SAE INTERNATIONAL

ARTICLE INFO
Article ID: 11-01-01-0003
Copyright © 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.
doi:10.4271/11-01-01-0003

Anomaly-Based Intrusion Detection Using the Density Estimation of Reception Cycle Periods for In-Vehicle Networks

Yoshihiro Hamada, *Sumitomo Electric Industries, Ltd.*
Masayuki Inoue, Hiroshi Ueda, and Yukihiro Miyashita, *AutoNetworks Technologies, Ltd.*
Yoichi Hata, *Sumitomo Electric Industries, Ltd.*

Abstract

The automotive industry intends to create new services that involve sharing vehicle control information via a wide area network. In modern vehicles, an in-vehicle network shares information between more than 70 electronic control units (ECUs) inside a vehicle while it is driven. However, such a complicated system configuration can result in security vulnerabilities. The possibility of cyber-attacks on vehicles via external services has been demonstrated in many research projects. As advances in vehicle systems (e.g., autonomous drive) progress, the number of vulnerabilities to be exploited by cyber-attacks will also increase. Therefore, future vehicles need security measures to detect unknown cyber-attacks. We propose anomaly-based intrusion detection to detect unknown cyber-attacks for the Control Area Network (CAN) protocol, which is popular as a communication protocol for in-vehicle networks. For the easy deployment and maintenance of the IDS, the proposed method learns the behavior model online. We compared the proposed method with conventional methods using captured CAN traffic data, and confirmed that, under attack, only the proposed method simultaneously achieved a high illegitimate frame detection rate and correct frame detection rate.

History

Received: 17 Aug 2017
Revised: 27 Dec 2017
Accepted: 01 May 2018
e-Available: 16 May 2018

Keywords

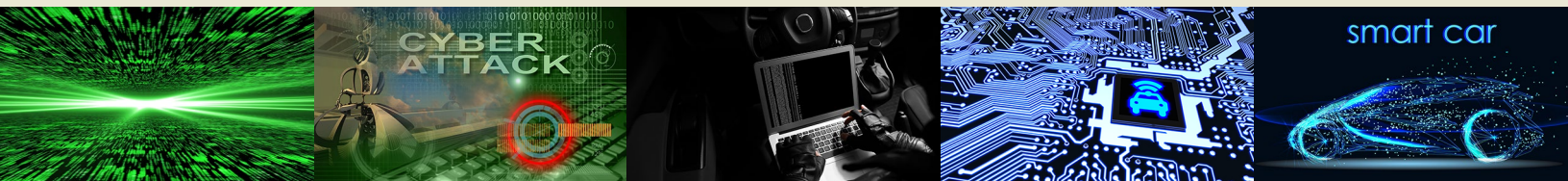
Embedded security,
In-vehicle network,
Controller area network,
Intrusion detection system

Citation

Hamada, Y., Inoue, M., Ueda, H., Miyashita, Y. et al., "Anomaly-Based Intrusion Detection Using the Density Estimation of Reception Cycle Periods for In-Vehicle Networks," *SAE Int. J. Transp. Cyber. & Privacy* 1(1):39–56, 2018, doi:10.4271/11-01-01-0003.

ISSN: 2572-1046
e-ISSN: 2572-1054

© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.; Published by SAE International. This Open Access article is published under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0/, which permits use, distribution, and reproduction in any medium, provided that the original author(s) and the source are credited.



Introduction

Modern vehicles, which have more than 70 electronic control units (ECUs), have improved in terms of safety and convenience by sharing control data through various service networks outside the vehicle [1]. However, such a complicated system configuration can result in security vulnerabilities for vehicles. Possible cyber-attacks on vehicles from external services have been demonstrated in many research projects [2, 3, 4, 5]. The purpose of these attacks is to illegally control in-vehicle functions by injected spoofed frames into in-vehicle networks from outside the vehicle. These attacks are very dangerous; they can place passengers in danger, contrary to the intention of the vehicle design. Moreover, to date, the scale of the program code for the premium-class vehicle system is 100 million lines, and that for technology such as autonomous drive and advanced driver-assistance systems is expected to reach 200-300 million lines in the future [1]. The remaining vulnerabilities in the vehicle system increase proportionally to the size of the program code. Furthermore, a particular vehicle model lifecycle is said to be more than a decade. This length of time gives attackers opportunities to learn about target vehicles. Therefore, security measures, such as detecting unknown cyber-attacks that exploit vulnerabilities, will be important in future vehicles.

An intrusion detection system (IDS) is a known security measure for unknown cyber-attacks [6]. If this method is combined with a prevention function, then it can detect cyber-attacks that a vehicle experiences quickly and reduce damage. Implementing a detection function that detects cyber-attacks from a previously unknown vulnerability is needed. Anomaly-based intrusion detection is useful for this purpose. Therefore, many research projects have been conducted in the field of the Control Area Network (CAN) protocol, which is popular as a communication protocol for in-vehicle networks. In many research projects, the communication cycle periods of each CAN identifier (CAN-ID) frame have been used as monitored objects. Because a current injected spoofed frame attack causes a change in the communication cycle period, anomaly-based IDS that monitors the communication cycle period can detect this type of attack efficiently. When deploying anomaly-based IDS, the system needs to learn the behavior model of the monitored objects. It seems simple to monitor the communication cycle period; however, a typical vehicle has many CAN-IDs. To deploy and maintain this type of IDS, it needs to determine many behavior models from many data and continue to retain these learned behavior models. Therefore, this task requires large personnel expenses, which raises the cost of the IDS.

In this article, we propose a new method that can learn the behavior model automatically in the ECU. The proposed method allows the easy deployment and maintenance of anomaly-based intrusion detection in vehicles. Additionally, we propose an anomaly detector for the prevention function that allows the vehicle to run continuously when under attack without stopping the service targeted by the attacker.

Can Protocol and Security Risks

Features of the CAN Protocol

CAN is divided into the conventional CAN protocol and CAN with Flexible Data-Rate (CAN FD) protocol. The CAN protocol is a communication protocol standardized in ISO11898-1(2003), and the CAN FD protocol was reflected in the CAN protocol specification in 2015 as a revision of ISO11898-1(2015) [7, 8]. These protocols enable low-latency frame transmission for control systems with a broadcast transmission of up to an 8-byte data field (CAN) or 64-byte data field (CAN FD) using the node that has acquired the transmission right in communication arbitration out of several nodes (ECUs) in a bus topology.

© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

Transmission Patterns of CAN Frames

The transmission patterns of CAN frames in in-vehicle networks can be classified into two categories. One is periodic frames, which are sent in constant time and used to convey control information, such as controlling the brake, engine, and speedometer reading. Figure 1 shows the reception cycle periods of periodic frames. From both figures, the communication cycle periods fluctuate around the particular communication cycle period (R ms), which is the designed communication cycle period. The other category is aperiodic frames, which are sent irregularly and used to carry control information, such as locking or unlocking doors, and opening or closing windows. The communication cycle periods of aperiodic frames do not necessarily fluctuate around the particular reception time.

FIGURE 1 Periodic frame features. (A) Periodic frame time propagation. (B) Periodic frame histogram.

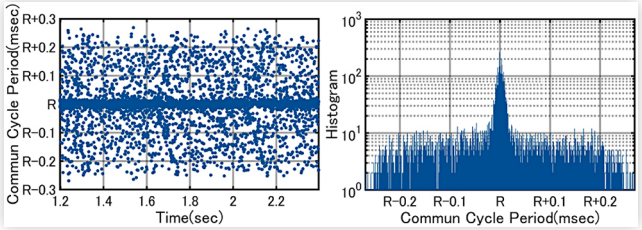
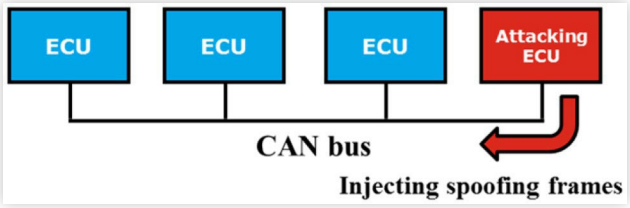


FIGURE 2 Injection of a spoofed CAN frame by a malicious ECU.



Security Threats

Koscher et al. highlighted the three security vulnerabilities of the CAN protocol [9]: (1) is easy to analyze control information flowing through the network; (2) it is easy to inject spoofed frames; and (3) the protocol is weak against denial-of-service (DoS) attacks. An injected spoofed frame attack or DoS attack is conducted through a malicious ECU that is connected to a CAN bus, as shown in Figure 2. A malicious ECU is a device whose firmware has been illegally altered to behave contrary to the intended design or illegally connected to the network.

