Venice PhD course: Lab 3

April 15th 2013

1 Modelling a B2B Application

This exercise involves constructing a more substantial PEPA model and carrying out some analysis to determine the performance of the system represented. It is based on a genuine case study of a bank process undertaken during a European collaborative research project. Read the system description carefully before you start to construct your model. The actions to use in your model have been identified for you.

1.1 Overview

A credit portal is a business-to-business (B2B) application where a bank wishes to offer its loan services to business customers. It is important that loan applications are processed efficiently (so that the bank does not lose customers to a rival bank). A high degree of automation is provided to meet this demand for efficient processing. Where human intervention is needed in the decision-making process this too must be driven by a deep sense of urgency in processing the request in a timely manner.

Some businesses which use the services of the bank are long-established customers with a lengthy and well-documented history of financial solidity and probity, together with substantial credit reserves and securities. Modest loan applications from these customers can be directly approved by a *pre-decision* process which validates the content of the loan request and the securities. In such a case it is possible that the decision to lend can be taken entirely by the application of the predefined rules stored on the server and executed by the local rule engine. This is the fastest route to approval of a loan request because it is one which completes entirely without human intervention.

Not all loan requests can be so rapidly approved. Many will require more lengthy scrutiny, evaluation and checking. In these cases a bank clerk will process the credit request. There are several possible next outcomes here. First, the clerk may approve the request but must then forward the credit request to a supervisor who must also approve the request. Second, the clerk may decline the request. Third, the clerk may enter into a negotiation with the customer with the intention of updating the request to reduce the capital requested, or change the terms of repayment. This then initiates another request from the customer which is to be processed in the manner just described.

1.2 Description

We begin with a description of the bank's customers and the process which they follow in order to secure a loan from the bank. This presents the customer's view of the process in terms of activities and choices along the way.

The customer's first action is to initiate a loan (*request*). To carry this through they must enter their balance data and securities (*enterData*) and send this to the credit portal with an XBRL upload (*uploadData*). (XBRL is the eXtensible Business Reporting Language.) The customer then waits to see if the request will be approved or declined (the *approve* and *decline* activities respectively, undertaken at the same rate, r_{inform}). If the request is approved the customer has no further business and the next waiting customer can be considered. If the request is declined the customer can try again (*reapply*) and will do so with probability t_0. If they do not wish to reapply they can yield to the next customer.

The service is a reactive system. It takes no action until the upload of XBRL data is complete. At this point it validates the data by using a validation web service which determines whether or not the balance data is valid (*validateData*). We are not interesting in the processing of invalid data in this scenario and so the next behaviour which we model is passing the valid data to the bank (*sendBank*). The service is then ready to receive the next request.

The relevant business functions of the bank are expressed in the *Bank* component. This documents the "predecision" phase which can have three possible outcomes. Some applications can be immediately approved (with probability p_0), and others immediately declined (with probability p_1). Some proportion need to be processed by a bank employee (with probability p_2) who will either decline the loan (with probability q_1), or approve it (with probability q_0). Approval requires confirmation, which may be forthcoming (with probability s_0) or not.

Values for rates and probabilities can be found in Tables **??** and **??**.

Customer rates		Service rates		Bank rates	
$r_{request}$	$= 3.11944$	$r_{validate}$	$= 1.43141$	$r_{predecide}$	$= 5.15757$
$r_{enterData}$	$= 0.04667$	$r_{sendBank}$	$= 1.53785$	r_{decide}	$= 0.01221$
r_{upload}	$= 0.88424$			r_{inform}	$= 0.45729$
$r_{reapply}$	$= 0.02036$				

Table 1: Table of rate values used in the model. All rates are expressed at the granularity of minutes. The reciprocal of the rate gives the mean or expected value of the duration of the activity. Thus, the average time for one bank employee to decide on a loan is about 82 minutes ($1/r_{decide} = 81.90008$).

1. Model the above system using PEPA and the PEPA Eclipse Plug-in.
2. Use your model to estimate the average waiting time for the customer from submitting a request to receiving a response in the form of *approve* or *decline* when there are 6 customers and one bank.

Predecision	**Employee decision**	**Supervisor decision**	**Reapplication**
$p_0 = 0.20907$	$q_0 = 0.17441$	$s_0 = 0.56423$	$t_0 = 0.08970$
$p_1 = 0.32075$	$q_1 = 0.82559$	$s_1 = 0.43577$	$t_1 = 0.91030$
$p_2 = 0.47018$			

Table 2: Table of probability values used in the model. Each column sums to 1.

3. Which one action within the bank process would you speed up to improve this waiting time? Justify your answer.

4. Additionally evaluate the B2B PEPA model to consider how well the system scales when there are hundreds of clients, based on performance measures of interest. You will need to use scalability analysis for this but it up to you whether to use simulation or ODEs.