Modeling and Simulating the controller behavior of an Automated People Mover using IEC 61850 Communication Requirements

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

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

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

***Abstract*—*Automated People Movers* (APM) are systems for passenger transport with fully automated operation and high frequency service. Trains controllers are traditionally centralized and based on wired circuits, although they generally have 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 and distributed. These concepts are largely addressed in the development of IEC 61850. In this study we proposed the adaptation of the standard IEC 61850, design to be used in electric power systems to be applied in an APM system named *Aeromovel* installed in Porto Alegre, Brazil. *Aeromovel* is a nonconventional Automatic People Mover whose operation principle is based on pneumatics. A model, based on timed automata formalism, is proposed for IEC 61850 communications requirements and respective simulation results are presented.**

I. 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 [1].

From the existing APMs, about one-quarter of them function as urban metros; the remainder are short-range, privately built shuttles and loops that operate as an integral part of the functioning of airports, amusement parks, institutions, and shopping centre across North America, Europe, and Japan. They all have in common a high level of frequent service. Some of these (that belong to the earlier generations), have been operating since the late 1960s [2]–[7].

An APM realizes automatically the control of movement, the execution of the safety instructions and of the direction of the trains. The automatic realization of these functions is assured by the *Automated Train Controller* (ATC) system that is composed by the following sub-systems:

- ATP - *Automatic Train Protection*. Protection against collisions, excess of speed, invasion of the train way, among other danger situations;
- ATO - *Automatic Train Operation*. Speed control, programmed stops at the stations and control of the doors, among other operations of the same kind (usually, in a non-automated transportation system, these operations would be associated at the train operator).
- ATS - *Automatic Train Supervision*. Functions of monitoring and adjustment of the individual performance of each train, in order to guarantee the schedule of departures and arrivals of trains.

An ATC must include, an ATP system and, optionally, it can include the ATO and/or ATS systems. In order to guarantee the communication among these systems, the standard *IEEE Standard for Communications-Based Train Control (CBTC) Performance and Functional Requirements* [1] 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). The main characteristic of CTBC include:

- Information about the precise positioning of the train, not-dependent of the sensors of the way.
- Continuous communication between the train and other processes that are not directly related with him.
- Verification of the train control conditions for the ATP. Functionalities of ATO and ATS can be also realized.

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

The IEEE Standard for Communications-Based Train Control (CBTC) Performance and Functional Requirements [1] describes the functional and performance requirements for using CBTC systems. This standard provides typical values, such as, for example, the resolution of speed, range and reaction time of communication equipment used in the CBTC.

For integration of the ATC, it is used the IEEE Standard for Communications Protocol Aboard Trains [8] that defines the communication protocol among instrument vehicles and among vehicles. This standard defines two solutions according to the application: the protocol 1473-L (LonWorks) and 1473-T (TCN).

The type 1473-L is based on EIA 709.1-1998 and on the EAI 709.3-1998. It can be configured to support buses sensors (local sensor bus - LSB) or applications (local bus vehicle applications - LVB). The LSB connects the sensors and the LVB connects embedded devices to the vehicle operating system.

The type 1473-T is based on IEC 61375-1-1999 and is dedicated for applications that require time determinism and is divided in protocols WTB (Wire Train Bus) and MVB (Multifunction Vehicle Bus). The WTB interconnect trains operating units and the MVB interconnect embedded devices to the vehicles operating system. One major benefit of this configuration is the possibility of using equipment from different manufacturers [9].

In [10], it 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 [11], protection system and train control are traditionally based on wired circuits with centralized operation. Although they generally have a simple design, there are serious difficulties in the installation and maintenance. In case where 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), that is replacing the older protective relays. The IEC 61850 combine functions of protection, control and communication in the same equipment. In general, its application results in the an following benefits [11], [12]:

- Reduced cabling.
- Reduced cost and installation time.
- Increased capacity for monitoring and control systems protection.
- Separated infrastructure and functionality.
- Interoperability.
- Comprehensive ”System Configuration Language” (SCL) for the whole life-cycle from design to engineering, operation and maintenance.
- Supports multi-vendor systems.

