Journal of Global Optimization manuscript No.
(will be inserted by the editor)

Markov modelling and parameterisation of genetic evolutionary test generations

Adriel Cheng · **Cheng-Chew Lim**

Received: date / Accepted: date

Abstract Genetic evolutionary algorithm is an effective and optimal test generation method. However, the method to select the algorithm parameters is often ad hoc relying on empirical data. We use Markov-based method to model the genetic evolutionary test generation process, parameterise the process characteristics, and derive analytical solutions for selecting the optimisation parameters. The method eliminates preliminary test generation calibration and experimentation effort needed to select these parameters used in current practice.

Keywords Genetic algorithm · parameter selection · Markov model · hardware design verification

1 Introduction

Genetic evolutionary algorithm is a randomised, global search optimisation technique used widely in applications where the objective function is complex for gradient-based optimisation methods. The effectiveness of the genetic evolutionary algorithm (GEA) method is governed by the parameters chosen for conducting the evolutionary optimisation. Selecting parameters in real-world applications using GEA often involves conducting preliminary test runs using empirical parameter values and iteratively refining the parameters. This approach facilitates immediate results, but it does not necessary give the best parameters possible and is time consuming.

We consider the parameter selection problem for GEA, in particular in generating test cases applicable to verify hardware design for the semiconductor industry. Verifying VLSI (very large scale integration) chip design is essential in checking the new design. Test generation plays a crucial role in not only verifying the intended

---

Adriel Cheng · Cheng-Chew Lim
The University of Adelaide, Australia 5005
Tel.: +618-8303-5893
E-mail: {acheng, cclim@eleceng.adelaide.edu.au}

functions but also determining the cost of chip testing. It is highly desirable to generate tests that optimally meet the verification objective function. In chip verification the objective function is complex with multiple goals and subject to nonlinear constraints. The GEA-based test generation has been shown to be effective [1], [2], but there is still no detailed parameter selection study reported.

In addressing the parameter selection problem, we establish the relationship between the self-adaptation characteristics of the evolutionary cycle and the selection of GEA parameters. We then model and parameterise the test case generation process, and derive analytical solutions for selecting the required parameters.

Markov modelling approach is used in this work as a GEA flow contains many sub-processes with independent states. Markov models have been previously employed to model the entire GEA process for analysis and extracting basic information [3], [4], [5]. However, a GEA process for most application domain requires too large a Markov chain for successfully modelling the entire flow and can be too complex to analyse. Rather than restrict one Markov chain to represent the entire flow, we use multiple smaller Markov models instead. We decompose the GEA process into sub-processes based on a key observation we made on the self-adaptation characteristics of the evolutionary cycle.

The contribution of this study is a systematical and analytical solution to the parameter selection difficulty in generating test using GEA. The method should provide the semiconductor industry an effective aid to verify new VLSI hardware design whose complexity in circuitry and functionality will only increase.

2 Problem statement

We present the genetic evolutionary algorithm (GEA) process of which tests are generated, then formulate the parameter selection problem in generating tests to verify hardware circuit designs.

2.1 Genetic evolutionary optimisation

Suppose the optimisation problem is to find x so as to

$$\begin{array}{cl} \max\limits_{x \in \Omega} & f_c(x) \\ \text{subject to} & f_s(x) \leq M, \end{array}$$

where x is a test from the input test space of the test generation Ω, $f_c(x)$ is the test coverage fitness function of the VLSI chip when exercised by x, $f_s(x)$ is the function that evaluates the size of the test, and M is the maximum size of the available memory capacity to hold a test for execution.

