Traffic Monitor Simulation Implementation by Using Lopez

Shichao Guan

School of Information Technology and Engineering

University of Ottawa

800 King Edward Avenue, Ottawa, Ontario, Canada, K1N 6N5

sguan049@uottawa.ca

Keywords: Traffic, Lopez, Cell-DEVS, RESTful CD++.

Abstract

Restful CD++ middleware provides a web container into which Lopez is able to be plugged as a web service. Users then can use httprequest and httprespond to call the simulator and get the corresponding simulation result. The web container is responsible for parsing an XML file included in the httprequest. The XML file defines how models assigned to relevant processors.

The real system which is focused on is the traffic flow on a one-way two-lane highway. The purpose of this modeling is to reduce traffic accident and improve efficiency by creating Traffic Monitor to detect unusual events.

# Intruductoin

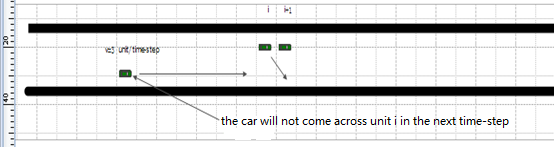
Each lane will be divided into a set of units. The size of one unit is the average length of a vehicle. Units may either be empty or be occupied by one vehicle.

If within 2 time-steps, one unit i keeps being occupied, integer value -1 will be sent to Traffic Monitor from unit i. Traffic Monitor will see whether it is an accident or congestion by the approach below:

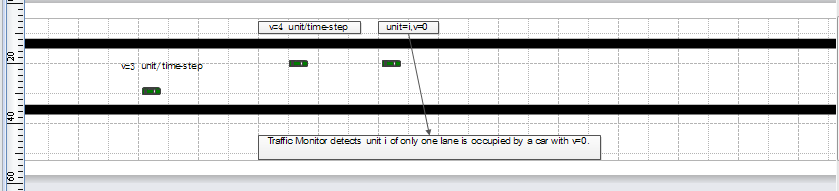
1. If Traffic Monitor receives the value -1 from unit i of both lanes at the same time-step, it is congestion.
2. If Traffic Monitor does only receive value -1 from unit i of only one lane at the time, it is an accident.

In terms of the vehicles, each car has an integer velocity with values between zero and Vmax (as 5 units per time-step). Assuming that the average car length is 7.5m and the time-step is 1 second, the Vmax is approximately 135km/h. The model will contain four behaviors as following:

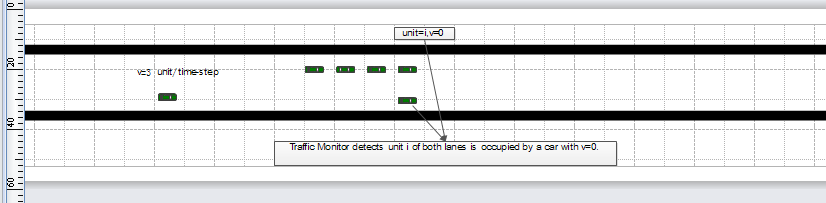
1. Acceleration: if the velocity v of a vehicle is lower than Vmax and the distance to the next car ahead is larger than v+1, the speed is advanced by 1 unit/per time-step (v to v+1).
2. Slowing down: if a vehicle at unit i sees the next vehicle at unit i+j(with j<=v),it reduces its speed to j-1 unit/per time-step(v to j-1).
3. Changing lane: if a vehicle at unit i sees the next unit i+1 is occupied by another stopped vehicle(this may due to congestion or an accident ), it will try to change to the other lane’s unit i+1 in the next time-step (with v=1) if that unit is empty and no vehicles may be able to come across the unit i+1 on the other line during the next time-step:



**Fig 1: Car at unit i changes lane.**

1. Traffic Monitor: if an accident occurs or congestion happens, Traffic Monitor is able to see the information by detecting the value from that exact unit.

**Fig 2: An accident occurs. Traffic Monitor detects the unit i where car’s v equals zero (one lane).**



**Fig 3: Congestion occurs. Traffic Monitor detects the unit i where car’s v equals zero (both lanes).**

1. Acceleration with different accelerated speed: Each car will be set with different accelerated speed. When cars begin to move and there are no cars in front of them in the next time step, the cars can move one unit for two based on their own accelerated speed.

# Backup environment and its DEPLOYMENT

We can use our lab server to do the simulation or, as an alternative way, we can build our own server.

The deployment of the environment is to integrate the Linux environment with Lopez and RISE. To achieve this, many support software s are required, including Tomcat, JDK, GCC, Netbeans, ect..

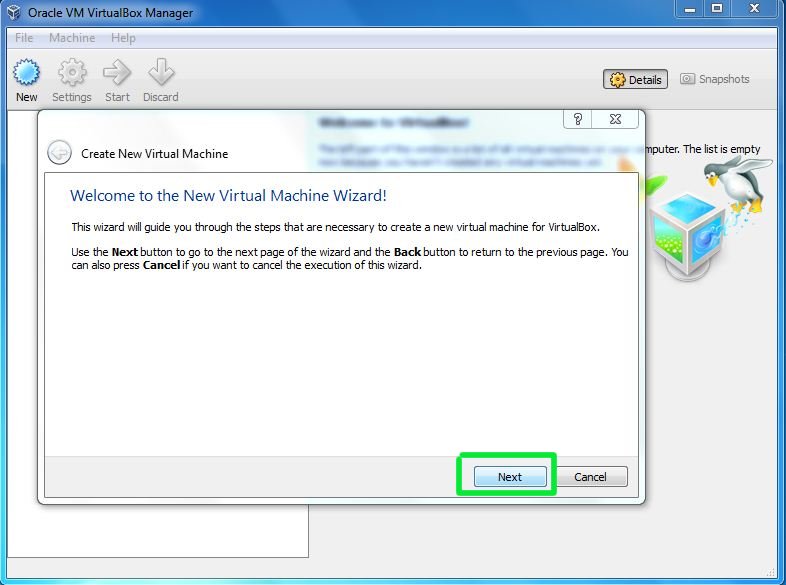
## Deployment of Linux

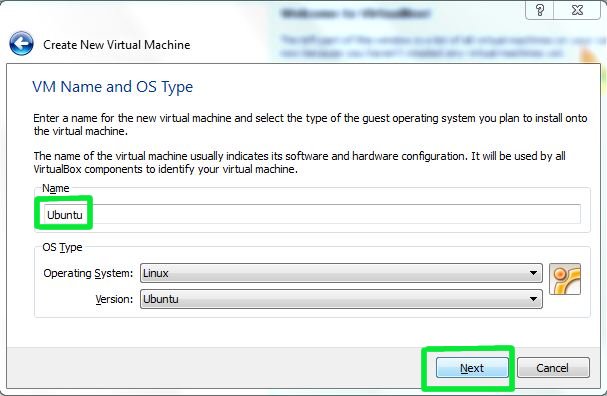
As the most successful open source, Linux is used in our project. Linux is free and performs better than Windows in terms of speed.

