Neural Comput & Applic
DOI 10.1007/s00521-015-2115-5

ORIGINAL ARTICLE

# Using timed automata for modeling, simulating and verifying networked systems controller's specifications

**Guilherme Kunz**[1] · **José Machado**[2] · **Eduardo Perondi**[3]

Received: 17 June 2015 / Accepted: 4 November 2015


**Abstract** The development of dependable controllers can be a very complex task. For this purpose, some synthesis and analysis modern computational techniques can be used. In this paper, simulation and formal verification analysis techniques are used in a concurrent way in order to validate formal communication requirements of generic object oriented substation event and sample value communication protocols from the IEC 61850 standard. Because these techniques are used in a complementary way, the formalism and tools used for both are the same: timed automata for modeling, and UPPAAL model checker for performing simulation and formal verification tasks. Also, we show that the use of timed automata formalism is suitable for modeling the controllers' specifications, specifying the time requirements for information exchanging taking into account networked controllers, and, as it is a non-deterministic formalism, for analyzing the plant behavior. The concepts developed in this study were successfully tested in an application in the control system of an automated people mover.

✉ José Machado
jmachado@dem.uminho.pt

Guilherme Kunz
guilherme.kunz@unioeste.br

Eduardo Perondi
eduardo.perondi@ufrgs.br

[1] Centre for Engineering and Exact Sciences, State University of Western Paraná, Foz do Iguazu, Brazil

[2] Mechanical Engineering Department, MEtRICs Research Center, University of Minho, Guimarães, Portugal

[3] Department of Mechanical Engineering, Federal University of Rio Grande do Sul, Porto Alegre, Brazil

**Keywords** IEC 61850 communication requirements · Simulation · Formal verification · Timed automata · Automated people movers

## 1 Introduction

The automated people mover (APM) is usually defined as an intense utilization, fully automated and grade-separated mass transport system. This term is generally used to describe systems that serve restrict areas, such as casinos, airports, or high-level transportation systems such as subways and trains which are served by local passenger stations [1].

About a quarter of the APM systems, function as urban trains and the remaining are private short-range transportation systems for airports, amusement parks, shopping malls and other institutions across the USA, Europe, Japan and Australia [2–7].

The APM control system controls autonomously the movement of the vehicles and supervises the correct achievement of the control instruction. For example, to avoid a collision between vehicles, it is necessary to activate the vehicle braking system in order to perform the stopping procedure in the desired deceleration curve. Therefore, the control system should be provided with continually updated information about speed, vehicles current location, data transmission delays, and the time interval necessary for the effective braking action. The correct fulfillment of these functions is guaranteed by the automated train controller (ATC) system, which, according to the 1474.1-2004-IEEE standard for communications-based train control (CBTC) performance and functional requirements [1], is organized comprising the following subsystems:

Published online: 21 November 2015

- *Automatic train protection (ATP)* This subsystem is responsible for providing protection against dangerous situations, such as train impacts, vehicle over-speed and train line obstruction, among others;
- *Automatic train operation (ATO)* performs cruising speed regulation and controls the vehicle and station doors. It is also responsible for the vehicle stopping procedure at the stations, among other similar operations that are assigned to the train drivers in a non-automatic means of transport;
- *Automatic train supervision (ATS)* supervises and makes corrections to each train operation in order to assure that the travel is conformed to the schedule.

According to the 1474.1-2004-IEEE standard for CBTC performance and functional requirements [1], an ATC must comprise an ATP system and it can include ATO and/or ATS subsystems. Aiming to assure the information interchange among these subsystems, functional and communication performance requirements concerning the APM control system should be taken into account. Among the main characteristics of CBTC, the following are of special importance to the current context:

- Precise information about the position of each vehicle obtained through sensors not belonging to the train line;
- Constant information interchanging among the vehicle and other systems that are not directly associated with its maneuver;
- Continuous monitoring of the vehicle control states transmitted to the ATP.

IEC 61850 provides typical values, such as, for instance, the resolution of speed measurement, the range and reaction time of the communication equipment used in the CBTC, and others. ATC is defined by IEEE Std 1473-1999-IEEE standard for communications protocol aboard trains [8], which establishes the communication protocol used for inter-car and intra-car vital messages. Two protocols are indicated from an application point of view: 1473-L and 1473-T.

The 1473-L (LonWorks) protocol has some characteristics which are similar to EIA 709.1-1998 [9] and from EAI 709.3-1998 [10]. This protocol is used to create the local vehicle bus (LVB) which integrates on-board devices and the local sensor bus (LSB) which connects the sensors.

The 1473-T (train communication network-TCN) protocol presents some characteristics of IEC 61375-1-1999 [11]. It is used in time deterministic applications and is subdivided into two protocols: wire train bus (WTB), which interconnects the trains operating units; and the multifunction vehicle bus (MVB), which interconnects embedded devices to the vehicles operating system. One main advantage of this arrangement is the facility in using devices from several suppliers [12].

Some limitations of the advanced train control system (ATCS) concerning types 1473-L and 1473-T were identified: the absence of integration protocols, of internet protocol (IP) interfaces discarding Ethernet data interchanging and the lack of support for different video broadcast demands [13].

Protection and vehicle control are usually centralized and designed to be implemented with wired circuits [14]. Even though they normally present a plain design [15], there are typically hard complications associated with its installation and maintenance. In the cases where constant revisions and improvements are necessary, it is advantageous to choose open architectures and a basic communication scheme. These general conceptions are broadly taken into account in the conception of the IEC 1850, which is a communication standard for modern electrical substations established on the concept of intelligent electronic devices (IED) which are replace the obsolete protection relays. In a general way, its use implies the following advantages [14].

