Finding Faults: a scoping study of Fault Diagnostics for Industrial Cyber-Physical Systems

Barry Dowdeswell[a,*], Roopak Sinha[a], Stephen MacDonell[a]

[a] *Auckland University of Technology, Auckland, New Zealand*

Abstract

Context: As Industrial Cyber-Physical Systems (ICPS) become more connected and widely-distributed, often operating in safety-critical environments, we require innovative approaches to detect and diagnose the faults that occur in them.

Objective: We profile fault identification and diagnosis techniques employed in the aerospace, automotive, and industrial control domains. Each of these sectors has adopted particular methods to meet their differing diagnostic needs. By examining both theoretical presentations as well as case studies from production environments, we present a profile of the current approaches being employed and identify gaps.

Methodology: A scoping study was used to identify and compare fault detection and diagnosis methodologies that are presented in the current literature. We created categories for the different diagnostic approaches via a pilot study and present an analysis of the trends that emerged. We then compared the maturity of these approaches by adapting and using the NASA Technology Readiness Level (TRL) scale.

Results: Fault identification and analysis studies from 127 papers published from 2004 to 2019 reveal a wide diversity of promising techniques, both emerging and in-use. These range from traditional Physics-based Models to Data-Driven Artificial Intelligence (AI) and Knowledge-Based approaches. Hybrid techniques that blend aspects of these three broad categories were also encountered. Predictive diagnostics or *prognostics* featured prominently across all sectors, along with discussions of techniques including Fault trees, Petri nets and Markov approaches. We also profile some of the techniques that have reached the highest Technology Readiness Levels, showing how those methods are being applied in real-world environments beyond the laboratory.

Conclusions: Our results suggest that the continuing wide use of both Model-Based and Data-Driven AI techniques across all domains, especially when they are used together in hybrid configuration, reflects the complexity of the current ICPS application space. While creating sufficiently-complete models is labour intensive, Model-free AI techniques were evidenced as a viable way of addressing aspects of this challenge, demonstrating the increasing sophistication of current machine learning systems. Connecting ICPS together to share sufficient telemetry to diagnose and manage faults is difficult when the physical environment places demands on ICPS. Despite these challenges, the most mature papers present robust fault diagnosis and analysis techniques which have moved beyond the laboratory and are proving valuable in real-world environments.

Keywords: Industrial Cyber-Physical Systems, Faults, Automotive, Aerospace, Avionics, Industrial Control.

1. Introduction

Industrial Cyber-Physical Systems (ICPS) are mechanisms that augment their computing elements with sensors and electromechanical actuators that allow them to interact with the physical environment they operate in [1]. By evaluating feedback, both from other ICPS they are connected to and from their local industrial environment, they perform a wide range of valuable and often hazardous tasks [2]. Varying widely in complexity and scale, they are found controlling equipment in aircraft, automobiles and factories.

ICPS should be thought of as being more than just computing devices. They form entire systems, viewed as a collection of seamless entities, including their multiple electrical, mechanical and computing subsystems. This homogeneity makes them fundamentally different to the earlier embedded Programmable Logic Controllers (PLCs) that were first used on General Motors automotive assembly lines in the 1960's [3, 4]. These devices controlled only the machinery they were installed or *embedded* in. They were seldom connected to other plant equipment and the sensors they used were often simpler devices such as limit switches, weight sensors or strain gauges. In contrast, modern ICPS act with higher degrees of autonomy than these earlier embedded systems, relying on sensors and actuators that often incorporate their own local data processing and conditioning. ICPS are therefore able to make control decisions based on their perception of their environment, driven by much deeper interaction with the physical characteristics of the world they operate in [5, 6]. Earlier embedded systems seldom featured this degree of complexity and capability.

Contemporary ICPS continue to present intriguing challenges as they have become increasingly more complex. Widely-distributed and now often physically-separated, ICPS are being used to create the Industrial Internet of Things (IIoT), where collections of discrete devices cooperate intelligently to perform large-scale industrial tasks [7]. ICPS differ from Cyber-Physical Systems (CPS) used in consumer or medical devices primarily in terms of their scale [8, 9], security [10, 11] and safety-critically [12, 13]. ICPS used in Smart Grids rely on industry-standard interfaces and sophisticated communications. They manage reliable power distribution across wide geographical areas by co-operating and co-ordinating the operations of the devices that control each sub-station. Examples of advanced ICPS include NASA's Mars rover *Curiosity* which operates semi-autonomously, controlled by one of the most remote ICPS ever deployed on another planet [14, 15].

Detecting and diagnosing ICPS faults quickly and correctly has become imperative to ensure they are fully-operational at all times. We have learnt how to rely on ICPS more and more to manage complicated and often safety-critical tasks. Today, undetected failures in ICPS are not just costly: in safety-critical or hazardous conditions they can be life-threatening [16, 17]. For example, ICPS in the aircraft and aerospace sector rely on accurate readings from sensors to inform guidance, vehicle health and maintain stable flight control. They do this with a degree of reliability, precision and repeatability that human pilots can no longer achieve alone [18, 19]. Similarly, in the automotive sector, vehicles have become increasingly reliant on large local networks of sophisticated subsystems such as anti-skid breaking and fuel-efficient engine controls [20, 21]. Within each subsystem, information is gathered using sensors designed to capture one or more physical characteristics of the local environment, both within and outside the vehicle. The overall operation of a typical ICPS is, therefore, reliant on the co-operative behavior of each of its specialized subsystems, each one dedicated to specific aspects of the vehicle's safe operation and reliability [22, 23, 24].

1.1. The focus and contributions of this study

We identified, categorized and analyzed fault identification and diagnosis strategies for ICPS employed across the aerospace, automotive and industrial control domains. Our goal was to present a snapshot of fault diagnosis as it is practiced today. We surveyed the differences in the approaches that have emerged in each sector and how they address the needs they describe. Our survey provides a guide to applicable techniques for designers seeking to implement fault identification, diagnosis and management into their ICPS.

We chose the aerospace, automotive and industrial control domains primarily because the ICPS they rely on must operate faultlessly for extended

*Corresponding author
Email address: barry.dowdeswell@aut.ac.nz (Barry Dowdeswell)

periods of time, often in close proximity to humans [25]. These sectors also exhibit high levels of integration between their computational cyber elements and the sensors that provide the information that all operational decisions are made on. For example, ICPS in automobiles now sense the position of highway lane markings accurately, extract information from signs and determine the relative positions of adjacent vehicles.

We were also interested in the similarities and differences in fault diagnostic approaches that have emerged in these three safety-critical sectors over the period we studied. The scope of our study was deliberately limited to representative domains that have become highly-dependent on ICPS to manage mission-critical tasks. It is in these sectors that we would expect to find that diagnostics are highly-advanced and widely-used. However, we chose not to include the medical sector in this study. Medical ICPS have distinctive biological characteristics, regulatory requirements and a scale that is worthy of a separate study later. We also excluded cyber-physical devices in the Consumer Electronics sector from our study. They are driving a large and expanding part of the market however they are often less complex than the ICPS in our chosen sectors and the tasks they manage are usually less safety-critical.

A scoping study was used to map the key approaches that underpin fault diagnosis in these sectors and the sources of both theory and case studies available from practitioners [26, 27]. We framed our study via three research questions:

RQ1: *What are the most common and widely-used fault identification and diagnosis techniques employed in ICPS in the aerospace, automotive and industrial control domains?*

RQ2: *What relative levels of maturity have the techniques identified in RQ1 achieved when assessed using a systematic scale that is applicable to these domains?*

RQ3: *What research gaps and challenges in ICPS fault identification and diagnosis are being highlighted in the literature surveyed to answer RQ1?*

This scoping study seeks to provide a thorough and systematic overview of the fault identification and diagnosis techniques currently in use in our sectors of interest. It profiles the diagnostic approaches we encountered and the techniques that are being used in different situations. By applying a systematic classification to each technique encountered, we are able to estimate the relative level of maturity of each approach, highlighting those which are being applied successfully in real-world environments.

1.2. How this paper is organized

Section 2 explores briefly what a ICPS fault is and the terminology used to describe the various stages in a fault management methodology. Section 3 then details the survey data capture and analysis protocol our scoping study employed. While scoping studies do not usually include assessments of the quality of studies uncovered, we chose to adapt and employ the NASA Technology Readiness Level (TRL) as an qualitative scale to compare the relative maturity of the fault diagnosis techniques we encountered [28].

Section 4 presents the results of the scoping study, mapping the fault diagnosis methodologies described in the papers that were included in this study. Finally, Section 5 presents our conclusions, briefly examining those studies that demonstrated the highest TRL. These exemplars discuss fault diagnostic techniques that have moved beyond the laboratory and are being applied in the real world.

2. Background - what is fault diagnostics?

ICPS bridge the connection between their "cyber" software, sensor and actuator hardware parts and the "physical" world they inhabit. Figure 1 illustrates the two distinct classes of devices that mediate communication across this divide for a warehouse package-handling robot. A *sensor* is a device that can convert an environmental characteristic such as proximity, pressure, temperature or light levels into an electrical signal that can be processed by a computer [29]. In contrast, an *actuator* is a mechanical device that can receive an electrical signal from a computer and cause a change, often as a result of moving something in its environment [30]. Motors are special classes of actuators that create movement, such as the mechanism that moves the package off the parcel tray once the robot has arrived at its destination.

Normal behavior for an ICPS such as this warehouse robot is to pick up packages, navigate reliably

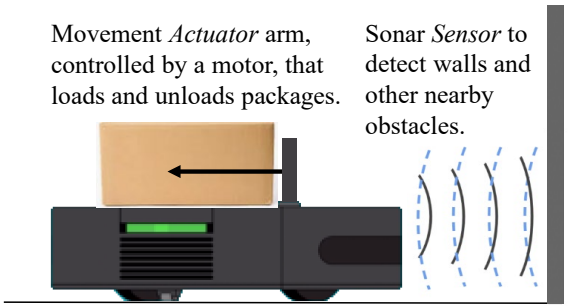

Figure 1: Sensors and Actuators for a Warehouse Robotic Package Handler.

and efficiently to another location, and then unload its cargo. The robot's activities rely on receiving inputs from its sensors and being able to co-ordinate the movements of its actuators to complete tasks that achieve previously-defined goals. Our example robotic package handler has pre-defined patterns of behavior that enable it to traverse warehouse aisles, locate shelves and deliver packages to specified locations. While it is working, it can detect both obstacles and humans, navigating safely around them.

The difficulty inherent in this interaction between the cyber and physical parts of an ICPS often results in faults occurring. Any change in the way that an ICPS operates that leads to unacceptable behavior or degraded performance is defined as a *fault* [31]. For example, the wheels of the robotic package handler might become entangled with warehouse rubbish from the floor and stop rotating. If the control program detects this problem, it can respond with an appropriate behavior, perhaps stopping and requesting a human for assistance. This sort of situation is not a fault: it is the ICPS managing its behavior in a way that is appropriate. In contrast, not detecting that it cannot move properly and carrying on regardless is a fault since the ICPS did not recognize the issue and change its behavior accordingly. Similarly, failing to detect the edge of stairs and falling down them is unacceptable behavior, possibly due to a faulty precipice sensor. Lee and Seshia comment that it is not enough to separately understand both the computational and electromechanical elements [30]. Rather, it is at the *intersection* of the cyber and the physical that the most challenging fault scenarios emerge.

2.1. Fault identification, diagnosis and management concepts

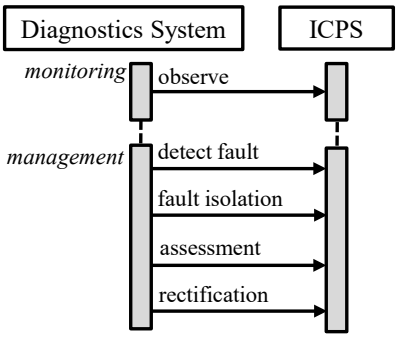

Figure 2: Sequence Diagram of Diagnostic Activities for a generalized ICPS.

Figure 2 illustrates the activities in a generalized fault management strategy. *Fault diagnosis* is primarily the analysis of the activities or interactions of an ICPS while it is being observed operating within the environment it is deployed in [32]. Milis [33] and Haririchi [34] define *Fault Detection* as the capability of a device to determine the difference between normal and abnormal modes of operation. This may be an after-the-fault examination of a system that has failed or a more proactive monitoring of the system's behavior, watching for issues before they occur. The evidence of a fault is therefore exhibited as unacceptable behavior or degraded performance. Hence, the previous example of the warehouse robot not stopping and requesting assistance is indicative of a fault, either on the part of a sensor or the ICPS software. Fault detection is recognizing that something is wrong but this realization alone does not necessarily categorize or analyze the problem. The purpose of fault detection is to trigger a response by the ICPS to take appropriate action by first recognizing abnormal activity. When faults are detected, the ICPS could just halt. However that is not always a viable strategy if the task the ICPS is performing is critical to some party other than itself.

