EUROPEAN COMMISSION
European Research Area

SEVENTH FRAMEWORK PROGRAMME

Funded under Socio-economic Sciences & Humanities

Chapter for delivrable D7.2

Using MATSim as a travel model plug-in to UrbanSim

Thomas W. Nicolai

Berlin Institute of Technology (TU Berlin) 1

FP7-244557 28/05/12

SustainCity

EUROPEAN COMMISSION
European Research Area
SEVENTH FRAMEWORK PROGRAMME
Funded under Socio-economic Sciences & Humanities

Contents

1 Overview 1

2 Data Requirements 3
2.1 UrbanSim Data Requirements 3
2.2 MATSim Data Requirements 4

3 UrbanSim Output 7

4 MATSim4UrbanSim Configuration 9

5 MATSim Default Settings and Adjustments 10

6 MATSim Demand Generation and Mobility Simulation 12

7 Access- and Accessibility Computation in MATSim 13

8 UrbanSim Data Set Updates 19

9 References 21

Using MATSim as a travel model plug-in to UrbanSim

Thomas W. Nicolai
TU Berlin
Salzufer 17-19
10587 Berlin Germany
phone: +49 30 314-29 592
fax: +49 30 314-26 269
nicolai@vsp.tu-berlin.de

28/05/12

Abstract

This paper summarizes the technical aspects of coupling MATSim with UrbanSim for the two case studies, Brussels and Zurich. It is described step by step how both frameworks interact with each other and what data is exchanged by referring to UrbanSim parcel and zone models.

The paper is organized as follows. The next section provides a brief overview about the current integration approach, were each integration step is described in detail in the subsequent sections. The following sections explain the UrbanSim and MATSim data requirements (Section 2), the UrbanSim output for MATSim (Section 3) and the joint MATSim and UrbanSim configuration (Section 4), where some settings are fixed or automatically adjusted (Section 5). The Sections 6 and 7 are describing the processes inside MATSim, which includes the initial demand generation, the mobility simulation and the computation of access- and accessibility measures. Finally Section 8 summarizes which UrbanSim data sets and attributes are updated by the MATSim feedback.

Keywords

UrbanSim, MATSim, MATSim4UrbanSim, Integration, SustainCity Case Studies, Brussels, Zurich

1 Overview

This section summarizes the integration approach coupling MATSim[1] (Multi-Agent Transport Simulation) with UrbanSim[2]. This integrated model framework is referred to as MATSim4UrbanSim (e.g. Nicolai *et al.* (2011); Nicolai and Nagel (2011); Nicolai (accessed 2012)). The underlying software design issues and decisions of this integration can be found in Nicolai and Nagel (2010). A user guide to using MATSim as a travel model plugin for UrbanSim can be found under `http://matsim.org/extensions/matsim4urbansim`.

UrbanSim (e.g. Waddell (2002); Miller *et al.* (2005); OPUS User Guide (2011)) is an extensible, microscopic urban simulation model. It consists of several models reflecting the decisions of households, businesses, developers and governments (as policy inputs). Figure 1 provides a simplified overview of the processing sequence of the UrbanSim main models, which are the Econometric and Demographic Transition Models, the Household and Employment Mobility Models, the Household and Employment Location Models, Real Estate Development Model, and the Real Estate Price Model. The Household and Employment Models are independent models and only illustrated jointly in Figure 1 for simplicity. Like other urban simulation models, for instance DELTA, CUFM, MUSSA, POLIS or RURBAN, UrbanSim does not model transport itself (Wegener, 2004). Instead, it relies on an interaction with external transport models like MATSim.

MATSim is a disaggregated, "agent-based" traffic simulation model that simulates each traveler individually (e.g. Raney and Nagel (2006); Balmer *et al.* (2009)). Therefore MATSim takes the synthetic UrbanSim population and directly simulates their travel behavior. The travel demand is, in principle, a result of individual decisions made by each agent trying to organize their day and perform activities at and out of home. Besides, MATSim provides additional advantages such as simulating time-dependent congestion, time-dependent mode choice, or speeding up the computation by running small samples of a scenario.

MATSim4UrbanSim at glance

When UrbanSim moves forward in time from year to year, it calls MATSim in regular intervals to update the access and accessibility indicators in its data sets (see also Figure 2). In order to do this the following steps are performed, which are described in detail for UrbanSim parcel (Zurich application) and zone (Brussels application) models in the subsequent sections:

- UrbanSim calls MATSim and passes the current land-use pattern together with a dynamically generated configuration including parameter settings and references to input-files

[1] see matsim.org/
[2] see urbansim.org/

like the road network; see Section 3, 4 and 5.

- Based on this input MATSim generates the traffic assignment and returns the resulting access and accessibility indicators; see Section 6 and 7. After that MATSim terminates.
- Once MATSim has terminated, UrbanSim takes back control. It reads the indicators and updates its data sets, which are used in the next iteration as input for various models; see Section 8.

