

UNIVERSITÀ DELLA SVIZZERA ITALIANA

MASTER THESIS

ACCELERATOR FOR EVENT-BASED FAILURE PREDICTION

Problem Statement

Author:

Simon MAURER

Supervisor:

Prof. Mirosław MALEK

January 31, 2014

Contents

1	Introduction	2
2	Problem and Research Area	2
3	State of the Art Outline	3
3.1	Failure Prediction	3
3.2	Accelerator	4
3.3	Log Standardization	4
4	Questions and Hypothesis	4
5	Theory Base	5
5.1	Taxonomy of Dependable Computing	5
5.2	Algorithms	5
6	Method, Design and Experiments	5
7	Preliminary Work Progress	6
8	Time Schedule	6

1 Introduction

This document provides an outline of the planned work for the master thesis "Accelerator for Event-based Failure Prediction". The document addresses the problem considered by the thesis, a state of the art outline as well as important questions and hypothesis that arise or are needed for the thesis. Further the document provides a list of necessary theory bases and introduces the design and experimenting methods applied in the thesis. Finally, the document gives an overview of necessary material in terms of literature and equipment, shows the preliminary work progress and provides a broad time schedule.

2 Problem and Research Area

In today's live it becomes increasingly important, that computer systems are dependable. The reason being, that computer systems are used more and more in areas where the failure of such a system can lead to catastrophic events. Banking, public transportation and medical engineering are only a few examples of areas employing large and extremely complex systems. The increasing complexity of computer systems has a direct impact on their maintainability and testability. It is simply impossible to guarantee that a piece of software comes without any faults. On top of that, the same problematic arises with the hardware components which also may contain faulty parts but also gets increasingly prone to failures due to decay of material.

In the event of a system failure it is of course desirable to fix the system as soon as possible in order to minimize the downtime of the system (maximize the availability). This can be accomplished by using different types of recovery techniques, e.g. Check-pointing (create checkpoints to roll back/forward), system replication (switch to a redundant system), fail over (reboot). All these techniques require a certain amount of time to complete the recovery process, time that is very expensive. In order to minimize this time, techniques have been developed to anticipate upcoming failures. Such a technique is described in [11].

The work presents a new algorithm to predict failures and compares the results with other techniques. The accuracy of the presented algorithm to predict failures proves to be better compared to the other techniques, has however the drawback of

increased complexity and hence increased computation time. It is very important to keep the computation overhead very low in order to maximize the time between the prediction of a failure and the actual event of the failure. One way to decrease the computation time is to design a hardware accelerator for the prediction algorithm. The design of such an accelerator is outlined in this document.

The presented algorithm uses system information, e.g. log entries, to predict failures during runtime (online). To be able to interpret the log files, it is important that they are structured in a standardized format. Further it is necessary input real log files to the algorithm in order to verify its performance (in terms of precision and speed). Unfortunately it is not possible to use the same data already used in the reference work, because of confidentiality issues. For this reason, new test data have to be generated by respecting the data properties described in the reference work as well as respecting a log representation standard.

3 State of the Art Outline

This section provides a brief outline of the state of the art in the different fields of research that are relevant for the thesis. This includes a small overview of failure prediction methods, existing solutions to accelerate failure prediction algorithms and concepts of log standardization.

3.1 Failure Prediction

A very detailed overview of failure prediction methods is given in [12]. The techniques used as comparison in the main reference [11] as well as the described technique are listed below.

- Event-based Failure Prediction [11]
- Dispersion Frame Technique (DFT) [6]
- Eventset [14]
- SVD-SVM [4]

Maybe some words about the counter ideas (SEER: a lightweight online failure prediction approach)

3.2 Accelerator

The following list presents designs to accelerate algorithms related to machine learning and the computation of hidden Markov models:

- [2] presents an architecture for a lightweight Viterbi accelerator designed for a embedded processor datapath.
- [3] describes a FPGA based accelerator for the SVM-SMO (support vector machine - sequential minimal optimization) algorithm used in the domain of machine learning.
- [5, 7, 9] describes a FPGA based accelerator for protein sequence HHM search. The Viterbi algorithm is used.
- [15] describes i.a. an approach to accelerate the Viterbi algorithm from the HMMER library using GPUs.

3.3 Log Standardization

[13] describes an approach on how to standardize logs. Still missing are sources for:

- data generation in order to verify an algorithm
- metrics to measure log quality
- more on log standardization

4 Questions and Hypothesis

The following list is intended to give an overview of open questions:

- how to parallelize the algorithm
- what kind of accelerator (SVM, GPU, just use multiple cores)
- how to generate test data to verify the accelerator
- ...

