Simulating Adaptive, Personalized, Multi-modal Mobility in Smart Cities

Andreas Poxrucker(✉), Gernot Bahle, and Paul Lukowicz

Embedded Intelligence, German Research Center for Artificial Intelligence, 67663 Kaiserslautern, Germany
{andreas.poxrucker,gernot.bahle,paul.lukowicz}@dfki.de

Abstract. Smart, multi-modal transportation concepts are a key component towards smart sustainable cities. Such systems usually involve combinations of various modes of individual mobility (private cars, bicycles, walking), public transportation, and shared mobility (e.g. car sharing, car pooling). In this paper, we introduce a large-scale multi-agent simulation tool for simulating adaptive, personalized, multi-modal mobility. It is calibrated using various sources of real-world data and can be quickly adapted to new scenarios. The tool is highly modular and flexible and can be used to examine a variety of questions ranging from collective adaptation over collaborative learning to emergence and emergent behaviour. We present the design concept and architecture, showcase the adaptation to a real scenario (the city of Trento, Italy) and demonstrate an example of collaborative learning.

Keywords: Multi-agent simulations · Smart urban mobility · Socio-technical systems · Collective adaptive systems · Collaborative learning

1 Introduction

Intelligent, multi-modal transport concepts are widely seen as a key component of smart sustainable cities (see [1–3] for example). Such systems involve a flexible combination of various modes of public transport, individual mobility (from private cars, through private bicycles to walking) and shared mobility (e.g. car sharing, public bicycles etc.).

Within the EU sponsored ALLOW ENSEMBLES project [4] we are investigating adaptive, evolvable, personalized versions of such systems. The key idea is to provide a journey planning system that combines global planning with a decentralized personalization component, taking into account the current conditions in the city. Thus, to get from A to B the system may propose a set of possibilities ranging from a straight forward trip with a private car, through various public transport routes to complex combinations of using a private car, ride sharing, walking, bicycling and taking different public transport offerings. Next each user's personalization component evaluates each route, combining current conditions in the city (weather, traffic situation, how full are buses/trains) with

A. Leon-Garcia et al. (Eds.): Smart City 2015, LNICST 166, pp. 113–124, 2016.
DOI: 10.1007/978-3-319-33681-7_10

the users preferences (how important is sustainability vs. personal comfort, how much he/she likes to walk, how much he/she dislikes crowds or walking in hot weather etc.) and makes a recommendation.

We assume that the personalization components are adaptive in three ways. First they check how far the users follow their recommendations. Second, during the journey they use sensors to record how far the conditions (travel time, environmental conditions etc.) correspond to the predictions. Third, systems of different users communicate such journey assessments to each other and thus learn from each other. What information is transferred to what other systems is subject to a personal privacy policy and may range from social networks based strategies to proximity (people sharing the same bus or living in the same suburb).

Allowing users' personal systems to control the recommendations fitted to individual preferences and to learn through local interactions controlled by users' privacy policy has a number of advantages. On the other hand predicting the effect of different individual policies on the overall system behaviour and state (traffic jams, total CO2 output, etc.) is a difficult problem. As the individual systems learn, exchange information and, as a result, recommend various travel options to their owners, they change the travel conditions. Thus, if every system predicts the car to be the best option, streets will fill and traffic jams will arise. If the same bus is recommended to a lot of users, it will be overcrowded. Such obvious effects are made more complicated by the way information spreads, and individual strategies change according to personal policies. Overall we have to deal with a complex dynamic system with non linear dynamics and a variety of potential emergent effects.

In this paper we describe a simulation environment that we have developed to investigate such effects in adaptive, personalized multi-modal transport systems.

1.1 Related Work

Traffic and transportation simulations are a well researched area and many different models exists. Available traffic simulators are commonly classified based on the granularity of the traffic flow model they are based on. Microscopic simulators such as SUMO [5], VISSIM [6], or CORSIM [7] on the one hand model the movement of every single vehicle in great detail. Usually, they also model properties of the transportation network such as lanes of streets or traffic lights. Macroscopic simulations like MASTER [8] or FREFLO [9] on the other hand work with global models of traffic and transportation networks using e.g. differential equations.

The same is true for pedestrian/crowd simulations [10–12]. In general both traffic and pedestrian simulations focus on an in-depth analysis of a particular transportation mode, which is not the focus of this work. In fact, our system could well integrate various more detailed simulation models for traffic and pedestrians if actually needed.