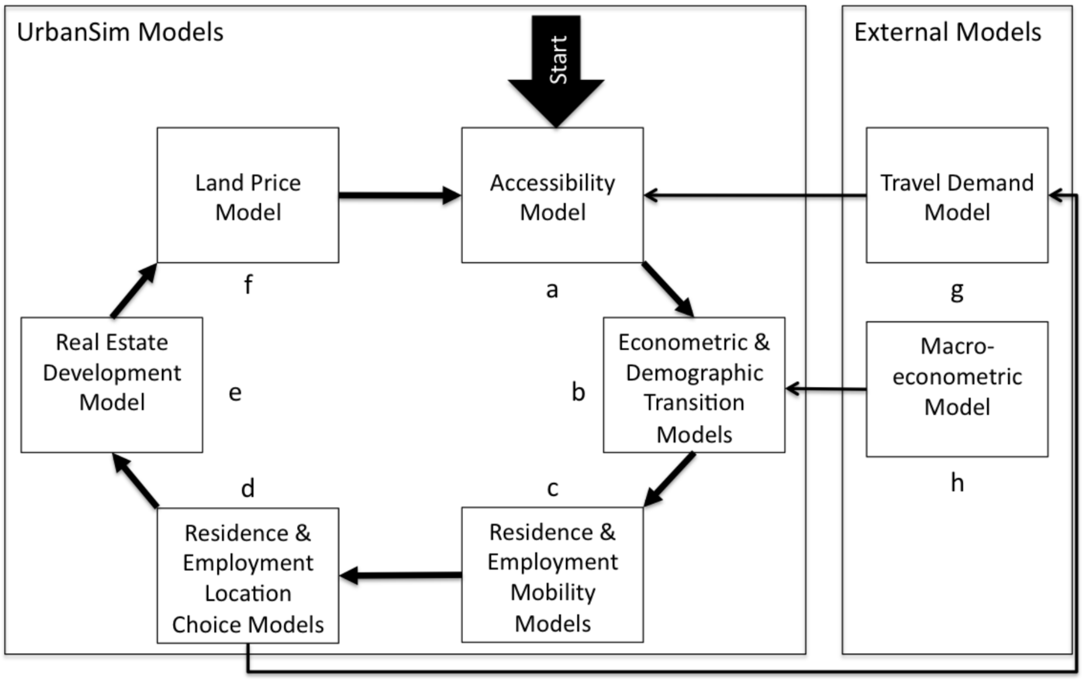

Figure 1: The sequence of UrbanSim main models after Waddell (2002).

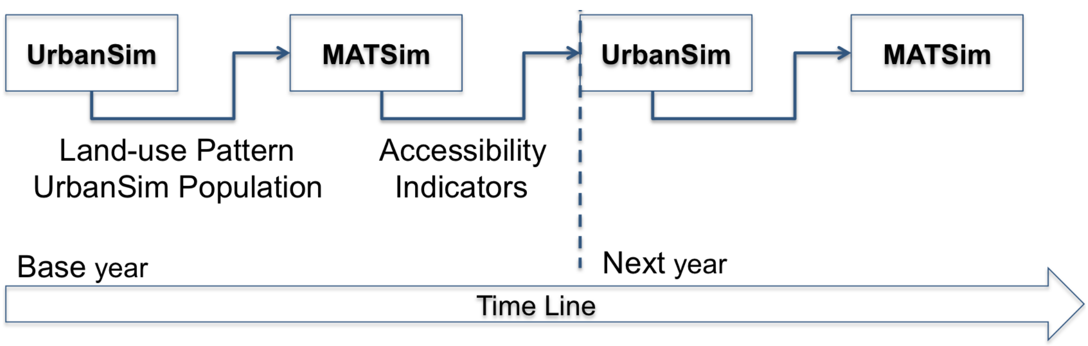

Figure 2: Interaction sequence between UrbanSim and MATSim.

2 Data Requirements

This section describes the data requirements for MATSim and UrbanSim in order to run the case studies.

UrbanSim is a very flexible tool. One consequence of this flexibility is that many variable names are not standardized. For example, an x-coordinate could be named “x_coord_sp” in one model, “xcoord” in another, and again something else in a third model. The reason why this works internally is that for every UrbanSim application the “models” (e.g. household location choice model) are specific to the application, and thus refer to the specific variable names that are in use for that specific application. Unfortunately, for the travel model plug-in this means that it needs to react to specifics of any given case study. This is why the following descriptions sometimes refer to specifics of the “Zurich parcel” or “Brussels zone” implementation. It is planned to make these items configurable before the end of the project.

2.1 UrbanSim Data Requirements

In order to create the input tables for MATSim UrbanSim requires certain data sets and attributes to reflect where a person lives and works. A compilation of these data sets and attributes is given below for the parcel and zone version of UrbanSim. Technically, persons are linked to residences and work places and these in turn are linked with a spatial reference or coordinate via unique id’s as illustrated in Figure 3 and Figure 4 for the UrbanSim parcel and zone version.

UrbanSim Parcel Models

For UrbanSim parcel models, e.g. the Zurich_parcel application, the following data sets and attributes in parentheses are required:

- persons (person_id, household_id, job_id)
- households (household_id, building_id)
- jobs (job_id, building_id)
- buildings (building_id, parcel_id)
- parcels (parcel_id, x_coord_sp, y_coord_sp, zone_id)[3]

[3] At this point, the variable names x_coord_sp and y_coord_sp are hard coded for parcel models that are to be integrated with MATSim as a travel model (see above). It is planned to make these variable names configurable before the end of the project.

- zones (zone_id)

UrbanSim Zone Models

For UrbanSim zone models, e.g. the Brussels_zone application, the following data sets and attributes in parentheses are required:

- persons (person_id, household_id, job_id)
- households (household_id, zone_id)
- jobs (job_id, zone_id)
- zones (zone_id, xcoord, ycoord)[4]

