A Neuro-Genetic Security Framework for Misbehavior Detection in VANETs

Ila Naqvi[1], Alka Chaudhary[2], Anil Kumar[3]
Amity Institute of Information Technology, Amity University, Noida, India[1, 2]
School of Computing, DIT University, Dehradun, India[3]

Abstract*—Genetic Algorithm (GA) is an excellent optimization algorithm which has attracted the attention of researchers in various fields. Many papers have been published on works done on GA, but no single paper ever utilized this algorithm for misbehavior detection in VANETs. This is because GA requires manual definition of fitness function and defining a fitness function for VANETs is a complex task. Automating the creation of these fitness functions is still a difficulty, even though studies have found several successful applications of GA. In this study, a neuro-genetic security framework has been built with ANN classifier for detecting misbehavior in VANETs. It leverages a genetic algorithm for feature reduction with ANN as a dynamic fitness function, considering both node behaviors and contextual GPS data. Deployed at the Roadside Unit (RSU) level, the framework detects misbehaving nodes, broadcasting alerts to RSUs, Central Authority and the vehicles. The ANN based fitness function has been employed in GA that enabled the GA to select the best results. The 10- fold CV used enabled the whole system to be unbiased giving a precision accuracy of 0.9976 with recall and F1 scores as 0.9977, and 0.9977 respectively. Comparative evaluations, using the VeReMi Extension dataset, demonstrate the framework's superiority in precision, recall, and F1 score for binary and multiclass classification. This hybrid genetic algorithm with ANN fitness function presents a robust, adaptive solution for VANET misbehavior detection. Its context-aware nature accommodates dynamic scenarios, offering an effective security framework for the evolving threats in vehicular environments.

Keywords*—*VANET security; genetic algorithm; ANN fitness function; misbehavior detection; hybrid detection

I. INTRODUCTION

Modern technology advancements and high transportation expectations have caused the global automobile utilization rate to rise quickly [1]. The transport industry is dealing with a variety of issues due to the quick rise in the number of automobiles and the limited space in the infrastructure of roads, including a spike in traffic accidents, prolonged traffic jams, damage to public property and human life, etc. To address these issues and improve the efficiency of the transportation sector, Vehicular ad hoc networks (VANETs) emerged as a particular kind of mobile ad hoc network (MANET) [2] in which mobile nodes are vehicles such as cars, trucks, buses, and motorcycles etc. Vehicles follow the design of the road, corresponding to traffic regulations and flow restrictions rather than moving at random. Vehicles exhibit different speeds, and their movement and their behavior are impacted by the traffic signals, road signs, and other vehicles. The density of these networks or topology of these networks varies very rapidly, depending on the area, the time of day, and recent occurrences (like traffic jams or accidents) [3].

Around the year 2000, Vehicular Ad-hoc Networks (VANETs) were the subject of investigation for many research laboratories [4]. VANET was first used to improve road traffic safety and lower the number of accidents and traffic jams [5]. Today it covers numerous integrated services employing other technologies in addition to the basic functionality provided by VANET architecture, indicating a significantly wider application [6].

As shown in Fig. 1, the key constituents of VANETs are typically Trusted Authority (TA), Roadside Units (RSUs), and Onboard Units (OBUs). As the only component in a VANET that can be completely trusted, TA oversees monitoring the whole setup and changing the parameters for the other components. RSU, on the other hand, is set up along roadsides as wireless infrastructure to link cars to TA. Every vehicle has an OBU, a wireless device that processes, transmits, and receives messages (such as road status, condition, and so on) from other cars [7]. Vehicles can communicate with each other through vehicle-to-vehicle (V2V) communication in VANETs as well as with infrastructure through RSUs through vehicle-to-infrastructure (V2I) communication. Every vehicle in the VANET transmits data messages and safety messages every 100 to 300ms to the vehicles in range in accordance with dedicated short-range communications (DSRC) requirements [8]. The transmission of data and safety-related information by vehicles in an open-access setting creates security and privacy challenges for VANETs. If appropriate precautions are not taken, attackers may utilize user information to launch a variety of attacks that might be harmful to the network and its users.

Predictable mobility patterns, a large network size, frequent disconnections, a high rate of topology changes, and strict delay constraints are only a few of the distinguishing characteristics of VANETs that make it extremely prone to a variety of misbehaviors. Even though VANET research has been ongoing for more than a decade, there are still many open challenges, including ineffective QoS, uneven flow traffic, security and privacy concerns, poor resource utilization, and inefficient information distribution [9].

Additionally, there is need to apply various contextual information to enhance the ability to differentiate between nodes that are genuinely malicious and those that exhibit anomalous behavior for contextual reasons. Fig. 2 provides a great illustration of misbehavior scenario in VANETs. The vehicle v1 drops packets in both scenarios, leading most of the current security systems to treat it as a misbehaving node without doing any more research. But taking a closer look at the setting in which packet loss occurs in, it is found that in case (a), v1 drops packets likely due to the busy channel; in case (b), no external factor prevents it from forwarding those packets, indicating that v1 is acting maliciously. This example makes it abundantly evident that context could be crucial in identifying misbehaving nodes in VANETs.

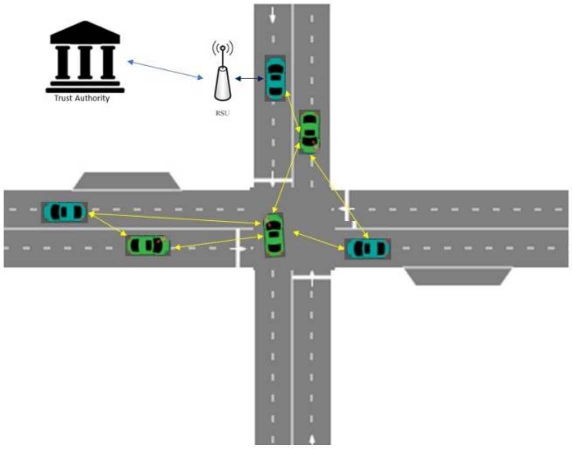

Fig. 1. Structure of VANET.