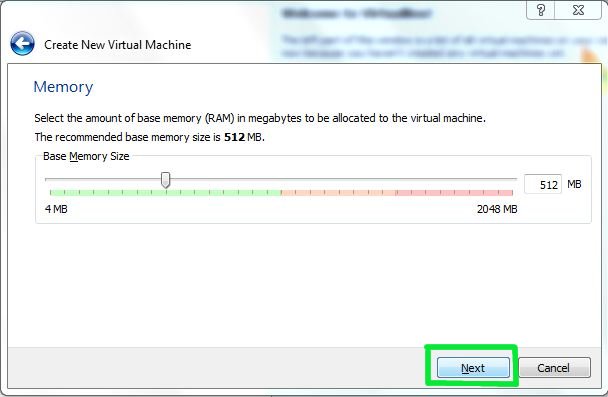
The deployment of Linux is able to be done in multiple ways. CD or USB setup can be used to setup the new operation system like Windows. Or, an alternative way to do this is to deployment a virtual machine and setup Linux on it. The second way enables us to exchange operation systems more flexibly which can significantly improve the efficiency. But on the other hand, using virtual machine will introduce more computer workload so it is not a good option for low-level computers.

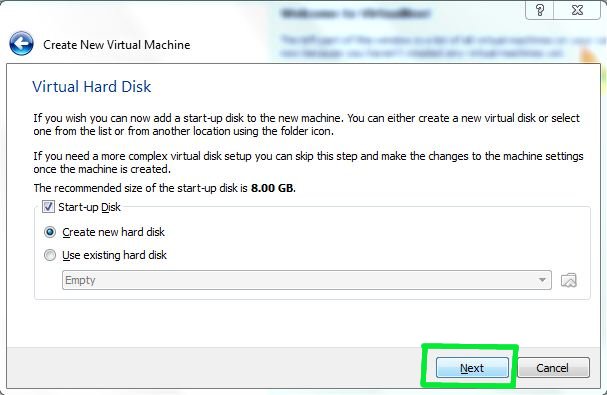
In my configuration, I used the VirtualBox as the VM software and deployed Ubuntu as the Linux system. The steps are as follows:

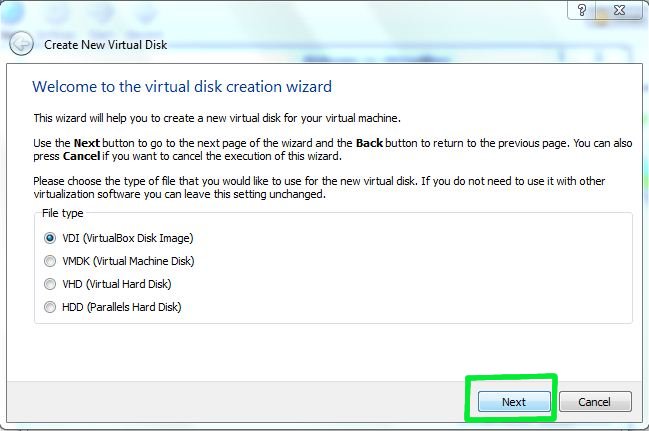
1. Download VirtualBox as an normal windows software and install it as suggested.
2. Download the Ubuntu Disk image -- the.iso file.
3. Create new Virtual Machine. Delegate RAM, create virtual Disk, create virtual hard drive, and install the Ubuntu Disk image boot on the hard drive.

