

Welcome to SUMO 2022



May 9th – 11th
Cyberspace



Knowledge for Tomorrow



SUMO Tutorial

Jakob Erdmann

SUMO2022, Online



Knowledge for Tomorrow



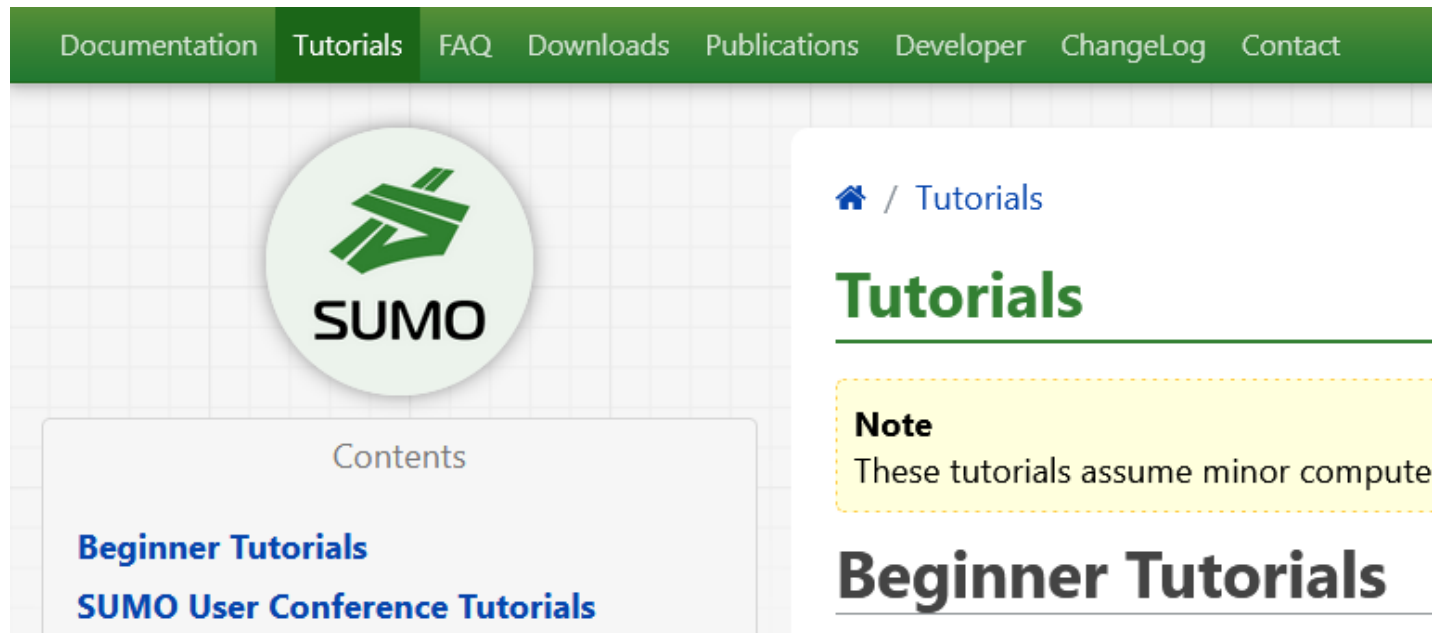
Outline

- Prerequisites
- 3-Click scenario generation with `osmWebWizard.py`
- Network editing (junctions, filtering, traffic light types)
- Creating traffic with individual flows
- Opposite direction driving
- Pedestrian crossings from scratch
- Traffic in search of parking



Prerequisites

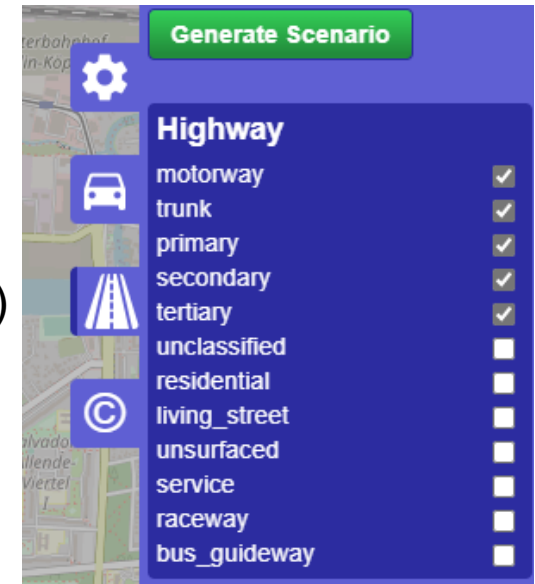
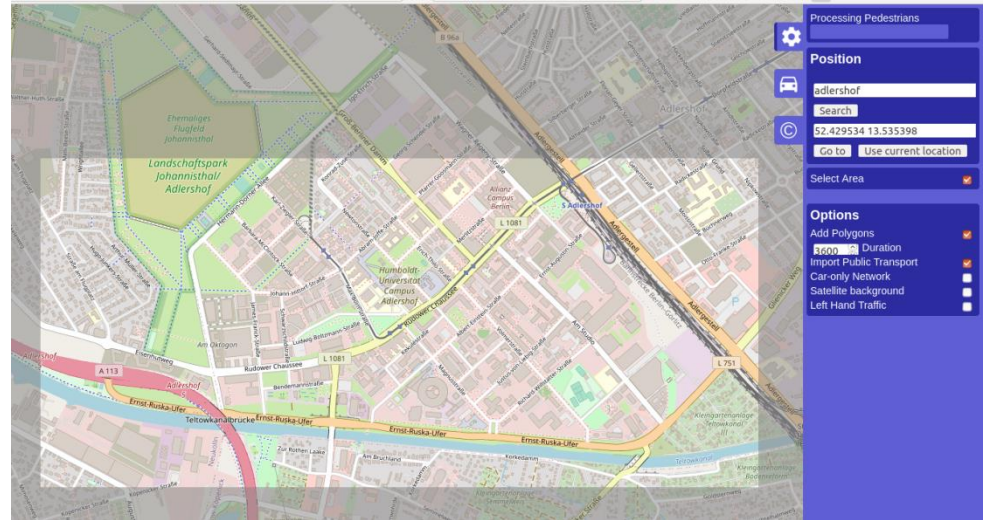
- SUMO 1.13.0 for running simulations
- Python: python.org/download/
- Text Editor (i.e. notepad-plus-plus.org/)
- Data files: sumo.dlr.de/daily/sumo2022_tutorial.zip



