

# Fully Agent-based Simulation Model of Multimodal Mobility in European Cities

Michal Čertický, Jan Drchal, Marek Cuchý, Michal Jakob

Agent Technology Center

Faculty of Electrical Engineering

Czech Technical University in Prague

Email: {certicky, jakob, cuchy}@agents.fel.cvut.cz, drchajan@fel.cvut.cz

**Abstract**—Even though the agent-based simulation modelling has become a standard tool in transport research, current implementations still treat travellers as passive data structures, updated synchronously at infrequent, predefined points in time, thus failing to cover within-the-day decision making and negotiation necessary for cooperative behaviour in a dynamic transport system. Leveraging the fully agent-based modelling approach, we have built large-scale activity-based models of multimodal mobility covering areas up to thousands of square kilometres and simulating populations of up to millions of inhabitants of several European cities. Citizens are represented by autonomous, self-interested agents which schedule and execute their activities (work, shopping, leisure, etc.) and trips in time and space. Individual decisions are influenced by agent's demographic attributes and modelled using the data from mobility surveys. The model is statistically validated against origin-destination matrices and travel diary data sets.

**Keywords**—mobility; modelling; simulation; agent-based simulation; activity-based modelling; transport

## I. INTRODUCTION

Increased complexity and frequency of communication and interactions in mobility systems changes not only the character of the systems but also the requirements on techniques and tools for their modelling. The agent-based simulation and activity-based modelling of travel demand have already become standard tools in transport research [5]. Current solutions manage to model traditional multimodal mobility systems by employing the paradigm of agent-based simulation in a broad sense. However, they treat travellers as passive data structures updated synchronously by central modules at infrequent, predefined points in time. Therefore they fail to cover frequent within-the-day decision making and negotiation that is necessary for cooperative behaviour in a dynamic transport system and for emerging interaction-rich mobility services, supported by increased pervasiveness of information and communication technology (ICT).

In order to model mobility systems with a wide variety of complex agent-to-agent and agent-to-environment interactions, it is often useful to fully leverage the conceptual foundation of multi-agent systems by employing the *fully agent-based approach*. In fully agent-based models, the agents (citizens) can communicate, adapt, make decisions and replan at any point in time, in order to react to perceived events immediately.

The fully agent-based, open-source simulation framework AgentPolis<sup>1</sup> has been designed and developed to support modelling of the mobility systems with frequent or complex interactions and decision making. Various models have already been built on top of the AgentPolis platform, including several classes of real-time ridesharing schemes, taxi auctions with dynamic pricing and negotiation, urban parcel logistics and game-theoretic models of public transport fare inspection [11].

This paper introduces the most recent and most comprehensive application of AgentPolis platform so far - a large-scale activity-based models of multimodal mobility, covering areas up to thousands of square kilometres and simulating populations of up to millions of inhabitants of European cities. The models aim to realistically reproduce travel in a multimodal urban transport system in a bottom-up manner, employing the fully agent-based and activity-based modelling approach, where autonomous, self-interested agents schedule their activities (work, shopping, leisure, etc.) and trips in time and space. The scheduling process is loosely based on the ALBATROSS scheduler [1], respecting its psychological foundations. Individual decisions taken by the agents are influenced by their demographics and modelled using the data from travel diary surveys.

The models of Barcelona, Helsinki and Milan were used as part of the Policy Optimiser in Mobility Policy Framework – a set of tools for optimizing and monitoring eco-friendly mobility policies within SUPERHUB project [2]. They allowed the framework users to analyse how potential mobility policies (e.g. changing the public transport schedules or modernizing the vehicle fleet) would affect the performance indicators like pollutant emissions or modal split.

After we finish summarizing the related work in Section II and explaining the activity-based modelling approach in Section III, we give a high-level overview of our model (Section IV). Sections V and VI describe in detail how we model the environment and the citizens and Sections VII and VIII enumerate modelled areas and explain how the model was validated.

---

<sup>1</sup><http://agentpolis.org>

## II. RELATED WORK

**Analytical Modelling:** Since 1970s, we have seen numerous attempts to study mobility and transport systems by analytical modelling. An extensive overview of analytical modelling methodology, along with relevant mathematical background, can be found in a monograph by Ortuzar and Willumsen [6]. Early models of mobility systems were largely based on mathematical programming and continuous approximations. The former technique relied on detailed data and numerical methods, whereas the latter relied on concise summaries of data and analytic models. Geoffrion [8] advocated the use of simplified analytic models to gain insights into numerical mathematical programming models. In a similar spirit, Hall [9] illustrates applications of discrete and continuous approximations, and notes that continuous approximations are useful to develop models that are easy to interpret and comprehend by humans.

