IMPLEMENTATION OF THE SOFTWARE-IN-THE-LOOP SIMULATION FOR ASSESSMENT OF OPERATIONAL CAPABILITIES IN THE NORTH AMERICAN ADVANCED TRANSPORTATION CONTROLLERS

Miloš N. Mladenović, MSc, Graduate Research Assistant
milosm@vt.edu
Montasir M. Abbas, PhD, Associate Professor
abbas@vt.edu
Miloš D. Vodogaz, BSc, MSc
m.vodogaz@sf.bg.ac.rs

Proffesional Paper

Abstract: *For improving the operational capabilities of the existing signal control system under their purview, traffic operation engineers need enhanced understanding and utilization of available traffic signal controller capabilities. This paper proposes methodology for obtaining in-depth information on controller capabilities based on Software-in-the-loop simulation (SILS). SILS is recognized as capable to assess unknown controller software capabilities. Testing and verification of the 2070 Advanced Transportation Controller software features is presented along with recommendations for further research.*

Key words: *traffic signal controller software, software-in-the-loop simulation testing, VISSIM signal time table*

I. INTRODUCTION

Increasing constraints for the construction of new road infrastructure are imposing higher performance expectations upon deployed traffic management and control systems. Traffic signals are considered as one of the vital control elements of those traffic management and control systems. They directly affect mobility, safety, and environmental parameters of the transportation networks where they are deployed. Transportation agencies nowadays are usually responsible for the increasing numbers of signalized intersections, thus often combining them into a traffic-signal control system. These traffic-signal control systems play an essential role in coordinating individual traffic signals to achieve desired operational objectives of an optimal network-wide traffic control.

Evident changes in traffic demands and patterns on transportation networks are often leading to poor safety, operational, and environmental parameters of traffic signal control system. The situation becomes even more complex when Department of Transportation (DOT) is responsible for managing large traffic-signal control systems containing obsolete technology. This underlines the need and pressure for extracting additional benefits from traffic-signal control systems.

In such a situation, traffic engineers often have to take into consideration many factors for improving system performance. Those factors are originating from specific localized requirements and design elements (e.g., human factors, geometric constraints, etc.). In addition to all these factors, there is one critical system component of traffic-signal control system – the actual control equipment.

Traffic engineers often face difficulties with control equipment. Those difficulties can be on an everyday, operational, basis. In addition to this, issues can be related to planning for the future signal control systems. In order to extract additional benefits from existing or successfully plan future traffic-signal control systems, traffic engineers need a thorough understanding of capabilities that equipment under their purview has. The focus on the equipment factor becomes even more important since traffic engineers are facing constantly evolving state-of-the-art technology. With this in mind, the list of influencing factors just keeps expanding.

In order to address the need for thorough understanding and assessment of equipment capabilities, the research presented in this paper presents the methodologies and techniques for assessment of traffic controller firmware.

II. ADVANCED TRANSPORTATION CONTROLLER

Use of traffic signal controllers at intersections had been a conventional task for many decades. Electro-mechanical controllers were the first devices used to control traffic in the early 1940s. However, the use of these electro-mechanical devices had become obsolete due to growing needs, increasing traffic demand and development of advanced microprocessor based controller technology. After this starting period, first microcontroller devices started to be widely implemented in traffic control.

The development of microcontroller implementation in Europe and North America had two different paths although both sides had very similar operational goals, safety constraints and basic phase structure. European controllers were interval-based with implementing mostly Programmable Logic Controllers or customized controllers developed by each vendor. The North American market bases on Ring-Barrier structure (Figure 1) with focus on creating standardized controller and signal cabinet equipment for application just in traffic control. North America had several generations and standards for microcontrollers used in traffic signal control starting from 1970s. Organizations involved in this hardware and firmware standardizations were the Institute of Transportation Engineers (ITE), National Electrical Manufacturers Association (NEMA), and American Association of State and Highway Transportation Officials (AASHTO). The latest effort is defined as

2070 Advanced Transportation Controller (ATC) standard. This standard has clearly defined an area of traffic signal cabinet, controller, engine board and operating system, with all the modular components for operation and communication.