Extensive system configuration language (SCL) for the entire life cycle, from the design to the manufacturing, including operation and maintenance;

- Small cost and time necessary to execute the physical implementation;
- Augmented ability for supervision and protection;
- Individual infrastructure;
- Individual functionality;
- Interoperability;
- Multi-vendor support;
- Small wiring necessity.

To reach these features, IEC 61850 establishes some requirements, such as real-time processing and distributed object orientation software design. Also, IEC 61850 provides a standardized procedure for substations interoperation based on the requirement of reporting specifications for data structure, functional features and data and devices nomenclature, as well as procedures for operational features, which determine requirements to integrate with IEC 61499 function architecture, to operate with the control devices applications, and how these features must be verified in the system compliance analysis procedure [16–18].

Concerning the for APM control protocol, the IEC 61850 was based on the ASN.1 [19], which is in accordance with requirements of the IEEE 1488 (IEEE trial-use standard for message set template for intelligent systems of transportation) [20]. The IEC 61850 used three classes of messages:

- Generic object oriented substation event (GOOSE), which is applied to real-time, high priority, asynchronous, unsolicited and heartbeat messages;
- Sample value (SMV), which is used for real-time data exchanging between machines, allowing signal processing in distributed equipment;
- Manufacturing messaging specification (MMS), which is used for non-real-time supervision communications and remote configuration.

The IEC 61850 divides the controller into three levels associated with the functional hierarchy of each component:

- *Process level* It comprises all the functions that show straight interface with the control process. The states are identified by analog indications. The signals are usually transferred by cabling in the form of electrical energy by using supplementary switches;
- *Bay level* It includes all the functions that operate the equipment of the process. These functions can, for instance, switch or turn on and off the electromechanical components.
- *Station level* The remainder characteristics are placed in this level and are distributed into two sets, according to their relation with the data interchange:
  - *Processes* Functions that manipulate information from two or more levels of bay;
  - *Interfaces* Data exchanging exchange between remote stations or supervision systems.

Those protocols, used in the mentioned standard (IEC 61850), are object-oriented and can be characterized by an open architecture concerning distributed control. The data types used are:

- *Physical device* Object accessed by using a network address;
- *Logical device* It comprises a set of logical nodes arranged in an IED—intelligent electronic device;
- *Logical nodes* It represents the real functions present in the system;
- *Data and attributes* It refers to parameters associated with logical nodes, as, for instance, velocity or arrangement.

Besides the advantages stated in [14], the IEC 61850 has characteristics that favor the integration of APM controller with power systems, allowing relating the APM general performance to minimizing energy consumption, improving the overall system efficiency. In this work, it is described how simulation and formal verification may be used to perform the validation of formal communication requirements of GOOSE and SMV communication protocols from IEC 61850 applied to an APM system performing a CBTC, resulting in a proposal for the expansion of the IEC 61850 standard to deal with APM systems. This extension is based on the following features that are common to both APM and power generation and distribution systems:

- Critical system (in the safety context);
- Complex systems;
- Long distances between equipments;
- Hybrid system (continuous/discrete).

To enable the use of IEC 61850 in APM design, safety conditions related to the system operation are clearly critical. These requirements are described and characterized by the aforementioned International Standards, including all the significant characteristics of the system controller.

In some practical applications, the use of the entire IEC 61850 requirements set could be impracticable. One appropriate alternative is to develop a global model that encloses all the data changing protocols defined by the IEC 61850, assuring that all the time delays are being considered. Therefore, it is necessary to choose appropriate computational resources to perform simulation and formal verification in order to assure the correct execution of all required behaviors defined by IEC 61850. In this study (as in [21]), both simulation and formal verification are used in a complementary way.

The modeling and simulation of the IEC 61850 communication requirements adapted to APM systems were addressed in [22, 23]. In this paper, besides the modeling of IEC 61850 communication requirements and simulation of the APM case, it is presented a strategy for executing the formal verification of the results, by establishing a strategy for supporting the future development of a standard for the design of APM control systems based on the IEC 61850 communication requirements. Hence, in the IEC 61850 standard protocols context, GOOSE and SMV procedures will be studied through simulation and formal verification techniques.

A set of different formalisms can be adopted for modeling real-time systems [24–26]. Timed automata formalism was adopted because the analysis of the proposed system needs to take into account real-time behavior; also, timed automata are the formalism used by UPPAAL model checker [27], regular free software available to perform the formal verification and simulation of real-time systems.

This paper is organized as follows: Sect. 2 presents the IEC 61850 requirements modeling, while Sect. 3 is devoted to the simulation analysis technique and respective results. In Sect. 4, the formal verification tasks and corresponding results are presented and discussed. Finally, in Sect. 5, the conclusions and perspectives for future work are presented.

## 2 Modeling of the IEC 61850 requirements

The main features of the peer-to-peer GOOSE and SMV messages and the full system (plant and controller) closed loop modeling via timed finite automaton modules are presented in this section.

The control system of the vehicle is frequently centralized; nevertheless, pointing toward a solution based on the IEC 61850 [14], the models of the subsystems are developed based on distributed control architecture.

Hence, in the proposed models, real-time procedures individually devoted to each device are taken into account. The control elements are linked to a communication channel that exchanges data with other channels to the user interfacing processing unit, thus reducing the individual processing effort [28, 29]. Generally, the choice of applying a distributed strategy is driven by cost reduction and by system flexibility purposes. The case described in Fig. 1a represents a theoretical overspeed protection system communicates with a position sensor by means of SMV Messages and with a valve controller via GOOSE messages. Both the GOOSE and SMV are real-time messages, and the units are connected through a communication bus embedded in the vehicle. The modeling of the plant, system devices and controllers are obtained through timed automata formalism, by using a modular approach, allowing the simulation and formal verification is performed using UPPAAL model checker.

