IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 5, 2019, accepted August 20, 2019, date of publication August 26, 2019, date of current version September 19, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2937576

An Intelligent Secured Framework for Cyberattack Detection in Electric Vehicles' CAN Bus Using Machine Learning

OMID AVATEFIPOUR[1], AMEENA SAAD AL-SUMAITI[2], (Member, IEEE), AHMED M. EL-SHERBEENY[3], EMAD MAHROUS AWWAD[4], MOHAMMED A. ELMELIGY[5], MOHAMED A. MOHAMED[6], (Member, IEEE), AND HAFIZ MALIK[1], (Senior Member, IEEE)

[1]Department of Electrical and Computer Engineering, University of Michigan–Dearborn, Dearborn, MI 48128, USA
[2]Advanced Power and Energy Center, Electrical Engineering and Computer Science, Khalifa University, Abu Dhabi 127788, United Arab Emirates
[3]Industrial Engineering Department, College of Engineering, King Saud University, Riyadh 11421, Saudi Arabia
[4]Electrical Engineering Department, College of Engineering, King Saud University, Riyadh 11421, Saudi Arabia
[5]Advanced Manufacturing Institute, King Saud University, Riyadh 11421, Saudi Arabia
[6]Electrical Engineering Department, Faculty of Engineering, Minia University, Minia 61519, Egypt

Corresponding authors: Ahmed M. El-Sherbeeny (aelsherbeeny@ksu.edu.sa) and Mohamed A. Mohamed (dr.mohamed.abdelaziz@mu.edu.eg)

The authors extend their appreciation to the Deanship of Scientific Research at King Saud University for funding this work through research group number RG-1440-047.

ABSTRACT Electric Vehicles' Controller Area Network (CAN) bus serves as a legacy protocol for in-vehicle network communication. Simplicity, robustness, and suitability for real-time systems are the salient features of CAN bus. Unfortunately, the CAN bus protocol is vulnerable to various cyberattacks due to the lack of a message authentication mechanism in the protocol itself, paving the way for attackers to penetrate the network. This paper proposes a new effective anomaly detection model based on a modified one-class support vector machine in the CAN traffic. The proposed model makes use of an improved algorithm, known as the modified bat algorithm, to find the most accurate structure in the offline training. To evaluate the effectiveness of the proposed method, CAN traffic is logged from an unmodified licensed electric vehicle in normal operation to generate a dataset for each message ID and a corresponding occurrence frequency without any attacks. In addition, to measure the performance and superiority of the proposed method compared to the other two famous CAN bus anomaly detection algorithms such as Isolation Forest and classical one-class support vector machine, we provided Receiver Operating Characteristic (ROC) for each method to quantify the correctly classified windows in the test sets containing attacks. Experimental results indicate that the proposed method achieved the highest rate of True Positive Rate (TPR) and lowest False Positive Rate (FPR) for anomaly detection compared to the other two algorithms. Moreover, in order to show that the proposed method can be applied to other datasets, we used two recent popular public datasets in the scope of CAN bus traffic anomaly detection. Benchmarking with more CAN bus traffic datasets proves the independency of the proposed method from the meaning of each message ID and data field that make the model adaptable with different CAN datasets.

INDEX TERMS Electric vehicles, controller area network (CAN Bus), anomaly detection, one-class support vector machine, optimization algorithm.

I. INTRODUCTION

Modern electric vehicles are composed of many hardware modules, known as Electronic Control Units (ECUs), which are controlled by sophisticated software components. ECUs read data measured by a range of sensors and perform relevant processing for various purposes, such as pedestrian detection, path planning, auto-parking, and collision avoidance. They also control the actuators in a vehicle [1]. The values of the sensors and actuators are transmitted over the in-vehicle network protocol to other ECUs, leading to the creation of a highly complex network of hardware and software

The associate editor coordinating the review of this article and approving it for publication was Hao Luo.

sub-modules. There are several in-vehicle network protocols, namely CAN, CAN Flexible Data-Rate (CAN FD), Local Interconnect Network (LIN), FlexRay, and Media Oriented Systems Transport (MOST). Among all of the aforementioned protocols, CAN Bus is the most well-known and widely used protocol in the automotive industry and is considered the de-facto standard for vehicular networks. In addition, the CAN bus has been applied to more than just automotive networks, finding a range of applications in other industries, such as aerospace, agriculture, medical devices, and even in some home and commercial appliances [1].

Although there are other protocols with more security features available, e.g. Ethernet, they cannot entirely replace the CAN bus for in-vehicle network communication due to the following reasons: 1) The CAN bus is designed to be perfectly applicable for hard real-time environments and guarantees deterministic communication with minimal time latency. 2) In the CAN bus protocol, there is a method of prioritization where lower priority messages do not interfere with higher priority messages. For instance, a message transferring more critical function, such as engine control or airbag control, has more priority than a message for door or climate control. 3) The CAN bus protocol is used in all modern vehicles as the backbone of in-vehicle network communication; replacing this protocol entirely with another protocols requires re-designing the whole vehicle network architecture and a tremendous amount of changes in vehicle software, which runs based on the CAN protocol. Therefore, other protocols will not entirely replace the role and application of the CAN bus but rather augment the CAN bus.

During the invention of the CAN bus protocol by Robert Bosch GmbH [2], vehicles were considered an isolated environment that did not have communication with the outside environment. Therefore, by design, the CAN bus suffers from a lack of authentication and security features, including data encryption and message authentication. This paves the way for adversaries to penetrate a network and launch malicious activities more easily than when other protocols, like the Transmission Control Protocol/Internet Protocol (TCP/IP), are used. For instance, given the lack of an effective message authentication method, attackers can compromise an ECU by injecting malicious messages and replay attacks. Fortunately, with the advancement of data-mining techniques, this type of attack has been addressed by researchers [3]–[7] in such a way that any anomalous communication traffic activities can be detected and ignored.

Recently, modern vehicles are not only considered a closed-loop system, but they also have several types of communication with the outside world. Attackers can penetrate and inject malicious messages into the CAN traffic via different internal and external interfaces, such as through physical access to the OBD-II port (an on-board diagnostics system that monitors emissions, mileage, speed, and other data about the car) in the vehicle, short-range wireless access, e.g. Bluetooth, long-range wireless access, e.g. Wi-Fi, a telematics control unit (TCU), and cellular radio. For instance, with the embrace of over-the-air (OTA) updates, ECUs can be reprogrammed remotely, which may provide more comfort and convenience to the vehicle owner and dealerships. However, these interfaces have also introduced more remote attack surfaces that can help attackers to compromise an ECU using a malicious message.

In general, anomaly detection refers to the problem of finding patterns in a dataset which do not follow the expected defined behavior [7]. Anomaly detection is considered an important topic, which has been studied within various research domains. In the CAN bus protocol, anomaly detection is the process of monitoring communication traffic among ECUs and identifying any abnormal behavior in traffic using machine learning (ML) algorithms. Nowadays, ML techniques have gained the attention of researchers in the cybersecurity community. One popular usage of ML techniques is designing intrusion detection systems (IDS) for a wide range of applications, namely outlier detection, novelty detection models, and antivirus/malware detection [8]–[10]. In particular, anomaly detection in automotive networks has also attracted the attention of researchers in this area, which is elaborated on in the Related Work section.

Automakers are aiming to develop fully-autonomous vehicles in the near future, a consequence of which is the introduction of more attack surfaces. Since data is not encrypted in the CAN bus protocol, attackers can launch replay attacks and inject malicious messages into a network (i.e. perform an intrusion-based attack) by performing a reverse-engineering procedure to interpret each CAN packet. To achieve this goal, attackers should send messages with very high frequency to beat the arbitration mechanism (explained further in section III) used on all messages on the bus. That said, this message injection procedure will create some anomalous behavior in the communication traffic, which can be detected by developing an anomaly detection method in the CAN bus protocol. In the world of desktop computers, the risk of attack and the securing of communication protocols have already received a huge amount of attention in recent years; however, considering the same standard measures in order to provide strong protection for vehicular networks is impractical and almost infeasible due to the complexities of embedded systems and the reality of a real-time environment with limited resources in terms of its processing unit and memory. Therefore, a different approach is required to detect anomalous behavior in a vehicular network. In this paper, we propose a prediction model that can detect anomalous traffic in a vehicular network protocol (here we consider the CAN bus protocol). The application of our anomaly detection method for vehicular network traffic is illustrated in Fig 1.

II. RELATED WORK

Anomaly detection in automotive network communication has been addressed by a growing number of researchers, which reflects the fact that this topic has been considered as one of the most critical issues to governments, industry, and academia. Different studies show the CAN bus's

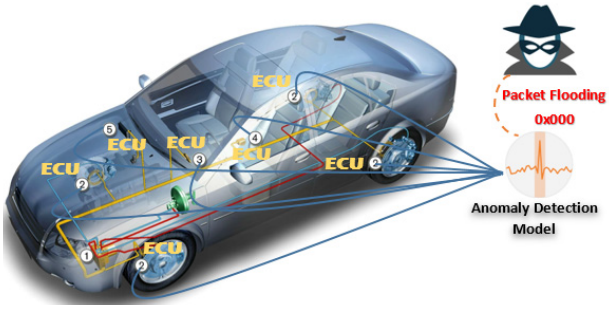

FIGURE 1. Application of anomaly detection model for vehicular network traffic.

vulnerabilities and weaknesses in terms of security features [11]. A number of these studies are presented in this section.