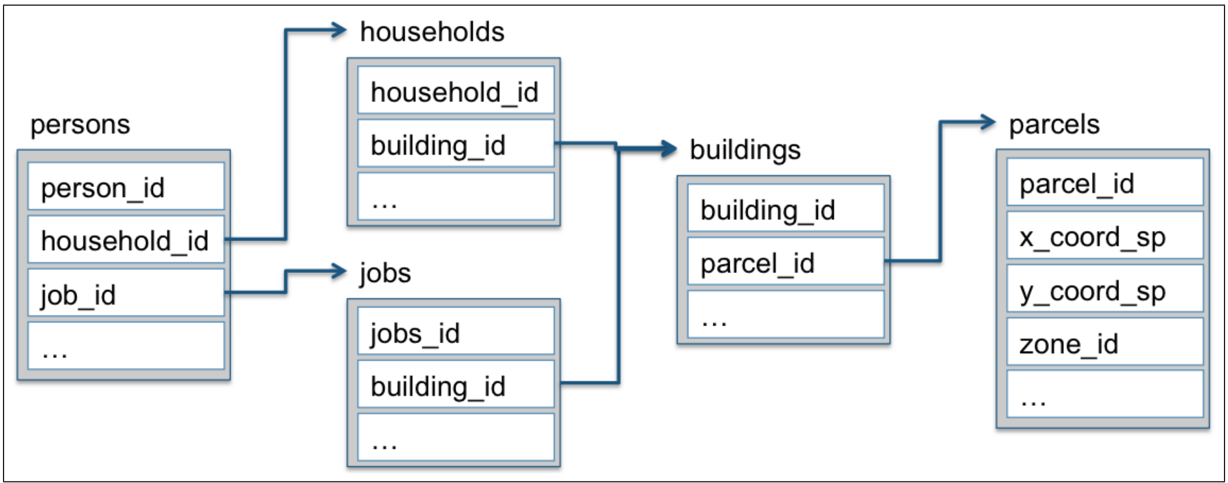

Figure 3: This illustrates the required data sets (grey boxes) and attributes (white boxes) for UrbanSim parcel models. The blue arrows indicate how these data sets are linked with each other via attribute names.

2.2 MATSim Data Requirements

The mandatory input for MATSim is a configuration file, a network file and the land-use and population pattern of the current UrbanSim simulation year. These components are introduced in the following.

Configuration File

The configuration file provides resources like the location of the network file and parameter

[4] At this point, the variable names xcoord and ycoord are hard coded for zone models that are to be integrated with MATSim as a travel model (see above). It is planned to make these variable names configurable before the end of the project.

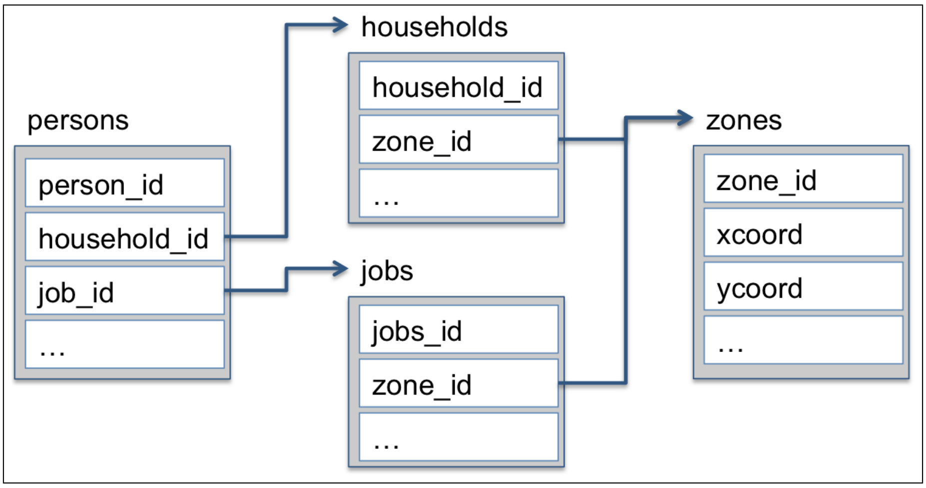

Figure 4: This illustrates the required data sets (grey boxes) and attributes (white boxes) for UrbanSim zone models. The blue arrows indicate how these data sets are linked with each other via attribute names.

settings that adapt the behaviour of MATSim. These settings needs to be done in the UrbanSim graphical user interface (GUI), as described in Section 4. Given these settings UrbanSim automatically generates a configuration file and passed it as an argument on each travel model execution to MATSim.

Network File

The network file gives information about the transport infrastructure. MATSim network files are XML[5] files that contains links and nodes each with its specific attributes such as the node coordinates or a description of a link. Links are defined by a start and an end node (from="<node-id>" and to="<node-id>"). Further important link attributes are length, capacity (vehicles per hour), free-flow speed in meters per second (freespeed) and number of lanes (permlanes). Recently an attribute specifying by which transport modes the link can be traveled was added (modes). An example network file is given in Listing 1.

Listing 1: An example for a MATSim network file.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE network SYSTEM "http://www.matsim.org/files/dtd/network_v1.dtd">

<network name="example">
    <nodes>
        <node id="0" x="505046.8125" y="137967.7969" />
        <node id="1" x="520580.9063" y="147882.7969" />
        <node id="2" x="594615.5" y="199259.2969" />
        ...
    </nodes>
```

[5]XML is an acronym for Extensible Markup Language

