Generic Fault-Diagnosis Strategy based on Diagnostic Directed Acyclic Graphs using Domain Ontology in Automotive Applications

Ali Behravan, Simon Meckel, Roman Obermaisser
University of Siegen, Siegen, Germany,
ali.behravan@uni-siegen.de, simon.meckel@uni-siegen.de, roman.obermaisser@uni-siegen.de

Abstract

In safety-critical systems the functionality in the presence of faults can be preserved using fault recovery in combination with robust diagnostic techniques. Fault-diagnosis is essential in many fault-tolerant control applications. In addition, low fault-positive and fault-negative rates are required in order to maximize the customer satisfaction and to reduce maintenance cost. The international standard ISO 26262 that is named functional safety for road vehicles is an adaptation of the international standard for electrical, electronic, and programmable electronic safety-related systems (IEC 61508). The automotive safety integrity level (ASIL) evaluates the failures based on severity, exposure, and controllability factors for assigning risk levels and is a key component of the ISO 26262. Embedded fault-diagnosis systems should be resource-efficient and simple for implementation with low-cost controllers, however, keeping a high diagnostic coverage. In the past, malfunction indicator lights (MILs) on the dashboard reported anomalies in a system but not the specific component. Nowadays, system-specific fault-diagnosis techniques are available, however, with the ongoing progress in vehicle's technology coming up with more and more complexity, describing a generic fault-diagnosis strategy is still challenging. In this paper two main goals are addressed. The first one is to get a full demonstration of the Fault-Error-Failure propagation in a domain ontology such as automotive ontology application which is also useful for the proposed diagnostic appraoch. Also this paper provides a solution for generic fault-diagnosis based on diagnostic directed acyclic graphs (DDAGs), distinguishing system, subsystem and component relationships thanks to the system ontology and Fault-Error-Failure propagation model. A key point is that the diagnosis system itself must be robust and able to overcome missing diagnostic inputs, e.g. missing sensor data, by proper signal substitutions. The approach is a part of condition-based maintenance (CBM) which uses run time data for fault-diagnosis. The diagnostic results are required to initiate repair or maintenance tasks prior to a failure.

1 Introduction

In the automotive domains, such as control, safety, information and entertainment systems, the complexity is becoming higher nowadays. This complexity induces an increasing failure potential and makes the automotive systems' maintenance very difficult. Due to this, the domain engineers are continuously facing new issues. Especially, the trend towards advanced driver assistance systems with the ambition to establish autonomous driving requires the vehicles to be equipped with various sensors to capture the current health status of the main systems and components as well as the surroundings. For the former, e.g. temperature, voltage or current sensors are employed, for the latter ultrasonic sensors, (infrared) cameras, RADAR, or LIDAR are used. For the real-time processing of the gathered information, powerful and reliable processing units are required to be installed and integrated into the car's E/E-architecture. Besides the existing safety-critical systems like the brakes, the newly introduced services are required to operate with a very high reliability, i.e. the overall reliability must be significantly higher than the reliability of the constituent components. Even in the case of a fault in a component, a subsystem, or system, a certain quality of the services must be ensured in order to enable the vehicle to manage difficult driving situations independently and overcome unexpected happenings. In short, the services must be fail-operational.

Fault-tolerance can be achieved in several ways. For instance, redundant sensors can be used to qualify the measured value for further processing via a voter, which allows a sensor to fail without dangerous consequences. Of course, this type of hardware redundancy is often not preferred due to weight, space, or cost limitations. In contrast, software redundancy is a feasible way to increase a system's reliability. For this, fault-detection and diagnosis capabilities are applied in order to isolate and overcome occurred faults in the system, preventing a service from failing. For instance, if a processing unit fails, the collapsed tasks can be rescheduled and executed on other processing units. In another typical scenario, if a sensor fails, the demanded data can be reconstructed from other sensor readings, e.g. through the fusion of multiple readings, potentially with further processing steps or a lower accuracy. For this, the

relationships and potential data fusion possibilities must be known, which is a difficult challenge since the modern mechatronic systems are very complex.