On another end of the spectrum are various agent based modelling techniques for complex social phenomena like [13] for example. Closest to this paper is previous work by our group which investigates collaborative indoor location [14] and learning [15].

2 Paper Contribution

Performing experiments to investigate emergent phenomena of the type outlined in the introduction in large-scale real-world urban mobility systems is hardly ever possible given usual constraints of time and money. A well established methodology is to use what is known about the real world to set up simulations in which interaction effects leading to various emergent effects can be studied.

In this paper, we present our large-scale, multi-agent simulation toolbox to investigate emergent phenomena arising in the context of adaptive, personalized multi-modal urban mobility. The simulator thereby simulates a public transportation system of an urban area and people travelling within this area using different means of transportation. It also incorporates various personalization, decision making and distributed learning strategies. In summary, our simulator has the following main features:

1. **Simulation of a real-world urban mobility system.** In order to generate possible emergent phenomena such as traffic jams or air pollution, it is necessary to simulate a more or less complete urban transportation network. This includes modelling a street network which incorporates a model of *congestion* depending on the movement of the involved entities. Additionally, different means of transportation such as a public transit system are included in order to have travel alternatives. Our simulation uses a model of the urban mobility system of Trento, Italy, as data about the street network, public transportation etc. is available from our partners there from a recent collaboration. We will go into more details about the used real-world data in Sect. 5.
2. **Planning of entity journeys.** In order to let entities perform journeys using different means of transportation, a special planner component is required which allows the planning of multi-modal itineraries. The planner should thereby take the underlying transportation system into account providing not only simple shortest path routings through a street graph, but also multi-modal journeys like walking to a bus stop, getting on the right bus and off again at the correct stop, and walking to the final destination. In our simulation, we use *OpenTripPlanner* [19] which has already been applied in several cities in the world including Trento.
3. **Simulation of entity journeys.** Entities must be able to execute queried journeys in the simulated traffic system. As the transportation network involves means of public transportation, the simulator includes logic to let agents use buses or trains for example. In our current implementation, entities can drive with a car, go by bike, walk, or use means of public transportation as suggested by the planner component.

At this point, it is important to understand that the simulation tool we present in this paper does not claim to be a realistic traffic simulator. The latter are used in traffic engineering and traffic research and have become popular for analyzing and optimizing traffic on the level of a whole city or focusing on a part like a problematic crossing or a roundabout.

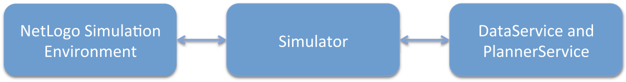


Fig. 1. Conceptual architecture of the simulation toolbox. The *NetLogo* multi-agent simulation environment is used as engine driving the simulation. The *DataService* and *PlannerService* are utility components providing information about the underlying transportation network and planning multi-model journeys, respectively.

3 Conceptual Architecture

As depicted in Fig. 1, our simulation tool conceptually consists of three components. The *NetLogo Simulation Environment* [16] is a popular multi-agent time-discrete simulation framework. It is used as an engine driving the step-based execution of the simulator and provides a graphical user interface (see Fig. 2) to interact with the simulation (e.g. starting and stopping the execution), to adjust input parameters, or to observe output parameters during execution.

The *Simulator* is the core component of our tool. It contains the definition of all the models to simulate as well as the specific logic which is executed by every instance of a certain model during every time step. We will explain the modelling in detail in Sect. 4. The simulator is realized as an extension to the NetLogo framework using its rich Java-based extension API. The coupling with

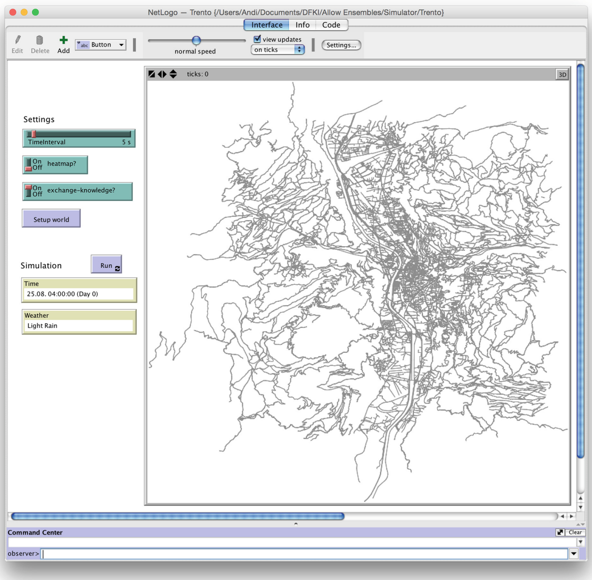


Fig. 2. NetLogo user interface with our simulation loaded.

