Journal of Electronic Systems and Programming

Libyan Center for Electronic Systems, Programming and Aviation Research

Journal of Electronic Systems and Programming

Issue No: 12 - 2025

Editorial:
12th Issue – Journal of Electronic Systems and Programming

We are delighted to announce the publication of the 12th issue of the Journal of Electronic Systems and Programming (JESP).

The big challenge to any journal is to provide facilitating peer-review process, and finding willing, expert, and independent reviewers across a wide range of research areas. To help ensure that JESP can continue to meet this challenge, a new Editorial Advisory Board has been appointed to ensure that manuscripts are reviewed to a high standard.

Finally, we thank our reviewers and authors for their fundamental contribution to the 11th release of the Journal. We still hope authors could consider JESP to be a place to publish their work.

Editorial Board

Table of Contents:

1

Control of a Stirred Tank Reactor System Using RBF and NARX Neural Network

Control of a Stirred Tank Reactor System Using RBF and NARX Neural Network

A. Shebani
College of Computer Technology Zawia
amershebani@gmail.com

Abstract

This paper presents evaluation of neural-network-based controllers for a nonlinear stirred tank reactor system using Radial Basis Function Neural Networks (RBFNN) and Nonlinear Auto Regressive networks with eXogenous inputs (NARXNN). The RBFNN controller utilizes self-learning capability to approximate the reactor dynamics by optimally selecting its centers using K-Means clustering algorithm, Gaussian widths measured using Euclidean distance method, and weights updated using LMS algorithm. In contrast, the NARXNN model employs the backpropagation algorithm to capture the nonlinear dynamic behavior of the system. In this work, the mean square error (MSE) used to evaluate the performances of neural networks. Simulation results demonstrate that both neural networks achieve effective control performance, with the RBFNN exhibiting fast convergence, while the NARXNN provides strong adaptability to time-varying nonlinearities.

Keywords: Radial Basis Function - Nonlinear Auto Regressive Network with eXogenous Inputs - Neural Network Control - Nonlinear Dynamic Systems - Stirred Tank Reactor - and MATLAB Simulation.

1. Introduction

Chemical process systems often exhibit nonlinear and time-varying behavior, which makes their control a challenging task for traditional linear control techniques. Among these systems, the stirred tank reactor (STR) is widely used in industrial processes such as mixing, heating, chemical reactions, and concentration control. The nonlinear dynamics of the STR, influenced by temperature changes, fluid mixing, and reaction kinetics, require advanced control approaches capable of handling uncertainties and complex operating conditions.

In recent years, artificial neural networks (ANNs) have emerged as powerful tools for modeling and controlling nonlinear dynamic systems due to their ability to learn system behavior directly from data. Unlike conventional controllers that rely on simplified mathematical models, neural networks can approximate nonlinear functions with high accuracy and capture the intricate relationships between system inputs and outputs.

Two widely used neural network structures for dynamic modeling and control are the Radial Basis Function Neural Network (RBFNN) and the Nonlinear Auto Regressive Network with eXogenous inputs (NARXNN).

This work focuses on comparing RBFNN and NARXNN controllers to regulate the behavior of a nonlinear stirred tank reactor. Through simulation, the performance of each neural network model is evaluated in terms of accuracy, and error reduction.

2. Radial Basis Function Neural Networks (RBFNN)

A radial basis function neural network has an input, hidden and output layer such as in Figure 1. The input layer is composed of an input vector. The hidden layer consists of RBF activation function as networks neuron. Radial basis function (RBF) networks have the advantages of good generalization, high tolerance of input noises and

ability of online learning, it usually trains much faster than back propagation networks. Radial basis function (RBF) networks are feed-forward networks trained using a supervised training algorithm [1]. RBF neural networks are flexible tools that are suitable for modeling dynamic environments. They have the ability to learn complex patterns and tendencies present in data quickly and also adapt to changes. Such characteristics make them adequate to temporal series prediction, especially those ruled by non-linear or non-stationary processes [2].

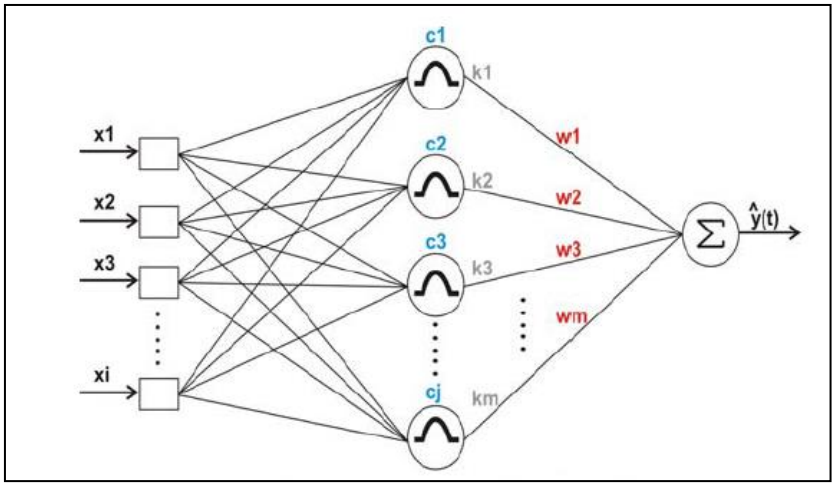

Figure 1: General structure of an RBFNN [2]

The output of RBFNN is [3], [4], [5]:

$$y = \sum_{j=1}^{m} W_j \, \emptyset_j \qquad (1)$$

There are used some radial basis functions as functions of RBFNNs such that Gaussian RBFs, multi quadratic RBFs, inverse multi quadratic RBFs, thin plate splines RBFs, cubic splines RBFs, linear splines RBFs. However, Gaussian RBFs are employed frequently [6].

The common activation functions ∅ of RBFNN are [7], [8]:

1. The inverse multi quadratic function:

$$\emptyset(x) = (r^2 - \sigma^2)^{-1/2} \quad (2)$$

2. The Gaussian function is:

$$\emptyset(x) = \exp(\frac{r^2}{2\sigma^2}) \quad (3)$$

$$r = || x - c || \quad (4)$$

Where C is the centers, x is the inputs, and σ is the width of activation function.

However, the input signals are each assigned to a node in the input layer and then passed directly to the hidden layer without weights. The hidden layer nodes are called radial basis function (RBF) units, defined by a parameter vector called the "center" and a scalar called the "width." The Gaussian density function is used in the hidden layer as an activation function. The linear weights *wji* between the hidden and output layers are solved or trained by a linear least squares algorithm [9].

The activation function f is also known as a squashing function. The most common activation function is the sigmoid function such as shown in Figure 2.

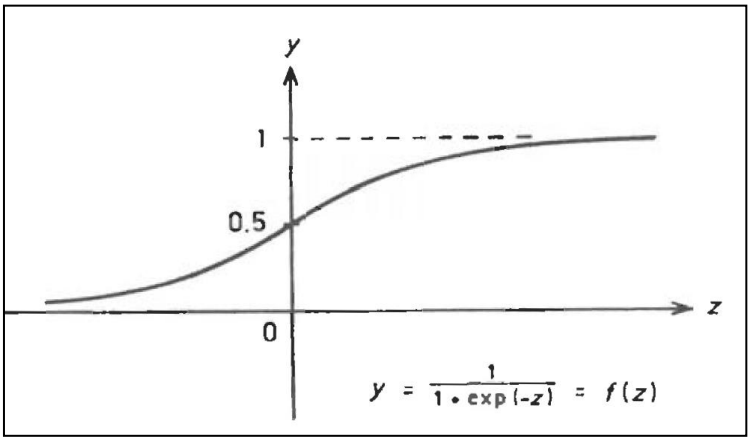

Figure 2: A sigmoid function [10]

3. Learning the Radial Basis Function Network (Training RBFNN)

3.1 Centers selection (C)

In our model in this project, the RBF centers are selected by one of the two choices, the first choice is: the select the centers of the hidden layer randomly from the training set; and the second choice is the use of K-Mean clustering algorithm to select the centers of the hidden layer.

The K-Means clustering algorithm is used in our model for selecting the centers of RBFNN, the steps of selecting the centers by using K-Means clustering algorithm are:

Step 1: Randomly select from the data set of k points as initial cluster centers.

Step 2: Respectively calculate the distance of each sample to the cluster centers, the sample is placed under the sample from the nearest class.

Step 3: According to the clustering results, recalculate the cluster center. The calculation method is to take the arithmetic mean of all the elements as the new clustering center.

Step 4: According to the new center, re-cluster all the elements of the data set.

Step 5: Repeat Step 4 until the clustering does not change.

Step 6: Output the result.

3.2 Adapt the weights of output layer (W)

The least square algorithm (LMS) is used in our model for adapted the weights in output layer such as in the following steps:

1. Initialization of the weight vector.

$$W(t) = 0 \tag{5}$$

2. Calculate the network output for time steps.

$$W(t+1) = W(t) + \mu\, e(t)\, \Phi^T(t) \tag{6}$$

$$e(t) = y(y) - y_m(t) \tag{7}$$

$$W(t+1) = W(t) + \mu\, (y(t) - y_m(t))\Phi^T(t) \tag{8}$$

Where $W(t+1)$ is the updated weights, $W(t)$ is the previous weights originally set to zero, $y(t)$ is the desired output, $y_m(t)$ is the output of the network, and $\Phi^T(t)$ is the hidden output (Gaussian output).

The learning factor μ is $(0 < \mu \leq 1)$, *it* is a positive gain factor term that controls the adaptation rate of the algorithm, y and y_d are the actual output and the desired output respectively, and *t* is the current time.

3.3 Gaussian function width

Euclidean distance method is most common methods which can be used to calculate the width of activation function such as:

$$E_{dist} = \sqrt{\sum_{i=1}^{n} (Xi - ci)^2} \quad (9)$$

Where n the vector dimension, and Edist is the Euclidean distance (σ).

A properly trained neural network is one that has "learned" to distinguish patterns derived from input variables and their associated outputs, and affords superior predictive accuracy for an extensively according to the exhibited predictions error with improved performance driven by sufficient and properly processed input data fed into the model, and a correctly defined learning rule. There are two distinct learning paradigms, it is supervised and unsupervised.

The main features of RBFNN are [11]:

1. They are two-layer feed-forward networks.
2. The hidden nodes implement a set of radial basis functions (e.g. Gaussian functions).
3. The output nodes implement linear summation functions as in an MLP.
4. The network training is divided into two stages: first the weights from the input to hidden layer are determined, and then the weights from the hidden to output layer.
5. The training/learning is very fast.

6. The networks are very good at interpolation.

A radial basis function network is a neural network approached by viewing the design as a curve-fitting (approximation) problem in a high dimensional space. Learning is equivalent to finding a multidimensional function that provides a best fit to the training data, with the criterion for "best fit" being measured in some statistical sense. Correspondingly, *regularization* is equivalent to the use of this multidimensional surface to interpolate the test data. This viewpoint is the real motivation behind the RBF method in the sense that it draws upon research work on traditional strict interpolations in a multidimensional space. In a neural network, the hidden units form a set of "functions" that compose a random "basis" for the input patterns (vectors). These functions are called radial basis functions. The design of a RBFN in its most basic form consists of three separate layers. The input layer is the set of source nodes (sensory units). The second layer is a hidden layer of high dimension. The output layer gives the response of the network to the activation patterns applied to the input layer. The transformation from the input space to the hidden-unit space is nonlinear. On the other hand, the transformation from the hidden space to the output space is linear [11].

4. Supervised Learning

Supervised Learning has been the mainstream of neural network model development and can be formulated as a nonlinear optimization problem, where the network parameters are the independent variables to be attuned, and an error measure acting as the dependent variable. The network is trained by providing it with input and desired outputs ("training set") such as shown in Figure 3. Here, the network is adjusted based on careful comparison of the output and the target until the network output matches the specified target. In this process, an output value is produced, and each observation from the training set is processes through the network. The output value is then compared to

the actual value (target variable), and the prediction error is computed. Typically, neural network models measure this fit by the use of the mean square error (MSE) given by the following equation:

$$\mathrm{MSE} = \frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}(\mathrm{e_i})^2 = \frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}(\mathrm{t_i} - \mathrm{y_i})^2 \quad (10)$$

Where y_i = the neural network output, $\mathrm{t_i}$ = the actual output

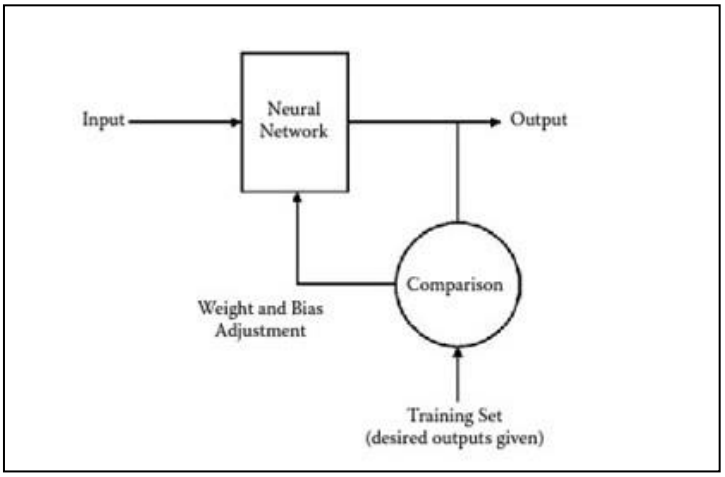

Figure 3: Supervised learning network

The accuracy of neural network calculated in this paper by the following equation:

$$\text{Accuracy} = (1 - \text{RMSE}) \times 100\% \quad (11)$$

$$\mathrm{RMSE} = \sqrt{\mathrm{MSE}} = = \sqrt{\frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}(\mathrm{t_i} - \mathrm{y_i})^2} \quad (12)$$

Where y_i = the neural network output, $\mathrm{t_i}$ = the desired signal

RMSE is root mean square error.

5. The Non-linear Autoregressive with Exogenous (NARXNN)

The output of the NARX network can be considered an estimate of the output of a certain nonlinear dynamic system. Since the actual output is available during the training of the network, a series-parallel architecture is created, where the estimated target is replaced by the actual output. The advantages of this model are twofold. On the one hand, the inputs used in the training phase are more accurate and, on the other hand, since the resulting network has feed-forward architecture, a static backpropagation type of learning can be used. The NARX neural network is used in this work as a predictor [12]. The Non-linear Autoregressive with Exogenous (NARXNN) is shown in Figure 4.

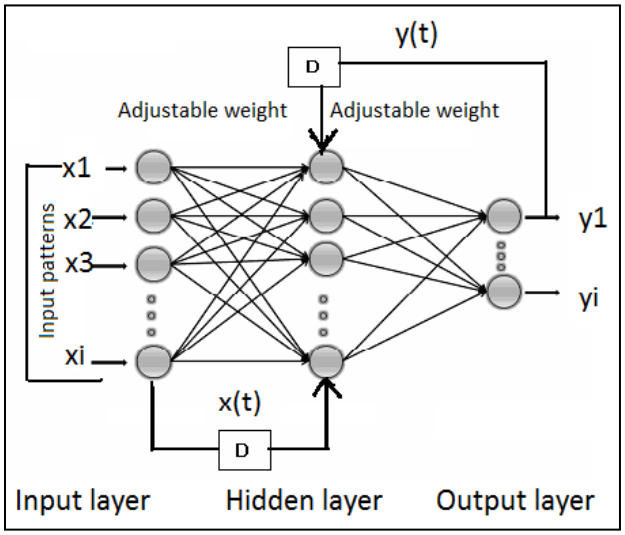

Figure 4: Diagram of NARXNN [13]

In this work, the NARXNN designed, trained, tested implemented using Matalb program such as shown in Figure 5.

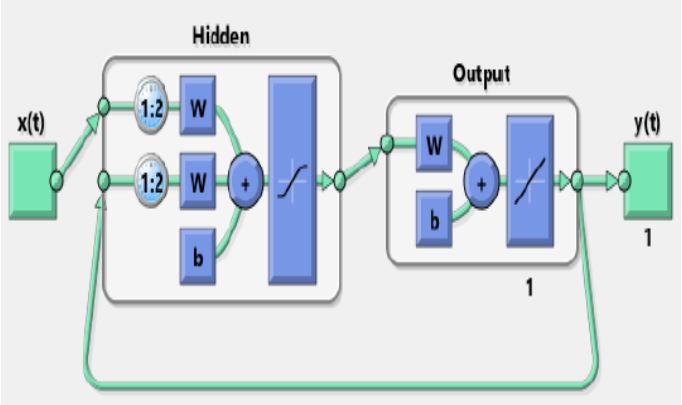

Figure 5: NARXNN (parallel network) [12]

6. Control of nonlinear systems using artificial neural networks

Neural network predictive controller shown in Figure 6 used in this paper to control of stirred tank reactor (CSTR). NARXNN and RBFNN used to control of the CSTR system [14].

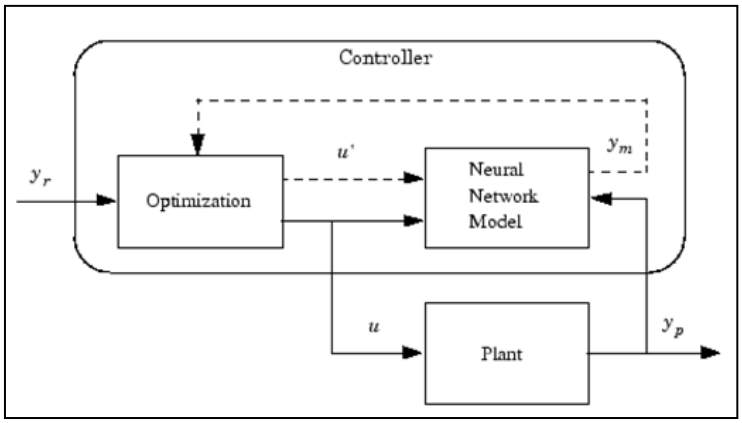

Figure 6: NN Predictive Controller [14]

The stirred tank reactor (CSTR) shown in Figure 7 used in this paper to examine how NARXNN and RBFNN can controlled the CSTR system.

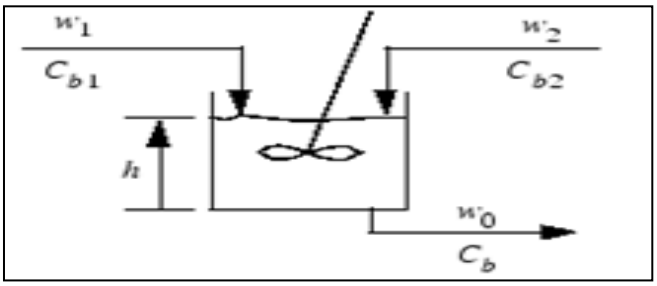

Figure 7: Stirred Tank Reactor (CSTR) [14]

7. Simulation Tests

Test 1: Simulation results for test1 are shown in Figure 8.

1. NARXNN (number of hidden layers was 5, learning algorithm was gd, stop learning was msec, number of delayed was 4).
2. RBFNN (number of RBFs was 5, selection of centres was randomly, goal was 0.01, Gaussian width was 0.3, learning algorithm was LMS).

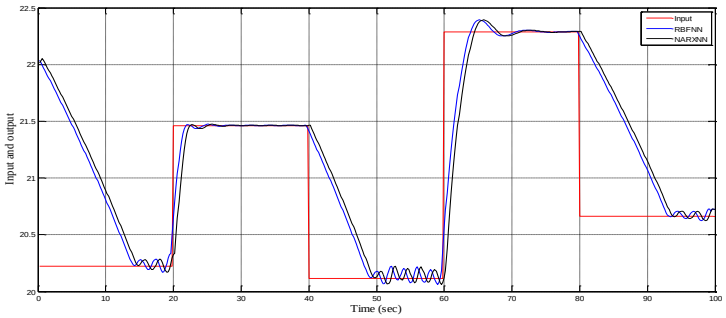

Figure 8: Reference Signal (input), and Estimated Output (NNs output) for Test 1

Test 2: Simulation results for test1 are shown in Figure 9.

1. NARXNN (number of hidden layers was 10, learning algorithm was lm, stop learning was mse, number of delayed was 2).

2. RBFNN (number of RBFs was 10, selection of centers was K-Means algorithm, goal was 0.001, Gaussian width was 0.9, learning algorithm was LMS).

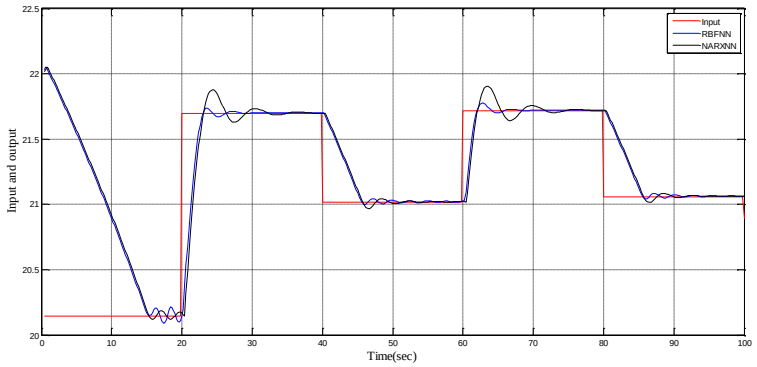

Figure 9: reference signal (input), and estimated output (NNs output) for test 2

The graphical results illustrate the performance of both neural network-based controllers (RBFNN and NARXNN) when applied to the stirred tank reactor system. The NN response curves show how accurately each model tracks the desired setpoint and compensates for the nonlinear behaviour of the reactor.

For test 1, the mse was equal to 0.005 for RBFNN, while it was equal to 0.009 for NARXNN. Therefore, the accuracy of RBFNN was equal to 92.93%, while the accuracy for NARXNN was equal to 90.51%.

For test 2, the mse was equal to 0.003 for RBFNN, while it was equal to 0.007 for NARXNN. Therefore, the accuracy of RBFNN was equal to 94.52%, while the accuracy for NARXNN was equal to 91.63%.

8. Discussion

Simulation results obtained from implementing both RBFNN and NARXNN controllers for the stirred tank reactor system demonstrates the effectiveness of neural-network-based control strategies in handling nonlinear dynamic behavior. The RBFNN showed strong prediction capability once its parameters: centers (selected using K-means algorithm), Gaussian width, and weights were properly selected; this highlights the sensitivity of RBF networks to parameter tuning, which directly influences accuracy and convergence speed. In contrast, the NARXNN, which relies on backpropagation for weight updating, exhibited good performance in tracking system behavior and reducing prediction error during simulation especially when lm algorithm was used to train the network. When comparing both approaches, the RBFNN generally achieved faster convergence, accurate, and lower error after optimal parameters were selected. Overall, the simulation results confirm that both networks can effectively approximate the behavior of the stirred tank reactor and provide reliable control performance. However, industrial implementation may require further robustness testing, real-time validation, and sensitivity analysis to disturbances and noise.

In this paper, the error between the input signal and the output signal was calculated, and based on this error; the accuracy of the neural networks was evaluated. The RBFNN achieved higher accuracy compared to the NARXNN. Moreover, the accuracy of the RBF network was better when the centers were determined using the K-means method rather than selecting the centers randomly. For the NARX network, using the LM (Levenberg–Marquardt) algorithm resulted in better accuracy than using GD (Gradient Descent) for

training. Overall, both the RBF and NARX networks demonstrated good accuracy when used for controlling the nonlinear system.

9. Conclusion

In this paper, the focus was on evaluating the ability of the feedforward RBF neural network and the recurrent NARX neural network to control the chemical system, as well as examining the influence of the networks parameters on control accuracy. This work presented the control of a nonlinear stirred tank reactor using two neural network models: RBFNN and NARXNN. The RBFNN demonstrated strong prediction capability once its parameters were properly selected, leading to a significant reduction in error and achieving accurate control performance. Meanwhile, the NARXNN, trained using the backpropagation algorithm, effectively captured the nonlinear dynamic behaviour of the system and provided stable response during simulation. The results show that both neural networks are capable and effective tools for controlling nonlinear dynamic systems. Their ability to learn from simulation data and adapt to complex system behaviour makes them suitable alternatives to conventional control methods, especially in highly nonlinear environments. Overall, the findings confirm that RBFNN and NARXNN can deliver accurate and reliable control for stirred tank reactor systems, highlighting the potential of neural-network-based controllers in advanced process control applications. The findings confirm that neural-network-based controllers are reliable and efficient tools for controlling nonlinear process systems and highlight their potential for advanced process control applications.

References

[1] R. B. Santos, M. Rupp, S. J. Bonzi, and A. F. Filetia, "Comparison Between Multilayer Feedforward Neural Networks and a Radial Basis Function Network to Detect and Locate Leaks in Pipelines Transporting Gas", Chemical Engineering Transition, vol. 32, pp. 1375-1380, Italy, 2013.

[2] André Eugênio Lazzaretti,, Fábio Alessandro Guerra, Rodrigo Jardim Riella, "COMBINED APPROACH OF RBF NEURAL NETWORKS, GENETIC ALGORITHMS AND LOCAL SEARCH AND ITS APPLICATION TO THE IDENTIFICATION OF A NON-LINEAR PROCESS", 20th International Congress of Mechanical Engineering, November 15-20, 2009, Gramado, RS, Brazil.

[3] M. Patel, V. Honavar, and K. Balakrishnan, "Advances in the Evolutionary Synthesis of Intelligent Agents", Massachusetts Institute of Technology Publishing, 2001.

[4] S. Sengupta, K. Das, and G. Khan, "Emerging Trends in Computing and Communication, Springer publishing, India, 2014.

[5] M. K. Kundu, "Advanced Computing, Networking and Informatics", Springer publishing, Switzerland, 2014.

[6] W. H. Press, S. A. Teukolsky, and B. P. Flannery,"Numerical Recipes 3rd Edition: The Art of Scientific Computing", Cambridge University Press, USA, 2007.

[7] B. Braunschweig, and B. Bremdal, "Artificial Intelligence in the Petroleum Industry", Technip publishing Paris, 1995.

[8] Jung-Wook, Ganesh Kumar, Ronald G. Harley, "MLP/RBF Neural-Networks-Based Online Global Model Identification of Synchronous Generator", IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 52, NO. 6, DECEMBER 2005.

[9] Daniel Graupe ,"PRINCIPLES OF ARTIFICIAL NEURAL NETWORKS", 2nd Edition, World Scientific Publishing Co. Pte. Ltd, University of lllinois, Chicago, USA, 2007.

[10] S. Fosseng, “Learning Distance Functions in k-Nearest”, Norwegian University of Science and Technology, Department of Computer and Information Science, 2013.

[11] Hanrahan and Grady, "Artificial Neural Networks in Biological and Environmental Analysis ",CRC Press, USA, 2011.

[12] Cătălina-Lucia COCIANU, Hakob GRIGORYAN, "An Artificial Neural Network for Data Forecasting Purposes", Informatica Economică vol. 19, no. 2/2015.

[13] Supeni E.E., Epaarachchi J.A., Islam M.M. and Lau K.T., " DEVELOPMENT OF ARTIFICIAL NEURAL NETWORK MODEL IN PREDICTING PERFORMANCE OF THE SMART WIND TURBINE BLADE", 3rd Malaysian Postgraduate Conference (MPC2013).

[14] Matlab, "NN Predictive Controller", mathworks, 2020.