**Simulation Modelling** Analytic models were unfortunately often too abstract for expressing relevant aspects in the structure and dynamics of various transport systems. To deal with this shortcoming, the paradigm of simulation modelling was adopted by the transport research community and has been employed in parallel with the analytical approaches. In 1969, Wilson [18] conducted a pioneer simulation-based study of the influence of the service area, demand density and the number of vehicles on the behaviour of a transport system. Simulations have since then been extensively utilized in the research of transport and mobility, as a powerful tool for the analysis of system's behaviour.

**Agent-based Simulation** In the above simulations, the system's behaviour was centralized – governed in a top-down manner by a system control mechanism or a single entity. Any self-initiated interactions, communication or negotiation among individual actors (e.g. travellers) was impossible, severely limiting their level of autonomy. To overcome these limitations, a new paradigm called *agent-based simulation* was introduced. Agent-based simulation has proven to be a highly valuable tool, especially when studying complex self-organizing systems in many domains [12]. Mobility systems modelled under this paradigm are implemented as multi-agent systems – i.e. composed of autonomous entities termed *agents* situated in a shared environment which they perceive and act upon, in order to achieve their own goals.

Agent-based transport simulation models are capable of simulating the travel activity of millions of individuals. The simulation performed at the level of individual travellers (citizens) and households enables highly granular analysis in time, space and demographic segments. Disaggregation avoids the errors and biases associated with generalization and averaging. In contrast to simulation tools for modelling road traffic, which have long tradition of decades, the tools supporting agent-based activity-based approach are still a relatively recent development. Two such tools are of high relevance to the objectives of agent-based simulation of multimodal mobility: *MATSim*<sup>2</sup> [4] and *TRANSIMS*<sup>3</sup> [16] (Transportation Analysis and Simulation

System). An open-source framework for agent-based mobility modelling, *MATSim*, takes an assumption that behaviour of humans is near-optimal and converges to equilibrium over time. It runs numerous subsequent simulations in order to discover such optimal behaviour for the simulated agents using genetic algorithms. *TRANSIMS* is an integrated set of tools for regional transport system analysis with a very similar experiment life cycle as *MATSim*, consisting of subsequent behaviour planning, routing and simulation stages.

**Fully Agent-based Simulation** Although termed agent-based and supporting individual-level modelling, simulations like *MATSim* and *TRANSIMS* treat travellers as passive data structures that can only be updated synchronously by central modules at infrequent, predefined points in time. This reduces the ability to add new types of agents to the model and to represent dynamic, interaction-rich and multi-agent behaviour. To overcome these limitations, a *fully agent-based* modelling approach was proposed [11], where entities of a transport system are autonomous agents with continuous, asynchronous control modules and the ability to interact freely with the environment and other agents at any point in time. This approach reduces coupling and allows modelling scenarios in which agents adjust their plans at any time during the day based on their observations of the environment and/or communication with other agents. The fully agent-based modelling approach can be realized using the open-source simulation platform *AgentPolis* [11].

## III. ACTIVITY-BASED MODELLING APPROACH

Regardless of the selected modelling paradigm, one of the hardest challenges in transport system modelling is the accurate estimation of transport demand. In contrast to traditional four-step demand models [14], which use trips as the fundamental modelling unit, agent-based models take a broader context into consideration and employ so-called *activities* (e.g. work, shop, sleep) and their sequences to represent transport-related behaviour of the citizens. The key idea behind such an *activity-based* approach [15] is that *travel is a derived demand* occurring due to the necessity of the agents to satisfy their needs through *activities* performed at different places at different times. These activities are arranged in time and space into sequential (daily) schedules that maximize an agent's utility. Trip origins, destinations and times are endogenous outcomes of activity scheduling. The activity-based approach considers individual trips in context and therefore allows representing realistic trip chains. It also allows considering inter-personal linkages and incorporating longitudinal adaptation effects. Finally, modelling decisions and utilities at an individual level also makes it easier to incorporate the impact of mobility policies on traveller choices and, consequently, on the operation of the transport system.

## IV. MOBILITY MODEL OVERVIEW

The model of multimodal mobility presented in this paper takes the activity-based modelling approach to travel demand

<sup>2</sup><http://www.matsim.org/>

<sup>3</sup><http://code.google.com/p/transims/>

generation and the fully agent-based approach to actual simulation modelling. It is implemented on top of the AgentPolis simulation platform.

### A. Model Architecture

Within the model (see Figure 1), the agents execute their planned activities and affect the environment by their actions and macro-actions and are notified about the changes via sensors. They can also communicate using the available communication protocols.

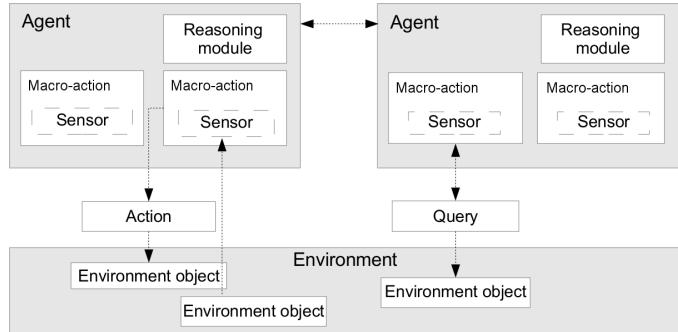


Figure 1. Internal architecture of the AgentPolis agent-based mobility model.