The screenshot shows the SUMO website's navigation bar with links: Documentation, Tutorials, FAQ, Downloads, Publications, Developer, ChangeLog, and Contact. The main content area features the SUMO logo (a green stylized 'S' with 'SUMO' text below it) and a 'Contents' section listing 'Beginner Tutorials' and 'SUMO User Conference Tutorials'. On the right, a sidebar shows a home icon and the path '/ Tutorials', followed by the heading 'Tutorials' and a yellow 'Note' box stating 'These tutorials assume minor compute'. Below the note is the heading 'Beginner Tutorials'.

osmWebWizard

- [tools/osmWebWizard.py](https://tools.osmWebWizard.py)
- OpenStreetMap network data
- **Random traffic**
- Configure
 - Area
 - **road types (new!)**
 - Traffic modes
 - Traffic volume
 - Fraction of through-traffic
 - Public Transport
 - Scenario duration
 - Building Shapes and Points-of-Interest (cosmetic)
 - Satellite background (cosmetic)
- Generated files allow rebuilding and adapting the scenario
- Example data in 01_wizard



osmWebWizard - Generated Files

- Scenario input

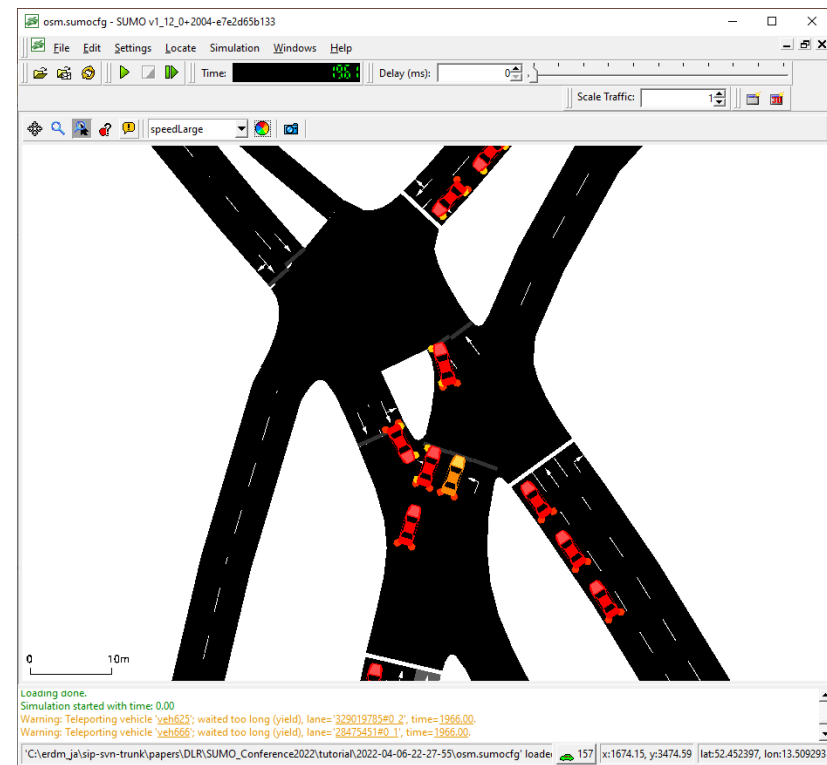
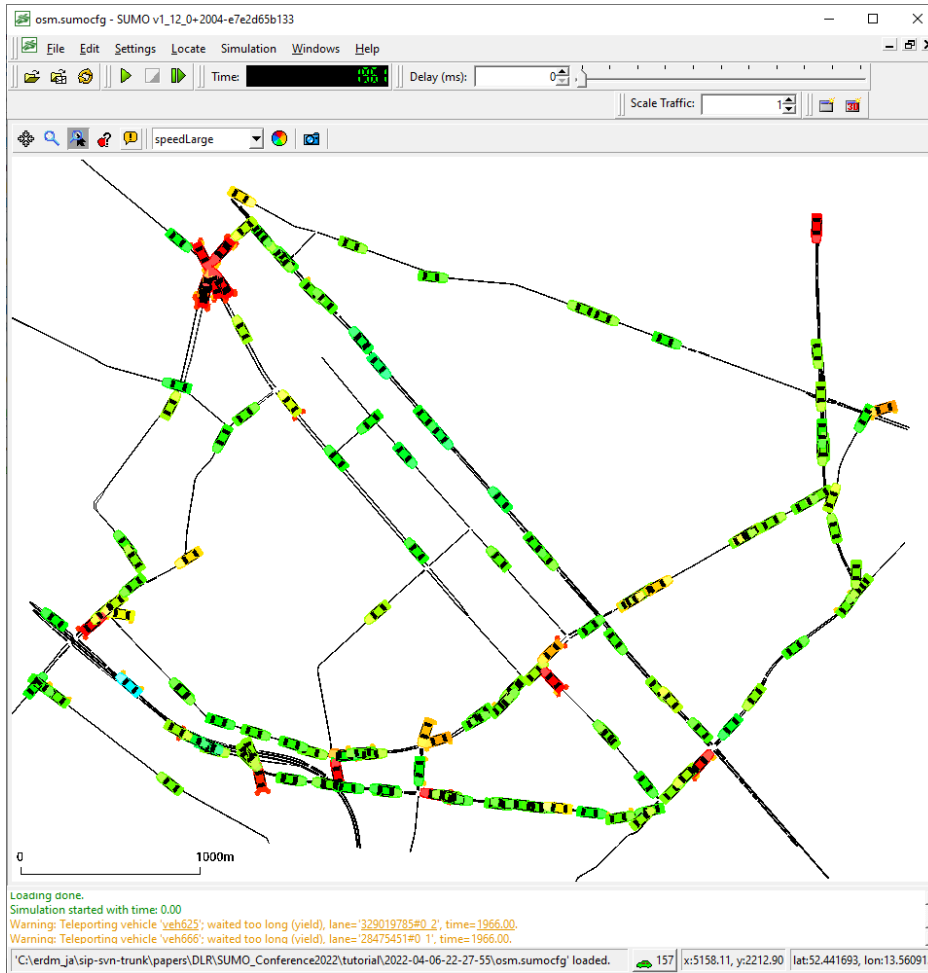
- `osm.sumocfg`: configuration file (load with **sumo**, **sumo-gui**)
- `osm.net.xml`: simulation network
- `osm.passenger.trips.xml`: passenger cars
- ~~• `osm.pedestrian.rou.xml`: persons~~
- ~~• `osm_pt.rou.xml`: busses, trams, ...~~
- ~~• `osm_stops.add.xml`: public transport stop locations~~
- ~~• `osm.poly.xml`: building shapes and POIs~~
- `osm.view.xml`: sumo-gui settings for delay, colors,...