Security Measures for In-Vehicle Networks

Conventional Security Measures

Conventional studies of security measures for in-vehicle networks are classified into the following two categories: (1) a security measure for a network protocol that can achieve secure communication between communication nodes; and (2) an IDS, which works on a network protocol and can detect the suspicious behavior of an application or network.

Intrusion Detection

Technology Types An IDS has two technology types: host-based IDS and network-based IDS. Host-based IDS is deployed on one computer node; thus, it can monitor a computer system and connected network segment. By contrast, network-based IDS is deployed where it can monitor all network segments. Generally, a vehicle system consists of more than 70 ECUs that are connected by a multi-segment network. Therefore, network-based IDS can monitor a vehicle system efficiently because it can monitor multiple ECUs through network behavior using a few IDSs.

Intrusion Detection Methods There are two types of intrusion detection: signature-based and anomaly-based [10]. They detect intrusions by detecting suspicious activities based on the behavior model. The signature-based method can detect misuse activities that match the misuse cases of a monitored object. Therefore, this method should define the misuse cases of a monitored object for a detection criterion. These

© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

misuse cases are learned from known cyber-attacks. The anomaly-based method allows the detection of the anomalous behavior of a monitored object. Therefore, this method learns the behavior model from typical cases of the monitored object.

Generally, the anomaly-based method has a higher misdetection rate than the signature-based method. However, only the anomaly-based method can detect unknown cyber-attacks that exploit the unknown vulnerabilities of a particular vehicle model during the long lifetime of its model. Therefore, the anomaly-based method is crucial for an in-vehicle IDS.

Conventional Methods of In-Vehicle Intrusion Detection Conventional methods of in-vehicle intrusion detection are assumed to apply to a network-based IDS where, for cost-efficient monitoring, all ECUs are connected to one CAN bus. Additionally, these methods have been based on anomaly-based intrusion detection because research projects have aimed at detecting cyber-attacks that are caused by unknown vulnerabilities in vehicle systems.

Larson et al. proposed specification-based intrusion detection for the CANopen2.1 specification, which works on the upper layer of the CAN protocol [11]. This method detects violations of communication rules as suspicious activities resulting from cyber-attacks.

Otsuka et al. proposed an intrusion detection method that monitors the communication frequency of each CAN frame. This method detects the violations of the communication frequency as suspicious activities resulting from a cyber-attack [12]. For detection, it defines the maximum deviation of the reception cycle period for each frame.

Müter et al. proposed entropy-based intrusion detection [13]. This method detects the violation of entropy of the communication cycle period or data in a data field of each CAN frame. Entropy H is defined as

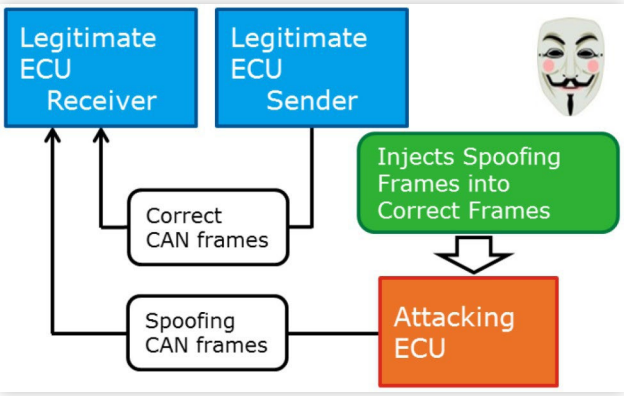
$$H(X) = \sum_{x \in C_X} p(x) \log \frac{1}{p(x)}, \qquad \text{Eq. (1)}$$

where C_X is a set of classes for dataset X and $p(x)$ is the probability of x in X. Thus, it is necessary to define $p(x)$ for the monitored object.

Markovitz et al. proposed another method to monitor data in the data field of each CAN frame [14]. This method detects the violations of the range of each piece of data in the data field. The range of each piece of data is not defined by the bit length of each piece; it is defined by the actual range of data while a vehicle is driven.

Wasicek et al. proposed a semantic-based intrusion detection method that detects the violations of control data using a physical property model of each piece of control data [15].

FIGURE 3 Attacker model: shared bus model.



Issues of Conventional In-Vehicle IDS

To consider issues of conventional in-vehicle IDS, we assume the attacker model shown in Figure 3 and call it the "shared bus" model. For this model, an attacker places a malicious ECU application into a vehicle system and then injects spoofed CAN frames into correct CAN frames, which are being transmitted between a sender and receiver of legitimate ECU, via/from a malicious ECU application. Therefore, a receiver of legitimate ECU receives correct and spoofed CAN frames for a particular CAN-ID. In this attacker model, the attacker can inject spurious control data to control the target vehicle illegally. Additionally, the attacker can intend that a DoS attack disables a particular function in the vehicle.

© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

TABLE 1 Learning features of a conventional in-vehicle IDS.

Class: One of the following two classes of conventional in-vehicle IDS.

"Frame pattern": Monitors the features of a communication pattern in a single CAN-ID frame.

"Context pattern": Monitors the features of context data in a single CAN-ID frame.

Ref: Reference index in section "Reference" for each method. object: Monitored object.

Feature: What each method learns for the behavior model to monitor the object.

Where: Where each method learns the behavior model.

Num.: Number of behavior models learned to monitor a single CAN-ID.

		Behavior model of each CAN-ID			
Class	**Ref**	**Object**	**Feature**	**Where**	**Num**
Frame Pattern	[12]	Reception Frequency	Maximum delay	Offline	1
	[13]	Reception Cycle Period	Probability	Offline	1
Context Pattern	[13]	Difference in context data as time progresses	Probability	Offline	Greaterthan 1
	[14]	Current context data	Range	Offline	Greaterthan 1
	[15]	Current context data	Physical property	Offline	Greaterthan 1

Issues of Conventional Anomaly-Based Intrusion Detection Methods

We classify the four conventional methods that were explained in the subsection "Conventional Methods of In-Vehicle Intrusion Detection" into two classes, as shown in Table 1, to consider the issues of conventional IDSs. The "frame pattern" class monitors communication cycle period and communication frequency. The "context pattern" class monitors the difference in context data as time progresses and the current value of context data. The "Behavior model of each CAN-ID" indicates the features of the learning behavior model for each method: there are four subcategories, "Object", "Feature", "Where" and "Num" as shown in table caption.

From Table 1, the "Frame Pattern" class has two conventional methods [12, 13]. Both methods monitor the reception cycle period of each CAN frame. One learns the maximum reception cycle period for a feature of a behavior model offline [12]. The other learns the probability of the reception cycle period for a feature of a behavior model offline [13]. Moreover, both methods monitor each frame for only one behavior model.

From Table 1, the "Context Pattern" class has three conventional methods [13, 14, 15]. The first method monitors the difference in context data as time progresses, and learns the probability of the monitored object [13]. For the feature of each behavior model, the second method learns a range of monitored objects, and the third method learns a physical property of the monitored object [14, 15]. Both methods also learn their features of the monitored object offline. Moreover, both methods need multiple behavior models to monitor each frame because, typically, a single CAN-ID contains multiple context data.

In Table 2, we consider the advantages and disadvantages of both classes. Offline learning requires a sufficient number of samples of real CAN traffic from multiple vehicles to determine a population of a property of a monitored object. Therefore, offline learning makes the deployment and maintenance of the IDS difficult.

For the deployment and maintenance of the IDS when monitoring all CAN-IDs, fewer behavior models is better than multiple behavior models.

The frame pattern class detects a change in the communication pattern of each frame. Therefore, this class can detect a DoS attack. The context pattern class cannot detect a DoS attack when injected spoofed frames do not include spurious data. This class monitors

TABLE 2 Advantages and disadvantages of two in-vehicle anomaly-based IDS classes.

	Frame pattern	Context pattern
Pro	✓ Fewer behavior models ✓ DoS detection	✓ Spurious data detection
Con	✓ Offline Learning	✓ Offline learning ✓ Multiple behavior models

© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

TABLE 3 Assumptions to estimate the processing time of each frame and result.

Model assumption		Value
Communication protocol	Protocol	CAN
	Communication speed	500 kbps
	CAN-ID	11 bits
System	Occupied CPU	1 CPU
	Number of buses	8
Bus condition	Data field size	1 byte
	Bus utilization	100%
Requirement	Processing time of each CAN frame	13.7 μs

context data directly; thus, the class can detect an injected spoofed frame that contains obvious spurious context data. The frame pattern class cannot detect an injected spoofed frame that contains obvious spurious context data when the change in the communication pattern of the injected spoofed frame is small.

