# CDL7417: Notes on Running the Traffic Simulator

1. Download sim.pl, domain\_model.pl, and initial\_state.pl from BRIGHTSPACE WEEK 9. Put them in an appropriate directory (you can see the absolute address of the directory I use in sim.pl – depending on which operating system I use) e.g.

% WHERE INPUT COMES FROM ..

initial('../Users/lee/Desktop/coursesim/initial\_state.pl') :- computer(mac).

initial('c:/Users/lapadmin/Desktop/coursesim/initial\_state.pl') :- computer(windows).

1. Assume you are using windows. Then change the fact in sim.pl at about line 15:

computer(mac).

To

computer(windows).

Then change the three lines:

initial('c:/Users/lapadmin/Desktop/coursesim/initial\_state.pl') :- computer(windows).

domainmodel('c:/Users/lapadmin/Desktop/coursesim/domain\_model.pl') :- computer(windows).

tracefile('c:/Users/lapadmin/Desktop/coursesim/trace.txt') :-computer(windows).

So that ***the quoted path fits with the computer you are using.*** Make sure that you change sim.pl so that these addresses for the initial state, domain model and trace file are correct, otherwise the program will not be able to access these external files.

1. Load sim.pl into SWI Prolog, and run the simulation for 1200 seconds at one second granularity. The commands I run for this in SWI Prolog (on my Mac) are:

compile('../Users/lee/Desktop/coursesim/sim.pl').

sim(1200,1).

But you might be able to use the window on SWI to consult the sim.pl file, so after you have done that all you need do is execute the simulator with “sim(1200,1).”

*NB the simulation run will start with some Prolog “warnings” – please ignore them, and wait for the progress notifications that should appear on the screen e.g. “Time is 30 seconds fox going to intergreen”.*

If you have an error, and Prolog halts, it must be because you have not carried out step 2 correctly.

The file “trace.txt” file logs all the information every time step, such as occupancies of links and whether they are saturated (over 90% capacity). The program will output to screen the time that junction stages change stage and go into their intergreen.

At the end of the simulation, summary information is written to screen and to the trace. This includes information on the queues of traffic waiting to enter the region, in number of PCUs, e.g.

Queue waiting in the EAST: 34

Queue waiting in the WEST: 16

Queue waiting in ALBERT STREET: 7

Queue waiting in HEPTON ROAD: 0

Queue waiting in the CALLANS: 2

Queue waiting in the TRADES: 3

For a well functioning traffic system the size of these queues should be small.

PS if you want to run the simulator on one of the five coursework scenarios, clearly you need to change the line of code in sim.pl from:

initial('<YOUR PATH>/initial\_state.pl') :- computer(windows).

To

initial('<YOUR PATH>/scenario1.pl') :- computer(windows).

So that the correct initial state etc is read in.

1. Do not use a Time Step (Delta) of more than 5 seconds as the results start to vary quite a bit because the ordering of the processes in the same time step then tends to make a significant difference. Likewise, 20 minutes operation (1200 seconds) is adequate for testing.
2. Every time you want to run the simulator, make sure you re-start Prolog first. The reason why you can’t re-run it is that the program uses a lot of global data that persists when the simulation finishes.
3. Every time you run the simulator, trace.txt will be over-written. So if you want to keep a trace file, give it a new name before re-running the simulator e.g. trace28\_11.txt.
4. To check on the progress or debug a program in Prolog I recommend using the “trace” and “spy” facilities. To find out about them look up the SWI Prolog manual.