vehicles

Article

An Innovative Technique for Fault Analysis of Electric Automated Vehicles

Shiqing Li *, Michael Frey and Frank Gauterin

Institute of Vehicle System Technology, Karlsruhe Institute of Technology (KIT), Kaiserstraße 12, 76131 Karlsruhe, Germany; michael.frey@kit.edu (M.F.); frank.gauterin@kit.edu (F.G.)
* Correspondence: shiqing.li@partner.kit.edu

Abstract: A serious accident might occur as a direct result of a defective component or system, especially at an automated vehicle. The objective of this study is to establish a fault analysis methodology that can effectively improve the reliability of electric and automated vehicles. To achieve this, We proposed a method for identifying potential faults by combining Failure Modes and Effects Analysis (FMEA) with Fault Tree Analysis (FTA), creating a database of possible (example) faults that maps the causal relationship between causes, symptoms and faults, which enables more thorough fault analysis and serves as the foundation for further study. Using the fault database, we demonstrate a practical application involving fault injection and simulation, which can provide a more intuitive and practical representation of the effects of faults. The methodology is validated with the demonstrator vehicle from the joint project. This approach is scalable and can also be well applied to other electric automated vehicles with similar structure, providing a reliable tool to the system fault analysis for future work.

Keywords: fault analysis; electric automated vehicle; FMEA; FTA; database; fault diagnosis; fault identification

Citation: Li, S.; Frey, M.; Gauterin, F. An Innovative Technique for Fault Analysis of Electric Automated Vehicles. *Vehicles* **2024**, *6*, 1995–2010. https://doi.org/10.3390/vehicles6040098

Academic Editor: Lihui Zhao

Received: 7 October 2024
Revised: 10 November 2024
Accepted: 20 November 2024
Published: 26 November 2024

1. Introduction

The occurrence of a malfunction or failure in a specific component or subsystem has the potential to result in a serious vehicle accident. There has been a reduction in the number of car accidents, but this trend is now being challenged by the development and growing acceptance of electric automated vehicles. The high reliability of electric and autonomous vehicles depends on competence in safety analysis, which includes fault analysis. The tasks associated with fault analysis include the definition of potential system faults, the identification and understanding of the root causes of these faults, the possible effects of the occurrence of these faults and the assessment of the risks. Systematic identification of potential faults can assist in further fault diagnosis and provide timely hazard warning. A variety of approaches such as FMEA (Failure Modes and Effects Analysis), FTA (Fault Tree Analysis), STPA (System Theoretic Process Analysis), etc. can be used to identify potential faults in mechanical, mechatronic and electrical systems. However, a theoretical analysis of the potential faults of a system is often not sufficient for the task of fault analysis, because the theoretical analysis cannot accurately assess the hazard that a fault can cause. The only solution is to combine theoretical analysis with practical application to ensure that the consequences of faults can be analysed as accurately as possible, with multiple possibilities and scenarios, without causing damage or danger to the environment or to oneself.

In response to these challenges, this study develops a comprehensive fault analysis approach that combines both theoretical and practical methodologies. Specifically, a novel variant based on FMEA and FTA is used to uncover holistic failure modes within electric automated vehicles. To create the foundation for a comprehensive FTA, this approach begins with a condensed FMEA and helps users identify important parts and functions. This basis is then developed into a thorough FMEA that identifies complex system-level

faults and component-level faults, making it suitable for both inexperienced users and seasoned engineers. A fault database is created to systematically record all identified faults and to facilitate quick access to previous analyses. To extend the practical applicability of the fault database, a database based fault injection and simulation technique is employed. This approach combines a structured fault database with fault injection and simulation, offers practical benefits for enhancing fault analysis and system reliability in electric automated vehicles. By systematically cataloging faults, causes, and effects, the database can streamline the diagnostic process, reducing both time and cost, provides a centralized, organized resource that enables both engineers and less-experienced users, such as vehicle operators to identify and prioritize potential risks quickly. In addition, the fault injection and simulation capabilities built on this database provide a safe, controlled testing environment through the connection with a digital twin of the vehicle. This enables engineers to virtually simulate faults without risking damage to the physical vehicle, which is particularly valuable during the design and development phase. By simulating a range of fault scenarios can provide a more intuitive understanding of the impact of faults in the fault database and the generated extensive data supports also the further refinement of diagnostic algorithms. The ability to simulate faults also allows engineers to identify vulnerabilities, prioritize critical faults, and develop redundancy strategies, ultimately contributing to safer, more robust vehicle designs.

Section 2 presents an overview of the current methodologies, highlighting their respective advantages and disadvantages. Section 3 introduces the fundamental procedures of the novel variant of the fault identification method. To illustrate the general faults of the demonstrator, an electric automated vehicle is employed as a case study, after which a fault database is constructed to document all pertinent fault information. Furthermore, this chapter elucidates fault injection and simulation techniques based on the fault database. Section 4 offers a synthesis of the preceding chapters and a prospective outlook.

2. Related Approaches

Failure Modes and Effects Analysis (FMEA) is a method of examining potential reliability problems in the early stages of product development within the product life cycle. It is easier to address faults as they occur and thus improve the reliability of the product [1]. This technique aims to identify fault causes, fault modes, the impact of faults on the product's operation, and actions taken to mitigate them [2]. FMEA primarily focuses on the most critical fault modes and analyses them. FMEA can be extended to encompass the severity of faults, referred to as FMECA (Failure Mode, Effects and Criticality Analysis), which assesses the gravity of the fault's impact and its likelihood of occurrence as well as the priorities for each recorded failure [2]. It was initially developed by the US military in 1949 to assess the impact of failures on mission success and safety [3], and found early use in the Apollo programme in the 1960s [4]. The automotive industry began to use it in the 1970s, with Ford particularly using it to improve the safety of its Pinto model [5]. In 1980, FMEA was standardized in Germany as DIN 25448 and after revisions by the German Association of the Automobile Industry, it became widely adopted in the automotive industry.

Fault Tree Analysis (FTA) is a deductive technique that identifies the causes of a specific undesired event, the TOP event [6]. This approach is frequently utilised to address reliability problems in industrial systems. FTA structures usually consist of three layers: complex system events at the top, functions, subsystems or system failures in the middle and basic events and component failures at the bottom. H.A. Watson created FTA in 1962 at Bell Labs to promote the Minuteman I ICBM launch control system [7], and was widely adopted by engineers as a failure analysis technique [8]. Since the 1960s, the Reliability Analysis Center at Rome Laboratory has provided FTA and reliability block diagram materials [9]. The 1986 Space Shuttle Challenger disaster highlighted the importance of Fault Tree Analysis (FTA) and Probabilistic Risk Assessment (PRA) in identifying system risks and improving safety assessments at NASA [10]. Nowadays, FTA is a key methodology for system reliability analysis that is applied across various disciplines.

FMEA and FTA employ contrasting approaches in failure analysis. FMEA takes a bottom-up approach, identifying component-level failures and their causes, whereas FTA adopts a top-down approach, starting with system-level failures and then tracing them to components. Each technique has its strengths and limitations, which are summarized in Table 1. Both methods are time-consuming, especially for unknown systems, but understanding the system early pays back in the future. FTA's benefit is in its multilevel structure and straightforward diagrams, while FMEA provides comprehensive information and evaluates the severity of failures. FTA scrutinises logical and causal relationships from the top of the tree, while FMEA builds a library of all possible failures and their consequences. The integration of these methods often provides more profound insights, as presented in Table 1, which minimises many limitations. The integration of FTA and FMEA is applied in various fields. For example, Peng et al. (2012) utilized them for analyzing failures in diesel engine combustion systems [11]. Tang et al. (2011) employed this combination for analyzing braking systems [12], and Han et al. (2013) applied it to early identification of software faults [13].