To summarize, for the installation and maintenance of an IDS, the frame pattern class is better than the context pattern class. Additionally, both classes monitor different features of each CAN frame. Therefore, the two classes can help each other to improve the detection performance of injected spoofed frames. However, to learn behavior models, conventional methods of both classes use offline learning. Offline learning makes the deployment and maintenance of IDS difficult. This is a basic problem of anomaly-based IDS. We believe that the more efficient learning method should be considered for anomaly-based IDS.

Issues of Deploying an In-Vehicle IDS

A gateway ECU is suitable for the deployment of in-vehicle IDS because the IDS can access all traffic in CAN buses. Additionally, to mitigate the damage of injected spoofed frames, it is possible to stop forwarding detected spoofed frames to other CAN buses. To deploy in-vehicle IDSs into a gateway ECU, we estimate a requirement for the processing time of each CAN frame. Table 3 shows some assumptions for this estimation that lead to an estimated requirement for the processing time of each CAN frame.

Proposed Method

Scope of the Solution

We propose a solution for the frame pattern class anomaly-based IDS for an in-vehicle system to detect injected spoofed frames under the shared bus model. To achieve the easy deployment of the IDS into a vehicle, the proposed method learns the behavior model of the monitored object online. For this online learning, the proposed method continuously learns a behavior model of each CAN-ID in the ECU. For the behavior model of the monitored object, the proposed method learns the histogram of reception cycle periods using the probability density function (PDF) and we call it the statistic detector (SDEC).

Additionally, we propose an anomaly detector for the prevention function that allows the vehicle to run continuously when under attack without stopping the service targeted by the attacker. Therefore, this method is designed to recognize illegitimate frames and legitimate frames. And, this method is able to start the frame assessment immediately when the frame arrives.

The application model of the proposed method is explained in the subsection, "Application Model of the Proposed Method." The learner learns each PDF of all CAN-IDs as explained in the subsection "Learner." The anomaly detector assesses each CAN frame to detect anomalous CAN frames as explained in the subsection "Anomaly Detector."

Application Model of the Proposed Method

Figure 4 shows the IDS application model for the proposed method. We assume that this model is deployed where all CAN frames of all CAN buses can be monitored so that it can fully understand the behavior of the CAN network. For this case, a gateway ECU is suitable for the implementation of this application. The model is composed of three functions: the learner, anomaly detector, and reporter. In Figure 4, these functions are represented by rectangles, and the input and output objects of each function are represented by ovals. The learner learns the behavior model to monitor each CAN-ID from each CAN frame that arrives and also determines the criterion used to assess the

© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

frame. The anomaly detector detects an anomalous CAN frame based on a behavior model as each CAN frame arrives. When the detection result of the anomaly detector indicates the detected anomalous frame, the reporter reports the detected frame to an in-vehicle prevention system that prevents damage by anomalous frames. Additionally, the reporter also sends information about the detected anomalous frames to a logging system that stores logs of information regarding detected anomalous frames.

Learner

The proposed method learns the PDF of the Gaussian mixture model, which is explained in the subsection "Gaussian Mixture Model," from the reception cycle periods of the monitored CAN-ID using the sequentially discounting expectation and maximization(SDEM) algorithm, which is explained in the subsection "SDEM Algorithm." In Figure 5, a black circle indicates each reception cycle period of the monitored CAN-ID and the middle row of the figure shows each estimated PDF. These reception cycle periods are calculated from the arrival time of each CAN frame of the monitored CAN-ID using the baseline method, which is explained in the subsection "Determination of the Current Reception Cycle Period." The learner continuously estimates each PDF from the reception cycle periods in each slot; a slot contains the expected number of samples, shown as slot 1 to slot 4 in Figure 5. To detect anomalous CAN frames in each slot, the anomaly detector assesses each CAN frame using the PDF that was estimated in the previous slot.

Additionally, the learner determines the criterion from the given policy, which is explained in the subsection, "Determination of the Threshold." This threshold is used to assess the frame in the anomaly detector.

Gaussian Mixture Model We use the PDF of the Gaussian mixture model because this model can create a more complex PDF than that of the Gaussian model. The PDF of the Gaussian mixture model is

$$P(x|\theta) = \sum_{1}^{K} C_k p_k\left(x|\overline{x}_k, {\sigma_k}^2\right), \qquad \text{Eq. (2)}$$

where P is the probability of sample x, K is the number of synthesized Gaussian distribution models, θ is $\left(\overline{x}_k, {\sigma_k}^2\right)$, $\overline{x}_k$ and ${\sigma_k}^2$ are the average and variation of each Gaussian distribution model, respectively, and c_k is a mix ratio that determines the degree to which each Gaussian distribution model is synthesized.

This model is constructed by synthesizing K Gaussian distribution models, where each Gaussian distribution model is given by

$$p_k\left(x|\overline{x}_k, {\sigma_k}^2\right) = \frac{1}{\sqrt{2\pi{\sigma_k}^2}} exp^{\frac{-\left(x-\overline{x}_k\right)^2}{2{\sigma_k}^2}}, \qquad \text{Eq. (3)}$$

where p is the probability of sample x, k is the number of each Gaussian distribution model, and $\overline{x}_k$ and ${\sigma_k}^2$ are the average and variation of each Gaussian distribution model, respectively.

FIGURE 4 IDS application model of the proposed method.

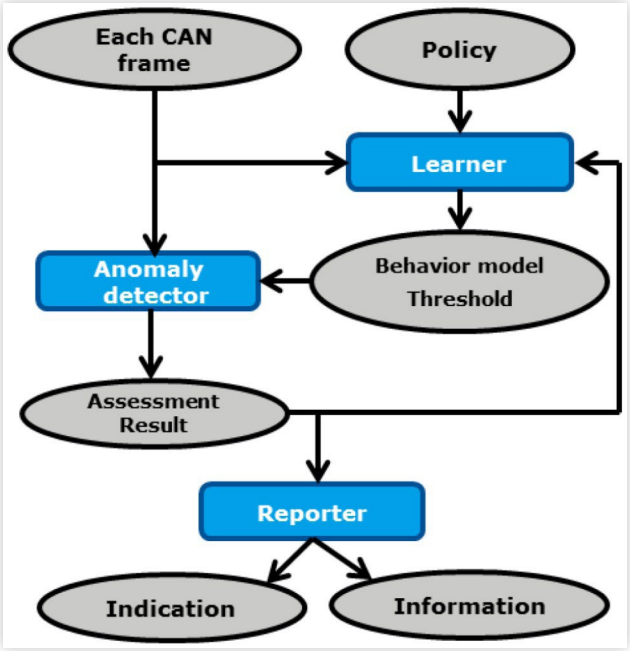
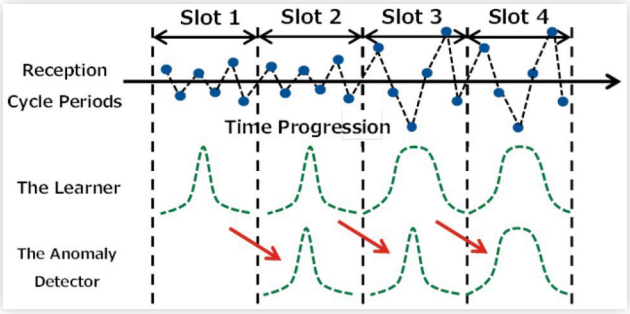


FIGURE 5 Learning process of the proposed method.



© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

SDEM Algorithm We use the SDEM algorithm to estimate the PDF of the Gaussian mixture model on the ECU. We provide a short explanation of the implementation of the SDEM algorithm for our proposed method [16]. The algorithm estimates the parameters of the PDF of each Gaussian mixture model; c_k, $\bar{x}_k$, and σ_k^2. Additionally, this algorithm reduces the effect of past samples on current parameters when updating parameters in a fixed time frame. Therefore, this algorithm can track variations of the PDF of the Gaussian mixture model in time. This updating of parameters is given by

$$S_k^{(t)} = (1-\eta)\cdot S_k^{(t-1)} + \eta\cdot\left(\gamma_k^{(t)}, \gamma_k^{(t)}\boldsymbol{x}, \gamma_k^{(t)}\boldsymbol{x}\boldsymbol{x}^T\right), \quad \text{Eq. (4)}$$

and reduces the effect of past samples using a discounting parameter η ($0 < \eta < 1$). $S_k^{(t)}$ denotes the current sufficient statistics, and it stores the parameters of the PDF of the Gaussian mixture model. $S_k^{(t-1)}$denotes the previous sufficient statistics. x is the value of the current sample. $\gamma_k^{(t)}$ is given by.