Wang and Sawhney [12] proposed a security framework for vehicular systems (VeCure), which can fundamentally solve the message authentication issue of the CAN bus. Their proposed method creates 2000 additional clock cycles compared to a system without this message authentication technique (equivalent to 50 microseconds running on a 40 MHz processor).

The authors of [13] proposed an intrusion detection system by recording the traffic of in-vehicle network communication (CAN bus). The basis of their proposed method is that there are specific ranges of randomness happening at the communication level of the in-vehicle network, which can be utilized by an information-theoretic measure, e.g. entropy.

Kang *et al.* [14] proposed an Intrusion Detection System (IDS) using Deep Neural Network (DNN). Probability-based feature vectors, which are extracted from the CAN bus messages, are utilized to train the DNN parameters. Statistical properties of each class will discriminate between normal and attack message in a given CAN bus packet.

Theissler [15] introduced an anomaly detection approach capable of detecting faults of known and unknown types without requiring the setting of expert parameters. An ensemble classifier, which consists of a two-class and a one-class classifier, is employed for univariate and multivariate anomalies.

Narayanan *et al.* [16] presented a Hidden Markov model (HMM), a stochastic model that follows the Markov property, to detect anomalous states of real data collected from a vehicle during operation. The underlying assumption when using an HMM is that the movement of the vehicle is a sequence of events in which each one is dependent on the previous event, like in Markov's processes.

Cho *et al.* [17] proposed a clock skew-based framework for ECU fingerprinting and used it for the development of Clock-based Intrusion Detection System (CIDS). The proposed clock-based fingerprinting method [17] exploited a clock characteristic that exists in all digital systems: "a tiny timing error known as clock skew." The clock skew identification exploits the uniqueness of clock skew and clock offset, which is used to identify a given ECU based on clock attributes of the sending ECU.

The authors of [18], however, found that there are potential vulnerabilities present in the CIDS by design, including parameter dependence on message periodicity and non-linearity of the clock skewness. They proposed an attack, called a clock-spoofing attack, which can easily be used to bypass the CIDS by replicating the clock parameters, hence challenging their assumed uniqueness.

Taylor *et al.* [19] proposed an intrusion detection system based on a long short-term memory (LSTM) recurrent neural network for CAN bus traffic. The neural network is trained to predict the next packet data values, and its errors are utilized as a signal for anomaly detection in a sequence.

Other solutions to CAN bus anomaly detection are proposed by researchers in [20]–[22] which are based on regression learning to estimate certain parameters by using correlated/redundant data among a group of sensors for a given in-vehicle network attack. For example, by pressing the accelerator pedal, the engine pump rotates faster; hence, the RPM value and vehicle speed will increase. When an attacker tries to manipulate these values, the existing balance of either the negative or positive correlation across that sensor group may deviate from the valid range.

Another approach automatically classifies fields in CAN messages. The authors of [5] developed a greedy algorithm to split the messages into fields and measure valid ranges based on previous data. The authors also designed a semantically-aware anomaly detection system for CAN bus traffic, but it was not evaluated with any attack scenarios.

Marchetti and Stabili [23] proposed an anomaly detection algorithm to identify anomalies in a sequence of CAN messages by using a transition matrix defined based on a reiterative CAN ID pattern sequence. The authors of [24] introduced a signature-based method to model both legitimate ECUs and the behavior of known attack signatures. Their method could detect intrusions in real-time, but it sometimes failed to detect an attack if it missed the first packets of the attack broadcast.

The authors of [25]–[27] presented an analysis of CAN broadcasts and subsequent testing of statistical methods to detect timing changes in the CAN traffic that were indicative of some of the predicted attacks.

The authors of [28] proposed a method for anomaly detection based on packets' time statistics analysis and a one-class support vector machine (OCSVM) in CAN traffic. Their method uses a flow-based anomaly detection approach, including a packet identifier, the total number of packets, and the time of occurrence. To the best of our knowledge, this is the only research work that considers OCSVM for anomaly detection in CAN traffic; however, this study lacks deep analysis of the kernel function type (linear or nonlinear) and optimal training of an OCSVM facing highly nonlinear data.

Based on the above discussions, this paper aims to propose a novel and effective anomaly detection model to avoid

cyberattacks on a vehicle's CAN bus. The proposed model is constructed based on a modified one-class support vector machine (OCSVM) to provide the highest security and accuracy. In a one-class SVM, the support vector model is trained on data that has only one class, which is referred to as the ''normal'' class. It emphasizes the properties of normal cases and from these properties can predict which examples are unlike the normal examples [29]. This is extremely practical for anomaly detection because the scarcity of training examples is what defines anomalies. Due to the high complexity and nonlinearity of the CAN bus datasets, a new meta-heuristic optimization algorithm, called Modified Bat Algorithm (MBA), is proposed to reduce the false positive rate and improve the overall hit rate of anomalous message detection. The high accuracy and satisfactory performance of the proposed model is examined using experimental data gathered from an unmodified licensed vehicle. Furthermore, in order to prove the proposed method's independence from a specific car model, we performed the proposed method with two other famous datasets in the area of CAN bus anomaly detection. To the best of our knowledge, this is the first time that the proposed MBA optimization algorithm has been applied in an OCSVM as an anomaly detection model for a CAN bus.

The contributions of this work are summarized as follows:

1. Proposing an intelligent anomaly detection model based on advanced machine learning for reinforcing the in-vehicle network communication CAN bus protocol.
2. Developing a new modified one-class support vector machine based on bat algorithm. The proposed bat algorithm would adjust the anomaly detection model parameters optimally for maximizing the efficiency and performance of the model against cyberattacks.
3. Introducing an effective two-stage modification approach for bat algorithm to increase population diversity when avoiding the premature convergence. The proposed modification method is constructed based on the crossover and mutation operators borrowed from genetic algorithm and will help the algorithm to look for the optimal solution in the entire search space.

The rest of this paper is organized as follows: 1) section III explains the CAN bus protocol and its requirements. Section IV explains the proposed improved anomaly detection model based on OCSVM and MBA. Section V provides the simulation results using the experimental dataset. Finally, the main conclusions are provided in Section VI.

III. CAN BUS PROTOCOL – AN OVERVIEW

To meet real-time systems' deadline requirements, each message in the CAN bus protocol has been assigned a unique identifier (ID) frame, which is utilized to define the message priority and is also used by every ECU in the car to identify whether the incoming message should be processed or ignored [2]. The lower the message identification value, the higher priority it has to gain bus access. This prioritization feature has solved the bus access conflict such that if two nodes want to send data simultaneously, whichever ECU has a lower ID value will publish its message first, due to its higher priority. This technique is also known as message arbitration [2]. Fig. 2 depicts a situation in which three nodes (first node: 11001011111 in binary, second node: 110011111111 in binary, and third node: 110010110010 in binary) try to transmit a message simultaneously. In order to prevent bus collision, the node with the lowest ID (in this case the third node) will transmit its information. Attackers can exploit this feature by sending a malicious message with the lowest ID at a very high frequency to create a condition in which a malicious message always wins the arbitration and does not allow other messages to be transmitted (Denial-of-Service attack). In the proposed method, this type of attack can be detected as the system learns the normal traffic behavior on the bus, allowing abnormal traffic behavior (e.g. sending the same message with high frequency) to be detected. The message data payload holds different values generated by multiple sensors and is managed by a particular ECU and encoded based on the database container (DBC) file specification. A DBC is a database file with a vehicle manufacturer proprietary format that holds the specifications of all the ECUs, CAN messages, signals, message IDs, message frequency, and data payload for a particular vehicle configuration [30].

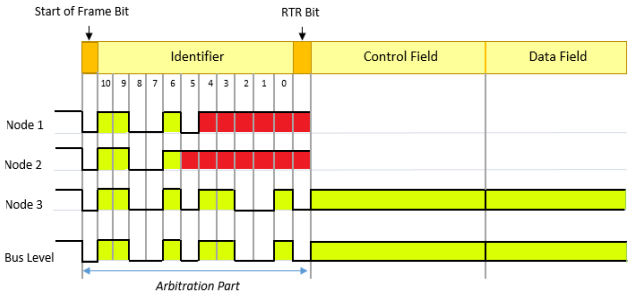

FIGURE 2. CAN bus arbitration technique.