There is one logical node associated with each function or equipment present in the overspeed protection system. For instance, if the overspeed protection scheme of the system presented in Fig. 1b needs to change the safety valve status, the GOOSE Producer will send a data packet command in a GOOSE Message to the valve controller logical node. This message, divided in two virtual local area networks (VLANs), one to GOOSE messages and another to SMV messages, will be sent to the Ethernet bus and, taking into account the corresponding delay, the message will be received by the safety valve controller GOOSE consumer. Therefore, the packet information will be verified by the GOOSE consumer and a message will be sent to the controller of the safety valve. The state of the valve will be changed within the time required to change the safety valve status.

There is only one GOOSE producer and a flexible amount of GOOSE consumers associated with a specific logical node. This configuration is also valid for the same configuration used in the SMV protocol. For example, if the overspeed protection controller has to verify the state of a set of safety valves, then the overspeed protection logical node needs only one goose producer, which will send the respective instructions to the valves, and one GOOSE consumers for each valve (necessary to receive the instructions).

An environmental model that comprises the GOOSE and SMV protocols is necessary to execute the conformance checking. Regarding the overspeed protection system, the communication between three devices must be taken into account: the position sensor (TDST logical node), overspeed control system (POSP logical node) and valve controller (KVLV logical node). There is a logical device with two logical nodes associated with each element

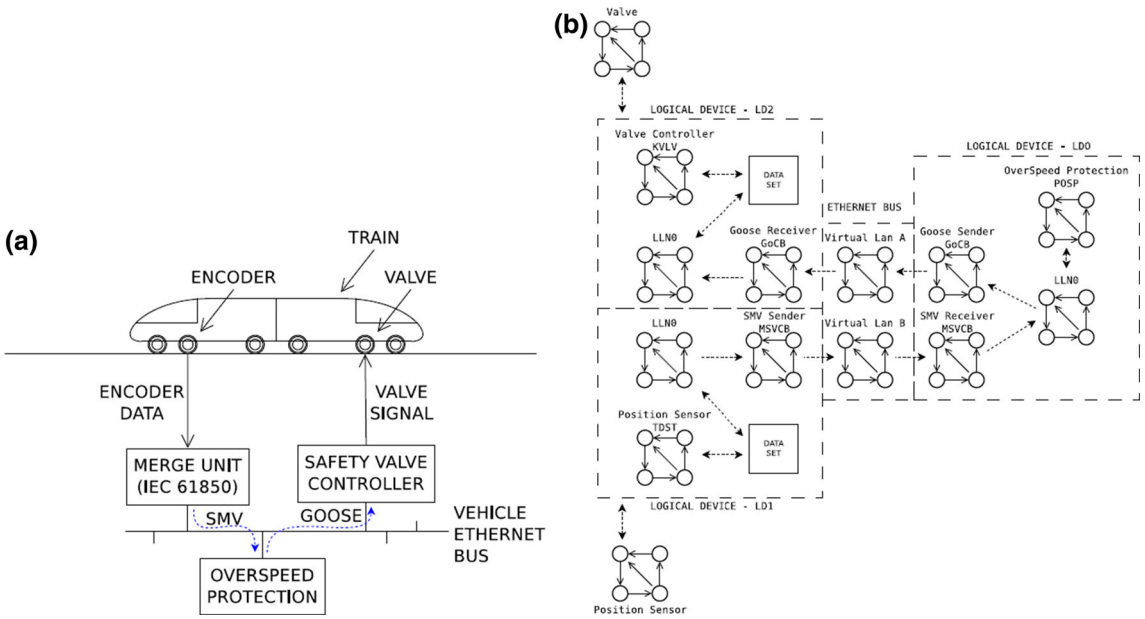


**Fig. 1** Automata network. **a** Overspeed protection, **b** integrated logical nodes

of the equipment (LLN0 node and the logical node of the equipment). The LLN0 node periodically sends the status (GOOSE) and an acquired value (SMV) of given equipment (for instance, a displacement transducer that measures the vehicle position detecting).

As GOOSE and SMV protocol messages receive distinct priorities, for each protocol (models BUS-VLAN0 and BUS-VLAN1) that represents diverse virtual LAN, a corresponding bus model must be introduced.

In the environment developed, the models used to perform simulation and formal verification are the following: one sensor, two buses (VLANs), one valve, six logical nodes (KVLV, POSP, TDST and three LLN0), two SMV (sender and receiver—used for data interchange between logical nodes), and two GOOSEs (sender and receiver). To execute the compliance testing it was necessary to add four specific modules to the ones that represent the control functions. As a result, 14 modules were used in the simulation and in the formal verification procedures. The relationship among the 14 functional models is presented in Fig. 1b, where the arrows indicate the direction of the data.

The TDST$LLN0$MSVCB model sends data from the TDST logical node by using SMV messages to the POSP logical node through his POSP$LLN0$GoCB SMV receiver, while the data of the POSP logical node are sent to the KVLV logical node via GOOSE messages by the POSP$LLN0$GoCB model.

The types of models considered are the following: logical nodes, check messages observers, GOOSE messages, communication bus, and sampled valued messages.

### 2.1 Logical nodes

The IEC 61850 standard establishes that a publish/subscribe architecture should be used. In this strategy, differently from the client/server one, where a client requests the data interchange from a given server by the request/reply methods, the method publisher (sender or producer) sends the messages without specifying a precise destination. The subscriber method (receiver or consumer) only takes into account the suitable messages. Therefore, periodically, the TDST$LLN0$GoCB and POSP$LLN0$MSVCB models send multicast messages and POSP and KVLV accept the following instructions sent by GOOSE and SMV subscribers:

- gse_updt. The GOOSE signal status was changed;
- gse_stop. Publisher cancels the data sending;
- gse_fail. The publisher was not able to send data packets within a predetermined time interval, generally caused by publishing or network error;
- smv_fail. The SMV publisher message was not able to send data packets within a predetermined time interval;
- smv_updt. The SMV signal state is unchangeable.