Table 1. Comparison the characteristics between FTA, FMEA and FMEA + FTA (× implies that the approach has no advantage in this attribute, while √ implies that the method does).

	FMEA	**FTA**	**FMEA + FTA**
timeconsuming in preparation	×	×	×
timesaving when using	√	√	√
clear level of analysis	×	√	√
structured approach	×	√	√
required experience of the process	√	×	√
visualization	×	√	√
connection of faults and causes	√	√	√
criticality of faults	√	×	√
applicability for complex system	×	√	√
consideration of human and Software fault	×	√	√
analyze the interaction of multiple systems	×	√	√
identifies non-failure related events	×	√	√
ability to focus on important components	√	×	√
comprehensive and detailed analysis	√	×	√
sufficient implementation example for reference	√	√	√

Combining FMEA and FTA can be accomplished through two methods: separately or mixed [14]. The separate approach conducts both methods independently, providing a thorough analysis but requiring more time and effort, possibly on less critical components [14]. The mixed approach comprises two categories [15]. The positive combination technique identifies causes using FMEA, choosing the top event with the highest criticality for the fault tree. The reverse combination technique identifies specific undesirable events as top events in the fault tree, taking into consideration failure descriptions and system requirements. This approach is more efficient, but requires detailed knowledge of the system. Furthermore, some research is still being done to develop conservative combination techniques. Ref. [16] have developed a novel approach that starts with the highest level fault tree event, iteratively applies FMEA to select the next level based on the highest risk, and continues down to the lowest level.

From the analysis of the current fault identification methods, it is easy to realise that no method is flawless. Conducting a comprehensive fault analysis requires significant investment of time. Nonetheless, this initial effort proves profitable in future applications as it thoroughly covers all potential failures. Furthermore, due to the development and interdisciplinary character of electric automated vehicles, there is a need for a flexible approach that allows people with different levels of expertise to access detailed and effective fault analysis information. The following chapter will examine the provided method in more detail.

3. Fault Analysis Technique

The fault analysis technique in this work can be divided into three parts: potential fault identification, fault database creation, fault injection and simulation.

3.1. Potential Fault Identification

The process of fault identification can be divided into six steps: analyze the architecture of the demonstrator, determine the system limits and precondition for fault identification, classify the crucial components, create FMEA information library, construct demonstrator FTA, determine failure mechanisms. Required sources for this process include technical models of the demonstrator, technical datasheets of components, literature research, etc. This information is relatively simple and can be analyzed by most people on their own.

To apply the proposed method, a demonstrator vehicle is used as an example. The architecture of the demonstration vehicle has to be decided first. This paper presents a modular architecture for electric automated vehicles, as illustrated in Figure 1, comprising seven essential subsystems: High-Level Control Unit, which is responsible for planning routes, interpreting remote driving instructions, and converting the instructions into speed and steering angle; the Low-Level Control Unit manages the vehicle's behavior, providing target torque to the actuators for execution; the Actuator System consists of motors for driving and steering tasks; the Sensor System includes environmental perception sensors (e.g., cameras and radars) and automotive sensors for monitoring vehicle performance parameters; the mechanical components system comprises linkages connecting actuators to the chassis and interaction with the vehicle's surroundings; the power supply system primarily uses the battery as the vehicle's energy source; the safety system is also the emergency braking system, which is essential for rapid deceleration in emergencies. This architecture provides a multi-layered foundation for electric automated vehicle development, ensuring effective control, safety and performance, especially for automated driving functions.

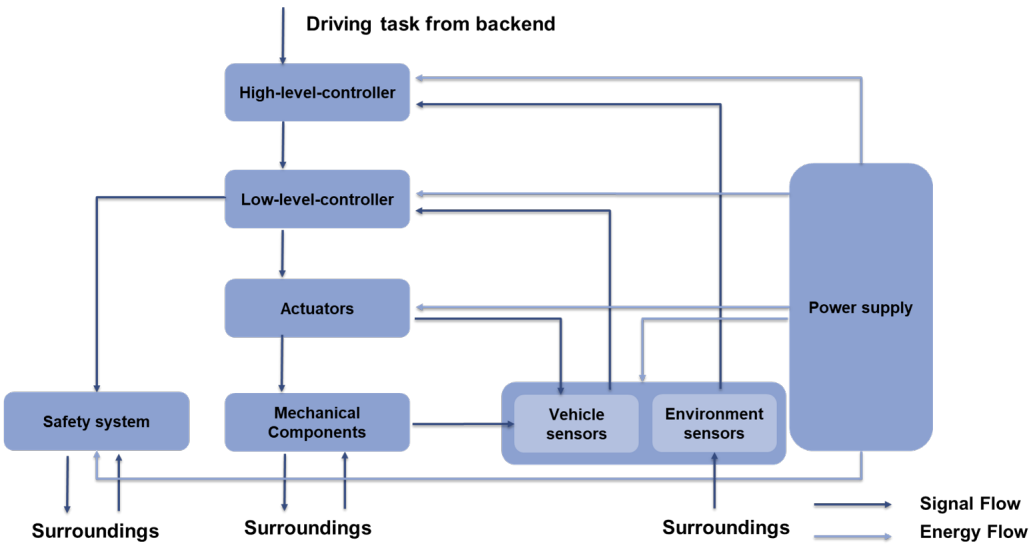

Figure 1. General architecture of an electric automated vehicle, which consists of seven subsystems: The direction of signal flow is represented by the dark blue arrow, indicating the direction in each subsystem transmits and receives information. The light blue arrow shows the direction of energy flow, illustrating the flow of electrical energy from each subsystem.

Once the general vehicle architecture has been determined, it can be subdivided into the seven subsystems described above, with each subsystem having clearly defined system boundaries. These subsystems operate independently and a fault in any of them will lead to a malfunction of the whole vehicle.

In the process of identifying potential faults, the analysis of faults within a particular subsystem must exclude consideration of the effects of faults on other subsystems. For

example, when evaluating actuator subsystem faults, any issues in the low-level controller subsystem (such as motor control unit faults) are not considered. In order to avoid confusion, this approach assumes that only one subsystem fails at a time during fault analysis.

Once the system architecture and its boundaries have been defined, it's important to examine the subsystem components with technical drawings of the demonstrator and component technical data sheets. The main objective here is to classify significant vehicle parts into the seven predefined subsystems, providing an organised framework for analysis and evaluation.

Following component classification, functions are identified. A simplified FMEA is then applied to identify potential component failure mechanisms. Unlike the full FMEA, the simplified version focuses on probable consequences and causes, forming a simple failure chain consisting of three elements: cause of failure, type of failure and resulting consequences. The simplified FMEA serves as an initial analysis for each component, eliminating non-essential parts to save time in subsequent analyses. Components that have a significant impact on subsystems are selected for detailed analysis, building on the preliminary understanding gained at this stage.

The next stage is to construct a fault tree for the entire system (shown in Figure 2). This process starts with identifying high-level system issues, followed by identifying subsystems based on their impact on system-wide faults. In each subsystem, faults and their associated components are identified. The simplified FMEA simplifies the component-level fault mode analysis and facilitates the final step in the fault tree analysis. By starting at the top of the system and proceeding systematically downwards, we can uncover previously unnoticed failures. For example, while a simple FMEA may ignore wire connection failures when evaluating the performance of single motor, the fault tree analysis, which considers the performance of the entire system, easily identifies such problems.

