The Construction of Sequences for Identification of Digital Circuits using Simulated Annealing

Zouaoui Ramzi
Department of Computer Technologies,
Higher Institute of Technological Studies ISET Nabeul, University Campus, 8000
Nabeul TUNISIA
Email : zouaoui_ram {at} hotmail.fr

***Abstract*— In this paper we propose the use of simulated annealing algorithm to solve the problems of the construction of initialization sequences for digital circuits. The proximity of this strategy with genetic algorithms allows us to use existing strengths to quickly build identification algorithms based on this approach.**

Keywords*—algorithm; cost function; digital circuits; Simulating Annealing (SA); Genetic Algorithm (GA); initialization sequences; verification sequences.*

I. INTRODUCTION

The building task of the input sequences of test with different properties represents the core of the development process of the digital devices. The control procedure according to various estimates takes up 80% of the conception.

To solve this problem, there are three basic approaches. The first approach is to develop structural methods for the construction of input sequences [1]. This approach is successfully applied to the circuits of small and medium sizes. However, with the growth of trees of the design it is essential to increase the memory of the computer. The second approach is based on converting symbolic expressions [2] but this approach has the same restrictions (limits). The third approach has emerged as an alternative to large circuits and is based on the modeling of circuits to find the solutions. Depending on the purpose of the task, we can use the modeling repairable or without repair. The possibility of solving the problem with large circuits is demonstrated since the task of modeling is simple and it offers for such circuits faster algorithms for resolution.

Often, in the third approach, the authors use genetic algorithms [3-4]. This is because they are well studied and on their bases are proposed several solutions for different tasks. The authors also used genetic algorithms for construction problems of input identification sequences [5-6]. However, beyond the attention of researchers remains another evolutionary strategy of optimization, which is the Simulation Annealing (SA).

The objective of this paper is to construct algorithms for generating input sequences for digital circuits based on the strategy of the simulating annealing with the use of developments in the field of genetic algorithms and comparing their yields.

II. THE STRATEGY OF SIMULATION ANNEALING

Formally, the strategy of simulation annealing refers to probabilistic polynomial algorithms for solving NP-complete problems.

We introduce the necessary definitions:

- The configuration K_i of an iteration i is an arbitrary potential solution. The configuration in the simulation annealing is similar to individuals in the genetic algorithm.
- At each configuration K_i is assigned a cost function $C_i=C(K_i)$. This function shows how the configuration is well chosen to solve the problem. The cost function for the simulation annealing corresponds to the fitness function for genetic algorithms. The calculation complexity of this function determines the complexity in the overall algorithm.
- The environment of the configurations K_i is considered to be the set of possible configurations that can introduce some disturbances (generally small sizes). The operations of perturbation in the algorithms of simulation annealing correspond to mutations for genetic algorithms.
- The temperature T in the process of the algorithm gradually decreases from a certain initial value T_0 to a cooling temperature T_f.

The description of the simulated annealing algorithm SA is given below:

1. At first, we form the initial configuration K_0 and evaluates its cost $C_0=C(K_0)$. The initial configuration is taken as equal to the current configuration $K_i= K_0$. The current initial temperature is determined $T_i=T_0$, then iterates the following steps until you reach the stop condition.
2. We form an environment of this current configuration K_i by performing a disturbance.
3. For all configurations belonging to the environment, the costs will be assessed and chooses the best configuration K_m.
4. The change of the cost function is calculated:

$$\Delta C_i = C(K_m) - C(K_i) \quad (1)$$

5. Changes made after the disturbance, or be accepted, or will be rejected if the change in the cost function is negative, so the intermediate configuration replaces the current. Otherwise, such a change will take place on the basis of the Boltzmann distribution:

$$K_{i+1} = \begin{cases} K_m, if\ \Delta C_i < 0 \\ K_m\ with\ probability\ P = \exp\left(\frac{-\Delta C_i}{kT_i}\right), if\ \Delta C_i > 0; \end{cases} \quad (2)$$

where k - a heuristic constant.