### 2.2 GOOSE messages

The GOOSE protocol model considers the following features:

- All the messages are asynchronous and unsolicited;
- An Ethernet layer encapsulates the GOOSE protocol. The messages have no connections; therefore, the model does not have confirmation from receivers about the connection dependability;
- GOOSE protocol messages are multicast. Each VLAN (virtual LAN) must have a bus model which connect one-to-many or many-to-many devices;
- When the validation from receivers is put into practice, the retransmission increases the probability of success in the reception.

The three fundamental states of the GOOSE Producer (Fig. 2a) are RETRANSMIT, RETRANSMIT-PENDING and NON-EXISTENT. The logical node is structured to transmit GOOSE messages (GoEna == true). StNum is used to describe how many times the device status has changed. SqNum is set to zero when the producer transmits the first message, and it is incremented in each broadcasting execution. SqNum rollover to zero when StNum is updated. In the first data transmission, StNum is set to 1 and timeAllowedtoLive is set to 2. These variables are associated with SendGooseMessage structure. The delay for the next data communication (timeAllowedtoLive) is set to an asynchronous changing state, being n incremented by 1 until reaching the heartbeat time limit (set as 1024 ms). The time delay for the subsequent data interchanging is related to an asynchronous status variation, from $t = 21 + i$ ms from $i = 0$ to $i = 1023$ (referent to the heartbeat interval).

According to VLAN, the GOOSE Producer sends messages to other devices by copying data to the bus model which, after the reception of the synchronization channel, makes a time registry and copies the information to a queue. Subsequently, the bus model copies the records to the busGsePdu structure and eliminates it from the queue, also opening a transmission channel to all GOOSE consumers connected to the VLAN. The bus model is equivalent to the SMV and GOOSE messages; nevertheless, some distinctions occur in the queue since the respective data packets present specific structures.

The GOOSE consumer (Fig. 2b) receives the data communication by means of the broadcast channel and replicates the busGsePdu in the local memory, thus validating the importance of the information. This procedure is predefined and, when the information is not significant, it is rejected and the GOOSE consumer returns to the reception state. If the incoming message is significant, the GOOSE consumer model makes a request to the logical node

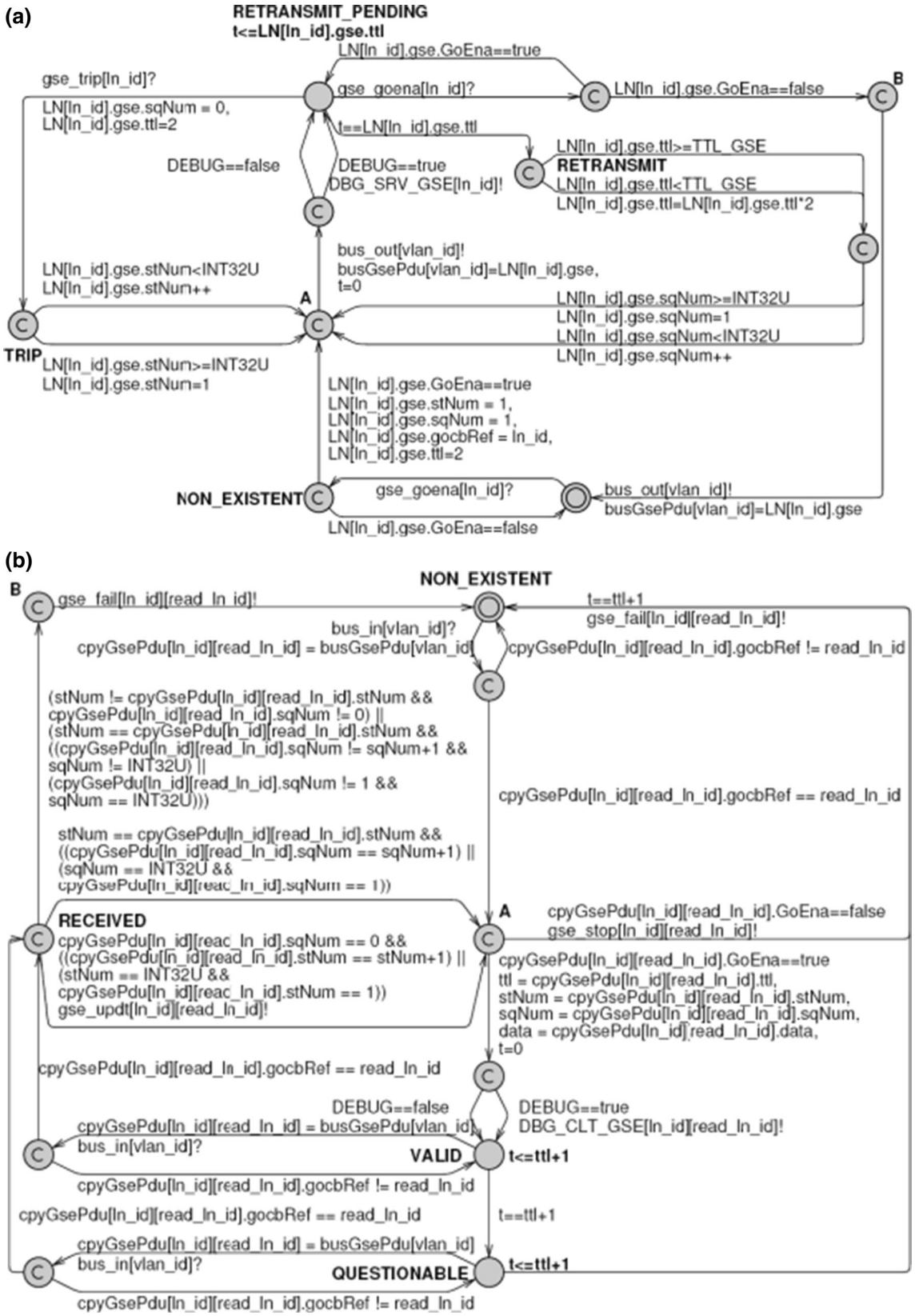


**Fig. 2** GOOSE models. **a** GOOSE producer model, **b** GOOSE consumer model

controller, executes the required functions and returns to the reception state.

