# Accessing SUMo from MATLAB

Using traci4matlab

# INITIALIZATION

- > Traffic control interface(TraCl), helps us use SUMo through other softwares. For MATLAB it is available as traci4matlab.
- > SUMo versions from 0.20.1 come with in-built traci4matlab. Latest version of SUMo is 0.32.0.
- > First we add SUMO\_HOME as an environment variable in our computer and traci4matlab in our MATLAB path.
- > Then a java path text file has to be created in the preference directory of MATLAB to traci4matlab.jar.
- > After this traci\_test.m can be run to confirm installation.

#### STARTING SUMo from MATLAB

import traci.constants

```
system( [ 'sumo-gui -c ' getenv("SUMO_HOME)...
'\docs\tutorial\traci_tls\data\cross.sumocfg&' ] );
```

## Starting TraCI

```
% Initialize TraCI
traci.init();

traci.inductionloop.subscribe('0');

for i=1:length(steps)

% Perform a simulation step (one second)
traci.simulationStep();

programPointer = min(programPointer+1, length(PROGRAM));
```

# Acessing SUMo variables

The SUMO objects are grouped in thirteen domains: gui, lane, poi, simulation, trafficlights, vehicletype, edge, inductionloop, junction, multientryexit, polygon, route, and vehicle.

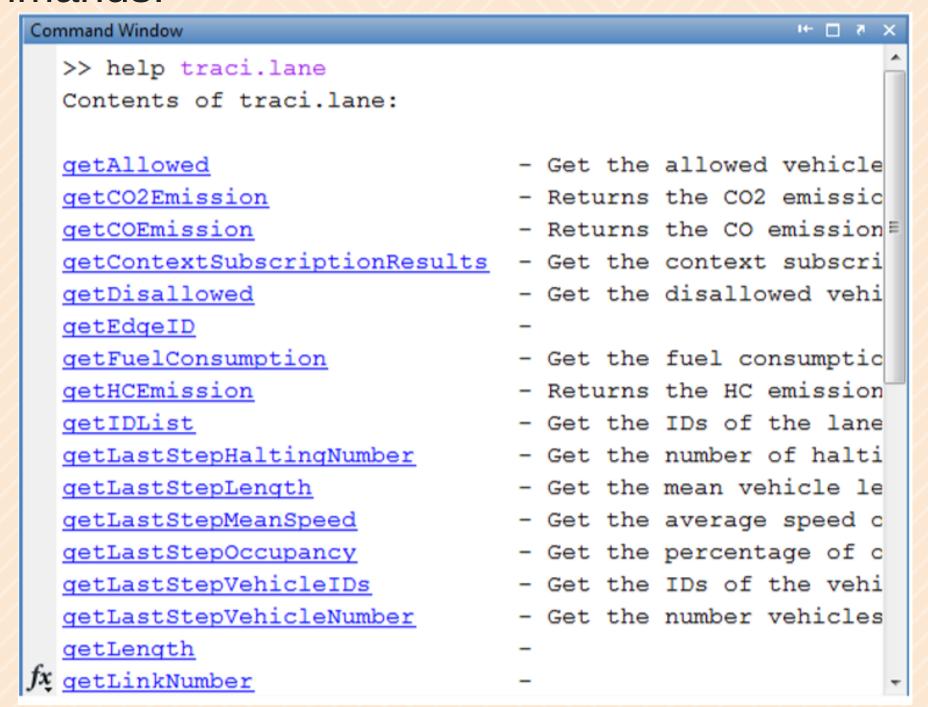
The general structure to access or modify a SUMO object is:

traci.<domain>.<get/set\_wrapper()>,

Where domain can take any of the domains listed previously and get/set\_wrapper() are the functions to acces the values (get) or modify (set) the attributes of the object of interest.

>If the velocity's value of the vehicle with ID veh\_1 in the current time step is required, the command current\_speed\_veh1 = traci.vehicle.getSpeed('veh\_1') can be executed.

#### >Some other commands:



# Getting Results

For example, suppose that it's desired to make a TraCl subscription to acces the values of the attributes LAST\_STEP\_VEHICLE\_NUMBER and LAST\_STEP\_MEAN\_SPEED of the *induction loop* with ID '0'. In this case, the command shown in figure 12 shall be used. Note that the import traci.constants command must be issued at the beginning of the *script*, as explained in the step 1.

```
indloopSubsResults = traci.inductionloop.getSubscriptionResults('0');
no = indloopSubsResults(constants.LAST_STEP_VEHICLE_NUMBER);
lsms = indloopSubsResults(constants.LAST_STEP_VEHICLE_NUMBER);
lsms = indloopSubsResults(constants.LAST_STEP_MEAN_SPEED);
```

### Closing TraCI

traci.close()