International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-2, July 2019

Usage of SUMO Computer Modeling Software for Road Traffic Control System Validation

Alexey Ivanov, Fares Abu-Abed

Abstract*: *Modern road traffic control systems become more complex with each year due to the growing number of cars, demand for more optimized road network designs, and desire to decrease road traffic delays at the intersections during rush hours because of the constantly growing metropolitan area and hence the need of moving farther away across the city. Therefore, a need is very relevant nowadays for technologies, which can help build more optimal and intelligent road traffic control systems, which in turn will help optimize already existing road network designs. With the help of computer modeling software, it is possible to recreate already existing segments of road network, and by implementing control logic, it is possible to observe how control system and road traffic would react to various changes. This paper looks at how computer modeling software (namely SUMO) can help with the validation of road traffic control system changes without the risks of road traffic accidents caused by manipulation of real control systems.

Index Terms*: *computer modeling, control system, open source, road traffic, software.

I. INTRODUCTION

Traffic congestion is still highly prevalent nowadays, especially in metropolitan areas of big cities [1]. Inefficiencies in modern traffic control systems cause not only significant traffic flow delays in day-to-day commute of many people, but also introduce air pollution and general monetary expenses due to the constant need of maintenance and improvement of already existing infrastructure. For example, average cost of being in congestion in America is estimated $1,700 a year [2], and that is not taking into account the price of toxic car emissions. Needless to say, any way to reduce traffic congestion is welcome.

Big cities try to conquer this problem by introducing automated road traffic control systems [6], which can control many various aspects of road traffic such as, for instance, traffic light phase durations for the intersections depending on the time of day. In case of emergency, for example, a malfunctioning traffic light, such system enables to react quickly by monitoring the state of listed intersections and send repair brigade to quickly eliminate the problem.

A problem with such systems arises at the time when further extension of automated infrastructure is required. Sometimes it is not obvious how to set up newly added intersection to the system. At this point operator, who does the setup has a choice: trust their gut feeling and hope for the best or use previous experience in setting up similar configurations. First case introduces the danger of increased traffic accident risk, due to the unknown driver reaction, which could show up later when it will be too late.

Computer modeling has been successfully used in numerous fields for quite a long time now. Almost any industry, which imposes a high risk of accidents, uses computer modeling for process validation, because it is cheaper to create a model, which can be observed in various states, rather than having a costly and potentially dangerous accident.

II. ROAD TRAFFIC CONTROL SYSTEM STRUCTURE AND VALIDATION

Road traffic control system (RTCS) is a system which allows to control various aspects of road traffic, such as: intersections, speed limit etc. Besides control RTCS allows to monitor their state – whether it is broken or malfunctioning. Such system enables centralized control using special place called situation center – a place from which an operator (or several operators) can monitor and control road traffic end devices (traffic lights, signs etc.) quickly and effectively in response to current traffic situation in that area.

Such system introduces higher road traffic safety. Because of quick response, it is possible to take control in situations of outage, which in turn reduces driver confusion and hence increases safety in the event of end device outage. Along with increased safety, such system allows to save on maintenance costs due to remote control capability.

Some of the weaknesses of road traffic systems might be considered the difficulty of initial system setup and operator training. Because it requires complete integration with end devices, initial setup cost might be high. Not all of such systems are intelligent: they require manual operating scenario creation, which introduces extra operator learning costs. Research for better road traffic control systems is an ongoing process in the modern world as can be seen in [7]-[9].

A. *Structure of automated road traffic control systems*

Typical structure of an automated RTCS consists of a combination of software and hardware components [6].

Revised Manuscript Received on 30 July 2019.

***** Correspondence Author

Alexey Ivanov*, Department of Electronic Calculating Machines / Tver State Technical University, Tver, Russia.

Fares Abu-Abed, Faculty of International Academic Cooperation / Tver State Technical University, Tver, Russian Federation.

Retrieval Number: B3396078219/19©BEIESP
DOI: 10.35940/ijrte.B3396.078219
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Control logic and operator user interface is usually implemented in software because of higher flexibility.

This software resides on a server – a special computer, which acts as a gateway between operators and RTCS end devices such as traffic lights and signs. Fig. 1 illustrates aforementioned structure of an automated RTCS. This structure assumes there exists some communication medium, which allows control system server to communicate with the end devices.