NetLogo is, however, only loosely with every agent we define in the toolbox being wrapped by a corresponding NetLogo agent. Thus, it would be easy to use another simulation engine if necessary.

The *DataService* component offers an interface to query information about the transportation network of urban mobility system. It provides information about the underlying street network as well as existing public transportation agencies and their routes and schedules. The data provided by the component is thereby read in from a precompiled street graph datastructure and from one or more GTFS [17] datasets which are parsed during instantiation of the service in the beginning.

The *PlannerService* provides multi-modal travel suggestions to persons who want to travel from a certain starting position to a destination. In essence, the *PlannerService* is a client application querying the REST-API of an instance of *OpenTripPlanner* which is, as explained above, a open-source multi-modal journey planner. *OpenTripPlanner* operates on one or more precompiled graphs which are loaded and registered to the service during start-up. The graphs are compiled from *OpenStreetMap* [18] data which allows the planning of journeys based on the street map only (car, walk, and bike) as well as GTFS data for planning itineraries involving public transportation.

To make the journeys proposed by the planner compatible with the underlying transportation system both the *DataService* and the *PlannerService* rely on the same set of basis of data. We will go into more detail about these data in Sect. 5.

4 Modelling

Generally speaking, the simulator relies on a set of models which are instantiated during simulation start-up. The state of these instances is then modified during the actual execution which allows to observe local state changes as well as changes of global system behaviour. In essence, have two types of models: The *environment model* on the one hand models the underlying transportation network including the street network and the public transportation system (see Subsect. 4.1). Models of the *entities* (or *agents* - we will use these terms interchangeable through the rest of the paper) on the other hand represent the actually acting agents (see Subsect. 4.2).

4.1 Environment

In summary, our model of the environment is given by the underlying street map and the public transportation system. The street network on the one hand is modelled as a graph consisting of *StreetNodes* and *StreetSegments* which carry a number of attributes such as the length l and the maximum allowed driving speed v_{max}. In order to realize a model of congestion, every *StreetSegment* keeps track of the number of entities n which are currently travelling on it. Using l, v_{max}, n, and a certain minimum driving speed v_{min} a *possible* driving speed on the segment v_p is computed according to the inverted sigmoid function given by the formula

$$v_p = \frac{v_{max} - v_{min}}{1 + \exp(a * (n/l) - b)} + v_{min} \tag{1}$$

where a and b are two parameters determining how fast the possible speed decreases. v_p is updated once ever time step if the number of entities on a segment has changed. Additionally, we have integrated a model of the weather as a context factor to the environment which also influences the possible driving on a segment which is reduced by e.g. a half in case it is snowing.

The public transportation system on the other hand is given by the routes and schedules of public transportation agencies. All these information is available through the *DataService* component. The current state of the public transportation system during simulation is determined by the state of the entities managing and executing it, which are, in essence, the transportation agencies and buses (see Subsect. 4.2).

4.2 Entities

Transportation Agency. A transportation agency is an entity which manages a set of *routes* specified in the GTFS dataset which is available through the *DataService* component. Every route is defined by a set of stops. A *trip* of a certain route is, in turn, defined by a sequence of stops of the route together with arrival and departure times. Additionally, the shape of every trip, i.e. geographical route is given by a sequence of GPS points which needed to be mapped to the street graph in order to influence buses by congestion as described above. For this purpose, we implemented the heuristics-based map matching procedure described in [20] which maps the GPS point sequences to the respective paths through the street graph. As this procedure is computationally intensive, all trip paths are precomputed and loaded during start-up.

In order to execute the trips of its routes, every transportation agency manages a set of bus agents. During each time step of the simulation, an agency entity checks whether there is a new trip (or possibly several) to depart. In case there is, it assigns the trip to one of its idle buses which then starts to execute it.

