MODELING AND SIMULATION OF IEC 61850 REQUIREMENTS APPLIED TO AN AUTOMATED PEOPLE MOVER'S CONTROLLER

Guilherme Kunz
Western Paraná State University, Centro de Engenharias e Ciências Exatas, Foz do Iguaçu, Brazil
guilhermekunz@gmail.com

José Machado
University of Minho, Mechanical Engineering Department, CT2M Research Centre, Guimarães, Portugal
jmachado@dem.uminho.pt

Eduardo Perondi
Federal University of Rio Grande do Sul, Mechanical Engineering Department, Porto Alegre, Brazil
eduardo.perondi@ufrgs.br

Keywords: IEC 61850, Modelling, Simulation, Automated People Movers.

Abstract: *Automated People Movers* (APM) are systems for passenger transport with fully automated operation and high frequency service. For this study, we proposed the adaptation of the standard IEC 61850 (design to be used in electric power systems based in intelligent electronic devices) to allow its application to an APM system named *Aeromovel* installed in Porto Alegre, Brazil. *Aeromovel* is a nonconventional *Automatic People Mover* whose operation principle is based on pneumatics. This paper proposes the use of two analysis techniques, Simulation and Formal Verification, in order to guarantee the desired behaviour for an APM propulsion system composed by a centrifugal fan and ten (on-off and proportional) pneumatic valves driven by pneumatic pistons. This approach is based on the use of timed automata and UPPAAL software.

1 INTRODUCTION

An *Automated People Mover* (APM) is a fully automated, grade-separated mass transit system. The term is generally used only to describe systems serving relatively small areas, such as airports, downtown districts or theme parks, but is sometimes applied to considerably more complex automated systems. Usually they circulate in headways that do not interfere with other traffic ways in order to guarantee safety for passengers and security for the system (IEEE, 2004).

An APM performs automatically the control of movement, the execution of the safety instructions and the direction of the trains. The automatic accomplishment of these functions is assured by the *Automated Train Controller* (ATC) system that is composed by the following sub-systems: ATP - *Automatic Train Protection*, ATO - *Automatic Train Operation* and ATS - *Automatic Train Supervision*.

In order to guarantee the communication among these systems, the standard *IEEE Standard for Communications-Based Train Control (CBTC) Performance and Functional Requirements* (IEEE, 2004) must be followed. This standard describes the functional requirements and also the communications performance concerning the described controller systems of the APM (*Communications Based Train Control* - CBTC).

For example, to activate the train braking system, it is needed that the central control has information, constantly updated, about the speed and current location of each vehicle on the highway and the time required for activation of the brake system, in order to perform the stopping under the deceleration curve thereby avoiding the collision between vehicles.

For integration of the ATC, it is used the standard IEEE Standard for Communications Protocol Aboard Trains (IEEE, 1999) that defines the communication protocol between vehicles and

inner vehicles. This standard defines two solutions according to the application: the protocol 1473-L (LonWorks) and 1473-T (TCN).

In Sullivan (2001) there was observed, in the types 1473-L and 1473-T, the lack of support for new demands for video transmission, the missing of IP interfaces preventing communication via Ethernet, for example, and the lack of protocols used for systems integration Advanced Train Control System (ATCS).

According to Hewings (2008) protection system and train control is traditionally based on wired and centralized circuits. Although they generally have a simple design, there are serious difficulties in the installation and maintenance. As there is increased demand on the system, there are advantages in choosing an open architecture, with a simple communication system. These concepts are largely addressed in the development of IEC 61850, designed to be a communication standard for electrical substations based on the use of IEDs (Intelligent Electronic Devices), which occupy the place of older protective relays, combining functions of protection, control and communication in the same equipment. In general, its application results in the following benefits (Hewings, 2008):

- Reduced cabling.
- Reducing the cost and installation time.
- Increased capacity for monitoring and control systems protection.
- Separate infrastructure from functionality.
- Interoperability.

The IEC 61850 standard has requirements such as real-time control and distributed object orientation and provides a standard for integration of substations from specification of reporting requirements, functional characteristics, data structure and the nomenclature for devices and data. It also provides standards for operational characteristics, such as how to interact with the applications of control devices and how they should be tested for compliance of the system.

Currently, there are applications in the areas of hydropower, wind energy and distributed generation. It is proposed, in the present study, the expansion of the IEC 61850 standard to APM systems, performing a control CBTC.

The application of all IEC 61850 requirements to an Automated People Mover's Controller is a large and very complex task. The approach that seems to accomplish the goals of this large project is to use, first, skills in modelling, creating a large global model that must consider the communication protocols proposed by the standard and guaranteeing the accomplishment of all time delays; second, to simulate the very large model using appropriated tools and software; and, third, to use Formal Verification techniques in order to guarantee a set of behaviours defined by the standard.

