Chapter 4

Model-Based Fault Analysis for Railway Traction Systems

Jon del Olmo, Fernando Garramiola, Javier Poza and Gaizka Almandoz

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.74277

Abstract

Fault analysis in industrial equipment has been usually performed using classical techniques such as failure modes and effects analysis (FMEA) and fault tree analysis (FTA). Model-based fault analysis has been used during the last several years in order to overcome the limitations of classical methods when complex industrial equipment has to be analyzed. In railway and automotive sectors, the development and validation of new products are based on hardware-in-the-loop (HIL) platforms. In this chapter, a methodology to enhance classical FMEAs is presented. Based on HIL simulations, the objective is to improve the results of the fault analysis with quantitative information about the effects of each fault mode. In this way, the impact of the fault analysis in the design of the traction system, the development of new diagnostic functionalities and in the maintenance tasks will increase.

Keywords: railway, traction, fault, models, FMEA

1. Introduction

Nowadays, reliability, availability, maintainability and safety (RAMS) are key features in the development of industrial equipment. Moreover, new maintenance approaches, such as condition-based maintenance (CBM) have emerged, as an alternative to preventive maintenance and run failure techniques [1]. In sectors such as railways, automotive and aviation, the business model is based on the sale of equipment and its long-term maintenance. Taking into account that locomotives might have a service life of up to 30 years, a long interval of the life cycle is related to maintenance tasks and technical services. This has been an incentive for

manufacturers to improve the reliability and maintainability of their products [2]. During the last years, several projects were launched in order to develop smart maintenance systems [3–6].

From the very first phases of the Life Cycle, reliability and safety analysis are performed. In these analyses, the effect of faults in the functionalities of the system is studied. In this field, techniques such as FMEA and FTA allow the designer to identify systematically fault modes (FMs) and effects. However, lately some limitations have been identified in the application of these classical methodologies, especially when a complex control system has to be analyzed. Moreover, more and more manufacturers are using Model-based techniques [7] in the development of new systems, following the global tendency of Model-based engineering. Fault analysis is not an exception and several authors have proposed to use models to analyze the effects of faults [8–10].

The extensive use of models for the development of railway systems has allowed the introduction of tools such as hardware-in-the-loop (HIL) platforms. These platforms have a key role in the validation of embedded control units, just as they have in automotive applications. Thanks to this kind of platforms, development time and costs are reduced considerably.

Taking into account the limitations of classical fault analysis methods and the extensive use of HIL platforms in the railway traction system manufacturing sector, in this chapter, a methodology for model-based fault analysis that takes advantage of HIL platforms is presented. Concretely, this methodology has been developed to improve the analysis of faults and effects of the railway traction system shown in **Figure 1**. This traction converter box has a three-phase inverter supplying two induction motors in parallel. Moreover, it has a breaking chopper, a DC-Link, an input filter and voltage, current and speed sensors. The control of the converter is executed by an embedded traction control unit (TCU).

The main goal of the methodology is to quantify the effects of the faults using analytical models and simulation platforms. Generally, the conclusions of an FMEA are used to improve designs, in future redesigns, identify new maintenance tasks and develop new fault detection and identification algorithms. With model-based approaches, the quality of the analysis improves and its impact on the three aforementioned aspects is increased.

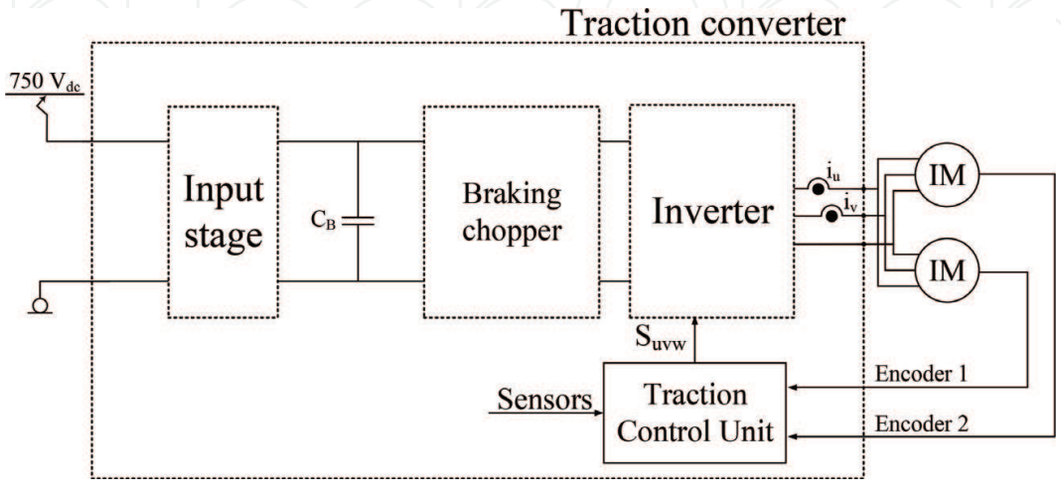


Figure 1. Tramway traction system schematics.

The chapter is organized as follows. In Section 2, classical fault analysis methods and their limitations are described. In Section 3, a brief state-of-the-art about model-based fault analysis is presented. In Section 4, the new model and HIL-based methodology is shown, accompanied by a case study. Finally, in Section 5, the conclusions are drawn.

2. Fault analysis in complex control systems: classical methods and limitations

2.1. Classical faults and effects analysis methods

In this section, a brief description of classical fault analysis methods will be presented. It provides the reader with a basic understanding of the most used techniques in the analysis of faults and their effects in complex control systems.

2.1.1. Failure modes and effects analysis (FMEA)

Fault modes and effects analysis (FMEA) is an inductive method used in the development of products in order to identify and classify fault modes and effects. This technique establishes a systematic approach to identify effects for each fault mode and classify them in terms of occurrence of probability and severity. Occurrence and detection probabilities are combined with severity indexes to obtain a risk priority number (RPN). Hence, corrective actions can be prioritized [11]. Recently, due to the deficiencies of RPN application in real-world cases, enhanced approaches have been proposed [12].

The main steps to perform the FMEA analysis are [11]:

Before the analysis:

1. Gather information about requirements, components, architecture and fault modes.
2. Organize the system under analysis in a structured way. Describe the architecture and its limits in a block diagram.

During the analysis:

3. Fault mode analysis and FMEA worksheets:
   - **a.** Determine fault modes.
   - **b.** Determine the effects of each fault mode.
   - **c.** Determine the causes of each fault mode.
   - **d.** Determine the existing protection actions.
   - **e.** Obtain the RPN of each fault mode.
   - **f.** Prepare FMEA tables.

After the analysis is completed:

4. Summarize the analysis in an FMEA report.

5. Establish corrective actions to mitigate the effects of the faults.

At the end of this process, an FMEA worksheet is obtained. Several worksheet examples can be found in the literature [11, 13].

Table 1 shows an excerpt from an FMEA worksheet of a railway traction system. In particular, the behavior of a railway traction drive under phase current sensor faults is analyzed.

