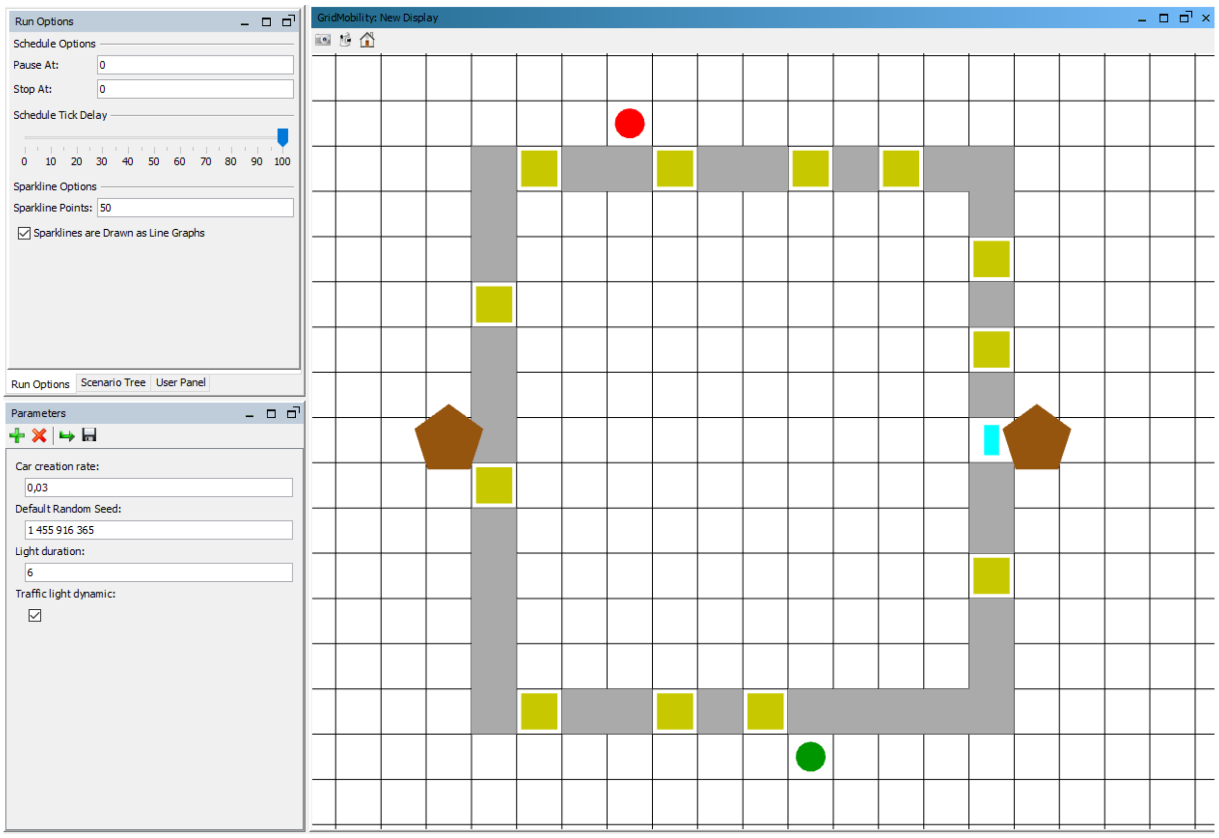
Report Multi-Agent system
Project 1: mobility simulation

By: Alexandre THEROND

Summary

How it works

My representation of movement simulation is as bellow:

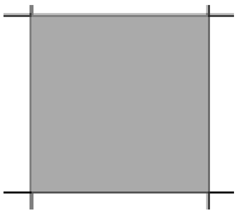


Schedule Options
Pause At: 0
Stop At: 0
Schedule Tick Delay
Sparkline Options
Sparkline Points: 50
☑ Sparklines are Drawn as Line Graphs

Parameters
Car creation rate: 0,03
Default Random Seed: 1 455 916 365
Light duration: 6
Traffic light dynamic: ☑

There are five different types of element that I will describe in the next parts.