- Rebuilding:

- `osm_bbox.osm.xml`: raw OSM data
- `osm.netccfg`: rebuild network and stops (**netconvert**)
- ~~• `osm.polyccfg`: rebuild shapes (**polyconvert**)~~
- `build.bat`: rebuilt traffic (cars, persons, public transport schedule,...)
- ~~• `osm_ptlines.xml`: intermediate public transport data~~



osmWebWizard - Simulation



osmWebWizard - Simulation

- Network is an interpretation of **filtered** OSM data
 - heuristic interpretation of complex intersections is challenging
- Traffic is random: 1538 vehicles departing over 3600s



© google maps

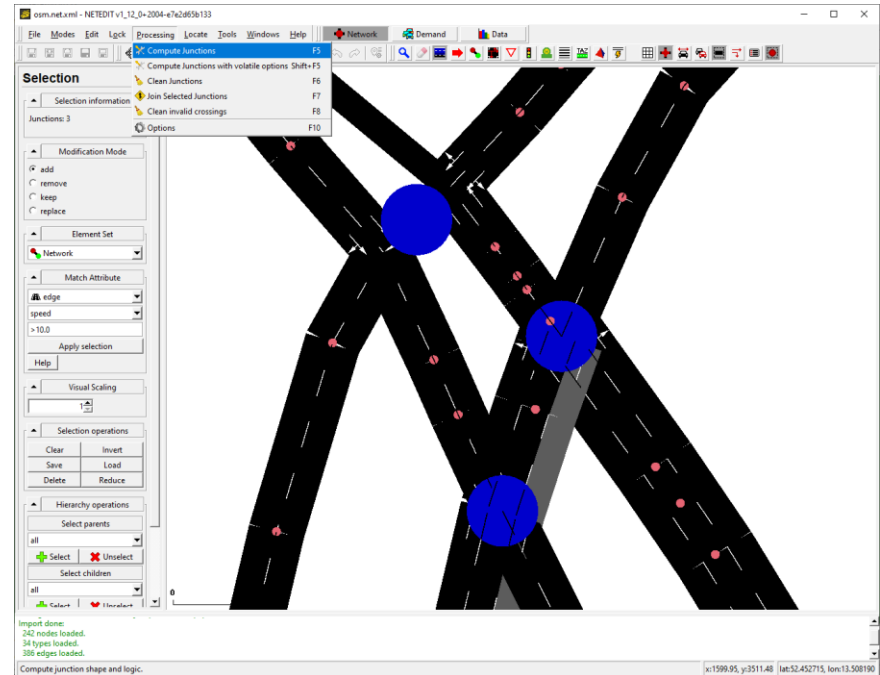


© openstreetmap



Network Editing - Join Junctions

- Load network osm.net.xml (open from sumo-gui with **CTRL+t**)
- Select mode (S)
- Select cluster of junctions which should become a single junction
- press F7
- optionally press F5 to update
- save



- Background: The heuristic that joins junctions rejected the cluster:

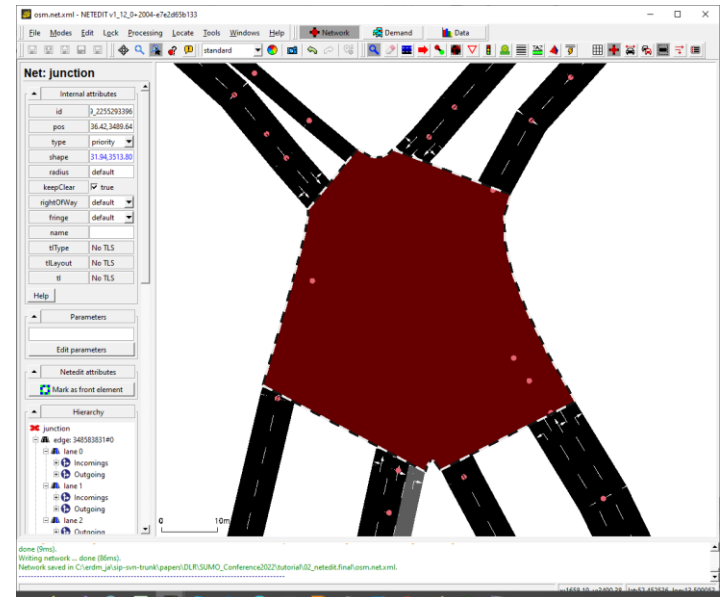
Warning: Not joining junctions

31358103,567607181,7600072730,9104198029 (not compact
(maxEdge=329019785#1 length=6.40)).



Network Editing - Join Junctions

- Example data in 02_netedit
- run **build.bat** to adapt traffic to changes
 - **On Windows, sumo-gui must be closed!**