Detecting faults is the first stage of a *Fault Management Strategy* [35]. Detecting a fault should start a multi-step process that attempts to diagnose and potentially correct problems so that the ICPS can resume operating at optimal levels. This implies that the ICPS needs to be able to hold a dynamic representation of what normal behavior is so that it can recognize misbehavior.

Fault management strategies include *Fault Isolation*, which is the process of accurately identifying the location of the fault and its nature [34]. This can be difficult to determine reliably in large systems that contain many interconnected subsystems. Hence, fault isolation includes the analysis of multiple possible fault sites to determine the nature of the real, underlying fault. The fault symptoms presented, or *Fault Evidence*, often include secondary system misbehaviors that are the result of the primary fault but which are not the root cause. Bradatsch [36] defines *fault latency* as the time between the occurrence of the fault and its recognition by the device's fault management system.

Once a list of possible fault candidate locations has been identified, the next step is *Fault Assessment*. This examines evidence and seeks to determine the most likely root locations of the fault, as the problem may include a compound failure located at multiple, distinct points [37, 38]. This leads to the final stage of the diagnosis, *Fault Risk Assessment*. Not all faults are important enough to require intervention if the system is able to operate satisfactorily in a degraded condition. Ghadhab [39] discusses the use of "limp-home" strategies for automobiles that allow them to continue to operate safely in a degraded mode until they can be repaired.

Mortellec et al. [40] provide a wider perspective on what an ideal diagnostic system should provide. Besides being able to uniquely identify the true location and nature of a fault, diagnostic systems must be able to communicate effectively with other systems to help facilitate fault rectification. They must deliver their findings rapidly, especially in safety-critical situations. Finally, it is paramount that they must not report false information.

3. Research method

Scoping studies are one method of rapidly mapping the key concepts that appear within a research area [27, 41]. Often smaller in scope than full systematic reviews or mapping studies, scoping studies allow the breadth of coverage and the depth of the information extracted to be tailored to address research questions appropriately [42, 43]. Arksey and O'Malley [27] and Antman et al. [44] both explain how scoping studies are an appropriate way to quickly capture and present both the available information and the gaps. They can also be used to focus and inform later literature searches for practitioners when they do not have time to perform a thorough initial analysis themselves. Our scoping study protocol follows the four steps of framing research questions, identifying relevant studies, analysis and then presentation of the results as outlined by Arksey and O'Malley [27] and refined by Cacchione [26].

3.1. Step One: Framing our research questions

Scoping studies are also effective where the researchers do not have a single or highly-focused research question that they are seeking to answer [45]. The research questions detailed in Section 1.1 were designed to identify, highlight and categorize practical fault recognition and diagnosis techniques that have been found to be effective both in laboratory studies and in the field. Since this scoping study examines multiple yet similar sectors with potentially differing needs, understanding the focus and spread of the challenges and how they are being addressed should be of interest to practitioners who are designing their own ICPS.

3.2. Step Two: Identification of relevant studies

Scopus was used to search for papers that included the terms "cyber-physical", "aerospace", "aircraft", "automotive", "industrial" and "manufacturing" for the fifteen-year period from 2004 to 2019. This starting period for the search was chosen since it coincides with the emergence of the term *Cyber-Physical System*. The first use of the term can be traced to the National Science Foundation meetings in 2001 that discussed networked embedded control systems [46, 47]. In 2006, Lee [48] highlighted the implications of these discussions about connecting discrete embedded systems. Prior to this, Wiener's earlier pioneering work on cybernetics informed much of the thinking on control systems theory, arguably setting the agenda for later CPS research [49].

3.3. Step Three: Study selection and classification

From an initial pool of 1,700 candidate papers returned by our queries, we performed a pilot study on thirty of these papers. Particular papers were chosen primarily because they contained well-written explanations of fault identification and diagnosis techniques that provided valuable background information. These were used to create an initial set of fault identification or diagnosis

Table 1: Fault Identification and Diagnosis Classification Categories.

Category	Example Sub-Categories
Physics-Based Modeling approaches	Kalman Filters, Markov Models, Fault Trees, Other Stochiometric processes, Model Validation/Invalidation, Monitor-based oracles
Data-Driven AI and Machine Learning approaches	Artificial Neural Networks, Machine Learning, Fuzzy Logic, k-Nearest Neigbour, Big Data/Data Mining
Knowledge-based approaches	Bayesian Decision Theory, Binary Trees, Petri nets, Network Message Analysis, Expert Systems

approach classifications that identified both broad conceptual differences and a list of specific techniques applicable to those approaches. Table 1 lists these categories. RQ1 asks what the nature of fault identification and diagnostics is within our chosen domains. The broadest primary classifications that emerged divided the approaches into three high-level categories that helped to delineate the research activity. We encountered Physics-Based Model-Driven diagnostics, Data-Driven Model-Free Artificial Intelligence (AI) techniques and Knowledge-Based graph approaches. Hybrid techniques that blend aspects of these approaches were also encountered. The similarities and differences between these broad classes are profiled in more detail in Section 4.

To examine the specific fault-finding methods found within our three primary approaches, subcategories were created to identify the characteristics of each technique. Beyond these classifications, trends such as Predictive diagnostics or *prognostics* became of particular interest to us since this approach featured more widely than we initially expected. The complete list of studies, classified according to these category codes, is available via this link [50].

Our publication sources included peer-reviewed journal papers, conference papers and open-access journals. Outside of the academic databases, we also sought technical publications and position papers written by industry-based authors with current, practical experience in their field. Examples include automotive-industry papers from SAE International (https://www.sae.org) and aerospace papers written by NASA researchers or their industry partners (e.g. Lockheed, Boeing). While the papers published by non-academic sources such as SAE were not necessarily peer-reviewed, they often contained detailed results from specific case studies. Arksey and O'Malley stress the importance of including such "grey matter" in scoping studies.

Our minimum inclusion criteria for a study required it to present and explain the fault identification or diagnosis approach that was being applied. We also sought papers that included case studies demonstrating the effectiveness of their techniques. Many papers were excluded because they only mentioned "faults" or "diagnosis" as an aspect of the nature of ICPS without presenting specific examples.

3.4. Step Four: Analyzing and Presenting the Data

During the first phase of the analysis, the classification categories allowed us to perform a thematic analysis [51, 52]. Each of our categories and subcategories represent a technique or approach used or proposed by a practitioner as a way of identifying, diagnosing or rectifying a fault [53]. The analysis also included examining where diagnostic research is focused in each sector and is presented in Figures 5, 6 and 7.

Scoping studies do not usually attempt to assess the quality of the studies uncovered [26]. However, we chose to adapt and apply a qualitative scale during the classification phase to rank the relative level of maturity of the diagnostic techniques we found. Each study was evaluated using the NASA Technology Readiness Level scale [54, 55]. This is a systematic metric for assessing how mature a particular technology is that is now widely used in both aerospace and defense for technology planning. The TRL has been progressively refined since the 1980's through its use at both NASA, ESA and the US military [28, 54, 56]. It is now embodied in the standard ISO 16290 [57]. In 2014, the European Association of Research & Technology Organisations (EARTO) identified an increased use of the TRL amongst its members as a planning tool to manage innovation [58].

RQ3 sought to identify research gaps, especially those exhibited amongst the most promising approaches. The TRL provide criteria for assigning a classification between TRL 1, representing basic principles being observed or reported through to TRL 9, characterized by technologies proven in real

environments that are ready for widespread adoption. We calibrated our fault diagnostic TRL descriptions using the approach of Terrile et al. [59]. They note that the relative TRL steps are not linear with the steepest steps being in the range TRL 6 to 8. Section 5 details the four divisions we chose to classify studies into an appropriate range. The granularity of the resulting TRL categories allowed us to distinguish between studies that were purely theoretical and those that were profiling fault diagnostic techniques that are being applied in live environments. Table 5 illustrates the fault diagnostic level characterizations we adapted from the NASA categories.

By the end of the classification and analysis phases we had identified fourteen studies that could be ranked at the highest TRLs between 7 and 9. These report mature, field-proven fault-finding and diagnostic strategies that have been deployed in production environments. In those papers, we should expect to see state-of-the-art exemplars that detail how ICPS respond to and recover from fault situations they encounter.

3.5. Threats to validity

In scoping surveys such as ours, the primary threats to the validity are our choice of which papers to include and our thematic classifications. Surveys are by definition *secondary studies* that report broad, summarized characteristics of *primary studies*, the source papers published that present research about an area of interest [60]. As distinct from Systematic Literature Studies (SLS) that provide highly-detailed evaluations of a smaller set of papers [61], scoping studies show where research activity is concentrated and what aspects of a topic are attracting interest, often examining a larger number of papers in less depth.

Internal validity is concerned with the risks that might lead to an incorrect conclusion [62]. This was partially mitigated during the analysis phase by ensuring that each primary paper was initially scanned to determine if it did indeed contain one of our classification classes. For some classes, a list of appropriate synonyms was built iteratively. Our inclusion criteria for a paper included a check to see if groups of related terms were present. The classifications defined in Section 3.3 such as "Model-Based" were expected to show up where models were discussed. However, within the same paper, the classification of "Model-Free" was expected to be applicable when discussions featured AI, Neural Networks, Markov approaches or Data-Driven techniques. Intellectual property restrictions on what can or cannot be published may also be a contributing factor to the amount of detail that can be published about implementations. This was considered when evaluating the relative TRL across sectors.

4. Diagnostic Techniques in Industrial CPS

Examples of fault identification and diagnostic methods examined initially during the pilot study were described by authors as having evolved along three primary pathways: Physics-Based modeling and analysis frameworks, Data-driven or Model-free AI techniques, and Knowledge-Based graphical approaches [63]. While classifying our studies, we also identified hybrid approaches which blend aspects of these methods.

4.1. Physics-Based, Model-Driven Diagnostics

Modeling is used by designers to gain a deeper understanding of a system. By creating models that imitate the physical characteristics of the ICPS components, they can explore the interaction the sensors and other physical devices have with the cyber parts of the ICPS [30]. Physics-Based Modeling techniques for diagnostics rely on consistency checks against these models. These detect the differences between the telemetry captured from the live ICPS and the values predicted by the model. Table 2 summarizes physics-based modeling diagnostic techniques across our survey domains.

Consistency checks use data captured by observers who filter the individual readings to distinguish between noise caused by telemetry errors and values that indicate faulty behavior [118]. These differences will often be small but seldom non-zero when the ICPS is performing within acceptable tolerances [21]. Techniques for determining when an aspect of a model is invalidated were discussed in 48% of papers, especially in the industrial control domain. Both Kalman Filters and Markov Models were discussed as ways of recognizing model invalidation. These techniques implement observers that can process sequential measurements that vary over time. Kalman Filters are more applicable when the range of possible readings is highly-linear. They apply recursive algorithms where weighted-averages are used to estimate the next value. They work well in noisy environments that produce sequences

Table 2: Physics-based Modeling fault identification and diagnosis techniques across all sectors.

Technique	Aerospace	Automotive	Industrial	All	Publications
Kalman Filters	23%	0%	14%	13%	[23] [64] [65] [66] [67] [68] [69] [70]
Markov models	17%	8%	9%	11%	[39] [65] [19] [71] [72] [73] [74] [75] [76]
Fault Trees	17%	19%	5%	14%	[22] [77] [78] [79] [63] [80] [81] [12] [82] [83] [84] [85] [74] [86] [87]
Model invalidation	43%	41%	64%	48%	[88] [8] [89] [90] [91] [64] [77] [78] [79] [63] [80] [81] [92] [93] [94] [95] [71] [96] [97] [98] [99] [100] [101] [102] [103] [104][105] [14] [106] [107] [108] [109] [110] [111] [112]
Monitor-based Oracles	10%	11%	9%	10%	[8] [113] [100] [114] [115] [116] [69] [117]

of unreliable readings. Zolghadri et al. describe an implementation of a Kalman filter to detect jamming of a flight control surface by filtering the error signal before it is processed by the on-board avionics [68]. The authors explain how the number of sensors providing input to the model affects both the design and worst-case performance. Tuning the model parameters requires trade-offs against the real-time capacities of the diagnostic systems that rely on the model. Shraim et al. discuss fault management for quadrotor unmanned vehicles to improve rotor positioning accuracy [23]. Unmanned Aerial Vehicles (UAV) require real-time fault tolerance since they now rely on autonomous, sensor-driven stability control that is no longer managed entirely by the pilot. The models used have to take into account the complex aerodynamic characteristics of the UAV. Dearden et al. discuss similar aspects of autonomous operation, describing fault diagnostics for Mars Rovers where Kalman filtering provides situational awareness to indicate fault conditions [75]. They contrast the number of sensors required to manage rover operations with the low computational power available to perform fault identification using multiple sub-system models.

