Multi-agent Systems Project Report: Traffic Simulation based on Real-Time Information

Konstantin Kloster[1] and Oliver Berg[1]

[1]Technical University Kaiserslautern

October 30, 2018

Abstract

TODO write abstract

Contents

1 Introduction

Traditionally, real-time information (RTI) was used by transit providers for operation and control. In today's age of always-on telecommunication devices and sensor input of an armada of connected input devices through the "Internet of Things" (IoT) movement, this information is also increasingly utilized by travellers themselves for route planning. This includes car routes as well as coordinating public transport timetables or available ride-sharing resources.

In this work, we will simulate, display and analyze the dynamic movement and interaction of travelling & planning agents within a traffic coordination setting.

To this end, a multi-agent system (MAS) is implemented which takes formal network graphs and specific user input regarding the simulation scale to then simulate a continuously spawning group of car-agents traversing the network with the goal to arrive at an assigned location as fast as possible. An internal coordination unit in the form of a planner-agent distributes route information according to the type of agent it is being queried from. The architecture setup is based on propositions from provided sources ([Mastio et al., 2015], [Brakewood and Watkins, 2018]). Random events altering the state of the network take place to simulate traffic incidents and to provoke agents to deviate from initial plans.

Having run the simulation and persisted the simulation run in respective log-files, a web-based frontend implementation reads network- and log-files to visualize the network graph and temporal agent-behavior. The visualization allows for selection of graph and run-data and displays the read and interpreted data. This should allow the observer to understand the car-agent's behavior and make numeric analysis results more visually interpretable.

This report is structured as follows: Following this introductory description of context and task, section 2 will inspect related work surrounding the field of RTI traffic simulation and derived research and system implementation propositions. These are then being put into application context in section 3 where the implemented MAS and respective architectural constraints are outlined. Next, section 4 continues to describe the frontend visualization and transitions into the traffic performance analysis in form of agent arrival-time metrics being is read from persisted system logs and put into perspective within this report's section 5. Finally, section 6 presents the findings and concludes this project's report.

2 Related Work

Simulating development of traffic is a well-traversed research topic regarding scheduling and network traversal simulation problems. Throughout application, dedicated simulators have been applied frequently and early-on like the popular "Multi-agent Transport Simulator" (MATSim) [Sezen, 2003] or "Repast Simphony" [Zargayouna et al., 2013] alongside dedicated extensions like the Symphony-based "SM4T Simnulator" [Ksontini et al., 2016]. Such applications provide advantages like automated timetables for public transport, advanced types of travelling agent and unified logging formats for simulation runs. Resorting to holistic solutions like the above mentioned is especially useful for non-technological research regarding behavioral analysis or city planning [Brakewood and Watkins, 2018]. If the multi-agent system aspects are predominant though (as is the case in this work), adapting similar implementation structures whilst doing the actual agent implementation work (done here) proves beneficial.

In contrast to fully incorporated applications, some approaches in literature already shift focus towards surrounding aspects surrounding of traffic simulation, like focusing on the distributing simulation [Mastio et al., 2015].

As of [Mastio et al., 2015], which depicts the simplest yet most fundamental approach to general network traversal, it utilizes a basic graph structure with vertices as intersection points and edges connecting these intersections. Agents are assigned a path over a fixed set of vertices as the shortest path over weighted edges. The travel time t there depends on the number of agents currently on the edge in question. Calculating this t can be done based on different kinds of functions modelling e.g. a certain free-flow capacity where for a given n number of vehicles the weight (t) of an edge will not be impacted and only after reaching a certain threshold number n_{thresh} the weight will increase to depict a slower movement (to a degree of α) of traffic along said edge; like shown in Equation 1 - being a slightly modified version of the formulas used in [Mastio et al., 2015] and [Ksontini et al., 2016].

$$t = \begin{cases} s, & \text{if } n < n_{\text{thresh}} \\ s * \alpha(n - n_{\text{thresh}}), & \text{otherwise} \end{cases} \tag{1}$$

To this agent-centric travel choice procedure, a literature review as is being depicted in the transport review of [Brakewood and Watkins, 2018] adds a theoretical framework of traveler agent's perspective concerning travel- and mode choice as well as choices regarding route, boarding and departure. Incorporating this thinking, one ends up with a tight path-choosing procedure across multiple channels which theoretically boils down to the simple procedure of Equation 1 adapted to modes, routes and fixed schedules. As this framework is a theoretical methodology at heart, the practical implementation aspect of this formal procedure raises multiple performance and architectural issues which without utilizing afore-mentioned well-established holistic framework solutions is expected to introduce bottlenecks.

As proposed by [Zargayouna et al., 2013], agents adhere to a specific simulation workflow where they continuously query for updates regarding path choice whenever reaching nodes of the underlying network. Opposing the precise setup depicted here and utilizing in addition the thinking of [Mastio et al., 2015], no precise time-step-based approach of procedurally checking all agents for their position but rather an event-based truly (distributed) multi-agent approach is implemented where agents query after complete time lapses instead of single short global time intervals.

The visualization procedure of a given simulation then needs to adhere to the time lapse nature of the event logs, which not necessarily adheres well to standards defined by holistic framework solutions (like used in [Zargayouna et al., 2013][Ksontini et al., 2016]). As visualization is a side-aspect of most MAS-focused implementations (see [Mastio et al., 2015]), this is generally speaking seen as an added bonus for both debugging and reporting purposes.

Regarding performance metrics of a traffic simulation, metrics like use-of-transit, satisfaction and travelling-time are frequently proposed ([Brakewood and Watkins, 2018]). With the use of dedicated car- and planner-agents (alongside [Zargayouna et al., 2013]) and the omittance of public transport time-table-based scheduling / availability-based carsharing, this leaves travelling-time as well as deviation from planning to execution performance as major indicators for performance. Enhancing this may be subjective monitoring of network behavior ([Brakewood and Watkins, 2018], [Zargayouna et al., 2013]).

3 MAS Traffic Simulator Backend

TODO write backend-section

4 Web-based Simulation Frontend

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

5 Agent Performance Analysis

TODO write performanceAnalysis-section

6 Conclusion

TODO write conclusion-section

References

[Brakewood and Watkins, 2018] Brakewood, C. and Watkins, K. (2018). A literature review of the passenger benefits of real-time transit information. *Transport Reviews*, pages 1–30.

[Ksontini et al., 2016] Ksontini, F., Zargayouna, M., Scemama, G., and Leroy, B. (2016). Building a realistic data environment for multiagent mobility simulation. In *Agent and Multi-Agent Systems: Technology and Applications*, pages 57–67. Springer.

[Mastio et al., 2015] Mastio, M., Zargayouna, M., and Rana, O. (2015). Towards a distributed multiagent travel simulation. In *Agent and Multi-Agent Systems: Technologies and Applications*, pages 15–25. Springer.

[Sezen, 2003] Sezen, B. (2003). Modeling automated guided vehicle systems in material handling.

[Zargayouna et al., 2013] Zargayouna, M., Zeddini, B., Scemama, G., and Othman, A. (2013). Agent-based simulator for travelers multimodal mobility. In *KES-AMSTA*, pages 81–90. Citeseer.