Adaptive fault detection and emergency control of autonomous vehicles for fail-safe systems using a sliding mode approach

Jongmin Lee1,3, Kwangseok Oh2, Member, IEEE, Youngmin Yoon1, Taejun Song3, Taehoon Lee1 and Kyongsu Yi1, Member, IEEE
1Department of Mechanical and Aerospace Engineering, Seoul National University, Seoul 08826, South Korea
2School of ICT, Robotics & Mechanical Engineering, Hankyong National University, Anseong-si 17579, South Korea
3Department of research engineering at Smart Mobility Lab, SML, CO., LTD, Seoul 08826, South Korea.

Corresponding author: Kyongsu Yi (kyi@snu.ac.kr).

This work was partly supported by the Engineering Research Institute at the College of Engineering of Seoul National University, Seoul, South Korea. This work also was supported by the Brain Korea 21 FOUR Project in 2022. This research also was supported by KOSEF/SNU-IAMD. This work was partly supported by Future Mobility Technical Center (FMTC) of Seoul National University, Seoul, South Korea. This work also was supported in part by the Program (0690-20200039, Sensor Fusion with 4D Image Radar and Commercialization Elements Analysis for Automated Vehicle Demonstration) funded by the Ministry of Science and ICT, South Korea.

ABSTRACT This paper presents a sliding mode-based adaptive fault detection and emergency control algorithm for implementation in fail-safe systems of autonomous vehicles. The overall algorithm is comprised of a fault detection part and a fail-safe control part. For the former, sliding mode observer-based fault detection algorithms were developed for environment and chassis sensors, including LiDAR, Radar, and acceleration sensors. Unidentified fault signals from the sensors are reconstructed through the adaptive sliding mode observer. The reconstruction is based on the MIT rule through the use of an estimated sensitivity parameter. For the latter, a sliding mode control (SMC)-based emergency control method designed to respond to fault occurrences has been proposed to ensure the functional safety of autonomous vehicles. An adaptive gain parameter was designed, taking convergence time into consideration, to secure consistent and rapid responses from the controller. When the detection algorithm detects a fault, the appropriate control input is computed by a lower controller for the vehicle. This control input is calculated based on the last scene information obtained from an upper controller. The performance of the proposed fault detection and control algorithms has been evaluated through simulations and actual vehicle tests of various scenarios.

INDEX TERMS Adaptive sliding mode observer; Autonomous vehicle; Fail-safe; Fault detection; Fault reconstruction; Sliding mode control.

I. INTRODUCTION
There is no contest regarding the safety of autonomous vehicles being of the utmost importance within the industry. The inherent nature of autonomous vehicles brings forth the issue that even a minor vehicle fault may lead to serious, possibly fatal, accidents. Hence, studies on fail-safe systems (also known as fallback systems) are particularly important for autonomous vehicle research. Autonomous vehicles are equipped with various equipment such as sensors, computers, and actuators. While the various parts are communicating with each other, monitoring the condition of each part is essential; Fail-safe systems form decisions based on these conditions. For autonomous vehicle sensors, abnormal conditions can be identified with relative ease as manufacturers provide essential information regarding the sensor conditions. However, a general solution to the autonomous vehicle fail-safe system still proves to be challenging as the hardware and software configurations differ depending on the vehicle [1]. Each vehicle system requires a corresponding and independent fail-safe structure. Because numerous companies and research institutes independently develop their own autonomous vehicles, the challenge of developing a universal fail-safe system persists.

Prior studies have shown that several vehicle companies have developed a dual-system and fallback system for safety. For example, a fallback system can be found in Google Waymo’s autonomous vehicles. Waymo's fallback system is composed of 4 backup systems: Backup computing, backup...
braking, backup steering, and backup power [2]. GM CRUISE, on the other hand, has a backup computer, backup actuators, and redundant signal communication [3]. Baidu’s Apollo autonomous vehicle has a guardian module and a fail-safe system. The open-source nature of the Apollo software structure was applied to all control commands through the guardian module [4]. The studies mentioned so far introduce the various independent fail-safe methodologies applied to different autonomous vehicles.

Other areas of studies apart from autonomous vehicles have also shown progress in fail-safe systems through the use of fault diagnosis and countermeasure methods for system malfunction or performance degradation. Of such, fault detection and diagnosis methods have been used in fail-safe structures to develop the stability and reliability of their fail-safe systems. Fault diagnosis usually utilizes the concept of hardware and software redundancy. The hardware redundancy concept relies on the usage of 2 pieces of the same hardware with the same input. The output signals of both pieces of hardware are compared to detect faults. However, the downside to this method is cost and limitations on mounting locations; Environment sensors such as LiDARs, RADARs, and Cameras cannot be installed on the exact same locations. Consequently, the progression of fault diagnosis has led to software redundancy (analytical redundancy) methods becoming the mainstream of fault detection and diagnosis study since the 1980s [5]. Some examples of software redundancy implementation are given below. Fault reconstruction and detection were achieved through the use of a sliding mode observer with parameters obtained from the LMI technique [6]. Fault diagnosis on applications for satellite formation flight was done through a robust unknown input observer. In this case, the isolation of faulty actuators was done while taking model uncertainty, input, and output environmental disturbance into account [7]. Fault estimation and fault-tolerant control problems of a class of switched stochastic systems were characterized using an independently designed observer; sensor faults were estimated and used for fault diagnosis. The dimensions of the designed observer were reduced when compared to existing results. [8] The advantages and practical implications of artificial intelligence (AI) were reviewed in rotating machinery. [9] Motor bearing condition monitoring and fault diagnosis via a vibrational signal were achieved using a wireless sensor network’s amplitude and frequency data [10]. Real-time monitoring and fault detection for motor arrays through vibration, current, and temperature data was achieved via a wireless sensor network, in consideration of time delays [11]. A fault diagnosis system for autonomous vehicles based on neural networks using the Internet-of-things (IoT) was developed to collect various types of information [12]. Fail-safe motion planning and verification were conducted to generate vehicle trajectories that could allow vehicles to maintain desired maneuvers with enhanced safety [43, 44, 45]. A fail-safe priority-based intersection approach was developed to manage signal-less intersections autonomously [46]. A fault detection, isolation, and identification architecture for multi-faults in multi-sensor systems has been introduced in [47]. Algorithms designed to diagnose faults in functional parts of autonomous vehicles were developed for fail-safe systems [48-51]. The following studies have placed greater focus on designing and analyzing environmental sensor faults. For example, vehicle chassis sensor faults [51, 52] handled faults related to vehicle chassis sensors, whereas faults in environment sensors such as LiDARs and Radars were tackled in [48-51, 53, 54, 55, 56], [57-59], on the other hand, focused on actuator faults. All these studies have performed validations through simulations and vehicle tests. Various methodologies regarding fault diagnosis and fault-tolerant control were proposed [48, 49, 52, 56, 57, 58].

A thorough review of the studies mentioned above has shown that aspects of fail-safe systems such as fault detection and reconstruction have been studied through several methods, including observer-approaches, statistical methods, sensor monitoring, and artificial networks. However, most methods rely on the observer or estimator assumption that fault is bounded by parameter design.

This paper proposes two fault detection algorithms for longitudinal sensor fault detection. Other studies have shown the occasional use of the sliding mode observer method to detect faults. However, the conventional sliding mode observer requires an output error boundary to design the injection term parameters. Designing this observer still proves to be challenging due to the time-variant nature of the output error boundary. Therefore, in this paper, an adaptive sliding mode observer was developed to update the injection term parameters in accordance with output errors affected by acceleration faults. Methodologies with the acceleration fault can be reconstructed using the Lyapunov stability concept, also applicable in detecting acceleration sensor faults through a threshold approach. Moreover, a linear model-prediction algorithm has been designed for faults in environmental sensors as well.

In their current state, environmental sensors utilize clearance data and relative velocity to make predictions based on the discretized system model. Two predicted states can be computed from the minimum and maximum acceleration usage. The feasible boundaries are then configured from each predicted state. Together with past accumulated states, comparisons are made with the current state. The fault index can be calculated by counting values sitting outside the configured boundaries. Finally, an environmental sensor fault is determined based on a defined threshold after averaging the count. Three representative fault types are classified as follows: Power off, Holding Signal, Offset Signal. Of the type of faults mentioned so far, this paper will only consider and demonstrate the offset fault. This is because the detection of offset faults results in easy identification of the other types.

This paper also contains an emergency control portion essential for an autonomous vehicle to be in accordance with level 4 autonomous driving. This research utilizes an SMC method that takes adaptive convergence time into account.
Because the convergence time of the control error was determined mathematically, the corresponding control values were designed to change this predetermined convergence time. The method incorporating convergence time was applied to the test vehicle under the fail-safe control portion. Later sections describe the method’s proof of stability in greater detail. Shown below are some of the relevant literature surveys regarding SMCs and the usage of such a method.