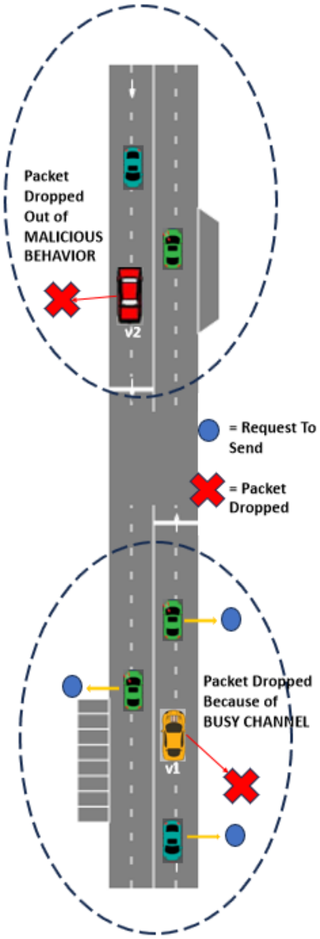

Fig. 2. Misbehavior example scenario.

In this paper, a context aware framework is proposed for detecting misbehavior in VANETs. In the proposed framework, both node behaviors (taken from BSM messages) and contextual information (taken from GPS data) are represented as features in the feature vector to train a genetic algorithm, with an artificial neural network (ANN) serving as its fitness function. The genetic algorithm takes the feature vector as input, dynamically reducing features based on ANN accuracy, and subsequently classifies whether a node exhibits malicious behavior. This hybrid genetic algorithm could offer a solution by combining the strengths of genetic algorithms, which excel in optimization and exploration of solution spaces, with ANN, creating a more dynamic and effective security framework for VANETs.

The main contributions of the study are to:

- Propose a security framework for misbehavior detection for VANETs using hybrid genetic algorithm with ANN fitness function.
- Compare multiple ML algorithms to be used as fitness function for genetic algorithm for better misbehavior detection.
- Compare the proposed framework with the existing ones for evaluation of the results.

Section II of this paper presents the overview of the existing works done in the field; Section III presents the proposed framework, including the communication architecture and the processing steps. Section IV discusses the simulation setup and results that include a comprehensive exploration of the framework, and a series of experiments that demonstrate the framework's effectiveness in detecting misbehaviour across a range of scenarios. Furthermore, it provides evidence of the framework’s superiority over traditional machine learning models and existing misbehaviour detection methods, underscoring the critical role of context-awareness and the ANN-based fitness function in VANET security. Section V presents the discussion of the results, and the conclusion of the paper is provided in Section VI.

II. Existing Works

Over the past years, several security methods have been investigated to identify and address these misbehaviors in VANETs. The proposed Trust-Based Event Detection Algorithm (TB-EDA) compares the trust values of the neighboring cars of a node with the threshold trust value measured to identify misbehaviors [10]. In study [11], the authors introduced the Vehicular Reference Misbehavior dataset (VeReMi) to assess various misbehavior detectors. They also assessed different detectors on their datasets using metrics such as precision and recall. While misbehavior detection systems based on rules or specifications can provide security against known attacks, they lack the ability to identify unknown attacks.

To enhance robustness against Sybil attacks in VANETs, [12] proposed anonymous authentication and Sybil attack detection protocol. In a broader framework employing subjective logic, [13] improved two position verification mechanisms for misbehavior detection. The ML-based

Intrusion Detection System (IDS) proposed in [14] focuses on thwarting spoofing attacks using a probabilistic cross-layer approach in a VANET consisting of Electric Vehicles. The research in [15] presented SVM-based IDS for VANETs, incorporating an enhanced penalty function to strengthen the classifier's regularization. The study in [16] suggested ML-based IDS for VANETs, where XGBoost demonstrated superior performance in binary class and multi-class classification problems. The research in [17] introduced a data-centric approach to identify position falsification attacks, employing machine learning (ML) algorithms. The proposed method combines information from two consecutive Basic Safety Messages (BSMs) for both training and testing purposes.

The majority of these security solutions use one or more pre-established, predefined thresholds to identify abnormal nodes from regular ones. However, it is not possible to determine a single set of thresholds that perform effectively in every situation due to the very dynamic nature of VANETs. However, as the use of machine learning solutions for misbehavior detection is rising, the studies have showcased more dynamic and adaptive approaches for VANETs. But there is still a gap: there are limitations of traditional machine learning approaches in handling the dynamic and complex nature of VANET security. While machine learning has shown promise, it may struggle with the rapidly changing and unpredictable nature of vehicular environments and the ever-evolving threats. It is proposed that hybrid algorithms could fill this gap by introducing a more adaptive and robust approach by combining the best of two or more algorithms for identifying the border between normal and misbehaving nodes automatically.

III. Proposed Framework

A. Proposed Architecture

Fig. 3 presents the communication architecture of the proposed framework. Most studies of misbehavior detection in VANETs applied misbehavior detection in On Board Units (OBUs) of individual vehicles. Keeping in mind that the communication range of RSUs is much broader than the communication range of vehicles [18], in the proposed scheme, our detection framework will be deployed at the RSU level. RSUs have greater computing capacity available for misbehavior detection than vehicle OBUs, which are more resource restricted. Notifying legitimate vehicles of a possible attack even before they come into communication range of the misbehaving vehicle is another advantage of the suggested approach. It will work as follows: Every vehicle gets its credentials for communication during the registration process with the authority (CA). When a vehicle sends the Basic Safety Messages (BSMs), are received by all vehicles and RSUs within the sender vehicle's communication range. These BSMs along with the GPS data are used by the RSUs for identifying the misbehaving nodes through the hybrid detection module of the proposed framework. On detecting misbehavior, an alert is generated and is broadcasted to the vehicles and other RSUs in the communication range. When such alert is received by any vehicle, it updates the misbehaving node's data into its local OBU to prevent future communication. The RSUs broadcast the alert to their respective vehicles in range and in such a way alert reaches the other vehicles through the network of RSUs.

