A Train Control System Case Study
in Model-Based Real Time System Design

Armin Zimmermann and Günter Hommel

Real-Time Systems and Robotics Group
Technische Universität Berlin
Einsteinufer 17, D-10587 Berlin, Germany
E-mail: azi | hommel @cs.tu-berlin.de

Abstract

The future European Train Control System (ETCS) will be based on mobile communication and overcome fixed blocks in order to increase track utilization and interoperability throughout Europe. Data processing on board the train and in radio block centers as well as the radio communication link are crucial factors for the safe and efficient operation. Their real-time behavior under inevitable link failures needs to be modeled and evaluated. The paper presents a first simplified model of communication failure and recover behavior as well as safety-critical data exchange. Performance evaluation of the stochastic Petri net model shows that the official quality of service specifications may lead to a bad utilization.

1. Introduction

Train control is an important part of the railway operations management system. Traditionally it connects the fixed signaling infrastructure with the trains. With the European Union ERTMS/ETCS project (European Rail Traffic Management System/European Train Control System), a standardised European train control system is designed, which will gradually replace the great number of different train control systems in use today. It will allow trains to cross borders without the need to change locomotive or driver, as it is still necessary today. The system forms the cornerstone of a common system for train control and traffic management.

At the final stage of ETCS implementation throughout Europe, more or less all train control infrastructure will be either on-board the trains or distributed in control centers. There is no need for optical signals, wheel counters, or a fixed arrangement of track parts into blocks. Trains and control centers are connected by mobile communication links. The safety of passengers depends on the communication system reliability. Real-time communication and information processing play a major role for the implementation of ETCS. The case study presented in this paper is thus a truly distributed real-time system.

The importance of quality of service parameters for the communication and specification of the real-time behavior of subsystems has been addressed in the specifications of ETCS (see e.g. [10, 11]). The requirements are however not very detailed, e.g. no distributions are considered, but only probabilities of meeting certain deadlines. While it is important to specify subsystem characteristics, the real-time behavior of the system as a whole can only be assessed by looking at their interaction. This paper goes a first step into that direction by evaluating one safety-critical communication structure together with its failure behavior.

In addition to offer interoperability between the different European railroad companies, another major goal is to increase track utilization with higher throughput of high-speed trains. It is obvious that dropping the standard block synchronization of trains and migrating to a virtual block system has the potential of allowing closer distances between trains. However, we show that the anticipation of driving in brake distance behind another train cannot be reached with ETCS under worst-case assumptions.

The mentioned evaluations can only be done using some kind of model, independent of whether it is a simulation program or based on a formal modeling technique. In this paper variants of stochastic Petri nets [1, 5] are used to describe the functional and timing behavior of ETCS train communication.

Petri nets [22] and their stochastic timed extensions have proven to be a useful formalism for real-time systems. They are considered to describe discrete event systems in a concise and appropriate way. An additional advantage is the

0-7695-1926-1/03/$17.00 (C) 2003 IEEE

availability of many different analysis and simulation techniques as well as software tools. However, the drawbacks of the different evaluation techniques appear for the presented example as well. Petri nets have been used in the context of real-time systems many times, see e.g. [3, 15, 19]. A comparison between continuous and discrete time stochastic Petri nets for real-time systems was given in a previous paper [27].

Most of the work in the area of train control systems deals with *qualitative* aspects like validation of correctness, absence of forbidden safety-critical states etc. Yet in a real-time system like a distributed communication-based train control system, critical safety questions can only be answered when also *quantitative* aspects are considered and evaluated. Failures and other external influences on the model require stochastic model values, but fixed values for deadlines or known processing times are equally important. Modeling and evaluation techniques need to support both in order to be applicable in this area.

In [17, 20] the ETCS communication structure is modeled with colored Petri nets. The model is used for a division of the system into modules, visualization of the functional behavior, and a check of different scenarios.

A verification of the radio-based signaling system together with a case study of a rail/street crossing is carried out in [8]. Live sequence charts are used to model the system, which is analyzed with the STATEMATE software tool.

The ETCS radio block center is formally modeled and validated in [4]. Message sequence charts are used to model and check different scenarios.