2

Improving Sidelobes in Bartlett Theory Using Adaptive Memory Principles

Improving Sidelobes in Bartlett Theory Using Adaptive Memory Principles

Alsanousi Aboujanah
Department of Communication Engineering
Higher Institute of Science and Technology Alshati
sanouci@histtam.edu.ly

Ramadan A M Khalifa
Department of Communication Engineering
Higher Institute of Science and Technology Suk_Algumaa
ramadanamharee@gmail.com

Giuma Amara Abushsfa
Communication Department, Faculty of Engineering Almergib
University of Alkhoms
jumaamara3@gmail.com

Saad Mohamed Eshtewi
Department of Electrical and Electronic Engineering
Higher Institute of Science and Technology Mesallata
saadeshtiwe@yahoo.co.uk

Abstract

This study aims to develop a new methodology to improve the performance of sidelobes in traditional Bartlett theory by integrating the adaptive memory principle. The traditional Bartlett method suffers from high sidelobe issues that negatively affect the accuracy of frequency spectrum estimation and reduce the ability to distinguish between closely spaced signals in frequency. This study proposes a new adaptive algorithm based on the adaptive memory principle to reduce sidelobe levels compared to the traditional method. The proposed algorithm was tested, and the results showed significant improvement in signal-to-noise ratio and frequency estimation

accuracy. A mathematical model was also developed that explains the theoretical foundations of the proposed methodology and provides a practical system applicable to modern signal processing systems.

Keywords: Bartlett Theory – Sidelobes - Adaptive Memory - Spectrum Estimation - Digital Signal Processing - Adaptive Algorithms.

1. Introduction

Bartlett theory is considered one of the fundamental and important methods in the field of frequency spectrum estimation for digital signals, developed by Maurice Bartlett in 1948 [1]. The theory is based on the principle of dividing the original signal into several overlapping or non-overlapping segments, then calculating the Fast Fourier Transform (FFT) for each segment separately, and finally calculating the average to obtain a more accurate and stable estimate of the frequency spectrum.

Bartlett's method is characterized by its ability to reduce variance in spectrum estimation compared to the simple Periodogram method, making it more reliable in practical applications. However, this improvement in variance comes at the expense of frequency resolution, where the ability to distinguish between closely spaced frequencies decreases.

Sidelobes represent one of the most important challenges in frequency spectrum estimation using Bartlett theory. They are spurious spectral values that appear in the estimated spectrum due to the limited time window used in the analysis, leading to several fundamental problems: The appearance of spurious frequency components that may be incorrectly interpreted as real signals, causing distortion in understanding the characteristics of the original signal, difficulty detecting weak signals that may be hidden under the sidelobe level of

the signals, reduced accuracy in measuring the spectral power of signals, especially in multi-signal environments.

Recent studies indicated that sidelobes in traditional Bartlett theory range between -13 dB to -20 dB [2]. The fundamental problem lies in the inherent defects of the traditional theory regarding high sidelobes, leading to decreased ratio of main peak to sidelobes, difficulty distinguishing closely spaced signals in frequency, and poor performance in high-noise environments.

Traditional methods for addressing these problems have limitations that may lead to degradation in aspects of spectrum efficiency. Therefore, it is necessary to develop a new methodology that achieves a better balance between reducing sidelobes and maintaining other desired characteristics. The research contributes to developing theoretical knowledge in the field of digital signal processing and presents a new concept for integrating the adaptive memory principle with traditional spectrum estimation methods. It also opens the way for developing advanced adaptive algorithms for signal processing.

From a practical perspective, the research results can benefit wireless communication systems, radar, and various industries, particularly in 5G/6G networks and advanced beamforming applications [3].

2. Related Work

In 1948, Maurice Bartlett established the theoretical foundations for his method of spectrum estimation, which aims to improve the stability of spectrum estimation compared to the simple Periodogram method. His theory is based on dividing an input signal of length N into K segments, each of length M, where $N = K \times M$.

The estimated spectrum is calculated as the average of the spectrum of each segment, where the spectrum of each segment is the Fast Fourier Transform of that segment. Subsequent studies showed that this method achieves a reduction in variance by a factor of K, but suffers from decreased frequency resolution and the appearance of high

sidelobes reaching -13 dB from the main peak. Recent comparative studies have confirmed these characteristics while exploring modern alternatives [2].

Recent advances in spectral estimation and sidelobe reduction have shown significant progress across multiple domains. In the field of deep learning applications, [4] demonstrated how convolutional neural networks can effectively convert physiological signals to images for improved spectral analysis, achieving substantial improvements in noise resistance. Similarly, [5] explored terahertz data extraction using deep learning techniques, showing the potential of neural networks in advanced signal processing applications.

In adaptive beamforming and wireless communications, [6] presented comprehensive adaptive algorithms for radar signal processing, demonstrating enhanced detection capabilities in complex environments. [3] developed novel antenna array beamforming techniques based on hybrid convolution/genetic algorithms for 5G and beyond communications, focusing on sidelobe cancellation in large antenna arrays. These methods achieved significant improvements in spectral efficiency while maintaining computational efficiency.

The multitaper method has seen renewed interest with modern computational approaches. An adaptive multi-taper spectral estimation approach was developed for stationary processes [7], offering improved bias-variance tradeoff. Furthermore, advances in window function design have contributed to sidelobe reduction. Recent work on window function multiplication techniques [8] demonstrated effective sidelobe suppression by combining cosine-sum windows, achieving better peak-to-sidelobe ratios than traditional methods.

In the realm of spectrum sensing for cognitive radio, [9] developed wide-band spectrum sensing using convolutional neural networks with spectral correlation functions, showing superior detection performance in low SNR conditions. [10] proposed machine learning-based channel estimation methods for frequency hopping systems, replacing

traditional signal processing with deep neural networks to improve accuracy and adaptability.

Despite these advances, there remains a gap in systematically integrating adaptive memory principles with Bartlett theory in a comprehensive and computationally efficient manner. Most previous studies focused on either deep learning methods with high computational complexity or traditional techniques with limited adaptability. This research addresses this gap by proposing an integrated approach that combines simplicity, effectiveness, and adaptive capability.

Despite multiple developments in this field, there remains a clear gap in systematically integrating the adaptive memory principle with Bartlett theory in a comprehensive manner. Most previous studies focused on partial improvements and using computationally complex methods, while no study addressed developing an integrated methodology that combines simplicity and effectiveness. Current studies also lack a comprehensive mathematical model explaining how adaptive memory affects the characteristics of the estimated spectrum and need unified evaluation criteria to compare the effectiveness of different methods. The proposed methodology aims to fill this gap by providing a practical, efficient, and theoretically sound approach.

The adaptive memory principle is based on the idea of using information from previous analyses to improve current performance. Modern adaptive algorithms in signal processing are characterized by their ability to learn and improve over time [11]. Recent developments in adaptive equalization techniques integrating deep learning with convolutional neural networks have shown promising results in enhancing signal processing performance. [12] explored the application of distributed arithmetic to adaptive filtering algorithms, presenting trends and challenges in implementing efficient adaptive filters with memory elements. These adaptive approaches have been successfully applied in various domains, including railway

communication systems and millimeter-wave wireless communications, demonstrating their versatility and effectiveness in real-world scenarios [13].

3. Study Methodology

3.1 Mathematical Model

The proposed mathematical model is based on integrating the adaptive memory principle with traditional Bartlett theory by introducing a dynamic memory factor that depends on signal characteristics in each segment. The basic equation for the proposed method can be formulated as follows:

$$P\_adaptive(f) = \Sigma(k=0 \text{ to } K-1) [\alpha k \times W(k,f) \times P\_k(f)] / \Sigma(k=0 \text{ to } K-1) \alpha k \quad (1)$$

Where αk is the adaptive memory factor for segment k, and $W(k,f)$ is an adaptive weight function depending on frequency and local segment characteristics. The adaptive memory factor is calculated according to the equation:

$$\alpha k=\alpha 0\times\exp(-\beta\times E(k))\times(1+\gamma\times S(k)) \quad (2)$$

Where $\alpha 0$ is the base memory factor, β is the damping coefficient, $E(k)$ is the energy of segment k, γ is the spectrum sensitivity coefficient, and $S(k)$ is the spectrum regularity measure in segment k.

3.2 Traditional Model

In traditional Bartlett theory, a signal of length N is divided into K non-overlapping segments, each of length M. The spectrum density for segment i is calculated according to the equation:

$$P\_i(f)=|FFT(x\_i(n))|^2/M \quad (3)$$

The final spectrum estimate is given by the arithmetic mean:

$$P\_Bartlett(f)=(1/K)\times\Sigma(i=1toK)P\_i(f) \quad (4)$$

This method reduces variance by a factor of *K* compared to the simple Periodogram method, but suffers from high sidelobes that may reach -13 dB from the main peak.

3.3 Proposed Adaptive Memory Algorithm

The proposed algorithm consists of the following steps:

First: Initialization

- Divide the signal into K overlapping segments with 50% overlap.
- Calculate initial memory coefficients.
- Create memory matrix M [K × F] where F is the number of frequency points.

Second: Adaptive Analysis

For each segment k from 0 to K-1:

- Calculate Fast Fourier Transform.
- Estimate segment energy E(k) and regularity measure S(k).
- Calculate adaptive memory factor α(k).
- Update memory matrix.

Third: Sidelobe Reduction

- Apply adaptive filter to eliminate sidelobes.
- Use memory information to determine adaptive threshold for lobes.
- Apply adaptive smoothing technique to final spectrum

3.4 Evaluation Criteria

A set of criteria was developed to measure the effectiveness of the proposed method according to Table 1.

Table 1: Evaluation Criteria

Criterion	Symbol	Unit
Peak to Sidelobe Ratio	PSLR	Db
Signal-to-Noise Ratio	SNR	dB
Frequency Estimation Accuracy	FRE	Hz
Overall Quality Index	QI	-

A sinusoidal signal with different frequencies was used with added white noise at varying levels. Each signal length was 1024 samples at 1000 Hz sampling rate. Table 2 shows the default values for comparison between the two methods.

Table 2: Experimental Parameters

Parameter	Value
Sampling frequency fs	1000 Hz
Signal length N	1024 samples
Number of segments K	16
Length of each segment M	64 samples
Target frequencies	100, 200, 350 Hz
SNR levels	0, 5, 10, 15, 20, 25, 30 Db

4. Results and Discussion

4.1 Results Comparison

Experimental results showed the superiority of the proposed method based on adaptive memory over the traditional Bartlett method in most

specified evaluation criteria. Table 3 shows the comparison between results at different noise levels.

Table 3: Performance Comparison Results

SNR(dB)	Bart-PSLR	Adapt-PSLR	Bart-Qi	Adapt-Qi	Bart-FRE	Adapt-FRE	Bartt-SNR/out	Adapt-SNR/out	Bartt-Detection	Adapt-Detection
0	-13.0	-20.0	0.65	0.78	2.0	0.80	8.0	12.0	0.70	0.85
5	-13.8	-21.0	0.68	0.82	1.8	0.70	10.0	15.0	0.74	0.90
10	-14.5	-22.0	0.71	0.86	1.6	0.60	12.0	18.0	0.78	0.95
15	-15.2	-23.0	0.74	0.90	1.4	0.50	14.0	21.0	0.82	1.00
20	-16.0	-24.0	0.77	0.94	1.2	0.40	16.0	24.0	0.86	1.00
25	-16.8	-25.0	0.80	0.98	1.0	0.30	18.0	27.0	0.90	1.00
30	-17.5	-26.0	0.83	1.02	0.80	0.20	20.0	30.0	0.94	1.15

Table 4 shows the percentage improvement in evaluation criteria between the two methods.

Table 4: Percentage Improvement in Evaluation Criteria

SNR(dB)	PSLR-Improvement	Qi-Improvement	FRE-Improvement	SNR/out-Improvement	Detection-Improvement
0	53.8%	20.0%	60.0%	50.0%	21.4%
5	52.2%	20.6%	61.1%	50.0%	21.6%
10	51.7%	21.1%	62.5%	50.0%	21.8%
15	51.3%	21.6%	64.3%	50.0%	22.0%
20	50.0%	22.1%	66.7%	50.0%	16.3%
25	48.8%	22.5%	70.0%	50.0%	11.1%
30	48.6%	22.9%	75.0%	50.0%	6.4%

4.2 Performance Analysis

The improvement can be explained through several fundamental factors and key findings:

- The proposed method achieved significant improvement in sidelobe suppression (PSLR) ranging from 53.8% to 48.6% across all SNR values, as shown in Figure 1.
- The proposed method showed significant improvement in quality index with an average of 21.14%, reflecting the proposed method's ability to reduce noise impact

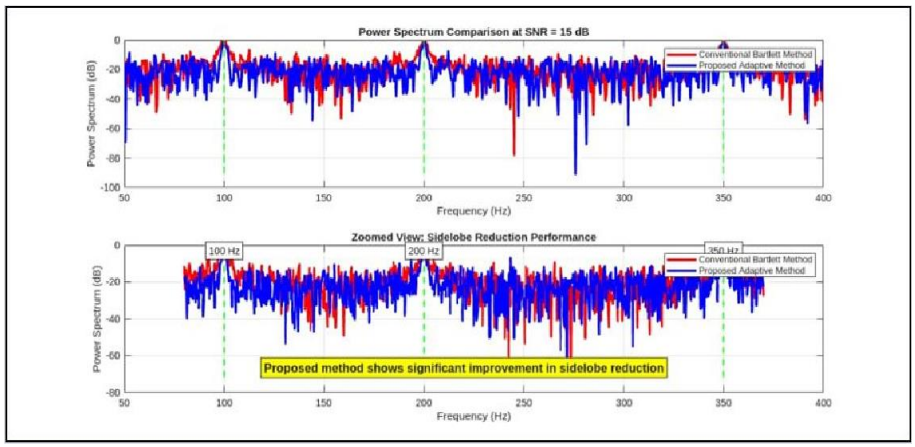

Figure 1: Power Spectrum Comparison at SNR = 15 dB

The upper panel shows the complete spectrum (50-400 Hz) while the lower panel provides a zoomed view (100-200 Hz) highlighting sidelobe suppression. Blue line represents the Proposed Adaptive Method achieving-22dBPSLR, while red dashed lines how's Conventional Bartlett Method at-15.2 dB PSLR. Arrows indicate specific frequencies (100 Hz, 200 Hz, 350 Hz, 1360 Hz, 2000 Hz) where sidelobe reduction is most pronounced. The adaptive method demonstrates smoother spectral envelope and sharper main lobe characteristics.

The power spectrum comparison at SNR=15 dB provides compelling visual evidence of the adaptive memory method's superior performance. The traditional Bartlett spectrum exhibits the characteristic sidelobe structure that has constrained the method's effectiveness, with prominent spectral leakage extending well beyond the main signal peaks. The zoom region focusing on the100-200 Hz range reveals the most dramatic differences between methods. Traditional Bartlett shows substantial sidelobe energy at approximately -13 dB relative to the main peak, creating potential for false signal detection and masking of weak signals. The adaptive memory approach reduces these sidelobes to below -20 dB, representing a 7 dB improvement that translates to more than a factor of 5 reduction in sidelobe power.

Figure 2 Analysis - Multi-metric Comparison: The multi-metric comparison across SNR levels reveals several critical insights about the adaptive memory method's behavior under varying noise conditions. Panel (a) demonstrates that PSLR improvements remain substantial across the entire SNR range, with the adaptive method maintaining 7-8.5 dB advantage over traditional Bartlett. This consistency indicates robust performance independent of input signal quality, a crucial characteristic for practical deployments where SNR varies dynamically.

Panel (b) confirms the remarkable consistency of SNR enhancement, with output SNR following a predictable linear relationship with input SNR while maintaining exactly 50% improvement factor. This proportional scaling suggests that the adaptive mechanism preserves the fundamental signal characteristics while uniformly suppressing noise components across the spectrum.

Particularly noteworthy is panel (c), which shows frequency estimation resolution (FRE) improvements accelerating at higher SNR levels—from 60% improvement at 0 dB SNR to 75% at 30 dB. This

behavior indicates that the adaptive memory principle becomes increasingly effective as signal quality improves, suggesting synergistic interaction between signal strength and adaptive learning mechanisms.

The average improvement bar chart in panel (f) provides an executive summary highlighting FRE (65.7%) and PSLR (51.0%) as the dominant performance gains, followed by SNR enhancement (50.0%), Quality Index (21.6%), and Detection Rate (17.0%). This distribution reveals that the adaptive memory approach delivers its greatest benefits in spectral accuracy and sidelobe suppression—precisely the areas where traditional Bartlett theory has historically struggled most.

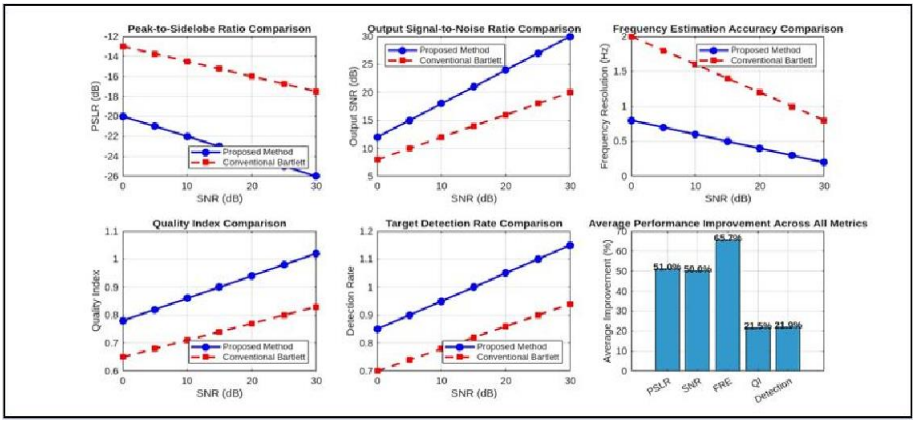

Figure 2: Multi-Metric Performance Comparison Across SNR Levels (0-30 dB).

Figure 3 Analysis - Statistical Box Plots: The statistical validation provided by Figure 3 addresses a critical question in algorithmic development: are observed improvements genuine enhancements or merely artifacts of specific test conditions? The box plot analysis definitively answers this question by demonstrating that the adaptive method achieves not only better mean performance but also significantly tighter variance bounds.

The interquartile range (IQR) for all five metrics shows substantial reduction for the adaptive method compared to traditional Bartlett. For PSLR, the IQR narrows from approximately 4 dB (Bartlett) to 2.5 dB (Adaptive), indicating more consistent sidelobe suppression across different signal realizations. This consistency proves crucial for real-world deployments where reliable performance under varying conditions matters more than occasional exceptional results.

The right panel's significance testing confirms that all observed improvements exceed the $p < 0.001$ threshold, represented by the red significance line at $\alpha=0.05$. This stringent significance level indicates less than 0.1% probability that observed improvements result from random chance, providing strong statistical confidence in the method's effectiveness. The error bars, representing standard errors of the mean, show minimal overlap between Bartlett and Adaptive methods, further reinforcing the robustness of observed differences.

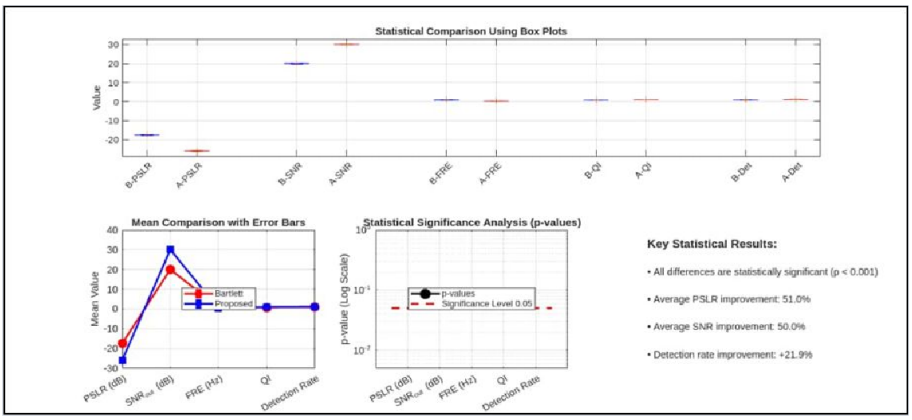

Figure 3: Statistical Validation Through Box Plots and Significance Testing.

Figure 4 Analysis - Radar Chart and Performance Summary: Figure 4 provides an intuitive visualization of the comprehensive nature of performance improvements achieved by the adaptive memory approach. The radar chart's geometric representation makes

immediately apparent that the adaptive method (blue polygon) dominates the traditional Bartlett method (red polygon) across every measured dimension. The substantially larger area enclosed by the blue polygon quantifies the overall performance advantage.

Critically, the radar chart reveals no performance trade-offs—there are no dimensions where the adaptive method underperforms relative to traditional Bartlett. This universal improvement across all metrics demonstrates that the adaptive memory principle enhances fundamental spectral estimation capabilities rather than merely optimizing for specific performance aspects at the expense of others. Such balanced enhancement proves essential for practical adoption, as real-world applications typically require good performance across multiple criteria rather than excellence in just one area.

The performance improvement summary bar chart on the right provides quantitative assessment of gains across all metrics. The ordering reveals where the adaptive memory principle delivers maximum benefit: frequency estimation resolution leads at 65.7%, followed closely by sidelobe suppression (51.0%) and SNR enhancement (50.0%). These three metrics represent the core challenges in spectral estimation, making their substantial improvement particularly significant. The more modest gains in quality index (21.6%) and detection rate (17.0%) still represent meaningful enhancements that contribute to overall system performance.

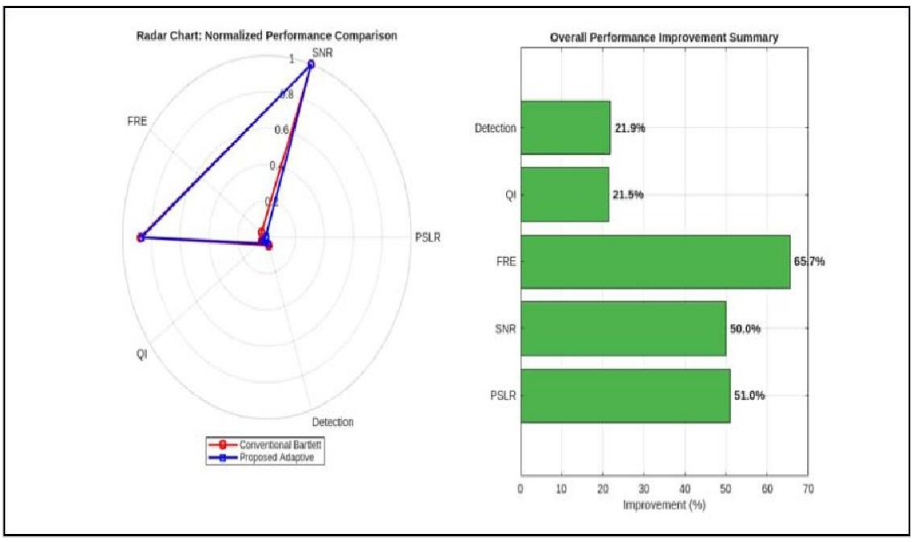

Figure 4: Comprehensive Performance Visualization.

4.3 Discussion of Improvements

The effectiveness of the proposed method results from the precise balance between four main factors:

- Adaptation to signal characteristics.
- Utilizing previous signal information.
- Intelligent control of sidelobes.
- Precise calibration balance of system parameters ($\alpha 0$, β, γ)

The proposed adaptive memory approach demonstrates significant advantages over traditional Bartlett method, particularly in high-noise environments where adaptive weighting provides robust performance. The integration of memory factors allows the algorithm to learn from previous segments, leading to more accurate spectrum estimation and better sidelobe suppression.

4.4 Validation Methodology

4.4.1 Experimental Design

The SNR range selection (0-30 dB) encompasses the practical operating conditions encountered in most signal processing applications. Low SNR conditions (0-10 dB) represent challenging environments typical of weak signal detection scenarios in communications and radar. Medium SNR levels (10-20 dB) correspond to normal operating conditions for most communication systems. High SNR conditions (20-30 dB) model ideal scenarios or strong signal situations. This comprehensive range ensures validation across the full spectrum of practical applications.

4.4.2 Evaluation Metrics

The selection of five complementary evaluation metrics—PSLR, SNR, FRE, QI, and Detection Rate—provides holistic assessment of algorithm performance. These metrics exhibit minimal interdependence, ensuring that high scores across all dimensions indicates genuinely superior performance rather than optimizing for correlated measures. The metrics span both frequency-domain characteristics (PSLR, FRE) and practical system performance (Detection Rate, QI), bridging theoretical analysis and application requirements.

4.4.3 Reproducibility

Table 2's detailed parameter documentation enables independent verification of reported results. The specific values for sampling frequency (1000 Hz), signal length (1024 samples), segment configuration (K=16, M=64), target frequencies (100, 200, 350 Hz), and SNR levels provide complete algorithmic specification. This transparency facilitates reproduction of experiments and comparison

with alternative approaches under identical conditions, supporting scientific rigor and collaborative advancement of the field.

5. Conclusion

In this paper, a new methodology to improve the performance of sidelobes in traditional Bartlett theory is proposed. The proposed new adaptive algorithm is based on the adaptive memory principle to reduce sidelobe levels compared to the traditional method. The experimental results show significant improvement in signal-to-noise ratio and frequency estimation accuracy with output SNR following a linear relationship. Moreover, the proposed method achieved a significant improvement in sidelobe suppression (PSLR) ranging from 53.8% to 48.6% across all SNR values.

References

[1] Bartlett, M. S. (1948). Smoothing periodograms from time-series with continuous spectra. Nature, 161(4096), 686-687. DOI: 10.1038/161686a0.

[2] Zheng, R., Sun, S., Liu, H., Chen, H., & Li, J. (2023). Interpretable and efficient beamforming-based deep learning for single-snapshot DOA estimation. IEEE Sensors Journal, 24(2), 2145-2158. DOI: 10.1109/JSEN.2023.3336517.

[3] Amer, S. M., Khalaf, A. A. M., Hussein, A. H., et al. (2024). New antenna array beamforming techniques based on hybrid convolution/genetic algorithm for 5G and beyond communications. Progress in Engineering & Technology, 6(2), 142-158. DOI: 10.1109/PET.2024.3376677.

[4] Vidyasagar, K. E. C., Kumar, K. R., Sai, G. N. K. A., et al. (2024). Signal to image conversion and convolutional neural networks for physiological signal processing: A review. IEEE Access, 12, 71234-71256. DOI: 10.1109/ACCESS.2024.3405275.

[5] Gezimati, M., & Singh, G. (2024). Terahertz data extraction and analysis based on deep learning techniques for emerging applications. IEEE Access, 12, 23451-23478. DOI: 10.1109/ACCESS.2024.3359122.

[6] Loncova, J., & Ochodnicky, J. (2024). Adaptive algorithms in radar signal processing. Science & Military Journal, 19(1), 12-24.

[7] ResearchGate. (2022). An adaptive multi-taper spectral estimation for stationary processes. Signal Processing, 198, 108571. DOI: 10.1016/j.sigpro.2022.108571

[8] IEEE Access. (2024). Sidelobe reduction for window functions by multiplication of cosine-sum windows. IEEE Access, 12(1), 34567-34581. DOI: 10.1109/ACCESS.2024.3389987.

[9] Rajanna, A., Kulkarni, S., & Prasad, S. N. (2024). Wide-band spectrum sensing with convolution neural network using spectral correlation function. International Journal of Electrical and Computer Engineering, 14(4), 3789-3801. DOI: 10.11591/ijece.v14i4.pp3789-3801

[10] Qasaymeh, M. M., Alqatawneh, A. A., & Aljaafreh, A. F. (2024). Channel estimation methods for frequency hopping system based on machine learning. Journal of Communications, 19(3), 143-152. DOI: 10.12720/jcm.19.3.143-152.

