SUMO Tutorial

Jakob Erdmann, Michael Behrisch, Peter Wagner

ITSC2015, Las Palmas

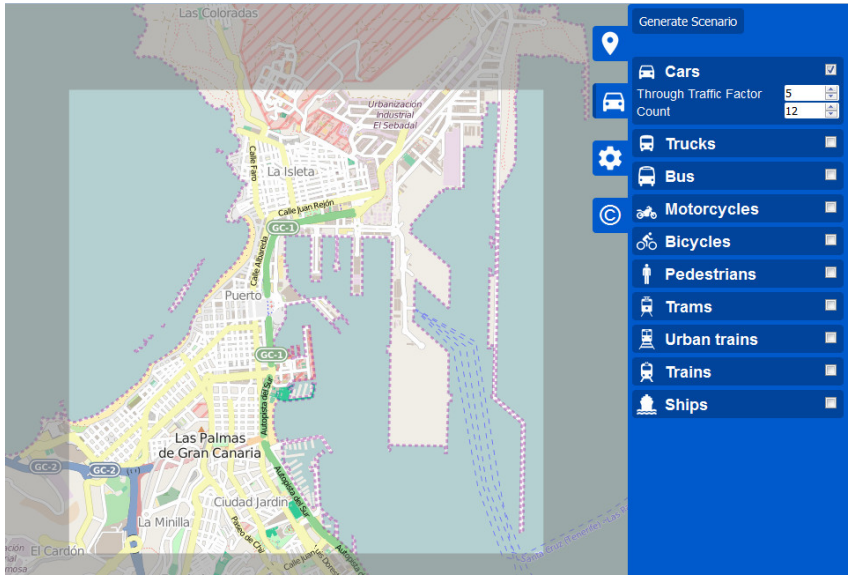
Outline

- Setup
- Introduction to SUMO
- 3-click scenario generation
- Editing a scenario with NETEDIT
- Multi- and intermodal simulation
- Outputs
- TraCI
- Recent additions & future developments
- Time for Questions

Folder:01_server.py

server.py

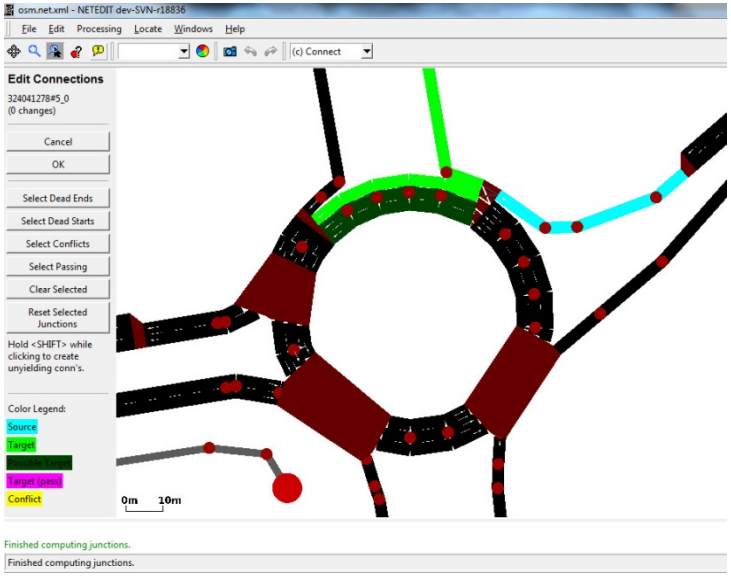
- Getting a basic scenario with tools/import/osm/server.py
  - Mode-specific network options
  - Random traffic
- Configure
  - Area
  - Traffic modes
  - Traffic volume
  - Fraction of through-traffic
- Generated files allow rebuilding and adapting the scenario



Folder:02_net_modified

Fixing network problems with NETEDIT

- Documentation at http://sumo.dlr.de/wiki/NETEDIT
- Open *osm.net.xml* in NETEDIT
- Fix traffic problem at roundabout
  - Join some junctions
  - Add and modify connections
- Run build.bat again to regenerate random demand (edges lost due to joining)



