IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 30 May 2024, accepted 21 June 2024, date of publication 25 June 2024, date of current version 5 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3419028

RESEARCH ARTICLE

Optimizing Structural Health Monitoring Systems Through Integrated Fog and Cloud Computing Within IoT Framework

MUHAMMAD HASSAN[1], (Student Member, IEEE), ALI HUSSEIN[2], AMR A. NASSR[3,4], RAID KAROUMI[5], USAMA M. SAYED[1,6], AND MOHAMED ABDELRAHEEM[1,7], (Senior Member, IEEE)

[1]Electrical Engineering Department, Assiut University, Asyut 71515, Egypt
[2]Information Technology Department, Faculty of Computers and Information, Assiut University, Asyut 71515, Egypt
[3]Department of Civil Engineering, Ajman University, Ajman, United Arab Emirates
[4]Department of Civil Engineering, Assiut University, Asyut 71515, Egypt
[5]KTH Royal Institute of Technology, 100 44 Stockholm, Sweden
[6]Faculty of Engineering, Sphinx University, Asyut 17090, Egypt
[7]EGC-TECH, Misr International Technological University, Asyut 71515, Egypt

Corresponding author: Amr A. Nassr (a.nassr@ajman.ac.ae)

This work was supported by the Science, Technology and Innovation Funding Agency (STIFA), Egypt, titled "A Smart Internet of Things Based Monitoring System for Critical Infrastructures" (Project 33427). Authors would like to acknowledge Ajman University's financial support (Research Travel Grant RTG-2023-ENIT-01 and Article Processing Charges APC).

ABSTRACT In this paper, we propose the design, operation, and implementation of an Internet of Things-based hybrid structural health monitoring. This innovative system leverages the capabilities of both fog and cloud layers in computing and monitoring. The system architecture consists of leaf nodes deployed on a target structure. These nodes, synchronously, collect acceleration signals from accelerometers attached directly to the structure and transmit the data to an on-site central node via a short-range communication protocol. At the fog layer, the central node, applies damage detection algorithms on the collected data. If a damage is detected, it forwards the acceleration signals to a cloud-based monitoring server using cellular internet connectivity, where more complex algorithms are used to identify and locate the damage. We provide detailed information about the design of the different system nodes, the implementation of damage detection algorithms, and the architecture of the monitoring server. To evaluate the effectiveness of the proposed system, several practical experiments were conducted. The results demonstrate that the hybrid system presented in this paper provides an efficient, reliable and cost-effective approach to damage detection and identification in civil infrastructures.

INDEX TERMS Internet of Things, fog computing, cloud computing, hybrid IoT solutions, structural health monitoring system, synchronized sampling, cellular internet.

I. INTRODUCTION

Urban and essential infrastructures (e.g., airports, bridges etc.) are considered one of the main pillars of any economy. However, these infrastructure systems are susceptible to damage caused by various factors, including excess loading beyond the capacity of the structure, changes in operation, and age deterioration resulting from wear and tear, exposure to environmental elements, and insufficient maintenance. Earthquakes can also trigger other infrastructure damage mechanisms at different levels, ranging from minor cosmetic damage, such as cracking of walls and floors, to severe damage, such as total structural collapse. The location of the damage can also vary, with the beam-column joints being

The associate editor coordinating the review of this manuscript and approving it for publication was Peng-Yong Kong.

particularly vulnerable. The intensity and duration of an earthquake can significantly impact the extent and severity of the damage.

The capability to assess the status of the infrastructures by detecting and identifying any damage is crucial in order to prevent total collapse and protect the lives of the citizens.

Structural Health Monitoring (SHM) involves the process of observing and critically evaluating the target structure [1]. SHM systems can determine the type and location of failures or deterioration in a structure, as well as estimate the remaining lifespan of the infrastructure. These systems typically provide different levels of monitoring as follows [2]:

1) Level I: Detecting the presence of deterioration in the structure.
2) Level II: Pinpointing the location of deterioration.
3) Level III: Evaluating the severity of deterioration.
4) Level IV: Assessing the remaining lifetime of the structure.

SHM systems can be used to characterize the building's response before, during and after seismic events by detecting any presence of a damage. Structural deterioration causes changes in the physical properties of a structure, including stiffness, damping, and mass. These changes can affect the dynamic characteristics (natural frequencies, modal damping, and mode shapes) of the structure [3]. Based on the modal analysis, it becomes feasible to identify the structural properties that have been impacted and necessitate remedial measures to mitigate potential risks.

In the past decades visual inspection was the common approach used to monitor the condition of structures. This approach is of high cost, time consuming nature, and has uncertainty particularly for large structures. Alternatively, wired-based systems (each sensor connected to a data-logger through cables) are used to assess structural health [4]. However, this method has several drawbacks including large installation and maintenance efforts. The rapid development in the wireless and sensing technologies allows for developing monitoring systems containing data processing unit (microcontroller), sensors (accelerometers), wireless communication modules, and structural health assessment mathematical models [5], [6]. Additionally, the Internet of Things (IoT) technology [7] allows devices and objects to connect to the internet and exchange data that can be accessed remotely. By incorporating IoT technology into SHM systems, the performance and efficiency of these systems can be improved by sending the measured data to cloud platforms or remote centers for further analysis and evaluation [8], [9].

Monitoring and assessing the seismic behaviour are often based on deploying a substantial number of sensor nodes on the structure, including displacement transducers, strain gauges, and accelerometers, which can record the structure's dynamic properties [10]. In this regard, highly customized sensing technologies, such as transducer devices (accelerometers and geophones), are essential parts for creating a continuous recording system. However, the cost of these sensors should be considered duo to the large number of sensors that may be deployed. Micro-Electro-Mechanical System (MEMS) technology is now being used in several applications, particularly for accelerometer manufacturing, which has contributed to reducing both the costs and size of the stations while also improving sensor reliability [11].

The measurement of structural responses of large structures requires a dense array of sensor nodes that collect huge amounts of data during long-term monitoring. For a high performance SHM system, an efficient data analysis and management platform is used for extracting useful information from the structural responses and assessing the structural health. Our previous work [12], [13] presented the design and implementation of a cloud-based, synchronized IoT-Based SHM system. This system consists of a group of leaf nodes, a central node, and a cloud monitoring server. The leaf nodes acquire the acceleration signals from different locations in the structure via MEMS accelerometers and then transmit these signals to the central node via short-range communication protocol (WiFi). The data acquisition process is timely synchronized among the different leaf nodes by using the Global Positioning System (GPS) Pulse Per Second (PPS) timing signal. The central node continuously collects the acceleration signals received from the leaf nodes, then forwards them to the remote server over the internet using cellular service, eliminating the fixed infrastructure requirements. The central node does not contribute in the damage detection and identification process; it only relays the received data to the remote server. On the remote server, a number of algorithms are implemented to detect and identify any damage presented in the monitored structure. Nonetheless, this system involves the incessant transmission of data from the central node to the remote server, which is deemed unfavorable for SHM systems due to multiple reasons. Firstly, long-range communication is facilitated through the cellular internet, which does not necessitate a fixed infrastructure but incurs monetary charges for data transfer volume. Secondly, the substantial amount of data transferred from central nodes poses a computational challenge for monitoring servers that are responsible for overseeing multiple infrastructure.

An alternative approach is by utilizing a hybrid fog-cloud SHM system. This system involves executing the damage detection algorithm on the processor of the central node in the fog layer and transmitting the data solely when the algorithm identifies structural damage to the remote server (cloud layer). The monitoring server employs more complex algorithms for identifying and locating this damage.

The proposed system operates as follows: Leaf nodes synchronously gather the accelerometers' acceleration signals. Then, they transmit the collected signals to a central node via a WiFi connection. After receiving the leaf nodes' signals, the central node performs real-time damage detection to assess the structure's condition. If damage is detected, the signals are relayed to the cloud-based monitoring server over the internet using cellular communication. Finally, the received data is