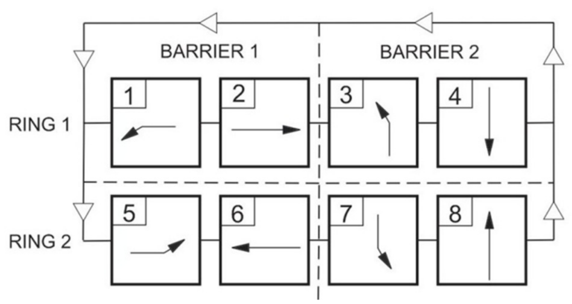

Figure 1: Dual ring eight-phase control defined by NEMA nomenclature

ATC standard is a combination and improvement of all the previous North American operational and hardware standards (NEMA TS1, NEMA TS2, 170). This standard defines all the signal cabinet elements, such as traffic signal controller, power supply and detector rack, malfunction management unit, back panel, etc. In addition to this, this standard defines all the controller elements, such as chassis, central processing unit, controller bus, field input/output, serial communication and modem units, power supply, front panel, and NEMA interface module. In addition, this standard had defined operating system as Microware OS9 or Linux that deals with device management, clock management, file management, and procedure startup. However, this standard has given the opportunity for various third-party vendors to develop their own signal control software upon the standardized hardware structure and operating system. Signal control software is usually purchased separately. This has allowed increased flexibility in development of actual controller software. However, just as this is an advantage, the stated flexibility led to multitude of signal control firmware. Today, there are over 10 different companies developing signal controller firmware on the North American market. This has introduced significant complexity in assessing the actual controller capabilities.

Typical ATC-based signal control software is developed in C or C++ code. An example of that software menu is presented on the Figure 2. On the other hand, a typical programming of European controllers is done using stage sequencing logic (Figure 3). The essential differences among North American ATC and European interval-based controllers is that North American controller software has all the possible operational options already developed in separate submenus. On the other hand, European controllers do not have such established programming structure. This approach leaves more space for development of customized control features (using IF/THEN decisions) for each signalized intersection but is also requires more programming efforts. It is important to emphasize that, with proper level of programming, European controllers can map the same level of functionality of modern ATC.

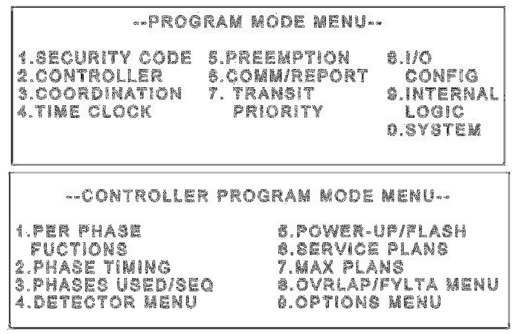

Figure 2: Example of typical 2070 controller menu programming options [1]

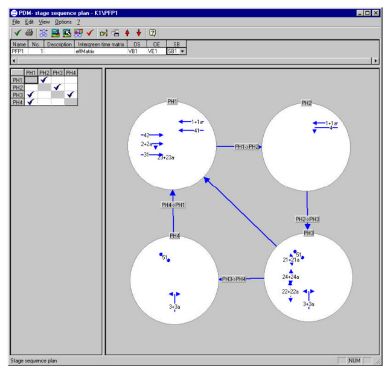

Figure 3: Stage sequencing of typical European controller [2]

III. SOFTWARE-IN-THE-LOOP SIMULATION

Software-in-the-loop simulation (SILS) is a generally known technique used in testing and simulation procedures among various engineering implementations. In the field of traffic control, it is related to traffic simulation. In this implementation, SILS is comprised of a microscopic simulation model, virtual traffic controller and interface for communication between these two components comprise [3]. This system integrates all the advantages of software for microscopic simulation and virtual traffic signal controller. Traffic signal controller element of SILS bases on the virtual software replica of real controller firmware, thus having the same operational logic. The main components of traffic signal controller emulator implemented in this research are virtual controller and virtual database editor. Virtual controller consists of Dynamic-link Library code that is implemented in the tested ATC 2070 controller. The database editor is a Graphical User Interface that allows viewing, editing and printing traffic signal database of the virtual controller. The result is implementation of signal control component in the microscopic simulation with all the control capabilities as there are programming options in tested ATC 2070 controller.

Previous research in the field of evaluation of signal controller features was mostly based on hardware-in-the-loop simulation (HILS). HILS is similar to SILS, except it uses real traffic signal controller input trough hardware connection in real time [4-6]. SILS, on the other hand, is a part of establishing state-of-the-practice for assessment of generally known controller features and their impact on efficiency parameters [7, 8]. The reasoning behind implementing SILS (compared to HILS) is that it has the same flexibility for different testing scenarios with less application complexity that HILS. This way, simulation speed is not constrained with the controller processor communication in real time.

In this research, SILS is the technique identified as able to test the defined controller specifications of operations. SILS is an ideal tool for testing and verification of operational usefulness and usability of controller features.

