Vrije Universiteit Brussel

Programeer Project 2

2016-2017

User Guide Project

Naam: Carlos Montero

Rolnummer: 0500677

Augustus 21, 2017

Contents

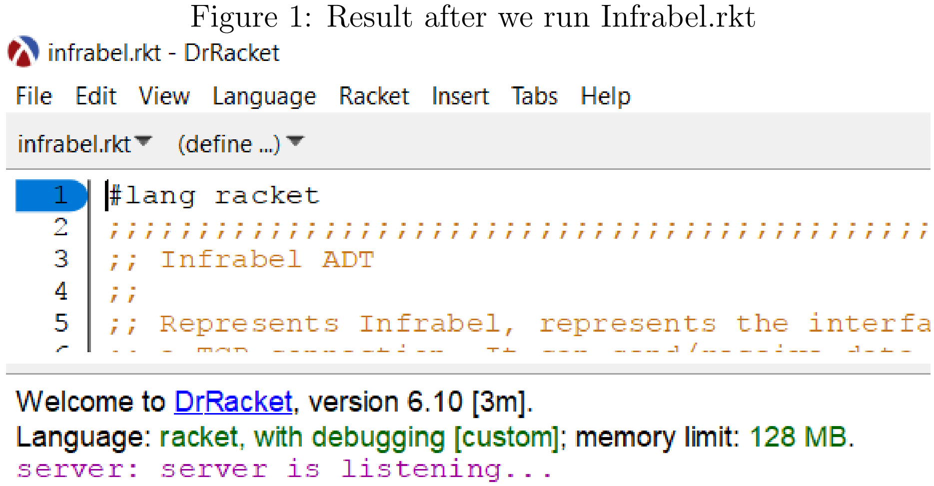
1 Introduction

In the next sections we will discuss the different functionalities that can be used for the implemented program. In the first section we will review how to run the program in a secure way. Section Input parameters explains how to change the input for NMBS and INFRABEL. For example changing the file that is used to compose the railway model.
The last section motivates how to use and interpret the GUI. It explains which commands NMBS understands and how to read the data presented in the GUI.

2 Running the Program

To be able to run the program we need to run two different files: "nmbs.rkt" and "infrabel.rkt", in the mentioned order. Whenever Infrabel is started the server listener will output "server: server is listening..." (see Figure 1). After the server is initiated it is safe to start nmbs. As soon as nmbs is started it will connect to the server of Infrabel through a TCP connection.

Figure 1: Result after we run Infrabel.rkt



3 Input parameters

The only parameters that can be changed before execution of "Infrabel.rkt" and "nmbs.rkt" are:

- rwm-file-hardware: contains the path to the file that will be used to compose the railway-model during a real life connection.
- rwm-file-simulation: contains the path to the file that will be used to compose the railway-model during simulation.
- simulation? : a Boolean that indicates if Infrabel should connect with the simulator or Z21.

Each of the last mentioned variables are defined in "nmbs.rkt" and "Infrabel.rkt". Hence to change the the file to be used, we need to make modification in "nmbs.rkt" and "Infabel.rkt". Also in "simulation.rkt" only when simulation? Boolean has been set on true.
Whenever we run nmbs we give it as input the value for "simulation?". This value will afterwards be send automatically through the TCP connection to Infrabel.

Figure 2: Variables that can be changed in nmbs.rkt

```
(define rwm-file-hardware "../NMBS/hardware_model.txt")

(define rwm-file-simulation "../NMBS/simplemodel.txt")

(define (nmbs simulation?)
  (define file (if simulation?
                   rwm-file-simulation
                   rwm-file-hardware))
```

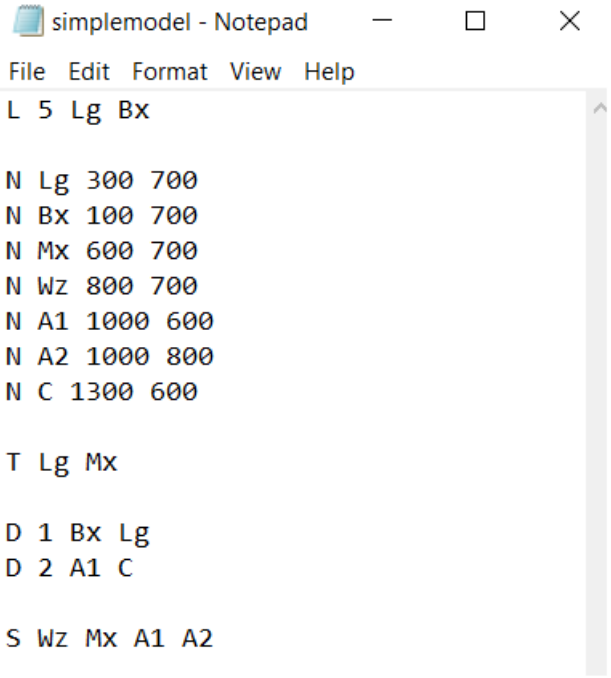
In figure 2 we choose to use file "hardware_model.txt" for a real life UDP connection with Z21 and file "simplemodel.txt" for connection with the simulator.

3.1 The input File

The way how the data from the input file is used to map to components of the railway-model is similar to the way how the simulator composes his

railway-model.