2.1.2. Fault tree analysis (FTA)

A fault tree is a graphical representation of all the basic events that can cause an undesired event in a process or system. The faults may be related to hardware components, human errors or any other event that can generate an undesired situation. Therefore, a fault tree describes the logical relation between basic events that can lead the system to a faulty state.

It is important to understand that a fault tree is created for its main undesired event. Hence, it does not describe all the faults that can occur in a system and more than one Fault Tree is needed to describe the faulty behavior. **Figure 2** shows an example of a fault tree and the steps to build it.

The analysis performed with the FTA technique consists of the following steps:

1. Define the undesired event under study. A unique fault tree is obtained for each event.

2. Understand the system. All the causes that can lead to the main event are analyzed.

3. Fault tree construction. Events and causes are linked using logic gates. This step is an iterative process. An event is selected and its causes are identified. These causes are classified as basic events, undeveloped events or intermediate events and the logical gates

Fault mode	Cause	Local level effect	Traction unit level effect	Train level effect
Measured value bigger than real value	Internal failure	False measurement	Inappropriate control Overcurrent Disabled converter.	Loss of traction unit
No measurement	Internal failure	No measurement	Inappropriate control Overcurrent Disabled converter	Loss of traction unit
Open-circuit	Internal failure	No measurement	Inappropriate control Overcurrent Disabled converter	Loss of traction unit
Measured value smaller than real value	Internal failure	False measurement	Inappropriate control Overcurrent Disabled converter	Loss of traction unit

Table 1. Excerpt from an FMEA worksheet of railway traction systems: Phase current sensor FMEA.

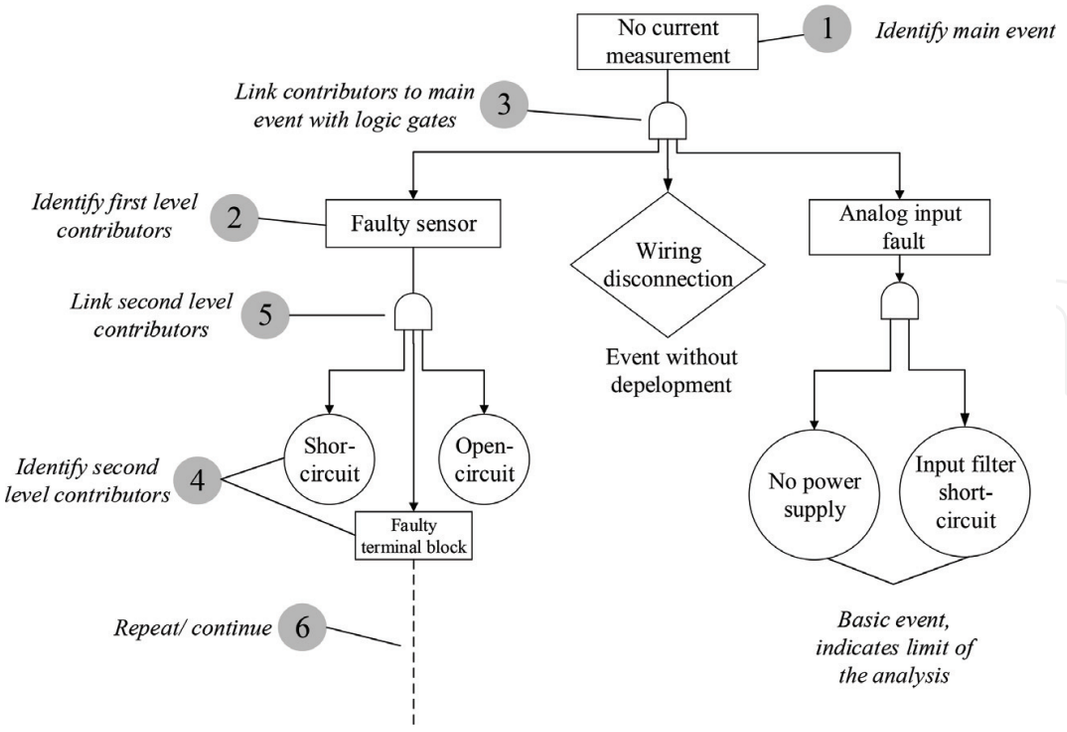


Figure 2. FTA for current sensor.

to link all of them are chosen. In this way, the tree is drawn from the top event to the basic events (see **Figure 2**).

4. Evaluate the fault tree. It is analyzed in order to suggest improvements. Moreover, the risk each fault generates is assessed.

5. Control measures are proposed. Once the risks associated with each fault or undesired event are identified mitigation measures are proposed.

2.2. Limitations

One of the challenges that railway traction system manufacturers have to face is the difficulty of combining and coordinating product development and safety analysis tasks. The integration of systems engineering and safety analysis is addressed in many other research works [14–16]. Nowadays, the fault analysis is mainly performed using the tools described in the previous section. During the design stage, the safety analysis is performed in order to achieve two main objectives. The first one is to draft a safety case document that allows the manufacturer to obtain the corresponding safety certificate. The second is to analyze the architecture of the system under development to ensure that meets availability, reliability and maintainability requirements. Nevertheless, as the system is analyzed from the high level (only the architecture is studied and implementation details are not taken into account), design teams rarely receive any feedback about the details of the effects and the potential improvements to mitigate those effects.

If the fault analysis is intended to be a source of information for future system improvements, redesigns and maintenance task, this analysis has to be more detailed and performed in collaboration between the design and the safety teams. Classical fault analysis methodologies lack tools to manage this need.

In addition, techniques such as FMEA and FTA are performed manually and they are based on requirement documents and informal models of the system [9]. With informal models, we refer to architecture models developed in the early stages of the design process. In the life cycle of a product, the safety analysis is part of the design step and it does not usually reflect the changes that the system has experienced later in the implementation and validation phases. Hence, these analyses are usually incomplete.

Apart from the aforementioned shortcomings, the following limitations have been also identified by other authors [8, 9, 17, 18]:

- They represent the fault logic in a static way and they do not allow analyzing neither the time information nor the dynamic behavior of the system. For instance, in railway traction applications a conventional FMEA does not usually reflect the behavior of the control strategy under faulty conditions. It is difficult to take into account the following aspects using classical methods: the implementation details of the control strategy, the transitions between control modes, the changes between operation points and the interaction between different subsystems when a fault occurs.
- The analysis depends on the skills of the analysist to predict the effects of each fault. Apart from the difficulty to reflect the dynamic behavior with a classical method, the analyst needs to know the specifics of the control strategy and its implementation. It is worth mentioning that under faulty conditions, a closed loop control reacts to compensate the effects of the fault. This is an added difficulty in the fault analysis process since fault effects are modified by the control system itself.
- Classical methods do not provide with efficient tools to manage the complexity and the number of components of current industrial equipment.

3. State of the art of model-based fault modes and effects analysis techniques

3.1. Introduction

Model-based safety analysis has been developed during the last years to help the analysis of complex systems, taking as a central element the model and automating the analysis of extended fault models [19]. This new approach intends to overcome the limitations mentioned in Section 2.2.