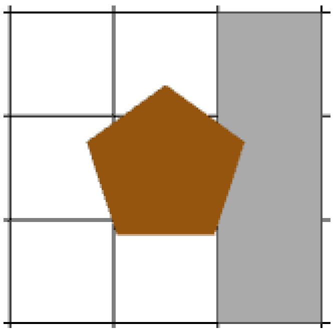
Road



The road element is represented by a grey square of size 15x15. 2 consecutive road elements form a possible path. The road elements are arrange at the start of the simulation to form a retangle path of size 11x12.

The different vehicles can only advance on a road element.

Bus Stop

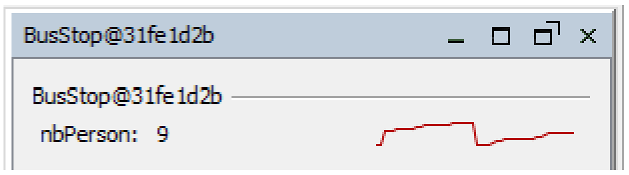


The bus stops are represented by a brown figure, created with the function createCircle and the values 12 and 5. As you can see on the figure, a bus stop is next to a road element.

At every tick of the simulation, a person can randomly be added to the bus stop. The amount of person currently waiting can be shown by double clicking on the bus stop during the simulation and look at the value "nbPerson".

Every time a bus pass at a bus stop, all the person waiting will go inside the bus and so the amount of person waiting will be reset to 0.

As we can see on the screenshot below, the amount of people currently waiting is 9. Also, the line chart shows us the evolution of people waiting at the bus stop. We can see that at some point, after continuously increasing, the number of persons drop, resulting of the passage of a bus.

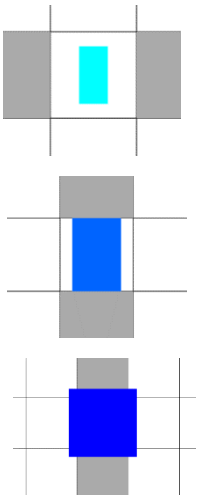


BusStop@31fe1d2b

BusStop@31fe1d2b

nbPerson: 9

Bus



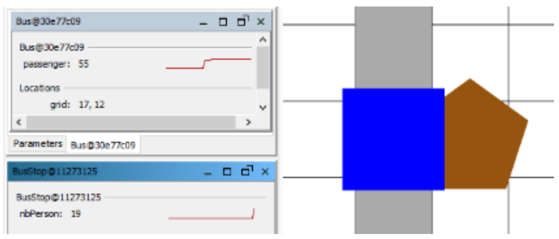
At the beginning of the simulation a bus is created on the road. It is defined by is number of passengers, which equals 0 at the start. There are 3 forms defining a bus depending on its passengers' number. Below ten passengers, the bus is a little rectangle of size 5x10 and color cyan. Between 10 and 24 passengers, the bus as a medium size (10x15) and color light blue. Above 25 passengers, the bus is overcrowded. Its size is 20x20 and its color is blue.

The number of current passengers on a bus can be desplayed by clicking on it.

To gain and loose passenger, a bus should stop at a bus stop. Most of the time, the stop will be short, and the bus will just continue directly. But it may arrive that a lot of people decide to descend off the bus, and so the bus will stop to as many ticks that people will want to get off. The number of people getting off is random but if a lot of people wants to get off at the same time, the bus will tend to keep its position and not move, and so, more people will have the opportunity to get off at the next tick.

