TECHNISCHE UNIVERSITÄT
ILMENAU

FG System- und Software-Engineering

Master Thesis

Methods for detecting the warmup period of stochastic Petri net simulations

Starting Date: 01. July 2015
Submission Date: 30. October 2015

Supervisor: Prof. Armin Zimmermann

Submitted By: Sushma Nagaraj
Matrikel-Nr. 51385
sushma.nagaraj@tu-ilmenau.de

Contents

List of Figures

List of Tables

2

Abstract

The cardinal dynamic nature of a discrete event simulation and its instantaneous state changes at occurance of specific events, make it difficult for the distribution of the model output values to attain a constant, steady behaviour; especially, at the very start of the simulation run. The presence of this prominent bias at the beginning of the simulation leads to an imprecise estimation of the system parameters.Removing the transient period by estimating the appropriate truncation point and looking for the start of the steady state, ignoring the transient phase have been the two promising solutions to this start-up bias, but however, most approaches from the literature have been subjective. A transient solution algorithm that can be evaluated for its performance has to be more comprehensive.

The intent of this master thesis is to find a method that can achieve the best accuracy out of any simulation run. There are many methods that have already been documented and the many other which might be suggested for the dealing with the initial transient. However, there are no pre-fixed parameters for quantizing the evaluation of these methods. One of the main deliverable from this work is a software framework, that provides an interface for comparing various different truncation methods with each other. The integrity of the evaluated algorithms is the output obtained from the developed interface, and is decided by running the heuristics against the different benchmarks, with random number seeds.

A variety of methods which included online methods and offline - converted into online methods were implemented in the current framework and tested with different model data that was obtained from TimeNET. The final artifacts of this work include an executable framework, the comparison between different methods deduced from the results given by the framework.

1 Introduction

1.1 Motivation

The accuracy of the performance estimates obtained from a simulation model can be ascertained in two ways : Removal of any initialization bias that may be present and establishing sufficient output data for obtaining an accurate estimate of the performance. Initialization bias iccurs when a model is started in an unrealistic state This work mainly focuses on the the first issue, concentrating on developing an automated framework for estimating and removing the initialization bias from the simulation output data. There is also description about the comprehensive literature research that was carried out in order to shortlist and test the suitable methods.

This research work deals with a few pre-existent algorithms like MSER-5 and Schruben's tests, which are already popular in the simulation field and a few other algorithms like the sequential steady state detection and the Euclidean distance method which are not so popular.

There are five main methods for dealing with initialisation bias [HRD09]:

1. Run-in model for a warm-up period until it reaches a realistic condition (steadystate for non-terminating simulations) and delete data collected from the warm-up period.
2. Set initial conditions in the model so that the simulation starts in a realistic condition.
3. Set partial initial conditions then warm-up the model and delete warm-up data.
4. Run the model for a very long time making the bias effect negligible.
5. Estimate the steady-state parameters from a short transient simulation run.

1.2 Goals

The aim of this project is to develop a conceptual test environment for testing the LiSARD softcore programs. The developed environment should

- Enable the user to enter a program.
- Provide input data for execution during run-time.

- Point out the syntax errors.
- Point out the errors in the variable usage.
- Build/Compile the program.
- Compute the output of the program with user-provided input values.
- Display the computed output to the user.

The tasks in the project can be summarized as:

- Understand the program structure of the LiSARD softcore programs.
- Understand the code-conversion that happens from LabVIEW to FPGA.
- Saving the compilation time during programming and testing.
- Develop a tool with the features mentioned above.
- Compare the tool computed output to the FPGA computed output for a user-program (testing).

1.3 Structure of the paper

The basic idea of System on Chip, FPGA and need for LiSARD for the FPGA was described in this section. The remainder of this paper is organized as follows.

Section 2 gives an overview of the LiSARD softcore processor, its program structure and its memory details. It explains the program syntax, assumptions used during the development and the basic operations that the tool can now support.

Section 3 explains the code conversion that happens during the execution of the softcore program.

Section 4 explains the designed algorithm for testing the softcore processor program that is included in the tool. This section only explains the algorithm for the program evaluation omitting the user-interface details.

Section 5 explains the implementation details of the developed tool. This includes the installation instructions, the user-interface description and the various functionalities that can be used while running the tool and a detailed walk through of the various developed features.

Section 6 provides a comparison in the performance of the actual hardware and the advantages of using the developed tool. The section also emulates the expected output and the obtained results for a predefined program and set of inputs. This section infers the precision that can be obtained by the developed tool, recommending its usage in a real-time environment.

Section 7 provides a round-up to the project work with brief details about the research work, challenges faced, advantages and drawbacks of having such a tool and its possible useful enhancements.

Bibliography

[HRD09] Hoad, K. ; Robinson, S. ; Davies, R.: Automating warm-up length estimation. In: *Journal of the Operational Research Society* (2009). `http://dx.doi.org/10.1057/jors.2009.87`. – DOI 10.1057/jors.2009.87