Model-based safety analysis methodologies can be divided into two main groups: failure logic modeling (FLM) and behavioral fault simulation (BFS), also known as fault injection (FI) [8, 10].

In the following section, these two alternatives will be described and the selection of fault injection as the basis for the new methodology will be explained.

3.2. Model-based fault modes and effects analysis techniques

3.2.1. Failure logic modeling approach

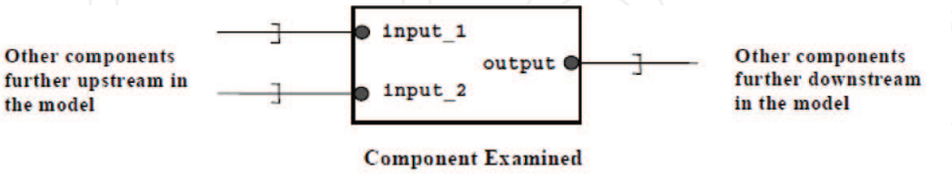
This methodology is based on the automatic generation of fault trees and FMEA worksheets using the information stored in models [8]. For each component of the model, its inputs, outputs and behavior under fault are defined [10, 20]. To specify the behavior, the following elements have to be described:

- Input fault modes.
- Internal fault modes.
- The logic that defines the effect of the input and internal fault modes in the output.

Figure 3 shows the definition of fault modes using this methodology.

Once the fault logic is defined for each component, a fault model can be developed linking the outputs of each block with the inputs of the next block. The components are connected as defined in the architecture, which allows reflecting the structure of the system in the posterior FMEA analysis. With this new model, the propagation of each fault mode can be studied. Moreover, the homogeneous and systematic description of each element (with output fault modes as a function of input fault modes and internal fault modes) allows automating the analysis process.

In the literature, several techniques and tools can be found to automate this process following the failure logic modeling approach [8]. Among others, from the railway traction application point of view, the HiP-HOPS methodology and its associated tools are the most interesting ones [19, 22]. This tool, implemented as a Matlab/Simulink tool, allows the analyst to define



Output Failure Mode	Description	Input Deviation Logic	Component Malfunction Logic	λ(f/h)
Omission-output	The component fails to generate the output	Omission-input_1 AND Omission-input_2	Jammed OR Short_circuited	$5x10^{-7}$ $6x10^{-6}$
Wrong-output	The component generates wrong output	Wrong-input_1 OR Wrong-input 2	Biased	$6x10^{-8}$
Early-output	Early output	...	...	...

Figure 3. Definition of fault modes for fault analysis using FLM and hip-HOPS methodology [21].

the fault behavior of the system over Simulink models. This is an advantage in the field of electric drives because Matlab/Simulink is one of the main tools used for their development.

With FLM methodologies, some of the limitations mentioned before are overcome. The efforts have been directed towards the automation of the analysis and characterization of faulty systems using extended models. Nevertheless, as it was mentioned in [18], the main limitation that FLM techniques have to face is the lack of tools to analyze the dynamic behavior of the system. They do not consider the changes in the states of a system and are not able to model its dynamic or temporal behavior [19]. A common framework to present the structure and analyze the propagation of the system is defined, but the information about the dynamic behavior of the system (operation modes, reaction to faults, etc.) is not used. In this respect, some authors have proposed alternatives that take into account the dynamic behavior of the system. In the study by Kabir [23] one of the main objectives is the modeling of the behavior of the system using state machines.

3.2.2. Behavioral fault simulation/failure injection approach

The BFS technique is based on the injection of faults using executable models of the system to define their effects [8]. The starting point for the analysis is a formal or nominal model (without faults) known and used during the design stage (see **Figure 4**). This model is extended with information related to the faults of the system. In this way, the effects of the faults and the behavior of the system when the faults occur can be analyzed. The extended and common model assures that the results of the fault analysis are relevant and that are updated with respect to design changes.

The key to this approach is that the models are executable and they allow analyzing the dynamic behavior of the system under faults. An analysis platform is used to simulate the extended models and assure that the system meets safety requirements.

Up to now, the work related to fault injection has been focused on the development of simulation platforms to apply the described methodology. In [24] and [25] the results of two European projects were presented, where this technique was applied to models developed in

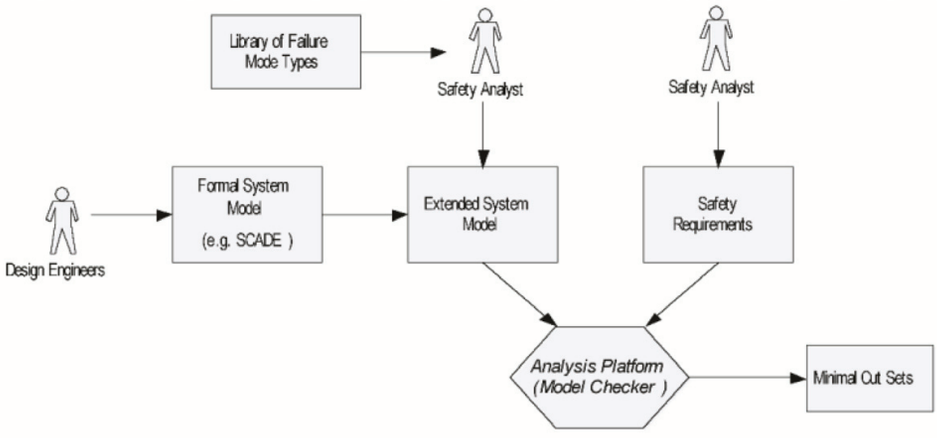


Figure 4. Principal components of failure injection approach [10].

SCADE and Statement software. In [9], the nominal models developed in Simulink were translated into SCADE that has a module for fault mode analysis.

Compared to FLM methods, BFS approaches allow simulating the dynamic behavior of the system. Moreover, as the models already validated in the design phase are the basis for the analysis, the results obtained about the effects are more accurate. Hence, the fault analysis does not entirely depend on the knowledge of the analysist because the system and its behavior have been defined in the model. In any case, some disadvantages have been also identified. First, since extended dynamic models are used, there is an inherent difficulty to process automation, as well as, not to generate an excessive amount of models. It has to be taken into account that in complex systems there could be a lot of different fault modes. Each fault mode could demand a different model and simulation in order to perform the fault analysis. In some systems, the number of simulations could be unmanageable. Second, another drawback of BFS methods is that it can be only applied after the designs and the models are developed, in the later stages of the development process [10]. This makes the introduction of changes difficult if the system does not comply with requirements.

However, it is worth mentioning that nowadays many manufacturers have chosen model-based systems engineering [26] as their main approach for the development of complex systems, avoiding document-based product development. This methodology is based on the extensive use of models during the whole life cycle of a product, from the requirement definition phase to the validation stage. Therefore, this context is ideal for the deployment of model-based fault injection methods for fault analysis.

4. Hardware-in-the-loop-based FMEA for railway traction applications

4.1. Need for a quantitative fault analysis methodology

As it was mentioned in the introduction, this work is related to the development of electric drives for railway traction applications. The results obtained using the classical fault analysis methods were assessed using reference information provided by the manufacturer CAF Power & Automation. The limitations identified in the literature were also found in the industrial application.

