Michael Zilske, Andreas Neumann, Kai Nagel

OpenStreetMap for traffic simulation

OpenStreetMap For Traffic Simulation

Michael Zilske, Andreas Neumann, Kai Nagel

TU Berlin, Fachgebiet Verkehrssystemplanung und Verkehrstelematik

Abstract. Micro-simulations of traffic systems are becoming more important as highly disaggregated data, such as mobility diaries or GPS traces, become available. For accurate results, a high-quality model of the road network is required. Recently, OpenStreetMap has proven to be a valuable data source, often on par with and in some respects surpassing proprietary network models provided by administrations in usefulness.

Keywords: Traffic simulation

1. Introduction

MATSim, a software package for micro-simulation of traffic in urban systems, which has successfully been applied in transport policy analysis, is available on an open-source basis (Balmer et al. 2009). However, it is often difficult to obtain and maintain input data for such simulations. For instance, information about the road network needs to be collected from different government agencies and map providers, who often deliver it in non-standard formats, and datasets provided by different agencies are typically not explicitly referenced to each other.

In this paper, we describe our experience using OpenStreetMap (OSM) as a data source for traffic simulation studies, either on its own or augmented with additional data. The focus is on use cases in the preparation of simulation scenarios where OSM comes into play. MATSim couples a time step based queuing simulation of road traffic flow with a multi-agent travel demand model based on individual people who carry out daily plans. Both parts of the model can benefit from OSM as a data source.

2. Generating the Network

2.1. Road Characteristics

The queuing model of traffic flow (Gawron 2008) uses a graph-based model of the road network, where nodes are intersections and edges are road segments. The characteristics of each road segment are modeled as three parameters: *free speed*, *flow capacity* and *storage capacity*.

The flow capacity limits the number of vehicles which can leave a link in each time step, while the free speed sets a lower bound for the time each car has to spend on a link. Storage capacity is the maximum number of vehicles which a link can contain at any given time. When the link is full, the simulation does not add any more vehicles to a link. The congestion "spills back" upstream. Storage capacity is calculated from the link length and the number of lanes.

When such a network is constructed from OpenStreetMap, the link length is taken directly from the geometry. This is the only geometric information required for the model. After taking the link length into account, a road segment is simplified to a straight line, and intermediary nodes between road crossings are discarded. We use a custom tool which simplifies the network when it is already in our own data format. It has the option to block certain link and nodes from being removed, which is important for things like bus stops or links referenced by traffic counts (*Section 2.2*).

Free speed and flow capacity are characteristics of road types (FGSV 2009) which we identified with the values of the OSM `highway` tag. Free speed is not identical to the legal speed limit, but the two are related. For instance, while large roads within city limits typically have a legal speed limit of 50 km/h, the average free speed is decreased by traffic signals. An average of 25 km/h was determined for the primary road network of Berlin. During a green wave or when driving faster than the legal limit, this can at times rise to 44 km/h, but this is countered by poorly coordinated signals on other parts of the route. On the freeways (Autobahnen), on the other hand, the legal limit (which typically does exist within city boundaries) is seldom honored. Here, we defined the free speed to be 1.2 the legal limit. In areas where speed limit zones are extensively used (mostly to 30 km/h), time which is lost at unsignaled nodes governed by „right-before-left“ priority is often countered by driving in excess of the speed limit, leading to an overall factor of 0.8 for tertiary roads. On streets tagged as `living_street`, the legal speed limit is prescribed as „walking speed“, which in reality is almost never honored. In lack of better data, we assume 15 km/h. The parameters per highway tag which we used in our studies are summarized in the table below.

	Legal speed limit (km/h)	Average speed / Legal speed limit	Capacity (veh/h)	Number of lanes
`motorway`	100	1.2	2000	2
`motorway link`	60	1.2	1500	1
`trunk`	50	0.5	1000	2
`trunk link`	50	0.5	1000	1
`primary`	50	0.5	1000	1
`primary link`	50	0.5	1000	1
`secondary`	50	0.5	1000	1
`tertiary`	30	0.8	600	1
`minor`	30	0.8	600	1
`unclassified`	30	0.8	600	1
`residential`	30	0.6	600	1
`living street`	15	1.0	300	1

Table 1. Highway characteristics inside city boundaries

The queuing model uses directed edges, so for each road segment except for those which are tagged as `oneway`, two links are created.