IV. TESTING FRAMEWORK

Modern traffic signal controllers have over 200 control parameters settings. They are usually grouped in submenus such as Phases, Overlap, Coordination, Plans, Preempt, Transit Signal Priority, Detectors, etc.
The programming options in these submenus can be timing parameters, on/off flags, and selection. Timing options are related to specific time dedicated to some function, such as minimum green or cycle length. Their resolution is at 0.1 seconds, with most of the timing parameters restricted to 255 seconds (such as minimum green or cycle length). Very rarely, some timed parameter is available for setting up to 65535 seconds (e.g., delay on queue detector). On/off flags are there to enable or disable certain function, such as minimum recall or dual entry. Selection settings are used when there are more than two options available for a programmable parameter, such as transition mode (e.g., Best, Long, Best No Ped) or offset reference point (e.g., Lag Force Off, Lead Green, Lag End, Green End). The result of operation for most of the programming options in modern ATC can be tested using SILS. However, due to the scope of this paper, we will only present examples of testing and verification of several selected controller features.

The developed testing framework is intended to investigate questionable controller software capabilities. This is a different approach from the previous research, which had an emphasis on testing the operational effects of generally known controller software features. In the developed framework, testing and verification has been conducted fully utilizing flexibility of virtual traffic simulation environment to trigger control events using various traffic users, their volumes and patterns. Signal control elements have been assumed as constant for finite periods, while the change in traffic conditions was aiming to activate chosen control functions. The different behavior observed as a response to imposed traffic conditions is the key information from this part of assessment process.

Virtual traffic signal controller has been integrated within VISSIM microscopic simulation. Controller resolution was set to 10 Hz, as implemented in the field controllers. Developed model was a coordinated traffic signal control system having fully-actuated operation, based upon 28 detectors (vehicle, pedestrian, transit, PE, and queue). Different users have been included in the model (passenger vehicles, heavy vehicles, pedestrians, bus transit, and emergency vehicles), for providing the reality component of the testing environment. The input of different user to the simulated traffic signal control system is done in various time steps – from 300 to 1800 seconds, aiming to cover a wider range of traffic conditions. Arrival time for special usage vehicles (transit buses and emergency vehicles) has been set at discrete time points. Based on the Time of Day schedule, the control system initially operates in Free operation, after which it transitions to Coordinated operation defined by Time of Day coordination pattern and events. Investigation of controller firmware capabilities is based on the graphical representation of signal time table. This table is an inbuilt VISSIM option and intuitively represents the signal time change and detector actuation calls (y axis) in relation to respective cycle length or Free operation (x axis).

As mentioned before, modern ATC software has significant number of programming features. For the purpose of this paper, research team has decided to show testing of several selected features. Selected features are in the domain of signal preemption and queue detection programming options.

V. SILS TESTING AND RESULTS

Signal preemption (SP) defines as a fully guaranteed termination of normal signal traffic control operation and transfer to a special control operation mode. This mode is usually used for the purpose of servicing rail vehicle or emergency vehicle, and rarely for public transport vehicle passage, and other special tasks (e.g., certain non-intersection locations such as at approaches to one-lane bridges and tunnels, movable bridges, highway maintenance and construction sites, metered freeway entrance ramps, etc.) [9, 10]. SP is usually used for maximizing emergency response. Settings of one preempt can have over 20 programming features, related to different stages of SP process. SP process has three distinct stages: Transitioning in, Serving (Dwell), and Transitioning out (Figure 4).

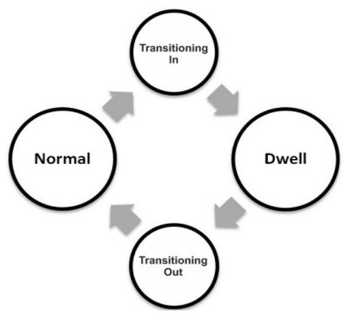

Figure 4: Cycling of signal preemption process

Transition In is usually very abrupt action since it sends the system into Free operation. Transit Signal Priority (TSP) tries to give priority for the public transportation vehicle, while maintaining the system in Coordinated operation. Transition In action is also simpler than Transition Out. During Transition Out, the intention is to return the system back into coordination with minimum effects possible. This is the reason why there are usually several algorithms/programming options available for Transition Out. In the case of tested controller, there are Normal, Next, and In Step options for Transition Out to be selected per specific SP. All of these options have been tested using SILS, and the results are presented in the Figures 5-7. In three testing cases, all the traffic and control parameters were maintained the same, except the selection of Transition Out option.