IV. PROPOSED ANOMALY DETECTION MODEL AND FORMULATION

A. PROPOSED CAN BUS ANOMALY DETECTION METHOD

The underlying idea behind the proposed method is to establish a model based on normal CAN bus traffic, which contains recurring patterns in the message IDs that are transmitted. From the analysis of several traces taken from a car, some recurring message ID patterns have been identified from the logged traces, which means every message ID is followed by a particular recurring message ID subset. Hence, we developed a model to identify these patterns in normal traffic and any deviation from them can be considered as malicious activities, which the proposed method can detect as anomalous behavior. The proposed method consists of two main phases: the training phase and the testing and/or evaluation phase. During the training phase, the normal behavior of the CAN bus traffic is logged from an unmodified licensed

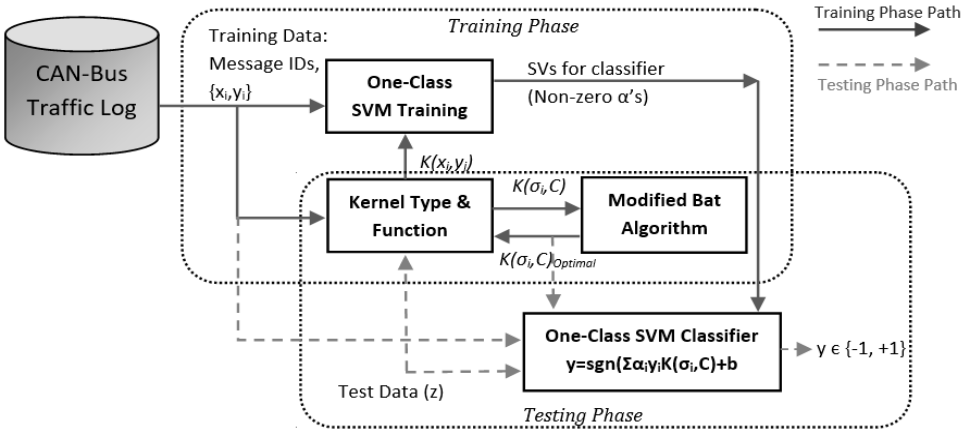

FIGURE 3. Block diagram of the proposed CAN bus anomaly detection method.

vehicle under normal operation to generate all the possible transitions between consecutive IDs without any attacks. Then, the model developed in the training phase can be employed as a reference to detect anomalous behaviors in the CAN traffic launched by an attacker.

A conceptual illustration of the proposed anomaly detection model is provided in Fig. 3. According to this figure, there are two paths for the training and testing phases (solid and dotted arrows, respectively). During the training phase, the CAN bus traffic is captured and each CAN message ID is extracted from the traffic and imported into the one-class SVM training algorithm. Within the one-class SVM training phase, the kernel type and function should be determined and their parameters (σ and C) should be optimally tuned (this process will be explained in the next section). To this end, a modified bat algorithm (flowchart is presented in Fig. 4) is applied as a meta-heuristic optimization algorithm to tune these parameters and feed them into the one-class SVM training algorithm to reach a better more matching hyperplane, as well as the optimal support vectors. Having completed the training phase, any CAN message in the traffic log can be classified as containing an abnormality or not by the proposed model.

It should be noted that although there is some literature that has considered different features, the authors have considered all possible features, including the sequence of frames, time, and frequency of occurrence, in the initial analysis. Through an appropriate feature selection procedure, it was seen that only frequency and frame ID suffice for a proper and reliable anomaly detection model. In other words, considering other features did not add any improvement to the classification model and only increased the anomaly detection model's complexity, simultaneously increasing the risk of over-complexity issues. In this study, a fuzzy-based feature selection method [35] is applied to select only the most informative features for developing a powerful and appropriate anomaly detection model in our case study.

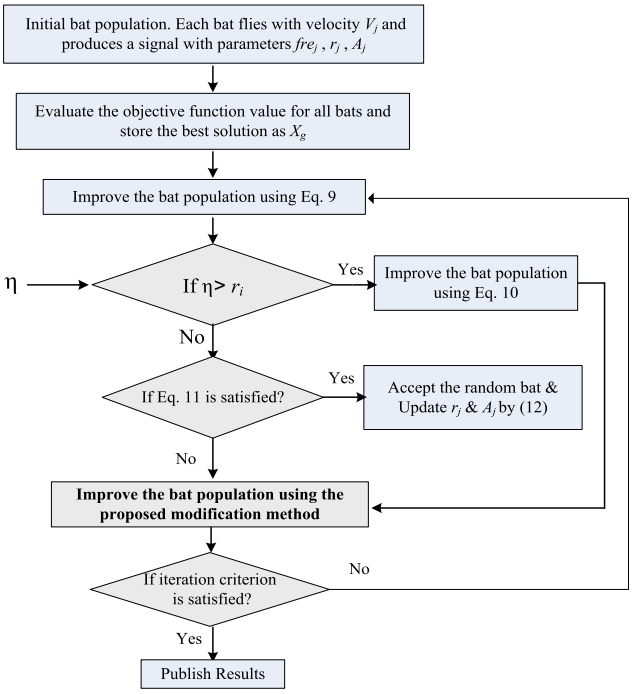

FIGURE 4. Modified bat algorithm flowchart.

B. ONE-CLASS SUPPORT VECTOR MACHINE (OCSVM)

One Class Support Vector Machine (OCSVM) is a classification algorithm that estimates the minimal subsets in an input space that contain a predefined fraction of the data. Consider a given dataset, which is designated as the normal dataset. OCSVM solves (1) to make the optimal classification as follows [29]:

$$\begin{aligned} &\min_{w,\rho} \frac{1}{2} \| w \|^2 + \frac{1}{\upsilon N} \sum_{i=1}^{N} \xi_i - \rho \\ &s.t.\ w.\Phi(x_i) \geq \rho - \xi_i \\ &\xi_i \geq 0 \end{aligned} \tag{1}$$

where N is the number of data points and $\xi = [\xi_1, \ldots, \xi_n]$ is the set of slack variables for the data points that allows a given data point to be located outside of the decision boundary. Here,$\upsilon \in (0, 1]$ is a trade-off parameter that represents an upper bound on the fraction of outliers and a lower bound on the fraction of support vectors. The parameters ρ and w define the decision boundary. Decision boundary function $f(x)$ is defined in (2), where x is a target and f returns $+1$ when a given data point falls within the normal CAN traffic; otherwise, $f(x)$ returns -1 for abnormality [29].

$$f(x) = w \cdot \Phi(x) - \rho \tag{2}$$

where $x \in R^M$ and Φ is a feature map. The inner product $\Phi(x_i) \cdot \Phi(x_z)$ is considered as the kernel function, which is represented by K, e.g. $K(x_i, x_z) = \Phi(x_i) \cdot \Phi(x_z)$. This paper considers a radial basis function (rbf) kernel as $K(x_i, x_z) = e^{-\| x_i - x_z \|^2 / 2\sigma^2 + C}$. Here C is a constant and σ is the width of radial basis function (rbf). When σ and C are selected properly, the rbf kernel can approximate the most suitable kernel function. Hence, these two parameters play a major role in the performance of the kernel and should be chosen carefully. To this end, we applied the bat algorithm with a proposed two-step modification, which resulted in a powerful meta-heuristic optimization algorithm, to find the optimal values for σ and C. In order to solve the optimization problem, a Lagrange equation is formulated as follows (Eq. 1):

$$L(w, \xi, \rho, \alpha, \beta) = \frac{1}{2} \| w \|^2 + \frac{1}{\upsilon^N} \sum_{i=1}^{N} \xi_i - \rho - \sum_{i=1}^{N} \alpha_i (w.\Phi(x_i) - \rho + \xi_i) - \sum_{i=1}^{N} \beta_i \xi_i \tag{3}$$

The partial derivatives for the above equation w.r.t. w, ξ, and ρ are set to zero. Hence, w and α can be defined as follows:

$$w = \sum_{i=1}^{N} \alpha_i \Phi(x_i) \tag{4}$$

$$\alpha_i = \frac{1}{\upsilon^N} - \beta_i \quad \sum_{i=1}^{N} \alpha_i = 1 \tag{5}$$

Having substituted (4) and (5) into the Lagrangian equation in (3), its dual form can be defined as:

$$\min_{\alpha} \alpha^T H \alpha \quad s.t.\ 0 \le \alpha_i \le \frac{1}{\upsilon^N} \quad \sum_{i=1}^{N} \alpha_i = 1 \tag{6}$$

where $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_N)$ is the vector form of the Lagrange multipliers of the constraints and H is the kernel matrix for the training set, $H(i, z) = K(x_i, x_z)$. Any points that have α_i greater than zero are called support vectors. H as the kernel matrix is as follows:

$$H_{ij} = K(x_i, x_j) = \Phi(x_i).\Phi(x_j) \tag{7}$$

After solving the optimization problem in (6) to attain α, ρ can be defined as:

$$\rho = \frac{1}{n_s} \sum_{i=1}^{n_s} \sum_{j=1}^{N} \alpha_j K(x_i, x_j) \alpha \tag{8}$$

where n_s is expressed as the number of support vectors that fulfill the criteria of $\xi_i = 0$ and where $0 < \alpha_i < 1/\upsilon^N$.