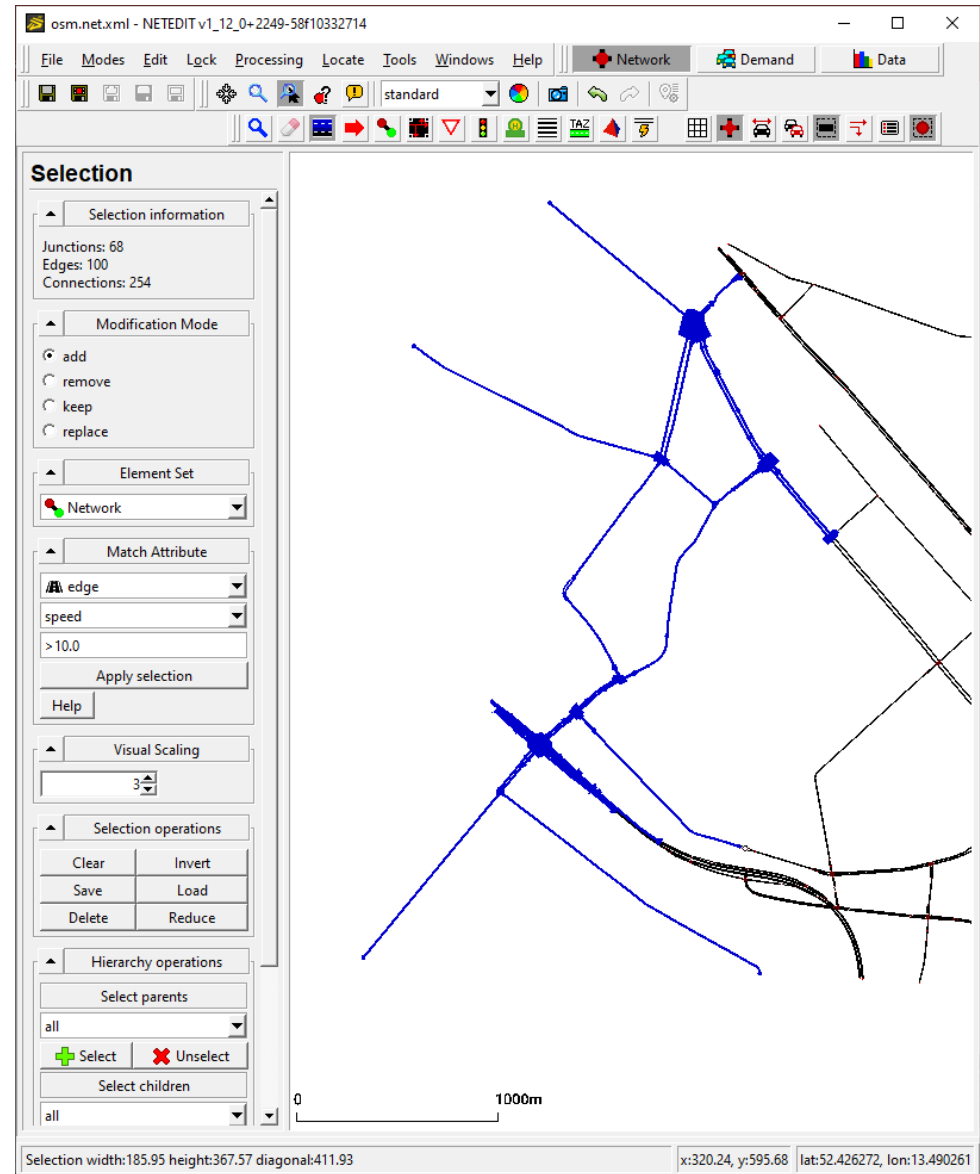


Trip Statistics	Original	Joined
RouteLength	3946	3918
Speed	11.71	11.97
Duration	351	329
WaitingTime	35	16
TimeLoss	73	53



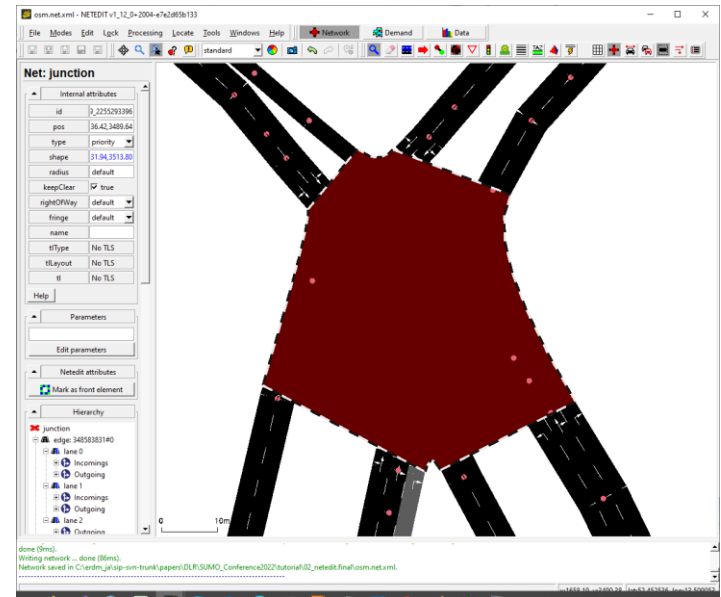
Network Editing - Reduce

- Load network osm.net.xml (open from sumo-gui with **CTRL+t**)
- Select mode (S)
- Shift-drag to select edges and junctions in rectangle (repeatedly)
- press 'Reduce' button
- save
- run **build.bat** to adapt traffic to changes
 - **On Windows, sumo-gui must be closed!**



Network Editing - Traffic Light

- Example data in 02_netedit
- run **build.bat** to adapt traffic to changes
 - **On Windows, sumo-gui must be closed!**

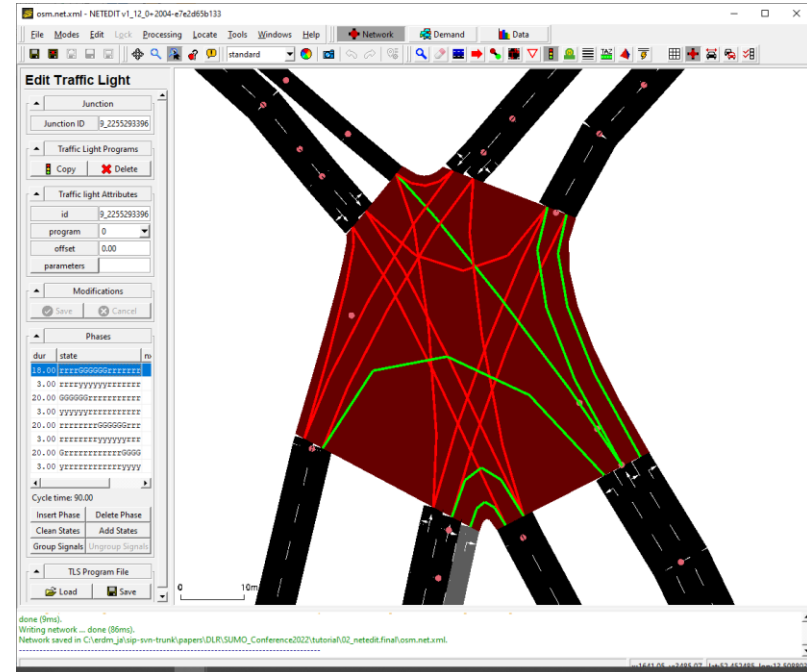


Trip Statistics	Reduced	Traffic Light
RouteLength	2183	2183
Speed	11.25	11.03
Duration	198	203
WaitingTime	11	15
TimeLoss	39	43



Network Editing - Traffic light (actuated)

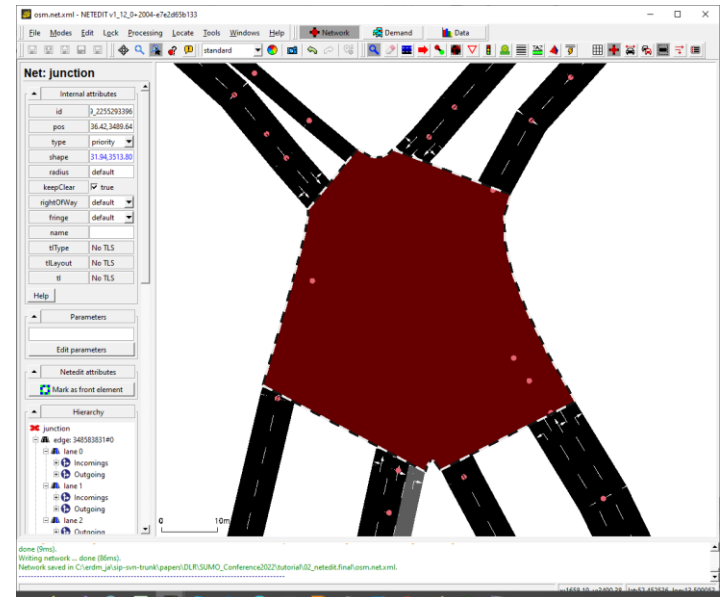
- Load network osm.net.xml (open from sumo-gui with **CTRL+t**)
- inspect mode (I)
- click joined junction
- set tlType=actuated
- optionally press F5 to update
- save



