The Zurich Scenario: A Technical Overview 1.0

Andreas Horni
Basil J. Vitins

Working paper
Transport and Spatial Planning
December 2011
Contents

1 Introduction 2

2 Input Data 3

2.1 Population and Initial Demand 3

2.1.1 Switzerland Scenario 3

2.1.2 Cross-Border Traffic Demand 5

2.1.3 Freight Traffic Demand 5

2.1.4 Zurich Demand 6

2.2 Networks 6

2.2.1 Road Networks 7

2.2.2 Public Transport Networks 7

2.3 Facilities 8

2.4 Public Transport Stops and Schedules 8

2.5 Count Data 9

2.6 Traffic Lights 9

2.7 Parking 9

3 Modules and Functionality 11

3.1 Network Loading Simulations a.k.a. Mobility Simulations (mobsims) 11

3.1.1 queueSimulation 11

3.1.2 QSim 11

3.1.3 DEQSim (deprecated) 11

3.1.4 JDEQSim 11

3.1.5 The Coming Mobsim (working title senozim) 12

3.2 Multi-Modal Simulation 12
3.2.1 Public transport ................................................. 12
3.2.2 Walk and Bike ................................................. 12
3.2.3 Ride ............................................................... 13
3.2.4 Mode Choice on Tour and Sub-Tour Level ................. 13
3.2.5 Mobility Tools .................................................. 13
3.3 Replanning Modules .............................................. 13
3.4 Analysis Tools ..................................................... 14
3.4.1 Visualizers ....................................................... 14
3.4.2 Additional Tools ................................................ 14

4 Large-Scale Projects Based on Switzerland or Zurich Scenario 15

4.1 Westumfahrung ..................................................... 15
4.1.1 Performance ..................................................... 15
4.2 KTI Frequencies ................................................... 16
4.2.1 Performance ..................................................... 16
4.3 Herbie ............................................................... 16
4.3.1 Performance ..................................................... 17

5 Configuration .......................................................... 18

5.1 Running a large Scenario ......................................... 18
5.1.1 Access to servers .............................................. 18
5.1.2 Required Resources .......................................... 18
5.1.3 Background Processes ....................................... 18
5.1.4 Improving Performance ..................................... 18
5.2 Configuration File Settings ...................................... 19
5.2.1 Choosing a Mobsim ......................................... 19
Abstract

The MATSim Zurich scenario is frequently used and well calibrated and validated. It has been applied in numerous projects and serves as real-world example for research. It is generated by extracting the Zurich region from the Switzerland scenario. This report is designed as a read-me file. In combination with the referenced papers the reader should be able to run the Zurich scenario autonomously. A brief overview is given of input data, central MATSim modules (such as the mobility simulations), prior large-scale projects and common configurations.

The document is primarily written for internal usage as data used are not open to the public. It is planned to continuously update this report and periodically publish a snapshot of it. Hints for running other scenarios are also given.

Keywords
MATSim, Switzerland, Demand, Network, Zürich, Calibration

Preferred citation style
1 Introduction

MATSim is an activity-based, extendable, open source, multi-agent simulation toolkit implemented in JAVA, designed for large-scale scenarios. It follows a co-evolutionary approach. A good overview of MATSim is given in Balmer et al. (2006); Meister et al. (2009), a very short introduction is available in Horni et al. (2011a).

The basic principle of MATSim and its required input and calibration/estimation data are depicted in Figure 1. A synthetic population is derived from census data. The initial transport demand of this population—consisting of an activity chain specified by attributes such as number, scheduling, timing, location of activities and mode for connecting trips—is created from travel surveys. This demand is then subjected to a constrained equilibration, i.e., a user equilibrium subject to constraints is searched. An iterative scheme is required due to a circular dependency of choices and infrastructure conditions.

Clearly, most models are not comprehensive from its beginnings. In MATSim not yet all dimensions are handled endogenously but given as input and unchanged during simulation. Some first experimental and explorative modules exists for activity chain choice (Feil, 2010). However, in productive scenarios number and type of activities are fixed input and not yet subjected to equilibration.

This paper’s core topic is the description of input data to generate the population, estimate the choice models and define the constraints and the infrastructure (Section 2). Main modules and functionality for network loading (mobsims), replanning, analysis and handling of different transport modes are also treated (Section 3). Furthermore, details of three large-scale MATSim projects are provided in Section 4. In Section 5 configuration hints are given related to the Zurich scenario as well as to software tools such as Maven. Calibration and runtime results are reported for the most recent project Herbie. The paper concludes by identifying conceptual limitations and reporting technical difficulties in applying the Zurich scenario (Section 6).

An introductory tutorial on MATSim can be found at (I).
2 Input Data

All scenario data are available internally in the revision control system svn under (2). Most data used are derived from raw data provided by the Swiss Federal Statistical Office. It is not publicly available, for research, however, licenses and contracts are issued; contact information can be found at (3).

2.1 Population and Initial Demand

The Switzerland scenario forms the basis for the Zurich scenario. The later is basically a cut-out from the Switzerland scenario and, thus, we start with describing the Switzerland scenario. The Zurich scenario, however, is the more detailed scenario; it is enhanced with data only available for this region, e.g., traffic light data or freight demand data is only included for Zurich city and canton respectively.

2.1.1 Switzerland Scenario

The scenario population is derived from the Swiss Census of Population 2000 (Swiss Federal Statistical Office, 2000). Home locations are given at hectare level and work locations are known at municipality level from the commuter matrices (4), a component of the Swiss Census of Population 2000 (see Section 6.2) (Balmer et al., 2009a, p.35). Other activity locations are derived from the Federal Enterprise Census 2001 (Swiss Federal Statistical Office, 2001) providing hectare level information. In MATSim all locations are termed activity facilities or in short facilities. The generation of facilities is described in Section 2.3 and Balmer et al. (2009a, p.33).

The travel demand is taken from the National Travel Survey for the years 2000 and 2005 (Swiss Federal Statistical Office, 2006).

A very good overview in German of the population generation, its initial individual demand and activity locations can be found in Meister et al. (2009). Further information is given in Ciari et al. (2008); Meister et al. (2010); Balmer et al. (2009a, 2010, 2008c).

Two versions of the MATSim population and initial demand for Switzerland are in use, where version 2 is richer and more recent. The two versions differ in activity type classification and technical improvements.
Version 1: The activity types for version 1 are:

- home, work, education, leisure, shop.

Available modes are:

- walk, bike, car, pt, undefined.


These 17 types are mapped to the 5 MATSim types mentioned above, using (6) and (68). Comparison of person id and trip id indicates the following mapping (house_hold_number.person_number.trip_number):

1. walk: 15. Zu Fuss (777.1.1)
2. bike: 14. Velo (1387.1.1)
3. car: 9. Auto (575.1.1)

Subsequent mode choice on the tour-level as described in Section 3.2.4 is done on the basis of (7). The variable TRIP_MODE is subject to above assignment. The variable MODE_CHOICE is derived from the microcensus variable ausmittel (see e.g., (41049.1.2),(22578.1.1) or (8497.1.2)). Car trips done by persons without driving license are car passenger trips (789.2.1). According to personal communication, tour-based mode choice models are estimated based on variable MODE_CHOICE.

Based on this mapping, for demand v1, it is probably more appropriate to talk about car traffic rather than talking about motorized individual traffic when describing the mode 3.car in papers and publications, although the quantitative difference is small.

Demand was created with the (deprecated) package (8).