Therefore, the objective of the work presented here was to propose a new fault analysis methodology in the field of electric drives. The requirements of this methodology were:

- To take advantage of the information generated during the design phase to perform the fault analysis. During the design and validation phases, models of the system are used. These models contain all the information about the architecture and the behavior of the traction converter and should be used in the fault analysis process, as part of the model-based development of industrial products.
- Fault effects must be quantified. If the conclusions of the fault analysis are intended to condition the design, development and maintenance, these conclusions should be

quantitative. For example, in the field of electric drives, the effects should be described in terms of traction/braking capacity loss, consumption increase, harmonic component increase or comfort deterioration. There is also a lack of indicators for the detection and identification of faults.

- In order to achieve a better coordination between the design and safety teams, both should share the same tools. The main tool for the design and the safety analysis should be the model of the system. Nowadays the resources available in the development of traction systems are Matlab/Simulink for the simulation of models and HIL platforms for the validation of embedded traction control units.

In order to tackle these needs, in the next section, a new methodology for fault analysis is proposed. The methodology is based on models and an HIL platform, following the fault injection approach presented in Section 3.2.2. In the literature, most of the work presented about HIL platforms was focused on the development and validation of new control strategies and embedded units [27–29] and in some cases, authors mention the use of such platforms for the analysis of specific faults [30–33]. Nevertheless, it is difficult to find publications where an HIL platform is used systematically in the analysis of fault modes and effects of a system.

4.2. Hardware-in-the-loop platform

The methodology presented here uses an HIL platform to obtain information about the behavior of the faulty railway traction system and its control strategy. The structure of the system is shown in **Figure 5**.

The model of the traction system, designed in Matlab/Simulink, is simulated in an OPAL-RT Real-Time Simulator. Thanks to its analog and digital inputs and outputs, it communicates with a commercial TCU developed by CAF Power&Automation. The tests are monitored from an auxiliary PC where the data is stored and analyzed.

The Real-Time Simulator allows simulating the same models developed in the design stage. In the case of electric drives, these models are usually implemented using Matlab/Simulink, which is the language also used by the simulator. If the Real-Time Simulator requires the adaptation of the models, the implementation of the extended models itself becomes an objective, which is an obstacle in the integration of the design and the safety analysis tasks. In this case, as an OPAL-RT simulator is used, Matlab/Simulink models can be imported directly into this device, reducing the development and adaptation time, and enabling to reuse the existing know-how. **Figure 6** shows the simulation model for the traction system. This model is the same as the one used in the design stage and contains some additional blocks to manage the interaction between the simulator, the TCU and the monitoring PC. The extension of the model to make its execution in real time possible takes little time and there is no need for expert knowledge about real-time simulations.

Furthermore, it has to be taken into account that the aim of the simulations is to replicate the behavior of the system under faults, so a flexible simulation environment is needed. This environment has to allow an effective and simple way to simulate faults. The second

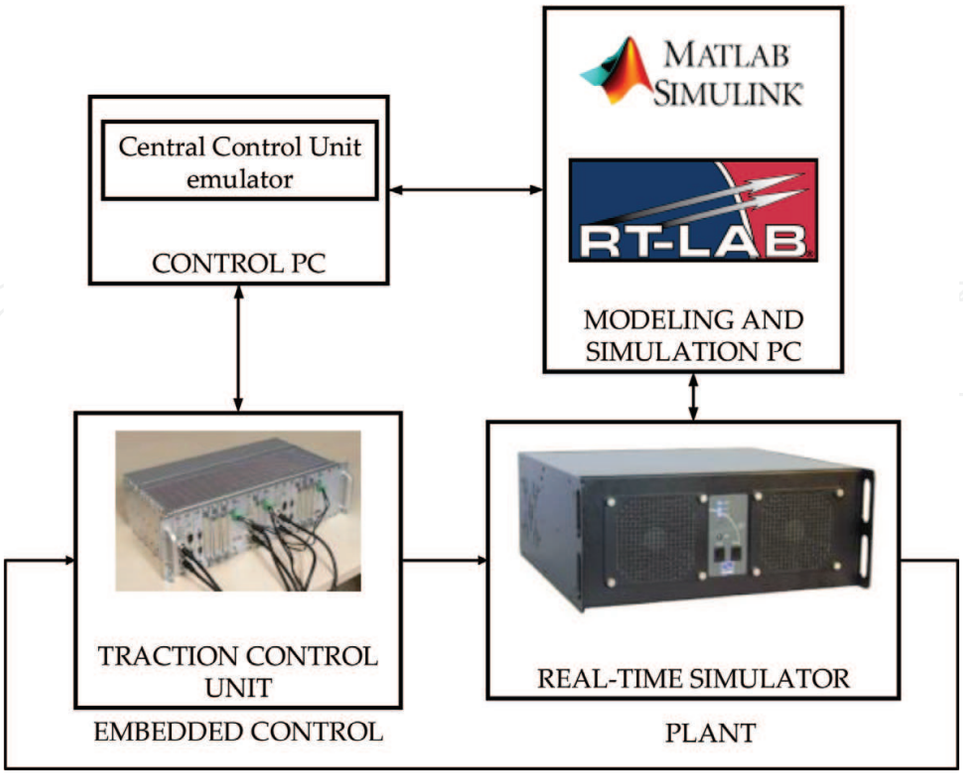


Figure 5. HIL platform structure.

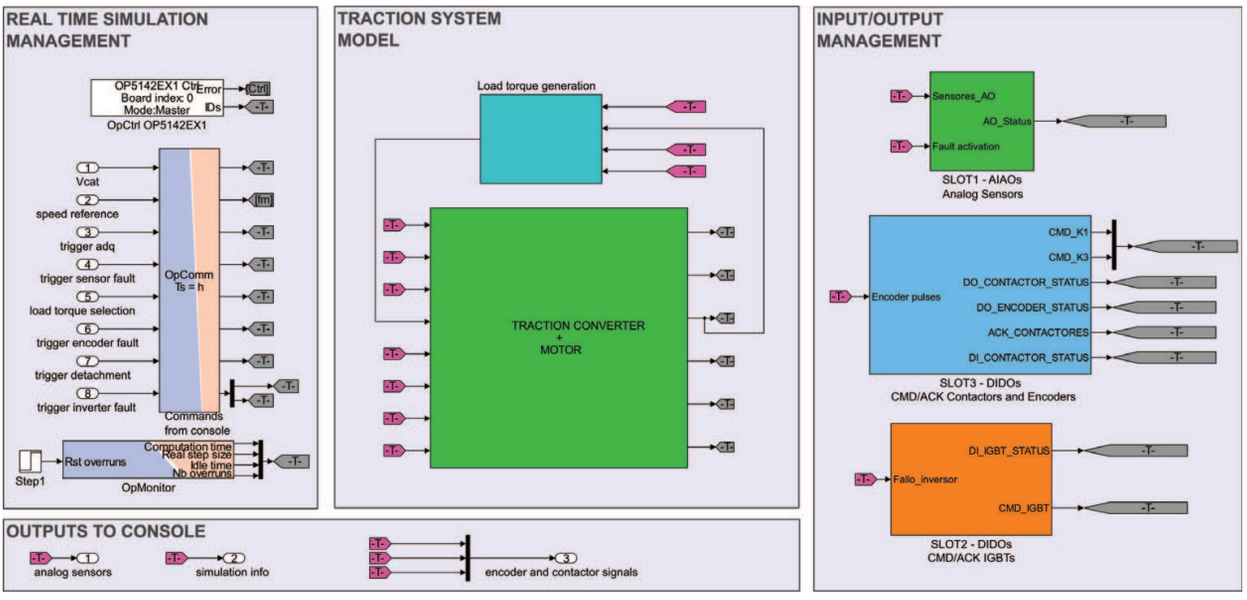


Figure 6. Real-time simulation model of railway traction system.