- Background: The default traffic light type is 'static' but this can be changed via options (F10)

Network Editing - Traffic Light

- Example data in 02_netedit
- run **build.bat** to adapt traffic to changes
 - **On Windows, sumo-gui must be closed!**

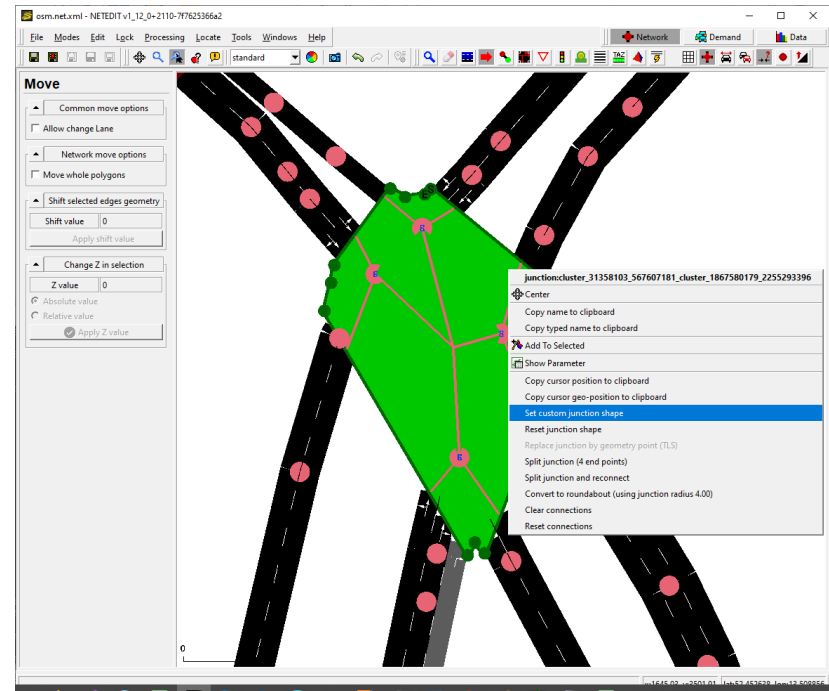


Trip Statistics	Reduced	Traffic Light	Actuated
RouteLength	2183	2183	2183
Speed	11.25	11.03	11.88
Duration	198	203	200
WaitingTime	11	15	12
TimeLoss	39	43	41



Network Editing - Junction shape

- Load network osm.net.xml (open from sumo-gui with **CTRL+t**)
- inspect mode (I)
- rightclick joined junction
- "set custom shape"
- drag shape points, drag for new points, shift-click deletes
- confirm with <ENTER>
- F5
- save
- Background: The heuristic that guesses junction shape may not always reproduce the real shape.
- No significant impact on traffic metrics **in this case**



Traffic

- Example scenario traffic has random vehicular traffic (build.bat)

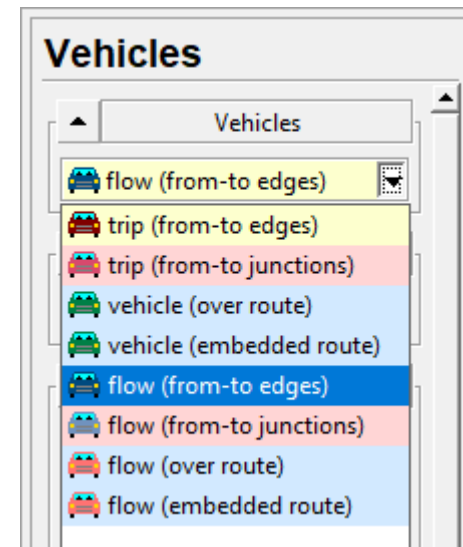
```
python "%SUMO_HOME%\tools\randomTrips.py"  
-n osm.net.xml  
-o osm.passenger.trips.xml  
-p 2.108033 -e 3600  
--vehicle-class passenger --vclass passenger --prefix veh  
--trip-attributes "departLane=\"best\""  
--min-distance 300  
--lanes  
--fringe-factor 5  
--fringe-start-attributes "departSpeed=\"max\""  
--allow-fringe.min-length 1000  
--validate
```

- Next: Add custom traffic flow



Demand Editing - Flow

- Load network `osm.net.xml` (open from `sumo-gui` with **CTRL+t**)
- Demand Mode (**F3**)
- Vehicle mode (**V**)
- *flow (from-to edges)*
- click origin edge
- click destination edge
- confirm with <Enter>
- Configure flow properties: before creation or later in inspect mode (**I**)
- save with <ctrl+shift+d> (or via *file* menu)
- Edit `osm.sumocfg` to include new file
- `<route-files`
`value="osm.passenger.trips.xml,flow.rou.xml"/>`
- Example data in `03_flow`