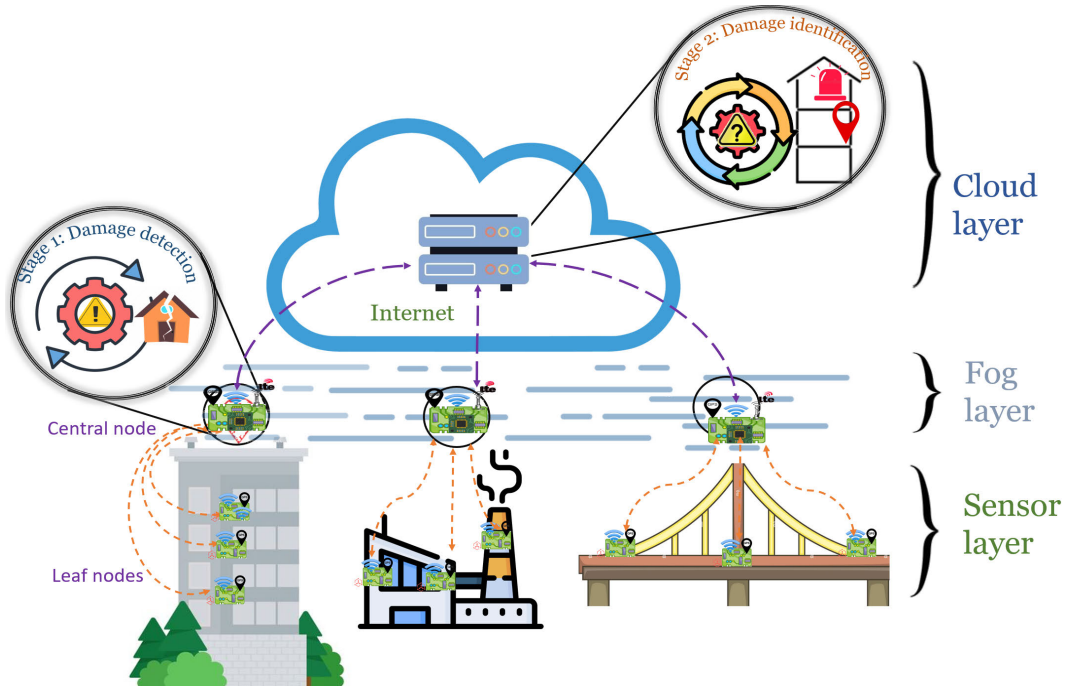

FIGURE 1. Two stages IoT SHM system. In the first stage the damage detection algorithm runs on the central node. If a damage is detected, the data is transmitted to the monitoring station where the second stage is activated by running a damage identification algorithm.

stored by the monitoring system, which employs damage identification algorithms to determine the precise location of the damage. Figure 1 depicts a general view of the presented system highlighting different layers and components.

The contribution of this paper can be summarized in the following points:

1) We present the hardware and power-efficient firmware design of different IoT nodes (central node and leaf nodes)
2) We developed a computationally-efficient implementation of the damage detection algorithms on the central node microcontroller.
3) We designed a cloud based monitoring server with a web interface implementing the damage identification algorithm to visualize the structure data. Also, the administrator can control and tune the system through the server dashboard.
4) We present the operation of the overall system through the different layers.
5) We evaluate the performance of the system using a series of practical experiments.

Paper Organization –The rest of the paper is organized as follows: Section II presents a review of the theoretical background for SHM algorithms, including the modal parameter estimation methods and the different algorithms used for damage detection and identification. Section III describes the main components of the proposed system. Section IV discusses the practical implementation of different algorithms on the IoT nodes. Section V introduces the different practical experiments carried out to evaluate the proposed system. Section VI discusses the proposed system's results. Finally, the paper is concluded in Section VII.

II. SHM THEORETICAL BACKGROUND

In this section, we will present the theoretical background for SHM algorithms used in this research. We started with discussion about feature extraction, quantitative analysis, and decision-making. Then we introduce the methods used for the estimation of the natural frequency and mode shapes of the monitored structure. After that, we present the different algorithms used for damage detection and identification.

A. FEATURE EXTRACTION AND DECISION-MAKING

Feature extraction is essential in structural health monitoring as it allows precise information to be extracted from the raw data collected by numerous sensors mounted on the structure. These features can be categorized as local and global features. Gathering and analyzing data from critical areas within a structure to identify subtle changes or damages is referred to as local feature extraction. This method is vital for identifying localized phenomena such as stress concentration, cracks, corrosion, or elements debonding which could potentially compromise the entire structure if undetected [8], [14]. Advanced sensing technologies like piezoelectric transducers, fiber optic sensors, and capacitance sensors are typically employed for this purpose [15], [16], providing high-resolution data specific to the areas where they are installed. On the other hand, global features refer

to characteristics or measurements that represent the overall behavior or condition of the entire structure. These features are typically derived from data collected from multiple points across the structure, providing a comprehensive view of its health. Techniques such as time-domain, frequency-domain, and modal analysis are commonly used to extract meaningful features from sensor data. These features, including frequency domain characteristics and modal parameters (natural frequencies, mode shapes and damping ratios), provide valuable insights into structural behavior and health status [8], [17]. These features are extracted from real-time signals acquired using different types of sensors including accelerometers, strain gauges, and piezoelectric sensors. In SHM systems, global fusion techniques combine data from various sensors to enhance the accuracy and reliability of health assessments. Quantitative analysis in SHM involves the systematic evaluation of extracted features to assess structural status using different techniques such as baseline comparison that compares current feature values with baseline (undamaged) values to detect anomalies [8] using natural frequency drop, and variation in mode shapes. Also, quantitative analysis involves damage identification techniques such as mode shape curvature method or the flexibility method, and machine learning models that can be used to assess the severity of damage and predict the remaining useful life of the structure based on current health metrics. Techniques such as regression, Neural Networks (NN), or Principal Component Analysis (PCA) and Support Vector Machine (SVM) are used to find patterns and categorize structural states using sensor data and can be trained on historical data to make predictions [18], [19]. However, these methods need a large amount of data for different structure conditions to properly train the model which may not be feasible to generalize and apply on real structures. The integration of wide data analysis approaches that support real-time strategies significantly improves decision-making processes in SHM systems. IoT technology can provide continuous monitoring and instantaneous feedback, which is critical for early assessment of structural health and prompt implementation of corrective actions. Furthermore, integrating predictive analytics into SHM enables the prediction of future problems based on statistical analysis, informing maintenance decisions and resource allocation in advance. This strategic integration increases operational efficiency and considerably decreases the expenses associated with unscheduled repairs and structural failures [8].

B. STRUCTURE PARAMETER ESTIMATION

Changes in physical parameters such as mode shapes, damping, and natural frequencies can indicate alterations in a structure's behavior, allowing for the identification of damage. These physical parameters can be obtained by measuring the ambient or forced structural response. While experimental (forced) modal analysis needs information about the input excitation, operational (ambient) modal analysis only needs information about the output and depends on the input being generated by the surrounding environment, such as earthquakes, passing vehicles, or wind [20].

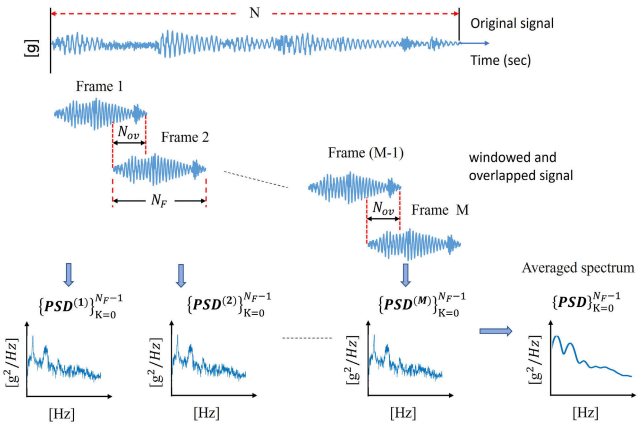

FIGURE 2. Averaged PSD spectrum for N acceleration samples, one channel.

Frequency Domain Decomposition (FDD) is one of the Operational Model Analysis (OMA) vibration-based techniques that are used for SHM thanks to its ability to precisely identify natural frequencies and mode shapes that are closely spaced [21]. The FDD method is based on analyzing the acceleration data in the frequency domain through estimating the Power Spectral Density (PSD) of ambient response that reveals dynamic amplification in the signal near the structure's natural frequencies, making it useful for studying modes or general frequency characteristics [22]. The averaged PSD can be calculated using the Welch periodogram method, which allows data segments to overlap by applying a windowing function to each data segment before computing the periodogram. This results in a reduction in variance and a more accurate estimate of the PSD [23].

For a single channel (sensor) time history, PSD can be estimated by dividing the data into M equal length and overlapped frames (segments), and then the PSD for each segment is calculated based on the Fast Fourier transform (FFT) as shown in Fig. 2. Where N is the total number of samples, N_{ov} is the number of overlapped samples in the proposed system (50% overlapped samples are used), and N_F is the number of samples in each frame.

In case of a Multi Degree of Freedom (M-DOF), i.e. multi-channels, system the dynamic amplification (peaks) in the averaged PSD spectrum represents the distribution of signal frequency components, many of which cannot be associated with known vibrational modes. For example, three PSD (three channels) with a peak at the same frequency may not necessarily indicate three modes [20]. The number of modes (peaks) can be determined by plotting the eigenvalues of the power spectral density matrix's real values versus frequency. The number of modes within a certain band can be determined by the number of lines (eigenvalues) that are significantly higher than the others. This plot is called the "singular value (SV) spectrum." that comes from the

singular value decomposition process applied to any complex rectangular matrix.

Recently, various singular value decomposition algorithms that guarantee good performance in most applications have been developed to solve singular value decomposition (SVD), such as Golub Reisch [24], Demmel-Kahan [25], Jacobi rotation [26], one-sided Jacobi rotation [27], and divide and conquer [28] algorithms. Also, there were a number of attempts for developing and implementing different techniques of SVD algorithms in embedded processors [29] taking into consideration the processing capabilities limitation of these platforms. In [30] the authors presented a comparative analysis of these techniques, which were implemented on a high-performance embedded processor (Cortex-M4). Also, the authors concluded that one-sided Jacobi rotation is the best algorithm for the SVD that surpasses the other SVD algorithms in terms of speed, precision, and energy efficiency. The singular value spectrum produced from the averaged sample power spectral density, has certain properties related to the contributing modes within the resonance band [20]. In the case that there is only one dominant mode in the band, the largest eigenvalue represents the dynamic amplification factor. The eigenvector of the largest eigenvalue corresponds to the partial mode shape near the natural frequency.