The ETCS train traffic is compared with today's standard train control operations in Germany in a simulation study of Deutsche Bahn (German railways company) [23]. Using a proprietary simulation program, the movement of a set of trains through an example line is simulated. The results say that ETCS operation in its final stage will increase track utilization by about 30% for the example. However, the communication is not modeled, and failures are not taken into account.

Modeling and evaluation of complex systems is only feasible with the support of appropriate software tools. Design, analysis and simulation of the models presented in the paper is done using the tool TimeNET [26]). It offers non-Markovian uncolored and colored Petri net modeling and numerical analysis as well as simulation algorithms.

The remainder of the paper is organized as follows: After a brief overview of the ETCS communication architecture, a model for the communication system failures is developed an analyzed in Section 3. A condensed model is derived from the results in the sequel. Section 5 describes how a safety-critical part of the ETCS communication system is modeled and presents results of a real-time behavior evaluation.

2. The European Train Control System

In order to facilitate fast and efficient train traffic across borders in Europe, a unified European Train Control System (ETCS) [10] is under development in several European countries. ETCS is the core part of the more general European Railway Traffic Management System (ERTMS). The normal fixed block operation with mechanical elements, interlockings and optical signals will be substituted by a radio-based computerized train control system. The system receives commands about the train routes that are to be set, and then directs wayside objects along these routes. To simplify migration to the new standard, ETCS defines three levels of operation.

ETCS Level 1 uses spot transmission of information to the train via passive transponders. It is a supplement for the conventional, existing trackside signaling technology for lines with low to moderate train density. The block sections are defined by the existing signaling system. This level increases safety against passing signals at danger and in areas of speed restriction.

With the ETCS Level 2 system, radio communication replaces the traditional trackside signals, which allows considerable savings in infrastructure and maintenance costs. The system enhances safety considerably by monitoring train speed and, if necessary, intervening automatically. This allows higher speeds and shorter headways and thus increases capacity. The traffic management system processes and sends information and instructions for the train driver directly onto a monitor in the drivers cab via radio communication. A Radio Block Center (RBC) traces the location of each controlled train within its area. The RBC determines and transmits track description and movement authorities according to the underlying signaling system for each controlled train individually. The first ETCS Level 2 track has been installed between Olten and Luzern, Switzerland in April 2002 for Swiss Federal Railways (SBB).

ETCS Level 3 additionally takes over functions such as the spacing of trains. No trackside monitoring system is necessary as trains actively report their head and tail positions as well as train integrity to control centers. Moving block can be applied to increase line capacity. An essential advantage of level 3 is the reduction in life cycle costs through the abolition of the devices for track occupancy monitoring and trackside signals. The only trackside hardware necessary are so-called balises, small track-mounted spot devices which communicate their identity and exact position to trains that drive over them. They are used to re-calibrate the on-board position tracking system, which otherwise relies on wheel sensors and can thus be inaccurate during a longer trip.

Figure 1 depicts a simplified view of the communication architecture underlying ETCS. Each train features a train

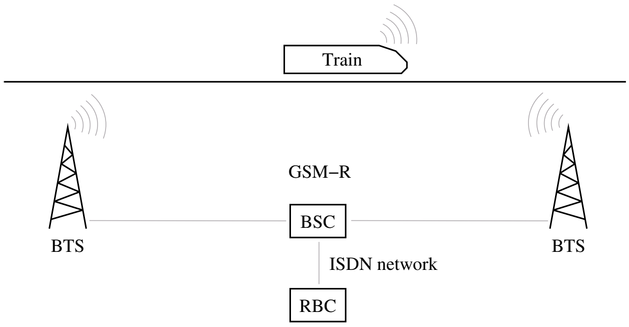

Figure 1. Simplified ETCS communication architecture

integrity control system and a computer that can control train speed. It communicates via GSM-R radio with base transceiver stations (BTS), which are connected to base station controllers (BSC) by cable. The BSCs are communicating with radio block centers via ISDN.

