Design of DDSM Modulator without Unwanted Tones with Effective Random Dither

Ghasem Hematipour [1], Seyed Ali Sadatnoori[2]
1-Department of electrical engineering, Shoushtar branch, Islamic Azad University, Shoushtar, Iran.
Email: Hematipour@gmail.com
2- Department of electrical engineering, Shoushtar branch, Islamic Azad University, Shoushtar, Iran.
Email: ssadatnoori@yahoo.com (Corresponding author)

ABSTRACT:
The output of a Digital Delta-Sigma Modulator (DDSM) is always a periodic signal and the input is constant. The frequency synthesizer is commonly used due to its flexibility and convenient frequency adjustment. This paper presents a new method to implement random dither in MASH 1-1-1 modulator. There are two different methods to remove unwanted tones from the MASH modulator output, which are deterministic and random. Deterministic methods include setting the initial conditions of the internal registers, using the first-module quantizer, and the HK-MASH structure. On the other hand, the random method involves applying a random dither signal to the circuit in different ways to eliminate the periodic behavior of the output string. Deterministic methods are not useful for most practical applications. In most practical applications, dither signal is a suitable method. In this article, the disadvantages of different dither methods will be investigated and a new vibration method for MASH sigma-delta modulators for use in fractional frequency synthesizers will be presented.

KEYWORDS: Fractional frequency synthesizers, Digital sigma delta modulator, dither signal, undesirable tones.

1. INTRODUCTION

In digital PLL circuits, fractional synthesizers, and data converters, the technique of quantization noise shaping via sigma-delta modulation is widely used. In the digital sigma delta modulator, the discrete time input is oversampled and quantized to produce an output signal with lower accuracy and reduce the quantization noise power in the desired frequency band.

The block diagram of a first-order digital sigma-delta modulator is shown in Fig. 1. The functions F(z), G(z) are the forward and feedback path filter functions, respectively, and Q(.) is the quantizer. The signal and noise conversion functions of this modulator STF(z), NTF(z) are expressed by the following relationship.

$$STF(z) = \frac{F(z)}{1+F(z)G(z)}, NTF(z) = \frac{1}{1+F(z)G(z)} \qquad (1)$$

Therefore, the z domain model of the modulator output is expressed by equation (2).

$$Y(z) = STF(z)X(z) + NTF(z)E(z) \qquad (2)$$

where E(z) is the quantization noise which is modeled as white noise.

Paper type: Research paper
https://doi.org/
Received: 2 December 2023; revised: 13 January 2024; accepted: 15 February 2024; published: 1 June 2024
How to cite this paper: Gh. Hematipour , S. A. Sadatnoori, **“Design of DDSM Modulator without Unwanted Tones with Effective Random Dither”**, *Majlesi Journal of Telecommunication Devices*, Vol. 13, No. 2, pp. 59-62, 2024.

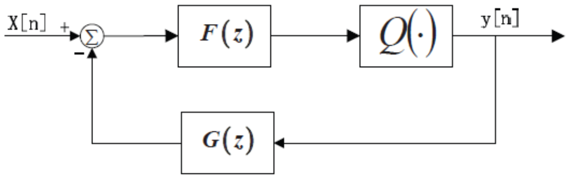


Fig. 1. Block diagram of first order DDSM modulator.

Fig. 2 shows a block diagram of a third-order MASH digital sigma-delta modulator, which is constructed from the cascade connection of three first-order blocks and a noise shaping network. In this structure, the output of each stage is entered into the noise shaping network, where the quantizer noise will be modulated and removed with the D(z) filter. The modulator output relationship is as follows.

$$Y(z) = STF_{out}(z)X(z) + NTF_{out}(z)E_3(z) \qquad (3)$$

where $STF_{out}(z)$ and $NTF_{out}(z)$ are general signal and noise transformation functions expressed by the following relation.

$$STF_{out}(z) = STF_1(z)D_1(z), NTF_{out}(z) = \prod_{i=1}^{3} NTF_i(z) \qquad (4)$$

where 〖$STF_i(z)$ and $NTF_i(z)$ are the i-th stage signal and noise transformation functions. In this article, the MASH 1-1-1 modulator signal and noise conversion functions are considered as follows.

$$F_1(z) = F_2(z) = F_3(z) = \frac{1}{1-z^{-1}}, G_1(z) = G_2(z) = G_3(z) = z^{-1}, D_1(z) = 1, D_2(z) = D_3(z) = -(1-z^{-1}) \quad (5)$$

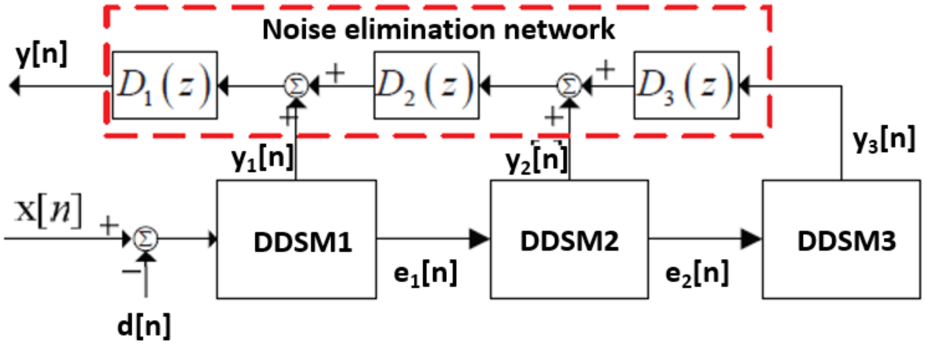


Fig. 2. MASH 1-1-1 block diagram.

2. REDUCTION OF UNDESIRABLE TONES IN MASH MODULATORS

Some important methods that have been proposed to reduce unwanted tones in MASH modulators are:

- Prime Modulo Quantizer: In these MASH modulators, the initial conditions of the internal registers must be chosen correctly. Hosseini and Kennedy showed that the length of the output string is equal to the quantization level M, if the value of the modulus of M is prime. This design reduces the range of undesirable tones. But quantizers with prime number modulus are more complicated than power-2 quantizers. Also, the initial condition of the quantization error in the modulator must be zero. [1,2]
- -Output feedback: Lie provides a new method for randomizing the output string of a digital sigma-delta modulator. In this method, the output of the MASH modulator is entered into a digital processor and processed. The output of this accumulator storage provides transportation to the next stage. [3,4]
- -Modified feedback error modulator (HK): Hosseini and Kennedy presented a new method to increase the string length of the MASH modulator, which uses a feedback loop. In this MEFM MASH each hardware has an additional accumulator. Therefore, MASH modulator hardware will increase. [5,6]
- -Shaped accumulated dither: In this method, a one-bit dither signal will be added to the input of the modulator

to reduce unwanted tones. This problem can be solved by using filtered vibration. In this article, it is proposed to add dither to the other stages of the MASH modulator to obtain the shaped dither. [7,8,9]

3. PROPOSED METHOD FOR APPLYING DITHER

If a dither signal from a pseudo-random source is added to the input of the third stage to reduce low-frequency quantization noise, the amplitude of the unwanted tones will not be reduced unless the dither string length is very large. But if the dither signal is applied to another path, it can greatly reduce the range of unwanted tones. There are a few different ways to apply dither. In these paths, a compromise must be considered between the loss of the alternating mode of the string and the increase of low frequency noise. In this article, the best way to add dither in MASH 1-1-1 modulator is shown in Fig. 3. This dither can be applied to the input of the last two stages of the modulator. This will not increase the circuit hardware. In this circuit, the three-bit output has the following relationship.

$$\mathrm{Y}(\mathrm{z}) = \mathrm{X}(\mathrm{z}) + \frac{1}{\mathrm{M}}\mathrm{D}(\mathrm{z})((1-\mathrm{z}^{-1})^{1} + (1-\mathrm{z}^{-1})^{2}) + (1-\mathrm{z}^{-1})^{3}\mathrm{E}_{3}(\mathrm{z}) \qquad (6)$$

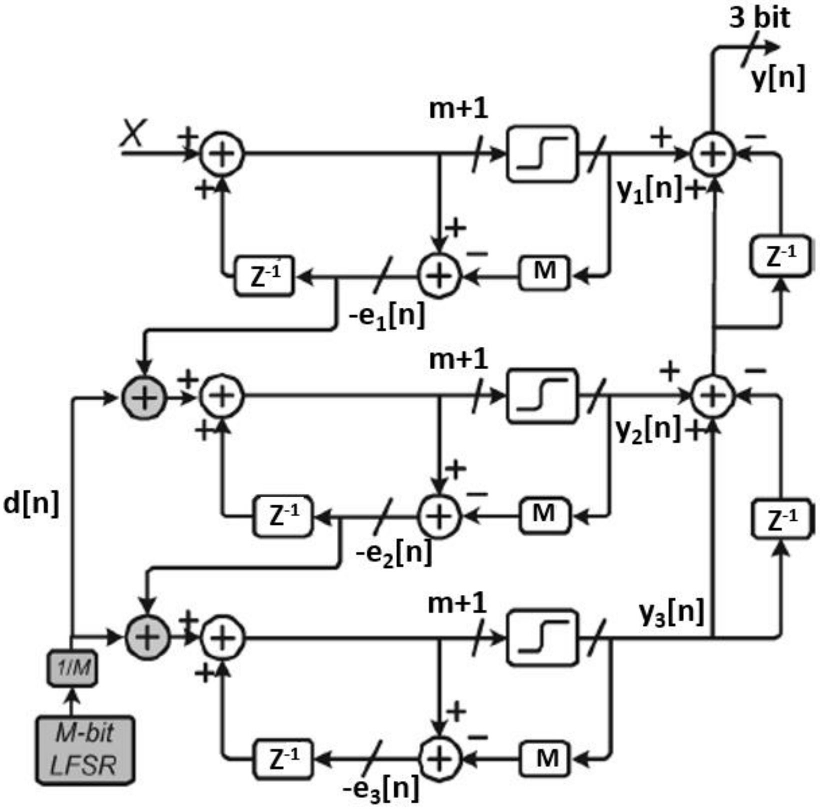


Fig. 3. MASH 1-1-1 sigma-delta modulator with recommended effective dither signal.

In this section, an optimal method of applying dither to MASH digital sigma-delta modulators is presented in order to reduce the undesirable tones of the output spectrum. In this method, the digital dither signal is applied according to Fig. 4.

In this structure, high frequency noise shaping is done and undesirable tones are reduced. The output spectrum of MASH 1-1-1 modulator with dither signal is shown in Fig. 5. The vibration signal is shaped and the low frequency noise is reduced.

4. CONCLUSION

An optimal dither structure is proposed for MASH 1-1-1 sigma-delta modulator in fractional frequency synthesizers. In this method, a simple pseudo-random string generator is used to randomize the output spectrum without increasing the power and occupied area. The dither is not dependent on the sigma-delta modulator input and its shaping is done properly. The synthesizer output phase noise at high frequency does not have undesirable tones; Therefore, the loop filter can be designed with fewer restrictions.

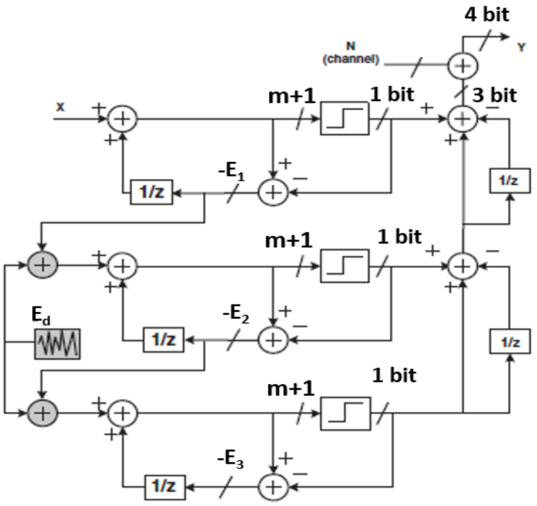


Fig. 4. Proposed dither method to MASH 1-1-1 sigma-delta modulator

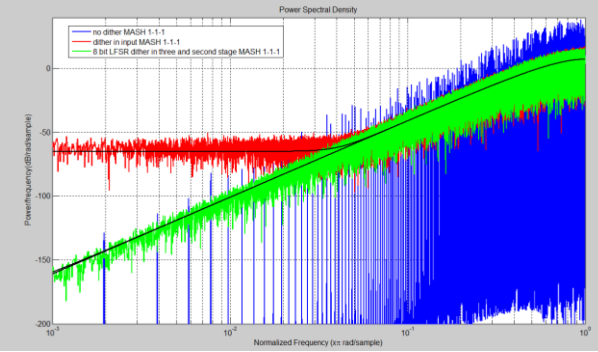


Fig.5. MASH modulator output spectrum 1-1-1 without dither, with dither at the input and 8-bit dither at the input of the second and third stage.

REFERENCES

[1] K. Hosseini, and M.P. Kennedy, "**Mathematical analysis of a prime modulus quantizer MASH digital delta-sigma modulator,"** IEEE Transactions on Circuits and Systems Part II: Express Briefs, vol. 54, no. 12, pp. 1105–1109, Dec. 2007.

[2] K. Hosseini, M.P. Kennedy (2011), "**Minimizing Spurious Tones in Digital Delta-Sigma Modulators",** Analog Circuits and Signal Processing, Springer, New York.

[3] M.P. Kennedy, H. Mo, B. Fitzgibbon, "**Spurious tones in digital delta-sigma modulators resulting from pseudorandom dither "** , journal of the Franklin Institute, vol: 352, No: 08, 2015, pp. 1-20.

[4] M.P. Kennedy, B. Fitzgibbon, K. Dobmeier, "**Spurious tones in digital delta sigma modulators with pseudorandom dither**", in 2013 IEEE International Symposium on Circuits and Systems (ISCAS),Beijing, China, 2013, pp. 2747-50.

[5] S.A.Sadatnoori, E. Farshidi, S. Sadughi,. **A novel structure of dithered nested digital delta sigma modulator with low-complexity low-spur for fractional frequency synthesizers**, *COMPEL - The international journal for computation and mathematics in electrical and electronic*, 35(1), 157-171, (2016).

[6] S.A. Sadatnoori, E. Farshidi, S. Sadughi, **A Novel Architecture of Pseudorandom Dithered MASH Digital Delta-Sigma Modulator with Lower Spur** , *Journal of Circuits, Systems and Computers*, 25(7):1650072 · (2016).

[7] Y. Liao, X. Fan, Z. Hua, **Influence of LFSR Dither on the Periods of a Mash Digital Delta- Sigma Modulator,** *IEEE Transactions on Circuits and Systems II: Express Briefs*, 66 (1), 66-70, (2019).

[8] V. Mazzaro, M.P. Kennedy, **Mitigation of "Horn Spurs" in a MASH-Based Fractional-N CP-PLL**; IEEE Transactions on Circuits and Systems II: Express Briefs, 67 (5), 821-825, (2020).

[9] D. Mai, M.P. Kennedy, **Analysis of Wandering Spur Patterns in a Fractional-** ***N*** **Frequency Synthesizer With a MASH-Based Divider Controller**; IEEE Transactions on Circuits and Systems I: Regular Papers, 67 (3), 729-742, (2020).

Deep Learning for Line Road Detection in Smart Cars

Zohreh Dorrani
Department of Electrical Engineering, Payame Noor University, Tehran, Iran.
Email: dorrani.z@pnu.ac.ir (Corresponding author)

ABSTRACT:
In recent years, smart cars have advanced rapidly and use artificial intelligence technology to predict behavior, make decisions, and control goals. This technology significantly determines the knowledge level of vehicles. In the complex and dynamic environment of road traffic, negligence, and inattention can lead to irreparable damage. Real-time identification and positioning of road lines are key to improving the safety of driving cars. To improve the performance of safe driving assistance, this paper shows how to detect road lines using the YOLOV8 algorithm and then make decisions to continue driving straight, turn right, and turn left. Simulation results and accuracy comparison show that this approach can be used as a reliable source for creating driving assistance scenarios in natural road traffic environments. The use of artificial intelligence and the precise architecture of YOLOV8 promise high speed and accuracy in smart cars.

KEYWORDS: Artificial Intelligence, Deep Learning, Smart Cars, YOLOV8.

1. INTRODUCTION

Lane detection [1] is a crucial task in autonomous driving systems. Lane markings provide essential information about road structure and traffic flow, which is critical for navigation and automated vehicle control.

Various methods have been employed for lane detection. Traditional computer vision approaches utilize geometric features of lane markings to identify them. However, these methods may prove unreliable under poor visibility or complex road conditions.a

Recently, deep learning methods [2, 3] have been adopted for lane detection. These methods employ deep neural networks [4, 5] to learn the characteristics of lane markings from images. The YOLOV8 architecture is a deep neural network architecture [6] designed for object detection in images and videos. This architecture is renowned for its fast and accurate object detection [7] [8] capabilities. Consequently, this paper utilizes this architecture for lane detection.

The YOLOV8 architecture can detect lane markings in real time, which is essential for autonomous driving systems. Additionally, it can detect lane markings with high accuracy under various road and visibility conditions. Moreover, the proposed method can be applied to detect lane markings on various types of roads and streets.

The paper begins by introducing the YOLOV8 architecture. Subsequently, the proposed method is elaborated. The proposed method is evaluated using simulations. The results are presented, and the conclusion follows.

2. RELATED WORK

Road lane detection [9] is one of the key tasks in advanced driver assistance and automated driving systems. This allows cars to detect their lane on the road and steer their movements safely in traffic. In recent years, deep learning has emerged as a powerful tool for solving various computer vision problems, including road line detection. A new method for real-time lane detection is presented, utilizing U-Net to extract road lane features from input images and classify pixels as lane or foreground [10]. This method achieves good accuracy in detecting road lines under various lighting and weather conditions. This method achieves good accuracy in detecting road lines in different light and weather conditions.

Paper type: Research paper
https://doi.org/
Received: 18 December 2023; revised: 5 January 2024; accepted: 7 February 2024; published: 1 June 2024
How to cite this paper: Z. Dorrani, **"Deep Learning for Line Road detection in Smart Cars"**, *Majlesi Journal of Telecommunication Devices*, Vol. 13, No. 2, pp. 63-68, 2024.

A deep learning method for recognizing road features in advanced driver assistance systems is presented. This method uses two deep learning models, YOLOv7 and Faster R-CNN, to detect road types, traffic signs, and road lines [11].

Attention-based deep neural networks have also been used to detect road lines [12]. The attention mechanism allows the model to focus on important parts of the image that are likely to contain road lines. This method has improved the road lane detection performance compared to traditional methods that do not use attention.

It also uses deep reinforcement learning to recognize road contours [13]. Reinforcement learning is a type of machine learning that allows the agent to interact with its environment and learn through trial and error. In this research, a reinforcement learning agent has been used to learn how to recognize road lines from input images.

Another research has presented a new dataset of real roadway images for training and evaluating deep learning models for roadway detection [14]. This dataset contains high-resolution images and accurate labeling of road lines in different light and weather conditions.

3. YOLOV8 ARCHITECTURE

The YOLOv8 algorithm is a speedy object detection method that tackles the task in a single go. It's broken down into four main parts: the input segment, the backbone, the neck, and the output segment.

Prepping the Image (Input Segment): The first step involves some creative data manipulation on the input image. This is done through a technique called mosaic data augmentation, which combines information from several images into one. It also calculates anchors that best fit the objects in the image (adaptive anchor calculation) and adds grayscale information where needed (adaptive grayscale padding).

Extracting Features (Backbone and Neck): This is the core processing unit of YOLOv8. The image goes through a series of Conv and C2f modules. These C2f modules are an upgrade over the older C3 modules used in YOLOv7. They're the workhorses for extracting features from the image at various scales.

C2f modules are efficient thanks to the ELAN structure, which reduces the number of layers needed.

They also leverage Bottleneck modules to improve the flow of information within the network.

This approach keeps the model lightweight while allowing it to capture more intricate details.

Fusing the Features (Neck): After the feature extraction, the SPPF module comes into play. It uses pooling layers with different sizes to combine the extracted feature maps. These combined features are then passed to the neck layer.

The neck layer in YOLOv8 is a powerhouse for feature fusion. It borrows the FPN (Feature Pyramid Network) and PAN (Path Aggregation Network) structures to create a more robust model.

This structure merges high-level (global details) and low-level (local details) feature maps using upsampling and downsampling techniques. This allows the network to share information between features of different sizes, which is crucial for detecting objects of varying scales.

Making the Detection (Detection Head): The detection head follows a common approach, separating the classification head (identifies object type) from the detection head (locates the object). This stage involves calculating loss functions and filtering out irrelevant detection boxes.

A method called Task Aligned Assigner helps identify positive and negative samples during loss calculation. Positive samples are chosen based on a combination of classification and localization scores.

The loss calculation itself has two parts: classification loss and regression loss (excluding the Objectness branch from previous versions).

Classification loss uses Binary Cross-Entropy (BCE).

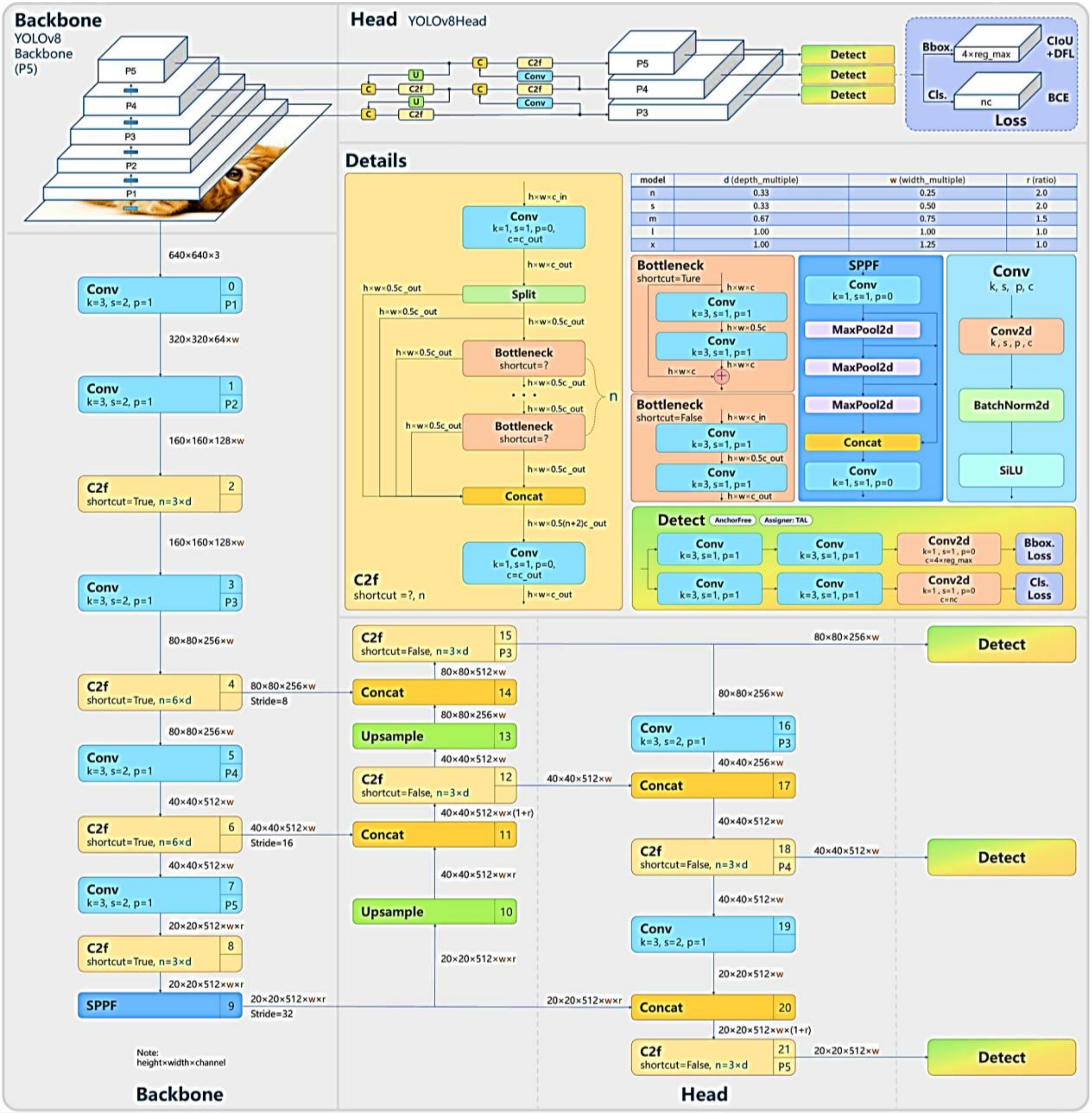
Regression loss employs a combination of Distribution Focal Loss (DFL) and CIoU loss functions.

YOLOv8 uses separate heads (decoupled heads) to predict classification scores (object type) and bounding box coordinates (object location) at the same time.

Classification scores are shown in a two-dimensional matrix, indicating the likelihood of an object existing in each pixel location.

Bounding box coordinates are represented by a four-dimensional matrix, specifying how far the object's center is from each pixel.

Finally, YOLOv8 uses a task-aligned assigner to evaluate how well these predictions align with the actual objects in the image. This metric considers both classification scores and Intersection over Union (IoU) values, allowing the network to optimize both object identification and localization simultaneously.



model	d (depth_multiple)	w (width_multiple)	r (ratio)
n	0.33	0.25	2.0
s	0.33	0.50	2.0
m	0.67	0.75	1.5
l	1.00	1.00	1.0
x	1.00	1.25	1.0

Fig. 1. YOLOv8 algorithm [15].

4. PROPOSED METHOD

Procedures for Extracting Road Line Features with YOLOV8:

A. Image Pre-processing: Car camera images undergo pre-processing before being fed into YOLOV8. This pre-processing includes resizing, noise removal [16], brightness enhancement, and removal of extraneous information.

B. Road Line Detection: YOLOV8 is trained to detect road lines within the pre-processed images. The model leverages deep learning algorithms to extract road line features such as position, direction, and lane type. By utilizing the backbone of this architecture, which consists of a series of convolutional layers, relevant features are extracted from the input image. SPPF processes features at different scales, allowing the model to detect road contours of varying sizes. C2f layers combine high-level features with background

Paper type: Research paper
https://doi.org/
Received: 10 January 2024; revised: 13 February 2024; accepted: 12 March 2024; published: 1 June 2024
How to cite this paper: S. Eskini, S. Karimi, **“Automated Diagnosis of Cardiac Diseases Using Machine Learning and Non-Stationary Heart Sound Signals”**, *Majlesi Journal of Telecommunication Devices*, Vol. 13, No. 2, pp. 69-85, 2024.

information, which improves detection accuracy. Finally, the final convolutional layer predicts the class and location of the lines.

C. Road Line Tracking: Following road line detection, YOLOV8 tracks these lines across consecutive images. This enables the automated driving system to comprehend the road geometry in real-time.

D. Post-processing: After tracking the road lines, the extracted information can be used to guide the car's behavior, such as making right or left turns, or continuing straight.

5. RESULT

The results of the simulation stage and how to extract the white and yellow lines are shown in Fig. 2.

