Part IV Collect Vehicular Traffic Statistics

Purpose: collect vehicular traffic statistics in Veins-3.0 of OMNeT, e.g., average vehicle speed, vehicle interarrival time

Prerequisite: Correctly installed SUMO 0.21.0 and OMNeT++ 4.6

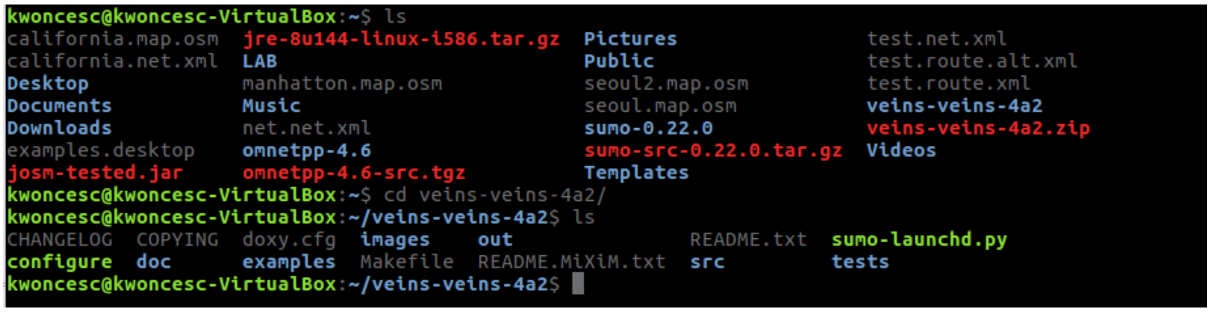
Procedures:

1. Download Veins-3.0 packet

   a. http://veins.car2x.org/download/

   b. Unzip Veins-3.0.zip:

   c. Go to the veins-veins-3.0 folder which we just unzipped

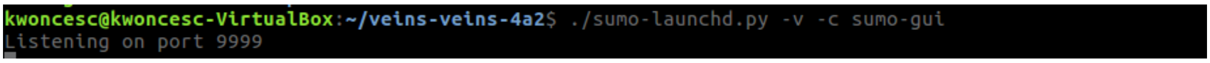


```
kwoncesc@kwoncesc-VirtualBox:~$ ls
california.map.osm  jre-8u144-linux-i586.tar.gz  Pictures                 test.net.xml
california.net.xml  LAB                          Public                   test.route.alt.xml
Desktop             manhatton.map.osm            seoul2.map.osm           test.route.xml
Documents           Music                        seoul.map.osm            veins-veins-4a2
Downloads           net.net.xml                  sumo-0.22.0              veins-veins-4a2.zip
examples.desktop    omnetpp-4.6                  sumo-src-0.22.0.tar.gz  Videos
josm-tested.jar     omnetpp-4.6-src.tgz          Templates
kwoncesc@kwoncesc-VirtualBox:~$ cd veins-veins-4a2/
kwoncesc@kwoncesc-VirtualBox:~/veins-veins-4a2$ ls
CHANGELOG  COPYING  doxy.cfg  images    out               README.txt  sumo-launchd.py
configure  doc      examples  Makefile  README.MiXiM.txt  src         tests
kwoncesc@kwoncesc-VirtualBox:~/veins-veins-4a2$ 
```

   d. Type:

      ./sumo-launchd.py -v -c sumo-gui

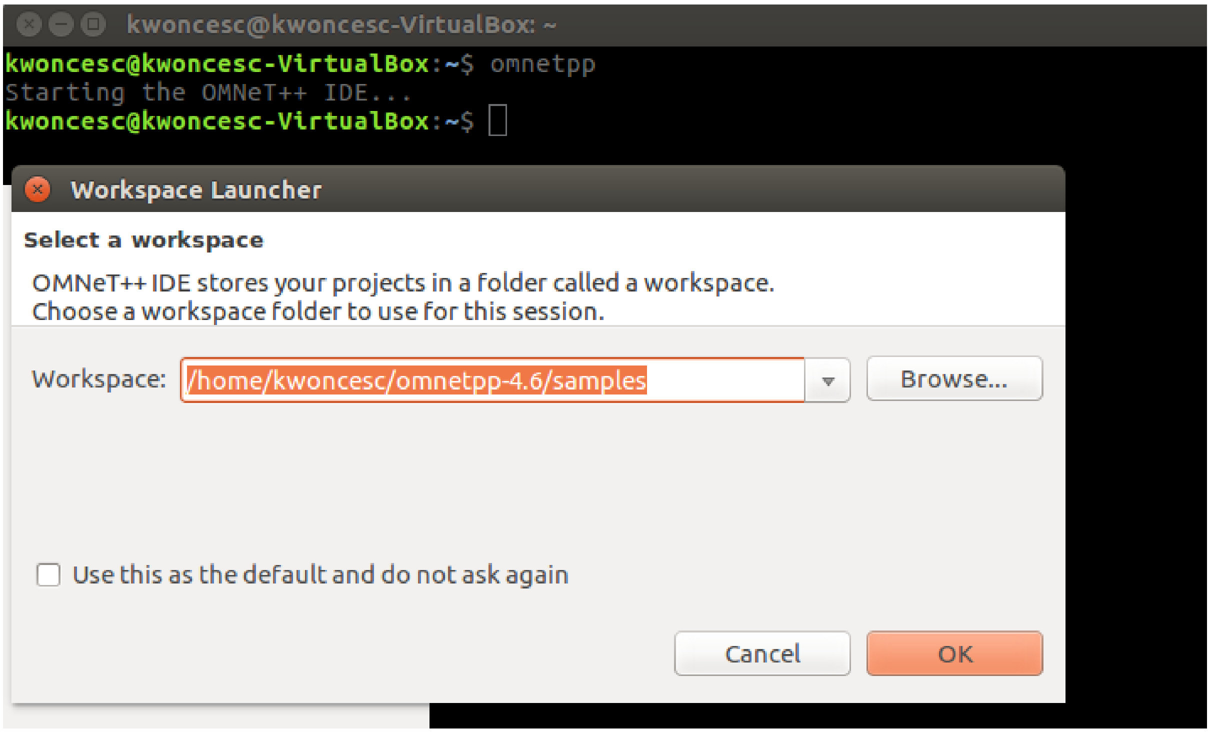
      to run the TraCI server of SUMO (Notice: the port number shall be same as *.ini file in Veins of OMNeT++)



```
kwoncesc@kwoncesc-VirtualBox:~/veins-veins-4a2$ ./sumo-launchd.py -v -c sumo-gui
Listening on port 9999
```

2. Open OMNeT++ 4.6 by typing omnet++ in a new terminal (Environment setting in .bashrc is done before, if not refer to the

Installation Manual)



```
kwoncesc@kwoncesc-VirtualBox: ~
kwoncesc@kwoncesc-VirtualBox:~$ omnetpp
Starting the OMNeT++ IDE...
kwoncesc@kwoncesc-VirtualBox:~$
```

Workspace Launcher

Select a workspace

OMNeT++ IDE stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.