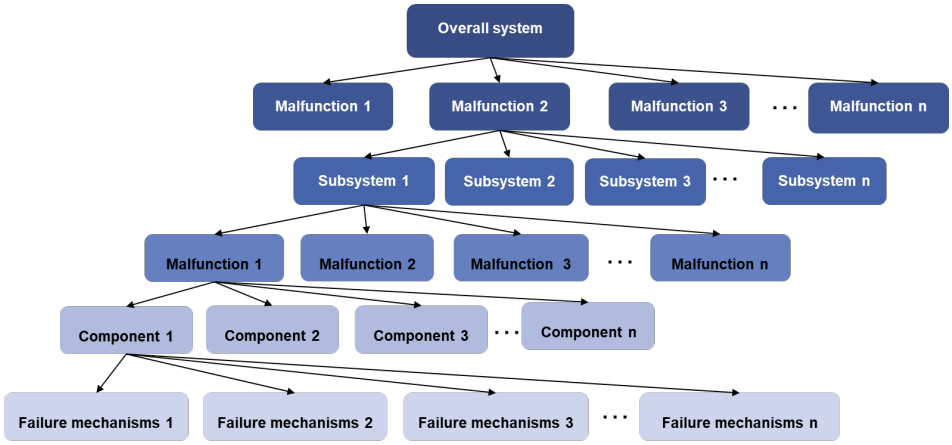

Figure 2. Structure of the fault tree from overall system level to component level.

The entire system fault tree leads to a detailed FMEA, which is the next step. The detailed FMEA includes specific details such as fault location or function, a brief description, detection measures, upper event identification and causes, and lower events associated with the fault. Another essential part of the detailed FMEA is the calculation of the Risk Priority Number (RPN), which combines three factors: severity, probability of occurrence and probability of detection, typically rated on a scale from "1" (low risk) to "10" (high risk). The RPN helps to prioritise faults, allowing the focus to be on critical problems. The FMEA in this step is performed with the help of FTA, so it is not limited to component analysis, but also performed for each layer of the entire system. For example, if a car's driving function failed resulting in a track deviation, FTA identifies subsystems associated with the system fault. FMEA then identifies the most critical subsystem likely to fail based on the FTA findings. Next, it identifies the components associated with the subsystem

fault, pinpointing the most likely faulty components and probable causes. The FMEA also records all failure causes and detection actions.

3.2. Application to Exemplar Vehicle

This chapter provides an illustrative application of the methodology described in Section 3.1. The example vehicle used is a demonstrator from the "SmartLoad" joint research project [17]. As mentioned above, electric automated vehicles have a similar structural framework. Accordingly, the demonstrator is divided into seven subsystems as in Figure 1. In the case of system-wide failures, the cause can be traced back to one of these subsystems.

The demonstrator's high-level controller is tasked with automation, information exchange, and trajectory determination, while environmental sensing is managed by the sensor system, as illustrated in Figure 1. The low-level controller includes motor control units and the autobox, which separately controls the steering and drive motors and manages the trajectory. The actuators comprise two steering and two drive motors, providing the drive and steering functions, as well as redundancy for the steering. Sensors are divided into driving condition sensors (e.g. steering angle, torque, wheel speed, temperature, IMU, tie rod force) and environmental sensors (e.g. camera, lidar). The power supply module contains two batteries: a 12 V battery that powers the control units and sensors, and a 48 V battery that powers the steering and drive motors and charges the 12 V battery. Mechanical components transmit motor torque and interact with the road to move the vehicle. In addition, a hydraulic braking system provides emergency braking and stopping.

The components of the demonstrator were categorised into different subsystems to identify their roles. A simplified FMEA was then performed to identify the relationships between components and functions. An illustrative example is shown in Figure 3, which details motor failures, causes and potential consequences, each row corresponds to a fault chain. An electric motor consists of several components. The faults in the stator winding, such as a short-circuit between windings or damage to the insulation material, can cause the windings to burn, overheat or fluctuate in torque. The problems in the stator core, such as a loose connection to the motor or local overheating, resulting in vibration, noise or an increase in no-load current. The rotor fault, including cracks in the brackets or eccentricity, will create a one-sided magnetic pull, causing vibration or friction between the rotor and stator and damaging the motor. The rotor may also be partially or completely demagnetised, resulting in a reduction in torque. In addition, bearing failures also cause vibration [18]. If the cooling system fails, the motor will experience a rise in temperature, resulting in motor malfunction.

The general fault tree topology for a complex system is stated in Figure 2. If a vehicle is a system as a whole, then malfunctions include improper longitudinal driving, deviating cornering, unexpected mistake, emergency stop, etc. Each malfunction is the result of a subsystem failure. Each malfunction of the problematic subsystem is likewise caused by the failure of one of the components, and the failure of each component is triggered by various bottom events, thus constituting a full vehicle fault tree.

Cause	Failure	Consequence
Incorrect assembly	Loose stator core	Vibration and noise increase
Cooling failure	Local overheating	No-load current increase
Damage to the insulation	Short circuit between winding	Winding burned and torque fluctuations
Incorrect assembly	Rotor eccentricity	Cause vibration, friction between the rotor and the stator
Overheating	Rotor demagnetization	Torque reduction

Figure 3. An example of the part of the simplified FMEA for actuator.

After analyzing the whole demonstrator, obtaining a comprehensive fault tree is possible. Below, just the partial instances were provided. Figure 4 depicts the higher levels

of the fault tree, which range from the top events of the entire system (entire vehicle) to the component level. Figure 5 depicts a portion of the electric actuator's fault tree. The actuator's faults are a lack of torque due to total breakdown or incorrect torque output due to partial damage. Each malfunction relates to distinct component failure mechanisms. Through the fault tree, one can visualize the vehicle's overall defects and the links between the various issues.

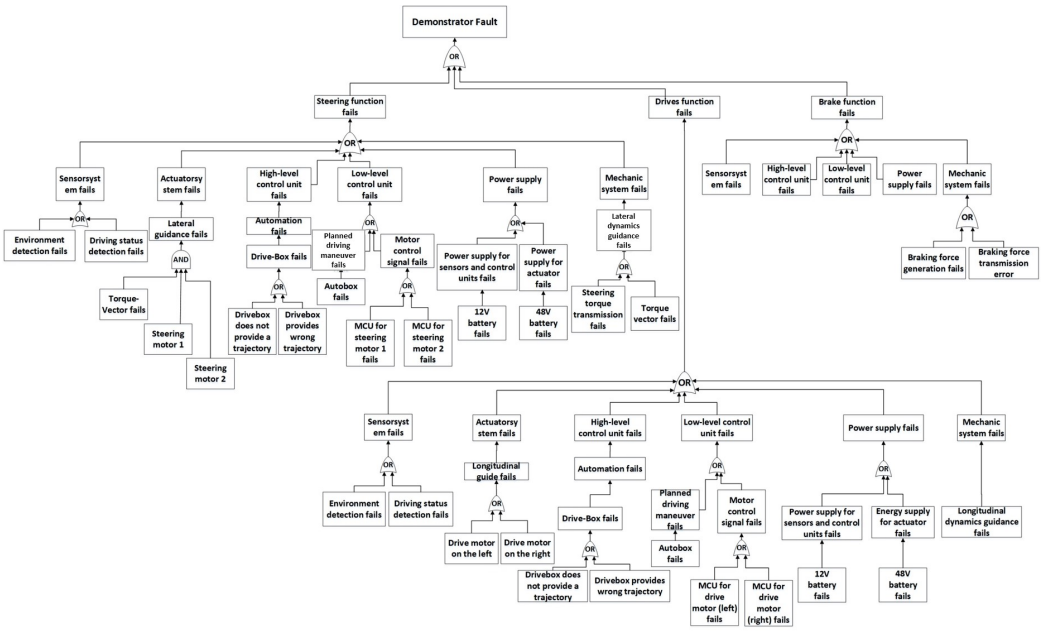

Figure 4. Fault tree of the overall system: The upper level of the fault tree is the vehicle fault event and the lower level is the key component fault event.