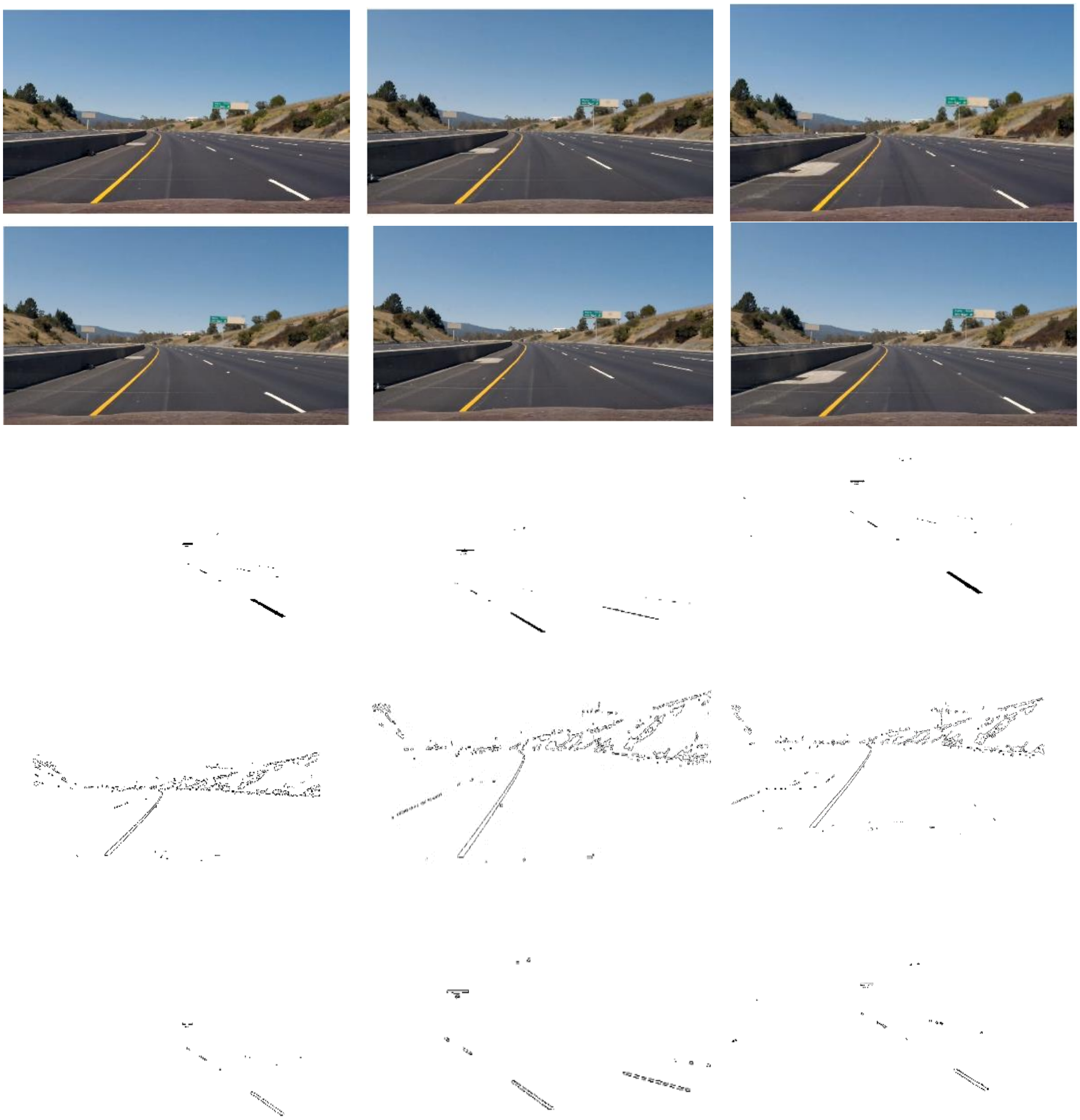


Fig. 2. Steps of the proposed method.

Lane markings play a vital role in identifying the designated path for vehicles on a road. They assist drivers in maintaining proper lane position, ensuring a safe distance from other vehicles, and determining the correct direction of travel.

Fig. 3 illustrates the results of road lane detection on dataset [17]. The red surface represents the detected lane area, and the green lines indicate the lane boundaries. However, achieving complete accuracy can be challenging due to uneven road surfaces and lane markings that may not be perfectly parallel.

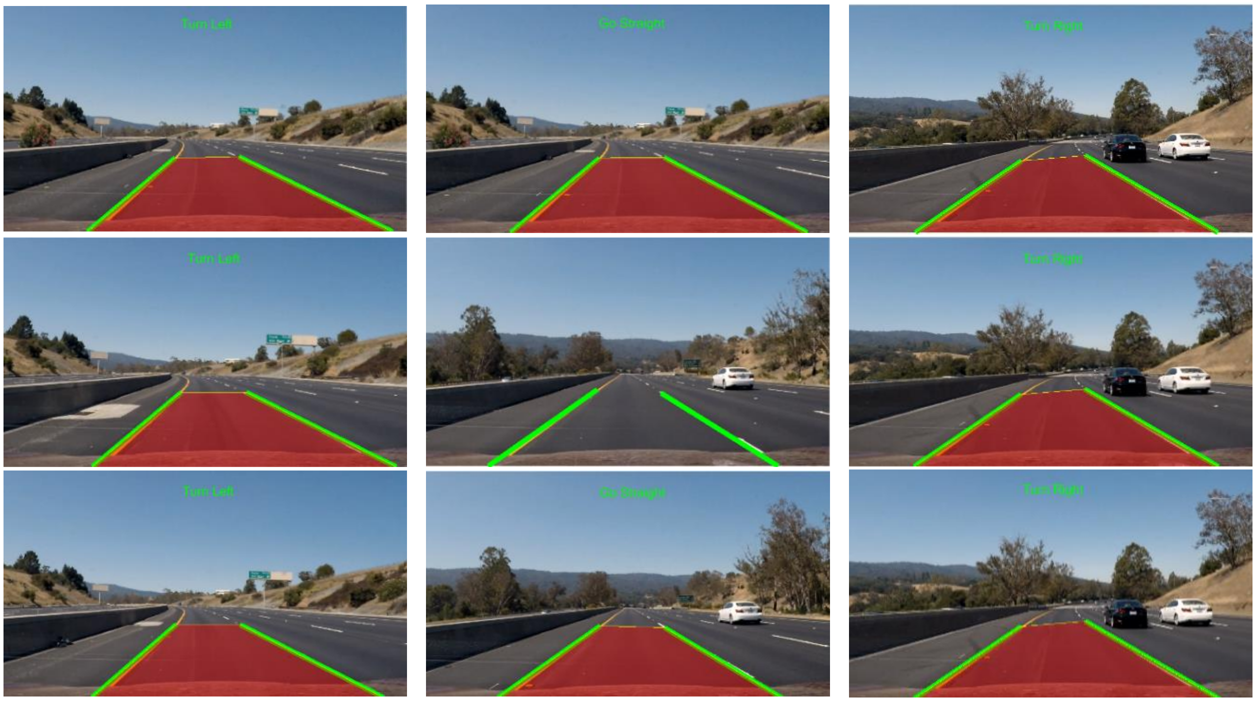


Fig. 3. lane line detection with proposed method.

To address this limitation, deep learning is employed. The YOLOv8 architecture is adopted, with adjustments made to the main channel and training process. It's important to note that if the weights of the initial layers are too small during loss propagation, the gradients may vanish within the network, hindering the learning process.

Line detection remains a crucial task for intelligent vehicles. However, environmental factors can sometimes obscure lane markings. These factors include shadows cast by vehicles or trees, poorly maintained lane markings, adverse weather conditions, and rough road surfaces. In such scenarios, the proposed scheme can be implemented to handle non-linear image features by extracting line segments.

The process involves pre-processing the image to remove noise and correct unsuitable background elements. Subsequently, line feature maps are extracted and sampled to classify the presence or absence of lane lines.

Table 1 compares the performance of the proposed method with several other methods in the field of detection accuracy. The first column shows the name of the method and the second column shows its accuracy.

Table 1. Performance evaluation of proposed method and comparison with some other methods.

Method	Accuracy
VGG16	89.91

Paper type: Research paper
https://doi.org/
Received: 10 January 2024; revised: 13 February 2024; accepted: 12 March 2024; published: 1 June 2024
How to cite this paper: S. Eskini, S. Karimi, **“Automated Diagnosis of Cardiac Diseases Using Machine Learning and Non-Stationary Heart Sound Signals”**, *Majlesi Journal of Telecommunication Devices*, Vol. 13, No. 2, pp. 69-85, 2024.

VGG19	90.24
YOLO	91.14
YOLOV5	91.86
YOLOV8	92.85

Based on this table, the proposed method with 92.85% accuracy provides the best performance among the 5 methods for detecting objects in images. YOLOV8 with 92.85% accuracy, YOLOV5 with 91.86% accuracy, YOLO with 91.14% accuracy, and VGG19 with 90.24% accuracy are in the next ranks respectively. VGG16 has the lowest accuracy with 89.91% accuracy.

6. CONCLUSION

Road lane detection using the YOLOv8 architecture is a promising method for autonomous driving systems. This method offers several advantages, including high speed, high accuracy, and generalizability. This paper presents a novel approach for lane detection in smart cars using deep learning. The approach utilizes the YOLOv8 algorithm to identify road lines. Simulation results demonstrate that the approach can accurately detect road lanes under a wide range of conditions. This approach can be a reliable source for creating driving assistance scenarios in natural road traffic environments.

For future research, the approach could be tested in real-world conditions to assess its performance in low-light conditions and the presence of various objects on the road. Additionally, developing more computationally efficient approaches for road lane detection remains an important area of research.

REFERENCES

[1] D.-H. Lee and J.-L. Liu, **"End-to-end deep learning of lane detection and path prediction for real-time autonomous driving,"** Signal, Image and Video Processing, vol. 17, no. 1, pp. 199-205, 2023.

[2] A. Arshaghi and M. Norouzi, **"A Survey on Face Recognition Based on Deep Neural Networks,"** Majlesi Journal of Telecommunication Devices, 2023.

[3] A. A. Abed and M. Emadi, **"Detection and Segmentation of Breast Cancer Using Auto Encoder Deep Neural Networks,"** Majlesi Journal of Telecommunication Devices, vol. 12, no. 4, pp. 209-217, 2023.

[4] N. Habibi and S. Mousavi, **"A Survey on Applications of Machine Learning in Bioinformatics and Neuroscience,"** Majlesi Journal of Telecommunication Devices, vol. 11, no. 2, pp. 95-111, 2022.

[5] Z. Dorrani, "**Traffic scene analysis and classification using deep learning,"** International Journal of Engineering, 2023.

[6] M. Ghasemzade, **"Extracting Image Features Through Deep Learning,"** Majlesi Journal of Telecommunication Devices, vol. 9, no. 3, pp. 109-114, 2020.

[7] Z. Dorrani, H. Farsi, and S. Mohamadzadeh, **"Deep Learning in Vehicle Detection Using ResUNet-a Architecture,"** Jordan Journal of Electrical Engineering. All rights reserved-Volume, vol. 8, no. 2, p. 166, 2022.

[8] Z. Dorrani, H. Farsi, and S. Mohammadzadeh, **"Edge Detection and Identification using Deep Learning to Identify Vehicles,"** Journal of Information Systems and Telecommunication (JIST), vol. 3, no. 39, p. 201, 2022.

[9] Z. Dorrani, **"Road Detection with Deep Learning in Satellite Images,"** Majlesi Journal of Telecommunication Devices, vol. 12, no. 1, pp. 43-47, 2023.

[10] S.-W. Baek, M.-J. Kim, U. Suddamalla, A. Wong, B.-H. Lee, and J.-H. Kim, **"Real-time lane detection based on deep learning,"** Journal of Electrical Engineering & Technology, vol. 17, no. 1, pp. 655-664, 2022.

[11] H. Nadeem et al., "**Road feature detection for advance driver assistance system using deep learning,"** Sensors, vol. 23, no. 9, p. 4466, 2023.

[12] E. Oğuz, A. Küçükmanisa, R. Duvar, and O. Urhan, "**A deep learning based fast lane detection approach,"** Chaos, Solitons & Fractals, vol. 155, p. 111722, 2022.

[13] Z. Zhao, Q. Wang, and X. Li, "**Deep reinforcement learning based lane detection and localization,"** Neurocomputing, vol. 413, pp. 328-338, 2020.

[14] N. J. Zakaria, M. I. Shapiai, R. Abd Ghani, M. N. M. Yassin, M. Z. Ibrahim, and N. Wahid, **"Lane detection in autonomous vehicles: A systematic review,"** IEEE access, vol. 11, pp. 3729-3765, 2023.

[15] L. Zhang, G. Ding, C. Li, and D. Li, **"DCF-Yolov8: An Improved Algorithm for Aggregating Low-Level Features to Detect Agricultural Pests and Diseases,"** Agronomy, vol. 13, no. 8, p. 2012, 2023.

[16] Z. Dorrani and M. Mahmoodi, **"Noisy images edge detection: Ant colony optimization algorithm,"** Journal of AI and Data Mining, vol. 4, no. 1, pp. 77-83, 2016.

[17] Detection of lanes on a road and prediction of turns based on vanishing point. https://github.com/ysshah95/Lane-Detection-using-MATLAB.

Automated Diagnosis of Cardiac Diseases Using Machine Learning and Non-Stationary Heart Sound Signals

Sadaf Eskini[1], Salman Karimi[2]
1, 2- Department of Electrical Engineering, Lorestan University, Khorramabad, Iran
Email: karimi.salman@lu.ac.ir (Corresponding author)

ABSTRACT:
The objective of this research is to employ machine learning techniques for the accurate diagnosis of cardiac diseases by leveraging a combination of diverse features. This study introduces an automated methodology that involves the analysis of non-stationary heart sound signals to effectively identify various heart conditions. In contrast to conventional approaches relying on techniques such as analysis of variance, signal average or standard deviation comparison, the proposed diagnosis is primarily based on diagnostic labels, ensuring a more reliable and robust assessment. The integrated system proposed in this study utilizes the renowned MIT-BIH database to analyze heartbeats and discern different functional states. Training and testing of the data are performed using the K-fold cross-validation procedure, and a novel model is employed to enhance the learning process. Through this innovative diagnostic system, the detection of cardiac abnormalities in electrocardiogram (ECG) signals is achieved with an impressive accuracy ranging from 96% to 99% across a broad spectrum of cases. By harnessing the power of machine learning algorithms and leveraging a comprehensive set of features, this research significantly advances the field of cardiac disease diagnosis. The proposed methodology outperforms traditional approaches by providing a more accurate and efficient means of identifying heart conditions. The utilization of diagnostic labels as the basis for diagnosis ensures enhanced reliability, enabling healthcare professionals to make informed decisions regarding patient care. Ultimately, this research contributes to the ongoing efforts to improve cardiac healthcare, enabling early detection and intervention, and potentially saving numerous lives.

KEYWORDS: Heart Diseases, Machine Learning, Heart Rate Analysis, MIT-BIH Database.

1. INTRODUCTION

Cardiovascular diseases (CVDs) are a significant cause of mortality worldwide, with an estimated 17 million deaths globally due to CVDs according to a 2005 report by the World Health Organization (WHO) [1]. CVDs are also responsible for an estimated 30% of all deaths worldwide [1]. Within this group, approximately 7.2 million deaths are attributed to coronary heart disease (CHD), and 5.7 million deaths are attributed to stroke [1]. By 2030, it is projected that approximately 23.6 million people will die from CVDs [2].

Healthcare professionals typically diagnose heart disease manually by interpreting electrocardiogram (ECG) signals, but recent technological breakthroughs have led to the development of multiple automated diagnostic tools for arrhythmia classification and diagnosis by physicians [3-4]. Some studies in the literature have suggested that heart rate could be a suitable criterion for diagnosis. Khan et al. [5, 6] provided a comprehensive introduction to the research conducted in this field and proposed deep learning (DL)-based methods for classifying three types of arrhythmias. DL has become a practical tool in medical settings, particularly for classifying cardiac arrhythmias, with several studies investigating one-dimensional (1D), two-dimensional (2D), and/or merged 1D/2D deep convolutional neural networks (CNN) [7, 8]. For example, Xiao et al. [7] proposed a novel arrhythmia classification technique that involves pre-processing, a 1D deep CNN using a block-stacked style architecture including clique and transition blocks, and an

Paper type: Research paper
https://doi.org/
Received: 10 January 2024; revised: 13 February 2024; accepted: 12 March 2024; published: 1 June 2024
How to cite this paper: S. Eskini, S. Karimi, **"Automated Diagnosis of Cardiac Diseases Using Machine Learning and Non-Stationary Heart Sound Signals"**, *Majlesi Journal of Telecommunication Devices*, Vol. 13, No. 2, pp. 69-85, 2024.

attention mechanism and majority voting decision strategy for prediction. However, their experiments did not account for low-frequency noise recorded in arrhythmic beats, and the 2D representation requires additional computations that are not feasible without building and adjusting a large set of hyperparameters. In a similar study, Noman et al. [8] proposed a framework based on a 1D-CNN that directly learns features from raw heart-sound signals and a 2D-CNN that takes inputs of 2D time-frequency feature maps. Despite the numerous advantages of DL, these networks require feeding with an immense volume of input data, and proper decision-making in these networks relies on the adjustment of multiple parameters.

Automated systems for arrhythmia diagnosis have been developed in recent research [9, 10], but these systems are still undergoing strict pre-approval evaluations by healthcare professionals. Other proposed systems suffer from learning challenges and uncertainty [11, 12]. The development of arrhythmia detection systems requires overcoming fundamental issues, such as manual feature selection, feature extraction techniques, and classification algorithms, particularly when using unbalanced data for classification. Extracting features from ECG signals for automatic arrhythmia diagnosis requires an immense volume of data and information. Afkhami et al. [13] proposed a novel method for accurately classifying cardiac arrhythmias, utilizing two inter-beat (RR) interval features as time-domain information exemplars. They also used Gaussian mixture modeling (GMM) with an enhanced expectation maximization (EM) solution to fit the probability density function of heartbeats. In addition, GMM parameters and shape parameters (e.g., skewness, kurtosis, and 5th moment) were included in the feature vector, which was then used to train an ensemble of decision trees. Mathunjwa et al. [9] designed a new DL method for effective arrhythmia classification using 2-second segments of 2D recurrence plot images of ECG signals. Marinho et al. [14] proposed a novel approach to detect cardiac arrhythmias in ECG signals. In [15], an improved CNN called the modified visual geometry group network (mVGGNet) was introduced for automatic heart-abnormality classification using ECG signals. Ref. [16] combined three groups of features for arrhythmia classification, including RR distances, signal morphology, and higher-order statistics (HOS), and validated the proposed method using the MIT-BIH database based on an inter-patient paradigm. The robustness of each group of features against classification faults was also investigated. Despite the promising results of these studies, automated arrhythmia diagnosis systems still need to overcome challenges such as learning difficulties and unbalanced data for classification.

The proposed jitter-based classification system demonstrated a sensitivity of 93.7%, 89.7%, and 87.9% for N, S, and V classes, respectively, according to the experimental results. Kaya et al. [17] used long short-term memory (LSTM) neural networks to classify ECG signals by combining LSTM and angle transform (AT) methods. Rahul et al. [18] proposed an improved RR interval-based cardiac arrhythmia classification approach that utilized the discrete wavelet transform (DWT) and median filters to remove high-frequency noise and baseline wander from the raw ECG. Lee et al. [19] proposed a beat–interval–texture convolutional neural network (BIT-CNN) model for arrhythmia classification by transforming variable-length 1D ECG signals into fixed-size 2D time-morphology representations. They learned comprehensible characteristics of beat shape and inter-beat patterns over time. Zhang et al. [20] proposed a Multi-Lead-Branch Fusion Network (MLBF-Net) architecture for arrhythmia classification by integrating multi-loss optimization to jointly learn the diversity and integrity of multi-lead ECG. The experimental results showed that MLBF-Net achieved the highest arrhythmia classification performance, with an average equation F_1 score of 0.855. Rahul et al. [21] proposed a technique for classifying lethal CVDs, such as atrial fibrillation (Afib), ventricular fibrillation (Vfib), ventricular tachycardia (Vtec), and normal (N) beats.

In this study, an automated method for analyzing non-stationary heart sound signals to identify disease-related classes is proposed. Unlike other methods that do not consider the non-stationarity of the signal, this study uses signal windowing to address this issue. By combining features obtained from analyzing cardiac signals, the diagnosis and classification of cardiac arrhythmias can be improved. The proposed feature extraction structure integrates features from the time and frequency domains based on statistical and fractal indicators, using only two leads. The structure also employs a decision tree for learning. However, the contribution of frequency and time domain features to classification accuracy may differ depending on the feature extraction method. Therefore, an appropriate feature extraction process is crucial for achieving a comprehensive method for combining and classifying features. Additionally, a multi-core overlearning model using group learning methods can address challenges such as overfitting and uncertainty, while also reducing computational complexity. The primary goal of this study is to diagnose heart disease and reduce false and negative faults, thereby improving evaluation criteria.

2. ELECTROCARDIOGRAPHY

Accurate and consistent evaluation of arrhythmias by cardiologists is crucial for effective diagnosis, but it can be challenging and time-consuming. Therefore, automated detection systems are necessary for the accurate identification of cardiovascular diseases. Sophisticated diagnostic systems can assist cardiologists in precise and rapid diagnosis of ECG recordings, reducing the time and cost of clinical interpretation. In recent years, various machine learning-based models have been proposed for the analysis of cardiac signals to detect different types of arrhythmias. An ECG is a machine that records the heart's electrical activity as a graph, where electrodes are placed at specific spots on the skin to record the electrical impulses generated by the heartbeat. Each recorded ECG graph contains three distinct waves, namely the P wave, the T wave, and the QRS complex. Researchers have used cardiac signals and analyzed various data to propose a range of techniques for identifying heart diseases and classifying the heart's diverse functional states. The developed models are based on feature extraction, feature selection, and classification. Fig. 1 shows segments of a 2s 1D ECG signal for six classes and their corresponding RPs, including data on AF rhythm, normal beats, PAC beats, PVC beats, VF, and noise. As asserted by them, this is the first report of using SCM to extract features with ECG signals.

3. CVDs

According to the World Health Organization (WHO), cardiovascular diseases (CVDs) account for 32% of all global deaths [22]. Congestive heart failure (CHF) is a severe cardiac disorder and a leading cause of mortality worldwide. In CHF, the heart muscle fails to pump enough blood to maintain the bloodstream and meet the needs of body tissues for oxygen and metabolism. Globally, over 26 million adults suffer from CHF, and its prevalence is increasing by 3.6 million annually [23]. However, early diagnosis of CHF can significantly improve treatments and prevent disease progression. Arrhythmias are the next leading heart disorder responsible for several cases of sudden cardiac death (SCD). An arrhythmia refers to an abnormality of the heart's rhythm caused by irregular heartbeats.

Various non-invasive and invasive methods can be used to diagnose CVDs. Invasive methods are available, but they are costly and uncomfortable, particularly in remote areas. In contrast, early diagnosis of CVDs using non-invasive methods is cheap and painless. ECG and phonocardiograph (PCG) signals are among the non-invasive techniques used to diagnose CVDs. However, cardiologists require the necessary equipment and facilities for heart monitoring to perform respective analyses, which are not easily accessible, particularly in remote areas [24].

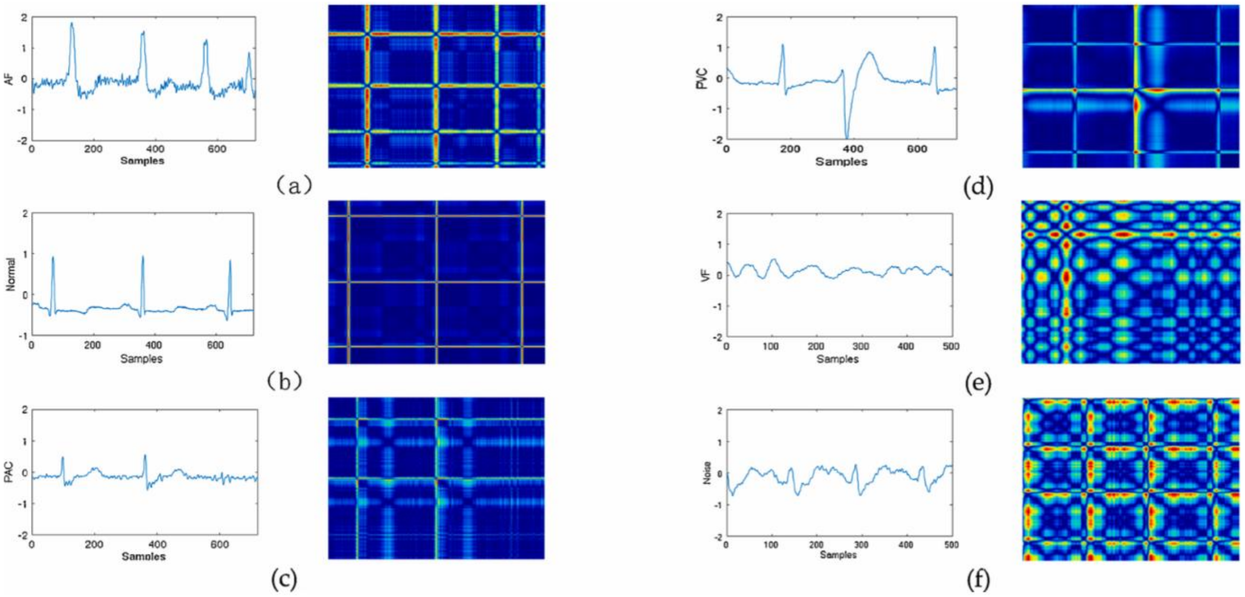


Fig. 1. Segments of the signal for a 2s normal state for six classes and relevant RPs. The 2s segments are shown for AF (a), normal beats (b), PAC beats (c), PVC beats (d), VF (e), and the likely noise (f). The difference in sample size between ECG segments are emanated from various sampling rates for the databases [9].

An ECG is a non-invasive test that monitors and records the heart's function by detecting the electrical activity of the heart muscles. ECGs provide essential data on heart diseases to cardiologists, making them efficient tools for the identification of various CVDs.

4. MIT-BIH DATABASE

This study used data from the MIT-BIH database to conduct experiments. This database includes various common and life-threatening arrhythmias and comprises 48 ambulatory ECG records, each lasting 30 minutes, obtained from 47 patients. Each record consists of two distinct scenarios, where for 45 records, the first lead is the MLII (modified limb lead), and for the remaining records, the lead is modified V5. The second lead is pericardial, i.e., V1 (for 40 records) and V2, V4, or V5 (for the remaining records). For this study, 23 records were randomly selected from 4000 ambulatory ECG records in the database collected from a diverse population of inpatients. The remaining 25 records were selected from the same collection, featuring less common but clinically significant arrhythmias that are not well-represented in a small random set [14]. Table 1 shows the main labeling of the database, including 16 classes of rhythms.

Table 1. Data from the MIT-BIH arrhythmia database.

No.	Type of heartbeat	Abbreviation	Signal annotation	Total
1	Normal rhythm	NOR	N	74607
2	Left bundle branch block	LBBB	L	8069
3	Right bundle branch block	RBBB	R	7250
4	Atrial premature contractions	APC	A	2514
5	Premature ventricular contraction	PVC	V	7127
6	Heartbeat	PB	/	7020
7	Premature atrial abnormality	AP	a	150
8	Ventricular Fibrillation	VF	!	472
9	Integrated ventricular beat	VFN	F	802
10	Non-conductive P wave	BAP	x	193
11	Nodal escape	NE	X	229
12	Merged fast and normal beat	FPN	J	982
13	Ventricular escape	VE	F	106
14	Nodal escape beat	NP	E	83
15	Atrial escape beat	AE	J	16
16	Unclassified	UN	E	38
Total	-	-	Q	109655

The classes were labeled, and the results of cardiac arrhythmia classification algorithms were represented following the standards set out by the Association for the Advancement of Medical Instrumentation (AAMI). AAMI defines five clinically relevant classes as "N" (sinus node beats), "S" (supraventricular ectopic beat), "V" (ventricular aberrant beats), "F" (fusion beats), and "Q" (unclassified hits). Table 2 maps the classes of the MIT-BIH arrhythmia database to AAMI heart rate classes. Additionally, this table contains plans to train data division and the test set (DS1 and DS2, respectively) [2] for subject-oriented classification. This plan carefully splits the dataset so that the training and test samples are equally distributed among the five mentioned classes, and it also creates false-negative (FN) for four speed records (i.e., 102, 104, 107, and 217) recommended by the AAMI. All 48 records and 16 MIT-BIH annotation classes were used in the class-oriented plan, while the subject-oriented plan employed 44 non-rhythmic records with 5 classes in Table 2 recommended by AAMI.

Table 2. AAMI considering the label of five classes for classification.

No.	AAMI class	MIT-BIH class	Total
1	N	NOR, LBBB, RBBB, AE, NE	89665
2	S	APC, AP, BAP, NP	2940
3	V	PVC, VE, VF	7478
4	F	VFN	802
5	Q	FPN, UN	17

5. THE PROPOSED METHOD

Generally, this research is comprised of three fundamental sections: signal windowing, feature extraction, and classification.

a) Signal Windowing: The necessity addressed here is overcoming the non-stationarity of cardiac signals. This issue holds significant importance in improving the accuracy of the examined system. To extract information from the signal, it needs to be divided into short segments in a manner that, according to signal processing sciences, ensures the time required to generate a cardiac complex, which is at least 500 seconds. Essentially, the goal is to apply processing to segments that are sufficiently short, preventing temporal changes in the properties of cardiac signals within them.