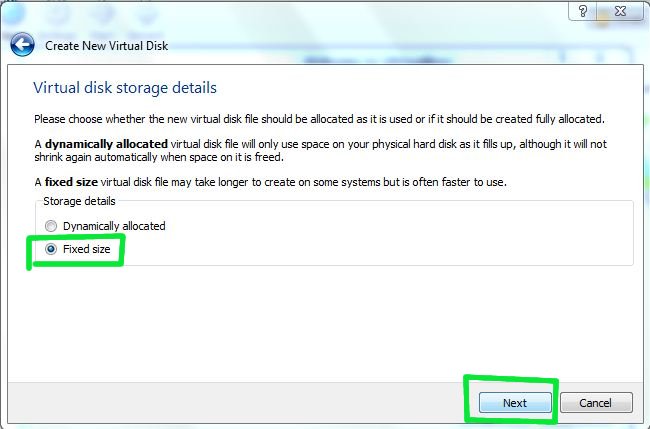


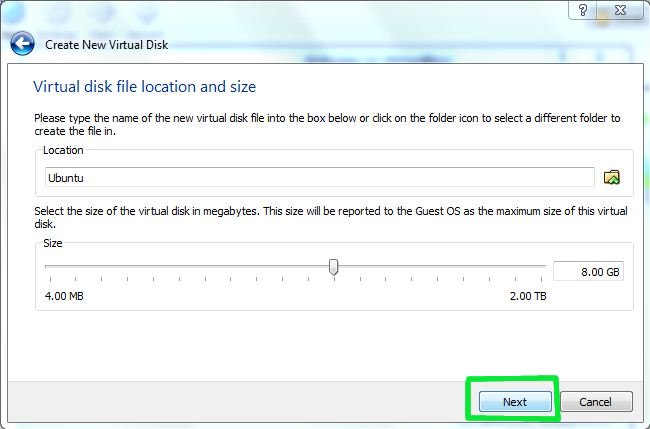


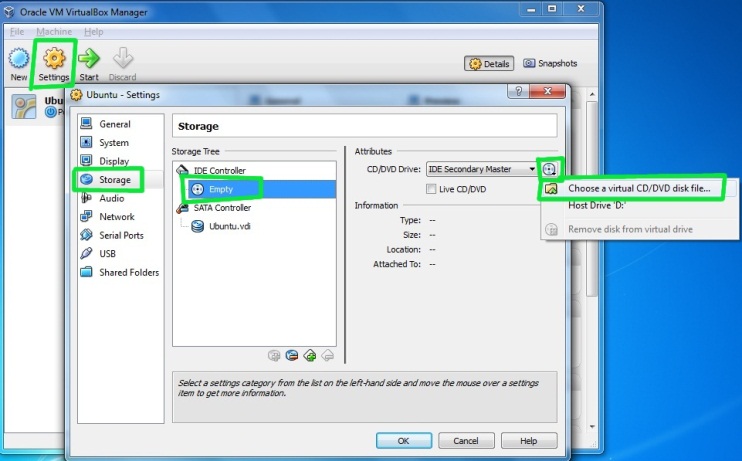






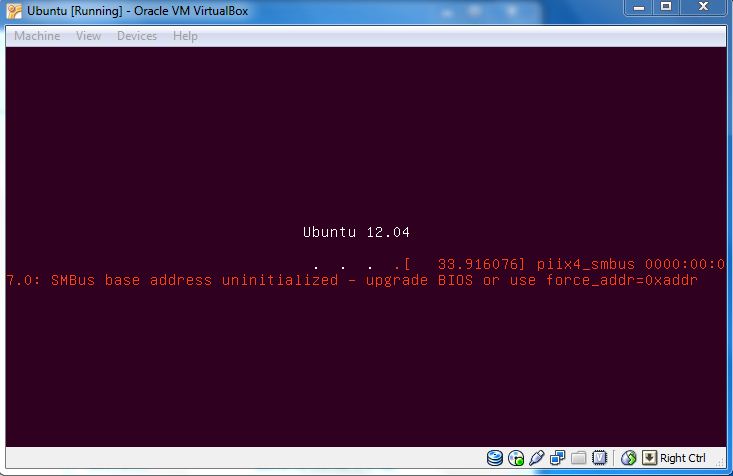


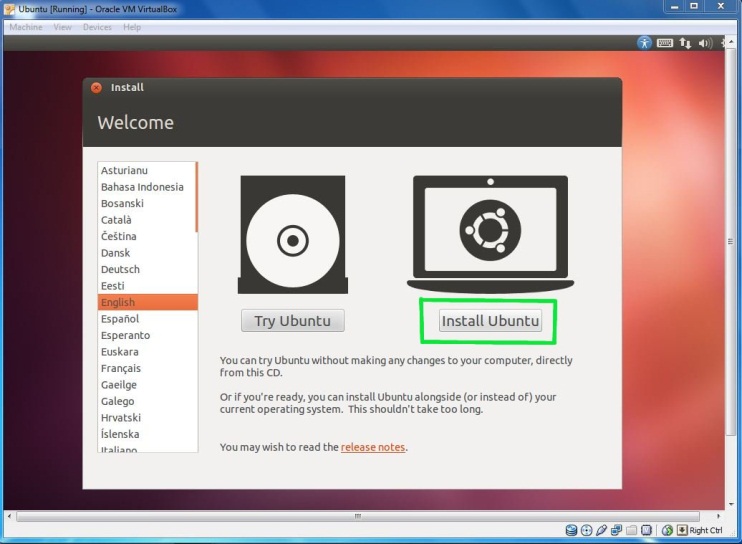




**Fig 4: Install Linux**

1. Wait for Ubuntu to boot up





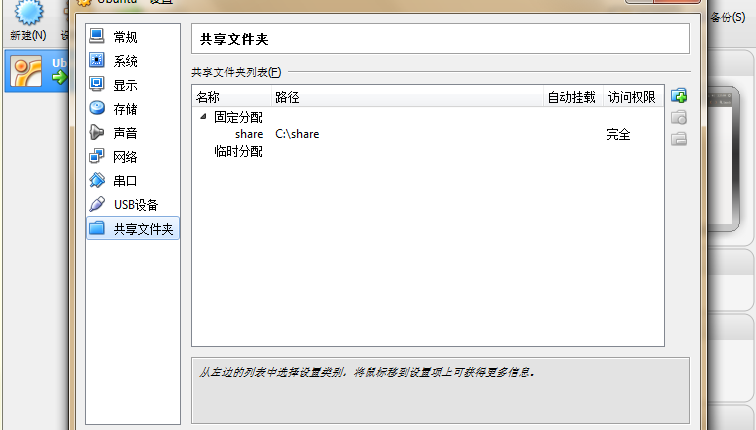
**Fig 5 Install Linux**

1. According to the guidance to finish the remaining setup.

When all above are done, we can go to the Linux system by start VirtualBox’s virtual machine.

In order to exchange files between different systems, one ‘share’ file is needed to be defined. The steps are as follow:

1. Design a share file in Windows: Setting button-->Share file-->new. In the pop-up dialog, input share file name and path

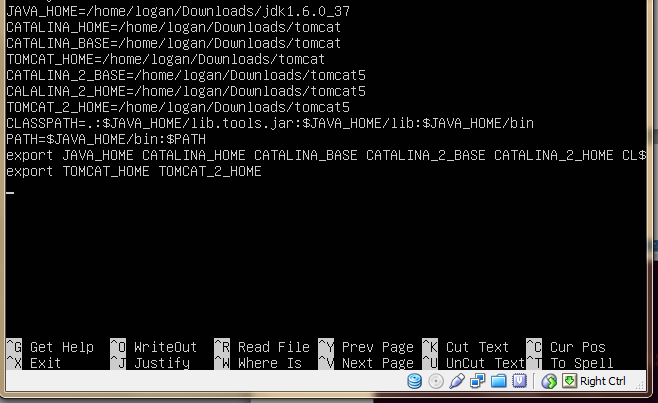


**Fig 6 Setup share point**

1. In Ubuntu, set the user as a manager: sudo su. Then create an corresponding share file space: mount –t vboxsf <share file name in windows> <share file path and name in Linux>. The default path and name in Linux is /mnt.

## Install JDK

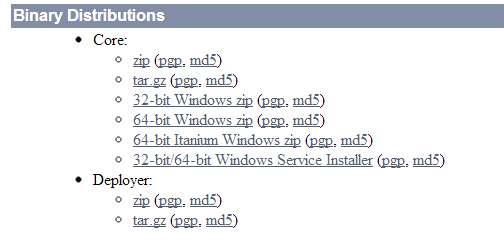
Because of some specific reasons, it is impossible for us to download JDK by using the commond ‘apt –get install’. It is , instead, needed to go to <http://www.oracle.com/> for installation. The steps are simple. First, we need to download the jdkxx.bin, then run it in Linux. After the installation, some environment valuables are required to be set. These valuables are located in /etc/profile. We can use ‘nano’ tool to edit this file by the command—nano /etc/profile. Valuables should be export in the end of the file as below: (This also includes some other configurations I added for tomcat later, only JAVA\_HOME, CLASSPATH, PATH are related to JDK)