As it is an ongoing work, this paper presents some aspects related with the communication protocols – proposed by the IEC 61850 standard – and aspects related with Simulation and Formal Verification of the communications requirements specified by those protocols.

Considering the IEC 61850 standard protocols, the GOOSE (Generic Object Oriented Substation Event) is the first one being analyzed. The Automated People Mover that is treated in this study uses pneumatic power for displacement, in which the combination of a pneumatic propulsion system control and the control of a set of *on-off* and *proportional* valves is crucial to guarantee the system dependability.

In order to handle with this complex problem, the main idea is to use, in a complementary way, Simulation and Formal Verification analysis techniques (Machado et al. 2011)

Several formalisms can be used to model timed systems. Timed automata were adopted as the modelling formalism for system modelling due to two main reasons: first, the study of the proposed system needs to take time into account; and, second, it is the input formalism of the UPPAAL model-checker (Behrmann et al., 2004). Even if UPPAAL is a Model-Checker, in this step of this very large project, and specifically in this paper, it is used, only, as a Simulator. As the next step of our approach is to use Formal Verification Technique, we believe that it is a good solution for this task.

In order to achieve the main goals of this paper the Section 2 presents the case study; Section 3 presents the GOOSE protocol model; Section 4 is devoted to presentation of the simulation results and finally, Section 5, presents some conclusions about the study presented herein.

2 CASE STUDY: *AEROMOVEL*

The main features of the technology are the exclusive *Aeromovel* traffic on the route, the high ratio of useful load/weight carried and external traction. These characteristics are due, respectively, of the fact that car travel above ground in a unique way and have external power system. This makes it relatively lighter than other similar transportation systems, allowing less robustness for the beams

where it operates, reducing the costs of construction, installation and maintenance of the system.

The power unit, known as power train group or propulsion system, is responsible for generating pressure differential and is basically composed of an asynchronous electric motor that drives the industrial centrifugal fan. Each power train group is connected to the main duct through a pipeline with $1m^2$ of cross-sectional area.

The proposed fluidic power system (Fig. 1) consists of an industrial centrifugal fan (with air flow of up to $10^6 m^3/h$) and a set of two proportional valves (*VP0* and *VP1*) that allow control of pressure and consequently the force imposed on the vehicle and eight on-off valves (*V0*, *V1*,…,*V7*) They allow the effect of the fan switch on the main duct through which the vehicle moves, and can perform inflation or exhaust air as seen in Fig. 1. The valves used in the *Aeromovel* system are characterized by causing obstruction of flow from angular movement. Pneumatic pistons are used to rotate the flaps of the valve due to high flow rates involved.

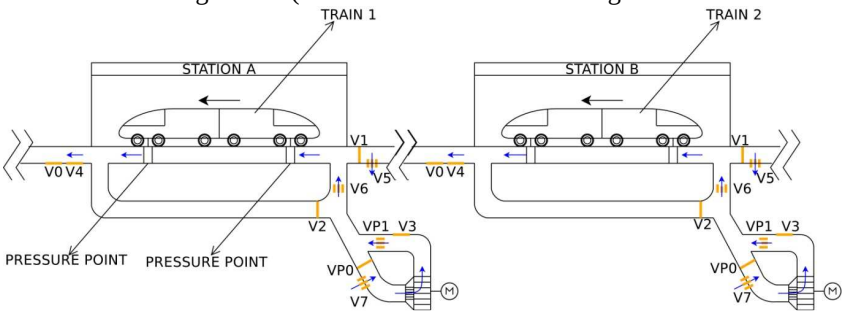

Figure 1. Layout of power train group - Push to Left

3 GOOSE PROTOCOL MODEL

In order to detail the explanation of the study realized, this paper presents, only, the study of one to one GOOSE messages. It must be highlighted that all the system (controller and plant, in closed-loop behaviour) was modelled and the entire model is composed by sixty-two (62) timed finite automaton modules.

The train control system is usually centralized, but, aiming a solution based on the IEC 61850 standard (Hewings, 2008), the models were developed based on distributed controllers so, in the models, it is considered real-time dedicated to each individual device. The units are connected to a communication bus that provides information exchange with other processing unit responsible for interfacing with the user, thus reducing the processing request individually. In general, the decision to use a distributed control system is motivated by cost reduction and increased system flexibility and control, in this particular case, the distance between the elements of the system.

Models of plant system devices and controllers were developed using timed automata formalism and analyzed using UPPAAL software for simulation. The model was divided into the following templates: Goose Server, Goose Client, Bus and Logical Node.