Version 2: The activity types for version 2 are:

- home, work_sector_2, work_sector_3, education_kindergarten, education_primary.

\[1\] Remember, that the number and type of activities are not changed in the MATSim iterations.
In [Balmer et al. 2008a,b] shopping and leisure activities have been further categorized in

- shop\_lt100sqm, shop\_get100sqm, shop\_get400sqm, shop\_get1000sqm, shop\_gt2500sqm,
  shop\_other and leisure\_culture, leisure\_restaurant, leisure\_sports

The version including subtypes of shopping and leisure can be found in svn history. Available modes are:

- walk, bike, motorized individual traffic, pt, ride.

Mode choice is included on sub-tour-level as described in Section 3.2.4.

Technical improvements of version 2 are as follows. Desired activity durations do not have to be stored in the configuration file (see Section 5.2) but are included in plans file via attribute <d-\_si\_res>. Primary activities to be performed at specific locations are defined with <knowledges> attribute. Facilities are used in this demand version, although the plans of version v1 can be adapted easily in a pre-preprocessing step to contain facilities as well.

Be aware that persons, which are registered at a different municipality than their residential location appear twice in the Swiss census with different ids. This has been taken into account in demand v2.

Car occupancy rate should be verified in future demand creation (mode ride).

This demand version can be found in folders called v2. It works best with the package at (9). Demand was created with the (deprecated) package (10).

Microcensus reports 8 shopping types and type combinations and 19 leisure types and type combinations. The model can be enhanced in the future in this regard. Microcenssi of other years may be added to avoid small sample sizes.

### 2.1.2 Cross-Border Traffic Demand

Cross-border traffic is derived from mode-specific hourly origin-destination matrices given by [Vrtic et al. 2007]. They are disaggregated to 572’749 individual MATSim plans for all of Switzerland, which contain the cross-border traffic that originates outside Switzerland. Non-Swiss cross-border traffic starting in Switzerland is supposed to be negligible. Cross-border traffic was included in the project Westumfahrung but not in the project KTI Frequencies. It will be applied in the project Herbie. Further information on creation of cross-border traffic plans is available at (11). The cross-border plans are archived under (12) and code for demand creation can be found at (13).
2.1.3 Freight Traffic Demand

Canton Zurich raw data for freight traffic is taken from the KVM (Kantonales Verkehrsmodell) provided by Amt für Verkehr, Volkswirtschaftsdirektion Kanton Zürich (2011) and documented in Gottardi and Bürgler (1999). Matrices given at zonal level (1341 zones, 182'687 trips) are disaggregated to single MATSim plans (Shah, 2010). Matrices for small delivery trucks and heavy trucks are combined into one activity called freight. An additional 181’886 agents are generated for Zurich region. Freight traffic is applied in project Herbie (Section 4.3).

Base data from the KVM includes OD-matrices for three specific hours of the day and the daily volumes (14). These matrices have been manually converted from Visum format to text files (15). Zoning information can be found under (16). A deprecated zones file used for Shah (2010) is stored under (17).

Generated MATSim plans are stored under (18) and deprecated plans of Shah (2010) are stored under (19). The code to create the freight demand can be found at (20).

It is not fully clear to the authors if the Swiss microcensus captures some of the freight traffic through the activity type "Geschäftliche Tätigkeiten und Dienstfahrt". However, only 6% of all trips are done for this purpose, thus, the quantitative error is expected to be small in any case.

2.1.4 Zurich Demand

The Zurich base scenario is a cut-out from the Switzerland scenario and enhanced by further data. The study area is delineated by a circle with radius 30 km around Bellevue, a central and prominent location in Zurich. Again two versions of the Zurich scenario exist, namely the Zurich diluted scenario and the Zurich cut scenario.

Diluted Scenario: The diluted scenario, created with (21), contains all persons that perform at least one activity in that circle or have at least one trip intersecting it. Plans are located in the folders called dilZH. The Zurich diluted scenario is depicted in Figure 2 and Figure 3.

Cut Scenario: The Zurich cut scenario (22), created with (23), contains only the persons who stay in the study area during the complete day. The Zurich cut scenario has been used in an experimental manner only in Hackney (2009). It is thus recommended to use the Zurich diluted scenario for production runs.

Enhancement of the Zurich base scenario by cross-border and freight traffic is described in Section 4.
2.2 Networks

Network data are stored under the svn path (24).

2.2.1 Road Networks

There are 3 different road networks available at IVT, namely the ivtch, navteq and teleatlas networks.

ivtch:  ivtch is a planning network with approximately 24’000 nodes and 60’000 links (see Balmer et al. (2008c) p.5) (Figure 5). Links of neighboring countries are also modeled but with a lower resolution. This network has been generated for Vrtic et al. (2005) and it is property of IVT and ARE (ARE, 2006). Different versions have been created for the project Westumfahrung.

1. ivtch: version generated from original shape files. Shape files links.shop.zip and nodes.shp.zip are available in the same folder.

2. ivtch-changed: modifications of the free speed and number of lanes and capacity attributes. See the provided read-me file in the same folder.

3. ivtch-changed-wu: includes Westumfahrung Zürich

4. ivtch-changed-wu-flama*: includes Westumfahrung Zürich and different reconstruction projects (Flankierende Massnahmen) as described in Balmer et al. (2009a).

5. ivtch-osm: The authors are not aware of studies using this network. Supposedly, its attributes are taken from OpenStreetMap (2011).

The networks including Westumfahrung should be used now.

teleatlas:  teleatlas (Tele Atlas MultiNet 2010) is a navigation network with almost 500’000 nodes and more than 1’000’000 links (see e.g. Meister et al. 2010). It has been used for the project KTI Frequencies and will be used for project Herbie. Road capacities and number of lanes were not part of the network and were thus imputed using (25) as described in Balmer et al. (2010) p.6. Research licenses are also available from (26).

navteq:  navteq is a navigation network (NA VTEQ 2011) with more than 400’000 nodes and almost 900’000 links. It has been used for Schüssler and Axhausen (2009). The version internally available at the IVT has flaws; significant errors in terms of multi-lane roads and wrong link speeds exist. If you plan to use this network contact Dr. M. Balmer, senozon for more information.
2.2.2 Public Transport Networks

The MATSim multi-modal network is derived from the KVM (Kantonales Verkehrsmodell) as provided by Amt für Verkehr, Volkswirtschaftsdirektion Kanton Zürich (2011). The version file is stored under (27). A .net export (28) serves as input to the MATSim converter (29). Using this converter a public transport network (30) and some further files as described in Section 2.4 are generated.

By merging the ivtch network with the public transport network a multi-modal MATSim network for the Zurich region is generated (31). Public transport and motorized individual traffic do not share joint links. However, the KVM (Kantonales Verkehrsmodell) includes a VISUM multi-modal network containing shared links for different modes. A new converter to generate truly multi-modal MATSim networks from VISUM networks is available from senozon AG (2011).

2.3 Facilities


Different MATSim activity classification schemes have been defined; see the read-me file at (32) and (33). In the most recent version 382’979 activity locations of type work_sector2 or work_sector3 are available. Dependent on the NOGA classification these locations are additionally defined as education_kindergarten, education_primary, education_secondary, education_higher, education_other, shop or leisure locations. Earlier versions are available in svn history.