In this paper we highlight the possibilities of recovering failed sensor data for an online-diagnosis system utilizing Diagnostic Directed Acyclic Graphs (DDAG) in combination with semantic knowledge of ontology about the overall domain information considering a fault-error-failure propagation model. By this, we aim at automating the process of determining sensor data relations, i.e. which data can be recovered from other readings (from where) and how additional processing steps can be applied to the remaining data.

As the knowledge about potential ways to substitute certain sensor values from other sensor's readings is a complex task and can be formulated as an optimization problem, a genetic algorithm is a feasible and supportive instrument to solve this kind of problem: by providing appropriate boundary conditions in the form of a start solution via the ontology, as well as signal processing and merging tasks, e.g. integration, differentiation, interrelationships of signals are identified. This method has two advantages: (1) compared with a manual definition the process is faster as it directly uses the expert knowledge from an ontology; (2) it reveals solutions otherwise potentially overlooked by human system experts, e.g. if signal correlations between sensed data are weak. It is of major importance that the result of the genetic algorithm can be easily reviewed by human system experts. The method therefore serves as a supportive tool rather than a replacement of the well-established strategies used nowadays. Substitutions for failed sensor data must be immediately known by the system during run time to allow fast reaction. Hence, a genetic algorithm might be applied at different stages of the system development, e.g. during test phases, in parallel to normal system execution, or within a simulation environment.

Future developments yield systems where it is tremendously difficult, even for a team of human system experts, to overview the system as a whole, including all intended but also unintended (sub-)system interdependencies and interactions, especially in critical situations. However, the knowledge base that can be created and updated by human system experts is of uttermost importance. In particular, ontologies that merge the system knowledge on a higher semantic level form an important basis for the concept introduced in this paper. Future work in this project will not only concentrate on substituting failed sensor data, but also address the possibilities to find substitutions for whole diagnostic tasks utilizing the information from the ontology.

1.1 Related Work

Ratasich et al. [1] propose a very specific platform that assists structural adaptation and demonstrates its capabilities with an example from the automotive domain: a fault-tolerant system that estimates the state-of-charge (SoC) of the battery. In their platform they use the ontology to support the SoC estimator, however, in this paper the authors extend the concept to a generic fault-diagnosis strategy based on diagnostic directed acyclic graphs that includes all the fault cases and all the available signals. Besides, the literature offers a broad range of fault-diagnosis methods, e.g. [2], that can be applied in the automotive domain, where different methods suit differently well to different components, devices, or systems. Model-based fault-diagnosis, signal-based diagnosis, or knowledge-based diagnosis are widely utilized. For instance, in [3] model-based diagnosis was applied for electric drives and in [4] we find a robust fault-diagnosis technique for the traction system of an electric vehicle. Signal-based and data-based diagnosis method have been also applied by many researchers, e.g. [5], [6]. For system-level fault-diagnosis, component and signal interrelations can be modeled in a diagnostic directed acyclic graph, which is additionally able to combine multiple fault-diagnosis methods [7]. The goal of this DDAG is to take as many sensor readings and status variables into account, extract diagnostic features, and narrow down a faulty system behavior detected in the signals to the faulty subsystem, the component and if possible, specify the faulty sub-component, e.g. a defect cell in the battery of an electric vehicle. This procedure allows sound decisions on consequent recovery actions that prevent greater failure. These days it is an ongoing challenge to identify and prepare all important diagnostic features necessary for a fast and reliable fault-diagnosis. In [8] an automated feature selection method based on genetic algorithms for gear boxes is introduced.

All these papers demonstrate the working procedure of the fault-diagnosis method for the relevant application, however, it is assumed that the required sensor measurements are available. In this paper, we introduce a strategy to keep a fault-diagnosis system alive, even if sensor data is failing. This is done by an ontology-based automated evaluation of substitutions for failed sensor data.