```

    <links capperiod="01:00:00" effectivecellsize="7.5"
      effectivelanewidth="3.75">
        <link id="0" from="0" to="1" length="6243.0"
          freespeed="27.77777777777778" capacity="4000.0" permlanes="2.0"
          oneway="1" modes="car" />
        <link id="1" from="1" to="0" length="6243.0"
          freespeed="27.77777777777778" capacity="4000.0" permlanes="2.0"
          oneway="1" modes="car" />
        <link id="2" from="1" to="2" length="949.0"
          freespeed="33.333333333333336" capacity="4000.0" permlanes="2.0"
          oneway="1" modes="car" />
        ...
    </links>
</network>
```

Land-Use & Population Pattern

The current land-use and population pattern are provided to MATSim as input by several data set tables (see Section 3). Such tables are including the residence and job location of each individual person in UrbanSim. Based on this information MATSim generates the traffic assignment, as described in Section 6.

3 UrbanSim Output

This section explains the MATSim input tables build by UrbanSim. It is assumed that UrbanSim contains the required data sets and attributes that are listed in Section 2. The tables are written in a tabulator-separated format.

UrbanSim Parcel Models

The UrbanSim output for parcel models, e.g. the Zurich application, consists of the following tables, see also Table 1:

- **Person Table**: This file includes a person id and the corresponding home and work parcel id for each individual UrbanSim person, where the parcel id's are referring to the parcel table described below.
- **Job Table**: This gives the job-, parcel- and zone id for each job in UrbanSim. The parcel id refers to the parcel table (next item).
- **Parcel Table**: This table consists of a parcel id, the parcel centroid given as x, y coordinates and the zone id in which the parcel is located. Given this information home and work locations can be located geographically, but also the structural zone centroid can be determined in MATSim by averaging over all parcel coordinates of an associated zone.

UrbanSim Zone Models

In case of UrbanSim zone models, e.g. the Brussels application, the following three tables are written; see also Table 2:

- **Person Table**: This file includes the person id as well as the zone id of the home- and work location of each individual UrbanSim person. The zone id's are referencing the zone table described below (last item).
- **Job Table**: This gives the job- and zone id for each available job in UrbanSim, where the zone id refers to the zone table (next item).
- **Zone Table**: This table stores the zone id and the x and y coordinates of the zone centroid for each UrbanSim zone, see also Figure 7. By means of this file home and work locations, given by the previous files, can be located in MATSim.

UrbanSim Parcel Output		
Table	**Indicators**	**Data Type**
Person data set table	person_id	long
	parcel_id_home	long
	parcel_id_work	long
Job data set table	job_id	long
	parcel_id_work	long
	zone_id_work	long
Parcel data set table	parcel_id	long
	x_coord_sp	double
	y_coord_sp	double
	zone_id	long

Table 1: Constructed output from UrbanSim parcel (e.g. Zurich scenario) for MATSim.

UrbanSim Zone Output		
Table	**Indicators**	**Data Type**
Person data set table	person_id	long
	zone_id_home	long
	zone_id_work	long
Job data set table	job_id	long
	zone_id_work	long
Zone data set table	zone_id	long
	xcoord	double
	ycoord	double

Table 2: Constructed output from UrbanSim zone (e.g. Brussels scenario) for MATSim.

4 MATSim4UrbanSim Configuration

An integral part of the integration is the joint configuration of both frameworks. In terms of usability a subset of MATSim parameters are embedded in the travel model configuration section as part of the UrbanSim configuration. This allows a convenient setup of both frameworks via the OPUS GUI and reduces the user's maintenance effort of keeping different configuration files consistent. A detailed description of the configuration and parameter settings for MATSim4UrbanSim can be found in the MATSim4UrbanSim users guide (Nicolai, accessed 2012).

In order to initialize MATSim properly UrbanSim takes the parameter settings from the travel model configuration section and generates a separate configuration file in XML format. This is done each time the travel model is called by UrbanSim. The following provides a brief overview about the software design issues behind this approach. A full description can be found in (Nicolai and Nagel, 2010):

The generated MATSim XML configuration is specified by an "XML Schema Document" (XSD). An XSD is an abstract collection of meta data about an XML document, or in other words a "formalization of the constraints, expressed as rules or a model of structure [...]" (van der Vlist, 2002). Technically, the XSD is used to generate customized XML parser's in MATSim and UrbanSim that reads, writes and validates XML documents. With this technique, called XML data binding, two important requirements are fulfilled:

- It allows a convenient adaption of the MATSim configuration to new requirements, for instance if additional configuration parameters are required, by just adjusting the XSD and regenerating the XML parsers on both sides.
- Moreover, each time a parser reads an XML document it is validated by the XSD, which ensures a robust and reliable communication between both frameworks.

5 MATSim Default Settings and Adjustments

In addition to the adjustable MATSim parameters in the OPUS GUI certain MATSim parameters are fixed or adjusted automatically:

- **Strategy Modules**: Three strategy modules are provided, which are **Route Choice** (10), **Departure Time Choice** (10) and **Plan Choice** (90), where the weights in parentheses (configurable via OPUS GUI) determine the probability that a strategy is applied to an agent. If for instance all three strategy modules have a weight of 100 each strategy has a probability of $33.\overline{3}\%$ being applied to an agent.

  The **Route**- and **Departure Time Choice** module are disabled for the last 20% of MATSim iterations. The so-called "mutation range" of the **Departure Time Choice** module is fixed to 2 hours. This means that agents can shift their departure times within this time window.

- **Network adjustments**: To preserve congestion effects the flow- and storage capacity of the road network are adjusted by a flow or storage capacity factor respectively according to the population sampling rate.

  The flow capacity gives the maximum number of vehicles per time unit that can pass a link (Nagel, 2007). The flow capacity factor is equal to the given $SamplingRate$.

  The storage capacity gives the maximum number of vehicles on a link (Nagel, 2007). The corresponding storage capacity factor is defined by

  $\frac{SamplingRate}{HeuristicFactor}$, where the $HeuristicFactor = \sqrt[4]{SamplingRate}$.

  The $HeuristicFactor$ is a fit function based on engineer heuristics. It aims to raise the storage capacity especially at low sampling rates (see Figure 5) to avoid network breakdowns caused by strong backlogs, which are described in Rieser and Nagel (2008).

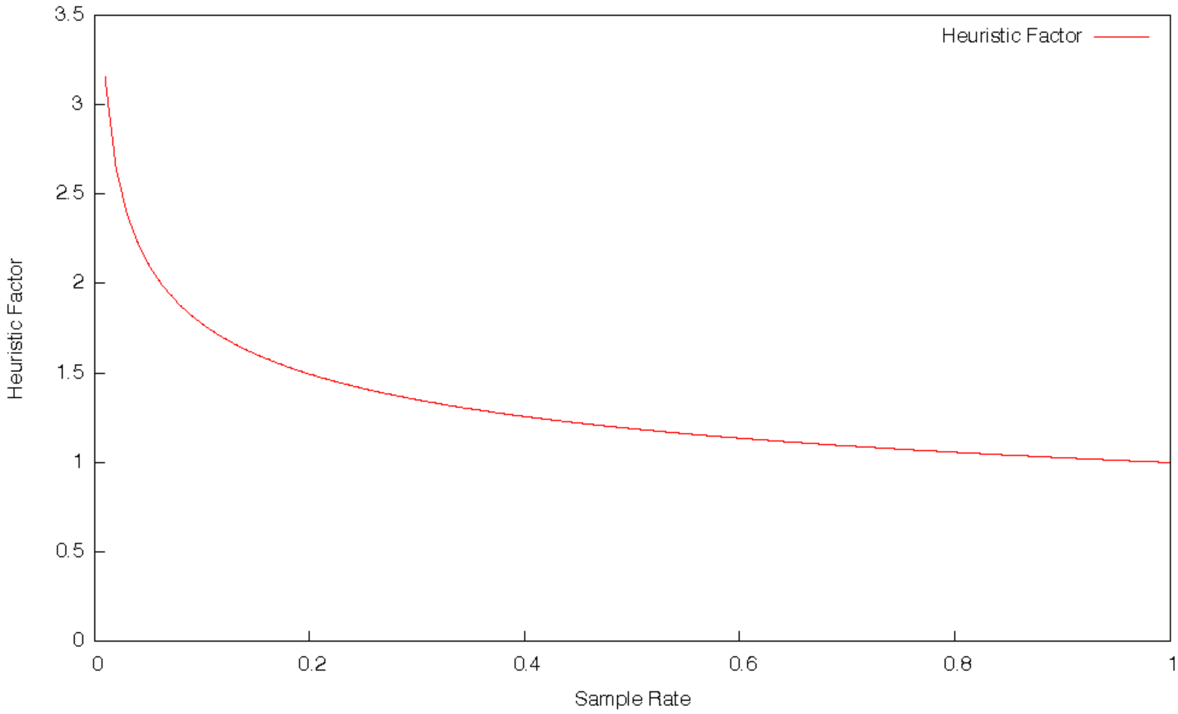

Figure 5: The heuristic factor raises the storage capacity especially at low sampling rates. For full samples, i.e. 100 percent, this factor becomes one.

6 MATSim Demand Generation and Mobility Simulation

After MATSim is initialized it starts with the initial demand generation and mobility simulation. At this point a brief overview is provided, for more details see (Balmer *et al.*, 2005). The following steps are processed:

- **Initial demand generation**: Given the input tables from UrbanSim (see Section 3), MATSim constructs agents. All agents independently generate daily plans with "home-to-work-to-home" activity chains based on their home and job locations in UrbanSim. A plan reflects a typical day of an agent.
- **Execution**: The mobility simulation executes all selected plans simultaneously on the road network.
- **Scoring**: All executed plans are scored by a utility function.
- **Re-planning**: Some of the agents obtain new plans for the next iteration by modifying existing plans with respect to two choice dimensions, route and departure time choice. With the default **strategy modules** settings, explained in Section 5, about 10% of the agents obtain new routes another approximately 10% obtain new activity starting and ending times. All other agents select between existing plans according to a logit model. The last 20% of the iterations route and departure time choice are switched off.
- **Analysis**: At the end of the iteration cycle MATSim obtains a congested road network. Based on this MATSim computes the access- and accessibility indicators as feedback for UrbanSim; see Section 7.

The repetition of the iteration cycle coupled with the agent memory, the capability to remember more than one plan per agent, enables the agents to improve their plans over several iterations (Balmer *et al.*, 2005).

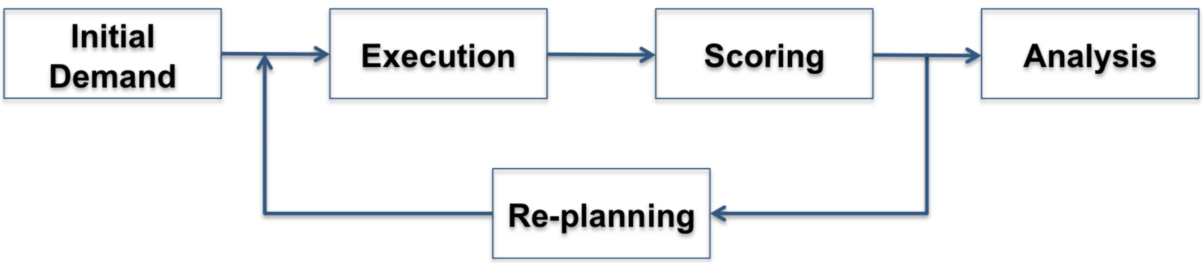

Figure 6: The simulation inside MATSim consists of an iterative loop with the following important steps: the initial demand generation, the execution of agent plans, a utility scoring function, the re-planning, were agents obtain new plans, and the calculation of access- and accessibility indicators as feedback from MATSim to UrbanSim.

7 Access- and Accessibility Computation in MATSim