6. The current temperature is changed:

$$T_{i+1} = actualize(T_i) \quad (3)$$

7. Go to step 2.

III. Simulated Annealing Algorithms To Build Sequences Entry Of Identification

A formal declaration of construction problems of initialization sequences and equivalence checking is proposed in [9]. The informal use of algorithms can be formulated as follows:

- A logical initialization algorithm to construct an input sequence which translates the maximum triggers of an unknown state to a known state (0 or 1).
- Algorithm verification (control) to construct an input sequence that distinguishes the behavior of the two specified circuits.

To solve these problems by using a simulated annealing algorithm we must define its components.

As the configuration is adopted as an input sequence unitary and the input unit is adopted as a configuration sequence wherein the number of series is not known in advance and the vector length of a series is equal to the number of all the external input circuit [6].

The evaluation function determines the type of the input sequence during construction. For the initialization logic problem of circuits, it is written in the form:

$$C(S) = f(n_1, n_2, n_3) = (c_1 * n_1 + c_2 * n_2) * {c_3}^{n_3} \quad (4)$$

Where

- n_1 - the ratio of the number of triggers initiated by their total number, the higher the number of triggers passed of an undetermined state to determined state, better is the quality of the sequence.
- n_2 - activity of diagram or the number of events modeling, higher the number of events of a given sequence following a modeling, is great, the greater the probability of passing of the number of triggers in the determinate state is high.
- n_3 - the length of the sequence, two sequences placed under equal conditions we must choose the shortest.
- C_1, C_2 and C_3 are constants for normalization.

The evaluation function in the task of verification the equivalence of circuits is as follows:

$$C(S) = f(n_1, n_2, n_3) = n_1 + c_1 * n_2 + c_2 * n_3 \quad (5)$$

Where n_1 is the number of different indications at the external outputs of the two circuits analyzed, n_2 is the number of changes of pseudo-outputs of circuits, n3 is the number of valves of the two circuits having different signals, and c_1, c_2 are two normalization constants.

To construct the environment, we use three types of disturbances that correspond to mutation operations of genetic algorithms for the tests [6]:

- We randomly remove a sequence of the set of inputs.
- We add randomly a vector to any position in the sequence.
- We change randomly a column (row) in the configuration.

IV. Experimental Data

The proposed simulated annealing algorithm for the construction of input sequences is implemented in software (program). The tests are conducted with circuits of the international catalog ISCAS-89.

The heuristic parameters of the algorithms are determined experimentally. The algorithm of construction of the initialization sequence: $T_0 = 120$, $T_f = 1$, the number of configurations for the same temperature = 100, the Boltzmann distribution, Boltzmann constant $k = 0.01$. The verification algorithm of equivalence: $T_0 = 120$, $T_f = 1$, the number of configurations for the same temperature = 50, the Boltzmann distribution, Boltzmann constant $k=0.00001$.

The experimental results for some large circuits are shown in TABLE I. To test the effectiveness of the algorithm for equivalence checking, we have built almost equivalent circuits in accordance with the procedure in [8].

Three results are possible for such an algorithm:

- The algorithm builds the necessary sequence, that is to say capable of distinguishing between the behaviors of circuits, the result are presented in the column "Different circuits".
- The algorithm does not construct the required sequence and the evaluation function in the research process remains zero, ie, for all input sequences considered, there was no change behavior even inside of the circuit; the results are presented in the column "identical circuits".
- The algorithm does not construct the required sequence but differences were observed in the circuit and have not been distributed to the external output. In this case, it is possible to construct a sequence by deepening the search; the results are presented in the column "?".

Based on these results, the high efficiency of the proposed algorithm can be pended out: the number of circuits discriminated is 96,71% while with other verification algorithms, the results are: Genetic Algorithms for verification equivalence-94,7% [8-10], the algorithm VEGA-88,35% [11], the algorithm AQUILA-65,00% [12].

The digital data in the genetic algorithms corresponding to the two types of experiments are indicated [7-8]. Comparison of digital data for simulated annealing algorithms and genetic algorithms shows that they are very similar in the meaning of optimization and the two types of algorithms are virtually equivalent.

