INTELLIGENT TRANSPORTATION SYSTEMS:

LECTURE 9:
SIMULATING URBAN MOBILITY

Amnir Hadachi, PhD.
hadachi@ut.ee
Fall 2020
SIMULATING URBAN MOBILITY

(INTRO)

- Reasons for an open-source traffic simulation
- Needed models, data, and its processing
- SUMO Overview
SIMULATING URBAN MOBILITY
(OPEN SOURCE ROAD TRAFFIC SIMULATION & APPLICATIONS)

• Reproducible, computerised evaluation of
  – real-world network performance
  – new concepts for
    • modern traffic signal control
    • traffic surveillance
    • traffic forecasting
    • traffic management
    • dynamic routing methods
    • car2car/car2infrastructure communication
  – traffic models
• Instructive visualisation
• Planning
SIMULATING URBAN MOBILITY
(OPEN SOURCE ROAD TRAFFIC SIMULATION & APPLICATIONS)

road network

vehicles / flow

signalling
Why an open source road traffic simulation?

Common procedure

• An academic organization wants to evaluate an aspect of traffic
• Builds an own traffic simulation; needs
  – A network representation including speed limits, right-of-way – rules, etc.
  – A representation of vehicles, their movement, routes, etc.
  – A representation of signals, variable message signs, etc.
... yields in
Many (incomplete) simulations which results can not be compared

→ **Solution:** an extendable simulation as a base for own development
SIMULATING URBAN MOBILITY

(OPEN SOURCE ROAD TRAFFIC SIMULATION & APPLICATIONS)

SUMO

• Portability
  —Achieved by using c++ and portable libraries only

• Performance
  —High execution speed
  —No limitations in network size and number of simulated vehicles

• Extendibility

• Open source
  —Licensed under GPL
  —Hosted at sourceforge (http://sumo.sourceforge.net)

• Microscopic
  —Each vehicle is modelled explicitly
SIMULATING URBAN MOBILITY
(OPEN SOURCE ROAD TRAFFIC SIMULATION & APPLICATIONS)

Road Traffic Flow Dynamics (one of):

Fundamental diagram of traffic (counted through induction loops)

As interpreted by Kerner

As interpreted by Kim and Keller
Road Traffic Flow Simulation Classes:

- Macroscopic
- Microscopic
- Sub-microscopic
- Mesoscopic
Mostly:

- discrete in time
- “Car Following Models”: vehicle’s speed depends on the preceding vehicle

\[ d(v_l) + g \geq d(v_f) + v_f \tau \]
SIMULATING URBAN MOBILITY
(OPEN SOURCE ROAD TRAFFIC SIMULATION & APPLICATIONS)

Car Following Model by Krauß*

Features:
• continuous in space
• discrete in time
• accident-free
• stochastic driver model

Parameters:
• acceleration $a$
• deceleration $b$
• max. speed $v_{\text{max}}$
• driver’s imperfection $\epsilon$

\[
v_{\text{safe}}(t) = v_l(t) + \frac{g(t) - v_l(t)\tau}{\frac{v}{b(v)} + \tau}
\]

\[
v_{\text{des}}(t) = \min\{v_{\text{safe}}(t), v(t-1) + a, v_{\text{max}}\}
\]

\[
v(t) = \max\{0, \text{rand}[v_{\text{des}}(t) - \epsilon a, v_{\text{des}}(t)]\}
\]

Further Models needed:

- Lane Changing
  - Navigation (I have to turn left on next junction)
  - Tactical (left lane allows me to move faster)
- Interaction with the network
  - Stops at red traffic lights
  - Decelerate if other vehicles have a higher right to move on the next junction
- Extensions
  - Vehicle classes, such as busses, which use own lanes
  - Bus stops
Microscopic Road Networks: Needed vs. given Information

Needed street attributes:
- Number of lanes
- Allowed speed
- Per-lane restrictions
- Allowed continuations

Needed junction attributes:
- Right-of-way
- Traffic lights program

Given: a graph with only few information
- Number of lanes (often vague)
- Allowed speed (often vague)
Microscopic Road Networks: Size and Complexity

Additional problems:
- very large networks
- complex junctions
Microscopic Road Networks: SUMO Solution: Heuristics for automatic computation

Example: computation of lane-to-lane connections:
1. for each edge: compute turnaround edges
2. for each node: sort each node’s edges
3. for each node: compute each node’s type
4. for each node: set edge priorities
5. for each edge: compute edge-to-edge connections
6. for each edge: compute lanes-to-edge connections
7. for each node: compute lane-to-lane connections
8. for each edge: recheck lanes
9. for each edge: append turnarounds
Traffic Demand: Needed Information

Each vehicle is modeled explicitly

Mandatory attributes:
– ID (name)
– Vehicle type (including Krauß-parameter)
– Complete route through the network
– Depart time