In contrast, Markov models are used to model non-linear, randomly changing systems with discrete states. A dynamic model is Markov or has the Markov Property if the future state of a system depends only on a limited number of previous states. Markov Chain and Markov Decision processes rely on observing the full set of values or *states* for the aspect of the ICPS that is being diagnosed. In contrast Hidden Markov Models operate where the sequential state of a system is not fully observable. Kunst et al. profile damage propagation through ICPS using Hidden Markov models [19]. Similarly, Windmann and Niggeman [65] and Ribero et al. both apply Markov Models to monitor industrial processes and identify faults as they propagate.

Fault Trees are a way of modeling all reasonably-probable fault scenarios [22]. They are tree structures that facilitate a top-down, systematic approach to identify chains of possible faults. Logical operators can be applied to nodes to identify likely fault pathways. Fault trees are usually considered to be knowledge-based approaches but they were most often encountered in studies that employed hybrid approaches. Mohre et al. demonstrate correlations between fault tree nodes and compositional safety analysis models [12]. Kassmeyer et al. apply fault trees to track fault scenarios across multiple automotive feature variants [86].

Across all sectors, a wide range of specialized Model-Invalidation approaches were encountered, both theoretical and in-use. Provan [88] discusses how acceptable inputs can be modeled, an important pre-requisite to detecting misbehavior. Monitors [117] are code within a fault identification system that is responsible for detecting anomalous situations or behavior. Similarly, Monitor-Based Oracles provide ways of both capturing and evaluating possible fault occurrence [8, 113, 100].

Formal modeling languages including the Architecture Analysis & Design Language (AADL) [119] and Modeling and Analysis of Real-time and Embedded Systems (MARTE) [120] model ICPS during their design phases. AADL originated in the aerospace sector to model embedded systems and has now found wide use in the automotive domain. MARTE extends the UML to provide similar capabilities. Huang et al. [121] describe a simulation platform modeled in AADL that allows transient faults to be evaluated. Khlif and Shawky demon-

strate how to use AADL to design co-simulations that are easier to diagnose later [102]. Shulte proposes a state machine architecture for fault detection based on SysML [22]. However, no papers in the survey discussed production ICPS implementations that employed either AADL or MARTE models from the design phases directly. Procter and Feiler present an introduction to the the AADL EMV2 Error Library where they discuss the use of an error ontology during modeling [122]. We searched the literature for examples of the use of EMV2 in production fault diagnostic systems beyond the design phase but found few applicable examples. Lu et al. discuss redundancy approaches using AADL and EMV2 however their work does not demonstrate how to apply their fault trees in a production, real-world example [123]. Similarly, Zhang et al. discuss the design of fault tolerant systems using EMV2, but it is applicable only to early-stage modeling [124].

Creating and maintaining models is labor-intensive. Many of the techniques rely on detecting situations where a model is invalidated. However, Milis et al. [33] highlight the amount of effort needed to calibrate models. Provan [88] also discusses two practical impediments to effective model-based diagnosis: the failure to integrate diagnostic modeling early enough in the requirements process and ambiguities in the models themselves at run-time.

4.2. Data-Driven fault diagnostics

Data-Driven diagnostic techniques employ training and learning to forge a representation of the system's behavior [21]. Unlike Physics-Based models, Data-Driven fault detection does not rely on the existence of pre-built models. This approach is preferred when the ICPS can provide telemetry that contains enough information to distinguish between either normal or degraded operations. AI fault diagnosers make sense of that information by using discriminating logic that copes with the changes seen in the ICPS as they occur. This ability to make intelligent decisions distinguishes AI from machine learning, which involves ICPS learning without being explicitly programmed. Milis [33] discusses cognitive agents that apply expert reasoning to mimic the behavior of human experts.

Artificial Neural Networks (ANN) [131, 11, 127] and pattern-recognition algorithms [144] are illustrative of data-driven techniques. Since they do not rely on static, pre-built models as reference points, they remove the need to keep the model up-to-date as the system evolves. Data-Driven diagnostic systems learn behaviors through training. Detection logic allows them to compare current values with previously learnt values [101]. Hence these Model-Free methods do not have to completely understand the underlying architecture of the system being examined [132].

Data-Driven approaches often scale better than Model-Based techniques [9, 8]. As long as sufficient computational resources are available, Data-Driven techniques work as effectively with a large number of sensors as they do with a few [132]. Since they construct knowledge representations dynamically, they are often easier to update than formal models [133].

Unlike Model-Based methods, Data-Driven approaches do not assume the probabilistic distributions of sampled values that Markov processes rely on [9]. Similarly, AI methods, including machine learning, do not rely on processes being stochastic or random. The trade-off is that while Physics-Based models are labor-intensive to create, model-free techniques require large example data sets to train the observers [125, 126, 127]. Iverson et.al [132] explain that for avionic ICPS, large volumes of archival sampled values are collected during routine operations that are can be used for training neural networks.

Fuzzy logic employs truth values that are real numbers between zero and one rather than being boolean [13, 38, 147]. This allows decisions to be made about non-numerical or imprecise data from ICPS, stored in structures called fuzzy sets. These sets represent partial truths and decisions are made by arriving at a consensus. Fuzzy logic algorithms are able to re-evaluate thresholds for situations where values are expected to change dynamically as the system is being observed. Song [148] discusses recognizing faults using threshold predictions. Each sampled value is checked to see if it falls within a range defined by the previous value read.

Condition monitoring allows Data-Driven fault observers to obtain real-time data about the ICPS they are monitoring. These data points replace the reference values that pre-built Model-Based solutions rely on since AI and machine-learning approaches are model-free [9]. Lee et al. [2] and Fleischmann et al. [152] describe these techniques in terms of system health monitoring. Where deviations from the norm are observed, the result is

Table 3: Data-driven A.I. Model-Free fault identification and diagnosis techniques across all sectors.

Technique	Aerospace	Automotive	Industrial	All	Publications
Artificial Neural Networks	54%	50%	47%	50%	[33] [11] [125] [126] [127] [21] [128] [129] [130] [131] [128] [132] [133] [134] [135] [136] [137] [138] [139] [140] [141]
Machine Learning	38%	17%	18%	24%	[9] [142] [24] [21] [133] [134] [143] [144] [145] [146] [139]
Fuzzy Logic	8%	25%	18%	17%	[13] [38] [147] [148] [138] [135] [149]
Big Data	8%	0%	18%	10%	[150] [151]
Condition Monitoring	8%	8%	24%	14%	[9] [2] [152] [89]

similar to the model-invalidation discussed earlier. Wang et al. discuss this in the context of cloud computing and predictive maintenance [89].

Wang et al. [153] caution against over-reliance on AI approaches. They suggest that given the complexity of some fault scenarios, the conclusions drawn by data-driven systems may not be sufficiently robust enough to be free of false positives and negatives. However, Iverson et al. profile fault finding for the International Space Station (ISS), reporting that when a large amount of nominal data is available, Data-Driven systems can become highly effective at detecting anomalies [132].

4.3. Knowledge-Based approaches

Knowledge-Based approaches are applicable where large amounts of historical data are available. The underlying dependencies that define the system are derived from these sources using a range of techniques. All fault diagnosis systems need to observe real-time data, basing their evaluations on either qualitative or quantitative aspects of the telemetry. However, only knowledge-based approaches utilize significant amounts of historical data to inform their classifiers [2]. Unlike Data-Driven AI approaches, Knowledge-Based methods do not require pre-classified training sets. Rather, they mine the historical data using statistical methods. Chen et al. explain the value of historical information gathered from experts in building knowledge bases to inform current fault diagnoses [38].

The resultant dynamic models they construct are represented using dependency graphs. Petri nets are directed bipartite graphs where nodes represent discrete fault events that may occur. The graph arcs define possible transitions between states [158, 159].

Bayesian Belief Networks are knowledge-based directed graphs that model probabilities [38, 154]. Each node represents a step in a cause and effect chain with a conditional probability. While observing, the fault system updates the probability at a node when new information is available. Hence, Bayesian networks can provide both diagnostic and predictive evaluations.

Binary Decision Diagrams are directed acyclic graphs. Waszecki et al. [156] encode observation patterns extracted from messages exchanged by automotive ECUs to capture fault scenarios that can be evaluated during diagnosis. Network message analysis also complements other knowledge-based approaches, either as a carrier of fault messages or as an indicator of misbehavior [148]. Schweppe et al. [160] discuss the Automotive Keyword Protocol ISO 14230:2000 [161], a widely-accepted standard for analysing faults via network messages exchanged over a vehicles CAN bus. Pons et al. [157] outline a similar approach using Causal Graphs rather than Binary Decision Diagrams.

4.4. Hybrid fault diagnostic approaches

Hybrid approaches that blend techniques from any of the three broad approaches were encountered in 14% of the papers but featured in 19% of all industrial control studies. Hybrid techniques skewed the overall ratios of our three primary categories since practitioners can adopt any combination of methods to create their fault identification and diagnostic methodologies. Figure 3 illustrates the spread of Hybrid approaches across our three domains. Lee et al. [89] employs Model Invalidation from the Physics-Based Modeling category with Condition Monitoring from the Data-Driven AI category in an intelligent manufacturing scenario. This allows their system to analyze and predict faults from patterns shared via a cloud-based system. The system is implemented using intelligent agents. Chen et al. [38] combine Bayesian

Table 4: Knowledge-Based fault identification and diagnosis techniques across all sectors.

Technique	Aerospace	Automotive	Industrial	All	Publications
Bayesian Networks	0%	17%	43%	31%	[91] [38] [154] [155]
Binary Decision Trees	0%	33%	0%	15%	[156] [157]
Petri nets	0%	0%	43%	23%	[158] [159]
Network Message Analysis	0%	16%	29%	62%	[156] [160] [148] [143]

networks with Fuzzy Logic to diagnose faults in automotive braking systems while Banerjee et al. [135] profiles a system with an amalgam of Fuzzy Logic Data-Driven predictors and Model-Based statistical data.

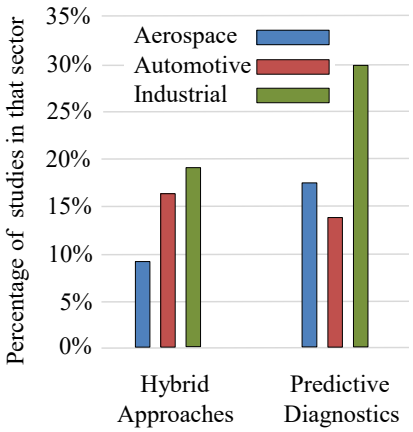

Figure 3: Hybrid and Predictive Approaches across all sectors.

Using multiple approaches in this way allows practitioners to apply the most appropriate technique to different aspects of an ICPS. Rizzoni et al [162] discuss how both model-based and neural network techniques facilitated the development of on-board diagnostics and fault monitoring to measure vehicle emissions in automobiles. They trace the motivation for continuously assessing emission compliance in each vehicle back to the California Air Resource Board (CARB) requirements that came into force in 1991. Each vehicle is required to monitor its own emissions to ensure compliance. That required neural network approaches to facilitate the tasks of data capture and sensor filtering followed by model invalidation to test compliance.

4.5. Predictive Diagnostic techniques

Predictive Diagnostics or Prognostics is the ability to detect the signs of an impending fault before a failure occurs and to estimate when it might happen [136]. Figure 3 suggests that the ability to predict ICPS faults in advance is of interest in all three domains. Predictive Diagnostics becomes feasible when it is possible to both capture and process large amounts of high-fidelity data about the operation of an ICPS and recognize the fault symptoms in-advance. Janasak and Beshears [163] state that one aim of European air carriers is that by 2050, all flights should arrive within one minute of their scheduled time. Current delays and disruptions can be up to fifteen minutes due to undiagnosed faults, an issue that better predictive capabilities might alleviate.

4.6. Overall trends in the data

In each sector, there is an emphasis on the development of smart sensors and the conditioning of the sensor data using a range of techniques such as Kalman Filters or Markov models. Coupled with that, the representation of ideal values or behavior was described using either models or dynamically using AI data-mining techniques. Once a definition of what is normal can be determined, deviations from expected values or behaviors can be detected. Artificial Neural Networks and Machine Learning were evidenced as alternatives to Model Invalidation in the Data-Driven AI category. However, the widespread use of hybrid techniques in different parts of the ICPS reflects the complexity of the systems being profiled: no single technique for fault recognition and analysis predominates or is sufficient for all needs. The predominance of Data-Driven techniques in aerospace is in contrast to the lack of evidence for the use of Knowledge-based approaches in that sector while Network Message Analysis was a technique profiled in 29% of the industrial studies that employed Knowledge-based approaches. Those contrasts are explored more deeply in Section 5 where we examine the most mature techniques in more detail.