Bus. A bus entity executes a trip assigned by the transportation agency it belongs to. During each time step, it moves along the street segments of its current trip resulting from the map matching process described above or waits at a stop to pick up waiting passengers.

Person. The core work-flow person agents execute during simulation is the following: When a person agent wants to go from its current position to a certain destination within the street network, it first sends a request to the multi-modal journey planner. In this request, the agent specifies, among others, its starting position, the destination it wants to reach, the time when the trip should depart, and a number of modes of transportation it wants to use (e.g. walk, car, bus). Based on these request parameters, the planner responds with a set of possibilities satisfying the constraints and preferences specified in the request.

The person then decides on one of the solutions by taking into account a utility ranking derived from estimates for travel time, costs and personal preferences [21] and executes it using the underlying transportation network. In case of the example above, the agent may walk to the given bus station, wait for the correct bus, then enter it and ride to its destination. During its journey, the person keeps track of travel parameters such as time, costs, bus fillgrades etc.

5 Towards a Real-World Simulation

To connect the simulation to the real world, we use real-world data for creating and calibrating our model of the environment and our different agent models. In our concrete case, we use data from the city of Trento. While the intention was not a complete and finely detailed representation of the urban environment (e.g. there are no traffic lights, no car density statistics for individual streets, etc.), we do aim for a realistic representation of behaviour. To that end, we conducted a survey among 15 people living and working in Trento. Participants marked different regions of interest on the map (residential areas, industrial zones, the university). They also specified main routes taken into the and out of the city and typical travel patterns (e.g. rush hours on workdays). Figure 3(a) shows the extracted regions of interest, Fig. 3(b) one possible distribution of agents created based on this partitioning. As a realistic source of weather patterns, we are using historical weather data queried from the wunderground API [22]. This provides an additional context factor influencing entity movement (e.g. when it snows, cars move very slow) which can be learned by the system. Furthermore, an agent is assigned a role which encapsulates different behaviour patterns, e.g. industrial workers, students, elderly (new roles can be added as needed). Based on these roles, a persistent agent population taking into account Trento demographic information is created and used for the subsequent simulation. All agents have a place of residence; some (depending on role) also have a place of work or go to university. Workers for example may travel to their workplace with the aim of being there at a given time, then start their travel home at another fixed time in the afternoon.

6 Example Use-case

Using our real-world setting of Trento described in Sect. 5, we present the results of an early experiment with our simulator investigating its ability to produce emergent phenomena. In this experiment, we rely on a set of assumptions. We define that every person agent of the simulation has a certain preference for going by car p_c and going by bus p_b, respectively, with $p_c + p_b = 1$. We further assume that every person decides on a means of transportation by directly comparing the preferences. If $p_c >= p_b$ the person chooses to go by car and in case $p_c < p_b$ the person decides to use the bus. Additionally, we define that travel time of buses is less affected by traffic than that of cars due to special bus lanes and traffic lights common in Trento.

(a)

(b)

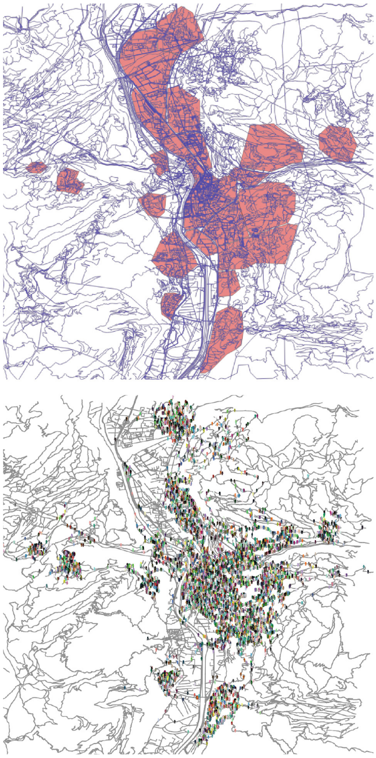
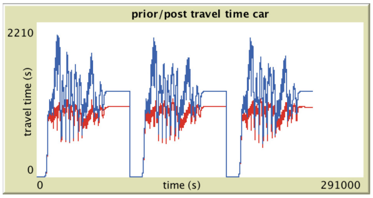


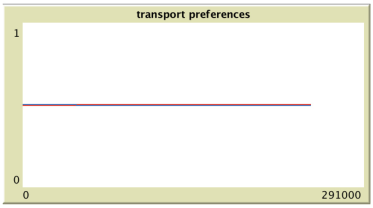
Fig. 3. (a) Partitioning of the city of Trento into different areas e.g. residential, industrial, shopping, or university as an overlay to the street network. (b) Distribution of agents created based on the defined regions of interest.