The underlaying idea of GEA applied to solve the problem is as follows. Let $P_\mu(z)$ (respectively, $P_\lambda(z)$) be the parent population set (respectively, offspring population set), of the z^{th} generation with its individual population in Ω. The Q set denotes a set of individuals for possible selection. The execution flow of the GEA process is expressed in Figure 1.

```
Genetic evolutional algorithm
begin
$z = 0$
initialise $P_\mu(z)$
evaluate $P_\mu(z)$
while termination condition is false {
    begin
    $P_\lambda(z)$ = variation $P_\mu(z)$
    evaluate $P_\lambda(z)$
    $P_\mu(z+1)$ = select ($P_\lambda(z) \cup Q$)
    adjust weights of variation operators
    $z = z+1$ }
end while
end
```

Fig. 1 Genetic evolutionary algorithm with self adaptation for test generation

The initialise function creates an initial population of solution $P_\mu(0)$ via a random selection procedure. Each individual in $P_\mu(0)$ is evaluated to obtain feasible solutions. The objective function used in the fitness evaluation can be single-objective or multi-objective (more details in [2]). Based on this evaluation, a new set of individuals are created with variation operations on individuals in $P_\mu(0)$. The objective function is then evaluated for each new individual in $P_\lambda(0)$ to assess its fitness. The population to be selected for the next evolutionary cycle is chosen from the created population $P_\lambda(z)$ and the set Q, which is derived from using two evolutionary strategies [3] such that

$$Q = \begin{cases} P_\mu(z) & \text{if } (\mu+\lambda) \text{ evolutionary strategy is used;} \\ \emptyset & \text{if } (\mu,\lambda) \text{ evolutionary strategy is used.} \end{cases}$$

The operation of adjusting variation weights for self-adaptation then follows. The procedure repeats iteratively, generating populations $P_\lambda(1)$, $P_\mu(2)$; $P_\lambda(2)$, $P_\mu(3)$; $\ldots$, until an appropriate stopping criterion is met.

2.2 Parameter selection in test generation

Test generation provides the stimulus that drives a hardware circuit to verify the design and uncover bugs. The GEA-based test generation utilises a set of environment level parameters provided by the test engineer to operate. The selection of the parameter values determines the evolutionary characteristics and controls the quality of the tests generated. The GEA parameter selection problem then involves finding a technique for selecting the parameters that facilitates effective test generations. We pose the problem as to parameterise the GEA optimisation process with identifiable parameters that lead to effective test generation, and to derive an analytical-based technique for selecting these parameters.

3 Approach

Markov representation is used to model the evolutionary cycle. This is based on the observation that the new population of tests depends on the GEA variation applied to the test population from the previous evolutionary cycle only.

Modelling the entire GEA test generation with one Markov chain is not practical. It can lead to a high dimensional state model that makes the problem intractable. Instead, we consider using multiple smaller Markov models. When analysing the GEA process, it becomes clear that the operations within the GEA iteration, i.e., inside the ‘while loop’ in Figure 1, are separatable into sub-processes, with each sub-process satisfying the Markov properties.

Three sub-processes have emerged from the analysis of the evolutionary iteration of the GEA test generation process: (i) the self-adaptation in the iterative cycle, (ii) the self-adaptation in the entire evolutionary process, and (iii) the progression of test population. In the GEA test generation, the parameters regarded as important for the evolutionary process are: the number of evolutions, the parent population size and the offspring population size. These parameters and the three sub-processes are directly associated.

We will describe these sub-processes and present the Markov models and their use in selecting the optimisation parameters.

4 Modelling self-adaptation in the iterative cycle

The GEA self-adaptation adjusts the variation weight variables after every evolution cycle. As shown in Figure 1, the operation of adjusting variation weights for self-adaptation takes place inside the ‘while loop’.

The goal of GEA self-adaptation is to optimise the objective function of the test generation via appropriate usage of the variation operators. We use the ratio of successful variation to non-successful variation as a measure of usage, and adopt the Rechenberg’s rule, which states that a target success ratio should be $1/5$ [6].

Within a GEA test generation context, a variation is considered successful if the test coverage acquired from the newly varied test is greater than the coverage of the original test which was varied to create the new test.

The operation of variation adjustments is conducted such that if the achieved success ratio is outside an acceptable margin, then apply increment or decrement adjustments to the weights of the variation operators. The weight represents the probability of usage of the variation operator. We model the state transitions of this variation adjustment by a Markov chain.

4.1 The variation adjustments Morkov model

Let the set of states of the Markov chain be $\{A,D,S\}$, where A represents the state the usage of a variation is increased, D is the state the variation usage is decreased, and S represents the state when the variation’s target success ratio has been achieved.