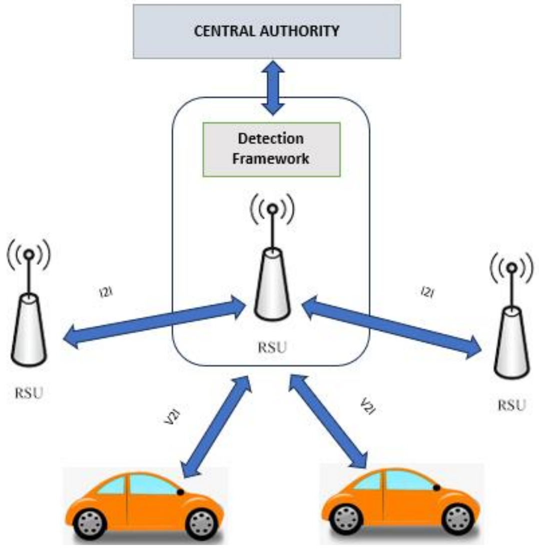

Fig. 3. Communication architecture.

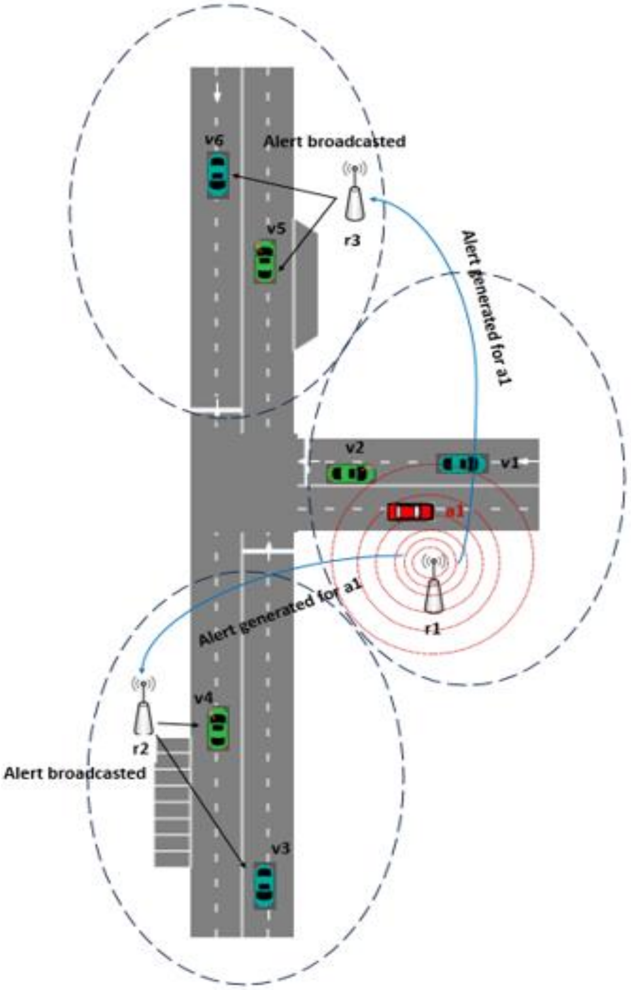

Fig. 4. VANET example scenario.

An example scenario is provided in Fig. 4 where v1, v2, v3, v4, v5, v6 are legitimate vehicles, a1 is a misbehaving vehicle, r1, r2, r3 are RSUs. Vehicle a1 is detected misbehaving by r1 which then broadcasts the alert to vehicles and RSUs in range. v1 and v2 being in range get this alert directly while v3, v4, v5 and v6 get this alert through the RSUs r2 and r3. In this way vehicles that are not even in range get the alert and are prevented from being attacked or misled. After detection of misbehavior, additional action may be taken by the CA depending on its own policy and procedures, which are not in the scope of this study.

B. Proposed Framework

The core contribution of this study is the framework for misbehaviour detection for the VANETs. The performance efficiency and the effectiveness of the suggested security framework are both significantly impacted by the entire features that are employed by the detection system. Detection accuracy, computational time and memory requirements have been identified as the primary factors for which the reduction in the total number of features is required by the system.

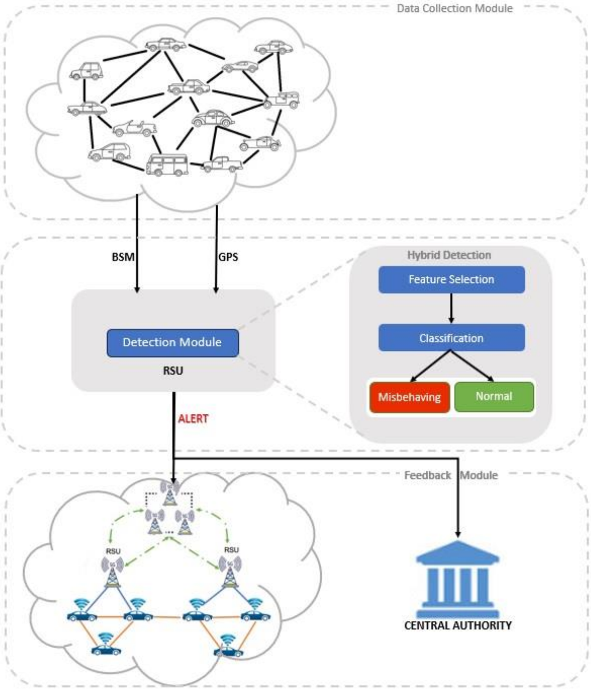

Fig. 5. Proposed Framework

The proposed framework (see Fig. 5) consists of three modules which include:

- Data collection module: This module collects the behavioural data and the contextual data from the network and sends it to the detection module.
- Hybrid Detection: This module primarily consists of a Genetic Algorithm model, which analyses the data, filters out the irrelevant characteristics, and reconstructs a low-dimensional feature dataset then uses supervised algorithms to categorize traffic, judge if it is being subjected to an attack, and decide whether to provide a warning in response to the findings.
- Feedback module: Using the machine's output status and alarm information, this module modifies its operations.