With respect to the implemented GOOSE protocol model, were taken in account the following characteristics (see Fig. 2):

- The messages are asynchronous and unsolicited;
- The GOOSE protocol is encapsulated directly in Ethernet layer. The messages are connectionless so the model does not verify the connection stability (without confirmation from receivers).
- The messages are multicast. To the multicast only clients or servers in the same VLAN (virtual LAN) can send or listen packages. Must be a Bus Model to each VLAN (the template model has facilities to do this configuration).
- How there isn't confirmation from receivers, the retransmission is used to increase the probability of successful reception.

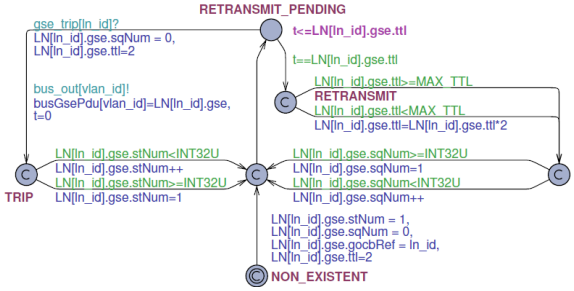

Figure 2. Goose Server Model

The Bus Model has a FIFO (First In, First Out) queue with 4 ms (milliseconds) delay and the total delays of frames flow introduced by network and communication processors are allocated only in the Bus Model (typical GOOSE total transfer time is 4ms) (see Fig. 3).

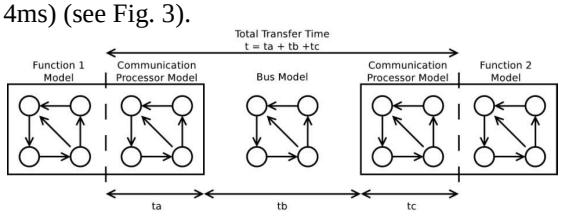

Figure 3. Communications Delay

The Gosse Server has three basic states: NON-EXISTENT, RETRANSMIT-PENDING and RETRANSMIT. In the case of Logical Node has been configured to send Goose Messages (*GoEna==true*) the server transmit the first package setting *SqNum* to 0 (this variable will be incremented for each transmission, but will rollover to 1 and set to zero when *StNum* is updated), *StNum* to 1 (this variable is used to define how many times the equipment has changed state) and *timeAllowedtoLive* to 2 (this variables are in the structure called *SendGooseMessage*). The time to wait for the next transmission (*timeAllowedtoLive*) is set to 2^n (n=1) and is incremented by *n+1* until 1024ms.

The Fig. 4 shows the waiting time for the next re-transmission when *th* is the *heartbeat* time (1024ms), *t0* is an indeterminate time by an asynchronous changing status, *t1* is 2^1 ms, *t2* is 2^2 ms, *t3* is 2^3 ms, *t4* is 2^4 ms and so forward.

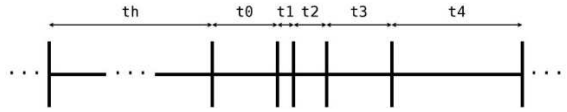

Figure 4. Time to wait for the next re-transmission.

Goose Server send messages to Bus Model by copying the struct *SendGooseMessage* to struct *busGsePdu* (according VLAN) and send a signal by channel to Bus Model. The Bus Model receives the signal and copies the *busGsePdu* structure to a queue and does a time registry. After the delay (4 ms) the Bus Model remove the data from the queue and copy again to *busGsePdu* structure, sending a broadcast channel to all Goose Clients which are listening the VLAN. The Bus Model is the same to Sample Value and Goose Messages, but has difference in the queue because those messages have different structures.

The Goose Clients receive by broadcast channel signal and copy the *busGsePdu* structure to local memory, verifying interest (initially configured). If has not interest the data is discarded and the Goose Client comeback to the listen state. If it is important information arriving, then the Goose Client model call the Logical Node Controller do the necessary actions and comeback to the listen state.

4 SIMULATION RESULTS

For all the models, the range of all variables has been limited in order to decrease the necessary computational capacity to obtain results, when executing formal verification tasks. For all the locations of the entire automata model - with exception of the "committed" locations - it is necessary a time interval to allow evolutions, in all automaton models, from a location to another location.

Concerning simulation results, the data of the file *XTR* (simulation registry) have been used to obtain the diagram of Fig. 5. In this figure, it is possible to see the retransmission messages by the increment of the *SqNum* and *stNum* variables in the time. The simulated behavior is the expected one for this system. However, the step considered - after this one – will be to consider also formal verification in order to be sure about the behavior of the Goose communications.