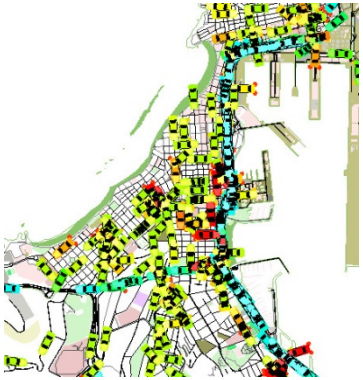
Folder:02_net_modified

Adapting Demand

- Parameters for random demand in build.bat
- Define custom traffic flow (*custom.rou.xml*):

```
<flow id="toTown" begin="0" end="3600" probability="0.25"
   from="310772654#0" to="30534383#2"/>
<flow id="fromTown" begin="0" end="3600" probability="0.25"
   from="205013618#0-AddedOffRampEdge" to="176908062#1"/>
```

- Reproduce/run scenario by running custom.sumocfg



Folder:03_multimodal

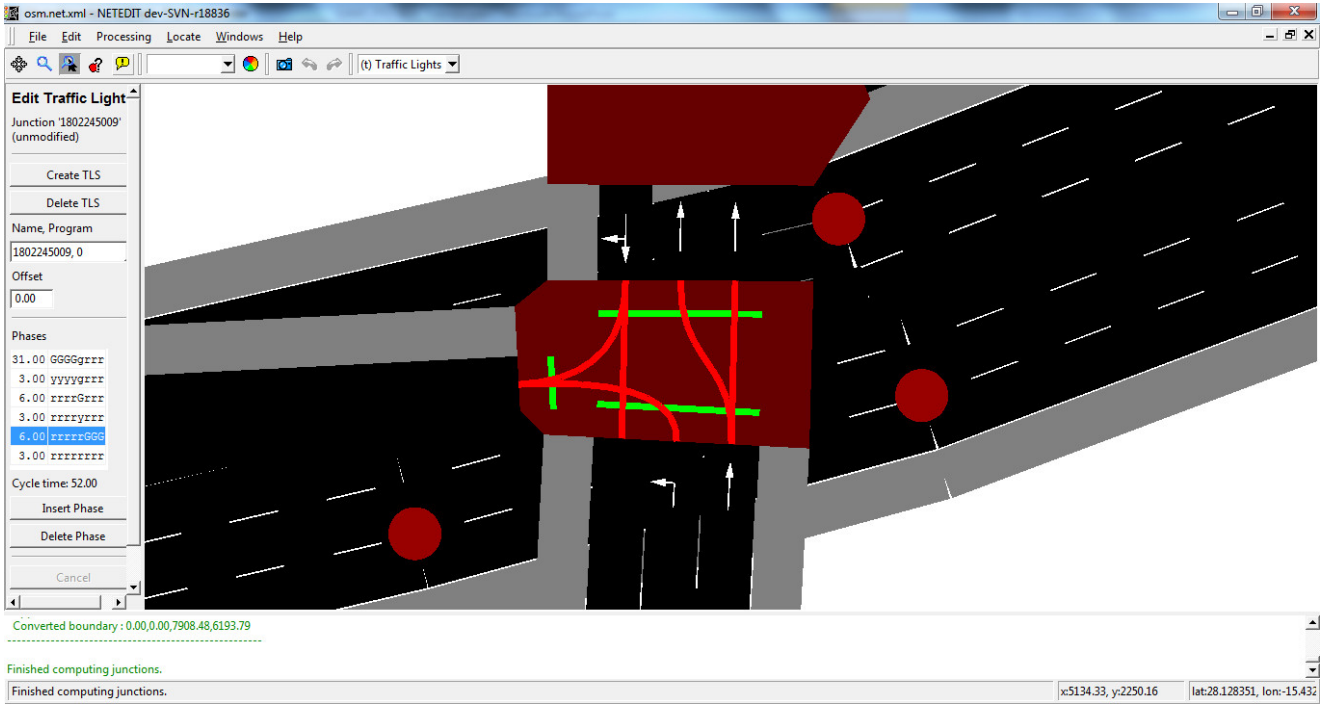
A multi-modal scenario

- Add more traffic modes using the web-interface of server.py
  - Sidewalks and crossings are added heuristically when adding pedestrians