C. Processing Steps

There are two primary phases to implementing the suggested framework: dataset preparation and hybrid classification.

1) Dataset preparation: Every BSM has a unique message ID, the sender's ID, and a time stamp showing when it was sent in addition to the pertinent status information. A labelled VeReMi Extension dataset [19] was used for training and testing our proposed framework. It consists of message logs for each vehicle, which include BSM messages (labelled as type=3) received from other vehicles via DSRC, as well as GPS information (labelled as type=2) about the vehicle. There is one ground truth file and several unique log files for every simulation, which include the BSMs that each vehicle received. As a result, there are exactly as many log files as there are receivers. Every BSM is logged in several distinct log files as it is received by numerous vehicles. First to get rid of redundant information, the processing of the merged log files was carried out. After that, the combined log files and the ground truth file are merged, and a labelled dataset is produced for every simulation using this combined file.

2) Hybrid detection: After the required labelled dataset is ready, Artificial Neural Network (ANN) fitness function based Genetic Algorithm (GA) was used for feature reduction and detecting misbehavior.

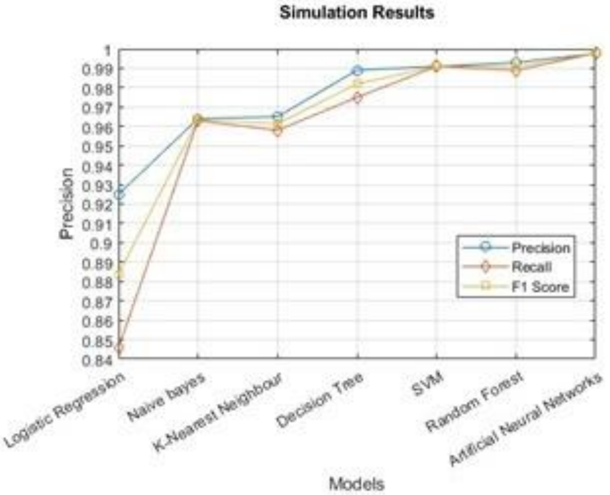

Fig. 6. Simulation Results for different fitness functions

a) Genetic Algorithm with ANN: Genetic Algorithm is used to solve a problem from a pool of potential solutions. GA is based on a fitness function, through which the generated candidates are iteratively developed, modified, and chosen for survival. Fitness functions are often manually constructed heuristics that rank candidate solutions according to how near they are to being accurate, with the candidate solutions that score higher being more likely to be picked for next generations [20]. Automating the creation of these fitness functions is still a difficulty, even though studies have found several successful applications of GA. In [21], authors have proposed an approach called NetSyn to automatically generate these fitness functions by representing their structure with a neural network. While they investigated this technique in the context of Machine Programming, they presented the technique to be applicable and generalizable to other domains also. Using that approach various classification algorithms have been explored that could be used as fitness function for GA. Among these, the following seven classifiers were selected: ANN, K-Nearest Neighbor (kNN), Random-Forest, Decision tree, Logistic Regression, Support Vector Machine (SVM) and Naïve Bayes, and compared the results (See Fig. 6). The results show that ANN yielded the best results among all. Algorithm 1 outlines the use of Genetic Algorithm with ANN Fitness function. The parameters used in GA are provided in Table I and the flowchart for the process is shown in Fig. 7.

Algorithm 1: Genetic Algorithm with ANN

```
BEGIN
Random initialization of population
For each generation (t) from 1 to max_generation:
    For each solution: evaluate fitness of each solution in the population
            Split the solution into 10 folds for cross-validation into ANN.
            Calculate and return the average accuracy as the ratio of correct predictions to the total number of predictions
    End
    Display progress information every 10 generations (optional).
    Sort the solutions based on their fitness.
    Select the two best solutions for reproduction.
    Apply Genetic Algorithm operators (crossover and mutation) to create new solutions.
    Update the population with the new solutions.
END
```