Considering the overlap between frames, the impact of discontinuities in the signal decreases, and this is a crucial step in improving accuracy, which we have addressed in this research. While some researchers do not use signal windowing and apply feature extraction directly to the entire signal, this approach leads to a reduction in classification accuracy.

b) Feature Extraction: In the feature extraction section, various features are employed to describe the signal, commonly used in vital signals and cardiac signals. The following features are introduced in order, encompassing time and frequency domain features. In (1), the signal integral is represented, where x denotes the input signal and N is the length of the signal.

$$I_{ECG} = \sum_{i=1}^{N} |x_i| \tag{1}$$

In equations (2) to (5), the absolute mean value of the signal, the absolute value of third-order time moments, the absolute value of fourth-order time moments, and the absolute value of fifth-order time moments are respectively represented.

$$MAV = \frac{1}{N}\sum_{i=1}^{N} |x_i| \tag{2}$$

$$TM3 = \frac{1}{N}\sum_{i=1}^{N} |x_i^3| \tag{3}$$

$$TM4 = \frac{1}{N}\sum_{i=1}^{N} |x_i^4| \tag{4}$$

$$TM5 = \frac{1}{N}\sum_{i=1}^{N} |x_i^5| \tag{5}$$

In the set of equations (6) to (9), the signal variance, root mean square, waveform length, and zero-crossings are respectively represented.

$$VAR = \frac{1}{N-1}\sum_{i=1}^{N} |x_i| \tag{6}$$

$$RMS = (\frac{1}{N}\sum_{i=1}^{N} x_i^2)^{0.5} \tag{7}$$

$$WL = \sum_{i=1}^{N-1} |x_{i+1} - x_i| \tag{8}$$

$$ZC = \sum_{i=1}^{N-1} [\operatorname{sgn}(x_i \times x_{i+1}) \cap |x_i - x_{i+1}| \geq threshold]$$

$$f(\mathrm{x}) = \begin{cases} 1 & if\ x \geq threshold \\ 0 & otherwise \end{cases} \tag{9}$$

In the set of equations (10) to (13), the mean absolute mean deviation, autoregressive coefficients, V order moments, and separable and simple square integrals are respectively represented.

$$\mathrm{MAVSLP}_{\mathrm{k}} = \mathrm{MAV}_{\mathrm{k+1}} - \mathrm{MAV}_{\mathrm{k}}\ ;\ k = 1,2,\ldots k-1 \tag{10}$$

$$\mathrm{x}_{\mathrm{i}} = \sum_{\mathrm{p=1}}^{\mathrm{P}} \mathrm{a}_{\mathrm{p}} \mathrm{x}_{i-p} + \mathrm{w}_{\mathrm{i}} \tag{11}$$

$$\mathrm{x}_{\mathrm{i}} = \left(\gamma \mathrm{m}_{\mathrm{i}}^{\alpha}\right)\mathrm{n}_{\mathrm{i}}$$

$$\mathrm{V} = \left(\sum_{i=1}^{N} \mathrm{x}_{\mathrm{i}}^{v}\right)^{\frac{1}{v}} \tag{12}$$

$$\mathrm{SSI} = \sum_{i=1}^{N} x_i^2 \tag{13}$$

Similarly, in equations (14) to (16), the average range of change, the difference in absolute standard deviation, and finally, the Wilson amplitude range are respectively presented.

$$\mathrm{AAC} = \frac{1}{\mathrm{N}} \sum_{i=1}^{N-1} \left| x_{i+1} - x_i \right| \tag{14}$$

$$\mathrm{DASDV} = \sqrt{\frac{1}{\mathrm{N}} \sum_{i=1}^{N-1} \left(x_{i+1} - x_i \right)^2} \tag{15}$$

$$\mathrm{WAMP} = \sum_{i=1}^{N-1} \left[\mathrm{f}(\left| x_i - x_{i+1} \right|) \right]$$

$$f\left(\mathrm{x}\right) = \begin{cases} 1 & if\ x \geq threshold \\ 0 & otherwise \end{cases} \tag{16}$$

Continuing, frequency domain features are introduced for the analysis of cardiac signals. From (17) to (22), frequency domain features, including mean frequency, median frequency, peak frequency, average power, final power, and frequency ratio, are respectively presented.

$$\mathrm{MNF} = \sum\nolimits_{\mathrm{i=1}}^{N} \mathrm{f}_i\ P_i \Big/ \sum\nolimits_{i=1}^{N} P_i \tag{17}$$

$$\sum_{\mathrm{i=1}}^{\mathrm{TTP}} P_i = \sum_{i=\mathrm{MDF}}^{\mathrm{N}} P_i = \sum_{\mathrm{i=1}}^{N} P_i \tag{18}$$

$$PKF = \max\left(p_i\right).\ i = 1,2,\ldots N \tag{19}$$

$$\mathrm{MNP} = \frac{1}{M} \sum_{\mathrm{i=1}}^{N} P_i \tag{20}$$

$$\mathrm{TTP} = \sum_{\mathrm{i=1}}^{N} P_i = \mathrm{SMO} \tag{21}$$

$$\mathrm{FR} = \sum\nolimits_{\mathrm{i=LLC}}^{\mathrm{ULC}} P_i \Big/ \sum\nolimits_{i=\mathrm{LHC}}^{\mathrm{UHC}} P_i \tag{22}$$

In addition to the aforementioned features, we also utilize fractal dimension features. These features are generally considered among the important features that reflect the oscillations and intricacies within the signal, with a high calculation speed. Among these features, known as fractal dimension features, are included Katz dimension, Higuchi dimension, Petrosian dimension, correlation dimension, and other similar measures. Equations (23) and (24) describe the signal's correlation dimension, where we have N points in an M-dimensional space.

$$Z\left(i\right) = \left[Z_1\left(i\right).Z_2\left(i\right)\ldots.Z_m\left(i\right)\right]\quad i = 1,2,\ldots,N \tag{23}$$

Then, the correlation integral C(ε) is calculated according to equation (24).

$$C(\varepsilon) = \lim g / N^2 (N \to \infty) \tag{24}$$

Where g is the total number of pairs of points with distances less than ε. As the number of points tends to infinity and the distance between them approaches zero, the correlation integral for small values of ε is expressed as in equation (25).

$$C(\varepsilon) \sim (\varepsilon) \wedge v \tag{25}$$

In the fractal method, the estimation of the fractal dimension is carried out for a new series of input data x according to the equation (26).

$$x = \{x(1), x(2) \ldots . x(N)\} \tag{26}$$

Where:

$$x_m^k = \left\{ x(m).x(m+k).x(m+2k) \ldots . x\left(m + \left[\frac{N-m}{k} \right] k \right) \right\} \tag{27}$$

Where m = 1, ..., k represents the starting point of each series, k is the distance between samples, and $\left[\frac{N-m}{k}\right]$ is the integer part of each number. Therefore, for each m, we have the length L_m(k).

$$L_m(\mathrm{k}) = \frac{\sum \left| x(m+ik) - x(m+(i-1)k) \right| (N-1)}{\left[\frac{N-m}{k} \right] k} \tag{28}$$

In this equation, N represents the number of required samples, and for each value of K, K lengths will be estimated. Then, the average is calculated as the component of the mean length. This process is repeated up to K_{max}, and if L(k) is obtained for the next k^{-1}, Higuchi is then used to form the best approximation line with the least squares error in Log(L(k)) versus Log(k^{-1}).

c) **Classification**: Classification in the network is performed by considering a suitable classifier, initial training of the network, and finally testing it for the detection of cardiac arrhythmias. The classification utilizes an ensemble method based on a decision tree. There are various ways to combine predictions of a group, or bagging, but the most common method used in the research is majority voting. The main causes of error in learning stem from noise, bias, and variance. This ensemble approach helps minimize these factors. The combination of multiple classifiers reduces variance. To use bagging and boosting, we choose a base learner algorithm. For instance, if we select a classification algorithm, bagging and boosting would involve an ensemble of trees as large as desired. Bagging and boosting methods create N learning patterns by generating additional data during the training phase. New training datasets are produced by random sampling with replacement from the original dataset. With sampling and replacement, some observations in each new training dataset may be repeated. In the case of bagging, each element has an equal probability of being present in the new datasets. To enhance the method, observations are assigned weights, so some of them participate more in the new datasets. In bagging, for training data with size D and dimension n, the model creates a new training sample set Di with size n' by sampling from D uniformly and with replacement. These types of samples are essentially bootstrap samples. Next, m models are created using m bootstrapped samples, and their outputs are combined by averaging in regression or majority voting in classification. This method can be a combination of numerous classifiers, each utilized in the bagging process. However, decision tree-based learning methods are efficient and can be beneficial in ensemble methods like bagging, even in semi-supervised settings. Bootstrap aggregation, also known as bagging, leverages the aggregation of inputs in a way that an algorithm is used to create stability and improve accuracy, typically applicable in statistical classification and regression processes. This approach not only improves accuracy but also reduces variance and dispersion, aiding in preventing overfitting. While decision tree-based methods are commonly used in bagging, this technique is applicable with various methods. In the stages of bootstrap aggregation in bagging, with sampling and replacement, some observations may be repeated in each *Di*. In a scenario where it can be assumed that a set of observations is independently and equally distributed from an accessible population, bootstrap can be implemented by creating multiple subsamples, where each of these subsamples consists of randomly selected observations with replacement from the original dataset. The use of the bootstrap method in bagging can significantly contribute to improving accuracy and other factors such as variance and dispersion. However, in decision tree-based learning, it essentially involves modeling using one of the prediction model methods that utilize a decision tree (as a predictive model) to draw conclusions about the target value of a case (as presented in the leaves) from observations related to that case (discussed in the subsections). This model, where the target variable can have a set of discrete values, is referred to as decision trees in classification. In these tree structures, the leaves represent class labels, and branches represent combinations of features that lead to the assignment of class labels. Decision trees in which the target variable can have continuous values, typically real numbers, are called regression trees. The predictions of decision tree models are aggregated to create a final combined prediction. This aggregation can be based on the predictions made by individual bootstrapped models or the probability of the predictions made by the individual models. The main drawback of the bagging method is that it does not allow for fine-tuning the aggregation of learner models. If not modeled properly, it may lead to overfitting and become a challenging decision-making problem. Another weakness is that while bagging provides higher accuracy, it can be computationally time-consuming and resource-intensive, making it less desirable for large datasets depending on the use case. The main advantage of the bagging or bootstrap aggregation method used in this study has been its ability to improve the accuracy of a model without risking an increase in its variance. This makes it a suitable choice for situations where we want to reduce the variance of predictions without sacrificing a significant amount of accuracy. The precise and automated identification of the presence or absence of a disease based on cardiac signal analysis in the algorithm set forth in this study can contribute to establishing an effective connection between individuals and healthcare professionals. It can also help prevent a reduction in the error rate in the design of assistive systems and assess changes in error rates. Through proper differentiation of binary or multi-class states, the system could estimate the severity of the disease based on cardiac signals among samples or individuals, fostering more accurate diagnoses.

Overall, for cardiac signals, the accuracy of the proposed method is more than 98% using 30 segment division and 40% overlap. Also, the overlapping situations less than 20% and more than 30% are well analyzed.

In other studies, the issue of uncertainty has not been addressed, but in this study, by plotting the ROC curve, it is possible to make a statement about this issue to a large extent. Therefore, in other experiments, the criterion of the area under the curve was calculated and estimated, which was done for the experimental data and also for the credit data, respectively.

Previous research did not address the non-stationary nature of the ECG signal, but we have addressed it in this research, and to overcome this problem, we have used windowing, which is very effective in improving accuracy. Also, by considering the overlap between frames, the effect of discontinuous points on the signal is reduced, which is an important step in improving accuracy.

In previous studies, techniques such as variance analysis, mean comparison, or standard deviation were used to diagnose the disease, but this study is based on diagnostic labels.

Furthermore, in other studies, the issue of uncertainty has not been addressed. However, in this research, we have addressed this matter by utilizing ROC curves and calculating the area under the curve to overcome uncertainty.

In general, the use of signal windowing involves creating input structures from signals, reducing the length of windowed signals, diminishing the dimensions of extracted features, improving processing time, and enhancing accuracy. This is considered an innovative aspect of the research**.**

The proposed method in this study consists of three main parts, as illustrated in Fig. 2: 1) windowing of various types of cardiac signals, including different cardiac arrhythmias, 2) feature extraction, and 3) final classification. The algorithm used in this study was simulated in MATLAB (version 2022) on a computer with a Windows 10 operating system for both the learning section and the splitting of the signal into equal frames. The hardware used was conFig.d with an Intel® Core™ i7-7500U Processor (Processor Base Frequency 2.70 GHz; 4GB RAM). All processes were performed offline. The data were split into several stages, where the first stage involved dividing the data into 80% (operational) and 20% (unseen) data groups.

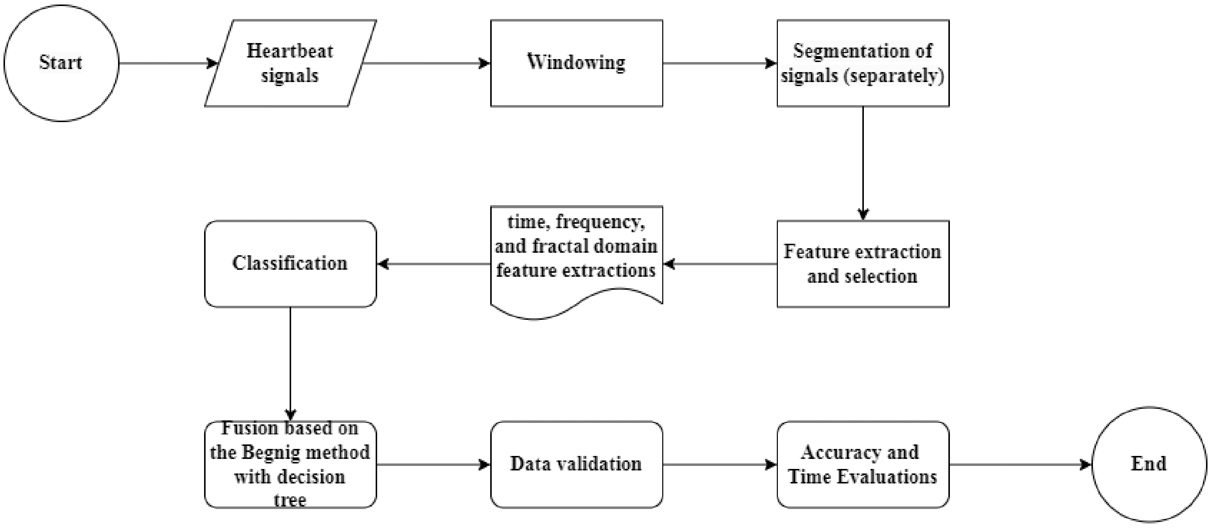

Start
Heartbeat signals
Windowing
Segmentation of signals (separately)
Feature extraction and selection
time, frequency, and fractal domain feature extractions
Classification
Fusion based on the Begnig method with decision tree
Data validation
Accuracy and Time Evaluations
End

Fig. 2. Schematic representation of the proposed method in recognizing cardiac arrhythmias from the ECG signals.

The 20% of data mentioned in the final stage was used to estimate the performance of the final model obtained. The data were split using the K-fold validation method with K=5, where two groups of data (training and testing) were used for the evaluation process. The training data were included in the model, and then re-split into new training data and validation data using the K-fold method to adjust the fusion classification appropriately. The classification errors were estimated based on the obtained features and the application of validation data. Fault calculation was also performed based on the validation data. After that, the parameters were transferred to the model and evaluated based on the testing data 2. The built model and the unseen data were eventually applied to the model. To determine the best window length, the length of the windows was varied from 800 ms to 2500 ms, considering the properties of the cardiac signal to cover at least one cardiac cycle. The rate of overlap between two consecutive frames was measured by balancing accuracy and time.

6. RESULTS

To evaluate the effectiveness and success of the proposed model, the confusion matrix and the ROC curve (receiver operating characteristic curve) were used. The confusion matrix was used to analyze the classification and identification of the target class and other classes, and it was estimated by examining various states when diagnosing the disease or an abnormal heart condition in the cardiac signal. The confusion matrix consists of four moods, including true positive (TP), true negative (TN), false positive (FP), and false negative (FN), each having a specific meaning. TP is the number of signals or segments of the signals that indicate the presence of a cardiac abnormality, while TN is the number of signals or segments of the signals that do not indicate the presence of a cardiac abnormality. FP refers to the number of signals or segments of the signals that indicate the presence of a cardiac abnormality, but the simulation software has wrongly determined the presence of that class of cardiac abnormality. FN refers to the number of signals or segments of the signals that indicate the presence of a cardiac abnormality, but the simulation software has wrongly determined the absence of that class of cardiac abnormality. Based on these definitions, the accuracy, sensitivity, and specificity values were estimated using equations (29) to (31).

$$\mathit{Accuracy}\text{: } \left(\mathrm{N}_{\mathrm{TP}}+\mathrm{N}_{\mathrm{TN}}\right)/\left(\mathrm{N}_{\mathrm{TP}}+\mathrm{N}_{\mathrm{TN}}+\mathrm{N}_{\mathrm{FP}}+\mathrm{N}_{\mathrm{FN}}\right) \tag{29}$$

$$\mathit{Sensitivity}\text{: } \left(\mathrm{N}_{\mathrm{TP}}\right)/\left(\mathrm{N}_{\mathrm{TP}}+\mathrm{N}_{\mathrm{FN}}\right) \tag{30}$$

$$\mathit{Specificity}\text{: } \left(\mathrm{N}_{\mathrm{TP}}\right)/\left(\mathrm{N}_{\mathrm{TP}}+\mathrm{N}_{\mathrm{FN}}\right) \tag{31}$$

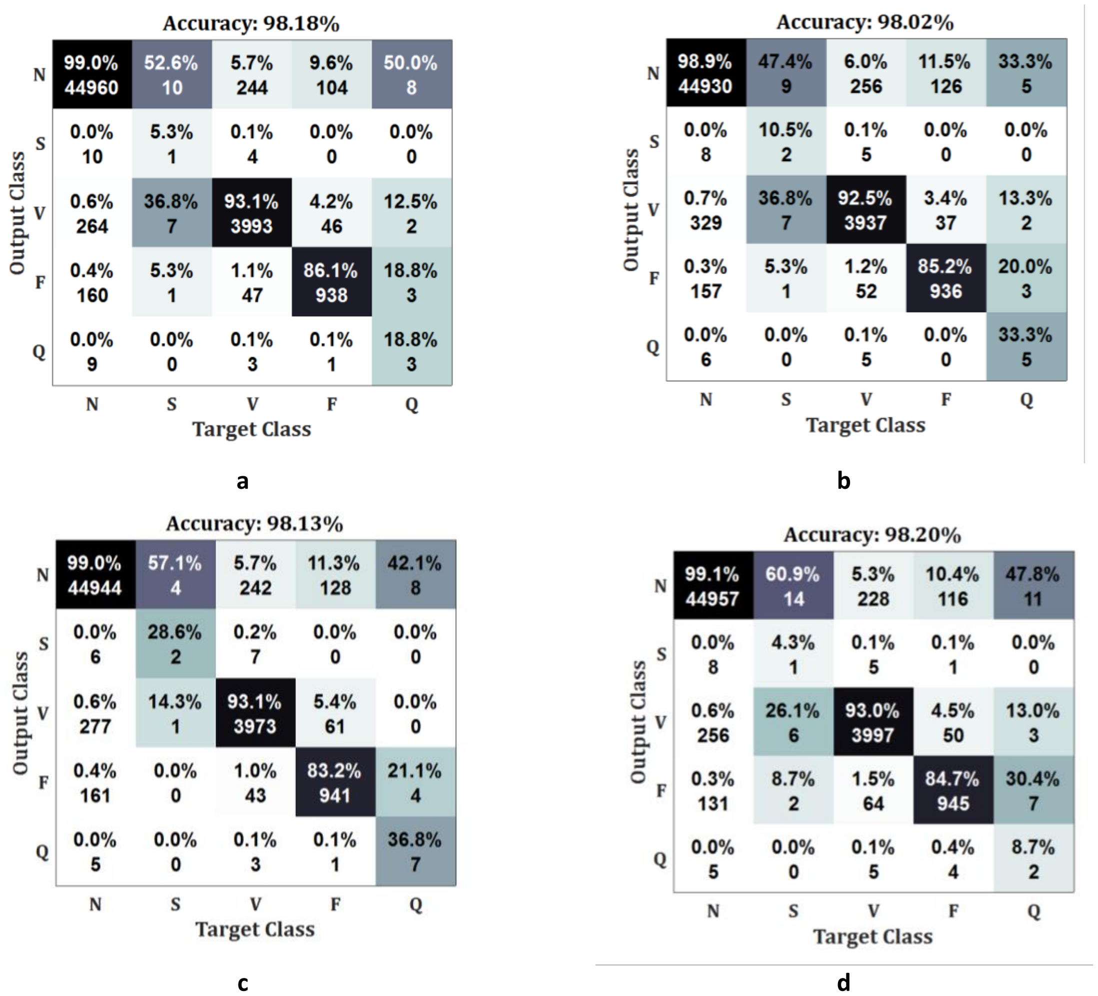
NTP represents the number of true positives, NTN represents the number of true negatives, NFP represents the number of false positives, and NFN represents the number of false negatives in the diagnosis of a specific class of cardiac abnormality. In the first step, it is necessary to evaluate the performance of the heart recognition model for signals selected from the database. The average recognition accuracy should be acceptable, and the average false rate should be less than 5% (with a small number of features). These values are ideal when the model is applied to the database.

Table 3. An estimation of criteria such as accuracy by estimating the maximum (Max), minimum (Min), and average (Ave) values using the proposed method and considering the change in frame length.

Results of test folds while altering the frame length		**A model without feature fusion and classification fusion**	**A model with feature fusion and classification fusion**
K-fold 1	Long frame length	0.89 ± (0.07)	0.95 ± (0.03)
	Average frame length	0.88 ± (0.07)	0.97 ± (0.03)
	Short frame length	0.89 ± (0.07)	0.95 ± (0.04)
K-fold 2	Long frame length	0.90 ± (0.07)	0.95 ± (0.03)
	Average frame length	0.91 ± (0.05)	0.97 ± (0.03)
	Short frame length	0.90 ± (0.07)	0.95 ± (0.04)
K-fold 3	Long frame length	0.90 ± (0.06)	0.95 ± (0.03)
	Average frame length	0.89 ± (0.07)	0.98 ± (0.04)
	Short frame length	0.90 ± (0.07)	0.95 ± (0.03)
K-fold 4	Long frame length	0.89 ± (0.07)	0.95 ± (0.03)
	Average frame length	0.89 ± (0.06)	0.96 ± (0.03)
	Short frame length	0.89 ± (0.07)	0.95 ± (0.04)
K-fold 5	Long frame length	0.89 ± (0.07)	0.95 ± (0.03)
	Average frame length	0.89 ± (0.06)	0.98 ± (0.03)
	Short frame length	0.89 ± (0.07)	0.96 ± (0.04)

The irregular heart signals were analyzed separately to compare cardiac functioning. Table 3 presents the differences in the results obtained from the proposed techniques. The cardiac dataset was analyzed independently using the proposed algorithm with the K-fold method, where K is 5. The short frame length varied from 800 ms to 1300 ms, the average frame length ranged from 1300 ms to 1800 ms, and the long frame length varied from 1800 ms to 2300 ms.

The representation of the confusion matrix can significantly contribute to the algorithm analysis process in terms of classification results. Thus, Fig. 3 illustrates the confusion matrix for 5 folds of a single run of the algorithm for classification in the set. The estimation is performed separately for each of the five classes, and investigating the dispersion among the algorithm outputs contributes to the estimation and investigation of challenges like repeatability and uncertainty. The final accuracies were estimated to investigate the dispersion among the responses in the next experiment, as shown in Fig. 4. Nearly 80% of the features could result in an acceptable accuracy when included in the classification process, indicating that the steadiness in selecting the number of features largely depends on the initial volume of the features. In other words, an efficient feature-selection strategy will select fewer features, and the selected features will function appropriately.



Accuracy: 98.18%

Output Class

	N	S	V	F	Q
N	99.0% 44960	52.6% 10	5.7% 244	9.6% 104	50.0% 8
S	0.0% 10	5.3% 1	0.1% 4	0.0% 0	0.0% 0
V	0.6% 264	36.8% 7	93.1% 3993	4.2% 46	12.5% 2
F	0.4% 160	5.3% 1	1.1% 47	86.1% 938	18.8% 3
Q	0.0% 9	0.0% 0	0.1% 3	0.1% 1	18.8% 3

Target Class

a

Accuracy: 98.02%

Output Class

	N	S	V	F	Q
N	98.9% 44930	47.4% 9	6.0% 256	11.5% 126	33.3% 5
S	0.0% 8	10.5% 2	0.1% 5	0.0% 0	0.0% 0
V	0.7% 329	36.8% 7	92.5% 3937	3.4% 37	13.3% 2
F	0.3% 157	5.3% 1	1.2% 52	85.2% 936	20.0% 3
Q	0.0% 6	0.0% 0	0.1% 5	0.0% 0	33.3% 5

Target Class

b

Accuracy: 98.13%

Output Class

	N	S	V	F	Q
N	99.0% 44944	57.1% 4	5.7% 242	11.3% 128	42.1% 8
S	0.0% 6	28.6% 2	0.2% 7	0.0% 0	0.0% 0
V	0.6% 277	14.3% 1	93.1% 3973	5.4% 61	0.0% 0
F	0.4% 161	0.0% 0	1.0% 43	83.2% 941	21.1% 4
Q	0.0% 5	0.0% 0	0.1% 3	0.1% 1	36.8% 7

Target Class

c

Accuracy: 98.20%

Output Class

	N	S	V	F	Q
N	99.1% 44957	60.9% 14	5.3% 228	10.4% 116	47.8% 11
S	0.0% 8	4.3% 1	0.1% 5	0.1% 1	0.0% 0
V	0.6% 256	26.1% 6	93.0% 3997	4.5% 50	13.0% 3
F	0.3% 131	8.7% 2	1.5% 64	84.7% 945	30.4% 7
Q	0.0% 5	0.0% 0	0.1% 5	0.4% 4	8.7% 2

Target Class

d

Fig. 3. The representation of multiple folds of a single running of the algorithm: a) accuracy98/18%, b) accuracy98/02%,c) accuracy 98/13%,d) accuracy98/20%. the average accuracy is estimated at 98.2%. The purpose of displaying multiple folds of a single running of the algorithm is to indicate the quantitative dispersion of the proposed algorithm.

The following experiment examined the changes in results when the performance of the proposed method was evaluated without reflecting the effect of the "fusion step of feature selection." It was observed that the feature selection step improves the response by up to about 2%. This observation is also reflected in the confusion matrix, where four out of five folds are displayed without the feature selection conditions. The dispersion among the responses is significant, and on average, a 1.5 to 2% reduction in accuracy occurs. This classification is illustrated in Fig. 4.

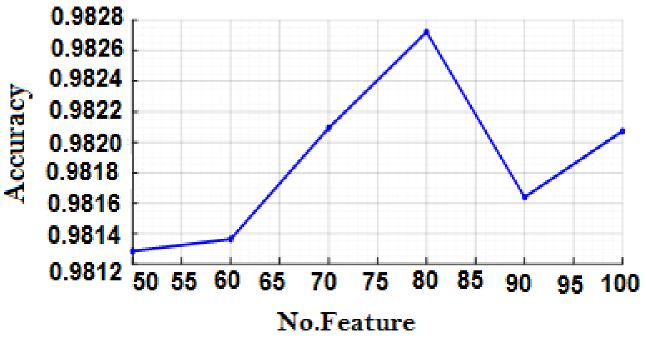
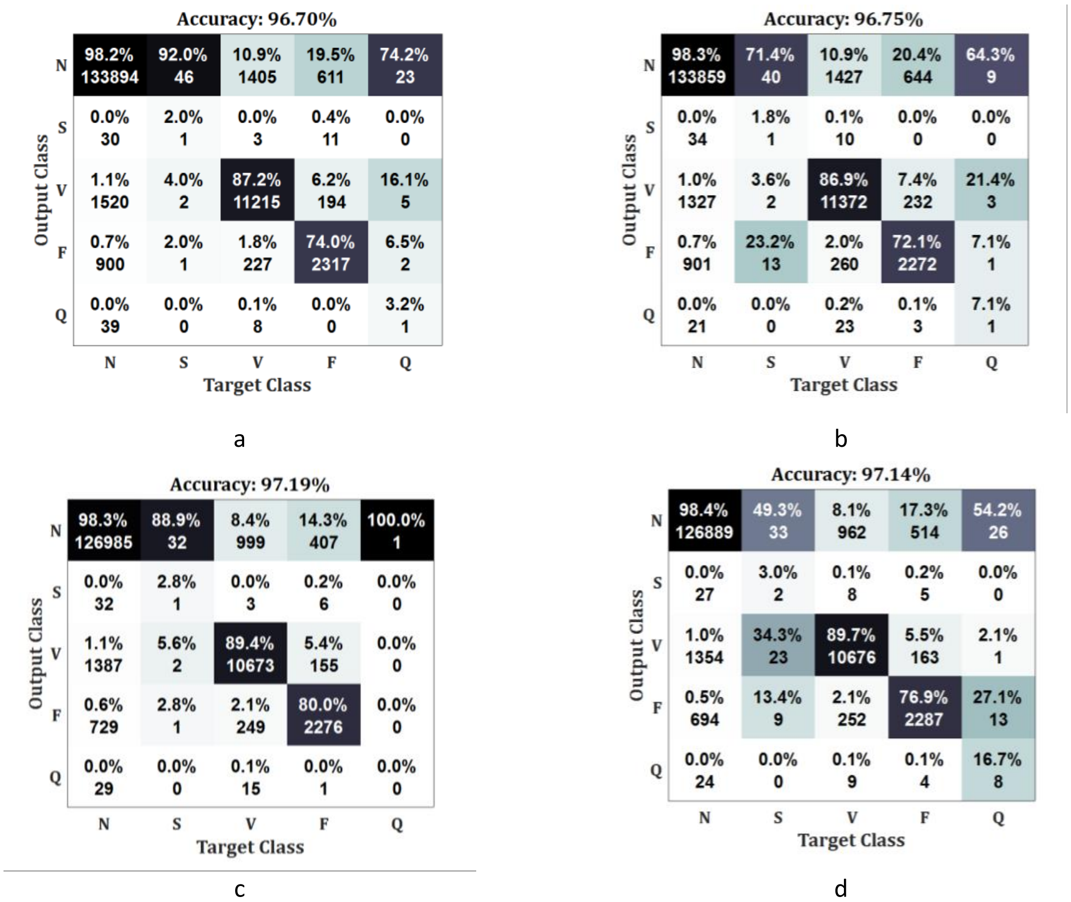


Fig. 4. A representation of the classification accuracy versus the number of features, indicating the maximum accuracy that is obtained when selecting 80% of the features.