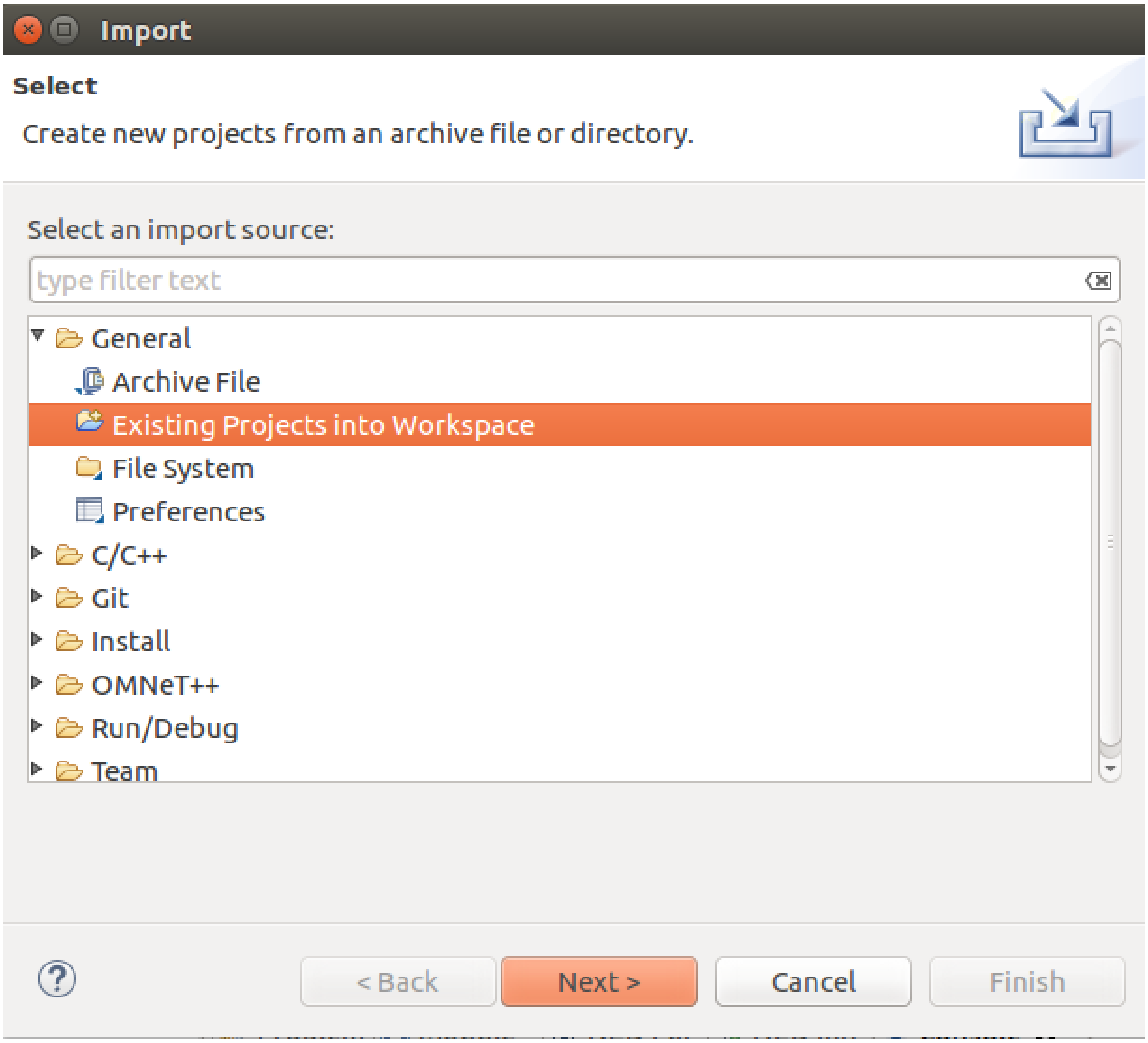
Workspace: /home/kwoncesc/omnetpp-4.6/samples

□ Use this as the default and do not ask again

3. Wait, and open a workspace, and wait, you will see the GUI of OMNeT++ 4.6

4. Import Veins-3.0

   a. File->Import, then you will see:



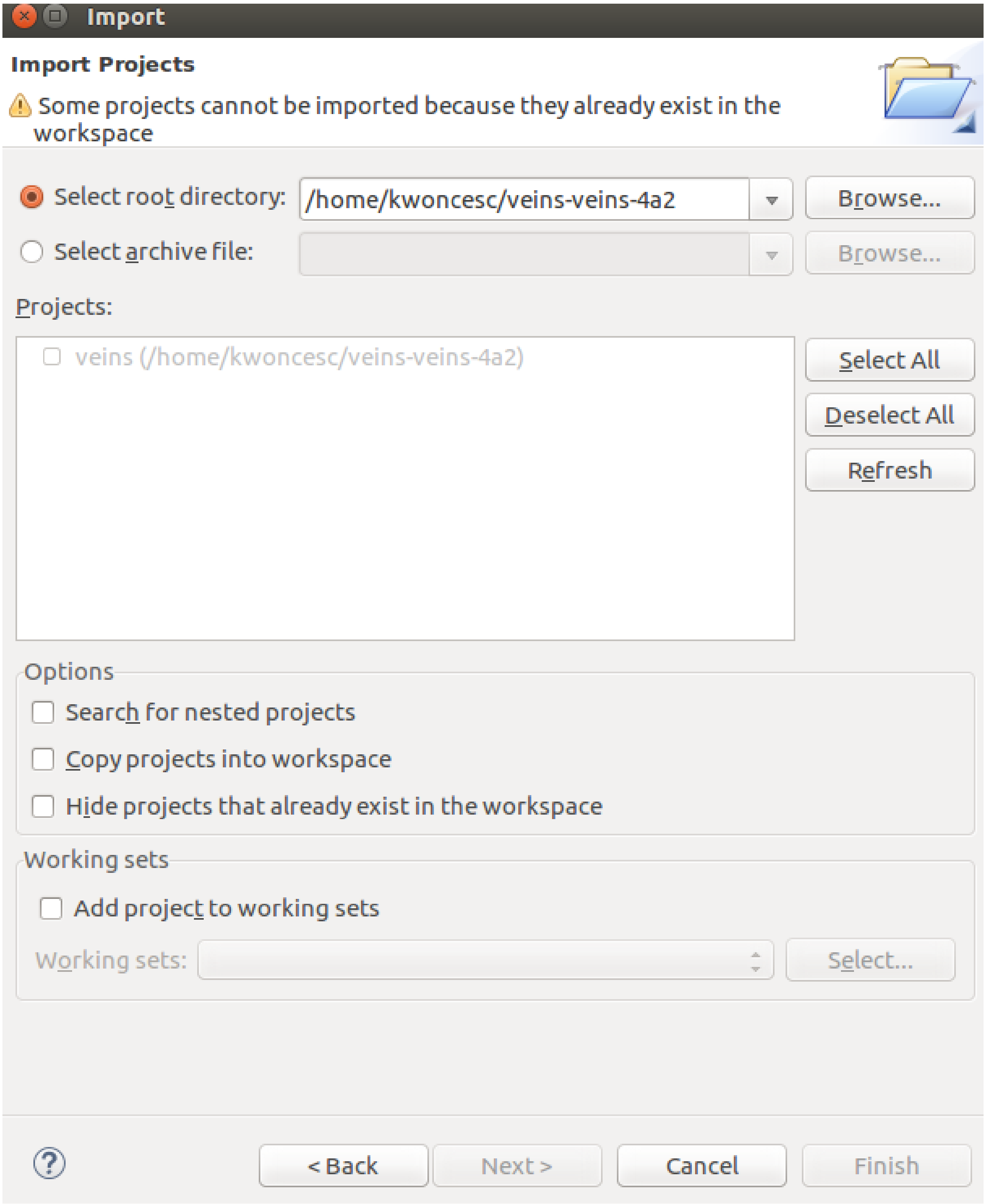
Import

Select

Create new projects from an archive file or directory.

Select an import source:

b. Select Existing Projects into Workspace, then Next, then Browse, select the veins-veins-3.0 folder we unzipped, then OK, it will back to the above window, then Finish.



Import Projects

⚠ Some projects cannot be imported because they already exist in the workspace

● Select root directory: /home/kwoncesc/veins-veins-4a2

○ Select archive file:

Projects:

□ veins (/home/kwoncesc/veins-veins-4a2)