Example scenario of a bus at a bus stop:

1. A bus arrive with 55 passengers at a bus stop containing 19 persons.



Bus@30e77c09

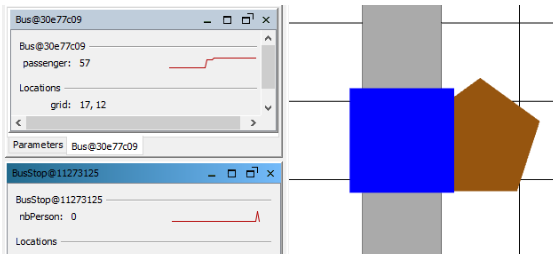
Bus@30e77c09
passenger: 55
Locations
grid: 17, 12

Parameters Bus@30e77c09

BusStop@11273125

BusStop@11273125
nbPerson: 19

2. The 19 persons goes inside the bus and some people get off. Now the bus have 57 passengers and the bus stop 0.



Bus@30e77c09

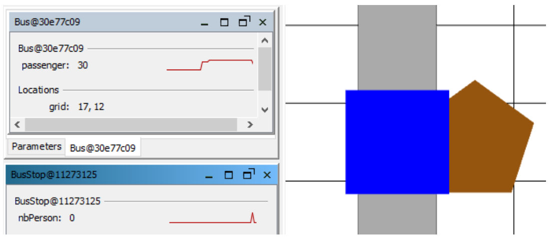
Bus@30e77c09
passenger: 57
Locations
grid: 17, 12

Parameters Bus@30e77c09

BusStop@11273125

BusStop@11273125
nbPerson: 0
Locations

3. People continue to get off the bus, having now 30 passengers.



Bus@30e77c09

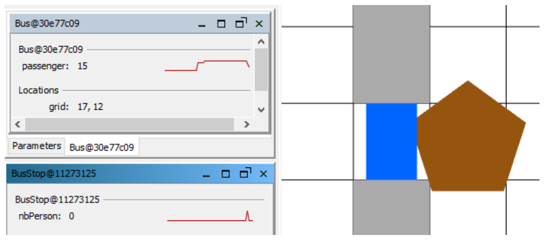
Bus@30e77c09
passenger: 30
Locations
grid: 17, 12

Parameters Bus@30e77c09

BusStop@11273125

BusStop@11273125
nbPerson: 0

4. As people continue to get off, the number of passengers pass under 25 and so the bus change its shape and color.



Bus@30e77c09

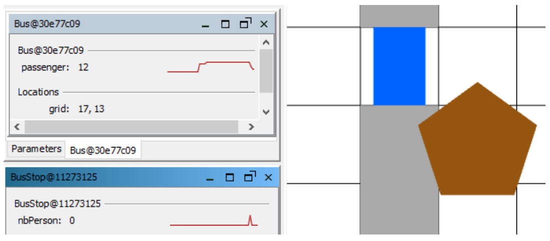
Bus@30e77c09
passenger: 15
Locations
grid: 17, 12

Parameters Bus@30e77c09

BusStop@11273125

BusStop@11273125
nbPerson: 0

5. Finally, the bus continue its road with 12 passengers.



Bus@30e77c09

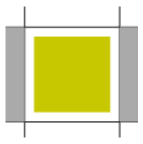
Bus@30e77c09
passenger: 12
Locations
grid: 17, 13

Parameters Bus@30e77c09

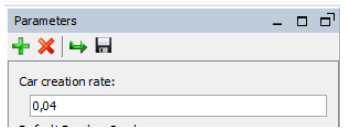
BusStop@11273125

BusStop@11273125
nbPerson: 0

Car



Randomly, at every tick of the simulation, a car can be generated, having a destination. A car can only be created on an empty road element. The car agent allows me to simulate a random movement of travellers around the neighbourhood.

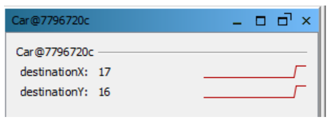


Parameters

Car creation rate:

0,04

It is represented by a yellow square of size 12. The creation rate of cars can be defined in the properties under the Car creation rate parameter. A value under 0.05 create a good simulation representation. A car can only be created if the road in front and behind it are empty, with no vehicles.



