SUMO

Simulation of Urban MObility

computer simulation
Winter semester 1401-1402

Table of Contents

Introduction

"Simulation of Urban MObility", or "SUMO" for short, is an open source, microscopic, multi-modal traffic simulation. It allows to simulate how a given traffic demand which consists of single vehicles moves through a given road network. The simulation allows to address a large set of traffic management topics. It is purely microscopic: each vehicle is modelled explicitly, has an own route, and moves individually through the network. Simulations are deterministic by default but there are various options for introducing randomness.

Features

- Includes all applications needed to prepare and perform a traffic simulation (network and routes import, DUA, simulation)

- Simulation
  - Space-continuous and time-discrete vehicle movement
  - Different vehicle types
  - Multi-lane streets with lane changing
  - Different right-of-way rules, traffic lights
  - A fast openGL graphical user interface
  - Manages networks with several 10.000 edges (streets)
  - Fast execution speed (up to 100.000 vehicle updates/s on a 1GHz machine)
  - Interoperability with other application at run-time
  - Network-wide, edge-based, vehicle-based, and detector-based outputs
  - Supports person-based inter-modal trips

- Network Import
  - Imports VISUM, Vissim, Shapefiles, OSM, RoboCup, MATsim, OpenDRIVE, and XML-Descriptions
  - Missing values are determined via heuristics

- Routing
  - Microscopic routes - each vehicle has an own one
  - Different Dynamic User Assignment algorithms

- High portability
  - Only standard C++ and portable libraries are used
  - Packages for Windows main Linux distributions exist

History

The development of SUMO started in the year 2000. The major reason for the development of an open source, microscopic road traffic simulation was to support the traffic research community with a tool with the ability to implement and evaluate own algorithms. The tool has no need for regarding all the needed things for obtaining a complete traffic simulation such as implementing and/or setting up methods for dealing with road networks, demand, and traffic controls. By supplying such a tool, the DLR wanted to i) make the implemented algorithms more comparable by using a common architecture and model base, and ii) gain additional help from other contributors.

Software design criteria

Two major design goals are approached: the software shall be fast and it shall be portable. Due to this, the very first versions were developed to be run from the command line only - no graphical interface was supplied at first and all parameter had to be inserted by hand. This should increase the execution speed by leaving off slow visualization. Also, due to these goals, the software was split into several parts. Each of them has a certain purpose and must be run individually. This is something that makes SUMO different to other simulation packages where, for instance, the dynamical user assignment is made within the simulation itself, not via an external application like here. This split allows an easier extension of each of the applications within the package because each is smaller than a monolithic application that does everything. Also, it allows the usage of faster data structures, each adjusted to the current purpose, instead of using complicated and ballast-loaded ones. Still, this makes the usage of SUMO a little bit uncomfortable in comparison to other simulation packages. As there are still other things to do, we are not thinking of a redesign towards an integrated approach by now.

Traffic Management and Other Structures

- Traffic Lights
- Public Transport
- Variable Speed Signs
- Rerouter / Alternative Route Signage
- Vaporizer (deprecated, use Calibrator instead)
- Dynamic calibration of flow and speed and type
- Parking areas
- Turnarounds

Model details

- Vehicle speed
- Vehicle insertion
- Vehicle permissions (access restrictions)
- Road capacity
- Intersection dynamics
- Randomness
- Routing and Re-routing
- Sublane Model
- Opposite Direction Driving
- Safety
- Mesoscopic model
- Lengths and Distances

Traffic modes

- Pedestrian simulation
- Bicycle simulation
- Railway simulation
- Waterway simulation

Output

sumo allows to generate a large number of different measures. All write the values they collect into files or a socket connection following the common rules for writing files. Per default, all are disabled, and have to be triggered individually. Some of the available outputs (raw vehicle positions dump, trip information, vehicle routes information, and simulation state statistics) are triggered using command line options, the others have to be defined within *additional-files*.

All output files written by SUMO are in XML-format by default. However, with the python tool xml2csv.py you can convert any of them to a flat-file (CSV) format which can be opened with most spread-sheet software. If you need a more compressed but still "standardized" binary version, you can use xml2protobuf.py. Furthermore all files can be written and read in compressed form (gzip) which is triggered by the file extension .gz.

By default, sumo will print some "heartbeat" information to indicate that it is still running. The following information will be printed every 100 simulation steps:

- Step #: current simulation time
- duration of the latest step in (ms)
- real-time factor (step-length / duration). (RT)
- number of vehicles updated per second (UPS)
- TraCI: time spent with TraCI processing in the current step (including external script)
- vehicles TOT: number of vehicles that departed so far
- ACT: number of currently running vehicles
- BUF: number of vehicles with delayed insertion

This output can be disabled with the option --no-step-log. It's period can be configured with the option --step-log.period TIME.

Thank you for your cooperation

- Samaneh Naghdbishi
- Fatemeh Akbari
- Amirhossein Shateri
- Sajjad Ranjbar