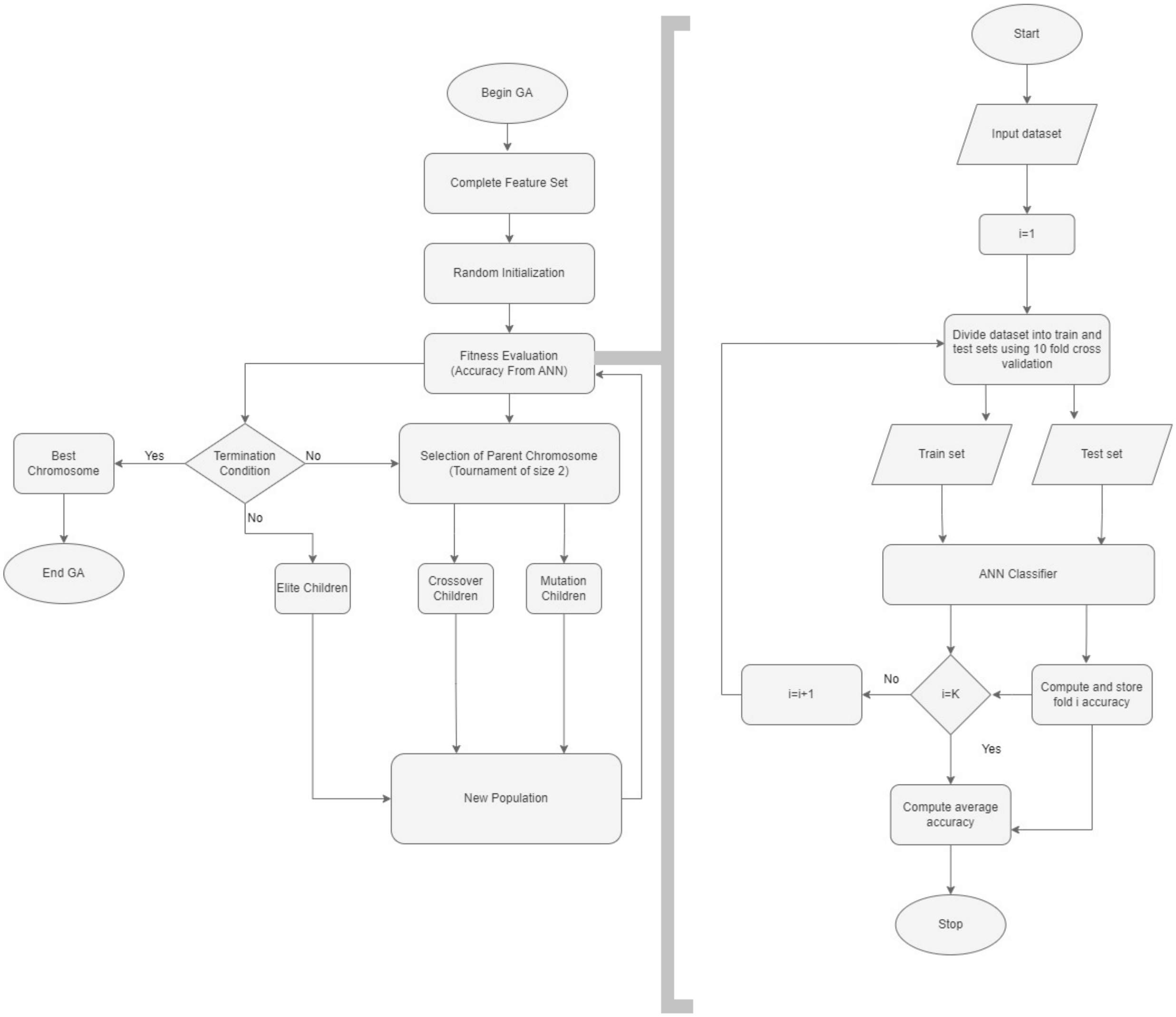

Fig. 7. Simulation Results Genetic Algorithm-based Feature Selection.

b) Cross-validation: To prevent the model from overfitting and effectively measure its accuracy, the entire dataset was split into k folds of train and test sets, with one split serving as the validation set and the remaining k-1 split as training set. Depending on the dataset, the value of k typically ranges from 5 to 10, and in this implementation, k = 10 has been used.

TABLE I. PARAMETERS USED IN GA

Parameter	Value
Genome length	33
Population size	300
Number of generations	500
Mutation	Uniform Mutation
Mutation Probability	0.1
Crossover	Arithmetic Crossover
Crossover Probability	0.8
Fitness Function	ANN-Based Classification Accuracy
Selection scheme	Tournament of size 2
Elite Count	2

IV. SIMULATION SETUP AND RESULTS

A publicly accessible VeReMi Extension dataset [19] has been utilized for training and testing the framework, to evaluate the suggested framework and guarantee fair comparisons. Using common metrics on this dataset, the suggested technique was evaluated and compared its results with those of previous approaches.

Information from both normal and misbehaving cars make the VeReMi Extension an imbalanced dataset [22]. The measures listed in Eq. (1) through Eq. (3) have been utilized for evaluating and comparing the performance of the proposed framework because accuracy by itself is insufficient as a metric for an imbalanced dataset. The misbehaving vehicle is indicated by a 1 in our dataset, whereas the genuine vehicle is shown by a 0.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (1)$$

$$Recall \quad = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (2)$$

$$F1\ Score \ = \quad 2 * \frac{Precision * Recall}{Precision + Recall} \quad (3)$$

The implementation was carried out in two ways: binary classifications, to simply classify vehicles as normal or misbehaving and multiclass classification, to identify the specific misbehaviour being carried out.

A. Results

The results of the proposed framework's binary classification can be seen in Fig. 8. 99.99% of the cases may be identified accurately when utilizing the binary classification approach. When the ROC is analysed, the framework's good measure of separability is shown by an AUC (area under the curve) that is close to 1. With binary classification the framework has shown the precision, recall and F1 scores as 0.9999 for all three metrics.

The results of the multiclass classification of the proposed framework are displayed in Fig. 9. When employing the multiclass classification approach, the proposed framework has 99.76% detection accuracy. Since the AUC is close to 1, the framework is also performing quite well when it comes to multiclass categorization. With multiclass classification, the proposed framework has shown precision, recall and F1 scores as 0.9976, 0.9977, and 0.9977 respectively.