TABLE I. THE NUMERICAL RESULTS OF EXPERIMENTS WITH THE CIRCUITS ISCAS-89

Circuit (Scheme)	**Number of triggers**	**Initialization algorithm**			**Verification algorithm**			
		triggers initialized	***Length of sequence***	***Time (Second)***	***Total of experiences***	***Different circuits***	***identical circuits***	***?***
s298	145/14	14	2	0	25	25	0	0
s344	198/15	15	2	0	25	25	0	0
s349	199/15	15	2	0	25	25	0	0
s382	191/21	21	1	0	25	23	1	1
s386	182/6	6	3	0	25	25	0	0
s967	465/29	10	1	11	25	21	4	0
s1196	578/18	18	1	0	25	25	0	0
s1238	557/18	18	1	0	25	25	0	0
s1423	756/74	74	4	0	25	25	0	0
s1488	289/6	6	1	0	25	23	0	2
s1494	683/6	6	1	0	25	23	0	2
s3271	1731 / 116	116	11	0	25	25	0	0
s3330	2037 / 132	132	5	0	25	22	2	1
s3334	1940 / 183	183	8	0	25	25	0	0
s4863	2514 / 104	104	12	0	25	25	0	0
s5378	3045 / 179	179	110	40	25	24	0	1
s6669	3460 / 239	239	7	2	25	25	0	0
Total					425 **(100%)**	411 **(96,71%)**	7 **(1,645%)**	7 **(1,645%)**

V. CONCLUSION

In our work we have proposed to use simulated annealing to solve construction problems of sequences identification of input for digital circuits. On this basis, we have developed algorithms to build sequences for initialization and verification of equivalence. According to the experimental results, these algorithms have shown more effective than algorithms based on the genetic approach.

As prospects for our work, we can try to use simulated annealing for the elaboration of excessive tests and development of parallel versions of simulated annealing.

REFERENCES

[1] Niermann T., Patel J.H. HITEC: A Test Generation Package for Sequential Circuits // Proc. European Design Automation Conf.- 1991.- p.214-218.

[2] Sellers F.F., Hsiao M.Y., Bearnson L.W. Analysing errors with the boolean difference // IEEE Transactions on Computers.- 1967.- №5.- p.675-680.

[3] Goldberg D.E., Genetic Algorithm in Search, Optimization, and Machine Learning.- Addison-Wesley.- 1989.

[4] Skobtsov Y.A.,. Principes de base de calcul évolutif– Donetsk: DonNU, 2008.- p326.

[5] Y.A. Skobtsov, D.E. Ivanov, V.Y. Skobtsov Evolutionary distributed test generation methods for digital circuits // Proc. of 8th International Workshop on Boolean Problems, September 18-19, 2008, Freiberg, Germany.- pp.213-218.

[6] Skobtsov Y.A., El-Khatib, Ivanov D.E. Distributed Fault Simulation and Genetic Test Generation of Digital Circuits // Proceedings of IEEE East-West Design&Test Workshop(EWDT'06).-2006: Sochi.- p.89-94.

[7] Skobtsov Y.A., El-Khatib, Ivanov D.E. Construction des séquences d'initialisation pour les circuits numériques synchrones en utilisant des algorithmes génétiques -. Problèmes des technologies de l'information-2007.-№1.-p.158-164.

[8] Ivanov D.E, Zouaoui .R Approche génétique pour la vérification de l'équivalence des circuits séquentiels // «Département Radio-électronique Informatique ».- Zaparojia, ZNTU.- 2009.- №1(20).- p.118-123.

[9] Ivanov D.E, Zouaoui .R Les algorithmes génétiques pour la construction des séquences d'identification pour les circuits numériques avec mémoire // Articles scientifiques de l'Université National des Techniques de Donetsk. Série: "Technique Informatique et Automatique". Issue 14(129).-Donetsk: DonNU. – 2008.- p.97-106.

[10] Ivanov D.E. The genetic approach to the verification of the equivalence of Sequential Circuits // Radioelectronics. Informatics. Office – Zaparojia, ZNTU. 2009-№1(20) PP.118-123

[11] F. Corno, M. Sonza Reorda, G. Squillero VEGA: A Verification Tool Based on Genetic Algorithms // ICCD98, International Conference on Circuit Design, Austin, Texas (USA). – 1998. – P.321-326.

[12] Shi-Yu Huang , Kwang-Ting Cheng , Kuang-Chien Chen, Forrest Brewer, Chung-Yang Huang, AQUILA: An Equivalence Checking System for Large Sequential Designs. // IEEE Transactions on Computers. – 2000. – v.49, N.5. – P.443-464.