Car@7796720c

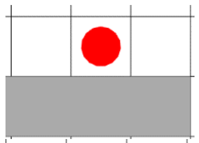
Car@7796720c

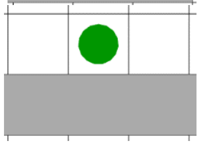
destinationX: 17

destinationY: 16

By double clicking on a car, we can see its destination coordinates. Once it arrive at destination, the car vanish, simulating the fact that the car park, and so, is no more moving.

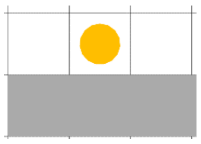
Traffic lights





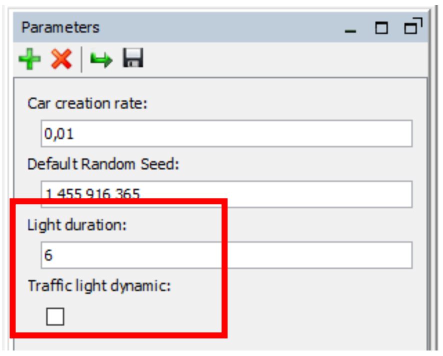
A traffic light is represented by a circle and a color and is always at the right side of the road.

If the traffic light is red, that means that no vehicle in front of the light can move. If the color is yellow, no vehicle can move also. But if the traffic light color is green, vehicles can continue.



It exist two type of traffic lights, the dynamic and the non-dynamic one.

By default, all trafic lights are non-dynamic. Their default duration value is indicated in the parameters of the simulation. If the default duration is of 6, this means that there will be 6 ticks with green light, 1 with yellow and 5 with a red light.



Parameters

Car creation rate:

0,01

Default Random Seed:

1 455 916 365

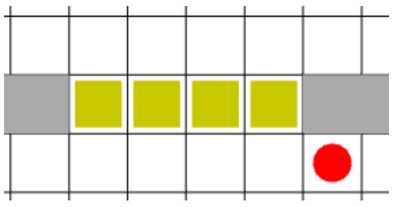
Light duration:

6

Traffic light dynamic:

☐

So, this means that there is probably a lot of car that can wait their turn in front of the traffic light:



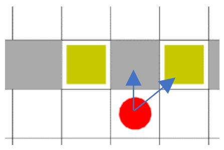
Here, 4 cars are waiting in front of a red light.

We can choose to tick the Traffic light dynamic option to change the traffic light strategy. By doing so, the light duration parameter is no longer use and changing it will not affect the simulation until the dynamic option is unchecked.

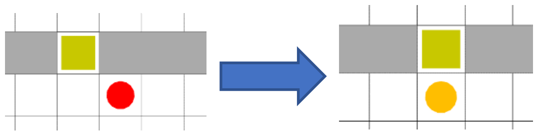
Once the dynamic option is ticked, the light will progressively turn red and remain red.

Different scenarios are then checked before turning the light green again. The examples below are with a sense of direction of vehicles from left to right.

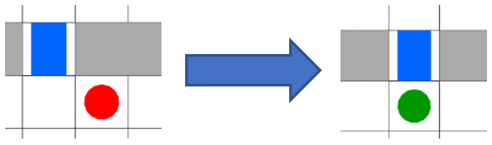
1. There is a vehicle right after the light or next to the light, the traffic light remains red to let the vehicle unclog the road. These two cells are represented with the blue arrows on the example.



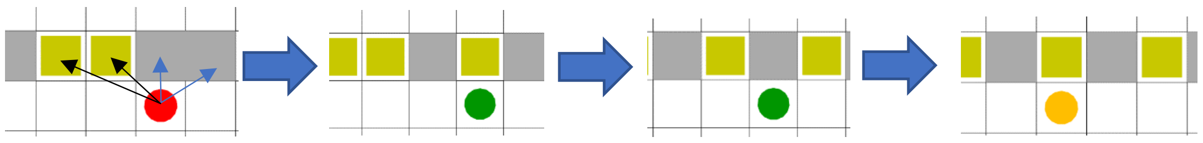
2. A car is waiting at a red light and there is no car in front of him nor behind him for at least 2 ticks, then the light allows the car to pass.



