Annals of DAAAM for 2011 & Proceedings of the 22nd International DAAAM Symposium, Volume 22, No. 1, ISSN 1726-9679
ISBN 978-3-901509-83-4, Editor B. Katalinic, Published by DAAAM International, Vienna, Austria, EU, 2011
Make Harmony between Technology and Nature, and Your Mind will Fly Free as a Bird
Annals & Proceedings of DAAAM International 2011

COMBINATORIAL LIMITATIONS OF COVERAGE TESTING FOR CONTROL SYSTEMS

RASKIN, M[ikhail] A[leksandrovich]

Abstract: *Present article considers purely combinatorial limits on effectiveness of black box testing. Even simple controlling structure (formally represented as a small finite automaton) can create rare combinations of events when acting on inputs without such rare combinations, which suggests that pure testing-based approach cannot prove reliability.*

Key words: *finite transducers, black box testing, theoretical bounds, mathematical methods*

1. INTRODUCTION

Development of more and more complex automated control system makes verification of correctness increasingly important. One of the ways to check whether control system works properly is testing it using realistic input data. Unfortunately, while it allows to verify behaviour under normal conditions, such testing may miss some problems caused by rare combinations of events.

One of the approaches to testing is “black-box” testing, which implies using program specification to create a set of test inputs and supposed outputs. There are some promising developments in creating black-box test sets with less human effort, like (Bareisa et al.) presenting a model of faults to improve test generation and (Last et al.) proposing a method for generating black-box test cases using genetic algorithms.

In the present paper limitations of black-box testing for control systems are analized for the case of a stateful control system that runs in a loop processing its inputs. The combinatorial result applies most to the case of testing control system with realistic sequence of inputs. It is shown that even simple control system can have rare sequences of reactions even when all possible sequences of input conditions are frequently observed.

2. PROBLEM IN CONSIDERATION

For long-running systems rare combinations of input are the hardest cases to test with realistic input data. If the control system has some inner state (and it usually does), it may be that every possible input is cheked, but some possible sequences of events are left without attention. These combinations may put the control system in the state where a rare event will be mishandled.

If we have more than one logical component in the system, it may be of in terest whether we have very rare combinations in the output of a controller given that there are no too rare combinations in inputs.

Frequency of rare combinations of inputs has been studied combinatorially before. For example, in (Muchnik et al. 2003) and in (Pritykin, 2006) it was proven that finite automata preserve almost periodicity, i.e. property of infinite seuqence not to have finite subsequences that occur progressively less often as we look through the sequence.

In (Pritykin, 2010), an upper bound was proven, i.e. it was shown that a finite automaton cannot create arbitrarily rare short combinations of outputs for a given lower bound on frequency of input combinations. Unfortunately, the upper bound is very high and impossible to improve substantially; in the current paper it is illustrated with a corresponding lower bound.

These considerations show that for checking handling of exceptional situations some white-box methods, like deducing the inner states of the system and checking the handling of all possible inputs in all possible states may be needed for testing system on realistic inputs.

3. DEFINITIONS

Definition. Infinite sequence A is almost periodic if and only if there is a function f such that for every natural n every word w of length n either doesn't occur in A or occurs in every fragment of A with length more than f(n). Minimal such function f is regulator of almost periodicity of the sequence A.

We will use standard definition of finite automaton. Finite automaton is defined by its set of states, its input and output alphabets and by a transition table that gives new state and output symbol for every combination of previous state and received input symbol. We will also assume that one of the states of automaton is marked as initial.

For every finite automaton A and for every infinite sequence of symbols from its input alphabet S we will define an infinite sequence I called image of S under action of A. To obtain I, we put A in its initial state and enter an infinite loop; on each step we consider current state of A and symbol in S with the number equal to the number of current step. We look up this pair in the transition table, replace the state of A with the resulting new state and put the output symbol on the corresponding position into I.

4. MAIN RESULT

Theorem. There exists a binary sequence S with regulator of almost periodicity f such that for every natural number n there exists a finite automaton A(n) with n states, input alphabet {0,1} and output alphabet {0, .., n-1} the regulator of almost periodicity of image of S under A has regulator of almost periodicity which is greater than f iterated n/10 times for infinitely many word lengths.

5. EXAMPLE CONSTRUCTION

To build sequence S, we will first define finite words X(k) and Y(k). First, **X(0) = 0; Y(0) = 1**. Next, if X(k) and Y(k) are already defined,

X(k+1) = Y(k) Y(k) X(k) X(k) X(k);
Y(k+1) = Y(k) Y(k) X(k) X(k) Y(k).

Note that for every m>n X(m) has X(k) as its beginning. That allows us to define S as limit of X(k) when n tends to infinity, i.e. on the k-th position in S we find the symbol from k-th position in some X(k) and it doesn't matter which n we choose as long as X(k) has length more than k.

Now let us define A(n). A(n) has states numbered from 0 to n-1 and its initial state is 0. If current state is n and input symbol is x, output symbol is pair (n, x) and new state is (n-1) when n+x is odd and (n+1) when n+x is even. If the new state should be -1 or n, values 0 and n-1 are used, correspondingly. Note that feeding the same symbol to A(n) twice leaves A(n) in the same state. It follows that feeding X(k) or Y(k) to A(n) leaves A(n) in the same state as feeding 0 or 1, correspondingly, would. Indeed, X(0) and Y(0) are equal to 0 and 1; and if X(k-1) and Y(k-1) act as 0 and one, Y(k-1) Y(k-1) and X(k-1) X(k-1) act as 11 and 00, i.e. leave A(n) in the same state, so X(k) and Y(k) act just as X(k-1) and Y(k-1).