The IEC 61850 standard has requirements such as real-time control and distributed object orientation programming 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, defining, for example, how to interact with the applications of control devices and how they should be tested for compliance analysis of the system.

With regard to the requirements of APM control systems, IEC 61850 is based on ASN.1, according to the IEEE Trial-Use Standard for Message Set Template for Intelligent Systems Transportation (IEEE, 2000). IEC 61850 uses the following communication protocols:

- Generic Object Oriented Substation Event (GOOSE). Used for asynchronous, unsolicited and heartbeat messages. This message is sent to the network with high priority and real-time requirements;
- Sample Value (SMV). Used for exchanging data between machines. Through this message it is possible to perform signal processing distributed equipment. It also has real time requirements;
- Manufacturing Messaging Specification (MMS). It is used for supervisory system communications and remote configuration. It has no real-time requirements.

According to the IEC 61850, the control system is divided into three levels related the hierarchy of functionality:

- Functions of the process level: In this level, all functions that have direct interface with the process are located. The state of the system are determined through the use of analog or binary indications. Traditionally, these signals are transmitted via wiring in the form of current intensity or voltage and auxiliary switches.
- Functions Bay Level: All the functions that act directly on the level of process equipment are located in this level. This functions have features such as switching equipment.
- Function Level Station: In this level all the other features are located. These are divided into two groups:
  - Related to Processes: Functions that use data from more than one level of bay.
  - Related to interfaces: Functions related to communication with HMI (human machine interface), SCADA systems (Supervisory and Data Acquisition) or with a remote station.

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 CBTC.

Besides the benefits mentioned by [11]. The IEC 61850 presents the possibility of integration of APM control system with power systems, such as distributed generation systems, which may relate the output from the vehicle with the goal of minimizing costs.

All protocols are object-oriented and based on distributed control. The data types are described below:

- Physical Device. Object accessed by the network address.
- Logical device. Contains a collection of logical nodes deployed in an IED that is not distributed.
- Logical Node. Functions that represent the actual functions in the system.

- Attributes and Data. Properties of logical nodes, for example, position or configuration.

To use IEC 61850 standard in APM systems, safety aspects related with operation of these systems are crucial. There are safety requirements that must be accomplished when these systems are operating. These safety requirements are defined by International Standards - as mentioned above - and cover all the aspects of the system controller.

The application of all IEC 61850 requirements to an Automated People Mover Controller is a large and very complex task. One suitable approach to accomplish the goals of this large project is to create a large global model that considers the communication protocols proposed by the standard, guaranteeing the accomplishment of all time delays. To simulate the very large model it is necessary to choose the appropriated tools and software, so, it could be used Formal Verification techniques in order to guarantee a set of behaviors defined by the standard.

As it is an ongoing work, in this paper, it is presented a model for IEC 61850 communication requirements and respective simulation results when developing a controller specification of an APM.

In the IEC 61850 standard protocols context the GOOSE is the first one that must be analyzed. The Automated People Mover that is used in this study uses pneumatic power for moving the vehicles. In this system 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. To this we used Formal Verification and Simulation Techniques.

Several formalisms can be used to model timed systems. Timed automata were adopted as the modeling formalism for system modeling due to two main reasons: first, the study of the proposed system needs to take time evolution into account; and, second, time is the input formalism of the UPPAAL model-checker [13]. Even if UPPAAL is a Model-Checker, in this paper, it is used only as a Simulator. The next step of our approach will be to use Formal Verification Technique.

This paper os organizes as follows: Section II presents the *Aeromovel* system while Section III is devoted to the IEC 61850 requirements modeling. Section IV presents the results of the simulation of the controller specification and, finally, conclusions and perspectives of future work are presented in Section V.

