Uppaal Lab

GIAN 2016

Exercise

Note: Please show your work to a T.A. before leaving.

The purpose of this lab is to give you a beginner's tutorial of UPPAAL model checker. You will learn the following concepts in this lab.

- Locations (Initial / Committed)
- Transitions (Select / Guard / Sync / Update)
- Channels (Urgent)
- Clocks
- Variables, Constants and Arrays
- Functions and Type declarations
- Simulation and Verification using UPPAAL

Description

Your task is to create a model of Train-Gate example using UPPAAL. The created model will be simulated and verified using UPPAAL.

Train-Gate Example

The system consists of a bridge that has to be shared between multiple train tracks. A (virtual) gate on the bridge allows one train to pass at a time. A train approaching the gate informs it and attempts to pass through, unless instructed to stop by the gate. Multiple simultaneous request may arrive (Max = 6), which are served on a FIFO basis. While processing a request e.g., letting a train pass through, the gate instructs all other approaching trains to stop until their turn. Upon its turn, a stopped train is instructed to go. The train thus starts again and leaves as soon as possible.

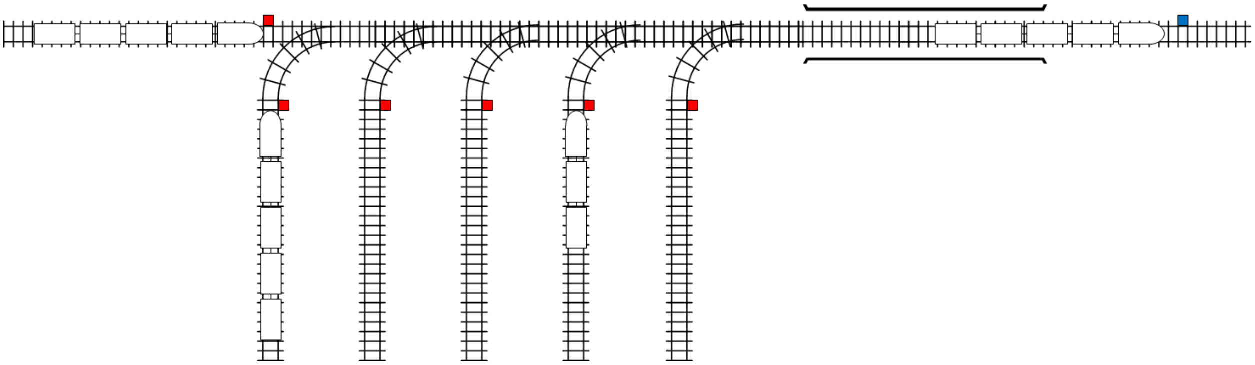

Fig. 1. A railway bridge, shared between multiple tracks. Approach and Leave sensors are visible.

Instructions

- Download "UPPAAL Lab1 Instructions.pdf" and follow the instructions given in the file.

- Create the Train-Gate System model and perform **Simulation** and **Concrete Simulation** using UPPAAL.

- Perform **Verification** of the following properties on this system.

1. *Train 2 can cross.*
2. *A situation is possible where Train 0 is crossing, while Train 1 is waiting.*
3. *When Train 0 is passing, all other trains must wait.*
4. *The wait queue never overflows.*
5. *The wait queue never underflows.*
6. *Whenever a train approaches, it eventually crosses.*
7. *System is deadlock free.*
8. *There is never more than one trains crossing at a time.*
9. *Create a property from your understanding of the system and verify it.*

- There are two errors in the model that you will create using the instructions. Find and fix those errors with the help of verification.

This is a 2-hour lab; however, you can always finish early. Demonstrate your understanding of the created model, simulation and verification process to a T.A. before leaving.