School of Innovation,
Design and Engineering

7. Starting to use Model-based Testing

This document provides the laboratory instructions for the third session on Model-based testing. The objective of the lab is to get experience of using Model-based Coverage Based Test Generation.

Please use the following link section to install UPPAAL 4.0:

http://www.it.uu.se/research/group/darts/uppaal/download.shtml

To find out more about UPPAAL, read this short introduction:

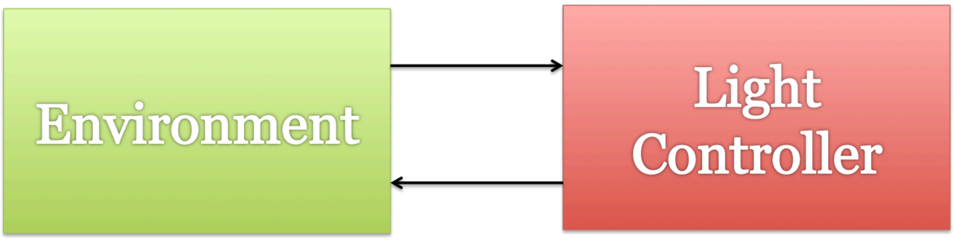
http://www.it.uu.se/research/group/darts/uppaal/about.shtml#introduction

Exercise 3 - Model-based Testing

In this exercises we will focus on model-based testing. The exercises are performed using the UPPAAL tool and a set of pre-implemented models.

Light Controller Description

Consider a simple light switch controller. The user interacts with the Light Controller by touching a sensitive pad. The light has three level of intensity: OFF, DIMMED, and BRIGHT. Depending on the timing between successive touches, the controller is in charge for changing between the light levels. For example, in the DIMMED state, if a second touch that caused the controller to enter the DIMMED state, the Light Controller increases the level to BRIGHT. On the other hand, if the second touch happens after the switching time, the Light Controller switches the light OFF. If the Light Controller has been in the OFF state for a long time it should reactivate upon a touch by going directly to the bright level.



Lets assume that the test specification is given as two models: the Environment model and the Light Controller model. We consider that the Light Controller is the (System-Under-Test) SUT that can operate in a specified Environment model.

Web:
http://www.es.mdh.se/staff/349-Eduard_Paul_Enoiu

E-mail: eduard.paul.enoiu@mdh.se

Tasks: Model-based Single Purpose Test Generation

1. Unzip the pre-implemented models from the following link: http://www.promptedu.se/promptwp/wp-content/uploads/2014/10/9.Lab3_.zip
2. Open the provided "`LightController_1.xml`" file.
3. Read the specifications for the Light Controller problem (above)
4. Double-click in the navigation panel on the "`LightController`" item. This is a simple implementation of the Light Controller specification. Have a brief look at the "`LightController`" model. Next, have a brief look in the "`Environment`" model.
5. Check whether the Light Model can become bright. Test your query.
6. Select the Options Menu and the "Diagnostic Trace" item. Click on the "Shortest" sub-item. In the Verification tab, select the "Check" item.
7. A new dialog window will be opened. Click "Yes" for storing a new trace in the simulator. Click on the "Simulator" view of the Uppaal tool and select from the "Simulation Trace" "(OFF,-)" element. Click on the "Next" button and have a brief look on the Light Controller and the Environment models. The test sequence, that checks that the light can become bright, is generated and can be examined.
8. Click on the "Verifier" view and select the Options Menu and the "Diagnostic Trace" item. Click on the "Fastest" sub-item. Select the "Check" item. Now repeat the steps used at task 6.
9. Open the provided "`LightController_2.xml`" file. Model an environment that can be used to serve the following test goal: "*The light switches off after three successive touches*". Check this test goal and generate the shortest /fastest test sequence.

Tasks: Model-based Coverage Based Test Generation

10. Open the provided "`LightController_3.xml`" file. Click on the "Simulator" view of the Uppaal tool and select the "View". Click on the "Mark Visited" item.
11. Check whether the Light Model can become bright.
12. Select the Options Menu and the "Diagnostic Trace" item. Click on the "Fastest" sub-item. Select the "Check" item.
13. In the "Simulator" view, select the "View" Menu, and click on "View Coverage" item. A new window will open in which you can find the edge and location coverage information. Hover the mouse over the graphics to get more details. Compare the current coverage information of the test property with the one generated for the "Shortest" trace.
14. Using the information you have from the Light Model check a property that ensures the *edge-coverage criteria*.

1. Send the solutions containing the `LightController_2.xml and LightController_3.xml` files to eduard.paul.enoiu@mdh.se.