After calculating the eigenvalues and eigenvectors, a peak-picking algorithm is applied to extract the mode shape and natural frequency. One of the powerful peak-picking approaches is the Scale-Space Peak Picking (SSPP) algorithm that is based on Scale-Space theory and involves smoothing the data at multiple scales [31]. The main benefit of this algorithm is that it defines and detects peaks based on their stability over iterative smoothing, rather than relying on local derivative assumptions. This allows for the identification of peaks in a global manner, since it emphasizes the detection of existing peaks at multiple scales in the data.

The procedure of OMA algorithm for extracting modal parameters of specific M-DOF system can be summarized as:

1) Selecting the suitable window, the appropriate number of overlapped samples N_{ov}, number of samples/frame N_F and number of frames M, for the given time history data $\{\mathbf{y}_j\}_{j=0}^{N-1}$ $(n \times 1)$, where n is the number of channels and N is the number of samples/channel.
2) Dividing the data $\{\mathbf{y}_j\}_{j=0}^{N-1}$ into M overlapped frames and multiplying each frame with the selected window; calculating the scaled FFT and the PSD for each frame.
3) Calculating the (averaged) PSD matrix $\{\boldsymbol{PSD}\}_{\mathrm{K}=0}^{N_F-1}$
4) Determining the singular values and singular vectors.
5) Peak picking of the singular values and singular vectors to estimate the natural frequencies f and mode shapes ψ.

C. DAMAGE DETECTION AND IDENTIFICATION ALGORITHMS

After estimating and calculating the natural frequency f and mode shapes ψ of the target structure as discussed in Section II-B, different vibration-based techniques can be used to identify the structure status. Below, we provide the theoretical background of the techniques used in this research. Later in Sections IV and III-D, we will present the implementation of these algorithms in the processing node and the remote server, respectively.

1) **Variation in the mode shape:**
The Modal Assurance Criterion (MAC) [32] is a statistical parameter used to measure the correlation or similarity between two mode shapes, one corresponding to the structure healthy state and the other corresponding to the mode shape of a structure test state. The value of the MAC ranges between 0 and 1, with a value of 1 indicating a perfect correlation between the two mode shapes and a value of 0 indicating no correlation. The MAC can be calculated as:

$$MAC_{j,k} = \frac{\left(\sum_{i=1}^{n} [\psi]_{i,j}^{u} * [\psi]_{i,k}^{d}\right)^2}{\sum_{i=1}^{n} \left([\psi]_{i,j}^{u}\right)^2 * \left([\psi]_{i,k}^{d}\right)^2}, \tag{1}$$

where $MAC_{j,k}$ quantifies the correlation level between the j^{th} and k^{th} mode, The numerical value of j ranges between 1 and N', where N' is the mode shape number of the structure in the healthy state, and the value of k ranges from 1 to M', where M' is the mode shape number for the damaged (test) state. The elements of the mode shape matrix of the damaged and healthy structures are represented by $[\psi]_{i,k}^{d}$ and $[\psi]_{i,j}^{u}$ respectively. The symbols d and u represent the damaged and healthy states, respectively, and n indicates the total number of sensor nodes.
In practical applications, the MAC values between two corresponding mode shapes of a system are slightly less than 1 due to errors from numerical calculations and uncertainty resulting from conditions surrounding the experiment [33]. So, a MAC value greater than 80% is considered to be a strong indication of a healthy structure, while a value less than 80% suggests the presence of damage. However, this method can only indicate the presence of damage with no indication of its location, and it also requires higher order modes for detection [34].
2) **Natural frequency drop:**
Damage to a structure causes changes in its physical and modal properties, notably a decrease in its natural frequency. The natural frequency is highly sensitive to damage, with more severe damage resulting in a significant decrease in the natural frequency. The differences in natural frequencies can be calculated by (2) [35]

$$\Delta f = f_j^u - f_j^d, \tag{2}$$

The notation f represents the natural frequency, j denotes the j^{th} mode, u stands for undamaged, and d represents the damaged status. The natural frequency value

may slightly decline even in the absence of damage due to variations in the surrounding experimental conditions. Therefore, damage is considered to have occurred when there is a shift of more than 5% in the frequencies [36]. Similar to the MAC method, this approach can only detect the damage but cannot pinpoint its location. On the other hand, this technique requires fewer modes if compared to the MAC-based technique.

3) **Mode shape curvature damage indicator:**
The curvature at the j^{th} measured DOF of the mode shape-based deflections can be estimated by using equation (3) with measurement points evenly spaced apart by a distance of Δh:

$$\psi''_{i,j} = \frac{\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}}{\Delta h^2}, \tag{3}$$

where $\psi''_{i,j}$ represents the element of the mode shape curvature matrix and it is computed for both the healthy and damaged structure states. Also, the mode shape component is represented by $\psi_{\mathrm{i,j}}$ where it refers to the i^{th} coordinate or measurement point (such as a sensor) at the j^{th} mode. A constant distance, represented by Δh, separates each two consecutive nodes.
The Curvature Damage Indicator (CDI) method is a way to detect and locate damage in a structure by analyzing the relationship between the flexural stiffness[1] of the structure and the shape of its bending (mode shape). When a damage is present, the structure stiffness decreases, leading to an increase in the value of the CDI. By comparing the CDI values of a healthy and damaged structure, the location of the damage can be identified [37]. To calculate the CDI value, the curvature of the structure's mode shape is calculated using a specific formula called the central difference formula [37].
The CDI vector is obtained by taking the average of the mode shape curvature matrices for both damaged and healthy states at a particular sensor or coordinate point, using the given number of modes N. An increase in the CDI value at a specific sensor or node indicates the presence of damage at that location.

$$C_i = \frac{\sum_{j=1}^{N} \left| \psi''^{u}_{i,j} - \psi''^{d}_{i,j} \right|}{N}, \tag{4}$$

where the i^{th} coordinate (node/sensor) at the j^{th} mode has mode shape curvature matrices for both healthy and damaged states, denoted by $\psi''^{u}_{i,j}$ and $\psi''^{d}_{i,j}$, respectively. The CDI vector has a length equal to the number of sensor nodes, with each element represented by C_i.
If a system is undamaged (in a test state), the mode shape curvature matrix values should closely resemble those of the initial (healthy) state's mode shape curvature matrix. However, if there is damage present at a specific location, there is a significant difference in the mode shape curvature values at that location compared to the healthy state, which can be used to identify the location of the damage.

4) **Flexibility method:** When damage occurs, it leads to a decrease in stiffness and an increase in flexibility. As a result, tracking the flexibility matrix can be used as a way to identify damage. In case of normalized mode shape mass $\left(\psi M \psi^t = 1\right)$, the flexibility matrix can calculated using the mode shapes and eigenfrequencies as follows [38]:

$$F = \sum_{i=1}^{N} \frac{1}{\omega_i^2} \psi_i \psi_i^t \tag{5}$$

The primary goal is to compare the flexibility matrices of the undamaged and damaged states.

$$\Delta F = F_u - F_d \tag{6}$$

where ψ_i represents the i^{th} vector of mode shape, ω_i corresponds to the natural frequency of the i^{th} mode, and N denotes the total number of modes. Due to the inverse proportionality between the flexibility matrix and the square of the eigenfrequencies, the matrix tends to approach a stable value more rapidly for lower modes. Therefore, the flexibility matrix can be accurately estimated using only a few lower modes. Typically, the first two modes are sufficient [39].
In the matrix ΔF, each column represents a specific measurement location. The location of the damage is determined by finding the maximum absolute value in each column $\overline{\gamma_j}$. with j denoting the measuring point coordinate.

$$\overline{\gamma_j} = \max_i \left| \Delta F_{ij} \right| \tag{7}$$

When a system is tested and remains undamaged, the flexibility matrix values should closely resemble those of its initial (healthy) state. On the other hand, if there is damage at a specific location, the flexibility values at that location significantly differ from those of the healthy state, thus indicating the damaged location.

III. THE PROPOSED IOT-BASED SHM SYSTEM

In this section we present and discuss, in detail, the different parts of the system. This includes the design of the leaf and central nodes and the function and operation of each of their components. Also, we detail the two network models used between the leaf and central nodes and between the central node and the cloud-based server. At each model we justify and detail the used transmission technique used. At the end we describe the architecture of the cloud-based server and highlight the operation of each different part.

[1]Flexural stiffness refers to the structural resistance exhibited by a structure when subjected to bending forces.