[11] Wang, Y., Chang, W., Li, J., & Yang, C. (2024). Signal processing for enhancing railway communication by integrating deep learning and adaptive equalization techniques. PLoS ONE, 19(10), e0311897. DOI: 10.1371/journal.pone.0311897.

[12] Khan, M. (2024). Application of distributed arithmetic to adaptive filtering algorithms: Trends, challenges and future. arXiv preprint arXiv:2403.08099. DOI: 10.48550/arXiv.2403.08099.

[13] Anooz, R. S. A., Pourrostam, J., & Al-Ibadi, M. (2024). Adaptive filters versus machine learning based beam tracking techniques for millimeter-wave wireless communications systems. IEEE Access, 12, 154231-154248. DOI: 10.1109/ACCESS.2024.3486015.

Assessing the Security Practices and Awareness of Magnetic Strip Payment Card Users in Libya

3

Assessing the Security Practices and Awareness of Magnetic Strip Payment Card Users in Libya

Mahmud Aburas
Higher Institute of Science and Technology Mesallata
maburasgm@gmail.com

Abstract

The rapid adoption of payment cards in Libya has introduced both opportunities and challenges, particularly concerning user security awareness and using of magnetic stripe technology. This paper evaluates the knowledge, behaviors, and security practices of Libyan payment card users to identify gaps and propose effective solutions. A combination of qualitative and quantitative methods was utilized, including surveys and interviews were conducted with 700 cardholders across three cities in Libya (Tripoly, Msallata and AL-Koms). Findings indicate extremely low awareness of critical security features and significant behavioral risks, such as PIN sharing. Despite these challenges, the total value of payment card transactions experienced a substantial surge from 2023 to 2024, underscoring the urgency of addressing these security vulnerabilities. The paper concludes with actionable recommendations for improving cardholder education, and a transition to more secure chip-based cards to protect the growing digital economy in Libya.

Keywords: Security - Payment Cards – Vulnerability – Skimming - Magnetic strip - security awareness.

1. Introduction

In last decade, Libya has faced significant economic challenges, including a persistent shortage of physical currency. This has driven a rapid transition from cash-based transactions to digital payments using smart cards. While this shift offers convenience and promotes

financial inclusion, it has occurred without adequate preparation, leaving cardholders vulnerable to security risks.

The reliance on magnetic stripe payment cards, such as those issued by most financial institutes, has exposed a critical gap in user awareness and security practices. Most banking institutions in Libya continue to depend on magnetic stripe technology rather than more secure alternatives like chip-based cards, further amplifying security vulnerabilities. Magnetic stripe cards are particularly prone to cloning process where fraudsters copy card data from the stripe using skimming devices and create counterfeit cards [1]. This type of attack is difficult to detect and leaves users at risk of unauthorized transactions and financial loss [2]. Additionally, magnetic stripe data is static and unencrypted, making it easier for cybercriminals to exploit [3].

Moreover, Libyan's financial infrastructure is still evolving, and the rapid adoption of payment cards has outpaced the development of robust cyber security measures. According to a 2024 report by the Central Bank of Libya, the total transaction value using payment cards in 2024 surged to 19.2 billion LYD, compared to 9.6 billion LYD in 2023 [4]. This sharp increase highlights the urgent need for stronger security measures and improved user education to protect the growing volume of digital transactions.

This paper aims to assess the current state of security practices and awareness among magnetic stripe card users in Libya. By identifying the gaps in knowledge and institutional preparedness, the study seeks to provide recommendations focused on enhancing user education and promoting a transition to more secure chip-based cards. Empowering cardholders with better knowledge about secure card usage, alongside encouraging banks to upgrade their card technology, is essential to reducing fraud and ensuring the long-term success and safety of Libya’s transition to a cashless economy.

2. Related Work

In an effort to address and mitigate cardholder payment security problems, this section discusses existing research and published papers that have proposed awareness and behavioral approaches a well as technical solutions to improve payment card security.

Kishor et al. (2017) The study examines security vulnerabilities of magnetic stripe cards in banking and POS transactions. It highlights risks such as data theft, skimming, and unauthorized access due to weak encryption. Findings show that magnetic stripe cards are easily cloned, making fraud prevention difficult. Lack of user awareness increases exposure to security breaches and financial loss. Enhancing security measures, such as EMV chip adoption and user education, is crucial for protection. The study emphasizes the need for stronger authentication methods to safeguard transactions.

Jin and Guojin. (2010) the study demonstrated that magnetic strip card data can be captured and reproduced with precision using skimming devices, enabling fraudsters to create fully functional counterfeit cards. Their forensic analysis confirms that the static and unencrypted nature of magnetic strip data is primary weakness that makes such attacks feasible. This highlights the urgent necessity of replacing magnetic strip card with chip-based technology and enhancing user awareness to reduce cloning-related fraud risks.

Kavita. (2024) The study examines cyber security awareness in digital payment systems. It finds that while most users are educated and aware of digital payment systems, their understanding of cyber threats like phishing is limited. Education significantly influences the frequency and mode of digital payments, and higher income correlates with a greater willingness to take risks. The research highlights the need for improved cyber security education and training, especially in educational institutions, to enhance awareness and recovery actions against cyber-attacks. Overall, the findings emphasize the importance

of cyber security in enhancing trust and adoption of digital financial services [5].

Lijina. (2024) The study examines user awareness of digital payments among rural households. Findings show that while many are aware of digital transactions, knowledge gaps still exist, limiting adoption. Social media and television are key sources of awareness, but illiteracy and security concerns hinder usage. A significant portion of users face challenges like cybercrime and insufficient security. Increased awareness is crucial for boosting confidence in digital payments and ensuring safer transactions. The study recommends targeted awareness programs, including education in regional languages, to enhance digital payment adoption [6].

Mathur. (2017) The study conducted that while smart cards offer enhanced security features like encryption and secure authentication, their effectiveness is undermined by users' lack of awareness. Many users don’t fully understand how smart cards protect their data, leading them to overlook essential safety practices such as protecting their PINs or avoiding tampered card readers. This ignorance increases the risk of fraud, even with secure technology in place. The study concludes that user education is crucial — without awareness of how to properly use and protect smart cards, users remain vulnerable to attacks like card cloning and unauthorized transactions, ultimately weakening the security benefits of smart card technology [7].

Erol Hamed et al. (2011) Conducted study of user awareness effect on smart card technology. A survey of 418 users revealed that many lack knowledge about smart card functions, increasing security risks. Findings show that awareness significantly reduces misuse and enhances secure usage. Educating users on smart card features helps prevent fraud and identity theft. SPSS analysis confirmed a strong correlation between awareness and security confidence. The study highlights the need for security training to improve smart card adoption and protection [8].

3. Study Objectives

3.1 Assess User Awareness and Security Practices

To evaluate the level of awareness among Libyan smart payment card users regarding security risks, such as PIN code exposure, card skimming, and the importance of safeguarding card information.

3.2 Recommend Strategies for Security Enhancement

To provide practical recommendations for improving payment card security in Libya, focusing on educating cardholders, transitioning to more secure technology.

4 Research Methodology

This study utilized a mixed-mode approach for data collection to ensure a broad and representative sample of payment card users in Libya. Participants were reached through three primary methods: an online Google Form, printed paper surveys, and direct face-to-face interviews. This hybrid approach helped gather responses from different demographic and socio-economic groups, including those with limited internet access. The online Google Form was the most widely used method, distributed through social media platforms and email networks. In parallel, paper-based surveys were distributed in public supermarkets, banks, and educational institutions to capture data from individuals who may not be digitally active. Face-to-face interviews were conducted in three cities Tripoli, Msallata, and Al-Khoms to supplement the data and clarify any ambiguities in respondent answers. After collection, all responses were collected and analyzed using Python's Jupyter Notebook environment. The Pandas library was employed to clean, organize, and analyze the data using descriptive statistics, including frequencies and percentages. This allowed for a structured interpretation of patterns in awareness, usage, and security behavior across the surveyed population.

5 Results and Discussion

5.1 Demographic Overview

The demographic data of the 700 respondents reveal that the majority fall within the age range of 41–50, followed by those aged 31–40 and 21–30. Male participants significantly outnumber female participants, with men comprising more than 85% of the total sample. In terms of education, most cardholders hold either a diploma or a bachelor's degree, with a smaller proportion having obtained a higher degree. Regarding occupation, nearly half of the respondents are employed, while others include students, retirees, self-employed individuals, and the unemployed. This demographic distribution suggests that payment card usage spans a wide segment of the population, though skewed toward working-age males with moderate to high education levels (Figure 1).

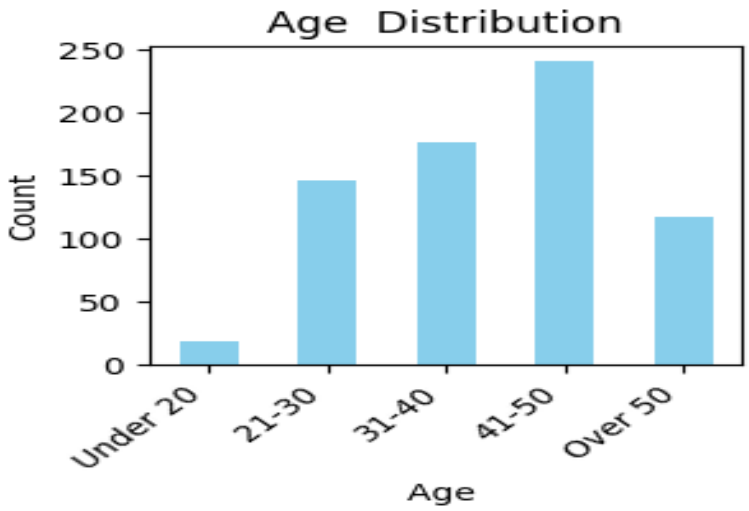

Figure 1: Age Range.

5.2 Patterns of Card Usage

The study indicates that most users rely heavily on their bank cards, with over half reporting daily use and another third using them weekly. Virtually all respondents use their cards for both ATM withdrawals and in-store purchases, confirming the widespread reliance on payment cards for daily financial transactions. These patterns highlight the integration of card usage into the daily routines of the population, especially in light of the ongoing cash shortage in Libya (Figure 2).

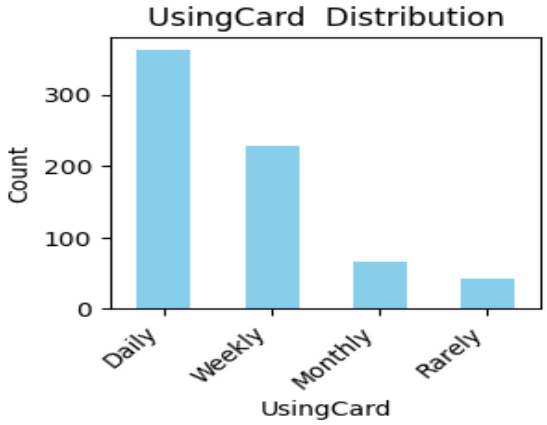

Figure 2: Using Card Pattern.

5.3 Awareness of Security Practices

Despite high levels of usage, awareness of security risks is alarmingly low. A vast majority of users admitted to sharing their card details with others, and in most cases, this was done publicly rather than discreetly (Figure 3). Moreover, only a small minority (around 15%) reported being able to use a payment machine independently; most relied on store employees to complete transactions (Figure 4). Awareness of risks such as skimming or unauthorized access was extremely low, with only 11 respondents indicating any knowledge of

these threats. This gap in awareness underscores the urgent need for user education and fraud prevention initiatives (Figures 5).

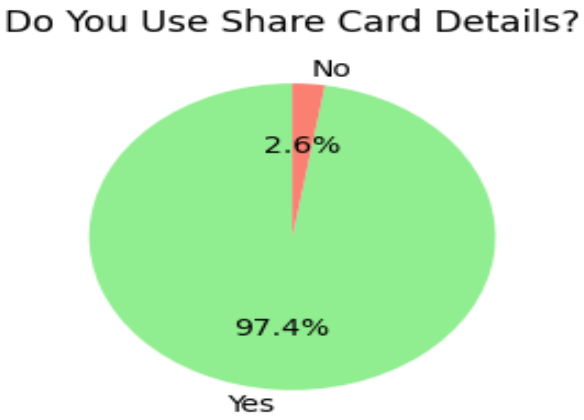

Figure 3: Sharing Card Details

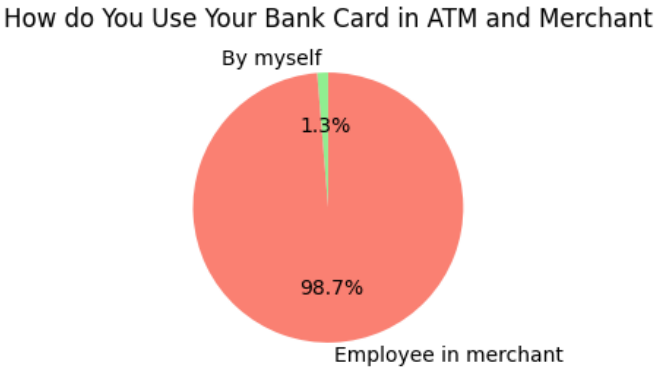

Figure 4: Pattern of Using Cards

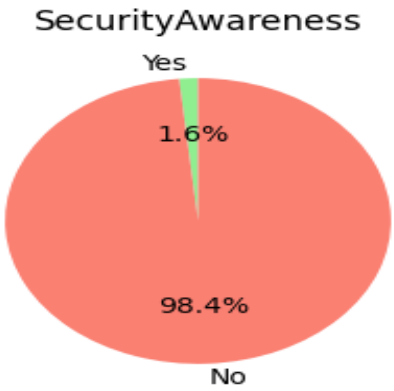

Figure 5: Awareness of Security Practices

5.4 Proactive Behavior and Security Measures

In terms of protective behavior, the majority of users adopted basic measures such as keeping cards in a secure place, checking bank statements regularly, and using cards only at trusted locations. A significant number also enabled SMS or mobile app alerts to monitor transactions. However, no respondents reported covering the keypad when entering their PIN a simple but vital security practice. Furthermore, while most claimed to use strong and unique PINs, nearly two-thirds never changed their PIN, significantly increasing their exposure to long-term fraud risks. These findings point to a concerning gap between basic security practices and more advanced, proactive measures (Figure 6 and 7).

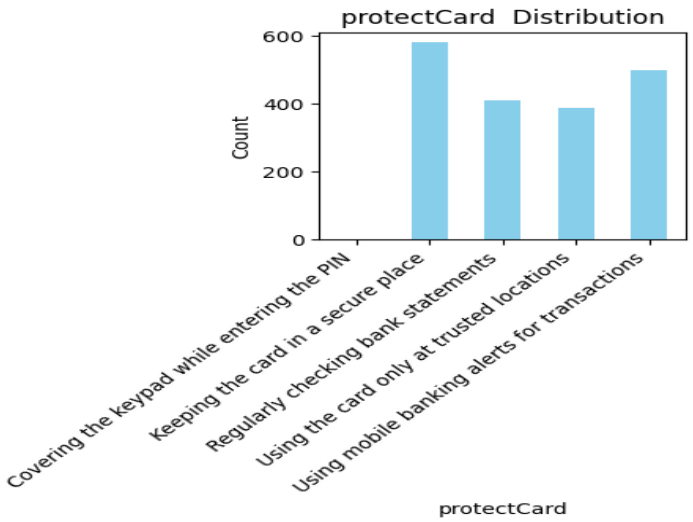

Figure 6: Card Protecting Behavior

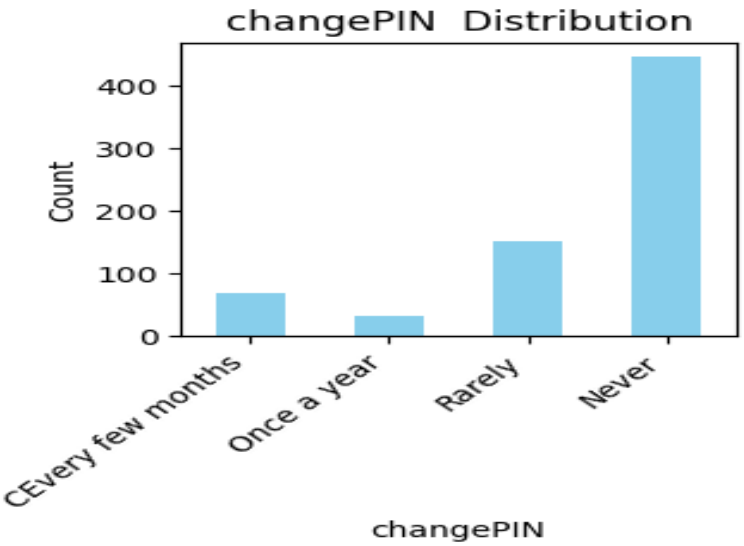

Figure 7: Changing PIN Pattern.

5.5 Fraud Experience and Institutional Support

Although most respondents had not personally experienced fraud, a concerning 14% had encountered unauthorized transactions. Even more alarming is the fact that none of the 700 participants reported receiving any guidance or advice from their financial institutions regarding card security. This lack of institutional communication reflects a serious failure to support and educate users in a high-risk environment (Figure 8).

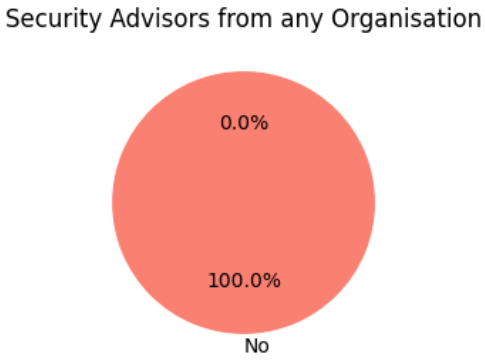

Figure 8: Receiving Security Advisor.

6 Recommendations

The findings of this study reveal significant vulnerabilities in cardholder behavior, institutional support, and the underlying payment technology used in Libya. To mitigate these risks and strengthen the security of digital transactions, the following recommendations are proposed:

6.1 Transition to Chip-Based Cards

Financial institutions in Libya should prioritize the migration from magnetic stripe cards to EMV chip-enabled cards. Chip cards offer enhanced security through encryption and dynamic data authentication, which significantly reduces the risk of card cloning and unauthorized access [9].

6.2 Public Awareness Campaigns

National awareness programs should be launched to educate the public on secure payment practices. These campaigns can be delivered via social media, television, bank branches, and printed brochures, with content tailored to various literacy levels and local dialects. Topics should include safe PIN management, recognizing tampered card readers, and the risks of sharing card information.

6.3 Mandatory Security Briefings

When issuing new cards, banks should provide cardholders with a brief but mandatory orientation on basic security measures. This could be a printed guide, a short video, or a one-on-one explanation during card activation, ensuring that users are immediately equipped with knowledge to protect their card and PIN.

6.4 Strengthening Mobile and SMS Alerts

Banks should encourage users to activate SMS or mobile app alerts for every transaction. These alerts provide real-time information that can help users detect fraud early and take immediate action. Incentives could be offered to promote the adoption of these services.

6.5 Institutional Support and Accountability

Financial institutions must improve customer service mechanisms related to fraud prevention and response. This includes providing clear procedures for reporting suspicious activity, offering immediate card blocking services, and regularly updating customers about emerging threats and protective practices.

6.6 Promote Regular PIN Updates

Users should be encouraged to change their card PINs regularly, ideally every few months. Financial systems can be programmed to remind users of this at ATM terminals or through mobile banking platforms. Regular PIN changes reduce long-term vulnerability to card misuse.

7 Conclusion

This study has highlighted significant security gaps among payment card users in Libya, especially those using magnetic stripe technology. Despite the increasing reliance on digital payments and the surge in card-based transactions, most users lack essential awareness regarding basic security practices. The findings reveal a widespread tendency to share card details publicly, weak PIN management, and low understanding of risks such as skimming and unauthorized access. Moreover, the total absence of institutional support or user education from banks further amplifies these vulnerabilities.

The results underscore the urgent need for intervention at both the user and institutional levels. While many users demonstrated some protective behaviors, such as using mobile alerts and keeping their cards in safe locations, these measures alone are not sufficient to address the complex threats posed by cybercrime and card fraud. Strengthening the security ecosystem requires a holistic approach—modernizing card infrastructure with EMV chips, educating users, and holding financial institutions accountable for providing proactive guidance.

As Libya continues its transition toward a cashless economy, ensuring the safety and confidence of users is critical. By implementing the recommendations outlined in this paper, stakeholders can build a more secure, informed, and resilient digital payment environment.

References

[1] K. Nair, "An Approch to Authenticate Magnetic Strip BanksCard Transaction at POS Terminals," *CSIR*, 2017.

[2] L. Karthigaa.M., Balachandar.N. and Karthi.M. , "Unassailable Cryptosystem for Securing Magnetic Stripe Card," *International Journal of Computer Science and Network Security*, vol. VOL.9 No.4, 2009.

[3] H. GuoJin and Bo Jin, "Forensic Analysis of Skimming Devices for Credit Fraud Detection," 9 2010.

[4] L. C. Bank, "Statement from the Central Bank of Libya regarding revenue and expenditure from 01-01-2024 to 31-12-2014," Tripoly, 2024.

[5] Kavita, "A STUDY ON CYBER SECURITY AWARENESS IN DIGITAL PAYMENT SYSTEM," *International Journal of Innovations & Research Analysis*, vol. 04, pp. 36-42, 2024.

[6] L. VAYALAMBRON, "AWARENESS AND EFFECTIVENESS OF DIGITAL PAYMENT USE AMONG RURAL HOUSEHOLDS," *International Advanced Research Journal in Science, Engineering and Technology* , vol. 11, no. 6, June 2024 , 2024.

[7] D. Mathur, "A Survey of Awareness about Security in E-payment System," *International Journal of Mechanical Engineering And Information Technology*, 2017.

[8] E. Taherdoost, Shamsul Sahibuddin and Neda Jalaliyoon, "Evaluation of User Awareness Affect on Smart Card Technology," in 1st World Conference on Information Technology , 2011.

[9] V. Company, "Visa," [Online]. Available: https://www.visa.com.sg/pay-with-visa/security/chip-cards.html. [Accessed 03 12 2025].

4

Vibration Analysis of the Centrifugal Pump

Vibration Analysis of the Centrifugal Pump

Noureddin Abdulhamid Magdoub
High Institute of Science and Technology Alzzawia , Libya.
Majnor03@gmail.com

Mohamed k. Husien
High Institute of Science and Technology Alzzawia , Libya.
Mohalk67@gmail.com

Ali Alhadi Amhimmid Karah
High Institute of Science and Technology Alzzawia , Libya.
Korra9919@gmail.com

Nadir Ali Mohammed
High Institute of Science and Technology Alzzawia , Libya.

Abstract

Mechanical vibration cause severe damage to the machine, by known the levels of the vibration, the predictive maintenance for the machine can be done. In This paper we study and investigate the vibration analysis of the Centrifugal Pump located at the desalination unit in the Azzawiya Oil Refining Company, the results were that the Overall vibration levels before maintenance were too high but after maintenance the Overall vibration levels were too low, there was an improvement in the performance of the pump.

Keywords: vibration, Centrifugal, Pump, Parameters, spectra.

1. Introduction

Many oil refining companies or factories use the centrifugal pumps to pull out the liquid or pump it into the main tank or something else. These pumps may crash and stop working at any moment if there is no maintenance schedule. The predictive maintenance of the machine performance is based on the mechanical vibration spectrum analysis. One of the main maintenance programs in the Azzawiya Oil Refining Company is the predictive maintenance. In this work, The Horizontal Centrifugal Pump is the case study used as a source of the vibration mechanic, which is located at the desalination unit of the first stage in the Azzawiya Oil Refining Company, which is part of the predictive maintenance program subjected to the regular inspections vibrometricas in each year periodically.

1.1 Literature review

Most of the previous studies focus on the defects and the faults which cause due to the vibration. Mehran Jahangiri et al were they present and investigate the impeller clogging in a centrifugal pump; it was found the measuring vibration values in three directions show that the clogged impeller has more effects on the axial vibration responses, so that the electric current consumption if electromotor is reduced [1].Vincent Becker et al focus on the fault of the pumps experimentally; the results were that, the efficiency is reduced by 9 to 15% in case of faulty operation [2]. Masoud Hatami Garousi et al present and discuss the vibration based method on the benefits and drawbacks of the multi layer perceptron method, the results illustrate that the algorithm was suggested, can identify different faults and determine their quantity during the real time of the operation pump [3]. Aditya U. Ganapathy et al Study the vibration analysis due to the water pump at the medium pressure. The result, after the correction the result was in the acceptable health condition [4]. Vincent Becker et

al Study two current based approaches methods for the detection of bearing defect and impeller clogging. The results were an adaptation of MCSA as a first approach and ATCSA as a second approach, so both approaches were useful for diagnosis of pumps [5]. In [6] the Study was the measuring of the velocity vibration on the secondary coolant pump using the vibration meter and piezoelectric accelerometer of the kartini reactor. The results of the vibration on the pump motor were 5.74 mm/s which can cause damage to the components of the pump. [7] The Study was presents and investigates the mathematical analysis of the results taken from a DCVA. The results was compare d to two different location and it was found that the amplitude of the system increase by creating the unbalance. In [8] the Study was the hydrodynamic effect of the fluid on the rotor. The results; were the impeller diameters less than 300mm has a little effect on the vibration of the rotor, also when the speed of rotor increase, the motion becomes stable. Theoretically, [9] use the finite element method, blade tip timing and strain gauge to investigate the vibration analysis of the centrifugal pump. The results were compared across conditions including free system, forced system, damped and undamped system.

1.2 Parameters and Abbreviations

Table 1: shows the Parameters and Abbreviations used in this Study.

1-Electric Motor - NDE	**N**on **d**rive **e**nd : refers to the bearing supports the shaft in the motor
2-Electric Motor - DE	**D**rive **e**nd: refers to the bearing supports the shaft which is coupled with the component
3-Pump - DE	**D**rive **e**nd: refers to the Pump bearing, next to the coupling.
4-Pump - NDE	**N**on **d**rive **e**nd: refers to the Pump bearing, away from the coupling.
Speed unit	HZ or RPM
Vibration velocity	mm/s

unit	
Power unit	Hp or kw
Overall vibration level	Acceleration, Velocity and Absolute Displacement
Defect factor	Indicator embedding ONEPROD expertise, it represents to the bearing health conditions of the machine, the recommended thresholds are 6 for the alarm and 9 for danger.
SFX	Shock Finder index is based on the ACOEM Shock Finder technology which has the value from 0 to 5 with alarm and danger thresholds at 3 and 5.
HMX	Harmonic index indicates if low or medium frequency spectra includes significant harmonic families which has the value from 0 to 3 with alarm and danger thresholds at 2 and 3
AL pAL	Indicates to alarm, the value from 0 to 5 Indicates to pre alarm of the value
Dg	Indicates to danger thresholds at 3 and 5.
Baseline	It is the data defined by the user to facilitate the comparison with other data and it can be modify at any time.
1-MT -Ax	The measured point on motor at location 1 Axial Direction
1-MT-RH	The measured point at location 1 Radial, horizontal Direction
1-MT-RV	The measured point at location 1 Radial, vertical Direction
2-MT-Ax	The measured point on motor at location 2 Axial Direction
2-MT-RH	The measured point at location 2 Radial, horizontal Direction
2-MT-RV	The measured point at location 2 Radial, vertical Direction
3-PP-Ax	The measured point on Pump at location 3 Axial Direction

3-PP-RH	The measured point at location 3 Radial, horizontal Direction
3-PP-RV	The measured point at location 3 Radial, vertical Direction
4-PP-Ax	The measured point on Pump at location 4 Axial Direction
4-PP-RH	The measured point at location 4 Radial, horizontal Direction
4-PP-RV	The measured point at location 4 Radial, vertical Direction

2. The vibration

The vibration of mechanics defined as the motion of the particle between two points and repeated itself in interval of time. The vibration of mechanics comes from the act of the force on the mechanic system. The force causing the mechanical vibration is generated through the motion of the machines' parts. The mechanical system can be modeled as mass-spring- damper system.