**Fig 7 Configure environment vals for JDK**

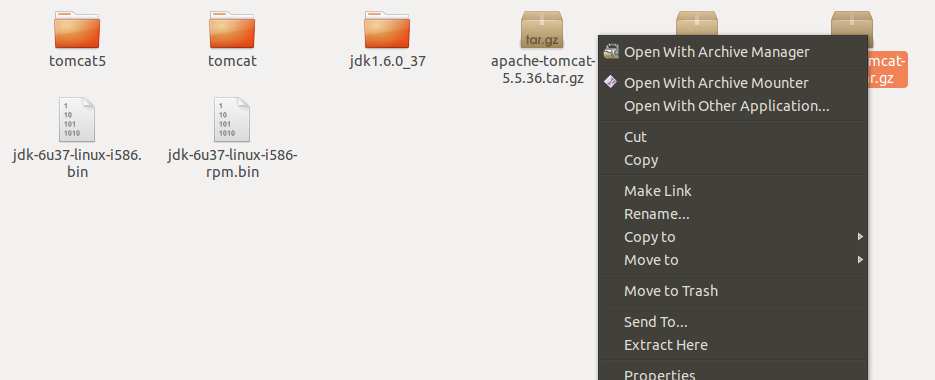
## Install Tomcat

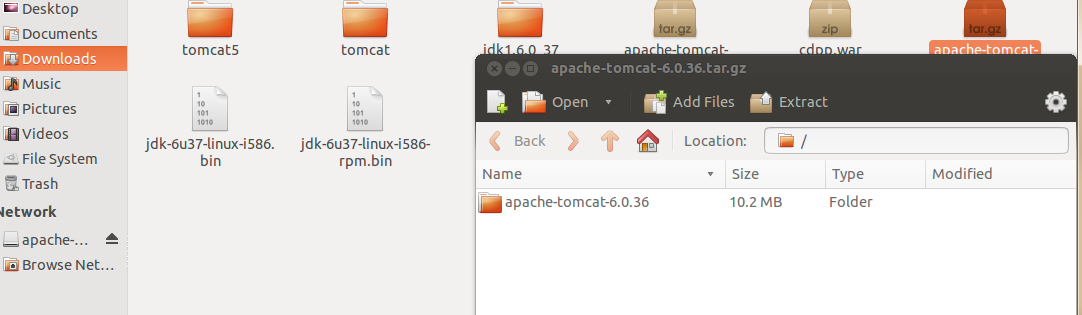
For some versions of Linux, tomcat is setup automatically during the installation of Linux. To setup a specific version of Tomcat (like Tomcat 6.0), it is still needed for us so far to install it from its website: <http://tomcat.apache.org>. First, we download the tar.gz from the website as below:



**Fig 8 Install Tomcat**

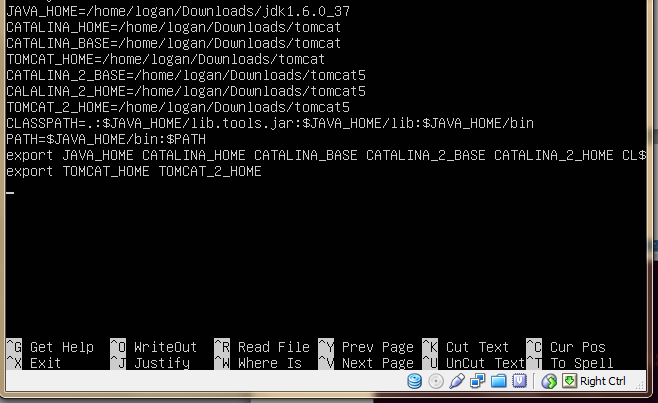
Then, open it by using ‘ArchiveMounter’ tool and copy the ‘tomcat’ file out:





**Fig 9 Install Tomcat**

Also, we can simply change the file name to ‘tomcat’ so that it will be easier for us to set the environment valuables. These valuables are also located in the file /etc/profile. The sets are as below: (Only CATALINA\_HOME, CATALINA\_BASE, TOMCAT\_HOME are related to tomcat)



**Fig 10 Setup Environment Vals for Tomcat**

In addition, if more than one tomcat is needed, more configurations are required:

1. Each tomcat should have its own valuables in the /etc/profile including CATALINA\_2\_HOME, CATALINA\_2\_BASE, TOMCAT\_2\_HOME.
2. In order to avoid the ports’ conflict between different tomcats, file server.xml needs to be modified. This file is located in /conf. The changes are as follow:

<Server port="9005" shutdown="SHUTDOWN">

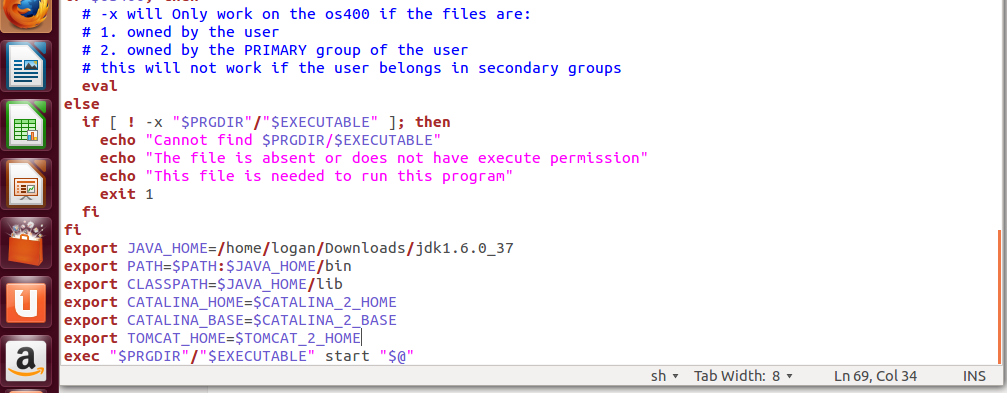
<!--port:8005->9005-->  
<!-- Define a non-SSL HTTP/1.1 Connector on port 8080 -->  
    <Connector port="9080" maxHttpHeaderSize="8192"

<!--port:8080->9080-->maxThreads="150" minSpareThreads="25" maxSpareThreads="75"  
       enableLookups="false" redirectPort="8443" acceptCount="100"  
      connectionTimeout="20000" disableUploadTimeout="true" />  
<!-- Define an AJP 1.3 Connector on port 8009 -->  
    <Connector port="9009"

<!--port:8005->9005-->  
enableLookups="false" redirectPort="8443" protocol="AJP/1.3" />

These modifications exchange port 8080, 8005, 8009 to other ports in the second tomcat to avoid conflict. Port 8443 is not necessary to be modified because we do not use SSL (https) for our project. If we need to use SSL in another project, then port 8443 should also be modified accordingly.

1. For the second tomcat, we need also change the startup.sh and shutdown.sh in /bin. This is because for the second tomcat, we use CATALINA\_2\_HOME, CATALINA\_2\_BASE, TOMCAT\_2\_HOME instead of the original ones. It is required to export these changes in both of the files.

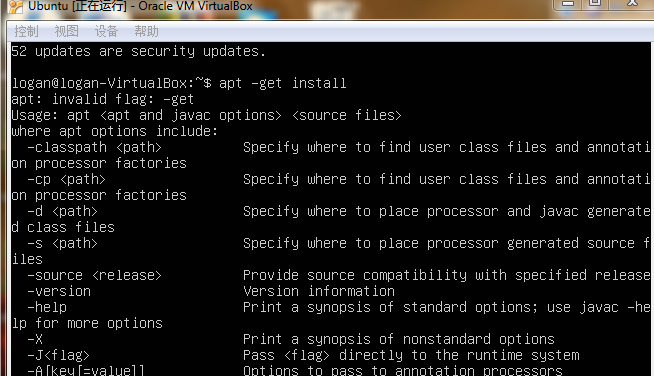


**Fig 11 Configure the second Tomcat**

1. After these changes, we can test if our tomcat is ready to use by the command ./startup.sh in different $TOMCAT\_HOME/bin files. The result shown below indicates that our configuration is OK.