Jun Hu et al. proposed a survey on SMC for networked control systems [60]. This survey was able to provide a greater understanding of the current application trends of the SMC. Jun Hu et al. also investigated the uncertain nonlinear systems of actuator faults [61]. In this research, the team designed various uncertain actuator faults and validated a newly proposed controller, the ISMC scheme, to ensure stability. Validations were carried out with the rocket fairing structure model. SMC for Networked control systems (NCSs) shows great promise in fail-safe research in that methodologies used here can also be effectively utilized in fail-safe systems as well. Moving on, the SMC was effectively utilized in reducing the effects of actuator attack failures [62]. The proposed approach was able to guarantee a type-2 fuzzy system and the input-to-state stabilization of sliding motion. Among the studies showing various SMC methodologies, their approach to time delay could be categorized into 4 types: constant time-delay, time-varying delay, distributed delay, and probabilistic interval time delay [60]. The constant time-delay type could be further divided into present-based SMCs and time-shift SMCs. A related study proposed a predictive SMC for a networked control system with time delay and packet dropout [63]. A chattering-free law was also designed and utilized in the predictive SMC. Within the field of time-varying delay, recent trends have leaned towards delay-fraction SMC research.

D. Ao et al. proposed a super-twisting sliding mode control algorithm (STA), based on the Lyapunov theory, to enhance the robustness and attenuation of the chattering issue [64]. S. Wang et al. suggested a new sliding mode control strategy based on an RBF (Radial Basis Function) neural network to solve tracking errors and substantial chattering phenomena [65]. A mathematical nonlinear longitudinal model was proposed, and their algorithm was validated through a simulation study under typical driving conditions. B. Peng et al. proposed an improved sliding mode control strategy for vehicle platoons, capable of forming a platoon in shorter time periods in simulated environments with better stability [66]. The proposed strategy was able to reach a stable and controllable area rapidly. P. Wang et al. proposed a robust automatic control strategy based on the back-stepping sliding mode control theory [67]. The study conducted a co-simulation with CarSim/Simulink under various scenarios. Y. Han et al. proposed a sliding mode control method based on the improved reaching law [68]. The method was developed to control the attitude of an automated guided vehicle, where an outer loop controller outputs the desired attitude. Validations were conducted through simulations.

So far, a thorough review of the numerous related studies has shown current trends and achievements made with the SMO and SMC. However, some issues are yet to be addressed. Firstly, methodologies proposed by most studies are developed under the assumption that faults are bounded by parameter design. Secondly, there is still a lack of research on the usage of the SMO and SMC in emergency controls. Additionally, a “PC (Upper controller) Shutdown” situation, where algorithms may only work with the last available values, is a genuine possibility that cannot be ignored. This paper aims to address all these issues with the algorithms proposed by our research team. The main contributions of this work can be summarized as follows:

1) A newly designed framework for fail-safe modules in Autonomous Vehicles has been proposed and implemented in an actual autonomous vehicle.
2) An MIT rule-based adaptive SMO methodology has been proposed for the detection of longitudinal faults in chassis and environment sensors (lidar, radar) and for the reconstruction of unknown faults in real driving scenarios.
3) The proposed adaptive SMC (In consideration of convergence time) has been successfully implemented and tested in fault scenarios involving actual vehicle experiments. Possible fault types of AVs have been defined through these experiments as well.

The remainder of this paper is divided into six sections and organized as follows. Section II provides an overview of the proposed fail-safe system and a brief introduction to the controller. Section III is further divided into 5 subsections that provide detailed explanations of the proposed observer. Section IV describes the vehicle actuator system model. Section V is further divided into 5 subsections that provide a description of the fail-safe algorithm and the evaluation results of the emergency controller. Finally, Section VI finalizes the paper with a conclusion and planned future works.

II. OVERVIEW OF A FAIL-SAFE SYSTEM IN AUTONOMOUS VEHICLES

This section introduces the fail-safe system applied to the autonomous vehicle used in this paper. While the system has only been applied and tested on autonomous vehicles used in university laboratories, its uses are not limited to select test vehicles. The methods proposed in this paper are still extensively applicable in other areas. The overall controller is composed of an upper controller and a lower controller: A commercial, industrial PC (IPC) as the upper controller, and a MicroAutobox (dSPACE) as a lower controller. It is to be noted that the lower controller is more robust than the upper controller. The test vehicle’s sensor configuration is depicted in Fig. 1. In this research, the focus is primarily placed on developing a fail-safe module for an automated driving
This work also utilizes a longitudinal control algorithm encompassing the fault detection algorithms. The following three consecutive parts make up the control algorithm. First, a decision part determines the vehicle’s current state through the CAN bus signal. Next, confirmation regarding a fault where the CAN communication gets stuck occurs. Indexes based on the best reference target model are determined. Finally, the lower-level controller determines the desired control input: The steering angle is calculated through a dead reckoning method, whereas the longitudinal acceleration control input is determined based on a sliding mode control method that tracks the reference target model. Table 1 describes the overall algorithm used in this paper for the fail-safe system in autonomous vehicles. In Table 1, the fault detection method corresponding to each hardware module can be found. An appropriate maneuver strategy for each module is also proposed.

In Table 1, a single asterisk indicates that the method was applied to an actual autonomous test vehicle. A double asterisk indicates that the method was only verified through simulations.

### III. ADAPTIVE SLIDING MODE OBSERVER AND LINEAR PREDICTION BASED SENSOR FAULT DETECTION

The following driving condition was proposed to test for various fault scenarios. The test vehicle was to follow another leading vehicle located in its longitudinal direction. The environmental sensors were used to measure and obtain vehicle clearance and relative velocity information. The acceleration of the leading vehicle was measured through a wireless communication sensor. The observer’s kinematic model defined for the longitudinal direction is given in the following form:

\[
\dot{x} = Ax + Bu \tag{1a}
\]

\[
y = Cx \tag{1b}
\]
\[ x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \] refers to the clearance and relative velocity of the leading vehicle. \( u \) indicates the relative acceleration, the difference in longitudinal accelerations of the leading and the test vehicles, defined by \( a_p - a_v \). The adaptive sliding mode observer and linear prediction algorithm for fault detection in the longitudinal direction are designed utilizing Equations (1a) - (1c).

A. Adaptive sliding mode observer for acceleration sensor fault reconstruction and detection

The adaptive sliding mode observer is designed to reconstruct the acceleration fault: see model schematics in Fig. 3. The description of Fig. 3 is as follows.

![Model schematics of the acceleration fault reconstruction and detection](image)

The environment, communication, and acceleration sensors transmit various sensor values to the observer. The SMO then calculates an estimated error, \( e_j \), and sends it to the forgetting factor module. The forgetting factor module then calculates the relationship coefficients \( C_r, C_p \). Upon receiving the coefficients \( C_r, C_p \), the Adaptive MIT rule module updates parameter \( \rho \) and transmits it back to the SMO. Throughout the sequence, the fault detection threshold block continuously receives the threshold results and reconstructed fault from SMO, and exports a fault detection signal as an output. Since this process was designed to work in real-time, the period of each step was set to 10ms (0.01Hz). The overall algorithm was then implemented and tested on an actual vehicle.

The kinematic model inclusive of the fault term \( f_a \) (if it exists) is represented as follows:

\[
\dot{x} = Ax + Bu + Ff_a
\] (2a)

\[
F = \begin{bmatrix} 0 & -1 \end{bmatrix}^T
\] (2b)

Here, we introduce a coordinate transformation \( x \rightarrow Tx \), while taking the system model into consideration. \( T \) is defined as

\[
T = \begin{bmatrix} Null(C) & C \end{bmatrix}^T
\] (2c)

The following assumptions are made when designing the observer:

1. When observer error is zero, acceleration faults can exist after convergence.
2. When convergence time is extremely small, the observer performs appropriately.