C. MODIFIED BAT ALGORITHM (MBA)

As explained in the previous section, the one-class SVM is a powerful method for CAN bus traffic anomaly detection. Nevertheless, the performance of this model depends on its adjusting parameters, including the kernel function, and setting values σ and C. In order to adjust these parameters globally, this paper proposes making use of an effective optimization algorithm based on the modified bat algorithm (MBA).

Bat Algorithm (BA) is a meta-heuristic optimization algorithm inspired by the echolocation process used by bats for detecting prey. Each bat X_j sends a loud signal into the air with the emission rate, frequency, and loudness of r_j^k, f_j, and A_j^k, respectively. By listening to the echolocation sound, the bat can update its velocity and position relative to its food (prey). Similar to other meta-heuristic algorithms, an initial bat population is generated. Through the echolocation process, the bat with the best position (most optimal fitness function) is stored as X_g and the rest of population is updated using the following equation:

$$\begin{aligned} V_{j,k+1} &= V_{j,k} + f_j \left(X_g - X_{j,k} \right) \quad \forall j \in \Omega^{bat} \\ X_{j,k+1} &= X_{j,k} + V_{j,k+1} \quad \forall j \in \Omega^{bat} \\ f_j &= f_{\min} + \theta_1 \left(f_{\max} - f_{\min} \right) \quad \forall j \in \Omega^{bat} \end{aligned} \tag{9}$$

where $V_{j,k+1}$ is the velocity of j^{th} bat in $(k+1)^{th}$ iteration, Ω^{bat} is the set of bats, $f_{\min/\max}$ is the min/max values of the bat signal frequency, and θ_1 is a random value in the range (0,1]. The mechanism behind BA represents a global search in the problem search space. On top of this global search, BA is equipped with local search capability, looking for the optimal solution in the neighborhood of each bat.

To this end, a random value η is generated in the range (0,1]. If η is bigger than the pulse emission rate$r_{m,k}$, the bat position is updated as follows:

$$X_{j,k+1} = X_{j,k} + \varepsilon A_{j,k} \tag{10}$$

where $\varepsilon \in [-1, 1]$. In the case where η is smaller than r_n, a new solution X_j^{new} is generated randomly. The new solution is considered subject to the below two criteria:

$$[\eta < A_j] \& [f(X_j^{new}) < f(X_{best})] \tag{11}$$

The loudness and rate of each bat signal is updated after each iteration, as follows:

$$\begin{aligned} A_{j,k+1} &= \lambda A_{j,k} \\ r_{j,k+1} &= r_j^0 \left[1 - \exp\left(-\gamma k\right)\right] \end{aligned} \tag{12}$$

where λ and γ are two constant parameters. In each iteration, the above steps are repeated until the algorithm converges.

While BA has shown great performance in the face of non-linear, non-convex and multi-modal optimization problems, in some situations it may find itself trapped in local optima or face premature convergence. In the literature, there are some modified versions of BA proposed. In [34], the authors proposed a special modification method for BA to provide a balance between the exploration and exploitation capabilities of the algorithm. From a technical point of view, while the BA's exploration feature helps produce a better global search, its exploitation feature offers better focus on local searches. In this case, the authors tried to improve the exploration mechanism of the algorithm by modifying the equation of the pulse emission rate and the loudness of the bats.

On the other hand, the modification method proposed in this paper is constructed based on two quite different and powerful approaches: 1) a new modification method based on the mutation and crossover operators to increase the bat population diversity. This not only avoids possible premature convergence but also will absolutely improve the global search ability of the BA. 2) A new math-based modification method to increase the convergence rate of the BA. The best bat in each iteration tries to improve the positions of other bats via (16). This is made possible by first evaluating the mean of the bat population and then trying to improve each bat's position according to its distance from the best bat. These two newly introduced modification methods are quite powerful and compatible with the high nonlinearity existing in the CAN bus dataset in this paper. We explain them in more detail below.

With regard to the first modification, in each iteration k and for each bat j, three dissimilar bats z_1, z_2 and z_3 are chosen from the population, such that $z_1 \neq z_2 \neq z_3 \neq j$. Using these random solutions, a new mutated bat is generated as follows:

$$\begin{aligned} X_{mut} &= X_{z1,k} + \theta_1 \times (X_{z2,k} - X_{z3,k}) \\ X_{mut} &= [x_{mut,1}, x_{mut,2}, \ldots, x_{mut,N}] \end{aligned} \tag{13}$$

where N equals the number of control variables and $x_{mut,v}$ represents the v^{th} element in bat vector X_{mut}. Using (11) and the crossover operator, two new test bats are generated as follows:

$$x_{j1,v}^{test1} = \begin{cases} x_{mut,v} & \text{if } \theta_1 \leq \theta_2 \\ x_{g,v} & otherwise \end{cases} \tag{14}$$

$$X_{j2}^{test2} = \theta_3 \times X_{mut} + \theta_4 \times (X_g - X_{mut}) \tag{15}$$

where θ_1, L, θ_4 are random values in the range of (0,1]. The best solution among X_{j1}^{new}, X_{j2}^{new} and X_j will replace X_j.

The second modification method is constructed based on the idea that the bats should try to move their positions toward the best current bat, X_g. Therefore, first the mean value of the bat population is evaluated column-wise as A_D. Now, the position of the i^{th} bat is updated as follows:

$$X^{test3} = X_j + \theta_5 (X_g - \phi_F A_D) \tag{16}$$

where θ_5 is a random value in [0,1] and ϕ_F is a random integer equal to 1 or 2, representing the moving acceleration rate. The flowchart of the modified bat algorithm is shown in Fig. 4.

Moreover, the intelligent fuzzy-based feature selection which is proposed by authors in [35] is applied to extract the most effective features for the anomaly detection model [35].

V. RESULTS AND DISCUSSION

This section provides simulation results based on practical data gathered from a licensed unmodified vehicle and also two other public CAN bus traffic dataset to examine the accuracy of the proposed model. The dataset, with its original format, is a collection of text files containing comma separated values (csv) with a timestamp, message ID, and data field. For performance evaluation, random partitioning is performed to divide the dataset into three sets, namely training, validation, and testing. 70% of the dataset is assigned to the training model, 10% to the validation to avoid any overfitting in the training, and the remaining 20% is assigned to the testing phase, which is sliced up into the normal traffic log and a simulated attacked traffic log. It should be noted that even though the CAN bus protocol specification is openly documented, the meaning of each message ID, the corresponding data field, and the expected broadcast frequency are not available to the researchers due to the proprietary nature of the dataset for different car manufacturers. However, the independency of the proposed method from the meaning of each message ID and data field make the model adaptable with different CAN datasets during the training of the OCSVM. In order to prove the superiority of the proposed method compared to other anomaly detection algorithms for CAN traffic, we benchmarked the proposed MBA-OCSVM with famous anomaly detection algorithms including Isolation Forest and classical one-class SVM. Isolation forests was introduced in [36] as a powerful classification algorithm for real datasets. This method is based on the assumption that it is easier to isolate anomalies from the rest of the observations than to construct a model describing normal behavior. To isolate an observation, it recursively partitions the data set randomly until the observation is the only data point in a partition. Recursive partitioning can be represented by a tree structure. A forest of random trees can collectively give a measure of normality for an observation. Three datasets are gathered from different sources to evaluate the effectiveness and performance of proposed method. One of the sources is an unmodified licensed electric vehicle in normal operation where CAN traffic was logged by the VN1630A device via the ODB-II port existing in the vehicle. Furthermore, in order to prove the independence of the proposed method from a specific car model, we gathered additional CAN traffic traces from two more available datasets. One CAN dataset is available on the

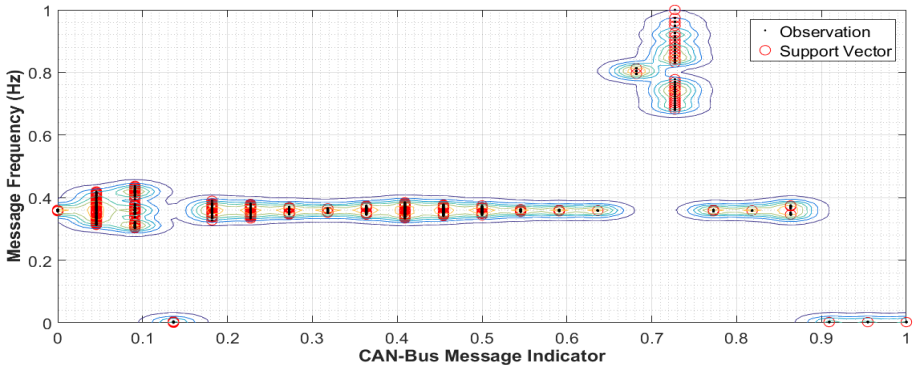

FIGURE 5. Observed frequency, support vectors, and decision boundaries for selected CAN identifier.