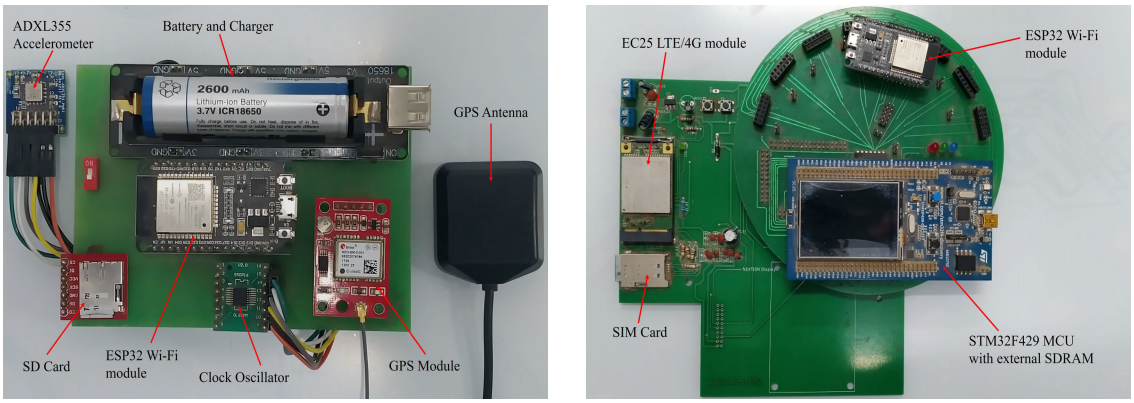

FIGURE 3. The hardware implementation of the proposed system nodes (a) The leaf (sensor) node (b) The central node.

A. LEAF NODES

The proposed system consists of five number of leaf nodes[2] with a design depicted in Fig.3 (a) the main role of each leaf node is to synchronously collect the acceleration signals, collected by the attached accelerometer, of the target structure and then transmit them to the central node. The following components make up the Leaf node:

1) **Processing/communication Unit:** ESP32, a Microcontroller (MCU) with built in WiFi 802.11 b/g/n (2.4GHz) speed up to 150 Mbps module with three main functions. The first role involves gathering the accelerometer's acceleration signals. The second role is signal filtering through the Kalman filter and saving them to an Secure Digital Card (SD-Card), and then adapting them to the required format. The third function is sending the collected data to the central node.
2) **GPS Module:** The GPS module, which is typically used for location identification is used in our system for another very crucial function which is to achieve perfect sampling synchronization, which is importance to achieve accurate signal analysis collected from more than one source, among the different leaf nodes without direct communication between them. Synchronizing the sampling procedure of multiple nodes is performed using the GY-NEO6MV2 with a u-blox 6 GPS module [40]. This GPS module receives accurate Pulse Per Second (PPS) signals, every exactly one second, sent from the GPS satellite network. These signals serve as starting points for the sampling procedure for all leaf nodes. Also, the PPS signals serve the purpose of restricting synchronization errors within a one-second, thus preventing the accumulation of errors [41].
3) **Battery:**
A 2600 mAh lithium-ion battery with a voltage of $3.7V$ powers the device. The battery is installed on a charger. This makes it possible to charge the battery with a 10 Watt solar panel connected to it using USB connector. The efficient design of the leaf node enable it to run on the used battery for time up to 15 hours before need to be charged.
4) **External memory (SD-Card):**
The system has a high availability and redundancy feature through saving the collected samples by each leaf node on an external SD-card. If there is a transmission failure between a leaf node and the central node, the stored samples can be re-transmitted from the SD-card to the central node.
5) **Accelerometer Sensor:**
The digital 3-axis MEMS accelerometer ADXL355 is used to measure the acceleration. It is a minimal noise, drift, and power with its own internal 20-bit Sigma-Delta ADC. It has the ability to capture 3-axis acceleration signals at various output data rates ranging from 3.9 Hz to 4 KHz. The bandwidth of the ADXL355 is constrained to roughly 1.5 KHz by an built-in analogue low-pass filter. In order to lessen the noise, it also offers additional digital filtering options, including a high-pass filter with a cut off frequency set at 1.94 Hz and a low-pass digital decimation filter with a frequency adjustment to 31.25Hz with a sampling rate equals to 125 samples/sec. The accelerometer board connects to the ESP32 using the Serial Peripheral Interface (SPI) protocol with the SPI clock configured to 10 MHz based on the bus clock and baud rate prescaler of the ESP32. As illustrated in Section II C, our system is based on tracking relative shifts in natural frequencies rather than measuring absolute magnitudes of vibrations. This indicates that for our system, the initial baseline measurements are sufficient to detect changes in the structural behavior over time. By establishing a reliable baseline, we can effectively monitor and identify any deviations that may indicate potential issues or changes in the structural integrity without any additional calibration [42]. Our methodology involves attaching the accelerometers at the required points

[2]In this study the number of leaf node is set to 5 to be suitalbe for the lab-scale structure used.

on the structure, recording baseline data under stable conditions, and continuously monitoring the data for any relative changes. This process allows us to detect trends and anomalies with high sensitivity, ensuring that any shifts in the natural frequencies or amplitudes are accurately captured.

B. CENTRAL NODE

The central node is a key component of the damage detection system as it receives the collected acceleration signals from the leaf nodes, stores them in the SDRAM, and implements the real-time damage detection algorithms discussed in Section II. In the presence of any damage, the central node aggregates the data and sends it to the remote server using the cellular internet. The central node, depicted in Fig.3 (b), comprises the following components.

1) The STM32F429 microcontroller [43] is a powerful ARM architectur 32-bit microcontroller with a clock speed of 180MHz and features a Floating Point Unit (FPU). The FPU enhances the microcontroller's capabilities to perform complex floating-point arithmetic operations with high precision and speed which is crucial to perform the damage detection algorithms discussed in Section II.
2) ESP32 Wi-Fi module. This module is identical to their leaf node counterpart which from the hub in a WiFi star network with the leaf nodes and is responsible for the short range communication between the leaf and central nodes.
3) The Quectel EC25 [44] module is a fourth generation (A.K.A Long Term Evolution (LTE)) cellular communication device that facilitates internet connectivity for the central node. It is responsible for establishing and maintaining the internet connection between the central node and the monitoring server.
4) Secure Digital Card (SD-Card) is employed for temporary storage of data when the internet service in unavailable. The central node sends the saved data once the internet connection has been re-established.

C. WIRELESS COMMUNICATION PROTOCOLS

The proposed system depends on using two different communication protocols. The first communication protocol involves short-range wireless transmission between the leaf and central nodes, utilizing the ESP-NOW protocol. The second protocol uses cellular internet for long-distance communication between the central node and the monitoring sever based on the TCP/IP protocol.

1) COMMUNICATION PROTOCOL BETWEEN SENSOR AND FOG LAYERS

The leaf nodes and the central node have the capability to communicate using a modified Wi-Fi protocol called ESP-NOW, which is designed for short-range communication [45]. The ESP-NOW functionality operates within Layer two of the Open Systems Interconnection (OSI) model, eliminating the need for preliminary handshaking in data transfer. This protocol employs Media Access Control (MAC) address for the two communicating nodes, enabling the transmission of data frames of up to 250 bytes. Compared to TCP/IP, ESP-NOW exhibits a faster setup process, consumes lower power, and incurs less payload overhead. This is primarily attributed to its operation at the data-link layer, which eliminates the necessity for data transmission through higher layers such as TCP/IP [46]. It should be noted that our proposed system employs point-to-point direct communication between the sensor layer and the fog layer, without requiring higher layers services such as routing, complex TCP flow, and congestion control.[3] Our system utilizes a sampling rate equals 125 samples/sec., with each sample is stored in a float format data type where each sample occupies 4 bytes. As a result, only two packets are required to be transmitted from each one of the leaf nodes to the central node within a one-second interval. The utilization of this approach leads to a decrease in overall power consumption when compared to the alternative of sending samples individually. This reduction is attributed to the elimination of the need to repeat packet overhead, as explained in [13].

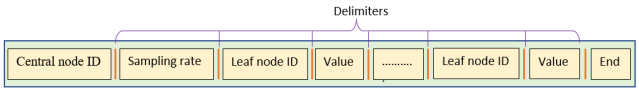

FIGURE 4. Data packet format used between the central node and the monitoring server.

2) COMMUNICATION PROTOCOL BETWEEN FOG AND CLOUD LAYERS

To establish communication between the fog layer and the cloud layer, a cellular based internet utilizing TCP/IP. The cellular communication module on the central node (fog layer) acts as a gateway for the internet by communicating with the internet service provider. The central node is responsible for forwarding the frames of the collected data to the monitoring server (when the damage is detected). By utilizing the Quectel EC25 cellular internet connection (4G) module with a maximum data payload of length of 1460 bytes per packet, the central node is capable of transmitting all the data collected by the leaf nodes within a one-second using only four packets. This is done by utilizing the packet-format of the update message illustrated in Fig. 4.[4]

D. CLOUD BASED MONITORING SERVER

The role of the cloud-based monitoring server is to receive the raw acceleration data (in the case of damage) from the

[3]Re-transmission capabilities are present in both the TCP/IP and ESP-NOW protocols, ensuring that the delivered data packets reach their intended locations. User Datagram Protocol (UDP) is not appropriate for this application since it does not ensure lost packet re-transmission.