Figure 2 is the Markov state diagram showing the states A, D and S and their transition relationships. The variable w represents the probability of usage that can increase, decrease, or be no change. The transition probabilities of the Markov chain are deduced as follows. The probability of transition within the same states A or D is w. The S state is absorptive[1], therefore the transition probability to itself is 1.

The probability of entering S from previous increase or decrease state is always lower than the probability to continue increasing or decreasing. It is therefore a fraction of w, taken as $e \cdot w$, where $0 < e < 1$. As transition probabilities exiting either A or D must sum up to one, the transition probabilities for $A \rightarrow D$ and $D \rightarrow A$ are both $(1-(1+e)w)$.

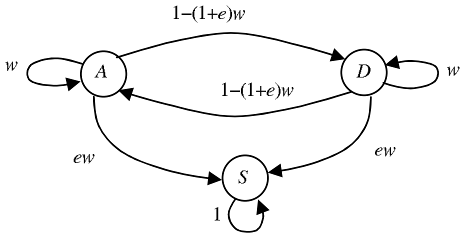

Fig. 2 Markov model for the variation adjustments in the GEA self-adaptation process

The transition matrix for A, D and S is

$$P = \begin{array}{c} \\ \\ \\ \end{array}\begin{array}{cccc} A & D & S & \\ \end{array}$$

$$P = \begin{array}{cccc} & A & D & S \\ & \left(\begin{array}{c} w \\ 1-(1+e)w \\ 0 \end{array}\right. & \begin{array}{c} 1-(1+e)w \\ w \\ 0 \end{array} & \left.\begin{array}{c} ew \\ ew \\ 1 \end{array}\right) & \begin{array}{c} A \\ D \\ S \end{array} \end{array} \tag{1}$$

Here the matrix element p_{ij} specifies the transition probability from the i state to the j state. The transition can remain in the same state i, and its probability is p_{ii}.

4.2 Determining the parameters

We now analyse the Markov chain to show how to extract from P information regarding the number of evolutions required for the GEA test generation.

Mapping P in (1) into the canonical form of an absorbing Markov chain transition matrix, as in [7], we obtain

$$P = \begin{array}{ccc} \text{TR} & \text{ABS} & \\ \left(\begin{array}{c} Q \\ 0 \end{array}\right. & \left.\begin{array}{c} R \\ I \end{array}\right) & \begin{array}{c} \text{TR} \\ \text{ABS} \end{array} \end{array} \tag{2}$$

with

$$Q = \begin{pmatrix} w & 1-(1+e)w \\ 1-(1+e)w & w \end{pmatrix} \quad \text{and} \quad R = \begin{pmatrix} ew \\ ew \end{pmatrix},$$

---

[1] A Markov state is said to be absorbing if the probability of leaving that state is zero.

where TR and ABS denote the transient and absorbing states, Q gives the probabilities of transition within transient states, R shows the probabilities of transition from a transient state to an absorptive state, and I is the identity matrix.

Let n denote the number of state transitions. The expression for P^n is

$$P^n = \begin{pmatrix} Q^n & N_n R \\ 0 & I \end{pmatrix}, \tag{3}$$

where Q^n indicates the probabilities of transitioning within the transient self-adapting increase or decrease states after n transitions and before entering the absorbing state, and $N_n = I + Q + Q^2 + \ldots + Q^{n-1}$.

In an absorptive Markov chain, if $n \to \infty$, then $Q^n \to 0$ and N_n tends to $N = (I-Q)^{-1}$. Thus, N is evaluated from

$$N = \frac{1}{ew(2-ew-2w)} \begin{array}{cc} \begin{array}{cc} A & D \end{array} & \\ \begin{pmatrix} 1-w & 1-(1+e)w \\ 1-(1+e)w & 1-w \end{pmatrix} & \begin{array}{c} A \\ D \end{array} \end{array} \tag{4}$$

The expected number of transitions that transverses A or D before being absorbed, called the time to absorption, is obtained by summing the row elements of N. This result gives the number of evolutions required for the GEA test generation process.

We use the verification of a VLSI chip design as an application example to illustrate the method.

To solve N from (4), the values for e and w are provided with the following consideration.