3. There is a bus in front of a red light. The traffic light turns automatically green for at least two ticks.



4. There are at least two vehicles waiting in front of the traffic light and the road after the light is unclog, then the traffic light turns green. The traffic light analyses if there are no car in front of him (blue arrows) and verify that there are two cars waiting (black arrows).



Difficulties encountered

The coordinates of the grid are inverted. When double clicking on an element on the grid, the first coordinate display is the column, the second one corresponds to the line number. Its not much but it takes some time to jump from the (line, column) way of operating to (column, line).

As I simulate the movement of travellers by random apparition at bus stop and on the road, taking their car, this design doesn't allow travellers going by foot to a specific destination.

To improve my code, I should add to each road the direction of which vehicle should go. Indeed, some part of my code are built only for the road created, changing the coordinates of the road break the simulation. Because of this I had to stay with the road I had already created, having no time to redesign all my project.

So, some part of the design choice I have made are good, as it had done the work, but choose straight from the beginning a little different design would have helped me going further in the simulation.

Finally, I miss a good documentation of Repast Simphony. As my code was built around a specific road, I knew at which location I can have a road and where I can't. I looked for a function returning the object at a specific location, but I didn't find it (I was looking for the inverse of the getLocation(Object) function). Also, I assign specific design to each element of my simulation. But I didn't find any documentation about which shapeFactory2D element already existed and how to create it, as the Repast Simphony Development Library does not allow modifications, I couldn't see the implementation.

User manual

Requirements

You will need:

- Repast Simphony installed.
- Java 11 installed.

Download project from: https://github.com/AlexandreTherond/Multi-Agent_System/

Create project

1. File -> New -> Repast Simphony Project:
   a. Give the name “Mobility” (without “”) -> next
   b. Libraries -> Add External JARs -> Path/to/java11 -> select java11
   c. Finish

2. Under /src/ create 2 Packages:
   a. mobility
   b. styles

3. Under src/mobility/ create 6 java classes and copy paste their code taken from git:
   a. Bus.java
   b. BusStop.java
   c. Car.java
   d. GridMobility.java
   e. Lights.java
   f. Road.java

4. Under src/styles/ create 5 java classes and copy paste their code taken from git:
   a. BusStyle.java
   b. BusStopStyle.java
   c. CarStyle.java
   d. LightsStyle.java
   e. RoadStyle.java

