Paper 23:

Testing software response times using a genetic algorithm

Jarmo T. Alander, **Timo Mantere** and **Ghodrat Moghadampour**
University of Vaasa
Department of Information Technology and Industrial Economics,
P. O. Box 700, FIN-65101 Vaasa, Finland
phone +358-6-324 8444, fax +358-6-324 8467
E-mail: `firstname.lastname@uwasa.fi`

Abstract. In this work we are studying possibilities to automate the testing of response times in embedded software using genetic algorithm search. The idea is to automatically produce test cases in order to find problematic situations and processing time extremes in an electric network protection relay's software. The proposed test method comes under the heading of automated dynamic stress testing.

Keywords: genetic algorithms, genetic algorithms, power distribution automation, relays, simulation, software engineering, software testing

23.1 Introduction

Failure in software, for power distribution system may have severe consequences; the requirements for correctness and reliability are high. In very reliable sequential programs, the rate of errors should be less than 10 errors/1000 lines of code, to avoid functional failure. Achieving this level is very labourious, because the amount of program testing work grows exponentially with code size.

Testing software manually is slow, and expensive, and demands inventiveness. Automated testing can reduce both the time and costs needed for making tests [132]. Automatic test data generation cannot solve the problem that every test alternative has to be dealt with. The most common way of generating test data is random, which is considered weak [325]. For this reason, efforts have been made to optimise data using various methods, for example heuristic ones.

In this method genetic algorithm optimises data inputs for tested program. The response time of the tested software is used as a fitness function. We are trying to find the situation in which software response is slowest by giving it more and more difficult situations generated by GA.

23.1.1 Related work

The automatic generation of test data using genetic algoritms has been studied by Xanthakis et al [521], Watkins [507] and Jones et al [250]. Studies have been mostly based on white-box testing methodology. Here we are using a black-box technique.

Recently there has been a growing interest to use GA based methods to test VLSI circuits [34, 106, 200, 221, 292, 363, 387, 424, 425, 435, 472]. See [11] for further references.

23.1.2 Network protection relay

Network protection relay is an apparatus that coverns electric network from overloading, overvoltage, short circuit and ground contact.

The relay itself is a microprocessor based embedded system and it normally contains several processing and interface units, which communicates with each other and with other relays and control units of the distribution network.

Because of the nature of the task, protection relay should work pretty fast, it should react within 30 ms (legal norm) from the discovery of a serious failure.

Time demand for the embedded relay software is even shorter, it should work in 20 ms. However, in the simulated environment response times of relay software are naturally somewhat longer than in the target environment.

23.1.3 ESIM simulation tool

In this work we have used ESIM simulator, which is designed to a test automation environment for embedded software development. ESIM is a product of Finnish software company Prosoft Oy. ESIM uses a workstation for testing embedded software written in the C programming language. Software is compiled with C compiler and linked to the ESIM environment library. The user describes the input and output system (the application-specific hardware). ESIM then simulates the I/O system and the operating system of the application, allowing the user to monitor what is happening in each of them. [395]

The GA and the tested program run separately in their own ESIM-tasks, which communicate with each other through simulated ESIM hardware ports.

Both programs, genetic algorithm and tested program runs under their own ESIM applications, either in the same Windows NT workstation or both in their own NT, which are connected together.

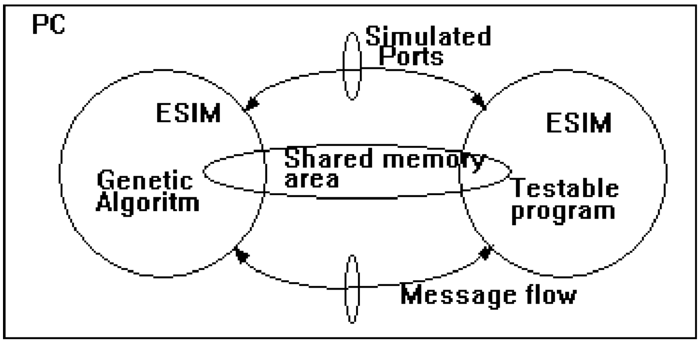

Figure 23.1: Program testing in a simulated environment

GA writes inputs to the tested program and measures the response time. Response time is the time it takes for the tested program to perform the operations caused by the input parameters it received. When tested program is completed, it sends a response signal. The response time is the fitness value for the GA.