Table 5: Adapting the NASA Technology Readiness Levels for Assessing Fault Diagnosis.

TRL	NASA categorization	Proposed Fault categorization
9	Actual system "flight proven" through successful mission operations.	Actual fault diagnostic system proven through successful identification and classification of real faults in a production environment.
8	Actual system completed and "flight qualified" through test and demonstration (ground or space).	Actual fault diagnostic system qualified through test and demonstration in a production environment.
7	System prototype demonstration in a space environment.	Functioning prototype demonstrated in a production environment.
6	System/subsystem model or prototype demonstration in a relevant environment (ground or space).	Functioning prototype demonstrated finding and/or diagnosing faults in a relevant environment beyond the laboratory.
5	Component and/or breadboard validation in a relevant environment.	Creation of a breadboard and/or software validation that can search for and/or identify faults in a relevant environment.
4	Component and/or breadboard validation in a laboratory environment.	Creation of a breadboard and/or software validation that can search for and/or identify faults in a laboratory environment.
3	Analytical and experimental critical function and/or characteristic proof-of-concept.	Proof-of-concept experiment with an appropriate simulation of the fault environment.
2	Technology concept and/or application formulated.	Concept and technology to perform detection and/or diagnosis proposed, including a mathematical formulation.
1	Basic principles observed and reported.	Basic fault detection or diagnosis principles observed and reported.

5. Investigating mature fault diagnostic techniques

RQ2 asked what levels of maturity the diagnostic techniques adopted in each sector have achieved. The TRL fault classifications we developed for our study are shown in Table 5 in parallel with the matching NASA descriptions.

Mankins [55] explains that each level in the TRL scale represents a different maturation of the technology or methodology. Heder [164] notes that the TRL has drawn criticism for its use outside of the environment it was originally designed for, explaining that in the European Union the approach has not always been tailored properly for specific disciplines. However in NASA the concept of "flight-readiness" was already deeply ingrained in their culture [165]. Adapting this concept to machinery to establish what stage of technological readiness it has reached was a natural step within their context. We considered this when designing our study, carefully crafting our adaptations of the individual level descriptions to ensure we stayed true to the intent of the TRL.

Assessing the maturity of a technological approach requires a careful evaluation of the context that it is being trialled or applied in. Our TRL categories are divided into four distinct groups. Studies classified as TRL 7 to 9 represent the most mature implementations. They provide a fascinating glimpse of techniques which are either close to or fully operational in live production environments.

Studies from TRL 5 and 6 provide evaluations from trials performed in highly-realistic environments beyond the laboratory. They often use case studies to illustrate how the diagnostics will work in particular situations. In contrast, studies at TRL 3 and 4 present functioning prototypes that are being evaluated in either a laboratory or simulated environment.

Levels 1 and 2 categorize fault identification and diagnosis techniques that are purely theoretical or are presented with a formal mathematical treatment. Papers at this level do not report concrete outcomes from case studies or field trials.

Figure 4 highlights the TRL maturity levels we observed across all our domains of interest. Amongst the survey papers are studies of fault identification and diagnostic techniques that have moved beyond the laboratory and are being applied

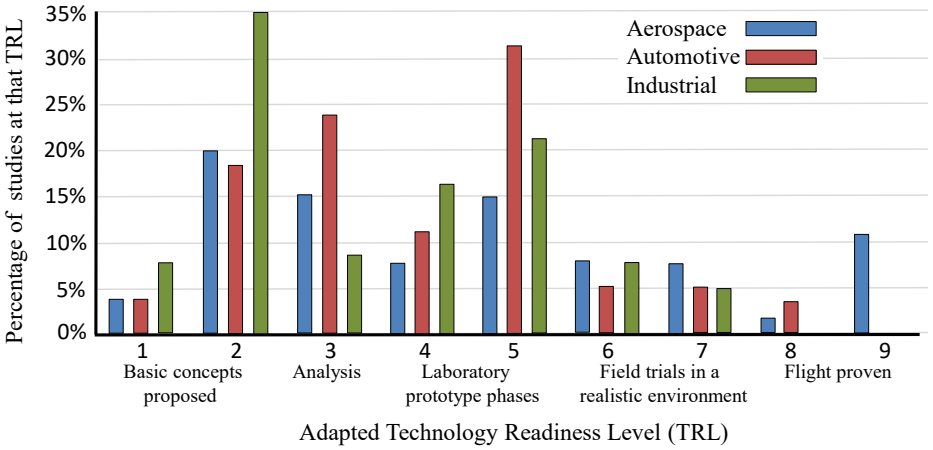

Figure 4: Technology Readiness Level by Sector

in real-world environments. In these papers, we should expect to see state-of-the-art exemplars that detail how ICPS respond to and recover from fault situations they encounter. The papers we classified at TRL 7 and above present evaluations of how well these techniques detect and analyse faults and why these approaches were adopted.

5.1. Studies from Aerospace and Avionics

Figure 5 highlights where the diagnostic research is focused in the aerospace sector. The research focus on flight control, high-dependability and predictive fault management aligns with the observations from the studies at the highest TRL discussed in this section.

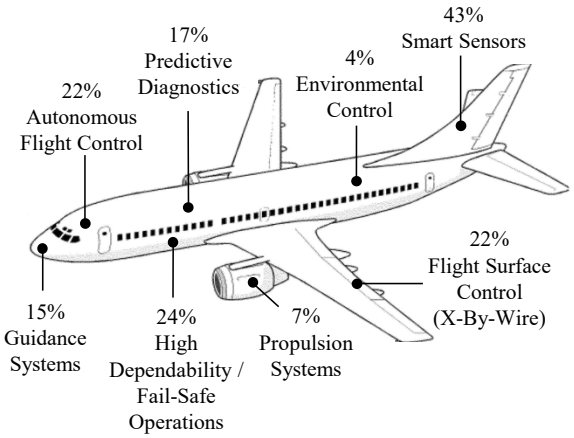

Figure 5: Where diagnostic research is focused in the aerospace sector.

Benowitz [117] profiles the Fault Protection Engine currently used by the Mars Curiosity rover. Since the rover is too far away to rely on external systems for assistance, the fault protection engine has to proactively manage faults within a large number of interrelated subsystems autonomously.

Earlier rover designs implemented discrete fault management within each subsystem. On Curiosity, the architecture implements *monitors*, code within each module whose responsibility is to recognise anomalous behavior. Each module has specific knowledge of the subsystem they are operating within that informs their judgements while filtering sensor readings. Monitors signal problems by raising an error flag. As well as detecting faults, they maintain a count of the occurrences that is later used by the fault protection engine to ascertain how persistent or serious the fault is.

Benowitz explains that error flags are latched but never cleared by the ICPS module-level monitors. This allows the fault protection engine to manage the overall health of the rover by polling in its own time, making decisions without being flooded by messages from subsystems. The fault engine maintains a model that contains a response that is appropriate to each situation the monitors are signalling. Curiosity has over 1,000 monitors operating at any one time. Since the rover may be performing any number of different tasks at any time, ranging from landing to exploration, fault management has to be contextual.

Curiosity's Model-Based approach is in contrast to the hybrid Model-Based and Data-Driven ap-

proach employed by Zolghadri et al. [68]. They profile the flight surface control systems they developed for the Airbus A380. Like Curiosity, their fault management is situation-aware. They note that fault signatures are often difficult to detect when an aircraft is parked or taxiing, or when the data rates from sensors are low. Their approach calculates *residuals*, the result obtained by comparing the current servo positions with the estimated position predicted by the model. They tune the sensitivity of Kalman filters to establish a trade-off between reliably detecting signals and robustness with respect to normal environmental variations. Azam et al. [133] take a similar approach using neural networks to dynamically model and monitor fifty flight parameters. They discuss the difficulty of using model-based approaches that cannot manage the complexity of accommodating all reasonable parameters in all flight modes. Their data-driven approach also provides estimates of fault severity.

Iverson et al. [132] and Schwabacher et al. [126, 136, 166] provide a highly detailed treatment of the hybrid fault monitoring system certified by NASA for International Space Station (ISS) operations and for Ares I-X launch pre-diagnostics. The Inductive Monitoring System (IMS) is a ground-based ICPS that processes telemetry from the ISS in near real-time. It relies on rule-based, Model-Based and Data Driven algorithms in three distinct subsystems of the IMS. They employ a clustering approach from a fixed number of training points, an approach that allows them to rapidly tailor IMS for new situations. Schwabacher et al. note that there is a need for mission-critical systems such as these to be flight-certified since ground controllers rely on them to make go/no go decisions about launches. They note that many Space Shuttle launches were delayed due to unreliable fault diagnoses. When launch faults can be evaluated more rapidly, redundant or hot-swappable modules can be deployed to reactivate launch sequences to meet critical time windows.

Studies such as these help to explain the proliferation of hybrid techniques encountered. In aerospace, 54% used Artificial Neural Networks and 38% employed Machine Learning, coupled with a range of Model Invalidation methods that were discussed in 43% of all aerospace studies.

5.2. Studies from the Automotive sector

The automotive ecosystem is built up of millions of discrete, complex and mostly unconnected ICPS. Each vehicle operates as a self-contained network of co-operating subsystems. Stout's Automotive Defect and Recall Report shows that in 2018, nearly eight million vehicles were recalled in the US to address software-based defects [167]. That total is higher than all the recalls for software issues in the previous five years. Figure 6 highlights where diagnostic research is focused in the automotive sector.

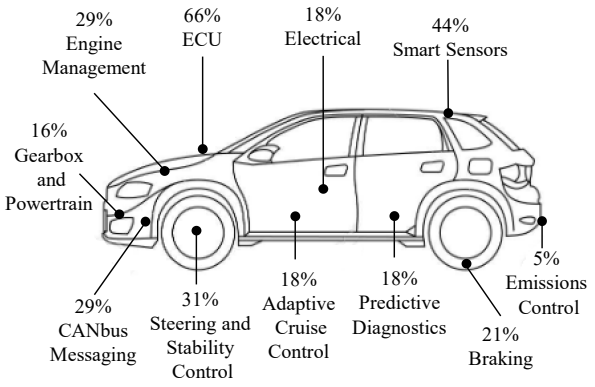

Figure 6: Where diagnostic research is focused in the automotive sector.

Modern vehicles feature up to 120 embedded ECUs, connected by five or more system buses [168, 169]. Sarecco highlights how large and complex the software is currently in vehicles, reporting that the 2017 Ford 150 pickup requires 150 million lines of code [170]. Charet [171] contrasts this with the F-35 Joint Strike Fighter that required only 5.7 million lines of code while the Boeing 787 Dreamliner uses only 6.5 million lines.

This complexity is reflected in the automotive survey papers at the highest TRL. Nasri et al [172] explain that the increasing sophistication of in-car electronics, including Adaptive Cruise Control, Lane Detection and Light Detection And Ranging (LIDAR) technologies, leads to more intricate fault scenarios. They detail the implementation of diagnostics that analyse messages flowing between subsystems on the vehicles Controller Area Network (CAN). Many of the current diagnostic tools rely on proprietary software from vendors that are not easy to integrate into system-wide diagnostic frameworks. They detail their implementation of a hierarchical chain of localised diagnosers that are monitored by a single global fault analyser. A Directed Graph approach is used to identify faults, capturing CAN messages via hardware-in-loop connections.

The scope of what is deemed a "safety-

critical" component in the automotive sector is also changing. In May 2018, back-up cameras became mandatory on US vehicles, transforming an optional luxury item into something that required much more rigorous quality control and deeper vehicle integration [167].

Over-the-Air (OTA) access to diagnostic data from automobiles is profiled as one route to addressing the difficulty of fault-finding in disconnected automotive ICPS. The global remote diagnostics market is forecasted to grow at 17% annually over the next five years, driven primarily by the potential operational cost savings to automakers [173]. Steinkamp et al. describe General Motors new OTA system which is capable of handling 4.5 TB of data per hour from vehicles [167].