MATSim computes several access- and accessibility indicators as feedback for UrbanSim such as zone-to-zone skims, individual agent-based performances as well as aggregated accessibilities per spatial unit, meaning zones or parcels. A general MATSim feedback overview is given in Table 3. In this context, the terms access and accessibility are used as follows: "access" refers to a two-point-value like travel time impedances between an origin-destination (OD) pair; "accessibility" just belongs to one location and thus refers to an aggregated single-point-value.

MATSim Computation	MATSim Output	Output Format	UrbanSim Data Set
Zone-to-zone Impedance Matrix	travel_data.csv	Comma-Separated Values	travel_data
Agent-Based Performance Feedback	persons.csv	Comma-Separated Values	persons
Zone-Based Accessibility	zones.csv	Comma-Separated Values	zones
Parcel-Based Accessibility	parcels.csv (under development)	Comma-Separated Values	parcels

Table 3: MATSim calculates zone-to-zone travel costs (like travel times, etc.), individual agent-based performances (including travel times and distances commuting from home to work and back) and aggregated accessibilities per spatial unit, i.e. per zone or parcel. UrbanSim uses this output to update the corresponding data sets as listed in the table.

Zone-to-zone Impedance Matrix

The zone-to-zone matrix is an origin-destination-matrix (OD-matrix) comprising travel times for different transport modes like car, bicycle and walk, generalized travel costs and vehicle trips for each pair of zones. An overview of these indicators can be found in Table 4.

To calculate these indicators zones are assigned to the road network by connecting the zone centroid to the closest node in the network. The coordinates of the zone centroids are either directly given in the MATSim input tables, as in case of UrbanSim zone models, or are determined by averaging over all parcel coordinates that belong to a zone like in UrbanSim parcel models; see also Section 3. Figure 7 depicts the zone centroids represented as blue dots at the example of the Brussels zone application.

The car travel times are based on link travel times from the congested MATSim road network. The generalized travel costs at this point consist of car travel time and toll (as time equivalent). Since the calculation of toll is not supported yet congested car travel times and generalized travel costs are equal. The walk travel time is based on the shortest path on the road network with a walking speed of 5km/h. The vehicle trip gives the number of trips for any pair of zones.

In case of within zones impedances, i.e. the origin and destination zone is the same, a minimal duration of 1.2 minutes for car travel time and 12 minutes for traveling on foot is assumed.

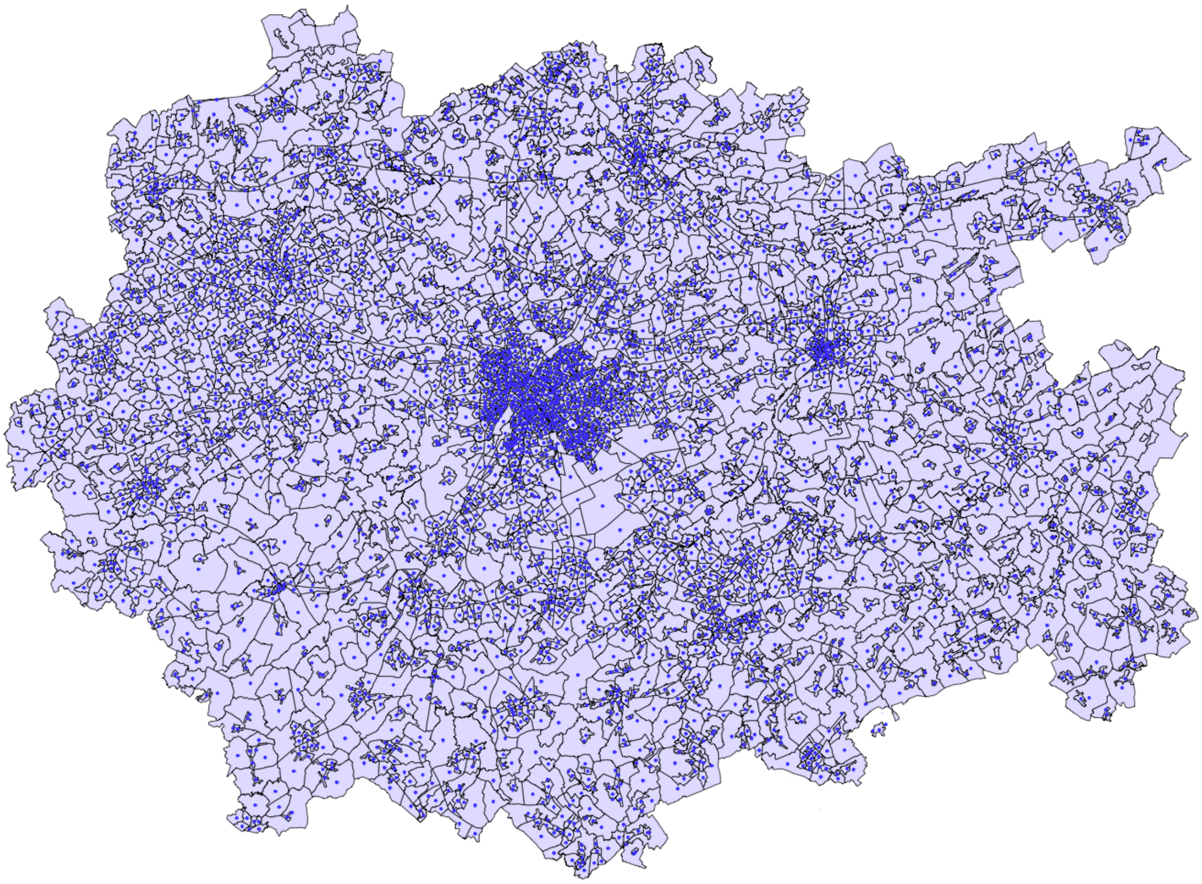

Figure 7: This illustrates the zone centroids (blue dots) at the example for the Brussels zone application.

Agent-Based Performance Feedback

This feedback contains the individual travel performance for each MATSim agent including congested car travel times and travel distances for both directions, commuting from home to work and back. In case of unemployed agents these indicators are -1. An overview of these indicators is provided in Table 5.

Accessibility Computation

MATSim measures accessibilities to opportunities such as workplaces. This is performed (i) for different spatial units, meaning zones or parcels, (ii) using different generalized costs of travel, like travel time or distance and (iii) for two modes, car and walk.

Currently, the only implemented accessibility measure is based on the so-called logsum term (e.g. Ben-Akiva and Lerman, 1985). This can be seen as a placeholder for other accessibility measures. It should be noted that accessibility measures that are not related to the transport system, such as “number of shops within 1km Euklidean distance”, can also be implemented directly in UrbanSim.

The logsum is a utility-based measure of accessibility reflecting the (economic) benefits, as the maximum expected utility, that someone gains from access to spatially distributed opportunities like work (Geurs and Ritsema van Eck, 2001; de Jong *et al.*, 2005). This measure includes both the number of opportunities (e.g. workplaces) that can be reached from a given location and the effort to get there (Geurs and Ritsema van Eck, 2001). The logsum term is defined as:

$$A_i := \frac{1}{\beta_{scale}} \ln \sum_k e^{-\beta_{scale} \, c_{ij}} \,, \tag{1}$$

where

- A_i is the accessibility at location i
- β_{scale} is the logit model scale parameter
- c_{ik} are generalized costs of travel in order to get from location i to location j

The generalized costs c_{ij} are composed of:

$$\begin{aligned} c_{ij} &:= (\alpha * ttime) + (\beta * ttime^2) + (\gamma * ln(ttime)) \\ &+(\delta * tdistance) + (\epsilon * tdistance^2) + (\zeta * ln(tdistance)) \end{aligned} \tag{2}$$

where