2 Project Overview

Prior to the introduction of the main concept of this paper in Section 3, the following sections deal with the three main aspects important for the bigger picture understanding, i.e. (1) ontologies, (2) Fault-Error-Failure propagation model, and (3) fault-diagnosis based on DDAGs.

2.1 Ontology

In the field of computer science and information engineering, a set of representational principals are defined by an ontology to model a domain of knowledge. Implicit redundancy which is adaptable for unexpected failures as a new approach can be implemented using the ontology, instead of traditional explicit, but expensive, redundancy. Ontologies are useful to show explicit specifications of conceptualizations at the highest level of abstraction while keeping its clarity as high as possible. The main goal of an ontology is the creation of a model which proposes a common understanding among people with different specializations and expertise [9].

The other aspects of description of ontology is that recove-

ry action will be done using run time reconfiguration based on ontology. Reconfiguration includes addition, removing, replacing, changing interaction, substitution, and rearrangement of components. The information flow can be changed during the reconfiguration. Therefore, a comprehensive ontology plays a significant role.

Systems often include various subsystems that are connected together via an information flow or a communication network. The subsystems themselves embody several components e.g. hardware or software components. One component provides its service either to another component via a service interface or to another system (or subsystem) via a service interface. A service is the predestinate behavior of a component or system.

There are several studies describing a step-by-step approach to create the ontology in the automotive domain using the Web Ontology Language (OWL), protégé, and the HermiT to check and evaluate of the ontology [10], [11], [12], [13], [14], [15].

2.2 Fault-Error-Failure Propagation Model

Once all the systems and components are available, a system ontology is created. However, the form of interaction of different components and systems in case of faults will be unknown. This paper implements the fault-error-failure propagation concept into this ontology to get a deep understanding of the propagation flow in the whole domain ontology. A fault as an unpermitted deviation of at least one characteristic property of the system from the expected normal behavior [16] may arise due to internal or external effects. When a fault occurs in a component of a system, so-called the fault is activated, it may cause an incorrect or undefined system state that is named error [17]. If the error observably propagates internally in the component and causes deviation of the intended functionality from its specification, it may lead to a component failure and often the component stops its execution which is named fail-stop. Besides, the error may propagate from one component to another component via service interfaces. Therefore, the service delivered by component *A* to component *B* becomes incorrect and it originates an error into component *B* (fail-silent). This propagation may continue out a system to another system via system interface. That means in the ontology, the error may propagate from one subclass to another subclass. Figure 1 shows an ontology-based fault-error-failure propagation model for a vehicle example with different systems of safety, body electronics, driver information system, automotive networking/communication, chassis, and powertrain control. Each system includes a number of subsystems or components in it. For example, safety system includes airbags, electronic stability program, collision avoidance and adaptive cruise control. Body electronics system consists of body control module, seat, door, and window control, remote control, HVAC control, and lighting control. Driver information system embraces infotainment and telematics. Automotive networking/communication system includes communication systems, controller area network, and local interconnect network. Chassis system contains braking systems, electronic power steering, and active suspension. Powertrain control consists of engine control and transmission. Besides these systems two sample subsystems which are named "System *A*" and "System *B*"are placed into this figure to show a zoomed view of them including their components e.g. "Component *A*" and "Component *B*"and so on.