**Macro-actions:** Macro-actions are reactive control structures used as the building blocks of agent's behaviour (walking, vehicle driving, etc.). They affect the environment by invoking Actions (see below). Macro-actions can access the information about the environment via Sensors and Queries (see below).

**Actions:** Actions represent lower-level, simpler interactions that directly change the environment (moving to an adjacent location, entering vehicle, etc.). Actions have associated logic determining their durations and how exactly they affect environment objects or agents.

**Environment objects:** Environment objects (e.g. vehicles) primarily hold observable information about their own state (e.g. vehicle capacity, speed or current location) and can be affected by agent's Actions, according to predefined logic.

**Sensors and Queries:** Agent (or agent's macro-action) can access the information about an environment object via Sensors or Queries. Sensors are activated by environment objects (information push), whereas Queries are used by the agents to ask for the current state of the Environment object (information pull).

**Communication protocols:** Used by the agents to communicate with other agents. Two kinds of communication protocols are available at this time: *I-to-1 messaging* and *I-to-many messaging*.

All the actions, macro-actions, environment objects, sensors, queries and communication protocols, along with additional modelling elements, are available as part of the AgentPolis simulation platform. The open-source platform written in Java can be accessed as part of the Flexible Mobility Services Testbed [3]<sup>4</sup>.

<sup>4</sup><http://github.com/agents4its/mobilitytestbed/>

### B. Input: Data Sources

Our model of multimodal mobility is data-driven and combines various types of data and uses them for three tasks:

- 1) initialization of the environment model,
- 2) synthesis of simulated population (agents),
- 3) model validation.

#### Environment Initialization:

The data required to initialize the environment model are quite heterogeneous. Specifically, it is used to build a model of:

- *Modelled area infrastructure*, using the following data:  
 1) *Transport network* (roads, railways, etc.). The physical part of the transport network is based mainly on the Open Street Maps (OSM) - a publicly accessible maps database<sup>5</sup> released under Open Data Commons Open Database Licence (ODbL). OSM uses a topological data structure with four core elements: Nodes, Ways, Relations and Tags. It is usually distributed in Extensible Markup Language (XML) format.  
 2) *Transport restrictions* (speed/mode limits, etc.). Non-physical properties of the transport system, such as speed limits, are also extracted from OSM.  
 3) *Points of interest* (schools, shopping places, residential areas, commercial areas, etc.). Points of interest (POIs) are initialized using two additional publicly accessible data sources, in addition to OSM: FourSquare API<sup>6</sup> and Google Maps API<sup>7</sup>.
- *Vehicles*. Parameters of vehicle types, e.g. the length of the vehicle, fuel or electricity consumption and emission production, are contained in user-provided custom XML files along with statistics about the entire vehicle fleet (e.g. overall number of vehicles, proportion of individual types of vehicles etc.).
- *Public transport (PT)*. In addition to actual PT vehicles, the model contains public transport stations, routes, timetables and individual vehicle trips specifying sequences of stations and arrival/departure times. All this is loaded from General Transit Feed Specification (GTFS)<sup>8</sup> files. GTFS feed usually contains several CSV-like (Comma Separated Value) text files compressed in a ZIP archive. Each file models a particular aspect of transit information: stops, routes, trips, and other schedule data.

#### Population Synthesis:

Additional types of data are needed to generate the simulated population (agents) with all their demographic attributes and assign them into shared households. Data sources for this purpose are specific for every modelled area. In general, we use census data with varying levels of aggregation and spatial granularity and combine them to generate a synthetic

<sup>5</sup><http://openstreetmap.org/>

<sup>6</sup><http://developer.foursquare.com/>

<sup>7</sup><http://developers.google.com/maps/>

<sup>8</sup><http://developers.google.com/transit/gtfs/reference>

population that resembles the reference real-world system the most. Census data are usually available from the office of statistics of a given country in a simple format such as CSV.

#### Validation:

We perform a statistical validation to evaluate the behaviour of the agents and the whole system and measure how well it corresponds to the reference modelled system. The validation process requires two additional sources of the data:

- 1) *Travel diaries* obtained by long-term surveys (taking up to several days), during which participants log all their trips. The data contain anonymized demographic information about participants and a collection of all their trips.
- 2) *Origin-Destination Matrices* – two-dimensional square matrices specifying the number trips travelled between every combination of origin and destination locations during a specified time period (e.g. one day or one hour). Origin and destination locations are usually predefined zones. O-D matrices may be obtained by roadside monitoring, household surveys or derived from global system for mobile communications (GSM) mobile networks.