Accuracy: 96.70%

Output Class \ Target Class	N	S	V	F	Q
N	98.2% 133894	92.0% 46	10.9% 1405	19.5% 611	74.2% 23
S	0.0% 30	2.0% 1	0.0% 3	0.4% 11	0.0% 0
V	1.1% 1520	4.0% 2	87.2% 11215	6.2% 194	16.1% 5
F	0.7% 900	2.0% 1	1.8% 227	74.0% 2317	6.5% 2
Q	0.0% 39	0.0% 0	0.1% 8	0.0% 0	3.2% 1

a

Accuracy: 96.75%

Output Class \ Target Class	N	S	V	F	Q
N	98.3% 133859	71.4% 40	10.9% 1427	20.4% 644	64.3% 9
S	0.0% 34	1.8% 1	0.1% 10	0.0% 0	0.0% 0
V	1.0% 1327	3.6% 2	86.9% 11372	7.4% 232	21.4% 3
F	0.7% 901	23.2% 13	2.0% 260	72.1% 2272	7.1% 1
Q	0.0% 21	0.0% 0	0.2% 23	0.1% 3	7.1% 1

b

Accuracy: 97.19%

Output Class \ Target Class	N	S	V	F	Q
N	98.3% 126985	88.9% 32	8.4% 999	14.3% 407	100.0% 1
S	0.0% 32	2.8% 1	0.0% 3	0.2% 6	0.0% 0
V	1.1% 1387	5.6% 2	89.4% 10673	5.4% 155	0.0% 0
F	0.6% 729	2.8% 1	2.1% 249	80.0% 2276	0.0% 0
Q	0.0% 29	0.0% 0	0.1% 15	0.0% 1	0.0% 0

c

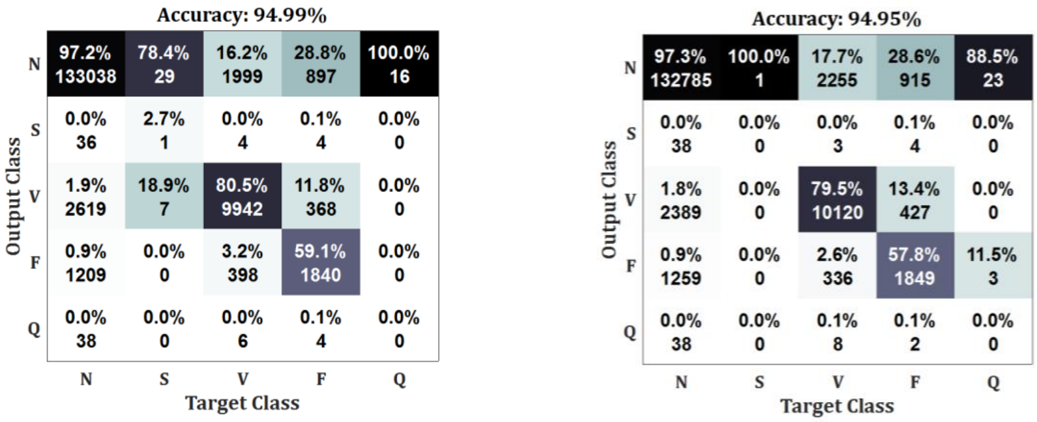
Accuracy: 97.14%

Output Class \ Target Class	N	S	V	F	Q
N	98.4% 126889	49.3% 33	8.1% 962	17.3% 514	54.2% 26
S	0.0% 27	3.0% 2	0.1% 8	0.2% 5	0.0% 0
V	1.0% 1354	34.3% 23	89.7% 10676	5.5% 163	2.1% 1
F	0.5% 694	13.4% 9	2.1% 252	76.9% 2287	27.1% 13
Q	0.0% 24	0.0% 0	0.1% 9	0.1% 4	16.7% 8

d

Fig. 5. The representation of four folds of a single running of the algorithm: a) accuracy96/70%, b) accuracy96/75%,c) accuracy 97/19%,d) accuracy97/14%. the effect of "the fusion step of feature selection" has been ignored. Under this condition, the classification accuracy will be reduced by 1.5 to 2% and the dispersity among the responses will be significant.

The impact of fusion in classification was examined in the subsequent experiment, and the results are presented through the confusion matrix. This state takes into account the feature selection step, but the classification is based on decision-making using the decision-making tree. The accuracy drop is more significant in this state, where the results are 2 to 3% less than the ideal state. Fig. 6 illustrates four random folds of a single run of the algorithm. Under some conditions similar to classification (such as classification while excluding the effect of fusion in feature selection and/or excluding the effect of fusion in classification), the diagnostic accuracy of some classes is notably low. However, when using the proposed method, the reduction in accuracy (even with fewer samples) is much lower than in similar states.



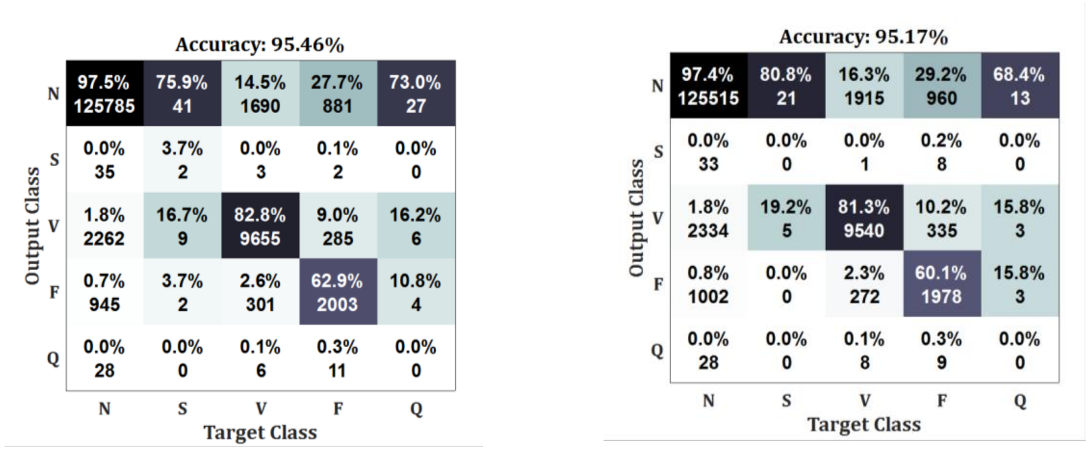
Accuracy: 94.99%

Output Class \ Target Class	N	S	V	F	Q
N	97.2% 133038	78.4% 29	16.2% 1999	28.8% 897	100.0% 16
S	0.0% 36	2.7% 1	0.0% 4	0.1% 4	0.0% 0
V	1.9% 2619	18.9% 7	80.5% 9942	11.8% 368	0.0% 0
F	0.9% 1209	0.0% 0	3.2% 398	59.1% 1840	0.0% 0
Q	0.0% 38	0.0% 0	0.0% 6	0.1% 4	0.0% 0

Accuracy: 94.95%

Output Class \ Target Class	N	S	V	F	Q
N	97.3% 132785	100.0% 1	17.7% 2255	28.6% 915	88.5% 23
S	0.0% 38	0.0% 0	0.0% 3	0.1% 4	0.0% 0
V	1.8% 2389	0.0% 0	79.5% 10120	13.4% 427	0.0% 0
F	0.9% 1259	0.0% 0	2.6% 336	57.8% 1849	11.5% 3
Q	0.0% 38	0.0% 0	0.1% 8	0.1% 2	0.0% 0

a b



Accuracy: 95.46%

Output Class \ Target Class	N	S	V	F	Q
N	97.5% 125785	75.9% 41	14.5% 1690	27.7% 881	73.0% 27
S	0.0% 35	3.7% 2	0.0% 3	0.1% 2	0.0% 0
V	1.8% 2262	16.7% 9	82.8% 9655	9.0% 285	16.2% 6
F	0.7% 945	3.7% 2	2.6% 301	62.9% 2003	10.8% 4
Q	0.0% 28	0.0% 0	0.1% 6	0.3% 11	0.0% 0

Accuracy: 95.17%

Output Class \ Target Class	N	S	V	F	Q
N	97.4% 125515	80.8% 21	16.3% 1915	29.2% 960	68.4% 13
S	0.0% 33	0.0% 0	0.0% 1	0.2% 8	0.0% 0
V	1.8% 2334	19.2% 5	81.3% 9540	10.2% 335	15.8% 3
F	0.8% 1002	0.0% 0	2.3% 272	60.1% 1978	15.8% 3
Q	0.0% 28	0.0% 0	0.1% 8	0.3% 9	0.0% 0

c d

Fig. 6. The representation of several folds of a single running of the algorithm: a) accuracy94/99%, b) accuracy94/95%,c) accuracy 95/46%,d) accuracy95/17%. the effect of "the fusion step of feature selection" has been ignored. Under this condition, the classification accuracy will drop by 2 to 3% and the dispersity among the responses will be significant.

Some other conditions can be considered to demonstrate the capability of the proposed algorithm. For instance, fusion has improved classification in both the feature selection step and the classification step compared to the typical state (i.e., feature selection and classification using a simple strategy), where the average accuracy reaches 95%. However, other factors, such as the type of features, the length of the selected frames, the number of selected features, and the initial parameters of the adjustment process, contribute to classification. When evaluating the algorithm's ability to process both testing and validation data, the issue of uncertainty needs to be addressed. Plotting the ROC can help assess this problem to a great extent. Therefore, in the other experiments, the area under the ROC was investigated, and estimations were made for both testing and validation data (Fig. 6).

Fig. 7 displays the ROC, which shows the area under the curve for both testing and validation data. Fig. 8 examines sensitivity and specificity, where the estimation of these factors indicates the performance of the proposed method in terms of unseen cardiac signals. In this experiment, these factors were estimated separately for each class of the two sets of unseen cardiac signals.

The proposed method has shown greater efficiency in detecting N, V, and F states in classifying various cardiac disorders in the ECG signal. However, it has been more prone to errors in detecting S and Q states. The main reason for this issue is the lack of data on S and Q states, which has resulted in insufficient training of the algorithm. If more data on S and Q classes were available, the learning model would be more effective in distinguishing the various classes optimally.

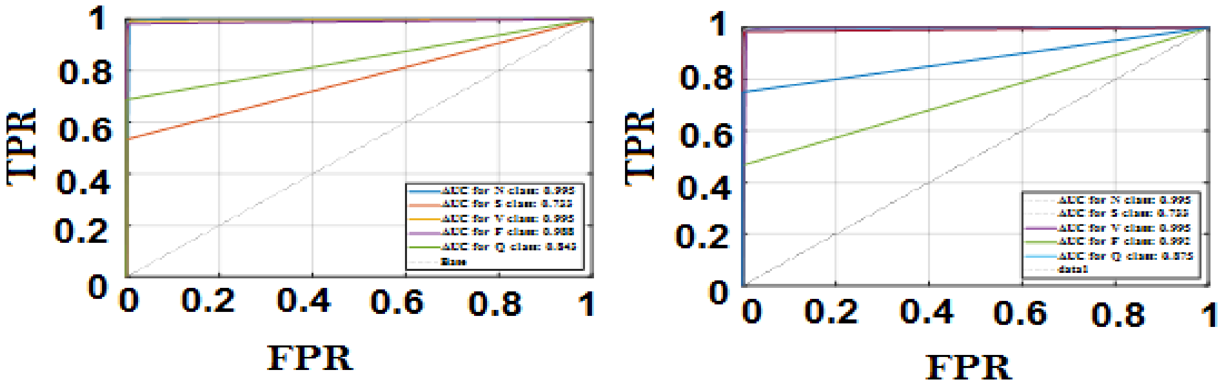


Fig. 7. The representation of the ROC matrix for validation data (left inset) and testing data (right inset).

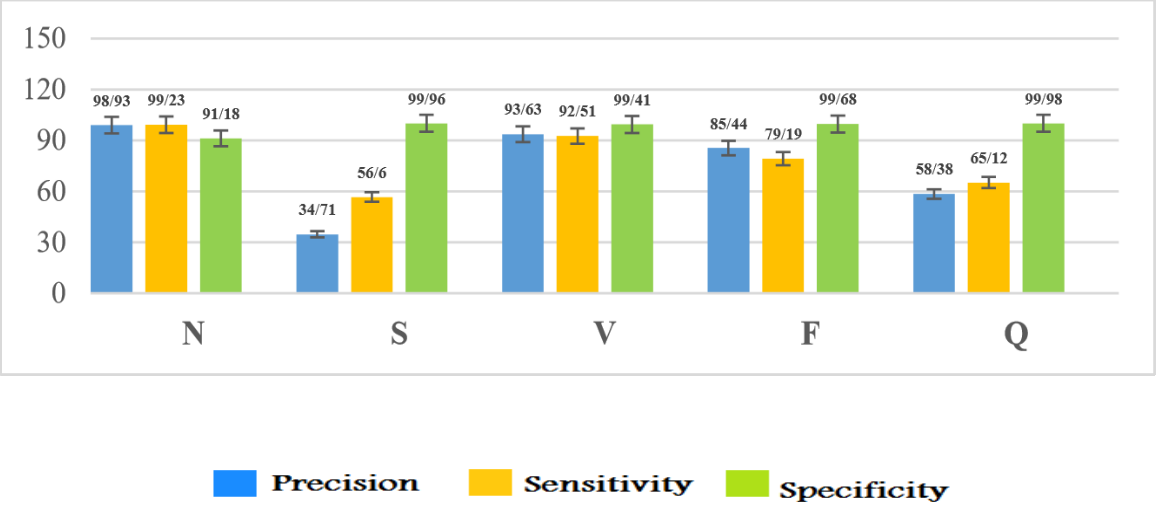


Fig. 8. An estimation of the sensitivity, accuracy, and specificity for cardiac unseen signals.

Overall, the performance of the proposed method is satisfactory. Although it has created some errors in relation to two classes, previous algorithms have either ignored these classes or considered the estimation criteria for all the concerned classes. The proposed model can respond to various sets of cardiac signals. In previous methods, the classification error has been even higher than that in the method proposed in this study. For instance, in previous studies such as Llamedo et al. [25], Ye et al. [26], De Chazal et al. [27], Afkhami et al. [13], both the number of classes and samples are lower than those in this study. Furthermore, these studies offer moderate accuracy, and when considering the excluded classes and samples, the accuracy of their proposed methods will significantly decrease. Some methods have high computational complexity, while others suffer from prolonged response times in both training and testing steps. Table 5 compares the proposed method with methods proposed in other studies.

7. CONCLUSION

The proposed integrated system in this study is a procedure that identifies various states of cardiac functioning by analyzing heartbeats through exploring the MIT-BIH database. Unlike other methods proposed thus far, which simply use techniques such as ANOVA, comparison of the mean or SD of signals, and/or estimation of the accuracy in diagnosing cardiac diseases, the method proposed in this study diagnoses CVDs based on diagnosis labels.

Table 5. The drawbacks and advantages of the proposed method compared to the methods proposed in recent years in relation to the diagnosis of various cardiac functioning states based on heartbeats.

Ref.	Features	Classifier	Classification Accuracy (%)	Advantages	Drawbacks
Afkhami et al. (2016) [13]	Modeling of statistical features and RR intervals	GMM	96.15	Creating low-level, appropriate features; the possibility of decision-making for 4 classes	Excluding some classes and samples; sensitivity to the number of samples; the possibility of overfitting
Ye et al. (2012) [26]	Wavelet; analysis of independent components; RR pick	Support vector machine (SVM)	99.71	Proper accuracy in the two-class state; utilizing low-level features; satisfactory outcomes	Investigation of only two classes; high volume of features; sensitivity to noise

Rodriguez et al. (2005) [27]	Waveform and signal morphology	Classification based on the “strict voting method”	96.13	Proper accuracy in the two-class state; instant classification	Investigation of only two classes; high volume of features; sensitivity to noise
de Oliveira et al. (2011) [28]	Waveform and signal morphology; RR features	Dynamic Bayesian Network (DBN)	98	High number of classes for analyzing heartbeat in the signal	Sensitivity to noise in the input signal; high computational complexity
Zubair et al. (2016) [29]	Deep CNN	Softmax function	97	Creating high-level features; the capability of decision-making for four classes	The need for a high volume of data; high dimension of the extracted features; high computational complexity
Chandra et al. (2018)	A 8-layer CNN and learning	Softmax function	92.7	Creating high-level features; the capability of decision-making for five classes	The need for a high volume of data; high dimension of the extracted features; relatively high computational complexity
Acharya et al. (2017) [31]	A 11-layer CNN and auto-learning	Softmax function	94.9	Creating high-level features; the capability of decision-making for five classes	No interpretability of features; dependence on the length of the window in the signal
Chu et al. [32]	A 12-layer CNN with the LSTM structure; SVM; particle swarm optimization (PSO) algorithm	Optimized SVM	97.8	Creating high-level features; the capability of decision-making for five classes	No interpretability of features; dependence on the length of the window in the non-stationary signal
Avanzato et al. (2020) [33]	CNN with the normalizing step	LSTM and Softmax function	98.3	Creating high-level features; the capability of decision-making for three classes	No interpretability of features; dependence on the length of the window in the non-stationary signal
Zhang et al. (2023) [34]	Deep CNN; features of the time and frequency domain	Softmax function	99.1	Creating high-level and low-level features; the capability of decision-making for eight classes	The need for a high volume of data; high dimension of the extracted features; relatively high computational complexity; dependence on the length of the window in the non-stationary signal
The proposed method	Feature extraction using the	Feature selection using the	98.28	Investigation of five classes; investigation of non-stationary signals and the problem of	Inappropriate output in cases where the signal is affected

	features of the time, frequency, and fractal domains	fusion method; classification based on the fusion among the decision-making trees		uncertainty; acceptable accuracy of classification; low sensitivity to noise	with too noises and/or when there are artifacts in the recorder. There is no functional mechanism for when the signal and classification are lost.

This is achieved by training and testing data through the data splitting procedure, the K-fold validation method, and using a classic and efficient model in the learning process. The classification accuracy in this study was 98.28%. The diagnosis system in this study is inspired by soft computing algorithms and serves as a tool to identify the presence of abnormalities in ECG signals and heartbeat signals in the dataset of samples.

The proposed method uses manual feature extraction and appropriate model adjustment, which presents some challenges. Using other feature extraction methods, such as nonlinear features based on efficient descriptors of the ECG signal, and combining manual and automated features will likely enhance the diagnosis accuracy. Therefore, it is recommended to extract features based on various methods to obtain more optimal features. The integration of features increases the likelihood of choosing the best feature and enhances the rate of diagnosing abnormalities through analyzing ECG signals.

In this study, the classes were classified using a classic learning algorithm. However, it is suggested to enhance the function of the classifier. There are various classifiers used to distinguish output labels, but clustering would be more efficient if it can produce low errors and classify new data based on the correct separation. Furthermore, deep learning methods could be useful in these settings.

REFERENCES

[1] F.P.D. Carvalho & C.F.D. Azevedo. Coronary Slow Flow Phenomenon-Adding Myocardial Fibrosis to the Equation. Arquivos brasileiros de cardiologia, (2020), 114, 552-553.

[2] R.A. Sanders, T.A. Kurosawa, & M.D. Sist. Ambulatory electrocardiographic evaluation of the occurrence of arrhythmias in healthy Salukis. Journal of the American Veterinary Medical Association, (2018), 252(8), 966-969.

[3] O. Deperlioglu. Segmentation of heart sounds by re-sampled signal energy method. BRAIN. Broad Research in Artificial Intelligence and Neuroscience, (2018), 9(1), 17-28.

[4] A. Giorgio, C. Guaragnella, & M. Rizzi. An Effective CAD System for Heart Sound Abnormality Detection. Circuits, Systems, and Signal Processing, (2022), 1-26. https://doi.org/10.1007/s00034-022-01909-6

[5] A.H. Khan, M. Hussain, & M.K. Malik. Cardiac disorder classification by electrocardiogram sensing using deep neural network. Complexity, (2021), 2021, Article ID 5511307. https://doi.org/10.1155/2021/5511307

[6] A.H. Khan, M. Hussain, & M.K. Malik. Arrhythmia classification techniques using deep neural network. Complexity, (2021), 2021, Article ID 5511307. https://doi.org/10.1155/2021/5511307

[7] B. Xiao, Y. Xu, X. Bi, J. Zhang, & X. Ma. Heart sounds classification using a novel 1-D convolutional neural network with extremely low parameter consumption. Neurocomputing, (2020), 392, 153-159.

[8] F. Noman, C.M. Ting, S.H. Salleh, & H. Ombao. Short-segment heart sound classification using an ensemble of deep convolutional neural networks. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 1318-1322). IEEE, (2019).

[9] B.M. Mathunjwa, Y.T. Lin, C.H. Lin, M.F. Abbod, & J.S. Shieh. ECG arrhythmia classification by using a recurrence plot and convolutional neural network. Biomedical Signal Processing and Control, (2021), 64, 102262.

[10] O. Yildirim, U.B. Baloglu, R.S. Tan, E.J. Ciaccio, & U.R. Acharya. A new approach for arrhythmia classification using deep coded features and LSTM networks. Computer methods and programs in biomedicine, (2019), 176, 121-133.

[11] F.M. Dias, H.L. Monteiro, T.W. Cabral, R. Naji, M. Kuehni, & E.J. Luz. Arrhythmia classification from single-lead ECG signals using the inter-patient paradigm. Computer Methods and Programs in Biomedicine, (2021), 202, 105948.

[12] A. Chandrasekar, D.D. Shekar, A.C. Hiremath, & K. Chemmangat. Detection of arrhythmia from electrocardiogram signals using a novel Gaussian assisted signal smoothing and pattern recognition. Biomedical Signal Processing and Control, (2022), 73, 103469.

[13] R.G. Afkhami, G. Azarnia, & M.A. Tinati. Cardiac arrhythmia classification using statistical and mixture modeling features of ECG signals. Pattern Recognition Letters, (2016), 70, 45-51.

[14] L.B. Marinho, N. de MM Nascimento, J.W. Souza, M.V. Gurgel, P.P. Rebouças Filho, & V.H. de Albuquerque. A novel electrocardiogram feature extraction approach for cardiac arrhythmia classification. Future Generation Computer Systems, (2019), 97, 564-577.

[15] M. Jangra, S.K. Dhull, & K.K. Singh. ECG arrhythmia classification using modified visual geometry group network (mVGGNet). Journal of Intelligent & Fuzzy Systems, (2020), 38(3), 3151-3165.

[16] Y. Kaya, F. Kuncan, & R. Tekin. A New Approach for Congestive Heart Failure and Arrhythmia Classification Using Angle Transformation with LSTM. Arabian Journal for Science and Engineering, 47 (2022), 2075-2081.

[17] J. Rahul, M. Sora, L.D. Sharma, & V.K. Bohat. An improved cardiac arrhythmia classification using an RR interval-based approach. Biocybernetics and Biomedical Engineering, 41(2) (2021), 656-666.

[18] H. Lee & M. Shin. Learning explainable time-morphology patterns for automatic arrhythmia classification from short single-lead ECGs. Sensors, 21(13) (2021), 4331.

[19] J. Zhang, D. Liang, A. Liu, M. Gao, X. Chen, & X. Zhang. MLBF-Net: a multi-lead-branch fusion network for multi-class arrhythmia classification using 12-Lead ECG. IEEE Journal of Translational Engineering in Health and Medicine, 9 (2021), 1-1.

[20] J. Rahul & L.D. Sharma. Automatic cardiac arrhythmia classification based on hybrid 1-D CNN and Bi-LSTM model. Biocybernetics and Biomedical Engineering, 42 (2022), 1-11.

[21] M.R. Rajeshwari & K.S. Kavitha. Arrhythmia ventricular fibrillation classification on ECG signal using ensemble feature selection and deep neural network. Cluster Computing, 25 (2022), 1-8.

[22] S. Kaptoge, L. Pennells, D. De Bacquer, M.T. Cooney, M. Kavousi, G. Stevens, et al. World Health Organization cardiovascular disease risk charts: revised models to estimate risk in 21 global regions. The Lancet Global Health, 7(10) (2019), e1332-e1345.

[23] T.A. McDonagh, M. Metra, M. Adamo, R.S. Gardner, A. Baumbach, M. Böhm, et al. 2021 ESC Guidelines for the diagnosis and treatment of acute and chronic heart failure: Developed by the Task Force for the diagnosis and treatment of acute and chronic heart failure of the European Society of Cardiology (ESC) With the special contribution of the Heart Failure Association (HFA) of the ESC. European heart journal, 42(36) (2021), 3599-3726.

[24] G. Sannino & G. De Pietro. A deep learning approach for ECG-based heartbeat classification for arrhythmia detection. Future Generation Computer Systems, 86 (2018), 446-455.

[25] M. Llamedo & J.P. Martínez. Heartbeat classification using feature selection driven by database generalization criteria. IEEE Transactions on Biomedical Engineering, 58(3) (2010), 616-625.

[26] C. Ye, B.V. Kumar, & M.T. Coimbra. Heartbeat classification using morphological and dynamic features of ECG signals. IEEE Transactions on Biomedical Engineering, 59(10) (2012), 2930-2941.

[27] P. De Chazal, M. O'Dwyer, & R.B. Reilly. Automatic classification of heartbeats using ECG morphology and heartbeat interval features. IEEE Transactions on Biomedical Engineering, 51(7) (2004), 1196-1206.

[28] L.S. de Oliveira, R.V. Andreão, & M. Sarcinelli-Filho. Premature ventricular beat classification using a dynamic Bayesian network. In 2011 annual international conference of the IEEE engineering in medicine and biology society (pp. 4984-4987) (2011).

[29] M. Zubair, J. Kim, & C. Yoon. An automated ECG beat classification system using convolutional neural networks. In Proceedings of the 2016 6th international conference on IT convergence and security (ICITCS) (pp. 1-5) (2016).

[30] S. Chandra, C.S. Sastry, & S. Jana. Robust heartbeat detection from multimodal data via CNN-based generalizable information fusion. IEEE Transactions on Biomedical Engineering, 66(3) (2018), 710-717.

[31] U.R. Acharya, H. Fujita, O.S. Lih, Y. Hagiwara, J.H. Tan, & M. Adam. Automated detection of arrhythmiasusing different intervals of tachycardia ECG segments with convolutional neural network. Information sciences, 405 (2017), 81-90.

[32] J. Chu, H. Wang, & W. Lu. A novel two-lead arrhythmia classification system based on CNN and LSTM. Journal of Mechanics in Medicine and Biology, 19(03) (2019), 1950004.

[33] R. Avanzato & F. Beritelli. Automatic ECG diagnosis using convolutional neural network. Electronics, 9(6) (2020), 951.

[34] Y. Zhang, J. Yi, A. Chen, & L. Cheng. Cardiac arrhythmia classification by time-frequency features inputted to the designed convolutional neural networks. Biomedical Signal Processing and Control, 79 (2023), 104224.

Distributed Jammer-Equipped Target Tracking with Hybrid Extended Kalman and Particle Filter in Sensor Network

Iman Maghsudlu[1], Meysam Raees Danaee[2], Hamid Arezumand[3]

1- Faculty of Electrical Engineering Department, Imam Hossein Comprehensive University, Tehran, Iran.
Email: iman.maghsudlu@gmail.com

2- Imam Hossein Comprehensive University, Tehran, Iran.
Email: mrdanaee@gmail.com, (Corresponding Author)

3- Imam Hossein Comprehensive University, Tehran, Iran.
Email: arezumand.h@ihu.ac.ir

ABSTRACT:
In this paper, we propose a distributed tracking algorithm for a jammer-equipped target that passes through the nodes of a wireless sensor network (WSN). The jammer that the target carries is of the deceptive type, which means that it can mimic the signal of the target and confuse the sensors. Unlike other existing methods, our proposed algorithm does not require any additional hardware installation on each WSN node. It only relies on signal processing and solving the average consensus problem to detect the presence of jamming effects in the observations of the contaminated nodes and exclude them from the distributed tracking problem. For the distributed tracking problem, we use the hybrid extended Kalman filter (EKF) and particle filter to reduce the number of parameters needed for solving the average consensus problem and to decrease the communication overhead. The simulation results show that our algorithm improves the tracking performance compared to the case where the nodes with jammed observations are used.