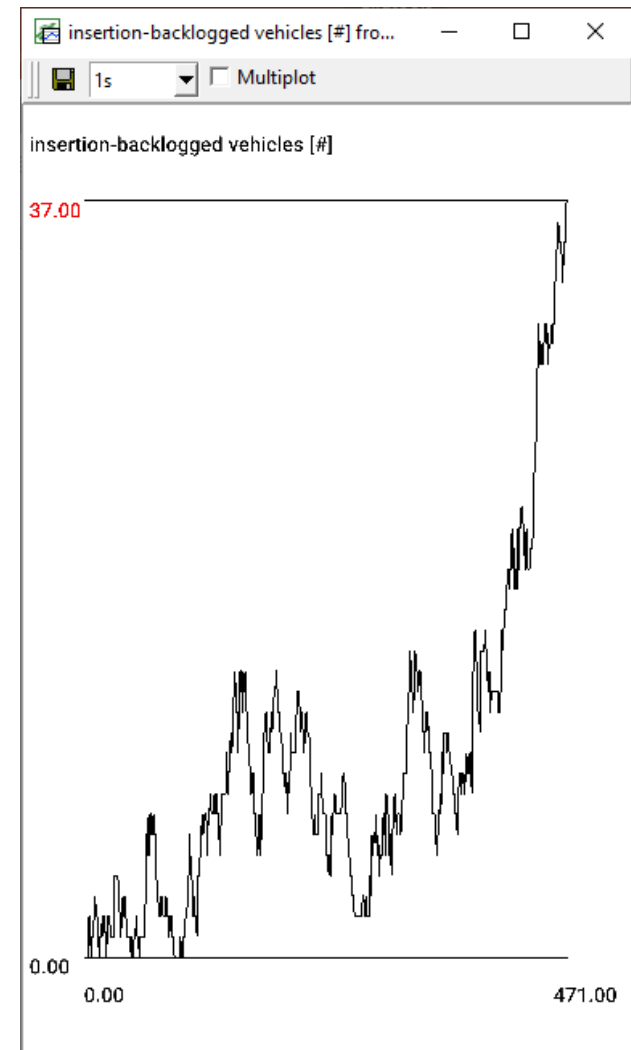
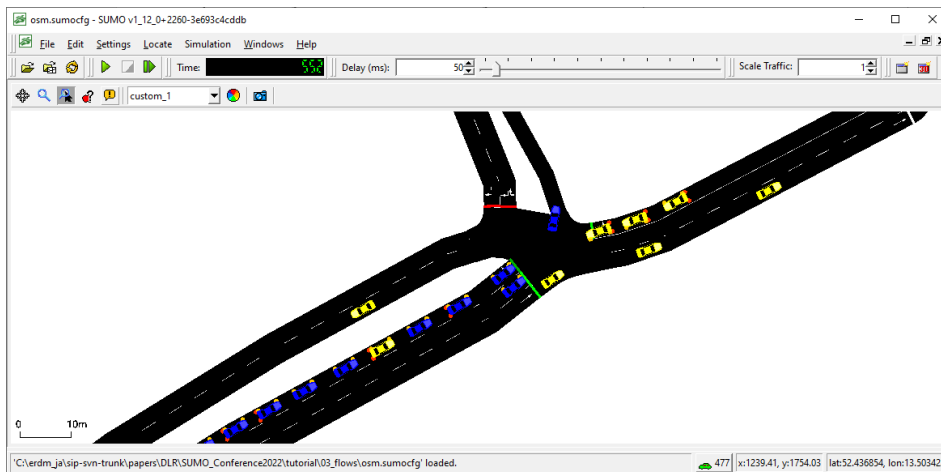
color	blue
departLane	random
departPos	base
departSpeed	avg

Flow attributes	
terminate	end
spacing	poisson
end	3600.00
rate	1.50



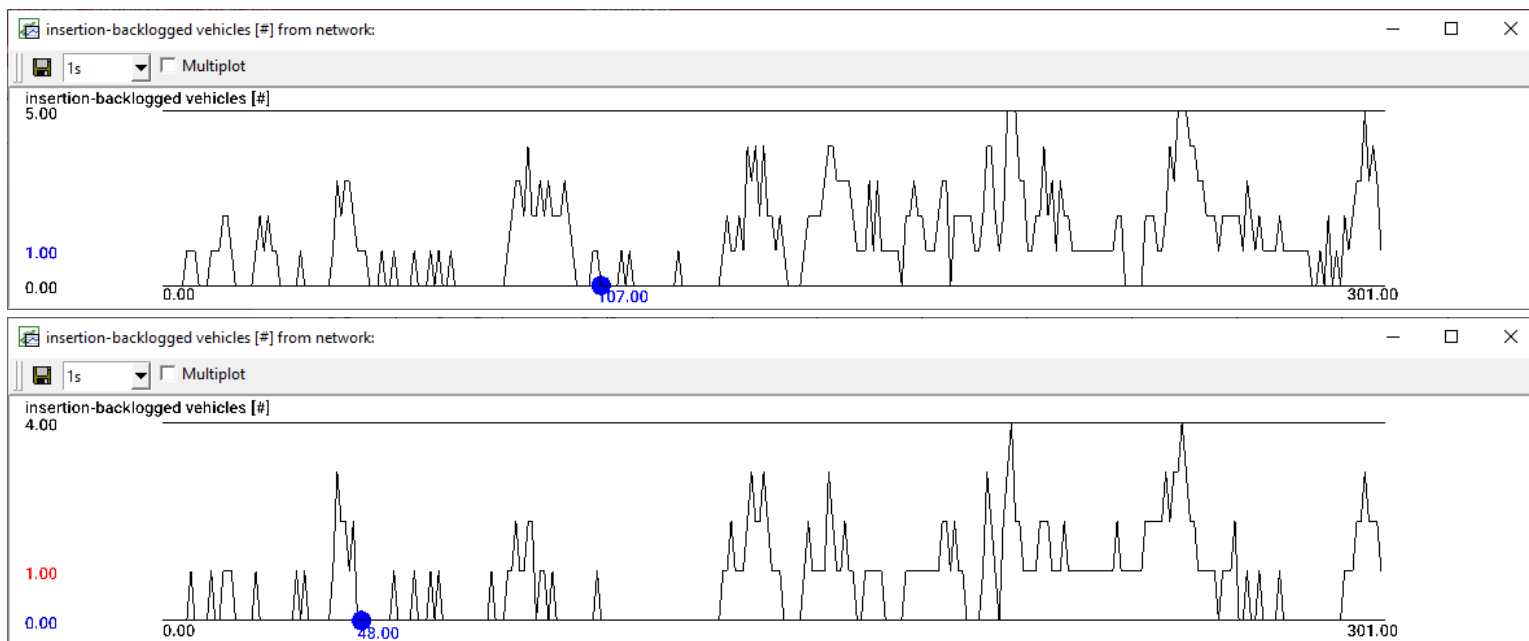
Simulation- Flow

- Exponentially distributed headways between insertions ("arrivals") correspond to a Poisson Process.
- `period="exp(1)"` has an expected rate of 1 veh/s
- Alternations in traffic density - bounded by road capacity!
- insertion-queue affects spacing



Simulation- Flow

- insertion queue first builds due to insufficient insertion capacity, later *explodes* due to road capacity.
- insertion queue smoothens out heterogeneity
- reduce flow: `period="exp(0.7)"`
- increase insertion capacity with sumo option `<extrapolate-departpos value="true">`



Opposite-Direction driving (1)

- Prepare network for opposite driving
- Load network osm.net.xml (open from sumo-gui with **CTRL+t**)
- Options (**F10**)
- Processing -> opposites.guess -> check > OK
- save