6. PROOF

Proof. First, we estimate almost periodicity regulator of S. Length of X(k) and Y(k) is equal to 5^k. Let us prove that X(k) and Y(k) occur only on positions with number divisible by 5^k (if the sequence begins with the zeroth symbol). For X(0) and Y(0) it is obvious; if it is true of X(k-1) and Y(k-1), X(k) and Y(k) may only occur at positions with numbers divisible by 5^k-1; but they both begin with Y(k-1) Y(k-1) X(k-1) and this only happens at positions divisible by 5^k.

It is easy to see that in any place of S there can be at most three instances of X(k) or Y(k) in row, which allows us to add 5^k-1 symbols from each side to get 5^(k+1)-2 symbols in row without instances of either X^k or Y^k. But out of 5^(k+1)-1 symbols in row there will be an instance of both X^k and Y^k. It is easy to see that every word of length 5^k in S can overlap with no more than two instances of X(k) and Y(k). All such combinations occur in Y(k+1). So we can conclude that 5^(k+1)-2<f(5^k)<5^(k+2)-1.

To analyze image of S under action of A(n) we will consider words U(k,l) and V(k,l), which are images of X(k) and Y(k), correspondingly, under action of A(n) with initial state l. Obviously, U(k, l) and V(k,l) can only occur in the image of S at the positions with numbers divisible by 5^k, because the second components of symbols in U(k, l) and V(k,l) form X(k) and Y(k). Let us see how many distinct U(k,l) and V(k,l) can occur in U(m,p) and V(m,p). U(k,p) contains only U(k, p). U(m+1,p) and V(m+1,p) cannot contain U(m,q) or V(m,q) with q>p+1 or q<p-1. So we see that U(m,p) cannot contain U(k,l) with m-k<|p-l|. On the other hand, it is also easy to see that U(m,0) contains either U(k,l) or V(k,l) if k+l=m and l<n. That allows us to conclude that there are words of length 5^k which occur in I, but do not occur inside some segments of I of length 5^(k+n-1). This proves lower bound on regulator for I which exceeds n/4 iterations of regulator for S.

7. GENERALISATIONS

In this proof, we used one example of S and one specific construction of A(n). This was done for relative simplicity of construction and proof. This can be done for some other A(n), including simple cyclical counter. It is also easy to make regulator of almost periodicity of S grow faster than in our example. The proof in the present paper is enough, though, to show impossibility of simple a priori limitations of frequency of unique combination of events.

8. LIMITATIONS

Of course, a proof like this is applicable only to idealized black box testing. Although it can show the source of complications in the testing procedures that check reactions to predefined input conditions, present result cannot help to estimate how much consideration should be given to the inner structure of the control system under test.

One of the limitations of this proof can probably be useful as a motivating principle in the control system design. The problem of rarely occurring situations rises because long chains of some events lead to inner state slowly drifting to some rare state and back. In practice, many complex control systems become simpler to design if they operate in a event loop with little memory and small iterations, compared to control system trying to execute some long procedure and fixing deviations on each step.

9. CONCLUSION

Even with quite simple finite automaton used as controller, input with quasiperiodical oscillation can lead to reaction drift. In our opinion this can show impossibility of determining of testing effectivity without analysis of extreme cases not only in input data, but also control logic itself.

10. FUTURE DIRECTIONS

This topic can be further researched from both theoretical computer science and software testing points of view.

In theoretical computer science there is a field of research devoted to preservation of various properties of input sequences under the action of relatively simple transformation procedures. For example, in the present paper we reference an upper bound for the regulator and prove a lower bound. The gap between the lower and the upper bound can still be reduced. On the other hand, there are various definitions of “a relatively simple sequence”. How they are related and how they are preserved under simple transformations are also interesting topics of research in theoretical computer science.

From the point of view of automated software testing, it will be interesting to find any way of estimating coverage of extreme conditions by a fully automated testing package having limited access to the structure of control system in consideration. For example, if we know what simple combinations of input conditions occur in the real input and we know what are the states of the control program, it would be of interest to generate automatically a short testing methodology that would ensure that automated testing will check control system’s reaction to every possible input in every possible state.

Of course, unlike theoretical finite automata with relatively small set of states, many control systems have a relatively small amount of inner memory. This translates to huge set of states. For the present result this problem simply illustrates our point that automated testing cannot be done reliably without knowledge of the inner structure of the control system. Positive results, though, will have to deal with this problem. Probably, estimates for small finite automata will have to be obtained first.

11. REFERENCES

Bareisa, E.; Jusas, V.; Motiejunas, K. & Seinauskas, R. Black-box fault models. *Available from:* http://itc.ktu.lt/itc353/motieju353.pdf *Accessed: 2011-08-02*

Last, M.; Eyal, S. & Kandel, A. Effective black-box testing with genetic algorithms. *Available from:* http://www.haifa.ibm.com/Workshops/verification2005/papers/testing/ibm05_ga2.pdf *Accessed: 2011-08-01*

Muchnik, An.; Semenov, A. & Ushakov, M. (2003). Almost periodic sequences. *Theoretical Computer Science,* 304.

Pritykin, Yu. (2006). Finite-automaton transformations of strictly almost periodical sequences. *Mathematical Notes,* 80, 5-6

Pritykin, Yu. (2010). Almost periodicity, finite automata mappings, and related effectiveness issues. *Russian Mathematics (Iz VUZ),* 54, 1