website of the Crash Reconstruction Research Consortium of the University of Tulsa [37]. This dataset was recorded from a Dodge RAM Pickup. They used a Dearborn Group (DG) Gryphon S3 [38] to record the data and export it to a .csv file using DG Technologies Hercules software [39]. The data was recorded under normal driving conditions. The car drove away from a normal residential driveway and pulled up on the street. After three right hand turns, the car was backed up into a parking space. The other available CAN traffic dataset is utilized from [40]. This dataset was constructed by logging CAN traffic via the OBD-II port from a real vehicle while message injection attacks were performing, e.g. Denial of Service (DoS) Attack in which messages of '0x000' CAN ID were injected in a short cycle. The dataset includes 2,369,868 attack free states CAN message and 656,579 messages where DoS attacked was launched. The proposed anomaly detection algorithm was implemented in MATLAB 2018b software on a computer with a Core i7 processor 4GHz and 4GB RAM with 100 iterations for OCSVM training. To evaluate the effectiveness of the proposed anomaly detection method, we simulated a condition in which an attacker tries to send a message that is not expected to be sent. For this purpose, we replicated an attack scenario by increasing the frequency of an exemplary message ID in the CAN traffic (here 3F0). During the normal operation of the vehicle, the cycle time of ID 3F0 is 100 ms, which means that this message is triggered in the traffic every 100 ms. For this attack, we intentionally doubled the frequency of this message and sent another message every 100 ms. Since all nodes share a single bus, increasing occupancy of the bus can produce latencies of other messages and cause threats regarding availability with no response to driver's commands. From the CAN traffic captured during the normal driving, we observed that each message occurrence has its own frequency pattern in the traffic, an observation that is leveraged during the training phase of the proposed method. Any deviation from the pre-defined message occurrence frequency can then be detected by this method. Table 1 shows some of the message IDs and their corresponding frequency during 10 minutes of driving for one of the main dataset that we used. The same attack simulation scenarios such as DoS attack was developed for the other datasets that are employed in this study.

TABLE 1. Can identifier and frequencies.

CAN Identifier	Frequency
6FF	101.010101
308	85.74311927
340	50
2A0	48.7804878
670	99.00990099
3F0	100.1666667
D21	38.7804878
210	51.02040816
238	108.6956522
410	93.45794393
200	61.02040816
A7F	49.01960784
B61	10
212	68.54368932
240	78.01010101
4EB	113.6363636
2C1	110.3595506
312	50
5AE	80.01960784

As shown in the above table, each CAN identifier possesses its own occurrence frequency. Before beginning the training phase of the proposed method, data preprocessing (here data numericalization) is required to store the data in the correct structural format for analysis, such that CAN identifiers are transformed into decimal indicators while maintaining the same frequency. Additionally, in order to avoid a very large range between the maximum and minimum values of the dataset, attributes are normalized through attribute rescaling. Having completed the data preprocessing steps, each CAN identifier and its corresponding frequency is imported into the proposed method as an input parameter, and then the MOCSVM training phase will start. Fig. 5 presents the observed frequency, support vectors, and decision boundaries for selected CAN identifiers, shown in Table 1.

A. PERFORMANCE EVALUATION OF MODIFIED BAT ALGORITHM

In order to assess the performance of the proposed MBA, the convergence characteristic of this algorithm is provided

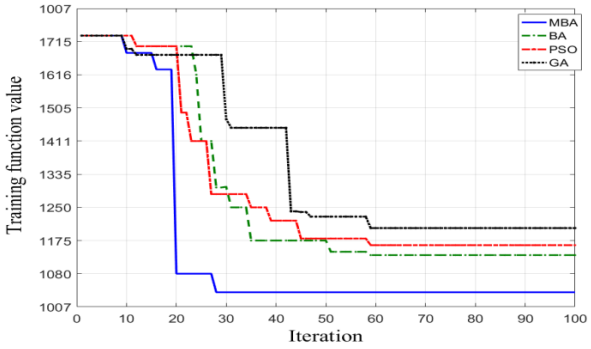

FIGURE 6. Convergence characteristics of the MBA, BA, PSO and GA in the training process.

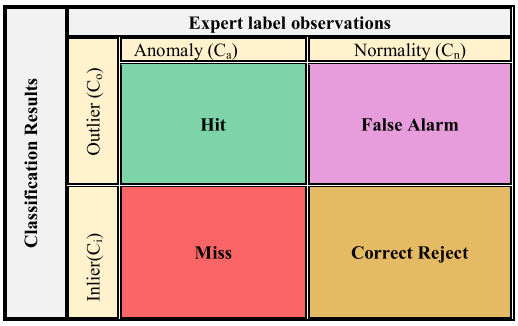

FIGURE 7. Confusion matrix showing the four possible outcomes.

in Fig. 6. To have a better comparison, the convergence characteristics obtained by other well-known algorithms, such as Particle Swarm Optimization (PSO) [31], Genetic Algorithm (GA) [32], and original BA [33] are plotted, simultaneously. The setting parameters of the GA, PSO, BA and MBA algorithms are determined as follows: for the PSO algorithm, the initial number of particles in the swarm equals 40 and the maximum number of iterations is 100. Also, the social parameters and inertia weight factor are set to 1.5 and 0.8, respectively. For the GA, the crossover and mutation probability values are set to 0.7 and 0.08, respectively. The initial population size equals 60 individuals with 100 iterations. For the BA and MBA algorithms, the initial size of the population is 25, the termination criterion is to reach 100 iterations, $\lambda = \gamma = 0.2$ and $\varepsilon = 0.5$. According to Fig. 6, the proposed MBA not only converged first but could reach a more optimal solution, which was not found by the other algorithms. This figure also shows the high search ability and convergence characteristics of the MBA, making it an appropriate tool for use in our case. Therefore, the simulation results provided in the remainder of this paper are all evaluated using MBA as the optimization tool.

B. PERFORMANCE EVALUATION OF PROPOSED MBA-OCSVM

Comparing the proposed MOCSVM results {outlier, inlier} with the real-world observation labels {anomaly, normality} is an essential task for performance evaluation of an anomaly detection method. A confusion matrix is used for performance evaluation, which represents the four possible outcomes when we compare the actual data point labels given by an expert to the corresponding data point results generated by a given classification algorithm. In this case, the four possible outcomes include: hit (*Hi*), false alarm (*FA*), miss (*Mi*), and correct reject (*CR*). If a given data point in the training data with normal CAN traffic is labeled as an anomaly and the outlier-detection algorithm classifies that data point as an outlier as well, the outcome will be a "hit." In addition, if both normal CAN traffic and the anomaly detection algorithm agree with each other about an inlier data point, the result will be "correct reject." The other two possible confusion matrix outcomes ("miss" and "false alarm") are the results of disagreement between the expert (i.e. normal CAN traffic) and the anomaly detection algorithm. If a given data point is considered an outlier but the anomaly detection algorithm detects that point as normal, it will be labeled as a miss. Similarly, if a given data point in the CAN traffic is presented as a normal (inlier) data point but the anomaly detection algorithm wrongly classifies it as an anomalous (outlier) point, the result is considered a "false alarm". The computed outlier scores are converted to the classification outcomes {outlier, inlier} using the threshold defined by the expert, which can be compared to the data point labels {anomaly, normality}, resulting in the aforementioned outcomes. Fig. 7 represents the four areas of the confusion matrix's outcomes, namely hit, miss, false alarm, and correct reject, graphically.

As is observed from Fig. 8, if an attacker tries to manipulate the occurrence frequency of a given CAN message, the proposed algorithm can detect that behavior as an anomalous situation and discard it. Usually, an attacker tries to double the message frequencies to win the arbitration scenario so they can publish their data on the bus. This behavior falls into the hit area in Fig. 6, where the compromised message frequency is higher than the maximum valid frequency for a given message in the normal CAN traffic. The four confusion matrix outcomes have four associated performance rates, as follows:

$$Hit\ Rate = \frac{|H_i|}{|C_A|} \tag{17}$$

$$False\ Alarm\ Rate = \frac{|F_A|}{|C_N|} \tag{18}$$

$$Miss\ Rate = \frac{|M_i|}{|C_A|} \tag{19}$$

$$Correct\ Reject\ Rate = \frac{|C_R|}{|C_N|} \tag{20}$$

To compare the performance of the different algorithms we need a way to quantify the correctly classified windows in the test sets containing attacks. Precision, recall (true positive rate), and specificity are the three famous evaluation indices to measure the reliability of the anomaly detection system in CAN traffic. Precision is defined as ratio of marked anomaly streaming data that is a true anomaly. Recall is the proportion of ratio of identified anomaly CAN data to the actual

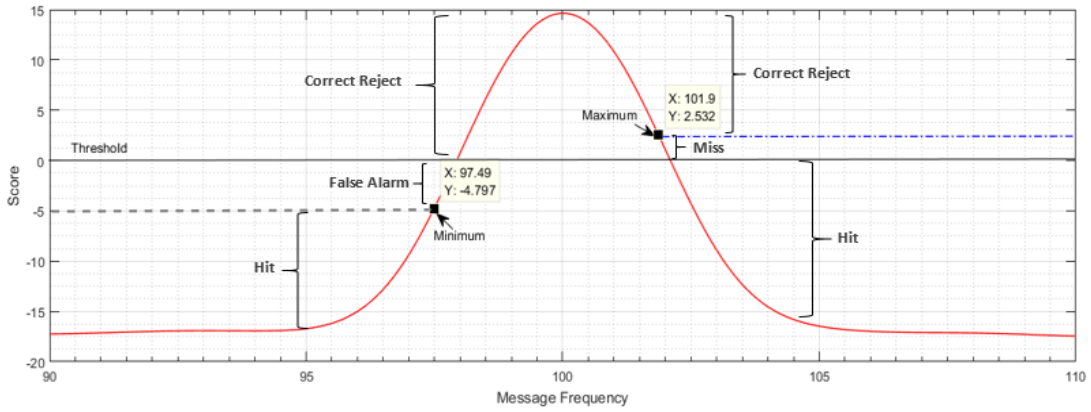

FIGURE 8. Hit, miss, correct reject, and false alarm area for a given CAN bus message.