[4]It is also possible to continuously transmit all the leaf nodes data to the remote server without processing at the fog layer. However, this is not the default operation.

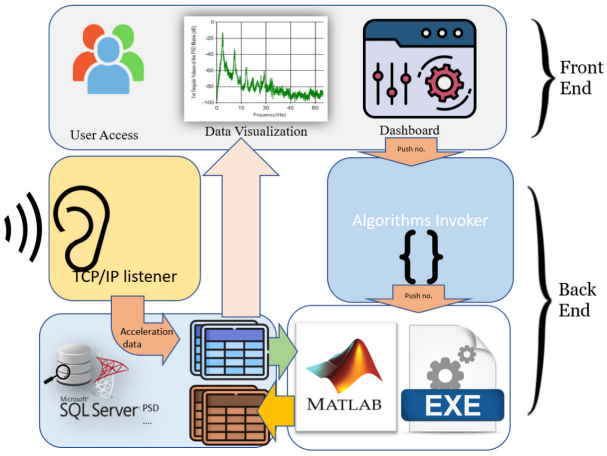

FIGURE 5. The architecture of the cloud-based monitoring server.

central nodes (in the fog layer), store them and apply damage detection and identification algorithm on the received data. The design of the server is shown in Fig. 5. The main system's building blocks are:

- **Front end services** This component is responsible for providing the user with access and control of the server as well as visualization of the damage detection and identification algorithm charts. The main features of this module are as follows:
  - User registration, confirmation, and login: allows the users to create an account and access the site's features.
  - Monitoring and visualizing data: provides the user with real-time information and a graphical representation of the data.
  - Site Dashboard: allows the user to control the server, add/delete central nodes, manually invoke the damage detection and identification algorithms.
  - Accessing historical data: enables the user to view past events and data.
- **Back end services**
  - *The algorithm invoker:* This module invokes the excludable files created using MATLAB software. These executable files runs the proposed damage detection/identification algorithms. The main reason for this module is to facilitate the communication and data processing between the MATLAB written executable algorithm and .Net framework in which the server is built.
  - *TCP listener:* This is a module that permanently listens to packets which contains sensory info from leaves, after that the data are parsed and stored in the Systems' database.
  - *The Database module:* The database stores the raw, timestamped, acceleration data received from the central nodes in addition to the algorithms outputs. The database contains the following tables:
    - Accounts: hold users' data such as username, password, and other related information.
    - Central node ID: identifies the presence of all central nodes and sends an alert if data is not received from a specific node.
    - Values: store the array of sensed acceleration variables' values and the corresponding time stamps when the values were collected.
    - Structure modal parameters: contain the modal parameters of the structure, such as mode shapes, natural frequency, and damping ratio.
    - Status of the structure: saves the structures status, indicating whether it is damaged or not, and also records the location of the damage.

Microsoft .NET technologies were utilized to deploy a monitoring server and website for the purpose of managing user activities and visualizing data. The back-end technologies include a database management system, specifically MS SQL Server 2019. Upon logging in, users can access monitoring functions and historical data navigation. The system offers two modes of operation: manual mode for running damage detection and identification algorithms offline on stored data, and live mode for receiving input from central nodes through the TCP listener module. The application is designed to provide users with a comprehensive view of the system, enabling them to monitor and analyze data, review past events, and make informed decisions based on historical data.

The establishment of a connection-oriented communication between the central node and the server TCP/IP listener is achieved through the utilization of a predefined port and message format, as illustrated in Fig. 4. This enables the server to effectively filter incoming packets and extract acceleration signals from each central node, which are subsequently stored in a database. Upon receipt of all data, various algorithms are executed by the algorithm invoker, commencing with the calculation of the PSD. The modal parameters are then estimated. Finally, damage detection and identification algorithms presented in Section II-C are employed by the server. The monitoring server also facilitates visualization of different mode shapes and their curvatures as well as changes in flexibility.

IV. ALGORITHMS IMPLEMENTATION AND SYSTEM OPERATION

In this section, we present the actual implementation of the different algorithms implemented in the leaf nodes and central node as well as the other implementation aspects including power saving procedure. Fig. 6 depicts the flow chart of data processing steps carried out on the leaf nodes, central node and the monitoring server.

A. LEAF NODE

The leaf node is tasked with the acquisition of acceleration signals, their filtration, and transmission to the central node.

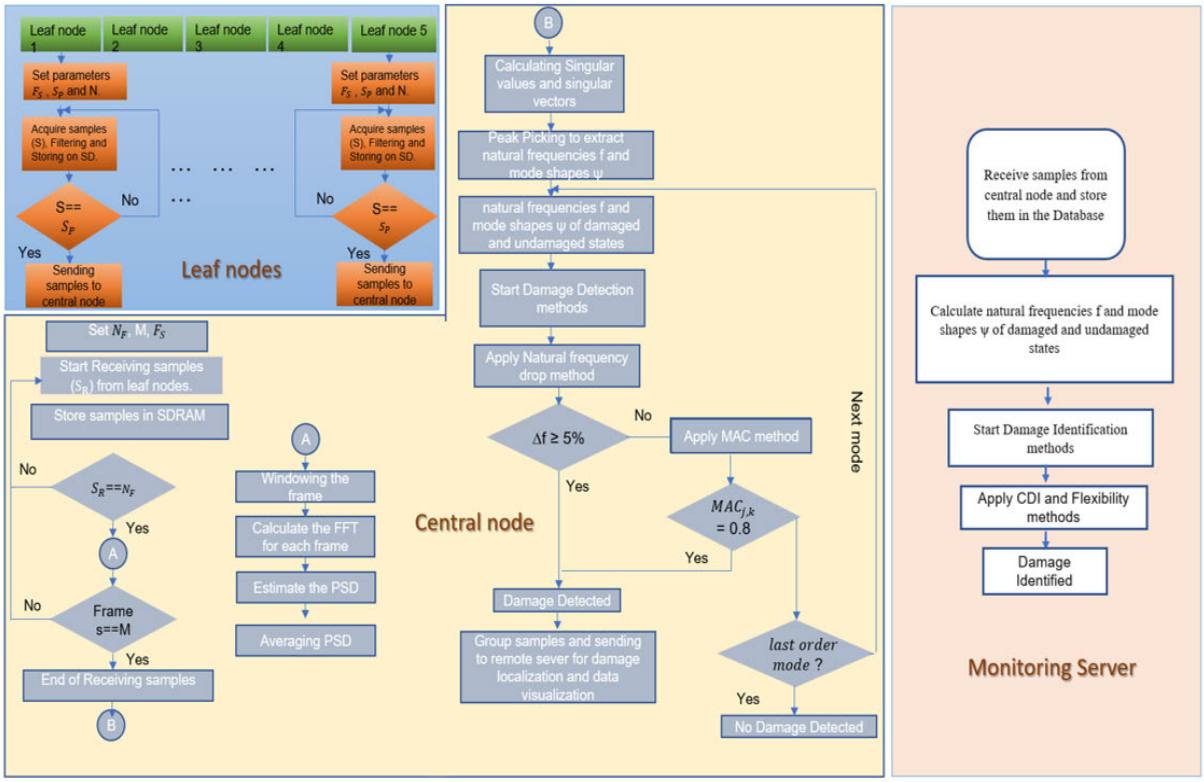

FIGURE 6. Data processing flowchart over the leaf node, central node and the monitoring server.

Furthermore, the leaf node must function in a low power mode to conserve its battery life. Below, we list the various techniques employed by the leaf node.

1) **Leaf node low power operation:**
Our proposed system utilizes GPS modules to synchronize the sampling process across various leaf nodes. In addition to providing location information, these GPS modules also offer precise timing information through an automated PPS signal from the GPS satellite system. The GPS's PPS signal synchronizes leaf nodes, ensuring simultaneous start of their sampling process by aligning the start of each second.[5] Triggered by the PPS signal, the acceleration sensor (ADXL355) begins its sampling step. Once the data is collected, an interrupt signal is sent to the microcontroller, which reads the data from the SPI buffer. This synchronization method guarantees that all leaf nodes successfully attain the necessary synchronization, and the repeatability of this PPS triggering signal every one second ensures that any sampling errors are confined to the one-second interval between successive PPS signals. The timing of the sampling process on the leaf node microcontroller operates independently of other tasks, preventing any adverse effects on the sampling synchronization.

[5]The PPS feature of the u-blox 6 GPS Modules offers an accuracy of 2.35 ns [47].

To minimize the error duo to the drift between different clocks used by different ADXL355 sensors and improve sampling synchronization within the one-second interval, an accurate external Quartz-based Phase Locked Loop (PLL) oscillator clock with a very low drift value is used to by the ADXL355 sensor. The utilization of this external clock mitigates the drift between the sampling of different accelerometers, restricting it to a maximum of $50\mu sec$.