The facilities file provides opening hours and capacity limits. Capacity limits are derived from number of full-time jobs associated with respective facility as described in (32). Opening hours are imputed from a sample data set containing opening hours and address of Migros, Coop, Denner and Pick Pay. These data have been collected from websites and are stored at (34). Approximately 1.3 Million home facilities are derived from Swiss Census of Population 2000 (Meister et al., 2009, p.272). The 1’697’196 facilities provide 1’798’752 activity options and are stored under the svn path (35). In the Zurich region there are 373’169 facilities providing 394’700 activity options.

Freight facilities are stored under (36). The deprecated version of Shah (2010) is stored under (37).
2.4 Public Transport Stops and Schedules

Travel time matrices and public transport stops table as used in project KTI Frequencies are derived from Vrtic et al. (2005). Travel time matrix is stored under (38) and list of public transport stops can be found under (39).

For simulation of public transport with mobsims QSim (see Section 3.1.2) and senozim (see Section 3.1.5) the procedure in Section 2.2.2 is applied. Two additional files (43) and (44) are generated containing public transport schedule and available public transport vehicles. These two files together with the multi-modal network is needed to run the public transport simulation as described in Section (3.2.1).

2.5 Count Data

Count data on city level, cantonal level and national level (ASTRA, 2006) are available (45) from various sources. An average working day (Monday to Thursday, excluding public holidays) is used for comparisons in current scenarios.

For the ivtch network, count data contains ASTRA and Zurich city count stations (478 ASTRA links and 98 Zurich city links, in total 570 measured different links). In Balmer et al. (2009a, p.40)) 578 links are mentioned. However, some stations are mapped to the same links (2 ASTRA links and 6 Zurich city links). These double entries are ignored for the actual automatic evaluation. Within the 30 km circle around Bellevue 159 links (70 ASTRA links and 89 Zurich city links) are measured for the ivtch network (center node is ivtch node 2531). One direction of station ZHS040 is wrongly mapped to non-existing link 108045. For the Zurich diluted scenario not all count stations within the study area are used to reduce boundary effects as described in Section 6.2. Instead, for most studies only 123 links (34 ASTRA links and 89 Zurich city links) located in the circle with radius 12 km around Bellevue are taken into account (see Balmer et al. (2009a, p.20)).

For the teleatlas network used for project KTI Frequencies additional data from the cantons Basel Land, Neuenburg, Graubünden and Zurich are used. Together with ASTRA and Zürich city stations this totals in 655 measured links (46) (see also Balmer et al. (2010, p.6).

Count data is generated converted with the package (47). The evaluation above is done with (48). Public transport count data are not yet available but could be extracted from the KVM (Kantonales Verkehrsmodell).
2.6 Traffic Lights

Information about signal light timing are available for the city of Zurich (Stadt Zürich, Dienstabteilung Verkehr, 2008). Modeling of traffic lights was implemented for the project Westumfahrung through reducing the available link capacity. However, the used mobsim (DE-QSim) is deprecated (see Section 3.1.3).

Research and implementation efforts are undertaken to include individual traffic lights in other mobsims (see e.g., (50) and Neumann (2008)).

2.7 Parking

Park and ride data for Switzerland is available at (51). Parking data for Zurich city is available at (52). This also includes private parking lots (53), street parking lots and load curves of public underground parking lots (54). Data of parking lots for the canton Zurich is available at (55).

However, except for experiments (Waraich and Axhausen, 2011), parking is not yet included in MATSim.
3 Modules and Functionality

3.1 Network Loading Simulations a.k.a. Mobility Simulations (*mobsims*)

An overview of MATSim mobility simulations is given in [Dobler and Axhausen](2011). See also the presentation of [Rieser](2011).

3.1.1 queueSimulation

*queueSimulation* is the basic MATSim network loading simulation. However, it is not the most frequently used one as it has been stripped down to the very basic functionality. It is queue-based and time-step based.

3.1.2 QSim

At the moment in most applications *QSim* (56) and (57) is used despite being under constant further development. It has started as a software branch of *queueSimulation* before supplanting this later simulation. Through multiple engines it can be run in parallel and simulate multi-modal traffic. However, for now, there is no interaction between different modes. The simulation is furthermore able to handle time-variant networks, within-day replanning ([Dobler](2009)), public transport ([Rieser](2010)) and in an experimental manner also traffic lights ([Neumann](2008)). Inclusion of public transport is described in Section 3.2.1.

3.1.3 DEQSim (deprecated)

*DEQSim* (58) was used for project *Westumfahrung*. It is a queue-based, event-based parallel simulation written in C++ ([Charypar et al.](2007); [Charypar](2008)). This simulation includes handling of reduced capacities due to traffic lights in an aggregate manner ([Charypar](2008), p.139 ff). It further supports modeling of gap back propagation at junctions ([Charypar](2008), p.98 ff). Events handling is done via file input and output. This represents a major bottleneck in terms of framework performance. Development of this simulation has been stopped.

3.1.4 JDEQSim

*JDEQSim* was used for project *KTI Frequencies*. It is a java reimplementation of *DEQSim* ([Waraich et al.](2009)) and provides parallel event handling but no parallel simulation ([Balmer]...
et al. (2010, p.11). Back-propagating gaps, traffic lights, public transport and within-day replanning are not supported. Advice for JDEQSIm configuration can be found at (59).

3.1.5 The Coming Mobsim (working title senozim)

senozim, an improved version of QSim, is in its final development stage (Rieser, 2011).

3.2 Multi-Modal Simulation

Simulation of different modes and their interactions is a crucial topic for further development of (multi-modal) mobsims. Current state of development and its history is sketched below.

3.2.1 Public transport

Project Westumfahrung did only take into account car traffic. In Balmer et al. (2008a) public transport is included. Based on an MNL (Ciari et al., 2007) the chosen mode was applied for the complete day plan. Public transport plans were not routed and simulated in investigations of that paper. Travel time is estimated based on the crow-fly distance and assumed travel speeds, where crow-fly travel time is multiplied by a factor usually set to 1.5.

For project KTI Frequencies travel time matrices and public transport stop tables of Vrtic et al. (2005) have been used (Balmer et al., 2010, p. 7), but public transport was not simulated. Data used is described in more detail in Section 2.4.

MATSim is now able to simulate public transport with QSim and senozim. However, there are no interactions between the different modes and there is not yet a true multi-modal network readily available. The data for Switzerland is available as described in Section 2.4. Required steps to simulate public transport are described at (60) and in Rieser (2010, p.100ff). The three input files required for the simulation are described in Section 2.4.

3.2.2 Walk and Bike

For the time being all walking and biking travel times are calculated by using crow-fly distances and assumed average travel speeds. In other words, these modes are not routed in the network and, thus, also here no interactions between agents occur. Navigation networks provide enough information to build bike and walk routes. Walk and bike modes have been included in Balmer et al. (2008a) and project KTI Frequencies but not in project Westumfahrung.
3.2.3 Ride

Handling of rides is not yet productively implemented in MATSim. Research to integrate carpooling (Ciari and Axhausen 2011) and joint trips (Dubernet 2011) is ongoing.

According to personal communication with M. Balmer and K. Meister in project KTI Frequencies the rides were teleported assuming car travel speeds. This means that rides are routed using the loaded network but they are not simulated. Furthermore, mode ride was not provided in the mode choice replanning as its share is very low (4.4% in the microcensus) and as its handling is only experimental in MATSim (Balmer et al. 2010 p.15).

3.2.4 Mode Choice on Tour and Sub-Tour Level

The first mode choice model for initial demand is described in Ciari et al. (2007). Therein, mode choice is made on tour level. A tour in demand \( v_1 \) encompasses the complete day plan. This mode choice model was applied in project Westumfahrung (Balmer et al. 2009a p.37).