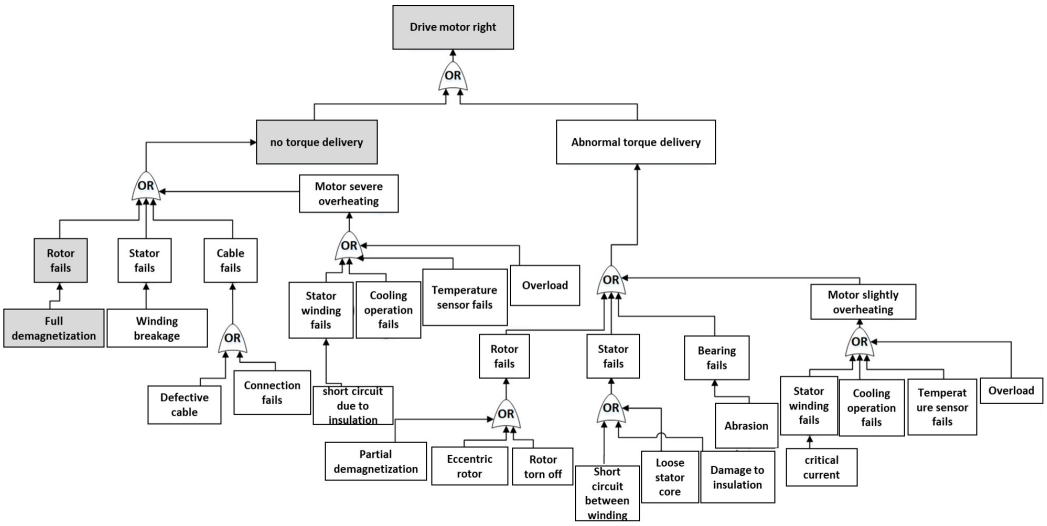

Figure 5. Drive actuator fault tree: The upper level of the fault tree is the key component fault event and the lower level is failure mechanism for the key component. The rectangles marked in grey correspond to the example given in Section 3.2.

An illustration has been provided to assist interpretation. If the vehicle is not running at the desired speed, one of the actuators may be faulty. The demagnetisation of the rotor could have a significant effect on the malfunction of the actuator. The reasons for demagnetisation could be further analysed, such as extremely high temperature due to cooling system failure or inaccurate temperature measurement due to sensor failure. As

illustrated in Figure 5, the relationship between the variables is evident within the grey-squared area.

Following the creation of the fault tree, a thorough FMEA was carried out under the guidance of the fault tree analysis. This enables a comprehensive FMEA to be performed on the whole vehicle and facilitates the construction of the database. Figure 6 is an example of a partial FMEA for the drive system.

System/ Component/ Function	System/ Component/ Function description	Fault name	Consequences/ Upper Events	Causes/ intermediate events	Detection measures	Occurrence	Severity	Probability of detection	RPZ
Longitudinal drive actuators	Longitudinal drive actuators contain 2 Heizmann motors, which are used to drive the demonstrator	Longitudinal drive actuators failure	Longitudinal movement failure; Drive failure	1.Heizmann motor left fails; 2.Heizmann motor right fails	Check both motors, observe longitudinal movement of the demonstrator.	8	7	7	392
Heizmann Motor(left)	Heizmann motor (left) drives the left tire	Heizmann motor left fails	Longitudinal drive actuators failure> Longitudinal movement failure> Drive failure	1.Heizmann motor (left) does not deliver torque; 2.Heizmann motor (left) delivers abnormal torque	Check Heizmann motor (left) before operation; motor tests;	8	7	7	392
Heizmann Motor (right)	Heizmann motor (right) drives the right tire	Heizmann motor right fails	Longitudinal drive actuator failure>Longitudinal movement failure> Drive failure	1.Heizmann engine (right) delivers no torque; 2.Heizmann motor (right) delivers abnormal torque;	Check Heizmann motor (right) before operation; motor tests;	8	7	7	392
Deliver Torque (left)	Heizmann Motor (left) does not deliver torque	No torque (Heizmann motor (left))	Heizmann motor left failure> Longitudinal drive actuator failure> Longitudinal movement failure> Drive failure	1. Rotor failure; 2. Winding failure; 3. Cable failure; 4. Motor overheating;	Check important components of the Heizmann motor (left); motor tests;	6	9	9	486
Deliver Torque (right)	Heizmann Motor (right) does not deliver torque	No torque (Heizmann engine (right))	Heizmann motor right Failure> Longitudinal drive actuator failure>Longitudinal movement failure> Drive failure	1. Rotor failure; 2. Winding failure; 3. Cable failure; 4. Motor overheating;	Check important components of the Heizmann motor (right); motor tests;	6	9	9	486

Figure 6. Example excerpt from the detailed FMEA for an actuator.

3.3. Create the Fault Database

The vehicle is a highly complex system with many potential problems. To efficiently identify and display faults in specific subsystems or components, the next stage of fault analysis is to establish a fault database. Microsoft Access is chosen as the database platform because of its user-friendly interface and compatibility with the data in the spreadsheet. The user interface design of the fault database as Figure 7 consists of two sections: the input field (in red box) and the display field (in green box). Users can search the database by entering search terms such as "FAULTCODE", "FAULTNAME", "COMPONENT" or "CONSEQUENCE" in the input area and clicking the "SEARCH" button. The display field shows detailed information, including item descriptions, fault causes, consequences, detection methods, risk level ratings, etc.

The fault database possesses various practical applications that enhance the efficiency and accuracy of fault identification. For example, engineers can conduct searches targeting specific outcomes, such as "steering failure." Inputting this word prompts the database to display the principal event associated with the issue, frequently a malfunction in the lateral drive actuator. Technicians can utilize the "Next" button to identify associated defects across different subsystems, for example the high-level controller. This consequence-based search method offers a transparent perspective on the evolution of faults and aids in identifying fault chains that may result in significant system failures. The database enables targeted searches within particular subsystems, such as identifying faults in the mechanical subsystem associated with steering. By inputting a combination of search terms, such as "M-" (denoting the mechanical subsystem) and "steering failure", the database narrows results to display only mechanical defects associated with steering failure, facilitating targeted troubleshooting within particular vehicle sectors. Other method like component specific searches, allowing users to enter a specific part name (e.g., "Rotor") to investigate any issues associated with that component in the demonstrator vehicle. This level of information enables engineers to track fault progression and identify direct causes or effects

within the system, so easing root cause analysis and assisting in the prioritization of repairs or interventions for critical components. The database enhances diagnostic efficiency, enables systematic problem identification, and ultimately increases vehicle performance reliability. To effectively manage and analyze these faults, a structured selection mechanism is employed, primarily based on the Risk Priority Number (RPN). The RPN is a critical metric that combines three factors: severity, probability of occurrence, and probability of detection. Each factor is rated on a scale from "1" (low risk) to "10" (high risk), with the overall RPN calculated as the product of these three values. This quantifiable measure allows for a systematic approach to fault prioritization, ensuring that attention is directed toward the most critical issues.