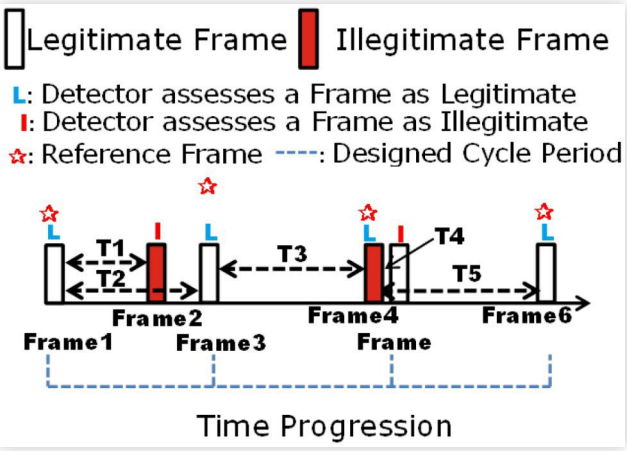
$$\gamma_k^{(t)} = (1-\alpha\eta)\frac{c_k^{(t-1)} p_k\left(x|\bar{x}_k^{(t-1)}, \sigma_k^{2(t-1)}\right)}{\sum_{i=1}^{K} c_i^{(t-1)} p_i\left(x|\bar{x}_i^{(t-1)}, \sigma_i^{2(t-1)}\right)} + \frac{\alpha\eta}{k},\ k = 1,2\ldots K \quad \text{Eq. (5)}$$

and is part of the process of updating the parameters. α is introduced for the stability estimation of c_k, and is set from 1.0 to 2.0. However, we did not make this parameter estimation stable using α in our experimental environment. We introduced the mini-batch method instead of parameter α. The mini-batch method updates the sufficient statistics for each set of m samples, where m is the mini-batch size. Each value of the samples is summed for each set of m samples, and then the current sufficient statistics are updated from the average of m samples. For the proposed method, we need to adjust discounting parameter η and the mini-batch size m to learn the PDF of the Gaussian mixture model. We set α to zero. We call this method the mini-batched SDEM algorithm.

Determination of the Current Reception Cycle Period We define two determination methods for the current reception cycle period: the baseline method and previous method. For the proposed method, we use the baseline method. This is because the proposed method intends to recognize legitimate and illegitimate frames for the frame assessment, thus, the proposed method needs to know the reception cycle period of each frame from the reference frame in the designed cycle period. However the previous method will be used in the subsection "Experiments;" thus, we explain it here.

The baseline method is shown in Figure 6. In this figure, a white box indicates a legitimate frame and a red box indicates an illegitimate frame. "L" and "I" indicate assessment results of the anomaly detector, where "L" denotes legitimate and "I" denotes illegitimate. "Reference frame" indicates the frames that are assessed to be legitimate by the proposed method; they are marked at the top of the assessment results. The baseline method determines the reception cycle period for each frame by calculating the elapsed time from the reference frame, which arrived faster than the current arrived frame. The advantage of this method is that this method can determine the reception cycle period from reference frame in the designed cycle period. The disadvantage of this method is that this method needs to retain the reference frame.

FIGURE 6 Baseline method: how the current reception cycle period is determined.



The previous method is shown in Figure 7. This method determines a reception cycle period for each frame by calculating the elapsed time from the previous arrived frame. This method has two advantages. The one of them is that this method does not need to retain the reference frame. The other is that this method is more sensitive illegitimate frames than the baseline method. This is because the determination result of the reception cycle period of this method is shorter than the baseline method when injected illegitimate frames. The disadvantage of this method is

© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

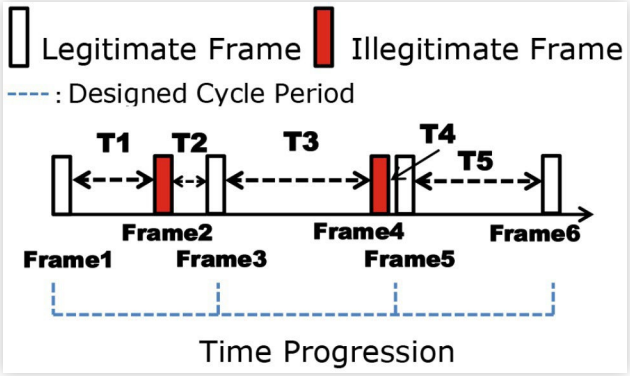
that this method cannot determine the reception cycle period from reference frame in the designed cycle period.

Determination of the Threshold The learner determines the static threshold using the given policy and the assessment result from the anomaly detector, as shown in Figure 4. The given policy is the allowable false positive rate (FPR) for each CAN-ID. The FPR indicates the incorrect rate of the legal assessment results of all legitimate frames as explained in the subsection "Terms for detection performance." The learner experimentally determines the static threshold that is satisfied for the given FPR.

In the determination sequence, the learner uses the given FPR as the target FPR. First, the learner sets the temporal threshold, and then aggregates the frame assessment results from the anomaly detector to calculate the FPR. If the calculated FPR is greater than the target FPR, then the learner changes the previous threshold. If the calculated FPR is greater than the target FPR, then the threshold should be larger than the previous threshold. If the calculated FPR is less than the target FPR, then the threshold should be lower than the previous threshold. This sequence is repeated until the threshold satisfies the convergence condition. This convergence condition is given as the allowable error. When the difference between the target FPR and calculated FPR is within the allowable error, the learner determines the static threshold for each CAN-ID.

This determination sequence assumes that all CAN frames are legitimate. Therefore, this sequence should be performed in an environment in which it is possible to isolate the vehicle from cyber-attacks.

FIGURE 7 Previous method: how the current reception cycle period is determined.



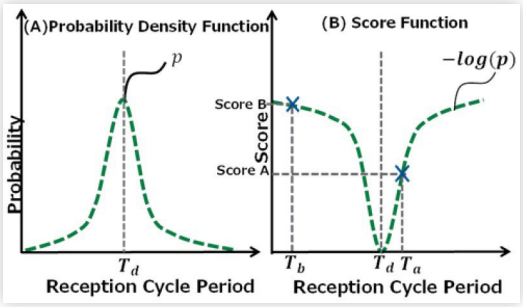
Anomaly Detector

To achieve the countermeasure using the IDS in the vehicle, we designed an anomaly detector that assesses when the CAN frame arrives immediately, as explained in the subsection "Assessment Method." Each reception cycle period of the monitored frame is assessed using a score calculated as

$$Score = -\log\big(P(x)\big), \qquad \text{Eq. (6)}$$

where P is the PDF and x is the current reception cycle period. Figure 8 shows the relationship between the PDF and score. Figure 8(A) shows the PDF of the Gaussian mixture model given in Equation 2. A bold dotted line indicates the PDF of the Gaussian mixture model. "T_d" denotes the designed communication cycle period. Figure 8(B) shows the score function given in Equation 6. The horizontal axis indicates the reception cycle periods and the vertical axis indicates the score. A bold dotted line indicates the score function. "T_d" denotes the designed communication cycle period. "T_a" and "T_b" denote the scores. Equation 6 calculates "Score A" from "T_a" and "Score B" from "T_b". If these values are close to the designed reception cycle period, then the score is at a minimum.

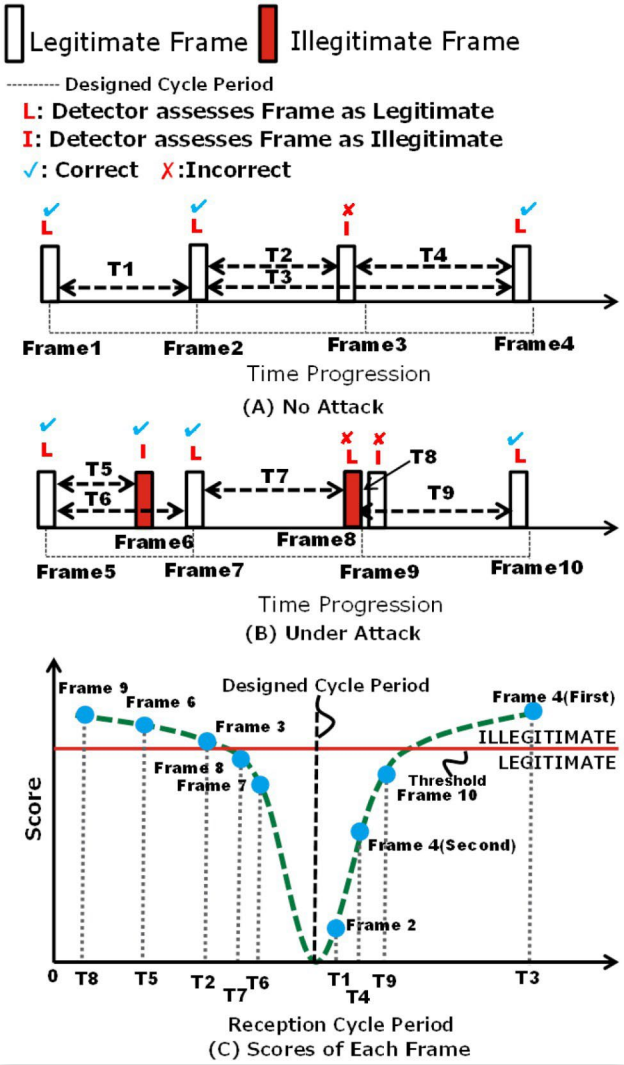
FIGURE 8 Relationship between the PDF and score.