Folder:04_multimodal_modified

Fixing the network

- Disable network building heuristics (not well adapted for Las Palmas)
  - netconvert -c osm.netccfg --junctions.join false --ramps.guess false
- Fix traffic light plan in NETEDIT
- Rebuild demand using build.bat



Folder:05_public_transport

Public transport

- Define bus stops

```
<busStop id="SantaCatalina" lane="37350442_0"
    startPos="100" endPos="140" lines="30-Aeropuerto"/>
```

- Define busses

```
<vType id="bus" vClass="bus"/>
<flow id="bus_line30" type="bus" begin="0" end="3600"
  period="600" from="310772654#0" to="176908062#1"
  departSpeed="max" line="30">
      <stop busStop="SantaCatalina" duration="60"/>
</flow>
```

- Run scenario by running custom5.sumocfg

Folder:05_public_transport

Intermodal trip chain

- Walk from the hotel to the bus station and ride to bus to the airport

```
<person id="leaving_early" depart="0" color="green">
    <walk from="4526354#6" to="37350442" arrivalPos="112"/>
    <ride to="176908062#1" lines="30"/>
</person>
```

- May use multiple lines serving the same destinations
- No online-rerouting (i.e. when missing a bus)
- Run scenario by running custom5a.sumocfg

Simulation Outputs

- Generate edgeData (outputs.sumocfg)
  `<edgeData id="10min_agg" freq="600" file="edgedata.xml"/>`
  - Visualize *waitingTime* in SUMO-GUI (showEdgeData.sumocfg)
- Generate *tripinfos*: `--tripinfo-output`
  - Convert to csv, open as spreadsheet
  - python %SUMO_HOME%\tools\xml\xml2csv.py tripinfos.xml
- Generate traces (outputs_fcd.sumocfg)
  - Convert with traceExporter.py (exportTraces.bat)
  - Visualize traces with showTraces.sumocfg
- V2X Frameworks (*veins.car2x.org* , VSimRTI)
- Other outputs
  - Induction loops
  - Traffic lights
  - Batteries

TraCI - Traffic Control Interface

- Learn how to influence a running simulation

- Start with 02_net_modified
- Add an induction loop to trigger a traffic light (detectors.add.xml)

```
<inductionLoop id="0" lane="290619181#3_0" pos="-5"
freq="1" file="det.out.xml"/>
```

- Write a small script to trigger the traffic light from outside (runner.py)
- Watch the traffic light being triggered

- Bonus:
- Analyse outputs
- Create collisions

- For documentation see http://sumo.dlr.de/pydoc/traci.html

Recent extensions

Version 0.24.0

- Electric vehicle model (battery, charging stations)
- Support for left-hand networks
- Improved TraCI support for pedestrians (new Tutorial)
- Vehicle-class specific speed limits
- Deliberate junction blocking (spillback of traffic jams onto an intersection)
- Import of bus lanes and bike lanes from OSM
- Updated VISSIM import *(.inpx*)

Development version

- NETEDIT is now open

Next Steps

- Improved support for intermodal simulation
  - Intermodal router
  - Better network building heuristics (connections, traffic lights, etc.)
  - Import of public transport from OSM
  - More TraCI support for persons
  - Better NETEDIT support for multi-modal networks
- Extended traffic models
  - Support for vehicle widths (heterogeneous traffic)
  - Driving through oncoming traffic
  - Better support for bicycles
  - Additional pedestrian models
- Ongoing validation of all models

Conclusion

- Use <SUMO_HOME>/tools/import/osm/server.py to get a quick start
- Read the documentation / FAQ at http://sumo.dlr.de/wiki
- Report any bugs you find to sumo-user@lists.sourceforge.net
- Share your scenarios and results

- Talks to us. We are always looking for project partners! sumo@dlr.de