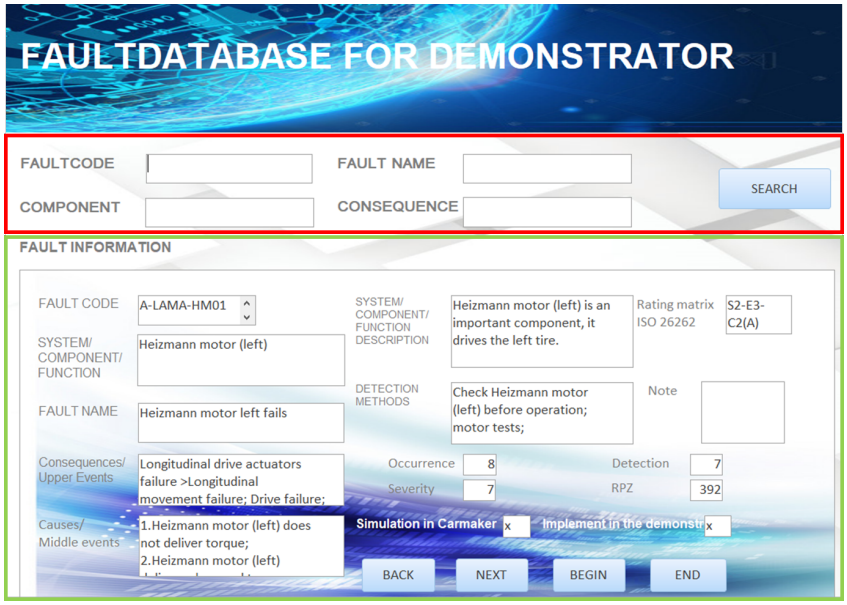

Figure 7. The user interface of the fault database for the demonstrator: The database can be searched by entering search terms such as "FAULTCODE", "FAULTNAME", "COMPONENT" or "CONSEQUENCE" in the input field (in the red box) and the display field (in the green box) shows detailed fault information.

Figure 8 shows an example of abnormal driving performance. In this scenario, the demonstrator initially follows its planned path, but suddenly deviates to the right from the expected path. The comparison between the shadowed shape (representing normal vehicle performance) and the solid shape (representing an abnormal state) indicates this deviation. Based on the fault identification analysis, potential problems may exist in the mechanical component, the steering actuator or the low level controller, as depicted in Figure 9. The RPN analysis gives first priority to the actuator system. According to the database, a complete breakdown of the steering motor is one type of malfunction, with possible causes including issues with the rotor, winding, or high motor temperature. For more detailed information on the causes of motor failure and associated inspection or prevention actions, users can click the 'NEXT' button or enter the event name in the search bar. This provides access to comprehensive information and maintenance guidelines to address the underlying events and related issues.

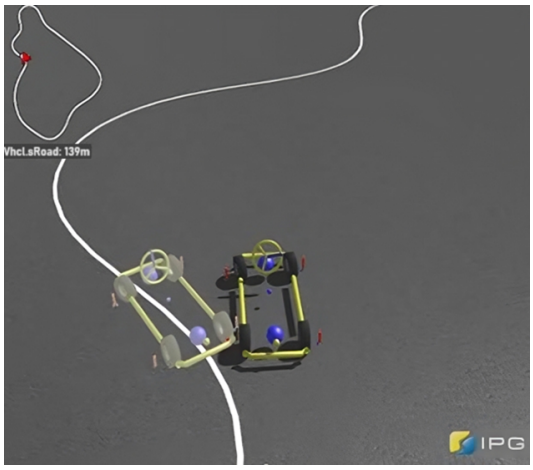

Figure 8. An example of the abnormal driving of the vehicle (The shaded vehicle indicates that the vehicle is in the state of normal driving without fault, and the solid vehicle indicates that it is in the fault state of deviating from the desired track).

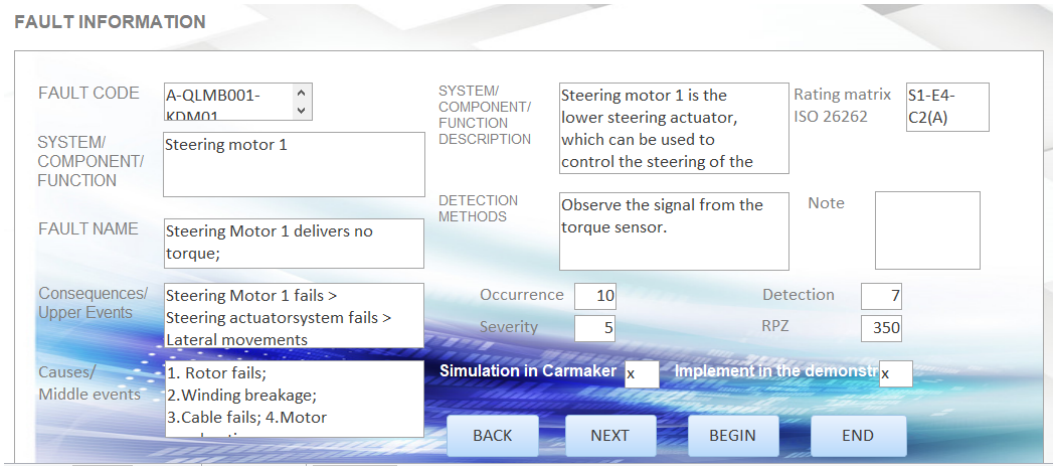

Figure 9. An Example of the information about the actuator faults in the database: The information shows the root causes of the possible fault in the steering motor and the impact of the fault event on the upper levels.

3.4. Fault Database Based Fault Simulation

Electric automated vehicles have the potential to transform road transport, but their reliability is challenged by the complexity of driving tasks and unpredictable situations. Although hazard analysis techniques are essential, the consequences of failures are often unknown in advance. To ensure the highest reliability, a comprehensive approach is required. This includes knowledge-based analysis and test-based research methods such as fault injection and simulation [19]. Fault simulation is the equivalent of creating a digital twin of the real vehicle in a virtual environment, which helps to understand system behaviour in fault conditions and to analyse how subsystems and components interact during faults. Simulated testing reduces time and cost, complementing real-road experiments. This approach plays a significant role in improving the reliability and safety of autonomous vehicles. We have presented a fault database based on the described fault identification method. The function of a fault database with sufficient information should not only provide information but also have some practical implementations, such as supporting fault injection and simulation. Fault database-based fault injection and simulation offer additional advantages over conventional modeling, including systematization and flexibility.

In the previous chapter we used Microsoft Access to create the fault database. Although it's convenient and adaptable for creating applications, its speed and connectivity are limited. To overcome such limitations, we have chosen to utilise MySQL as the backend

database for fault injection and simulation due to its improved connectivity with Simulink and data acquisition capabilities. The data transfer was completed using an ODBC connection, allowing to export tables directly from Access to MySQL. Our simulation environment uses CarMaker for Simulink, which provides comprehensive vehicle and traffic models. To integrate the database, we connected MATLAB to MySQL for efficient data retrieval. To simplify this process during testing, we have developed a user-friendly GUI using the toolbox "App Designer" in MATLAB R2018b, as Figure 10, linked to MySQL through Java Database Connectivity. To enable fault injection and simulation, a Simulink model is developed that can trigger fault injection by input mode. This mode also includes features such as injected time, fault severity and parameter adjustments. These functions can be accessed through the GUI, which improves visibility and usability. In addition, the GUI offers an real-time visualisation of key variables such as acceleration, steering angle and vehicle speed, which helps to understand how faults affect the vehicle's behaviour.