- Example data in 04_opposite

Net: lane

Overlapped elements

1 / 2

Help

Internal attributes

id	3234937#0_0
index	0
speed	13.89
allow	pm1 custom2
disallow	rail_fast ship
width	default
endOffset	0.00
acceleration	<input type="checkbox"/> false
customShape	
opposite	3234937#0_0

Configure Options

Input	geometry.min-dist
Output	geometry.max-angle
Processing	geometry.min-radius
Building Defaults	geometry.min-radius.fix <input type="checkbox"/>
TLS Building	geometry.min-radius.fix.railways <input type="checkbox"/>
Ramp Guessing	geometry.junction-mismatch-threshold
Edge Removal	geometry.check-overlap
Unregulated Nodes	geometry.check-overlap.vertical-threshold
Junctions	geometry.avoid-overlap <input checked="" type="checkbox"/>
Pedestrian	join-lanes <input type="checkbox"/>
Bicycle	ptline.match-dist
Railway	ptstop-output.no-bidi <input type="checkbox"/>
Formats	geometry.max-grade
Netedit	geometry.max-grade.fix <input type="checkbox"/>
Visualisation	offset.disable-normalization <input checked="" type="checkbox"/>
Time	offset.x
Report	offset.y
Random Number	offset.z
	flip-y-axis <input type="checkbox"/>
	roundabouts.guess <input checked="" type="checkbox"/>
	roundabouts.visibility-distance
	opposites.guess <input checked="" type="checkbox"/>
	opposites.guess.fix-lengths <input checked="" type="checkbox"/>
	fringe.guess <input type="checkbox"/>
	lefthand <input type="checkbox"/>
	speed.offset
	speed.factor
	speed.minimum
	edges.join-tram-dist

OK



Opposite-Direction driving (2)

- Create some blocking vehicles
 - Demand mode (**F3**),
 - Vehicles (**V**)
 - Create trips, each with a single edge
 - Vehicle Stop mode (**A**)
 - Create a stop for each vehicle
 - save demand (**CTRL+SHIFT+D**) *blockers.rou.xml*
-
- Example data in 04_opposite
 - *osm.sumocfg* loads blockers
 - *2.sumocfg* loads blockers and extra flow

Vehicles

Vehicles

trip (from-to edges)

Parent vType

DEFAULT_VEHTYPE

Internal attributes

id	t_3
color	yellow
departLane	first
departPos	stop

Stops

Parent element

t_0

Stops

stopLane

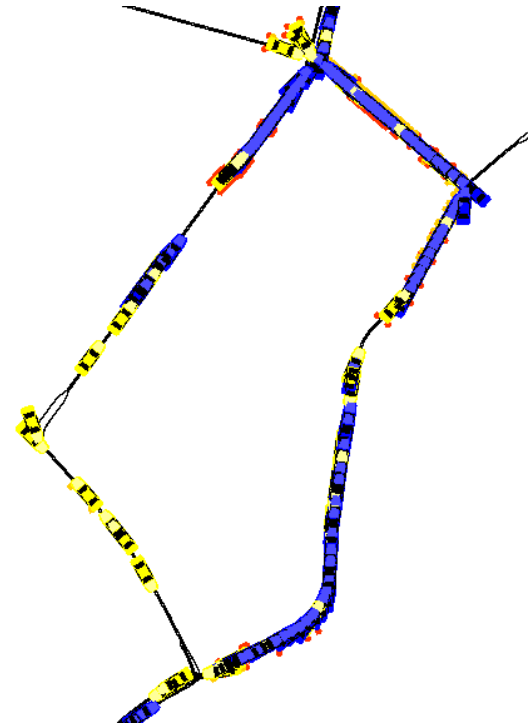
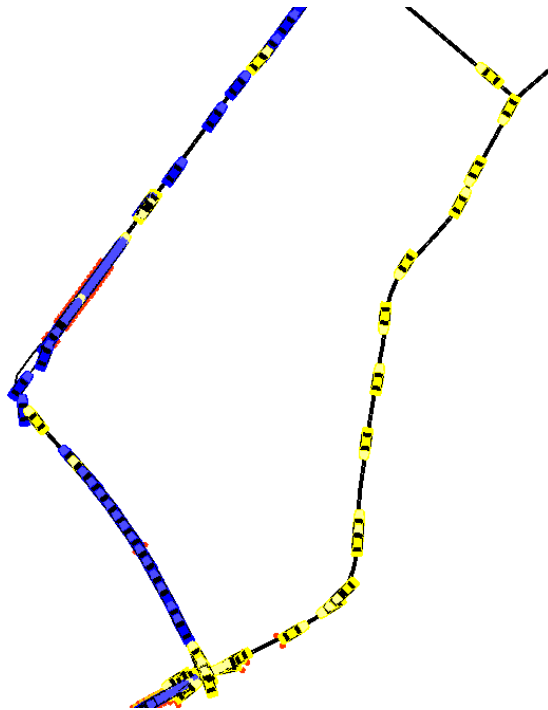
Internal attributes

friendlyPos	<input type="checkbox"/> false
posLat	
<input checked="" type="checkbox"/> duration	3600.00
<input type="checkbox"/> until	
<input type="checkbox"/> extension	



Opposite-Direction driving (3)

Stopped vehicles reduce capacity

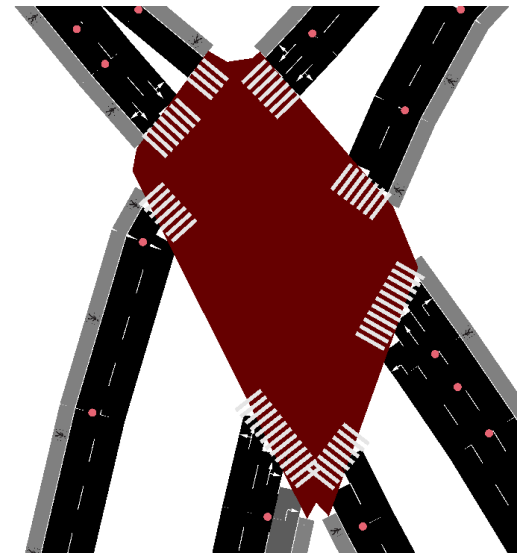
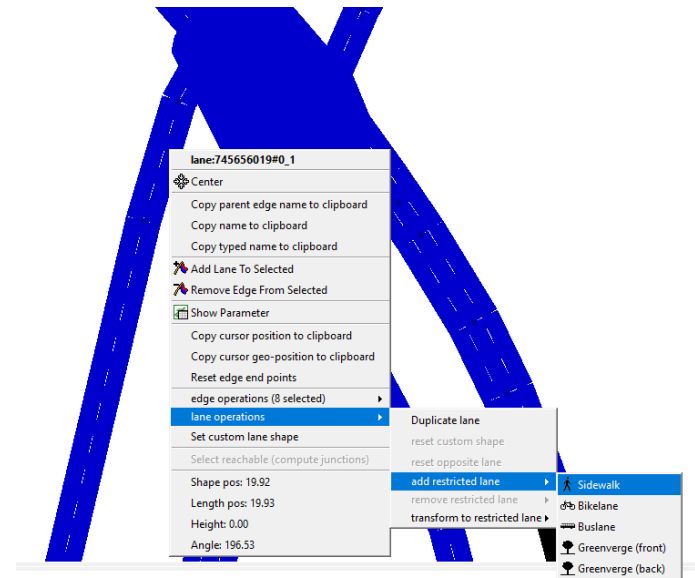


