Performance Modelling

Practical 2

26th October, 2017

This is the second of two practicals. The coursework accounts for 25% of the marks for the course, and this practical accounts for 50% of the coursework mark. Each practical is marked out of 50. The deadline for the second practical is 16:00 on **Wednesday 29th November**. Submissions should made on paper to the ITO AND electronically, as described, at the end of this document. This is an individual practical: whilst general discussion with classmates is fine, the solutions that you submit must be your own work.

This practical is intended to give you some hands-on experience of developing larger models using GSPNs and PEPA. It is recommended that you use the PIPE Tool v4.3.0 (available from `http://sourceforge.net/projects/pipe2/files/PIPEv4.3.0/`) and the Eclipse PEPA Plug-in Project (available from `http://www.dcs.ed.ac.uk/pepa/tools/plugin/`). Download and install these on your own computer or in your Informatics DiCE account by following the instructions on the Web sites. For the PIPE tool you will find useful information in Lecture Note 8. For the PEPA tool you will also find it useful to refer to Lecture Note 11. Example models, for both GSPN and PEPA are available from the course webpages.

Good Scholarly Practice: Please remember the University requirement as regards all assessed work for credit. Details about this can be found at:

`http://web.inf.ed.ac.uk/infweb/admin/policies/`
`academic-misconduct`

Furthermore, you are required to take reasonable measures to protect your assessed work from unauthorised access. For example, if you put any such work on a public repository then you must set access permissions appropriately (generally permitting access only to yourself, or your group in the case of group practicals).

1 Using GSPN to analyse an automated manufacturing system

In an automated widget factory, widgets are assembled from two parts, an A part and a B part. A parts are processed by machine 1 while B parts are processed by machine 2; machine 3 then assembles one A part and one B part to make one widget. A single robot transports parts from a conveyor belt to the appropriate machine; it is also responsible for moving completed A parts from machine 1 to machine 3, and completed B parts from machine 2 to machine 3. Machine 1 cannot start to load the next A part until the current one has been moved to machine 3; similarly for machine 2 and B parts. Machine 3 cannot accept parts of either kind until it has completed the assembly of the previous widget. There are always A and B parts available from the conveyor belt. If both machine 1 and machine 2 need to use the robot at the same time they are equally likely to acquire it. Loading parts from the conveyor belt, or transferring them to machine 3 takes 10 seconds on average. The mean duration of processing A parts at machine 1 is 125 seconds, while the mean duration of processing B parts at machine 2 is 200 seconds. Assembling a widget from A and B parts takes 100 seconds on average. Assume that all times are exponentially distributed.

a.) *Draw a GSPN model to represent the system as described above, taking care to either give a meaningful label to each place and transition, or alternatively provide a separate table explaining the role of each place and transition. You should also include the firing rate or firing weight for each transition.*

4 marks

Note that there is more than one correct way of representing the system as a GSPN.

Machine 1 is getting old and rather temperamental: for approximately 1 part in 20 it jams during processing and needs to be repaired. On average, the repair time is 500 seconds. The processing of that part then continues. Extend your GSPN model to include the breakdown and repair of machine 1.

b.) *Extend your GSPN model to include the breakdown and repair of machine 1.*

2 marks

Using this GSPN and the PIPE tool answer the following questions.

c.) *What is the utilisation of each of the Machines 1, 2 and 3?*

3 marks

d.) *What is the throughput in terms of widgets per second?*

1 mark

Since machine 1 soon blocks the whole system when it is out of action ways of avoiding this problem are being investigated. In particular, it has been decided to experiment with placing a buffer between machine 1 and machine 3. Now, a completed A part is transferred to the buffer, not to machine 3 directly. When machine 3 is ready to assemble another widget it loads an A part from the buffer, taking 2 seconds on average. The buffer only has limited capacity because space is limited within the workcell; initially the buffer capacity is set at 3.

e.) *Construct another GSPN to reflect the new situation.*

3 marks

With this new GSPN and the PIPE tool answer the following questions:

f.) *What is the effect of introducing the buffer on the throughput of Machine 3? Does the size of the buffer matter?*

3 marks

g.) *What other steps could be taken to improve the throughput of the system? Which do you think are likely to be most effective?*

5 marks

h.) *The foreman has noticed that if Machine A is run at half the speed it does not jam and never needs to be repaired. He thinks that at least as good throughput as that derived in part d.) could be achieved without the buffer if Machine A was run at this speed. Is he right? Justify your answer.*

3 marks

24 marks total

2 Modelling an online-game platform in PEPA

2.1 Overview

An online-game platform allows users to play various online-games. It is important to the platform operators to keep the response time to users low even when the number of simultaneous online players is very large. Thus, they would like a PEPA model to be constructed and used to investigate the present and possible future performance of the platform.