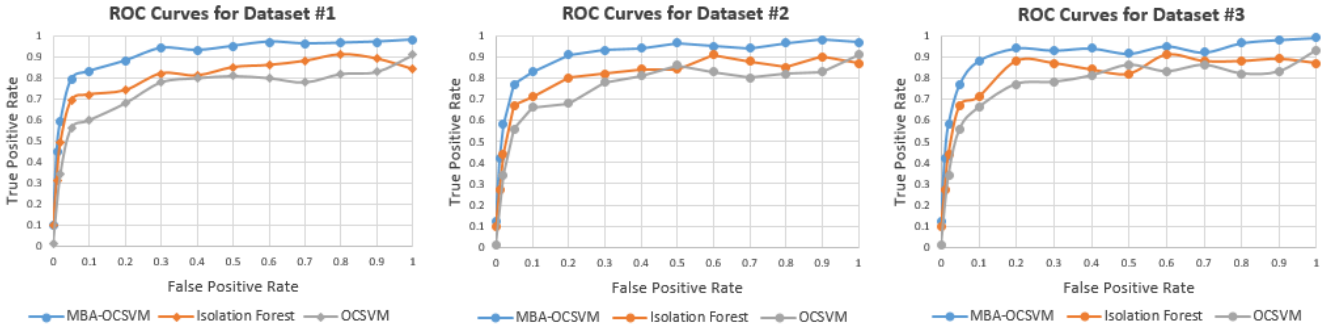

FIGURE 9. ROC performance curve for proposed MOCSVM, Isolation Forest, and classic SVM.
Dataset#1: CAN Bus traffic from unmodified licensed vehicle under normal operation - Dataset#2: CAN Bus traffic dataset from Dodge RAM Pickup track (Crash Reconstruction Research Consortium of the University of Tulsa) – Dataset#3: Available online CAN Bus traffic dataset [40].

anomaly data. Also, specificity is the fraction of the total number of anomalies that the algorithm recognizes as the real normal behavior. The recall value close to 1.0 represents the higher performance of the anomaly detection model. We can measure different detection models by Receiver Operating Characteristic (ROC). ROC is a diagram of the relationship between hit rate (recall) and false alarm rate (1-specificity). The ROC curve should approach the top-left corner as close as possible with a steep slope, as this would indicate a high number of detected anomalies, i.e. a high hit rate, with a low number of false positives, i.e. a low false alarm rate. To draw this diagram, it is important to understand how an anomaly detection algorithm determines which label, e.g. normal or anomalous, to give to an observation. This is done by calculating a so called decision score. The decision score is a measure of how normal, or how anomalous, an observation is. The algorithm sets a threshold and all observations with a decision score on one side of the threshold are labelled as normal behavior and all observations with a decision score on the other side are labelled as anomaly. There is a trade-off between the hit rate (recall) and false alarm rate (1-specificity), for which the expert can determine a threshold value. Choosing the optimal value of the threshold depends on the application and the cost of misclassification. In this study, the MOCSVM is employed as an optimization algorithm to find the optimal value of the thresholds to minimize the number of false alarms while maximizing the hit rate. Fig. 9 shows the ROC for three different datasets which have been used for CAN anomaly detection. In addition, Table 2 summarizes the four aforementioned performance rates for conventional one-class SVM, Isolation Forest, and proposed modified one-class SVM based on MBA.

As it is shown in Fig. 9, the proposed MBA-OCSVM is the one that achieved the highest hit rate (True Positive Rate) and lowest false alarm rate (False Positive Rate) in the ROC curve compared to the Isolation Forest and classical OCSVM. The kernel that is used in the proposed method has the biggest impact on anomaly detection performance and has been optimized by the proposed modified bat algorithm (MBA) as a powerful optimization algorithm which resulted in achieving higher hit rate and lower false alarm rate compared to the other two algorithms. According to the results of Table 2, the proposed method achieves the highest hit rate and correct reject in all three different datasets, compared to those of the other two methods shown here. This high hit rate shows that the proposed method has adequate capability to recognize

TABLE 2. Confusion matrix for OCSVM, isolation forest AND MBA-OCSVM.

Dataset #1	Hit Rate	Miss Rate	False Alarm Rate	Correct Reject Rate
One-Class SVM	83.12%	16.88%	18.07%	81.93%
Isolation Forest	89.01%	10.99%	13.57%	86.43%
Proposed MBA-OCSVM	95.5%	4.5%	8.54%	91.45%
Dataset #2	Hit Rate	Miss Rate	False Alarm Rate	Correct Reject Rate
One-Class SVM	85.62%	14.38%	16.04%	83.96%
Isolation Forest	89.99%	10.01%	13.34%	86.66%
Proposed MBA-OCSVM	96.99%	3.01%	7.11%	92.89%
Dataset #3	Hit Rate	Miss Rate	False Alarm Rate	Correct Reject Rate
One-Class SVM	86.01%	13.99%	15.8%	84.2%
Isolation Forest	88.85%	11.15%	13.34%	86.66%
Proposed MBA-OCSVM	97.01%	2.99%	6.45%	93.55%

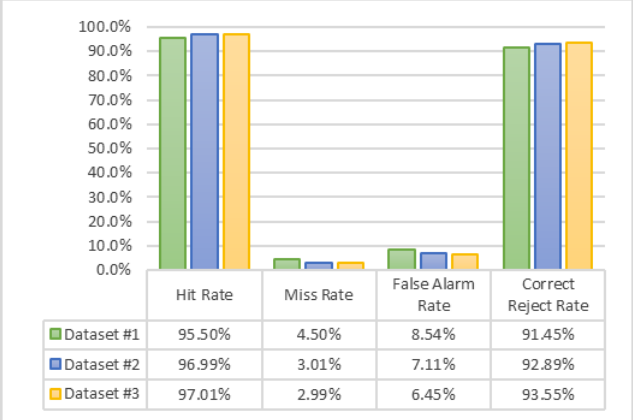

FIGURE 10. Benchmarking of three different datasets for the proposed MBA-OCSVM anomaly detection method.

legitimate messages as normal messages in the traffic. In a similar way, the high correct reject rate shows that malicious messages can be recognized as anomalous behavior in the traffic. In addition, in the proposed method, there is an acceptable drop in the false alarm rate compared to in the other two methods, respectively. This reduction shows the very low percentage of legitimate messages that are incorrectly identified as anomalous by the proposed method. It should be noted that the low false alarm rate of the proposed method mostly exists in frequencies far from the original message frequency. Since it is quite rare for the original message frequencies to reach the overlapping area of abnormality and normality, the proposed method will not affect the performance of the normal CAN bus traffic behavior. Please note that we have simulated a wide range of message hacking with frequencies ranging from 200% to the very small range of a 1% increase/decrease. Therefore, our model is assessed with a varied range of message frequencies to check its true capability as an anomaly detection model. While accurate anomaly detection when dealing with frequencies very close to the real vehicle frequency may be hard to achieve, our model could show good performance with regard to its high hit and correct reject indices. On the other hand, since most practical message flooding starts from high frequencies (generally the hacker does not know the exact frequency of the target message frame), the capability of the proposed model needs to be assessed at high frequency ranges, such as 150% to 200% of the actual message frequency. Furthermore, in order to prove the independence of the proposed method from a specific car model, we gathered additional CAN traffic traces from two other unmodified licensed vehicles. Fig. 10 shows the benchmark results of the proposed MBA-OCSVM anomaly detection model for three different datasets. As shown in Fig. 10, the proposed anomaly detection method is independent from the different message IDs contained in each DBC for each vehicle model and achieves a high hit rate and correct reject rate.