Radio Block Centers (RBC) are the trackside part for radio at ETCS level 2 and 3. Their major functions include safe train separation based on safe allocation of routes by regulation and interlocking. Position and integrity reports are sent by the trains periodically or upon request. Based on this information and the train routes, safe track areas are assigned to trains. This is done with so-called *movement authority* messages.

The European Integrated Railway Radio Enhanced Network (EIRENE) project was started on behalf of the European Railways to define a new digital radio standard for application on the European High Speed Rail System. The EIRENE System Requirements Specification [13] defines the set of requirements which a railway radio system shall comply with in order to ensure interoperability between national railways. GSM (Global System for Mobile Communications) was chosen as the base technology because of availability and cost considerations. Additional functions which are tailored to the needs of railroad use (like area addressing, automatic international roaming etc) have been defined as Railway GSM (GSM-R [7]). For up-link and down-link there are different frequency bands reserved for GSM-R around 900 MHz.

The EURORADIO layer of the communication link specifies the Radio Communication System requirements to the air gap interface between train and trackside equipment [11, 18]. The MORANE (Mobile Radio for Railway Networks in Europe [21]) project was set up to specify, develop, test and validate prototypes of a new radio system. Trial sites exist in France, Italy and Germany. Results of a quality of service test at one of these sites are presented in [24].

3. An ETCS Communication System Failure Model

The ability to exchange data packets with position and integrity reports as well as movement authority packets will be crucial for the reliable operation of ETCS. In this section, a quantitative model of the failure and recovery behavior of the communication base system is presented and analyzed. The results will be used in the subsequent section to examine moving block operation and the necessary data exchange while taking into account the reliability of the communication channel.

The model is based on the following sources of information about the qualitative and quantitative behavior of the communication system and its failures:

- A Quality of Service parameter specification (maximum connection establishment delay etc.) is given in the *Euroradio form fit functional interface specification* (FFFIS) [11].
- Allowed parameter ranges for some system design variables (like the minimum time between two subsequent position reports sent by a train) are specified in *ERTMS Performance Requirements for Interoperability* [12].
- Definitions of requirements of reliability, availability, maintainability and safety (RAMS) as well as acceptable numbers of failures per passenger-kilometer due to different reasons can be found in the *ERTMS RAMS Specification* [9].
- Some additional assumptions (mean time to complete the on-board train integrity check etc.) have been taken from a description of simulation experiments carried out by the German railways company [23].
- Another detailed description of communication quality of service parameters is provided in [16], serving as acceptance criteria for future measurements and tests of actual ETCS communication setups.
- Results of such a quality of service test at a railway trial site are presented in [24], thus facilitating a comparison with the original requirements. It turns out that the QoS parameters are in the required range, although often close to and even sometimes worse than the requirements. In the following, however, we adopt worst case assumptions based on the requirements, because future implementations will be bound to perform as required.

The communication link between train and RBC is always connected in normal operation mode. In that situation the following failures may happen:

Transmission errors occur from time to time, possibly due to temporarily bad radio signal conditions. There is no action necessary, because after a short time the link is operable again.

Connection losses may happen e.g. due to longer radio signal problems in areas where the radio coverage is not complete. The train hardware detects this state after some timeout and tries to establish a new connection. There is a slight chance of failing to establish such a connection until a certain timeout has elapsed, after which the connection establishment procedure starts over again.

Handovers take place every time the train crosses the border between the communication areas of two neighboring base transceiver stations (BTS). The train connects to the next BTS automatically, but this may take some time.

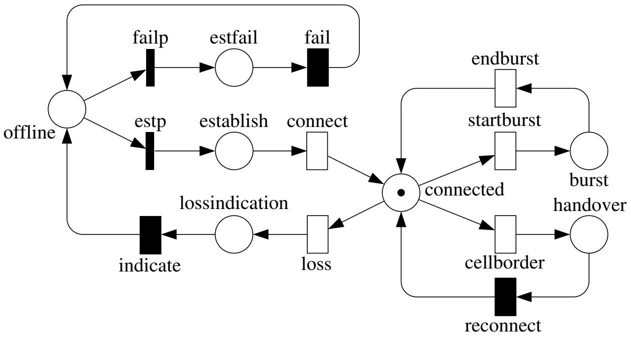

Figure 2. Failure and recovery model for GSM-R communication channel

Figure 2 shows a *deterministic and stochastic Petri net* [2] model of the described behavior. The firing delays and distributions have been chosen as follows. One unit of model time means one second in reality.