II. *Aeromovel* SYSTEM

The main features of the *Aeromovel* technology are the exclusive traffic on the route, the high ratio of useful load/weight carried and external traction. These characteristics are due, respectively to the fact that the car travels above the ground in a unique way and have external power system. This makes it relatively lighter than other similar transport systems, allowing less weight for the beams over which it operates, reducing the building installation and maintenance of the system [14].

The *Aeromovel* uses rail technology in the interface between the vehicle and the ground. Thus, as the friction metal/metal below the rubber/concrete less energy is lost duw to friction effects. The vehicle has four-wheel independent sets, allowing the *Aeromovel* to make curves with radii smaller than conventional trains, which have fixed wheels on the axes. The flaps are articulated, allowing the vehicle to make turns and moves uphill and downhill without contact with the duct wall [14].

The power unit, known as power train group or propulsion system, is responsible for generating differential pressure and is basically composed of an asynchronous electric motor that drives and industrial centrifugal fan [15]. 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 to control the pressure and, consequently, the force imposed to the vehicle and eight on-off valves (*V0*, *V1*,...,*V7*). That impose the effect of the fan switching on the main duct through which the vehicle moves. They can perform inflation or exhaust of the air on the duct, as seen in Fig. 1. The valves used in the *Aeromovel* system are characterized by causing obstruction of flow throught angular movement. Pneumatic pistons are used to rotate the flaps of the valve.

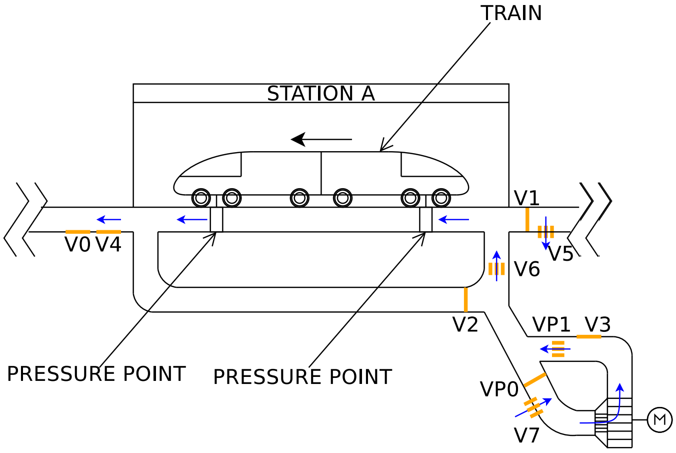


Fig. 1. Layout of the power train group

According to [16] the *Aeromovel* system can be segmented into sections between two stations, which are called ”Standard-Block”. The standard block is formed by two power train groups, one at each station and a vehicle. This configuration allows three different types of operation modes of the system:

- *Push* - the vehicle is pushed by the pressure caused by the operation of the power train group upstream of vehicle. In the chamber downstream of the vehicle, the atmospheric valve is opened, communicating the duct to the atmosphere.
- *Pull* - the vehicle is pulled by the low pressure caused by the operation of power train group downstream of the vehicle. In the chamber upstream of the vehicle, the atmospheric valve is opened, connecting the duct to the atmosphere.

- *Push-Pull* - both power train groups are connected to the duct and the two atmospheric valves are closed. Thus, the vehicle moves due to the upstream pressure and downstream *vacuum*. In this form of operation, the vehicle may develop higher speeds.

One of the difficulties of working with the power train group is that the change of states (from push to pull for example) may cause safety problems for people and security problems for the equipment because the valves can briefly set up a power train group in addition to the three states mentioned above. To avoid making changes of states of the valves in sequence (which implies a longer time to change) it is proposed in this paper, the inclusion of a condition called *OFFLINE*, where the power train group does not influence the movement of the vehicle, independently of the state of the motor, since the segments valves remain closed (*V1* and *V4*) while the atmospheric segment valves remain opened (*V0* and *V5*). Thus, independently of the other valves, there is no interference in the movement of the vehicle while the propulsion system remains in the *OFFLINE* state. This state is used during the exchange process between the states *PUSH* and *PULL* or when the vehicle remains stopped at the station.