The assumptions are further discussed in subsection B. We denote the observer estimated equation as follows:

\[
\dot{\hat{x}} = A_1 \hat{x} + B_1 u + G_n v
\] (2d)

\[
y = C_1 \hat{x}
\]

\[
A_1 = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix},
B_1 = \begin{bmatrix} B_{11} \\ B_{12} \end{bmatrix},
C_1 = \begin{bmatrix} 0 & I \end{bmatrix}
\] (2e)

\[
G_n = \begin{bmatrix} L & -I \end{bmatrix},
v = p \text{sign}(e)
\] (2f)

Here, \( \hat{x} \) represents the observer’s estimated state. \( G_n \) represents the distribution matrix, inclusive of the design parameter \( L \) in equation (2f). The transformed system matrix is \( A_1 = TAT^{-1} \), \( B_1 = TB \), and \( C_1 = CT^{-1} \). \( \rho \) represents a coefficient bounded through a stability analysis, where parameters were determined through a vehicle test, shown in Table III. Subsections B and C provide further explanations of the coefficient \( \rho \). Error dynamics can be derived using system and observer models by defining error as \( e = \hat{x} - x \). The error dynamics are given as follows:

\[
\dot{\hat{e}}_1 = A_{11} e_1 + A_{12} e_2 + Lv
\] (3a)

\[
\dot{\hat{e}}_2 = A_{21} e_1 + A_{22} e_2 - v
\] (3b)

Here, \( v \) represents the injection term. A proper design of the injection term \( v \) results in \( v \) being large. In this case, the errors \( \hat{e}_1 \) and \( \hat{e}_2 \) in equation (3a) converge to zero within finite time. This is further described in the next section. Equation (3a) can be rewritten as

\[
v = A_{21} \hat{e}_1
\] (3c)

Substituting equation (3c) to equation (2f), a dynamic of error \( e \) can be calculated as follows:

\[
\dot{e}_1 = (A_{11} + L A_{12}) e_1
\] (3d)

Error \( \hat{e}_1 \) and \( e \) can also converge to zero when the eigenvalue of \( A_{11} + L A_{12} \) is negative definite. If a fault occurs after the observer error \( e \) converges to zero, only the fault term \( f_a \) remains in the error dynamic model. It can then be described as follows:

\[
0 = F_{11} f_a +Lv
\] (4a)

\[
0 = F_{12} f_a -v
\] (4b)

Here, \( F_{11} \) and \( F_{12} \) are variables of transformed fault matrix calculated using \( F \) in equation (2b). The designed parameter \( L \) and reconstructed acceleration fault \( f_a \) are described as follows:

\[
L = -F_{11} (F_{12})^{-1}
\] (4c)

\[
f_a = - (F_{11})^{-1}Lv = -(F_{12})^{-1}v
\] (4d)
$F_{1,1}, F_{1,2}, \ldots$ are variables of the transformed fault matrices. $v_e$ denotes the equivalent injection term. Here, equation (4c) should simultaneously satisfy the condition derived from equation (3d). The reconstructed fault is used to detect acceleration fault based on a threshold approach, referred to as the fault detection signal in equation (4c). The fault detection index $I_a$ is described as follows:

$$I_a = \begin{cases} 
1(\text{Fault}) & \text{if } f_{a,rec} \geq T_h_a \\
0(\text{No Fault}) & \text{if } f_{a,rec} < T_h_a 
\end{cases}$$ (4e)

where $f_{a,rec}$ and $T_h$ represent the reconstructed fault and threshold index, respectively. $T_h$ is introduced in Table III of the next section. The following subsection introduces a design parameter for the sliding mode observer to satisfy the stability condition of output error $e_y$.

### B. Stability analysis of sliding mode observer

The convergence stability of the output error can be ensured through the use of the injection term $v$, defined in expression (3c). The Lyapunov direct method-based injection term, magnitude $\rho$, is designed by the finite-time convergence condition. The Lyapunov function and conditions are defined to ensure stability in the design of the injection terms. It is given as follows: [21] [22]

$$V = \frac{1}{2} e_y^2$$ (5a)

$$\dot{V} < 0, \ orall e_y \neq 0$$ (5b)

$$\lim_{t \to \infty} V = \infty$$ (5c)

To secure the stability within finite time, the following conditions are considered:

$$\dot{V} \leq -\alpha V^{1/2}$$ (5d)

These are calculated based on a variable separation method and $\alpha$ can be derived with the assumption that the cost function reaches zero within a finite time through the following relation:

$$t_f \leq -\frac{2V^{1/2}(0)}{\alpha}$$ (5e)

From equation (5d), $\alpha$ can now be defined. Based on equations (5a) and (5e), the Lyapunov term can be given as:

$$\dot{V} \leq -e_y \dot{e}_y$$ (5f)

Equation (5c) can be rewritten using equation (3b) as follows:

$$\dot{V} \leq -e_y (A_{21} e_i + A_{22} e_y) - \rho |e_y|$$ (5g)

Where $\rho$ is predefined from equation (5d). The parameter $\rho$ is designed from equations (5f) - (5g). The boundary conditions are described as follows:

$$|A_{21} e_i + A_{22} e_y| \leq L_b$$ (5h)

where $\rho$ is designed as follows:

$$\rho = L_b + \frac{\alpha}{\sqrt{2}}$$ (5i)

Next, boundary condition, $L_b$, needs to be determined in consideration of the fault magnitude, obtained from the experiment. Through parameter $\rho$, the output error of the sliding mode observer could converge to zero. $L_b$ represents the boundary value that includes acceleration fault information from equation (5h). Note that $L_b$ is a time-varying value. An MIT rule-based adaptive algorithm to design the magnitude of the injection term parameter is proposed in a later subsection. The adaptation method is utilized to account for unexpected values of acceleration error boundary.

### C. Adaptation algorithm based on the MIT rule

This subsection introduces the utilization of an MIT adaptive rule that does not require system model parameters. Here, estimated coefficients and MIT rules were used to update feedback gain. The parameter $\rho$ is designed using the Lyapunov function under the assumption of boundary condition $L_b$. However, considering the nature of the acceleration fault, we cannot forgo the possibility that the error boundary itself could be the unexpected value. This indicates that the conventional sliding mode observer has a fixed parameter and cannot be adjusted for fault reconstruction. The adaption rule methodology is proposed to overcome this limitation of the algorithm.

The equations for the cost function $J(\hat{\theta}(k), k)$ of the recursive least square term, optimal gain $L_g$, covariance $P$, and forgetting factor $\lambda$ are given as follows:

$$\min J(\hat{\theta}(k), k) = \frac{1}{2} \sum_{i=1}^{N} \left[ (y(i) - \phi(i) \hat{\theta}(k)) \right]^2$$ (6a)

$$\hat{\theta}(k) = \hat{\theta}(k - 1) + L_g (k) (y(k) - \phi(\hat{\theta}(k - 1)))$$ (6b)

$$L_g (k) = P(k - 1) \phi(k) \lambda + \phi^T(k) P(k - 1) \phi(k)$$ (6c)

$$P(k) = (I - L_g (k) \phi^T(k)) P(k - 1) / \lambda$$ (6d)

Here, $\hat{\theta}(k), k, y(i), \phi(i), \phi^T(i)$, and $\lambda$ are estimated terms at the k-th step, time step number, output for RLS, regressor, transpose of regressor, and forgetting factor, respectively. The MIT rule-based cost function $J_g$ and $J_p$, designed for the adaptation algorithm, are given as follows: [23] [24]
\[ J_e = \frac{1}{2} \varepsilon_e^2, \quad J_p = \frac{1}{2} \varepsilon_p^2 \]  
\[ \frac{d \rho}{dt} = -\gamma_e \varepsilon_p \hat{\varepsilon}_p - (1-w)\gamma_p \varepsilon_p \hat{\varepsilon}_p \]  
\[ \rho = \int_0^t \left( -w \gamma_e \varepsilon_p \hat{\varepsilon}_p - (1-w)\gamma_p \varepsilon_p \hat{\varepsilon}_p \right) dt \]

Where \( \gamma_e \) and \( \gamma_p \) are adaptation gains designed to increase and decrease the \( \rho \) term. Here, \( \hat{\varepsilon}_e \) and \( \hat{\varepsilon}_p \) are coefficients estimated from \( J_e \) and \( J_p \). \( \varepsilon_e \) refers to the error of \( \rho \) that can be simply described as \( \hat{\rho} - \rho \). The equations (7a) and (7b) can be integrated into one equation using a weighting factor \( w \). The integrated equation determines the magnitude of the output error and is given as follows:

\[ \frac{d \rho}{dt} = -w \gamma_e \varepsilon_p \hat{\varepsilon}_p - (1-w)\gamma_p \varepsilon_p \hat{\varepsilon}_p \]  
\[ \rho = \int_0^t \left( -w \gamma_e \varepsilon_p \hat{\varepsilon}_p - (1-w)\gamma_p \varepsilon_p \hat{\varepsilon}_p \right) dt \]  
\[ w = \begin{cases} 1, & \varepsilon \geq \varepsilon_c \\ 0, & \varepsilon < \varepsilon_c \end{cases} \]  
\[ x(k) = A_x x(k) + B_x u(k) \]  
\[ A_x = \begin{bmatrix} 0 & \Delta T \\ 0 & 0 \end{bmatrix}, \quad B_x = \begin{bmatrix} 0 \\ \Delta T \end{bmatrix} \]

Here, \( \varepsilon \) represents a small positive value and a criterion for determining whether an increase or decrease in parameter \( \rho \) is necessary, depending on the status of the output error \( \varepsilon \). \( w \) represents a switching factor that changes (increase or decrease) the parameter \( \rho \), which is updated through equation (8b). From equation (8b), it is evident that \( \rho \) is dependent on output error \( \varepsilon_p \). Therefore, the primary purpose of the design adaptation algorithm is satisfied at this point. The following subsection describes the methodology for fault detection in environmental sensors.