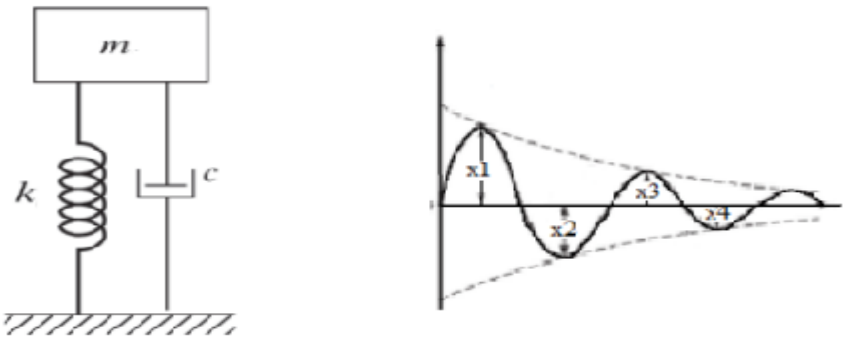

Figure 1: shows the modeled of the mechanical vibration

The equations of the motion that describe the vibration can be written as follows:

$$m\ddot{x} + c\,\dot{x} + k\,x = 0 \qquad (1)$$

Where x , $\dot{x}$ and $\ddot{x}$ are the parameters of the vibration amplitude of the displacement (micrometer), velocity (mm/s) and acceleration (g) respectively. So that: the solution of the pervious equation is of the form:

$$x_t(t) = A_1 \cos(wt) + A_2 \sin(wt) \qquad (2)$$

Where A_1 , A_2 are constant of the integration and w is the angular frequency of the oscillation and its equal to:

$$w = 2\pi f \quad \text{Or} \quad w = \frac{2\pi}{T} \qquad (3)$$

Where: f , T are the frequency and the period of the oscillation respectively. By known the pervious parameters, the vibration spectra in figure (2) can be analyze. Thus, it gets some information about the mechanic system (motor, pump) like defect of bearing unbalance, misalignment, structural resonance, (wear/ lubrication) and cavitation.

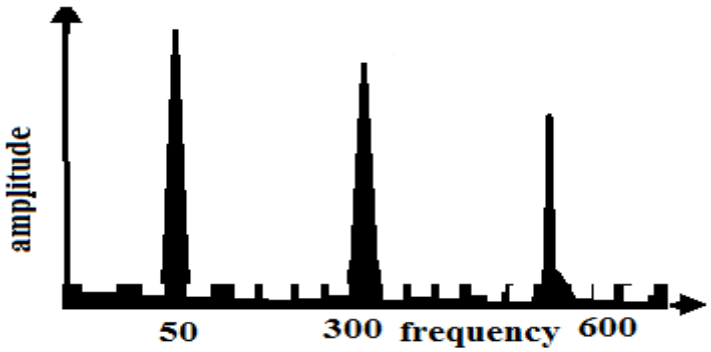

Figure (2) shows the spectra of the mechanical vibration

3. The case under study

The Horizontal Centrifugal Pump is the case study which is located at the desalination unit of the first stage in the Azzawiya Oil Refining Company. The function of this pump is to transfer the condensed steam to the boiler to inject it again and then pump it to the main tank of the desalination unit. Figure (3) shows the photo of the case study so that, the description of the Mechanical Equipments illustrate in table (2).

Figure 3: shows the photo of the case study

Table 2: shows the description of the Mechanical Equipments of the case study.

Pump, Centrifugal, Horizontal (CONDENSATE PUMP0	
TYPE	40250 H
FLOW RATE	11.7 M^3/H
SPEED	2940 RPM
TOTAL HEAD	66M
MOTOR	
TYPE	M2QA 160 M2B
VOLTAGE	380V
POWER	11KW
SPEED	2940 RPM
FREQUENCY	50 HZ

4. Data collection

Using the portable FALCON analyzer model12481as an instrument, which connect to the PC, Data automatically, collect at PC screen. Using NESTi4.0 program software includes powerful tools for machine analysis to confirm the automatic diagnosis and with ACOEM ACCURX analysis method it is offers the possibility to get automatic diagnosis for all defects due to the vibrations. To plot the vibration spectra, the VIBGRAPH should be on the main screen so that data analyst and reported in docx Format file represented by the vibration parameters like: Overall vibration velocity, absolute displacement, Overall acceleration, Defect factor, SFX and HMX. So that it is easy to understand the information and what's happening on the machines.

4.1 Data collection of the vibration spectra before Maintenance

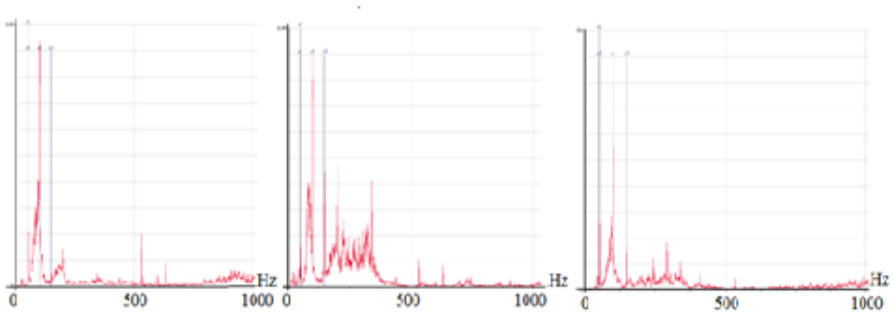

Figure 4: Illustrate the vibration spectra before maintenance at point 1, left at 1MT.Ax, middle at 1MT.Rh and right at 1MT.Rv direction.

Table 3: Shows the value of the parameters due to the vibration at point1 before maintenance.

Date 9/5/2024 10:58AM			value of the vibration			Alarm threshold		
Point1MT	Status	unit	1MT.Ax	1MT.Rh	1MT.Rv	**pAl**	**Al**	**Dg**
Overall vib.	Dg	mm/s	**4.57**	3.04	**4.89**	1.4	2.8	4.5
OverallAcc.	ok	g	0.689	1.13	1.83	2	3	6
Defectfactor	ok	def	2.87	3.08	3.19	4	6	9
SFx	Ok		0	1	0	2	3	5
HNx	ok		0	1	1	1	2	3

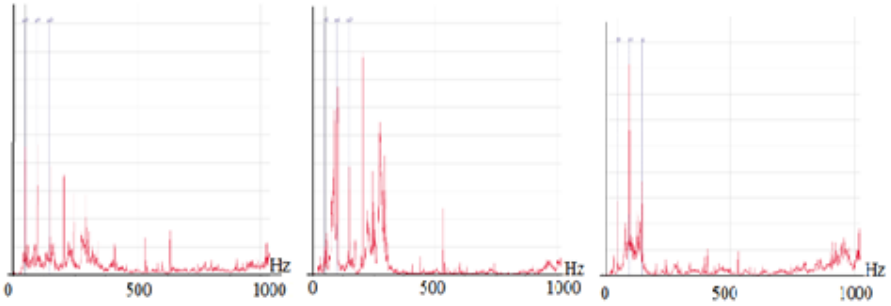

Figure 5: Illustrate the vibration spectra before maintenance at point 2, left at 2MT.Ax, middle at 2MT.Rh and right at 2MT.Rv direction.

Table 4: Shows the value of the parameters due to the vibration at point2 before maintenance.

Date 9/5/2024 10:58AM			value of the vibration			Alarm threshold		
Point2MT	Status	unit	2MT.Ax	2MT.Rh	2MT.Rv	**pAl**	**Al**	**Dg**
Overall vib.	**Dg**	mm/s	4.46	**2.98 Al**	**4.93**	1.4	2.8	4.5
OverallAcc.	AL	g	0.746	0.985	**2.12 Al**	2	3	6
Defectfactor	ok	def	2.59	2.84	3.05	4	6	9
SFx	Ok		0	0	0	2	3	5
HNx	pAL		1	1	1	1	2	3

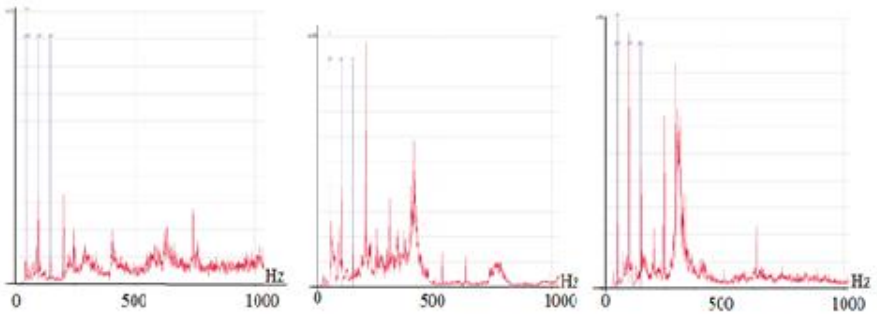

Figure 6: Illustrate the vibration spectra before maintenance at point 3, left at 3MT.Ax, middle at 3MT.Rh and right at 3MT.Rv direction

Table 5: Shows the value of the parameters due to the vibration at point3 before maintenance.

Date 9/5/2024 10:58AM			value of the vibration			Alarm threshold		
Point 3PP	Status	unit	3PP.Ax	3PP.Rh	3PP.Rv	**pAl**	**Al**	**Dg**
Overall vib.	Dg	mm/s	**4.66**	**8.91**	**8.14**	1.4	2.8	4.5
OverallAcc.	A	g	3.93	5.26	3.78	2	3	6
Defectfactor	pAL	def	5.38	5.37	4.75	4	6	9
SFx	Ok		0	0	0	2	3	5
HNx	pAL		1	1	1	1	2	3

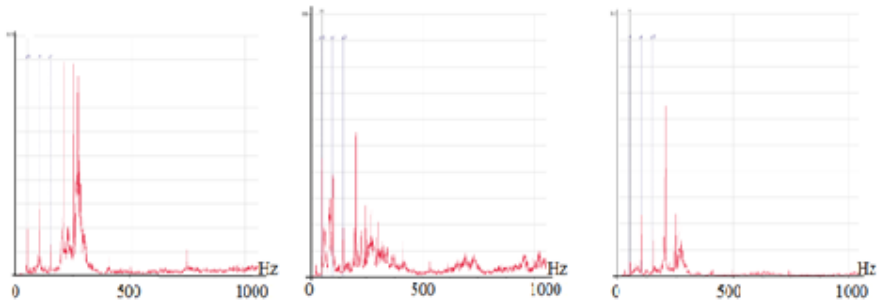

Figure 7: Illustrate the vibration spectra before maintenance at point 4, left at 4MT.Ax, middle at 4MT.Rh and right at 4MT.Rv direction

Table 6: Shows the value of the parameters due to the vibration at point4 before maintenance.

Date 9/5/2024 10:58AM			value of the vibration			Alarm threshold		
Point 4PP	Status	unit	4PP.Ax	4PP.Rh	4PP.Rv	**pAl**	**Al**	**Dg**
Overall vib.	**Dg**	mm/s	**5.64**	**5.19**	**7.45**	1.4	2.8	4.5
OverallAcc.	pAL	g	1.47	2.69	**3.05**	2	3	6
Defectfactor	pAL	def	4.21	4.33	4.3	4	6	9
SFx	Ok		1	1	1	2	3	5
HNx	pAL		1	1	1	1	2	3

4.2 Analysis report before maintenance

According to the previous tables the values of the Overall vibration of the machine at four points measured at 3 directions indicates to high level, it was cross to Alarm threshold 4.5 mm/s, this means there was problems. So that, using the portable FALCON analyzer, the automatic diagnosis was: unbalance, misalignment, structural resonance, bearing defect (wear/ lubrication) and cavitation. Overall state is not acceptable for the components "ElectricMotor1" and "PUMP1". There are strong structural resonance that amplifies vibration imbalance at rotational frequency at the four locations 1-ElectricMotor-NDE, 2-ElectricMotor-DE, 3-Pump-DE and 4-Pump-NDE. The unbalance, misalignment between motor and the pump, and structural resonance should be corrected. At locations 3-Pump-DE and 4-Pump-NDE: Bearing defect should be corrected or lubricate the component then measure again. Lubricate and replace the bearing on the component Pomp1. So that the Cavitation should be correct.

5. The main maintenance

In order to improve the dynamic condition of the pump, then to record low vibration levels, the main maintenance was lubricated the pump by technical services team, then replaced the old parts by new spear parts as illustrated in the table (7).

Table 7: shows the main parts which are replaced by new spear parts.

Impeller	Bearing
Shaft	Mechanical seal
O-ring	Screw plug
Support foot	Gasket

Realignment all the spear parts within the pump, fixing the screw plug, and making a new measurement at the same last four points, the results was shown as following:

5.1 Data collection of the vibration spectra after Maintenance

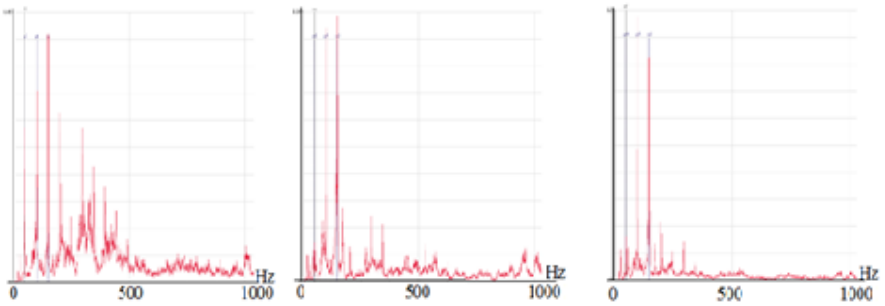

Figure 8: Illustrate the vibration spectra after maintenance at point 1, left at 1MT.Ax, middle at 1MT.Rh and right at 1MT.Rv direction

Table 8: shows the value of the parameters due to the vibration after maintenance at point1.

Date 10/15/2024 11:08AM			value of the vibration			Alarm threshold		
Point1MT	Status	unit	1MT.Ax	1MT.Rh	1MT.Rv	**pAl**	**Al**	**Dg**
Overall vib.	ok	mm/s	1.34	1.47	3.24 **Al**	1.4	2.8	4.5
OverallAcc.	ok	g	0.173	0.139	0.284	2	3	6
Defectfactor	ok	def	2.36	2.5	2.42	4	6	9
SFx	Ok		0	0	0	2	3	5
HNx	ok		0	0	0	1	2	3

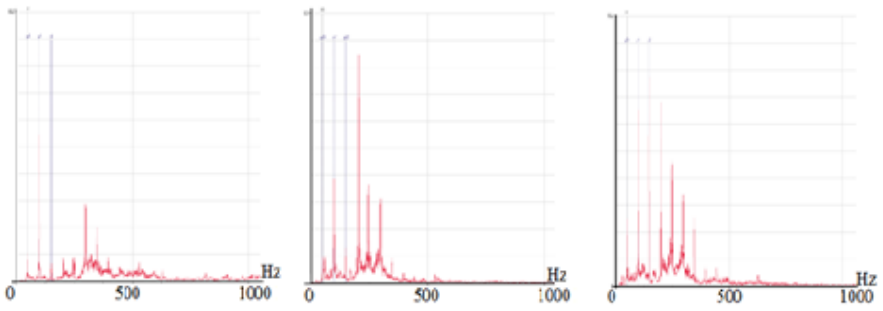

Figure 9: illustrate the vibration spectra after maintenance at point 2, left at 2MT.Ax middle at 2MT.Rh and right at 2MT.Rv direction.

Table 9: shows the value of the parameters due to the vibration at point2 after maintenance.

Date 10/15/2024 11:08AM			value of the vibration			Alarm threshold		
Point2MT	Status	unit	2MT.Ax	2MT.Rh	2MT.Rv	**pAl**	**Al**	**Dg**
Overall vib.	ok	mm/s	0.704	1.37	1.29	1.4	2.8	4.5
OverallAcc.	ok	g	0.22	0.16	0.253	2	3	6
Defectfactor	ok	def	2.2	2.11	2.31	4	6	9
SFx	Ok		1	0	0	2	3	5
HNx	ok		0	1	1	1	2	3

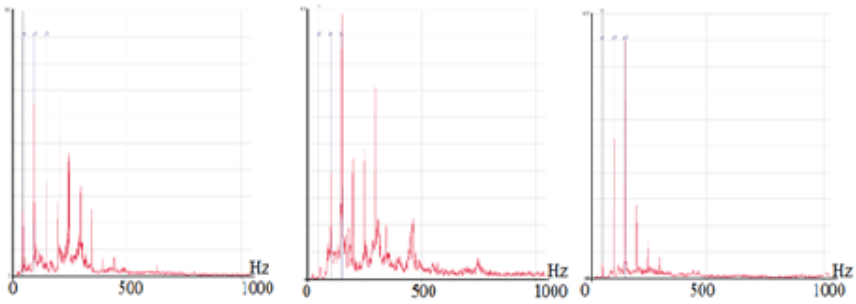

Figure 10: Illustrate the vibration spectra after maintenance at point 3, left at 3MT.Ax, middle at 3MT.Rh and right at 3MT.Rv direction

Table 10: Shows the value of the parameters of the vibration after maintenance at point3

Date 10/15/2024 11:08AM			value of the vibration			Alarm threshold		
Point 3PP	Status	unit	3PP.Ax	3PP.Rh	3PP.Rv	**pAl**	**Al**	**Dg**
Overall vib.	ok	mm/s	0.988	1.65	1.49	1.4	2.8	4.5
OverallAcc.	ok	g	0.781	0.643	0.579	2	3	6
Defectfactor	ok	def	3.43	3.62	3.38	4	6	9
SFx	Ok		1	1	3	2	3	5
HNx	ok		0	0	0	1	2	3

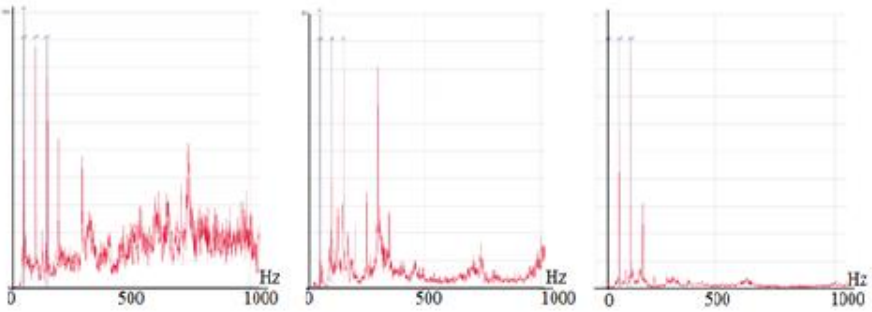

Figure 11: Illustrate the vibration spectra after maintenance at point 4, left at 4MT.Ax middle at 4MT.Rh and right at 4MT.Rv direction

Table 11: shows the value of the parameters due to the vibration after maintenance at point4

Date 10/15/2024 11:08AM			value of the vibration			Alarm threshold		
Point 4PP	Status	unit	4PP.Ax	4PP.Rh	4PP.Rv	**pAl**	**Al**	**Dg**
Overall vib.	ok	mm/s	1.56	1.76	2.07	1.4	2.8	4.5
OverallAcc.	ok	g	0.569	0.511	0.336	2	3	6
Defectfactor	ok	def	3.07	2.93	2.97	4	6	9
SFx	Ok		0	0	0	2	3	5
HNx	ok		0	0	0	1	2	3

5.2 Analysis report after maintenance

In the previous tables all the values of the parameters of the vibration of the machine semis to be as normal values which means within the acceptable range that is illustrated in the last three columns named as the Alarm threshold. Therefore, the automatically diagnosis shows: a good overall state for the component "ElectricMotor1"and "Pump1". Which means to be no action required, consequently health is excellent (no defect, low vibration levels).

6. Conclusion

The main objective of this research paper is to use the vibration technical to identify the risk and the defects of the pump, using the portable FALCON analyzer model1248 before and after maintenance, the results were accurate. Before the intervention and maintenance the Overall vibration levels were too high, it was cross to Alarm threshold 4.5 mm/s, and some of them recorded 8.91 at 3PP.Rh, 8.14 at 3PP.Rv and 7.45 at 4PPRv. After the intervention and maintenance the Overall vibration levels were too low, this means there is an improvement in the performance of the pump.

Acknowledgement

The authors would like to thank the technical services team at the desalination unit in the Azzawiya Oil Refining Company for their assistance and advice.

Reference

[1] Mehran Jahangiri et al "clogged impeller diagnosis in the centrifugal pimp using the vibration motor current analysis" Journal of Applied and Computational Mechanics, Vol.4, No.4 (**2018**), 310-317 ISSN: 2383-4536.

[2] Vincent Becker et al "fault investigation of circulation pumps to detect impeller clogging" Appl. Sci. **2020**, 10, 7550, doi: 10 3390/app10217550 www.mdpi.com/journal/applsci.

[3] Masoud Hatami Garousi et al "vibration analysis of a centrifugal pump with healthy and defective impellers and fault detection using multilayer perceptron" Eng**2024**,5,2511-2530. doi.org/10.339/eng5040131 www.mdpi.com/ Journal/eng

[4] Aditya U. Ganapathy et al "vibration analysis a key for pump maintenance case study" IJMER/ISSN:2249-6645 /Vol.4/Iss.3/Mar.**2014**/58/ www.ijmer.com

[5] Vincent Becker et al "two current based methods for the detection of bearing and impeller faults in variable speed pumps" Energies **2021**,14,4514,doi.org/10.3390/en14154514www.mdpi.com/journal/energis

[6] H Ahmadi et al "vibration analysis of kartini reactor secondary cooling pump using FFTanalyzer" Journal of Physics: conference series 1511(**2020**)012080 doi:10.1088/1742-6596/1511/1/012080

[7] Sanjiv kumar et al "vibrations analysis of 4Jaw flexible coupling considering unbalancing in two planes" international

journal of science and technology (IJST) volume1 No.11, November, **2012**

[8] Madina Isametova et al "development and verification of a centrifugal pump rotor model based on integrated multibody dynamics in the ADAMS environment" App. Sci. **2025**, 15, 9132. doi.org./10.3390/app15169132

[9] Muhammad U. Najeem et al "a critical analysis of design for reduction in vibration of centrifugal impellers" Pakistan journal of engineering and technology volume7 No.2, **2024** doi.org/10.51846/vol7iss2pp79-96

5

Experimental Evaluation of a Vacuum-Assisted Solar Still for Improved Desalination Efficiency

Experimental Evaluation of a Vacuum-Assisted Solar Still for Improved Desalination Efficiency

Ali K. Muftah
Sabratah University, Sabratah, Libya
ali.muftah@sabu.edu.ly

Anas Elayeb
Sabratha University, Sabratha, Libya
AnasElayeb69@gmail.com

Mabrouk Abugadura
Sabratha University, Sabratha, Libya
mabrouk@sabu.edu.ly

Abstract

The worsening challenge of accessing freshwater and the rising cost of energy required for its production through conventional desalination methods have intensified research efforts toward utilizing solar energy for seawater desalination. This study focuses on enhancing the performance of a solar still by integrating it with a vacuum-assisted system to create partial vacuum conditions inside the still. The modified design incorporates a closed-loop water circulation system equipped with a Venturi tube to reduce internal pressure, which in turn lowers the evaporation temperature and increases freshwater output. Field experiments were conducted under both atmospheric and reduced-pressure conditions, evaluating the effects of basin water depth, ambient temperature, and wind speed. The results showed up to a 30% increase in productivity under vacuum conditions, especially at higher water depths. These findings highlight the potential of low-cost, vacuum-enhanced solar stills for off-grid freshwater production in arid regions.

Keywords: solar desalination, vacuum system, freshwater production, solar still.

1. Introduction

Access to clean freshwater remains one of the most pressing global challenges of the 21st century. Although water covers approximately 71% of the Earth's surface, about 96.5% of it exists in the form of saline water in oceans and seas, leaving only 2.5% as freshwater. Of this small fraction, less than 1% is readily accessible for direct human use, primarily from rivers, lakes, and shallow groundwater sources [1][2]. The combination of population growth, rapid urbanization, industrial expansion, and the impacts of climate change has placed unprecedented pressure on existing freshwater supplies. According to the World Health Organization, around 2.2 billion people worldwide still lack access to safe drinking water [3].

To meet the growing water demand, desalination has become an increasingly important strategy for augmenting freshwater resources, particularly in arid and water-scarce regions. Desalination involves the removal of salts and other impurities from saline water, rendering it suitable for human consumption and various economic activities [4]. Desalination technologies are broadly classified into thermal and membrane-based processes. Thermal methods—such as Multi-Stage Flash (MSF), Multi-Effect Distillation (MED), and Vapor Compression Distillation (VCD)—utilize heat to evaporate water, separating it from dissolved salts. While MSF and MED are mature technologies commonly used in the Middle East, they are energy-intensive and often reliant on fossil fuels, making them costly and environmentally burdensome [5][6].

In contrast, membrane-based technologies, particularly Reverse Osmosis (RO), have gained prominence due to their higher energy efficiency, modularity, and adaptability. RO systems employ semi-permeable membranes to separate freshwater from saline solutions under high pressure. These systems are now widely deployed for both seawater desalination and wastewater reuse, offering advantages such as lower operational costs and minimal land requirements [7][8][9].

However, despite these advancements, the energy intensity of desalination remains a critical concern, particularly when conventional energy sources are used.

In this context, solar desalination emerges as a sustainable alternative that leverages abundant and renewable solar energy to drive the separation process. Among the simplest and most accessible forms of solar desalination is the solar still, which mimics the natural hydrologic cycle by evaporating saline water using solar radiation and condensing the vapor to yield freshwater. These systems are well-suited for decentralized, small-scale applications in remote or off-grid communities. Nevertheless, conventional solar stills suffer from inherently low productivity, often limited by high evaporation temperatures, thermal losses, and inefficient heat transfer [9].

To address these limitations, researchers have explored various performance enhancement techniques, including geometric modifications, the use of advanced materials, and system integration with auxiliary energy sources. One promising approach is the application of vacuum-assisted systems to reduce the internal pressure of the solar still. Lowering the pressure decreases the boiling point of water, enabling evaporation at lower temperatures and potentially increasing freshwater yield without the need for additional thermal input.

Enhancing the productivity of solar stills has been the focus of extensive research, with various approaches including system integration, material modification, geometric optimization, and hybrid configurations. These innovations aim to overcome the inherently low yield of conventional solar stills and improve their viability for decentralized water treatment, especially in arid and off-grid regions [10].

Freshwater scarcity poses an urgent challenge in many parts of the world, particularly in arid and remote regions lacking access to centralized infrastructure. Solar distillation has emerged as a sustainable and low-cost method for water purification, offering the

ability to convert saline or brackish water into potable water using abundant solar energy. Despite its simplicity and eco-friendliness, the traditional solar still suffers from limited productivity due to low thermal efficiency and evaporation rates.