KEYWORDS: Distributed Particle Filter, Sensor Network, Multi-Sensor Target Tracking, Data Fusion, Consensus-Base Algorithms.

1. INTRODUCTION

Estimating the state vector of a target that passes through the nodes of a WSN has recently attracted attention [1]. This task can be performed either centrally (where all the sensors send their information to a central node) or distributedly (where each sensor obtains an estimate of the state vector of the target by exchanging information with its one-hop neighbors). The distributed tracking method is more preferable because the nodes do not need to know the network topology and the whole system does not fail if the central node breaks down [2]. In the nonlinear environment, particle filters have shown better results than other nonlinear filters [3]. In the following, we review different ideas that have been proposed in the field of distributed tracking. We categorize them based on the type of communication, estimation, and fusion methods used by the sensors and the fusion center.

In [4], the probability vector of particles at the location of each node is projected on a set of eigenvectors of the Laplacian matrix of the graph and then the obtained projections are averaged with the help of a consensus-based algorithm in the network to calculate the global probability vector of the network in each node. [5] proposes a method for achieving average consensus on the sufficient statistics of the target state, which are computed by using a particle filter to approximate the posterior distribution. The method is based on a distributed weighted average algorithm that allows the nodes to exchange and update their local estimates iteratively until they converge to a global estimate. In [6], a distributed particle filter for bearing-only tracking applications is proposed, based on a constrained sufficient statistic approach. The method uses local particle filters at each sensor node and computes the global sufficient statistics as a sum of the local sufficient statistics. The method reduces the communication complexity and bandwidth requirement by running average consensus algorithms only on the means of the local sufficient statistics. The method achieves near-

Paper type: Research paper
https://doi.org/
Received: 11 January 2024; revised: 25 February 2024; accepted: 23 March 2024; published: 1 June 2024
How to cite this paper: I. Maghsudlu, M. Raees Danaee , H. Arezumand, **“Distributed Jammer-Equipped Target Tracking with Hybrid Extended Kalman and Particle Filter in Sensor Network”**, *Majlesi Journal of Telecommunication Devices*, Vol. 13, No. 2, pp. 87-96, 2024.

optimal performance, comparable to the centralized particle filter.

Reference [7] presents a method for distributed estimation of the target state based on a Gaussian mixture model (GMM). Each node uses a particle filter to approximate the local posterior distribution by a GMM, and then exchanges and updates the GMM parameters with its neighbors using a distributed weighted average algorithm. This results in a global GMM that represents the posterior distribution projected onto the basis functions. Each node can then compute its own estimate of the target state from the global GMM. In [8], the key idea is to use consensus algorithms to approximate the joint likelihood function of all sensors, which summarizes the measurement information. This approach is applicable if the local likelihood functions belong to the exponential family. The paper then develops distributed particle filters and distributed Gaussian particle filters based on the consensus approximation of the joint likelihood function. Each sensor runs a local (Gaussian) particle filter that computes a global state estimate. The paper also proposes a method to reduce the number of particles in the distributed Gaussian particle filter by using another consensus scheme. Despite these advances in distributed tracking, there is no suitable method for addressing and solving the problem of jamming in WSNs when the target tracking is performed in a distributed manner. Some targets have installed a hardware device on themselves that acts as a self-screening jammer to prevent WSNs from tracking them effectively [9-11].

In this paper, we aim to propose a method that can cope with such a jammer and preserve the distributed tracking performance. In [12-13] the methods of dealing with deceptive jammer in a sensor network have been investigated. This jammer can create collaborative or non-collaborative targets for network nodes. In this paper, jammer is considered a non-collaborative type. In the scenario presented by [12] to detect a collaborative jammer it is necessary to place a number of passive sensors scattered in a number of network nodes which is not always possible. In the second type of sensor network jammers, the communication links of the network nodes are disrupted and in [14-17] some algorithms are presented that can estimate the location of the jammer. To learn more about the second type of jammers, we refer the reader to the relevant literature [9-11]. Target tracking in wireless sensor networks is vulnerable to jamming attacks, which can degrade the performance of the distributed estimation algorithms and cause the target to escape detection.

In this paper, we propose a method that can identify the jammed nodes based on the disturbance in their measurements and exclude them from the consensus-based tracking algorithm. By removing the jammed nodes, we can improve the accuracy and reliability of the target tracking in the network.

In this paper, we propose an idea that if only a small portion of the sensors are contaminated by the jamming disturbances, then we can obtain the average of the observations by solving the average consensus problem. Then, since the contaminated sensors cannot change the average value, we can identify them by comparing them with the obtained average (because they are far from it). Therefore, the network can ignore their data and then solve the distributed tracking problem assuming that all the data are clean, as in the usual algorithms in this field. The advantage of this method is that, unlike the proposed methods for controlling the jamming problem in WSNs that require additional hardware installation for jamming detection, the proposed method only detects the contamination of the observations by using the solution of the average consensus problem on the received observations.

The rest of the paper is organized as follows: Section 2 formulates and states the problem. Section 3 introduces the distributed state estimation problem based on the graph theory idea. In this section, we first describe the characteristics of the jammer that the target is equipped with, and then we present the algorithm for detecting the jammed sensors by solving the average consensus problem. Section 4 presents and analyzes the simulation results. Section 5 concludes the paper.

2. PROBLEM FORMULATION

The sensor network used in this paper consists of S nodes that are scattered in an area where we want to track the jammer-equipped target. Jammer signal specifications are explained in the relevant section. This sensor network is modelled with an undirected graph G={V,E}. V is the set of all the nodes of the network and E is the set of all edges. E⊂V×V is established between these two sets. *(i,j)* is a member of this set if node *i* and *j* are connected through a communication link. $N_i = \left\{ j \middle| (i, j) \in E \right\}$ is the set of all one-hop neighbours of node *i* where $d_i = \left| N \right|$ is called degree of node *i*. in each time step, each node can exchange its data with its neighbouring nodes for a specified number of iterations. $X_k = \left[x_k, y_k, \dot{x}_k, \dot{y}_k \right]^T$ is the state vector of the target that we want to track in each time step and this vector includes the position and speed of the target in two-dimensional coordinates. The target state transition model in this network is as

$$\mathbf{x}_{k+1}=\mathrm{f}(\mathbf{x}_k,\xi_k) \tag{1}$$

Also, the observation made of the target in node *s* and time step *k* is as

$$\mathbf{z}_k^{\mathrm{s}}=\mathrm{h}^{\mathrm{s}}(\mathbf{x}_k,\mathbf{v}_k,,\mathbf{j}_k) \tag{2}$$

Where $\mathbf{z}_k^{\mathrm{s}}=[r_{\mathbf{z},k}^{\mathrm{s}},\theta_{\mathbf{z},k}^{\mathrm{s}}]^{\mathrm{T}}$ is an observation made in node s. ξ_k, $\mathbf{v}_k$ and $\mathbf{j}_{\mathrm{k}}$ are noise vectors of state, observation and jamming whose covariance matrices are equal to $\mathbf{Q}_k$, $\mathbf{R}_k$ and $\mathbf{J}_{\mathrm{k}}$ respectively. Observation vector $\mathbf{z}_k^{\mathrm{s}}$ is as

$$\mathbf{z}_k^s=[r_{\mathbf{z},k}^s,\theta_{\mathbf{z},k}^s]^{\mathrm{T}}=[r_k^s,\theta_k^s]^{\mathrm{T}}+\mathbf{z}_k^n=[r_k^s,\theta_k^s]^{\mathrm{T}}+[r_k^n,\theta_k^n]^{\mathrm{T}} \tag{3}$$

Where r_k^s and θ_k^s are the range and angle of the target relative to the node s as (4) and (5) respectively. $\mathbf{z}_k^n$ is zero mean Gaussian noise with covariance matrix $\begin{bmatrix}\sigma_r^2 & 0\\ 0 & \sigma_\theta^2\end{bmatrix}$.

$$\mathrm{r}_{\mathrm{k}}^{\mathrm{s}}=\sqrt{(x_k-x_s)^2+(y_k-y_s)^2} \tag{4}$$

$$\theta_k^s=\mathrm{Arctan}(\frac{y_k-y_s}{x_k-x_s}) \tag{5}$$

$[x_s,y_s]^{\mathrm{T}}$ and $[x_k,y_k]^{\mathrm{T}}$ are coordinates of node s and target in the reference coordination system. If in (1) and (2) the functions f and h^{s} have a lot of nonlinear properties or if the target manoeuvre is very high, distributed particle filter will be a good choice for target tracking.

3. GRAPH-BASED DISTRIBUTED HEPF

In this paper, jammer-equipped target tracking in WSN is presented. This jammer can be a type of noise or a type of deception as what is considered here.

We consider the problem of distributed estimation of a target that is equipped with a jammer. We assume that the target can be tracked by a network of sensors, which communicate with each other using consensus-based algorithms. We propose a method to identify and exclude the sensors that are affected by the jamming signals, and thus reduce the impact of the jamming on the estimation accuracy. Moreover, we propose a hybrid approach that switches between different filters depending on the target's motion dynamics. When the target's motion is high, we use a particle filter, which can handle the nonlinearity and uncertainty of the target's state. When the target's motion is low, we use an extended Kalman filter, which is faster but less accurate than the particle filter.

Hybrid Extended Kalman Filter (HEPF) consists of a combination of particle and extended Kalman filters and can be used in distributed target tracking in a sensor network. By doing so, we can balance the trade-off between the communication overhead and the tracking speed. We compare our proposed method with the conventional methods that use only one sensor or all the sensors for tracking, and show that our method can significantly improve the performance in terms of the root mean square error (RMSE) of the target state estimation.

In the first case when the combination of these two filters is used for tracking the tracking speed will be low due to the complex calculations of the particle filter and in the second case due to the removal of the particle filters the calculation volume will be low and as a result the tracking speed will be higher. One can refer to [18] to study more details of HEPF and its related relationships. Also, the particle filter used here is a graph-based distributed particle filter the details of which are described in [4]. As can be seen in Fig. 1 a switch is considered that compares the dynamic value of the target movement with the threshold level and selects the appropriate filter. Threshold level and target dynamic value can be defined at the discretion of the designer.

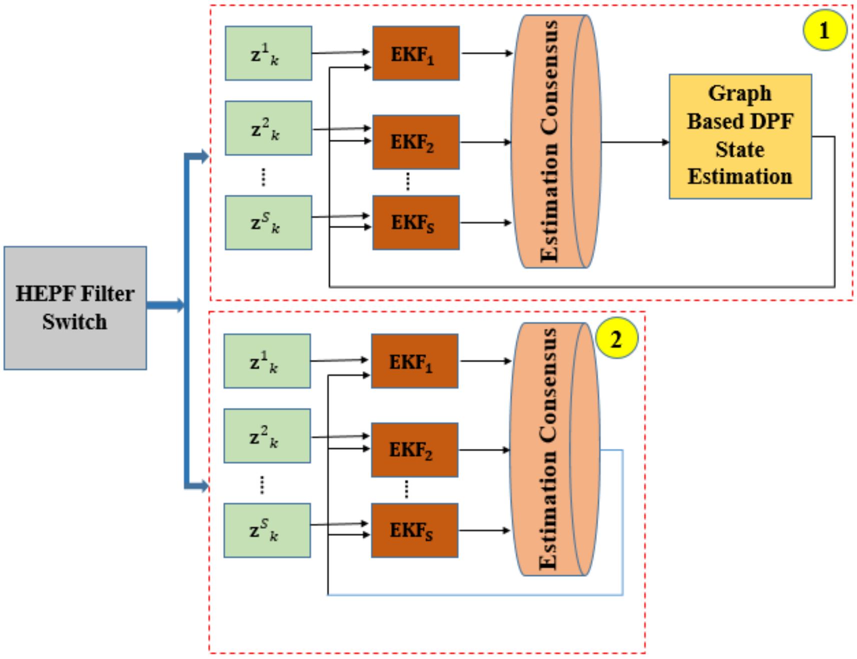


Fig. 1. Distributed HEPF in sensor network.

3.1. Target self-screening jammer

In general, jammer can be of the type of suppression or deception [19]. The target that we want to track in the sensor network is equipped with a deceptive jammer which can send the deception targets to the sensors of the network with the help of DRFM[1] technology [20]. Deception targets are of collaborative type and it means that the deception targets created for all network sensors are produced at the same location in the reference coordinates even though the location coordinates of the network nodes are different [12]. The point that seems important here is that the number of signals received by DRFM is limited, therefore after saturation of this system the deception signal does not affect the sensors located at a further distance.

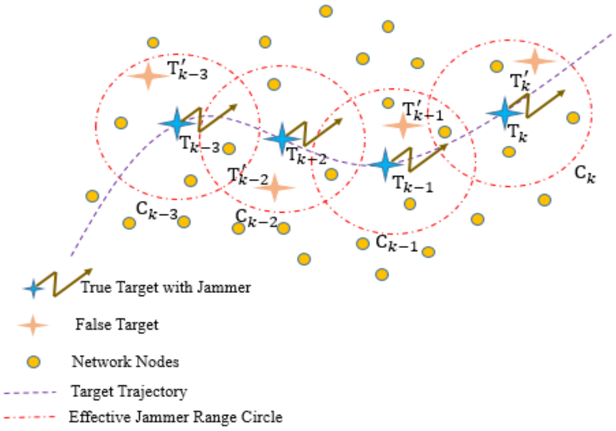


Fig. 2. Jammer effect on observation model of in-range nodes.

As can be seen in figure 2, the real target that carries a jammer and is marked with a blue star, is at the point T_k of its true path at the time step k. Jammer that is installed on the target jams all the sensors inside the hypothetical circle C_k at the time step k so all the sensors inside C_k do not see the real location of the target as can be seen in figure 4. T'_k indicates the deception target in the hypothetical circle. the nodes that are outside of C_k are not jammed and the correctness of their operation is preserved. Therefore, it can be said that at any time step k a limited number of nodes

[1] Digital Radio Frequency Memory

are jammed and their state estimations are incorrect and instead there are a much larger number of nodes that are not jammed and their state estimations are completely correct. Here we presented a method by which the network detects the jammed nodes with the help of the consensus algorithm and makes a correct distributed estimation of the target state and prevents the target tracking from diverging.

3.2. Distributed jammer detection algorithm in network sensor

Consider a situation where we have only one node to track the jammer-equipped target. This issue is shown in figure 3.

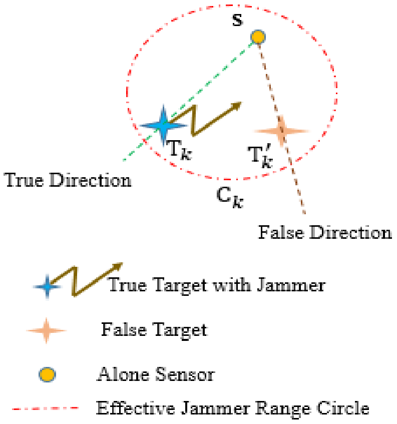


Fig. 3. Jammer-equipped target tracking with only one node.

This node determines the state of the target independently by processing the observation received from the target. Since there is no other node, it is impossible to compare the received observations with other nodes to reveal the presence of a jammer effect in the received observation signal. The main idea of this paper is to use a sensor network and define a consensus-based criterion for evaluating the presence of jamming in the observations nodes have been received by comparing the observations obtained in the nodes of the network. The presented distributed consensus-based algorithm can identify the observations containing the jammer signal and then remove the jamming effect from the target tracking. Therefore, the method presented here generally has two steps and is shown in figure 6. As can be seen in figure 6, before the target tracking is done with a distributed particle filter (DPF), local observations are entered into two blocks that implement jamming detection in stage 1 and jamming cancellation in stage 2. These two stages are explained below.

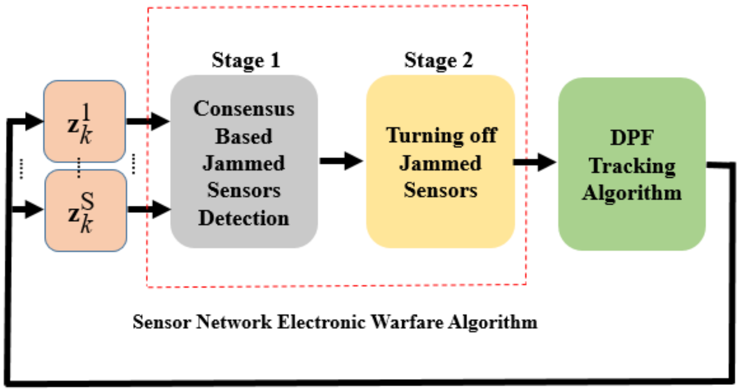

Stage 1
Stage 2
z_k^1
z_k^S
Consensus Based Jammed Sensors Detection
Turning off Jammed Sensors
DPF Tracking Algorithm
Sensor Network Electronic Warfare Algorithm

Fig. 4. Distributed target tracking algorithm with ability to detect and remove jamming effect .

Stage 1: in stage 1, after receiving the observations in the network nodes, the observations are first converted into the reference coordinate system through the following relationships.

$$x_{o,\,k}^{s} = x_s + r_{\mathbf{z},k}^{s} \cos(,\theta_{\mathbf{z},k}^{s}) \tag{16}$$

$$y_{o,\,k}^{s} = y_s + r_{\mathbf{z},k}^{s} \sin(,\theta_{\mathbf{z},k}^{s}) \tag{17}$$

Where $\mathbf{z}^{s}_{o,k}=[x^{s}_{o,k},y^{s}_{o,k}]^{T}$ is the coordinate of the observation of the node s in the reference coordinate system. reference coordination system concept is shown in figure 7. converting local coordinates to reference one is because the consensus algorithm can only be applied to observations in the same coordinate system.

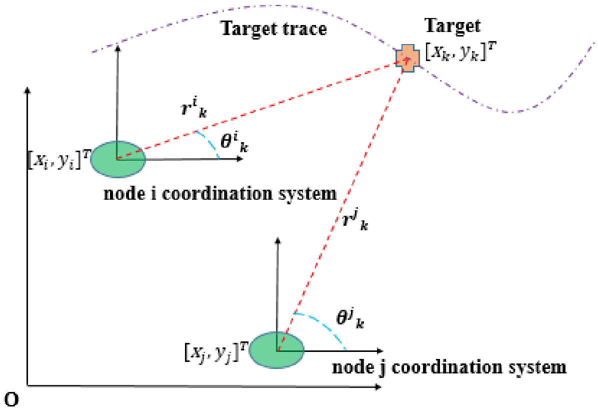


Fig. 5. Reference coordination system.

After transferring the observation coordinates of the network nodes to the reference coordinate system, the consensus algorithm is executed in the form of (18) in each node

$$\mathbf{z}^{i}_{o,k}(t+1)=w_{ii}\,\mathbf{z}^{i}_{o,k}(t)+\sum\nolimits_{j\in \mathrm{N_i}} w_{ij}\,\mathbf{z}^{j}_{o,k}(t) \qquad (18)$$

According to (18), each node exchanges information only with its neighboring nodes for a certain number of iterations. $\bar{\mathbf{z}}^{i}_{o,k}$ is the observation resulting from the average consensus at node i and time step k. According to (19) the difference in the observation of each node in the reference coordinate system $\mathbf{z}^{i}_{o,k}$ with the observation resulting from the average consensus algorithm $\bar{\mathbf{z}}^{i}_{o,k}$ is calculated at the location of each node.

$$\Delta\bar{\mathbf{z}}^{i}_{o,k} = \bar{\mathbf{z}}^{i}_{o,k} - \mathbf{z}^{i}_{o,k} \qquad (20)$$

In the nodes where the jamming signal is added to their observation, it is expected that the jamming infected observation will have more difference than the average observation of the network. The amount of this difference can be determined in (21)

$$\left|\frac{\Delta\bar{\mathbf{z}}^{i}_{k}}{\bar{\mathbf{z}}^{i}_{k}}\right| > \alpha \qquad (21)$$

α is called Relative Difference Threshold (RDT). In the jammed nodes, the left side of the inequity (21) is larger than the nodes on which jamming had no effect, and the greater the power of jamming RDT should be chosen larger to detect the jammed nodes more effectively.

Algorithm1: Distributed Jammer-Equipped Target Tracking with Hybrid Extended Kalman and Particle Filter

HEPF filter (according to Fig 1):
Depending on the maneuverability of the target, different filters are selected for the estimation process. For high maneuver targets, the first filter is chosen, while for low maneuver targets, the second filter is applied.
First filter (EKF+DPF):

```
1.  for i=1, …, S
        • Observations of all sensors will be averaged using EKFs
          and then local estimates are obtained in each node.
        • If $|\frac{\Delta\bar{z}_k^i}{\bar{z}_k^i}| > \alpha$, then the node is considered to be
          contaminated by jammer and its estimate is excluded from
          the fusion process.
    end
2.  for i=1, …, S
        • After purifying the observations from jamming signals, we
          apply the graph-based algorithm proposed in [4] for
          distributed estimation of the target state.
    end
second filter (EKF):
1.  for i=1, …, S
        • Observations of all sensors will be averaged using EKFs
          and then local estimates are obtained in each node.
        • If $|\frac{\Delta\bar{z}_k^i}{\bar{z}_k^i}| > \alpha$, then the node is considered to be
          contaminated by jammer and its estimate is excluded from
          the fusion process.
        •
    end
2.  for i=1, …, S
        • local state estimations are calculated by using EKF in a
          distributed manner.
    end
```

Stage 2: by identifying the jammed nodes in the first stage, observations of these nodes can be removed from entering the distributed tracking filters. The jammed nodes that are detected in the previous stage, remain in the network only for the distributed processing of the information of other nodes as well as telecommunication relay, but they don't inject the information obtained from the observation into the network. When the distributed particle filter runs average consensus on the local state estimations as shown in figure 1 the jammed nodes copy the estimation of the first node from their neighbouring nodes and discard estimation of their own due to jamming effects and the rest of the DPF steps will be done same as the normal case.

The pseudocode of distributed Jammer-Equipped Target Tracking with Hybrid Extended Kalman and Particle Filter is shown in Table 1.

4. MONTE CARLO SIMULATIONS

In this section, we simulate the distributed jammer-equipped target tracking algorithm in a sensor network and compare its performance with the results of tracking the same target by a single node. In the second case, we compare the results of the proposed algorithm with the results of tracking the mentioned target in the conventional sensor network, which does not use jamming detection and cancellation blocks in its distributed tracking filters. As seen in figure 6, the network graph used in this simulation includes 16 vertices, and the connections of each node with neighboring ones are also specified with dashed line in the figure. As explained, each node communicates information only with its one-hope neighbors in each repetition of consensus. The state transition model in this simulation is in the form of

$$\mathbf{x}_{k+1}=\mathrm{f}(\mathbf{x}_k,\xi_k)=\mathbf{A}\mathbf{x}_k+\xi_k \tag{22}$$

Which is a motion with a constant turn in the clockwise direction. **A** is constant turn matrix which is as

$$\mathbf{A}=\begin{bmatrix} 1 & 0 & \frac{\sin(C(k)T)}{C(k)} & -\frac{1-\cos(C(k)T)}{C(k)} \\ 0 & 1 & \frac{1-\cos(C(k)T)}{C(k)} & \frac{\sin(C(k)T)}{C(k)} \\ 0 & 0 & \cos(C(k)T) & -\sin(C(k)T) \\ 0 & 0 & \sin(C(k)T) & \cos(C(k)T) \end{bmatrix} \tag{23}$$

We make a simplifying assumption that the communication links between the one-hop sensors are ideal, i.e., there is no AWGN noise or packet loss in the data transmission To compare the performance of the used algorithms here, we have used Root Mean Square Error (RMSE) as

$$\mathrm{RMSE(k)}=\sqrt{\frac{1}{n_{\mathrm{MC}}}\frac{1}{S}\sum_{j=1}^{n_{\mathrm{MC}}}\sum_{l=1}^{S}(x_k^j-\hat{x}_k^l)^2+(y_k^i-\hat{y}_k^l)^2} \tag{24}$$

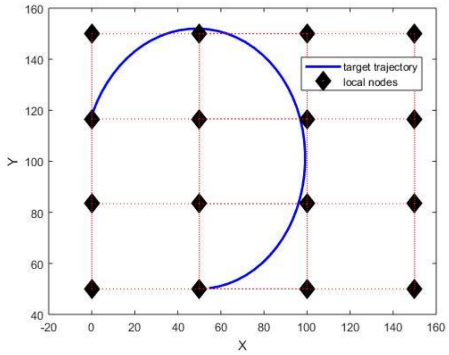


Fig. 6. target trajectory in the sensor network.

Where n_{MC} is number of Monte Carlo simulations, $\hat{x}_k^l$ and $\hat{y}_k^l$ are longitudinal and transverse components of target state estimation in node l respectively. We consider a scenario where 16 active radar sensors are deployed in a cluttered environment to track a moving target. Fig. 6 shows the sensor network topology and the target trajectory. Two sensors are considered to be single-hop neighbors if they can communicate directly with each other. The red dotted lines in Fig. 6 indicate the single-hop communication links between the sensors. If there is no dotted line between two sensors, it means that they are out of each other's communication range. The observation variance in distance measurement is equal to $\sigma_r^2 = 0.3m$ and simulation is repeated for 500 Monte Carlo simulations. In the nodes that are subject to jamming, the covariance value of the jamming noise is 30 dB higher than the observation noise, which makes the observation of these nodes completely incomprehensible. In this simulation ten percent of the network nodes that have a smaller distance from the target are jammed at each time step. We assume that all the nodes are identical, but since the target is equipped with a digital radio frequency memory (DRFM) jammer, only a limited number of WSN nodes are affected by this jammer due to the following two reasons:

1. The jammer has a limited memory for generating false targets, so it can only deceive a fraction of the nodes.
2. The antenna pattern for the DRFM is not omnidirectional, and only a few nodes are in its main lobe. In the simulations, we denote this fraction by α, which is set to 0.1, meaning that we assume that 10% of the nodes are jammed.

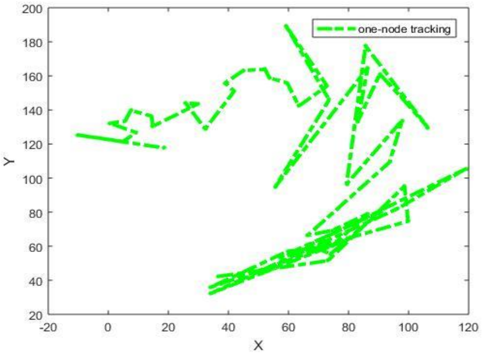


Fig. 7. Estimated target trajectory for single node.

In the figure 8 the performance of distributed target tracking in the sensor network using jamming detection and cancellation blocks is compared with single node target tracking.

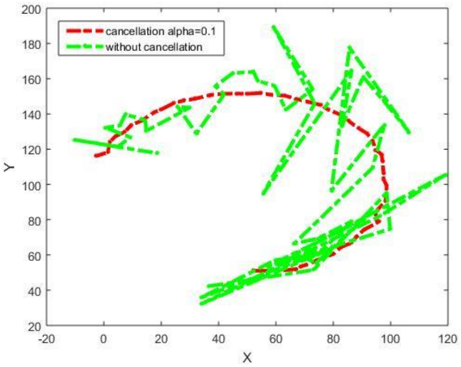


Fig. 8. Estimated target trajectory for sensor network with jamming cancellation and tracking by single node.

The tracking performance for single node is shown in Table 1. We adopt the root mean square error (RMSE) as the performance metric to assess the accuracy of the proposed algorithm for distributed estimation of the target state. As can be seen in table 1, with the defined criterion, tracking RMSE of a single node is high and as a result the tracking performed is not acceptable. The estimated trajectory of the target for single node is drawn in the figure 7. RMSE results of distributed target tracking using the jamming cancellation algorithm is shown in Table 1 at different time steps

Table 1. RMSE (m) for different scenarios.