**D. Linear model prediction-based environment sensor fault detection**

This subsection introduces the linear model prediction algorithm for environmental sensor fault detection. A predictive and accumulative method is used to derive the feasible boundary of the current measured state. The same longitudinal driving kinematic model in equations (1a) - (1c) has been used. The discretized kinematic model is used for prediction and is given as follows:

\[ x(k+1) = A_x x(k) + B_x u(k) \]

Where the value of \( N \) is 20 (indicating a 2 second interval). Vehicle tests have determined the 2 second time frame to be an appropriate estimation time. In order to consider the predicted input term \( \sum_{i=0}^{N-1} u(k+N-i) \) of equation (9c), an acceleration value (maximum or minimum) is applied to a differential acceleration value, defined as \( \text{Jerk} \). Equation (9c) can then be divided into two separate equations for prediction, utilizing the maximum or minimum acceleration information instead of the predicted input term. The equations are given as follows:

\[ x_u(k+N) = A_y^x x(k) + \sum_{i=0}^{N-1} A_y^u B_u u(k) \]  
\[ + \sum_{i=0}^{N-2} (N-i-1)A_y^u B_u \Delta u_i \]  
\[ x_l(k+N) = A_y^x x(k) + \sum_{i=0}^{N-1} A_y^u B_u u(k) \]  
\[ + \sum_{i=0}^{N-2} (N-i-1)A_y^u B_u \Delta u_i \]

Here, \( x_u(k+N) \) and \( x_l(k+N) \) represent the predicted upper and lower limit values. The state is predicted from the first to the \( N \)-th step through the equations above. The predictions are then accumulated for precision. Terms \( \Delta u \) and \( \Delta u_i \) represent the upper and the lower bound of input difference respectively. Fig. 4, visualizes the linear prediction algorithm’s concept; each step’s lower subscript indicates the past step from which the prediction was made. The present time shows an accumulation of predicted states (red point in Fig. 4). Predicting the states of \( x_u \) and \( x_l \) allows the derivation of the \( N-1 \) feasible boundary between \( x_u \) and \( x_l \). A deviation of the current state from the feasible boundary derived by the accumulation of predicted states
method is given below. Fig. 4 shows that the current index determines a fault by comparing the value predicted over 20 steps with the current measured value. The clearance fault index $F_{1,i}$, relative velocity fault index $F_{2,i}$, and fault sum average index $F_{i}$ are given as follows:

$$F_{1,i} = \begin{cases} 1, & x_{i,k} < x_{i,k-1} \text{ or } x_{i,k} > x_{i,k-1}, \\ 0, & \end{cases}$$

(10a)

$$F_{2,i} = \begin{cases} 1, & x_{2,k} < x_{2,k-1} \text{ or } x_{2,k} > x_{2,k-1}, \\ 0, & \end{cases}$$

(10b)

$$F_{i} = \left( \sum_{i=1}^{N} (F_{1,i} + F_{2,i}) \right) / N$$

(10c)

The final fault index is derived using the threshold approach as follows:

$$F_{s} = \begin{cases} 1 & (F_{s} \geq Th_{s}) \\ 0 & (F_{s} < Th_{s}) \end{cases}$$

(10d)

$F_{s}$ and $Th_{s}$ represent the final fault index and threshold of the fault sum average index $F_{s}$. The algorithms presented above have been proposed as a methodology for fault detection of acceleration and environment sensors in the longitudinal direction.

E. Vehicle test results of fault detection algorithm

This subsection introduces vehicle test scenarios, the test environment, and test results. Table II summarizes the proposed fault detection algorithm for the environment sensor and the vehicle chassis sensor. The designed fault scenario is distinguished as follows: Three scenarios were designed as shown in Table II. A square wave fault was used with different magnitudes for acceleration sensors. Faults with the environmental sensors are designed as fault for $x_{1}$, fault for $x_{2}$, and fault for $x_{1}$ and $x_{2}$ simultaneously. In the last case in which faults of the environment sensor and acceleration sensors occur simultaneously, the sliding mode observer could not be converged using the wrong state information. To get reasonable results, the vehicle test was carried out using the results from the environment sensors (lidar, radar), the RT range, and the RT differential GPS equipment.

A vehicle test situation is designed as shown Fig. 5 to utilize equation (1c) in the kinematic model. The parameters for adaptive sliding mode observer and linear prediction algorithm are described in Table III.

Two small and large fault signals are applied to acceleration information to evaluate the performance of the adaptation algorithm for the sliding mode observer. The results from applying a large fault are shown in Fig. 6 for comparison. The results from applying a small fault are shown in reference papers [15], [41]. Fig. 6 (a) shows comparison of results of applied and reconstructed fault between the timestamps 25 sec ~ 35 sec. From the applied fault plot, it can be observed that output error results are large. A switch in the weighting factor to 1 from 0 can be seen as well. The parameter $\rho$ is updated using the estimated coefficient from output error. Even though output error represents trembling on the region with applied fault in Fig. 6 (b), it maintains stability using the adaptive parameter in the injection term. The switched weighting factors are shown in Fig. 6 (c) and Fig. 6 (d). The estimated coefficients are shown in Fig. 6 (e) and Fig. 6 (f) used for the adaptation algorithm. Based on
reconstructed fault, the acceleration fault has been detected as shown in Fig. 6 (g). Fig. 6 shows the scenario applied to a relatively large acceleration fault. The results are similar to the results of the small fault scenario. However, the results of the adaptive parameter in Fig. 6 (c) are more significant than small fault scenarios. This indicates that the method has a large output error from a large acceleration fault, shown in Fig. 6 (b). Estimated coefficients are also shown in the large estimation changing rate (compare Fig. 6 (e) and (f)). Therefore, the applied fault can be reconstructed using a switched weighting factor based on adaptive parameters. Despite the large acceleration fault, the performance of fault detection is still improved.

FIGURE 6. The results of adaptive sliding mode observer: large magnitude of fault.

FIGURE 7. The results of clearance fault detection.
reconstruction and detection was checked and is secure (Fig. 6 (a) and (g)). A large value is seen because it is the initial convergence of observer output error. A high adaptive gain can be defined to ensure initial convergence performance, but the results are over a relatively large adjustment of parameters and can lead to unreasonable results. Fig. 7 shows the fault detection results for environment sensors based on the linear prediction method. Three test scenarios were considered in Table II for reasonable performance evaluation: clearance fault, relative velocity fault, and simultaneous faults in both sensors. The square wave fault was induced at around 0 seconds, an increase in the fault-sum-average index becomes larger at around the 20 second mark. However, around 0 seconds, an increase in the fault-sum-average index can be observed as well. Such faults at the initial interval are ignored in the vehicle test.

IV. Vehicle actuator system model

The test vehicle performs longitudinal control by applying the desired longitudinal acceleration commands to the vehicle gateway. The gateway connects the lower controller and the vehicle system through a CAN communication platform to achieve vehicle behavior that follows the desired command. As depicted in Fig. 8, the vehicle actuator system can be defined as a dynamic system consisting of a gateway and a vehicle system. The dotted line in Fig. 8 represents a schematic diagram of an actual vehicle actuator system. For the actuator system, the desired acceleration command is the input, and the actual acceleration of the vehicle is the output. Consider the case of an emergency braking scenario. The emergency control corresponds to a command is continuously applied to the gateway. e.g., negative acceleration. The vehicle then decelerates while tracking the braking input command.

From a vehicle control perspective, immediate responses from the actuator system are critical in ensuring that the desired motion planning gives an acceptable performance. An ideal actuator system outputs a vehicle acceleration response instantly when an acceleration input command is given. However, response delays exist in realistic vehicle actuator systems. This is due to the vehicle’s inherent characteristics, such as body weight and powertrain performance. These actuator delays lead to an accumulation of error within the control objective, resulting in an overshoot response to compensate for the error. Therefore, actuator response characteristics need to be accounted for to reduce the disparity between the desired and actual performance.