component of the HIL platform is the TCU. This device is the electronic control unit that controls the traction system. Thanks to the HIL platform, a commercial version of the TCU was used, executing the same control code as in a real application. This allows to gather information about the control strategy that otherwise, with a simple FMEA analysis, would

be impossible to obtain. Moreover, using a commercial TCU, implementation and manufacturing details are taken into account and fed back into the enhanced FMEA. It is important to note that thanks to this kind of platforms the results of the fault analysis are quantitative and more detailed than the results obtained with classical methods.

4.3. Methodology

The starting point in the methodology for model-based fault analysis with HIL platforms is an FMEA based in conceptual/empirical knowledge about the system (see **Figure 7**). As conceptual FMEA we refer to the initial fault modes and effects analysis that is performed during the design stage of railway traction systems. This initial or existing FMEA is usually based on design and requirement documents and standards, following the classical methodologies for fault analysis.

The methodology is composed of the following steps:

1. Complete conceptual FMEA

   In the first step, all the information related to the architecture and the behavior of the traction system is gathered. The main information source is the initial FMEA (if there already was one), but other information sources should be considered:

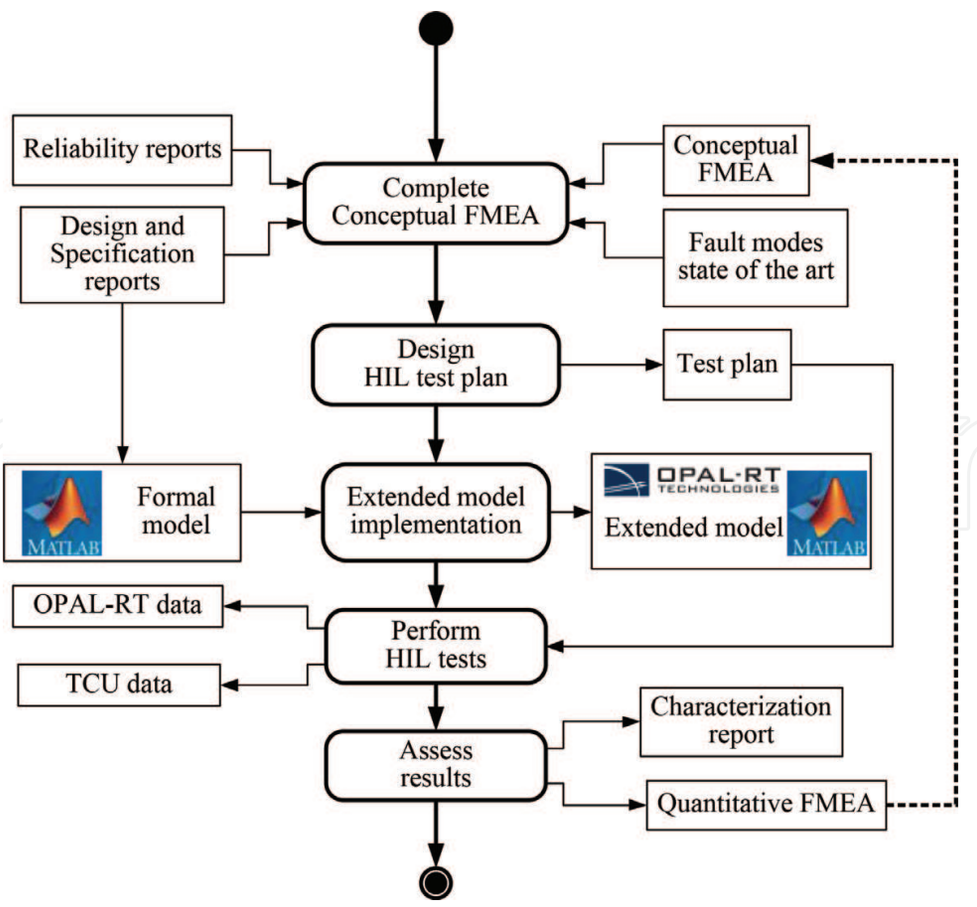


Figure 7. Flux diagram of the methodology for the model-based characterization of faults and effects.

- Design and specification documents where it is defined as what the system is (architecture) and how does it work (behavior).
- Reliability reports written by the after-sales department, where information about fault rates, mean time between failures and availability is presented.
- Updated state of the art about fault modes, causes and effects of the components of the traction system.

2. Design a test plan

   This plan is a document where all the conditions for each test are stated. Among others, these aspects should be defined:

   1. General description of the test
      a. Objective
      b. Fault mode: description and magnitude of the fault.
      c. Subsystem under fault. Part of the system that needs to be tested.
   2. Operation point. Operation phase where the system is working when the fault occurs (steady state, transient state, in traction, braking, and so on)
   3. Expected fault effects based on the conceptual FMEA. Thanks to the FMEA, preliminary fault effects are identified and the variables where the effects are visible established.
   4. Platform configuration. Define the configuration of the components of the HIL platform.
   5. Summary of the test to perform.

3. Implement and validate the extended model

   In this step, the formal model is extended in order to simulate the faulty behavior. This extended model is already defined by the fault modes that need to be tested and the expected effects considered in the test plan. The expected effects should be clearly stated in order to know if the extended model would be able to replicate them.

4. Perform HIL test

   HIL tests are performed following the test plan for the selected operation conditions, storing the required data.

5. Analyze and assess results

   A fault characterization report is written where the details of the faulty behavior are described. The changes that the fault has generated in the operation of the system are reported using tables and graphs. The data recorded in the TCU about the control strategy is analyzed and converted to quantitative information about the effects of the faults. Moreover, the conclusions are fed back to the initial FMEA, which is sent to the diagnostics and maintenance teams.

It is worth mentioning that this methodology can be applied iteratively adding information to the initial FMEA during the whole life cycle. It could be a tool for fault analysis throughout different phases of the life cycle. In this chapter, the model-based FMEA has been proposed as a tool to improve fault analysis during the validation process, in which a commercial TCU or a prototype is available for unit testing, as shown in **Figure 8**. In this diagram, a traction system development V-model is presented. The model describes the different phases of the life cycle. As it was mentioned, the HIL simulations are commonly used in the validation and testing process, but the methodology could be used in the design stage with model-in-the-loop (MIL) or software-in-the-loop (SIL) simulations.