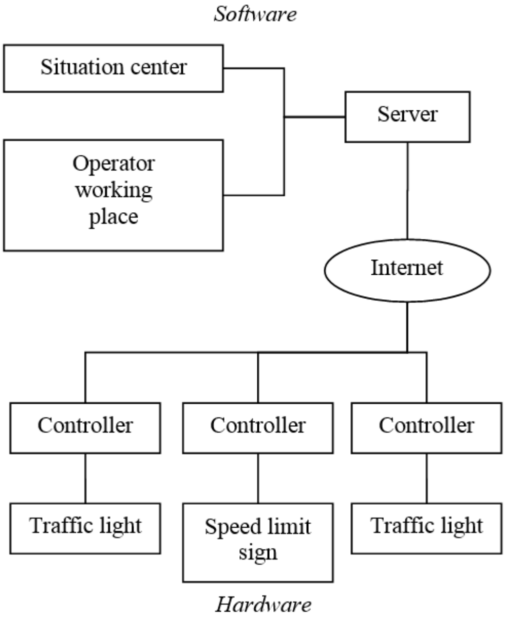


Fig. 1. Typical structure of automated RTCS.

Each end device is typically connected to some kind of controller. Controller manages connection between the end device and the server and can guard end device against invalid control signals coming from the server, i.e. conflicting traffic light phases, which would cause intersection deadlock.

B. *Intelligent road traffic control systems*

There has been extensive work done in the field of intelligent systems since the rise of machine learning and artificial intelligence algorithm practical usage.

There have been numerous researches, such as [4] and [5], which introduce artificial intelligence to allow better control over road traffic situation. By employing reinforcement learning algorithms, researchers build computer programs, which allow to control traffic flow using real-time data as well as previous experience. Fig. 2 shows a typical structure of a fully connected neural network, which could serve as a base for intelligent system algorithm implementation in application to road traffic control. Inputs of this network would receive current state of the model (current traffic load, delays of traffic light phases etc.) while outputs usually provide data to make a control decision.

Often researches have to build models themselves to simulate real life traffic flow and control devices (most commonly – traffic lights). In order to verify their control program they introduce metrics, such as traffic average speed, which is maximized or the minimum of traffic wait time at the red signal of a traffic light.

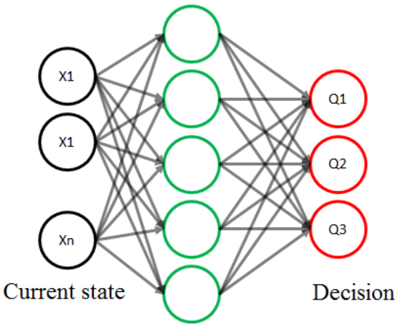


Fig. 2. Typical structure of a neural network.

C. *Computer modelling of a road traffic control system*

In order to develop a computer model of an RTCS, developer usually needs two parts: an environment model and a control system. If developer wishes to model an intelligent system, they could extend control system with algorithms that would allow intelligent control (based on machine learning algorithms or some other algorithm).

Environment model is a model that represents environment, such as cars, pedestrians, signs, traffic lights, road lanes etc. Setting up correct environment for modeling is crucial for the validity of decisions, which are taken after analysis of the model. For correct modeling developer has to take into account both microscopic [10] details, such as non-uniform driver behavior (some drivers are less prone to errors or aggressive driving than others), vehicle physical limits and capabilities (acceleration, breaking etc.) and macroscopic [11] details, such as general structure of the road network (number of lanes, their direction etc.)

Usually there is a need to create a model of a real object. That requires replication of a real traffic situation during required time of day in order for the model to be as close to the real situation as possible. Developer could leverage statistical data about the place or gather the data themselves, but that often deems itself not feasible or even impossible. In this case, developer could get the data with the help of online map services that provide statistical traffic delay data (Fig. 3).

Control system is a part of the model, which controls RTCS end devices, such as traffic lights or signs in order to improve the efficiency of moving traffic flow. This is usually done by a program that executes some sort of algorithm (i.e. monitors vehicle count and switches traffic lights accordingly).