We observed that achieving the target success ratio of variations was usually difficult. In the best case, it requires at least ten evolutions of increasing or decreasing weight adjustment before reaching the target success ratio. This is a tenth of the variation application for the likelihood of realising the ratio. Thus, the fraction e is taken as $e = 0.1$ in this application example.

For w, we provide the lower and upper values. We obtain the lower value by observing that the probability of changing between increase or decrease states is not greater than the probability of maintaining the same self-adaptive variation. Thus, we express their relationship as $w \geq 1-(1+e)w$. If $e = 0.1$ is used, then $w \geq 0.48$. For the upper value for w, we assign $w = 0.9$; it represents the near maximum probability value of the variation weights that bring about changes between transient states in the GEA test generation process.

The numerical solutions of N are

$$N = \begin{pmatrix} 13.1 & 9.9 \\ 9.9 & 13.1 \end{pmatrix}\Bigg|_{w=0.48,\ e=0.1} \quad \text{and} \quad N = \begin{pmatrix} 10.1 & 1.0 \\ 1.0 & 10.1 \end{pmatrix}\Bigg|_{w=0.9,\ e=0.1}$$

The results indicate that before the Markov chain is finally absorbed into S, variation can (i) remain in an increasing A or decreasing D state for 10.1 to 13.1 evolutionary cycles, and (ii) transition between A or D states from 1 to 9.9 cycles. In addition, the times to absorption are

$$t = \begin{pmatrix} 23.0 \\ 23.0 \end{pmatrix}\Bigg|_{w=0.48,\ e=0.1} \quad \text{and} \quad t = \begin{pmatrix} 11.1 \\ 11.1 \end{pmatrix}\Bigg|_{w=0.9,\ e=0.1}$$

From a parameter selection viewpoint, we are interested in the number of transient states the variation undergoes. In order for the GEA process to achieve absorption, at least 23 evolutions must be conducted. If ideal conditions are assumed, then only 11 evolutions are needed. However, in real applications, one must allow at least 23 evolutions to take modelling uncertainties into account.

5 Modelling self-adaptation over the entire life cycle

We now model the characteristics of the variation and its usage over the evolution life cycle. The model is used for analysing the long-term usage behaviour of the variation operators.

A GEA process can end with one of the following states: the goal state G, which represents the level of variation at which the 1/5 target is successfully attained; the upper boundary state U, which represents an undesired condition when the upper variation bound is being applied; and the lower boundary state L, which represents another undesired condition when the lower variation bound is being applied. These three states are all absorptive.

We further define two transient states to represent the condition of variation before it becomes absorbed. We use E to represent the state in which variation weight holds the intermediary value above the minimum value represented by L and the eventual target value at which G occurs. Similarly, H is the state for values between G and U. The E and H states here and the A and D states of the Markov model in Figure 2 are closely linked and will be discussed shortly.

The transition relationships of G, L, U, E and H that model the self-adaptation characteristics in terms of variations over the entire life cycle are represented by the Markov chain in Figure 3. Their transition probabilities between states are now derived.

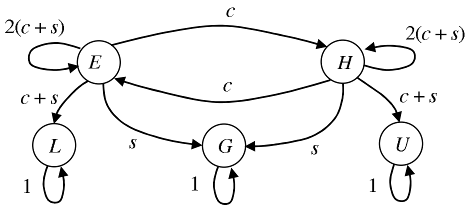

Fig. 3 Markov model for the self-adaptation of the process life cycle

Let c be the number of times a variation usage is continually increasing or decreasing, and s be the number of times a variation usage is switching between increment and decrement. In effect, c is proportional to the probability of the variation under continual change from one evolution to the next, and corresponds to the state transition of the Markov model in Figure 2. Similarly, s is proportional to the probability of the variation switching. It follows that c is the transition probability for E transversal to H, and also for the $H \rightarrow E$ transition. For the transition E to G to occur, it requires a change opposite to what was previously applied in order to avoid overshooting past the goal state G. Thus, the transition probability is s. The same applies for the transition $H \rightarrow G$.