However, Dragojevic et al. [174] identify remote access to diagnostic data from a vehicle as a significant technical challenge. Traditional automotive architectures featured highly-specialized ECUs that were optimized for minimal functionality to balance safety concerns. Full operating systems for vehicles emerged though middleware such as Adaptive AUTOSAR [175], leading to greater opportunities to aggregate diagnostic data that could be shared with remote fault analysis systems. Without functionality such as OTA, remote vehicle diagnostics cannot be performed in an IIoT ecosystem. Dragojevic et al. profile their work on an OTA bridge solution that connects with the on-board vehicle network. However, they note that Adaptive AUTOSAR needs to encompass safety aspects to certifiable levels before it can be widely deployed.

Kane, Fuhrman and Koopman detail the use of runtime monitor-based oracles that mine the data used by OTA systems for fault finding [114]. Runtime monitors analyze system traces to see if they conform to acceptable behavior patterns. They tune their oracles using large amounts of previously captured telemetry and describe methods used during live vehicle trials. Since monitors operate as hardware-in-loop devices and often interact with safety-critical components, they have to be designed as high-integrity devices. They address this by creating isolated monitors with well-defined interfaces.

5.3. Studies from Manufacturing and Control

Unlike the automotive and aerospace sector, most industrial systems are stationary in one location and are therefore easier to connect into factory-wide monitoring systems. Industrial production machinery therefore offers numerous opportunities to perform local or remote diagnostics. Ramos et al cite maintenance costs of up to 60% of the production costs as a key driver for factory diagnostics and prognostics [101]. Figure 7 highlights where diagnostic research is focused in the manufacturing and industrial control sector.

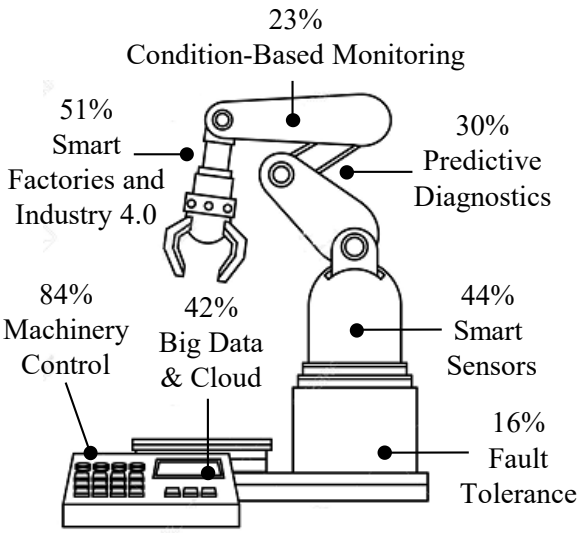

Figure 7: Where diagnostic research is focused in the industrial control sector.

International, industry-wide initiatives foster standardization across this sector. Chen et al. [91] discuss trials of sensors for gearboxes in the context of manufacturing initiatives such as the Machinery Information Management Open Systems Alliance (MIMOSA) [176]. Lee, Jin and Bagheri [2] discuss Industry 4.0 and Big Data as similar driver of standardization. Their approach demonstrates end-to-end factory machinery feeding sensor data into multiple analytical systems for near real-time fault identification and prediction. They employ deep-learning for Data-Driven prognostics.

Ramos et al. [101] also profile Service Orientated Architectures to expose fault-finding services at multiple factory levels. Their case study focuses on self-recovering machinery that is supported by the factory infrastructure using hardware-in-loop techniques. Manufacturing is typically managed by multi-layer IT infrastructures that connect higher-level Enterprise Resource Planning (ERP) through layers down to factory automation systems such a SCADA. Ramos et al. profile their eSonia system which manages assets on multiple levels. Many pro-

duction operations require assembly lines to be able to be re-configured dynamically to suit changes in demand. This requires a degree of self-awareness from plant equipment, which must be able to signal if it is available when changes are requested.

6. Conclusions, gaps and future work

This scoping study was written with a view to providing an overview of mature fault identification and diagnosis techniques for practitioners who are seeking to understand the state of the current practice and who are creating ICPS. The wide use of Model-Based (62%) and alternative Data-Driven AI (33%) techniques across the aerospace, automotive and industrial control domains reflects the complexity of the current ICPS application space.

As the number of interconnected ICPS increases along with the intricacy of the tasks they manage, the use of Model-Based approaches alone was often profiled as becoming intractable. Milis [33] discussed the difficulties of calibration to align with real systems. Scalability of models was also discussed in this context but only Yen et al [8] discussed partial models, a technique for segmenting models into sub-models. No studies profiled Digital Twins as a solution. Model-Based diagnosis remains a viable strategy, yet how we create complete-enough or partial models quickly and reliably remains a challenge. The AADL EV2 Error Annex has potential to be used beyond the early modeling stages however we found no evidence of its use in the field.

Model-Free AI approaches were evidenced as a viable way of addressing this challenge, demonstrating the increasing sophistication of current machine learning systems. However, there was no discussion of explainable AI, where the decisions made by algorithms could be justified.

The proliferation of hybrid fault systems that blend different aspects and techniques reached 19% in the industrial control sector, indicating the importance of further research into multiple-method solutions, where models are tuned by real-time data. Design-for-Certification was highlighted as a significant driver to ensure products could be deployed beyond the laboratory [174, 126, 151].

Predictive diagnostics is a promising area that was often discussed in-context with the ability to mine sensor data with enough granularity to allow faults to be predicted. Predictive techniques were prevalent in 30% of all industrial control studies, driven by the availability of large amounts of local data. Further research to develop remote connectivity in the aerospace and automotive sectors should lead to more powerful predictive capabilities. However, the potential volume of the data available from these ICPS also presents challenges of scale.

Statistical aspects of Knowledge-based diagnostic approaches were poorly represented across the aerospace sector. Most applications of the technique in the automotive and industrial control sectors discussed Bayesian approaches and various Petri net derivatives. This may be due to the increasing presence of hybrid approaches which employ Knowledge-Based methods in the midst of other techniques. There was little evidence of traditional Expert Systems.

Connectivity is a key characteristic of ICPS yet it has deeper implications in our sectors of interest. Table 6 illustrates how connectivity for facilitating diagnostics is made more challenging because of the different environments ICPS operate in. Brief discussions in the papers of emerging cloud technologies pointed towards ways of establishing connectivity in more achievable ways.

While the TRL analysis provided a way of identifying and profiling the most mature approaches, those results cannot always be extrapolated across all three sectors. Almost all the avionic and aerospace studies profiled originated from organizations who were partnering with agencies such as NASA and ESA. These do not face the same intellectual property restrictions that restrict what we might expect to find published in the automotive and industrial control sectors.

During our paper selection, promising papers from the medical device ICPS sector gave a tantalizing glimpse of the differences and challenges that sector presents. We look forward to exploring that domain in a later study, where complex, safety-critical devices and regulatory certification are the norm rather than the exception.

References

[1] R. Alur, Principles of cyber-physical systems, MIT Press, 2015.

[2] J. Lee, B. Bagheri, H.-A. Kao, A cyber-physical systems architecture for industry 4.0-based manufacturing systems, Manufacturing Letters 3 (2015) 18–23. doi:10.1016/j.mfglet.2014.12.001.

[3] M. A. Laughton, M. G. Say, Electrical engineer's reference book, Elsevier, 2013.

Table 6: Recurring themes across sectors.

Theme	Aerospace	Automotive	Industrial Control
Existing degree of connectivity?	Low.	Becoming more connected.	Already highly connected.
Difficulty of becoming more connected?	Hard due to distance and low bandwidth.	Hard due to large number of discrete vehicle instances.	Already highly-connected due to high degree of localization.
Amount of diagnostic data available?	High.	Becoming very high.	Already very high.
Need for autonomous operation?	Very high in remote planetary rovers and drones	Very high due to cost of local fault repair.	Already established via predictive diagnostics and self-management.

[4] E. A. Parr, Industrial control handbook, Industrial Press Inc., 1998.

[5] M. Bajracharya, M. W. Maimone, D. Helmick, Autonomy for Mars Rovers: Past, Present, and Future, Computer 41 (12) (2008). doi:10.1109/MC.2008.479.

[6] G. Jacoby, et al., Testing adaptive probabilistic software components in cyber systems, in: Monterey Workshop, Springer, 2010, pp. 228–238. doi:10.1007/978-3-642-21292-5_13.

[7] P. Leitão, S. Karnouskos, L. Ribeiro, P. Moutis, J. Barbosa, T. I. Strasser, Common practices for integrating industrial agents and low level automation functions, in: IECON 2017-43rd Annual Conference of the IEEE Industrial Electronics Society, IEEE, 2017, pp. 6665–6670. doi:10.1109/IECON.2017.8217164.

[8] I.-L. Yen, S. Zhang, F. Bastani, Y. Zhang, A framework for IoT-based monitoring and diagnosis of manufacturing systems, in: 2017 IEEE Symposium on Service-Oriented System Engineering (SOSE), IEEE, 2017, pp. 1–8. doi:10.1109/SOSE.2017.26.

[9] Wu, Dazhong and Liu, Shaopeng and Zhang, Li and Terpenny, Janis and Gao, Robert X and Kurfess, Thomas and Guzzo, Judith A, A fog computing-based framework for process monitoring and prognosis in cyber-manufacturing, Journal of Manufacturing Systems 43 (2017) 25–34. doi:10.1016/j.jmsy.2017.02.011.

[10] A. Tanveer, R. Sinha, S. MacDonell, On Design-time Security in IEC 61499 Systems: Conceptualisation, Implementation, and Feasibility (2018). doi:10.1109/INDIN.2018.8472093.

[11] A. Sargolzaei, C. D. Crane, A. Abbaspour, S. Noei, A machine learning approach for fault detection in vehicular cyber-physical systems, in: Machine Learning and Applications (ICMLA), 2016 15th IEEE International Conference on, IEEE, 2016, pp. 636–640.

[12] F. Mohrle, M. Zeller, K. Hofig, M. Rothfelder, P. Liggesmeyer, Automated compositional safety analysis using component fault trees, in: Software Reliability Engineering Workshops (ISSREW), 2015 IEEE International Symposium on, IEEE, 2015, pp. 152–159. doi:10.1109/ISSREW.2015.7392061.

[13] M. H. Kim, S. Lee, K. C. Lee, A fuzzy predictive redundancy system for fault-tolerance of x-by-wire systems, Microprocessors and Microsystems 35 (5) (2011) 453–461. doi:10.1016/j.micpro.2011.04.003.

[14] G. J. Holzmann, Mars Code, Commun. ACM 57 (2) (2014) 64–73. doi:10.1145/2560217.2560218.

[15] J. A. Starek, B. Açıkmeşe, I. A. Nesnas, M. Pavone, Spacecraft autonomy challenges for next-generation space missions, in: Advances in Control System Technology for Aerospace Applications, Springer, 2016, pp. 1–48. doi:10.1007/978-3-662-47694-9_1.

[16] J. D. McGregor, D. P. Gluch, P. H. Feiler, Analysis and Design of Safety-critical, Cyber-Physical Systems, ACM SIGAda Ada Letters 36 (2) (2017) 31–38.

[17] P. Feiler, D. Gluch, J. Mcgregor, An architecture-led safety analysis method, in: 8th European Congress on Embedded Real Time Software and Systems (ERTS 2016), Toulouse, 2016.

[18] A. Chamseddine, M. H. Amoozgar, Y. M. Zhang, Experimental validation of fault detection and diagnosis for unmanned aerial vehicles, in: Handbook of Unmanned Aerial Vehicles, Springer, 2015, pp. 1123–1155. doi:10.1007/978-90-481-9707-1_41.

[19] N. Kunst, J. Judkins, C. Lynn, D. Goodman, Damage propagation analysis methodology for electromechanical actuator prognostics, in: 2009 IEEE Aerospace conference, IEEE, 2009, pp. 1–7. doi:10.1109/DSN.2014.28.

[20] A. Kodali, Y. Zhang, C. Sankavaram, K. Pattipati, M. Salman, Fault diagnosis in the automotive electric power generation and storage system (EPGS), IEEE/ASME Transactions On Mechatronics 18 (6) (2013) 1809–1818. doi:10.1109/TMECH.2012.2214397.

[21] C. Sankavaram, A. Kodali, K. Pattipati, An integrated health management process for automotive cyber-physical systems, in: 2013 International Conference on Computing, Networking and Communications (ICNC), IEEE, 2013, pp. 82–86. doi:10.1109/ICCNC.2013.6504058.

[22] P. Z. Schulte, A State Machine Architecture for Aerospace Vehicle Fault Protection, Ph.D. thesis, Georgia Institute of Technology (2018).

[23] H. Shraim, A. Awada, R. Youness, A survey on quadrotors: Configurations, modeling and identifica-

tion, control, collision avoidance, fault diagnosis and tolerant control, IEEE Aerospace and Electronic Systems Magazine 33 (7) (2018) 14–33. doi:10.1109/MAES.2018.160246.