Figure 3: Example of an input file called simplemodel.txt

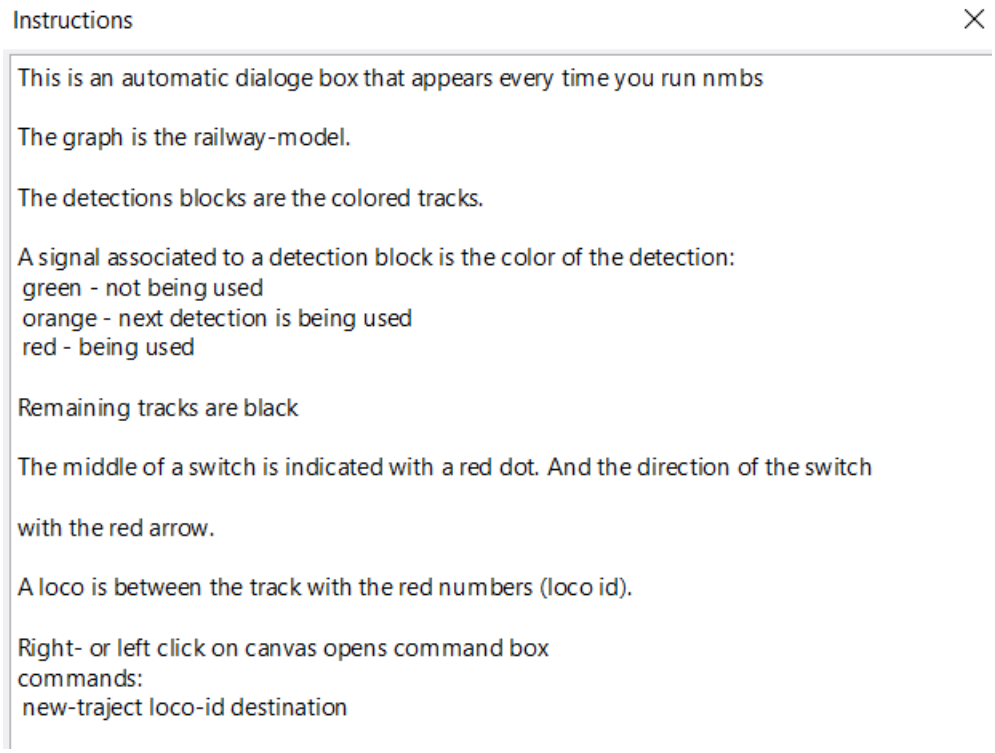


Each datum from input file "simplemodel.txt" is interpreted as follow:

- L 5 Lg Bx : a loco with ID 5 is located between stations Lg and Bx.
- N Lg 300 700: a Station with ID Lg and coordinates (300, 700).
- T Lg Mx: there exists a track between two stations Lg and Mx.
- S Wz Mx A1 A2: a switch where station Wz is the middle of the switch. The switch is composed by three tracks Wz-A1, Wz-A2 and Wz-Mx. To go from station Wx to A1 or A2 switch needs to be respectively at position one or two.
- D 2 A1 C: there exists a detection track with ID 2 between two stations A1 and C.

4 GUI

Whenever we start the program as mentioned in section "Running the Program" the GUI will pop up with the next window.

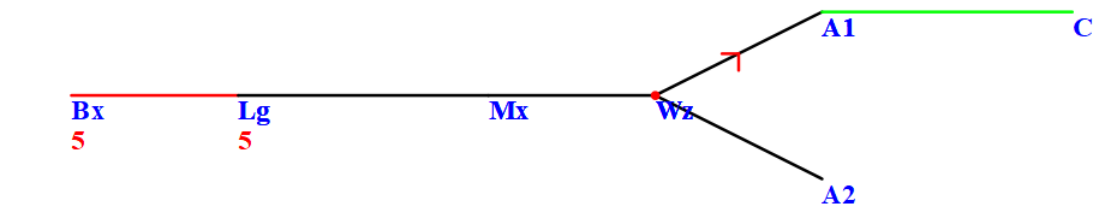


The window appearing is a briefly remainder on how to interpret and use the GUI.

4.1 Interpretation of the graph

After we close the Instruction window a canvas will remain.
The railway-model is represented in the GUI by a graph. Each station is denoted by it's ID (blue text). For each connection between two stations their exist a line.

Figure 4: Railwaymodel constructed based on file "simplemodel.txt".



A coloured line indicates that their exits a detection track between the two stations of the line. The state of the signal associated with a detection track is the colour of the detection line. The colour of the line can be:

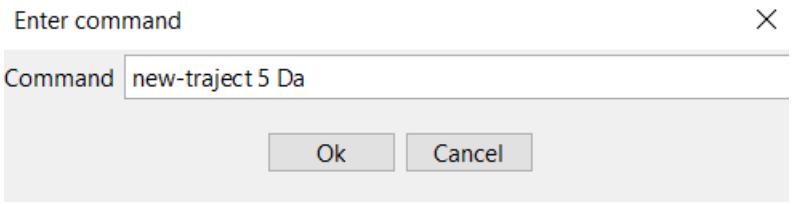
- green: indicates that the detection track is not used.
- red: indicates that the detection track is being used.
- orange: indicates that the next detection track is being used.

The remaining black lines are switch tracks or regular tracks.
The middle of a switch is indicated by a red dot. And the direction where the switch is set is indicated by the red arrow. The position of a loco is always between two stations and is represented by it's ID (red text).

The graph in figure 4.2 was constructed based on the file "simplemodel.txt". Loco with ID 5 is between stations Bx and Lg. The two stations are connected with a detection track, and the signal associated to the detection track is red. Stations Lg and Mx are connected with a regular track. Switch Wz is set to go to A1. Detection track between A1 and C is not used, the associated signal is set on green.

4.2 Introducing commands

To be able to start using the command prompt we need to right- or left-click on the canvas. This will make the next window appear



The only command understood by nmbs is of the form "new-traject loco-ID station-ID". This command will generate a new traject for the given loco-ID. The traject will be from the current position of the loco to the destination station. It is mandatory that each traject starts from and ends into a detection track. Therefore the station-ID has to be connected to a detection track.
All other type of command are unknown, or trajects that lead to a non detection track will be ignored.