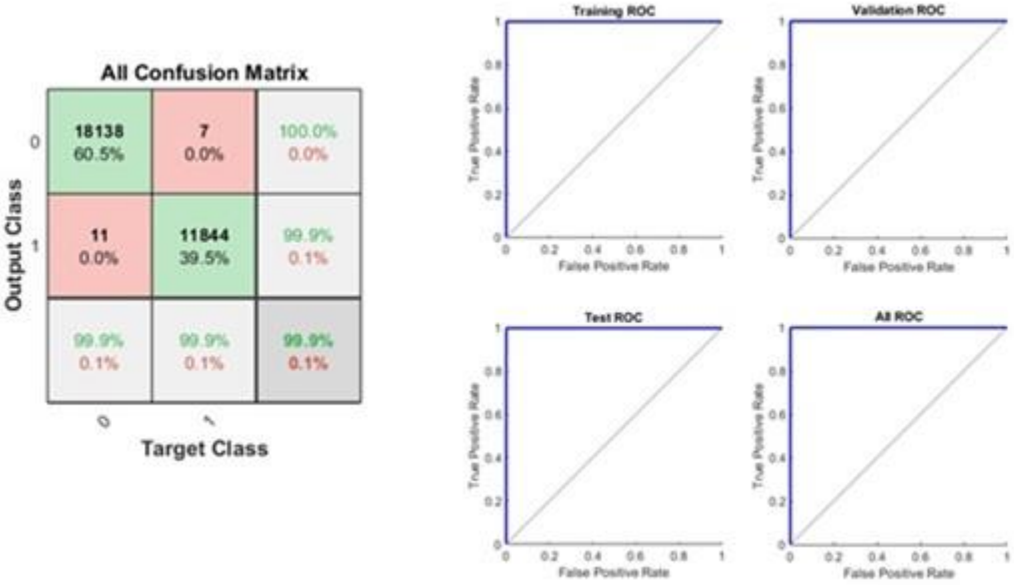

Fig. 8. Confusion matrix (left) and ROC (right) for binary classification.

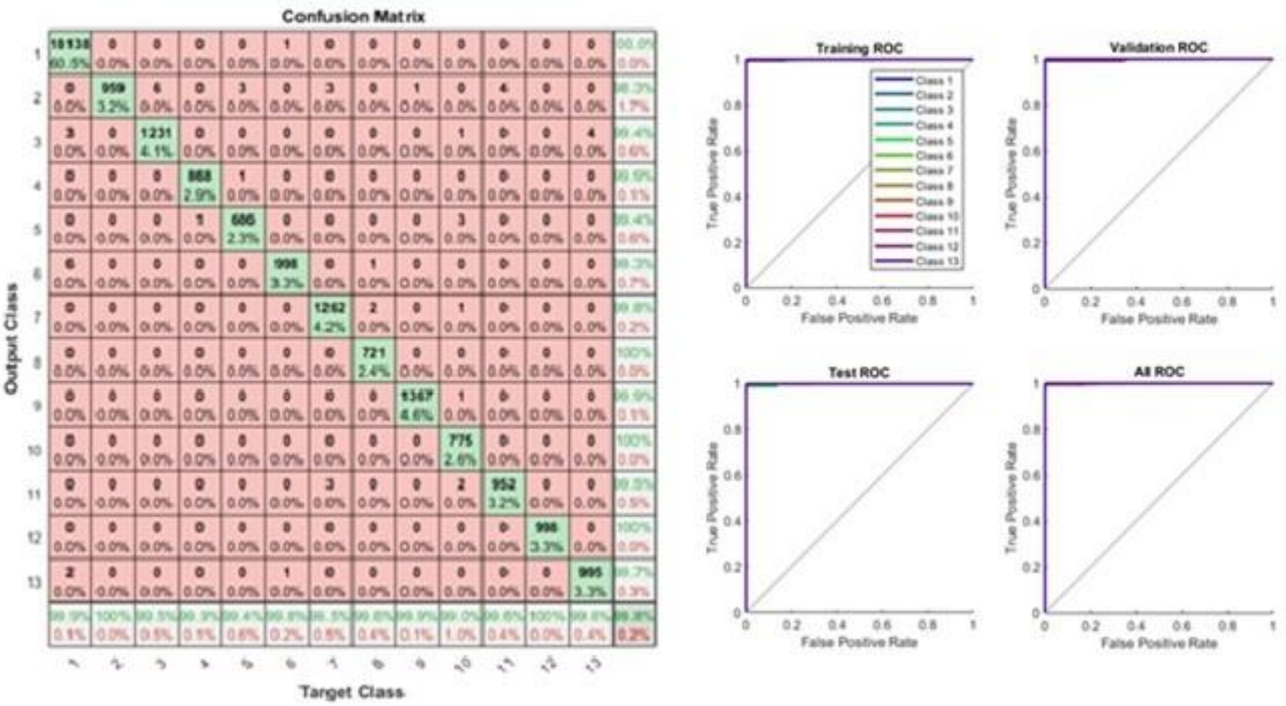

Fig. 9. Confusion matrix (left) and ROC (right) for multiclass classification.

B. Performance Evaluation with Varying Misbehaving Node Densities

Total five datasets were created with varying percentages of misbehaving nodes (10%, 20%, 30%, 40%, and 50%), and evaluated for the framework's performance under various misbehavior node densities. The simulation's outcomes are displayed in Fig. 10. The findings demonstrated that the framework demonstrated 100% accuracy with precision, recall, and F1 score all pointing to 1 when the proportion of misbehaving nodes was just 10% of the total number of nodes. The accuracy, recall, and F1 score values decreased as more and more misbehaving nodes were added to the dataset. Though the framework's performance was worse at 50% misbehaving nodes than in the 10% case, it still demonstrated 0.9967 recall, 0.9966 F1 score, and accuracy.

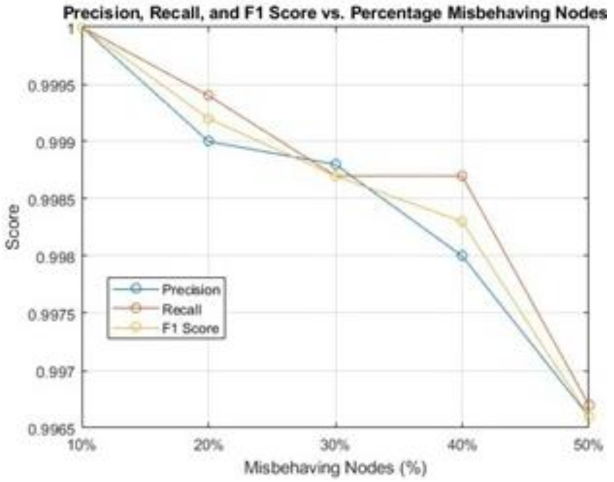

Fig. 10. Simulation Results for different node densities.