III. IEC 61850 REQUIREMENTS MODELING

This paper details the study of peer-to-peer GOOSE messages. All the system (controller and plant, in closed-loop behavior) is modeled 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 [11], the models were developed based on distributed controllers so, in the models, it is considered real-time operation 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 by an increased system flexibility.

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 and Bus and Logical Node. With respect to the implemented GOOSE protocol model, the following characteristics were taken in account (see Fig. 2 and Fig. 3):

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

The Bus Model (see Fig. 4) 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 (a typical GOOSE total transfer time is 4ms) (see Fig. 5).

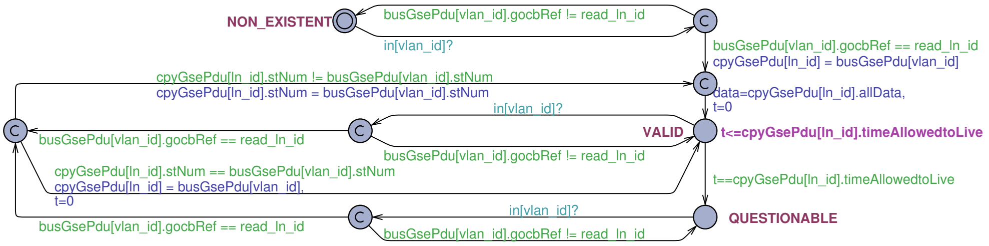
The GOOSE Server has three basic states: NON-EXISTENT, RETRANSMIT-PENDING and RETRANSMIT. The Logical Node has been configured to send GOOSE Messages (*GoEna==true*). The server transmits the first package setting *SqNum* to 0 (this variable is incremented for each transmission, but will rollover to 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.

Fig. 6 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 2^1ms, $t2$ is 2^2ms, $t3$ is 2^3ms, t4 is 2^4ms and so on.

GOOSE Server sends messages to Bus Model by copying the structure *SendGooseMessage* to the structure *busGsePdu* (according VLAN) and sends 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 removes the data from the queue and copies again to the *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).

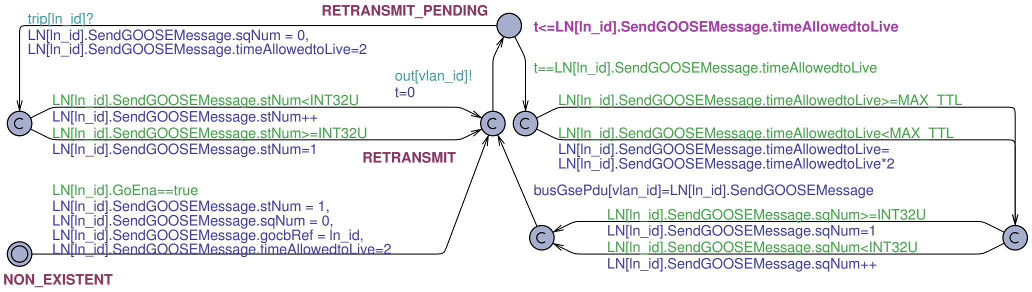
The GOOSE Clients receives the signal by broadcast channel and copies the *busGsePdu* structure to the local memory, verifying the interest (initially configured). If the data is not important, it is discarded and the GOOSE Client comeback to the *listen* state. If the information that is arriving is important, then the GOOSE Client model ”calls” the Logical Node Controller, does the necessary actions and comeback to the *listen* state.

