Run Vehicle in the road network
9/5 - 9/11 by Chris Shen

Tools
Generating random traffic in road network:

ls ../sumo-0.21.0/tools/trip

```
chris@chris-ubuntu:~/usr/sumo-0.21.0/tools/trip$ ls
generateTripsXml.py  removeGeometryInTrips.pl      route2trips.py
randomTrips.py       removeTripsIfNotInInterval.pl splitTripsByHours.pl
```

Run:

./randomTrips.py --help

```
chris@chris-ubuntu:~/usr/sumo-0.21.0/tools/trip$ ./randomTrips.py --help
Usage: randomTrips.py [options]

Options:
  -h, --help              show this help message and exit
  -n NETFILE, --net-file=NETFILE
                          define the net file (mandatory)
  -a ADDITIONAL, --additional-files=ADDITIONAL
                          define additional files to be loaded by the router
  -o TRIPFILE, --output-trip-file=TRIPFILE
                          define the output trip filename
  -r ROUTEFILE, --route-file=ROUTEFILE
                          generates route file with duarouter
  --pedestrians           create a person file with pedestrian trips instead of
                          vehicle trips
  --prefix=TRIPPREFIX     prefix for the trip ids
  -t TRIPATTRS, --trip-attributes=TRIPATTRS
                          additional trip attributes
  -b BEGIN, --begin=BEGIN
                          begin time
  -e END, --end=END       end time (default 3600)
  -p PERIOD, --period=PERIOD
```

Please check the detail explanation for this tool:
http://sumo.dlr.de/wiki/Tools/Trip

Usage:
Input the net.xml file we generated before.
Run:
./randomTrips.py -n *.net.xml -e 7200 -p 0.5

Several important parameters:
-e END, end time (default 3600)
-p PERIOD, repetition period (default 1)   // the default means every 1 second inject 1 vehicle
-o TRIPFILE, define the output trip filename   // if not specify, the default name is
trips.trips.xml

```
chris@chris-ubuntu:~/usr/sumo-0.21.0/tools/trip$ ./randomTrips.py -n /home/chris
/usr/sim/chris10x10uturn.net.xml -e 7200 -p 0.5
chris@chris-ubuntu:~/usr/sumo-0.21.0/tools/trip$ ls
generateTripsXml.py         removeTripsIfNotInInterval.pl  trips.trips.xml
randomTrips.py              route2trips.py
removeGeometryInTrips.pl   splitTripsByHours.pl
```

Then you can get trips file, you can directly use this file as **a route file** working with **net file**.
The contents of trip file:

```
 trips.trips.xml  ×

 1 <?xml version="1.0"?>
 2 <!-- generated on 2015-09-11 00:55:29.973673 by $Id:
   randomTrips.py 16365 2014-05-13 08:50:42Z namdre $
 3   options: -n /home/chris/usr/sim/chris10x10uturn.net.xml -e 7200
   -p 0.5
 4 -->
 5 <trips>
 6     <trip id="0" depart="0.00" from="RC5" to="UB3" />
 7     <trip id="1" depart="0.50" from="LF7" to="UC2" />
 8     <trip id="2" depart="1.00" from="UD2" to="DH3" />
 9     <trip id="3" depart="1.50" from="UC3" to="DF0" />
10     <trip id="4" depart="2.00" from="RB4" to="DH3" />
11     <trip id="5" depart="2.50" from="RH2" to="DD1" />
12     <trip id="6" depart="3.00" from="UA3" to="DH4" />
13     <trip id="7" depart="3.50" from="DA4" to="RB6" />
14     <trip id="8" depart="4.00" from="RD3" to="LE8" />
```

Specifying route

Or by the following tool some shortest path can be generated as a route file based on the trip file made above:

duarouter

```
chris@chris-ubuntu:~/usr/sim$ duarouter
SUMO duarouter Version 0.21.0
 Copyright (C) 2001-2014 DLR and contributors; http://sumo-sim.org
 License GPLv3+: GNU GPL Version 3 or later <http://gnu.org/licenses/gpl.html>
 Use --help to get the list of options.
```

For a complete information of this tool, please check the following page:
http://sumo.dlr.de/wiki/DUAROUTER

Several important parameters:
-t, --trip-files FILE   // Read trip-definitions from FILE(s), which is generated from generateTrips.py
-n, --net-file FILE   // Use FILE as SUMO-network to route on, specify the net file

-o, FILE   // specify output file name

The generated file looks like following:

```
route.xml ×
1  <?xml version="1.0" encoding="UTF-8"?>
2
3  <!-- generated on Fri Sep 11 01:10:55 2015 by SUMO duarouter Version 0.21.0
4  <?xml version="1.0" encoding="UTF-8"?>
5
6  <configuration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:noNamespaceSchemaLocation="http://sumo-sim.org/xsd/duarouterConfiguration.xsd">
7
8      <input>
9          <net-file value="/home/chris/usr/sim/chris10x10uturn.net.xml"/>
10         <trip-files value="/home/chris/usr/sumo-0.21.0/tools/trip/trips.trips.xml"/>
11     </input>
12
13     <output>
14         <output-file value="route.xml"/>
15     </output>
16
17 </configuration>
18 -->
19
20 <routes xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:noNamespaceSchemaLocation="http://sumo-sim.org/xsd/routes_file.xsd">
21     <vehicle id="0" depart="0.00">
22         <route edges="RC5 LC5 LC4 LC3 UB3"/>
23     </vehicle>
24     <vehicle id="1" depart="0.50">
25         <route edges="LF7 UE7 UD7 LD6 LD5 LD4 LD3 LD2 UC2"/>
26     </vehicle>
27     <vehicle id="2" depart="1.00">
28         <route edges="UD2 RD2 DD3 DE3 DF3 DG3 DH3"/>
29     </vehicle>
30     <vehicle id="3" depart="1.50">
31         <route edges="UC3 LC2 LC1 LC0 DC0 DD0 DE0 DF0"/>
32     </vehicle>
33     <vehicle id="4" depart="2.00">
```

Running in SUMO-GUI

2 ways: command line and in-gui

command line:
sumo-gui -n *.net.xml -r *.route.xml   // read net file and route file, for more
information, please see the sumo-gui --help

in-gui:
1. make configuration file:
   file name shall be ***.sumocfg**

```
<configuration>
    <input>
        <net-file value="test.net.xml"/>
        <route-files value="test.rou.xml"/>
        <additional-files value="test.add.xml"/>
    </input>
</configuration>
```
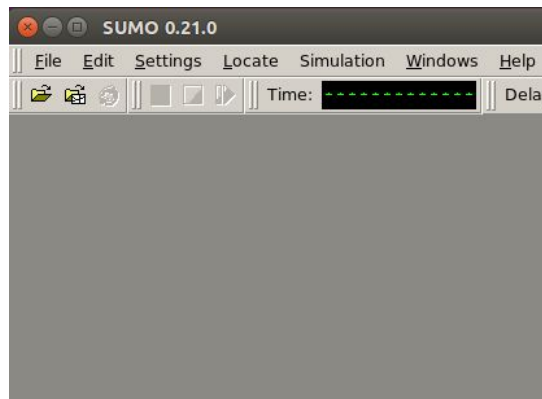
<configuration>
  <input>
    <net-file value="*.net.xml"/>
    <route-files value="*.rou.xml"/>
    <additional-files value="*.add.xml"/>
  </input>
</configuration>

2. open sumo-gui, and load the configuration file (or directly use command line mode)
File->Open Simulation->load configuration file

Then run the simulation, you will see vehicles running in the road network.