Assessment Method The proposed method evaluates the score of a reception cycle period for an arrived frame by comparing it with a threshold. If the score of a reception cycle period is higher than the threshold, then the proposed method assesses the frame to be an illegitimate frame. Figure 9 shows how the proposed method works. Figure 9(A) and (B) show a time chart of arrived frames when a CAN bus is not under attack and when a CAN bus is under attack, respectively; a non-hatched box indicates a legitimate frame that the legitimate ECU sent and a hatched box indicates an illegitimate frame that is an injected spoofed frame. Thin dashed lines indicate the designed reception cycle period of

© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

FIGURE 9 Proposed method detection process for a spoofed frame.



the monitored CAN-ID. Additionally, T1 to T9 indicate the reception cycle periods for each frame, Frame 1 to Frame 10. "L" and "I" indicate assessment results of the proposed method, where "L" denotes legitimate and "I" denotes illegitimate. Figure 9(C) shows a score chart. A bold dotted line indicates the score of the PDF that is calculated by Equation 6. The bold solid line indicates the threshold. If a score of a reception cycle period for a frame, T1 to T9, is less than the threshold, then the proposed method assesses the frame to be a legitimate frame. If the score is greater than the threshold, then the proposed method assesses the frame to be an illegitimate frame. If the assessment is correct, then the "Correct" symbol is marked at the top of the assessment result. If the assessment is incorrect, then the "Incorrect" symbol is marked at the top of the assessment result.

For the assessment of the proposed method when a CAN bus is not under attack, the Frame 2 score is calculated by Equation 2 from T1 (we assume that the reference frame is Frame 1 at first) and the calculated score of Frame 2 is less than the threshold (shown as a black circle in Figure 9(C)); thus, Frame 2 is assessed to be the legitimate frame. This decision is correct; the "Correct" symbol is marked at the top of the assessment result. The Frame 3 score is calculated from T2 (the reference frame is the previously legitimate Frame 2) and the calculated score is greater than the threshold; thus, Frame 3 is assessed to be the illegitimate frame. This decision is incorrect. Then, the Frame 4 (first) score is calculated from T3 (the reference frame is still Frame 2), and the calculated score is substantially greater than the threshold. In this case, our proposed method needs to recover the reference frame because after this, Frame 2 will be used as the reference frame. This recovery is performed using the previously arrived frame as the new reference frame (Frame 3 is the new reference frame). Then, Frame 4 is assessed again. The Frame 4 (second) score is calculated from T4, and the calculated score is less than the threshold; thus, Frame 4 is assessed to be the legitimate frame. The recovery process is complete after the second assessment, regardless of its result.

For the assessment of the proposed method when a CAN bus is under attack, the Frame 6 score is calculated from T5 (the reference frame is the previously legitimate Frame 5) and the calculated score of Frame 6 is greater than the threshold; thus, Frame 6 is assessed to be the illegitimate frame. This decision is correct. The Frame 7 score is calculated from T6 (the reference frame is still the previously legitimate Frame 5) and the calculated score of Frame 7 is less than the threshold; thus, Frame 7 is assessed to be the legitimate frame. This decision is correct. The Frame 8 score is calculated from T7 (the reference frame is the previously legitimate Frame 7) and the calculated score of Frame 8 is slightly less than the threshold; thus, Frame 8 is assessed to be the legitimate frame. This decision is incorrect. The Frame 9 and Frame 10 scores are calculated from T8 and T9, respectively (the reference frame is the previously legitimate Frame 7), and the calculated score of Frame 9 is greater than the threshold; thus, Frame 9 is assessed to be the illegitimate frame. This decision is incorrect. Additionally, the calculated score of Frame 10 is less than the threshold; thus, Frame 10 is assessed to be the legitimate frame. This decision is correct.

Experiments

Using the traffic data obtained from a test vehicle, we evaluated the processing time and detection performance of the proposed method. Both experimental results are

© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

presented in the subsections "Processing Time" and "Detection Performance," respectively.

Evaluation Environment

Table 4 summarizes the environment used for evaluation. The evaluation was performed on MATLAB® with its statistical analysis option. We used parameters of the mini-batched SDEM throughout our experiments, as shown in Table 5. We prepared two types of CAN traffic data for the detection performance evaluation: one was CAN traffic data when not under attack, which is explained in the subsection "CAN Traffic Data under No Attack," and the other was CAN traffic data under attack, which is explained in the subsection "CAN Traffic Data Under Attack."

CAN Traffic Data under No Attack For evaluations under no attack, we used captured traffic data from No. 1 to No. 7 shown in Table 6. For this experiment, we drove a test vehicle at the company site so as not to exceed the upper limit speed, while stopping to follow traffic rules. Additionally, we selected the periodic CAN-ID with periodic deviation variations. The designed communication cycle period of the selected CAN-ID was 30 ms.

CAN Traffic Data under Attack Figure 10 shows one illegitimate traffic dataset. The thin line indicates the designed communication cycle period, 30 ms, for the selected CAN-ID. "Injection time" indicates the injected time measured from the previous legitimate frame.

For evaluations under attack, we created 300 variations of the illegitimate traffic dataset for each captured CAN traffic data file. One illegitimate traffic dataset had one illegitimate frame for every two legitimate frames. This illegitimate frame always had the same injection time for each illegitimate traffic dataset. The injected time was varied from 0.1 ms to 30 ms at 0.1 ms intervals. This resulted in 300 illegitimate traffic datasets for one captured dataset. We evaluated the detection performance for all possible injection times of the spoofed frame for the selected CAN-ID using this dataset with the transmission time (0.11 ms) of the shortest CAN frame (shortest frame length was 55 bits and the communication speed was 500 kbps).

Processing Time To evaluate the processing time of the proposed method, we implemented it on a popular CPU for the in-vehicle network, which had a clock speed of 120 MHz and a floating point unit. The implemented proposed method had the learner and the anomaly detector. In this evaluation, we assumed that the proposed method worked on the gateway ECU with one occupied CPU core, as shown in Table 3.

Table 7 shows the evaluation results. "Default" and "Speed" indicate the optimization options of the compiler. "Frame assessment" indicates the processing time of the assessment of a CAN frame, as explained in the subsection "Anomaly Detector." "Mini-batched SDEM with the second Gaussian mixture model" indicates the processing time of the learner, as explained in the subsection "Learner." "Other time" indicates the overhead time to call the learner and anomaly detector.

TABLE 4 Evaluation environment.

Experimental item	Value
PC	Dell OptiPlex 7040 Intel Core i7-6700 CPU 3.4 GHz Memory 16 GB
MATLAB	Version 9.2 R2017a
Statistics and Machine Learning Toolbox	Version 11.1 R2017a

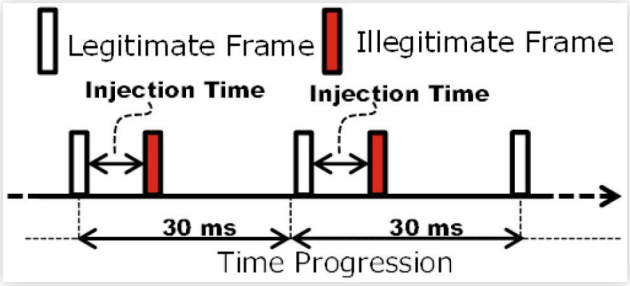
TABLE 5 Mini-batched SDEM parameters.

Mini-Batched SDEM parameters	Value
Discounting parameter η	0.7
Mini-batch size m	20
Parameter for stability estimation α	0

TABLE 6 Captured CAN traffic data for evaluation.

No	Driver	Duration (min)	Upper speed limit (km/h)	Course
1	A	63	15	Ground
2	B	31	15	Ground
3	C	30	15	Ground
4	B	31	20	Ground
5	B	30	20	Ground
6	B	30	35	Ground
7	B	31	35	Ground

FIGURE 10 Example of one illegitimate traffic dataset.



© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

TABLE 7 Processing time of the proposed method.

Evaluated function of the proposed method	Maximum processing time (μs)	
	Default	Speed
Anomaly detector Frame assessment (Equation 6)	3.3	3.2
Learner Mini-batched SDEM with the second Gaussian mixture model	15.0	13.2
Other time	0.2	0.1
Total processing time (120 MHz CPU)	18.5	16.5

TABLE 8 Captured CAN traffic data for evaluation.

