

MATSim Tutorial

Using the Deterministic Event-driven Queue-based Traffic Flow Simulation (DEQSim) with MATSim

Release 2007-05-16

Content

Introduction
Running DEQSim Stand-Alone
Using DEQSim Integrated in MATSim

Introduction

The Deterministic Event-driven Queue-based Traffic Flow Micro-Simulation (DEQSim) is a fast traffic simulation written in C++. It can read the usual MATSim XML files for input, which allows it to be integrated in MATSim¹.

This tutorial will show you how you can use DEQSim together with MATSim.

DEQSim is written in C++. You will need at least a C++-Compiler to get it running. We provide a Makefile to use with cmake², which allows you to compile DEQSim on most major operating systems. DEQSim requires the expat-library for XML parsing, which must be available at compile time. See the file README.TXT for more information about compiling DEQSim.

To integrate DEQSim in MATSim, you need MATSim, Release 2007-05-16 or newer. MATSim requires the Java Runtime Environment (JRE) 5 or newer.

If you are new to MATSim, we suggest that you start first with the tutorial "Getting Started with MATSim" to get a glimpse what MATSim is about and to get a better understanding on how DEQSim interacts with MATSim.

Running DEQSim Stand-Alone

DEQSim can read XML files generated and used by MATSim. If you have a network and some plans, you could use DEQSim to simulate the traffic generated by agents following the plans. You could again run the equil-scenario, which you may remember from the "Getting Started"-Tutorial. You find all the required input files in the directory `data`.

If you have built DEQSim according to the description in the file README.TXT, you can run the simulation with:

```
./build/microsim/microsim ./configs/deqsim_standalone.xml
```

This will generate the files (among others) `deq_event.txt` and `loads_out.txt`. `deq_events.txt` contains the events that the micro-simulation generated and which can be read by MATSim. `loads_out.txt` gives you a list of all nodes and the number of events for each of those nodes. Because there are two events per node and vehicle (a vehicle wants to cross the node and actually crosses it), the numbers are always double the number of vehicles that traveled across those nodes. The microsimulation does **not** create any output that can be immediately visualized.

¹ <http://www.matsim.org/>

² <http://www.cmake.org/>

If you use DEQSim with XML plans, make sure that all legs have a departure time set: DEQSim only reads the departure time of legs and not end-time or duration of activities.

Using DEQSim Integrated in MATSim

It is possible to use DEQSim as traffic simulation for MATSim. In this case, you can start your simulation runs with MATSim, configure the re-planning of the agents and so on. Use the configuration file `./configs/deqsim_matsim.xml` together with MATSim to use DEQSim together with MATSim. Make sure to set the correct path to the executable `microsim` in the configuration file. Then you can run MATSim together with DEQSim with:

```
java -cp MATSim.jar org.matsim.demandmodeling.controller.DEQSimController
    ./configs/deqsim_matsim.xml
```

In this case, plans and events are written in a binary format. Parsing large XML files, combined with the transformation of numeric or time values to string and back takes large amounts of processing power. To speed up the data exchange between MATSim and the mobility simulation DEQSim, a binary data format was specified for temporal exchange data.