Transition `startburst` models the beginning of a transmission error. It has an exponentially distributed firing delay because of the stochastic nature of transmission errors. The corresponding firing time is comparable to a mean time to failure of the communication link due to transmission errors. The specification requires this value to be greater than or equal to 7 seconds for 95% of all cases. From the density and distribution functions of the exponential distribution

$$f(x) = \lambda e^{-\lambda x} \text{ and } F(x) = 1 - e^{-\lambda x}$$

we can calculate the necessary parameter λ value:

$$\lambda = -\frac{\ln p}{x} \approx 0.00733 \text{ with probability } p = 0.95 \text{ and } x = 7.$$

Transition `endburst` models the end of the transmission problem. The delay is assumed to be memoryless and the specification requires it to be smaller than one second in 95% of all cases. Thus the transition is exponential with parameter $\lambda \approx 3$ $(F(1) = 1 - e^{-\lambda} = 0.95)$.

The crossing of a cell border and connection setup with a new BTS is modeled by transitions `cellborder` and `reconnect`, respectively. Normally the BTS are situated a few meters away from the track and have a typical density of $0.1 \ldots 0.3$ BTS per km. Another source specifies 7 km as the mean BTS distance, which is adopted here. Unlike for personal use of a mobile phone, handovers happen quite often due to the speed of the train. ETCS is required to work for speeds up to 500 km per hour (139 meter per second). Thus the worst-case mean time between two handovers is 50.4 seconds. The firing delay of `cellborder` is thus exponentially distributed with parameter 0.0198 (the mean delay being equal to $1/\lambda$). From the specification we know that a reconnection is required to take at most 300 msec, which is taken as a worst case with a deterministic transition `reconnect`.

Following the specification, a complete connection loss takes place only rarely, namely 10^{-4} times per hour or $2.77 * 10^{-8}$ per second. The parameter of the exponential transition `loss` is set accordingly. There is a certain amount of time needed to detect the communication loss, which is required to be not greater than one second. This is modeled by the deterministic transition `indicate` with one as the fixed delay.

After being offline, the train communication system tries to reestablish the link at once. The requirements specify that a connection attempt must be successful with probability 99.9%, while in the remaining cases the establishment is canceled after 7.5 seconds and retried. This behavior is modeled with immediate transitions carrying the success/fail probabilities `estp` and `failp`, and the deterministic transition `fail` with delay 7.5. Connection establishment times are random, but required to be less than 5 seconds for 95% of the cases. The corresponding firing distribution of transition `connect` is thus exponential with parameter 0.6.

The model shown in Figure 2 depicts states and state transitions of the communication link. The initial state is `connected`. It is obvious that there will always be exactly one token in the model, letting the Petri net behave like a *state machine*, and the reachability graph is isomorphic to the net structure.

Because in every marking there is at most one transition with non-exponentially distributed firing delay enabled, the model can be numerically analyzed with standard DSPN algorithms [6]. Because of the state machine structure it would also be possible to exchange all deterministic transitions (delay τ) with their exponential "counterpart" (with

Place/State	Probability
connected	0.99166
burst	$2.4305 * 10^{-3}$
handover	$5.9027 * 10^{-3}$
lossindication	$2.7546 * 10^{-8}$
establish	$4.5910 * 10^{-8}$
estfail	$2.0680 * 10^{-10}$

Figure 3. Performance results of the communication failure model

firing rate $\lambda = 1/\tau$), without changing the resulting steady-state probability vector. It could then be analyzed as a simple GSPN [1].

Numerical analysis of the example is computationally inexpensive due to its small state space. Despite of the "stiffness" of the problem (e.g. firing rates of transitions `endburst` and `loss` differ by eight orders of magnitude) the exact solution is a matter of seconds. A simulation with proper confidence interval control would take quite some time because of the mentioned rare events.