With this methodology, fault analysis can be applied not only in the design phase but also in the system integration process, enhancing the quantity and the quality of the FMEA. It allows the continuous improvement of products in many aspects such as safety, maintenance and fault detection and identification functionalities.

An immediate result of the approach and the quantitative FMEA is the improvement of the maintenance manual, which is completed with quantitative information about the effects and the indicators of each fault.

4.4. Use case: quantitative FMEA of current sensors

In the following sections, a use case of the proposed methodology will be presented. In this case, the example will be focused on the analysis and identification of phase current sensor fault modes (FMs) and effects. The steps shown in **Figure 7** are explained for current sensor faults.

4.4.1. Complete conceptual FMEA

In this step, the initial FMEA is extended with additional information obtained from the literature and from the specification documents of the traction system (see **Table 2**).

First, it should be noted that the number of fault modes has increased. Some authors point out [34, 35] that an unbalanced measurement of three-phase currents in a traction drive generates a

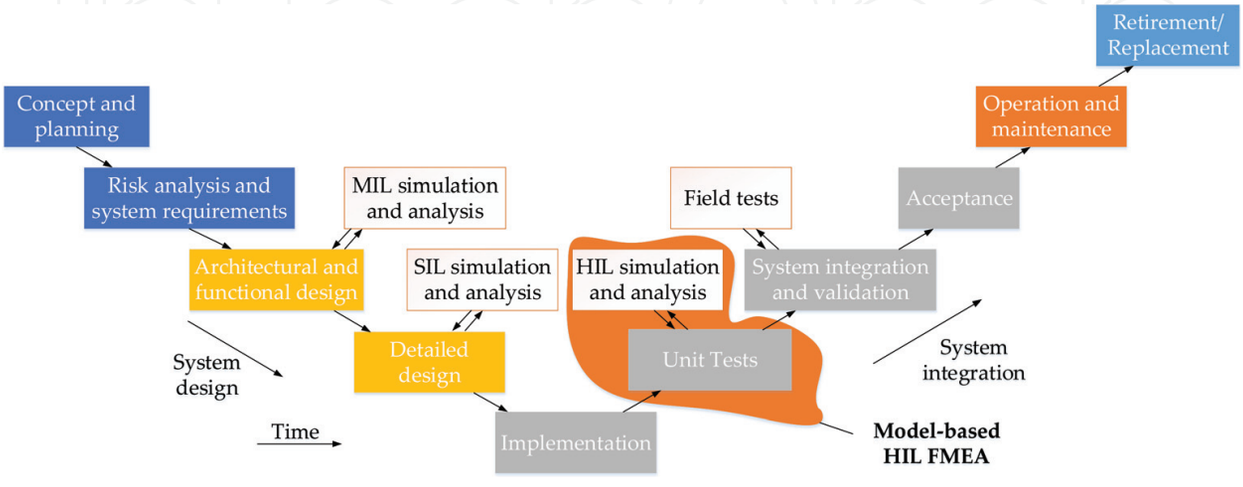


Figure 8. Railway traction system life cycle.

Operation phase	Fault mode	Cause	Local effect	Traction unit effect	Train effect
Normal operation	**FM1** Measured value bigger than real value (offset)	Internal failure: offset	False measurement	Oscillations in the torque at motor operation frequency	Loss of traction unit
Normal operation	**FM2** Measured value bigger than real value (gain)	Internal failure: gain	False measurement	Oscillations in the torque at twice stator frequency. Torque controlled below reference value	Loss of traction unit
Normal operation	**FM3** No measurement (sensor connected but no signal)	Internal failure	No measurement	Inappropriate control. Overcurrent. Disabled converter	Loss of traction unit
Start-up and motor fluxing	**FM4** Open-circuit	Internal failure	No measurement	Inappropriate fluxing. Overcurrent. Disabled converter	Loss of traction unit
Normal operation	**FM5** Open-circuit	Internal failure	False measurement	Inappropriate control. Overcurrent. Disabled converter	Loss of traction unit
Normal operation	**FM6** Measured value smaller than real value (offset)	Internal failure: offset	False measurement	Oscillations in the torque at stator frequency	Loss of traction unit
Normal operation	**FM7** Measured value smaller than real value (offset)	Internal failure: offset	False measurement	Oscillations in the torque at twice stator frequency. Torque controlled above reference value	Loss of traction unit

Table 2. Conceptual FMEA of a railway traction systems: Phase current sensor FMEA.

low-frequency oscillation in the torque. Depending on the type of deviation, offset or gain, the oscillation has different frequencies, and the effects change. Hence, the fault modes (FMs) have been described in detail.

Moreover, knowing that the control of the traction motor has different operation phases, it was considered interesting to analyze the effect of a sensor disconnection fault during the fluxing of the motor (FM4).

4.4.2. Design HIL test plan

Once the fault modes and effects are selected, the HIL test plan is defined. The test plan contains all the information mentioned in section 4.2 and will not be reproduced here due to lack of space.

4.4.3. Implement and validate the extended model

In this step, the extended model for the simulation of phase current sensor faults was implemented. Taking into account the fault modes described in **Table 2**, a fault injection bock was implemented to inject gain and offset faults in the measurement of the sensor (see **Figure 9**).

4.4.4. Perform HIL tests

The tests are performed following the test plan, for the required operation points and fault modes.

As a result, two data files about the evolution of the system are obtained. On the one hand, a file gathers the evolution of the internal variables of the traction system. In this case, the data is obtained from the real-time simulator. On the other hand, there is another group of variables that is stored and downloaded from the TCU. These variables reflect the behavior of the control strategy.

4.4.5. Assess results

In the evaluation step, the data from the tests is converted into information to improve and quantify the effects in the FMEA. In the case of the phase current sensor faults, two effects were identified. Due to gain and offset deviations, a new harmonic component appears in the

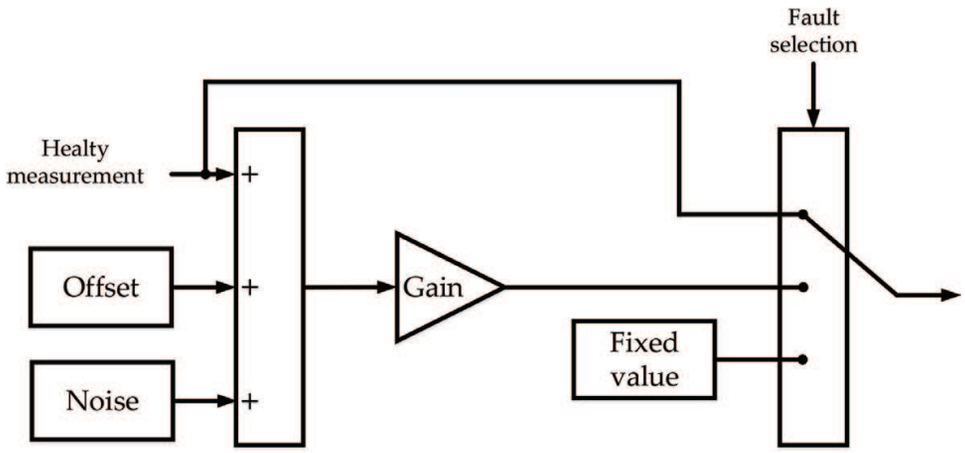


Figure 9. Fault injection block for phase current sensors.

