

MobMod Lab 2

Marco Sorbi *EURECOM*
Antibes, France
marco.sorbi@eurecom.fr

I. IMPORTING THE MAP

The Sophia Antipolis map has been imported using the provided `map.sophia.antipolis.osm` and the following configuration file from `netconvert`:

```
<?xml version="1.0" encoding="UTF-8"?>

<configuration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="http://sumo.dlr.de/xsd/netconvertConfiguration.xsd">

  <input>
    <osm-files value="map.sophia.antipolis.osm" synonymes="osm" type="FILE"
      help="Read OSM-network from path &apos;FILE(s)&apos;"/>
  </input>

  <output>
    <output-file value="sophia.net.xml" synonymes="o output sumo-output" type="FILE"
      help="The generated net will be written to FILE"/>
    <ptstop-output value="ptstop.xml" type="FILE"
      help="Writes public transport stops to FILE"/>
    <ptline-output value="ptline.xml" type="FILE"
      help="Writes public transport lines to FILE"/>
  </output>

  <tls_building>
    <tls.default-type value="actuated" type="STR"
      help="TLSs with unspecified type will use STR as their algorithm"/>
  </tls_building>

  <edge_removal>
    <keep-edges.by-vclass value="passenger" type="STR[]"
      help="Only keep edges which allow one of the vclasses in STR[]"/>
    <remove-edges.isolated value="true" synonymes="remove-isolated" type="BOOL"
      help="Removes isolated edges"/>
  </edge_removal>

  <junctions>
    <no-internal-links value="true" type="BOOL" help="Omits internal links"/>
    <no-turnarounds value="true" type="BOOL" help="Disables building turnarounds"/>
  </junctions>

</configuration>
```

The imported map, visible in Fig. 1, is simpler than the one shown in the assignment, visible in Fig. 2.

II. TRAFFIC ASSIGNMENT ZONES

TAZ are shown in Fig. 3. `taz_0` and `taz_1` correspond respectively to `poly_1` and `poly_2` in the assignment, visible in Fig. 2. As the obtained area corresponding to `poly_4` in the assignment has only a few nodes, it has been merged with `poly_3` into `taz_2`.

III. ORIGIN DESTINATION MATRIX

The OD matrix has been modified from the assignment by conveying `poly_3` and `poly_4`'s incoming and outgoing traffic into `taz_2`, by summing their weights. Table I reports the obtained OD matrix.



Fig. 1: Downloaded map

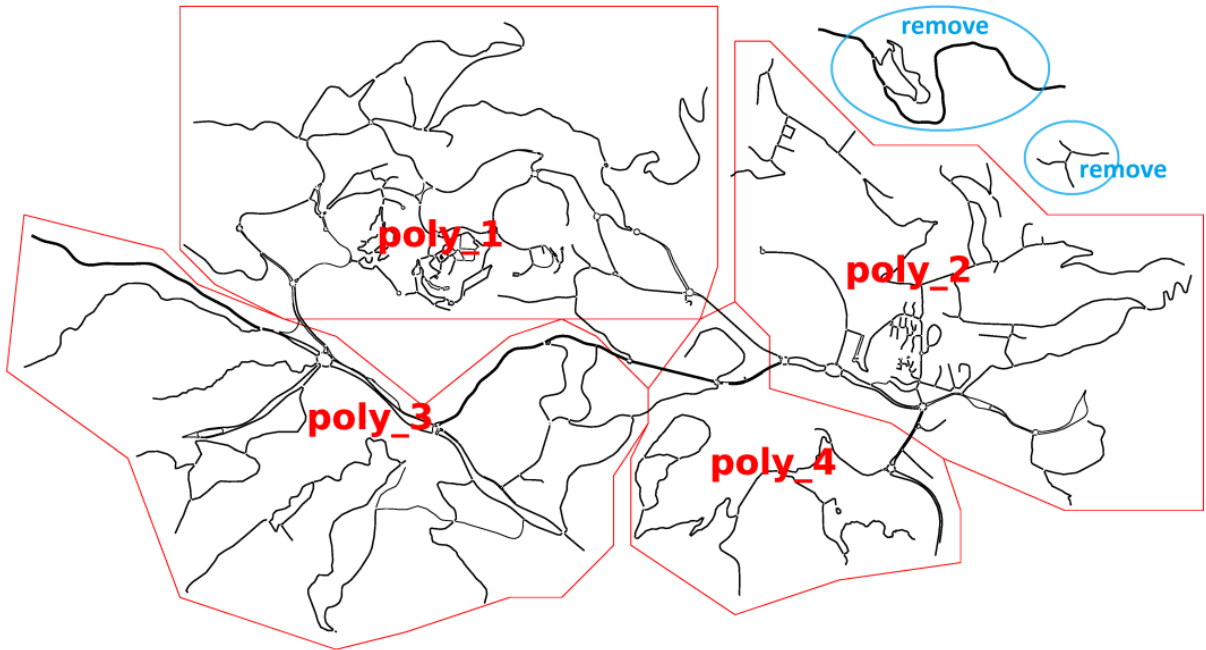


Fig. 2: Assignment map with TAZ

TABLE I: Origin Destination matrix

From TAZ	To TAZ	Vehicles
taz_0	taz_1	0.14
taz_0	taz_2	0.76
taz_1	taz_0	0.86
taz_1	taz_2	1.03
taz_2	taz_0	1.33
taz_2	taz_1	1.33