The accelerator will be designed by taking the following hypotheses into account:

- Algorithm of Felix is accurate but slow (to verify by online tests)
- the system using the accelerator provides standardized log events
- only the online part of the algorithm needs to be accelerated
- ...

5 Theory Base

5.1 Taxonomy of Dependable Computing

- Fault-Error-Failure chain [1]
- Precision and Recall [10]
- F-Measure [8]

5.2 Algorithms

- Tupling and Grouping of log events
- Forward algorithm
- Baum-Welch algorithm
- Classification

6 Method, Design and Experiments

This enumeration gives a broad overview on how the design of the accelerator will be accomplished.

1. analyze the possibility of parallelization of the online algorithm
2. generate sets of test data (log events)

3. implement the complete algorithm with Octave (Matlab)
4. run the Octave model with the generated test data for verification
5. implement the online algorithm in C (create executable)
6. run the online part without accelerator (on CPU)
7. implement an accelerator for the online part
8. run the online part with the accelerator
9. compute speedup

7 Preliminary Work Progress

todo...

8 Time Schedule

todo...

References

- [1] A. AVIZIENIS, J.-C. LAPRIE, B. RANDELL, AND C. LANDWEHR, *Basic concepts and taxonomy of dependable and secure computing*, IEEE Transactions on Dependable and Secure Computing, 1 (2004), pp. 11–33.
- [2] M. AZHAR, M. SJALANDER, H. ALI, A. VIJAYASHEKAR, T. HOANG, K. ANSARI, AND P. LARSSON-EDEFORS, *Viterbi accelerator for embedded processor datapaths*, in IEEE International Conference on Application-Specific Systems, Architectures and Processors (ASAP), July 2012, pp. 133–140.
- [3] S. CADAMBI, I. DURDANOVIC, V. JAKKULA, M. SANKARADASS, E. COSATTO, S. CHAKRADHAR, AND H. GRAF, *A massively parallel FPGA-based coprocessor for support vector machines*, in IEEE Symposium on Field

Programmable Custom Computing Machines (FCCM), April 2009, pp. 115–122.

- [4] C. DOMENICONI, C.-S. PERNG, R. VILALTA, AND S. MA, *A classification approach for prediction of target events in temporal sequences*, in Principles of Data Mining and Knowledge Discovery, T. Elomaa, H. Mannila, and H. Toivonen, eds., vol. 2431 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2002, pp. 125–137.
- [5] A. JACOB, J. LANCASTER, J. BUHLER, AND R. CHAMBERLAIN, *Preliminary results in accelerating profile hmm search on fpgas*, in IEEE International Parallel and Distributed Processing Symposium (IPDPS), March 2007, pp. 1–8.
- [6] T.-T. Y. LIN, *Design and evaluation of an on-line predictive diagnostic system*, PhD thesis, Carnegie-Mellon University, Pittsburgh, PA, 1988.
- [7] R. P. MADDIMSETTY, J. BUHLER, R. D. CHAMBERLAIN, M. A. FRANKLIN, AND B. HARRIS, *Accelerator design for protein sequence hmm search*, in International Conference on Supercomputing, ICS '06, New York, NY, USA, 2006, ACM, pp. 288–296.
- [8] J. MAKHOUL, F. KUBALA, R. SCHWARTZ, AND R. WEISCHEDEL, *Performance measures for information extraction*, in In Proceedings of DARPA Broadcast News Workshop, 1999, pp. 249–252.
- [9] T. OLIVER, L. YEOW, AND B. SCHMIDT, *High performance database searching with hmmer on fpgas*, in IEEE International Parallel and Distributed Processing Symposium (IPDPS), March 2007, pp. 1–7.
- [10] C. J. V. RIJSBERGEN, *Information Retrieval*, Butterworth-Heinemann, Newton, MA, USA, 2nd ed., 1979.
- [11] F. SALFNER, *Event-based Failure Prediction*, PhD thesis, Humboldt-University of Berlin, February 2008.
- [12] F. SALFNER, M. LENK, AND M. MALEK, *A survey of online failure prediction methods*, ACM Comput. Surv., 42 (2010), pp. 10:1–10:42.

- [13] F. SALFNER, S. TSCHIRPKE, AND M. MALEK, *Comprehensive logfiles for autonomic systems*, in IEEE International Parallel and Distributed Processing Symposium (IPDPS), April 2004, pp. 211–218.
- [14] R. VILALTA AND S. MA, *Predicting rare events in temporal domains*, in IEEE International Conference on Data Mining (ICDM), December 2002, pp. 474–481.
- [15] J. WALTERS, V. BALU, S. KOMPALLI, AND V. CHAUDHARY, *Evaluating the use of gpus in liver image segmentation and hmmer database searches*, in IEEE International Parallel Distributed Processing Symposium (IPDPS), May 2009, pp. 1–12.