Eventually, traffic reroutes



Pedestrian Crossing (1)

- Select mode (**S**)
- Select edges around junction
- right click edge -> lane operations -> add sidewalk
 - **important: this automatically prohibits walking on road lanes**
- Crossing mode (**R**)
- click junction,
- repeat 8 times:
 - click edge
 - <ENTER>
- Background: To avoid creating "islands", click two edges + <ENTER>



Pedestrian Crossing (2)

- Demand mode (**F3**)
 - Person mode (**P**)
 - personFlow
 - click origin edge
 - click destination edge
 - <ENTER>
 - save with <**CTRL+SHIFT+D**> (or via *File* menu)
-
- Example data in 05_crossing

Persons

Persons

personFlow

Parent vType

DEFAULT_PEDTYPE

Internal attributes

id	f_3
color	yellow
departPos	base
begin	0.00

Help

Flow attributes

terminate	end
spacing	poisson
end	1000.00
rate	0.2

Person plans

personTrip: edge-> edge



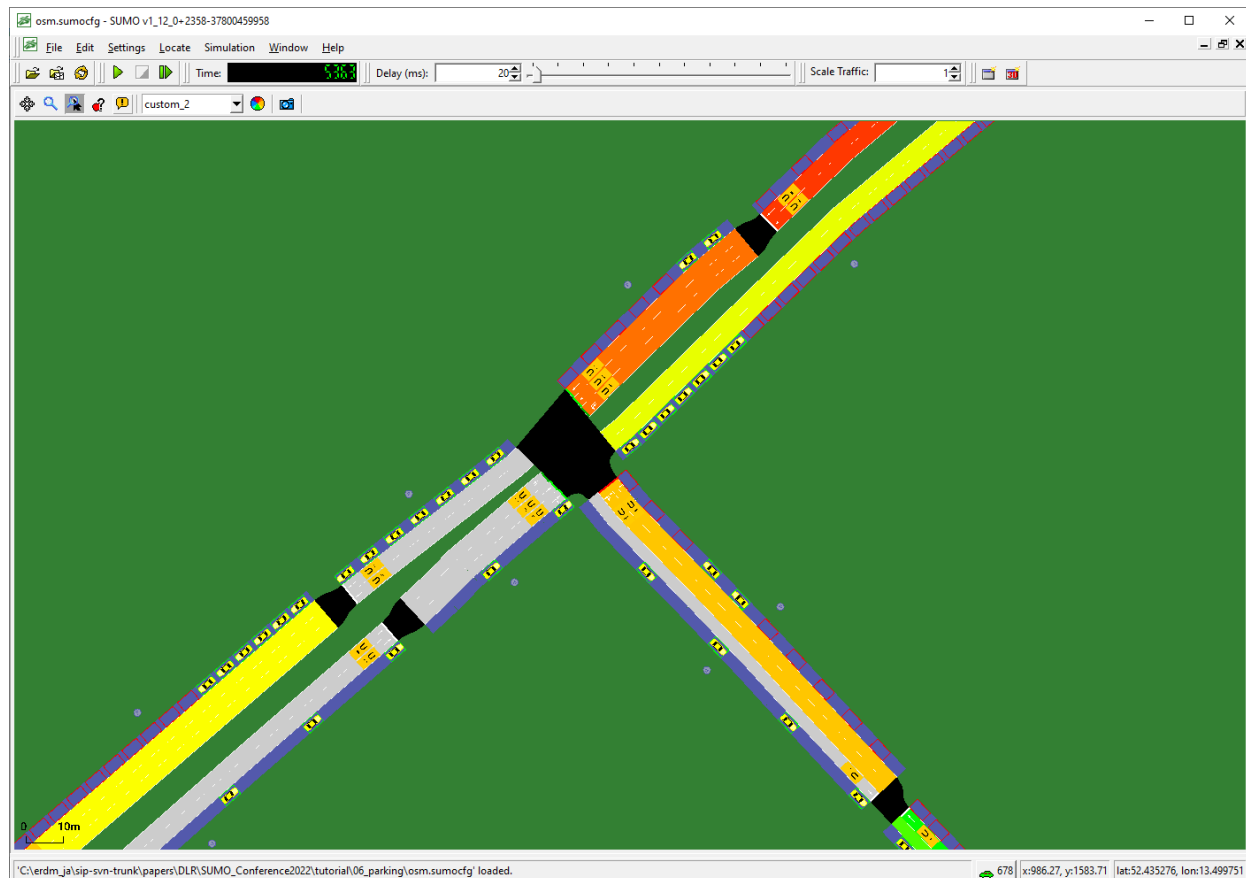
Parking Search - Setup

- add synthetic parkingAreas
generateParkingAreas.py -n osm.net.xml -o parking.add.xml
--random-capacity --keep-all
--edge-type.remove highway.motorway,highway.motorway_link
- every car must park at the end of their route
addStops2Routes.py -n osm.net.xml -r osm.passenger.trips.xml
--parking-areas parking.add.xml -o parking.rou.xml --duration 3600
- add synthetic parking connectivity/visibility information
generateParkingAreaRerouters.py -n osm.net.xml -a parking.add.xml
-o rerouter.add.xml --opposite-visible --max-distance-alternatives 4000
- Example data in 06_parking
 - generate everything with **build.bat**



Parking Search - Simulation

- Whenever a vehicle attempts to park and there is no capacity left, it selects from a list of alternatives (defined in **rerouter.add.xml**)
- Wide range of parameters to configure search strategy



Conclusion

- Use [tools/osmWebWizard.py](#) to get a quick start
 - Read the documentation / FAQ at <http://sumo.dlr.de/docs>
 - Report any bugs you find to sumo-user@eclipse.org
 - Share your scenarios and results
-
- Talks to us. We are always looking for project partners! sumo@dlr.de