Table 3 shows the results of the numerical analysis. The connection is working with a probability of 99.166%. This is much worse than the required availability of 99.95% as specified in [11]. This requirement is commented to be a coverage requirement, although we see from the model evaluation that already the allowed handover downtimes violate this requirement.

In fact, handovers account for more than 70% of the overall unavailability. To avoid their impact on the communication link, there are discussions about installing two independent GSM-R devices in each train. For instance in the Rome-Naples ETCS installation all electronic units have been duplicated for a higher reliability and availability. The connection to the next BTS can then be carried out when the train gets close to the cell border, thus avoiding any offline state due to handovers.

4. Derivation of a Condensed Failure Model

The goal of this paper is to evaluate the operation in moving block mode (ETCS level 3) under communication failures. In the subsequent section a model for the real-time communication between trains and radio block centers is presented. However, its performance evaluation is computationally expensive, which is in part due to the combination of the failure model with the normal operation model. The failure model as presented in Section 3 is therefore condensed into a smaller model in this section, facilitating a less complex evaluation of the combined model. This is possible because the failure model does not depend on the operation model.

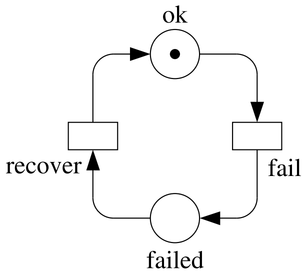

Figure 4. Condensed failure model

By doing so, there will be a tradeoff between model *complexity* and model *accuracy*. We decided to condense the failure model into a two-state system with the basic states `ok` and `failed`. The corresponding stochastic Petri net (GSPN) is shown in Figure 4.

The question is then how to specify the transition firing rates to minimize the approximation error. The main characteristic of the failure model is the mean availability, which shall be equal in the exact and condensed model. Thus the probability of having one token in place `ok` needs to be 0.99166.

Even with a correct availability an error can be introduced by selecting a wrong speed of state changes between `ok` and `failed`. If the speed would be too high, no correct packet transmission is possible, because a certain undisturbed time (given by the packet length and transmission bit rate) is always necessary. The second restriction which we impose on the condensed model is thus to keep the mean sojourn time in state `ok` exactly as it was in the full model. This time is the reciprocal value of the sum of all transition firing rates going out from the state, in our case $1/(\lambda_{startburst} + \lambda_{cellborder} + \lambda_{loss}) \approx 36.77$.

With these two restrictions the transition rates can be easily calculated. Let λ denote the transition rates and π the state probability vector in steady-state.

Then

$$\begin{aligned} \text{probabilities:} \quad & \pi_{ok} = 0.99166,\ \pi_{ok} + \pi_{failed} = 1 \\ \text{balance equations:} \quad & \pi_{ok}\,\lambda_{fail} = \pi_{fail}\,\lambda_{recover} \\ \text{sojourn time:} \quad & \lambda_{fail} = \frac{1}{36.77} \\ \text{and thus} \quad & \lambda_{recover} = 3.236 \end{aligned}$$

The model is then completely defined and will be used as a simplified failure model in the subsequent section.

5. A Moving Block Operation Model

In this section a model of the position report message exchange and emergency braking due to communication problems is developed and analyzed. We are interested in the dependency between maximum throughput of trains and reliability measures of the communication system. ETCS is being introduced in order to maximize track utilization by high-speed trains. The maximum utilization will be achieved if trains are following each other with a minimum distance. The question is then how close after each other can trains be operated under ETCS? We assume in the following a continuous track without stops, on which trains follow each other with a maximum speed v (current high-speed trains have a maximum speed of 300 km/h) and a distance s. Moreover, for the following considerations we arbitrarily select w.l.o.g. two trains (*Train1* and *Train2*) that directly follow each other. To ensure safety of the system, worst-case assumptions are made for all timings, distances etc.

Continuous operation is facilitated by the new notion of virtual *moving blocks*. Because there is no fixed block assigned to a train, and no physical block borders exist, the train movement is controlled by exchanging messages with the radio block center (RBC). Each train checks periodically its integrity and sends the integrity information together with the current position of the train head to the RBC. The time needed to check the train integrity is specified to be in the range between 2 to 5 seconds. Let Δt denote the time between two successive position reports of *Train1*. The requirements definition specifies $\Delta t \geq 5sec$. It is obvious that more frequent position reports will facilitate smaller train distances s, thus we choose $\Delta t = 5sec$.