23.2 Testing system

Genetic algoritm is connected to the relay program via ESIM in three different connection points, it it directly connected to the relay programs memory database (shared memory area), so that it can read and write information directly there, secondly it is connected to the control area network connection point, where it can send and receive control messages and thirdly it is connected into local area network connection point, where it can send and receive network messages.

GA simulates

- digital input/output (and manual commands card) cards of protection relay, GA sends the digital input information as the input card would throught the control area network into the main relay program.
- the analog inputs (current/voltage information from the electric network) by writing information directly to memory database of the relay program.
- other relays in the local area network by sending network messages to the relay program.

23.3 Test result analysis

First test case was to compare GA based test method against random testing method. The most important communication task in the relay is the communication between digital input/output cards and relays main processing unit. In

	GA-total	*Random-total*
`Mean` [ms]	223.45	190.86
`Std. deviation` [ms]	116.66	87.85
`Count`	4750	4751
`Minimum` [ms]	90.00	80.00
`Maximum` [ms]	761.00	811.00
`Range` [ms]	671.00	731.00
`Skewness`	1.20	1.28
`Kurtosis`	1.39	2.61
`Median` [ms]	200.00	161.00

Table 23.1: Descriptive statistics of the total response time of input data.

this test that communication is interrupted or bothered by lower priority network messages.

In order to examine the performance of the GA based input data generator we ran the GA and random based input data simulators 20 times. In order to combining the input data cases generated by the both methods we use the descriptive statistics.

In table 23.1 values are from simulated environment, does not equal with target environment times.

From the table 23.1 it looks obvious that the average and the median processing time of test case generated by GA is longer than with the test case generated by random method. Standard deviation in GA method is also greater, but the processing time distribution range is narrower.

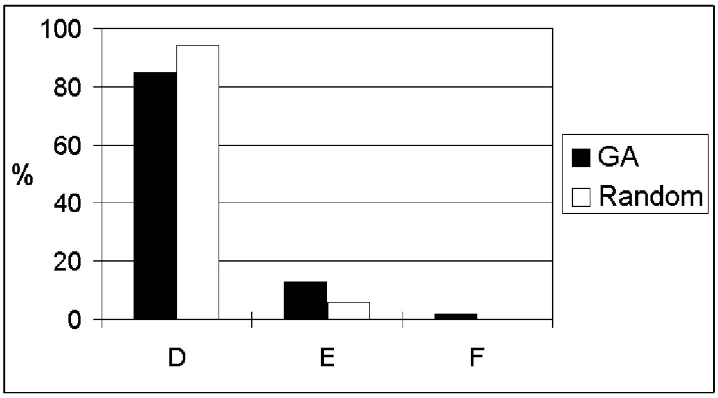

Figure 23.2: Distribution of total response time for test case

In figure 23.2 D is the range of [80, 330] ms, E is [330, 580] ms and F is [580, 830] ms.

From figure 23.2 it seems that GA generates more input data cases with long response times. Random method generates 11% more cases to the shortest time range, in turn GA method generates 153% more input data cases into the two longer ranges.

By calculating the statistical significance of difference between average processing times, we can be 95% confident that the average response time of test case generated by GA is at least 15% and at most 19% bigger than average response time of test case generated by the random method.

The second test case was to determine in which phase during the relay's internal message handling (messages between main processing unit and digital I/O cards) the network message arriving interrupts it the most. It seemed that there were statistical signifigance found in this test also.

23.4 Conclusions and future

It seems that genetic algoritm can be used for software testing, at least it can test suitable things in the software. The problem might be finding a characteristic fitness function and factors that influence to the test results. Problem complexity and non-determinism might also cause problems, which increases the difficulty of software testing. [32, 507]

There seems to be significant difference between test data cases produced by GA and random method, in advantage to the GA method.

The occurrance of network message interruption also has a significant statistical influence to the mean handling time of relay's internal messages. There is also difference in which point of the handling the interruption messages are sent.

Future research contains more precise analysis of message interruptions, what is the influencence of frequence of sending I/O card messages, what influence has the time between sent I/O card message and interrupt message.

The new version of tested software is under implementation to the test environment. New version includes more modules from the actual target environment embedded software, some modules were left off from the first implementation in order to make adjustment to the simulated environment easier. Tests with the updated software version cointains first the comparison to the old test and then new tests, which the implementation of more modules from the embedded software allows.

Acknowledgements

This work is supported by Finnish Technology Development Center (Tekes) and ABB Corporate research.

23.5 Bibliography

See chapter bibliography at the end of this proceedings.