The transition $E \rightarrow L$ is where the variation usage undergoes decrement. It corresponds to the transitions $D \rightarrow D$ and $A \rightarrow D$, where A and D are the states of the Markov chain in Figure 2. The transition probability is $(c+s)$. Similarly, the transition probability for $H \rightarrow U$ is $(c+s)$.

For the transitions of E to itself and H to itself, they are the usage of increment or decrement. Here the variation weights are adjusted but remain in the same intermediate state. Such changes correspond to any of transition $A \rightarrow A$, $A \rightarrow D$, $D \rightarrow A$, and $D \rightarrow D$. The transition probability for $E \rightarrow E$ is $2(c+s)$, and is the same for $H \rightarrow H$. As the states G, L and U are absorptive, the transition probability to itself is 1.

Transforming the transition matrix of the Markov chain of Figure 3 into the absorptive Markov chain canonical form, we acquire the fundamental matrix N and associated R as

$$N = \frac{16(c+s)^2}{4(c+s)^2-4c^2} \begin{array}{cc} \begin{array}{cc} E & H \end{array} & \\ \begin{pmatrix} \frac{1}{2} & \frac{c}{4(c+s)} \\ \frac{c}{4(c+s)} & \frac{1}{2} \end{pmatrix} & \begin{array}{c} E \\ H \end{array} \end{array} \tag{5}$$

$$R = \begin{pmatrix} \frac{1}{4} & \frac{s}{4(c+s)} & 0 \\ 0 & \frac{s}{4(c+s)} & \frac{1}{4} \end{pmatrix}. \tag{6}$$

The long-term probabilities of the GEA process being absorbed into G, L or U are obtained from the matrix product NR with $n \rightarrow \infty$. Taking $w = 0.48$ and $e = 0.1$, and obtaining c and s from (4) by noting that they are the transition probabilities of the A and D, we have

$$NR = \begin{array}{cc} \begin{array}{ccc} L & G & U \end{array} & \\ \begin{pmatrix} 0.54 & 0.31 & 0.15 \\ 0.15 & 0.31 & 0.54 \end{pmatrix} & \begin{array}{c} E \\ H \end{array} \end{array} \tag{7}$$

This matrix product is interpreted as follows. If the test generation began in the intermediate transient state E, there is a 0.54 probability of absorption into the boundary state L, or a 0.15 probability of absorption into U when the GEA process terminates. These outcomes are vice versa if the initial state is H. Regardless of which state variation starts from, there is a 0.31 probability of attaining the target success ratio. Avoiding absorption into U or L is important because once a variation attains the maximum or minimum boundary values, applying further variation will continue to be either excessive (U) or insignificant (L). This results in a runaway condition for the GEA process and recovering from such a condition is difficult. In short, attaining G is a desired goal, and avoiding U and L is just as important.

Besides revealing possible long-term value ranges of variation usage adjustments and weights, the Markov chain in Figure 3 is used to build the Markov chain that models the progression of generating test population.

6 Model the progression of test population

Consider the GEA execution flow shown in Figure 1. At every cycle, the GEA process creates new tests and selects existing parent and new offspring tests for the next evolution population. Modelling these operations enables us to examine the ratio of existing and newly created tests that are selected to make up the next evolution population. We can then deduce from the analysis appropriate test population size parameters.

We define the following states to represent the ratio of offspring to parent tests making up the test population during each evolutionary cycle.
V: Half the population is entirely new offspring tests.
X: Half the population consists of more offspring tests than parent tests.
Y: Half the population is less than or equal to the number of offspring to parent tests.
J: Half the population is mostly parent tests, with only a few offspring tests.
The states V, X and Y are transient states, and J is absorbing state. Shown in Figure 4 is the Markov chain. The transition probabilities are made up of μ, the number of parent tests, and λ, the number of offspring tests.

For the transitions $V \rightarrow V$ and $X \rightarrow V$, the test population selects only offspring tests, hence the transition probability is proportional to a population ratio of $2\lambda/\mu$. For $V \rightarrow X$, $X \rightarrow X$ and $Y \rightarrow X$, as greater offspring tests are selected, the ratio is also a fraction of $2\lambda/\mu$. There are more parent tests being selected in $X \rightarrow Y$ and $Y \rightarrow Y$, therefore, λ/μ is appropriate. When $Y \rightarrow J$ occurs, much more parent tests are now selected, thus, λ/μ is used again. Additional factors a_1 to a_8 based on the Markov model in Section 5 and other practical chip verification considerations are also incorporated into the transition probabilities. For the purpose of this section, where only population sizes are of interest, we express the transition probabilities and focus our analysis using μ and λ only.