In this use-case, we want to demonstrate the difference between a traffic system in which entities adapt their preferences only based on their own experiences (scenario 1) and a system in which the agents learn from experience and actively exchange their knowledge when they get spatially close to others (scenario 2). In the first case, agents rely on their own experiences following one simple rule. If they go by car and are late due to heavy traffic (i.e. arrive later than initially predicted by the planner), they decrease their preference for car by a certain value and, in turn, increase their preference to use the bus. Additionally, we keep track of how many times an agent has changed its preferences due to personal experience. In the second scenario, agents also incorporate the knowledge of others whenever they meet using a weighted fusion of self and foreign preferences. Factors taken into account are the number of samples (how often an agent already had a good / bad trip and the (dis)similarity of experiences).

To measure system behaviour in both scenarios, we look at average prior (i.e. predicted by the planner) and posterior (actual simulated) travel time of trips by cars and buses respectively. We also track the average preferences for both car



(a) Prior (red) and posterior (blue) travel time by car.



(b) Preferences for bus (red) and car (blue).

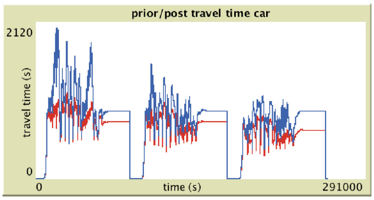
Fig. 4. Simulation results for scenario 1 (without knowledge exchange): The actual travel time differs significantly from the estimated one during the rush hour times. However, there is almost no change in the transportation preferences and consequently the posterior travel times look identical during all three days. (Color figure online)

and bus. Figure 4 shows the prior and posterior travel times by car (Fig. 4(a)) and mean transportation preferences averaged over all agents (Fig. 4(b)) for three simulated days. It can be seen that the difference between estimated and actual travel time is especially high during the morning and afternoon rush hours. The peaks in-between are caused by agents representing students who attend their lectures at university which causes high congestion in the university area. Figure 4(b) shows almost no change in the transportation preferences. Consequently, the travel behaviour of the agents does not change and the posterior travel times on subsequent days look almost identical to the first day.

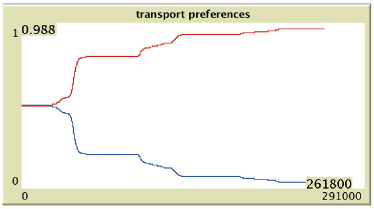
Figure 5 shows the statistics for the second scenario in which agents actually do share information and adapt their preferences based on feedback received from others. In this case, a clear change in preferences is visible. This, in turn, leads to different choices by agents, which generates a shift from cars to buses. This frees up street capacity, reducing difference between predicted and actual travel time.

While this is a rather simple example, it is a show case of the system capabilities. Based on the simulation framework, arbitrary experiments can be designed and deployed quickly. Areas of research that can be explored include:

1. different data exchange strategies, e.g. based on network models, privacy considerations, etc.
2. different knowledge fusion and learning techniques, e.g. graph based models.



(a) Prior (red) and posterior (blue) travel time by car.



(b) Preferences for bus (red) and car (blue).

Fig. 5. Simulation results for scenario 2 (with knowledge exchange): With agents exchanging knowledge about their prior travel experiences (high delays from traffic jams), the preference for using a car decreases rapidly. The more agents in turn decide to go by bus, the less vehicles are on the streets. Consequently, traffic jams are reduced and the estimated and actual travel time become more aligned. (Color figure online)

3. different ways to automatically rank solutions or even learn user preferences automatically.

All of the above can be represented by different and easily customisable parameters (like we did with travel time).

7 Conclusion and Future Work

In this paper, we have presented a large-scale multi-agent simulation tool for simulating adaptive, personalized, multi-modal mobility in the context of smart cities which can be used, among others, for investigating effects of collaborative agent behaviour on emergent global system properties. The simulator is calibrated using real-world data in an easily customisable way. An early demonstration has been done to show its capabilities. Our next step will be the application of the simulator for investigating collaborative learning techniques based on conditional random fields. We also strive to make the framework open source and public soon so it can be used as a research tool by others.