#### C. Output: Key Performance Indicators

The model needs to be able to assess the behaviour of the urban mobility system under different circumstances specified by the input data. Technically, this functionality is implemented by evaluating a set of key performance indicators (KPIs). The list of KPIs supported by our models is the following:

- Travel times: total sum of travel times, average time of one trip.
- Modal split: percentage of legs by mode, trip purpose and citizen profile, average trip duration by mode, total sum of travel times by mode, number of park-and-ride trips.
- Public transport (PT): average time deviation from PT schedule, average wait time on PT stations.
- Ecology: total pollutant emission by the whole system, total fuel consumption per mode, total energy consumption per mode.
- Comfort: average trip duration, average cruising time when parking, average number of PT transfers per journey, average walk time to PT station.

## V. ENVIRONMENT MODEL

The description of mobility model is divided into two parts: the model of the environment (described in this section) and the model of the agents acting in it (the following section).

The environment model needs to support both the supply and demand for the transportation. Supply is enabled by the *transport infrastructure* model, *vehicles* model and *public transport* model. Demand for the transportation is supported by the model of *points of interest*.

#### A. Transport infrastructure

The transport infrastructure is represented by a series of interconnected oriented graphs corresponding to individual transport modes: *road graph*, *pedestrian graph*, *metro graph* and *tram graph*. The intersections are represented by graph nodes and road/walkpath segments or rails between them by oriented edges (two-way road is represented by two separate edges with opposing orientation). Selected nodes are mapped to nodes of other graphs, allowing the agents to change transport modes. In addition to simulation graphs, there is a *planner graph* – a more complex, time-aware graph structure used by the multimodal journey planner included in AgentPolis.

The graph structures are built using the OSM and GTFS data sets. Data importers of the AgentPolis platform filter and cross-reference these heterogeneous inputs. For example, the OSM maps represent the transport network in unnecessary detail, so it is important to filter out the unnecessary nodes to improve the computational performance. Also, the *tram graph* needs to be synchronized with the *road graph*, so that the trams and cars can share the same space on the road (the graphs need to have shared edges).

#### B. Vehicles

Vehicles are another important part of the environment – either public transport vehicles such as buses, trams or trains, or private vehicles such as cars, bikes or motorcycles. Each vehicle is defined by a set of properties including capacity, length in meters, average speed, pollutant emission values and its fuel/electricity consumption. All the types of vehicles with corresponding properties, as well as distributions of vehicle instances among these types are defined in the user-provided VehicleModel XML file (specific for each modelling area).

#### C. Public Transport

The public transport (PT) model is built using the GTFS data.<sup>9</sup> The following information is used to initialize corresponding parts of the model:

- Stations: Places where passengers can be picked up or dropped off by scheduled PT vehicles. GTFS contains station ids, names, geographic coordinates, and types (e.g. bus or tram).
- Routes: A route is a group of trips with assigned route number and the type (e.g. bus or tram). The routes are also used to initialize a special type of agents - PT drivers and their daily plans.
- Trips: A trip is a planned movement of a PT vehicle through a sequence of stations that pertain to a specific route. They have an arrival and departure times specified for each station in a sequence.
- Operating dates: A specification of availability of each PT trip.

<sup>9</sup>Full reference of GTFS is available at <https://developers.google.com/transit/gtfs/reference>

- Connections: Time intervals needed to make a transfer from one route to another at a specific station.

At the implementation level, a GTFS feed contains several CSV-like text files compressed in a ZIP file. Each file models a particular aspect of transit information: stations, routes, trips, and other schedule data.

#### D. Points of Interest

Environment model also needs to cover points of interest (POIs) to support activity-based transport demand generation. POI in our model is a data structure holding the following information about a single location where agents can execute activities: 1) latitude-longitude coordinates, 2) types of activities that can be executed at a given POI, 3) POI opening hours, and possibly 4) additional information such as POI's name, attractivity etc.

POIs represent all the workplaces, residential locations, leisure time venues such as restaurants, cinemas, tourist attractions, etc. We extract and merge them from publicly available sources in the following three-step sequence:

- 1) Collect the initial set of POIs from OSM maps by searching for tags having specific keys (e.g., *amenity*, *landuse*, *building*, *tourism*, *shop*). As an example: OSM tags having *amenity* set to *school* are used to create school POIs while tags with *shop* = *supermarket* are used to create shop POIs.
- 2) Collect additional POIs by making HTTP requests to the public Foursquare API. Foursquare API provides a large database of POIs (these are called “Venues” in the Foursquare terminology). Each POI comes with a collection of metadata including its location, opening hours (in some cases), numbers of daily check-ins by the users of Foursquare mobile application (check-ins can be used as a measure of attractivity) and the category, which our script uses to assign an activity types to POI by a number of hard-coded rules. New POIs either expand our POI set or are merged with the ones from previous step if their location and name are similar enough.
- 3) Finally, we use Google Maps API to fill in the missing information about the collected POIs - especially the opening hours. This step is relatively slow due to Google Maps API restrictions.