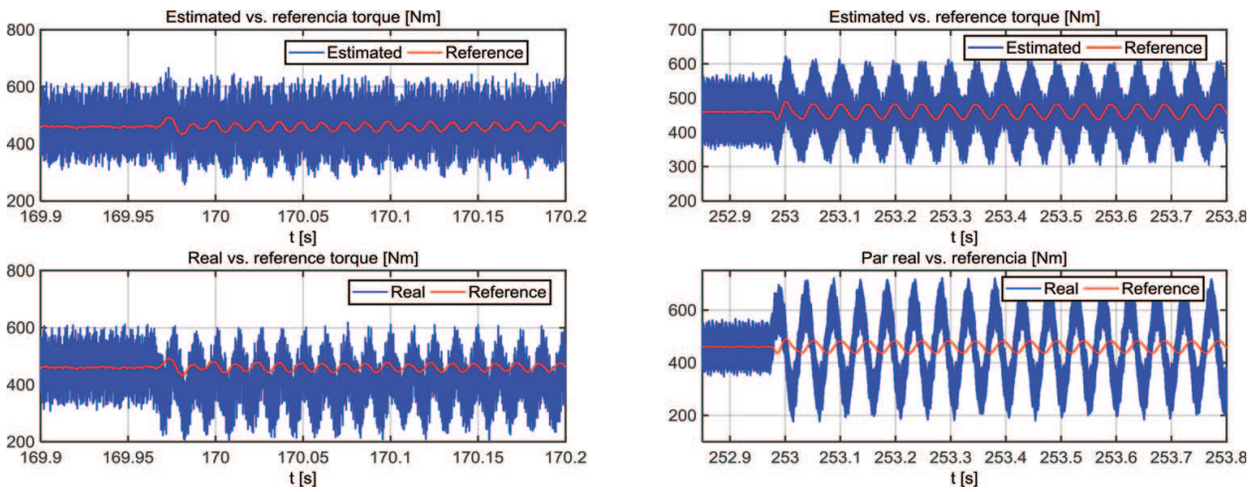


Figure 10. Reference, real and estimated torque with phase current sensors under +20% gain deviation (left) under 100A offset deviation (right).

torque, as it is shown in **Figure 10**. If the deviation is caused by an offset, the oscillation has the same frequency as the supply current. This frequency is twice the supply frequency when the fault mode is a gain deviation. Moreover, when the gain deviates, the torque is controlled below or above the reference value.

With this methodology, each fault mode has been characterized by different deviation levels. For example, as it is presented in **Figure 11** the relation between de gain deviation and the torque oscillations and deviations was obtained.

Table 3 shows the summary of the results.

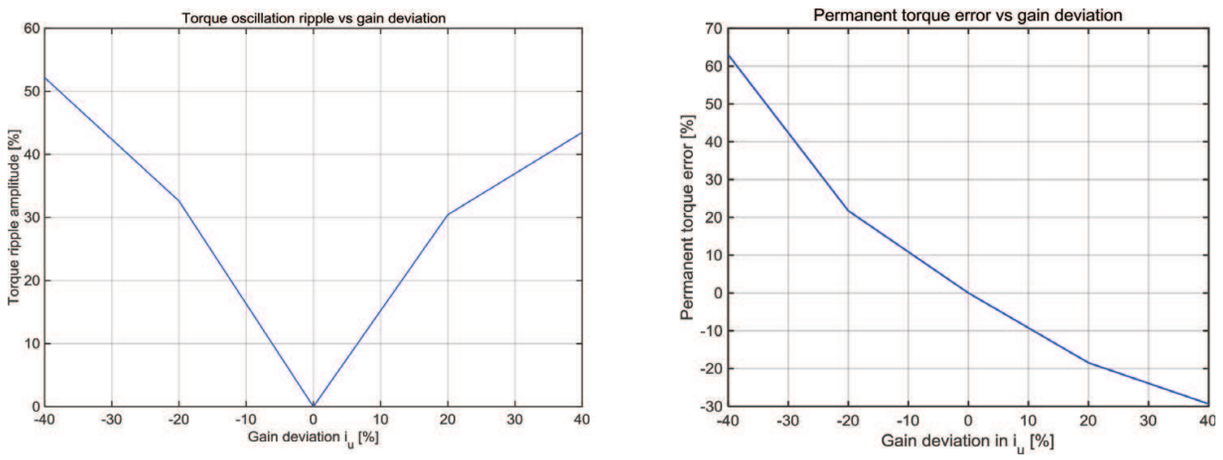


Figure 11. Phase sensor gain deviation due to phase sensor fault (left) torque ripple(right) torque permanent deviation.

Traction unit level effect	Effect 1	Oscillation in the torque and the bus voltage at twice the supply frequency of the motor. The quantitative amplitude of the torque ripple vs. the gain deviation is shown in **Figure 11**.
	Effect 2	Permanent torque error. Torque controlled below the reference value. The quantitative amplitude of the torque ripple vs. the gain deviation included in **Figure 11**.
	Effect 3	Overcurrents and overvoltages that activate the protections. Above ±35% of deviation, the inverter is disabled to assure component safety.
Train level effect	Effect 1	Comfort loss at low speeds. Low-frequency oscillations in the torque. Using the torque ripple curves, for each railway application, a quantitative evaluation of oscillations of the train linear acceleration could be computed by obtaining the maximum allowable torque ripple for a good comfort travel.
	Effect 2	Maximum acceleration capacity is decreased. Torque capacity lost. For each application, using the information of the train route, the overtime duration associated with a gain fault could be assessed.
	Effect 3	Availability loss of the one Traction Converter above ±35% of deviation of the gain fault.

Table 3. Quantitative FMEA for phase current sensor under gain faults.

5. Conclusions

In this chapter, a methodology for model-based HIL fault analysis was presented. Using models and real-time simulations, an improved quantitative FMEA for complex systems can be obtained. Following the described steps, a quantitative FMEA for railway traction systems was obtained as an example. Thanks to the data obtained from the models and the simulations, the effects of the faults are characterized in detail. The improved FMEA can be used as a reference document to improve designs, to implement new diagnostic functionalities or to elaborate new maintenance procedures.

Acknowledgements

This research work was supported by CAF Power & Automation. The authors are thankful to the colleagues from CAF Power & Automation, who provided expertise that greatly assisted the research.