To integrate the *Aeromovel* models with GOOSE Protocol we use, according to IEC 61850, one Logical Node to one function or equipment on the system. In the Fig. 7, the GMP Controller communicates with Valve Controller by GOOSE Messages. For example, if the GMP Controller needs to change the GMP to offline mode, a command will be sent to the GMP Controller Logical Node end packet (array form) in a GOOSE Message format by the GOOSE Server. This message is sent to Bus and, according to the delay, will be received by Valve Controller GOOSE Client. GOOSE Client will verify the packet and will send a message by channel to the Valve Controller which changes the valve model according with time to change the valve status.



NON_EXISTENT
busGsePdu[vlan_id].gocbRef != read_ln_id
in[vlan_id]?
busGsePdu[vlan_id].gocbRef == read_ln_id
cpyGsePdu[ln_id] = busGsePdu[vlan_id]
cpyGsePdu[ln_id].stNum != busGsePdu[vlan_id].stNum
cpyGsePdu[ln_id].stNum = busGsePdu[vlan_id].stNum
data=cpyGsePdu[ln_id].allData,
t=0
in[vlan_id]?
VALID
t<=cpyGsePdu[ln_id].timeAllowedtoLive
busGsePdu[vlan_id].gocbRef == read_ln_id
busGsePdu[vlan_id].gocbRef != read_ln_id
cpyGsePdu[ln_id].stNum == busGsePdu[vlan_id].stNum
cpyGsePdu[ln_id] = busGsePdu[vlan_id],
t=0
t==cpyGsePdu[ln_id].timeAllowedtoLive
in[vlan_id]?
QUESTIONABLE
busGsePdu[vlan_id].gocbRef == read_ln_id
busGsePdu[vlan_id].gocbRef != read_ln_id
C

Fig. 2. GOOSE Client Model



trip[ln_id]?
LN[ln_id].SendGOOSEMessage.sqNum = 0,
LN[ln_id].SendGOOSEMessage.timeAllowedtoLive=2
RETRANSMIT_PENDING
t<=LN[ln_id].SendGOOSEMessage.timeAllowedtoLive
out[vlan_id]!
t=0
t==LN[ln_id].SendGOOSEMessage.timeAllowedtoLive
LN[ln_id].SendGOOSEMessage.stNum<INT32U
LN[ln_id].SendGOOSEMessage.stNum++
LN[ln_id].SendGOOSEMessage.stNum>=INT32U
LN[ln_id].SendGOOSEMessage.stNum=1
RETRANSMIT
LN[ln_id].SendGOOSEMessage.timeAllowedtoLive>=MAX_TTL
LN[ln_id].SendGOOSEMessage.timeAllowedtoLive<MAX_TTL
LN[ln_id].SendGOOSEMessage.timeAllowedtoLive=
LN[ln_id].SendGOOSEMessage.timeAllowedtoLive*2
busGsePdu[vlan_id]=LN[ln_id].SendGOOSEMessage
LN[ln_id].GoEna==true
LN[ln_id].SendGOOSEMessage.stNum = 1,
LN[ln_id].SendGOOSEMessage.sqNum = 0,
LN[ln_id].SendGOOSEMessage.gocbRef = ln_id,
LN[ln_id].SendGOOSEMessage.timeAllowedtoLive=2
LN[ln_id].SendGOOSEMessage.sqNum>=INT32U
LN[ln_id].SendGOOSEMessage.sqNum=1
LN[ln_id].SendGOOSEMessage.sqNum<INT32U
LN[ln_id].SendGOOSEMessage.sqNum++
NON_EXISTENT
C