Options

□ Search for nested projects

□ Copy projects into workspace

□ Hide projects that already exist in the workspace

Working sets

□ Add project to working sets

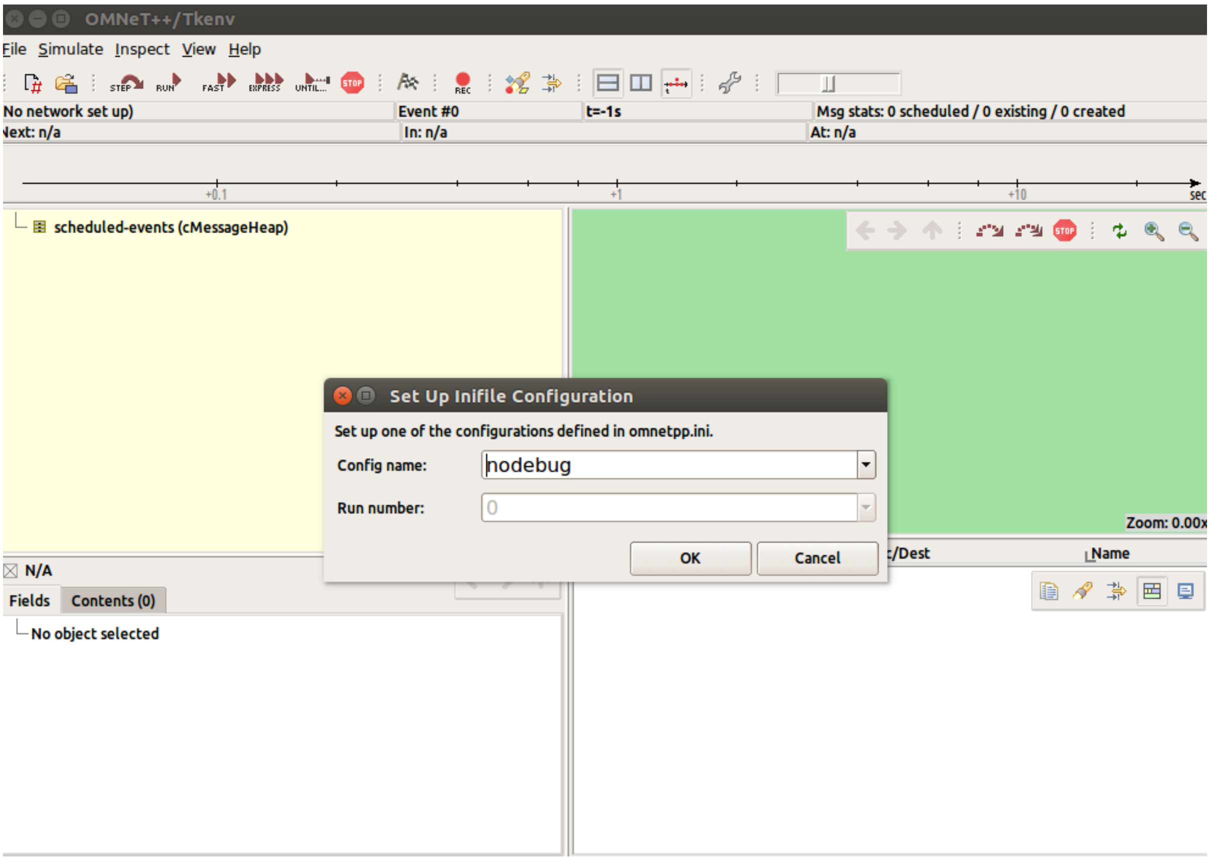
Working sets:

c. In the left side, the Project Explorer, you will see a new project (item): veins.

d. Right click veins, click Build Project, then wait, it will compile the source code.

e. Open examples->veins, find the omnetpp.ini, right click, find Run As

->1 OMNeT++ Simulation. Then Click, the default simulation will run, which is erlangen city map of Germany.



Set Up Inifile Configuration

Set up one of the configurations defined in omnetpp.ini.