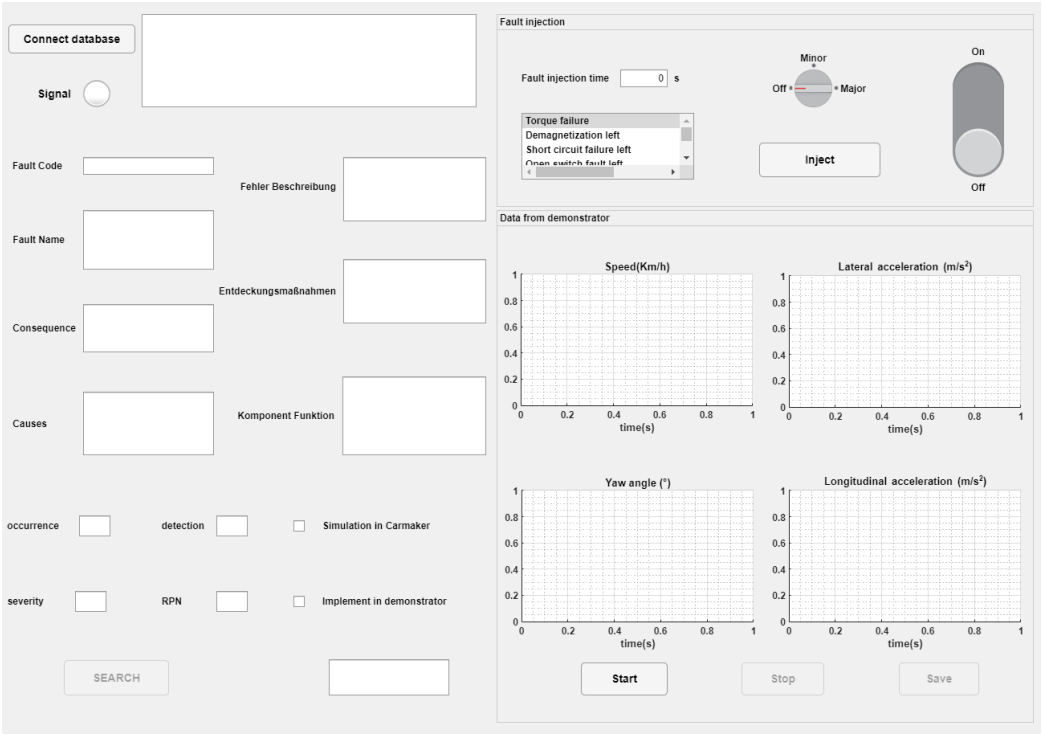

Figure 10. The GUI for the fault injection and simulation, which connected the fault database and simulation modell in Simulink.

Here we presented the implementation of this approach for injecting and simulating demagnetization faults in electric motors, building on the fault model developed by [20] as part of the "SmartLoad" project. Originally, ref. [20] developed a mathematical model to analyze multiple open-switch faults within two-level converters that supply Permanent Magnet Synchronous Motors (PMSM). This model facilitates the examination of fault impacts on electrical current by configuring specific switch states within the converter, hence permitting a comprehensive analysis of how open-switch faults affect motor performance through modified converter outputs. The model computes phase currents and switching patterns, modeling the dynamic reactions of the PMSM under fault conditions to evaluate converter and motor fault tolerance.

We modified the model of [20] for our application by including a magnetic flux parameter to qualitatively depict the demagnetization process in permanent magnet synchronous motors. This adaption illustrates typical phenomena, such as variations in torque and current, that occur during a demagnetization event, rather than offering an exact demagnetization model. Demagnetization, which diminishes motor torque production, may result from physical damage, thermal stress, adverse magnetic fields, or aging [21].

Our fault database, associated with this revised model, facilitates systematic fault injections and adaptable simulation configurations. Users possess complete control over the timing and intensity of the fault injection via the GUI. The injection time can be configured to simulate the beginning of demagnetization at a designated operational point, while severity is calibrated through changing magnetic flux levels, with increased severity indicating more significant reductions in flux. In the fault simulation, magnetic flux parameters are manipulated. If the demagnetization fault level is set to "Off" via the GUI and the fault injection time is set to 0 s (which represents a non-fault scenario), the courses of the motor moment with (purple dash-dot line) and without (blue solid line) fault are observed in the GUI display, as shown in Figure 11, and the course of current with (purple dash-dot line) and without (blue solid line) fault are shown in Figure 12. When the electric motor rotor is demagnetized at 0.5 s, it necessitates increased current to meet the torque requirement. The augmented current subsequently amplifies the demagnetization action, resulting in a further increase in current. Despite torque compensation, oscillations persist approximately 0.5 s. It should be noticed that the strength of demagnetization doesn't halt the vehicle instantaneously, but the increase in current does cause secondary damage.

The mechanism behind this instability lies in the feedback cycle triggered by reduced magnetic flux. A reduction in magnetic flux, which causes the motor controller to boost stator current in order to maintain torque. This decline indicates that the motor requires an increased current, given that torque is proportional to flux. A feedback loop is thus established, as the increased current also produces more heat, thereby hastening demagnetization. In addition to increasing the risk of motor wear, this cycle causes torque instability and oscillations, which demonstrate that even when the controller can temporarily compensate, the motor remains susceptible to additional harm as demagnetization advances.

This paper aims not to attain precise fault modeling, but to demonstrate an effective approach for database-driven fault injection. Future model modifications may integrate more sophisticated flux variations, including the simulation of flux variation with temperature during slow demagnetization. The primary objective in this paper is to establish a practical framework for fault injection, equipping engineers with a versatile tool to investigate fault impacts and response techniques, rather than delivering the ultimate fault model.

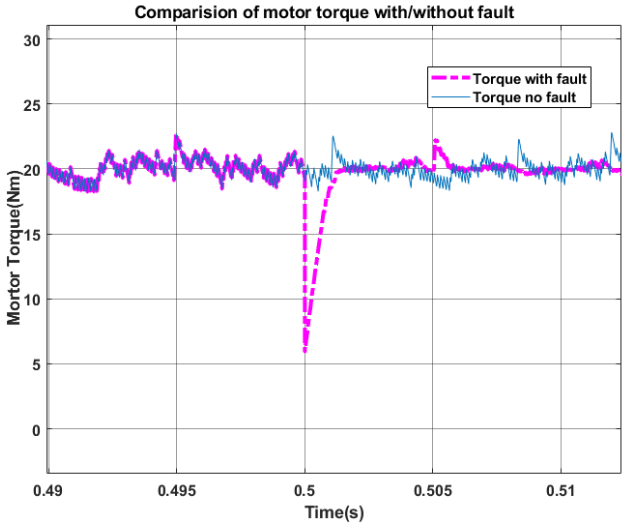

Figure 11. Comparison of motor torque under demagnetization failure and without failure (Purple dash-dot line: the torque with fault, blue solid line: the torque without fault): at the 0.5 s, the motor hat demagnetization failure, the torque changed momentarily, due to the reduced electromagnetic intensity, the current will increase until the required torque is reached, so the torque will continue to return to its normal value.