This demonstrates that even in the worst-case situations, where the fraction of misbehaving nodes is 50% of all nodes, our methodology is producing good outcomes.

C. Comparison with Existing Works

Table II presents a comparison between the existing works and the accuracy, recall, and F1 scores achieved using our proposed framework. The results make it evident that, in comparison to all other frameworks, Paper 4 [24] has extremely poor accuracy, recall, and F1 score values. While Paper 1 [23] had a high accuracy value of 0.9999 and performed comparable to the suggested model, it had a somewhat lower F1 Score and recall value. When it came to multiclass classification, the suggested model outperformed the other methods, with classifications showing 0.9976, 0.9977, and 0.9977 precision, recall, and F1 scores, respectively. When it came to binary classification, the framework displayed 0.9999 precision, recall, and F1 scores.

TABLE II. COMPARISON WITH EXISTING WORKS

Paper	Precision	Recall	F1 Score
Proposed Framework (with Binary Classification model)	0.9999	0.9999	0.9999
Proposed Framework (with Multiclass Classification model)	0.9976	0.9977	0.9977
Paper 1 [23]	0.9999	0.9554	0.977144
Paper 2 [11]	0.9886	0.8277	0.901023
Paper 3 [17]	0.988	0.99	0.988999
Paper 4 [24]	0.887	0.616	0.727069
Paper 5 [25]	0.978	0.932	0.954446

V. Discussion

The results of this study align closely with the theoretical framework presented in the introduction. The use of Genetic Algorithm (GA) with an Artificial Neural Network (ANN) fitness function for misbehavior detection in Vehicular Ad-hoc Networks (VANETs) is supported by the findings, which demonstrate high accuracy and robustness across different scenarios. This alignment validates the theoretical underpinnings of using a hybrid genetic algorithm approach for effective misbehavior detection in VANETs. The results of this study have several important implications for theory, practice, and future research. From a theoretical perspective, the success of the GA-ANN framework underscores the importance of context-awareness in misbehavior detection, as demonstrated by the use of GPS data and node behaviors as features in the classification process. Practically, this framework offers a robust and adaptive solution for VANET security, capable of detecting misbehaving nodes with high accuracy and efficiency. Compared to existing works in the field, the proposed GA-ANN framework demonstrates superior performance in terms of accuracy, precision, recall, and F1 scores. This highlights the effectiveness of the hybrid genetic algorithm approach in addressing the challenges of misbehavior detection in VANETs. The framework also outperforms existing methods in terms of adaptability to dynamic scenarios and robustness against evolving threats.

One of the key strengths of this study is the use of a comprehensive dataset and rigorous evaluation methodology, including 10-fold cross-validation, to assess the performance of the framework. However, one limitation is the reliance on simulated data, which may not fully capture the complexities of real-world VANET environments. Future research could involve testing the framework in real-world settings to validate its effectiveness further.

VI. Conclusion

This study presents a novel approach to detect misbehavior in VANETs using Genetic Algorithm with Artificial Neural Networks. Through the implementation of the misbehavior detection framework in the RSUs, which may broadly disseminate this information with other RSUs and vehicles, the proposed solution moves the computational burden from vehicles (OBUs). In contrast to existing methods, the proposed strategy uses ANN based fitness function in Genetic Algorithm. It was discovered after comparing several ML algorithms for fitness evaluation that ANN produces the best results.

The performance of the proposed framework was also compared with the existing solutions that have been published in the literature. The collected findings show that, in terms of precision, recall and F1 score, the suggested framework consistently outperforms the existing approaches across a variety of misbehavior types. Developing strong frameworks that can identify various misbehaviors using various GPS and BSM characteristics (such as heading, acceleration, and speed) is essential for safe VANET functioning.

Future research directions could focus on further enhancing the framework's performance by exploring different feature selection techniques or integrating other machine learning algorithms to improve classification accuracy. Additionally, extending the framework to address other types of misbehavior and incorporating real-time data processing capabilities could enhance its practical utility in real-world VANET environments.

Acknowledgment

This research was partially supported by Gigasheet.co. We thank gigasheet.co for providing us with enterprise access to their website for converting json data to CSV.

References

[1] D. Oladimeji, K. Gupta, N. A. Kose, K. Gundogan, L. Ge, and F. Liang, “Smart Transportation: An Overview of Technologies and Applications,” *Sensors*, vol. 23, no. 8, p. 3880, Apr. 2023, doi: 10.3390/s23083880.

[2] M. S. Corson, J. P. Macker, and G. H. Cirincione, “Internet-based mobile ad hoc networking,” *IEEE Internet Computing*, vol. 3, no. 4, pp. 63–70, 1999, doi: 10.1109/4236.780962.

[3] J. J. P. C. Rodrigues, *Advances in Delay-Tolerant Networks (DTNs)*. Woodhead Publishing, 2020.

[4] M. J. Haidari and Z. Yetgin, “Veins based studies for vehicular ad hoc networks,” *2019 International Artificial Intelligence and Data Processing Symposium (IDAP)*, Sep. 2019, Published, doi: 10.1109/idap.2019.8875954.

[5] M. A. Al-Shareeda, M. Anbar, I. H. Hasbullah, and S. Manickam, “Survey of Authentication and Privacy Schemes in Vehicular ad hoc Networks,” *IEEE Sensors Journal*, vol. 21, no. 2, pp. 2422–2433, Jan. 2021, doi: 10.1109/jsen.2020.3021731.