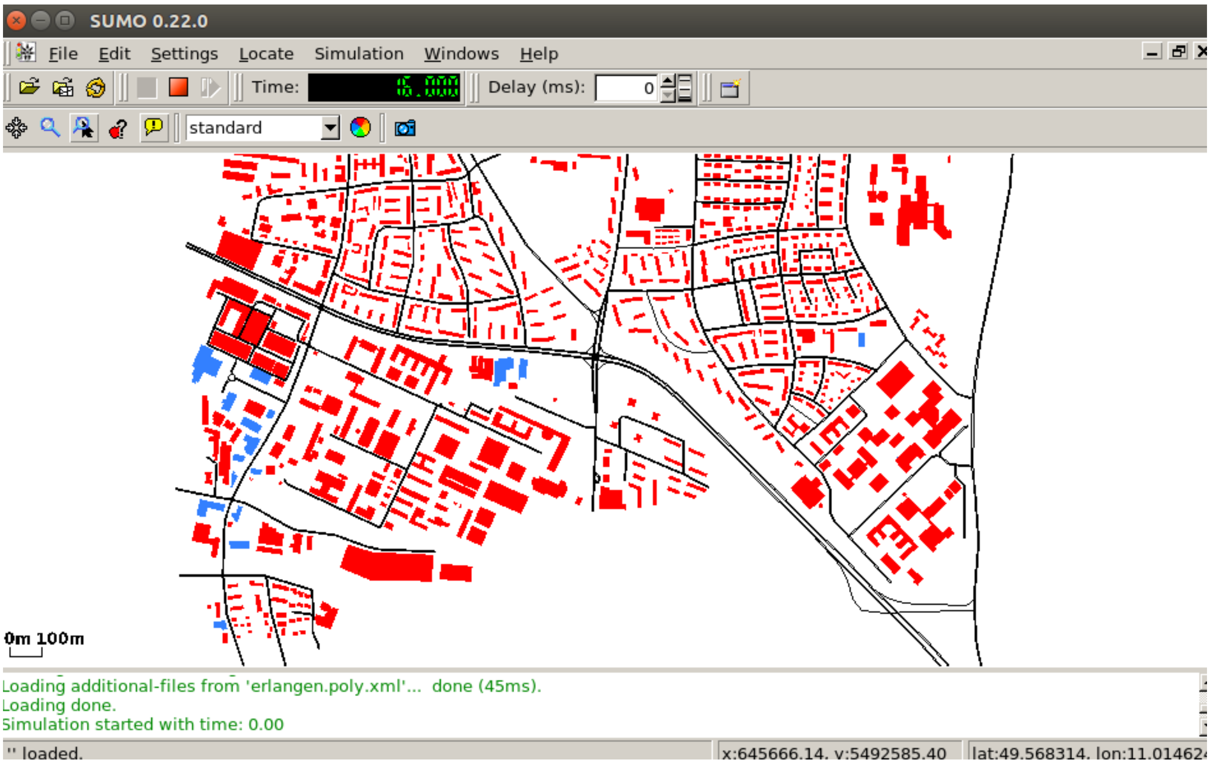
Config name: nodebug

Run number: 0

OMNeT++/Tkenv
File Simulate Inspect View Help
STEP RUN FAST EXPRESS UNTIL... STOP REC
No network set up)
Event #0
t=-1s
Msg stats: 0 scheduled / 0 existing / 0 created
Next: n/a
In: n/a
At: n/a
+0.1
+1
+10
sec
scheduled-events (cMessageHeap)
Zoom: 0.00x
N/A
Fields
Contents (0)
No object selected
:/Dest
Name
OK
Cancel

f. Set UP Inifile Configuration, just click OK. Later you would need to revise ini file, then may change here, now just click OK.

g. The simulation window comes out, click top menu: RUN, you will see SUMO window jumping out, and the simulation is running.



```
Loading additional-files from 'erlangen.poly.xml'...  done (45ms).
Loading done.
Simulation started with time: 0.00
'' loaded.    x:645666.14, y:5492585.40    lat:49.568314, lon:11.01462
```