### 2.3 Sampled value messages

The analysis of Fig. 3a allows one to verify that the Sampled Value arriving messages model initiates by waiting for the first data packet sent to his logical node, described by ln_id. Once received this data package, previously resetting the timer, the inspection of both reception interval (smpRate) and message counter (smpCnt) takes place. Whether there is a lost packet verified by means of the smpCnt counter or the message is delayed ($t >$ smpRate), an error communication is directed to the logical node through the smv_fail command.

Figure 3b shows the sampled value messaging model. This template sends, the updated value of the model TDST, regardless of the message received by the other logical nodes, in accordance with a configured rate (TTL_SMV). The data sending is done through the MULTICAST method, because the messages are restricted to the logical nodes connected in the same VLAN.

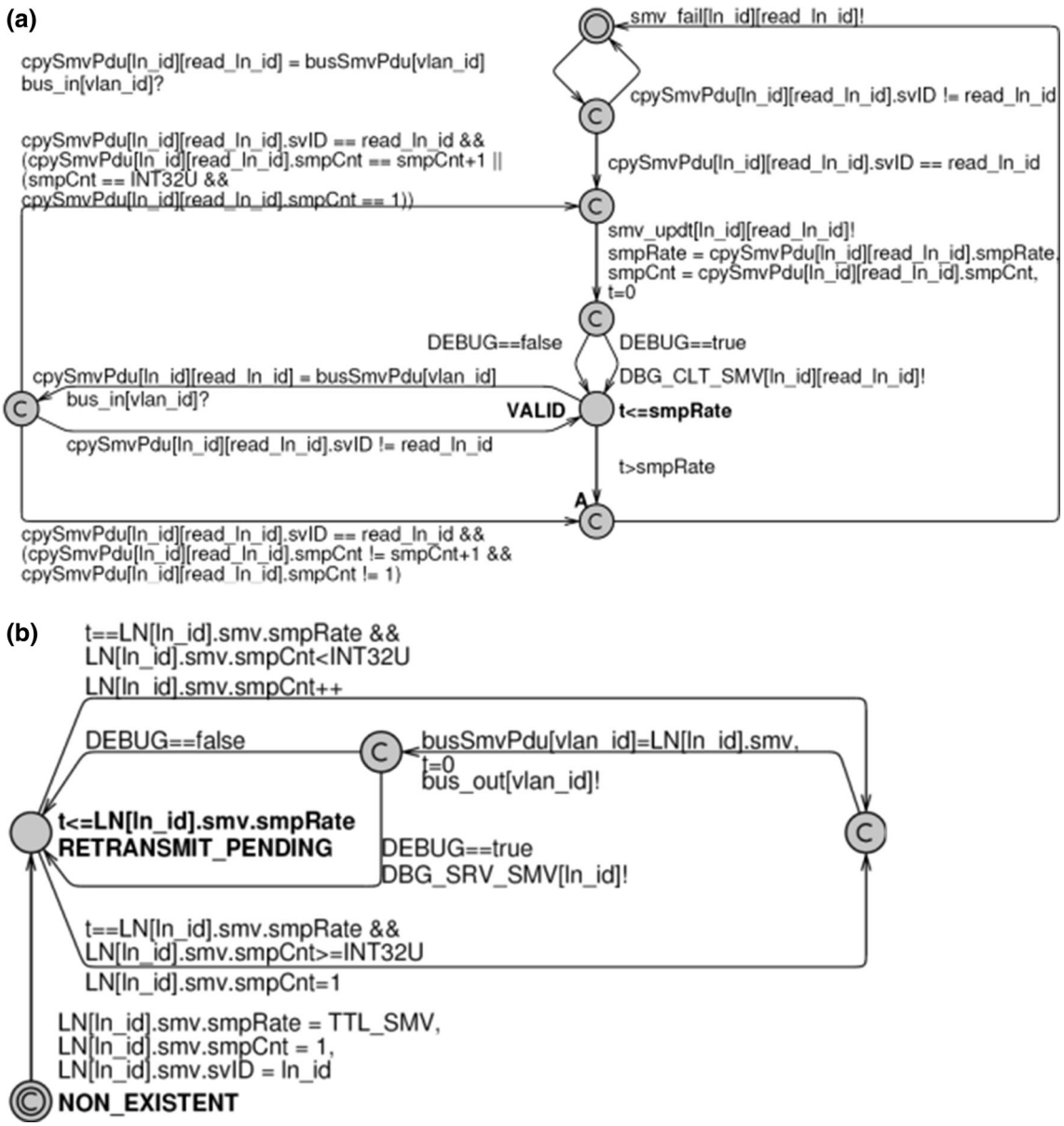


**Fig. 3** Sampled value models. **a** Sampled value consumer model, **b** sampled value producer model

### 2.4 Communication bus and check messages observer

The bus model presented in Fig. 4a comprises a first in, first out (FIFO) queue with delay. All of the delays associated with the frames flow introduced by the communication processors and the network are assigned in the bus model.