Fig. 3. GOOSE Server Model

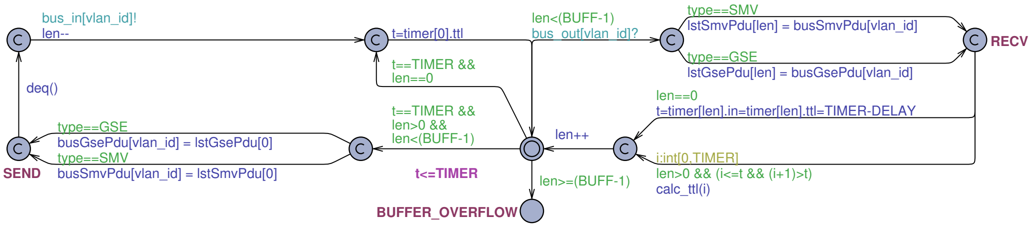


Fig. 4. Bus Model

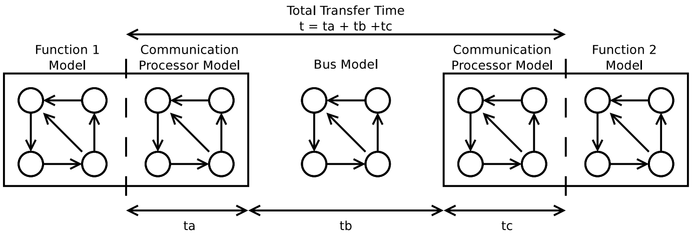


Fig. 5. Communications Delay

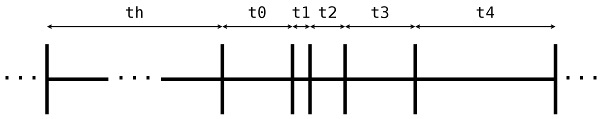


Fig. 6. Waiting time for the next re-transmission

For each Logical Node, there is only one GOOSE Server and no one or more GOOSE Clients. For example, if the GMP Controller needs to know the status of ten valves, then the Logical Node GMP Controller needs one GMP Server (sending the corresponding commands to valves) and ten GMP Clients for each valve (listening).

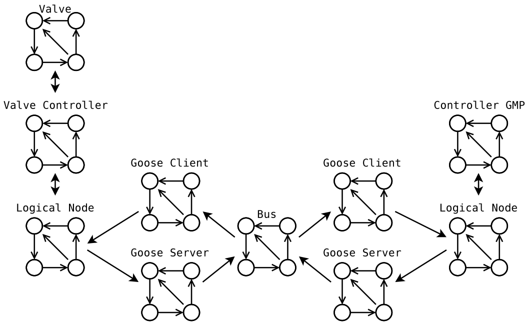


Fig. 7. Integrated Logical Nodes

IV. Simulation Results of the Controller's Specification

For all the models, the range of all variables has been limited in order to decrease the necessary computational effect, allowing to obtain results. For all the locations of the entire automata model - with exception of the ”committed” locations - it is necessary a time interval to allow the evolutions in all automaton models, from one to another location.

Concerning simulation results, the data of the file *XTR* (simulation registry) was been used to obtain the diagram of Fig. 8. In this figure, it is possible to see the retransmission messages made by the increment of the *SqNum* and *stNum* variables in the time.

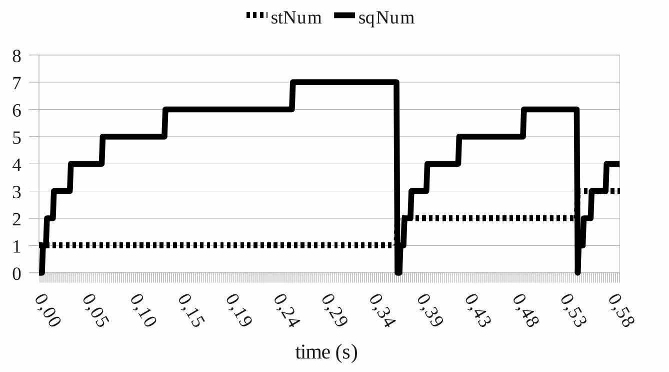


Fig. 8. Simulation Results

The simulated result is the expected one for this system. However, formal verification is also necessary in order avail the behavior of the GOOSE communications.