## VI. AGENT MODEL

### A. Agent's Architecture

The population of citizen agents forms a cornerstone of the simulation system. The citizen agent is a relatively complex software component – see the architecture depicted in Figure 2.

The agents are organised in *households* which allows us to simulate cooperative behaviour and interactions between them (e.g. escorting children to school, cooperative planning of shopping or joint leisure time activities).

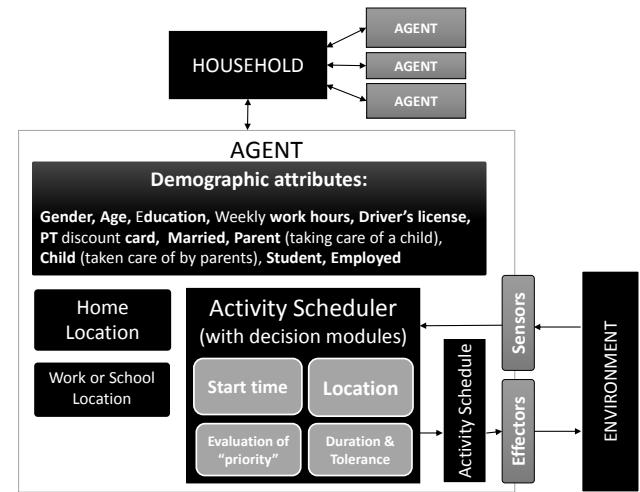


Figure 2. Architecture of the simulated citizen agent.

Agent's *sensors* (implemented by AgentPolis *sensors* and *queries*) are used to access all the information about the environment needed to create daily *activity schedules*, while *effectors* (implemented by *actions* and *macro-actions*) are used to execute them (e.g. initiate transport to reach an activity). Each agent also has a set of *demographic attributes* (gender, age, education, etc.) as well as fixed *home* and *work/school* locations. All these descriptors are stochastically sampled by the population synthesizer using the data available for a given scenario (see subsection VI-B for details).

The most important part of the agent structure is the *Activity Scheduler* (described in subsection VI-C).

### B. Population Synthesis

The population of agents needs to be synthesized during the model initialization phase by the following four-step process:

- 1) *agent* generation,
- 2) *household* generation,
- 3) agent to household *assignment*,
- 4) *location* selection.

The agents and households are generated in space, using a multi-level area hierarchy (with the *country* on the highest level and the *statistical area*<sup>10</sup> on the lowest). Multiple levels of spatial granularity are needed because the input data sources are in general available with varying degrees of aggregation.

The citizen agent needs two spatial attributes and a set of demographic attributes. The complete list with allowed values is summarized in Table I.

In the first step, we generate the collection of agents ignoring *home/work/school* locations based on the available aggregated data. The source is usually a census data set for

<sup>10</sup>Statistical area is the smallest area covered by available census data. It usually covers up to 400 citizens.

Table I. AGENT ATTRIBUTES AND THEIR POSSIBLE VALUES.

Attribute	Values
Age	Non-negative integer
Gender	{Male, Female}
Marital status	{Single, Married, Divorced, Widowed}
Education	{Basic, Secondary without final exam, Secondary with final exam, Higher professional, University}
Economic activity	{Student, Working, Inactive}
Driver's licence (car) ownership	Boolean
PT discount card ownership	Boolean
Bike ownership	Boolean
Home location	Node ID
Work/school location	Node ID or NULL

the given modelled region containing the attribute statistics with fine spatial granularity (up to the level of statistical areas with only a few dozens of inhabitants).

We use similar input data sources for household generation. The census data contain three classes of households (*full family*, *lone-parent family* and *non-family* household), each with a specific number of agent slots (*parent*, *dependent child* or *other*). As an exact household location, we select a random graph node within the hierarchical area into which the household belongs.

Next, we assign the agents to these slots. The assignment is solved as a constraint optimization problem with a collection of hard and soft constraints imposed on solution candidates. Hard constraints ensure the validity of the assignment (e.g. a dependent child must only be assigned to a *dependent child* slot), while soft constraints minimize the fitness of unlikely assignments (e.g. 80-year old parents with 2-year old children). We solve this optimization problem using *simulated annealing* metaheuristic algorithm [17].

Once the agents are assigned to households, they automatically adopt these households as their *home location*. The census data set also contains the information on commuting travel times of the citizens. Which, along with multimodal path planner of AgentPolis, are used to point out areas where the agents are likely to work (areas reachable from the home location in a given time period). The *work/school* location of the agent is selected from POIs within those areas that allow work or school activity.