Time-step number	10	20	30	40	50
RMSE (m) (using only single node)	51	38	68	48	62
RMSE (m) (excluding contaminated sensors)	1.96	1.46	1.2	1.35	2.01
RMSE (m) (using all contaminated sensors)	10.52	12.24	13.35	11.5	14.35

5. CONCLUSION

This paper presented jammer-equipped target tracking in a sensor network using distributed particle filters. A robust distributed method based on the consensus algorithm was presented here which can discover the nodes of the network that are subject to jamming and remove them from the distributed target tracking algorithm. To illustrate the effectiveness of the proposed method, we compare the RMSE values of the target state estimation at the 30th time step under different scenarios. When only one sensor is used for tracking, the RMSE value is 68 meters. When all the sensors in the network, including the ones affected by jamming, are used, the RMSE value drops to 13.35 meters. When the proposed method is applied to identify and exclude the jammed sensors, the RMSE value reduces to 1.2 meters. This shows that using all the sensors improves the performance by a factor of 5, and using the proposed method further improves the performance by a factor of 56.6. Therefore, the proposed method can significantly enhance the accuracy of the distributed estimation in the presence of jamming signals.

REFERENCES

[1] W. Xia, S. Meiqiu, W, Qian, "**Direct Target Tracking by Distributed Gaussian Particle Filtering Based on Delay and Doppler**" IEEE Transactions on Signal Processing, 2020, vol. 68, pp. 1361-1373.

[2] J. Fan, Y. Guo, K. Wang. "**Communication-efficient accurate statistical estimation**" Journal of the American Statistical Association, vol. 118 (542), 2023, pp. 1000-1010.

[3] M. Coates, "**Distributed particle filters for sensor networks**" 3rd international symposium on Information processing in sensor networks, 2004.

[4] I J. Y. Yu, M. J. Coates, M. G. Rabbat, "**Graph-Based Compression for Distributed Particle Filters**" IEEE Transactions on Signal and Information Processing over Networks. vol 5(3), 2019, pp. 404-417.

[5] A. Mohammadi, and A. Asif, "**Distributed Consensus + Innovation Particle Filtering for Bearing/Range, Tracking with Communication Constraints**" IEEE transactions on Signal Processing, 2014, vol. 63, no. 3, pp. 620–635.

[6] A. Mohammadi, and A. Asif, "**A constraint sufficient statistics based distributed particle filter for bearing only tracking**" IEEE International Conference on Communications (ICC), 2012.

[7] X. Sheng, Y.H. Hu, P. Ramanathan, "**Distributed particle filter with GMM approximation for multiple targets localization and tracking in wireless sensor network**" Fourth International Symposium on Information Processing in Sensor Networks, Los Angeles, California, USA, 2005, pp. 181-188.

[8] O. Hlinka, O. Sluciak, F. Hlawatsch, P. Djuric, and M. Rupp, "**Likelihood consensus and its application to distributed particle filtering**" IEEE Transactions on Signal Processing, 2012, vol. 60, no. 8, pp. 4334–4349.

[9] S.A.Vakin,"**Fundamentals of Electronic Warfare**" Artech House, 2001.

[10] A,Graham,"**Communications, Radar and Electronic Warfare**" John Wiley & Sons, 2011.

[11] D, Adamy, "**Introduction to Electronic Warfare Modeling and Simulation**" Artech House, 2003.

[12] Y. Ye, D. Kai, Z. Yongfeng, X. Shengwen and F. Qiang, "**Consensus based target tracking against deception jamming in distributed radar networks**" IET Radar, Sonar & Navigation, 2023.

[13] Y. Ye, D. Kai, Z. Yongfeng and F. Qiang, "**Consensus-based distributed target tracking in the presence of active false targets**" 2021 IEEE 2nd International Conference in Big Data, Artificial Intelligence and Internet of Things Enfinneering (ICBAIE). IEEE, 2021, pp. 753-757.

[14] J. Thangapoo Nancy, K.P. Vijaya Kumar and P. Ganesh Kumar, "**Detection of Jammer in Wireless Sensor**" 2014 Internatonal Conference on Communication and Signal Processing. IEEE, 2014, pp. 1435-1439.

[15] C. Tianzhen, L. Ping and Z. Sencun, "**An Algorithm for Jammer Localization in Wireless Sensor Networks**" 2012 IEEE 26th international conference on advanced information networking and applications. IEEE, 2012, pp. 724-731.

[16] X. Wenyuan, T. Wade and Z. Yanyong, "**Jamming Sensor Networks: Attack and Defense Strategies**" IEEE network. vol. 20, no. 3, 2006, pp. 41-47.

[17] K.P. Porkodi, I. Kartika and H.K. Gianey, "**Localization and Tracking of Mobile Jammer Sensor Node Detection in Multi-Hop Wireless Sensor Network**" Recent Advances in Computer Science and Communications (Formerly: Recent Patents on Computer Science). vol. 14, no. 1, 2021, pp. 54-61.

[18] P. Aggarwal, Z. Syed and N. El-Sheimy, "**Hybrid Extended Particle Filter (HEPF) for Integrated Civilian Navigation System**" 2008 IEEE/ION Position, Location and Navigation Symposium. IEEE, 2008.

[19] A. Farnia, M. Skolnik, "**Electronic counter-countermeasures**" Radar Handbook, vol. 2. McGraw-Hill, New York, 2008. pp. 0964-0969.

[20] S .D Berger, "**Digital radio frequency memory linear range gate stealer spectrum**" IEEE Transactions on Aerospace and Electronic Systems, vol. 39, no. 2, 2003, pp. 725-735.

Semi-Blind Separation of Multiple Asynchronous Wideband Frequency Hopping Signals Based on MWC and Spectral Entropy Method

Mohsen Rezaee[1], Morteza Babaei [2], Mohammadreza Motedayen [3]
1- ICT Research Institute, Tehran, Iran.
Email: m.rezaeekh@itrc.ac.ir (Corresponding Author)
2- Imam Hossein Comprehensive University, Tehran, Iran.
Email: mbabaei @ihu.ac.ir
3- Faculty of Electrical Engineering Department, Imam Hossein University, Tehran, Iran.
Email: p9519894887@ ihu.ac.ir

ABSTRACT:
Wideband Frequency Hopping Spread Spectrum (FHSS) communications are widely used in both military and commercial applications. In military applications, it is very important to investigate these communications, especially when frequency hopping signals are received simultaneously by a single antenna. This paper investigates the problem of estimating interfering wideband asynchronous frequency hopping (FH) signals parameters with the same hop rate, and using narrow-band receivers. Due to minimal knowledge about the transmitted signals, the problem is analyzed in semi-blind mode. For this purpose, time-frequency (TF) processing has been applied to the modulated wideband converter (MWC) output. The proposed method consists of two stages; In the first stage, frequency-hopping signals with different amplitudes are received by a single antenna. By passing through baseband receivers, the TF distribution of the converter's output signal is obtained. In the next stage, by computing instantaneous spectral entropy (SE), and finding the local maxima in the spectrum, the hop time of each signal is obtained. We use MWC for sub-Nyquist sampling and simultaneous extraction of time and frequency information from signals while eliminating irrelevant signals. The results obtained from estimating hop time parameters demonstrate improved performance compared to other traditional source separation methods such as sparse linear regression (SLR). Based on evaluation metrics such as root mean squared error (RMSE), in lower signal-to-noise ratio (SNR) values, the amount of error has been substantially reduced.

KEYWORDS: Semi-Blind Separation, Modulated Wideband Converter, Time-Frequency Distribution, Spectral Entropy, Wideband Frequency-Hopping Signal.

1. INTRODUCTION

In recent years, spread spectrum telecommunications has been expanded due to its appropriate performance in different applications [1]. In the meantime, Frequency Hopping (FH) communication is one of the most essential types of spread spectrum communications in which the signal is controlled by a pseudo-random sequence and is spread across the entire spectrum [2].

Frequency-hopping communication is widely used in military and civilian communications due to its low probability of detection (and intercept) [3-5], high reliability, anti-jamming capability, and low sensitivity. Considering these features, estimating the parameters of FH signals is difficult, especially when there exist multiple interfering signals with no prior information about hop patterns [6-7]. In practical communications, the electromagnetic environment is very complex, especially in the short wave band, where all types of environmental signals are mixed and make it challenging to

Paper type: Research paper
https://doi.org/
Received: 13 January 2024; revised: 1 February 2024; accepted: 19 March 2024; published: 1 June 2024
How to cite this paper: M. Rezaee, M. Babaei, M. R. Motedayen, **"Semi-Blind Separation of Multiple Asynchronous Wideband Frequency Hopping Signals Based on MWC and Spectral Entropy Method"**, *Majlesi Journal of Telecommunication Devices*, Vol. 13, No. 2, pp. 97-111, 2024.

detect FH signals [8]. The identification of FH signals typically occurs in two stages: parameter estimation and signal sorting [9-11]. Estimating parameters such as hop time [12], frequency [13], and angle of arrival (DOA) [14-16], is crucial for sorting multiple users without prior information in practical applications.

On the other hand, MWC structure is widely used in cognitive radar [18], estimating the direction of arrival [19], and detection of FH signals of spread spectrum systems [20], due to its features such as flexibility, sub-Nyquist sampling [17], easy implementation, etc.

In many articles, it is used to implement compressed sensing (CS) methods. However, the CS-based MWC structure has several disadvantages, such as imposing sparse condition constraints on input signals, analyzing the problem only in an underdetermined state, poor performance with reduced SNR, etc. In [21], the structure of MWC is used based on channeling and energy detection, but it is focused on a single signal without time information. In [20], MWC-based frequency analysis with dynamic programming (DP) is considered track and sort wideband FH signals in two steps. Also, the stable performance of the DP section is only for signal-to-noise ratios above 9 dB, indicating poor performance in low SNR.

FH signal analysis requires a suitable method to express time-varying signals that can simultaneously represent time and frequency variations of the signal [22]. In this regard, TF analysis is widely used to estimate FH signal parameters [3], [23-24]. In a classification for non-stationary signals, the analysis of FH signals is divided into two aspects: estimation methods based on time-frequency analysis and estimation methods without it [25]. Generally, TF analysis is divided into two parts: linear and quadratic-order analysis. Linear TF analysis mainly includes Short Time Fourier Transform (STFT), Gabor Transform, and Wavelet Transform (WD) [22]. The window used in STFT distribution is rectangular, while the window used in the Gabor transform is a Gaussian function. The rectangular function in STFT has a fixed width, so STFT has poor time-frequency resolution [26]. In addition, the time-frequency resolution of the Gabor transform is better than STFT. Also, among these transformations, wavelet transformation is one of the most effective TF analysis tools, providing a suitable approximation for signals that exhibit sharp discontinuities. However, it is a method that requires prior knowledge of a completely known signal. Generally, the main TF analysis tools [3], [27], include the STFT analysis, which has less computational complexity compared to other TF distributions. Still, its estimation error is slightly higher than other distributions [28]. The spectrogram (SPEC) distribution is considered one of the simplest quadratic-order TF analysis methods, where the time-frequency representation of the non-stationary signal energy is shown [29]. An analysis based on the Wigner-Ville distribution (WVD) provides better resolution than the SPEC distribution. Still, it introduces cross-term interference components [3]. An analysis based on the Smoothed Pseudo Wigner-Ville Distribution (SPWVD) detects cross-term in a non-cooperative environment. However, it leads to increased computational complexity and reduced performance at low SNR [24], [30-33]. An analysis based on the Re-Smoothed Pseudo Wigner-Ville Distribution (RSPWVD), which overcomes the effects of cross-term in the SPWVD distribution, and leads to increasing computational load [23]. Meanwhile, the STFT distribution analysis is linear, and the rest of the distributions are of second order. Also, in [34-35], the combined STFT-SPWVD method has been used to estimate FH signal parameters. However, the performance of this method is significantly affected by noise, and it has a high computational load [36]. Moreover, in [37] which aimed at direction finding of FH signals, a method based on relative entropy has been employed to extract a hop-free dataset, and focused on estimating the direction of arrival of the signal.

In this article, semi-blind separation of multiple asynchronous wideband FH signals using MWC and spectral entropy in the TF domain is addressed. For this purpose, in Section 2, the signal model for receiving N signals by a single antenna is investigated. In Section 3, the mathematical relationships for the proposed method, including the analysis of the MWC structure and instantaneous SE method in estimating hop times are presented. In Section 4, the results obtained from simulating the proposed method will be evaluated. Finally, in Section 5, the conclusions are presented.

2. SIGNAL MODEL

Suppose a set of N frequency-hopping signals $\mathrm{S} = \{s_{\mathbf{1}}(t)\text{،}s_{\mathbf{2}}(t)\text{،} \dots \text{،}s_{\mathbf{N}}(t)\}$, which are time-varying, asynchronous, non-stationary signals generated by fixed sources and received through a single antenna in the far field. For asynchronous FH signals, we assume the dwell time is the same. The analytical representation of the n-th received signal, at time $t \in [t_{k-1}\text{،}t_k)$ is:

$$s_n(t) = \sum_{k=1}^{\mathrm{N_h}} \alpha_{n\text{،}k}(t) e^{j\left(2\pi f_{n\text{،}k}(t)t + \varphi_{n\text{،}k}(t)\right)} \quad \text{،}\ t_{k-1} \le tk \tag{1}$$

Where $\alpha_{n\text{،}k}(t)$, $f_{n\text{،}k}(t)$ and $\varphi_{n\text{،}k}(t)$ are the amplitude, instantaneous frequency, and phase of the n-th FH signal in the k-th hop instant, respectively, and $\mathrm{N_h}$ is the number of frequency hops of $s_n(t)$. The k-th hop timing is defined as $t_k \in [0\text{،}\mathrm{T}]$, and T is the observation time.

Considering linear combining model for a single antenna, the received signal $x(t)$ reads

$$x(t) = \sum_{n=1}^{N} k_n s_n(t) + n(t) \qquad ،0 < t \leq T \quad (2)$$

Where k_n is the attenuation coefficient of the n-th signal's amplitude, and $n(t)$ is the added white Gaussian noise with zero mean and variance σ^2.

3. THE PROPOSED METHOD

In non-cooperative communications, the receiver lacks a pseudo-random sequence generator similar to the transmitter and cannot receive the frequency-hopping pattern of the signals. These receivers consist of three main parts: signal detection, parameter estimation, and separation of combined signals. Non-cooperative receivers usually receive a combination of FH signals simultaneously. Therefore, to separate the signals, the parameters of each signal are estimated [21].

We propose a semi-blind method for separating multiple asynchronous FH signals, using the MWC, TF analysis and applying an instantaneous SE-based method. The MWC structure is utilized due to its sub-Nyquist sampling [17], TF analysis is used because of its simplicity and fast computations [40] in analyzing time-varying signals [41] and also for evaluating the parameters of FH signals in blind mode [42]. Additionally, SE method is used for estimation of signal parameters such as hop times [37]. Some features of the proposed MWC_TF_SE method for the Blind Source Separation (BSS) process are as follows:

1- Signal processing capability in the baseband [38] and sub-Nyquist sampling [17], [39], employing a single antenna.
2- Analysis of MWC structure in TF domain and simultaneous extraction of time and frequency information of signals.
3- Separation of signals based on the TF_SE method.
4- Removal of other interfering signals.

In summary, the proposed method consists of two stages for solving the BSS problem:

(1) Passing signals through MWC, and transfer to the TF domain.
(2) Instantaneous SE method in the estimation of signal parameters.

In the following, the proposed method is presented in detail.

3.1. MWC structure

In the context of the BSS process using a narrow-band receiver, two flexible and popular structures of random demodulator (RD) and MWC can be mentioned. These structures can be employed in spectrum sensing to alleviate the limitations caused by the bandwidth shortage of receivers for wideband signals. In the meantime, the RD is a well-known single-branch structure for recovering the input signal, with m_0 limited frequency components, such as $\{e^{jl\omega_0}\}_{l=1}^{m_0}$. It is suitable for implementing analytical methods like CS. However, it leads to complex integral computations. Therefore, it is possible to use an alternative wideband converter, such as MWC, which is a generalization of the RD structure Some of the features of multi-branch compared to single-branch MWC are:

1. Parallel Processing and Data Diversity: Multiple branches allow parallel processing of the input signal, which can lead to the extraction of diverse features from different branches simultaneously. This can enhance the system's ability to capture various characteristics of the input signal.

2. Improved Signal Separation: In cases where the input signal contains different components or sources, multiple branches can help separate and process these components independently, potentially leading to more accurate signal decomposition.

3. Enhanced Noise Robustness: By processing the input signal through multiple branches and then combining the results, noise and interference from one branch may be mitigated by the information gathered in other branches, resulting in a more robust output.

4. Feature Combination: Having multiple branches allows for the combination of features extracted from each branch, potentially yielding more comprehensive and informative representations of the input signal.

Fig. 1 illustrates the standard structure of MWC. Consider the frequency band range $\mathcal{F} \triangleq [-f_{nyq}/2،f_{nyq}/2]$ for the signal $x(t)$, where $f_{nyq}/2$ is the upper limit of the frequency band $x(t)$.

According to the figure (1), the signal $x(t)$ is multiplied by a set of periodic sequences $p_i(t)،i \in \{1، \ldots ،M\}$. These periodic sequences have a period of $T_p = 1/f_p$ and a frequency range of $\mathcal{F}_p \triangleq [-f_p/2،f_p/2]$, as defined by

(3

Then, MWC passes the resulting product through the low-pass filter $h(t)$ in the frequency range $\mathcal{F}_s \triangleq [-f_s/2،f_s/2]$, and it samples the filter output uniformly at a frequency of f_s, while satisfying the conditions of $f_s \geq f_p$ ، $f_s \ll f_{nyq}/2$ [17]. In this case, the output signal of the i-th branch is equal to

$$y_i[n] = \left(\left(x(t) \times p_i(t)\right) \underset{t}{*} h(t)\right)\Big|_{t=nT_s} \qquad = \int_{-\infty}^{\infty} x(\tau)p_i(\tau)h(t-\tau)dt\Big|_{t=nT_s} \tag{4}$$

Where $x(t)$ is the input signal, $p_i(t)$ is the periodic sequence of the i-th branch, $h(t)$ is the baseband filter, $\boldsymbol{y}_i = \{y_i[0]، y_i[1]، \dots ،y_i[\mathrm{L}_y - 1]\}$, L_y is the length of the $\boldsymbol{y}_i$, and $T_s = 1/f_s$ is the sampling time interval. Considering M branches of MWC, the set of vectors $\boldsymbol{y} = \{\boldsymbol{y}_1 ،\boldsymbol{y}_2 ، \dots ،\boldsymbol{y}_\mathrm{M}\}$ is obtained at the output of MWC.

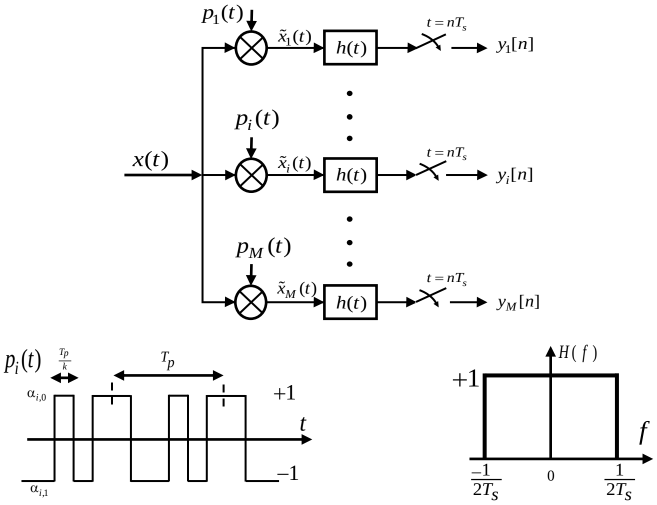


Fig1. MWC structure consisting of M parallel channels [17]

By simultaneously processing each of the signals in the set $\boldsymbol{y}$, a set of signal features will be obtained. By processing these features, the desired parameter can be extracted. But the increase in the number of MWC branches leads to an increase in the volume of calculations. Therefore, it is possible to select one of the signals of the set $\boldsymbol{y}$ and perform further processing on it. In this case, the effects of noise and interference in the selected signal are less observed compared to other members. In this article, the second mode is chosen.

3.1.1. TFD of MWC output data

TF distribution is an analytical tool that provides time and frequency information of signals simultaneously. In the BSS process using TF distribution leads to recovery and estimating signal parameters.

According to the Eq. (*a*-5) in Appendix (A), the discrete TF distribution of the observed signal at the output of the i-th MWC branch is equal to

$$\boldsymbol{\rho}_{y_i}[n،k] = \sum_{|m| < \frac{N_{\boldsymbol{G}}}{2}} \boldsymbol{G}[n،m] \underset{n}{*} (y_i[n+m]y_i{}^*[n-m])e^{-j2\pi km/N_{\mathbf{G}}} \tag{5}$$

where y_i^* is the conjugate of y_i, $\boldsymbol{G}[n،m]$ is the time-lag kernel, and $[n،m] \in \mathbb{Z}$. By considering M branches of MWC, the set of distributions $\{\boldsymbol{\rho}_{y_i}[n،k]\}_{i=1}^{\mathrm{M}}$ is obtained where $\boldsymbol{\rho}_{\boldsymbol{y}_i}[n،k]$ is the discrete TF distribution of the observed vector $\boldsymbol{y}_i$.

In the next stage of the signal separation process, by applying the instantaneous SE method on the distribution $\boldsymbol{\rho}_{\boldsymbol{y}_i}[n،k]$, we will take a significant step toward recovering the signals.

3.2. Spectral entropy

The spectral entropy of a signal is a measure of its spectral power distribution. The concept is based on the Shannon entropy, or information entropy. The SE treats the signal's normalized power distribution in the frequency domain as a probability distribution, and calculates its Shannon entropy. This property is useful for feature extraction [43-44], and is also widely used in areas such as signal processing [37], and biomedical signal processing [45].

FH signal structure has a unique feature compared to other signals. When the frequency changes, a discontinuity will appear in the frequency spectrum of the signal. Methods based on entropy, variance, etc, are sensitive to discontinuous points in the signal, and can be used to extract information from these points. In this article, by applying the instantaneous SE method to the $\boldsymbol{\rho}_{y_i}[n،k]$ matrix, such as the STFT matrix, the hopping times of FH signals are determined.

3.2.1. Instantaneous spectral entropy estimation

To estimate SE, we obtain the TFD matrix in the output of one of the MWC branches, such as the SPEC distribution. According to Eq. (a-7) of Appendix (A), the STFT distribution for the i-th MWC branch is equal to

$$\boldsymbol{\rho}_{\boldsymbol{y}_{i\_STFT}}[n،k]=\sum_{m=1}^{\mathrm{M}_\omega}\boldsymbol{y}_i[m+n]\omega[m]e^{\frac{-j2\pi km}{\mathrm{M}_\omega}} \qquad (6)$$

where $\boldsymbol{\omega}=\{\omega[1]، \omega[2]، \dots ،\omega[\mathrm{M}_\omega]\}$ is the window function, and M_ω is the number of samples of $\boldsymbol{\omega}$. In the STFT distribution, an analysis window, such as Hamming window, slides over the signal and then the Discrete Fourier Transform (DTF) is computed for the windowed data. The number of rows in the STFT matrix equals the number of DFT points, and the number of columns is given by $l=\left\lfloor\frac{\mathrm{L}_y-\mathrm{L}}{R}\right\rfloor ، l\geq 2$, where L_y is the length of $\boldsymbol{y}_i$, and $R=\mathrm{L}_\omega-\mathrm{L}$ represents the hop distance between consecutive DFT, L_ω is the length of $\boldsymbol{\omega}$.

Each column of the STFT matrix is a spectral segment of the signal and provides a short-term estimate of the time-frequency content of the signal [43]. The STFT matrix in terms of spectral segments can be expressed as $\boldsymbol{Y}_i(f)=[Y_{i،1}(f)\,Y_{i،2}(f)\dots\,Y_{i،m}(f)\dots\,Y_{i،l}(f)]$. The element corresponding to m-th row and h-th column of the matrix $\boldsymbol{Y}_i(f)$ is equal to

$$Y_{i،m}(f_h)=\sum_{n=-\infty}^{\infty}y_i[n]\omega[n-mR]e^{-j2\pi f_h n} \qquad (7)$$

For $f_h\in\left\{f_1، f_2، \dots ، f_{N_f}\right\}$, where N_f is the number of DFT points,and the vector $Y_{i،m}(f)=\left[Y_{i،m}(f_1)، Y_{i،m}(f_2)، \dots ، Y_{i،m}\left(f_{N_f}\right)\right]^{\mathrm{T}}$ is obtained [43].

The probability function for each column of the STFT matrix is used for detecting hop times. Columns, with frequency hops have high entropy due to the presence of instantaneous frequency variations. Entropy can be regarded as a measure of uncertainty between pairwise elements of a discrete set. Any possibility of displacement between the elements of this set, which have close probabilities, increases the entropy of the collection. In a time window with frequency hopping from one moment to another, it causes the signal to show its spectrum dispersion with a high entropy value.
Generally, the spectral entropy in the domain $(t،f)$ is defined in terms of TF distributions as follows [22]:

$$\boldsymbol{\mathcal{SE}}=-\sum_{n=1}^{N_t}\sum_{k=1}^{N_f}\frac{\boldsymbol{\rho}_{\mathbf{y}}[n،k]}{\sum_n\sum_k\boldsymbol{\rho}_{\mathbf{y}}[n،k]}\log_2\left(\frac{\boldsymbol{\rho}_{\mathbf{y}}[n،k]}{\sum_n\sum_k\boldsymbol{\rho}_{\mathbf{y}}[n،k]}\right) \qquad (8)$$

where N_t , N_f are the total time and frequency points, respectively. The above relation is only valid for the TFDs that do not assume negative values, e.g., the SPEC. Considering the STFT distribution, the instantaneous SE of the i-th MWC branch is equal to

$$\mathcal{SE}_{i،(t)}=-\sum_{k=1}^{N_f}\frac{\boldsymbol{\rho}_{\mathbf{y}_{i\_STFT}}[n،k]}{\sum_k\boldsymbol{\rho}_{\mathbf{y}_{i\_STFT}}[n،k]}\log_2\left(\frac{\boldsymbol{\rho}_{\mathbf{y}_{i\_STFT}}[n،k]}{\sum_k\boldsymbol{\rho}_{\mathbf{y}_{i\_STFT}}[n،k]}\right) \qquad (9)$$

where $\mathcal{SE}_{i،(t)}\in\mathbb{C}^{N_t}$. In every time window, as long as the frequency is constant, the elements of $\mathcal{SE}$ remain constant, as soon as the frequency changes, the elements of $\mathcal{SE}$ show significant change and is considered a hop time.

Therefore, by applying the SE method to the STFT matrix of the SPEC distribution, a set of peaks is created in the spectrum. In the next step, by detecting and processing these peaks, signal parameters are separated and estimated.

3.2.2. Estimating hop times of signals

In practice, the presence of frequency hop in the signal leads to an increase in the SE spectrum. Therefore, detecting peak points is essential for extracting time and frequency measurement information from the signal.

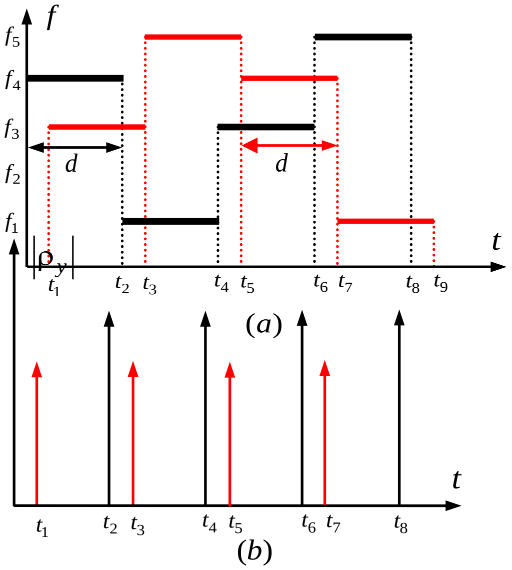

f
f_5
f_4
f_3
f_2
f_1
d
d
t
ρ_y
t_1 t_2 t_3 t_4 t_5 t_6 t_7 t_8 t_9
(a)
t
t_1 t_2 t_3 t_4 t_5 t_6 t_7 t_8
(b)

Fig. 2. (*a*) shows the sample distribution of SPEC for two asynchronous FH signals with four frequency hops, (b) the SE of Figure (*a*) in the ideal state, and the creation of impulse functions at hop times.

Fig. (*a*-2) shows the SPEC distribution of two asynchronous FH signals with four frequency hops. The frequency dwell time, d, is the same for all frequency hops in both signals. Figure (*b*-2) shows the performance of the TF_SE method in an ideal state, where a set of impulse functions is created at the time instants of signal hops.

According to Eq. (10), extraction of signal peaks based on instantaneous SE and obtain a set of frequency-hopping times $\{\hat{t}_{i،hop}\}$ of the signals.

$$\hat{t}_{i،hop} = \arg\left\{\max_t\left[\left|\mathcal{SE}_{i،(t)}\right|\right]\right\}، \; 0 \le \hat{t}_{i،hop} \le \mathrm{NN_h} \qquad (10)$$

where $\mathrm{N_h}$ represents the number of frequency hops per signal, and N is the number of signals. Increasing SNR in the SE method leads to a reduction in false peaks and improvement in the estimating the set of time instants $\{\hat{t}_{hop}\}$. Considering that all equations are related to a single branch, in order to simplify the equations, the i variable has been omitted in the following equations.