[6] K. L. K. Sudheera, M. Ma, G. G. Md. N. Ali, and P. Han Joo Chong, “Delay efficient software defined networking based architecture for vehicular networks,” *2016 IEEE International Conference on Communication Systems (ICCS)*, Dec. 2016, Published, doi: 10.1109/iccs.2016.7833564.

[7] M. A. Al-Shareeda and S. Manickam, “A Systematic Literature Review on Security of Vehicular Ad-Hoc Network (VANET) Based on VEINS Framework,” *IEEE Access*, vol. 11, pp. 46218–46228, 2023, doi: 10.1109/access.2023.3274774.

[8] F. M. Salem and A. S. Ali, “SOS: Self-organized secure framework for VANET,” *International Journal of Communication Systems*, vol. 33, no. 7, Jan. 2020, doi: 10.1002/dac.4317.

[9] G. G. Md. Nawaz Ali, P. H. J. Chong, S. K. Samantha, and E. Chan, “Efficient data dissemination in cooperative multi-RSU Vehicular Ad Hoc Networks (VANETs),” *Journal of Systems and Software*, vol. 117, pp. 508–527, Jul. 2016, doi: 10.1016/j.jss.2016.04.005.

[10] R. P. Nayak *et al.*, “TFMD-SDVN: a trust framework for misbehavior detection in the edge of software-defined vehicular network,” *The Journal of Supercomputing*, vol. 78, no. 6, pp. 7948–7981, Jan. 2022, doi: 10.1007/s11227-021-04227-z.

[11] R. W. van der Heijden, T. Lukaseder, and F. Kargl, “VeReMi: A Dataset for Comparable Evaluation of Misbehavior Detection in VANETs,” *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 318–337, 2018, doi: 10.1007/978-3-030-01701-9_18.

[12] T. B. M. de Sales, A. Perkusich, L. M. de Sales, H. O. de Almeida, G. Soares, and M. de Sales, “ASAP -V: A privacy-preserving authentication and sybil detection protocol for VANETs,” *Information Sciences*, vol. 372, pp. 208–224, Dec. 2016, doi: 10.1016/j.ins.2016.08.024.

[13] R. W. van der Heijden, A. Al-Momani, F. Kargl, and O. M. F. Abu-Sharkh, “Enhanced Position Verification for VANETs Using Subjective Logic,” *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, Sep. 2016, Published, doi: 10.1109/vtcfall.2016.7881000.

[14] D. Kosmanos *et al.*, “A novel Intrusion Detection System against spoofing attacks in connected Electric Vehicles,” *Array*, vol. 5, p. 100013, Mar. 2020, doi: 10.1016/j.array.2019.100013.

[15] A. Alsarhan, M. Alauthman, E. Alshdaifat, A.-R. Al-Ghuwairi, and A. Al-Dubai, “Machine Learning-driven optimization for SVM-based intrusion detection system in vehicular ad hoc networks,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 5, pp. 6113–6122, Feb. 2021, doi: 10.1007/s12652-021-02963-x.

[16] A. R. Gad, A. A. Nashat, and T. M. Barkat, “Intrusion Detection System Using Machine Learning for Vehicular Ad Hoc Networks Based on ToN-IoT Dataset,” *IEEE Access*, vol. 9, pp. 142206–142217, 2021, doi: 10.1109/access.2021.3120626.

[17] A. Sharma and A. Jaekel, “Machine Learning Based Misbehaviour Detection in VANET Using Consecutive BSM Approach,” *IEEE Open Journal of Vehicular Technology*, vol. 3, pp. 1–14, 2022, doi: 10.1109/ojvt.2021.3138354.

[18] Chenxi Zhang, Xiaodong Lin, Rongxing Lu, Pin-Han Ho, and Xuemin Shen, “An Efficient Message Authentication Scheme for Vehicular Communications,” *IEEE Transactions on Vehicular Technology*, vol. 57, no. 6, pp. 3357–3368, Nov. 2008, doi: 10.1109/tvt.2008.928581.

[19] J. Kamel, M. Wolf, R. W. van der Hei, A. Kaiser, P. Urien, and F. Kargl, “VeReMi Extension: A Dataset for Comparable Evaluation of Misbehavior Detection in VANETs,” *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, Jun. 2020, Published, doi: 10.1109/icc40277.2020.9149132.

[20] A. Lambora, K. Gupta, and K. Chopra, “Genetic Algorithm- A Literature Review,” *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, Feb. 2019, Published, doi: 10.1109/comitcon.2019.8862255.

[21] S. Mandal, T. Anderson, J. Turek, J. Gottschlich, S. Zhou, and A. Muzahid, “Learning fitness functions for machine programming”. *Proceedings of Machine Learning and Systems*, *3*, pp. 139-155, Jan. 2021, doi:10.48550/arXiv.1908.08783.

[22] J. Brownlee, “A Gentle Introduction to Imbalanced Classification,” *MachineLearningMastery.com*, Jan. 14, 2020. https://machinelearning mastery.com/what-is-imbalanced-classification/

[23] C. Mangla, S. Rani, and N. Herencsar, “A misbehavior detection framework for cooperative intelligent transport systems,” *ISA Transactions*, vol. 132, pp. 52–60, Jan. 2023, doi: 10.1016/j.isatra.2022.08.029.

[24] S. So, P. Sharma, and J. Petit, “Integrating Plausibility Checks and Machine Learning for Misbehavior Detection in VANET,” *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Dec. 2018, Published, doi: 10.1109/icmla.2018.00091.

[25] S. Gyawali, Y. Qian, and R. Q. Hu, “Machine Learning and Reputation Based Misbehavior Detection in Vehicular Communication Networks,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 8, pp. 8871–8885, Aug. 2020, doi: 10.1109/tvt.2020.2996620.