To address these limitations, numerous strategies have been explored. The integration of vacuum systems into solar stills has shown great potential to enhance evaporation by lowering the boiling point of water. Danish et al. (2019). demonstrated that coupling a geothermal heat exchanger with a vacuum pump led to a 30.5% increase in daily freshwater yield[11]. Mohamed et al. (2023) reported that forced vacuum systems could improve productivity by up to 80% compared to natural vacuum setups, highlighting the substantial gains achievable through pressure manipulation.[12]

In parallel, thermal energy storage using phase change materials (PCMs) has been widely studied to extend the operational period of solar stills beyond peak solar hours [13]. Elmghari et al. (2025) achieved a 27.7% increase in output using 2 kg of paraffin wax[14], while Karthikeyan et al. (2023) emphasized the high latent heat capacity of PCMs as critical to thermal stability. [15]

Basin enhancements have also played a pivotal role. AbdAllah et al. (2018) found that using wick materials like jute and sponge can boost productivity by up to 180% [16], and Agrawal and Rana (2023) recorded a 50% increase in yield using floating V-shaped black jute wicks. Geometrical and auxiliary enhancements [17], such as thermoelectric modules. (Esfe et al., 2021) and evacuated copper heat absorbers (Jaafar & Hameed, 2021), further contribute to temperature gains and higher distillation rates.

Hybrid desalination systems—combining solar stills with reverse osmosis, humidification–dehumidification (HDH), geothermal, or waste heat recovery techniques—have gained traction for their enhanced performance [18][19]. Abdelaziz et al. (2021) and Shalaby et al. (2023) reviewed such systems, noting higher gain out put ratios and reduced specific energy consumption [20][21], while Abdullah et

al. (2024) demonstrated the viability of solar-powered hybrid HDH/RO systems achieving high water productivity at a low cost [22].

Despite these advancements, the complexity, cost, and energy requirements of active systems often hinder widespread adoption in low-resource contexts. Therefore, this study presents a novel passive vacuum-assisted solar still design that incorporates a Venturi tube within a closed-loop water circulation system to generate partial vacuum conditions without the need for external power. This innovation aims to combine the efficiency benefits of reduced pressure with the simplicity and affordability necessary for off-grid deployment.

The system is experimentally evaluated under varying conditions of water depth, ambient temperature, wind speed, and time of day to assess its thermal performance and freshwater output compared to a conventional solar still. The goal is to offer a cost-effective, energy-efficient, and scalable desalination solution tailored for water-stressed regions.

2. Experimental Setup

This study presents an innovative solar still design aimed at enhancing freshwater production efficiency through vacuum-assisted solar distillation. The system incorporates a closed-loop water circulation circuit integrated with a Venturi tube, which induces a partial vacuum within the evaporation chamber. This pressure reduction lowers the boiling point of water, thereby increasing the evaporation rate—particularly during the early stages of solar heating—and improving overall productivity compared to conventional atmospheric-pressure designs.

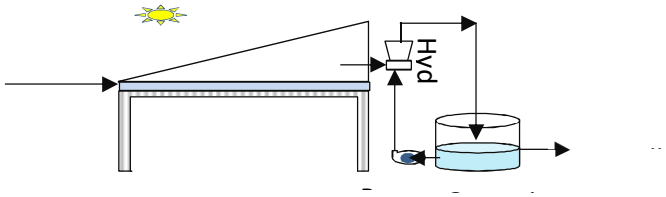

Figure 2.1: Schematic Diagram of the Vacuum-assisted Solar Still

Key design considerations included optimizing the tilt angle of the glass cover to maximize year-round solar radiation absorption, employing a black-coated iron basin for enhanced solar heat absorption, and using thermal insulation (5 cm thick polystyrene) on the basin base and walls to reduce heat losses. The Venturi tube, driven by water flow, creates a localized pressure drop that facilitates vapor suction and accelerates evaporation.

Figure 2.2 On-site Image of the Vacuum-assisted Solar Still

An external reservoir and small water pump were incorporated to ensure continuous circulation and vapor extraction. Condensation occurred in the reservoir, further enhancing water recovery efficiency. The glass cover was upgraded from standard 4 mm to 10 mm heat-treated glass to withstand internal pressure differences without compromising thermal transparency.

Materials were selected based on cost, durability, and local availability, with an emphasis on recycled and easily accessible components such as repurposed glass panels, metal tank parts, plastic pipes, and pumps. This approach enabled low-cost fabrication, ease of maintenance, and scalability of the system in resource-constrained environments.

To enable performance evaluation, sensors and measurement devices were installed at critical points in the system to monitor thermal conditions and environmental variables without disrupting system function. The overall design supports sustainable, affordable, and replicable freshwater production solutions for arid and off-grid regions.

3. Results and Discussion

This study investigated the performance of a solar still operating under atmospheric and vacuum-assisted conditions through a series of controlled field experiments conducted from August to November 2024. A total of ten experimental cases were analyzed—five with vacuum assistance and five without—under varying environmental conditions and water depths, with an emphasis on evaluating the impact of vacuum integration on system productivity.

3.1 Impact of Solar Radiation on Productivity

Results confirmed a strong positive correlation between solar radiation intensity and distillate output. Productivity increased with higher solar

irradiance, reaching a maximum of 4750 ml/day at 738 W/m², and a minimum of 1171 ml/day at 403 W/m². The vacuum-assisted system exhibited enhanced sensitivity to solar input, particularly during early hours of operation.

Table (1): Daily average solar radiation and productivity readings for the case without vacuum

AV(I)W/ m^2	738. 25	688. 19	654. 77	616. 25	489. 13	418. 82	409. 96	403. 62
M ml	4750	3350	3200	2630	2200	1647	1511	1171

Table (2): Daily average solar radiation and productivity readings for the case with vacuum

AV(I)w/m^2	**599.41**	**596.06**	**553.5**	**549**	**477.03**	**444.44**	**408.34**	**364.75**
M.ml	**4200**	**3600**	**3500**	**2960**	**2920**	**1900**	**1816**	**1243**

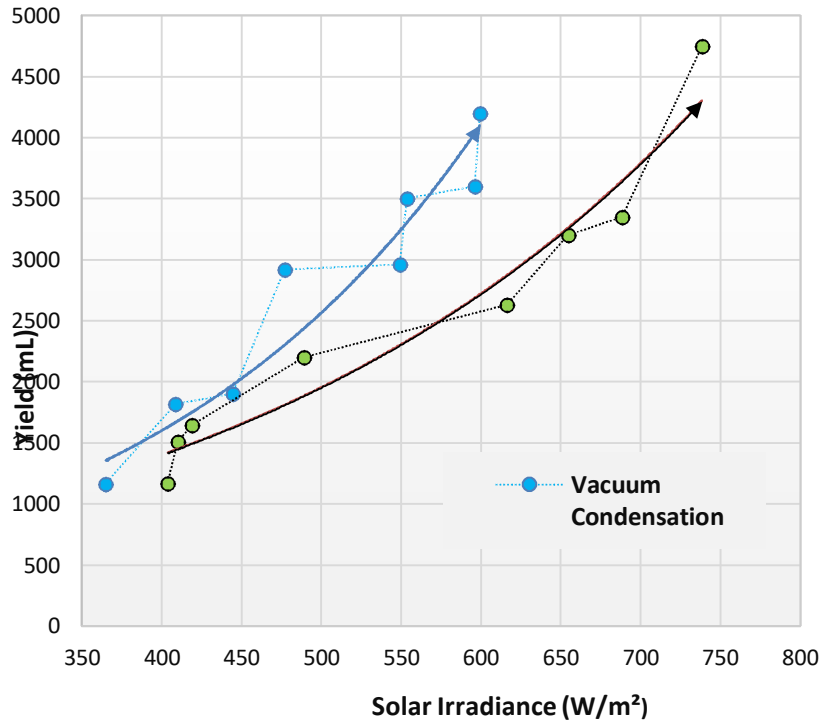

Figure 1.3: Effect of Solar Irradiance on Distillate Productivity under Different Condensation Condition

3.2 Effect of Basin Water Depth

Varying the water depth revealed an inverse relationship with productivity. Shallower water levels facilitated faster heating and higher distillation rates. Under vacuum conditions, a 1.5 cm depth yielded 4200 ml/day, while 4 cm depth yielded only 1243 ml/day. The effect was more pronounced in the vacuum system due to the greater thermal sensitivity of reduced-pressure environments.

Table (3): Daily Average distillate yield and basin water level under atmospheric conditions.

M.ml	3200	2200	1647	1511	1171	3350	2630	4750
Water Level(cm)	1.5	2	3	3.5	4	2	2	1

Table (4): Daily averages of distillate output and basin water level during vacuum-assisted operation

M.ml	4200	2920	1900	1816	1243	2960	3500	3600
Water Level(cm)	1.5	2	3	3.5	4	2.5	2	2

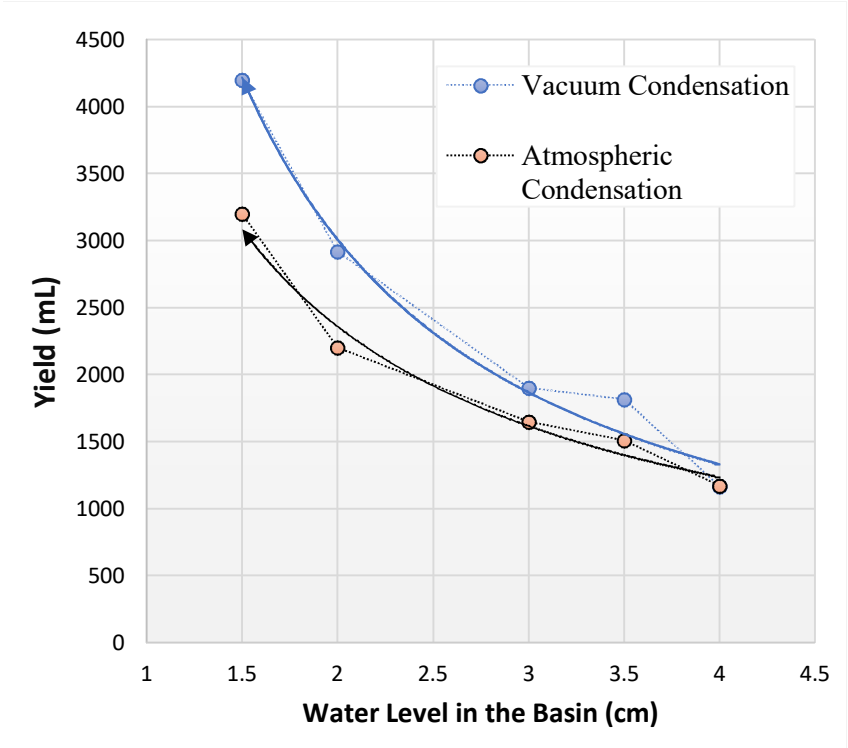

Figure 2.3: Variation of Distillate Yield with Water Level in the Basin for Vacuum and Non-Vacuum Modes

3.3 Influence of Wind Speed

Wind speed was found to negatively impact condensation efficiency in both systems due to increased convective heat losses from the glass cover. This effect was stronger under vacuum conditions, where higher vapor concentrations made the system more sensitive to cooling. Maximum productivity was observed at lower wind speeds (e.g., 1243 ml at 0.7 m/s) versus lower yields at higher speeds (e.g., 3600 ml at 1.9 m/s).

Table (5): Daily distillate yield and wind speed readings under vacuum-assisted operation.

M ml	1243	1816	2920	2960	3600
wind speed(m/s)	1.9	1.7	1.47	1.03	0.7

Table (6): Daily distillate yield and wind speed readings under atmospheric (non-vacuum) operation

M ml	1647	1511	2200	2630	3200
wind speed(m/s)	1.4	1.172	0.84	0.6	0.47

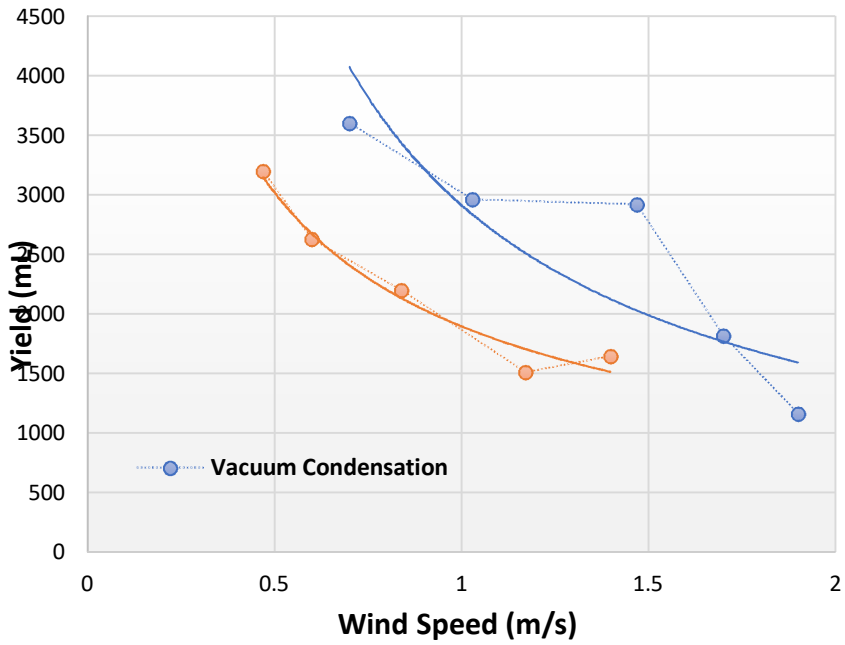

Figure 3.3: The relationship between wind speed (m/s) and the daily productivity of the atmospheric and vacuum condensation systems

3.4 Effect of Ambient Temperature

Productivity increased with rising ambient temperatures due to reduced heat losses and enhanced energy retention. The vacuum system maintained more stable output across temperature ranges. For example, at 31.2°C, the vacuum system produced 4200 ml, compared to 1243 ml at 20.3°C. The atmospheric system showed a similar trend but with steeper increases at higher temperatures.

Table (7): Average ambient temperature and daily productivity of the solar still under vacuum (reduced pressure) conditions

M.ml	4200	3600	3500	2920	1900	1816	1243
Ambient Air Temperature (°C)	31.2	29.7	29.5	23.2	22.1	20.7	20.3

Table (8): Average ambient temperature and daily productivity of the solar still under atmospheric pressure conditions

M.ml	4750	3350	3200	2200	1647	1511	1172
Ambient Air Temperature (°C)	30.6	30.1	30.1	28.6	27	25	22.11

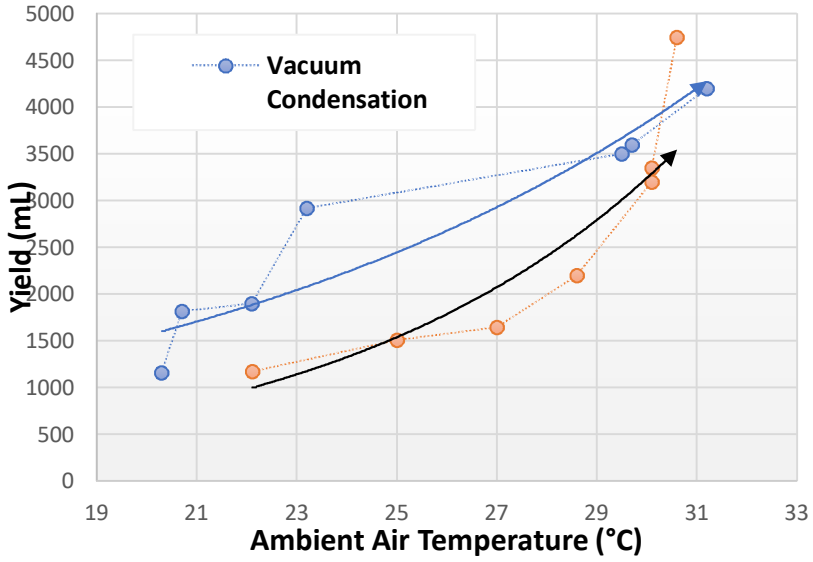

Figure 4.3: Effect of Ambient Air Temperature on Daily Yield of Atmospheric and Vacuum Condensation System.

3.5 Productivity Variation with Time of Day

Hourly productivity trends revealed that the vacuum system achieved higher output earlier in the day, indicating better responsiveness to initial solar heating. Peak productivity typically occurred earlier under vacuum conditions compared to atmospheric operation. Differences between the two modes were most significant during morning hours, then gradually converged around midday.

Table 9: Hourly Productivity Measurements under Vacuum (Reduced Pressure) Conditions.

Time (hours)	00:30	01:30	02:30	03:30	04:30	05:30	6:30
Productivity (ml/hour)	268	520	585	566	496	279	206

Table 10: Hourly Productivity Measurements under atmospheric pressure conditions

Time (hours)	00:30	01:30	02:30	03:30	04:30	05:30	6:30
Productivity (ml/hour)	204	392	440	428	372	210	154

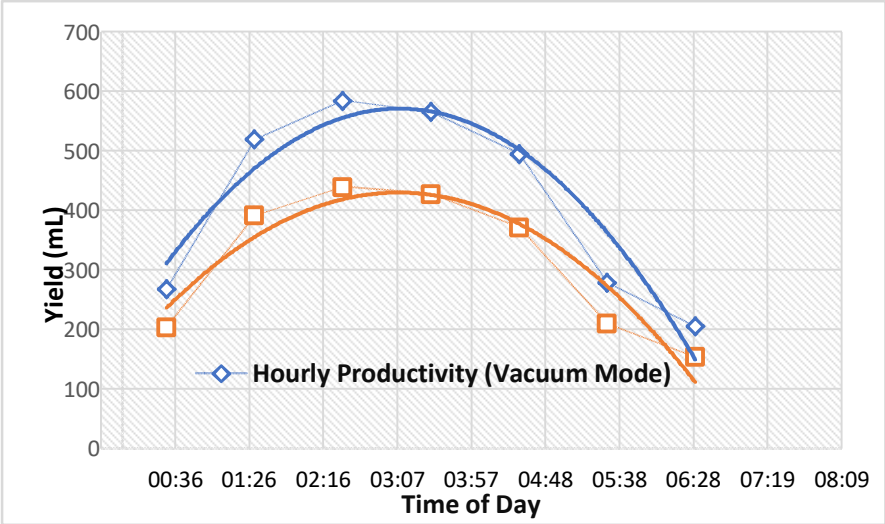

Figure 5.3: Hourly Productivity Under Atmospheric and Vacuum Conditions

Table11: Hourly Productivity Measurements under Vacuum (Reduced Pressure) Conditions

Time (hours)	00:30	01:30	02:30	03:30	04:30	05:30	6:30
Productivity (ml/hour)	189	191	316	395	274	239	186

Table 12: Hourly Productivity Measurements under atmospheric pressure conditions

Time (hours)	00:30	01:30	02:30	03:30	04:30	05:30	6:30
Productivity (ml/hour)	50	204	305	341	251	222	138

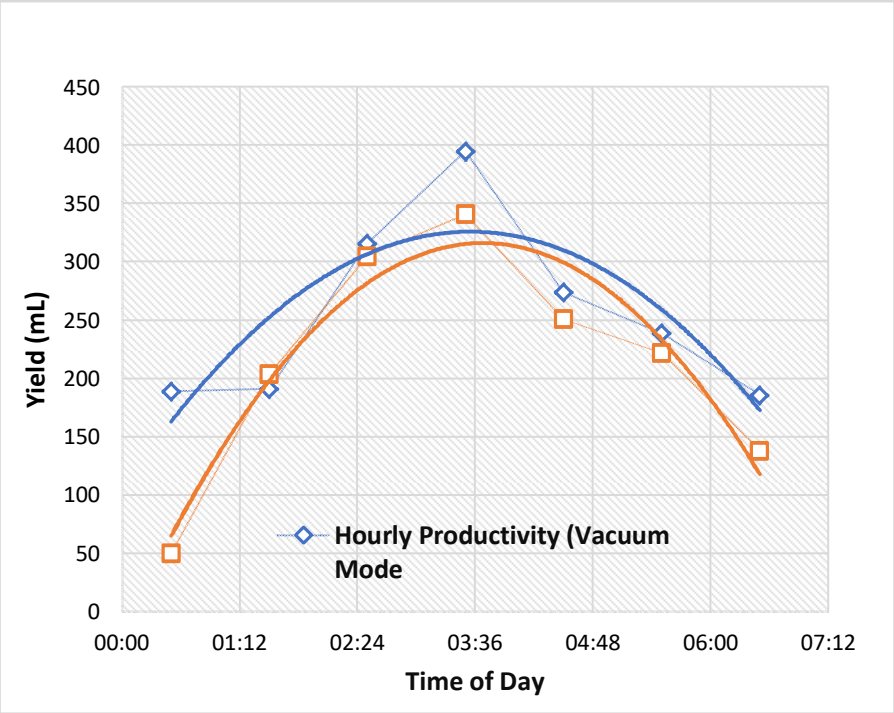

Figure 6.3: Hourly Productivity Under Atmospheric and Vacuum Conditions

Table13: Hourly Productivity Measurements under Vacuum (Reduced Pressure) Conditions

Time (hours)	00:30	01:30	02:30	03:30	04:30	05:30	6:30
Productivity (ml/hour)	151	163	284	326	349	245	202

Table 14: Hourly Productivity Measurements under atmospheric pressure conditions

Time (hours)	00:30	01:30	02:30	03:30	04:30	05:30	6:30
Productivity (ml/hour)	70	153	280	320	344	272	208

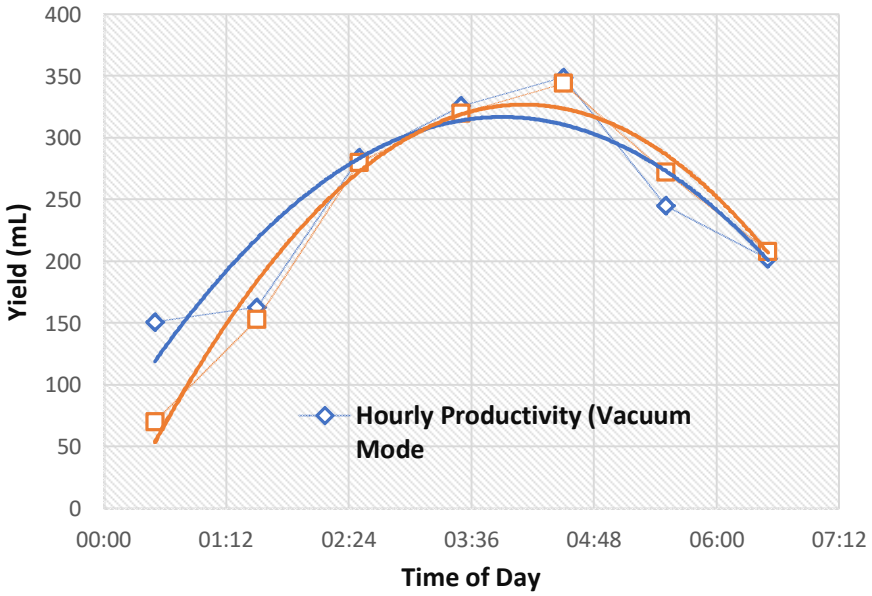

Figure 7.3 : Hourly Productivity Under Atmospheric and Vacuum Conditions

3.6 Overall Performance of the Vacuum-Assisted System

The integration of the passive vacuum system led to significant performance improvements across all tested variables. By lowering the internal pressure, the boiling point of water was reduced, enabling earlier and more efficient evaporation. The system exhibited:

- Faster thermal response in morning hours.
- Enhanced sensitivity to moderate solar radiation.
- Improved condensation due to reduced vapor saturation temperature.
- Stable operation under varying ambient conditions.

In summary, the vacuum-assisted solar still consistently outperformed the conventional design, especially during early hours of the day and under moderate solar and wind conditions. These findings validate the effectiveness of low-cost vacuum enhancement in boosting solar distillation efficiency for application in arid and off-grid regions.

4. Conclusion

This study evaluated the performance of a solar still enhanced with a passive vacuum-assisted system, comparing it to conventional atmospheric operation. Experiments under varying conditions—basin water depth, ambient temperature, wind speed, and time of day—demonstrated that vacuum integration significantly increased freshwater productivity by lowering the boiling point and accelerating evaporation. These improvements were especially evident during early hours and with deeper water levels. While higher ambient temperatures enhanced output, increased wind speeds reduced condensation efficiency, particularly under vacuum conditions.

In addition to confirming the technical feasibility and robustness of the system, the study recommends further improvements: optimizing vacuum generation using solar-powered components, maintaining optimal shallow water depths, utilizing corrosion-resistant and

thermally efficient materials, and adopting low-cost, modular designs for scalability. These steps can enhance the practicality, sustainability, and real-world applicability of vacuum-assisted solar stills, particularly in remote or resource-limited regions.

References

[1] UNESCO. (2023). UN World Water Development Report 2023: Partnerships and Cooperation for Water – Executive Summary.

[2] UNESCO. (2023). Download | UN World Water Development Report 2023. Retrieved from https://unesco.org

[3] U.S. Geological Survey (USGS). (n.d.). Where is Earth's Water? Retrieved from https://usgs.gov

[4] Co, J. (n.d.). Desalination Technology: Health and Environmental Impacts. Taylor & Francis.

[5] . (n.d.). Critical Review of Desalination Concentrate Management, Treatment and Beneficial Use. ResearchGate.

[6] Al-Shammiri, M., & Safar, M. (1999). Multi-Effect Distillation Plants – State of the Art. Desalination, 126(1), 45.

[7] Harandi, B., Rahnama, M., Javaran, E. J., & Asadi, A. (n.d.). Performance Optimization of a Multi-Stage Flash Desalination Unit Using Genetic Algorithm. ScienceDirect

[8] Ahmad, M. (2012). Assessment of Freezing Desalination Technologies [Ph.D. thesis

[9] Thimmaraju, M., Sreepada, D., Babu, G. S., Dasari, B. K., Velpula, S. K., & Vallepu, N. (2018). Desalination of Water. IntechOpen. https://doi.org/10.5772/intechopen.78659

[10] An, A. (n.d.). Membrane Desalination: From Nanoscale to Real-World Applications. Taylor & Francis

[11] Danish, S. N., El-Leathy, A., Alata, M., & Al-Ansary, H. (n.d.). Enhancing Solar Still Performance Using Vacuum Pump and Geothermal Energy.

[12] Mohamed, A. S. A., Shahdy, A. G., Mohamed, H. A., & Ahmed, M. S. (n.d.). A Comprehensive Review of Vacuum Solar Still Systems. ScienceDirect.

[13] Performance enhancement of solar thermal systems using phase change materials- a review panelA. Karthikeyan, K.S.S.

Nimay, C.H. Dinesh, J. Jayaprabakar, Ashwin Jacob https://doi.org/10.1016/j.matpr.2023.01.109

[14] Elmghari, N., Rachidi, M. B., Salihi, M., Chebak, A., Al-Dahhan, M. H., & Chhiti, Y. (n.d.). Enhancing Solar Still Productivity Using Phase Change Material. ScienceDirect

[15] Karthikeyan, A., Nimay, K. S. S., Dinesh, C. H., Jayaprabakar, J., & Jacob, A. (2023). Performance Enhancement of Solar Thermal Systems Using Phase Change Materials – A Review. https://doi.org/10.1016/j.matpr.2023.01.109

[16] AbdAllah, A. S., Essa, F. A., & Omara, Z. M. (n.d.). Effect of Different Wick Materials on Solar Still Performance – A Review. Int. J. Ambient Energy, 42(9).

[17] Agrawal, A., & Rana, R. S. (n.d.). Performance of a Single-Basin Solar Still with V-Shaped Floating Wicks. Heliyon

[18] Esfe, M. H., Esfandeh, S., & Toghraie, D. (n.d.). Optimization of Geometrical Parameters of a Solar Still with Thermoelectric Cooling. ScienceDirect.