The integrity/position report is sent via GSM-R to the RBC and processed there, which takes a maximum of 0.5 sec. The resulting information is sent to the following *Train2*, telling him either that everything is fine to go on driving (by sending a new *movement authority* packet) or that an emergency braking is necessary immediately.

However, if a communication packet is lost on either the communication up-link (*Train1*→RBC) or down-link (RBC→*Train2*), *Train2* needs to decide on its own at what point of time emergency braking is inevitable out of safety reasons. There is obviously a deadline d after the last movement authority has been received, after which the train needs to be stopped. The worst-case assumption is that after the last integrity check has been completed, a part of the train's carriages are lost from the main train and stop where they are. Another danger for the following train is an accident after sending the position/integrity report. The ETCS system requirements specification thus states that "In moving block operation the movement authority shall never exceed the min safe rear end of the preceding train" [10].

We would like to investigate the deadline and its dependency on the train distance s (see Figure 5 for an illustration). First of all the train length (about 410 m for the German high-speed train "ICE") needs to be subtracted from the distance. Second, when the results of the position/integrity report of *Train1* arrive at *Train2*, the information is already some time old. The worst-case delay can be estimated as follows: 5 seconds to complete the integrity check, 2.4 seconds end-to-end delay and 0.13 seconds (32 Bytes at 240 Bytes/second) to transfer the message to the RBC, 0.5 seconds to process the information there, again 2.4 + 0.13 seconds for the downlink transfer to *Train2* plus assumed 0.44 seconds to process the information in the train and start braking if necessary (12 seconds altogether). Then there is a location error possible in the position report of *Train1*, which cannot exceed 20m due to specifications of relative error and re-calibration at the balises. The emergency braking distance needs to be subtracted as well, being between 2300 and 2800m depending on the actual speed. For simplicity we assume braking distance plus train length plus position error as 3000m.

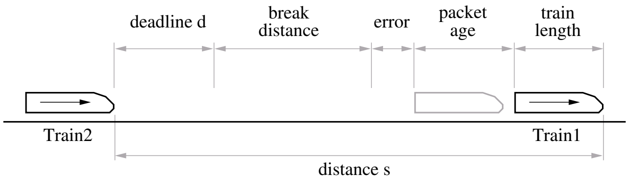

Figure 5. Train distance and deadline

The deadline d is then given by

$$d = \frac{s - 3000}{v} - 12$$

and the minimum theoretical distance for $v = 300km/h$ is thus $s_{min} = 4000$. This simple consideration already shows that the common term of "driving in braking distance" with ETCS is misleading, because even if everything would run perfectly trains cannot get closer than 4km.

Figure 6 shows a Petri net model for the above explained behavior. The upper part models the generation of the po-

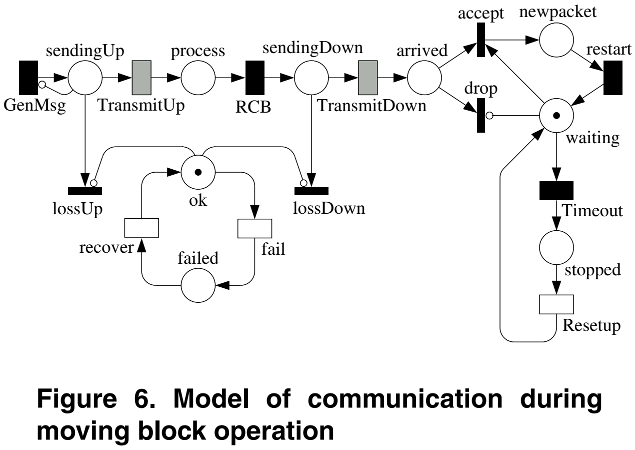

Figure 6. Model of communication during moving block operation

sition/integrity report and its transmission to the following train via the RCB. Every time a new movement authority arrives at the second train, the current deadline (transition `Timeout`) is reset and started again (transition `restart`). If the deadline is violated, the train stops and we assume a `Resetup` time of 15 minutes before the train can move on. Movement authority packets arriving during that time are dropped.