Optional attributes:
– stops

```xml
<vehicle id="bus100_west_0d" type="BUS" depart="0" color="1,0,1">
  <route>-572658025 -572658026 -572658027 -572658024 ...
  </route>
  <stop bus_stop="west1" duration="20"/>
  <stop bus_stop="west2" duration="20"/>
</vehicle>
```
Traffic Demand: Possible Sources

- Real (physical) measures are done at induction loops
  - Accurate but sparse
  - No information about who is driving from which place to which place, only how many vehicles have passed a certain place

- Counts at junctions may provide flows and turning ratios
  - Less accurate than induction loops and sparse, too – mainly for some major inner-city junctions
  - Information about flow spread on junctions

- Traffic scientists estimate demands on district level
  - Even less accurate (estimated), but cover a complete area
Traffic Demand Possible Sources

- Real (physical) measures are done at induction loops
  - Induction-loop measures → DFROUTER → SUMO-routes

- Counts at junctions may provide flows and turning ratios
  - Turning ratios → JTRROUTER → SUMO-routes
  - Flows

- Traffic scientists estimate demands on district level
  - OD-matrix → OD2TRIPS → DUAROUTER → SUMO-routes
  - Trips
SUMO - Modules: Included Applications

- **SUMO**: command line simulation
- **GUI SIM**: simulation with a graphical user interface
- **NETCONVERT**: network importer
- **NETGEN**: network generator
- **OD2TRIPS**: converter for O/D-matrices
- **JTRROUTER**: router based on turning ratios
- **DUAROUTER**: router based on a dynamic user assignment
- **DFROUTER**: router which uses detector data
SUMO Features

- Microscopic: all vehicles are modeled explicitly
- Time-discrete, space-continuous car-following model by S. Krauß
- Multi-lane traffic, right-of-way rules
- Around 100,000 cars in real-time (without graphical output)
- Traffic lights with time schedules, other traffic management devices
- Complex networks / Import support
- Dynamic Routing based on Dynamic User Assignment
- Other routing modules

SIMULATING URBAN MOBILITY
(OPEN SOURCE ROAD TRAFFIC SIMULATION & APPLICATIONS)

Road traffic simulation and CO$_2$ Emission

- Road traffic simulation using SUMO Tool
  - Simulating urban mobility dynamic
    - Vehicles
      - Private
      - Public transport
    - Pedestrians
    - Railway
  - Simulating vehicular mobile devices’ interactions
  - CO$_2$ Emission Models
    - Handbook of Emission Factors for Road Traffic Model (HBEFA) - Versions 2.1 & 3.1
      - Vehicle category: passenger cars, light duty vehicles, heavy duty vehicles, buses, coaches and motorcycles
      - Pollutants covered - CO$_2$, CO, HC, NO$_x$, PM$_x$
    - Passenger car and Heavy duty Emission Model (PHEMlight)
      - Pollutants covered - CO$_2$, CO, HC, NO$_x$, PM$_x$
SIMULATING URBAN MOBILITY

(PROJECTS & APPLICATIONS)
Projects at the DLR where SUMO was used: 2002-2006

- INVENT
  Implementation and verification of traffic management strategies for large urban areas
- OIS
  Verification of benefits arising from usage of new optical sensors
- Traffic Tower
  Virtual Traffic Management Environment
- WJT2005 / Soccer2006
  Integration of standard and airborne detectors into a traffic portal with forecast functionality
- TrafficOnline
  Traffic surveillance via in-vehicle GSM phones
SIMULATING URBAN MOBILITY
(PROJECTS & APPLICATIONS)

• Used in the city of Cologne during
  – The pope’s visit (world youth day 2005)
  – The world soccer cup 2006
• Traffic surveillance using
  – Highway induction loops
  – Inner-city induction loops
  – Airborne traffic recognition system
    (mounted on a zeppelin)
• Traffic visualization
  – Integration of gained information into a viewer
  – Presentation for the police
• Traffic forecast
  – 30 min into the future using a simulation (extended SUMO)
DELPHI: System Overview

**Simulation:**
- Areal extrapolation of measured data
- Forecast generation (30min)
- Fed from db using python

**Datenbank:**
- MySQL with InnoDB tables
- No built-in logic

**Aggregation/Correction:**
- Guessing missing values
- Data fusion
- Done using python scripts

**Webserver:**
- Tomcat Servlet Container
- Visualization (Images and traffic)

**Webbrowser:**
- JavaScript Browser
- Asynchronous XML-Requests

**ARGOS:**
Access via db, webserver

**Induction loops:**

**Floating Car Data:**
SUMO – Example Results
Flow Comparison

Flow [veh/h] vs. Time [t [h]]

- Chart 1: Time series of flow with peaks during rush hours.
- Chart 2: Comparison of flow under different scenarios.
Case Study