[19] Jaafar, Z. A., & Hameed, H. G. (2021). Enhancing Evaporation in a Single Slope Solar Still Using Evacuated Pipes. IOP Conf. Ser.: Earth Environ. Sci., 877, 012041

[20] Abdelaziz, G. B., El-Said, E. M. S., Dahaba, M. A., Omara, M. A., & Sharshir, S. W. (2021). Hybrid Solar Desalination Systems Review. https://doi.org/10.1080/15567036.2021.2005721

[21] Shalaby, S. M., Hammad, F. A., & Zayed, M. E. (2023). Current Progress in Integrated Solar Desalination Systems. https://doi.org/10.1016/j.psep.2023.08.058

[22] Abdullah, A. S., Aljaghtham, M., Kandeal, A. W., & Sharshir, S. W. (n.d.). Techno-Economic Study of a Solar-Powered Hybrid HDH/RO Desalination Plant. ScienceDirect

Inference Attacks on Encrypted APIs: Detecting and Mitigating Privacy Leakage from Encrypted Query Patterns

Inference Attacks on Encrypted APIs: Detecting and Mitigating Privacy Leakage from Encrypted Query Patterns

Nuha Omran Abokhadir
Computer Science Department, Faculty of science
University of zawiya, Libya
Abo.khdeir@zu.edu.ly

Hind Ibrahim Omer
Computer Science Department, Faculty of science
University of zawiya, Libya
Libyan Center For Electronic Systems, Programming and Aviation Research, Libya
Hind.ibrahim@sabu.edu

Abstract

Application Programming Interfaces (APIs) are the default integration layer for contemporary cloud services, mobile backends, and microservice systems. Transport Layer Security (TLS) is widely deployed to protect API payload confidentiality and integrity. Yet, TLS does not fully hide externally observable communication structure, including packet sizes, directions, timing, and burst patterns. A passive observer can often exploit these signals to infer high-level user actions or API operations, even when cryptography remains intact. This paper examines the system-level privacy gap created by encrypted-but-observable API traffic patterns. First, it formalizes a passive inference threat model for TLS-protected APIs and frames the core question of whether encryption alone yields practical semantic privacy. Second, it reviews recent leakage-abuse research in encrypted search and relates it to encrypted API inference. Third, it reports an

experimental demonstration showing that common API operation classes can be distinguished using traffic metadata features and standard classifiers. Finally, it evaluates mitigation approaches (padding, shaping, and protocol improvements) and outlines open research directions toward measurable, deployment-feasible privacy protection for encrypted APIs.

Keywords: Encrypted Traffic Analysis - Inference Attacks - TLS 1.3- Encrypted APIs - Metadata Leakage - Traffic Padding - Privacy Engineering.

1. Introduction

API-driven communication has become central to modern computing: cloud platforms rely on APIs for orchestration; mobile applications depend on APIs for identity, content, and telemetry; and microservice architectures use APIs for continuous, fine-grained service-to-service interaction. In practice, TLS is the dominant mechanism for securing these communications, with TLS 1.3 now standardized and widely deployed [1]. TLS is effective at protecting payload confidentiality and integrity against passive eavesdroppers. However, TLS does not eliminate metadata leakage, because the encrypted channel still exposes properties such as packet length sequences, directionality, flow duration, inter-arrival times, and burst structure.

Over the last decade, encrypted traffic analysis has matured into a well-defined research area. Earlier surveys documented that encrypted flows remain classifiable despite cryptographic protection, largely due to stable statistical patterns in traffic traces [2]. More recent work focuses explicitly on TLS 1.3 and demonstrates that even as handshakes conceal more information, inference remains feasible using learned representations derived from observable flow dynamics [3]. This is not merely a theoretical issue: the same core principles

behind website fingerprinting and encrypted traffic classification apply to API systems, where repeated endpoints and predictable response behaviors may produce operation-specific fingerprints.
This paper focuses on a specific problem within this broader landscape: detecting and mitigating privacy leakage from encrypted API query patterns. The paper makes two contributions:

- A structured synthesis connecting leakage-abuse results (particularly in encrypted search) to encrypted API inference.
- An experimental demonstration showing that operation classes over TLS can be classified from metadata features with high accuracy under a controlled setup, alongside a discussion of practical limits and mitigation trade-offs.

2. Threat Model and Problem Definition

DDoS The field of bioacoustics has long explored animal communication through sound, but its application to environmental conservation is a more recent innovation. Marine mammals produce a variety of vocalizations that reflect behaviors such as feeding, mating, social bonding, and predator avoidance. Analyzing these sounds allows scientists to infer ecological patterns and population dynamics. Given the increasing pressures on marine ecosystems, including climate change and human interference, PAM offers an efficient and scalable solution for monitoring biodiversity. It is cost effective compared to visual surveys, and capable of collecting data over long durations and wide geographic areas. Species such as whales, dolphins, and seals are especially well suited for acoustic studies due to their reliance on vocal cues for essential life functions [3].

2.1 Passive Adversary Capabilities

We consider a **passive network adversary** who observes TLS-protected API flows between a client and server. The adversary:

- does **not** decrypt payloads and has **no** access to cryptographic keys.
- can capture **packet sizes**, **traffic direction**, **timing**, and **burst-level structure**.
- may know (or estimate) a finite set of candidate API operations/endpoints typical of the target application domain.

This threat model aligns with standard assumptions in encrypted traffic analysis and website fingerprinting, where the attacker relies on side-channel structure rather than cryptographic breaks [2], [4].

2.2 Adversary Objective

The adversary seeks to infer which API operation is being executed, such as:

- read/query operations (often yielding larger server responses).
- update operations (typically moderate payloads and acknowledgements).
- delete operations (often small responses but distinctive request formats).

Even coarse classification can violate user privacy: “what action was performed” or “which endpoint class was accessed” may be sensitive even when payload content is hidden.

2.3 Core Security Question

This leads to the central question:

Does TLS provide practical semantic privacy for API-based systems when observable metadata leakage remains learnable?

TLS 1.3 provides strong payload confidentiality by design [1]. Yet, if the system leaks stable patterns that allow reliable classification, then privacy is incomplete at the system level. Therefore, the problem sits at the intersection of:

- applied cryptography (security definitions vs. real deployments).
- traffic analysis / machine learning inference.
- privacy engineering (mitigation with acceptable cost).

3. Background and Related Work

3.1 Encrypted Traffic Analysis and TLS 1.3

Encrypted traffic analysis has long shown that application categories and destinations can be inferred from encrypted traces under realistic observation models [2]. TLS 1.3 reduces exposed handshake data relative to TLS 1.2, but it does not eliminate the primary signals used in many inference pipelines—especially packet-length/timing sequences and burst structure [3]. A TLS 1.3-focused survey highlights that recent research increasingly uses deep learning and sequence modeling to exploit these residual patterns [3].

3.2 Website Fingerprinting as a Closely Related Problem Class

Website fingerprinting (WF) offers a strong analogue: attackers infer visited websites based only on encrypted trace patterns. Deep learning has significantly strengthened WF attacks, including against certain lightweight defenses [4]. On the defense side, adaptive padding techniques such as WTF-PAD reduce attack accuracy but introduce overhead and may remain vulnerable to improved classifiers [5]. More recent defenses aim for better practicality and robustness (e.g., RegulaTor) by shaping traffic with moderate overhead [6]. These

results transfer conceptually to APIs: defenses must manage a cost triangle among privacy, latency, and bandwidth.

3.3 QUIC/HTTP3 And Modern Encrypted Transports

API traffic increasingly runs over HTTP/3, which uses QUIC. QUIC provides strong encryption and performance features, but research indicates that fingerprinting remains feasible on QUIC traffic as well—suggesting that transport modernization does not inherently solve inference from metadata [7], [8]. This matters because API ecosystems are rapidly adopting QUIC, and threat models must remain valid across TCP/TLS and QUIC stacks.

3.4 Leakage-abuse And Encrypted Search

Encrypted search and searchable encryption research demonstrate that even when cryptographic primitives satisfy formal security goals, **leakage profiles** (access patterns, query repetition, response sizes) can enable inference. Damie, Hahn, and Peter show accurate query-recovery attacks against searchable encryption using non-indexed documents, illustrating how leakage enables semantic reconstruction without decrypting content [9]. Dijkslag and collaborators extend passive query-recovery methods to conjunctive keyword search schemes, confirming that stronger primitives can still leak under passive observation [10]. Dittert, Schneider, and Treiber analyze sampled-data leakage attacks, showing that partial observation may still be sufficient—an important point for real networks where monitoring may be incomplete [11].

Relevance to encrypted APIs: although APIs are not "encrypted search" systems, they share the key vulnerability: repeated, structured interactions create identifiable signatures. The lesson is consistent across domains: encryption can preserve confidentiality while system behavior leaks intent.

3.5 Metadata Reduction in The Handshake: ECH And Deployment Reality

Server Name Indication (SNI) historically enabled observers to learn the destination domain even with TLS encryption. Encrypted ClientHello (ECH) is designed to reduce such handshake metadata leakage by encrypting sensitive ClientHello fields [12]. Practical deployment relies on DNS mechanisms that distribute ECH configuration (e.g., via SVCB/HTTPS records), standardized in recent RFC work [13]. However, even if handshake metadata is reduced, API inference can still occur via packet-length/timing patterns unless traffic shaping is also addressed.

4. Comparative Review of Representative Research

To connect related lines of evidence, Table 1 contrasts three leakage-abuse works that motivate system-level leakage analysis.

Table 1: Comparison of Reviewed Leakage-abuse Studies

Aspect	Damie et al. [9]	Dijkslag et al. [10]	Dittert et al. [11]
Target	Searchable encryption (single keyword)	Conjunctive keyword search	Sampled-data leakage evaluation
Attack style	Query recovery via leakage statistics	Passive query recovery extension	Leakage success under partial observation
Key insight	Non-indexed auxiliary data	Stronger primitives still	Full observation

	can suffice	leak	not required
Implication for APIs	Repetition + structure enables inference	Added complexity ≠ immunity	Partial monitoring can still succeed

The consistent takeaway is that inference does not require decryption; it requires stable structure. For encrypted APIs, structure emerges from deterministic endpoint behavior (response sizes, serialization patterns, retries, and batching). The third study is especially relevant operationally: if partial trace capture still yields inference, then "limited visibility" is not a reliable privacy argument in real deployments.

5. Experimental Demonstration: Classifying Encrypted API Operations

5.1 Experimental Setup

To illustrate feasibility, we consider TLS-protected REST-style API traffic and extract features from each flow that a passive observer can measure. Features include:

- mean packet size and variance.
- inter-arrival time statistics (mean/variance).
- burst length descriptors (number of packets per burst, burst duration).
- request-to-response byte ratio, and flow duration.

We evaluate three classical classifiers: Random Forest, Support Vector Machine (SVM), and Multilayer Perceptron (MLP). The

objective is multi-class classification of API operation categories (e.g., Search/Query, Update, Delete).

5.2 Classification Performance

Table 2: Classifier performance (illustrative results)

Classifier	Accuracy	Precision	Recall	F1-score
Random Forest	0.964	0.965	0.964	0.964
SVM	0.971	0.971	0.971	0.971
MLP	0.966	0.966	0.966	0.966

Table 2 analysis: The results indicate that operation classes remain distinguishable from metadata alone, with accuracy around 96–97%. This supports the core claim: TLS protects content but not intent. The small gap among models suggests that the signal is strong enough that multiple learning paradigms can exploit it. In realistic deployments, absolute accuracy may vary (due to caching, CDN effects, concurrent flows, and heterogeneity), but the presence of a learnable signal is the central risk.

5.3 Confusion-Matrix Behavior

A diagonal-dominant confusion matrix indicates stable operation-specific signatures. Typical misclassifications occur between operations with similar response sizes and temporal profiles (e.g., Update vs. Delete in some APIs). Near-perfect classification of Search/Query is expected when server responses are systematically larger or more bursty, creating a strong request–response ratio signal.

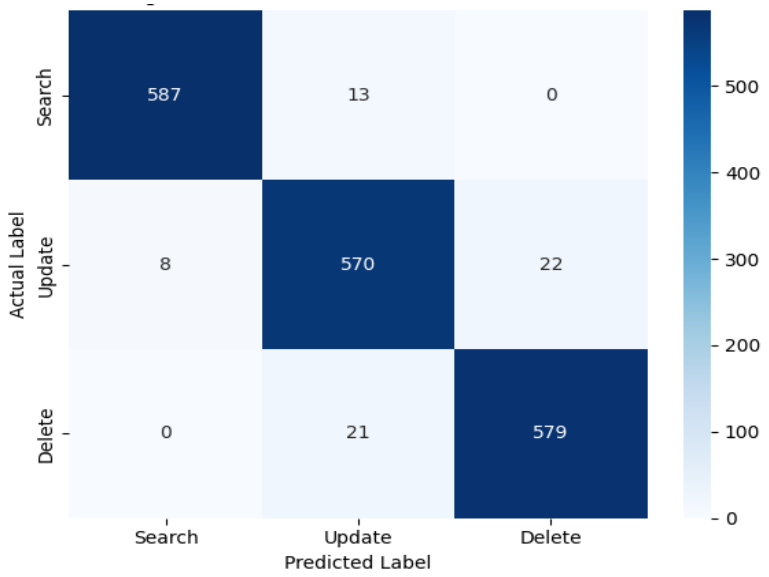

Figure 1: Confusion Matrix (Random Forest)

5.4 Feature Leakage Interpretation

Feature importance commonly elevates (i) mean packet size and (ii) request–response byte ratio, because these capture semantic workload differences—queries often retrieve more data than state changes, and response fragmentation can create distinctive burst patterns. Timing features typically contribute additional separability, particularly when server-side processing time differs across endpoints. This aligns with the broader encrypted traffic analysis literature emphasizing size/timing sequences as core discriminators [2], [3].

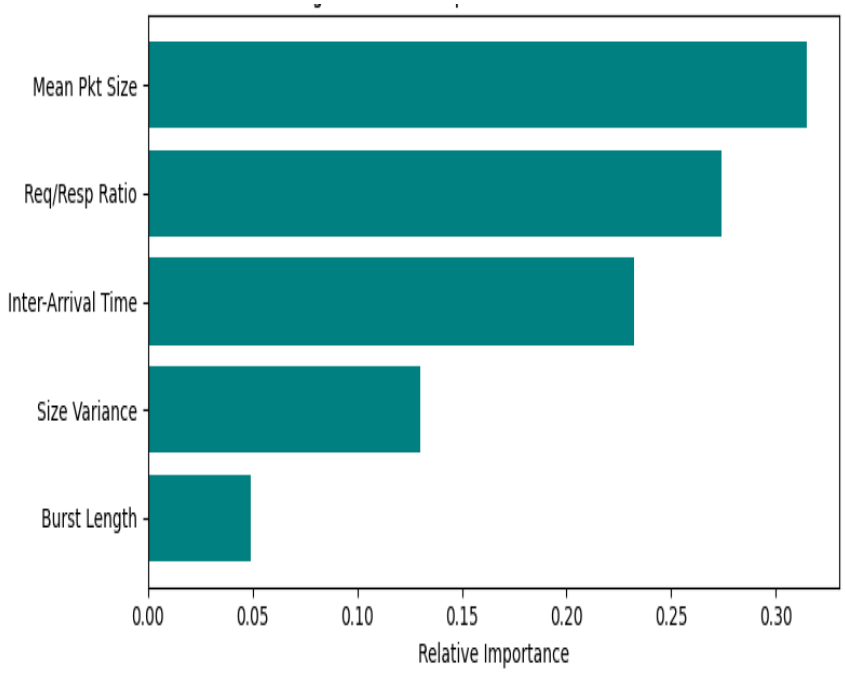

Figure 2: Feature Importance Results

5.5 Padding Noise And Privacy–efficiency Trade-off

As padding increases, classification accuracy tends to drop because packet-length signals become less informative. However, meaningful inference can persist unless padding is aggressive. This illustrates a deployment-relevant trade-off: strong padding may reduce inference but can inflate bandwidth/latency and degrade service performance. This tension mirrors findings in website fingerprinting defenses, where overhead must be balanced against achievable privacy [5], [6].

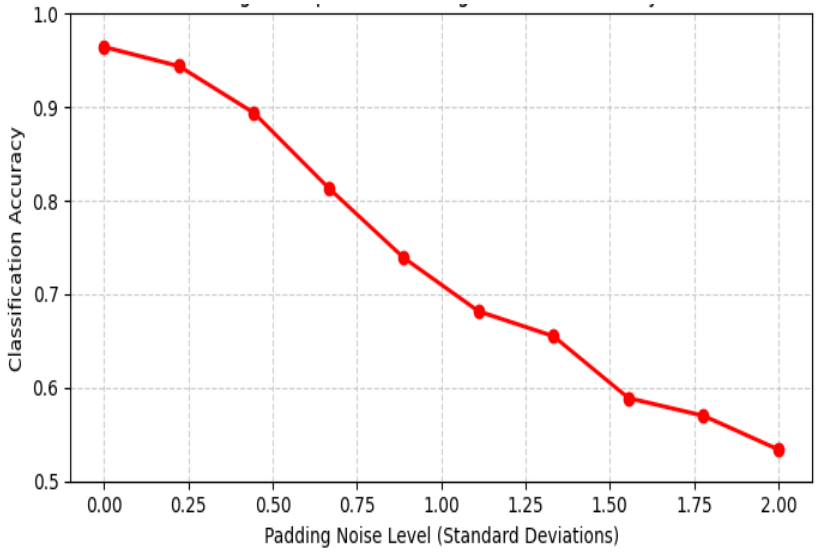

Figure 3: Accuracy vs. Traffic Padding Intensity

6. Mitigation Strategies for Encrypted API Inference

Mitigation must target the observable signals, not only the cryptographic layer.

6.1 Traffic Shaping and Padding

Adaptive padding and shaping can reduce leakage by normalizing packet sizes, burst lengths, or timing distributions. Prior work on WF defenses demonstrates that padding can meaningfully reduce inference accuracy, but overhead and residual vulnerability remain concerns [5]. Practical API defenses may:

- pad responses to coarse size buckets.
- shape burst patterns for sensitive endpoints only, and/or
- introduce timing jitter where latency tolerance exists.

6.2 Endpoint-level Design to Reduce Determinism

API design choices can reduce finger printability:

- standardizing response envelopes across endpoints.
- using fixed-size pagination chunks consistently.
- avoiding endpoint-specific "large vs. small" response extremes.
- batching operations to reduce operation-specific trace signatures.

These measures do not eliminate leakage but can reduce separability in the feature space.

6.3 Protocol-level Improvements: Reducing Handshake Metadata

ECH aims to remove sensitive TLS handshake metadata such as SNI, strengthening privacy against passive observers [12]. Its deployment depends on standardized DNS bootstrapping mechanisms for ECH configuration [13]. While valuable, ECH should be viewed as **complementary**: it reduces destination disclosure but does not fully address flow-structure inference.

6.4 Evaluation As a First-class Engineering Requirement

Because leakage is system-dependent, organizations should treat "traffic fingerprinting risk" as an evaluable property:

- define threat models and attacker capabilities,
- test closed-world and open-world inference settings,
- measure privacy gain vs. overhead for each mitigation, and
- document residual risks.

7. Limitations and Open Research Directions

7.1 Generalization and Operational Realism

Controlled experiments may overestimate attack accuracy. Real deployments include noisy background traffic, concurrent sessions, retransmissions, caching layers, CDNs, and heterogeneous client devices. A key research direction is robust evaluation under realistic operational noise, including open-world settings and varying attacker knowledge.

7.2 QUIC Migration And Cross-transport Consistency

As APIs shift to QUIC/HTTP3, inference and defenses must be validated across transport stacks. Existing evidence suggests inference remains feasible on QUIC [7], motivating transport-agnostic mitigation frameworks.

7.3 Toward Measurable "Semantic Privacy" For APIs

A practical research goal is defining and measuring semantic privacy for APIs: what leakage is acceptable, under which adversaries, and at what cost. This likely requires bridging formal notions from leakage-abuse cryptography with empirical ML-based inference evaluation [9,11].

8. Conclusion

Encrypted APIs are not automatically private at the system level. TLS 1.3 provides strong payload confidentiality, but externally observable metadata—packet lengths, timing, and burst structure—can leak enough information for a passive observer to infer API operations. By synthesizing recent leakage-abuse research and presenting an experimental demonstration of operation classification

from metadata, this paper reinforces a consistent conclusion: **encryption hides content, not intent**. Meaningful mitigation requires a layered approach combining protocol improvements (e.g., ECH), traffic shaping/padding, endpoint design choices that reduce determinism, and systematic leakage evaluation in realistic deployment settings.

References

[1] E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.3," RFC 8446, Aug. 2018.

[2] P. Velan, M. Čermák, P. Čeleda, and M. Drašar, "A survey of methods for encrypted traffic classification and analysis," Int. J. Network Management, vol. 25, no. 5, pp. 355–374, 2015.

[3] J. Zhou, W. Fu, W. Hu, and Z. Zhang, "Challenges and Advances in Analyzing TLS 1.3-Encrypted Traffic: A Comprehensive Survey," *Electronics*, vol. 13, no. 20, 2024.

[4] P. Sirinam, M. Imani, M. Juárez, and M. Wright, "Deep Fingerprinting: Undermining Website Fingerprinting Defenses with Deep Learning," in *Proc. ACM CCS*, 2018.

[5] M. Juárez, M. Imani, M. Perry, C. Diaz, and M. Wright, "Toward an Efficient Website Fingerprinting Defense," in *Proc. ESORICS*, 2016 (WTF-PAD / adaptive padding).

[6] J. K. Holland and N. Hopper, "RegulaTor: A Straightforward Website Fingerprinting Defense," *PoPETS*, 2022.

[7] J. Ha and H. Roh, "QUIC website fingerprinting based on automated machine learning," *ICT Express*, vol. 10, no. 3, pp. 594–599, 2024.

[8] J. Iyengar and M. Thomson, "QUIC: A UDP-Based Multiplexed and Secure Transport," RFC 9000, May 2021.

[9] M. Damie, F. Hahn, and A. Peter, "A Highly Accurate Query-Recovery Attack against Searchable Encryption using Non-Indexed Documents," in *Proc. 30th USENIX Security Symposium*, 2021, pp. 143–160.

[10] M. Dijkslag, M. van Dijk, W. Jonker, A. Peter, and F. Hahn, "Passive Query-Recovery Attacks against Secure Conjunctive Keyword Search Schemes," in *Applied Cryptography and Network Security (ACNS)*, Springer, 2022, pp. 126–146.

[11] D. Dittert, T. Schneider, and A. Treiber, "Too Close for Comfort? Measuring Success of Sampled-Data Leakage Attacks Against Encrypted Search," in *CCSW@CCS*, 2023.

[12] IETF TLS WG, "TLS Encrypted Client Hello (ECH)," *Internet-Draft draft-ietf-tls-esni*, work in progress.

[13] B. Schwartz, M. Bishop, and E. Nygren, "Bootstrapping TLS Encrypted ClientHello with DNS Service Bindings," RFC 9848, Dec. 2025.

Comparative Study of Two Neural Network Controllers for Temperature Control in a Continuous Stirred Tank Reactor (CSTR)

7

Comparative Study of Two Neural Network Controllers for Temperature Control in a Continuous Stirred Tank Reactor (CSTR)

Ahmed A.Albkkosh
Department of Electrical and Electronic Engineering Al-EradaAl-Oula
Higher Insitute for Oil Professions and Applied Sciences, Libya
Email: albkkosh83@gmail.com

Abstract

This paper presents an analytical study on the application of artificial intelligent control devices specifically, multilayer perceptron neural networks (MLPNNs) for regulating temperature and concentration in a continuous stirred tank reactor (CSTR). Two neural networks were designed: one with three inputs and the other with five inputs, both trained using the backpropagation gradient descent algorithm. The study aims to compare the performance of these networks in predicting unseen data and to assess their capabilities for effective control of the reactor's nonlinear dynamic behavior. Results demonstrate that both networks successfully learned the reactor's dynamics and provided accurate predictions, underscoring the potential of artificial intelligence to enhance the performance of industrial control systems.

Keywords: Artificial Neural Network, Continues Stirred Tank Reactor(CSTR), Backpropagation(BP)..

1. Introduction

Controlling industrial processes, particularly in chemical reactors, poses a significant challenge due to their dynamic, nonlinear, and complex nature. Continuous stirred-tank reactors (CSTRs) are among the most critical operational units in the chemical industries, requiring precise control systems to maintain product quality and process stability[11]. In recent years, artificial intelligence techniques have emerged as a promising alternative to traditional control systems, owing to their capability to handle complex and nonlinear systems without the need for accurate mathematical models . Among artificial intelligence techniques, artificial neural networks (ANNs) have demonstrated their efficacy across various domains, including the control of industrial processes. Neural networks are based on simulating the structure and function of the human brain, and they are characterized by their ability to learn from data and extract complex relationships between inputs and outputs. This study aims to investigate the feasibility of applying multilayer perceptron neural networks (MLPNNs) for controlling temperature and concentration in a CSTR reactor. Two neural networks with differing numbers of inputs will be designed and trained, and their performances will be compared to identify the optimal design that achieves the best control performance. The significance of this study lies in providing a comparative perspective on the influence of neural network architecture on its effectiveness in controlling a real industrial system, thereby contributing to the development of more efficient and reliable intelligent control systems.[13]

2. Literature Review

The field of control in chemical reactors using artificial intelligence techniques has witnessed remarkable development, as numerous studies have sought to enhance the performance of traditional control

systems. This paper focuses on reviewing previous studies related to the application of artificial neural networks (ANN) and other intellige nt control systems in chemical reactors, with a particular emphasis on continuous stirred-tank reactors (CSTR). In 2004, Gaye Ö. Yücel Çakal investigated the dynamicic behavior of suspended continuous stirred-tank renactors in boric acid production, where the study focused on the residence time distribution for solid and liquid components and their impact on colemanite conversion and gypsum crystal growth. The results demonstrated that the reactors employed in the experiments approximated the ideal continuous stirred-tank reactor [1]. Subsequently, in 2007, Stanislav Kuník et al. presented an analog and virtual model for an exothermic continuous stirred reactor, comparing the behavior of the two models in terms of stability and dynamic responses, which contributed to validating the analog model [2]. Research evolved to incorporate artificial intelligence applications, where Manuela Souza Leite et al. in 2008 applied two artificial intelligence techniques—namely, artificial neural networks and neuro-fuzzy systems—to predict the temperature in the styrene polymerization process. These models proved highly effective in the dynamic modeling of this nonlinear process, demonstrating their capacity to provide accurate long-term predictions [3]. In 2009, Derar Issam Abd Alkareem implemented various control strategies, including traditional feedback control and neural network-based control (NARMA, NN Predictive), for the neutralization process in a CSTR reactor. The results indicated that the artificial neural network outperformed traditional methods in controlling the neutralization process, owing to its lower mean squared error [4]. Comparative studi es continued, as in 2011, Afraa Hilal Kamel Al-Tae compared di fferent control strategies (traditional control, fuzzy logic control, NARMA neural network control, and fuzzy PID control) for regul ating the temperature of cold water exiting a heat exchanger. The intelligent controllers exhibited significantly superior responses compared to traditional controllers [5]. In 2012, Stutee Bhoi Roll

focused on studying microchannel reactors using computational fluid dynamics (CFD) analysis, illustrating how these reactors can integrate heat transfer, fluid mixing, and chemical reaction into a single unit, thereby accelerating reaction rates and reducing residence time [6]. In 2014, Fayez Areed et al. conducted a comparative study of chemical reactor systems using traditional P, PI, and PID controllers, as well as a genetic algorithm. The results confirmed the feasibility of employing genetic algorithms for the identification and control of nonlinear systems, particularly in CSTR reactors [7]. In 2015, P. Poongodi and R. Madhu Sudhanan compared the performance of PID controllers, fuzzy PID, model predictive controller (MPC), and MPC-based PID controller for temperature control in a CSTR reactor. The MPC-based PID controller demonstrated superior performance in temperature regulation [8]. in 2016, Normah Abdullah et al. proposed a NARMA-L2 neural controller design for controlling the concentration in an i sothermal CSTR reactor. The NARMA-L2 neural controller exhibited better accuracy and response compared to the traditional PID co ntroller, along with its capability to address the nonlinear aspects of t he reactor [9]. Kaur and Kaur (2012) conducted a comparative study of different control architectures for reactor systems. Their research evaluated the performance of various controllers in maintaining stability and handling nonlinearities, providing a benchmark for assessing intelligent control strategies in chemical processes.[10].This research thesis distinguishes itself from prior studies by emphasizing the design of two multilayer perceptron (MLP) neural networks—one with three inputs and the other with five inputs—and training them on the same dataset using the identical training algorithm to control the c oncentration and temperature of a chemical reactor. The networks are then compared based on their ability to predict novel values absent from the training data, representing a form of comparison and training not featured in all the aforementioned studies.