An improved sub-tour (sequence of activities that starts and ends at the same location) mode choice model is introduced in Ciari et al. (2008). It is used for the generation of demand \( v_2 \) and it is applied in project KTI Frequencies.

3.2.5 Mobility Tools

Mobility tools such as car ownership or season tickets for public transport are taken into account in project KTI Frequencies for both replanning and scoring Balmer et al. (see also 2010 p.29ff).

3.3 Replanning Modules

MATSim provides replanning modules for time (Balmer et al. 2005), route (Lefebvre and Balmer 2007), destination (Horni et al. 2011b) and mode choice. A special replanning module using a different logic than undirected trial-and-error is planomat. planomat does not evaluate just one random alternative per iteration but multiple alternatives within one single iteration to obtain a (at least locally) optimal solution. It uses a genetic algorithm (Meister et al. 2006a,b) for this purpose. planomat was applied for the project KTI Frequencies for time choice and mode choice for sub-tours (Balmer et al. 2010 p.10). Configuration of planomat applied in this project can be found in Section 4.2 General advice for the application of planomat is given at 61.
3.4 Analysis Tools

3.4.1 Visualizers

senozon via: It is the new and powerful MATSim visualizer. It is proprietary and can be bought from senozon AG (senozon AG, 2011). Research licenses are available. via is written in java and distributed as a jar file. See Figure 6.

On-the-fly-visualizer (OTFVis): It is developed in java and uses OpenGL leading to platform-dependency (Strippgen, 2009), (62). See Figure 7.

MATVisReplay (deprecated): Development of this visualizer, written in C++ (Charypar, 2007), is not continued. See Figure 8.

3.4.2 Additional Tools

A set of additional analysis tools were implemented for project KTI Frequencies (63) and (64). Leg times and distances as well as the score elements are evaluated. These tools are now migrated to (65) for project Herbie.

The same tools can be used to analyze microcensus, which has been converted to MATSim plans (66) or (67) using (68).
4 Large-Scale Projects Based on Switzerland or Zurich Scenario

4.1 Westumfahrung

This project is described in detail in Balmer et al. (2009a) and its basic configuration is as follows:

- **Demand:** For this project demand and population version 1 for the Zurich diluted scenario was used. Base demand is generated as described in Section 2.1, where mode choice is performed on tour level as described in Section 3.2.4. For this project only car traffic demand (see Section 2.1.1) was taken into account. For final results 10\% samples were used (10\% of 673’706 agents (Balmer et al., 2009a, p.50)). Also 100\% samples were simulated in earlier stages of the project. Plans used in this project are located in folders marked with miv. Cross-border traffic was included.

- **Network:** ivtch network was used including the adapted versions (see Section 2.2).

- **Replanning Modules:** Route choice (Landmarks-A*) and time choice are part of the optimization. planomat was not used for final results production.

- **Modes:** Only car traffic traffic (see Section 2.1.1) is handled (Balmer et al., 2009a, p.37 and 50) (see also Section 3.2).

- **Mobsim:** Final results are produced with DEQSim including light signals in the city of Zurich.

Results can be found at (69). Configuration file used for the base scenario in this project can be found in the log file at (70).

4.1.1 Performance

In the final report (Balmer et al., 2009a) no performance information is given. This is done in the reports Balmer et al. (2008b,a) published during the final project phase. For these experiments, run on satawal, configuration is substantially different: Not only car but also bike, walk and public transport are included. They are, however, not simulated but handled with "teleportation". A 10\% sample was used, which now results in 180’000 agents (instead of 67’370 agents). planomat was used for time choice, and replanning was performed on 2 CPU cores. A time-step-based non-parallel network loading simulation was used. A total runtime of 23 hours per run with 100 iterations was achieved.

Additional reports for complete Switzerland exist (Balmer et al. 2009b, 2008c). Balmer et al. (2009b, table 1, p.73) gives detailed information about computation times for different
configurations.

4.2 KTI Frequencies

This project is described in detail in Balmer et al. (2010) and its basic configuration is as follows:

- **Demand:** While the initial calibration was done with the Switzerland scenario v2, the final results were produced with the Switzerland datapuls population. Results are produced for a 25% sample of the datapuls population (Balmer et al., 2010, p.13).
- **Network:** Initial calibration was done on the ivtch network but final results are produced for the teleatlas network.
- **Replanning Modules:** Mode choice was part of the optimization and performed on sub-tour level, as described in Section 3.2.4. planomat was used for time and mode choice. Furthermore, route choice was performed with Landmarks-A*-algorithm.
- **Modes:** Motorized individual traffic is simulated. Public transport departure and travel times are based on travel time matrices and public transport schedules. Walk and bike trips are calculated based on straight-line distances. Ride trips are routed but not simulated (see also Section 3.2).
- **Mobsim:** Final results are produced with JDEQSim with parallel event handling.

Results for described production runs can be found at (71) and (72). A configuration file used in this project can be found at (73).

4.2.1 Performance

Experiments were run on the machine satawal, where 90 GB RAM were necessary for the 100% scenario using datapuls population (ca. 6 million agents) (Balmer et al. 2010, p.8) and configured as described above. One iteration took approximately 4.5 hours (Balmer et al. 2010, p.11) when executed with 16 CPU cores. Approximately 50-60 iterations were necessary to achieve a relaxed state. This results in almost 2 weeks of computation time. Thus, in this project 25% samples were simulated requiring 30GB RAM and 4 days of run time (Balmer et al. 2010, p.13) and (Meister 2010, p.20).

4.3 Herbie

Basic configuration is as follows:

- **Demand:** This project uses the diluted Zurich scenario with demand v2. Both freight traf-
fic and cross-border traffic are included. 196’947 agents are simulated in total, including freight and cross-border traffic.

- **Network:** Initial calibration is performed on the ivtech network whereas final results will be produced using the teleatlas network.

- **Replanning Modules:** A time mutator, Rerouting, TransitChangeLegMode, Transit-TimeAllocationMutator and TransitSubtourModeChoice is used as replanning modules. To this day, an integrated replanning module, such as planomat, was not yet available to suit the requirements of freight traffic and public transportation.

- **Modes:** Besides car traffic also public transport is simulated. Simulation of public transport for MATSim has been developed by Rieser (2010). For the Herbie project it will be applied for the first time in a project. As in earlier projects walk and bike are "teleported". Car passengers (mode "ride") have not been considered so far.

- **Mobsim:** qsim with the public transport engine will be used.

First calibration results are shown in Section 5.

### 4.3.1 Performance

Experiments were run on the machine pulusuk, where 60 GB RAM were necessary for the 10% scenario using population (ca. 200’000 agents) and configured as described above. One iteration took approximately 2.0 min when executed with 16 threads. Approximately 150 iterations were necessary to achieve a relaxed state. This results in 5 h of computation time.
5 Configuration

5.1 Running a large Scenario

Many scenarios, e.g. the 100% Zurich scenario, are too large to be run on a desktop computer. For running it on a high-performance server machine basically three possibilities exist, chosen by equal numbers of users at IVT.

5.1.1 Access to servers

1. The most direct approach is to use Maven and SVN and to “compile” the code with Maven as described at (74) (see also Section 6.1).
2. Some users create a jar file locally as described at (75), copy this jar file to the server and start it there.
3. It is also possible to run eclipse remotely on the server. Code can be built as described at (76).