SP is activated through check-in detector number 511 and Dwell is terminated after check-out call on the detector number 521. At the moment of SP call arrival, system is in Coordination, serving main street phases 3, 4, 7, and 8. Emergency vehicle arrives on the minor street north approach and operation transitions into Free. The dwelling phase is phase six, that serves the movement of emergency vehicle through the intersection. After the check-out call, as observed on the Figure 5, during Normal transitioning out the operation returns to the predefined exit phases for the preempt. In the case of SP on phase 6, the exit phases are set as phases 4 and 8, and this is where operation transitions out. In the case of Next Transition Out programmable option (Figure 6), operation exits to the phases that were following the phase interrupted by SP call. Since check-in call was received during timing of phases 3, 4, 7, and 8, the next phases after these in the RBC structure were phase 2 and 6. Those phases were transitioning out had a hold until local cycle timer counted back into coordination. Finally, In Step transition out keeps the local cycle timer counting and performs the transition directly into scheduled point of coordination, as if there was no interruption. This operation can be observed on the Figure 7.

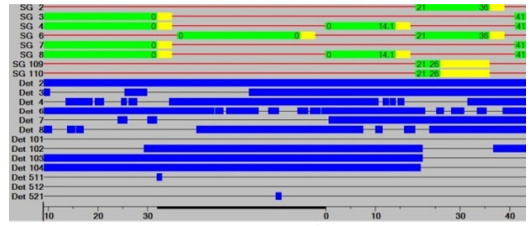

Figure 5: Transition Out - Normal

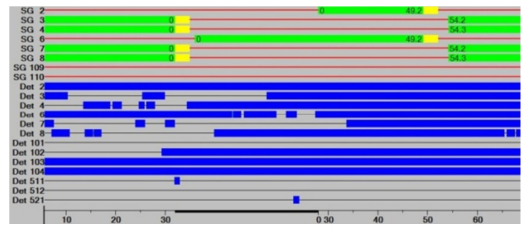

Figure 6: Transition Out - Next

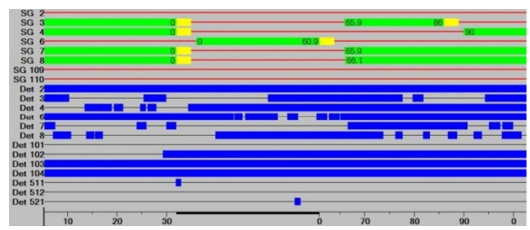

Figure 7: Transition Out – In Step

Beside presented SP programming options, this paper presents the testing of several queue detection programming options. Queue is defined as a line of vehicles, bicycles, or persons waiting to be served by the signal control system [11]. The focus of signal control is mostly on the queue of vehicles; usually overflow left over resulting from split/cycle failure. The inclusion of change in signal operation due to queue detection usually requires additional detectors. In the testing case, those detectors operate on the delay on/delay off principle. Delay on is the number of seconds a detector has a constant activation call. This is the trigger for activating queue detector and further queue detection actions. The abort of any actions taken is done after a predefined time or after queue detector loosing activation call for a Delay off number of seconds.

Due to queue detection, control mode can be changed into Free or alternative coordination pattern can be initiated. In addition, initiated action can be, for example, enabling of Max 2, Max 3, or start of Preempt call [12]. Queue detector in this testing is set to be upstream detector number 15. This detector is connected to detector 4 thus relating its activation to operation of phase 4.

Delay on for activation is set to 10 seconds, and delay off for deactivating is set to 5 seconds. Evaluation situation is assumed to appear after 900 seconds of simulation time, when the volume on the west approach is significantly increased to create queues.

Figure 8 presents the extension of green time beyond the cycle length limit after the queue detection. This is a result of activation of Max 3 time for phase 4.

Figure 9 presents another option for clearing the detected queue. As it can be noted from this figure, the activation of preempt on phase 4 is the result from queue detected on detector 15. With this action, controller is placed in Free operation and operates as under SP call. The “check-out” call in this case is Delay off call from the detector 15. From this figure we can also observe that Free operation is not caused by any other calls from SP or TSP detectors (2xx and 5xxx, respectively).