The definition of the type of messages addressed by the bus (GOOSE or SMV) is made previously. When network messages are received, independently of its arrival time, this template adds a data packet to the queue, waiting until the delay time is over to eliminate the message from the queue and send it back to the network.

The template presented in Fig. 4b is used to execute the formal verification checking by means of the following objects:

- CHECK_POSP_KV LV_GSE: besides the package integrity verification, this object is used to check the delay associated with the time required for exchanging GOOSE messages between KVLV and POSP logical nodes;
- CHECK_TDST_POSO_SMV: besides the package integrity verification, this object is used to check the delay associated with the time required for exchanging sampled value messages between POSP and TDST logical nodes.

After sending a sampled value or GOOSE message, both the transmission time and the message are replicated in a FIFO queue. When the data packet reaches its destination, package integrity is verified by comparing the replicated message and checked if the sending time interval did not surpass the predetermined delay, in accordance with IEC 61850 functional constraints.

## 3 Simulation of the controller specification

For the entire set of models, the amount of variables was limited in order to reduce the computational effort and, as well as possible, a specific time interval was defined to the transition between locations. The scheme presented in Fig. 5 was obtained by means of the analysis performed on the results furnished by the UPPAAL simulation registry (XTR file type).

The variable stNum increases one unit at each data interchanging, while SqNum increases one unit when it reaches the counter superior value (and, therefore, StNum restarts the counting process) or the device status changes. Figure 5 presents the simulations results for SMV, which shows that the value of smpCnt increases one unit in each packet sending and that the data are modified in a non-deterministic way, assuming the randomness behavior of a generic sampled data value.

The simulations results indicate that the system should present an appropriate behavior, being consistent with the preprogrammed requirements. Nevertheless, it is also important to execute a formal verification in order to evaluate the reliability performance of the GOOSE and SMV communications procedures.

## 4 Formal verification of the controller specification

Regarding formal verification, some points were identified as being required to verify some significant behavior related to IEC 61850 standard-based systems. In this paper, for use on the UPPAAL model checker, these behaviors have been summarized in the following three steps: (1) describing the proposed behavior by means of natural language; (2) formalizing the behavior (natural language) by means of a subset of timed computation tree logic, which is the temporal logic supported by UPPAAL; and finally, (3) decoding the behavior characteristics for the input UPPAAL language, regarding queries verification. The properties of steps 1 and 3 are presented in Table 1. The verification was achieved based on the given information [30]. After about 8 min, the entire set of properties was verified using a PC Intel ® Core™2 Duo CPU 2.10 GHz (4 Gb RAM) and UPPAAL release 4.1.6 set as first search order DBM—difference bounded matrices state space representation.

## 5 Conclusions

Based on the previous discussion, it can be concluded that the proposed specification for the development of dependable APM control systems based in the IEC 61850 has been successfully modeled, simulated and formally verified. The timed automata formalism used to develop the specification structure could be suitably adapted to the application in the simulation and formal verification techniques, and some critical characteristics of the system studied have been validated by using both techniques (simulation and formal verification) in a complementary strategy. We could also conclude that the UPPAAL software tool is appropriate for performing simulation and formal verification tasks, and the furnished results show that the proposed approach is adequate for improving the reliability of the proposed specification of networked controllers. Moreover, the results obtained through simulation and formal verification showed that the proposed specification is in accordance with the IEC 61850 standard and with the respective communication requirements.