The backend of the online-game platform consists of three kinds of server: the *authentication server*, the *data retrieval server* and the *game server*. The authentication server is responsible for the logging in and out of user accounts. The data retrieval server provides existing data to users (e.g., the rank list of players, the user profiles, etc.). Lastly, the game server serves users currently engaged in a game.

Your task will be to construct a PEPA model of the system, using appropriate components.

2.2 Description

We begin with a description of the online users and the process which they follow in order to play a game or browse some information on the platform. This presents the user's view of the process in terms of activities and choices along the way.

- An online user's first action is to make a *login request* (2 seconds) to the authentication server; then, they have to wait for the reply of the request.

- The outcome of the login request can either be *approved* or *declined*.
- If the request is approved, the user can do further actions on the platform; otherwise they have to try logging in again.
- Once the user is authenticated, they will have a *think* phase (5 seconds) to decide what action they are going to do next.
- Generally, 40% of the users choose to browse some data on the platform, 40% of them will play a game whereas the other 20% of the users simply *log out* (2 seconds) of the platform.
- To browse data on the platform, the users need to *request data* (2 seconds) from the data retrieval server.
- They have to wait for the data to be *returned*, and then they can *read* (30 seconds) the data.
- To play a game, they must *request a game service* (2 seconds).
- The game cannot start until a game service is *allocated* by the game server.
- On average, it takes about 3 minutes to *play a game*.
- Lastly, after finishing browsing data or playing a game, the user returns to the think phase to *think* about their next action.

You may assume that there is a fixed of online population of users who cycle through these behaviours (e.g. a user who logs out is immediately replaced by another user who makes a login request).

The authentication server is a passive system. It takes no action until a login or logout request is placed by a user. Then, it will *process the request* (1.5 seconds). It takes 1 second to reply the user with the authentication outcome. Statistics show that 90% of login attempts are *approved*, and 10% are *declined*.

The data retrieval server simply *processes* (1.5 seconds) the *data request* once a *data request* is placed by a user. Then, it *returns* (0.5 second) retrieved data to the user.

The game server is responsible for *creating a game service* (10 seconds) when it receives a game service request, and then *allocating* (0.5 second) it to the user.

Assume all times are exponentially distributed.

a.) *Model the above system using PEPA and the PEPA eclipse Plug-in.*

8 marks

b.) *Explain five steps you might take to verify and validate your model.*

5 marks

c.) *Use your model to estimate the utilisation of each server when there are 5 simultaneous online users and one copy of each server.*

2 marks

d.) *Keeping the response time of user actions low is of vital importance to the platform operators. If you are the platform operator, which actions will you be most interested in measuring the response time for (you may give more than one actions)? Give the response time of these actions when there are 10 simultaneous online users and one copy of each server.*

3 marks

e.) *How well will the system perform when there are hundreds of online users? What steps do you suggest the platform operators take to improve the user experience. You will need to use scalable analysis to investigate.*
Give a brief account of your investigation (1/2 - 1 page).

8 marks

26 marks total

Submission instructions

Your submission for this second practical is to be electronic only, with no additional paper documents.

1. A typed document containing your answers to Questions 1 and 2.
2. XML files with the GSPN models for Question 1 that can be loaded into the PIPE tool.
3. Your PEPA model in a file `practical2.pepa` that can be loaded into the PEPA Eclipse Plug-in Tool.

Submit your work for assessment using the following command:

```
submit pm cw2 <name of file 1> <name of file 2>...
```

You can submit more than once up until the submission deadline. All submissions are timestamped automatically. Identically named files will overwrite earlier submitted versions, so we will mark the latest submission that comes in before the deadline.

If you submit anything before the deadline, you may not resubmit afterward. (This policy allows us to begin marking submissions immediately after the deadline, without having to worry that some may need to be re-marked).

If you do not submit anything before the deadline, you may submit *exactly once* after the deadline, and a late penalty will be applied to this submission unless you have received an approved extension. Please be aware that late submissions may receive lower priority for marking, and marks may not be returned within the same timeframe as for on-time submissions.

The deadline for the homework is 16:00 on ***Wednesday 29th November 2017****. Marked work will be returned on or before 13th December 2017.*