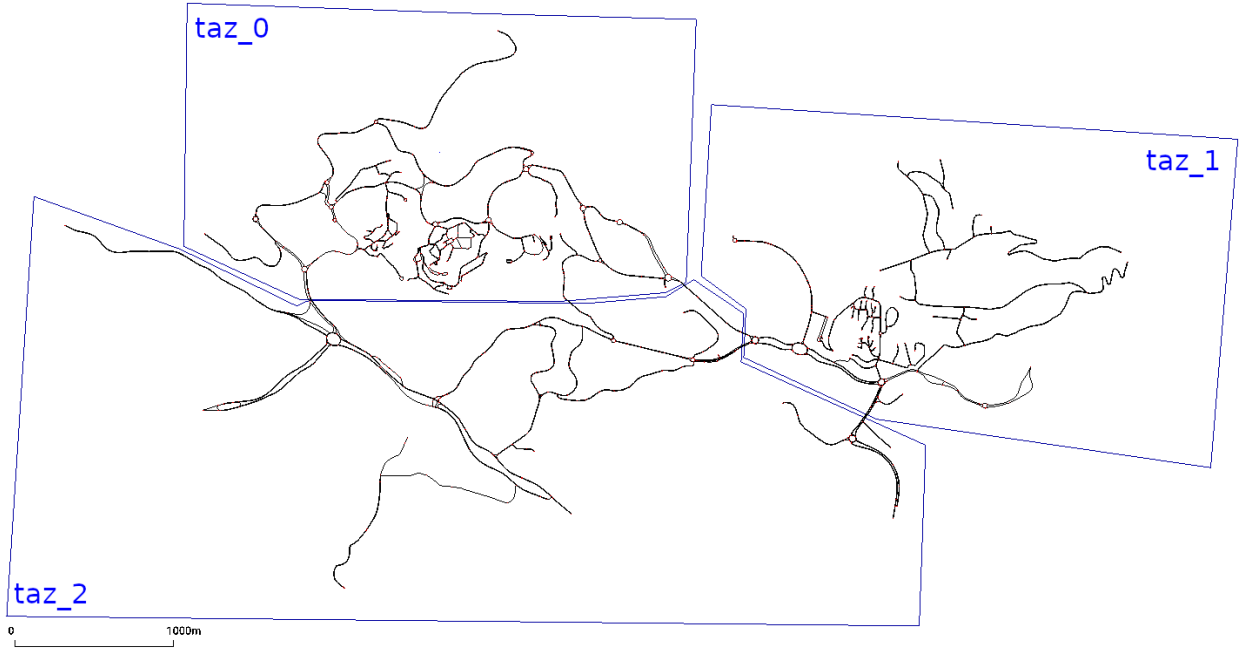


Fig. 3: Downloaded map, cleaned, with TAZ

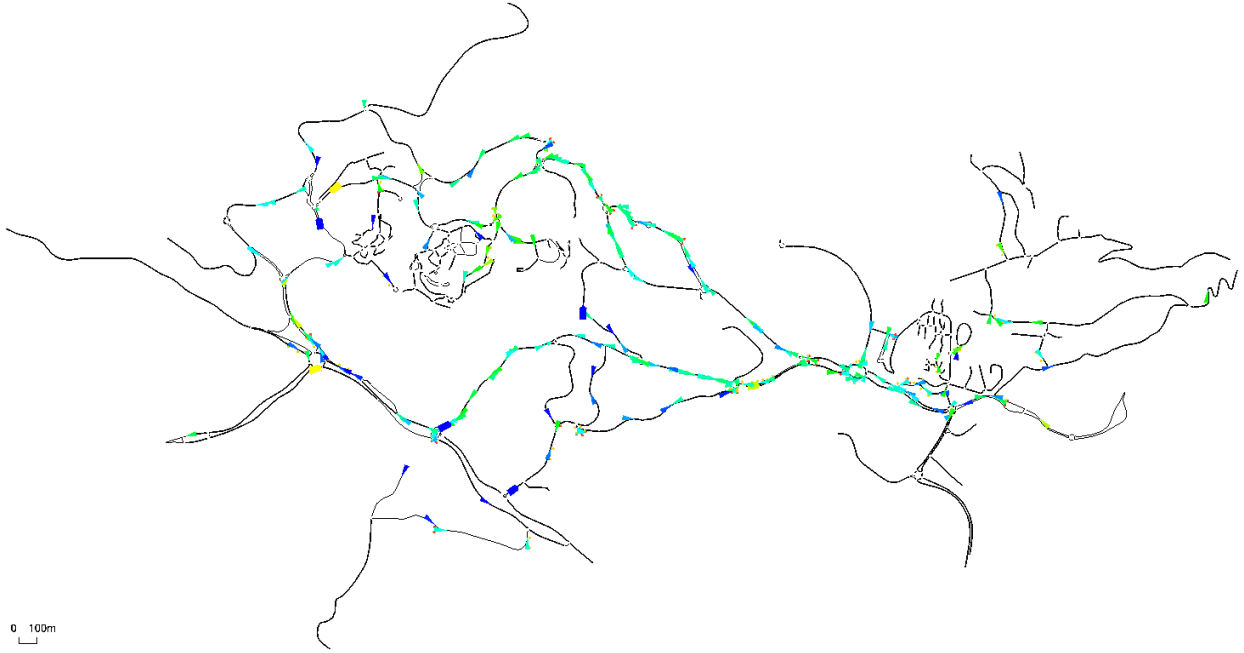
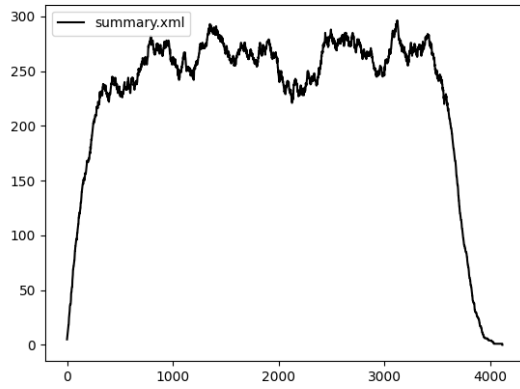


Fig. 4: Running simulation, vehicles colors span from green to red as timeLoss increases

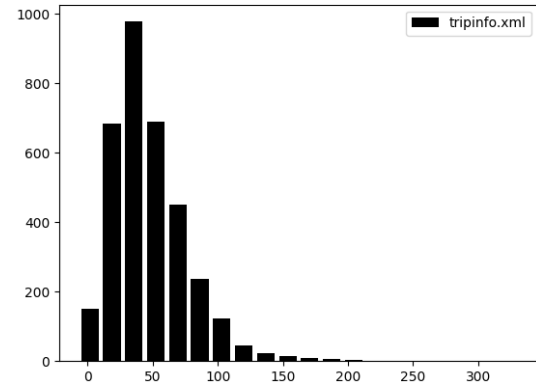
IV. TRAFFIC ANALYSIS

The simulation ran with SUMO does not show visible problems or bottlenecks apart from some short queue somewhere from time to time. A typical simulation step is shown in Fig. 4.

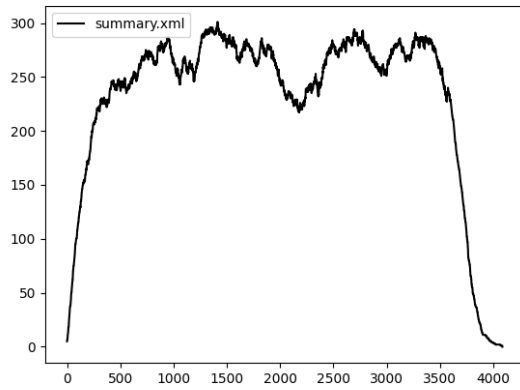
Adding the edges indicated in the assignment and analyzing the traffic, there aren't visible changes to the traffic. To check that there aren't major changes, it is possible to note from the running vehicles and timeLoss plots, in Fig. 5, that adding the new edges achieves only a little decrease in the latter.



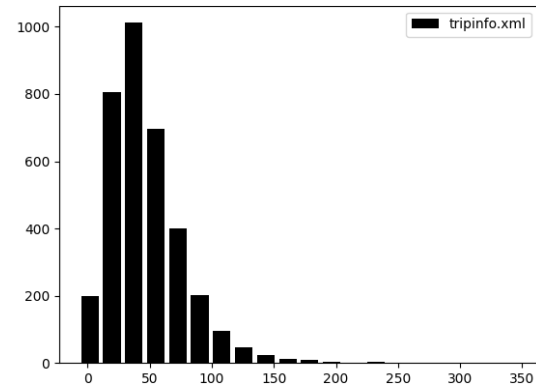
(a) Running vehicles in the original map



(b) Time loss in the original map



(c) Running vehicles in the modified map



(d) Time loss in the modified map

Fig. 5: Traffic analysis results