Truths of the assessed frame	Assessment results of the anomaly detector	
	Positive	Negative
Positive	True positive (TP)	False negative (FN)
Negative	False positive (FP)	True negative (TN)

The requirement for the processing time for each frame was 13.7 μs, as shown in Table 3. The evaluation results of the processing time using the proposed method exceeded 2.8 μs (speed option) or 4.8 μs (default option) compared with the requirement when using a 120 MHz clock speed CPU. However, at the present time, the clock speed of popular CPUs for in-vehicle networks is becoming fast. Thus, this technological advancement could be addressed short the processing time of the proposed method. In fact, we can use a 240 MHz clock speed CPU for the same model shown in Table 7, in which case the processing time would be halved for the "Total processing time (120 MHz)." The processing time would be half of 16.5 μs (speed option) or half of 18.5 μs (default option). Both processing times would satisfy the requirement of the processing time for each frame.

Detection Performance

Terms for Detection Performance Before evaluating detection performance, we define terms that are used in this section. Table 8 shows the terms used to express the results of the anomaly detector. "Positive" indicates that the frame is illegitimate, "Negative" indicates that the frame is legitimate and the four abbreviations "TP," "FN," "TN," and "FP" indicate the frame assessment results.

We evaluate the detection performance using the following equations: the correct rate of the positive assessment results with respect to all positive frames is defined as.

$$\text{true positive rate}(\text{TPR}) = \frac{TP}{TP + FN}, \quad \text{Eq. (7)}$$

the incorrect rate of the negative assessment results with respect to all negative frames is defined as.

$$\text{false positive rate}(\text{FPR}) = \frac{FP}{TN + FP}, \quad \text{Eq. (8)}$$

the correct rate of the negative assessment results with respect to all negative frames is defined as.

$$\text{true negative rate}(\text{TNR}) = \frac{TN}{TN + FP}, \quad \text{Eq. (9)}$$

and the incorrect rate of the positive assessment results with respect to all positive frames is defined as.

$$\text{false negative rate}(\text{FNR}) = \frac{FN}{TP + FN}. \quad \text{Eq. (10)}$$

Overview To evaluate detection performance, we compared the proposed method with the conventional method. The conventional method was designed to make the anomaly detector for the prevention function that stops forwarding illegitimate frames in the gateway ECU. More detailed information is shown in the subsection "Conventional Method." In this regard, the concept of the conventional method is the same as that of the proposed method. However, both methods are completely different regarding the policy of the frame assessment. The proposed method aims to allow the vehicle runs continuously without stopping target service of the attacker; thus, this method intends to

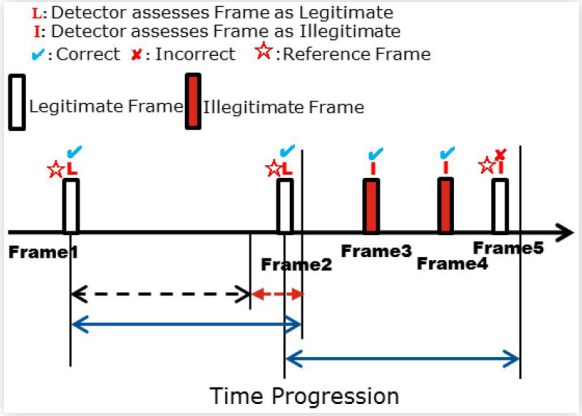
© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

achieve the high TPR and the high TPR under attack. By contrast, the conventional intends to achieve the high TPR, and fails to improve the TNR. And also, this method has the long delay to start the frame assessment to achieve high TPR.

First, we show the effect of the baseline method of the proposed method that calculates the reception cycle period and how much the tracking PDF of the proposed method improves the detection performance, as explained in the subsection "Improvement of the Proposed Method Detection Performance."

At the end of this section, we clarify the difference in detection performance between the proposed method and conventional method in the subsection "Detection Performance." Additionally, we clarify the difference in the delay before the frame assessment starts.

FIGURE 11 Conventional method: delayed-decision cycle detection.



Conventional Method Otsuka et al. proposed the "delayed-decision cycle detection" method that monitors the communication cycle period of periodic frames [12]. This method belongs to the frame pattern class. To detect illegitimate frames, this method waits for the frame assessment of all arrived frames until the expiration of the monitoring period. Then, the method counts the number of arrived frames. If this number is only one, then the method assesses the arrived frame as legitimate. If the number of arrived frames is greater than one, then this method assesses all arrived frames as illegitimate because it is difficult to identify the legitimate frame among illegitimate frames.

Figure 11 shows the more detailed behavior of this method. "M" indicates the monitoring period, which is determined by adding "T" to "B." "T" indicates the designed cycle, which is defined as the designed communication cycle period of the periodic frame. "B" indicates the margin. The designed communication cycle period fluctuates as a result of the CAN protocol; therefore, the monitoring period needs additional time as the margin. "L" and "I" indicate assessment results of the conventional method, where "L" denotes legitimate and "I" denotes illegitimate. The "Correct" symbol and "Incorrect" symbol are marked at the top of each frame to indicate each assessment result. The "Reference frame" symbol is marked at the top of each reference frame. This frame is used to measure the communication cycle period of each arrived frame, and the latest arrived frame in the previous monitoring period is used as the reference frame for the next monitoring period.

We show the detailed case that only a legitimate frame arrives. To assess Frame 2, Frame 1 is used as the reference frame. The assessment result is legitimate because there is only one frame in the monitoring period, which is Frame 2.

We also show the detailed case that two illegitimate frames and a legitimate frame arrive. To assess Frame 3, Frame 4, and Frame 5, Frame 2 is used as the reference frame. The assessment result of each frame is illegitimate because there are three frames in the monitoring period.

Improvement of the Proposed Method's Detection Performance First, we demonstrate the effect of the baseline method of the proposed method that calculates the reception cycle period in the subsection "Effect of the Baseline Method." Then, we demonstrate how much the tracking PDF of the proposed method improves the detection performance in the subsection "Effect of the Tracking PDF."

For these two evaluations, we compared the proposed method with two static methods: the static-baseline and the static-previous. Both methods use a static PDF with the Gaussian mixture model instead of the tracking PDF. This static PDF is learned using the expectation and maximization (EM) algorithm for all captured CAN traffic data shown in Table 6 [17]. These comparison methods differ in how they calculate the current reception cycle period. The static-baseline and the static-previous use the baseline method and the previous method to calculate the current reception cycle period, respectively.

© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

FIGURE 12 TPR and TNR of the proposed method, static-baseline, and static-previous method: (A) TPR (under attack), (B) TNR (under attack).

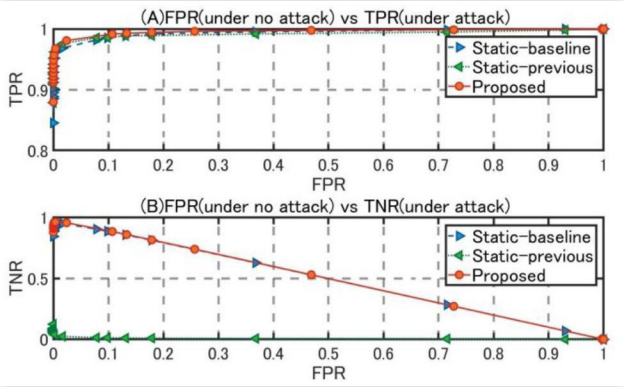


FIGURE 13 Enlarged area around FPR = 0 for (A) TPR (under attack), (B) TNR (under attack) as shown in Figure 12.

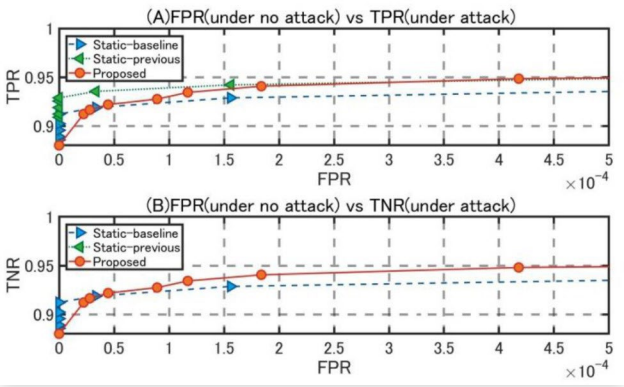
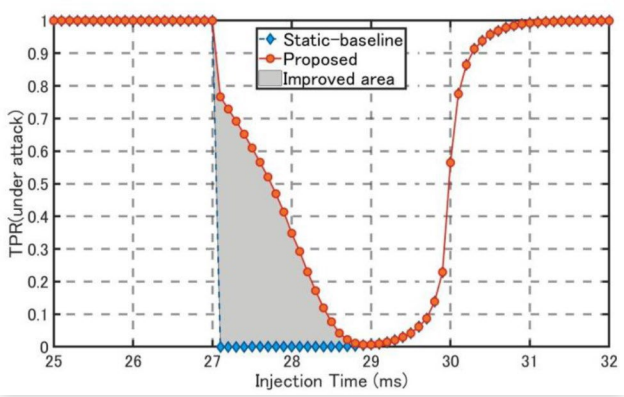


FIGURE 14 TPR improvement when the FPR was 1.5e-4.