3.3. Separation of hop times

By Applying the TF_SE method, a set of hop times, $\{\hat{t}_{hop}\}$, is obtained. In this stage, to separate the hop times of each signal, we first sort the set of times $\{\hat{t}_{hop}\}$ in ascending order. Then, by measuring the difference between each successive pair of time values $\{\hat{t}_{hop}\}$,

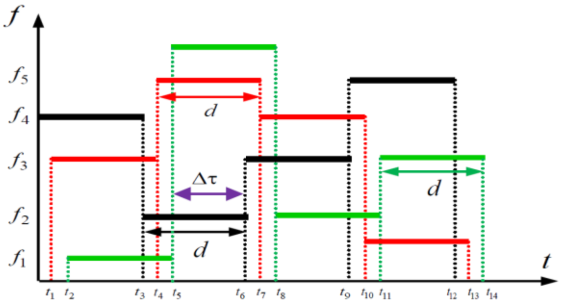

f
f_5
f_4
f_3
f_2
f_1
d
$\Delta\tau$
d
d
t
t_1 t_2 t_3 t_4 t_5 t_6 t_7 t_8 t_9 t_{10} t_{11} t_{12} t_{13} t_{14}

Fig. 3. The representation of three asynchronous FH signals, each with four hopping frequencies and the same dwell time.

a set of rounded numbers is obtained as follows:

$$\begin{aligned} t_{hopdif} &= \{\Delta\hat{t}_{hop}\} \qquad (11) \\ &= \mathrm{diff}\,\{t_1، t_2، t_3، t_4، t_5، t_6، t_7، \ldots\} \end{aligned}$$

$$= \{\Delta t_1 ‘ \Delta t_2 ‘ \ldots ‘ \Delta t_m ‘ \ldots ‘ \Delta t_{\hat{k}}\}$$

where $\hat{k} = \mathrm{NN_h} - 1$ and $\Delta t_m = t_m - t_{m-1}$ for $1 < m \leq \hat{k}$.

It is assumed that all three signals have the same dwell time as shown in Figure (3). Some of the features of the t_{hopdif} vector are as follows:

- In the t_{hopdif} set, the highest frequency (with a value of $\mathrm{N_h}$) corresponds to the time delays between the signals. that is,

$$Max\left(\Delta\hat{t}_{hop}\right) = \{\Delta t_1 ‘ \Delta t_2 ‘ \Delta t_4 ‘ \Delta t_5 ‘ \Delta t_7 ‘ \Delta t_8 ‘ \ldots$$
$$\ldots ‘ \Delta t_{\widehat{\mathrm{N}}_\mathrm{h}-5} ‘ \Delta t_{\widehat{\mathrm{N}}_\mathrm{h}-4} ‘ \Delta t_{\widehat{\mathrm{N}}_\mathrm{h}-2} ‘ \Delta t_{\widehat{\mathrm{N}}_\mathrm{h}-1}\}$$

where $\widehat{\mathrm{N}}_\mathrm{h} = \mathrm{NN_h}$, $\Delta t_1 = t_1 - 0$, $\Delta t_2 = t_2 - t_1$, $\Delta t_4 = t_4 - t_3$,… and that,

$$\Delta\tau_1 = \Delta t_1 = \Delta t_4 = \Delta t_7 = \Delta t_{10} = \cdots = \Delta t_{\widehat{\mathrm{N}}_\mathrm{h}-2}$$
$$\Delta\tau_2 = \Delta t_2 = \Delta t_5 = \Delta t_8 = \Delta t_{11} = \cdots = \Delta t_{\widehat{\mathrm{N}}_\mathrm{h}-1}$$

where $\Delta\tau_1$ represents the delay time between the first and second signals, and $\Delta\tau_2$ represents the delay time between the second and third signals.

- In the set t_{hopdif}, after $\Delta\tau_1$, the highest frequency (with a value of $\mathrm{N_h} - 1$) corresponds to the set $\{\Delta t_3 ‘ \Delta t_6 ‘ \Delta t_9 ‘ \ldots ‘ \Delta t_{\widehat{\mathrm{N}}_\mathrm{h}-3}\}$, In which, $\Delta t_3 = t_3 - t_2$, … and that,

$$\Delta\tau = \Delta t_3 = \Delta t_6 = \Delta t_9 = \ldots = \Delta t_{\widehat{\mathrm{N}}_\mathrm{h}-3}$$

So, for three signals, $t_\tau = \{\Delta\tau_1 ‘ \Delta\tau_2\}$ represents the time delay between the signals, and $t_C \approx \Delta\tau + \Delta\tau_1 + \Delta\tau_2$ will dwell time d.

- In the t_{hopdif} set, the repetition of all three consecutive components indicates the accuracy of the MWC_TF_SE method, in estimating the two parameters t_τ and t_C. According to Figure (3), and Eq.(11), the equality $\Delta\tau_1 = \Delta t_m$, which means that in the m-th hop, the m-th component of all three signals is correctly estimated.
- Previous properties of the t_{hopdif} set are discussed in the ideal state, but in practice, factors such as environmental noise, and SNR reduction lead to a decrease in the frequency of $\Delta\tau_1$ and $\Delta\tau_2$.
- So, Increasing the frequencies $\Delta\tau_1$ and $\Delta\tau_2$ in the t_{hopdif} set, leads to an increase the accuracy of the TF_SE method.

Generally, for N asynchronous FH signals, assuming the dwell time is the same, the set of hop times for each signal can be recovered and separated using two parameters t_τ and t_C, as follows:

$$\text{Signal\_1}: \{0 ‘ t_C ‘ 2t_C ‘ \ldots\}$$
$$\text{Signal\_2}: \{\Delta\tau_1 ‘ t_\mathrm{C} + \Delta\tau_1 ‘ 2t_\mathrm{C} + \Delta\tau_1 ‘ 3t_\mathrm{C} + \Delta\tau_1 ‘ \ldots\}$$
$$\text{Signal\_3}: \{\Delta\tau_1 + \Delta\tau_2 ‘ t_\mathrm{C} + (\Delta\tau_1 + \Delta\tau_2) ‘ \ldots \quad 2t_\mathrm{C} + (\Delta\tau_1 + \Delta\tau_2) ‘ \ldots\} \vdots$$
$$\text{Signal\_N}: \{(\Delta\tau_1 + \Delta\tau_2 + \cdots + \Delta\tau_{N-1}) ‘ \ldots$$
$$t_\mathrm{C} + (\Delta\tau_1 + \Delta\tau_2 + \cdots + \Delta\tau_{N-1}) ‘ \ldots$$
$$2t_\mathrm{C} + (\Delta\tau_1 + \Delta\tau_2 + \cdots + \Delta\tau_{N-1}) ‘ \ldots\}$$

By extracting the hop times from the STFT matrix of the TF distribution, other information, such as hopping frequencies, energy density of T-F plane points, period, and frequency hop of each signal based on matrix STFT, can be determined and separated.

One of the advantages of the proposed method is that in a real environment, FH signals have time interference with other environmental signals, such as burst, constant, periodic, and other signals. However, if the two parameters t_τ and t_C are accurately estimated, the interfering signals will not impact the estimation of signal parameters, and are easily eliminated.

Because, the basis of recovery of signal hopping times is based on the difference frequency of both consecutive components of the time vector $\{\hat{t}_{hop}\}$, and disturbing signals lack frequency due to scattering. Therefore, they have little effect on the estimation process of t_τ and t_C, and can be easily removed.

Also, in the proposed method, according to the level of signal contamination with noise, the accuracy of estimating time parameters decreases, and leads to an increase in estimation error. Therefore, improving processing variables, such as increasing SNR, increasing input samples, increasing experiment repetitions, etc, results in a reduction of estimation errors for time parameters t_τ and t_C, consequently leading to increased accuracy in separating the time intervals of each signal hop.

4. EVALUATION OF THE PROPOSED METHOD

This section is intended to validate the proposed method in the article, and it aims to evaluate, and calculate errors in the estimated parameters. We consider three asynchronous FH signals, each of these signals consists of six frequency hops (Nh $= 6$). The signals have different magnitudes [1.4 1.6 1.7], zero phases, different hop times, and the same hop length.

The signals are placed in the frequency band range of $50\text{MHz} \le f \le 550\text{MHz}$, without frequency interference. Table (1) shows the normalized frequency-hopping pattern of the three input sources.

Table 1. Frequency-hopping pattern of input signals.

Inputs	Normalized hopping frequencies
Signal (1)	$\{0.05،\ 0.14،\ 0.445،\ 0.28،\ 0.48،\ 0.165\}$
Signal (2)	$\{0.22،\ 0.51،\ 0.19،\ 0.07،\ 0.41،\ 0.225\}$
Signal (3)	$\{0.545،\ 0.385،\ 0.115،\ 0.32،\ 0.35،\ 0.09\}$

Figure (4) shows the SPEC distribution of asynchronous FH signal over a period. In the asynchronous mode, the hopping times of FH signals are different. According to the figure (4), the total hopping time for all signals is equal to $t_C = 2\mu$ seconds (hopping rate $f_h = 500k$Hz), but the signals have time delays relative to each other. To simulate MWC input, for a sampling rate of $f_S = 600$MHz, the number of sampling points is $N_S = t_C * f_S = 1200$. An increase in f_S or the duration of t_C leads to an increase in the value of N_S. Increasing the number of N_S points for each frequency- hopping leads to improved estimation of hopping parameters with less error and better accuracy. However, increasing N_S leads to increased computational complexity and hardware limitations.

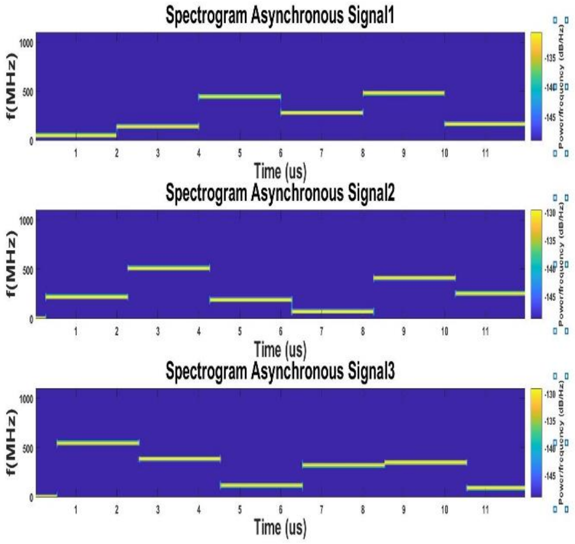


Fig. 4. SPEC, distribution of three asynchronous input signals.

Therefore, for designing a suitable system, a trade-off between variables should be considered. Table (2) shows the numerical values of MWC parameters.

Table 2. MWC parameters.

Parameter	Value	Parameter	Value
Number of branches (M)	**10**	Sampling frequency	**5MHz**

Sequence compression (k)	**30**	Filter bandwidth (B)	**2.5MHz**

According to Table (2), a low-pass filter with a limited bandwidth of 2.5MHz has been used to pass FH signals with a frequency-hopping range of 500MHz.

According to Eq. (10), Figure (5-a) shows the amplitude of $\mathcal{SE}_{(t)}$ in terms of time, to display the hop times of FH input signals. Each of these signal consists of six frequency hops. The duration of each hop for every signal is the same, and equal to $t_C = 2\mu$ seconds. According to Figure (5-a), all hopping times are equivalent to:

$$t_{hop}(\mu S) \quad = \begin{pmatrix} 0 & 2 & 4 & 6 & 8 & 10 \\ 0.17 & 2.16 & 4.16 & 6.17 & 8.17 & 10.16 \\ 0.34 & 2.33 & 4.33 & 6.34 & 8.32 & 10.33 \end{pmatrix}$$

The signals have a time delay relative to each other of approximately $t_\tau = 0.17\mu$ seconds.

Fig. (5-b) shows the results obtained from estimating the hop times of the signals in the MWC output. The estimation error is reduced by repeating the experiment, and increasing the SNR.

As shown in Figure (5-b), in Experiment 400, for over 100 iterations, the hop times of the signals with an error less than 1.068×10^{-7}, sorted from small to large, have been estimated as follows:

$$\hat{t}_{hop}(\mu S) = \{0 \text{ ، } 0.18 \text{ ،}0.35 \text{ ،}2 \text{ ،}2.18 \text{ ،}2.35 \text{ ،}4 \text{ ،}4.18 \text{ ،}4.35$$
$$\text{ ، } 6 \text{ ،}6.18 \text{ ،}6.35 \text{ ،}8 \text{ ،}8.18 \text{ ،}8.35\text{، } 10\text{، } 10.18 \text{ ، } 10.35\}$$

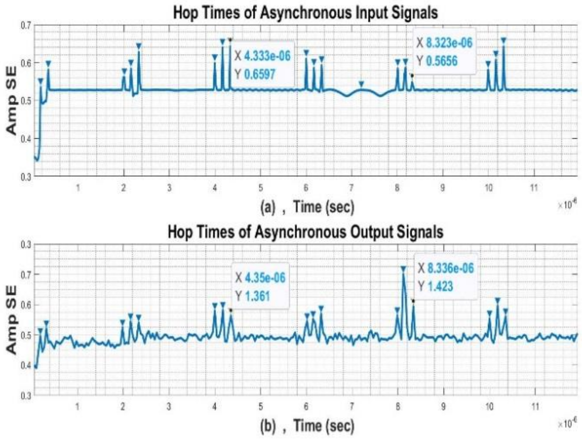


Fig. 5. (a) plotting the amplitude of $\mathcal{SE}_{(t)}$ in terms of time, and showing the hop times of three input asynchronous FH signals **(b)** Estimated hop times at MWC output.

In addition, In a noisy environment, decreasing the SNR leads to an increase in false peaks in the spectrum, and applying the SE method significantly increases the estimation error.

By measuring the difference of both consecutive components of the time vector $\hat{t}_{hop}$, A set of rounded numbers is obtained as follows:

$$t_{hopdif} = \{13.1 \text{ ،}166.4 \text{ ،}17.5 \text{ ،}17.5 \text{ ،}166.4 \text{ ،}17.5 \text{ ،}17.5\text{، } 166.4$$
$$13.1 \text{ ،}17.5 \text{ ،}166.4 \text{ ،}13.1 \text{ ،}21.9 \text{ ،}166.4 \text{ ،}17.5 \text{ ،}17.5\}$$

In this set, the number 17.5 is observed with the highest frequency value, which indicates the time delay between signals. Because, it is the same frequency dwell time for all FH signals in each frequency hop. The higher the frequency value, the higher the accuracy of the SE method. Additionally, increasing the SNR leads to an increase in the frequency value.

In the next step, the position of each pair of consecutive and repeating numbers, 17.5, in the vector t_{hopdif} represents the time delay of three signals at hop times. For example, in the t_{hopdif} vector, the third and fourth components have a common value of 17.5, indicating that in the second hop, the second component of all three signals has been correctly registered. Similarly, the sixth and seventh components of the t_{hopdif} vector have the same value. it means, The SE method has correctly estimated the third time component of all three signals in the third hop. Also, the total value of 166.4, along with two delays of 17.5, indicates the approximate duration of each signal at a constant frequency.

Therefore, using the t_{hopdif} vector, it is possible to obtain the time delay between signals (i.e., $t_{\tau} = 0.175 \times 10^{-6}$) , and the approximate time t_{C} (i.e., $t_{C} = 2 \times 10^{-6}$).

Considering the t_{hopdif} vector, the second, third, and sixth components of all three signals can be estimated using two-time parameters, t_{τ}, and t_{C}. For example, the third component of all three signals is equal to the following:

$\text{Signal\_1}(3) = 10^{-6}\{2 \times t_{C} + 0 \times t_{\tau}\} = 4\mu Sec$
$\text{Signal\_2}(3) = 10^{-6}\{2 \times t_{C} + 1 \times t_{\tau}\} = 4.175\mu Sec$
$\text{Signal\_3}(3) = 10^{-6}\{2 \times t_{C} + 2 \times t_{\tau}\} = 4.35\mu Sec$

Similarly, the sixth component of the signals is:

$\text{Signal\_1}(6) = 10^{-6}\{5 \times t_{C} + 0 \times t_{\tau}\} = 10\mu Sec$
$\text{Signal\_2}(6) = 10^{-6}\{5 \times t_{C} + 1 \times t_{\tau}\} = 10.175\mu Sec$
$\text{Signal\_3}(6) = 10^{-6}\{5 \times t_{C} + 2 \times t_{\tau}\} = 10.35\mu Sec$

Thus, using the estimated time parameters t_{τ} and t_{C}, the set of hop times for each signal can be recovered and separated as follows:
Signal_1(μS): {0 ،2 ،4 ،6 ،8 ،10}
Signal_2(μS): {0 ،2.17 ،4.17 ،6.17 ،8.17 ،10.17}
Signal_3(μS): {0 ،2.35 ،4.35 ،6.35 ،8.35 ،10.35}

As previously mentioned, improving processing variables, such as increasing SNR, increasing Ns, and increasing the repetition of experiments result in reducing the estimation errors of time variables t_{τ} and t_{C} to less than 10^{-8} (and even zero). As a result, the accuracy is increased in separating the hop times of each signal.

4.1. Calculating the RMS/RF error

To evaluate the results of the analysis, the estimated hop time errors have been measured using RMSE and RFE methods. The estimation error of hop times is defined using Eq. (12) and Eq. (13).

$$RMSE_{t} = \sqrt{\frac{1}{NRU'} \sum_{n=1}^{N} \sum_{r=1}^{R} \sum_{u=1}^{U'} \left(t_{nr}^{(u)} - \hat{t}_{nr}^{(u)}\right)^{2}} \qquad (12)$$

In Eq. (12), $\hat{t}_{nr}^{(u)}$ represents the estimated hop time, $t_{nr}^{(u)}$ represents the actual value of the hop time, n indicates the n-th signal, and r for the r-th frequency of the signal hop.

$$RFE_{t} = \frac{1}{U'} \sum_{u=1}^{U'} \frac{|\hat{t}_{h_u} - t_{h}|}{f_S} \qquad (13)$$

In Eq. (13), $\hat{t}_{h_u}$ represents the estimated time, t_{h} represents the actual value of the hop time, f_S for the sampling frequency of the MWC, and U$'$, represents the total number of experiments. It should be noted that for all calculations, the number of experiments is greater than $\text{U}' = 400$. Naturally, increasing U$'$ leads to improvements in the results.

Figure (6), displays the curve of the RMS error value compared to the RF error for estimating hop times of three asynchronous FH signals at different SNR values. The results are obtained by registering the error for $8 \times 20 = 160$ processing iterations.

According to the Figure (6), the RMS error has decreased compared to the RF error and recorded a smaller amount of error, so that in the value of $\text{SNR} = -1dB$, the RMS error is approximately 0.25 of the RF error and recorded a value of 0.54×10^{-6}. In addition, in low and negative SNR values, the RMS error is more reduced. Also, in the value of $\text{SNR} = 3dB$, the RMS error amount has almost reached zero, and in this case, the hopping times are estimated with better accuracy. It should be noted that the amount of recorded error has a direct relationship with the amount of rounding of the estimated hopping times during processing.

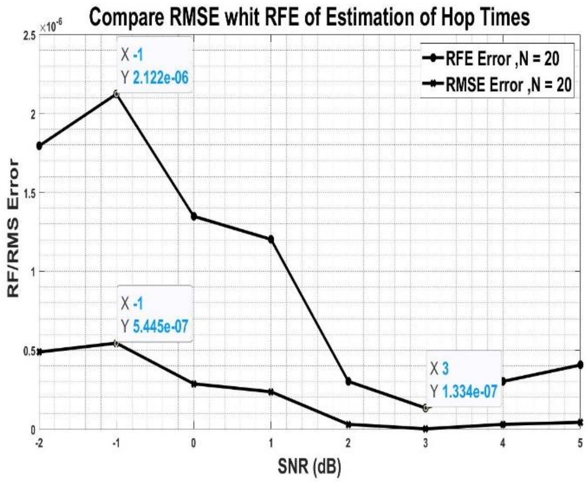


Fig 6. Compares, the RF error value with RMS in estimating the hopping times of three asynchronous FH signals.

Figure (7), shows the curve of the RMS error value compared to the RF error for estimating the hopping times of three asynchronous FH signals. The repetition of processing is $8 \times 200 = 1600$. According to the figure(7), increasing the repetition by 10 times has led to an error reduction of approximately 0.1 while maintaining the previous features compared to figure (6).

In another investigation, the increase in the number of experiments and its impact on the RMS error for three signals with six hop frequencies has been examined. As shown in Figure (8), increasing n results in a reduction in the RMS error of the system. Moreover, at low SNR values, the error slope becomes slow and nearly linear.

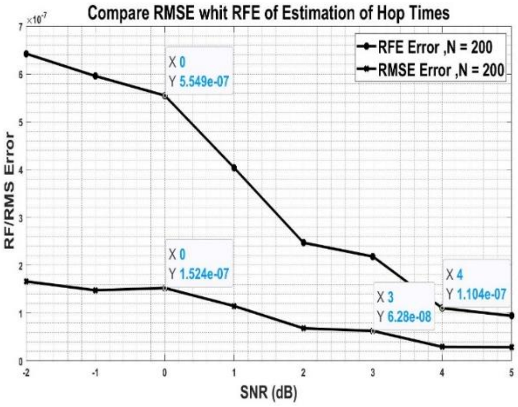


Fig. 7. Compares, the RF error with the RMS error in estimating the hopping times of three asynchronous FH signals with increased processing repetition.

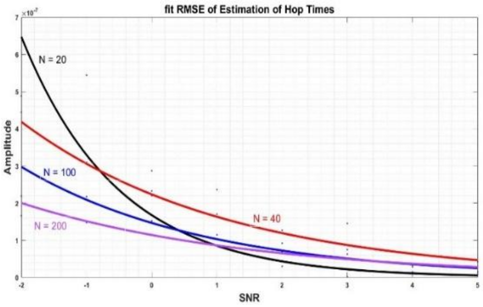


Fig. 8. Shows, the RMS error values in estimating the hopping times of three asynchronous HF signals for different numbers of processing repetitions.

4.2. Comparison with other methods

In this section, to validate the accuracy of the proposed method compared to other methods, the estimation accuracy of parameters has been evaluated in terms of SNR for three asynchronous signals. Figure (9), For $\mathrm{Nh} = 6$, compares the RMS error curve of the MWC_TF_SE method within the range of SNR -4dB to 10dB, simultaneously with two other methods, the STF distribution based on array processing [8], and the parameter estimation method using sparse linear regression (SLR) [46]. The results represent the recorded error rate for a processing repetition of $15 \times 100 = 1500$ times.

According to the figure (9), the proposed method has achieved better results than the $\mathrm{STFD} - \mathrm{based}$ method using ULA over the entire SNR range. Moreover, for $\mathrm{SNR} \leq +1.5\mathrm{dB}$, the RMS error of the proposed method is significantly reduced. So, in $\mathrm{SNR} = -4\mathrm{dB}$, the recorded error is approximately half of the other two techniques.

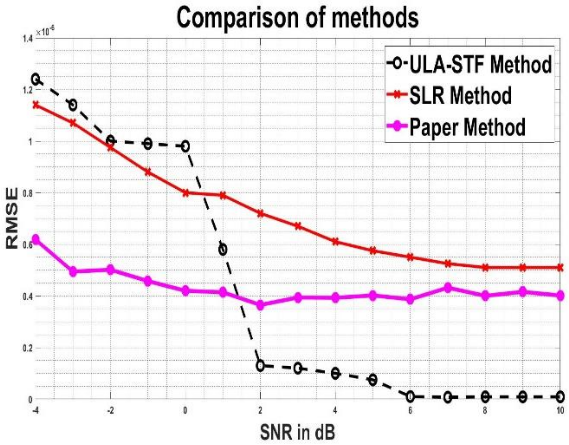


Fig. 9. Compares, the RMS error of different methods in estimating the hopping times of FH signals.

Additionally, in the entire SNR range, the slope of the error curve for the proposed method is slow, indicating its robustness compared to the other two techniques. However, for $\mathrm{SNR} > 7\mathrm{dB}$, the error of the research method converges towards a constant value of 0.4×10^{-6}. This is due to the rounding errors in the parameter extraction process of the signal. Another point to note is that increasing variables, such as the number of sampling points, SNR, and the number of iterations significantly contribute to improving results and reducing errors.

5. CONCLUSION

This paper proposes a novel Semi-BSS and parameters estimation method of FH signals. The proposed method exploits the structure of MWC and SE method to estimating the parameters of FH signals. In the proposed method, initially, three FH signals in the frequency range of $50 \leq f(\mathrm{MHz}) \leq 550$ were received by a single antenna and passed through MWC.

The MWC structure consists of ten branches with a constant sampling rate of $f_s = 5\mathrm{MHz}$, and a receiver with a cut-off frequency of $f_{cut} = 2.5\mathrm{MHz}$. Then, the instantaneous SE method was applied to the TF distribution of the output data from MWC. The accurate estimation of the time delay variables, t_τ, and the duration of stay, t_C, ensures the precision of the proposed method in signal separation and parameter estimation. Some features of the proposed MWC_TF_SE method for the BSS process are as follows:

- Signal processing capability in the baseband and sub-Nyquist sampling condition by a single antenna,
- Ability to remove interfering arrival of the signals,
- Improved performance of the method with reduced SNR, compared to similar methods for $\mathrm{SNR} \leq +1.5\mathrm{dB}$ values,
- Significant reduction in the volume of data in the input TF analysis, proportional to the MWC sampling rate, and increased program execution speed.

The obtained evaluation results show that this algorithm has an acceptable performance compared to other traditional separation methods, is applicable at lower SNR, and has higher accuracy. According to Figure (9), the error rate decreases significantly compared to other methods at $\mathrm{SNR} \leq +1.5\mathrm{dB}$. Specifically, around $\mathrm{SNR} = -4\mathrm{dB}$, the RMS error reaches an approximate value of 0.6×10^{-6}, representing a 50% reduction compared to other methods in parameter estimation.

REFERENCES

[1] Yun-Bin، Y. and T. Qing-Min ، “**DOA Estimation Methods of FH Signals and Follower Gamming Signals**”، J. Nav. Univ. Eng 28، pp. 228-234، 2016.

[2] R. L. Peterson، D. E. Borth، and R. E. Ziemer، “**An Introduction to Spread-Spectrum Communications**”، Prentice-Hall، Inc، 1995.

[3] S. Barbarossa and A. Scaglione، "**Parameter Estimation of Spread Spectrum Frequency-Hopping Signals Using Time-Frequency Distributions**، “ in Signal Processing Advances in Wireless Communications، First IEEE Signal Processing Workshop on، Paris، France، pp. 213-216، 1997.

[4] H. Alizadeh, M. Rezaee and M. Babaei, “**Amplitude Symmetry Detection for Interfering Signals in CNC Satellite Communications**”، Majlesi Journal of Telecommunication Devices (MJTD), Volume 12, Issue3, Pages 149-159, September 2023.

[5] Y. Yuan، Z. Huang، and X. Wang، ``**Detection of frequency-hopping radio frequency-switch transients**،'' **Electron. Lett.**، vol. 50، pp. 956_957، Jun. 2014.

[6] J. Viterbi، “**Spread spectrum communications: myths and realities**،” IEEE Communications Magazine، vol. 40، no. 5، pp.34–41، 2002.

[7] X. Liu، N. D. Sidiropoulos، and A. Swami، “**Joint hop timing and frequency estimation for collision resolution in FH networks**،” IEEE Transactions onWireless Communications، vol. 4، no. 6، pp. 3063–3073، 2005.

[8] J. Wan، D. Zhang، W. Xu، and Q. Guo، “ **Parameter Estimation of Multi-Frequency Hopping Signals Based on Space-Time-Frequency Distribution**”، Symmetry، Vol. 11، No.5، pp.648، 2019.

[9] L. Wan، X. Kong، and F. Xia، "**Joint Range-Doppler-Angle Estimation for Intelligent Tracking of Moving Aerial Targets**،" IEEE Internet of Things Journal، vol. 5، pp. 1625-1636، 2018.

[10] S. Tomar and P. Sumathi، "**Amplitude and Frequency Estimation of Exponentially Decaying Sinusoids**،" IEEE Transactions on Instrumentation and Measurement، vol.67، pp.229-237، 2018.

[11] L. Wan، G. Han، L. Shu، S. Chan، and T. Zhu، "**The Application of DOA Estimation Approach in Patient Tracking Systems with High Patient Density**،" IEEE Transactions on Industrial Informatics، vol. 12، pp. 2353-2364، 2016.