The failure behavior of the communication link is given by the condensed model as derived in the previous section. It is connected to the main model in a way that all messages are lost (tokens are removed from places `sendingUp` and `sendingDown`) as long as the link is failed. This is facilitated by firing transition `lossUp` or `lossDown`, which are enabled only when the communication channel is not in state `ok`.

The end-to-end transmission delay for messages is specified in the requirements as being between 0.4 and 0.5 seconds on the average, but being less than 0.5 for 95%, less than 1.2 seconds for 99%, and less than 2.4 seconds in 99.99% of all cases. For a realistic mapping of this timing behavior into the stochastic Petri net model we used two *generalized transitions* with expolynomial firing delays, how they are allowed in the class of *extended and deterministic stochastic Petri nets* (eDSPNs [14]). The actual data transmission times (0.13 seconds for a packet) have to be incorporated as well. The transition firing delay of `TransmitUp` and `TransmitDown` is defined by the following function:

$$f(x) = \begin{cases} 9.5 & \text{for } 0.53 \leq x < 0.63 \\ 0.057 & \text{for } 0.63 \leq x < 1.33 \\ 0.00842 & \text{for } 1.33 \leq x < 2.53 \\ 0 & \text{otherwise} \end{cases}$$

For the performance evaluation of the model the numeric analysis cannot be used, because the restriction of not more than one enabled non-exponential transition per marking is violated. Switching to an underlying discrete time scale [27] does not help either, because then the state space becomes so large that it cannot be handled by the available computing hardware. This is due to the fact that there are small delays and small differences between delays, necessitating a very small underlying time step, leading to a state space explosion.

Thus simulation was the only choice, but has its own problems. For a distance $s = 4500$ the deadline d is 6 seconds. In that case the model evaluation shows that the train is stopped with a probability of 94%. Even for a distance $s = 5000$ and a resulting deadline of 12 seconds, the probability of being stopped is 33%. For higher deadline values the simulation was not completed because it took too long to finish. The reason is that for a higher deadline, deadline misses are less frequent and the computation of the stopping probability depends exactly on that. It is well known that rare events as in this case cannot be handled efficiently by standard simulation algorithms. In the future we plan to apply acceleration techniques such as the RESTART method [25] to the train example.

However, the two computed probabilities of stopping are of course unacceptable and show that by assuming the worst case of the allowed specification, the communication based ETCS level 3 would be impossible to operate. Further investigations are necessary to study the behavior for different setups or more restrictive quality of service specifications in order to achieve a reliable train control system.

6. Conclusion

Model based performance evaluation is helpful during the design of distributed real-time systems. The paper presents the safety-critical communication inside the future European Train Control System as a case study. Stochastic Petri nets are used to model and evaluate the failure and recovery behavior of the communication link as well as its combination with the exchange of vital train information between trains and radio block centers. Numerical results are presented which put into perspective quality of service specifications and possible track utilization. The model evaluations show that worst-case communication behavior leads to unacceptable train operation situations. It will be crucial for the economic success of ETCS to further assess the real-time behavior of hardware and communication system under failures.

References

[1] M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis. *Modelling with Generalized Stochastic Petri Nets*. Series in parallel computing. John Wiley and Sons, 1995.

[2] M. Ajmone Marsan and G. Chiola. On Petri nets with deterministic and exponentially distributed firing times. In G. Rozenberg, editor, *Advances in Petri Nets 1987*, volume 266 of *Lecture Notes in Computer Science*, pages 132–145. Springer Verlag, 1987.

[3] G. Bucci and E. Vicario. Compositional validation of time-critical systems using communicating time Petri nets. *IEEE Transactions on Software Engineering*, 21(12):969–992, 1995.

[4] A. Chiappini, A. Cimatti, C. Porzia, G. Rotondo, R. Sebastiani, P. Traverso, and A. Villafiorita. Formal specification and development of a safety-critical train management system. In *SAFECOMP*, pages 410–419, 1999.