In this evaluation, we measured the FPR of each method using all data that are under no attack shown in the subsection "CAN Traffic under No Attack." Additionally, we measured the TNR and TPR of each method using all data that are under attack in the subsection "CAN Traffic under Attack." We measured the performance while changing the threshold.

Effect of the Baseline Method. In this evaluation, we compared the static-baseline method with the static-previous. Figure 12(A) shows the TPR: this is the detection performance of the illegitimate frames. Figure 12(B) shows the TNR: this is the detection performance of legitimate frames. From Figure 12(B), the TNR of the static-baseline method is approximately 0.9, whereas the static-previous method is close to zero. From the Figure 12(A), the TPR of the static-baseline method and the static-previous method are approximately 0.9. Figure 13 shows an enlarged section of the area of the two subfigures around the FPR (under no attack) that is equal to zero in Figure 12. From Figure 13(A), the TPR of the static-previous method was 1% higher than that of the static-baseline method.

To summarize, the advantage of the baseline method is that it maintains a high TNR. The disadvantage of the baseline method is that the TPR is lower than it would be for the previous method.

Effect of the Tracking PDF. In this evaluation, we compared the proposed method with the conventional-baseline method.

From Figure 12(B), the proposed method maintains a high TNR. Additionally, from Figure 13(A), the TPR is caught up the static-previous when the FPR is greater than 1.5e-4. This is the effect of the tracking PDF on the proposed method.

To analyze where the proposed method improves the TPR, Figure 14 shows the TPR of the proposed and static-baseline methods when the FPR is 1.5e-4 is considered. This figure shows that the TPR of the proposed method improves in the gray area, which is located in the reception cycle period from 27 ms to 30 ms, which is close to the designed cycle period (30 ms) of the test data in subsection "CAN Traffic Data under Attack." To emphasize the improvement of the TPR of the proposed method, we demonstrate two results that changed the average time interval, as shown in Table 9. In this table, "Entire time interval" refers to the TPR of the average of all measured values, while the injection time of the spoofed frame was changed from 2.0 ms to 32.0 ms. "Around the designed reception cycle period" indicates that the injection time of the spoofed frame changed from 27.0 ms to 32.0 ms. From this result, the proposed method showed a 4.8% improvement near the designed cycle period.

To summarize, the tracking PDF of the proposed method improved the TPR around the designed reception cycle period. From Figure 14, TPR of both method was low where was close to the designed cycle period. This means it was difficult to distinguish illegitimate frames and legitimate frames for both methods in this area.

Detection Performance We compared the detection performance of the proposed method with the conventional

© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

method, as explained in the subsection "Comparing with the Conventional Method." Additionally, we evaluated the delay to start the frame assessment of the conventional method, as explained in the subsection "Delay to Start the Frame Assessment."

In this evaluation, we measured the FPR for each method using all data that were under no attack in subsection "CAN Traffic Under no Attack", and the TNR and TPR for each method using all data that were under attack in subsection "CAN Traffic under Attack." For the proposed method, we measured this performance while changing the threshold. For the conventional method, we measured the performance while changing the margin from 0 ms to 3.0 ms.

Comparing with the Conventional Method. Figure 15(A) shows the TPR: this is the detection performance of the illegitimate frames. Figure 15(B) shows the TNR: this is the detection performance of legitimate frames.

From Figure 15(A), the TPR of the proposed method was maintained at approximately 1.0 when the FPR increased from 0 to 1.0. The TPR of the conventional method decreased from 1.0 to approximately 0.5 when FPR increased from 0 to approximately 0.4. From Figure 15(B), the TNR of the proposed method was approximately 1.0 when the FPR was approximately zero. By contrast, for the conventional method, the TNR was approximately zero.

Figure 16 shows an enlarged section of the area of the two subfigures around the FPR that was equal to zero in Figure 15. From Figure 16(A), the TPR of the conventional method was approximately 1.0. The TPR of the proposed method was approximately 0.95 when the FPR was greater than 1.5e-4.

To summarize, the TNR of the proposed method was 0.95 when the FPR was greater than 1.5e-4. The proposed method improved this performance by 94% compared with the conventional method. By contrast, the TPR of the proposed method was approximately 0.94 when the FPR was greater than 1.5e-4. Additionally, the TPR of the conventional method was approximately 1.0 when the FPR was greater than 1.5e-4.The proposed method degraded this performance by 6% compared with the conventional method.

Delay to Start the Frame Assessment

The proposed method can start the frame assessment immediately when each frame arrives. By contrast, the conventional method needs to wait to start the frame assessment until the expiration of the monitoring period, as explained in the subsection "Conventional Method." In this section, we evaluate the maximum delay to start the frame assessment for the conventional method.

The monitoring period of the conventional method is composed of the designed cycle and margin. The designed cycle is determined by what the CAN-ID of the proposed method monitors and is 30 ms. However, the margin is a variable; thus, a suitable value needs to be determined. Therefore, in this section, first, we evaluate a suitable margin for the conventional method. Then, we evaluate the maximum delay to start the frame assessment.

TABLE 9 TPR improvement using the proposed method compared with not tracking the PDF method.

Time interval for average	TPR		
	Proposed method	Static-baseline	Improvement (%)
Entire time interval (2.0 ms-32.0 ms)	0.939	0.929	1.0
Around the designed reception cycle period (27.0 ms-32.0 ms)	0.628	0.580	4.8

FIGURE 15 TPR and TNR of the proposed method and conventional method: (A) TPR (under attack), (B) TNR (under attack).

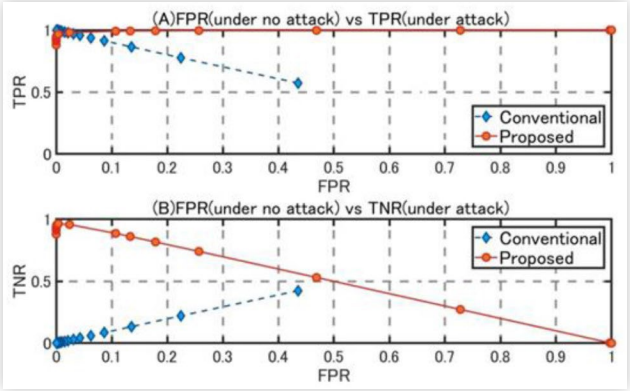
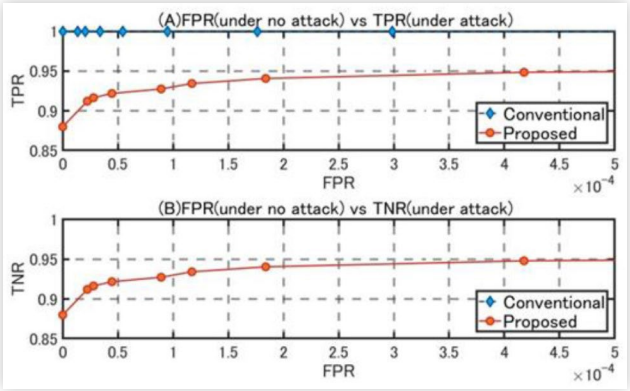


FIGURE 16 Enlarged area around FPR = 0 of (A) TPR (under attack), (B) TNR (under attack) as shown in Figure 15.



© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

FIGURE 17 Relationship between margin and (A) TPR (under attack), (B) FPR (under no attack).