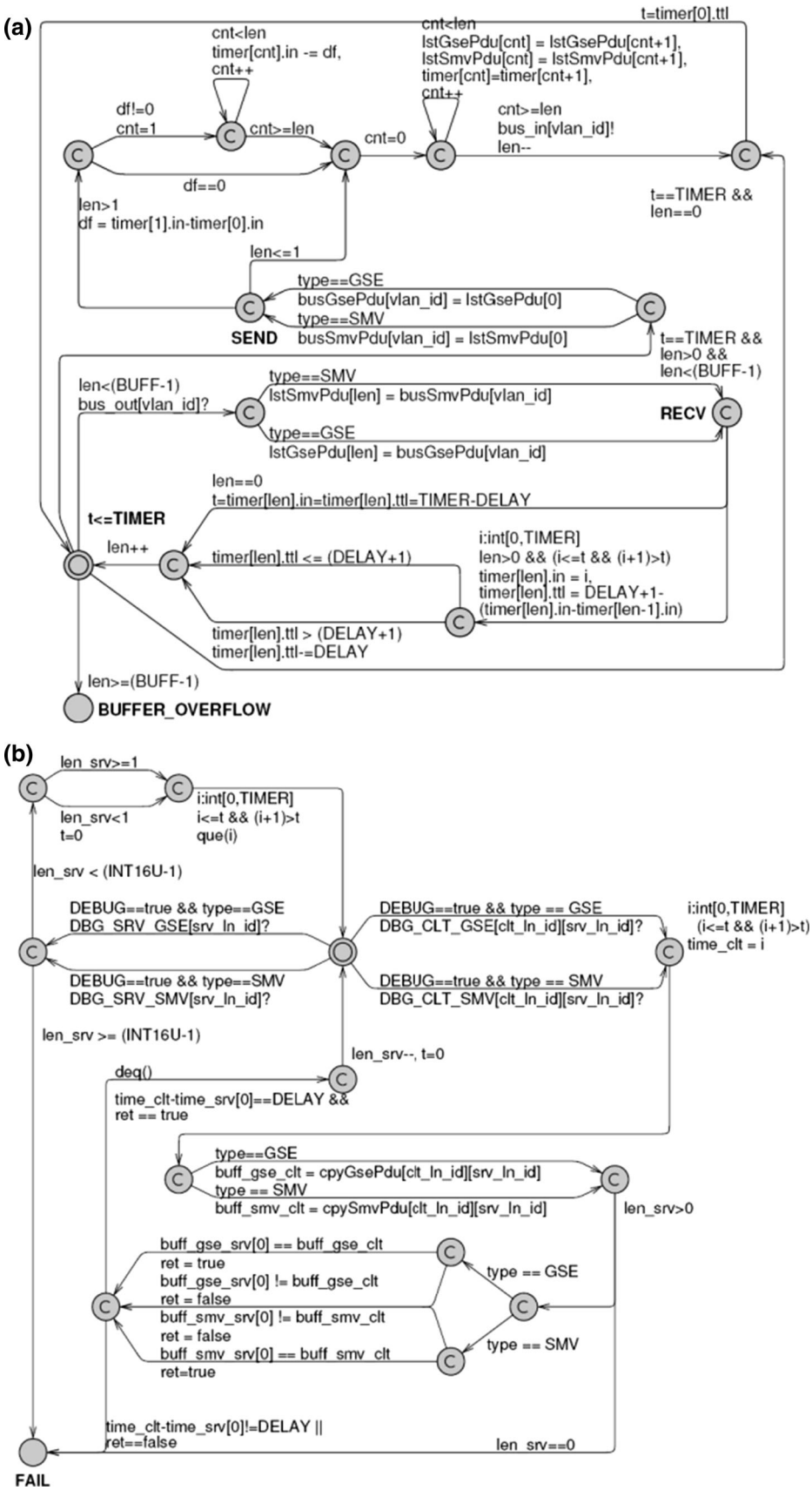


**Fig. 4** Simulation of the controller specification: **a** bus model, **b** check messages observer model

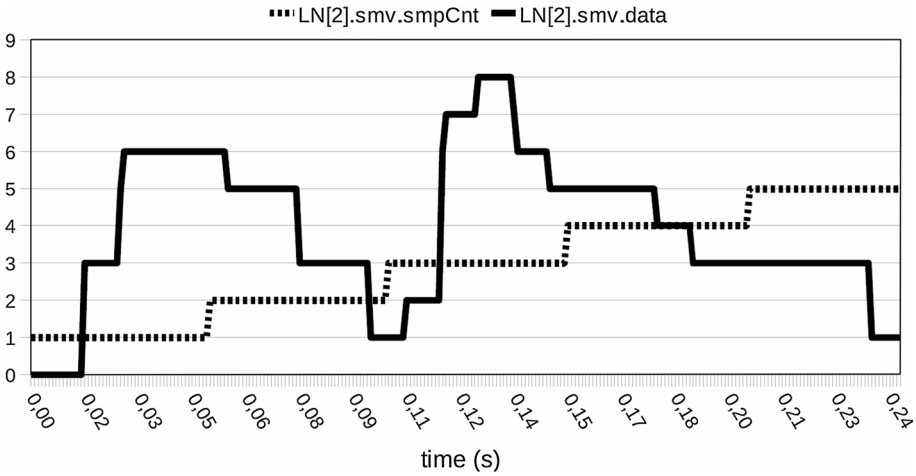


**Fig. 5** Simulation results—SMV

**Table 1** Behavior properties of the communication system

| Informal description | Formal description |
|---|---|
| Verify for GSE and GOOSE that stNum and sqNum present initial value equal to 1 on device wake-up procedure | A[] (POSP_LLN0_GoCB.A and POSP_LLN0_GoCB.t == 0) imply LN[LN_ID[2]].gse.sqNum == 1 and LN[LN_ID[2]].gse.stNum == 1 |
| Verify if GoEna enables and disables the sending of GOOSE messages | A[] (POSP_LLN0_GoCB.B) imply LN[LN_ID[2]].gse.GoEna == false |
| | A[] (POSP_LLN0_GoCB.RETRANSMIT) imply LN[LN_ID[2]].gse.GoEna == true |
| Confirm that GOOSE messages are published periodically, that sqNum increases and that stNum was not modified. When there is any changing of a data value in the GSE dataset, GOOSE messages should be published as specified, stNum increases and transmits GOOSE message with the new values and verifies if the sent data packet was received and whether the data contains the new information | A[] (KVLV_LLN0_GoCB.A and KVLV_LLN0_GoCB.stNum !=cpyGsePdu[1][2].stNum) imply (cpyGsePdu[1][2].stNum == KVLV_LLN0_GoCB.stNum + 1 and cpyGsePdu[1][2].sqNum == 0) or (cpyGsePdu[1][2].stNum == 1 and KVLV_LLN0_GoCB.stNum == INT32U and cpyGsePdu[1][2].sqNum == 0) and (cpyGsePdu[1][2].data !=KVLV_LLN0_GoCB.data) |
| If the GOOSE data changes, stNum increases and sqNum is set equal to zero | A[] (KVLV_LLN0_GoCB.A and cpyGsePdu[1][2].data !=KVLV_LLN0_GoCB.data) imply cpyGsePdu[1][2].sqNum == 0 and (cpyGsePdu[1][2].stNum == KVLV_LLN0_GoCB.stNum + 1 or cpyGsePdu[1][2].stNum == 1 and KVLV_LLN0_GoCB.stNum == INT32U) |
| Check for delayed, double, missing and out of order GSE messages | A[] (KVLV_LLN0_GoCB.VALID) imply KVLV_LLN0_GoCB.t <= KVLV_LLN0_GoCB.ttl |
| | A[] not KVLV_LLN0_GoCB.B |
| | A[] not KVLV_LLN0_GoCB.QUESTIONABLE |
| Check for out of order, double, missing, delayed and SMV messages | A[] not POSP_LLN0_MSVCB.A |
| Verify buffer overflow into Bus | A[] not BUS_VLAN1.BUFFER_OVERFLOW |
| | A[] not BUS_VLAN0.BUFFER_OVERFLOW |
| Send SMV and GSE messages with new data and check if they were received | A[] not CHECK_POSP_KVLV_GSE.FAIL |
| | A[] not CHECK_TDST_POSP_GSE.FAIL |
| The system can never present the deadlock State | A[] not deadlock |

The theoretical application of the proposed strategy to critical APM systems showed that it is promising to perform practical application of the analytical and computational models and concepts proposed in this work.