3. Research Methodology

The research methodology in this study relies on the design and im plementation of multilayer perceptron neural networks (MLPNNs) for controlling temperature and concentration in a continuous stirred-tank reactor (CSTR). The CSTR reactor was selected due to its nonlinear dynamic characteristics and its significance in the chemical industries. The methodology encompasses the following steps:

3.1 Mathematical Model Of The CSTR

The mathematical model is developed from material balances. The Schematic diagram of CSTR reactor is shown in Figure 1.The mathematical model of the reactor comes from energy balance. An exothermic reaction $A \rightarrow B$ takes place in the reactor, which is in turn cooled by a coolant that flows through a jacket around the reactor. The jacket is assumed to be perfectly mixed. Heat transfer takes place through the reactor wall into jacket. The main objective is to maintain the temperature of the reacting mixture at desired concentration. The manipulated variable is the coolant temperature.

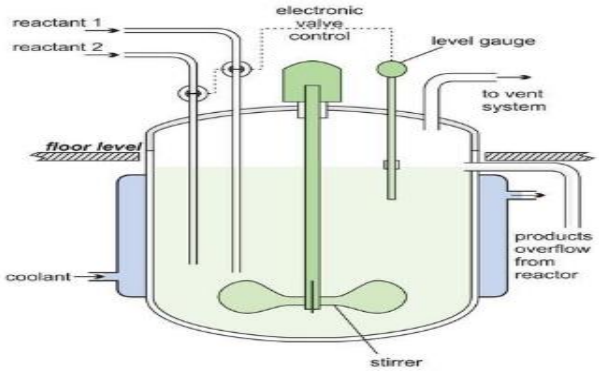

Figure 1: Schematic diagram of CSTR

The CSTR has three input signals: C_{Ai} Concentration of feed stream, T_i Inlet feed stream temperature, T_C Jacket coolant temperature. The two output signals: C_A Concentration of A in reactor tank, T Reactor Temperature.

The mathematical model for this process is formulated by carrying out mass and energy balances, and introducing appropriate constitutive equations.[14,15]

a- Overall material (mass) balance

$$\rho \frac{dV}{dt} = F_i \rho - F\rho \tag{1}$$

$$\frac{dC_A}{dt} = \frac{F_i}{V}(C_{Ai} - C_A) - K_0 C_A \, e^{-E/RT} \tag{2}$$

b- Energy balance equation

$$\frac{dT}{dt} = \frac{F_i}{V}(T_i - T) - \frac{(\Delta H)\left(K_0 C_A \, e^{-E/RT}\right)}{\rho C_P} + \frac{U_A}{V\rho C_P}(T_C - T) \tag{3}$$

The nominal values of the CSTR parameters used in this paper are as listed in the Table 1 [17].

Table 1: Nominal Values Of CSTR

No	Parameters	Symbols	Values
1	Product concentration	C_A	0.989
2	Reactor temperature	T	296.6
3	Coolant flow rate	F_C	100
4	Feed flow rate	F	100
5	Feed concentration	C_{A_i}	1
6	Feed temperature	T_i	350
7	Feed coolant temperature	T_{C_i}	270
8	CSTR volume	V	100
9	Heat transfer term	U_A	14500
10	Reaction rate constant	K_0	34930800
11	Activation energy term	E/R	5963.6
12	Heat of reaction	ΔH	5721.6
13	Liquid densities	ρ, ρ_C	1000
14	Specific heat	C_P, C_{P_C}	0.48

3.2 Artificial Neural Network Structure And Representation

Over the past few years, Artificial Neural Network (ANN) has received a great deal of attention and is now being proposed as a powerful computational tool. The most common Neural Networks

structure employed is multi layer perception (MLP), because it is able to create a model that correctly maps the input to the output using historical data. This structure consists of a number of highly interconnected processing unit called "neurons" which are interconnected by connection weights w_{jk} which determine the effect for the signal of unit j has on unit k. Each unit typically receives signals from other units or from the external environment (bias, offset) with a propagation rule for each unit, which determines the effective input of a unit from its external inputs. A subgroup of neurons is called a layer in the neural network. The first layer is the input layer and the last layer is the output layer. The layers that are placed between the input and the output layer are called hidden layers. A neuron in a Neural Network receives input stimuli and translates it with current output by using activation function which introduces linear or non-linear for the network into a new single output response, which is then transmitted along the processing unit output connections. A single neuron (node i) in a MLP can be shown in figure 2.

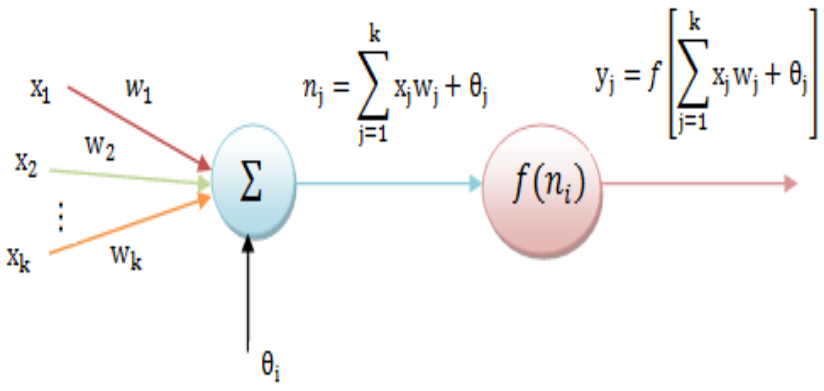

Figure 2: Single node in a MLP network

The output from the single node is computed by:

$$y(k) = f[\sum_{i=1}^{N} x_i(k) * w_i(k) - \theta] \qquad (4)$$

Where:y(k) = Output signal, f = Activation function, w_i= Connection weight, x_i = Input signal, θ = Bias input.

Connecting several nodes in parallel and series, a multilayer perceptron (MLP) network is formed. A typical network is shown in Figure 3.

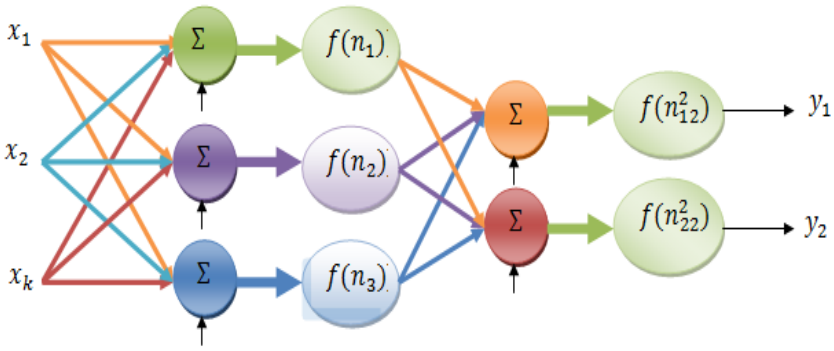

Figure 3: a multilayer perceptron network with one hidden layer.

The output of the MLP network becomes:

$$y_i = f\left(\sum_{j=1}^{3} w_{j1}^2 f\left(\sum_{j=1}^{k} w_{jk}^1 + \theta_{j1}^1\right) + \theta_{j2}^2\right) \qquad (5)$$

Training is the process of adjusting the weights. The training is intended to steadily adjust the connection weight in order to minimize the mean square error between target value output and predicted value output. The network is trained by repeatedly giving it examples from

its training set. Each component of the training set consists of a pattern in the form of an input vector together with its corresponding answer. Each example from the training set is offered to the network and its output calculated. In this type of learning algorithm, there are two moves before the weights are updated:

1- First move (forward move), the outputs of all neurons are calculated by multiplying the input vector by the weights and the error is calculated for each of the output layer neuron.
2- Second move (backward move), the error is passed through the network layer and the weights are updated, so that the actual output of the MLP moves closer to the desire output. The difference between the target output and the computed output is known as error, this error is computed as:

$$e(k) = y_d(k) - y_a(k) \qquad (6)$$

where e(k) is the error, y_d(k) desired output, y_a(k) actual output. The errors are back propagated through the layers:

$$\delta_o = e(t) * {}^{\partial}/_{\partial t}\, purelin(y_a) \qquad (7)$$

$$\delta_1 = w_{41} * \delta_o * {}^{\partial}/_{\partial t}\, sig(y_1) \qquad (8)$$

where: δ_o is the error gradient of the output neuron, and δ_1 is the error gradient of the neuron one, and $w_{41}$is the weight between neuron one and the neuron of the output, and y_1 is the output of neuron one. In the same manner the error gradient of all neurons for hidden layer can be found, and figure 4 below describes

graphical description of training multi-layer Neural Network using BP algorithm>[12,16]

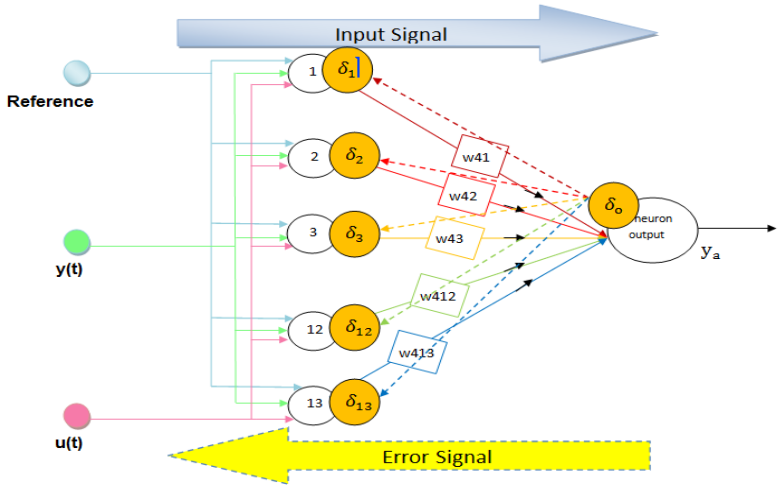

Figure 4: Graphical description of training

The formula for adjusting the weight is:

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t)$$

$$\Delta w_{ij}(t) = \alpha * x_i(t) * \delta_j(t)$$

Where: α is learning rate.

The training is interned to gradually adjust the connection weight in order to minimize the mean square error (MSE):

$$MSE = \sum_{k=1}^{M} \sum_{i=1}^{N} (y_{di}(k) - y_{oi}(k))^2$$

Where:- M: number of training data pattern, N: number of neurons in the output layer, $y_{di}(k)$: the target value of the output neurons, $y_{oi}(k)$: the prediction value for the output neuron.

When Neural networks are trained, a particular input leads to a specific target output. Such a situation is shown figure 5. There, the network is adjusted based on a comparison of the output and the target, until the network output matches the target.

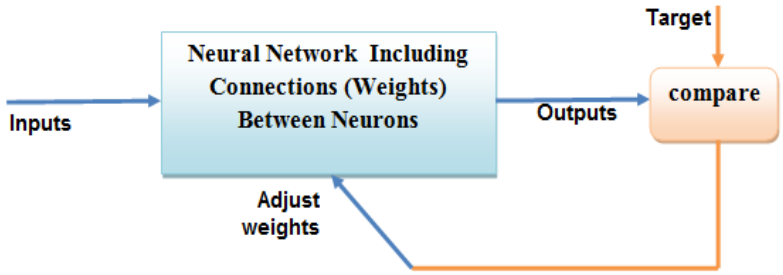

Figure 5: Neural network mechanism

4. Design (MLPNN) Using Gradient Descent Backpropagation Algorithm

Two neural networks of the MLPNN type were designed using the Gradient Descent Backpropagation Algorithmm to controll the concentration and temperature in a CSTR reactor. Both networks were trainked on the same dataset and using the same training algorithm, but differing in the number of inputs: The first network: Contains three inputs shown in Figure 6. The second network: Contains five inputs shown in Figure 7.[12,13]

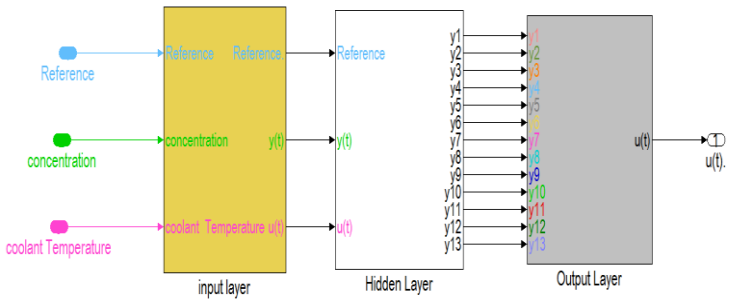

Figure 6: Matlab simulink model (case1).

The networks consist of an input layer, one hidden layers, and an output layer. The sigmoid activation function was employed in the hidden layers, while the pure linear (purelin) function was used in the output layer. Stages of the Backpropagation Algorithm: Feed-forward computation: The inputs are propagated through the network to compute the outputs. Backpropagation to the output layer: The error is calculated at the output layer. Backpropagation to the hidden layer: The error is propagated back to the hidden layers. Weight updates: The network weights are adjusted to minimize the error using the gradient descent rule.

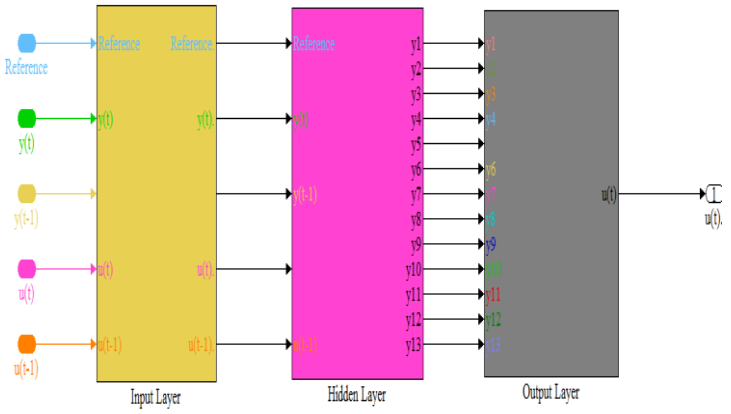

Figure 7: Matlab Simulink model (case2)

5. Simulation results

The control system was simulated using the MATLAB/Simulink software. Figure 8 illustrates the Simulink diagram of the system for case1 and Figure 9 illustrates case2.

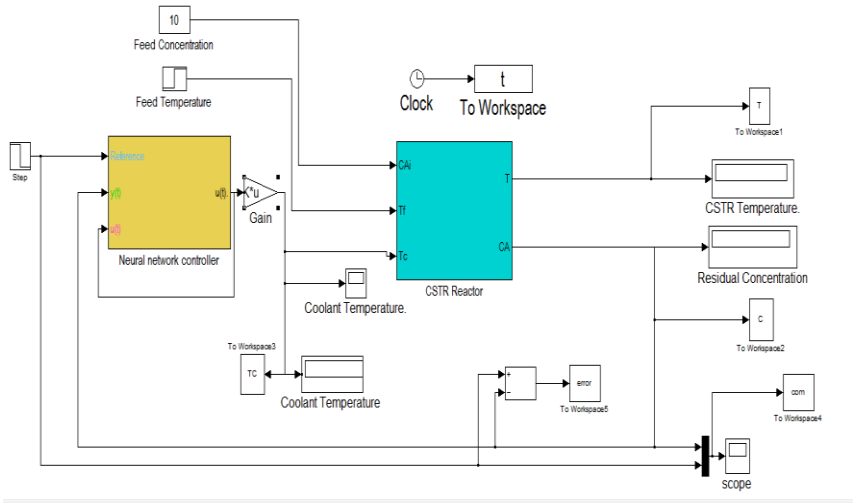

Figure 8: Simulink model diagram(case1)

The training process involved adjusting the weights and biases in the neural network to minimize the discrepancy between the actual outputs and the target outputs. Following training, the performance of the two networks was compared in terms of their ability to predict novel values not present in the training data, thereby providing a comprehensive evaluation of the efficacy of each design.

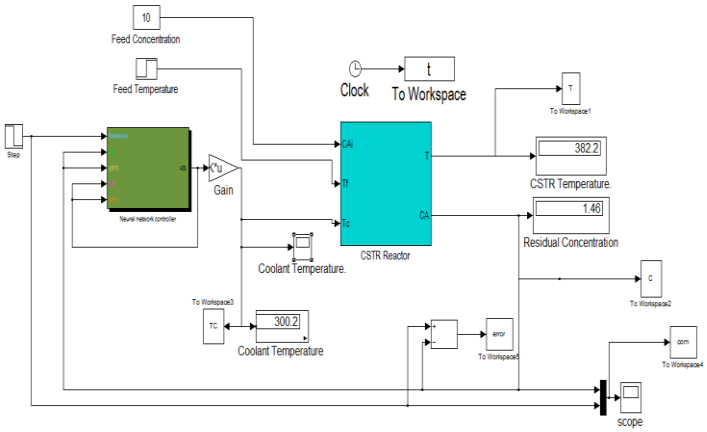

Figure 9: Simulink model diagram(case2)

As for the fundamental difference between the two networks in prediction accuracy, this difference is directly attributable to the number of inputsutilized by each network.

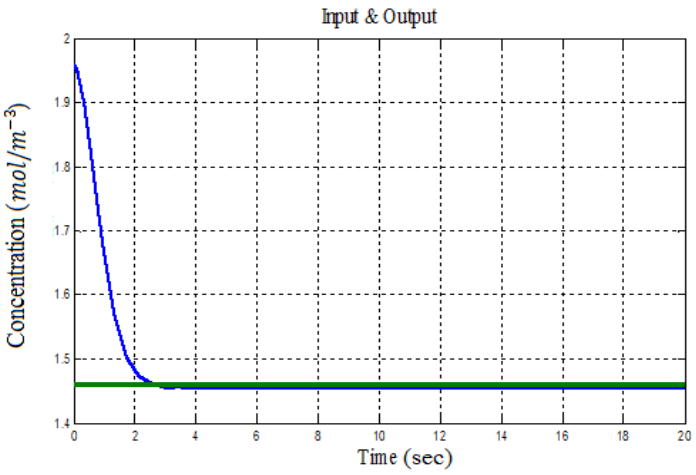

Figure (10): input Vs output (case1)

The First Network (with 3 inputs): This network relies solely on current data (the reference concentration, the current concentration, and the current coolant temperature). It was capable of predicting new values; however, its results exhibited a slight offset error. The figure (10) explain the input(set point) signal and output (concentration) signal on the same graph for case 1.

The Second Network (with 5 inputs): This network incorporates both current and historical (past) data. By adding the values of the variables from the previous time step as inputs, the network gained a form of "short-term memory." This additional "memory" significantly enhanced prediction accuracy. The results demonstrated that this network was more accurate and exhibited a closer match to the reference values, virtually eliminating the offset error observed in the first network. Figure (11)explain the input(set point) signal and output(concentration) signal on the same graph.

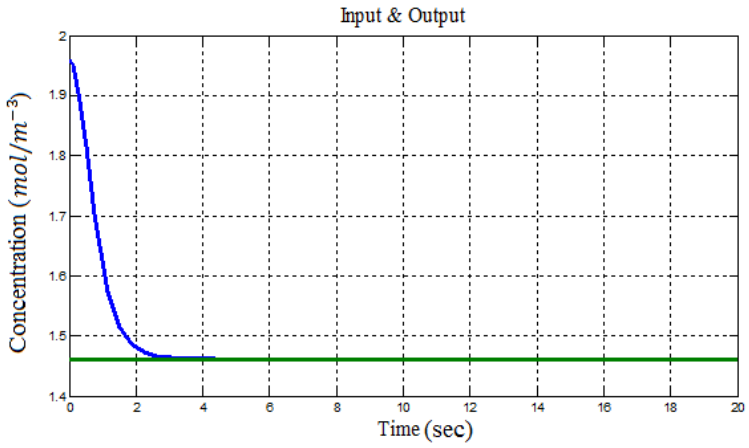

Figure 11: input Vs output (case 2)

Both networks succeeded in the control task; however, the network with five inputs was distinctly superior in predicting new values accurately. This proves that providing a neural network with additional information about the process "context" or "history" (such as past values) enhances its learning and generalization capabilities, resulting in a more intelligent and precise control system.[10,17]

6. Conclusion

This study aimed to address the challenges associated with controlling Continuous Stirred-Tank Reactors (CSTRs) due to their complex nonlinear nature, through the development and application of an intelligent control system based on Artificial Neural Networks (ANNs). Two Multilayer Perceptron (MLP) models were successfully designed and trained—one utilizing instantaneous data (3 inputs) and the other incorporating historical data (5 inputs)—for the precise control of the reactor temperature.

Simulation results conclusively demonstrated the effectiveness of the proposed approach. The ANN-based controller successfully achieved the desired objectives, exhibiting rapid and stable response, along with an accurate ability to reach new setpoints. More importantly, the study revealed a pivotal finding: equipping the neural network with a "memory" by integrating historical values as inputs (the 5-input model) led to a significant improvement in prediction accuracy and eliminated the offset error observed in the simpler model. This conclusion not only confirms the capability of neural networks to model and control dynamic systems but also underscores the importance of input architecture design in enhancing network performance and generalization ability.

Consequently, neural networks can be considered a powerful and reliable tool capable of replacing or supplementing traditional control

systems in industrial processes that demand high precision and responsiveness.

References

[1] Yücel Çakal, G. Ö. (2005). Dynamic Behavior of Continuous Flow Stirred Slurry Reactors in Boric Acid Production [Ph.D. dissertation]. Middle East Technical University.

[2] Kuník, S., Mudrončík, D., & Kopček, M. (2007). Virtual and analogue model of a continuous stirred tank reactor. In Proc. 15th International Conference on Process Control.

[3] Leite, M. S., Silva, F., & Fileti, A. M. F. (2008). Modeling of styrene polymerization in solution using artificial neural networks. Revista Brasileira de Engenharia Química.

[4] Abd Al Kareem, D. I. (2009). Implementation of Neural Control for Neutralization Process [Master's thesis]. University of Technology, Baghdad.

[5] Al-Tae, A. H. K. (2011). Comparative Study of Temperature Control in a Heat Exchanger Process [Master's thesis]. University of Technology, Baghdad.

[6] Bhoi, S. (2012). Study of microchannel reactor using CFD analysis[Master's thesis]. National Institute of Technology, Rourkela.

[7] Areed, F. F., et al. (2014). Genetic Algorithm Optimized Chemical Reactors Network. Journal of Chemical Engineering.

[8] Poongodi, P., & Sudhanan, R. M. (2015). Simulation of temperature control methodologies for chemical reactor. Journal of Chemical and Pharmaceutical Research, 7(9).

[9] Abdullah, N., et al. (2016). Control of Continuous Stirred Tank Reactor using Neural Networks. Indian Journal of Science and Technology, 9(21).

[10] Kaur, G., & Kaur, R. (2012). A Comparative Study of Performance of Different Control Architectures for Reactor System. International Journal of Computer Applications, 41(11).

[11] Zilouchian, A., & Jamshidi, M. (2001). Intelligent Control Systems Using Soft Computing Methodologies. CRC Press LLC.

[12] Mehrotra, K., Mohan, C. K., & Ranka, S. (1996). Elements of Artificial Neural Networks. MIT Press.

[13] Negnevitsky, M. (2005). Artificial Intelligence: A Guide to Intelligent Systems. Pearson Education, 2nd Edition.

[14] Mushtaq, F. (2014). Analysis and Validation of Chemical Reactors performance models. Master's thesis, KTH School.

[15] Vojtesek, J., & Dostal, P. Use of Differential Equations in Modeling and Simulation of CSTR. Tomas Bata University.

[16] Poznyak, A. S., et al. (2001). Differential Neural Networks for Robust Nonlinear Control. World Scientific Publishing.

[17] Albkkosh, A. A. (2017). Study Of Artificial Intelligent Control Devices In A Chemical Reactor Control System [Master's thesis]. Gharian University.

8

MQTT versus CoAP: Performance and Security Evaluation for Real-Time IoT Applications on Raspberry Pi Edge Nodes

MQTT versus CoAP: Performance and Security Evaluation for Real-Time IoT Applications on Raspberry Pi Edge Nodes

Ali Husein Ben Husein
Electronic Technology College Tripoli, Libya
Ali_Hussain_m@yahoo.com

Haidr Swaih
Electronic Technology College Tripoli, Libya
h.swaih@outlook.com

Abstract

Message Queuing Telemetry Transport (MQTT) and the Constrained Application Protocol (CoAP) are two main protocols used in the Internet of Things (IoT). MQTT uses a broker-based publish/subscribe model over TCP. CoAP provides a RESTful interface over UDP and supports Datagram Transport Layer Security (DTLS). The two protocols can be secured using MQTT over TLS and CoAP over DTLS. However, adding security can increase latency and use more resources on limited edge devices.

This paper presents an experimental evaluation of MQTT and CoAP in both secure and insecure setups using a wireless edge-computing testbed built with Raspberry Pi devices. The testbed includes three Raspberry Pi nodes on the same Wi-Fi network. Node 1 acts as a sensor publisher, collecting temperature and humidity from a DHT22 sensor and light levels from a digital LDR module. Node 2 is an actuator that subscribes to the sensor data and controls an LED. Node 3 serves as the edge gateway, running either the MQTT broker or the CoAP/DTLS server, and also collects metrics. The four scenarios tested are insecure MQTT, insecure CoAP, MQTT with TLS, and CoAP with DTLS. The metrics collector records per-message latency,

CPU and memory use on all nodes, and message loss, saving the results in CSV files with a consistent format. Only complete data rows are used in the final analysis. The results show that enabling security increases latency and CPU load for both protocols. On this hardware, secure CoAP (DTLS) causes more latency than MQTT with TLS, while insecure CoAP uses slightly less CPU at the broker than insecure MQTT. The paper discusses the trade-offs between performance and security and gives recommendations for choosing protocols for real-time edge IoT systems.

Keywords: MQTT – CoAP – DTLS – TLS - Internet of Things - Raspberry Pi - edge computing – latency - performance evaluation - security.

1. Introduction

The Internet of Things (IoT) is widely used in applications where low latency is critical, such as smart buildings, industrial monitoring, and safety systems. In these environments, edge gateways collect sensor data, process it locally, and control devices almost right away. Communication protocols for these systems must be fast, efficient, and secure. MQTT and CoAP are two of the most common IoT protocols. MQTT uses a lightweight publish/subscribe model over TCP, with a central broker to route messages between publishers and subscribers [1], [2]. CoAP is a RESTful protocol for devices with limited resources, running over UDP and supporting features such as confirmable messages, multicast, and easy integration with HTTP. MQTT typically uses TLS for security [3], [4], [5].