[24] C. Sankavaram, A. Kodali, K. Pattipati, S. Singh, Y. Zhang, M. Salman, An Inference-based Prognostic Framework for Health Management of Automotive Systems, International Journal of Prognostics and Health Management 7 (2016) 1–16.

[25] V. Bolbot, G. Theotokatos, M. L. Bujorianu, E. Boulougouris, D. Vassalos, Vulnerabilities and safety assurance methods in Cyber-Physical Systems: A comprehensive review, Reliability Engineering & System Safety (2018).

[26] P. Z. Cacchione, The evolving methodology of scoping reviews (2016). doi:10.1177/1054773816637493.

[27] H. Arksey, L. O'Malley, Scoping studies: towards a methodological framework, International journal of social research methodology 8 (1) (2005) 19–32. doi:10.1080/1364557032000119616.

[28] J. C. Mankins, Technology readiness assessments: A retrospective, Acta Astronautica 65 (9-10) (2009) 1216–1223. doi:10.1016/j.actaastro.2009.03.058.

[29] N. Jazdi, Cyber physical systems in the context of Industry 4.0, in: Automation, Quality and Testing, Robotics, 2014 IEEE International Conference on, IEEE, 2014, pp. 1–4. doi:10.1109/AQTR.2014.6857843.

[30] E. A. Lee, S. A. Seshia, Introduction to embedded systems: A cyber-physical systems approach, MIT Press, 2016.

[31] T. R. Thombare, L. Dole, Review on fault diagnosis model in automobile, in: Computational Intelligence and Computing Research (ICCIC), 2014 IEEE International Conference on, IEEE, 2014, pp. 1–4. doi:10.1109/ICCIC.2014.7238546.

[32] O. I. de Normalisation, Systems and Software Engineering: Systems and Software Quality Requirements and Evaluation (SQuaRE): System and Software Quality Models, ISO/IEC, 2011.

[33] G. M. Milis, D. G. Eliades, C. G. Panayiotou, M. M. Polycarpou, A cognitive fault-detection design architecture, in: Neural Networks (IJCNN), 2016 International Joint Conference on, IEEE, 2016, pp. 2819–2826. doi:10.1109/IJCNN.2016.7727555.

[34] F. Harirchi, N. Ozay, Guaranteed model-based fault detection in cyber-physical systems: A model invalidation approach, arXiv (2016). doi:arXiv:1609.05921.

[35] D. M. Johnson, A review of fault management techniques used in safety-critical avionic systems, Progress in Aerospace Sciences 32 (5) (1996) 415–431. doi:10.1016/0376-0421(96)82785-0.

[36] C. Bradatsch, T. Ungerer, R. Zalman, A. Lajtkep, Towards runtime testing in automotive embedded systems, in: Industrial Embedded Systems (SIES), 2011 6th IEEE International Symposium on, IEEE, 2011, pp. 55–58. doi:10.1109/SIES.2011.5953679.

[37] J. De Kleer, B. C. Williams, Diagnosing multiple faults, Artificial intelligence 32 (1) (1987) 97–130. doi:10.1016/0004-3702(87)90063-4.

[38] Y.-g. Chen, Applications of Bayesian network in fault diagnosis of braking system, in: Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2011 International Conference on, Vol. 1, IEEE, 2011, pp. 234–237. doi:10.1109/IHMSC.2011.63.

[39] M. Ghadhab, M. Kuntz, D. Kuvaiskii, C. Fetzer, A controller safety concept based on software-implemented fault tolerance for fail-operational automotive applications, in: International Workshop on Formal Techniques for Safety-Critical Systems, Springer, 2015, pp. 189–205. doi:10.1007/978-3-319-29510-7_11.

[40] A. Le Mortellec, J. Clarhaut, Y. Sallez, T. Berger, D. Trentesaux, Embedded holonic fault diagnosis of complex transportation systems, Engineering Applications of Artificial Intelligence 26 (1) (2013) 227–240. doi:10.1016/j.engappai.2012.09.008.

[41] D. Levac, H. Colquhoun, K. K. O'Brien, Scoping studies: advancing the methodology, Implementation science 5 (1) (2010) 69. doi:10.1186/1748-5908-5-69.

[42] C. Wohlin, Writing for synthesis of evidence in empirical software engineering, in: Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, ACM, 2014, p. 46. doi:10.1145/2652524.2652559.

[43] N. Mays, E. Roberts, J. Popay, et al., Synthesising research evidence, Studying the organisation and delivery of health services: Research methods 220 (2001).

[44] E. Antman, J. Lau, B. Kupelnick, et al., A comparison of results of meta-analyses of RCTs and recommendations of clinical experts. Treatments for myocardial infarction, Journal of the American Medical Association 268 (1992) 240–248.

[45] Z. Munn, M. D. Peters, C. Stern, C. Tufanaru, A. McArthur, E. Aromataris, Systematic review or scoping review? guidance for authors when choosing between a systematic or scoping review approach, BMC medical research methodology 18 (1) (2018) 143. doi:10.1186/s12874-018-0611-x.

[46] N. R. Council, Embedded, Everywhere: A Research Agenda for Networked Systems of Embedded Computers., The National Academies Press, 2001. doi:10.17226/10193.

[47] H. Gill, From vision to reality: cyber-physical systems, in: HCSS national workshop on new research directions for high confidence transportation CPS: automotive, aviation, and rail, 2008.

[48] E. A. Lee, Cyber-physical systems-are computing foundations adequate, in: Position paper for NSF workshop on cyber-physical systems: research motivation, techniques and roadmap, Vol. 2, Citeseer, 2006, pp. 1–9.

[49] N. Wiener, Cybernetics or Control and Communication in the Animal and the Machine, Vol. 25, MIT press, 1948.

[50] B. Dowdeswell, R. Sinha, S. MacDonell, Data set Finding Faults: a scoping study of Fault Diagnostics for Industrial Cyber-Physical Systems, Mendeley Data, v1 (2019). doi:10.17632/wg9shy9rsm.1.

[51] D. S. Cruzes, T. Dyba, Recommended steps for thematic synthesis in software engineering, in: 2011 International Symposium on Empirical Software Engineering and Measurement, IEEE, 2011, pp. 275–284. doi:10.1109/ESEM.2011.36.

[52] D. S. Cruzes, T. Dybå, Research synthesis in software engineering: A tertiary study, Information and Software Technology 53 (5) (2011) 440–455. doi:10.1016/j.infsof.2011.01.004.

[53] A. Castleberry, A. Nolen, Thematic analysis of qualitative research data: Is it as easy as it sounds?,

Currents in Pharmacy Teaching and Learning 10 (6) (2018) 807–815. doi:10.1016/j.cptl.2018.03.019.

[54] J. Straub, In search of technology readiness level (TRL) 10, Aerospace Science and Technology 46 (2015) 312–320.

[55] J. C. Mankins, Technology readiness levels, White Paper, April 6 (1995).

[56] ESA, ESA Software Engineering and Standardization. (2009) [cited 11 July 2017].
URL http://http://www.esa.int/TEC/Software_engineering_and_standardisation/TECP5EUXBQE_0.html

[57] ISO/BSS, Space systems. Definition of the Technology Readiness Levels (TRLs) and their criteria of assessment, ISO/BSS, 2019.

[58] EARTO, The trl scale as a research and innovation policy tool, earto recommendations. (2014) [cited 27 July 2019].
URL https://www.earto.eu/wp-content/uploads/The_TRL_Scale_as_a_R_I_Policy_Tool_-_EARTO_Recommendations_-_Final.pdf

[59] R. J. Terrile, F. G. Doumani, G. Y. Ho, B. L. Jackson, Calibrating the technology readiness level (TRL) scale using NASA mission data, in: 2015 IEEE Aerospace Conference, IEEE, 2015, pp. 1–9. doi:10.1109/AERO.2015.7119313.

[60] C. Wohlin, P. Runeson, P. A. d. M. S. Neto, E. Engström, I. do Carmo Machado, E. S. De Almeida, On the reliability of mapping studies in software engineering, Journal of Systems and Software 86 (10) (2013) 2594–2610. doi:10.1016/j.jss.2013.04.076.

[61] B. Kitchenham, R. Pretorius, D. Budgen, O. P. Brereton, M. Turner, M. Niazi, S. Linkman, Systematic literature reviews in software engineering–a tertiary study, Information and Software Technology 52 (8) (2010) 792–805. doi:10.1016/j.infsof.2010.03.006.

[62] A. Haghighatkhah, A. Banijamali, O.-P. Pakanen, M. Oivo, P. Kuvaja, Automotive software engineering: A systematic mapping study, Journal of Systems and Software 128 (2017) 25–55. doi:10.1016/j.jss.2017.03.005.

[63] D. L. Nuñez, M. Borsato, An ontology-based model for prognostics and health management of machines, Journal of Industrial Information Integration 6 (2017) 33–46. doi:10.1016/j.jii.2017.02.006.

[64] P. Goupil, J. Boada-Bauxell, A. Marcos, E. Cortet, M. Kerr, H. Costa, AIRBUS efforts towards advanced real-time fault diagnosis and fault tolerant control, IFAC Proceedings Volumes 47 (3) (2014) 3471–3476. doi:10.3182/20140824-6-ZA-1003.01945.

[65] S. Windmann, O. Niggemann, Efficient fault detection for industrial automation processes with observable process variables, in: 2015 IEEE 13th International Conference on Industrial Informatics (INDIN), IEEE, 2015, pp. 121–126. doi:10.1109/INDIN.2015.7281721.

[66] Y. Jiang, K. Li, S. Yin, Cyber-physical system based factory monitoring and fault diagnosis framework with plant-wide performance optimization, in: 2018 IEEE Industrial Cyber-Physical Systems (ICPS), IEEE, 2018, pp. 240–245. doi:10.1109/ICPHYS.2018.8387666.

[67] X. Chang, J. Huang, F. Lu, Robust in-flight sensor fault diagnostics for aircraft engine based on sliding mode observers, Sensors 17 (4) (2017) 835. doi:10.3390/s17040835.

[68] A. Zolghadri, J. Cieslak, D. Efimov, D. Henry, P. Goupil, R. Dayre, A. Gheorghe, H. Leberre, Signal and model-based fault detection for aircraft systems, IFAC-PapersOnLine 48 (21) (2015) 1096–1101. doi:10.1016/j.ifacol.2015.09.673.

[69] T. Ruppert, J. Abonyi, Software Sensor for Activity-Time Monitoring and Fault Detection in Production Lines, Sensors 18 (7) (2018) 2346. doi:10.3390/s18072346.

[70] G. Ducard, The SMAC fault detection and isolation scheme: Discussions, improvements, and application to a UAV, in: 2013 Conference on Control and Fault-Tolerant Systems (SysTol), IEEE, 2013, pp. 480–485. doi:10.1109/SysTol.2013.6693949.

[71] A. L. White, Designing fault-injection experiments for the reliability of embedded systems, in: Digital Avionics Systems Conference (DASC), 2012 IEEE/AIAA 31st, IEEE, 2012, pp. 9D5–1. doi:10.1109/DASC.2012.6382447.

[72] L. Ribeiro, J. Barata, Re-thinking diagnosis for future automation systems: An analysis of current diagnostic practices and their applicability in emerging IT based production paradigms, Computers in Industry 62 (7) (2011) 639–659. doi:10.1016/j.compind.2011.03.001.

[73] A. Kodali, Y. Zhang, C. Sankavaram, K. Pattipati, M. Salman, Fault diagnosis in the automotive electric power generation and storage system (EPGS), IEEE/ASME Transactions on Mechatronics 18 (6) (2012) 1809–1818. doi:10.1109/TMECH.2012.2214397.

[74] W. A. Syed, S. Khan, P. Phillips, S. Perinpanayagam, Intermittent fault finding strategies, Procedia CIRP 11 (2013) 74–79. doi:10.1016/j.procir.2013.07.062.

[75] R. Dearden, T. Willeke, R. Simmons, V. Verma, F. Hutter, S. Thrun, Real-time fault detection and situational awareness for rovers: Report on the mars technology program task, in: 2004 IEEE Aerospace Conference Proceedings (IEEE Cat. No. 04TH8720), Vol. 2, IEEE, 2004, pp. 826–840. doi:10.1109/AERO.2004.1367683.

[76] G. Zhou, W. Feng, Q. Zhao, H. Zhao, State tracking and fault diagnosis for dynamic systems using labeled uncertainty graph, Sensors 15 (11) (2015) 28031–28051. doi:10.3390/s151128031.

[77] D. Ko, T. Kim, J. Park, S. Kang, I. Chun, An approach to applying goal model and fault tree for autonomic control, Contemporary Engineering Sciences 9 (2016) 843–851. doi:10.12988/ces.2016.6697.

