

Title	モデル検査による大規模な計算機環境での検証手法
Author(s)	戸川, 博貴
Citation	
Issue Date	2010-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/8960
Rights	
Description	Supervisor: 青木利晃准教授, 情報科学研究科, 修士

Verified methodology in large-scale computer that uses model checking

Hiroataka Togawa 0810042

School of Information Science,

Japan Advanced Institute of Science and Technology

February 09 2010

Keywords model checking, formal methods,
cluster analysis, Promela environment model

Back ground

In late years the software is used in social every corner. Therefore it is necessary to guarantee reliability of the software. There is model checking as a method to guarantee reliability of the software. I studied the qualification methodology that applied model checking for OSEK/VDX till now. And we were able to generate much verification models. However, it is difficult to inspect it with one computer when there is much number of verification models. Because capacity of the memory of the computer is short, I cannot verification it. Moreover, I cannot grasp problems because the number of test results becomes enormous. This research inspects it with SPIN in large-scale computer environment. And I consider I analyze it and how display the result. A characteristic of this research is really to inspect it in large-scale computer environment. The SPIN has a characteristic to display the process of the wrong result(Counterexample). Therefore, the inspection of plural inspection models is enabled if I use it in large-scale computer environment. In addition, I can compare a test result.

Problem point

When I perform model checking with the computer environment of the large scale, there are problems. 1.Inspection time is different if I do not distribute an inspection model ideally. 2. Because I perform model checking with much inspection models, a test result becomes enormous and cannot grasp a problem. 3. I don't know whether the counterexample includes several kinds of errors. I test it to solve these problems and arrange a problem. And I think about applicable inspection technique with SPIN in the large-scale computer environment.

Approach

1-1. I distribute an inspection model ideally. The method to distribute an inspection model ideally demands time per one inspection model. And I distribute it to equalize the inspection time for each computer with inspection time. I suggest two experiments. I extract maximum 200 from the inspection model of each environmental model to demand inspection time. And I perform model checking. I demand inspection time per one inspection model. Next I demand total time of using all inspection models. I divide the total time when I demanded it by a computer to use. I do the value that I divided with inspection time per one computer. I distribute it till I achieve an inspection model at inspection time per one computer.

1-2. It is a method to demand the inspection time of the inspection model from relations of the number of lines of the inspection model script and the compilation time.

I perform model checking first. I examine relations of lineage and the compilation time next. And I find an approximate expression.

I divide it into the class that set the lineage of the inspection model script. I find a class value from the class that I set. I substitute the class that I found for an approximate expression.

I do the value that I found with inspection time of the class.

With the inspection time when I demanded it, I demand total time when I employed all the inspection models. I do the value that I divided with inspection time per one computer. I distribute it till I achieve an inspection model at inspection time per one computer.

2. The indication method of the test result

I suggest the method that can understand a test result by intuition. First, Abstraction degree makes an expensive list. For example, it is the list which is limited to the number of tasks. When I want to examine a detailed test result, I appoint the number of tasks and display it. Because I appoint the number of tasks then, I can make a list depending on the number of resources. When I want to examine a test result in detail, I pay attention to the priority of a task and resource, relations of a task and a resource, multiplicity and display it.

3. I distinguish the variation of the error

The method to distinguish the variation of the error applies cluster analysis. The cluster analysis is the method that can distinguish an individual by using the data which an individual has. I perform preprocessing to apply cluster analysis. I extract necessary information from a counterexample result with a grep command first. I find for

difference between the counterexample with a diff command next. And I convert a number for the difference between the counterexample for distance. I apply the distance that I found to cluster analysis.

An experiment result and consideration

1. I was able to inspect the first experiment result with the speed of about 18 times than inspection time was the longest and inspected it with one computer for about six hours. However, time lag for the and a half produced it for about three hours when number of tasks 2 compared the result of the computer which the number of resources allotted 4 inspection models to environment. The cause extracts the big environmental inspection model of the scale at random, and it is thought that an error at inspection time produced it because I demanded time per a thing of one inspection model. I perform model checking about all inspection models from this result, and it is necessary to examine a tendency of the inspection time. I was able to inspect the second experiment result in the range of an odd number with 66 computers for from two hours to two hours. However, it took six hours and nine hours for inspection with the computer of two remainder. It was predicted that lineage finishes the cause in the approximate expression between 700-800 lines for approximately 200 seconds, but the reason is because there was a model, actually, to need for 800-1000 seconds.

2. I can expect that I examine while deepening understanding by it pass, and examining a stage in detail. In addition, I understood that I considered where there was a problem by the border of the test result. You must think about the method that the problem extracts a test result from many test results in the item which you want to know and compares.

3. I extracted the contents of the error for preprocessing and was able to divide it into every contents of the result error that I performed a weighting every variation of the error, and applied cluster analysis. In addition, I extracted a true line of the counterexample to do detailed distinction by the same error and demanded distance. As a result, I understood what I could distinguish because for difference occurred for the true line. A problem is to think about a method to perform the grouping by the error same as a method grouping every kind of the error in once.