5.1.2 Required Resources

To get a rough idea of required resources, and to help users identifying misconfigurations, some numbers are provided in Section 4. Specifications of the machines are given at (77), (78) and (79). On the one hand many technical improvements have been achieved in the years between project Westumfahrung and the recent project Herbie, on the other hand the scenarios have been extended. Thus, clearly, the most relevant and recent numbers are given for Herbie project. However, information is also given for earlier settings as for some studies it may make more sense to run an earlier, i.e., less detailed, scenario.

5.1.3 Background Processes

When running MATSim on a remote server the GNU screen tool (GNU Project, 2011) does a great job. Simulations can be run in background while one disconnects from the server in the meantime.

5.1.4 Improving Performance

Our experience has shown that the JVM option
substantially improves performance if less than 32 GB of RAM are used. Gains in computation time are around 10% and gains in needed memory are around 20%. It is very important, in general, to work with the latest update of the JVM from Sun/Oracle. Further hints how to speed up MATSim Runs are given at (80).

5.2 Configuration File Settings

Detailed explanations of parameters used in the simulation and specified in the configuration file are given in the MATSim javadoc (81) as well as in the MATSim modules documentation (82) and (83).

References to very central settings are given below.

5.2.1 Choosing a Mobsim

To chose a mobsim the attribute mobsim in the controller module needs to be set. At the moment the following options are available: queueSimulation, qsim, jdeqsim, multimodalQSim. The option multimodalQSim does not specify its own mobsim but an engine in qsim.

<module name="controller">
    ...
    <param name="mobsim" value="queueSimulation" />
    ...
</module>

5.2.2 Utility Function Parameters

Especially important are utility function settings. Suggestions for implementation and configuration of the MATSim standard scoring function are given at (84). Recent research concerns a new MATSim utility function with an S-shape as proposed in Joh (2004) (instead of the logarithmic form) (Shah 2010; Feil 2010). The scoring function used in project KTI Frequencies is located at (85). An adapted version for the project Herbie can be found at (86). Utility function extensions and calibration results for the Herbie project are given below.
5.2.3 Public Transport

The modules "ChangeLegMode", "Strategy" and "Transit" need to be adopted if public transport is included in the microsimulation. A very detailed description of configuration settings necessary to simulate public transport is given in (60). This includes specification of replanning modules able to handle public transport such as TransitTimeAllocationMutator and TransitChangeLegMode.

The settings for handling public transport based on travel time matrices as done in the KTI Frequencies project are given below. However, this is deprecated and should only be used as a starting point.

```xml
<module name="planomat">
  ...
  <param name="possibleModes" value="car,pt,bike,walk" />
  ...
</module>

<module name="kti">
  <param name="distanceCostCar" value="0.12" />
  <param name="distanceCostPtNoTravelCard" value="0.28" />
  <param name="distanceCostPtUnknownTravelCard" value="0.14" />
  <param name="intrazonalPtSpeed" value="4.361111" />
  <param name="usePlansCalcRouteKti" value="true" />
  <param name="pt_traveltime_matrix_filename" value="&INBASE;/2005$_OEV_Befoerderungszeit.mtx" />
  <param name="pt_haltestellen_filename" value="&INBASE;/Haltestellen.txt" />
  <param name="worldInputFilename" value="&INBASE;/world-natModell.xml" />
  <param name="travelingBike" value="-16.0" />
  <param name="constBike" value="-0.48" />
  <param name="constCar" value="-0.48" />
</module>
```
5.2.4 Settings for the Large-Scale Scenario *Herbie*

Here, only configuration settings for the most recent project are described. Configuration files for earlier projects are referenced in Section 4.

The currently proposed configuration is shown in Appendix B. A new travel score calculator is included in the utility function. The goal is to provide a flexible structure to allow custom travel utility functions throughout MATSim. So far, only the scoring refers to the custom travel utility functions. Routing needs to be adapted to the new function.

Currently, a linear structure is proposed for the utility function:

\[ U_{m,i} = \text{const}_m + \beta_{tt,m} \cdot tt_i + \beta_{td,m} \cdot td_i, \]

where

- \( U_{m,i} \): utility of agent \( i \) traveling with mode \( m \).
- \( m \): travel mode \( m \).
- \( \beta_{tt,m} \): marginal utility of travel time with mode \( m \).
- \( tt_i \): travel time of agent \( i \).
- \( \beta_{td,m} \): marginal utility of travel distance with mode \( m \).
- \( td_i \): travel distance of agent \( i \).

The linear approach is adapted to a piecewise linear function in mode *walk* and *bike*.

The corresponding revision number is currently 16258 (July 29, 2011). A different revision number can lead to a different outcome due to changes in the code.
6 Conceptual and Technical Problems

6.1 Technical Problems and Debugging

Practice has shown that not yet every misconfiguration is properly handled by an error message. Sometimes the microsimulation is stuck without error message or java exception. Nevertheless, some symptoms can be observed often. They are described here together with the identification of the error.

Wrong or missing specification of activities for scoring:  Simulation stops without error message after the plans are dump. Often this is caused by wrong or missing specifications of the activity types in the scoring section of the configuration file, i.e., plans contain an activity that is not specified in the configuration file.

Maven configuration:  If done for multiple runs with different code some enhanced Maven configurations are required, which is a constant source of problems difficult to trace. For example JVM hotspot errors have been observed (see Figure 4). In this case you should use multiple local Maven repositories.

Compiler errors:  Compilation errors can occur in Eclipse during jar generation with maven. Errors can be avoided when using the commands “clean install” for compilation, instead of only “install”. Depending on methods and servers, the jdk version jdk1.6.0_24 sometimes works better than the newest jdk. The reason could not yet be found.

6.2 Conceptual Limitations and Problems

Too high travel speeds:  The queue-based mobsims are especially designed for handling large-scale scenarios. Consequently, they only implement relatively simplistic rules guiding the physics of traffic. However, even in this context a couple of flaws exist, which in the sum produce too high travel speeds compared to empirical data. First, qsim is or will soon be able to take into account traffic lights. However, data is only available for the city of Zurich. Moreover, junction dynamics of non-signaled intersections are only marginally taken into account. Furthermore, interaction between different modes is missing, often reducing travel speed in reality. Places with many pedestrians crossing streets such as the Central in Zurich intensify problematic traffic conditions. This is missed in MATSim. Finally, it can be assumed that the queue model is linear or almost linear. This is contradictory to the common non-linear cost-flow curves, such as
the BPR function ([U.S. Bureau of Public Roads](1964)). The simulation of back-propagating gaps in DEQSim can be seen as a first attempt in direction of relaxing this conflict. However, DEQSim is deprecated and the recent qsim does not support this feature.

**Missing prices and income:** Prices and incomes as crucial determinant in economic models are missing in MATSim. No VTTS can thus be employed. Incomes can be derived from [Swiss Federal Statistical Office](2008, 2007, 2006).

**Missing households:** To date households are missing. As the household is a common and natural modeling unit in transportation planning and economics, MATSim should be enhanced by such a data structure. Empirical data is available in the Swiss Census.

**Almost identical agents:** Apart from home and work location and mobility tools, agents are essentially identical, which misses much of the observed heterogeneity.

**Boundary effects for the diluted scenario:** When simulating an isolated study region with a diluted scenario agents may circumnavigate traffic jams by leaving the study area, especially when moving along the region’s boundaries. No mechanism have yet been developed to prevent this. Clearly, evaluation should be reduced to a smaller area in the center to avoid corruption of the results by boundary effects. For the cut scenarios this problem is inexistent.