The region of interest for our studies has typically been an urban area. The road network of the region of interested is extracted from `planet.osm`, or more conveniently from one of the excerpts made available by geofabrik.de, by means of the open-source tool Osmosis, which permits extracting the road network of the entire region of interest by specifying a bounding box for the city, and a tag filter selecting all highway types accessible by car. When traffic is later assigned to the network, it is helpful not to cut the network at the city limits, but to embed it into its environment by adding the primary roads of the commuting range of the city. It can then be determined by routing if and on which way a person commutes from A to B by passing through the city.

2.2. Working with Additional Data

To calibrate simulation parameters and to validate results, traffic counts are used. Local authorities obtain them by means of induction loops or traffic cameras. These data are typically referenced to a proprietary network used by the respective agency, or they only carry coordinates but no explicit reference to a road segment. For one of our studies, we manually referenced count stations to OSM nodes and saved the OSM node ID to the counts database. In rare cases, where no suitable node was available, we inserted a node into OSM specifically as a reference point for our data. It turned out that nodes are stable enough to enable this workflow. Still, with every update from OSM it has to be verified that the reference nodes still exist, and any changes to their location have to be monitored. This is not different from other data sources, where there are also no guarantees about map entities keeping any properties from one version to the next.

In another case, we obtained a large set of floating car data (FCD) within the study area. They were given as point measurements of velocity, georeferenced and annotated with the approximate driving direction, but not matched to road segments. We referenced each data point to the nearest network link which approximately matched the given direction and used it to update the free speed of the referenced road segment to the observed maximum speed.

Matching points to links is only a special case of the more general problem where a network generated from OSM must be augmented with data referenced to a different network. In a future project, we plan to evaluate using a general network matching algorithm to create a simulation scenario where the base data comes from OSM, but specific extra data like traffic signal plans or volume counts come from different network graphs which are semi-automatically matched to OSM.

2.3. Junction Layout

Apart from a higher tagging rate of attributes from which flow capacity, free speed and number of lanes can be inferred, traffic simulation applications would benefit from a detailed representation of junction layouts (Krajzewicz et al. 2005). In the queue model as described above, the link capacity controls the rate at which cars flow out of a link. In reality, different turn relations at nodes have different capacities, which among other things (like traffic signaling) depend on the number of turn lanes available. If this is modeled by decreasing the capacity of certain turn relations, this raises the question of how to handle the traffic going in other directions: If cars are removed from a link strictly on a first-in, first-out basis, a car going straight will be forced to wait behind a car going left which is kept by a capacity restriction on the turn, while in reality it may well be that turning traffic has its own lane and can be passed. On the other hand, if cars going in either direction actually share the same lane, this would again be inaccurate.

A straight-forward solution is to split the link into several ones, one for each lane. For this to work, it needs to be known how many lanes are available per turn relation, and also their length, since once a turn lane runs full, turning vehicles will spill back to the main lane. This kind of data is not yet available in OSM. Benjamin Schulz (2011), a student in our group, proposed a tagging scheme for this information as part of his thesis, and he also developed a graphical editor as a plug-in for the comprehensive

OpenStreetMap editor JOSM. It allows adding extra turn lanes to the links leading up to a node, specifying into which link each turn leads, and moving the beginning of the lane with a slider. The turn relation can lead through complex junctions consisting of multiple nodes and permits special cases like U-turn lanes. While the tagging scheme is an early proposal and the software is a prototype, we intend to develop it further and to use it to investigate if there is a demand for this kind of detailed information and a willingness on part of the community to contribute it once a convenient editor is available.