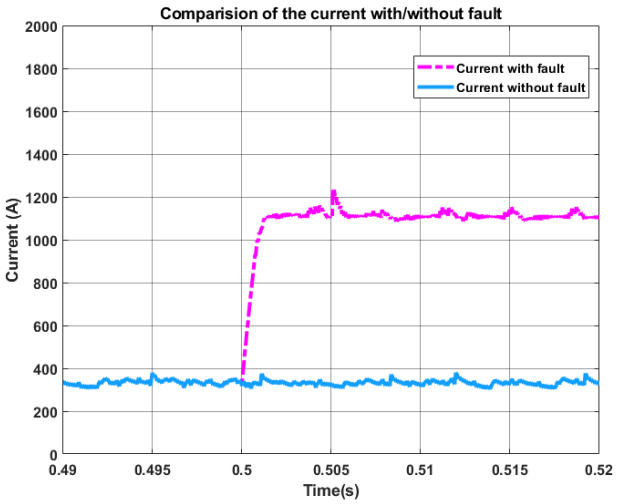

Figure 12. Comparison of current under demagnetization failure and without failure (Purple dash-dot line: the current with fault, blue solid line: the current without fault): at the 0.5s, the motor hat demagnetization failure, the current changed momentarily, due to the reduced electromagnetic intensity, the current must increase to ensure the required torque is reached, but the increased current will cause further damage to the motor.

Here we offered also another example, for presentation the application of the fault database based fault injection. We also modified the model of [20] for our another example by simulating short-circuit fault focuses on the motor's converter. A critical component responsible for converting direct current (DC) to the three-phase alternating current (AC) required by the permanent magnet synchronous motor (PMSM). In the simulated environment, the converter consists of six insulated-gate bipolar transistors (IGBTs), arranged to control each phase of the current flow. By manipulating the configuration of these transistors, specific fault conditions such as short circuits can be introduced. For this short-circuit scenario, one phase of the motor's windings is intentionally disconnected, forcing the motor to operate with only the remaining two phases, which affects both the motor's torque output and overall stability. This injection can also be applied in the models through Matlab S-function. When the short-circuit fault of the converter is set through the fault database GUI, the faulty module will be adjusted to the current state as required. At this time, the fault will be added in 0.5 s, where "phase number" in the operation GUI is "1" means that only one phase is short-circuited. At this situation, the virtual vehicle had an oscillating change in motor torque, as shown in Figure 13.

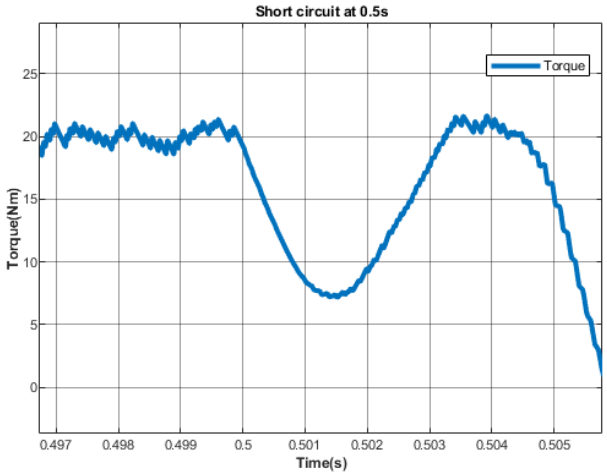

Figure 13. The motor torque oscillation because of the injection of one phase short-circuit fault at the 0.5 s.

The torque oscillations happen because the motor is constantly trying to balance the current across the remaining active phases after one phase is lost to the short circuit. With the current flow disrupted, the motor can't maintain a stable electromagnetic field, which leads to repeated torque fluctuations. This imbalance doesn't just make the motor unstable, it also adds extra strain on the components and generates more heat, which could speed up wear and increase the chance of failure. This example shows how a short-circuit fault can throw off motor stability, pointing to the need for strong converter design and solid fault response strategies in electric automated vehicles.

4. Summary and Outlook

A general method of fault identification for electric automated vehicles has been developed in this study. The method combines FMEA (Failure Modes and Effects Analysis) and FTA (Fault Tree Analysis) and can provide comprehensive information about possible faults for those who do not have sufficient understanding of the complex system. By conducting a simple FMEA analysis, one can identify the different components and functions, and establish a basis for the FTA. It is possible to quickly create the fault trees for an entire car and perform then a detailed FMEA for the whole system. A fault database is built using the information from the detailed FMEA to clean up and store the results of fault identification. The database can be used to reduce the time spent on problem finding, and a GUI makes it simple to retrieve the relevant information. For instance, by looking for a certain motor failure, users can get a list of potential causes. The diagnosis can then be ranked using severity levels. By simplifying diagnostics and reducing errors, this well-organized reference benefits not only professionals but also drivers or users with limited technical knowledge. The database can assist less experienced users in identifying potential issues before undertaking costly diagnostic procedures by providing convenient access to fault information. Moreover, The fault tree and database developed in this work can be easily transferred to other electric automated vehicles, as the structure of electric automated vehicle is similar. The linking of the fault database to the car model in Matlab enables the fault injection and simulation. This transition from theoretical study to practical application adds a practical layer to theoretical analysis, enabling virtual testing of faults in a digital twin environment, allowing for safe and reasonably priced virtual simulation testing of faults, since it is challenging to recreate direct component failures during vehicle development. For instance, using virtual simulation, we discovered that motor torque is greatly impacted by short-circuit defects in converter, which is crucial information for redundancy design. Engineers can improve design robustness and reliability by better anticipating failure modes using fault scenario simulation throughout the development stage. This setup provides crucial insights for developing redundancy strategies and improving system reliability, laying a foundation for comprehensive fault monitoring and robust fault-response designs in electric automated vehicles.

We recognize several limitations and issues inherent to this approach. The method's temporal and resource requirements might be substantial, due to the effort necessary to develop a comprehensive fault information for intricate vehicles. This early investment facilitates comprehensive fault identification but may provide difficulties for projects with constrained deadlines or resources. Scalability poses difficulties, particularly when applying this strategy to vehicles with varied designs or sophisticated technologies such as AI-driven control systems. Nonetheless, the more simple architecture of electric vehicles facilitates design modifications that mitigate this disadvantage, rendering the technique more applicable across diverse vehicle platforms. For example, due to the modular design, the technique can adapt to a vehicle with a different number of motors without requiring significant modifications to the fault identification framework. This can be achieved by simply adding or removing instances of the "motor module" within the drive system. The efficacy of fault injection and simulation is fundamentally reliant on the correctness of the used model, as a precise model facilitates more dependable forecasts of fault effects on the vehicle system. Although restrictions may occur if the database does not encompass

specific scenarios or environmental variables, frequent updates with empirical failure data can facilitate the alignment of simulations with real-world settings. Our approach's primary contribution is the integration of the FMEA- and FTA-based fault database with a digital twin, thereby augmenting the fault database's practical utility as a dynamic resource for real-time fault injection and simulation. This integrated methodology provides a more thorough understanding of fault pathways and interdependence compared to isolated FMEA or FTA or traditional pure theoretical fault identification methods. Although methodologies such as STPA (System Theoretic Process Analysis) effectively tackle software-related challenges and the interactions between human operators and automated systems, they frequently fall short in providing the component-level specificity required for attaining high dependability in electric vehicle applications. Anticipating the future, the incorporation of our methodology with STPA may enhance its flexibility for AI-driven vehicles that are predominantly software-dependent, thereby extending its applicability to highly automated vehicles.