## Deploy GCC

This step is relevant simple. By using the command apt –get install, we can install GCC automatically as follow:

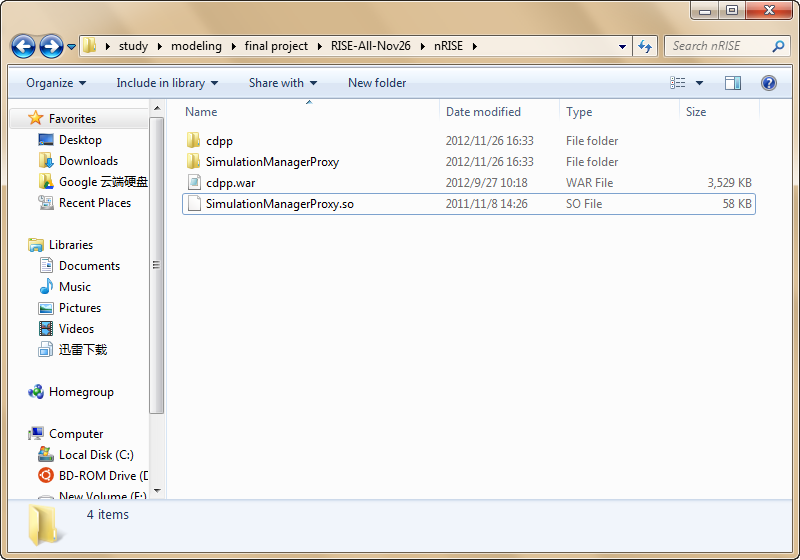


**Fig 12 Install GCC**

## Compile SimulationManagerProxy and cdpp

The next step is to compile SimulationManagerProxy and cdpp by GCC. Some errors may occur during the compilation. Based on the suggestion, several software are also requested to be installed. In addition, as to GCC 4.7, our code should be modified because of syntax change. For example, as to file ‘make file’, we need to change the line as

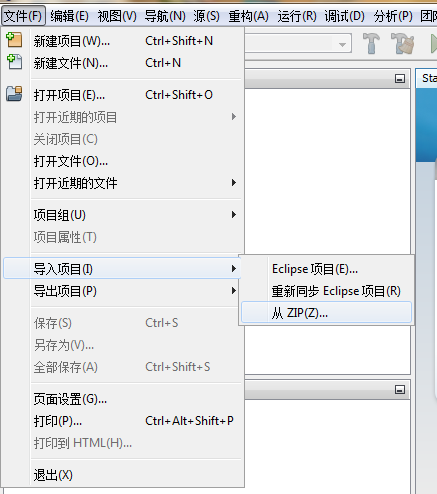
‘simu: ${ALLOBJS} libsimu.a ${CPP} ${LDFLAGS} -o $@ -lpthread ${EXAMPLESOBJS} ${LIBS}’. When these two files are successfully compiled, the last step is to copy file ‘SimulationManagerProxy.so’ out to the path as follow:



**Fig 13 File path of cdpp and proxy**

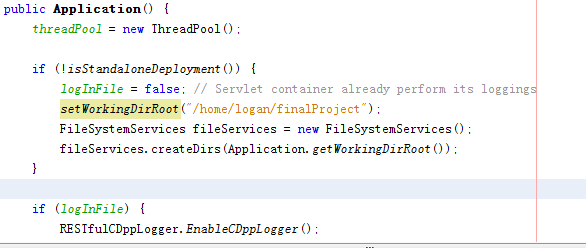
## Compile RESTful\_CDpp and cdpp by Netbeans

The first step is to import the files:



**Fig 14 Using Netbeans to compile cdpp and RESTful\_CDpp**

Then, it is needed to compile project RESTful\_CDpp and cdpp respectively. As to project RESTful\_CDpp, we have to set the Working dictionary root path for it (/home/logan/finalProject):

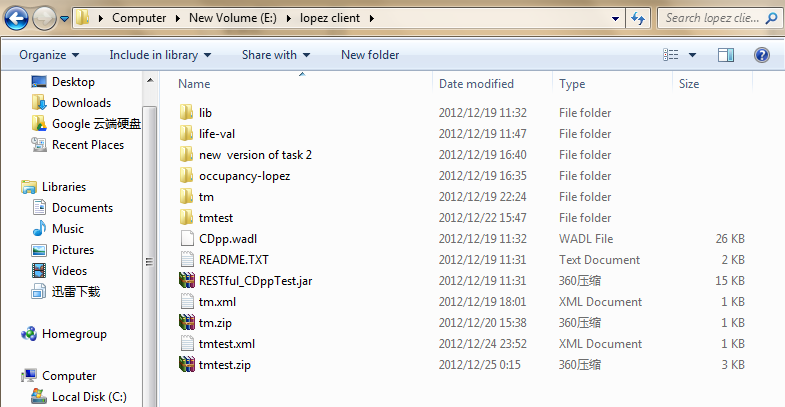


**Fig 15 Configure file path**

After compilation, we can get the .war file in the path /cdpp/dist. In the Linux system, we should add this .war file to Tomcat so that this service will be available to other users.

## Compile RESTful\_CDppTest

RESTful\_CDppTest is a client service to give users access to our web service. In this file, if we want to use the lab server, we should choose the URI as APPLICATION\_URI = <http://134.117.53.66:8080/cdpp>. On the other hand, if we want to choose our own computer as the server, we can set the URI as the other one: APPLICATION\_URI = <http://localhost:8080/cdpp>. After the setting, this file should also be compiled. The .war and the lib file should be stored in the same path. Here is an example:



**Fig 16 File path of client service**

After all the steps above, we can successfully build our own service and begin to use Lopez for our simulation.

# Model defination

In this chapter, I will introduce the design of the model. First of all, I will show some general information about the model then explain the rules of it.

## General Information

Based on the .ma file, basic information about the model is shown as below:

type : cell

dim : (2,40)

delay : transport

defaultDelayTime : 1000

border : wrapped

neighbors : tmtest(0,-5) tmtest(0,-4)

neighbors : tmtest(0,-3) tmtest(0,-2)

neighbors : tmtest(0,-1) tmtest(0,0)

neighbors : tmtest(0,1) tmtest(0,2)

neighbors : tmtest(0,3) tmtest(0,4)

neighbors : tmtest(1,-5) tmtest(1,-4)

neighbors : tmtest(1,-3) tmtest(1,-2)

neighbors : tmtset(1,-1) tmtest(1,0)

neighbors : tmtest(1,1)

initialvalue : 0

localtransition : move-rule

stateVariables : velocity init acce

stateValues : 9 0 0

initialvariablesvalue : tmtest.stvalues

neighborports: warning movement accele

As it can be seen that our model’s type is Cell-DEVS and it is a two dimension model with length of 40 and width of 2. The delay type is transport delay. The border is set as wrapped because our model is a one-way two lane highway and turn left is the same as turn right. It is like in the binary system, add one equals minus one. As to the neighbors, it contains seventeen neighbors: five behind of it in the same lane, itself, four ahead of it in the same lane and another seven in the other lane.

The model contains three state valuables: init, velocity and acce. Valuable init is designed to initialize the other two valuables. Velocity stands for the current speed of the car from zero to five. If the unit is not occupied by a car, the velocity of that unit is designed to be nine. As to acce, it is the accelerated speed of the car. The value of acce is one or two, which means a car can move one or two units when it begins to move from speed zero.

In addition, we have three neighbourports: movement, warning and accele. Movement mainly focuses on the speed change of each units. Warning is to show the situation when there are cars with speed zero (congestion or accident). Accele is to describe the accelerated speed change of each unit.