Fig. 3. Average traffic flow speed data availabe from online maps (in this case "Yandex.Maps").

D. *Road traffic control system model validation*

In this section, we provide

our definition of validation and what benefits it can bring in practice.

To quote Ian Sommerville [15]: “Software validation is the process of checking that the system conforms to its specification and that it meets the real needs of the users of the system.” That quote originally is for software development approach, but it is true in our case as well. By the term “validation” we mean not only correct function of a developed model, but also expected behavior in reaction to all anticipated situations. In practice that allows to conduct experiments on a finely tuned model, so later it is possible to transfer results that are shown in the model into real road traffic control situations. This, in turn, allows to cut costs on tuning already deployed RTCS and avoid dangerous situations on the road caused by misconfiguration.

By employing previously mentioned RTCS modeling approaches, it is possible to create a model of already existing element of a road network. For example, if there is a need to optimize a busy intersection it is possible, by using already existing theoretic and practical knowledge along with computer modeling techniques, to create a model of this intersection in order to inspect it. It will be trivial to see how it would react if some of the traffic light phases would get changed – all without real intervention into the work of a real intersection. By analyzing various changes for this intersection this allows to come to the conclusion that these changes would work in practice with a higher certainty opposed to doing it relying on gut feeling, hence this intersection will be validated to work correctly under new configuration.

III. SIMULATION OF URBAN MOBILITY COMPUTER MODELING SOFTWARE

In order to build a model for RTCS object validation, one often has to build a computer model. In the second chapter we list common pieces needed in order to make a functioning road traffic network model with realistic driver behavior. To ease the burden of traffic researchers (and consequently intelligent road traffic system developers) the Centre of Applied Informatics and the Institute for Transport Research at the German Aerospace Centre develops and maintains a microscopic computer modeling software called “SUMO” – this name being an abbreviation of “Simulation of Urban MObility” [3]. It is a de-facto standard in road traffic research world as it has been used in a number of works, such as [13], [14] and numerous others.

A. *Modeling software features*

SUMO is an impressive open-source software package that consists of a number of programs, which allow to create, set up and run traffic simulations. Capabilities of the program allow for simulations of large scale, while microscopic simulation approach means that each car is modeled individually, without any abstractions such as fluid flows that can be seen in macroscopic simulation approaches.

SUMO is described in detail by their authors in [12], however we will mention in brief about its structure, workflow while using it and generic features it provides compared to a model one would make themselves.

Being a framework for building and running road traffic simulations, it consists of several program modules. There is a program to run simulation itself, in two variations – most commonly used graphical interface simulation window (Fig. 6), which allows to observe the simulation while it runs and a command line interface (CLI) program without graphical interface, which greatly simplifies the program and allows to run big simulations faster. Besides that there is an editor program named NETEDIT (network editor) that provides a graphical user interface for road network editing. It is worth noting that in general all SUMO simulation files are human readable XML files which when required can be easily edited by hand.

Along with the main modules above there are numerous utility programs, which help with automatic simulation data generation, such as vehicle routing using several different algorithms and even give the ability to generate random road networks. Another impressive feature of SUMO is TraCI (Traffic Control Interface) – a protocol that allows to run software in client-server mode, where simulation program acts as a server and multiple client programs can connect to it either locally or over the network. The availability of an extensive application programming interface for TraCI allows researchers to build external programs that communicate with the simulation. This makes it much easier to integrate an intelligent control system into a model.

B. *Creating and setting up a model*

First step in modeling with SUMO is creating a model. Typically, it is some existing object, which is to be analyzed for how it can be optimized with minimal setup alterations. Fig. 4 shows a satellite image of an intersection to be modeled. Clearly visible road lanes and markings are enough to recreate this structure in a computer model.

Second step is to create the model of a given object. It is achieved by building a graph, which would represent road network. Each edge of the graph would serve as a road. SUMO gives the ability to configure road lane count for each side of the road. Edges are connected through the nodes – most of the time these points act as intersections of roads. Nodes can have a specific type, which would change its behavior and appearance in simulation interface. Typical node types are: traffic light, unregulated intersection etc.

Fig. 4. Sattelite image of a road network object.