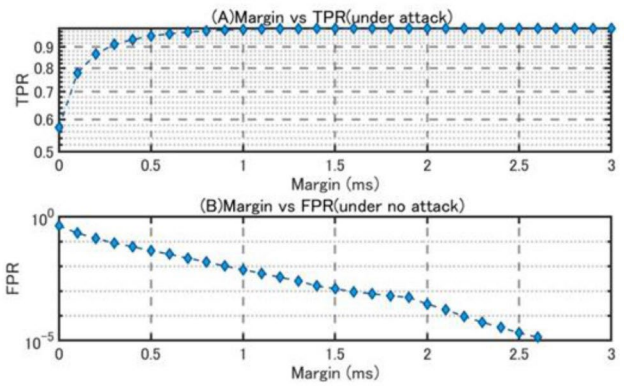
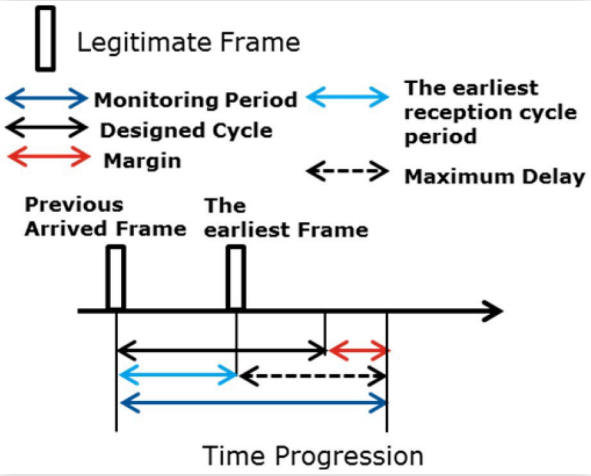


FIGURE 18 Relationship between the maximum delay and other parameters for the conventional method.



From Figure 17(A), the TPR decreased when the margin decreased from approximately 1.0. Additionally, from Figure 17(B), the FPR increased when the margin decreased. From these results, this method needs to sacrifice the TPR and FPR to improve the delay to start the frame assessment. To determine a suitable margin, we assumed that the allowable FPR was 1.5e-4, which is the same value used for the evaluation of the proposed method. In this case, from Figure 17(B), we obtained 2.1 ms as a suitable margin for the conventional method.

From Figure 18, the maximum delay can be determined from the designed cycle (30 ms), margin (2.1 ms), and earliest reception cycle period. To determine the earliest reception cycle period, we aggregated all captured data in the subsection "CAN Traffic Data under No Attack". Then, we determined that the earliest reception cycle period was 27.5 ms. Finally, the maximum delay to start the frame assessment of the conventional method was 4.6 ms.

To summarize, the proposed method does not to wait to start the frame assessment. To monitor the periodic frames the designed cycle period of which was 30 ms, the conventional method needed to wait for a maximum of 4.6 ms since the legitimate frame arrived.

Summary/Conclusions

To allow the easy deployment and maintenance of anomaly-based intrusion detection for the frame pattern class, we proposed a method that learns the behavior model of the monitored object online. For this online learning, the proposed method continuously learns a behavior model of each CAN-ID in the ECU. For the behavior model of the monitored object, the proposed method learns the histogram of reception cycle periods using the PDF. We called this proposed method the SDEC.

For the prevention function that allows the vehicle to run continuously when under attack without stopping the service targeted by the attacker, we designed the SDEC so that it was possible to recognize illegitimate frames and legitimate frames.

In the evaluation of the improvement of the proposed method's detection performance, to identify legitimate frames, the baseline method improved this performance by 94% against the previous method. For the detection performance of illegitimate frames, the baseline method degraded performance by 1% compared with the previous method. However, the tracking PDF of the proposed method improved this performance degradation that was 1% due to using the baseline method. The SDEC achieved to improve detection performances of illegitimate and legitimate frames by the baseline method and the tracking PDF.

In the evaluation of detection performance, we compared the SDEC to a conventional method that also aims to be used in the gateway ECU as a detector for a prevention function. According to the evaluation results, the SDEC could identify 94% of the legitimate frames under attack. The conventional method did not identify the legitimate frames. These results mean that the SDEC improved this performance by 94%. By contrast, the SDEC identified 94% of the illegitimate frames. The conventional method identified 100% of the illegitimate frames. These results mean that the SDEC degraded performance by 6% compared with the conventional method. We also evaluated the delay

© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

to start the frame assessment and found that the conventional method had to wait for a maximum of 4.6 ms and the SDEC did not need to wait to start the frame assessment.

The SDEC worked better than the conventional method as an IDS in the gateway ECU in view of the identification performance of the legitimate frames and the delay to start the frame assessment. However, the detection performance of the illegitimate frame was degraded by 6% compared with the conventional method. This degradation should be improved.

The degradation of the detection performance of the illegitimate frames was caused where around the designed cycle period. In this area, it was difficult to distinguish the illegitimate frames and the legitimate frame by using the communication cycle period. To improve the detection performance of illegitimate frames without increasing the delay to start the frame assessment, we believe that the anomaly detector should monitor the feature of the frame different from the frame pattern class. For this purpose, we believe that the context pattern class could improve this degradation. Therefore, to improve this degradation, we would like to expand the SDEC so that it can monitor the reception cycle period and context data.

Acknowledgements

We thank Maxine Garcia, PhD, from Edanz Group (www.edanzediting.com/ac) for editing a draft of this manuscript.

Definitions/Abbreviations

ECU - Electronic Control Unit

CAN - Control Area Network

SDEC - Statistic Detector

EM - Expectation and Maximization

SDEM - Sequentially Discounting Expectation and Maximization

PDF - Probability Density Function

IDS - Intrusion Detection System

References

1. Charette, R.N., "This Car Runs on Code," http://spectrum.ieee.org/transportation/systems/this-car-runs-on-code, accessed July 2017.
2. Checkoway, S., McCoy, D., Kantor, B., Anderson, D. et al., "Comprehensive Experimental Analyses of Automotive Attack Surfaces," *Proceedings of the 20th USENIX Conference on Security*, 2011.
3. Miller, C. and Valasek, C., "Adventures in Automotive Networks and Control Units," presented at *DEF CON 21*, Aug. 2013.
4. Miller, C. and Valasek, C., "Remote Exploitation of an Unaltered Passenger Vehicle," presented at *DEF CON 23*, Aug. 2015.
5. Miller, C. and Valasek, C., "Advanced Can Injection Techniques for Vehicle Networks," presented at *DEF CON 24*, Aug. 2016.
6. Scarfone, K.A. and Mell, P.M., "Guide to Intrusion Detection and Prevention Systems (IDPS)," Special Publication (NIST SP) - 800-94, Feb. 2007.
7. International Organization for Standardization, "Road Vehicles- Controller Area Network (CAN) - Part 1: Data Link Layer and Physical Signaling," ISO11898-1, Rev. 2003.
8. International Organization for Standardization, "Road Vehicles- Controller Area Network (CAN) - Part 1: Data Link Layer and Physical Signaling," ISO11898-1, Rev. 2015.

© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.

9. Koscher, K., Czeskis, A., Roesner, F., Patel, S. et al., "Experimental Security Analysis of a Modern Automobile," *2010 IEEE Symposium on Security and Privacy*, 2010, doi:10.1109/SP.2010.34.
10. Chandola, V., Banerjee, A., and Kumar, V., "Anomaly Detection: A Survey," *ACM Computing Surveys* 41(3), 2009, doi:10.1145/1541880.1541882.
11. Larson, U.E., Nilsson, D.K., and Jonsson, E., "An Approach to Specification-Based Attack Detection for In-Vehicle Networks," *2008 IEEE Intelligent Vehicle Symposium*, 2008, doi:10.1109/IVS.2008.4621263.
12. Otsuka, S., Ishigooka, T., Oishi, Y., and Sasazawa, K., "CAN Security: Cost-Effective Intrusion Detection for Real-Time Control Systems," SAE Technical Paper 2014-01-0340, 2014, doi:10.4271/2014-01-0340.
13. Müter, M. and Asaj, N., "Entropy-Based Anomaly Detection for In-Vehicle Networks," *2011 IEEE Intelligent Vehicle Symposium (IV)*, 2011, doi:10.1109/IVS.2011.5940552.
14. Markovitz, M. and Wool, A., "Field Classification, Modeling and Anomaly Detection in Unknown CAN Bus Networks," presented at *the 13th escar Europe Conference*, Nov. 11-12, 2015.
15. Wasicek, A., Pesé, M., Weimerskirch, A., Burakova, Y. et al., "Context-Aware Intrusion Detection in Automotive Control System," presented at *the 5th escar USA Conference*, Detroit Metropolitan, MI, June 21-22, 2017.
16. Yamanishi, K., Takeuchi, J., and Williams, G., "On-Line Unsupervised Outlier Detection Using Finite Mixtures with Discounting Learning Algorithms," *Data Mining and Knowledge Discovery Journal* 8(3):275-300, 2004.
17. Dempster, A.P., Laird, N.M., and Rubin, D.B., "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of Royal Statistical Society. Series B (Methodological)* 39(1):1-38, 1977.

© 2018 SUMITOMO ELECTRIC INDUSTRIES, LTD.; Published by SAE International. This Open Access article is published under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0/, which permits use, distribution, and reproduction in any medium, provided that the original author(s) and the source are credited. Positions and opinions advanced in this work are those of the author(s) and not necessarily those of SAE International. Responsibility for the content of the work lies solely with the author(s).