Acknowledgement. This work has been partially funded by the EU Project ALLOW Ensembles (600792) and Smart Society (600854).

References

1. Carreras, I., Gabrielli, S., Miorandi, D., Tamilin, A., Cartolano, F., Jakob, M., Marzorati, S.: 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, pp. 9–10. ACM (2012)
2. Motta, G., Sacco, D., Belloni, A., You, L.: A system for green personal integrated mobility: A research in progress. In: 2013 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI), pp. 1–6. IEEE (2013)
3. Motta, G., Sacco, D., Ma, T., You, L., Liu, K.: Personal mobility service system in urban areas: The IRMA project. In: 2015 IEEE Symposium on Service-Oriented System Engineering (SOSE), pp. 88–97. IEEE (2015)
4. Allow Ensembles. http://www.allow-ensembles.eu
5. Krajzewicz, D., Erdmann, J., Behrisch, M., Bieker, L.: Recent development and applications of SUMO - Simulation of urban MObility. In: International Journal on Advances in Systems and Measurements, pp. 128–138 (2012)
6. Fellendorf, M.: VISSIM: A microscopic simulation tool to evaluate actuated signal control including bus priority. In: 64th Institute of Transportation Engineers Annual Meeting, pp. 1–9 (1994)
7. Halati, A., Lieu, H., Walker, S.: CORSIM-corridor traffic simulation model. In: Traffic Congestion and Traffic Safety in the 21st Century: Challenges, Innovations, and Opportunities (1997)
8. Helbing, D., Hennecke, A., Shvetsov, V., Treiber, M.: MASTER: macroscopic traffic simulation based on a gas-kinetic, non-local traffic model. Transp. Res. Part B: Methodol. **35**(2), 183–211 (2001). Elsevier
9. Payne, H.J.: FREFLO: A macroscopic simulation model of freeway traffic. In: Transportation Research Record (1979)
10. Helbing, D., Buzna, L., Johansson, A., Werner, T.: Self-organized pedestrian crowd dynamics: experiments, simulations, and design solutions. Transp. Sci. **39**(1), 1–24 (2005)
11. Klügl, F., Rindsfüser, G.: Large-scale agent-based pedestrian simulation. In: Petta, P., Müller, J.P., Klusch, M., Georgeff, M. (eds.) MATES 2007. LNCS (LNAI), vol. 4687, pp. 145–156. Springer, Heidelberg (2007)
12. Kster, G., Hartmann, D., Klein, W.: Microscopic pedestrian simulations: From passenger exchange times to regional evacuation. In: Hu, B., Morasch, K., Pickl, S., Siegle, M. (eds.) Operations Research Proceedings 2010, pp. 571–576. Springer, Heidelberg (2011)
13. Epstein, J.M.: Agent-based computational models and generative social science. In: Generative Social Science: Studies in Agent-Based Computational Modeling, pp. 4–46 (1999)
14. Kampis, G., Kantelhardt, J.W., Kloch, K., Lukowicz, P.: Analytical and simulation models for collaborative localization. J. Comput. Sci. **6**, 1–10 (2015). Elsevier
15. Kampis, G., Lukowicz, P.: Collaborative knowledge fusion by Ad-Hoc information distribution in crowds. Procedia Comput. Sci. **51**, 542–551 (2015)
16. Wilensky, U.: NetLogo. http://ccl.northwestern.edu/netlogo/
17. What is GTFS? https://developers.google.com/transit/gtfs/
18. OpenStreetMap. http://www.openstreetmap.org
19. OpenTripPlanner. http://www.opentripplanner.org
20. Marchal, F., Hackney, J., Axhausen, K.: Efficient map matching of large global positioning system data sets: Tests on speed-monitoring experiment in Zrich. Transp. Res. Rec. J. Transp. Res. Board **1935**, 93–100 (2005)

21. Andrikopoulos, V., Bitsaki, M., Bucchiarone, A., Karastoyanova, D., Leymann, F., Nikolaou, C., Pistore, M.: A game theoretic approach for managing multi-modal urban mobility systems. In: 2th International Conference on the Human Side of Service Engineering Human Factors and Ergonomics (2014)
22. Wunderground weather data. http://www.wunderground.com