**Population generation:** The Swiss Census of Population 2010, and probably following, will not be full population surveys but a mixture of a register-based compilation and sample surveys, mainly due to cost reasons. Thus, population synthesizers containing work location choice models need to be applied. For MATSim the synthesizer described in [Müller (2011)] can be used.
References


Rieser, M. (2011) Mobility simulations in matsim, internal presentation, IVT, ETH Zurich, Zurich, February.


A Referenced Files and Links

1. http://www.matsim.org/docs/tutorials/8lessons
2. ivt/studies/*/ 
4. ivt/studies/switzerland/matrices/*
5. IVT-VPL Drive P:/Daten/Mikrozensen Verkehr Schweiz/MZ 2005/Wegeketten-Tabelle/Wegeinland.sav
6. IVT-VPL Drive P:/Daten/Mikrozensen Verkehr Schweiz/MZ 2005/MZ05_Datenbank_CH/2_DB_SPSS/Wegeketten_new.sav
7. ivt/studies/switzerland/externals/Mode_choice_all.sav
8. playground.balmermi.census2000.*
9. playground.ivtExt.herbie.*
10. playground.balmermi.census2000v2.*
11. ivt/studies/switzerland/plans/xy/crossborder/documentation/GQPV.doc
12. ivt/studies/switzerland/plans/xy/crossborder/*
13. playground.anhorni.crossborder.*
14. ivt/studies/switzerland/externals/KVMZH/freight/rawdata/2010_DWV_LKW_LI_Fzg.zip
15. ivt/studies/switzerland/externals/KVMZH/freight/processed/2010_DWV_LKW_LI_Fzg_fma.zip
16. ivt/studies/switzerland/externals/KVMZH/freight/processed/Bezirke_und_Koordinaten_GVMZH_2010_mitNamen.txt
17. ivt/studies/switzerland/externals/KVMZH/freight/rawdata/Zonierung_2009_11_19_zone.zip
18. ivt/studies/switzerland/plans/xy/freight/ZH/freightplans.xml
19. ivt/studies/switzerland/plans/xy/freight/ZH/deprecated/*
20. ivtExt.herbie.creation.freight/*
21. playground.balmermi.ScenarioDilute.*
22. ivt/studies/kt-zurich/plans/ivtch-zh-cut/*
23. playground.balmermi.ScenarioCut.*
49. ivt/studies/switzerland/gtfs/ivtch/*
51. ivt/studies/switzerland/facilities/parking/parkAndRide/parkandrail.pdf
52. ivt/studies/city-zurich/facilities/parking/*
53. ivt/studies/city-zurich/facilities/parking/basis/private parking/*
54. ivt/studies/city-zurich/facilities/parking/basis/loadCurvesPublicUndergroundParkings/*
55. ivt/studies/kt-zurich/facilities/parking/*
57. http://www.matsim.org/node/397
58. http://www.matsim.org/node/65
60. http://www.matsim.org/docs/tutorials/transit
63. playground.meisterk.org.matsim.analysis.*
64. playground.meisterk.kti.controler.listeners.*
65. playground.ivtExt.herie.running.analysis.*
67. IVT-VPL Drive P:/Daten/Mikrozensus Verkehr Schweiz/MZ 2005/Wegeketten-Tabelle/matsim-plans/*
68. playground.balmermi.mz.PlansCreateFromMZ.java
69. ivt/studies/switzerland/results/westumfahrung/*
70. ivt/studies/switzerland/results/westumfahrung/run365/run-analysis/run365.log
71. ivt/studies/switzerland/results/ktiCHFrequencies/*
72. ivt/studies/switzerland/results/datapuls2010/*
73. ivt/studies/datapuls/switzerland/results/datapuls2010/runs/ta2010car-dp25pct-rev9473-run00/config.xml
74. http://www.matsim.org/node/368
75. http://www.matsim.org/node/373
76. http://www.matsim.org/docs/devguide/eclipse
80. http://matsim.org/node/344
82. http://www.matsim.org/node/377
83. http://www.matsim.org/node/478
84. http://www.matsim.org/node/262
85. playground.meisterk.kti.scoring.*
86. playground.ivtExt.herbie.running.scoring.*
B Config File for Zurich Scenario

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE config SYSTEM "http://www.matsim.org/files/dtd/config_v1.dtd"
[
  <!ENTITY INBASE
  "/Network/Servers/pingelap/Volumes/ivt-shared/Groups/ivt/vpl/
  Projekte/matsim/data/>
  <!ENTITY OUTBASE
  "/Network/Servers/pingelap/Volumes/ivt-shared/Groups/ivt/vpl/
  Projekte/herbie/output/"
]
>
<config>

<!--
======================================================================
-->

<module name="changeLegMode" >
  <param name="modes" value="car,pt,bike,walk" />
  <param name="ignoreCarAvailability" value="false" />
</module>

<!--
======================================================================
-->

<module name="config" >
</module>

<!--
======================================================================
-->

<module name="controller" >
  <param name="enableLinkToLinkRouting" value="false" />