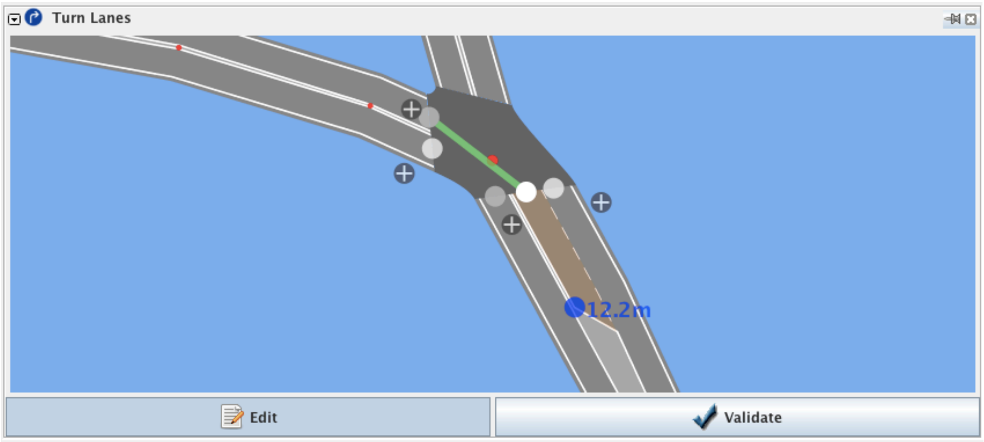

Figure 1. The turn lane editing pane in JOSM.

2.4. Public Transport

If the focus of a traffic simulation is on private cars, a user switching to public transport can be modeled as removing herself from the system and experiencing a travel time estimated by aggregate characteristics of the transit system, for example twice the time required by car on an uncongested network (Rieser et al. 2009).

This simple approach cannot take into account the varying transit accessibilities of different locations. A more fine-grained approach is possible if a complete transit schedule is available, with route definitions, stop locations and departure times. In this case, agents plan a route through the transit network, wait for the next departure and arrive at their destination according to the schedule, although the simulation can take vehicle capacities into account, as well as access and egress times. This means that effects like overcrowded vehicles, where passengers have to wait for the next departure, can be simulated, and vehicles are delayed if a larger number of passengers are boarding or leaving than the schedule accounts for (Rieser 2010). The users can be charged fares, so the utility of the ride can be weighed against using a private car. With the General Transit Feed Specification (GTFS), developed by Google, there is a data format available which contains all the required information. However, as of 2011, not one transit authority in Germany publishes a GTFS feed. Any study including public transport at schedule-level requires coordination with the local transit authorities, and manual data conversion.

While some public transport modes, like railway, normally operate without interaction with car traffic, this does not hold for busses, which can be caught in congestion together with private cars. If it is desired to simulate this kind of interaction, one requires not only the transit schedule, but the exact routes the busses take through the road network. Transit authorities do not publish their schedules referenced to the OSM network, of course. A GTFS feed can provide detailed geometric route information as an optional feature and at the discretion of the transit authority. But even if one is available, the shape of this route must still be matched against the OSM network to insert detailed link-based routes into the transit schedule.

Public transport tagging in OSM itself is in a proposal stage. Despite efforts, as of 2011, we have not been successful in extracting sufficiently consistent route information for even the main lines of the transit network in major German cities.

3. Generating the Synthetic Population

The synthetic population which models the transport demand is typically generated from survey data. The available data may be as fine grained as complete mobility diaries of a population sample, containing georeferenced activity locations, durations and the means of transport taken in between. However, this information is often not available. The least amount of data required for a meaningful simulation is home and work locations for the synthetic population. These can be drawn from a *commuter matrix*, which for each pair over a set of polygons contains the number of people commuting from a home location in polygon 1 to a work location in polygon 2 on a typical work day. These polygons are typically administrative areas.

Depending on the scale and the desired accuracy of the simulation scenario in question, it may be sufficient to randomly draw points from the entire administrative area and designate them as home and work locations for a person. In our simulation, activity locations are on links, so each randomly drawn point is matched to its nearest link. This approach is sufficient if the scale of the given polygons matches the simulated network level. For example, if only commuter traffic on the primary network is simulated, locations may be drawn from cities or counties without much consideration for different land use, since the locations are snapped to the next primary road. On the other hand, if commuter data and polygons are available on the level of city quarters or even building blocks, the entire road network may be used without any ill effects, such as small roads becoming unrealistically congested because the simulation places homes where, in reality, nobody lives.