This research provides a robust foundation for an intelligent fault monitoring technique in automotive systems, with a particular emphasis on the initial stages of problem recognition, the development of a comprehensive fault database, and the injection and simulation of faults. These approaches establish the basis for advanced processes, including fault detection and decision-making, which are based on the process of identifying potential faults and simulating critical failures, as outlined in this study. For readers interested in a comprehensive framework for fault monitoring, our other studies [22,23] clarify the subsequent research works, which offer a comprehensive approach for monitoring faults in vehicular systems.

Author Contributions: Methodology, S.L.; Software, S.L.; Validation, S.L.; Writing—original draft preparation, S.L.; writing—review and editing, M.F.; supervision, F.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been supported by the Federal Ministry for Education and Research (BMBF) in the project "New methods to increase the reliability of highly electric automated vehicles (SmartLoad)" with the funding reference number: 16EMO0362

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Acknowledgments: We acknowledge support by the KIT-Publication Fund of the Karlsruhe Institute of Technology.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

FMEA	Failure Modes and Effects Analysis
FTA	Fault Tree Analysis
STPA	System Theoretic Process Analysis
FMECA	Failure Mode, Effects and Criticality Analysis
PRA	Probabilistic Risk Assessment
RPN	Risk Priority Number

References

1. DIN EN. *Analysetechniken für die Funktionsfähigkeit von Systemen–Verfahren für die Fehlzustandsart-Undauswirkungsanalyse (FMEA)*; Deutsches Institut für Normung, DIN EN: Berlin, Gemany, 2006; Volume 60812.
2. Mraz, M.; Huber, B. *FMEA-FMECA*; University of Ljubljana: Ljubljana, Slovenia, 2005.
3. USDD. *MIL-P-1629—Procedures for Performing a Failure Mode Effect and Critical Analysis*; United States Department of Defense: Arlington, VA, USA, 1949.

4. Office of Manned Space Flight, Apollo Program; Office, Q.A. *Procedure for Failure Mode, Effects and Criticality Analysis (FMECA)*; NASA: Washington, DC, USA, 1966. Available online: https://ntrs.nasa.gov/api/citations/19700076494/downloads/19700076494.pdf (accessed on 22 November 2024).
5. Matsumoto, K.; Matsumoto, T.; Goto, Y. Reliability analysis of catalytic converter as an automotive emission control system. *SAE Trans.* **1975**, *84* , 728–738.
6. Geymayr, J.A.B.; Ebecken, N.F.F. Fault-tree analysis: A knowledge-engineering approach. *IEEE Trans. Reliab.* **1995**, *44*, 37–45. [CrossRef]
7. EricsonII, C. Fault tree analysis—A history. In Proceedings of the 17th International System Safety Conference, Orlando, FL, USA, 16–21 August 1999 .
8. Martensen, A.L.; Butler, R.W. *The Fault-Tree Compiler*; Technical Report; NASA: Washington, DC, USA, 1987.
9. Sharma, P.; Singh, A. Overview of fault tree analysis. *Int. J. Eng. Res. Technol.* **2015**, *4*, 337–340.
10. Stamatelatos, M.; Vesely, W.; Dugan, J.; Fragola, J.; Minarick, J.; Railsback, J. *Fault Tree Handbook with Aerospace Applications*; NASA: Washington, DC, USA, 2002.
11. Peng, W.; Yu, L.; Li, Y.; Liu, Y.; Huang, H.Z. Application of FMECA and FTA in the fault diagnosis for diesel engine. In Proceedings of the 2012 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering, Chengdu, China, 15–18 June 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 641–643.
12. Tang, T.; Lu, Y.; Zhou, T.T.; Jing, H.L.; Sun, H. FTA and FMEA of braking system based on relex 2009. In Proceedings of the International Conference on Information Systems for Crisis Response and Management (ISCRAM), Harbin, China, 25–27 November 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 106–112.
13. Han, X.; Zhang, J. A combined analysis method of FMEA and FTA for improving the safety analysis quality of safety-critical software. In Proceedings of the 2013 IEEE International Conference on Granular Computing (GrC), Beijing, China, 13–15 December 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 353–356.
14. Cristea, G.; Constantinescu, D.M. A comparative critical study between FMEA and FTA risk analysis methods. *IOP Conf. Ser. Mater. Sci. Eng.* **2017**, *252*, 012046. [CrossRef]
15. Jianfeng, T.; Shaoping, W.; Yiping, Y.; Peiqiong, L. Reliability analysis on combination of FMECA and FTA for redundant actuator system. In Proceedings of the Gateway to the New Millennium. 18th Digital Avionics Systems Conference, St. Louis, MO, USA, 24–29 October 1999; Proceedings (Cat. No. 99CH37033); IEEE: Piscataway, NJ, USA, 1999; Volume 1, pp. 3.B.2-1–3.B.2-6.
16. Peeters, J.; Basten, R.J.; Tinga, T. Improving failure analysis efficiency by combining FTA and FMEA in a recursive manner. *Reliab. Eng. Syst. Saf.* **2018**, *172*, 36–44. [CrossRef]
17. Neue Methoden zur Zuverlässigkeitssteigerung von Hochautomatisierten Elektrischen Fahrzeugen (SmartLoad); Teilvorhaben: Szenarienbasierte Validierung von Hochautomatisierten Elektrischen Fahrzeugen. Available online: https://www.tib.eu/de/suchen/id/TIBKAT:1814793585/Neue-Methoden-zur-Zuverl (accessed on 27 September 2022).
18. Liwei, Z.; Xianjin, H.; Yannan, Y.; Chen, X.; Jie, L. Summarize of Electric Vehicle Electric System Fault and Fault-tolerant Technology. *TELKOMNIKA Indones. J. Electr. Eng.* **2014**, *12*, 1094–1099. [CrossRef]
19. Juez, G.; Amparan, E.; Lattarulo, R.; Rastelli, J.P.; Ruiz, A.; Espinoza, H. Safety assessment of automated vehicle functions by simulation-based fault injection. In Proceedings of the 2017 IEEE International Conference on Vehicular Electronics and Safety (ICVES), Vienna, Austria, 27–28 June 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 214–219.
20. Pecha, U.; Parspour, N.; Gieck, F. Modeling and analysis of multiple open-switch faults in converters supplying permanent-magnet synchronous machines. In Proceedings of the 2021 IEEE 13th International Symposium on Diagnostics for Electrical Machines, Power Electronics and Drives (SDEMPED), Virtual, 22–25 August 2021; IEEE: Piscataway, NJ, USA, 2021; Volume 1; pp. 240–246.
21. Lee, J.; Jeon, Y.J.; Choi, D.C.; Kim, S.; Kim, S.W. Demagnetization fault diagnosis method for PMSM of electric vehicle. In Proceedings of the IECON 2013-39th Annual Conference of the IEEE Industrial Electronics Society, Vienna, Austria, 10–13 November 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 2709–2713.
22. Li, S.; Frey, M.; Gauterin, F. Evaluation of Different Fault Diagnosis Methods and Their Applications in Vehicle Systems. *Machines* **2023**, *11*, 482. [CrossRef]
23. Li, S.; Frey, M.; Gauterin, F. Model-based condition monitoring of the sensors and actuators of an electric and automated vehicle. *Sensors* **2023**, *23*, 887. [CrossRef] [PubMed]