[12] D. Angelosante، G. B. Giannakis، and N. D. Sidiropoulos، "**Estimating Multiple Frequency-Hopping Signal Parameters via Sparse Linear Regression**،" IEEE Transactions on Signal Processing،vol.58، pp. 5044-5056، Oct 2010.

[13] Z. C. Sha، Z. T. Huang، Y. Y. Zhou، and F. H. Wang، "**Frequency-hopping signals sorting based on underdetermined blind source separation**،" Iet Communications، vol. 7، pp. 1456-1464، 2013.

[14] L. Wan، G. Han، L. Shu، S. Chan، and N. Feng، "**PD Source Diagnosis and Localization in Industrial High-Voltage Insulation System via Multimodal Joint Sparse Representation**،" IEEE Transactions on Industrial Electronics، vol. 63، pp. 2506-2516، 2016.

[15] L. Wan، G. Han، L. Shu، and N. Feng، “**The Critical Patients Localization Algorithm Using Sparse Representation for Mixed Signals in Emergency Healthcare System**”، IEEE Systems Journal، vol. 12، pp. 52-63، 2018

[16] L. Wan، G. Han، L. Shu، N. Feng، C. Zhu، and J. Lloret، "**Distributed Parameter Estimation for Mobile Wireless Sensor Network Based on Cloud Computing in Battlefield Surveillance System**”، IEEE Access، vol. 3، pp. 1729-1739، 2015.

[17] M. Mishali، and Y. C. Eldar، “**From Theory to Practice: Sub-Nyquist Sampling of Sparse Wideband Analog Signals**”، IEEE Journal of Selected Topics in Signal Processing، Vol. 4، No. 2، pp. 375-391، 2010.

[18] E. Baransky، G. Itzhak، N. Wagner، I. Shmuel، E. Shoshan، and Y. Eldar، “ **Sub-Nyquist Radar Prototype: Hardware and Algorithm**”، IEEE Transactions on Aerospace and Electronic Systems، Vol. 50، NO. 2، pp. 809-822، 2014.

[19] S. S. Ioushua، O. Yair، D. Cohen، and Y. C. Eldar، “**CaSCADE: Compressed Carrier and DOA Estimation**”، IEEE Transactions on Signal Processing، Vol. 65، No.10، pp. 2645-2658، 2017.

[20] Z. Lei، Y. Peng، Z. Linhua، X. Hui، and D. Hong، “**Frequency Hopping Signals Tracking and Sorting Based on Dynamic Programming Modulated Wideband Converters**”، Applied Sciences، Vol. 9، No. 14، pp. 2906، 2019.

[21] Z. Lei، P. Yang، and L. Zheng، “**Detection and Frequency Estimation of Frequency Hopping Spread Spectrum Signals Based on Channelized Modulated Wideb and Converters**”، Electronics، Vol. 7، No.9، pp. 170، 2018.

[22] B. Boashash، “**Time-Frequency Signal Analysis and Processing: A Comprehensive Review**”، 2nd ed.: Elsevier Science، eBook، 2015.

[23] Y. Lei and Y. Wu، “**A New Hop Rate Estimation Method for High-Speed Frequency-Hopping Signals**”، in 2008 11th IEEE Singapore International Conference on Communication Systems، Guangzhou، China، pp.1330-1333، 2008.

[24] T. C. Chen، "**Joint Signal Parameter Estimation of Frequency-Hopping Communications**”، Iet Communications، vol. 6، pp. 381-389، Mar 2012.

[25] Y. Wang، Y. Lin، & X. Chi، “**A Parameter Estimation Method of Frequency Hopping Signal Based on Sparse Time-Frequency Method**”، In 2018 IEEE 23rd International Conference on Digital Signal Processing (DSP)، pp. 1-5، November 2018.

[26] R.G. Stockwell، L. Mansinha، R.P. Lowe، “**Localization of the complex spectrum: the S transform**”، IEEE Trans. Signal Process، vol. 44، no. 4، pp. 998-1001، 1996.

[27] W. Yang، M. Li، & L. Wang، H. Zhang، “**Parameter Estimation of Frequency Hopping Signals Based on Time Frequency Analysis**”، In Proceedings of the 26th conference of spacecraft TT\&C technology in China، pp. 131–140، 2013.

[28] Q. Zhang، Y. Liu، X. Zhang، “**Parameter Estimation of Non_modulated or Modulated Frequency-Hopping Signals**”، IEEE International Conference on Signal Processing، Communications and Computing. IEEE، pp. 1-4، 2016.

[29] L. Deng، T. Zhang، J. Jin، "**Application of Rearrangement Spectrogram in Parameter Estimation of Frequency-Hopping Signal**"، Computer Engineering and Design، Vol. 34، No. 10، pp. 3422–3426، 2013.

[30] T. Jo Lynn، "**Adaptive Optimal Kernel Smooth-Windowed Wigner-Ville Distribution for Digital Communication Signal**، EURASIP Journal on Advances in Signal Processing، 2009.

[31] Liang، Z.-J.; Lv، M، " **A joint rapid parameter estimate method of frequency-hopping signals**"، In Proceedings of the 2012 International Conference on Control Engineering and Communication Technology، Shenyang، Liaoning، China، 7–9 December، pp. 952–954، 2012.

[32] Zhang، H.-X.; Chen، C.-F.; Wang، H.-Q، "**A parameter estimation method for FH signal based on SPWVD**"، J. China Univ. Posts Telecommun، 18، pp. 133–136، 2011.

[33] Y. Yang، X. Sun، and Z. Zhong، "**A Parameter Estimation Algorithm for Frequency-Hopping Signals with a Stable Noise**"، C IEEE 3rd Advanced Information Technology، Electronic and Automation Control Conference (IAEAC)، pp. 1898-1904، 2018.

[34] W. H. Fu، Y. Q. Hei، and X. H. Li، "**UBSS and blind parameters estimation algorithms for synchronous orthogonal FH signals**"، Journal of Systems Engineering and Electronics، vol. 25، no. 6، pp. 911–920، 2014.

[35] W. H. Fu، W. Lu، K. Jia، et al، "**Blind Parameter Estimation Algorithm for Frequency Hopping Signals Based on STFT and SPWVD**"، Journal of Huazhong University of Science and Technology، vol. 42، no. 9، pp. 59–63، 2014.

[36] Y. Li، X. Guo، F. Yu، and Q. Sun، "**A New Parameter Estimation Method for Frequency Hopping Signals**"، IEEE USNC-URSI Radio Science Meeting، (Joint with AP-S Symposium)، pp. 51-52، 2018.

[37] M. Khazaee، and S. Akhlaghi، "**Direction-of-Arrival and Hop Tracking of A Special User in A FH Multi-User Based Network Using Uniformly Linear Arrays**"، Journal of Ra'ad،Vol. 5، No. 12، pp. 10-26، 2016. (In Persian)

[38] J. P. Changeux، and S. J. Edelstein، " **Allosteric mechanisms of signal transduction**"، Science، Vol. 308، No. 5727، pp. 1424-1428، 2005.

[39] M. Mishali، Y. C. Eldar، D. Oleg، and Sh. Eli، "**Xampling: Analog to digital at Sub-Nyquist Rates**"، IET Circuits، Devices & Systems Vol. 5، No. 1، pp. 8-20، 2011.

[40] Yang، W.-G.، Li، M.، & Wang، L.-B.، et al ، "**Parameter estimation of frequency hopping signals based on time frequency analysis**"، In Proceedings of the 26th conference of spacecraft TT\&C technology in China، pp. 131–140، 2013

[41] Guido، R.C.; Addison، P.S.; Walker، J، "**Introducing Wavelets and Time-Frequency Analysis Wavelet-Related Technologies in Biomedical Signal Processing**"، IEEE Eng. Med. Biol، vol. 28، no. 13، 2009.

[42] Feng، T.; Yuan، C.-W، "**Blind parameter estimation of frequency-hopping signals based on the time–frequency distribution maxima**"، Acta Electron، vol. 39، pp. 2921–2925، 2012

[43] Pan، Y. N.، J. Chen، and X. L. Li، "**Spectral Entropy: A Complementary Index for Rolling Element Bearing Performance Degradation Assessment**"، Proceedings of the Institution of Mechanical Engineers، Part C: Journal of Mechanical Engineering Science. Vol. 223، pp. 1223–1231، 2009.

[44] Sharma، V.، and A. Parey، "**A Review of Gear Fault Diagnosis Using Various Condition Indicators**"، Procedia Engineering. Vol. 144، pp. 253–263، 2016.

[45] Vakkuri، A.، A. Yli-Hankala، P. Talja، S. Mustola، H. Tolvanen-Laakso، T. Sampson، and H. Viertiö-Oja، "**Time-Frequency Balanced Spectral Entropy as a Measure of Anesthetic Drug Effect in Central Nervous System during Sevoflurane، Propofol، and Thiopental Anesthesia**"، Acta Anaesthesiologica Scandinavica. Vol. 48، pp. 145–153، 2004.

[46] L. Zhao، L. Wang، G. Bi، L. Zhang، and H. Zhang، "**Robust Frequency-Hopping Spectrum Estimation Based on Sparse Bayesian Method**"، IEEE Transactions on Wireless Communications، Vol. 14، No. 2، pp. 781-793، 2014.

[47] Civera, Marco, and Cecilia Surace، **"Instantaneous spectral entropy: An application for the online monitoring of multi-storey frame structures"**، Buildings, Vol.12, No. 3، pp, 310, 2022.

Appendix (A)

For the analytical signal $y[n]$, the discrete-time instantaneous autocorrelation function (IAF) is expressed by [22]:

$$\boldsymbol{K_y}[n،m] = y[n+m]y^*[n-m] \qquad (a\text{-}1)$$

where $\boldsymbol{y} = \{y[0]، y[1]، \dots ،y[\mathrm{L}_y - 1]\}$, L_y is the length of $\boldsymbol{y}$, $[n،m] \in \mathbb{Z}$, and y^* is the conjugate of y. Also, the smoothed IAF (SIAF) of signal $\boldsymbol{y}$ is equal to

$$\boldsymbol{R_y}[n،m] = \boldsymbol{G}[n،m] \underset{n}{*} \boldsymbol{K_y}[n،m] \qquad (a\text{-}2)$$

where $\boldsymbol{G}[n،m]$ is the time-lag kernel, and $\underset{n}{*}$ represents the sign of discrete time-convolution. In this case, the discrete time TF distribution of the signal $\boldsymbol{y}$ is expressed by

$$\boldsymbol{\rho_y}[n،k] = \underset{m \to k}{\mathcal{DFT}}\{\boldsymbol{R_y}[n،m]\} \qquad (a\text{-}3)$$

where $[n،k] \in \mathbb{Z}$, and $\mathcal{DFT}$ represents the discrete-time Fourier transform of variable m in matrix $\boldsymbol{R_y}[n،m]$.

Substituting Eq. (a-2) into Eq. (a-3), writing out the transform, and substituting from Eq. (a-1), the distribution time frequency $\boldsymbol{\rho}_{\boldsymbol{y}}[n،k]$ is obtained.

$$\boldsymbol{\rho}_{\boldsymbol{y}}[n،k] = \qquad (a\text{-}4)$$

$$\sum_{|m|<\frac{N_{\mathbf{G}}}{2}} \boldsymbol{G}[n،m] \underset{n}{*} (y[n+m]y^{*}[n-m])e^{-j2\pi km/N_{\boldsymbol{G}}}$$

which can be expanded as

$$\boldsymbol{\rho}_{\boldsymbol{y}}[n،k] = \qquad (a\text{-}5)$$

$$\sum_{|m|<\frac{N_{\mathbf{G}}}{2}} \sum_{|p|<\frac{P}{2}} \boldsymbol{G}[n-p،m]y[p+m]y^{*}[p-m]e^{-j2\pi km/N_{\mathrm{G}}}$$

In Eq. (a-4) and Eq. (a-5), P and N_{G} are a positive integer, $\boldsymbol{G}[n،m]$ is the time-lag, and $\boldsymbol{G} \in \mathbb{C}^{P\times N_{\mathrm{G}}}$ [22]. For example, the time-lag kernel of some common distributions is equal to

- WV distribution

  $\boldsymbol{G}[n،m] = \delta[n]$

- SPEC distribution

  $\boldsymbol{G}[n،m] = w[n+m]w[n-m]$

WV and SPEC distributions are bilinear transformations. By placing the time-lag kernel $\boldsymbol{G}[n،m]$ of the SPEC distribution in the equation (a-5), the frequency-time distribution $\boldsymbol{\rho}_{\boldsymbol{y}}[n،k]$ of the signal y[n] can be calculated.

The squared magnitude of the STFT, is called the spectrogram and is expressed as follows [22]:

$$\boldsymbol{\rho}_{\boldsymbol{y}-SPEC}[n،k] = \left|\boldsymbol{\rho}_{\boldsymbol{y}-STFT}[n،k]\right|^{2} = \left|\sum\nolimits_{|m|<\frac{M}{2}} y[m+n]w[m]e^{\frac{-j2\pi km}{M}}\right| \qquad (a\text{-}6)$$
$$⬚^{2}$$

where w[m] is the window, $\boldsymbol{\omega} = \{\omega[1]، \omega[2]، \ldots ،\omega[M]\}$ $\boldsymbol{\omega}$ is of length M. According to the Eq. (a-6), the TF distribution of the discrete STFT is equal to

$$\boldsymbol{\rho}_{\boldsymbol{y}-STFT}[n،k] = \sum_{|m|<\frac{M}{2}} y[m+n]w[m]e^{\frac{-j2\pi km}{M}} \qquad (a-7)$$

Existence Of Excitatory and Inhibitory Oscillators in The Small World Network and Its Dynamic Effect on Network Synchronization

Tayebeh Nikfard[1], Ravindra Kumar[2]
1- Department of Physics, Mobarakeh Branch, Islamic Azad University, Mobarakeh, Iran.
Email: tayebehnikfard@gmail.com (Corresponding author)
2- Department of Physics, Radha Govind University, Ramgarh 829122, India
Email: ravindk@gmail.com

ABSTRACT:
Synchronization was investigated in Watts-Strogats small world network with inhibited and excitable oscillators. According to the Kuramoto model in the small world network, with the increase in the limited number of inhibited oscillators, the synchronization in the system will be accompanied by network defects, and with their increase, the synchronization will also increase, and after reaching its maximum value, it will begin to decrease. That is, with a certain ratio of inhibitory oscillators to excitation depending on the coupling strength, network synchronization is maximum. As the coupling strength of the oscillators increases, the interval of the number of inhibitions for which the network is in synchronization decreases. This result is not related to a specific small world network and has been observed by repeating it in different small world networks. Excitatory and inhibitory oscillators are in phase up to a certain percentage of inhibitory oscillators in the network (depending on the coupling strength).

KEYWORDS: Kuramoto, Synchronization, Small World Network, Inhibitory Oscillator, Excitatory Oscillator.

1. INTRODUCTION

Since 1950, network science has become a living and interdisciplinary field. Today, networks play an important role for research in various fields, including social sciences, economics and psychology, biology, physics and mathematics [1,2] and as a forward-looking concept, it is used to describe the interactions of many systems. Several network models have been developed that have statistical properties consistent with real-world networks. In particular, we can mention random networks or René camp, small world network in network science.

Real-world networks such as brain networks, electrical networks, etc. [3] are characterized by a high clustering coefficient. Also, despite the large size, there is often a relatively short path between both nodes. Strugats presented a model with small-world network [3] that exhibits both features, small shortest path length and high clustering factor. These features are known as the small world feature, which consists of a regular network and is rewired with the probability p of edges, which is from 0.005 to 0.05 and is between the regular network ($p = 0$) and the random network ($p = 1$).

One of the main topics of network dynamics is synchronization [4,5]. Synchronization can be seen in many different contexts. In computer science, such as synchronization has been used to extract data in a large database [6]. Other applications in engineering where synchronization or asynchrony are important, such as wireless communication networks [7] and electric networks [8].

By simulating his model, Winfrey [9] found that spontaneous synchronization appears as a threshold process, a phenomenon similar to phase transition, and in his studies and Kuramoto [10], it is stated that the start of synchronization

Paper type: Research paper
https://doi.org/
Received: 19 February 2024; revised: 11 March 2024; accepted: 2 May 2024; published: 1 June 2024
How to cite this paper: T. Nikfard, R. Kumar, **"Existence Of Excitatory and Inhibitory Oscillators in The Small World Network and Its Dynamic Effect on Network Synchronization"**, *Majlesi Journal of Telecommunication Devices*, Vol. 13, No. 2, pp. 113-116, 2024.

in Oscillating populations represent phase transitions; Below the transition point, the individual movement of oscillators in a group is uncorrelated. As their interactions become stronger, the connections between the dynamic modes of the oscillators in one part of the set are established and the frequencies of these oscillators become the same. Near the transition point, the size of the coherent oscillator group is small, but the group grows and the number of interacting oscillators increases. The size of this group can be chosen as a synchronization parameter.

Based on Winfrey's method, The Kuramoto model consists of a population of phase oscillators whose interaction is determined by differential equations [11,12] and expresses the rotation of oscillators with heterogeneous natural frequency that are coupled in the form of phase difference sinusoids.

The paper is organized as follows. In Sect. II, we define the model and the numerical methods of quantifying the synchronization. Sect. III represents the results and discussion and sect. IV is devoted to the concluding remarks.

2. MODEL AND METHOD

We used the Kuramoto model in a network with N oscillators at the nodes of the network, which include two groups. One group (excitatory oscillators) has positive coupling and tries to be in phase with its neighbor, and another group (inhibitory oscillators) tries to be in the opposite phase (π) with it.

Therefore, in the Kuramoto equations:

$$\frac{d\theta_i^s}{dt} = \omega_0 + \frac{1}{k_i}\sum_{j=1}^{N} a_{ij}\lambda_j^s \sin\left(\theta_j - \theta_i^s\right), \qquad (1)$$
$$i = 1, \dots, N$$

In this equation, θ_i is the phase of the ith oscillator, ω_0 is the intrinsic frequency of the oscillators, which are equal and zero without losing any generality. a_{ij} are the elements of the adjacency matrix, where $a_{ij} = 1$, if oscillator i and j are connected, otherwise $a_{ij} = 0$, and k_i is the degree of node ith. λ_i^s is the coupling constant of the ith oscillator (s indicates excitatory and inhibitory) which λ_j^{excit} is positive if the oscillator is excitatory and λ_j^{inhib} is negative if the oscillator is inhibitory. Assuming $Q > 0$ and λ_j^{excit}, we will have: $\lambda_i^{inhib} = -Q\lambda_i^{excit}$, also $\tau = \lambda_i^f t$.

We determine the degree of synchronization of all oscillators in each time interval by the order parameter:

$$\mathrm{r}(\tau) = \frac{1}{N}\sum_{j=1}^{N} e^{i\theta_j(\tau)} \qquad (2)$$

we define the longtime averaged order parameter in the stationary state as:

$$r_\infty = \lim_{\Delta\tau\to\infty} \frac{1}{\Delta\tau}\int_{\tau_s}^{\tau_s+\Delta\tau} r(\tau)d\tau \qquad (3)$$

in which τ_s is the time of reaching a stationary state.

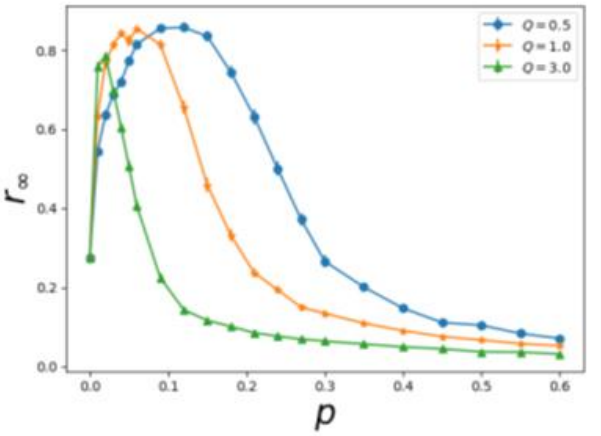


Fig. 1. (Color online) Stationary order parameters verses fraction of inhibitory oscillators for the excitatory-inhibitory model for Q=0.5,Q=1 and Q=1.5 for small world (with the probability of rewiring 0.03) networks of N=1000 oscillators and mean degree ⟨k⟩=10. The error bars indicate the standard error of mean (SEM).

3. RESULTS AND DISCUSSION

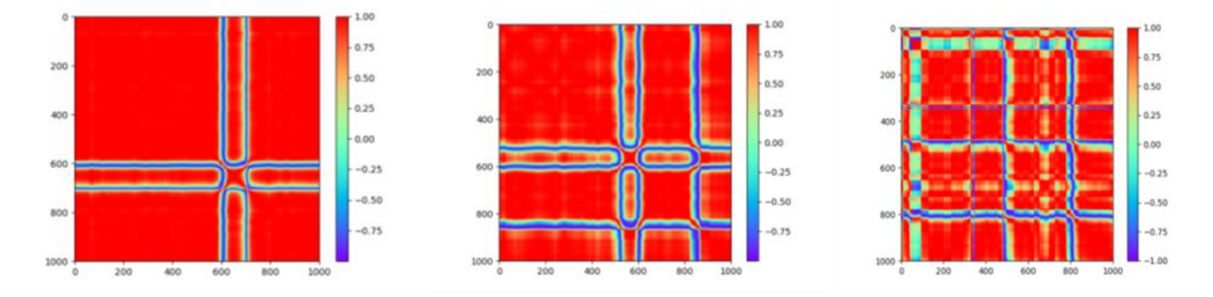
We considered the model for a small world network with 1000 nodes and an average degree of 10. Note that all edges are bidirectionally selected and we denote the ratio of inhibitory oscillators to excitatory oscillators by p. To obtain the time evolution of the oscillators, we used the fourth-order Rangkota method with a time step of 0.1 and considered the initial phase of the oscillators from the box diagram in the interval [π, -π], and the natural frequency distribution of the oscillators follows the Lorentz distribution function.

We obtained the average order parameter for 150 runs for the small world network and different initial conditions and 30 runs for 10 different networks and different initial conditions. $\backsim 8\times10^5$time step calculations have been done and from this number, 1000 final steps have been kept and averaged. In the calculations, it can be seen that the time step of the network is about $\backsim 6x10^5$ and reaches a stable state.

As it is thought, the power of coupling of inhibition and excitation oscillators can be effective in the order parameter and thus network synchronization by increasing the number of inhibition oscillators in the network up to a certain percentage. In Fig. 1, three states are considered for Q: $Q < 1$, $Q = 1$, $Q > 1$. By setting γ=0 in the model, the order parameter is drawn in terms of p. For all three modes, the initial phase of the oscillators is the same. With the increase of inhibitory nodes in the network, the order parameter increases and then decreases, and the higher the coupling strength of the nodes, the faster this decrease and the resistance of the network for synchronization is lower. This turning point depends on the value of Q and decreases with increasing Q.

It can also be seen in Fig. 1 when there is no inhibitory oscillator in the network, the network has not reached full synchronization and what is shown in this figure is the average of 10 initial conditions. By increasing the inhibited oscillators in the small world network in a certain range, not only the order parameter does not decrease, but the order parameter increases up to a certain percentage of the inhibited oscillators.

For a better review, the correlation matrix for $Q = 0.5$ is drawn in Fig. 2. In this Figure, it can be seen that network defects are seen for percentages of the inhibited oscillator which is the maximum order parameter. At first, when the percentage of inhibited oscillators in the network increases from zero, the network defects decrease and in fact the network becomes more regular. Then, with the increase of inhibitory oscillators, network defects increased and for higher percentages (for network with $Q = 0.5$, for $p > 0.18$, for network with $Q = 1$, for $p > 0.1$ and for network with $Q = 3$, for $p > 0.03$) disappears. In fact, around the maximum order parameter, the oscillators are divided into two groups that are in opposite phase (π).

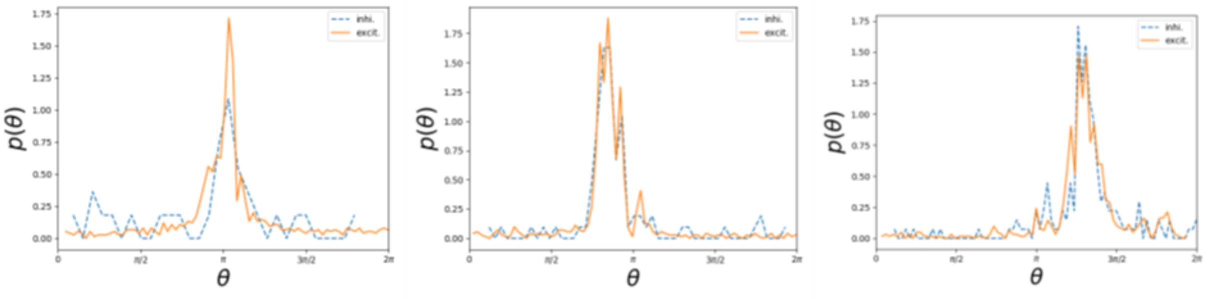


(a) (b) (c)

Fig. 2. (Color online) correlation matrix of inhibitory and excitatory oscillators for (a) p 0.03,(b)p=0.09 and (c)p=0.18 in a small world network of N=1000 oscillators, mean degree ⟨k⟩=10 and Q=0.5. p is the fraction of inhibitory to excitatory oscillators.

With the investigation, we came to the conclusion that the group of oscillators that are in opposite phase with other oscillators and cause network defects, are not only inhibited oscillators and include both inhibited and excited oscillators. The presence of a small percentage of the inhibited oscillator causes a number of oscillators to be in opposite phase with the rest of the oscillators and even for a certain percentage, they create a higher order in the network.

The phase density of the oscillators after the network reached a stable state, separately (inhibitory and excitatory) is drawn in Fig. 3 for Q=0.5 in the small world network for the percentages presented respectively in Fig. 2. As can be seen in these figures, for the percentage of inhibition oscillators that we observed network defects, the phase density diagrams of inhibition and excitation oscillators are in phase.



(a) (b) (c)

Fig. 3. (Color online) The probability density function of the phase of inhibitory and excitatory oscillators for (a) p= 0.03,(b)p=0.09 and (c)p=0.18 in a small world network of N=1000 oscillators, mean degree ⟨k⟩=10 and Q=0.5. p is the fraction of inhibitory to excitatory oscillators.

4. CONCLUSION

In summary, By using the Kuramato model in the small world network and defining inhibitory and excitatory oscillators, we found that the excitatory and inhibitory oscillators are always in phase. We also observed an increase in synchrony by increasing the fraction of inhibitors in the SW network, where the number of inhibitory oscillators to maximize synchrony depends on the coupling strength of the oscillators.

REFERENCES

[1] Solé, Ricard V, Corominas-Murtra, Bernat, Valverde, Sergi, and Steels, Luc. **Language networks: Their structure, function, and evolution.** ***Complexity***, 15(6):20–26, 2010.

[2] Newman, MEJ. **The structure and function of complex networks**, siam re-view, 45, 167–256. *Cerca con Google*, 2003..

[3] Watts, Duncan J and Strogatz, Steven H. **Collective dynamics of 'small-world'networks**. *nature*, 393(6684):440, 1998.Plasma Sci. [Online]. 21(3), pp. 876-880. Available:

[4] Strogatz, Steven H and Stewart, Ian. **Coupled oscillators and biological synchronization**. Scientific American, 269(6):102–109, 1993.

[5] Keane, Andrew, Dahms, Thomas, Lehnert, Judith, Suryanarayana, Sachin Aralasurali, Hövel, Philipp, and Schöll, Eckehard. **Synchronisation in networks of delay-coupled type-i excitable systems.** The European Physical Journal B, 85(12):407, 2012.

[6] Miyano, Takaya and Tsutsui, Takako. **Data synchronization in a network of coupled phase oscillators**. Physical review letters, 98(2):024102, 2007.

[7] Díaz-Guilera, Albert, Gómez-Gardenes, Jesús, Moreno, Yamir, and Nekovee, Maziar. **Synchronization in random geometric graphs**. International Journal of Bifurcation and Chaos, 19(02):687–693, 2009.

[8] Rohden, Martin, Sorge, Andreas, Timme, Marc, and Witthaut, Dirk. **Self-organized synchronization in decentralized power grids**. Physical review letters, 109(6):064101, 2012.

[9] Winfree, Arthur T. **Biological rhythms and the behavior of populations of coupled oscillators**. Journal of theoretical biology, 16(1):15–42, 1967.

[10] Kuramoto, Yoshiki. **Chemical oscillations, waves, and turbulence.** Courier Corporation, 2003.

[11] Kuramoto, Yoshiki. **Self-entrainment of a population of coupled non-linear oscillators.** in International symposium on mathematical problems in theoretical physics, pp. 420–422. Springer, 1975.

[12] Y. kuramoto: **Chemical oscillations, waves, and turbulence, springer-verlag,** berlin and new york, 1984, viii+ 156, 25× 17cm, 9,480 ff (springer series in synergetics, vol. 19). 40(10):817–818, 1985.