2.3 Fault-Diagnosis based on DDAG

For safety-critical systems it is crucial that the fault-diagnosis is performed online, i.e. at run time, in order to be able to immediately plan the healing or repairing steps. Since faults in components and systems are reflected in the monitored sensor measurements and state variables, faults can be detected from extracted diagnostic features as soon as an unintended signal behavior is discovered. A DDAG models the order of signals to be analyzed for the diagnosis and defines the relevant diagnostic procedures, e.g. limit or trend observation. The more information is processed during a diagnostic cycle, the higher becomes the confidence degree about the current health situation, and the potentially erroneous components, respectively. In [18] the authors introduce a Simulink model of a hybrid-electric vehicle (HEV) which serves as a test and evaluation platform to implement the algorithms for the proposed approach. According to an input driving cycle, i.e. a speed profile, the output signals of the various mechanical, electrical and control elements are monitored with sensors. By means of an established fault injection framework, faults of different severity can be induced during the simulation to make components drop in performance or fail. Based on this model, a library of fault-diagnosis methods was established in [7] with the goal to define standardized interfaces for the data exchange between the different diagnostic tasks in the DDAG. Besides, the automated generation of a first diagnosis model (the DDAG) was performed via machine learning techniques. The implemented online diagnosis system is then able to identify the induced faults by previously learned deviations in the signal patterns using a system to component approach. That means, at first, a faulty system or subsystem is identified before an internal component diagnosis can narrow down the fault further, e.g. isolating a defect cell in a battery of electric vehicles.

3 Concept Implementation

Assuming a fault-diagnosis system is missing a certain signal input, which is required to process diagnostic outcomes, a sound decision on occurred faults cannot be guaranteed. In automotive systems, we often find redundant sensor measurements, some of which are obviously highly correlated, e.g. rotational speeds of the four wheels. Electrical vehicles provide many electric signals as well that are required for diagnosis, e.g. the currents and voltages, and temperatures at the motors or the converters. Not always a correlation between different signals is obvious. Suitable algorithms can help to figure out even weak signal correlations and prepare replacement strategies for the DDAG in case of missing data. Figure 2 shows an excerpt of a DDAG