- $ttime$ are travel times in minutes
- $tdistance$ are traveled distances in meter
- $tcost$ are monetary travel costs
- α to ζ are marginal utilities converting travel times, distances and monetary costs respectively into utils.

The accessibility parameters α to ζ are separately configurable for different modes such as car, bicycle and walk, were β_{Scale} setting applies for all modes, see (Nicolai, accessed 2012). Table 6 depicts the MATSim default accessibility parameter settings.

As mentioned earlier accessibilities are calculated for zones and parcels:

- **Zone-Level**: Accessibility measures at a zone-level are using zone centroids as measuring points (origins), as shown in Figure 8 (left). In UrbanSim zone models like the

Brussels application zone centroids are directly given in the UrbanSim output tables; in case of UrbanSim parcel models, e.g. Zurich scenario, the structural zone centroid is determined by averaging all parcel coordinates that belong to a zone; see Section 3.

- **Parcel-Level**: In this approach the study area is subdivided into grid-cells of configurable size, where the resulting cell centroids serve as origins for the accessibility calculation; see Figure 8 (center). Once the calculation for each cell centroid is completed, MATSim interpolates the accessibility value for each UrbanSim parcel from adjacent grid-cell accessibility values.

In order to calculate the accessibility A_i, origin locations i and opportunity locations j are assigned to the MATSim road network. For every given origin i a so-called "least cost path tree" runs through the network and determines the best route to each opportunity location j by using the Dijkstra shortest path algorithm (Dijkstra, 1959). The best route from i to j depends on the given cost type such as link travel times or distances. Once the least cost path tree has explored all nodes, MATSim queries the resulting travel costs c_{ij} for all opportunities and calculates the accessibility as stated in Equation 1. A detailed description of the accessibility implementation can be found in Nicolai and Nagel (2012).

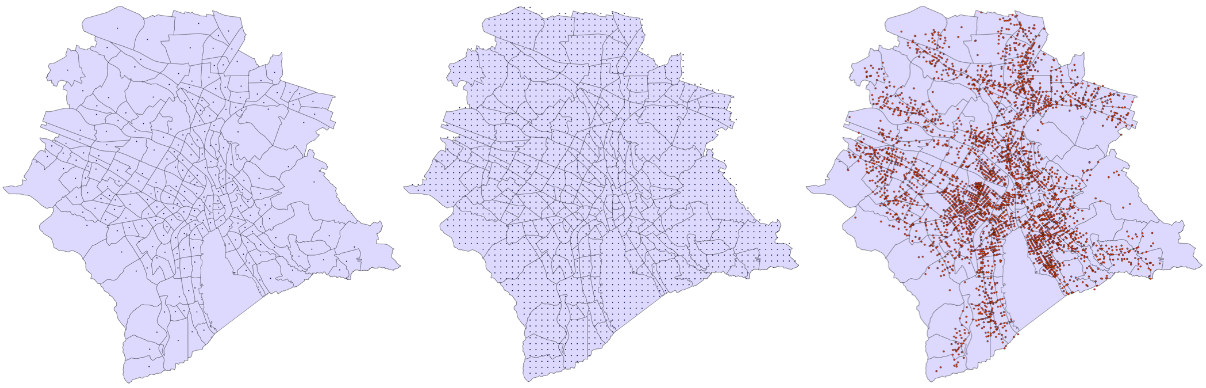

Figure 8: This illustrates the origins and destinations in accessibility calculation at the example of the Zurich parcel study area. For zone-based accessibility measures the zone centroid is used as a measuring point (left). In case of parcel-based accessibility measure, the study area is subdivided into grid-cells of configurable size using the cell centroids as origins for the accessibility computation (center). The red dots (right) depicts workplaces, which are aggregated to their nearest network node.

Zone-to-zone Impedance Matrix		
Indicator Name	**Description**	**Data Type**
from_zone_id	UrbanSim origin zone id	long
to_zone_id	UrbanSim destination zone id	long
am_single_vehicle_to_work_travel_time	Congested car travel time in minutes at 8a.m.	double
single_vehicle_to_work_travel_cost	This consist of car travel time and toll (as time equivalent), since no toll is implemented this is equal to congested car travel times	double
am_walk_time_in_minutes	walk travel time in minutes	double
am_pk_period_drive_alone_vehicle_trips	number of vehicle trips at 8a.m.	double

Table 4: The zone-to-zone impedance matrix includes congested car travel times, generalized travel costs, walk travel times and trips for any pair of zones.

Agent-Based Performance Feedback		
Indicator Name	**Description**	**Data Type**
person_id	UrbanSim person id	long
home2work_travel_time_min	Congested car travel time in minutes to get from home to work	double
home2work_distance_meter	Traveled distance in meter from home to work	double
work2home_travel_time_min	Congested car travel time in minutes to get from work to home	double
work2home_distance_meter	Traveled distance in meter from work to home	double

Table 5: This MATSim feedback contains for each UrbanSim person the individual travel performances commuting from home to work and back including congested car travel time and distance.

Default Accessibility Parameter Settings	
Parameter	**Description**
β_{Scale}	A scale factor, related to the scale parameter of a logit model. It is set to 2, which is the MATSim default setting. This parameter is configurable via the OPUS GUI.
α	By default α is -12 utils/hour. In MATSim terms, this is the sum of the marginal opportunity cost of time (typically -6 utils/hour) and the marginal additional disutility of travel (typically another -6 utils/hour)
β - ζ	These marginal utilities are set to zero by default. They are configurable via the OPUS GUI.

Table 6: Default accessibility parameters used by MATSim.

Zone-Based Accessibility		
Indicator Name	**Description**	**Data Type**
zone_id	UrbanSim zone id	long
car_accessibility	Workplace accessibility by car	double