The system’s response characteristics can be captured in a simple dynamic model. The First Order Plus Dead Time (FOPDT) model can aptly define the vehicle actuator system as follows:

\[ \dot{a}_s(t) = -\frac{1}{\tau} a_s(t) + \frac{1}{\tau} a_{s,des}(t-t_d) \]  

(11)

Here, \( a_s \) represents the actual longitudinal acceleration, \( a_{s,des} \) represents the desired longitudinal acceleration command, \( \tau \) is a time constant, and \( t_d \) represents the dead time. By fitting the time series data of the control command and actual sensor measurement against the FOPDT model, model parameters could be obtained. The reference data was obtained from actual vehicle tests in order to evaluate the step response of the vehicle actuator system with different step command scenarios. The model parameters were obtained using the MATLAB system identification tool, which estimates the parameters of a Laplace transfer function from a given time series data of input and output. The Laplace transfer function model corresponding to (14) in the frequency domain is given as follows:

\[ G(s) = \frac{\frac{a_s(s)}{a_{s,des}(s)}}{\frac{1}{1+\tau s}} = \frac{e^{-t_d s}}{1+\tau s} \]  

(12)

\( a_s(s) \) represents the Laplace transform of \( a_s(t) \), and \( a_{s,des}(s) \) represents the Laplace transform of \( a_{s,des}(t) \). Each braking and acceleration command results in responses with differing characteristics. Hence the representative model parameters need to be estimated individually for each command scenario. From a control perspective, the scenario of interest is of the vehicle braking. Hence, system identification was mainly conducted for a braking scenario.

Fig. 9 shows a comparison of the step input responses between the actual vehicle and a simulation based on the FOPDT model with the estimated parameters. The comparison shows that the FOPDT model has a satisfactory characterization of the vehicle response with the estimated parameters. When a step command input was given, the Root Mean Squared Error (RMSE) value between the actual and the simulated acceleration results was less than 0.12 m/s². Fig.9 shows that the FOPDT model with the appropriate parameters can accurately describe the actuator response’s overall characteristics. The corresponding parameters for the acceleration and braking scenario were found to be \((\tau, t_d) = (0.5s, 0.8s)\) and \((\tau, t_d) = (0.17s, 0.25s)\).
respectively. By utilizing the FOPDT model with the estimated parameters, the characteristics of the actuator response could be accounted for within vehicle system model design of the emergency control system.

V. FAIL-SAFE EMERGENCY CONTROL – ADAPTIVE SLIDING MODE CONTROL BASED (LONGITUDINAL, MODULE IN VEHICLE)

A. Overall hardware structure

The hardware concept schematic of the autonomous vehicle controllers (PC, Autobox), inclusive of fail-safe algorithms, is depicted in Fig. 10. The fail-safe module and the perception, decision and control algorithms in the upper controller, were configured under typical environmental circumstances, in consideration of the autonomous vehicle’s hardware structure. If the fault detection module detects a fault and no driver intervention is given, the last available information is used to predict and control the system. The drive-able path information received from the upper controller is utilized in two ways: in the lateral direction and the longitudinal direction. In the lateral direction, the dead reckoning algorithm utilizes the distance information to calculate the appropriate steering angle. Lateral control only utilizes the vehicle chassis information. This algorithm uses the last information (desired path) available from the upper controller to follow the desired path using the DR method. In the longitudinal direction, reference target building and sliding mode control-based deceleration algorithms are executed. Longitudinal control works by calculating the reference model based on the received drive-able path information. Information is delivered to the lower controller in real-time, and the control constantly operates in the vehicle. The entire module is composed of a fault detection part that classifies the fault and a control model that controls deceleration with limited information.

B. Reference deceleration model rebuilding and filtering

The typical reference deceleration model is made from general driver deceleration data [25]. The typical model is described in Fig. 10 under the fail-safe control portion. The model considers driver safety and ride comfort. A first-integrated velocity model and a second-integrated station model are used to construct an algorithm for stopping at safe distances. Images and formulas for the longitudinal acceleration, the longitudinal velocity, and the longitudinal distance model are illustrated in Fig. 11. Here, $V_0$ represents initial velocity, $a_m$ represents maximum used deceleration, $\theta$ represents time ratio, $t_d$ represents deceleration time, $m$ represents a model variable parameter, and $r$ represents a model parameter. Detailed information and model parameters regarding the reference model have been

$$a(t) = ra_m(1 - \theta^n)^2$$

$$V(t) = V_0 + 3.6ra_m t_\theta^2 \left[ \frac{1}{2} \frac{2\theta^n}{(m+2)} + \frac{\theta^{2m}}{(2m+2)} \right]$$

$$S(t) = V_f t / 3.6 + ra_m t_\theta^2 \theta^m \left[ \frac{1}{6} - \frac{20^n}{(m+2)(m+3)} + \frac{\theta^{2m}}{(2m+2)(2m+3)} \right]$$

$V_f$ = initial speed $(m/h)$

$a_m$ = maximum deceleration rate $(m/s^2)$

$\theta$ = time ratio $(\theta = t/t_d)$

$t_d$ = deceleration time

$m$ = model parameter

$r$ = model parameter

FIGURE 9. Acceleration results of vehicle actuator system (Step response)

FIGURE 10. Failsafe hardware concept diagram

FIGURE 11. Deceleration reference model. [25]
introduced in various studies [25]. The authors of [25] utilized normal driver deceleration data to build a deceleration model. Driver data-based models such as the one built in [25] may reflect the subject driver’s driving habits. Consequently, it has been shown in a simulation study that some reference models show characteristics that result in an uncomfortable feeling for the driver. The applied reference model was a function of time. This implies that the reference model is calculated in real-time in the event of a fault that requires emergency deceleration control.

This paper has adopted a new method to rebuild the reference model offline, according to the vehicle’s velocity. A reference model for each velocity was first created. Next, the Carsim simulator was used to evaluate the reference model's three main index types. In this paper, the proposed indexes regarding safety and comfort are described in two parts. Under safety, the indexes utilized are the time to collision inverse (TTC inv) index and longitudinal warning index. Under comfort, the indexes utilized are pitch, pitch rate, and vertical acceleration. More detail regarding the safety indexes can be found in [26], [27]. The index proposed here is shown in Fig. 12. For Comfort indexes, this study used the passenger’s ride comfort data and indexes from a related paper [28]. In [28], jerk value and vertical acceleration are proposed as riding comfort indexes. Other indexes utilized in [28] are shown in Fig. 13. Table IV summarizes the indexes regarding safety and comfort. For this paper, a new index was developed through a combination of the indexes investigated.

C. New index-based reference model rebuilding and filtering

This section introduces a methodology for filtering out improper reference models using the newly proposed indexes. Fig. 11 shows an example of an improper reference model [25]. Reference target models were first generated according to the velocity at the moment of vehicle failure. Filtering of these models using safety and comfort indexes was carried out to generate a final filtered reference model.

Through a Carsim simulation test, the pitch, vertical acceleration, and jerk value could be derived. In the Carsim simulation, each test vehicle was driven with the velocity and station of each reference model. A total of 78 derived cases were carried out. The Safety and Comfort indexes used and referred to are presented in Table IV, and the derived results are shown in Figs. 14 to 16. The guidelines for filtering are as follows: 1. overshoot of indexes occurs when its value is not within a reasonable range and the passenger is deemed to be uncomfortable [28]. The filtered referenced model used for the vehicle tests is shown in Fig. 17. The model was

FIGURE 12. AEB & longitudinal safety index [26, 27].

FIGURE 13. Riding comfort evaluation in paper [28].

FIGURE 14. Generated reference model filtering: pitch rate, warning index, and TTC inverse.

FIGURE 15. Generated reference model filtering - pitch, warning index, TTC inverse.
constructed for vehicle tests of velocities between 30-50 kph, shown in Fig. 17

D. Sliding mode control - adaptive converge time gain and stability

This subsection proposes a control methodology applicable for the fail-safe control portion in an autonomous vehicle. Reasons for the usage of the sliding mode control method and the adaptive convergence time method are detailed in this section. To ensure a strict level of safety with the designed conditions, knowledge regarding response time to a failure is vital. Therefore, the convergence time of the control error was determined mathematically, and the corresponding control values were designed to be changed based on the predetermined convergence time. Thus, this particular method was applied to the test vehicle within the fail-safe control portion. This subsection demonstrates the stability of the longitudinal emergency braking system. The proof of stability is also shown in this subsection. In the fail-safe control part, the longitudinal model is a nonlinear system without any disturbance and uncertainties. The model can be described as follows:

\[ \dot{x}(t) = f(x,t) + g(x,t) \cdot u(t) \]  \hspace{1cm} (13a)

From section IV (vehicle actuator module system identification), it was determined that the actuator system followed a first-order delay system [37], [38], [39]. We define the longitudinal model and error dynamics as follows:

\[ \dot{x}_{\text{long}}(t) = Ax(t) + Bu_{\text{long}}(t) \]  \hspace{1cm} (13b)

\[ A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1/\tau_{ax} \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 1/\tau_{ax} \end{bmatrix} \]  \hspace{1cm} (13c)

\[ e_1(t) = x_{1,\text{ref}}(t) - x_1(t) = s_{x,\text{ref}}(t) - s_1(t) \]

\[ e_2(t) = x_{2,\text{ref}}(t) - x_2(t) = v_{x,\text{ref}}(t) - v_2(t) \]

\[ e_3(t) = x_{3,\text{ref}}(t) - x_3(t) = a_{x,\text{ref}}(t) - a_3(t) \]  \hspace{1cm} (14)

