Assignment 4
Evacuation Analysis of Bmod

Cláudio Gomes
`claudio.gomes@uantwerp.be`

July 29, 2017

1 Practical Information

The goal of this assignment is to implement a translation that takes a Bmod floor plan model, and outputs a stochastic petri net model, that represents the movement of people in case of an emergency.

This is important because, in an emergency situation, there can be no agglomeration of people in any of the emergency routes.

1.1 Task Overview

Task 1 *Change the Bmod language to include information about emergency routes.*

Task 2 *Implement a model transformation that creates a stochastic petri net model that represents the movement of people in an emergency situation.*

Task 3 *Generate pipe2 inputs, from a given stochastic petri-net model.*

Task 4 *Analyse a non-trivial building floor model.*

Task 5 *Write a report.*

1.2 Deadline and Logistics

Complete this assignment in **groups of 2**.

One, and only one, person in the group must submit the solution on Blackboard before TBA.

Contact Cláudio Gomes (`claudio.gomes@uantwerp.be`) if you have questions.

2 Requirements

2.1 Task 1

An emergency route is a path that starts at any room, and ends in an emergency exit. An emergency exit is a special door.

Not all building entrances can be used as emergency exits though. So the user of Bmod needs to be able to specify which doors are emergency exist of the building.

Additionally, we need to know how much time a person, in an emergency situation, takes to cross each room, and each door. This is important because, using stochastic petri-nets, the average number of people crossing through a room, or standing at a door, can be computed. This information has to be specified by a user of Bmod.

Expert. The emergency routes are calculated automatically as a pre-processing step of the transformation made in Task 2.

2.2 Task 2

The goal of the resulting stochastic petri net, is to capture the movement of a given simultaneous number of people along the multiple emergency routes.

We assume that the given number of people start at one of the entrances of the building (one that is not an emergency exit), and the petri-net represents the routing of these people, along the multiple emergency routes.

Once a person leaves an emergency exit, it is teleported to the initial entrance immediately. This is because we want the analysis to capture an infinite number of movements along the emergency routes.

Assume that there is one building entrance which is not an emergency exist, and that there are at least two different emergency exits. Hence, at any point in the route, there can be at least two possible paths to the emergency exits.

It is up to you to decide how a person decides which of the paths to follow for the emergency exit. For example, a reasonable decision is that the probability of taking the shortest path to the exit is proportional to how close that emergency exit is. That way, you can model the fact that most people will exit the building via the shortest path, with the occasional person making the wrong choice (in panic off course) and taking the longer route to the exit.

Notice that at any point in the emergency route the person is faced with this kind of decision, as long as there are multiple routes to the emergency exit. You cannot assume that a person makes a decision to taken a certain route, and then sticks to that decision. In a panic situation, the decisions are immediately, and they affect very little the next decision.

This petri-net, after Task 3 is complete, can be loaded into the tool pipe2. Once loaded, the user can place a certain number of people (which are petri-net tokens), at one of the entrances of the building, and run the analysis.

Expert. When a person is at a point that has more than one possible emergency route, the probability that he/she takes a given route is proportional to how short that route is to the exit. For example, if there are two routes, one which is at a distance of 4, and another one at a distance of 5, then the likelihood that the first route is taken is $\frac{(4+5)-4}{(4+5)}$, and the likelihood that the second one is taken is $\frac{(4+5)-5}{(4+5)}$.

2.3 Task 3

The goal of this task is to create a file representing the petri-net model created in Task 2 that can be loaded into pipe2.

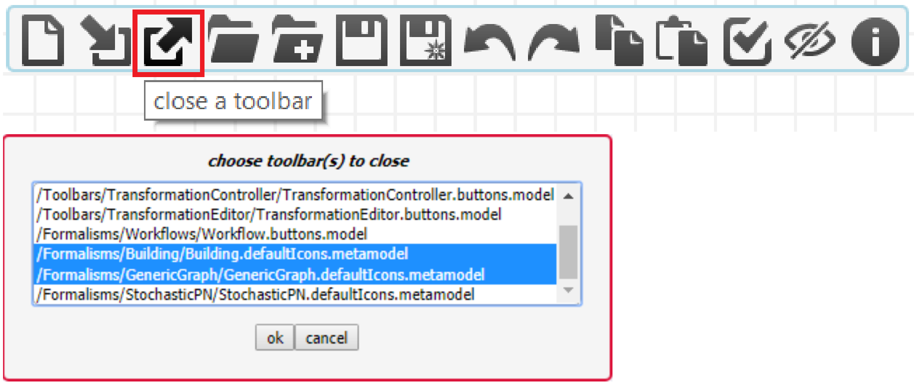
The petri net model created in Task 2 needs to be exported to a model in metaDepth. Below is an explanation on how to do this.

The exported file is then used as input to the .egl file that you will code.

Create a simple petri-net model in pipe2 and inspect the corresponding .xml file, to understand the format used.

3 Tutorial

Exporting petri-net models from AToMPM. After running the transformation created in Task 2, remove the traceability links and original building plant model by closing the respective toolbars:



Load the MetaDepth toolbar (inside of the */Toolbars/MetaDepth/* folder, it is called *Export.buttons.model*). This toolbar has two buttons: one for exporting models, and one for exporting metamodels.

The toolbar should like as follows:

If you open the Stochastic petri-net metamodel file (called *StochasticPNMM.model*), you can export it by pressing the export metamodel button.

With an open petri-net model, you can export it by pressing the export model button.

4 Tips, Tricks, Pitfalls, and Issues

Bug. If you get a null pointer exception when running EGL, it may be because you forgot a ";" somewhere.