It is crucial to prolong the lead nodes' battery lifespan by improving its power usage. This can be achieved by having the processor node (ESP32 MCU) enter into a sleep mode to minimize power consumption during idle periods when there are processes to perform. In addition, the leaf node groups multiple samples before sending them to the central node, ensuring that it reaches the Wi-Fi's packet Maximum Transmission Unit (MTU) size. This strategy reduces the total number of packets transmitted by each leaf node to only two packets per second. A time line of one second is shown in Fig. 7 which illustrates the operation of the ESP32 (leaf node microcontroller). According to the figure, the MCU switches between normal and sleep modes based on the required tasks to execute. The ADXL355 sensor generates a data-ready signal that prompts an interrupt in the leaf node processor, causing it to exit sleep mode. Upon receiving the data, the processor stores it in a

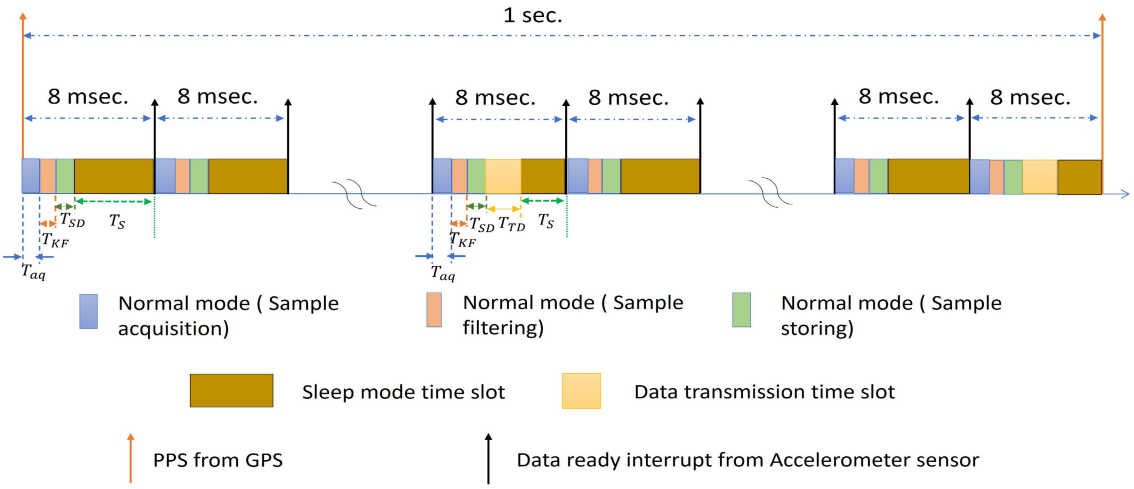

FIGURE 7. A timeline spanning one second illustrating the various operational modes of the leaf node (ESP32).

buffer whose size corresponds to that of the ESP-NOW packet payload, receiving each sample consumes time $T_{aq} = 18.135\ \mu$sec then filter the signal using Kalman filter which takes time $T_{KF} = 4\ \mu$sec after that the processor stores the sample in the SD-card which consumes time $T_{SD} = 25\ \mu$ sec before entering sleep mode again for $T_S = 7.975\ m$ sec until a new sample is available. This process is iterated until the buffer reaches its full capacity, at which point the leaf node remains in normal mode to send the buffered signal to the central node, taking approximately $T_{TD} = 0.645\ m$ sec. Once the data transmission is finished, the processor enters sleep mode until the accelerometer sends the next interrupt signal. Utilizing this technique enables the leaf node to conserve its battery power and prolong its lifespan.

B. CENTRAL NODE

As outlined in Section II, the identification of structural modal parameters is the first step in the damage detection process. This involves estimating the power spectral density, computing the singular values, selecting dominant peaks as natural frequencies, and estimating mode shapes associated with these frequencies. To minimize data transmission to the remote server, our system performs these steps on the central node. Below, we describe various algorithm implementations on the central node processor.

1) **Estimation of a PSD:**
The PSD can be estimated by using the Welch periodogram method that is based on converting each frame (segment) of data from a discrete time-domain into a discrete frequency-domain representation. This is typically achieved by applying a discrete Fourier transform to the time-domain signal by calculating the N_F-point complex DFT of the given discrete-time signal $y(n)$ using (8) where r is the number of frame.

$$\underset{n\times 1}{\mathcal{F}_k^{(r)}} = \sqrt{\frac{\Delta t}{N_F}} \sum_{j=0}^{N_F-1} \underset{n\times 1}{\mathbf{y}_j^{(r)}} \mathrm{e}^{-2\pi ijk/N_F} \tag{8}$$

Direct implementation of the DFT equation requires a large and complex multiplications, so using precalculated twiddle factors that are stored in a lookup table gives significant computational efficiency improvements. FFT algorithm yields the same results as the DFT with fewer required computations. This reduction becomes more important and advantageous with higher order FFTs. STMicroelectronics provides an optimized CMSIS-DSP library for Cortex-M processor.

TABLE 1. Execution times for different DFT functions.

Used method	Number of samples	Time (ms)
DFT	1024	7483629.3
DFTW	1024	1626.5
FFT	1024	7.4202
CMSIS FFT	1024	0.5827

Table 1 shows a comparison of the implementation of DFT, DFTW (DFT with precalculated twiddle factors), FFT (function implemented in pure c) and CMSIS FFT based on ARM processor (1024 samples length).
Using the DFT algorithm is computationally expensive while using twiddle factors can greatly reduce the computational effort by reducing the number of trigonometric calculations required. The FFT algorithm is indeed more efficient than the DFT due to its use of symmetries and the radix-2 structure which allows to perform less calculations and operations. The CMSIS DSP library function is specifically designed for Cortex-M processors, and it is more efficient than other FFT implementations because it is optimized for the specific

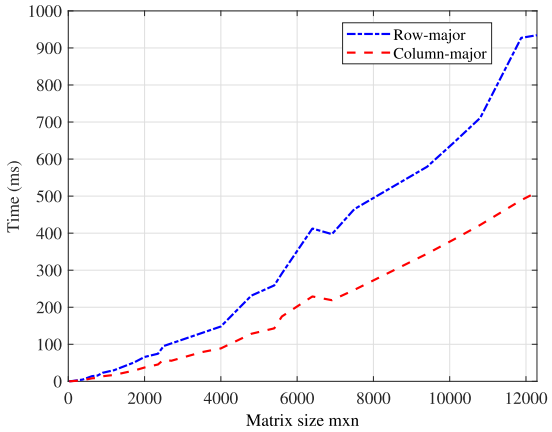

FIGURE 8. Timing of SVD algorithms on Cortex-M4 with respect to total matrix size mxn.

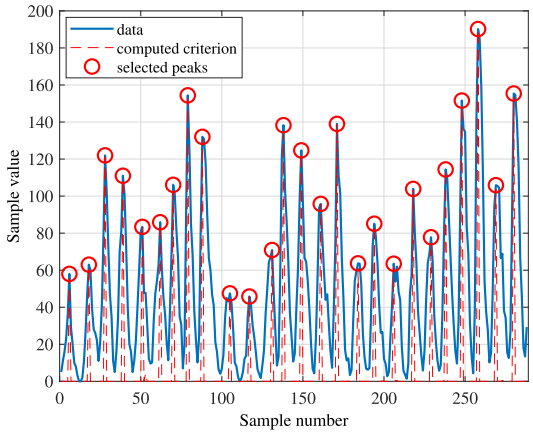

FIGURE 9. Peaks detection for SSPP algorithm performance.

architecture of the STM32F4 microcontroller. This function is more optimized in terms of memory usage and speed than a general-purpose FFT library that is not specific to the STM32F4.

FFT is calculated for each frame using the CMSIS DSP library function then the PSD is calculated as:

$$\underset{n\times n}{\mathbf{PSD}_k^{(r)}} = \underset{n\times 1}{\mathcal{F}_k^{(r)}}\,\underset{1\times n}{\mathcal{F}_k^{(r)*}} \tag{9}$$

where $\mathcal{F}_k^{(r)*}$ is the conjugate of the FFT $\mathcal{F}_k^{(r)}$ and r is the frame number.

After estimating the PSD for all frames, the averaged PSD can be estimated as follows:

$$\underset{n\times n}{\mathbf{PSD}_k} = \frac{1}{M}\sum_{r=1}^{M}\mathbf{PSD}_k^{(r)} \tag{10}$$

where M is the total number of frames.