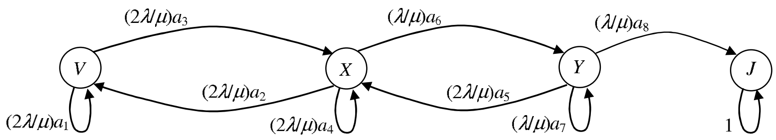

Fig. 4 Markov model for the progression of test population

A goal of GEA test generation is for greater offspring tests to be selected, which requires the offspring population size (λ) to be as large as possible. To establish an upper limit on λ, we analyse the Markov model of Figure 4. By evaluating the fundamental matrix N, we maximise the expected number of times the test generation achieves state V for the $V \rightarrow V$ transition; in order to determine the ratio of offspring and parent test sizes for GEA test generation such that greater offspring tests are produced and selected for as many evolutions as possible. Maximising the $V \rightarrow V$ transition matrix element of N, the analysis is as follows.

$$\max\left(\frac{0.56\mu-\lambda}{0.56\mu}\right) = \max\left(1-\frac{\lambda}{0.56\mu}\right) \Rightarrow \min\left(\frac{\lambda}{0.56\mu}\right) \Rightarrow \lambda < 0.56\mu. \quad (8)$$

Equation (8) suggests the desired ratio of offspring to parent population size can be chosen to be approximately 1:2. To determine actual population sizes, we first consider what the parent test population should be. The parent test population size is the test population used at the beginning of each evolutionary cycle. The parent and offspring sizes must be sufficiently large to provide a good diverse mixture of tests, but must not be too large to become inefficient to implement. In the chip verification application, our test building block library holds approximately 30 test building blocks, hence the parent population size (μ) is set to 30. From (8), the offspring population size (λ) is 15.

7 Closing remarks

The analysis of the self-adaptation characteristics process and model enables the number of evolutions to be chosen for generating test cases. It was shown to be at least 23 evolutions. Modelling and analysing the progression of test generation shows that the offspring to parent population size should be at an approximately 1:2 ratio. We have demonstrated that the devised Markov modelling and analytical parameter selection strategy can be applied to GEA procedures. In addition, the methodology is flexible, allowing practical considerations to be incorporated separately to handle practical constraints imposed by the application and the implementation of the GEA process. The outcome of the method is more accurately derived parameters that provides effective GEA processes without the need for ad hoc preliminary test runs or empirical results.

Acknowledgements This research was supported by the Australian Research Council under the grant ARC-LP0454838.

References

1. Corno, F., Cumani, G., Reorda, M. S. and Squillero, G.: Fully automatic test program generation for microprocessor cores, Design, Automation and Test in Europe (DATE2003), Munich, 1006-1011 (2003).
2. Cheng, A. and Lim, C. C.: Multi-objective genetic algorithms for system-on-chips verification, 1st World Congress on Global Optimization in Engineering and Science (WCGO2009), June, (2009).
3. Fogel, D. B.: Evolutionary computation: Toward a new philosophy of machine intelligence, 2nd ed, IEEE Press, New York (2000).
4. Nix, A. E. and Vose, M. D.: Modeling genetic algorithms with Markov chains, Anals of Mathematics and Artificial Intelligence, 5, pp. 79-88 (1992).
5. Mao, C. Y. and Hu, Y. H.: Convergence analyses of simulated evolution algorithms, Design Automation of High Performance VLSI Systems (GLSV'94), Madison, 30-33 (1994).
6. Rechenberg, I.: Evolution strategy: Optimization of systems according to principles of the biological evolution, Stuttgart: Frommann-Holzboog, Verlag (1973).
7. Grinstead, C. M. and Snell, J. L.: Introduction to probability, 2nd ed, American Mathematical Society (2006).