Introducing Cellular Network Layer into SUMO for Simulating Vehicular Mobile Devices’ Interactions in Urban Environment

- Motivation:
  - Constraints on accessing the mobile users related data
  - Producing the logs - enabling new possibilities in the research field.
  - Vehicle-To-Everything (V2X) communication

SIMULATING URBAN MOBILITY
(PROJECTS & APPLICATIONS)

Cellular Network

- Base station
- Base station transmitter
- Cells - signal coverage
- User equipment - mobile
Mobility Data

- Mobile technologies 1G to 5G
- Mobility data:
  - CDR - like logs
  - GPS data
  - RFID
  - Wireless access point related logs
  - etc
Challenges in analysing CDR logs

- Sparse Events
- Unreliable Coverage Map
- Changes in Coverage Map
- Features of 2G/3G/4G technologies
SIMULATING URBAN MOBILITY
(PROJECTS & APPLICATIONS)

Contribution

- Research of the microscopic road traffic simulator and cellular network technologies to design and model cellular network

- Integration of the cellular network entities into the road traffic simulation software

- Running command-line and visualise traffic simulation with integrated cellular network simulation
SIMULATING URBAN MOBILITY
(PROJECTS & APPLICATIONS)

Contribution
SIMULATING URBAN MOBILITY
(PROJECTS & APPLICATIONS)

Contribution: Support programs

- HexagonGen
- Mobility Event Simulation Generator - MESGEN
SIMULATING URBAN MOBILITY
(PROJECTS & APPLICATIONS)

Contribution: Classes

- GUI and microsim:
- Cellular antenna
- Signal propagation

\[ P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L} \]

- Cellular tower
- Cellular tower controller
- Mobile and GPS devices
- Mobile Event Data
- Data import and export
SIMULATING URBAN MOBILITY
(PROJECTS & APPLICATIONS)

Results: GPS

- Mobile mobility logs
  - GPS data
SIMULATING URBAN MOBILITY
(PROJECTS & APPLICATIONS)

Results: CDR

- Mobile mobility logs
  - CDR-like
- Event types:
  - Call initiation
  - Call receiving
  - SMS sending
  - SMS receiving
  - Web communication

```xml
<mobile-output>
    <vehicle id="16">
        <event timestamp="16.00" eventType="SMS_REC" instle="16" cgl="t-571-1"
        signalStrength="27.30" latitude="58.28" longitude="26.52935928" speed="0.080000000"/>
    </vehicle>
    <vehicle id="91">
        <event timestamp="91.00" eventType="CALL_INIT" instle="91" cgl="t-693-2"
        signalStrength="24.43858123" latitude="58.27194942" longitude="26.53703835" speed="0.000000000"/>
    </vehicle>
    <event timestamp="92.00" eventType="CALL_INIT" instle="91" cgl="t-693-2"
        signalStrength="24.40692827" latitude="58.27194625" longitude="26.53706682" speed="0.000000000"/>
    <event timestamp="93.00" eventType="CALL_INIT" instle="91" cgl="t-693-2"
        signalStrength="24.32271960" latitude="58.27193870" longitude="26.53713112" speed="3.93172514"/>
    <event timestamp="94.00" eventType="CALL_INIT" instle="91" cgl="t-693-2"
        signalStrength="24.22708629" latitude="58.27192817" longitude="26.53722397" speed="5.5497000001"/>
    <event timestamp="95.00" eventType="CALL_INIT" instle="91" cgl="t-693-2"
        signalStrength="24.09187215" latitude="58.27191446" longitude="26.53734378" speed="7.19185444"/>
    <event timestamp="96.00" eventType="CALL_INIT" instle="91" cgl="t-693-2"
        signalStrength="24.91395039" latitude="58.27189622" longitude="26.53750321" speed="9.57843992"/>
    <event timestamp="97.00" eventType="CALL_INIT" instle="91" cgl="t-693-2"
        signalStrength="23.69217562" latitude="58.27187303" longitude="26.53778074" speed="12.15771377"/>
    <event timestamp="98.00" eventType="CALL_INIT" instle="91" cgl="t-693-2"
        signalStrength="23.46155188" latitude="58.27184022" longitude="26.53792255" speed="13.51444867"/>
    <event timestamp="99.00" eventType="CALL_INIT" instle="91" cgl="t-693-2"
        signalStrength="23.24320905" latitude="58.27182381" longitude="26.53813582" speed="12.80231273"/>
</mobile-output>
```
SUMO
TASKS TO DO

• Check the readings section in course webpage

• Download and install the right version of sumo on your computer.

• Try to run it to be ready for the lab session
THANK YOU FOR YOUR ATTENTION

― Intelligent Transportation Systems - MTAT.08.040 - Lecture 9