Here, \( u_{\text{long}} \), \( \tau_{ax} \), and \( k \) represent the control input, time constant, and estimated \( s_{x,\text{ref}} \), \( v_{x,\text{ref}} \), \( a_{x,\text{ref}} \). The estimated model parameters were \( (\tau, t_d) = (0.5s, 0.8s) \) and \( (\tau, t_d) = (0.17s, 0.25s) \) in accelerating and braking scenarios, respectively. Considering the system (13b, 13c) and reference model, error states can be defined as shown in (14). The time derivative of the error dynamics are derived as follows:

\[ \dot{e}_1(t) = \dot{x}_1(t) - \dot{x}_{1,\text{ref}}(t) = \ddot{x}_1(t) + \dot{x}_{1,\text{ref}}(t) - x_{2,\text{ref}}(t) \]

\[ \dot{e}_2(t) = \dot{x}_2(t) - x_{2,\text{ref}}(t) = \ddot{x}_2(t) + \dot{x}_{2,\text{ref}}(t) - x_{3,\text{ref}}(t) \]

\[ \dot{e}_3(t) = -\frac{e_3(t)}{\tau_{ax}} - \frac{k}{\tau_{ax}} u_{\text{long}} + \ddot{x}_{3,\text{ref}}(t) - \frac{x_{3,\text{ref}}(t)}{\tau_{ax}} \]  \hspace{1cm} (15)

The sliding surface term and the first time-derivative term were defined as follows:

\[ s(t) = e_1(t) - \lambda e_3(t) \]

\[ \dot{s}(t) = \dot{e}_1(t) - \lambda \dot{e}_3(t) \]  \hspace{1cm} (16)

The control input term \( u_{\text{long}} \) is related to \( e_1 \), and the sliding surface term contains \( e_3 \). The first-time derivative of the sliding surface is defined as follows:

\[ \dot{s}(t) = \dot{e}_1(t) + \lambda \dot{e}_3(t) \]

\[ = \frac{e_1(t) + \dot{x}_{1,\text{ref}}(t) - \dot{x}_{1,\text{ref}}(t)}{\lambda e_3(t)} \]

\[ + \lambda \cdot \begin{bmatrix} -\frac{1}{\tau_{ax}} \cdot e_1(t) - \frac{k}{\tau_{ax}} \cdot u_{\text{long}}(t) \\ + \frac{1}{\tau_{ax}} \cdot x_{3,\text{ref}}(t) \end{bmatrix} \]

We define the Lyapunov function candidate as follows:

\[ V(t) = \frac{1}{2} s(t)^2 \]  \hspace{1cm} (18)

We define first-derivative term \( \dot{V}(t) \) as follows. It is to be noted that this term is always negative.
\[ \dot{V}(t) = s(t) \ddot{s}(t) = s(t) \left[ \dot{\lambda}(t) + \lambda \dot{\lambda}(t) \right] \]

\[ \dot{V}(t) = \left[ s(t) \right] \cdot \left[ R(t) + \frac{k \lambda}{\tau_v} u_{long}(t) \right] \]

If the following condition is satisfied,

\[ e_s(t) + \hat{x}_{1,ref}(t) - \hat{x}_{2,ref}(t) - \frac{\lambda}{\tau_v} \leq R[x(t)] \]

\[ R[x(t)] \text{ is bounded by } R[x(t)]. \text{ The } \dot{V}(t) \text{ equation then satisfies the following inequality equation.} \]

\[ \dot{V}(t) \leq [s(t) \cdot R[x(t)] - s(t)] \cdot \left[ -\frac{k \lambda}{\tau_v} u_{long}(t) \right] \]

\[ = [s(t)] \cdot R[\lambda] \cdot R[x(t)] + Q[\lambda] \cdot s(t) \cdot u_{long}(t) \]

From equation (19) and (20), one can obtain a longitudinal control input and an equivalent control input as follows:

\[ u_{long}(t) = -\rho_u sgn(s(t)) \]

\[ u_{eq}(t) = -\rho_u sgn(s(t)) - K \cdot sgn(s(t)) \]

\[ \rho_u = \left[ \frac{R[x(t)]}{Q[\lambda]} \right] + \alpha, \quad K > 0, \quad \alpha > 0 \]

Here \( u_{long}(t) \), \( u_{eq}(t) \) and \( \rho_u \) represent the control input, the equivalent control input, and the adaptation parameter for control input magnitude, respectively. From equation (21), the equality equation of \( V(t) \) can be defined as follows:

\[ \dot{V}(t) \leq [s(t)] \cdot R[x(t)] - s(t) \cdot R[\lambda] \cdot u_{eq}(t) \]

\[ = [s(t)] \cdot R[x(t)] - s(t) \cdot R[\lambda] \cdot sgn(s(t)) \]

\[ = -Q[\lambda] \cdot K \cdot s(t) - \alpha \cdot K \cdot s(t) \]

\[ = -Q[\lambda] \cdot K \cdot \sqrt{2} \]

Where \( Q[\lambda] \), \( K \), and \( \alpha \) are all positive values. The design \( W(t) \) is as follows:

\[ W(t) = \sqrt{2} \cdot [s(e(t))] \]

The value of \( W(t) \) is defined so that the Lyapunov function candidate converges to zero within finite time.

Using the Comparison Lemma found in [13] and [21], the following equation can be derived.

\[ \dot{V}(t) < -\alpha V^{1/2}(t), \alpha > 0 \]

Variables separating and integrating time interval \( \tau_{int} \) from 0 to \( t_{converge} \) is given as follows:

\[ t_{converge} \leq \frac{2V^{1/2}(t)}{\alpha} = t_{bound} \]

From equation (27a), an inverse relation between \( t_{converge} \) and \( \alpha \) can be observed. Therefore, from equation (27a), as \( \alpha \) becomes larger, \( V(t) \) converges to zero faster.

\[ u_{eq}(t) = \left[ \frac{R[x(t)]}{Q[\lambda]} \right] \cdot sgn(s(t)) - K \cdot sgn(s(t)) \]

However, equation (28a) shows that an increased magnitude of \( \alpha \) could result in a more substantial chattering phenomenon of the control. The eta-reachability condition in equation (28b) can be easily defined from the initial condition of error and occurs instantaneously. Thus, equation (28c) can be rewritten as follows:

\[ R_{\lambda}[x(t)] = \left[ e_s(t) + \dot{x}_{1,ref}(t) - \dot{x}_{2,ref}(t) - \frac{\lambda}{\tau_v} \right] \]

\[ \leq R[x(t)] \]

If an upper decision controller determines a need for emergency deceleration, then the exact value of \( R_{\lambda}[x(t)] \) can be determined at the moment of the error’s occurrence. Equations (27a, 27b) show that the magnitude of \( \alpha \) is adaptively defined, and it immediately has a proportional effect on the magnitude of \( \rho_u \), evident from equation (28a). However, there could be a case in which the inequality condition becomes an equality, as shown in equation (29).

\[ t_{converge} = \frac{2V^{1/2}(t)}{\alpha} = t_{bound} \]

From the eta-reachability condition,
\[
\rho_m \geq R[\chi(t)] + \frac{\alpha}{\sqrt{2}} + \eta
\]  
(30)

\(\rho_m\) is large enough such that the condition \(\eta \in \mathbb{R}\) is satisfied. Thus, it can be inferred that \(s(t)\) will converge to zero within finite time. The mathematical proof for the above convergence in finite time allows us to derive the main contribution of this algorithm. This is also the main idea and unique concept behind the algorithm. The bounded time can be defined by various vehicle experiments. Due to the mathematical derivation of \(t_{\text{bound}}\) values, the logic behind the control guarantees that the time error is at always less than \(t_{\text{bound}}\) when standard control is in operation.

**E. Vehicle validation**

The proposed algorithm has been validated through actual vehicle tests on two different testbeds. The first testbed is located in Seoul National University (SNU), Gwanak-gu Seoul-si. A straight road test at the SNU campus beltway testbed was validated. The second testbed is located in the Future Mobility Technology Center (FMTC), Gyeonggi-do Siheung-si Seoul National University. On the FMTC testbed, we validated the system with test scenarios inclusive of curved roads. The testbed environment is depicted in Fig. 18 (Regions highlighted in orange show the main test area).