In future work, it will be performed an integrated analysis of the IEC 61850 standard specifications joined with the APM protection system (ATP) by using timed automata and the application of formal verification

technique in order to validate and propose an expanded instance of the IEC 61850 standard suitable to be applied into APM systems.

## References


1. (1999) IEEE standard for communications-based train control (CBTC) performance and functional requirements. doi:10.1109/IEEESTD.1999.90611
2. Neumann ES, Bondada MVA (1985) Automated people movers: engineering and management in major activity centers. ASCE, New York
3. Inouye T, Kurokawa T (1993) Automated people movers III. ASCE, New York
4. Sproule WJ, Bondada MVA, Neumann ES (1993) Automated people movers IV. ASCE, New York
5. AFCET (1996) APMS toward the 21st century, Technical Report. Association Française des Sciences et Technologies de l' Information et des Systemes, Paris
6. Shen LD, Huang J, Zhao F (1996) APM applications: a worldwide review. Annual Transportation Research Record, Academy of Science, Washington, DC
7. (1999) APMs in Urban Development. In: 7th International conference on automated people movers. Technical report, Society of Danish Engineers
8. (2011) IEEE standard for communications protocol aboard passenger trains. doi:10.1109/IEEESTD.2011.5724313
9. Electronic Industry Association, EIA 907.1 (1998) Control network protocol specification. Arlington, VA
10. Consumer Technology Association Standards Groups (1999) Free-topology twisted-pair channel specification. ANSI
11. Institute of Electrical and Electronics Engineers (1999) IEEE standard for rail transit vehicle event recorders. IEEE Standard 1482.1-1999, Piscataway, New Jersey
12. Moreno JC, Laloya E, Navarro J (2007) A link-layer slave device design of the mvb-tcn bus (IEC 61375 and IEEE 1473-t). IEEE Trans Veh Technol 56(6):3457–3468
13. Sullivan T IEEE rail transit vehicle interface standards update. In: 4th International conference on communications based train control
14. Hewings D (2008) Introduction of integrated protection and control to railway electrification systems. In: Proceedings of IET 9th international conference on developments in power system protection DPSP 2008, pp 6873
15. Gao S, Dong H, Ning B, Chen Y, Sun X (2015) Adaptive fault-tolerant automatic train operation using RBF neural networks. Neural Comput Appl 26:141–149. doi:10.1007/s00521-014-1705-y
16. Zhabelova G, Vyatkin V (2012) Multiagent smart grid automation architecture based on IEC 61850/61499 intelligent logical nodes. IEEE Trans Ind Electron 59(5):2351–2362
17. Timbus A, Larsson M, Yuen C (2009) Active management of distributed energy resources using standardized communications and modern information technologies. IEEE Trans Ind Electron 56(10):4029–4037
18. Higgins N, Vyatkin V, Nair NKC, Schwarz K (2011) Distributed power system automation with IEC 61850, IEC 61499, and intelligent control. IEEE Trans Syst Man Cybern Part C Appl Rev 41(1):81–92
19. (2002) Information technology—abstract syntax notation one (ASN.1): specification of basic notation
20. (2000) IEEE trial-use standard for message set template for intelligent transportation systems
21. Machado J, Seabra E, Campos JC, Soares F, Leão CP (2011) Safe controllers design for industrial automation systems. Comput Ind Eng 60(4):635–653
22. Kunz G, Perondi E, Machado JM (2011) A dependable automated people mover system modeled and verified using timed automata: a case study. ABCM Symp Ser Mechatron 5:742-750
23. Kunz G, Perondi E, Machado J (2011) Modeling and simulating the controller behavior of an automated people mover using IEC 61850 communication requirements. In: 2011 9th IEEE International Conference on Industrial Informatics (INDIN). pp 603–608. doi:10.1109/INDIN.2011.6034947
24. Chen L, Shan Z, Tang T, Liu H (2011) Performance analysis and verification of safety communication protocol in train control system. Comput Stand Interfaces 33(5):505–518
25. Zhang Y, Tang T, Li K, Mera J, Zhu L, Zhao L, Xu T (2011) Formal verification of safety protocol in train control system. Sci China Technol 54(11):3078–3090
26. Lee J-H, Hwang J-G, Shin D, Lee K-M, Kim S-U (2009) Development of verification and conformance testing tools for a railway signaling communication protocol. Comput Stand Interfaces 31(2):362–371
27. Behrmann G, David A, Larsen KG A tutorial on uppaal. In: 4th international school on formal methods for the design of computer, communication, and software systems (SFM-RT'04), LNCS 3185
28. Lee J-D, Jung J-I, Lee J-H, Hwang J-G, Hwang J-H, Kim S-U (2007) Verification and conformance test generation of communication protocol for railway signaling systems. Comput Stand Interfaces 29(2):143–151
29. Lee C-H (2005) Evaluation of the maximum potential rise in Taipei rail transit systems. IEEE Trans Power Deliv 20(2):1379–1384. doi:10.1109/TPWRD.2004.833902
30. (2005) IEC 61850-10 communication networks and systems in substations—conformance testing