### C. Activity Scheduler

The task of the *activity scheduler* is to produce an *activity schedule* for each agent. Each activity schedule consists of a list of spatially and temporally determined activities planned by the agent. Our scheduler implementation is loosely based on ALBATROSS activity scheduler [1] generating two activity classes: *fixed* activities (sleep, work and school) and *flexible* ones (leisure and shop). *Fixed* activities form a so-called *skeleton*: a part of an activity schedule not likely to change on a daily basis while the *flexible* ones cover all other activity types which are fitted in gaps defined by the *skeleton*.

The scheduler produces *feasible* activity schedules which

means all trips between activities are possible<sup>11</sup>. It continually checks for the schedule *feasibility*. In a case of infeasible activity schedule, it reduces the number of *flexible* activities and/or completely reschedules. Activity schedules are generated by the following steps:

- 1) **Skeleton** is created. Skeleton is composed of *fixed* activities only. The *fixed* activity *start time* and *duration* is planned based on agent's demography and our probabilistic models trained using real-world travel diaries. The model takes demographic attributes as an input and outputs a PDF (probability density function) for a selected *target variable*. The PDF is represented by parameters of a selected distribution (we currently use normal distribution determined by mean and standard deviation for all *target variables*). The training dataset for the model is created by firstly clustering the travel-diary records based on the demography attributes and secondly deriving the distribution parameters for each cluster separately. We have experimented with both simple linear model and more expressive MultiLayer Perceptron (MLP) artificial neural network [10] favouring the latter. The modelled PDFs are in turn used to generate actual desired *start time* and *duration* for each fixed activity. Activity locations for the *fixed* activities are constant (assigned by the population synthesizer, see Section VI-B).
- 2) **Flexible activity templates** are created. Flexible activity templates are prototypes of flexible activities which should be inserted into gaps defined by the skeleton. Each template has the following information associated: *duration* (including tolerances), *activity type* and *priority*. The durations are modelled using the same approach as used for skeletons. There can be more templates of the same activity type allowing more instances of a single activity type in the final activity schedule. Their count is again generated using PDF approximation. As not all templates will make it to the final activity schedule, each template is assigned a priority which is used in the following decision making process of template selection.
- 3) Each template is **assigned a start time** to fit in the schedule. We check permutations of the *activity templates* as well as their assignments to *empty intervals* defined by the skeleton. A random configuration having a maximum sum of activity template priorities is selected. The templates are spread evenly in the corresponding *empty intervals*. Note, that during this process templates might be dropped in order to guarantee schedule feasibility.
- 4) Each template is **assigned an attractor** (location). A set of candidate attractors is selected for each template – a fixed number of possible candidate POIs closest to immediately preceding and following *fixed* activities is chosen. The actual configuration of attrac-

<sup>11</sup>The schedule might be still adaptively modified as a result of unforeseen events during simulation phase.

tors is then again selected by means of optimization: maximizing a number of remaining templates (as in the previous step templates might be dropped), minimizing a total expected trip duration sum and maximizing a total sum of POI attractiveness. Having assigned a *start time* and *attractor* the activity template becomes a fully-fledged activity.

## VII. MODEL INSTANCES

Our model has been instantiated in four different areas across Europe so far (see the overview in Fig. 3 and Tab. II):

- *Helsinki, Finland*: The model of coastal capital of Finland, Helsinki, is the smallest one in terms of area, covering only  $2,925.17 \text{ km}^2$ . The environment model was built using a 474MB OSM file and the population of 564,000 agents was synthesised based on statistics obtained from local statistical office.
- *Milan, Italy*: The inland city of Milan covers  $4,369.16 \text{ km}^2$  and was based on a 616MB OSM file. Total population of the model is 1.316 million agents.
- *Barcelona, Spain*: The coastal city of Barcelona takes up the area of  $11,182.10 \text{ km}^2$  populated by 1.621 million agents. The infrastructure model, based on a 735MB OSM, contains considerably more complex PT network, compared to Helsinki.
- *South-moravian region, Czech Republic*: The last and the largest model (in terms of area, covering  $35,816.07 \text{ km}^2$ ) is the only model that comprises of multiple smaller towns, instead of a single major city. The largest town in South-moravian region is Brno with the population of 379,000 but the whole model covers 1.136 million agents.