At this moment, formal verification has been used, only for the deadlock checking (formal description: ”A[] NOT DEADLOCK”), with DBM - Difference Bounded Matrices [17] 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 - composed by sixty-two (62) modules - and the computational available capacity is not enough to obtain the results of the full model. To deal with this problem, it will be necessary to use partial formal verification [18], [19] and/or abstraction modeling techniques [20] to handle with the full model, in order to obtain more complex formal verification results, associated to more complex behavior properties of the controller specification.

V. Conclusions and Future Work

IEC 61850 requirements were modeled in order to obtain a dependable specification for an APM system. The obtained results, concerning simulation of the obtained specification, are satisfactory and allow us to conclude that the developed specification accomplishes the requirements of the IEC 61850 standard.

The adopted formalism (timed automata) used to develop the specification of the system will allows to consider formal verification technique to validate some critical behaviors of the system. In order to accomplish there goals. The reducing of the variable size will be tested because it should be decrease the necessary processing time and computer memory used to perform the formal verification tasks. For example, *stNum* has 2^{32} bytes, but we intend to use it with 2^8 bytes because the functionality is same. Partial formal verification and/or abstraction modeling techniques will also be used in order to reduce the memory and time consuming for obtaining formal verification results.

Acknowledgments

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

[1] IEEE, “Ieee standard for communications-based train control (cbtc) performance and functional requirements,” 2004.

[2] E. S. Neumann and M. V. A. Bondada, “Automated people movers: Engineering and management in major activity centers,” *ASCE*, 1985, new York.

[3] T. Inouye and T. Kurokawa, “Automated people movers iii,” *ASCE*, 1993, new York.

[4] W. J. Sproule, M. V. A. Bondada, and E. S. Neumann, “Automated people movers iv,” *ASCE*, 1993, new York.

[5] A. F. des Sciences et Technologies de l Information et des Systemes (AFCET), “Apms toward the 21st century,” *5th Int. Conf. on Automated People Movers*, 1996.

[6] L. D. Shen, J. Huang, and F. Zhao, “Apm applications: A worldwide review,” *Annu. Meeting of Transportation Research Board*, 1996, washington, D.C.

[7] S. of Danish Engineers, *7th Int. Conf. on Automated People Movers*, 1999.

[8] “Ieee standard for communications protocol aboard trains,” 1999.

[9] J. C. Moreno, E. Laloya, and J. Navarro, “A link-layer slave device design of the mvb-tcn bus (iec 61375 and ieee 1473-t),” *IEEE Transactions on Vehicular Technology*, vol. 56 Issue: 6, 2007.

[10] T. Sullivan, “Ieee rail transit vehicle interface standards update,” *4th International Conference on Communications Based Train Control*, 2001.

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

[12] News on iec 61850 and related standards. [Online]. Available: http://blog.iec61850.com/

[13] G. Behrmann, A. David, and K. G. Larsen, “A tutorial on uppaal,” *4th International School on Formal Methods for the Design of Computer, Communication, and Software Systems (SFM-RT'04)*, 2004, lNCS 3185.

[14] J. F. F. H. Britto, “Modelo computacional do sistema aeromóvel de transportes,” Master's thesis, UFRGS, 2008.

[15] S. M. de Lima Furtado, “Análise comparativa entre o aeromóvel e outros sistemas de transporte urbanos guiados automáticos em vias exclusivas elevadas,” Master's thesis, Universidade de Brasília, 1994.

[16] Aeromovel, “Aeromovel system technical specification,” Aeromovel Brasil S.A., Tech. Rep., 1999.

[17] D. L. Dill, “Timing assumptions and verification of finite-state concurrent systems,” *LNCS 407*, pp. pp 197–212., 1989.

[18] G. J. Holzmann, “Design and validation of computer protocols,” *Prentice-Hall*, 1991.

[19] ——, “An analysis of bitstate hashing,” *Formal Methods in System Design*, pp. 13:289–307, 1998.

[20] F. Balarin, “Approximate reachability analysis of timed automata,” *17th IEEE, Real-Time Systems Symposium*, 1996.