## Rules

rule : {1} {$velocity := $velocity;$init := $init;$acce := $acce;} 0 { (0,0) = 0 }

This rule is to initialize the state value.

rule : {~movement := $velocity;~accele := $acce;} {$init := -1;} 100 {$init = 0}

This rule is to initialize ports value.

rule : {~movement := 5;~warning := 2;~accele := (0,-5)~accele;} {$velocity := 5;} 1000 { $velocity = 9 and (0,-1)~movement = 9 and (0,-2)~movement = 9 and (0,-3)~movement = 9 and (0,-4)~movement = 9 and (0,-5)~movement = 5}

This rule is to set the speed of the unit as five. In this situation, the car’s speed is five and it keeps the speed because there are no other cars in front of it in the next five units.

rule : {~movement := 5;~warning := 2;~accele := (0,-5)~accele;} {$velocity := 5;} 1000 { $velocity = 9 and (0,-1)~movement = 9 and (0,-2)~movement = 9 and (0,-3)~movement = 9 and (0,-4)~movement = 9 and (0,-5)~movement = 4}

This rule is to set the speed of the unit as five. In this situation, the car’s speed is four and it speeds up because there are no other cars in front of it in the next five units.

rule : {~movement := 4;~warning := 2;~accele := (0,-4)~accele;} {$velocity := 4;} 1000 { $velocity = 9 and (0,-1)~movement = 9 and (0,-2)~movement = 9 and (0,-3)~movement = 9 and (0,-4)~movement = 4 and (0,1)~movement != 9}

This rule is to set the speed of the unit as four. In this situation, the car’s speed is four and it keeps the speed because there are no other cars in front of it in the next four units but there is a car in the fifth unit ahead of it.

rule : {~movement := 4;~warning := 2;~accele := (0,-4)~accele;} {$velocity := 4;} 1000 { $velocity = 9 and (0,-1)~movement = 9 and (0,-2)~movement = 9 and (0,-3)~movement = 9 and (0,-4)~movement > 4 and (0,-4)~movement != 9 and (0,1)~movement != 9}

This rule is to set the speed of the unit as four. In this situation, the car’s speed is more than four and it slows down because there are no other cars in front of it in the next four units but there is a car in the fifth unit ahead of it.

rule : {~movement := 4;~warning := 2;~accele := (0,-4)~accele;} {$velocity := 4;} 1000 { $velocity = 9 and (0,-1)~movement = 9 and (0,-2)~movement = 9 and (0,-3)~movement = 9 and (0,-4)~movement = 3}

This rule is to set the speed of the unit as four. In this situation, the car’s speed is three and it speeds up because there are no other cars in front of it in the next four units.

rule : {~movement := 3;~warning := 2;~accele := (0,-3)~accele;} {$velocity := 3;} 1000 { (0,-1)~movement = 9 and (0,-2)~movement = 9 and (0,-3)~movement = 3 and $velocity = 9 and (0,1)~movement != 9}

This rule is to set the speed of the unit as three. In this situation, the car’s speed is three and it keeps the speed because there are no other cars in front of it in the next three units but there is a car in the fourth unit ahead of it.

rule : {~movement := 3;~warning := 2;~accele := (0,-3)~accele;} {$velocity := 3;} 1000 { (0,-1)~movement = 9 and (0,-2)~movement = 9 and (0,-3)~movement > 3 and (0,-3)~movement != 9 and $velocity = 9 and (0,1)~movement != 9}

This rule is to set the speed of the unit as three. In this situation, the car’s speed is more than three and it slows down because there are no other cars in front of it in the next three units but there is a car in the fourth unit ahead of it.

rule : {~movement := 3;~warning := 2;~accele := (0,-3)~accele;} {$velocity := 3;} 1000 { (0,-3)~movement = 2 and (0,-2)~movement = 9 and (0,-1)~movement = 9 and $velocity = 9}

This rule is to set the speed of the unit as three. In this situation, the car’s speed is two and it speeds up because there are no other cars in front of it in the next three units.

rule : {~movement := 2;~warning := 2;~accele := (0,-2)~accele;} {$velocity := 2;} 1000 { (0,-2)~movement = 1 and (0,-1)~movement = 9 and $velocity = 9}

This rule is to set the speed of the unit as two. In this situation, the car’s speed is one and it speeds up because there are no other cars in front of it in the next two units.

rule : {~movement := 2;~warning := 2;~accele := (0,-2)~accele;} {$velocity := 2;} 1000 { (0,-1)~movement = 9 and (0,-2)~movement = 2 and $velocity = 9 and (0,1)~movement != 9 }

This rule is to set the speed of the unit as two. In this situation, the car’s speed is two and it keeps the speed because there are no other cars in front of it in the next two units but there is a car in the third unit ahead of it.

rule : {~movement := 2;~warning := 2;~accele := (0,-2)~accele;} {$velocity := 2;} 1000 { (0,-1)~movement = 9 and (0,-2)~movement > 2 and (0,-2)~movement != 9 and (0,1)~movement != 9 and $velocity = 9}

This rule is to set the speed of the unit as two. In this situation, the car’s speed is more than two and it slows down because there are no other cars in front of it in the next two units but there is a car in the third unit ahead of it.

rule : {~movement := 1;~warning := 2;~accele := (0,-1)~accele;} {$velocity := 1;} 1000 { (0,-1)~movement = 1 and $velocity = 9 and (0,1)~movement != 9}

This rule is to set the speed of the unit as one. In this situation, the car’s speed is one and it keeps the speed because there are no other cars in front of it in the next unit but there is a car in the second unit ahead of it.

rule : {~movement := 1;~warning := 2;~accele := (0,-1)~accele;} {$velocity := 1;} 1000 { (0,-1)~movement > 1 and (0,-1)~movement != 9 and (0,1)~movement != 9 and $velocity = 9}

This rule is to set the speed of the unit as one. In this situation, the car’s speed is more than one and it slows down because there are no other cars in front of it in the next unit but there is a car in the second unit ahead of it.

rule : {~movement := 0;~warning := 1;~accele := (0,0)~accele;} {$velocity := 0;} 1000 { $velocity != 9 and (0,1)~movement != 9 and (0,1)~movement != 0}

This rule is to set the speed of the car as zero and output the warning as 1 because the car is stopped. The car’s speed is not zero but the next unit is occupied by a moving car at that time ahead of it.

rule : {~movement := 0;~warning := 1;~accele := (0,0)~accele;} {$velocity := 0;} 1000 { $velocity != 9 and (0,1)~movement = 0 and (1,1)~movement != 9}

This rule is to set the speed of the car as zero and output the warning as 1 because the car is stopped. The car’s speed is not zero but the next unit is occupied by a stopped car at that time ahead of it. The car cannot change lane because in unit (1,1) there is another car.

rule : {~movement := 0;~warning := 1;~accele := (0,0)~accele;} {$velocity := 0;} 1000 { $velocity != 9 and (0,1)~movement = 0 and (1,0)~movement != 9}

This rule is to set the speed of the car as zero and output the warning as 1 because the car is stopped. The car’s speed is not zero but the next unit is occupied by a stopped car at that time ahead of it. The car cannot change lane because in unit (1,0) there is another car.