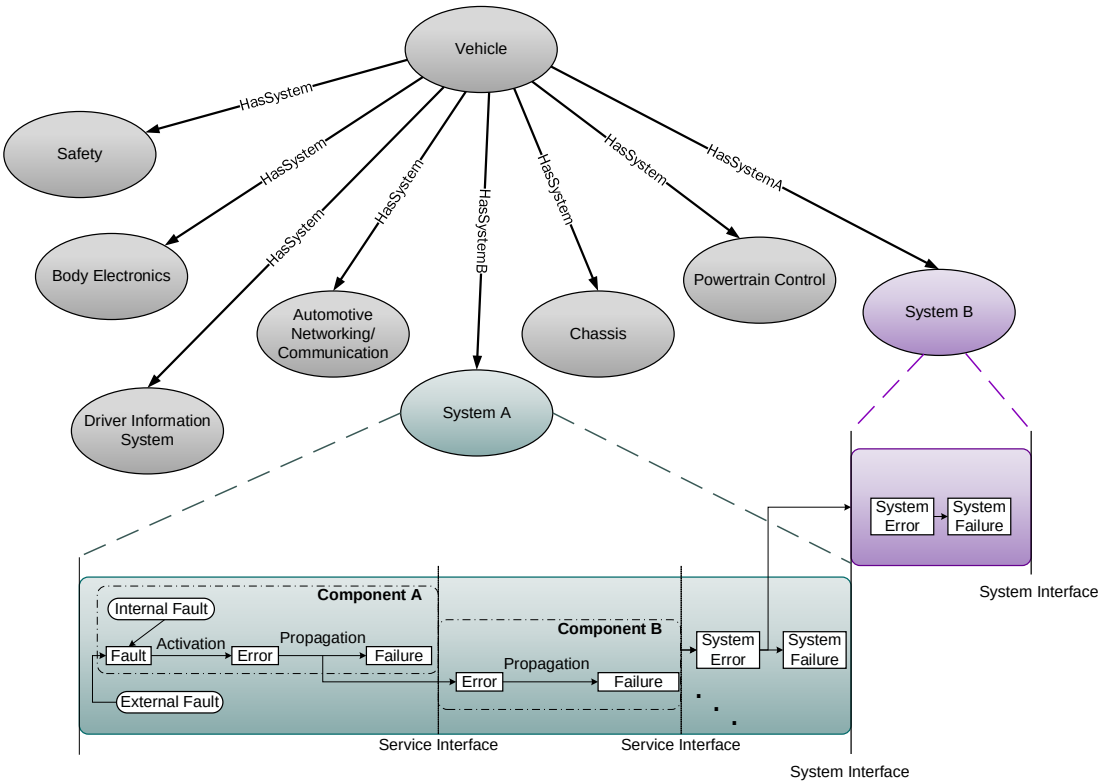

Figure 1 Ontology-based Fault-Error-Failure propagation model.

that processes a position value within a car, potentially derived from GPS. In case of missing this data stream it is the goal to substitute this value, which can be done, e.g. via an integration of the vehicle's speed measurements over time, originating from other sensors. The new values might not be as accurate, but may serve as replacements for the diagnosis system to continue to operate.

Of course, for these rather simple signals, finding replacement signals is easy. It is the goal of our strategy to use a semantic language to describe all the available signals and summarize them in the ontology. This is the first step towards a higher level description language that allows an easier reconfiguration of the diagnostic DAG during operation.

Once a faulty or missing signal value was detected, the DDAG uses the ontology-based fault-error-failure propagation model, described in Figure 1, to find the closest signal for the means of substitution with focusing on all the service interfaces and inputs/outputs of all the components around the faulty component. For example, if the output signal of component *B* in system *A* was detected faulty (which the DDAG needs for its calculations), this approach will focus on the service interface and input signals coming from component *A*.

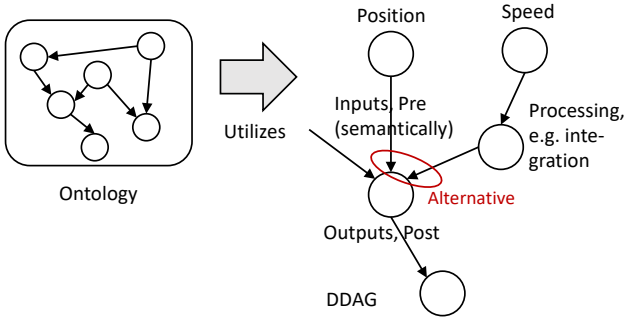

Figure 2 Finding substitutions for missed sensor values via the ontology

For future project steps the authors suggest the application of genetic algorithms for the substitute search. In contrast to machine learning techniques, the explicit semantic knowledge from the ontology can be directly exploited, e.g. by defining the start solution for the algorithm and providing suitable processing routines for the combination and fusion of sensor data. Considering the facts that genetic algorithms do not necessarily converge or provide the best possible solution, they show good results in short time if the boundary conditions are well defined. As a consequence, such an algorithm cannot perform a substitute search in real-time after a fault occurred, rather must the potential substitutes be precalculated, additionally under consideration of the time dependent signal behaviors, such that the dynamic reconfiguration of the DDAG is performed in real-time based on the precalculated possibilities.

4 Conclusion and Future Work

In this paper two main goals are addressed. The first one is to get a full demonstration of the fault-error-failure propagation in a domain ontology such as automotive ontology application. The second goal is a promising solution for a

fail-operational system thanks to a reconfigurable DDAG method using the domain ontology. The implementation of this solution can be done via a suitable method, e.g. a genetic algorithm is a candidate to precalculate the interrelationships of different components to find the best fits among all the signals coming from all the devices. Thanks to the domain ontology which helps to find the optimum solution by showing all the systems, components, and their information flows and connections for the goal of data substitution in case of a faulty system, component, or signal.

5 Acknowledgment

This work was supported by the DFG research grants LO748/11-1 and OB384/5-1.

6 Literature

[1] D. Ratasich, O. Höftberger, H. Isakovic, M. Shafique, and R. Grosu, "A self-healing framework for building resilient cyber-physical systems," in *Real-Time Distributed Computing (ISORC), 2017 IEEE 20th International Symposium on*, pp. 133–140, IEEE, 2017.