5. Replace Mobility.rs/parameters.xml and context.xml by the git files

6. Run Mobility Model

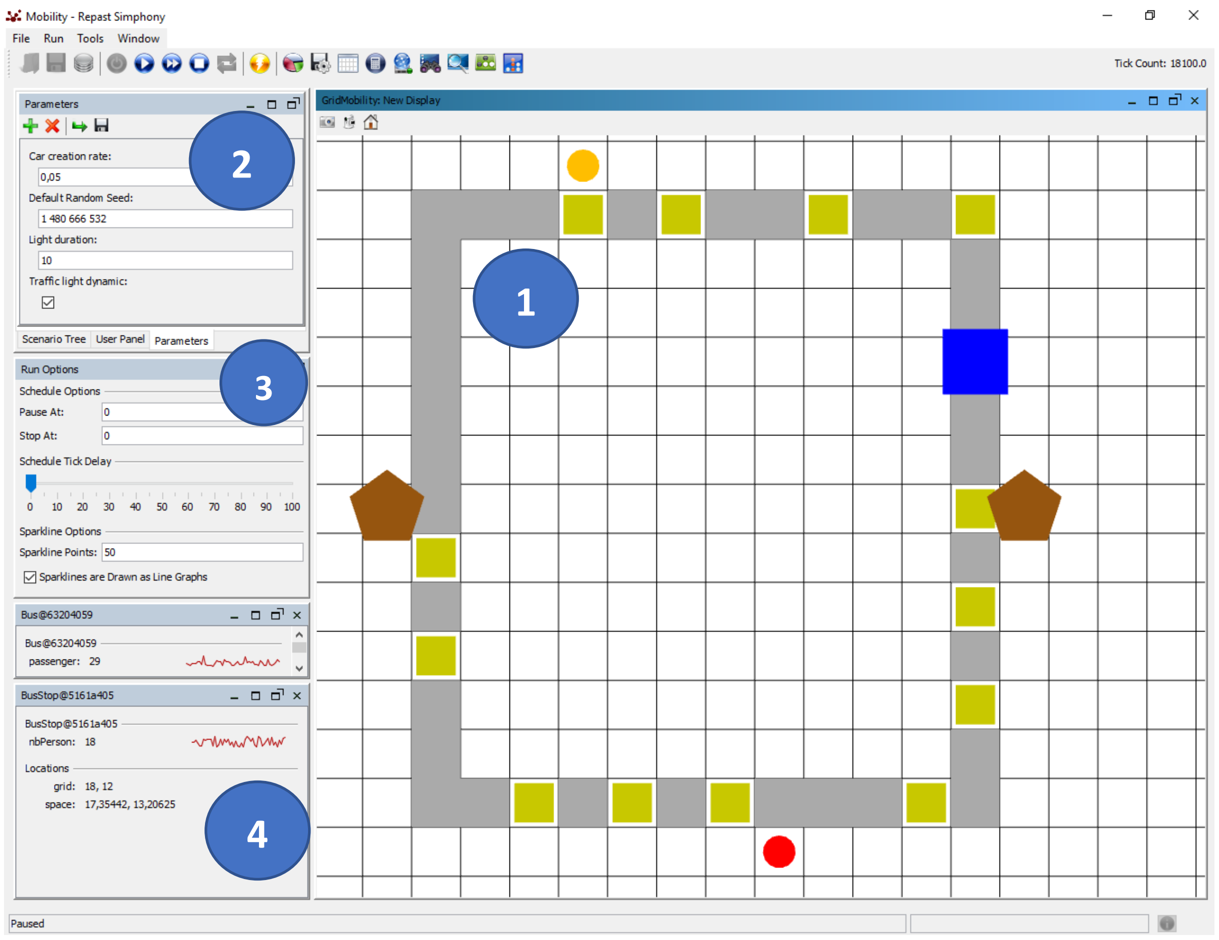
7. Data Loader -> set Data Loader -> Custom ContextBuilder Implementation -> Mobility.GridMobility -> Finished

8. Displays -> Add displays -> select grid -> next:
    a. Select Bus, BusStop, Car, Lights and Road -> next
    b. Select styles.BusStyles for bus, styles.CarStyles for car, etc. -> next
    c. Next -> Finish

9. Car creation rate, Light duration and Traffic light dynamic should appear under Parameters tab with default values

10. Run the program!

Use the program



Tick Count: 18100.0

Car creation rate: 0,05
Default Random Seed: 1 480 666 532
Light duration: 10
Traffic light dynamic: ☑

Pause At: 0
Stop At: 0
Sparkline Points: 50
☑ Sparklines are Drawn as Line Graphs

Bus@63204059
Bus@63204059
passenger: 29

BusStop@5161a405
BusStop@5161a405
nbPerson: 18
Locations
grid: 18, 12
space: 17,35442, 13,20625

Paused

1. The representation of the simulation with the different actors. By double clicking on one element, its characteristic will appear in the area of point 4

2. Parameter tab. We can change the car probability apparition, duration of lights and choose to have a dynamic light environment.

3. Run options tab, mostly used to change the tick delay.

4. Agents characteristics. We can see their names, grid location, and some variables as well as their evolution.