rule : {~movement := 0;~warning := 1;~accele := (0,0)~accele;} {$velocity := 0;} 1000 { $velocity != 9 and (0,1)~movement = 0 and (1,-1)~movement != 9}

This rule is to set the speed of the car as zero and output the warning as 1 because the car is stopped. The car’s speed is not zero but the next unit is occupied by a stopped car at that time ahead of it. The car cannot change lane because in unit (1,-1) there is another car.

rule : {~movement := 0;~warning := 1;~accele := (0,0)~accele;} {$velocity := 0;} 1000 { $velocity != 9 and (0,1)~movement = 0 and (1,-2)~movement != 9}

This rule is to set the speed of the car as zero and output the warning as 1 because the car is stopped. The car’s speed is not zero but the next unit is occupied by a stopped car at that time ahead of it. The car cannot change lane because in unit (1,-2) there is another car.

rule : {~movement := 0;~warning := 1;~accele := (0,0)~accele;} {$velocity := 0;} 1000 { $velocity != 9 and (0,1)~movement = 0 and (1,-3)~movement != 9}

This rule is to set the speed of the car as zero and output the warning as 1 because the car is stopped. The car’s speed is not zero but the next unit is occupied by a stopped car at that time ahead of it. The car cannot change lane because in unit (1,-3) there is another car.

rule : {~movement := 0;~warning := 1;~accele := (0,0)~accele;} {$velocity := 0;} 1000 { $velocity != 9 and (0,1)~movement = 0 and (1,-4)~movement != 9}

This rule is to set the speed of the car as zero and output the warning as 1 because the car is stopped. The car’s speed is not zero but the next unit is occupied by a stopped car at that time ahead of it. The car cannot change lane because in unit (1,-4) there is another car.

rule : {~movement := 0;~warning := 1;~accele := (0,0)~accele;} {$velocity := 0;} 1000 { $velocity != 9 and (0,1)~movement = 0 and (1,-5)~movement != 9}

This rule is to set the speed of the car as zero and output the warning as 1 because the car is stopped. The car’s speed is not zero but the next unit is occupied by a stopped car at that time ahead of it. The car cannot change lane because in unit (1,-5) there is another car.

rule : {~movement := 1;~warning := 2;~accele := (0,-1)~accele;} {$velocity := 1;} 1000 { (0,-1)~movement = 0 and $velocity = 9 and ((0,1)~movement != 9 or (0,2)~movement != 9 or (0,3)~movement != 9 or (0,4)~movement != 9)}

This rule is to set the speed of the car as one. There is one car with speed zero. But in the next five units ,there is more than one car there. In this situation, the car can only move one unit no matter its accelerated speed is 1 or two.

rule : {~movement := $velocity;~warning := 2;~accele := (0,-2)~accele;} {$velocity := (0,-2)~accele;} 1000 { (0,-2)~movement = 0 and $velocity = 9 and (0,-1)~movement = 9 and (0,1)~movement = 9 and (0,2)~movement = 9 and (0,3)~movement = 9 }

This rule is to set the speed of the car as its accelerated speed. There are no cars in the next five units ahead of the car so when the car starts to move, it is able to move more than one units based on its own accelerated speed.

rule : {~movement := 1;~warning := 2;~accele := (1,-1)~accele;} {$velocity := 1;} 1000 { $velocity = 9 and (0,-1)~movement = 9 and (0,-2)~movement = 9 and (0,-3)~movement = 9 and (0,-4)~movement = 9 and (0,-5)~movement = 9 and (1,0)~movement = 0 and (1,-1)~movement != 9 }

This rule is to set the speed of the car as one. The rule describes the situation when cars change lane. When the car changes lane, its speed will be one.

rule : {~movement := 9;~warning := 2;~accele := 0;} {$velocity := 9;} 1000 { $velocity != 9 and (0,1)~movement = 9}

When the unit is not occupied by a car, it should be empty

rule : {~movement := 9;~warning := 2;~accele := 0;} {$velocity := 9;} 1000 { $velocity != 9 and (0,1)~movement = 0 and (1,0)~movement = 9 and (1,-1)~movement = 9 and (1,-2)~movement = 9 and (1,-3)~movement = 9 and (1,-4)~movement = 9 and (1,-5)~movement = 9 and (1,1)~movement = 9}

When the unit is not occupied by a car, it should be empty.

rule : {} 0 {t}

When nothing happens to the unit, it will do nothing.

## The XML File

In our model, because we need the speed zero, I set the ‘Notprintzero’ as false.

If we want to observe the movement of the cars, we can set the neighborports as movement; If we want to observe the warning message, we can set the neighborports as warning; if we need to see the accelerated speed of the cars, we can set the neighborports as accele.

# Test method and simulation results

In this chapter, I will introduce how the model is tested and show the corresponding result. The test will be set based on the models behavior.

## Acceleration

This behavior describes the car speeds up if its speed is below the maximum speed 5 units/time-step.

The result is shown as below:

Line : 1 - Time: 00:00:02:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 1 9 9 2 9 9 9 3 9 9 9 9 4 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:03:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 2 9 9 9 3 9 9 9 9 4 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:04:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 3 9 9 9 9 4 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:05:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 9 9 9 9 4 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:06:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

As it can be seen that, there are five cars in the beginning, when the time moves, they will speed up unit they get the maximum speed. The result is as expected.

## Keep the Speed

This behavior describes the car keeps its speed and moves on. The car cannot speed up because there is another car in front of it.

The result is shown as below:

Line : 1 - Time: 00:00:01:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 1 9 2 9 9 3 9 9 9 4 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:02:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 1 9 9 2 9 9 9 3 9 9 9 9 4 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:03:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 2 9 9 9 3 9 9 9 9 4 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:04:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 3 9 9 9 9 4 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:05:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 9 9 9 9 4 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:06:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

We can see the result that from time 1 second to time 2 second, the car moves on with its speed not changed. After that, the car speeds up and finally gets the maximum speed. The result is as expected.

## Slow down

Cars need to slow down to avoid collisions. The result is shown as follow:

Line : 1 - Time: 00:00:01:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 1 2 9 3 9 9 4 9 9 9 5 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:02:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 1 9 9 2 9 9 9 3 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:03:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 1 9 9 2 9 9 9 3 9 9 1 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 1 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:04:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 2 9 9 9 3 9 9 2 9 9 2 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 2 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:05:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 3 9 9 2 9 9 2 9 9 9 3 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 3 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:06:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 9 9 2 9 9 2 9 9 9 3 9 9 9 9 4 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 4 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:07:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 5 9 9 9 9 9 9 9 9 2 9 9 9 3 9 9 9 9 4 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:08:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 3 9 9 9 9 4 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:09:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 4 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 5 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:10:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9 9 9 5 9 9 9|

1| 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

In this case, cars will slow down first in the time from 1 second to 2 second, then, gradually speed up until all cars get the maximum speed at 10 second. The result is as expected.