[2] R. Isermann, *Fault-diagnosis systems: an introduction from fault detection to fault tolerance*. Springer Science & Business Media, 2006.

[3] Y. L. Murphey, M. A. Masrur, Z. Chen, and B. Zhang, "Model-based fault diagnosis in electric drives using machine learning," *IEEE/ASME Transactions On Mechatronics*, vol. 11, no. 3, pp. 290–303, 2006.

[4] M. A. Djeziri, R. Merzouki, and B. O. Bouamama, "Robust monitoring of an electric vehicle with structured and unstructured uncertainties," *IEEE transactions on vehicular technology*, vol. 58, no. 9, pp. 4710–4719, 2009.

[5] H. Guo, J. A. Crossman, Y. L. Murphey, and M. Coleman, "Automotive signal diagnostics using wavelets and machine learning," *IEEE transactions on vehicular technology*, vol. 49, no. 5, pp. 1650–1662, 2000.

[6] H. Henao, G.-A. Capolino, M. Fernandez-Cabanas, F. Filippetti, C. Bruzzese, E. Strangas, R. Pusca, J. Estima, M. Riera-Guasp, and S. Hedayati-Kia, "Trends in fault diagnosis for electrical machines: A review of diagnostic techniques," *IEEE industrial electronics magazine*, vol. 8, no. 2, pp. 31–42, 2014.

[7] S. Meckel and R. Obermaisser, "Component-based combination of online-diagnosis methods using diagnostic directed acyclic graphs," in *2018 7th Mediterranean Conference on Embedded Computing (MECO)*, pp. 1–5, IEEE, 2018.

[8] M. Cerrada, R. V. Sánchez, D. Cabrera, G. Zurita, and C. Li, "Multi-stage feature selection by using genetic algorithms for fault diagnosis in gearboxes based on vibration signal," *Sensors*, vol. 15, no. 9, pp. 23903–23926, 2015.

[9] T. R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing?," *International journal of human-computer studies*, vol. 43, no. 5-6, pp. 907–928, 1995.

[10] A. Mallak, C. Weber, M. Fathi, and A. Holland, "Active diagnosis automotive ontology for distributed embedded systems," in *Technology and Engineering Management Summit (E-TEMS), 2017 IEEE European*, pp. 1–6, IEEE, 2017.

[11] I. Niles and A. Pease, "Towards a standard upper ontology," in *Proceedings of the international conference on Formal Ontology in Information Systems-Volume 2001*, pp. 2–9, ACM, 2001.

[12] M. Uschold and M. Gruninger, "Ontologies: Principles, methods and applications," *The knowledge engineering review*, vol. 11, no. 2, pp. 93–136, 1996.

[13] "protégé - a free, open-source ontology editor and framework for building intelligent systems." http://protege.stanford.edu, 2017. [Accessed: 2017-04-07].

[14] L. W. Lacy, *OWL: Representing information using the web ontology language*. Trafford Publishing, 2005.

[15] "Hermit owl reasoner." http://www.hermit-reasoner.com, 2017. [Accessed: 2017-06-12].

[16] A. Behravan, R. Obermaisser, D. H. Basavegowda, and S. Meckel, "Automatic model-based fault detection and diagnosis using diagnostic directed acyclic graph for a demand-controlled ventilation and heating system in simulink," in *Systems Conference (SysCon), 2018 Annual IEEE International*, pp. 1–7, IEEE, 2018.

[17] A. S. Tanenbaum and M. Van Steen, *Distributed systems: principles and paradigms*. Prentice-Hall, 2007.

[18] S. Meckel, R. Obermaisser, and J.-U. Yang, "Generation of a diagnosis model for hybrid-electric vehicles using machine learning," in *2018 21st Euromicro Conference on Digital System Design (DSD)*, pp. 389–396, IEEE, 2018.