[5] G. Ciardo, R. German, and C. Lindemann. A characterization of the stochastic process underlying a stochastic Petri net. *IEEE Transactions on Software Engineering*, 20:506–515, 1994.

[6] G. Ciardo and C. Lindemann. Analysis of deterministic and stochastic Petri nets. *Performance Evaluation*, 18(8), 1993.

[7] A. Coraiola and M. Antscher. GSM-R network for the high-speed line Rome-Naples. *Signal und Draht*, 92(5):42–45, 2000.

[8] W. Damm and J. Klose. Verification of a radio-based signaling system using the STATEMATE verification environment. *Formal Methods in System Design*, 19(2):121–141, 2001.

[9] EEIG ERTMS User Group. *ERTMS/ETCS RAMS Requirements Specification.* UIC, Brussels, 1998.

[10] EEIG ERTMS User Group. *ERTMS/ETCS System Requirements Specification.* UIC, Brussels, 1999.

[11] EEIG ERTMS User Group. *Euroradio FFFIS*. UIC, Brussels, 2000.

[12] EEIG ERTMS User Group. *Performance Requirements for Interoperability*. UIC, Brussels, 2000.

[13] EIRENE Project Team. *EIRENE System Requirements Specification.* UIC, Brussels, 1999.

[14] R. German. *Performance Analysis of Communication Systems, Modeling with Non-Markovian Stochastic Petri Nets*. John Wiley and Sons, 2000.

[15] C. Ghezzi, D. Mandrioli, S. Morasca, and M. Pezze. A unified high-level Petri net formalism for time-critical systems. *IEEE Transactions on Software Engineering*, 17(2):160–172, Feb. 1991.

[16] M. Göller and L. Lengemann. Measurement and evaluation of the quality of service parameters of the communication system for ERTMS. *Signal und Draht*, 94(1+2):19–26, 2002.

[17] L. Jansen, M. Meyer zu Hörste, and H. Schnieder. Technical issues in modelling the european train control system. In *Proc. 1st CPN Workshop, DAIMI PB 532*, pages 103–115, Aarhus University, 1998.

[18] D. Kendelbacher and F. Stein. Euroradio - communication base system for ETCS. *Signal und Draht*, 94(6):6–11, 2002.

[19] A. Mazzeo, N. Mazzocca, S. Russo, and V. Vittorini. A systematic approach to the Petri net based specification of concurrent systems. *Real-Time Systems*, 13:219–236, 1997.

[20] M. Meyer zu Hörste and E. Schnieder. Modelling and simulation of train control systems using Petri nets. In J. M. Wing, J. Woodcock, and J. Davies, editors, *FM'99 Formal Methods. World Congress on Formal Methods in the Development of Computing Systems.* Volume 1709 of *Lecture Notes in Computer Science*, page 1867, Springer Verlag Berlin, 1999.

[21] MORANE Project Group. *Radio Transmission FFFIS for Euroradio*. Brussels, 1998.

[22] T. Murata. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580, 1989.

[23] J. Osburg. Performance investigation of arbitrary train control techniques. *Signal und Draht*, 94(1+2):27–30, 2002.

[24] R. Schrenk. GSM-R: Quality of service tests at customer trial sites. *Signal und Draht*, 92(9):61–64, 2000.

[25] M. Villen-Altamirano and J. Villen-Altamirano. Enhancement of the accelerated simulation method RESTART by considering multiple thresholds. In *ITC-14*, J. Labetoulle and J.W. Roberts (Eds.), pages 797–810, North-Holland, 1994. Elsevier Science Publishers B. V.

[26] A. Zimmermann, J. Freiheit, R. German, and G. Hommel. Petri net modelling and performability evaluation with TimeNET 3.0. In *11th Int. Conf. on Modelling Techniques and Tools for Computer Performance Evaluation*, pages 188–202, Schaumburg, Illinois, USA, 2000. LNCS 1786.

[27] A. Zimmermann, J. Freiheit, and G. Hommel. Discrete time stochastic Petri nets for modeling and evaluation of real-time systems. In *Proc. Int. Workshop on Parallel and Distributed Real-Time Systems (WPDRTS01*, pages 282–286, San Francisco, 2001.