walk_accessibility	Workplace accessibility by walk	double

Table 7: This returns the workplace accessibilities by car and walk for each UrbanSim zone.

Parcel-Based Accessibility		
Indicator Name	**Description**	**Data Type**
parcel_id	UrbanSim parcel id	long
car_accessibility	Workplace accessibility by car	double
walk_accessibility	Workplace accessibility by walk	double

Table 8: This gives the workplace accessibilities by car and walk for each UrbanSim parcel.

8 UrbanSim Data Set Updates

In the last step of the UrbanSim/MATSim interaction UrbanSim updates its data sets for the next year (iteration) by importing the MATSim feedback given in Table 3. The MATSim feedback is directly assigned to a identically named UrbanSim data set. For instance the MATSim output “travel_data.csv” is imported to the “travel_data” data set. A list of the updated or newly included indicators for each MATSim feedback/output can be found in the corresponding Tables 4, 5, 7 and 8.

Acknowledgments

This chapter benefited from discussions with K. Nagel.

9 References

Balmer, M., B. Raney and K. Nagel (2005) Adjustment of activity timing and duration in an agent-based traffic flow simulation, in H. Timmermans (ed.) *Progress in activity-based analysis*, 91–114, Elsevier, Oxford, UK.

Balmer, M., M. Rieser, K. Meister, D. Charypar, N. Lefebvre, K. Nagel and K. Axhausen (2009) MATSim-T: Architecture and simulation times, in A. Bazzan and F. Klügl (eds.) *Multi-Agent Systems for Traffic and Transportation*, 57–78, IGI Global.

Ben-Akiva, M. and S. R. Lerman (1985) *Discrete choice analysis*, The MIT Press, Cambridge, MA.

de Jong, G., M. Pieters, A. Daly, I. Graafland-Essers, E. Kroes and C. Koopmans (2005) Using the logsum as an evaluation measure – literature and case study, Research report commissioned by the Transport Research Centre [AVV] of the Dutch Ministry of Transport, Public Works and Water Management.

Dijkstra, E. (1959) A note on two problems in connexion with graphs, *Numerische Mathematik*, **1**, 269–271.

Geurs, K. and J. Ritsema van Eck (2001) Accessibility measures: review and applications, *Technical Report*, National Institut of Public Health and the Environment, RIVM, P.O. Box 1, 3720 BA Bilthoven, June 2001.

Miller, E., K. Nagel, H. Ševčíková, D. Socha and P. Waddell (2005) OPUS: An open platform for urban simulation, `http://128.40.111.250/cupum/searchpapers/detail.asp?pID=428`.

Nagel, K. (2007) *Multi-agent transport simulation (draft-in-progress)*, October 2007, `https://svn.vsp.tu-berlin.de/repos/public-svn/publications/kn-old/book/`. Draft-in-progress for a book.

Nicolai, T. W. (accessed 2012) *MATSim4UrbanSim – Guide on UrbanSim usage of the travel model plug-in*, `http://matsim.org/extensions/matsim4urbansim`. See matsim.org/extensions/matsim4urbansim.

Nicolai, T. W. and K. Nagel (2010) Coupling MATSim and UrbanSim: Software design issues, *Technical Report*, **10-13**. See www.vsp.tu-berlin.de/publications.

Nicolai, T. W. and K. Nagel (2011) Investigating accessibility indicators for feedback from a travel to a land use model, paper presented at *European Regional Science Association*

Conference, no. 11-09, Barcelona, Spain. See www.vsp.tu-berlin.de/publications.

Nicolai, T. W. and K. Nagel (2012) Coupling Transport and Land-Use: Investigating accessibility indicators for feedback from a travel to a land use model, paper presented at *Latsis Symposium 2012 – 1st European Symposium on Quantitative Methods in Transportation Systems*, no. 12-16, Lausanne, Switzerland. See www.vsp.tu-berlin.de/publications.

Nicolai, T. W., L. Wang, K. Nagel and P. Waddell (2011) Coupling an urban simulation model with a travel model – A first sensitivity test, paper presented at *Computers in Urban Planning and Urban Management (CUPUM)*, no. 11-07, Lake Louise, Canada. See www.vsp.tu-berlin.de/publications.

OPUS User Guide (2011) *The Open Platform for Urban Simulation and UrbanSim Version 4.3*, University of California Berkley and University of Washington, January 2011, `http://www.urbansim.org`.

Raney, B. and K. Nagel (2006) An improved framework for large-scale multi-agent simulations of travel behaviour, 305–347.

Rieser, M. and K. Nagel (2008) Network breakdown “at the edge of chaos” in multi-agent traffic simulations, *European Journal of Physics*, **63** (3) 321–327.

van der Vlist, E. (2002) *XML Schema.*, 1 st edn., O’REILLY, June 2002.

Waddell, P. (2002) Urbansim: Modeling Urban Development for Land Use, Transportation, and Environmental Planning, *Journal of American planning Association*, **68** (3) 297 – 314.

Wegener, M. (2004) Overview of land-use transport models, in K. Hensher, D.A.; Button (ed.) *Transport Geography and Spatial Systems*, no. 5 in Handbook in Transport, 127–146, Pergamon/Elsevier Science.