## Change Iane

When there is a stopped car in the next unit, the car in the current unit will try to change lane in order to move on. The result is shown as below:

Line : 1 - Time: 00:00:01:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 1 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 2 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:02:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 1 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 1 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 1 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 3 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:03:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 2 9 9 9 9 9 9 9 9 9 9 9 9 9 9 1 9 9 2 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 2 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 4 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:04:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 9 3 9 9 9 9 9 9 9 9 9 9 9 9 9 2 9 9 9 3 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 3 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:05:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 9 9 9 9 9 4 9 9 9 9 9 9 9 9 9 9 9 9 3 9 9 9 9 4 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 4 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:06:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 4 9 9 9 9 9 5 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:07:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9|

1| 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Based on the result, we can see that from 1 second to 2 second, one car in the line 1 unit 3 changes to line two. After the change of line, the cars begin to speed up until get the maximum speed 5units/time-step. The result is expected.

## Start with Accelerated Speed

Cars are able to start to move with different accelerated speed. The accelerated speed differs from one to two, that is to say, some cars can move two units when they start to move. The result is show as follow:

Line : 1 - Time: 00:00:01:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 0 0 0 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:02:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 2 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 1 1 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:03:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 9 3 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 0 9 9 2 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:04:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 9 9 9 9 9 4 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 1 9 9 9 9 3 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:05:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 2 9 9 9 9 9 9 4 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:06:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 3 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:07:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 4 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9|

+------------------------------------------------------------------------------------------------------------------------+

Line : 1 - Time: 00:00:08:000:0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

+------------------------------------------------------------------------------------------------------------------------+

0| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9|

1| 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 5 9 9 9 9 9 9 9 9 9 9 9|

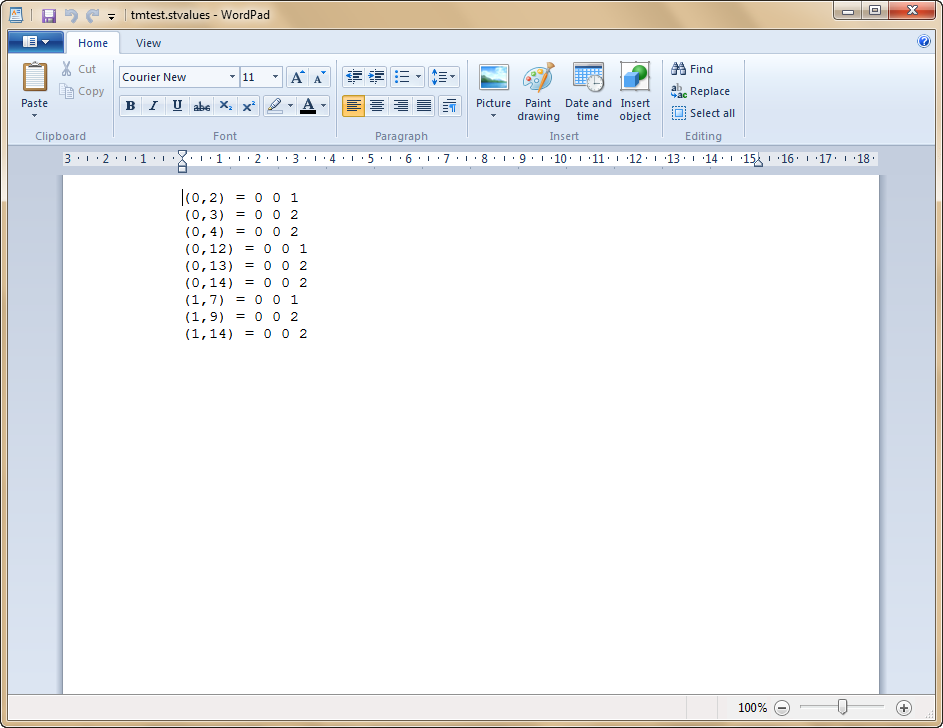
+------------------------------------------------------------------------------------------------------------------------+

Three cars start with speed zero. They start with different speed because they have different accelerated speed. After change line, they gradually speed up until get the maximum speed.

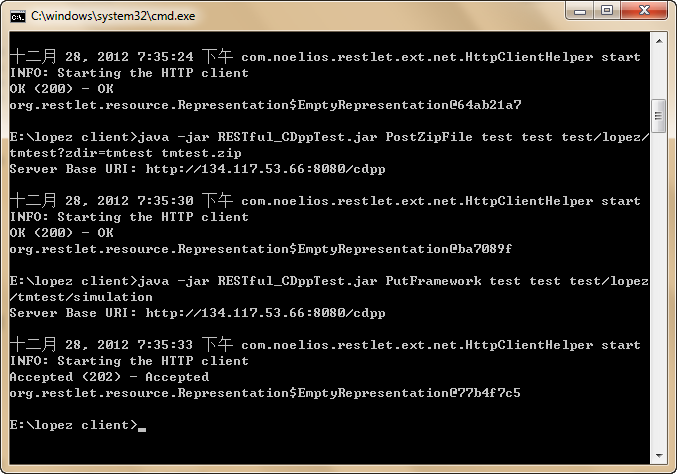
## Demo and Results

Please refer to speedUp.zip, showDown.zip, keepMoving.zip and acce2.zip as well as changeLane.zip for the details. The demo\_trafficMonitor file is for the result of the video version.

Here are some brief screenshots of the demo:



**Fig 17 File Initial value of the cells**



**Fig 18 Run simulation on the server**



**Fig 18 Get simulation result**

# conclusions

Based on the traffic monitor’s behavior, the Cell-DEVS model is created accordingly. With different data and methods chosen to test the model, the simulator works as expected. In conclusion, the M&S of traffic monitor satisfies the V&V requirement.

Reference List

[1] Kai Nagel and Michael Schreckenberg, “A cellular automaton model for freeway traffic”. J. Phys. I France 2 (1992) 2221-2229 DECEMBER 1992, PAGE 2221

[2] Wei-Lieh Hsu, Chang-Lung Tsai, Po-Lun Chang, “Automatic Traffic Monitoring Method Based on Cellular Model”. 2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing.

[3]http://www.psychocats.net/ubuntu/virtualbox

[4][Gabriel A. Wainer](http://cell-devs.sce.carleton.ca/publications/index.php?idAuthor=6), [Khaldoon Al-Zoubi](http://cell-devs.sce.carleton.ca/publications/index.php?idAuthor=47" \t "_blank),“[Distributed Simulation Using RESTful Interoperability Simulation Environment (RISE) Middleware”](http://cell-devs.sce.carleton.ca/publications/2010/WA10/), Springer-Verlag, page 1--100 – 2010.

[5][Khaldoon Al-Zoubi](http://cell-devs.sce.carleton.ca/publications/index.php?idAuthor=47" \t "_blank), [Gabriel A. Wainer](http://cell-devs.sce.carleton.ca/publications/index.php?idAuthor=6),["Using REST Web-Services Architecture for Distributed Simulation",](http://cell-devs.sce.carleton.ca/publications/2009/AW09/) 23rd Workshop on Principles of Advanced and Distributed Simulation, page TBD, Lake Placid, NY – 2009.