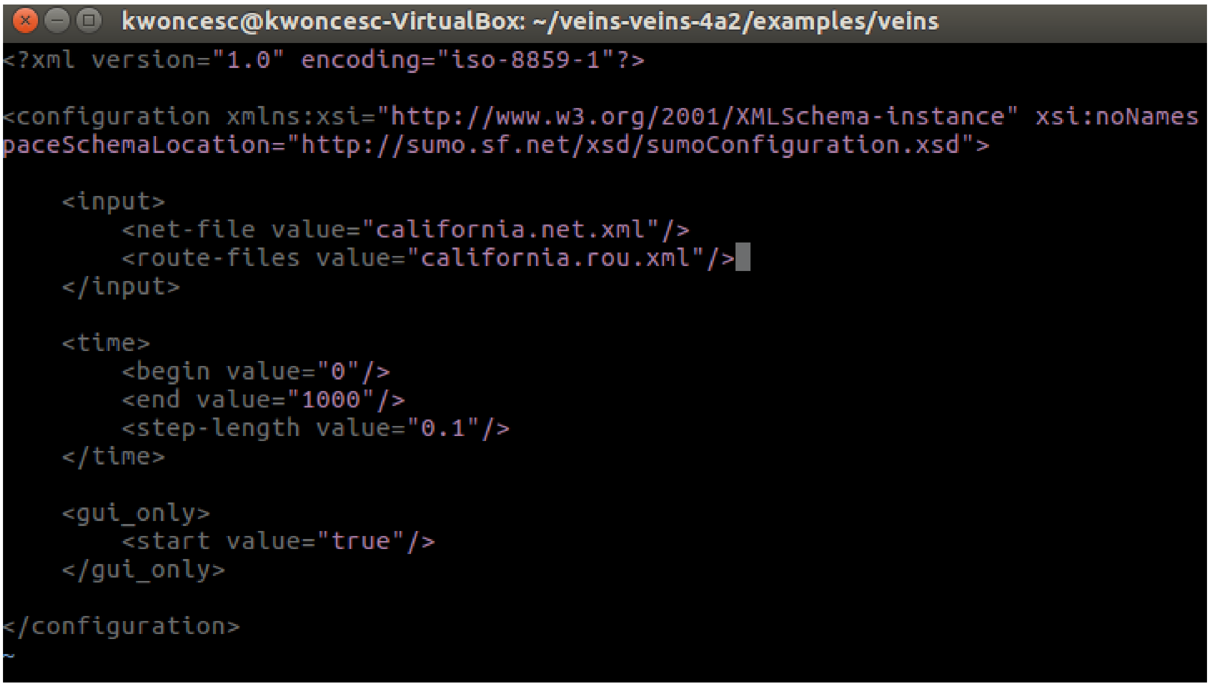
h. Stop simulation: click STOP in Tkenv of OMNeT

i. The Tkenv window shows the node moving out and an rsu[0], if you zoom in, you will see more details. The SUMO window just shows the map and moving vehicles.

5. Use map made by ourself

a. Copy the map files (*.map.xml, *.rou.xml) to the examples->veins in the Project Explorer window, you can either move or copy directly.

b. Revise erlangen.sumo.cfg file with imported *.map.xml and *.rou.xml file name and save as a new file with a new name, like *.sumo.cfg.


kwoncesc@kwoncesc-VirtualBox: ~/veins-veins-4a2/examples/veins

```
<?xml version="1.0" encoding="iso-8859-1"?>

<configuration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNames
paceSchemaLocation="http://sumo.sf.net/xsd/sumoConfiguration.xsd">

    <input>
        <net-file value="california.net.xml"/>
        <route-files value="california.rou.xml"/>
    </input>

    <time>
        <begin value="0"/>
        <end value="1000"/>
        <step-length value="0.1"/>
    </time>

    <gui_only>
        <start value="true"/>
    </gui_only>

</configuration>
~
```

c. Revise erlangen.launchd.xml with similar manner, *.net.xml, *.rou.xml, *.sumo.cfg and save as a new file with name *.launchd.xml (delete poly item).

d. Open omnetpp.ini file (source code view, switch at the bottom, Form<->Source), modify the TraCIScenarioManager parameters section, the parameter:

.manager.launchConfig = xmldoc(".launchd.xml")

e. Then you can run the simulation with your new map.

6. Source code we may need to deal with

a. MAC layer:

src -> modules -> mac -> ieee80211p -> Mac1609_4.cc, .h

b. Mobility:

src -> modules -> mobility -> traci -> TraCIMobility.cc, .h, (possible: TraCICommandInterface.h, .cc, TraCIConstants.h)

c. Modules (Network Entity Description files):

src -> modules -> nic -> Nic80211p.ned

src -> nodes -> Car.ned

src -> modules -> mac -> ieee80211p -> Mac1609_4.ned

Notice: when you modify these source code, you'd better save as a new file in case you want to refer to the original code after. When you rename the source code, you need to modify namespace and related head files and the corresponding *.ned file, or you rename the project name.

7. Example of getting traffic statistics

(if the font is too small, please zoom in)

This is to get vehicle mobility information, between // Chris and // Chris end are the code I wrote, notice this is in Mac1609_4.cc file.

The right side is the print out information, you can see each node name, Position, Speed, RoadID.

There are many other APIs in TraCIMobility.h and TraCICommandInterface.h or define your own API in TraCICommandInterface.cc and .h based on TraCIConstants.h.