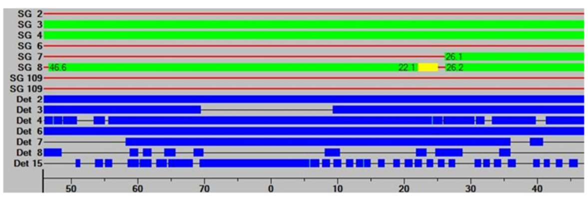

Figure 8: Max 3 activated after queue detection

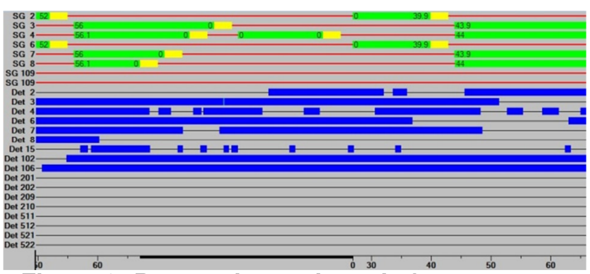

Figure 9: Preemption activated after queue detection

This testing resulted in additional information on the operation of programming features. As presented under reasoning for introducing SILS, this technique has successfully incorporated the requirements for in-depth assessment process. Testing of features presented here is a guiding example of testing any other unknown or unclear features in signal control software. The resulting information obtained could be consequently used as an input to the defendable decision-making process.

VI. Conclusion and Future Work

The desire for the improvement of the system’s and user’s efficiency, safety and environmental parameters in a specific traffic signal control system obviated the desire to enhance the utilization of existing controller features. For better understanding of these features, an essential need for an assessment of signal controller firmware features is identified. This research focuses on the specifics of 2070 ATC hardware and software and their similarities and dissimilarities with European controllers. In addition, this paper points out the complexities introduced from the perspective of users and vendors in relation to controller market. The critical point identified is the multitude of ATC signal control software and their programming options. In light of all this, the research presented here is focusing on developing an improved assessment process for vendor developed custom and unclear features. The focus of assessment on unknown control software capabilities is due to their potential to improve the control results. This paper is presenting the framework and methodology for conducting controller software assessment. This in-depth assessment is considered as essential for obtaining informed, defendable, and ultimately, optimal solutions that would shape current and next generation control system.

Finally, this testing procedure could stimulate and guide further development of user specific controller firmware features. This is possible through recognizing existing features in available field controllers and by developing functional requirements for signal operation under specific transportation agency purview. In addition, the additional information obtained from SILS-based assessment could give the guidance for developing features in European interval-based controllers using the readily available multitude of programming options existing in modern ATCs.

Acknowledgement

This work was supported by the Virginia Center for Transportation Innovation and Research (VCTIR) under the Next Generation Control Project.

REFERENCES

[1] "Voyage Software Operating Manual - Version 3.0.0," Northwest Signal Supply, Inc.2009

[2] "SITRAFFIC P2 - Planning and Configuration System for Signal-Controlled Intersections and Networks," Siemens Traffic Management Solutions, 2011.

[3] A. Stevanovic*, et al.*, "Microscopic Modeling of Traffic Signal Operations: Comparative Evaluation of Hardware-in-the-Loop and Software-in-the-Loop Simulations," *Transportation Research Record: Journal of the Transportation Research Board,* 2009.

[4] E. Nelson and D. Bullock, "Impact of Emergency Vehicle Preemption on Signalized Corridor Operation: An Evaluation," *Transportation Research Record: Journal of the Transportation Research Board,* vol. 1727, pp. 1-11, 2000.

[5] D. Bullock*, et al.*, "Hardware-in-the-loop simulation," *Transportation Research Part C: Emerging* Technologies*,* vol. 12, pp. 73-89, 2004.

[6] J. Yohe and T. Urbanik, "Advance Preempt with Gate-Down Confirmation: Solution for Preempt Trap," *Transportation Research Record: Journal of the Transportation Research Board,* pp. 40-49, 2007.

[7] J. Obenberger and J. Collura, "Methodology to Assess Traffic Signal Transition Strategies for Exit Preemption Control," *Transportation Research Record: Journal of the Transportation Research Board,* vol. 2035, pp. 158-168, 2007.

[8] A. Stevanovic*, et al.*, "Optimizing Traffic Control to Reduce Fuel Consumption and Vehicular Emissions: Integrated Approach with VISSIM, CMEM, and VISGAOST," *Transportation Research Record: Journal of the Transportation Research Board,* vol. No. 2128, 2009.

[9] Institute of Transportation Engineers. American Association of State Highway and Transportation Officials, National Electrical Manufacturers Association, "National Transportation Communications for ITS Protocol 1211," in *Object Definitions for Signal Control and Prioritization*, ed, 2008.

[10] Federal Highway Administration, U.S. Department of Transportation, "Manual on Uniform Traffic Control Devices," ed, 2009.

[11] Peter Koonce, *et al.*, "Traffic Signal Timing Manual," U.S. Department of Transportation, Federal Highway Administration2008.

[12] "D4 Traffic Signal Controller Software (Version 1.5c)," Fourth Dimension Traffic2009.