2) **Singular value decomposition calculation:**
Our work focused on One-sided Jacobi rotation for calculating singular values and singular vectors from the PSD matrix. The one-sided Jacobi algorithm operates on a matrix by successively applying a set of rotations to its rows and columns until the off-diagonal elements become small. This process is repeated until the desired level of accuracy is reached.
In addition to optimizations made by the compiler, optimizing the critical mathematical routines is important for keeping the use of program memory and required RAM memory low to speed up calculations as much as possible while fitting in the limited specs of the system resources. The following steps are implemented:

3) **Peak Picking algorithm:**
   a) Data ordering: Bi-dimensional matrices can be stored in either row-major or column-major order, and optimization can be achieved by selecting the optimum data arrangement in memory. The most popular method of storing data is sequentially by row. This method can significantly speed up access to sequential data on CPUs with cache memory since the cache is populated with sequential data from RAM. Even though a microcontroller has no cache memory, accessing data sequentially still has an advantage as a result of load/store assembly instructions with auto-increment functionality, which enables reading or writing of data from memory and simultaneously incrementing the address register within a single execution cycle.
   The one-sided Jacobi rotation algorithm accesses the input matrix exclusively by column. The effect of row-major or column-major data storing can be assessed by examining the timing of the algorithm for different sets of matrices. Fig. 8 shows the different timings of the one-sided Jacobi implementation based on the two methods
   b) Best procedure of operations: Because of the limitation in power and precision of the FPU, it is important to choose the appropriate procedure for operations that can be implemented in different ways. Loss of precision occurs in some operations (32-bit domain) as a result of limitation in the FPU.
   One of the recurring operations in the given algorithm is calculating $1/\sqrt{1+x^2}$ where in cases of $x \ll 1$ loss of many significant bits occurs in the original value of x.
   A more optimal solution is to apply logarithm to the value we intend to compute, perform intermediate computations within the logarithmic domain, and then apply exponentiation to obtain the final result.
   By using the optimized special function in C math library that is called "log1pf" where this function used to calculate $\log(1+x)$ with high accuracy when x is near to zero. We can use this function to calculate $\log\left(1/\sqrt{1+x^2}\right) = -0.5\log\left(1+x^2\right)$. Utilizing exponentiation and logarithm functions within a 32-bit domain is shown to be a faster and more efficient alternative to calculating the square root function.

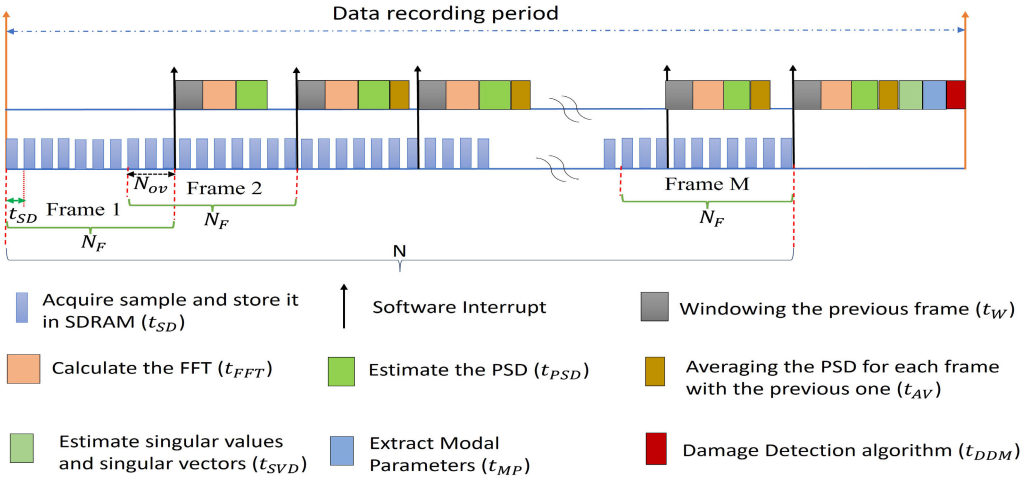

FIGURE 10. Time line for implementing real-time processing on the central node.

The scale-space peak picking (SSPP) approach [31] aims to identify the peaks in a 1D data vector (v) of size N x 1 through an iterative. This algorithm was tested on the central node by using a SUNSPOT dataset (288 samples) to detect the peaks as shown in Fig. 9. When the SSPP algorithm is implemented based on CMSIS DSP library functions, it takes only 36.27 m sec to execute, which is faster than the time required to execute when implementing in pure C without using the CMSIS DSP library (318.92 m sec).

4) **Central node operation:**
The central node starts by receiving samples from Leaf nodes (receiving one sample and storing it in the SDRAM takes time $t_{SD} = 13.46\ \mu$ sec.) where writing one sample to the SDRAM takes a 0.0878 μ sec and reading from it consumes a 0.17 μ sec. Once a certain number of frame samples (2048 samples/frame in our system) have been collected, a software interrupt is generated to process this data. The processing typically includes the following steps as shown in Fig 10:

- Frame windowing: After receiving a specific frame and during receiving the current frame multiply the previous frame of data with a window (Hamming window in the proposed system). This step consumes time $t_W = 4.4063\ m$ sec and it is performed to reduce the spectral leakage that occurs when performing an FFT on a finite length signal.
- FFT Calculation for each frame: After windowing the previous frame calculating its FFT is performed using the CMSIS DSP library which takes a time $t_{FFT} = 3.7395\ m$ sec.
- PSD Estimation: The PSD can be estimated for the previous frame and this consumes time $t_{PSD} = 16.4617\ m$ sec.
- Averaging PSD: After estimating the PSD for the previous frame the central node starts averaging the PSD for the previous frame with the pre-previous one which takes a time $t_{AV} = 1.0611\ \mu$ sec.
- Repeat: The above steps can be repeated until reaching the last frame at which the total averaged PSD is estimated.

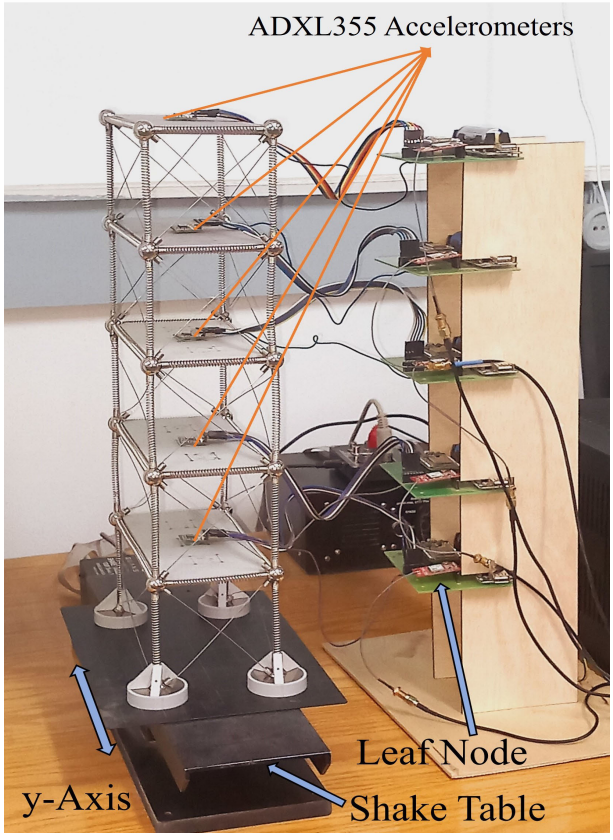

FIGURE 11. The experimental setup of a five-story building model mounted over a shaking table with lead node attached to each floor.

After estimating the power spectral density, the One-Sided Jacobi algorithm is used to calculate the singular values and singular vectors that consume time

$t_{SVD} = 438.1742\ m$ sec. The SSPP algorithm is used to select the dominant peaks in singular values that represent the natural frequencies and estimate the mode shape from the singular vector which takes time $t_{MP} = 227.9928\ m$ sec.

After estimating the natural frequencies and mode shapes, the damage detection algorithms are applied to identify the status of the structure and this consumes time $t_{DDA} = 28.1752\ \mu$ sec. When a damage is detected, the central node uses a 4G/LTE module to transmit the data to the monitoring center.

V. EXPERIMENTAL SETUP

In this section we describe the experimental setup used to evaluate the the feasibility and the performance of the system and the implemented algorithms required for monitoring the structural status.

The test structure used in this experiment is a five-story building model, constructed using Mola structural elements [48], is used for this experiment as depicted in Fig. 11. Each floor of this structure has a leaf node containing an accelerometer sensor that is attached to its center. The Quanser shake table 1-40 [49] is used to simulate seismic events with different intensities, including sine waves, chirps, and actual earthquakes records. The following steps detail the procedures for the experiment operation:

- The initial step includes configuring the essential parameters such as the desired sample count, the sampling frequency, and assigning a unique identifier (ID) to each leaf node, and then the data acquisition process is initiated in its initial state.
- The shake table is employed to externally stimulate the structure and induce an excitation.
- The leaf nodes synchronize the collection of accelerometer data using external clock with low drift and the precise GPS PPS signal.
- The leaf node applies a real-time processing to filter the signal, and then transmit them to the central node.
- At the fog layer, the central node starts real-time processing to the received data.
- The central node starts applying different techniques for modal parameter extraction for the structure's healthy (default) state.
- To introduce damage to the structure, two diagonal braces are removed from the second and fourth floors and then the above steps are repeated again after introducing an artificial damage to the structure.
- In the case of damage detection, the central node open a connection with the monitoring server and relays to it the collected.
- Upon receiving the data, the server stores it in a database and proceeds to apply modal parameter extraction, damage detection, and identification algorithms.