C. COMPUTATIONAL TIME EVALUATION

To evaluate the suitability of the proposed method in real-time systems, the computational time required for each major function is calculated. It is worth noting that all CAN messages, corresponding signals, message ID, message frequencies, and data payload for each message is defined in a proprietary database (DBC) file by each car manufacturer. This DBC file is defined during the manufacturing phase and might be changed in very rare circumstances after production. Therefore, since the construction of DBC file is not changing in real-time, the training phase of the proposed method can be performed offline to save computational time and memory in real-time environment. As a result, only detection/classification part of the algorithm with the optimal parameters of the detection model should be run online in the ECUs. This approach of offline training is widely applied in the industry for other Advanced Driver Assistance Systems (ADAS) features such as pedestrian detection, tracking, parking slot classification, etc. where the model requires computational time more than real-time standards for training. Our proposed method consists of the following major components which mainly affect the computational overhead compared to the other functions: FEATURE_EXTRACTION, MBA_OCSVM_TRAINING, and ANOMALY_DETECTION. The first two functions are performed as offline basis and ANOMALY_DETECTION part will be run online. All simulations are implemented in the MATLAB software which is run on a Core i7 processor, 4GHz and 4GB RAM. FEATURE_EXTRACTION takes 15 ms time to select the most useful features and extract them from CAN traffic log. MBA_OCSVM_TRAINING is the function that requires the longest computational time in the proposed method because calculating the optimal parameters for OCSVM classifier by modified bat optimization method is also included in this function. The computational burden required for MBA_OCSVM_TRAINING and ANOMALY_DETECTION functions for classical OCSVM, Isolation Forest, and proposed MBA-OCSVM are calculated separately which are summarized in Table 3. The computational times for ANOMALY_DETECTION should be minimized because our model needs the CAN traffic to be analyzed in real time.

TABLE 3. Computational time for major functions.

Required Computational time (ms)	Classical OCSVM	Isolation Forest	Proposed MBA-OCSVM
MBA_OCSVM_TRAINING	5,146	4,664	7,256
ANOMALY_DETECTION	1.27	1.43	0.96

According to ISO-11898, CAN protocol supports baud rates from 40 Kbit/s to 1 Mbit/s. Among them, the 500 Kbit/s CAN network (known as high speed CAN) is the most typically used baud rate in automotive industry. To ensure that no message gets lost, the CAN bus load should not be occupied 100%. Hence, in-vehicle CAN networks maintain their bus load no more than 60% [41]. Considering a high speed CAN bus with 500Kbit/s and 60% bus occupancy, it is realized that each message is publishing every 2*ms* by measuring the average time interval between two consecutive CAN messages in the logged traffic. Therefore, the ANOMALY_DETECTION function needs to take less than 2ms to identify the state of a given message whether it is anomalous or normal. As it can be seen from Table 3, the computational time required by the proposed anomaly detection model in the online case is around 1 *ms*, which guarantees the applicable aspect of the proposed model for real cases. Given the fact that recent ECUs that are deployed for ADAS features can support multi-thread processing to boost the computational performance, computational time for each message to be analyzed by ANOMALY_DETECTION function can still be much less negligible and readily implemented in the recent ADAS ECUs.

VI. CONCLUSION

This article proposed an effective anomaly detection model for CAN bus traffic. The proposed method, a modified one-class SVM, was constructed based on a new meta-heuristic optimization algorithm called the Modified Bat Algorithm (MBA), which helps prevent the algorithm from becoming trapped in local optima and avoids pre-mature convergence. The proposed MOCSVM method is used to detect malicious cyberattack behaviors in CAN traffic. The underlying idea behind the proposed method is to establish a model based on the normal CAN bus traffic, which contains recurring patterns in message IDs that are transmitted in a given normal traffic. To this end, any deviation from the normal traffic, e.g. increased message occurrence frequency or message flooding, can be detected by the MBA-OCSVM algorithm as an outlier. In order to demonstrate the high performance of the proposed model, three methods, namely conventional one-class SVM, Isolation Forest, and MBA-OCSVM have been compared. The experimental results show that the proposed method achieved the highest hit rate and lowest miss rate compared to other anomaly detection methods. From a cyber-resilience point of view, the proposed model can provide a highly secure and accurate model to prevent vehicles from being harmed by attackers. Last but not least, the proposed MBA could show high search ability and convergence characteristics, making it a good algorithm for optimization applications.

REFERENCES

[1] M. Steger, C. A. Boano, T. Niedermayr, M. Karner, J. Hillebrand, K. Roemer, and W. Rom, "An efficient and secure automotive wireless software update framework," *IEEE Trans. Indus. Informat.*, vol. 14, no. 5, pp. 2181–2193, May 2018.

[2] *Robert Bosch GmbH*, document 50, C. A. N. Specification, Postfach, 1991.

[3] O. Avatefipour, A. Hafeez, M. Tayyab, and H. Malik, "Linking received packet to the transmitter through physical-fingerprinting of controller area network," in *Proc. IEEE Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2017, pp. 1–6.

[4] Y. Yuan, H. Yuan, L. Guo, H. Yang, and S. Sun, "Resilient control of networked control system under DoS attacks: A unified game approach," *IEEE Trans. Ind. Informat.*, vol. 12, no. 5, pp. 1786–1794, Oct. 2016.

[5] M. Markovitz and A. Wool, "Field classification, modeling and anomaly detection in unknown CAN bus networks," *Veh. Commun.*, vol. 9, pp. 43–52, Jul. 2017.

[6] A. Theissler, D. Ulmer, and I. Dear, "Interactive Knowledge Discovery in recordings from vehicle tests," in *Proc. 33rd FISITA World Automot. Congr.*, Budapest, Hungary, 2010, pp. 1–10.

[7] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, p. 15, 2009.

[8] T. Kieu, B. Yang, and C. S. Jensen, "Outlier detection for multidimensional time series using deep neural networks," in *Proc. 19th IEEE Int. Conf. Mobile Data Manage. (MDM)*, Jun. 2018, pp. 125–134.

[9] C. Sommer, R. Hoefler, M. Samwer, and D. W. Gerlich, "A deep learning and novelty detection framework for rapid phenotyping in high-content screening," *Mol. Biol. Cell*, vol. 28, no. 23, pp. 3428–3436, 2017.

[10] S. Sharma, C. R. Krishna, and S. K. Sahay, "Detection of advanced malware by machine learning techniques," in *Soft Computing: Theories and Applications*. Singapore: Springer, 2019, pp. 333–342.

[11] O. Avatefipour and H. Malik, "State-of-the-art survey on in-vehicle network communication (CAN-Bus) security and vulnerabilities," Feb. 2018, *arXiv:1802.01725*. [Online]. Available: https://arxiv.org/abs/1802.01725

[12] Q. Wang and S. Sawhney, "VeCure: A practical security framework to protect the CAN bus of vehicles," in *Proc. Int. Conf. Internet Things (IOT)*. Cambridge, MA, USA, Oct. 2014, pp. 13–18. doi: 10.1109/IOT.2014.7030108.

[13] M. Marchetti, D. Stabili, A. Guido, and M. Colajanni, "Evaluation of anomaly detection for in-vehicle networks through information-theoretic algorithms," in *Proc. IEEE 2nd Int. Forum Res. Technol. Soc. Ind. Leveraging Better Tomorrow (RTSI)*, Sep. 2016, pp. 1–6.

[14] M.-J. Kang and J.-W. Kang, "Intrusion detection system using deep neural network for in-vehicle network security," *PLoS ONE*, vol. 11, no. 6, 2016, Art. no. e0155781.

[15] A. Theissler, "Detecting known and unknown faults in automotive systems using ensemble-based anomaly detection," *Knowl.-Based Syst.*, vol. 123, pp. 163–173, May 2017.

[16] S. N. Narayanan, S. Mittal, and A. Joshi, "OBD_SecureAlert: An anomaly detection system for vehicles," in *Proc. IEEE Int. Conf. Smart Comput. (SMARTCOMP)*, May 2016, pp. 1–6.

[17] K.-T. Cho and K. G. Shin, "Fingerprinting electronic control units for vehicle intrusion detection," in *Proc. 25th USENIX Secur. Symp. (USENIX Secur.)*. Berkeley, CA, USA: USENIX Association, 2016.

[18] M. Tayyab, A. Hafeez, and H. Malik, "Spoofing attack on clock based intrusion detection system in controller area networks," in *Proc. NDIA Ground Vehicle Syst. Eng. Technol. Symp.*, Aug. 2018, pp. 1–13.

[19] A. Taylor, S. Leblanc, and N. Japkowicz, "Anomaly detection in automobile control network data with long short-term memory networks," in *Proc. IEEE Int. Conf. Data Sci. Adv. Anal. (DSAA)*, Oct. 2016, pp. 130–139.

[20] H. Li, L. Zhao, M. Juliato, S. Ahmed, M. R. Sastry, and L. L. Yang, "POSTER: Intrusion detection system for in-vehicle networks using sensor correlation and integration," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2017, pp. 2531–2533.

[21] A. Ganesan, J. Rao, and K. G. Shin, "Exploiting consistency among heterogeneous sensors for vehicle anomaly detection," SAE Tech. Paper 2017-01-1654, 2017.