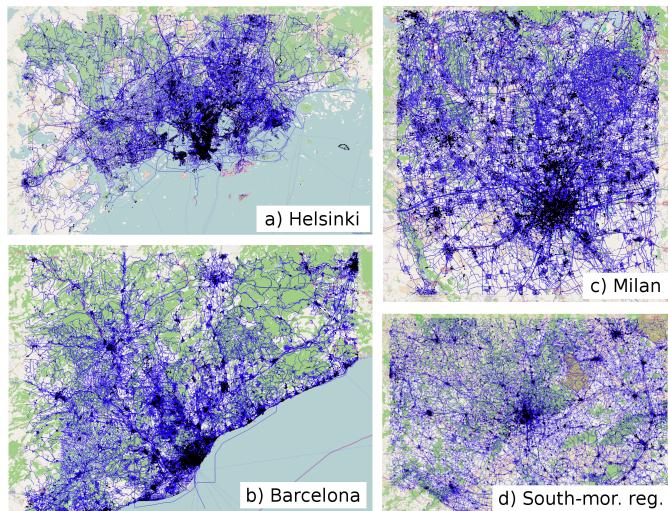


Figure 3. Model instances (Helsinki, Milan, Barcelona and South-moravian Region). Blue lines represent the transport infrastructure model and black dots are points of interest (higher concentration in city centers is apparent).

Table II. PUBLIC TRANSPORT MODES IN FOUR MODEL INSTANCES.

Model	Bus	Tram	Train	Trolley bus	Metro lines
Helsinki	yes	yes	yes	no	1
Milan	yes	yes	yes	yes	4
Barcelona	yes	yes	yes	no	11
South-m. r.	yes	yes	yes	yes	0

## VIII. VALIDATION

In order for any model to produce reliable and useful results, it needs to be *valid* enough. In fact, the validity is often considered the most important property of the models [12]. A process of quantifying the validity by determining whether the model is an accurate representation of the studied system is called *validation* and needs to be done thoroughly and throughout all the phases of model development [13].

Until recently there was no standardized method for statistical validation of activity-based models. In [7] we propose a six-step Validation Framework for Activity-based Models (VALFRAM). The framework compares temporal and spatial properties as well as structure of activity schedules (e.g. activity durations, trip distribution in space or typical activity sequences) against travel diaries and origin-destination matrices gathered in the reference system, using a number of numerical metrics to produce a collection of quantitative numeric validity indicators.

The validation of our model using VALFRAM is covered in detail in [7], where it is compared to its predecessor as well as with an alternative model based on recurrent neural networks. For the purpose of validation, the model generated a sample of 100,000 activity schedules for a single work day in South-moravian region in Czech Republic. The validation set consisted of approximately 1,800 schedules.

Figure 4 depicts and example of one of the validation steps. Distribution of travel times using the *car* mode in model and validation sets are compared using Kolmogorov-Smirnov statistic getting  $d_{KS} = 0.22^{12}$ . Further improvements of the model should decrease the value of the  $d_{KS}$ .

Figure 5 illustrates a validation of school activity spatial distribution showing PDFs of model and validation data. In most cases the model fits validation set reasonably, still there are several areas covered exclusively by either of them. Further analysis shown than these inaccuracies are caused by imperfections in the attractor (location) data employed by our model. To get a numerical statistic we use the same data to construct bivariate empirical cumulative distribution functions (ECDFs), discretize them and compute a Root Mean Square Error (RMSE) getting  $d_{ecdf} = 0.29$  in this case.

## IX. CONCLUSION

In this paper we have introduced a fully agent-based model of multimodal mobility built on top of open-source simulation

<sup>12</sup>Note that all statistics generated by VALFRAM are designed in a way where a higher number of the statistic value indicates less accuracy of the validated model.

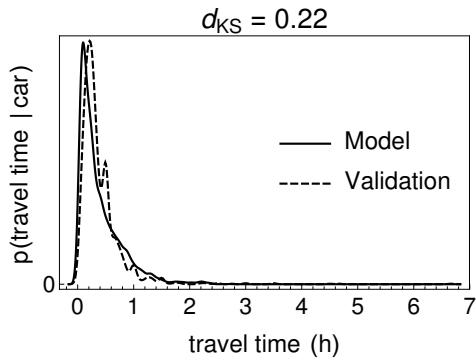


Figure 4. Car mode travel time distribution:  $p(\text{travel time} | \text{car})$ . Kolmogorov-Smirnov statistic  $d_{KS}$  comparing model and validation distribution is shown.

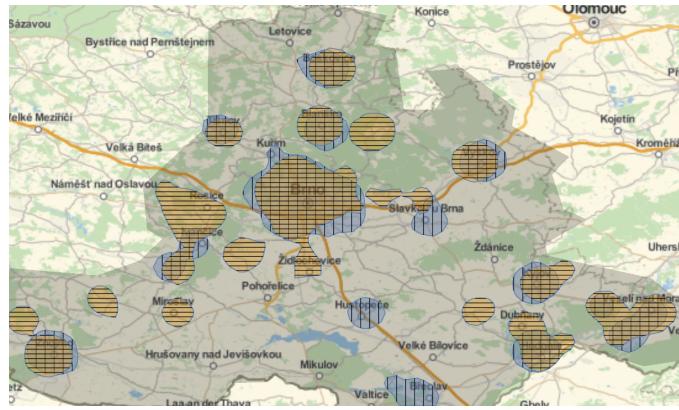


Figure 5. Modelled area (shaded) including school activity spatial PDFs is shown. Filled regions indicate PDF values of model (blue, vertical mesh) and validation set (yellow, horizontal mesh) surpassing a preselected threshold. The value of RMSE statistic is  $d_{ecdf} = 0.29$ .

platform AgentPolis, employing the activity-based modeling paradigm. We have given an overview of the model architecture and implementation details regarding the environment and agent models. The environment covers transport infrastructure, means of transport and points of interest extracted from multiple publicly available data sources. The synthesis of citizen agent population as well as generation of activity schedules from which an execution of actual traffic is derived are presented in detail.