Author details

Jon del Olmo*, Fernando Garramiola, Javier Poza and Gaizka Almandoz

*Address all correspondence to: jdelolmo@mondragon.edu

Electronics and Computing Department, Mondragon Unibertsitatea, Faculty of Engineering, Arrasate-Mondragón, Spain

References

[1] Jardine AKS, Lin D, Banjevic D. A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mechanical Systems and Signal Processing. 2006;**20**(7):1483-1510

[2] Farnsworth M, Tomiyama T. Capturing, classification and concept generation for automated maintenance tasks. CIRP Annals - Manufacturing Technology. 2014;**63**(1): 149-152

[3] Le Mortellec A. Proposition d'une architecture de surveillance active à base d'agents intelligents pour l'aide à la maintenance de systèmes mobiles-Application au domaine ferroviaire. Valenciennes: Université de Valenciennes et du Hainaut-Cambrésis; 2014

[4] Gandibleux J. Contribution a l'evaluation de surete de fonctionnement des architectures de surveillance diagnostic embarquees Application au transport ferroviaire. Valenciennes: Université de Valenciennes et du Hainaut-Cambresis; 2013

[5] Xue F, Yan W, Roddy N, Varma A. Operational data based anomaly detection for locomotive diagnostics. In: International Conference on Machine Learning, Models, Technologies and Applications. MLMTA. 2006. pp. 236-241

[6] European Commission, "Shift2Rail Joint Undertaking Multi-Annual Action Plan," Shift2Rail Joint Undertaking. Brussels. 2015

[7] Kabir S. An overview of fault tree analysis and its application in model based dependability analysis. Expert Systems with Applications. 2017;**77**:114-135

[8] Sharvia S, Kabir S, Walker M, Papadopoulos Y. Chapter 12 - Model-based dependability analysis: State-of-the-art, challenges, and future outlook. In: Software Quality Assurance. Boston: Morgan Kaufmann. 2016. pages 251-278

[9] Joshi A, Heimdahl MPE. Model-based safety analysis of simulink models using SCADE design verifier. In: SAFECOMP 2005 – International Conference on Computer Safety, Reliability and Security. LNCS. Vol. 3688. Berlin, Heidelberg: Springer. 2005. pp. 122-135

[10] Lisagor O, Pumfrey DJ, Mcdermid JA. Towards a practicable process for automated safety analysis. In: 24th International System Safety Conference (ISSC) Organized by The International System Safety Society. 2006. pp. 596-607

[11] Villacourt M. Failure mode and effects analysis (FMEA): A guide for continuous improvement for the semiconductor equipment industry. SEMATECH. 1992

[12] Liu H, Liu L, Liu N. Expert systems with applications risk evaluation approaches in failure mode and effects analysis : A literature review. Expert Systems with Applications. 2013;**40**(2):828-838

[13] Mikulak RJ, McDermott R, Beauregard M. The Basics of FMEA. 2nd ed. New York: CRC Press; 2008

[14] David P, Idasiak V, Kratz F. Reliability study of complex physical systems using SysML. Reliability Engineering & System Safety. 2010;**95**(4):431-450

[15] Sharvia S, Papadopoulos Y. Integrated application of compositional and behavioural safety analysis. In: Dependable Computer Systems. Vol. 97. Berlin, Heidelberg: Springer; 2011. pp. 179-192

[16] Belmonte F, Soubiran E. A model based approach for safety analysis. In: SAFECOMP 2012 – International Conference on Computer Safety, Reliability and Security. LNCS. Vol. 7613. Berlin: Springer; 2012. pp. 50-63

[17] Grunske L, Winter K, Ytapanage N, Zafar S, Lindsay PA. Experience with fault injection experiments for FMEA. Software – Practice and Experience. 2011;**41**:1231-1258

[18] Aizpurua JI, Muxika E. Model-based design of dependable systems: Limitations and evolution of analysis and verification approaches. International Journal on Advances in Security. 2013;**6**(1):12-31

[19] Sharvia S, Papadopoulos Y. Integrating model checking with HiP-HOPS in model-based safety analysis. Reliability Engineering & System Safety. 2015;**135**:64-80

[20] Lisagor O. Failure Logic Modelling: A Pragmatic Approach. York: University of York; 2010

[21] Papadopoulos Y, Maruhn M. Model-based synthesis of fault trees from Matlab-Simulink models. In: 2001 International Conference on Dependable Systems and Networks. Vol. 36. 2001. pp. 77-82

[22] Papadopoulos Y. Safety-Directed System Monitoring Using Safety Cases. York: The University of York; 2000

[23] Kabir S. Compositional Dependability Analysis of Dynamic Systems with Uncertainty. Hull: The University of Hull; 2016

[24] Bozzano M et al. ESACS: An integrated methodology for design and safety analysis of complex systems. In: Proceedings of the European Safety and Reliability Conference 2003, ESREL2003. 2003

[25] Akerlund O, Bieber P, Böde E. ISAAC, a framework for integrated safety analysis of functional, geometrical and human aspects. In: 3rd European Congress on Embedded Real Time System (ERTS). 2006

[26] Estefan JA. Survey of model-based systems engineering (MBSE) methodologies. In: International Council on Systems Engineering (INCOSE). 2008

[27] Terwiesch P, Keller T, Scheiben E. Rail vehicle control system integration testing using digital hardware-in-the-loop simulation. IEEE Transactions on Control Systems Technology. 1999;**7**(3):352-362

[28] Wang L, Zhang Y, Yin C, Zhang H, Wang C. Hardware-in-the-loop simulation for the design and verification of the control system of a series-parallel hybrid electric city-bus. Simulation Modelling Practice and Theory. 2012;**25**:148-162

[29] Isermann R, Schaffnit J, Sinsel S. Hardware-in-the-loop simulation for the design and testing of engine-control systems. Control Engineering Practice. 1999;**7**(5):643-653

[30] Poon JJ, Kinsy MA, Pallo NA, Devadas S, Celanovic IL. Hardware-in-the-loop testing for electric vehicle drive applications. In: Conference Proceedings – IEEE Applied Power Electronics Conference and Exposition – APEC. 2012. pp. 2576-2582

[31] Wu J, Dufour C, Sun L. Hardware-in-the-loop testing of hybrid vehicle motor drives at Ford Motor Company. In: 2010 IEEE Vehicle Power and Propulsion Conference (VPPC 2010); 2010

[32] Baccari S et al. Real-time hardware-in-the-loop in railway: Simulations for testing control software of electromechanical train components. In: Railway Safety, Reliability and Security: Technologies and Systems. 2012. p. 487

[33] Alvarez-gonzalez F, Member S, Griffo A, Wang J, Member S. Real-time hardware-in-the-loop simulation of permanent magnet synchronous motor drives under stator faults. IEEE Transactions on Industrial Electronics. 2017;**64**(9):6960-6969

[34] Chung D-WCD-W, Sul S-KSS-K. Analysis and compensation of current measurement error in vector-controlled AC motor drives. IEEE Transactions on Industry Applications. 1998;**34**(2):340-345

[35] Jung H, Kim J, Kim C, Choi C. Diminution of current measurement error for vector controlled AC motor drives. IEEE Transactions on Industry Applications. 2006;**42**(5): 1249-1256

IntechOpen

IntechOpen