In this process, OSM can be used in two different respects: First, the administrative areas themselves may be obtained from OSM. In real-life data obtained from local governments, it is interestingly not always the case that the polygons referenced by the commuter matrix are delivered in a common data format or even with a common coordinate reference system. If, however, the reference is made using a common identifier as a key (for example German administrative areas, which are multipolygon relations tagged with `de:amtlicher_gemeindeschluessel`), the shape may be taken directly from OSM, which saves writing a converter for the specific data.

Secondly, OSM may help to bridge different levels of aggregation. For instance, if commuter data is only available at the level of cities and counties, but a simulation of the entire road network is desired, drawing locations evenly is problematic, for the reasons stated above. This could be solved by employing a distribution which takes land use or even buildings into account, as available in OSM. A first step in this direction is to restrict the area from which activity locations are drawn to certain values of the `landuse` tag, or to assign weights to different values, so that home locations are concentrated in areas tagged as `landuse=residential`. How to refine this, and in particular how to deal with the problem that the availability of this information may vary widely within the area of one simulation scenario, will be the subject of future work.

4. Visualization

An agent-oriented traffic simulation produces trajectories of individual vehicles or pedestrians. The simulation environment MATSim features an interactive visualization tool which can display the current location of each agent during a simulation. This is an important tool for the user in order to spot regions with implausible traffic patterns caused by errors in link attributes or in the population structure.

Agents are rendered as discs moving over a simple schematic view of the network graph, where links are lines with a width corresponding to the number of lanes. A user who is familiar with the simulated scenario will be able to identify locations from the road layout alone. However, for a detailed analysis, it may be preferable to use a properly rendered street map as a background image.

We were able to extend our visualization tool with a background layer of OSM tiles using the JMapViewer component. In principle, this would also be usable if the network data for the simulation were taken from a different source. However, as with any map overlay, this would lead to inaccuracies. When the cars move on the same network which is rendered as a background, the visual impression is much clearer.

5. Conclusion

In this paper, we described a workflow for generating multi-agent traffic simulation scenarios based on OpenStreetMap. For detailed, calibrated studies, additional data will be necessary, but in principle, it is possible to set up a basic simulation based only on OSM and a commuter matrix. While it is still difficult to link public transport information to OSM, the ongoing efforts to establish a tagging scheme for transit routes is promising. Finally, using a network for the simulation which at the same time can be rendered as a high-quality background map has substantial benefits for the visualization of results. We think that Volunteered Geographic Information will play a large part in making it easier to set up and maintain large-scale traffic simulation scenarios.

The software component which converts OSM data to a MATSim network is part of the MATSim distribution. The JOSM plugin for turn lanes has been committed to the JOSM plugin repository by its author.

References

Balmer M, Rieser M, Meister K, Charypar D, Lefebvre N, Nagel K, Axhausen K (2009) MATSim-T: Architecture and Simulation Times. Multi-Agent Systems for Traffic and Transportation, IGI Global, 57-58

Forschungsgesellschaft für Straßen- und Verkehrswesen (2009) Handbuch für die Bemessung von Straßenverkehrsanlagen, FGSV Verlag GmbH

Gawron C (1998) An Iterative Algorithm to Determine the Dynamic User Equilibrium in a Traffic Simulation Model. International Journal of Modern Physics C 9(3):393–407

Krajzewicz D, Hertkorn G, Ringel J, Wagner P (2005) Preparation of Digital Maps for Traffic Simulation; Part 1: Approach and Algorithms. Proceedings of the 3rd Industrial Simulation Conference 2005:285–290

Pendyala R.M., Development of GIS-Based Conflation Tools for Data Integration and Matching. Research Center, Florida Department of Transportation

Rieser M, Grether D, Nagel K (2009) Adding Mode Choice to a Multi-Agent Transport Simulation. Transportation Research Record 2132:50–58

Rieser M (2010) Adding Transit to an Agent-Based Transportation Simulation. Dissertation, TU Berlin

Schulz B (2011) Abbiegespuren und Kreuzungslayout in OpenStreetMap. Bachelor's Thesis, TU Berlin

Strippgen D (2009) Investigating the Technical Possibilities of Real-time Interaction with Simulations of Mobile Intelligent Particles. Dissertation, TU Berlin