The models of three European cities (Barcelona, Milan and Helsinki) were successfully used as part of the Mobility Policy Framework within the SUPERHUB project [2] to analyse how modifications of the transport system would affect its performance and behaviour of the citizens.

We validated the model of a South-moravian region in Czech Republic covering 1.136 million inhabitants by comparing the activity schedules performed by the synthetic agents to real-world origin-destination matrices and travel diaries (this area was selected due to best availability of validation data).

## ACKNOWLEDGEMENT

This publication was supported by the European social fund within the framework of realizing the project “Support of inter-sectoral mobility and quality enhancement of research teams at Czech Technical University in Prague”, CZ.1.07/2.3.00/30.0034, period of the project’s realization 1.12.2012 – 30.6.2015. This publication was further supported by the Technology Agency of the Czech Republic (grant no. TE01020155) and by the European Union Seventh Framework Programme FP7/2007-2013 (grant agreement no. 289067).

## REFERENCES

- [1] T. Arentze and H. Timmermans. *Albatross: a learning based transportation oriented simulation system*. Eirass Eindhoven, 2000.
- [2] I. Carreras, S. Gabrielli, D. Miorandi, A. Tamilin, F. Cartolano, M. Jakob, and S. Marzorati. Superhub: A user-centric perspective on sustainable urban mobility. In *Proceedings of the 6th ACM Workshop on Next Generation Mobile Computing for Dynamic Personalised Travel Planning*, Sense Transport '12, pages 9–10, New York, NY, USA, 2012. ACM.
- [3] M. Čertický, M. Jakob, R. Píbil, and Z. Moler. Agent-based simulation testbed for on-demand mobility services. *Procedia Computer Science*, 32:808–815, 2014.
- [4] D. Charypar and K. Nagel. Generating complete all-day activity plans with genetic algorithms. *Transportation*, 32(4):369–397, 2005.
- [5] B. Chen and H. H. Cheng. A review of the applications of agent technology in traffic and transportation systems. *IEEE Transactions on Intelligent Transportation Systems*, 11(2):485–497, 2010.
- [6] J. de Dios Ortuzar and L. G. Willumsen. Modelling transport. 1994.
- [7] J. Drchal, M. Čertický, and M. Jakob. Data Driven Validation Framework for Multi-agent Activity-based Models. *ArXiv e-prints*, Feb. 2015.
- [8] A. M. Geoffrion. The purpose of mathematical programming is insight, not numbers. *Interfaces*, 7(1):81–92, 1976.
- [9] R. W. Hall. Discrete models/continuous models. *Omega*, 14(3):213–220, 1986.
- [10] S. Haykin. *Neural Networks and Learning Machines (3rd Edition)*. Prentice Hall, 3 edition, 2008.
- [11] M. Jakob, Z. Moler, A. Komenda, Z. Yin, A. X. Jiang, M. P. Johnson, M. Pěchouček, and M. Tambe. Agentpolis: towards a platform for fully agent-based modeling of multi-modal transportation. In *Proceedings of AAMAS 2012*, volume 3, pages 1501–1502, 2012.
- [12] F. Klügl. *Agent-based simulation engineering*. PhD thesis, Habilitation Thesis, University of Würzburg, 2009.
- [13] A. M. Law. How to build valid and credible simulation models. In *Simulation Conference (WSC), Proceedings of the 2009 Winter*, pages 24–33. IEEE, 2009.
- [14] M. G. McNally. The four step model. *Center for Activity Systems Analysis*, 2008.
- [15] M. G. McNally and C. Rindt. The activity-based approach. *Center for Activity Systems Analysis*, 2008.
- [16] L. Smith, R. Beckman, D. Anson, K. Nagel, and M. E. Williams. Transims: Transportation analysis and simulation system. In *Fifth National Conference on Transportation Planning Methods Applications-Volume II: A Compendium of Papers Based on a Conference Held in Seattle, Washington in April 1995*, 1995.
- [17] P. J. Van Laarhoven and E. H. Aarts. *Simulated annealing*. Springer, 1987.
- [18] N. H. M. Wilson, J. Sussman, L. Goodman, and B. Hignett. Simulation of a computer aided routing system (cars). In *Proceedings of the third conference on applications of simulation*, pages 171–183. Winter Simulation Conference, 1969.