All velocity-specific (30 - 50 kph) test results are shown simultaneously in Fig. 19. The results for station and acceleration errors from designed velocity, ranging from 30 kph to 50 kph, are shown in Fig. 19. Of the two kinds of errors show, most have values within an acceptable range. Detailed error analysis is tabulated in Table V. However, a high degree of variance in acceleration error can be observed. The right axis of Fig. 19 represents the gear state of the vehicle. A high degree of variance in acceleration error can be observed during a change in gear number. Further detailed and specific changes were obtained through additional experiments, shown in Fig. 20. The three error values exist between the range of -0.1445 and 0.9815. From Fig.21, it can be observed that these values are within an acceptable range. The value of the lateral axis in Fig. 21 includes numerically meaningful but physically meaningless units. However, the station error has the highest value at around 0.3 m. These values mainly occurred upon switching off the autonomous driving mode. When this fail-safe control methodology was applied to the autonomous vehicle, the control input value changed to a constant negative value when vehicle velocity was close to zero. It is to be noted that the error values beyond 10 seconds in Fig. 20 and Fig. 21 do not hold any meaning. Overall, the vehicle experiments have shown that errors occur within reasonable boundaries.

The RoA figure shows a locally and asymptotically stable value that could be calculated using the Lyapunov function [13]. Fig. 22 contains two figures. The first background figure represents a contour, indicated mainly in yellow, which is a part of the set and satisfies \(\dot{V}_{\text{ref}} < -V_{\text{ref}}\). The
In this paper, a newly designed fail-safe structure for autonomous vehicles has been proposed. The main contributions of this paper can be summarized into three parts. First, the framework of the fail-safe module for autonomous vehicles was introduced and implemented to an actual autonomous vehicle. Second, an adaptive sliding mode observer methodology for longitudinal fault detection about chassis sensors and environment sensors (lidar, radar) in autonomous vehicles was presented. An MIT rule-based adaptation rule to determine the magnitude of the observer injection term was proposed for the reconstruction of unknown faults. Third, fault types of autonomous vehicles are defined through vehicle experimentation. The designed logic has been applied to vehicles and operated to manage a fault situation. In the fail-safe control portion, fault was defined as any situation where a shutdown of the upper controller occurs, be it from continuous vibrations or large physical shocks. The lower controller conducts safety control using the last available scene information from the upper controller. The stability of the system was proved, and the proposed algorithm was investigated via actual vehicle tests. The test results have shown that the proposed algorithm was able to perform the desired vehicle actions successfully. Additionally, error variance has shown to converge within acceptable ranges. The vehicle tests have led to the discovery of a fault in autonomous vehicles' upper controls.

Fault detection can occur in a number of cases, including, but not limited to, sensor performance degradations and adverse weather conditions. Decision tree-based pull-over controls will be the topic of future research in this field for our research team.

For future research, our research team plans to further develop the fail-safe module for an automated vehicle. We plan to utilize a LiDAR point-wise de-noising approach to classify and filter out noise in adverse weather conditions. Our team also plans to develop a tolerant control scheme to cover fatal cases of autonomous vehicle faults, excluding actuator faults. Furthermore, an emergency pull-over algorithm for fail-safe systems corresponding to level-4 autonomous driving is currently being investigated, with emphasis on the control method for achieving a safe emergency pull-over maneuver.

REFERENCES

[1] National Highway Traffic Safety Administration. (2017). Automated Driving Systems 2.0: A Vision for Safety. US Department of Transportation, Washington.

[2] Waymo, Safety Report: ON THE ROAD TO FULLY SELF-DRIVING

[3] GM, SELF-DRIVING SAFETY REPORT, 2018, 11

[4] Baidu, “Apollo Auto,” https://github.com/ApolloAuto/apollo, 2019, [Online].

[5] Z. Gao, C. Cecati and S. X. Ding, “A Survey of Fault Diagnosis and Fault-Tolerant Techniques—Part I: Fault Diagnosis With Model-Based and Signal-Based Approaches,” in IEEE Transactions on Industrial Electronics, vol. 62, no. 6, pp. 3757-3767, June 2015, doi: 10.1109/TIE.2015.2417501.

[6] Dhaïri, Slim & Ben Hmida, Faycal & Sellami, Anis & Gossa, Moncef. (2007). LMI-based sliding mode observer design method for reconstruction of actuator and sensor faults. 1.
[7] Farzin Nemati, Seyed Mostafa Safavi hamami, Ali Zemouche. A nonlinear observer-based approach to fault detection, isolation and estimation for satellite formation flight application. Automatica, Elsevier, 2019, 107, pp.474-482. (10.1016/j.automatica.2019.06.007). (hal-02298581)

[8] J. Han, X. Liu, X. Wei, X. Hu, and H. Zhang, "Reduced-order Observer based Fault Estimation and Fault-Tolerant Control for Switched Stochastic Systems with Actuator and Sensor Fault," in *IEEE Transactions: Elsevier*, vol. 88, pp. 91-101, May 2019.

[9] R. Liu, B. Yang, E. Zio, and X. Chen, "Artificial Intelligence for Fault Diagnosis of Rotating Machinery: A Review," in *Mechanical Systems and Signal Processing: Elsevier*, vol. 108, pp. 33-47, Aug 2018.

[10] S. Lu, P. Zhou, X. Wang, Y. Liu, F. Liu, and J. Zhao, "Condition Monitoring and Fault Diagnosis of Motor Bearings using Undersampled Vibration Signals from a Wireless Sensor Network," in *Journal of Sound and Vibration: Elsevier*, vol. 414, no. 3, pp. 81-96, Feb 2018.

[11] J. Medina-Garcia, T. Sanchez-Rodriguez, and J. Galan, "A Wireless Sensor System for Real-Time Monitoring and Fault Detection of Motor Arrays," in *Sensors 2017: MDPI*, vol. 17, no. 3, pp. 1-22, 2017.

[12] Y. Jeong, S. Song, E. Jeong, and B. Lee, "An Integrated Self-Diagnosis System for an Autonomous Vehicle Based on an IoT Gateway and Deep Learning," in *Applied Sciences: MDPI*, vol. 8, no. 7, pp. 1-24, 2018.

[13] Z. Gao, C. Cecati, and S. Ding, "A Survey of Fault Diagnosis and Fault-Tolerant Techniques Part I: Fault Diagnosis with Model-Based and Signal-Based Approaches," *IEEE Trans. Industrial Electronics*, vol. 62, no. 6, pp. 3757-3767, 2015.

[14] K. Oh, J. Lee, and K. Yi, "Sliding Mode Observer and Control-Based Approach for Longitudinal Fault-tolerant Control Algorithm of Autonomous Vehicles" Trans. Korean Soc. Mech. Eng. B, Vol. 43, No. 1, p. 31-40, 2019 https://doi.org/10.3795/KSME-B.2019.43.1.031

[15] Khalil, Hassan K. (1996). Nonlinear Systems( vol. 2, no. 5). New pp. 4819–4830, Jun. 2018.

[16] H. Chen, B. Jiang, N. Lu, and Z. Mao, "Deep PCA based realtime 3874–3884, Jun. 2015.

[17] Incipient fault detection and diagnosis methodology for electrical dynamic systems," *IEEE Trans. Ind. Electron., vol. 62, no. 6, pp. 1655-1614, Sep 2007.

[18] F. Nemati, S. Hamami, and A. Zemouches, "A Nonlinear Observer-based Approach to Fault Detection, Isolation and Estimation for Satellite Formation Flight Application," in *Automatica: Elsevier*, vol. 107, pp. 474-482, Sep 2019.

[19] National Highway Traffic Safety Administration. Automated Vehicle 3.0: Preparing for the Future of Transportation. US Department of Transportation, Washington, 2018

[20] Seoul city Autonomous vehicle application website, http://topis.seoul.go.kr/intro/openTestBed.do

[21] Rajamani, R., Vehicle dynamics and control. 2011: Springer Science & Business Media.

[22] Suh, J. and K. Yi. A new adaptive uncertainty propagation method based Stochastic Model Predictive Control for automated driving vehicles. in 2017 American Control Conference (ACC). 2017. IEEE.

[23] H. Chae and K. Yi, "Virtual Target-Based Overtaking Decision, Motion Planning, and Control of Autonomous Vehicles," in IEEE Access, vol. 8, pp. 51363-51376, 2020, doi: 10.1109/ACCESS.2020.2980391.

[24] K. Oh, S. Park, J. Lee, and K. Yi, “Functional perspective based probabilistic fault detection and diagnosis algorithm for autonomous vehicle using longitudinal kinematic model,” Microsystem Technologies, Vol. 24, No. 11, pp. 4527-4537, 2018.

[25] T. Song, J. Lee, K. Chae, K. Yi, “Dual-sliding mode approach for separated fault detection and tolerant control for functional safety of longitudinal autonomous driving,” Proceedings of the Meche, Part D: Journal of Automobile Engineering, Vol. , No. , pp. -, 2020.

[26] C. Pek and M. Althoff, "Fail-Safe Motion Planning for Online Verification of Autonomous Vehicles Using Convex Optimization," in IEEE Transactions on Robotics, vol. 37, no. 3, pp. 798-814, June 2021, doi: 10.1109/TRO.2020.3036624.