[78] Z. Chen, X. Liu, R. Zhang, H. Liu, An Automotive Electronic Throttle Testing Equipment Based on STM32, in: Computer, Consumer and Control (IS3C), 2014 International Symposium on, IEEE, 2014, pp. 478–481. doi:10.1109/IS3C.2014.131.

[79] J. Hieb, J. Graham, J. Guan, An ontology for identifying cyber intrusion induced faults in process control systems, Critical Infrastructure Protection III (2009) 125–138doi:10.1007/978-3-642-04798-5_9.

[80] D. Trawczynski, J. Sosnowski, P. Gawkowski, Analyzing fault susceptibility of ABS microcontroller, Computer Safety, Reliability, and Security (2008) 360–372doi:10.1007/978-3-540-87698-4_30.

[81] F. Afonso, C. Silva, A. Tavares, S. Montenegro, Application-level fault tolerance in real-time embedded systems, in: 2008 International Symposium on Industrial Embedded Systems, IEEE, 2008, pp. 126–133.

doi:10.1109/SIES.2008.4577690.

[82] K. Höfig, M. Zeller, K. Schorp, Automated failure propagation using inner port dependency traces, in: Proceedings of the 11th International ACM SIGSOFT Conference on Quality of Software Architectures, ACM, 2015, pp. 123–128. doi:10.1145/2737182.2737191.

[83] W. Ahmad, O. Hasan, Formalization of fault trees in higher-order logic: a deep embedding approach, in: International Symposium on Dependable Software Engineering: Theories, Tools, and Applications, Springer, 2016, pp. 264–279. doi:10.1007/2F978-3-319-47677-3_21.

[84] K. Swearingen, K. Keller, Health ready systems, in: Autotestcon, 2007 IEEE, IEEE, 2007, pp. 625–631. doi:10.1109/AUTEST.2007.4374277.

[85] D. Klar, M. Huhn, Interfaces and models for the diagnosis of cyber-physical ecosystems, in: Digital Ecosystems Technologies (DEST), 2012 6th IEEE International Conference on, IEEE, 2012, pp. 1–6. doi:10.1109/DEST.2012.6227948.

[86] M. Käßmeyer, R. Berndt, P. Bazan, R. German, Product Line Fault Tree Analysis by Means of Multi-valued Decision Diagrams, in: International GI/ITG Conference on Measurement, Modelling, and Evaluation of Computing Systems and Dependability and Fault Tolerance, Springer, 2016, pp. 122–136. doi:10.1007/978-3-319-31559-1_11.

[87] C. Efkemann, T. Hartmann, Specification of conditions for error diagnostics, Electronic Notes in Theoretical Computer Science 217 (2008) 97–112. doi:10.1016/j.entcs.2008.06.044.

[88] G. Provan, A Contracts-Based Framework for Systems Modeling and Embedded Diagnostics, in: International Conference on Software Engineering and Formal Methods, Springer, 2014, pp. 131–143. doi:10.1007/978-3-319-15201-1_9.

[89] J. Wang, L. Zhang, L. Duan, R. X. Gao, A new paradigm of cloud-based predictive maintenance for intelligent manufacturing, Journal of Intelligent Manufacturing 28 (5) (2017) 1125–1137. doi:10.1007/s10845-015-1066-0.

[90] A. Palladino, G. Fiengo, D. Lanzo, A portable hardware-in-the-loop (hil) device for automotive diagnostic control systems, ISA transactions 51 (1) (2012) 229–236. doi:10.1016/j.isatra.2011.10.009.

[91] Z. Chen, Y. Yang, Z. Hu, A technical framework and roadmap of embedded diagnostics and prognostics for complex mechanical systems in prognostics and health management systems, IEEE Transactions on Reliability 61 (2) (2012) 314–322. doi:10.1109/TR.2012.2196171.

[92] P. Folkesson, F. Ayatolahi, B. Sangchoolie, J. Vinter, M. Islam, J. Karlsson, Back-to-Back Fault Injection Testing in Model-Based Development, in: International Conference on Computer Safety, Reliability, and Security, Springer, 2014, pp. 135–148. doi:10.1007/978-3-319-24255-2_11.

[93] M. Khlif, O. Tahan, M. Shawky, Co-simulation trace analysis (COSITA) tool for vehicle electronic architecture diagnosability analysis, in: Intelligent Vehicles Symposium (IV), 2010 IEEE, IEEE, 2010, pp. 572–578. doi:10.1109/IVS.2010.5548045.

[94] D. N. Lira, M. Borsato, Dependability Modeling for the Failure Prognostics in Smart Manufacturing., in: ISPE TE, 2016, pp. 885–894. doi:10.3233/978-1-61499-703-0-885.

[95] W. T. Hale, G. M. Bollas, Design of Built-In Tests for Active Fault Detection and Isolation of Discrete Faults, IEEE Access 6 (2018) 50959–50973. doi:10.1109/ACCESS.2018.2869269.

[96] J. Barata, L. Ribeiro, M. Onori, Diagnosis on evolvable production systems, in: 2007 IEEE International Symposium on Industrial Electronics, IEEE, 2007, pp. 3221–3226. doi:10.1109/ISIE.2007.4375131.

[97] J. Kurien, X. Koutsoukos, F. Zhao, Distributed diagnosis of networked, embedded systems, Tech. rep., XEROX Palo Alto Research Center CA (2002).

[98] J. Luo, K. Choi, K. R. Pattipati, L. Qiao, S. Chigusa, Distributed fault diagnosis for networked, embedded automotive systems, in: 2006 IEEE International Conference on Systems, Man and Cybernetics, Vol. 2, IEEE, 2006, pp. 1226–1232. doi:10.1109/ICSMC.2006.384882.

[99] S. Grimm, M. Watzke, T. Hubauer, F. Cescolini, Embedded Reasoning on Programmable Logic Controllers, in: International Semantic Web Conference, Springer, 2012, pp. 66–81. doi:10.1007/978-3-642-35173-0_5.

[100] M. Schoeller, M. Roemer, M. Leonard, M. Derriso, Embedded reasoning supporting aerospace IVHM, in: AIAA Infotech@ Aerospace 2007 Conference and Exhibit, 2007, p. 2820. doi:10.2514/6.2007-2820.

[101] A. V. Ramos, I. M. Delamer, J. L. Lastra, Embedded service oriented monitoring, diagnostics and control: Towards the asset-aware and self-recovery factory, in: Industrial Informatics (INDIN), 2011 9th IEEE International Conference on, IEEE, 2011, pp. 497–502. doi:10.1109/INDIN.2011.6034930.

[102] M. Khlif, M. Shawky, Enhancing diagnosis ability for embedded electronic systems using co-modeling, in: Novel Algorithms and Techniques In Telecommunications, Automation and Industrial Electronics, Springer, 2008, pp. 143–149.

[103] P. Castaldi, N. Mimmo, S. Simani, Fault diagnosis and fault tolerant control strategies for aerospace systems, in: 2016 3rd Conference on Control and Fault-Tolerant Systems (SysTol), IEEE, 2016, pp. 684–689. doi:10.1109/SYSTOL.2016.7739828.

[104] F. Harirchi, N. Ozay, Guaranteed model-based fault detection in cyber–physical systems: A model invalidation approach, Automatica 93 (2018) 476–488. doi:10.1016/j.automatica.2018.03.040.

[105] E. Bartocci, T. Ferrère, N. Manjunath, D. Ničković, Localizing faults in simulink/stateflow models with stl, in: Proceedings of the 21st International Conference on Hybrid Systems: Computation and Control (part of CPS Week), ACM, 2018, pp. 197–206. doi:10.1145/3178126.3178131.

[106] R. Cossé, D. Berdjag, S. Piechowiak, D. Duvivier, C. Gaurel, Meta-Diagnosis for a Special Class of Cyber-Physical Systems: The Avionics Test Benches, in: International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, Springer, 2015, pp. 635–644. doi:10.1007/978-3-319-19066-2_61.

[107] P. Struss, Model-based Analysis of Embedded Systems: Placing It upon Its Feet Instead of on Its Head-An Outsider's View., in: ICSOFT, 2013, pp. 284–291. doi:10.5220/0004596102840291.

[108] C. Modest, F. Thielecke, SPYDER: a software package for system diagnosis engineering, CEAS Aeronautical Journal 7 (2) (2016) 315–331. doi:doi.org/10.1007/s13272-016-0189-0.

[109] H. Khorasgani, G. Biswas, D. Jung, Structural methodologies for distributed fault detection and isolation, Applied Sciences 9 (7) (2019) 1286. doi:10.3390/app9071286.

[110] J. Chu, L. Zhang, P. Cui, Study on Integration Diagnosis System for Automobile Faults and Its Key Technologies, in: 2008 IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application, Vol. 1, IEEE, 2008, pp. 159–162. doi:10.1109/PACIIA.2008.207.

[111] H. G. M. Pakala, K. Raju, I. Khan, Integration testing of multiple embedded processing components, in: International Conference on Computer Science and Information Technology, Springer, 2011, pp. 200–209. doi:10.1007/978-3-642-17881-8_20.

[112] T. Sanislav, G. Mois, L. Miclea, A new approach towards increasing cyber-physical systems dependability, in: Proceedings of the 2015 16th International Carpathian Control Conference (ICCC), IEEE, 2015, pp. 443–447. doi:10.1109/CarpathianCC.2015.7145120.

[113] A. Abel, A. Adir, T. Blochwitz, L. Greenberg, T. Salman, Development and verification of complex hybrid systems using synthesizable monitors, in: Haifa Verification Conference, Springer, 2013, pp. 182–198. doi:10.1007/978-3-319-03077-7_13.

[114] A. Kane, T. Fuhrman, P. Koopman, Monitor based oracles for cyber-physical system testing: Practical experience report, in: 2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, IEEE, 2014, pp. 148–155. doi:10.1109/DSN.2014.28.

[115] M. Golagha, A. Pretschner, D. Fisch, R. Nagy, Reducing failure analysis time: An industrial evaluation, in: 2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP), IEEE, 2017, pp. 293–302. doi:10.1109/ICSE-SEIP.2017.15.

[116] M. B. Dwyer, R. Purandare, S. Person, Runtime verification in context: Can optimizing error detection improve fault diagnosis?, in: International Conference on Runtime Verification, Springer, 2010, pp. 36–50. doi:10.1007/978-3-642-16612-9_4.

[117] E. Benowitz, The Curiosity Mars Rover's Fault Protection Engine, in: 2014 IEEE International Conference on Space Mission Challenges for Information Technology, IEEE, 2014, pp. 62–66. doi:10.1109/SMC-IT.2014.16.

[118] R. Koitz, J. Lüftenegger, F. Wotawa, Model-based diagnosis in practice: interaction design of an integrated diagnosis application for industrial wind turbines, in: International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, Springer, 2017, pp. 440–445. doi:10.1007/978-3-319-60042-0_48.

[119] P. H. Feiler, B. A. Lewis, S. Vestal, The SAE Architecture Analysis & Design Language (AADL) a standard for engineering performance critical systems, in: 2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, IEEE, 2006, pp. 1206–1211. doi:10.1109/CACSD-CCA-ISIC.2006.4776814.

[120] OMG, What's Driving the connected car? (2020) [cited 31 March 2020].
URL https://www.omg.org/omgmarte/

[121] X.-F. Huang, C.-J. Zhou, S. Huang, K.-X. Huang, X. Li, Transient fault detection in networked control systems, International Journal of Distributed Sensor Networks 10 (11) (2014) 346269. doi:10.1155/2014/346269.

[122] S. Procter, P. Feiler, The AADL Error Library: An Operationalized Taxonomy of System Errors, ACM SIGAda Ada Letters 39 (1) (2020) 63–70. doi:10.1145/3379106.3379113.

[123] Y. Lu, Y. Dong, X. Wei, M. Xiao, A hybrid method of redundancy system reliability analysis based on aadl models, in: 2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C), IEEE, 2018, pp. 294–300. doi:10.1109/QRS-C.2018.00060.

[124] W. Zhang, G. Shen, Z. Huang, Z. Yang, L. Xue, An Analysis Tool Towards Fault Tolerance Systems based on AADL Error Model, International Journal of Performability Engineering 13 (6) (2017) 844–853. doi:10.23940/ijpe.17.06.p6.844853.

[125] J. Marzat, H. Piet-Lahanier, F. Damongeot, E. Walter, A new model-free method performing closed-loop fault diagnosis for an aeronautical system, in: 7th Workshop on Advanced Control and Diagnosis, ACD'2009, 2009, p. 6. doi:10.3182/20090706-3-FR-2004.00032.