  <!-- Specifies the file format for writing events. Currently
      supported: txt, xml. Multiple values can be specified separated by
      commas (','). -->
  <param name="eventsFileFormat" value="xml" />
  <param name="firstIteration" value="0" />
  <param name="lastIteration" value="150" />
</module>
```
<!-- Defines which mobility simulation will be used. Currently supported: queueSimulation, qsim, jdeqsim, multimodalQSim -->
<param name="mobsim" value="qsim" />

<!-- The type of routing (least cost path) algorithm used, may have the values: Dijkstra or AStarLandmarks -->
<param name="routingAlgorithmType" value="AStarLandmarks" />

<!-- An identifier for the current run which is used as prefix for output files and mentioned in output xml files etc. -->
<param name="runId" value="herbie" />

<!-- iterationNumber % writeEventsInterval == 0 defines in which iterations events are written to a file. '0' disables events writing completely. -->
<param name="writeEventsInterval" value="10" />

<!-- iterationNumber % writePlansInterval == 0 defines (hopefully) in which iterations plans are written to a file. '0' disables plans writing completely. Some plans in early iterations are always written -->
<param name="writePlansInterval" value="10" />

<!-- factor by which to re-scale the simulated values. necessary when simulation runs with something different from 100%. needs to be adapted manually -->
<param name="countsScaleFactor" value="10.0" />
<!-- distance to distanceFilterCenterNode to include counting stations. The unit of distance is the Euclidean distance implied by the coordinate system --><param name="distanceFilter" value="12000.0" />

<!-- node id for center node of distance filter --> <param name="distanceFilterCenterNode" value="2531" />

<!-- input file name to counts package --> <param name="inputCountsFile" value="/Network/Servers/kosrae.ethz.ch/Volumes/ivt-home/bvitins/tempMATSimData/countsIVTCH.xml" />

<!-- possible values: 'html', 'kml', 'txt', 'all' --> <param name="outputformat" value="all" />
</module>

<!--======================================================================
-->

<module name="events" />
</module>

<!--======================================================================
-->

<module name="facilities" >
<param name="inputFacilitiesFile" value="/Network/Servers/kosrae.ethz.ch/Volumes/ivt-home/bvitins/tempMATSimData/facilitiesWFreight.xml.gz" />
</module>

<!--======================================================================
-->

<module name="global" >
<param name="coordinateSystem" value="CH1903_LV03" />
<param name="numberOfThreads" value="16" />
<param name="randomSeed" value="4711" />
</module>

<!--======================================================================
-->

<!--The Zurich Scenario: A Technical Overview 1.0 December 2011-->

35
<module name="households">
   <param name="inputFile" value="null" />
</module>

<!--
======================================================================
-->

<module name="herbie">
   <param name="distanceCostPtNoTravelCard" value="1.0" />
   <param name="distanceCostPtUnknownTravelCard" value="1.0" />
   <param name="intrazonalPtSpeed" value="4.361111" />
   <!-- [unit_of_money/m] conversion of bike distance into money -->
   <param name="monetaryDistanceCostRateBike" value="-0.001" />
</module>

<!--
======================================================================
-->

<module name="locationchoice">
   <param name="algorithm" value="null" />
   <param name="centerNode" value="null" />
   <param name="constrained" value="true" />
   <param name="fixByActType" value="false" />
   <param name="flexible_types" value="null" />
   <param name="maxDistanceEpsilon" value="-1.0" />
   <param name="maxRecursions" value="5" />
   <param name="planSelector" value="SelectExpBeta" />
   <param name="radius" value="null" />
   <param name="recursionTravelSpeed" value="8.43333333" />
   <param name="recursionTravelSpeedChange" value="0.2" />
   <param name="restraintFcnExp" value="5.0" />
   <param name="restraintFcnFactor" value="0.1316872" />
   <param name="scaleFactor" value="4.88" />
   <param name="searchSpaceBeta" value="0.0001" />
   <param name="simple_tg" value="true" />
   <param name="travelTimes" value="true" />
   <param name="tt_approximationLevel" value="0" />
</module>

<!--
======================================================================
-->
<module name="multimodal"/>

<!-- Use this, if your network is not multi-modal. Links with free speeds that are lower than the specified cutoff value will be usable for walk and bike trips. -->
<param name="createMultiModalNetwork" value="false"/>

<!-- Only used, if createMultiModalNetwork is enabled (set value in m/s). -->
<param name="cutoffValueForNonCarModes" value="22.22222222222222"/>
<param name="dropNonCarRoutes" value="false"/>
<param name="multiModalSimulationEnabled" value="false"/>

<!-- Use number of threads > 1 for parallel version using the specified number of threads. -->
<param name="numberOfThreads" value="1"/>
<param name="simulatedModes" value="pt,walk,ride,bike"/>
</module>

</module>

======================================================================

<module name="network"/>
<param name="inputChangeEventsFile" value="null"/>
<param name="inputNetworkFile" value="/Network/Servers/kosrae.ethz.ch/Volumes/ivt-home/bvitins/tempMATSimData/network.multimodal-wu.xml.gz"/>
<param name="laneDefinitionsFile" value="null"/>
<param name="timeVariantNetwork" value="false"/>
</module>

</module>

======================================================================

<module name="parallelEventHandling"/>
<param name="numberOfThreads" value="1"/>
</module>

</module>

======================================================================

<module name="planCalcScore"/>
<!-- [utils] conversion factor of utility model for use in logit-based choice model. Set to 1 if your util function is estimated -->
<param name="BrainExpBeta" value="2.0" />
<param name="PathSizeLogitBeta" value="1.0" />

<!-- [utils] alternative-specific constant for bike. No guarantee that this is used anywhere. Default=0 to be backwards compatible for the time being -->
<param name="constantBike" value="-5.5" />

<!-- [utils] alternative-specific constant for car. No guarantee that this is used anywhere. Default=0 to be backwards compatible for the time being -->
<param name="constantCar" value="-0.5" />

<!-- [utils] alternative-specific constant for pt. No guarantee that this is used anywhere. Default=0 to be backwards compatible for the time being -->
<param name="constantPt" value="0.0" />

<!-- [utils] alternative-specific constant for walk. No guarantee that this is used anywhere. Default=0 to be backwards compatible for the time being -->
<param name="constantWalk" value="0.0" />

<!-- [utils/hr] utility for departing early (i.e. before the earliest end time). Probably implemented correctly, but not tested. -->
<param name="earlyDeparture" value="-180.0" />

<!-- [utils/hr] utility for arriving late (i.e. after the latest start time). Normally negative -->
<param name="lateArrival" value="-18.0" />

<!-- New score = (1-learningRate)*old_score + learningRate * score_from_mobsim. Learning rates close to zero emulate score averaging, but slow down initial convergence -->
<param name="learningRate" value="1.0" />

<!-- [utils/unit_of_money] conversion of money (e.g. toll, distance cost) into utils -->
<param name="marginalUtilityOfMoney" value="1.0" />

<!-- Discouraged! [utils/m] utility of walking per m, normally negative. This is on top of the (dis)utility. It is discouraged to use this but in some cases it may make sense. -->
<param name="marginalUtlOfDistanceWalk" value="-0.00065" />
<param name="monetaryDistanceCostRateCar" value="-0.001" />

<param name="monetaryDistanceCostRatePt" value="-0.0005" />

<param name="performing" value="2.26" />

<param name="traveling" value="-0.3" />
<param name="travelingBike" value="-0.5" />

<param name="travelingPt" value="-0.1" />

<param name="travelingWalk" value="-0.26" />

<param name="utilityOfLineSwitch" value="-1.0" />

<param name="waiting" value="-0.0" />

<param name="doLogging" value="false" />
<param name="jgapMaxGenerations" value="10" />
<param name="levelOfTimeResolution" value="7" />
<param name="populationSize" value="100" />
<param name="possibleModes" value="null" />
<param name="routingCapability" value="fixedRoute" />
<param name="simLegInterpretation" value="CharyparEtAlCompatible" />
<param name="tripStructureAnalysisLayer" value="facility" />
</module>

<module name="plans" >

<param name="inputPlansFile" value="/Network/Servers/kosrae.ethz.ch/Volumes/ivt-home/bvitins/tempMATSimData/plans.xml.gz" />
</module>

<module name="planscalcroute" >

<param name="beelineDistanceFactor" value="1.3" />
<param name="bikeSpeed" value="2.6" />

<param name="ptSpeedFactor" value="2.0" />
<param name="undefinedModeSpeed" value="13.88888888888889" />
<param name="walkSpeed" value="0.8333333333333333" />
</module>

<module name="qsim" >
<param name="flowCapacityFactor" value="0.1" />
<param name="storageCapacityFactor" value="1.0" />
<param name="startTime" value="00:00:00" />
</module>
<module>

<param name="endTime" value="30:00:00" />
<param name="removeStuckVehicles" value="false" />
<param name="stuckTime" value="600.0" />
</module>

<!--
======================================================================
-->

<module name="ptCounts" >

<!-- factor by which to re-scale the simulated values. necessary
when simulation runs with something different from 100%. needs to
be adapted manually -->
<param name="countsScaleFactor" value="1.0" />

<!-- distance to distanceFilterCenterNode to include counting
stations. The unit of distance is the Euclidean distance implied by
the coordinate system -->
<param name="distanceFilter" value="null" />

<!-- node id for center node of distance filter -->
<param name="distanceFilterCenterNode" value="null" />

<!-- input file containing the alighting (getting off) counts for pt -->
<param name="inputAlightCountsFile" value="null" />

<!-- input file containing the boarding (getting on) counts for pt -->
<param name="inputBoardCountsFile" value="null" />

<!-- input file containing the occupancy counts for pt -->
<param name="inputOccupancyCountsFile" value="null" />

<!-- possible values: 'html', 'kml', 'txt', 'all' -->
<param name="outputformat" value="null" />

<!-- every how many iterations (starting with 0) counts comparisons
are generated -->
<param name="ptCountsInterval" value="10" />
</module>

<!--
======================================================================
-->

<module name="roadpricing" >
<module name="scenario" >

<!-- Set this parameter to true if households should be used, false if not. -->
<param name="useHouseholds" value="false" />

<!-- Set this parameter to true if knowledge should be used, false if not. -->
<param name="useKnowledge" value="true" />

<!-- Set this parameter to true if lanes should be used, false if not. -->
<param name="useLanes" value="false" />

<!-- Set this parameter to true if roadpricing should be used, false if not. -->
<param name="useRoadpricing" value="false" />

<!-- Set this parameter to true if signal systems should be used, false if not. -->
<param name="useSignalsystems" value="false" />

<!-- Set this parameter to true if transit should be simulated, false if not. -->
<param name="useTransit" value="true" />

<!-- Set this parameter to true if vehicles should be used, false if not. -->
<param name="useVehicles" value="true" />
</module>

<!--
======================================================================
-->

<module name="signalsystems" >

<param name="ambertimes" value="null" />
<param name="intergreentimes" value="null" />
<param name="signalcontrol" value="null" />
<param name="signalgroups" value="null" />
<param name="signalsystems" value="null" />
</module>
<module name="strategy">
  <param name="ModuleProbability_1" value="0.8" />
  <param name="ModuleProbability_2" value="0.05" />
  <param name="ModuleProbability_3" value="0.10" />
  <param name="ModuleProbability_4" value="0.10" />
  <param name="ModuleProbability_5" value="0.10" />
  <param name="Module_1" value="SelectExpBeta" />
  <param name="Module_2" value="ReRoute" />
  <param name="Module_3" value="TransitChangeLegMode" />
  <param name="Module_4" value="TransitTimeAllocationMutator" />
  <param name="Module_5" value="TransitSubtourModeChoice" />
  <param name="maxAgentPlanMemorySize" value="4" />
</module>

<module name="transit">
  <param name="transitModes" value="pt,bus,train,tram" />
  <param name="transitScheduleFile" value="/Network/Servers/kosrae.ethz.ch/Volumes/ivt-home/bvitins/tempMATSimData/transitSchedule.networkOevModellZH.xml" />
  <param name="vehiclesFile" value="/Network/Servers/kosrae.ethz.ch/Volumes/ivt-home/bvitins/tempMATSimData/vehicles.oevModellZH.xml" />
</module>
<module name="travelTimeCalculator">
  <param name="calculateLinkToLinkTravelTimes" value="false" />
  <param name="calculateLinkTravelTimes" value="true" />
  <param name="travelTimeAggregator" value="optimistic" />

  <!-- The size of the time bin (in sec) into which the link travel times are aggregated for the router -->
  <param name="travelTimeBinSize" value="900" />
  <param name="travelTimeCalculator" value="TravelTimeCalculatorArray" />
</module>

<!--
======================================================================
-->}

<module name="TimeAllocationMutator">
  <param name="mutationRange" value="7200" />
</module>

<!--
======================================================================
-->}

<module name="vspExperimental">

  <!-- String: minOfDurationAndEndTime', 'tryEndTimeThenDuration', 'endTimeOnly' -->
  <param name="activityDurationInterpretation" value="minOfDurationAndEndTime" />

  <!-- set chainBasedModes, e.g. "car,bike", "car" -->
  <param name="chainBasedModes" value="car" />

  <param name="coloring" value="car" />

  <!-- Set this filename of MZ05 daily analysis -->
  <param name="inputMZ05File" value="" />

  <!-- set the traffic mode option for subTourModeChoice -->
  <param name="modes" value="car, pt" />
</module>

<!--
======================================================================
-->
--> 

</config>
Figure 1: MATSim Principle

input

population

situation (e.g. season, weather)

microsimulation (model)

choice model

constraints

feedback

generalized costs

network load simulation

output

U_{max} (day chains)

instantiation census travel surveys infrastructure data

e.g., socio-demographics

estimation

e.g., network constraints, opening hours

input

output
Figure 2: The Zurich Scenario
Figure 3: The Zurich Scenario (Google Earth)
Figure 4: Maven Error