[22] M. Pajic, J. Weimer, N. Bezzo, P. Tabuada, O. Sokolsky, I. Lee, and G. J. Pappas, "Robustness of attack-resilient state estimators," in *Proc. ACM/IEEE 5th Int. Conf. Cyber-Phys. Syst. (CPS Week ICCPS)*, Apr. 2014, pp. 163–174.

[23] M. Marchetti and D. Stabili, "Anomaly detection of CAN bus messages through analysis of ID sequences," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Los Angeles, CA, USA, Jun. 2017, pp. 1577–1583, doi: 10.1109/IVS.2017.7995934.
[24] I. Studnia, E. Alata, V. Nicomette, M. Kaâniche, and Y. Laarouchi, "A language-based intrusion detection approach for automotive embedded networks," *Int. J. Embedded Syst.*, vol. 10, no. 1, pp. 1–11, 2018.
[25] A. Tomlinson, J. Bryans, S. A. Shaikh, and H. K. Kalutarage, "Detection of automotive CAN cyber-attacks by identifying packet timing anomalies in time windows," in *Proc. 48th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw. Workshops (DSN-W)*, Luxembourg City, Luxembourg, Jun. 2018, pp. 231–238. doi: 10.1109/DSN-W.2018.00069.
[26] F. Martinelli, F. Mercaldo, V. Nardone, and A. Santone, "Car hacking identification through fuzzy logic algorithms," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*, Jul. 2017, pp. 1–7.
[27] H. Lee, S. H. Jeong, and H. K. Kim, "OTIDS: A novel intrusion detection system for in-vehicle network by using remote frame," in *Proc. 15th Annu. Conf. Privacy, Secur. Trust (PST)*, Aug. 2017, pp. 57–60.
[28] A. Taylor, N. Japkowicz, and S. Leblanc, "Frequency-based anomaly detection for the automotive CAN bus," in *Proc. WCICSS*, Dec. 2015, pp. 45–49.
[29] H.-J. Xing and M. Ji, "Robust one-class support vector machine with rescaled hinge loss function," *Pattern Recognit.*, vol. 84, pp. 152–164, Dec. 2018.
[30] A. Hafeez, H. Malik, O. Avatefipour, P. R. Rongali, and S. Zehra, "Comparative study of can-bus and flexray protocols for in-vehicle communication," SAE Tech. Paper 2017-01-0017, 2017.
[31] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. Int. Conf. Neural Netw.*, vol. 4, Nov./Dec. 1995, pp. 1942–1948.
[32] H. Zhi and S. Liu, "Face recognition based on genetic algorithm," *J. Vis. Commun. Image Represent.*, vol. 58, pp. 495–502, Jan. 2019.
[33] X.-S. Yang and A. H. Gandomi, "Bat algorithm: A novel approach for global engineering optimization," *Eng. Comput.*, vol. 29, no. 5, pp. 464–483, 2012.
[34] S. Yilmaz, E. U. Kucuksille, and Y. Cengiz, "Modified bat algorithm," *Elektronika ir Elektrotechnika*, vol. 20, no. 2, pp. 71–79, 2014.
[35] O. Avatefipour and A. Nafisian, "A novel electric load consumption prediction and feature selection model based on modified clonal selection algorithm," *J. Intell. Fuzzy Syst.*, vol. 34, no. 4, pp. 2261–2272, 2018.
[36] F. T. Liu, K. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. 8th IEEE Int. Conf. Data Mining*, Dec. 2008, pp. 413–422.
[37] The University of Tulsa. *Crash Reconstruction Research Consortium*. Accessed: May 5, 2019. [Online]. Available: http://tucrrc.utulsa.edu
[38] Dearborn Group. *Gryphon G2*. Accessed: May 5, 2019. [Online]. Available: https://www.dgtech.com/gryphon-g2/
[39] Dearborn Group. *Hercules Software*. Accessed: May 5, 2019. [Online]. Available: https://www.dgtech.com/software/#Hercules
[40] H. Lee, S. H. Jeong, and H. K. Kim. (2017). *OTIDS: A Novel Intrusion Detection System for in-Vehicle Network by Using Remote Frame*. PST (Privacy, Security and Trust). [Online]. Available: http://ocslab.hksecurity.net/Dataset/CAN-intrusion-dataset
[41] S. Nürnberger and C. Rossow, "-vatiCAN-Vetted, Authenticated CAN Bus," in *Proc. Int. Conf. Cryptograph. Hardw. Embedded Syst.* Berlin, Germany: Springer, Aug. 2016, pp. 106–124.

OMID AVATEFIPOUR received the master's degree in computer engineering from the University of Michigan–Dearborn. He has work experience at Vector CANTech Company, as an Embedded Software Engineer. He has also worked as a Researcher in Information System, Security, and Forensics (ISSF) Laboratory, Department of Electrical and Computer Engineering (ECE), University of Michigan–Dearborn. Additionally, he was a primary Researcher with the Laboratory of Control and Robotics, Institute of Advanced Science and Technology, IRAN SSP Research and Development Center. He is currently with Valeo North America Inc., as a System Engineer with the Research and Development Group. His research interests include in-vehicle network communication protocol security, autonomous vehicles, embedded systems, machine learning, intelligent control systems, and robotics.

AMEENA SAAD AL-SUMAITI received the B.Sc. degree in electrical engineering from UAE University, UAE, in 2008, and the M.A.Sc. and Ph.D. degrees in electrical and computer engineering from the University of Waterloo, Canada, in 2010 and 2015, respectively. She was a Visiting Assistant Professor with MIT, Cambridge, USA, in 2017. She is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Khalifa University, UAE. Her research interest includes intelligent systems.

AHMED M. EL-SHERBEENY received the master's and Ph.D. degrees in mechanical engineering from West Virginia University (WVU), in 2001 and 2006, respectively, where he was a Graduate Teacher and a Research Assistant. He has been an Assistant Professor with the Industrial Engineering Department, since 2010, was the former Head of the Alumni and Employment Unit, College of Engineering, King Saud University, from 2013 to 2018. His research interests include human factors engineering, manufacturing engineering, and engineering education.

EMAD MAHROUS AWWAD is currently pursuing the Ph.D. degree with the Electrical Engineering Department, King Saud University. He graduated and employed as a Teaching Assistant at Industrial Electronics and Control Engineering Department, Faculty of Electronic Engineering, Menofia University, Egypt. He developed his researches in the field of design, control, and implementation of autonomous mobile robot. He is interested in modeling, optimization, observer design, and MPC controller of vehicle dynamics under the wheel-terrain interaction slippage phenomenon. He is also interested in artificial intelligent, machine learning, and deep learning related to the field of robotics and image processing.

MOHAMMED A. ELMELIGY received the B.Sc. degree in information technology from Menoufia University of Egypt, in 2005. He has been a Software Engineer with King Saud University, Riyadh, Saudi Arabia, since 2009. His research interests include Petri nets, supervisory control of discrete event systems, database software, and network administration.

MOHAMED A. MOHAMED (M'16) received the B.Sc. and M.Sc. degrees from Minia University, Minia, Egypt, in 2006 and 2010, respectively, and the Ph.D. degree from King Saud University, Riyadh, Saudi Arabia, in 2016. He joined the College of Electrical Engineering and Automation, Fuzhou University, China, as a Postdoctoral Research Fellow, in 2018. He has been a Faculty Member with the Department of Electrical Engineering, College of Engineering, Minia University, Minia, Egypt, since 2008. He has supervised multiple M.Sc. and Ph.D. theses, worked on a number of technical projects, and published various articles and books. His current research interests include the areas of renewable energy, energy management, power electronics, power quality, optimization, and smart grids. He has also joined the editorial board of some scientific journals and the steering committees of many international conferences.

HAFIZ MALIK is currently an Associate Professor with the Electrical and Computer Engineering (ECE) Department, University of Michigan–Dearborn. He has published more than 70 articles in leading journals, conferences, and workshops. His research in cyber-security, multimedia forensics, information security, wireless sensor networks, steganography/steganalysis, pattern recognition, information fusion, and biometric security is funded by the National Academies, National Science Foundation and other agencies. Dr. Malik is also on the Review Board Committee of the IEEE Technical Committee on Multimedia Communications (MMTC). He organized Special Track on Doctoral Dissertation in Multimedia, in the 6th IEEE International Symposium on Multimedia (ISM) 2006. He is also organizing a special session on "Data Mining in Industrial Applications" within the IEEE Symposium Series on Computational Intelligence (IEEE SSCI) 2013. He is serving as a Vice Chair for the IEEE SEM, Chapter 16, since 2011. He is also serving on several technical program committees, including the IEEE AVSS, ICME ICIP, MINES, ISPA, CCNC, ICASSP, and ICC. He is serving as an Associate Editor for the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, since August 2014 and for the Springer *Journal of Signal, Image, and Video Processing* (SIVP), in May 2013.