Test Data Generation for Testing Parallel Real-Time Systems - A Summary

Lecture Notes in Computer Science, Springer, 2015

Nur Qamarina binti Mohd Noor

Subject: Embedded System Design and Analysis (MANN1023)
Lecturer: Dr. Azurati Ahmad @ Salleh

3rd March 2015
Outline

• Background of Study
• Objective of Study
• Methodology
• Result and Discussion
• Conclusion
Background of Study

• Growing use of multicore processors → deployment of real time applications on multicore platform (parallel) → non-sequential code running.

• Hence,
  – The design and testing of multicore embedded real-time systems specifically its Worst-Case Execution Time (WCET) testing must be developed.

• Typically WCET is derived by either using
  • static timing analysis or
  • measurement-based method - which suits parallel systems
Objective of Study

Developing a Search-Based Software Engineering technique specifically using Genetic Algorithm to generate test-data for testing parallel real-time systems by measuring end-to-end execution time of the ParMiBench benchmark suite using Gem5 architecture simulation.
Methodology

ParMiBench

Stringsearch

**WHAT?**
Searching a token in a pattern file from text file

**HOW?**
Regenerate
a) new text file of 102400 characters
b) pattern file with non similar tokens
   - 64 tokens in each pattern file
   - 5 characters in each token
Each pattern file simulated to generate trace

100 pattern files

from generated traces

select top five

create new ten

enhance ten

50G?

YES

END

NO
For 5 characters length token, the threshold WCET is $5.98 \times 10^{10}$
It can be seen that for 5 characters length token, the number of pattern files having WCET above 5.98e+10 increased from one generation to another except for 30th generation and then the WCET increased again up to 50th generation.
Fig. 3. End-to-End times (fitness values) in different generations for the pattern files with ten character long tokens

For 10 characters length token, the threshold WCET is now 5.90e+10
It can be seen that for 10 characters length token, the WCET decreases compared to the WCET for 5 characters length. The number of pattern files with WCET above 5.90e+10 increase from one generation to another except for 30\textsuperscript{th} generation and then the WCET increased again up to 50\textsuperscript{th} generation.
Conclusion

• A measurement-based technique specifically Genetic Algorithm (GA) for automatic test-data generation for parallel real-time systems running on a multicore architecture is proven to be able to produce large execution times closed to WCET.