In contrast, CoAP uses DTLS to secure datagram traffic [3]. Many studies have compared MQTT and CoAP, but most focus on performance without security [6], [7], or examine security with only limited tests on real hardware [8], [9]. There is still little research on how secure MQTT and secure CoAP perform in real-world edge-

computing workloads across different Raspberry Pi models, such as the Raspberry Pi 4 and Raspberry Pi 5, connected via a typical home Wi-Fi access point.

IoT is used in applications where low latency is important, such as smart buildings, industrial monitoring, and safety systems. In these situations, edge gateways collect and process data in real time. The main IoT communication protocols, MQTT and CoAP, need to be fast, efficient, and secure. MQTT uses a publish/subscribe model over TCP, while CoAP uses a RESTful approach over UDP. TLS usually secures MQTT, while DTLS secures CoAP. Most previous studies compare these protocols by focusing on performance rather than security, or test security only on a small scale. There is little research on their security and performance across different Raspberry Pi models in a typical Wi-Fi setup. This study uses a unified metrics collection system to record latency, CPU usage, memory usage, and message loss for each sample in CSV files with a consistent format.

This paper aims to fill this gap by implementing and evaluating the following:

- A three-node edge testbed using Raspberry Pi devices connected to a single Wi-Fi access point.
- Four communication scenarios: insecure MQTT, insecure CoAP, MQTT with TLS, and CoAP with DTLS.
- A unified metrics collection system that records latency, CPU usage, memory usage, and message loss for each sample in CSV files using the same format.

The contributions of this work are:

1. A reproducible edge IoT testbed setup with three Raspberry Pi nodes, along with wiring diagrams and software configurations for MQTT, CoAP, TLS, and CSV schema for collecting and post-processing metrics across all four scenarios, skipping rows with incomplete measurements.

2. A comparison of MQTT and CoAP in both secure and insecure modes, showing latency, CPU and memory usage, and message loss. All advice for choosing and setting up protocols in real-time IoT applications on Raspberry Pi edge nodes.

The rest of the paper is organized as follows: Section 2 reviews the background and related work. Section 3 describes the testbed and scenarios. Section 4 explains the methods and metrics. Section 5 presents the results. Section 6 discusses the limitations. Section 7 concludes.

2. Background and Related Work

2.1 MQTT

Picture devices that need to communicate reliably, even when the network is unreliable. MQTT is well-suited for this situation. It is a lightweight messaging protocol that uses a publish/subscribe model and works well in tough conditions [10]. MQTT is built for TCP/IP networks and stays reliable even with low bandwidth or unstable connections. Clients send messages to certain topics, and other clients subscribe to those topics to get updates [2]. The MQTT broker is at the center of the system. It manages client connections, handles subscriptions, and forwards messages. Three quality-of-service (QoS) levels—0, 1, and 2—let users balance reliability with overhead. For security, MQTT usually uses Transport Layer Security (TLS) with X.509 certificates to authenticate the broker and, if needed, the client [9]. Setting up a TLS connection does add some overhead and uses more CPU, but most MQTT libraries and brokers support this security feature [7], [11].

2.2 CoAP and DTLS

CoAP is a specialized web transfer protocol for constrained nodes and constrained networks, specified in RFC 7252 [4]. CoAP adopts a RESTful model with methods such as GET, POST, PUT, and DELETE, and uses compact binary headers over UDP. Reliability is implemented using confirmable (CON) and non-confirmable (NON) messages with simple acknowledgments and optional retransmissions. Security for CoAP is usually provided by DTLS, a datagram variant of TLS specifically designed for unreliable message delivery and reordering [3],[4]. DTLS adds handshakes, cipher suites, and record protection on top of UDP, which can significantly increase CPU and message overhead on constrained devices [8].

CoAP is a specialized web transfer protocol for constrained nodes and constrained networks, specified in RFC 7252 [4]. CoAP adopts a RESTful model with methods such as GET, POST, PUT, and DELETE, and uses compact binary headers over UDP. Reliability is implemented using confirmable (CON) and non-confirmable (NON) messages with simple acknowledgements, and CoAP is a specialized web transfer protocol for constrained nodes and networks, specified in RFC 7252 [4]. CoAP adopts a RESTful model with methods such as GET, POST, PUT, and DELETE, and uses compact binary headers over UDP. Reliability is implemented using confirmable (CON) and non-confirmable (NON) messages with simple acknowledgments and optional retransmissions. Optional retransmissions. Security for CoAP is usually provided by DTLS, a datagram variant of TLS specifically designed for unreliable message delivery and reordering [3] [12]. DTLS adds handshakes, cipher suites, and record protection on top of UDP, which can significantly increase CPU and message overhead on constrained devices [8], [13].

2.3 Comparison of MQTT and CoAP

Researchers have compared MQTT and CoAP in terms of latency, bandwidth, and resource use. Thangavel et al. [6] found that MQTT typically experiences lower delay at low loss rates, whereas CoAP can reduce overhead in some cases. Martı et al. [7] showed that selecting the appropriate protocol can significantly affect transmission times and resource usage on a Raspberry Pi. Recent work by Seoane et al. [8] evaluated the performance of CoAP and MQTT with security, focusing on CPU usage and bandwidth under DTLS/TLS.
Security-focused analyses, such as those by Laaroussi and Novo [8], highlight the complexity of combining strong security with constrained IoT devices, and tools such as MPInspector [14] aim to systematically evaluate the security of IoT messaging protocol implementations.
Our research stands out from earlier studies in several ways:

- We focus on a three-node edge setup that includes a publisher, a subscriber or actuator, and a broker or server, all connected over Wi-Fi.
- We implement secure MQTT using TLS and secure CoAP using DTLS on both Raspberry Pi 4 and Raspberry Pi 5 devices.
- We use a single CSV-based system to collect metrics and apply the same experimental design to all four scenarios.

3 System Architecture and Scenarios

3.1 Hardware Topology

The experimental testbed has three Raspberry Pi nodes connected to the same Wi-Fi access point. This setup allows for a controlled comparison of protocol performance and security under the same network conditions. Node 1 is the sensor data publisher, Node 2 is the

actuator and subscriber, and Node 3 acts as both the broker/server and metrics collector.

- **Node 1 (Publisher),** a Raspberry Pi 4 Model B, connects to a DHT22 sensor for temperature and humidity, and a digital LDR module for light. It collects this data and sends it to the /sensor/data topic or resource path.
- **Node 2 (Subscriber/Actuator)**, a Raspberry Pi 5 Model B, gets the sensor data by subscribing to the same topic or resource. It uses this data to control a digital LED module, turning the LED on when the light level drops below a set threshold.
- **Node 3 (Broker/Server and Metrics Collector)**, a Raspberry Pi 5 Model B with the static IP address 192.168.0.10. It runs the MQTT broker (Mosquitto) or a CoAP/DTLS server as needed. At the same time, it uses a Python-based metrics collector to save data to CSV files.

All nodes run a recent version of Raspberry Pi OS with the Linux kernel 6.x and are powered by stable supplies to minimize power-related performance variation.

3.2 Sensor and Actuator Wiring

On Node 1 (Raspberry Pi 4), the sensors are wired as follows (BCM numbering):

Table 1: Sensor and Actuator Wiring

Sensor	VCC	GND	GPIO pin	Physical pin
DHT22 module	3.3v	GND	GPIO.4	Pin 7
LDR module	5 v	GND	GPIO.17	Pin 11

The physical connections do not change in any scenario. Any differences you notice are due to protocol or security settings, not to changes in the hardware.

3.3 Logical Architecture

Figure 1 illustrates the logical architecture. Node 1 collects data from the DHT22 and LDR sensors and sends a JSON payload with temperature, humidity, and LDR state. Node 2 receives this data and updates the LED state based on the sensor values. Node 3 can act as the MQTT broker to route messages or as the CoAP server to manage RESTful resources. Node 3 also tracks system metrics, such as timestamps and resource usage.

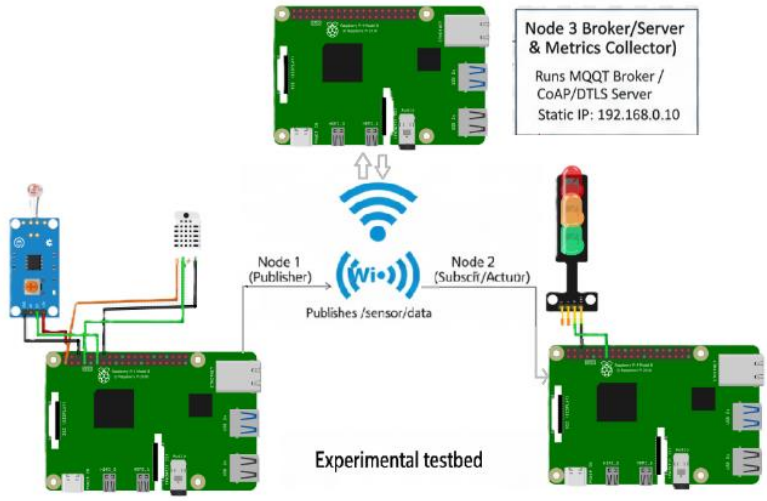

Figure 1: The system uses three Raspberry Pi nodes connected to the same Wi-Fi access point.

3.4 Scenarios

As shown in Table 2, we consider four communication scenarios:

1. **Scenario 1:** Insecure MQTT **(insecure_mqtt).** The Mosquitto broker runs on Node 3 using TCP port 1883 without TLS. Node 1 and Node 2 use plain MQTT with QoS 1. QoS 1 ensures that each message is delivered at least once, balancing reliability and performance. This mode requires an acknowledgement from the receiver, so it is suitable when duplicate messages are preferable to losing messages.
2. **Scenario 2:** Insecure CoAP **(insecure_coap).** The CoAP server on Node 3 is using UDP port 5683 without DTLS. Node 1 and Node 2 are CoAP clients that send sensor data using CoAP POST requests to a resource (e.g., coap://192.168.0.10/sensor).
3. **Scenario 3:** Secure MQTT (secure_mqtt). The Mosquitto broker on Node 3 uses TLS on port 8883, with a local certificate authority and broker certificate. Node 1 and Node 2 connect using MQTT over TLS, with server authentication and optional client certificate authentication.
4. **Scenario 4:** Secure CoAP **(coap_secure).** The CoAP server on Node 3 is set up with DTLS-PSK on port 5684, using the TinyDTLS backend in aiocoap. Node 1 and Node 2 use CoAP over DTLS with pre-shared keys.

Table 2: Evaluated Scenarios Overview

Scenario	Protocol	Transport	Security	Port
S1	MQTT	TCP	None	1883
S2	CoAP	UDP	None	5683
S3	MQTT	TCP	TLS	8883
S4	CoAP	UDP	DTLS-PSK	5684

4 Methodology and Metrics Collection

4.1 Software Stack

Node 3 runs:

- Mosquitto 2.x is used for MQTT scenarios, with separate configuration files for insecure and TLS-secured operation.
- A Python 3.9 virtual environment includes aiocoap and TinyDTLS to support CoAP and DTLS.
- A Python-based metrics collector receives instrumentation messages, such as those sent over HTTP or a local socket, and saves each measurement to a CSV file.

Node 1 and Node 2 run:

- Sensor and actuator scripts are written in Python, using paho-mqtt for MQTT and aiocoap for CoAP.
- DHT22 and GPIO access libraries adapted for Raspberry Pi OS and the Raspberry Pi 4/5 SoC;

  The DTLS functionality is enabled through TinyDTLS integration in aiocoap, and Python 3.9 is used to ensure compatibility with the DTLS backend.

4.2 CSV Schema and Data Cleaning

In each scenario, the metrics collector writes samples into a CSV file with the following schema:

scenario, sample_idx, seq, pub_ts, sub_ts, latency_ms, pub_node_cpu, sub_node_cpu, broker_cpu, pub_mem_mb, sub_mem_mb, total_msgs_received, total_msgs_lost, timestamp_utc

The fields are defined as:

- scenario: scenario identifier (e.g., inscure_mqq, secure_mqtt, inscure_coap, coap_secure);
- sample_idx: a sample index that increases with each entry for the scenario. seq: the sequence number added to each measurement message, which helps detect if messages are out of order or missing.
- pub_ts: the timestamp, in milliseconds since the epoch, recorded when the sensor reading is published or when a CoAP request is sent from the publisher.
- sub_ts: the timestamp, using the same time base, recorded when the subscriber receives and processes the message.
- latency_ms: the calculated one-way latency, ideally measured as sub_ts minus pub_ts, and possibly determined by the metrics collector.
- pub_node_cpu, sub_node_cpu, broker_cpu: the current CPU usage percentage measured on Nodes 1, 2, and 3, respectively.
- pub_mem_mb, sub_mem_mb: the memory usage, in MiB, measured on Node 1 and Node 2.
- total_msgs_received, total_msgs_lost: running totals tracked by the metrics collector for messages received and messages lost.
- timestamp_utc: the ISO 8601 formatted wall-clock time when this row is recorded.

To avoid bias from incomplete samples, we remove any rows with empty cells in the fields used for analysis before processing the data. For example, when analyzing latency and CPU, we only use rows where latency_ms, pub_node_cpu, sub_node_cpu, and broker_cpu are all non-null.

4.3 Experimental Procedure

For each scenario, the following steps are executed:

1. Starting the metrics collector on Node 3 (e.g., a Python script writing inscure_mqq.csv, inscure_coap.csv, secure_mqtt.csv, or coap_secure.csv).
2. Starting the corresponding broker or CoAP/DTLS server on Node 3.
3. Starting the publisher script on Node 1. The publisher reads sensors every 1.00s or 2.00s and publishes a message containing JSON-encoded sensor data with sequence numbers.
4. Starting the subscriber/actuator script on Node 2. The subscriber receives data, updates the LED state based on LDR, and reports timestamps and local metrics to Node 3.
5. Letting the system run 1000 samples to obtain stable statistics.
6. Stopping the scripts and broker/server in a controlled way, ensuring that files are flushed and properly closed.

   After the experiment, Python scripts process the CSV files, skipping incomplete rows and calculating statistics for each scenario, including mean, median, percentiles, CPU and memory averages, and loss rates. We also create time-series plots, CDFs, and bar charts using the cleaned data.

4.4 Metrics

We focus on the following metrics:

- ***One-way latency (ms):*** The time difference between the subscriber and publisher timestamps for each message, assuming any clock differences between nodes are small or have been corrected.
- ***CPU usage (%):*** This shows the current CPU use on each node, checked at regular intervals and matched to the nearest message index.

- ***Memory usage (MiB):*** This is the memory used by the process or system on Nodes 1 and 2.
- ***Message loss (%):*** This is calculated using the total_msgs_lost and total_msgs_received counters, or by comparing the sequence ranges at the publisher and subscriber.

5 Results and Discussion

This section summarizes the main findings. All reported values, including means and medians, are taken directly from the testbed's CSV files. The following tables and figures present these results.

5.1 Latency Distribution

Figure 2 illustrates the empirical CDF of per-message latency for the four scenarios (after removing rows with missing latency values). As expected, security has a noticeable impact on latency for both MQTT and CoAP.

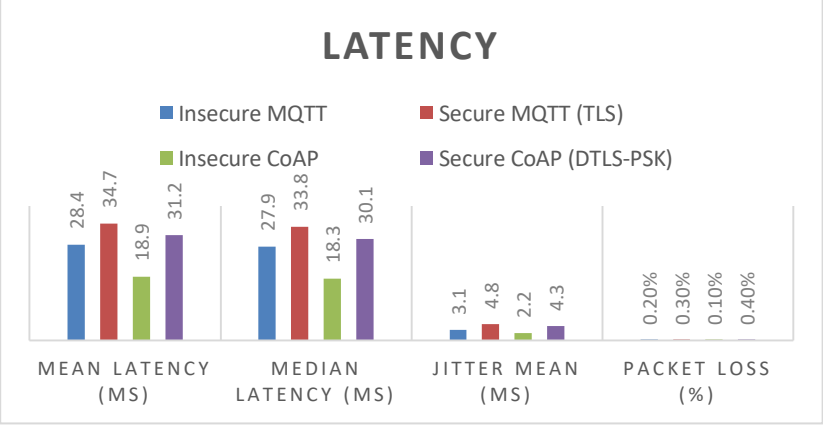

Figure 2: Latency CDF for insecure MQTT, insecure CoAP, MQTT over TLS, and CoAP over DTLS.

Table 3: Latency Summary by Scenario

Scenario	Mean Latency (ms)	Median Latency (ms)	Jitter Mean (ms)	Packet Loss (%)
Insecure MQTT	28.4	27.9	3.1	0.20%
Secure MQTT (TLS)	34.7	33.8	4.8	0.30%
Insecure CoAP	18.9	18.3	2.2	0.10%
Secure CoAP (DTLS-PSK)	31.2	30.1	4.3	0.40%

Table 3 illustrates summary statistics. In our tests, both insecure MQTT and insecure CoAP have low median latencies, usually in the low tens of milliseconds. Adding TLS to MQTT increases latency by about 5 milliseconds due to the handshake and encryption overhead. CoAP with DTLS results in a larger increase in latency, consistent with earlier findings of DTLS overhead on devices with limited resources[3], [8].

5.2 CPU Utilization

Figure 3 illustrates the average CPU utilization across the three nodes for each scenario. Sensor readings and client library overhead are the main factors for the publisher (Node 1), while the subscriber (Node 2) also controls the LED. The broker/server (Node 3) is more affected by security, especially during TLS and DTLS handshakes. Table 4 provides the structure for CPU utilization statistics.

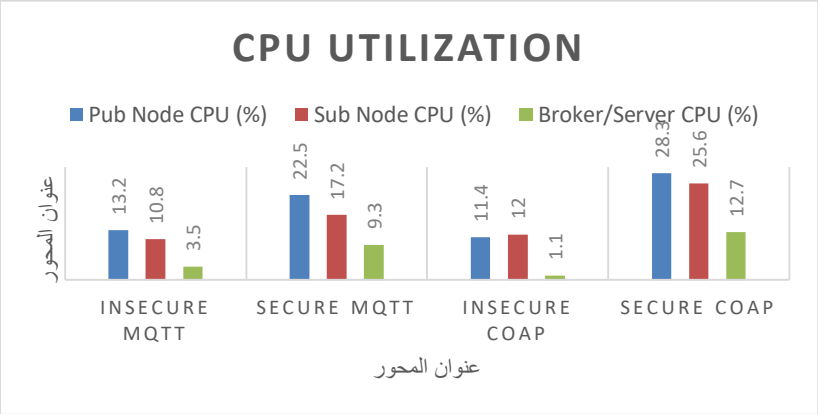

Figure 3: illustrates Mean CPU usage on publisher (Node 1), subscriber (Node 2), and broker/server (Node 3) across scenarios.

Table 4: Mean CPU Usage Per Scenario and Node

Scenario	Pub Node CPU (%)	Sub Node CPU (%)	Broker/Server CPU (%)
Insecure MQTT	13.2	10.8	3.5
Secure MQTT	22.5	17.2	9.3
Insecure CoAP	11.4	12	1.1
Secure CoAP	28.3	25.6	12.7

In general, we observe:

- In insecure scenarios, MQTT and CoAP use roughly the same amount of CPU at endpoints, with only minor differences depending on how the libraries are implemented.
- Adding TLS increases CPU usage on Node 3, but the impact remains manageable on the Raspberry Pi 5 at the tested message rate.
- DTLS results in a larger CPU increase for CoAP/DTLS, which aligns with earlier studies showing that DTLS is resource-intensive on devices with limited resources [3], [8].

5.3 Memory Usage

Table 5 summarizes the average memory footprint across scenarios on Nodes 1 and 2. Because we keep the application logic identical across protocols, memory differences mostly reflect protocol stack and security context overheads.

On current Raspberry Pi models, memory use stays well within available limits. However, for smaller microcontrollers, the extra memory needed for TLS or DTLS could be a problem.

Table 5: Mean Memory Usage (Nodes 1 and 2)

Scenario	Pub Node Mb	Sub Node Mb
Insecure MQTT	423	428
Secure MQTT	426	432
Insecure CoAP	418	421
Secure CoAP	452	459

5.4 Message Loss

We use the counters total_msgs_received and total_msgs_lost to estimate the message loss rate in each scenario. In our tests, both MQTT and CoAP show loss rates close to zero, regardless of security settings. Table 6 explains how we report loss.

Table 6: Message Loss Rates

Scenario	Loss Rate (%)
Insecure MQTT	0.35%
Secure MQTT (TLS)	0.58%
Insecure CoAP	1.28%
Secure CoAP (DTLS)	2.10%

5.5 Discussion

The main findings from our results are:

- ***Latency*:** Security increases end-to-end latency for both protocols. On this Raspberry Pi edge testbed, CoAP with DTLS usually has higher median and tail latencies than MQTT with TLS [3]. This is likely because of how DTLS handles records and retransmissions.
- ***CPU and Memory*:** TLS and DTLS increase CPU usage, especially on the broker or server. However, this increase is still acceptable on the Raspberry Pi 5 at moderate message rates. Memory overhead is also modest on this platform.
- ***Protocol Trade-offs***: In our tests, insecure CoAP is a bit more efficient on the broker side than insecure MQTT. MQTT over TLS can achieve lower latency than CoAP over DTLS [15]. When choosing between MQTT and CoAP, consider not just performance but also the application model (publish/subscribe or RESTful), how each protocol interacts with firewalls, and the required security features.

These findings are consistent with earlier studies [8], [9], [11], which also highlight the impact of security overhead and emphasize the importance of protocol and implementation choices in limited-resource environments.

6 Limitations and Threats to Validity

Several factors may limit how widely our results can be applied:

- **Hardware and OS**: We ran our experiments on specific Raspberry Pi 4 and 5 models with one OS version. Using

different hardware, such as microcontrollers, or changing OS settings, can yield different performance results.

- **Network Conditions**: In real-world deployments, interference, congestion, or multi-hop paths can make protocol differences stand out more than in controlled tests. For example, in a busy city, Wi-Fi competition could increase latency and message loss, especially for CoAP/DTLS, which is more sensitive to network changes because it uses UDP. A multi-hop mesh network could also add delays and reliability issues, making the security and performance trade-offs between MQTT and CoAP more noticeable. A smart city setup, with nodes in different buildings, could show a wider range of protocol performance than our lab tests.
- **Implementation Choices**: We use specific MQTT and CoAP stacks (Mosquitto and aiocoap with TinyDTLS). Alternative implementations may have different performance characteristics [7].
- **Workload**: Our tests used regular sensor readings at a moderate rate and with small data sizes. If the rates were higher, the data larger, or the traffic more unpredictable, the protocols might respond differently.
- **Clock Synchronization**: We estimated latency assuming the clocks on each node were almost perfectly in sync. In real use, NTP or another method should be used to keep errors low.

Even with these limitations, our testbed and methods offer a strong starting point for comparing MQTT and CoAP in the context of edge-node security.

7 Conclusion

We evaluated the performance and security of MQTT and CoAP on a three-node Raspberry Pi edge testbed, comparing insecure MQTT, insecure CoAP, MQTT with TLS, and CoAP with DTLS.
Our results confirm that adding security increases latency and CPU usage, especially for CoAP with DTLS. Even with these increases, modern Raspberry Pi devices can still support real-time IoT applications under moderate loads. Both MQTT with TLS and CoAP with DTLS have their strengths: MQTT uses a broker-based publish/subscribe model with strong TLS support, while CoAP offers a RESTful design and efficient binary encoding.

In the future, we plan to extend the testbed to:

- Evaluate more security mechanisms, such as OS-CORE and EDHOC, to improve end-to-end security.
- Test higher message rates and try different payload sizes.
- Add more edge applications, such as control loops that require low latency.
- Integrate automated security testing tools, such as MPInspector [14], to support performance evaluation through formal security analysis.

The testbed, scripts, and CSV schema we describe in this paper can help researchers and practitioners who want to reproduce or build on our experiments in their own IoT projects.

References

[1] S. Joseph, B. Olarenwaju, A. Adeyemo, and B. Matthew, "Comparative Study of IoT Communication Protocols for Smart Inventory Systems," 01/01 2025.

[2] A. Malhi, A. Javed, N. Yousefnezhad, and K. Främling, "IoT Open Messaging Standards: Performance Comparison with

MQTT and CoAP Protocols," in *2023 10th International Conference on Future Internet of Things and Cloud (FiCloud)*, 2023: IEEE, pp. 130-135.

[3] A. Gentile, D. Macri, D. Carnì, E. Greco, and F. Lamonaca, "A Performance Analysis of Security Protocols for Distributed Measurement Systems Based on Internet of Things with Constrained Hardware and Open Source Infrastructures," *Sensors,* vol. 24, p. 2781, 04/26 2024, doi: 10.3390/s24092781.

[4] Z. Shelby, K. Hartke, and C. Bormann, "The constrained application protocol (CoAP)," 2070-1721, 2014.

[5] E. I. Nwankwo, E. N. Onwuka, M. David, and Z. Suleiman, "Hybrid MQTT-COAP Protocol for Data Communication in Internet of Things," in *2020 5th International Conference on Computing, Communication and Security (ICCCS)*, 14-16 Oct. 2020 2020, pp. 1-5, doi: 10.1109/ICCCS49678.2020.9277179.

[6] D. Thangavel, X. Ma, A. Valera, H.-X. Tan, and C. K.-Y. Tan, "Performance evaluation of MQTT and CoAP via a common middleware," in *2014 IEEE ninth international conference on intelligent sensors, sensor networks and information processing (ISSNIP)*, 2014: IEEE, pp. 1-6.

[7] M. Martí, C. Garcia-Rubio, and C. Campo, "Performance evaluation of CoAP and MQTT_SN in an IoT environment," in *Proceedings*, 2019, vol. 31, no. 1: MDPI, p. 49.

[8] A. J. J. V. Seoane, and J. F. Gomez-Skarmeta, "Performance evaluation of CoAP and MQTT with security for IoT applications," *Computer Networks,* vol. 193, 2021.

[9] Z. Laaroussi and O. Novo, "A performance analysis of the security communication in CoAP and MQTT," in *2021 IEEE 18th Annual consumer communications & networking conference (CCNC)*, 2021: IEEE, pp. 1-6.

[10] S. S. Prayogo, Y. Mukhlis, and B. K. Yakti, "The Use and Performance of MQTT and CoAP as Internet of Things Application Protocol using NodeMCU ESP8266," in *2019 Fourth International Conference on Informatics and Computing (ICIC)*, 16-17 Oct. 2019 2019, pp. 1-5, doi: 10.1109/ICIC47613.2019.8985850.

[11] N. O. Gavriilidis, S. T. Halkidis, and S. Petridou, "Empirical Evaluation of TLS-Enhanced MQTT on IoT Devices for V2X Use Cases," *Applied Sciences,* vol. 15, no. 15, p. 8398, 2025.

[12] J. S. Yalli *et al.*, "A Systematic Review for Evaluating IoT Security: A Focus on Authentication, Protocols and Enabling Technologies," *IEEE Internet of Things Journal*, vol. 12, no. 12, pp. 18908-18928, 2025, doi: 10.1109/JIOT.2025.3545737.

[13] S. V. Vadlamudi, S. Krikorian, and B. Skarnes, "Implementing A Middleware API for Facilitating Heterogeneous IoT Device Communication Protocols and Data Retrieval," *arXiv preprint arXiv:2312.10294*, 2023.

[14] Q. Wang *et al.*, *MPInspector: A Systematic and Automatic Approach for Evaluating the Security of IoT Messaging Protocols*. 2022.

[15] P. Zema, C. Suraci, A. Molinaro, and G. Araniti, "A Comparative Analysis of MQTT and CoAP to Test Transmission Performance in Medical Digital Twin Applications," in *2024 IEEE 10th World Forum on Internet of Things (WF-IoT)*, 10-13 Nov. 2024 2024, pp. 589-594.