[126] M. Schwabacher, K. Goebel, A survey of artificial intelligence for prognostics, in: Aaai fall symposium, 2007, pp. 107–114.

[127] E. R. Lapira, B. Bagheri, W. Zhao, J. Lee, R. V. Henriques, C. E. Pereira, L. Piccoli, C. Guimarães, A systematic approach to intelligent maintenance of production systems with a framework for embedded implementation, IFAC Proceedings Volumes 46 (7) (2013) 23–28. doi:10.3182/20130522-3-BR-4036.00092.

[128] J. Lee, C. Jin, B. Bagheri, Cyber physical systems for predictive production systems, Production Engineering 11 (2) (2017) 155–165.

[129] A. Guo, D. Yu, H. Du, Y. Hu, Z. Yin, H. Li, Cyber-physical failure detection system: Survey and implementation, in: Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), 2016 IEEE International Conference on, IEEE, 2016, pp. 428–432. doi:10.1109/CYBER.2016.7574863.

[130] W. Yan, J.-H. Zhou, Early Fault Detection of Aircraft Components Using Flight Sensor Data, in: 2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA), Vol. 1, IEEE, 2018, pp. 1337–1342.

[131] F. Langer, D. Eilers, R. Knorr, Fault detection in discrete event based distributed systems by forecasting message sequences with neural networks, in: Annual Conference on Artificial Intelligence, Springer, 2009, pp. 411–418. doi:10.1007/978-3-642-04617-9_52.

[132] D. L. Iverson, R. Martin, M. Schwabacher, L. Spirkovska, W. Taylor, R. Mackey, J. P. Castle, V. Baskaran, General purpose data-driven monitoring for space operations, Journal of Aerospace Computing, Information, and Communication 9 (2) (2012) 26–44. doi:10.2514/1.54964.

[133] M. Azam, K. Pattipati, J. Allanach, S. Poll, A. Patterson-Hine, In-flight fault detection and isolation in aircraft flight control systems, in: 2005 IEEE Aerospace Conference, IEEE, 2005, pp. 3555–3565. doi:10.1109/AERO.2005.1559659.

[134] X. Xu, T. Chen, M. Minami, Intelligent fault prediction system based on internet of things, Computers & Mathematics with Applications 64 (5) (2012) 833–839. doi:10.1016/j.camwa.2011.12.049.

[135] T. P. Banerjee, S. Das, Intelligent Fault Tracking by an Adaptive Fuzzy Predictor and a Fractional Controller of Electromechanical System–A Hybrid Approach, in: International Conference on Swarm, Evolutionary, and Memetic Computing, Springer, 2013, pp. 574–582. doi:10.1007/978-3-319-03756-1_51.

[136] M. Schwabacher, R. Waterman, Pre-launch diagnostics for launch vehicles, in: 2008 IEEE Aerospace Conference, IEEE, 2008, pp. 1–8.

[137] C.-M. Vong, P.-K. Wong, W.-F. Ip, Simultaneous faults diagnosis for automotive ignition patterns, in: 2011 International Conference on Machine Learning and Cybernetics, Vol. 3, IEEE, 2011, pp. 1324–1330. doi:10.1109/ICMLC.2011.6016890.

[138] F. Song, W. Hou, L. Shi, The information-enhanced fault diagnosis system design of avionics power supply module, in: Quality, Reliability, Risk, Maintenance, and Safety Engineering (QR2MSE), 2013 International Conference on, IEEE, 2013, pp. 1758–1761. doi:10.1109/QR2MSE.2013.6625916.

[139] L. Losik, Using Generic Telemetry Prognostic Algorithms for Launch Vehicle and Spacecraft Independent Failure Analysis Service, International Foundation for Telemetering, 2009.

[140] J. P. N. Gonzalez, L. E. G. Castañon, A. Rabhi, A. El Hajjaji, R. Morales-Menendez, Vehicle Fault Detection and Diagnosis combining AANN and ANFI, IFAC Proceedings Volumes 42 (8) (2009) 1079–1084. doi:10.3182/20090630-4-ES-2003.00178.

[141] A. von Birgelen, D. Buratti, J. Mager, O. Niggemann, Self-organizing maps for anomaly localization and predictive maintenance in cyber-physical production systems, Procedia CIRP 72 (2018) 480–485. doi:10.1016/j.procir.2018.03.150.

[142] A. Bregon, M. Daigle, I. Roychoudhury, G. Biswas, X. Koutsoukos, B. Pulido, An event-based distributed diagnosis framework using structural model decomposition, Artificial Intelligence 210 (2014) 1–35.

[143] S. M. N. A. Sunny and X. Liu and M. R. Shahriar , Remote monitoring and online testing of machine tools for fault diagnosis and maintenance using mtcomm in a cyber-physical manufacturing cloud, in: 2018 IEEE 11th International Conference on Cloud Computing (CLOUD), 2018, pp. 532–539. doi:10.1109/CLOUD.2018.00074.

[144] H. Fang, H. Shi, Y. Dong, H. Fan, S. Ren, Spacecraft power system fault diagnosis based on DNN, in: 2017 Prognostics and System Health Management Conference (PHM-Harbin), IEEE, 2017, pp. 1–5. doi:10.1109/PHM.2017.8079271.

[145] U. Schubert, U. Kruger, H. Arellano-Garcia, T. de Sá Feital, G. Wozny, Unified model-based fault diagnosis for three industrial application studies, Control Engineering Practice 19 (5) (2011) 479–490. doi:10.1016/j.conengprac.2011.01.009.

[146] L. Losik, Using data-driven prognostic algorithms for completing independent failure analysis, International Foundation for Telemetering, 2011.

[147] A. Khoukhi, M. H. Khalid, Hybrid computing techniques for fault detection and isolation, a review, Computers & Electrical Engineering 43 (2015) 17–32. doi:10.1016/j.compeleceng.2014.12.015.

[148] Y. H. Song, M. H. Kim, S. Lee, K. C. Lee, Implementation of a fuzzy predictive redundancy system for tolerance of x-by-wire systems, in: IECON 2010-36th Annual Conference on IEEE Industrial Electronics Society, IEEE, 2010, pp. 3141–3145. doi:10.1109/IECON.2010.5675027.

[149] H. Hu, Z. Li, A. Al-Ahmari, Reversed fuzzy Petri nets and their application for fault diagnosis, Computers & Industrial Engineering 60 (4) (2011) 505–510. doi:10.1016/j.cie.2010.12.003.

[150] K. Nagorny, S. Scholze, J. Barata, A. W. Colombo, An approach for implementing ISA 95-compliant big data observation, analysis and diagnosis features in industry 4.0 vision following manufacturing systems, in: Doctoral Conference on Computing, Electrical and Industrial Systems, Springer, 2016, pp. 116–123. doi:10.1007/978-3-319-31165-4_12.

[151] M. Schwabacher, R. Martin, R. Waterman, R. Oostdyk, J. Ossenfort, B. Matthews, Ares IX ground diagnostic prototype, in: AIAA Infotech@ Aerospace 2010, 2010, p. 3354. doi:10.2514/6.2010-3354.

[152] H. Fleischmann, J. Kohl, J. Franke, A Modular Architecture for the Design of Condition Monitoring Processes, Procedia CIRP 57 (2016) 410–415. doi:10.1016/j.procir.2016.11.071.

[153] J. Wang, L. Ye, R. X. Gao, C. Li, L. Zhang, Digital twin for rotating machinery fault diagnosis in smart manufacturing, International Journal of Production Research (2018) 1–15doi:10.1080/00207543.2018.1552032.

[154] D. Kurz, J. Kaspar, J. Pilz, Dynamic maintenance in semiconductor manufacturing using Bayesian networks, in: Automation Science and Engineering (CASE), 2011 IEEE Conference on, IEEE, 2011, pp. 238–243. doi:10.1109/CASE.2011.6042404.

[155] T. A. N. Heirung, A. Mesbah, Input design for active fault diagnosis, Annual Reviews in Control (2019). doi:10.1016/j.arcontrol.2019.03.002.

[156] P. Waszecki, M. Lukasiewycz, S. Chakraborty, Decentralized diagnosis of permanent faults in automotive e/e architectures, in: Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS), 2015 International Conference on, IEEE, 2015, pp. 189–196. doi:10.1109/SAMOS.2015.7363675.

[157] R. Pons, A. Subias, L. Travé-Massuyès, Iterative hybrid causal model based diagnosis: Application to automotive embedded functions, Engineering Applications of Artificial Intelligence 37 (2015) 319–335. doi:10.1016/j.engappai.2014.09.016.

[158] X. Yang, L.-j. Chen, Design and fault diagnosis of Petri net controllers for Petri nets with uncontrollable and unobservable transitions, Journal of Manufacturing Systems 28 (1) (2009) 17–22.

[159] M. P. Cabasino, C. Seatzu, C. Mahulea, M. Silva, Fault diagnosis of manufacturing systems using continuous Petri nets, in: 2010 IEEE International Conference on Systems, Man and Cybernetics, IEEE, 2010, pp. 534–539. doi:10.1109/ICSMC.2010.5642021.

[160] H. Schweppe, A. Zimmermann, D. Grill, Flexible on-

board stream processing for automotive sensor data, IEEE Transactions on Industrial Informatics 6 (1) (2009) 81–92. doi:10.1109/TII.2009.2037145.

[161] ISO/IEC, (E)-Road Vehicles Diagnostic Systems Keyword Protocol 2000, ISO/IEC, 2000.

[162] G. Rizzoni, S. Onori, M. Rubagotti, Diagnosis and prognosis of automotive systems: motivations, history and some results, IFAC Proceedings Volumes 42 (8) (2009) 191–202. doi:10.3182/20090630-4-ES-2003.00032.

[163] K. M. Janasak, R. R. Beshears, Diagnostics to Prognostics-A product availability technology evolution, in: Reliability and Maintainability Symposium, 2007. RAMS'07. Annual, IEEE, 2007, pp. 113–118. doi:10.1109/RAMS.2007.328051.

[164] M. Héder, From NASA to EU: The evolution of the TRL scale in Public Sector Innovation, The Innovation Journal 22 (2) (2017) 1–23.

[165] S. P. Feldman, Micro matters: The aesthetics of power in NASAs Flight Readiness Review, The Journal of Applied Behavioral Science 36 (4) (2000) 474–490. doi:10.1177/0021886300364005.

[166] M. Schwabacher, R. Aguilar, F. Figueroa, Using decision trees to detect and isolate simulated leaks in the J-2X rocket engine, in: 2009 IEEE Aerospace conference, IEEE, 2009, pp. 1–7. doi:10.1109/AERO.2009.4839691.

[167] N. Steinkamp, R. Levine, R. Roth, 2019 Automotive Defect and Recall Report (2019) [cited 18 October, 2019].
URL https://www.stout.com/en/insights/report/2019-automotive-defect-and-recall-report

[168] C. Ebert, J. Favaro, Automotive software, IEEE Software (3) (2017) 33–39. doi:10.1109/MS.2017.82.

[169] R. Hegde, G. Mishra, K. Gurumurthy, An insight into the hardware and software complexity of ECUs in vehicles, in: International Conference on Advances in Computing and Information Technology, Springer, 2011, pp. 99–106. doi:10.1007/978-3-642-22555-0_11.

[170] R. Saracco, Guess what requires 150 million lines of code, EIT Digital. Internet. Available: https://www.eitdigital. eu/news-events/blog/article/guess-what-requires-150-million-lines-of-code/(13 Jan. 2016) (2016).

[171] R. N. Charette, This car runs on code, IEEE Spectrum 46 (3) (2009) 3.

[172] O. Nasri, N. M. B. Lakhal, L. Adouane, J. B. H. Slama, Automotive decentralized diagnosis based on can real-time analysis, Journal of Systems Architecture (2019). doi:10.1016/j.sysarc.2019.01.009.

[173] Technavio, Global commercial vehicle remote diagnostics market 2018-2022. (2018) [cited 27 July 2019].
URL https://www.technavio.com

[174] M. Dragojević, S. Stević, G. Stupar, D. Živkov, Utilizing iot technologies for remote diagnostics of next generation vehicles, in: 2018 IEEE 8th International Conference on Consumer Electronics-Berlin (ICCE-Berlin), IEEE, 2018, pp. 1–4. doi:10.1109/ICCE-Berlin.2018.8576249.

[175] S. Fürst, M. Bechter, AUTOSAR for connected and autonomous vehicles: The AUTOSAR adaptive platform, in: 2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshop (DSN-W), IEEE, 2016, pp. 215–217. doi:10.1109/DSN-W.2016.24.

[176] MIMOSA, Machinery Information Management Open Systems Alliance (MIMOSA) (2019) [cited 29 June, 2019].
URL https://www.mimosa.org