All of this can be done in a graphical road network editor called NETEDIT. After creating a basic graph, user must set up the network with proper lane and intersection configuration.

SUMO provides a flexible system, which allows to implement any configuration of connections from one lane to another. Fig. 5 demonstrates intersection path editing process in road network editor.

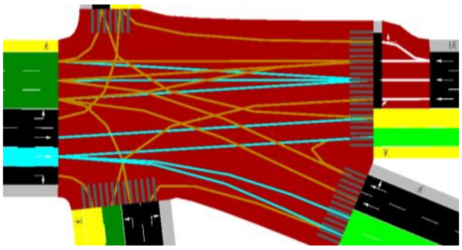


Fig. 5. Intersection path editing using NETEDIT.

After the environmental part of the model has been completed, the last required step before running the model is the creation of vehicle routes. SUMO has ability to define types of vehicles and model individual vehicle routes as well as traffic flows, which is convenient especially for large scale models.

C. *Interpretation of modeling results and object validation*

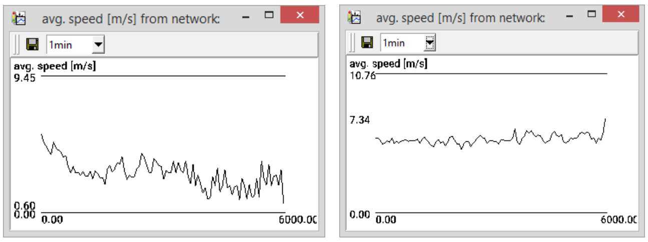
Besides building an adequate model (a model, which represents a real object with high enough accuracy), another crucial step is the correct interpretation of the results of modeling, after which the researcher could use these results to conduct object validation (i.e. confirming that new object configuration can be used in a real object).

Interpretation can be done using several tools provided by SUMO: either building your own program and using TraCI to gather all required model data or to use built-in tools into simulation program with graphical user interface, which allow to examine road network's both static and dynamic parameters, such as number of vehicles, their speed etc.

For example, intersection portrayed on Fig. 6 has a complex traffic light phase system that can be analyzed to see if it can be improved. The intersection has primary and secondary roads. Primary road is obviously busier, so it makes sense to tweak parameters, which affect its throughput. One of the core parameters is green phase duration – the longer it is, the less traffic is interrupted on the main road. However it cannot be too long, because then there would be many cars even on the secondary roads and that would cause congestion.

Fig. 6. SUMO graphical user interface demonstration with a running simulation of an intersection. Color of the cars represent their speed: red – complete stop, green – 54 km/h.

One way to conduct object validation procedure of the intersection is to change the duration of the green phase of the primary road to see how it would affect traffic throughput. As a metric for traffic throughput, we would choose average traffic speed – the higher it is, the better the throughput of the intersection. Of course, this assumption would work only if the model is adequate.



(a) green phase is 32 seconds (b) green phase is 16 seconds

Fig. 7. Average traffic speed before (a) and after (b) object configuration.

Fig. 7 shows, that after reducing green phase duration on the primary road from 32 seconds to 16 seconds average traffic speed became more stable and more importantly – higher. That means average speed of all vehicles on this intersection – both primary and secondary – has been increased. Assuming the model has been constructed using real parameters – dimensions, such as road length, traffic density, vehicle speed and their general physical capabilities, then it is safe to assume that reducing green phase duration would cause the same effect on the real object. Therefore, object validation has been conducted – we found the new and safe parameter value. Besides changing parameter of the object, researchers might want to change parameters of the environment – test how the object would handle increased traffic load is a common aim of object validation using computer modeling approach.

Numerous features of SUMO software package allow researchers to easily create sophisticated road traffic flow models in order to test their road traffic control systems, possibly including intelligent machine learning algorithms. This removes the need of creating their own, most often less complex models, which in turn helps to achieve high adequacy of the model they use to conduct experiments.

IV. CONCLUSION

In this paper, we highlighted the importance of using automated control systems for road traffic control and how computer modeling can be used to validate such systems as well as real objects such as intersections.

In the second chapter of this paper, we discuss road traffic control system structure, highlight modern tendency of intelligent components in such systems. Then we point out the advantage of the usage of computer modeling traffic that researchers can use in order to verify their control algorithms.