[27] Pek, C., Manzinger, S., Koschi, M. et al. Using online verification to prevent autonomous vehicles from causing accidents. Nat Mach Intell 2, 518–528 (2020). https://doi.org/10.1038/s42256-020-0225-y

[28] S. Magdici and M. Althoff, “Fail-safe motion planning of autonomous vehicles,” 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), 2016, pp. 452-458, doi: 10.1109/ITSC.2016.7794934.

[29] Fabian Hart, Mustafa Saraoglu, Andrey Morozov, Klaus Janschek, Prioritization and Safety-Critical Fault Tolerant Control Strategy for an Autonomous Vehicle. Sensors 2018, 18, 1893.

[30] T. Song, J. Lee, K. Oh, “Dual-sliding mode approach for separated fault detection and tolerant control for functional safety of longitudinal autonomous driving” in Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, vol. 235, issue. 2, 2020, doi: 10.1177/0954407020962627
Oh, Kwang & Lee, Jongmin & Yi, Kyong. (2019). Sliding Mode Observer and Control-Based Approach for Longitudinal Fault-tolerant Control Algorithm of Autonomous Vehicles. Transactions of the Korean Society of Mechanical Engineers - A. 43. 31-40. 10.3795/KSME-A.2019.43.1.031.

Song, Tae & Lee, Hye & Oh, Kwang & Park, Sung & Yi, Kyong. (2019). A Fault Level Decision Algorithm for Functional Safety of Autonomous Vehicles and Longitudinal Fault Detection Based Performance Evaluation. Transactions of the Korean Society of Mechanical Engineers - A. 43. 119-128. 10.3795/KSME-A.2019.43.2.119.

K. Oh, S. Park, J. Lee, and K. Yi, “Functional perspective based probabilistic fault detection and diagnosis algorithm for autonomous vehicle using longitudinal kinematic model,” Microsystem Technologies, Vol. 24, No. 11, pp. 4527-4537, 2018.

Garcia Daza, I.; Rentero, M.; Salinas Maldonado, C.; Izquierdo Gonzalo, R.; Hernández Parra, N.; Ballardini, A.; Fernandez Llorca, D. Fail-Aware LIDAR-Based Odometry for Autonomous Vehicles. Sensors 2020, 20, 4097. https://doi.org/10.3390/s20154097

L. Guo, S. Manglani, Y. Liu and Y. Jia, "Automatic Sensor Correction of Autonomous Vehicles by Human-Vehicle Teaching-and-Learning," in IEEE Transactions on Vehicular Technology, vol. 67, no. 9, pp. 8085-8099, Sept. 2018, doi: 10.1109/TVT.2018.2846593.

Q. Shi and H. Zhang, "Fault Diagnosis of an Autonomous Vehicle With an Improved SVM Algorithm Subject to Unbalanced Datasets," in IEEE Transactions on Industrial Electronics, vol. 68, no. 7, pp. 6248-6256, July 2021, doi: 10.1109/TIE.2020.2994886.

Geng, K.; Chulin, N.A.; Wang, Z. Fault-Tolerant Model Predictive Control Algorithm for Path Tracking of Autonomous Vehicle. Sensors 2020, 20, 4245. https://doi.org/10.3390/s20154245

C. Zhang and P. Woodland, “Parameterised sigmoid and ReLU hidden activation functions for DNN acoustic modeling,” in Proc. 16th Annu. Conf. Int. Speech Commun. Assoc., 2015, pp. 3224–3228.

Jong Min, L., Oh, K., and Song, T., "Development of Fault Detection and Emergency Control for Application to Autonomous Vehicle," SAE Technical Paper 2021-01-0075, 2021.

Jongmin Lee, Kyongyu Yi, Kwangseok Oh. "Development Of An Emergency Control Algorithm For A Fail-Safe System In Automated Driving Vehicles ", The 26th International Technical Conference and exhibition on the Enhanced Safety of Vehicles (ESV), June 2019.

Biddle, L., Fallah, S. A Novel Fault Detection, Identification and Prediction Approach for Autonomous Vehicle Controllers Using SVM. Automot. Innov. 4, 301–314 (2021).

Jun Hu, Hongyu Zhang, Hongjian Liu & Xiaoyang Yu (2021) A survey on sliding mode control for networked control systems, International Journal of Systems Science, 52:6, 1129-1147

H. Zhang, J. Hu and X. Yu, "Adaptive Sliding Mode Fault-Tolerant Control for a Class of Uncertain Systems With Probabilistic Random Delays," in IEEE Access, vol. 7, pp. 64234-64246, 2019, doi: 10.1109/ACCESS.2019.2917530.

Zhang, Z., Niu, Y., & Song, J. (2020). Input-to-state stabilization of interval-type-2 fuzzy systems subject to cyberattacks: an observer-based adaptive sliding mode approach. IEEE Transactions on Fuzzy Systems, 28(1), 190–203. https://doi.org/10.1109/TFUZZ.91

Zhang, Y., Xie, S., Ren, L., & Zhang, L. (2019). A new predictive sliding mode control approach for networked control systems with time delay and packet dropout. IEEE Access, 7, 134280–134292. https://doi.org/10.1109/ACCESS.6287639

D. Ao, W. Huang, P. K. Wong and J. Li, "Robust Backstepping Super-Twisting Sliding Mode Control for Autonomous Vehicle Path Following," in IEEE Access, vol. 9, pp. 123165-123177, 2021

S. Wang, Y. Hui, X. Sun and D. Shi, "Neural Network Sliding Mode Control of Intelligent Vehicle Longitudinal Dynamics," in IEEE Access, vol. 7, pp. 162333-162342, 2019, doi: 10.1109/ACCESS.2019.2949992.

B. Peng, D. Yu, H. Zhou, X. Xiao and Y. Fang, "A Platoon Control Strategy for Autonomous Vehicles Based on Sliding-Mode Control Theory," in IEEE Access, vol. 8, pp. 81776-81788, 2020, doi: 10.1109/ACCESS.2020.2990644.

P. Wang, S. Gao, L. Li, S. Cheng and L. Zhao, "Automatic Steering Control Strategy for Unmanned Vehicles Based on Robust Backstepping Sliding Mode Control Theory," in IEEE Access, vol. 7, pp. 64984-64992, 2019, doi: 10.1109/ACCESS.2019.2917507.

Y. Han, Y. Cheng and G. Xu, "Trajectory Tracking Control of AGV Based on Sliding Mode Control With the Improved Reaching Law," in IEEE Access, vol. 7, pp. 20748-20755, 2019

Jongmin Lee is Korean. He received his B.S. in department of automotive engineering from Tsinghua University, China in 2015. He received his M.S. in mechanical engineering from Seoul National University, Korea in 2018. He is a Ph.D. student in mechanical engineering at Seoul National University. His research interests are fail-safe systems for autonomous vehicles and autonomous vehicle environmental sensor degradation (De-noising point-wise noise data of Lidar sensor in adverse weather)

Kwangseok Oh received a BS degree in mechanical engineering from the University of Hanyang, Seoul in 2009 and an M.S degree and a PhD in mechanical and aerospace engineering from Seoul National University, Seoul in 2013 and 2016. Since 2017, he has been an assistant professor in the Mechanical Engineering Department of Hankyong National University. His research interests include fail-safe systems for autonomous driving, adaptive control.

Youngmin Yoon received a B.S. in mechanical engineering from Seoul National University, South Korea, in 2018, He is currently pursuing a Ph.D. in mechanical engineering at Seoul National University, South Korea. His research interests include motion prediction of surrounding vehicles, predictive motion planning, and automated driving vehicle control.

Taejun Song received the B.S. and M.S degrees in mechanical engineering from Hankyong National University, Korea, in 2018 and 2020, respectively. He is a research engineer at Smart Mobility Lab, Korea. His research interests include the path planning and tracking control using LiDAR, and model-based sensor fault detection for autonomous vehicles.

Taehoon Lee received his B.S. in Mechanical Engineering from Seoul National University in 2022. He is currently an aspiring Ph.D. student in the Mechanical and Aerospace Engineering Faculty of Seoul National University. His research interests include motion planning and situational decision making for autonomous vehicles.

Kyongyu Yi received B.S. and M.S. degrees in mechanical engineering from Seoul National University, South Korea, in 1985 and 1987, respectively, and a Ph.D. degree in mechanical engineering from the University of California, Berkeley, in 1992. He is currently a Professor at the School of Mechanical Engineering at Seoul National University, South Korea. He currently serves as a member of the editorial board of Mechatronics and Chair of KSME IT Fusion Technology Division. His research interests are control systems, driver assistant systems, active safety systems, and automated driving of ground vehicles.