At this moment, formal verification has been used only for deadlock violation (formal description: "`A[] not deadlock`"), with DBM - Difference Bounded Matrices (Dill, 1989) state space representation, but not yet considered, concerning all the system with GOOSE protocol, because the model of the system is now a very large and complex model - composed by sixty-two (62) modules - and the computational capacity that we have available is not enough to obtain results. To solve this problem, it will be necessary to use partial formal verification (Holzmann, 1991 and Holzmann, 1998) and/or abstraction modeling techniques (Balarin, 1996) to handle with this very large model, in order to obtain Formal Verification results.

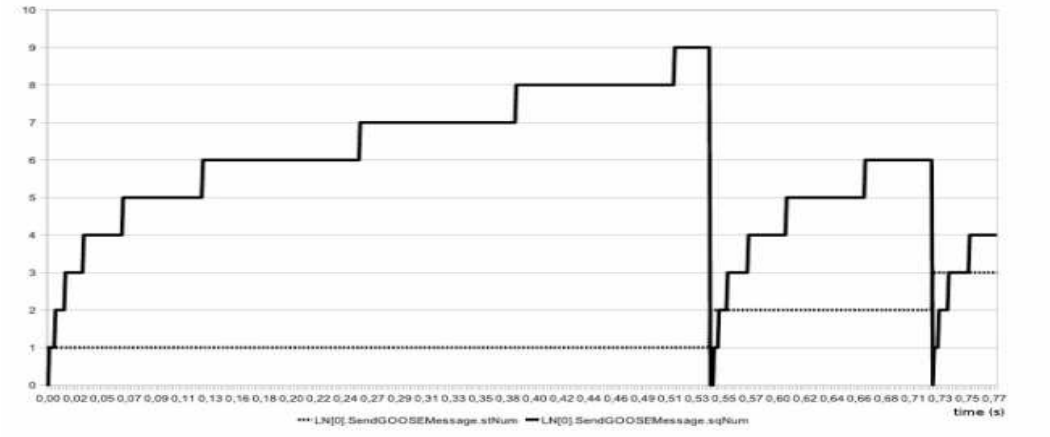

Figure 5. Simulation Results.

5 CONCLUSIONS AND FUTURE WORK

The use of simulation and formal verification techniques for analysis of the studied APM controller was very helpful and allowed us obtaining good results for studying the IEC 61850 GOOSE protocol.

With this study, it is shown, in this paper that a distributed controller - corresponding at a part of a complex system - has been verified and it is concluded, also, that the controller accomplishes the main behavior desired for the system and the delays proposed by IEC 61850 standard.

All range of variables, in the models, are used in conformance with IEC 61850. However, in order to accomplish the next step of our work - formal verification of the entire model - reducing variable size will be possible to improve formal verification performance, because it decreases the time and memory consuming used to do the formal verifications. For example, *stNum* has 2^{32} bytes but we intend to use it with 2^8 bytes because the functionality is the same.

As future work, other partial controllers will be verified - concerning the same system - and, finally, an abstraction of each part of the controller will be verified in order to guarantee the desired behavior for the system, considering all the distributed controller system.

ACKNOWLEDGEMENTS

Guilherme Kunz is supported by the PTI C&T program (*Fundação Parque Tecnológico Itaipu* - FPTI-BR). The authors would like to thank to PTI C&T/FPTI-BR for financial support and to CESUP-UFRGS for access to the clusters.

REFERENCES

Balarin, Felice. Approximate reachability analysis of timed automata. In 17th IEEE, Real-Time Systems Symposium. IEEE Computer Society Press, 1996.

Behrmann, G., David, A., and Larsen, K. G. (2004). A tutorial on UPPAAL. *4th International School on Formal Methods for the Design of Computer, Communication, and Software Systems (SFM-RT'04)*. LNCS 3185.

Dill, David L., Timing Assumptions and Verification of Finite-State Concurrent Systems. LNCS 407. Springer Berlin 1989, pp 197-212.

Holzmann , Gerard J.. Design and Validation of Computer Protocols. Prentice-Hall, 1991.

Holzmann , Gerard J.. An analysis of bitstate hashing. Formal Methods in System, Design, 13:289–307, 1998.

Hewings, D. Introduction of integrated protection and control to railway electrification systems. In *Proc. IET 9th International Conference on Developments in Power System Protection DPSP 2008*, pages 68–73.

IEEE (1999). Standard for Communications Protocol Aboard Trains.

IEEE (2004). IEEE standard for communications-based train control (CBTC) performance and functional requirements.

Machado, J., Seabra, E., Campos, J., Soares, F. and Leão C. (2011). Safe controllers design for industrial automation systems. Computers & Industrial Engineering 60 (2011) 635–653.

Sullivan, T., IEEE rail transit vehicle interface standards update, 4th International Conference on Communications Based Train Control, 2001.