In the third chapter, we described how by using SUMO computer modeling software it is possible to validate road traffic objects, such as intersections, to increase the certainty of a road traffic control system operator that changing observed value (for instance, duration of a traffic light phase) will not cause dangerous situations on the road.

Of course this approach can be applied to all sorts of road traffic regulation situations – given the created model is adequate.

In our future work, we would like to focus on a protocol, which could be used to communicate between a road traffic control system and a model. This would help further streamline the process of setting up the necessary interface between the control program and the model in order to give the researchers an ability to focus more on control system implementation instead of fiddling with low-level technical details of software communication.

REFERENCES

1. G. Weisbrod, D. Vary, G. Treyz, “Economic implications of congestion.” Transportation Research Board: NCHRP Report, issue 463, 2001.
2. The Economist. (2014, November 3). The cost of traffic jams [Online]. Available: https://www.economist.com/the-economist-explains/2014/11/03/the-cost-of-traffic-jams
3. D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, “Recent Development and Applications of SUMO – Simulation of Urban MObility,” *International Journal on Advances in Systems and Measurements*, issn 1942-261x, vol. 5, no. 3 & 4, year 2012, pp. 128-138.
4. E. van der Pol, F.A. Oliehoek, “Coordinated Deep Reinforcement Learners for Traffic Light Control,” *NIPS'16 Workshop on Learning, Inference and Control of Multi-Agent Systems*. Available: https://sites.google.com/site/malicnips2016/papers
5. H. Wei, G Zheng, H. Yao, Z. Li, “IntelliLight: A Reinforcement Learning Approach for IntelligentTraffic Light Control,” *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, doi 10.1145/3219819.3220096, year 2018, pp. 2496-2505.
6. P. Roychowdhury, S. Das, “Automatic Road Traffic Management System in a City,” *Trends in Transport Engineering and Applications*, vol. 1, issue 2, year 2014, pp. 38-46.
7. I. Ben Arbi, “A survey on intelligent urban road traffic control systems,” *ResearchGate*, year 2019.
8. S. Jianjun, W. Xu, G. Jizhen, C. Yangzhou, “The analysis of traffic control cyber-physical systems,” *13th COTA International Conference of Transportation Professionals*, year 2013, pp. 2487-2496.
9. A.M. de Souza, C.ARL Brennand, R.S. Yokoyama, “Traffic management systems: A classification, review, challenges, and future perspectives,” *International Journal of Distributed Sensor Networks*, doi 10.1177/1550147716683612, vol. 13, issue 4, year 2017.
10. E. Brockfeld, P. Wagner, “Testing and Benchmarking of Microscopic Traffic Flow Models,” *Proceedings of the Computional Physics Conference*, year 2002, pp. 775-776.
11. R. Mohan, G. Ramadurai, “State-of-the art of macroscopic traffic flow modelling,” *International Journal of Advances in Engineering Sciences and Applied Mathematics*, vol. 5, year 2013, pp. 158-176.
12. D. Krajzewicz, G. Hertkorn, P. Wagner, “SUMO (Simulation of Urban MObility); An open-source traffic simulation,” *4th Middle East Symposium on Simulation and Modelling (MESM2002)*, pp. 183-187.
13. M. Stevens, C. Yeh, “Reinforcement learning for traffic optimization,” unpublished.
14. P. Manniona, J. Duggana, E. Howley, “Parallel Reinforcement Learning for Traffic Signal Control,” *The 4th InternationalWorkshop on Agent-based Mobility, Traffic and Transportation Models,Methodologies andApplications (ABMTRANS)*, year 2015, pp. 956-961.
15. I. Sommerville. (2011). *Software engineering* (9th edition)

AUTHORS PROFILE

Alexey Ivanov B.Sc., Department of Electronic Calculating Machines / Tver State Technical University, Tver, Russia. lexuzieel@gmail.com

Fares Abu-Abed Ph.D., Associate Professor, Dean of the Faculty of International Academic Cooperation. Associate Professor at the Department of Electronic Calculating Machines at Tver State Technical University. Has more than 200 works and patents in the field of IT-technologies and artificial intelligence. aafares@mail.ru