```
+ A fatal error has been detected by the Java Runtime Environment:
+ SIGSEGV (0x0) at pc=0x00007f489937351e, pid=26126, tid=139950767590912
+ JRE version: 6.0_22-b04
+ Java VM: Java HotSpot(TM) 64-Bit Server VM (17.1-b03 mixed mode linux-amd64 )
+ Problematic frame:
+ C [<timp AppleWebKit_2.0.so.0+0x7031e] gdk_window_enable_synchronized_configure+0x2
+
+ If you would like to submit a bug report, please visit:
+ http://java.sun.com/webapps/blueprint/bugreport/crash.jsp
+ The crash happened outside the Java Virtual Machine in native code.
+ See problematic frame for where to report the bug.
+
+----------------------- Thread -----------------------

Current thread (0x000000004358c400): JavaThread "main" [__thread_in_native], id=26126, stack(0x00007f48d26490)

Thread info: si_signo=SIGSEGV, si_errno=0, si_code=1 (SEGV_MAPERR), si_addr=0x0000000000000018

Registers:
RA=0x0000000000000000, RAX=0x0000000041000060, RCX=0x0000007f48d0364700, RDX=0x0000007f48d1805280
RSIP=0x0000007f48d3626b0, RSP=0x0000000041000150, RSI=0x0000000000000000, RDI=0x0000000000000000
RS=0x0000000000000100, R9=0x0000000000000001, R10=0x0000000000000011, R11=0x0000007f489937310
R12=0x000000007f48d362c00, R13=0x0000000000000000, R14=0x00000000000000260, R15=0x0000000000000000
RIP=0x0000007f48d362731e, EFLAGS=0x0000000000010202, C55SFS=0x000000000000033, CR2=0x0000000000000000

Top of Stack: (sp=0x0000007f48d3626b0)
0x00007f48d3626b0: 0000000000000001 0000000041000060
0a00007f48d3626b0: 0000000041000150 00007f48990b38f9
0a00007f48d3626b0: 0000000000000000 00007f48d362c00
0a00007f48d3626b0: 0000000000000001 00007f48d362e0f
0a00007f48d3626b0: 0000000000000000 00007f48990b38f9
0a00007f48d3626b0: 0000000000000000 00007f48990b38f9
0a00007f48d3626b0: 0000000000000000 00007f48990b38f9
0a00007f48d3626b0: 0000000000000000 00007f48990b38f9
0a00007f48d3626b0: 0000000000000000 00007f48990b38f9
0a00007f48d3626b0: 0000000000000000 00007f48990b38f9
0a00007f48d3626b0: 0000000000000000 00007f48990b38f9
0a00007f48d3626b0: 0000000000000000 00007f48990b38f9
0a00007f48d3626b0: 0000000000000000 00007f48990b38f9
0a00007f48d3626b0: 0000000000000000 00007f48990b38f9
0a00007f48d3626b0: 0000000000000000 00007f48990b38f9
0a00007f48d3626b0: 0000000000000000 00007f48990b38f9
0a00007f48d3626b0: 0000000000000000 00007f48990b38f9
0a00007f48d3626b0: 0000000000000000 00007f48990b38f9
```
Figure 5: ivtch network

Figure 6: senozon via
Figure 7: OTFVis

Figure 8: MATVisReplay