VI. PERFORMANCE EVALUATION

The objective of this Section is to assess the effectiveness of a hybrid system that incorporates damage detection techniques at both the central node (fog layer) and monitoring server (cloud layer). The proposed system operates by collecting acceleration signals from specific points in the structure using synchronized accelerometer sensors at leaf nodes, which then send the data to the central node via short-range wireless communication. The central node processes the signals locally and transmits only the data indicating potential damage to a remote server using long-range cellular internet connectivity. This local processing at the central node enables the system to filter and analyze data on-site, reducing the frequency and volume of data transfers to the cloud for further analysis. By selectively sending only relevant data to the cloud, the system reduces data transmission and storage costs associated with cloud servers, thus enhancing the cost-effectiveness of data transmission and storage. Specifically, we aim at comparing the performance of the central node of the hybrid system with that of a cloud-based system where the server solely executes all algorithms, and the central node only aggregates and transmits data to the server without participating in damage detection and identification stages.

A. IOT CENTRAL NODES PERFORMANCE

In this experiment, comparison was made between the performance of the central node when operating in the hybrid mode and when operating in the cloud based mode in terms of the consumed energy and memory usage. Fig. 12 (a) shows the central node consumed energy in case of hybrid system and cloud based system. It is clear that the consumed energy resulted from applying damage detection algorithms on the central node is lower than that resulted from continuous data transmission using 4G/LTE to the remote server for cloud-based approach.

Fig. 12 (b) illustrates the utilization of different memories (Flash program memory and RAM) of the central node in both a hybrid and cloud-based system. The terms ''text'' and ''data'' sections pertain to the Flash memory size (code footprint). In the case of the cloud-based system, the Flash memory size is calculated as 58.512KB + 0.81KB, resulting in a total of 59.322KB. On the other hand, the hybrid system has a Flash memory size of 115.145KB + 9.176KB, which sums up to 124.321KB. Additionally, the RAM size comprises the ''data'' and ''bss'' (block start by symbol that is used for uninitialized variables) sections. In the hybrid system, the central node utilizes a RAM size of 0.81KB + 29.403KB, totaling 30.213KB. In contrast, In the hybrid system, the central node uses a RAM size of 9.176KB + 136.129KB, which amounts to 145.305KB. Overall, the STM32F429 microcontroller possesses 2 Mbytes of Flash memory and 256 Kbytes of RAM. It is evident that the hybrid system requires more RAM to operate its central node algorithms, and its code occupies a larger footprint than that of the cloud-based counterpart.

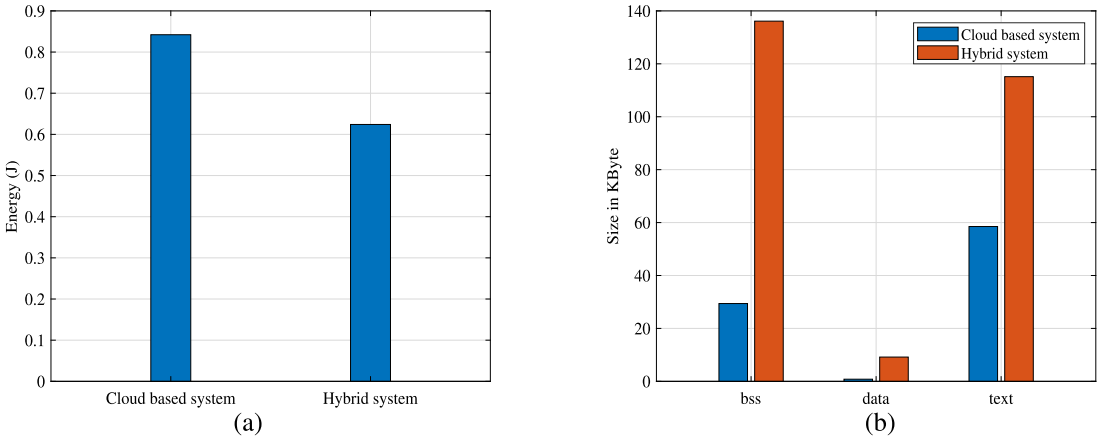

FIGURE 12. The comparison between hybrid and cloud approaches in the case of (a) consumed energy and (b) memory section usage.

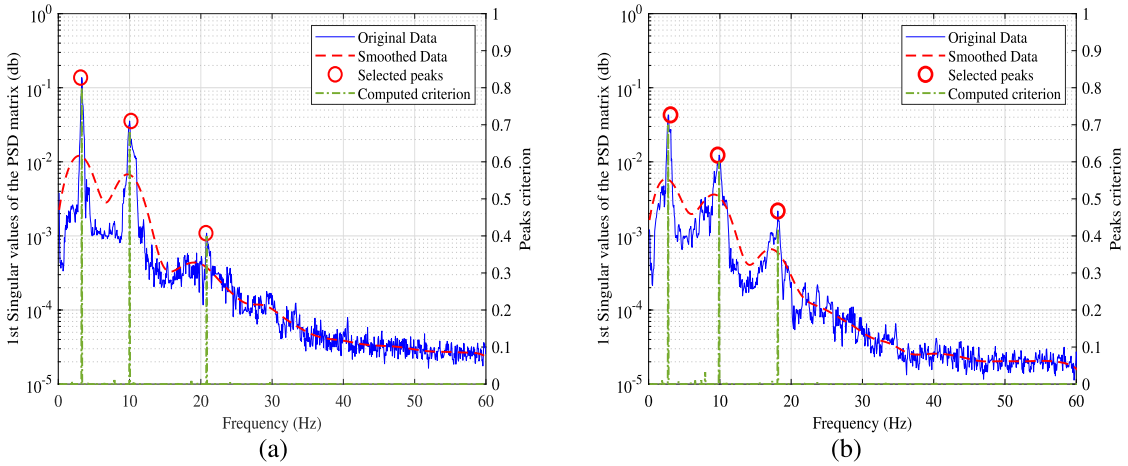

FIGURE 13. The PSD of the test model estimated using the proposed central node under (a) undamaged state (b) damaged state.

B. THE PERFORMANCE OF THE DAMAGE DETECTION/IDENTIFICATION ALGORITHMS

For this experiment, a model of a test building consisting of five floors, as depicted in Fig. 11, was used. The structural stiffness and flexibility were monitored both in its damaged and undamaged states. To simulate damage, two of the diagonal braces on the second and fourth floors were intentionally removed, resulting in a modification of the structure's stiffness.

The distinction between the undamaged and damaged states can be observed in Fig. 13 (a) and (b), which illustrates the PSD of the undamaged and damaged structures, respectively. The PSD estimation was initially performed on the central node as described in Section IV-B where the natural frequencies were determined by identifying the peaks in the PSD using the SSPP algorithm. Table 2 presents the natural frequencies obtained from the central node, which exhibit slight differences compared to those estimated on the monitoring server shown in Table 3. This variation can be attributed to the lower computational capabilities of the central node in comparison to the monitoring server. However, it is important to note that this variation does not affect the results of damage detection algorithms that rely on changes in natural frequencies.

TABLE 2. The values of the the first three natural frequencies for the building model extracted using the central node (in Hz).

State \ Mode	1	2	3
Undamaged	3.1436	10.1873	20.6316
Damaged	3.1653	9.5736	18.2577

TABLE 3. The values of the first three natural frequencies for the building model extracted using the monitoring server (in Hz).

State \ Mode	1	2	3
Undamaged	3.1351	10.1964	20.7163
Damaged	3.0746	9.6887	18.1274

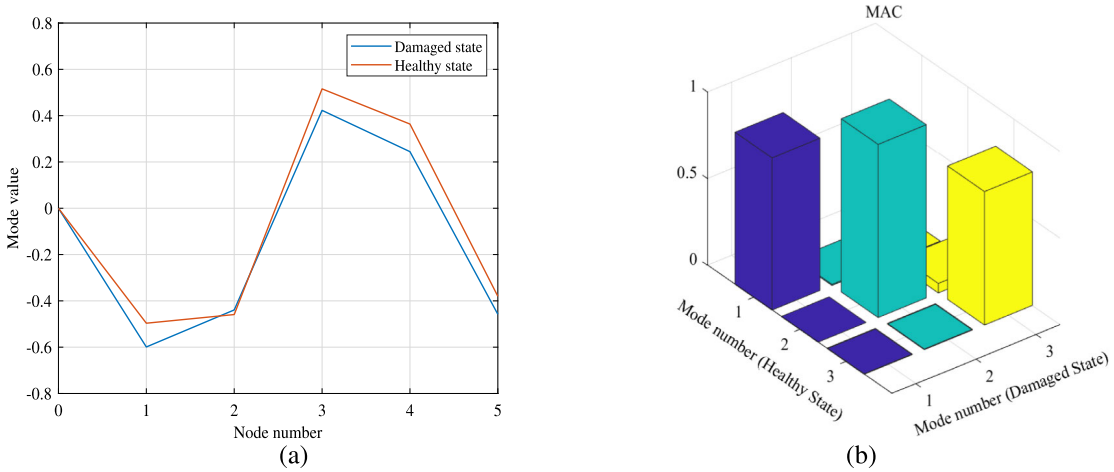

FIGURE 14. A comparison between the test structure in the healthy and damaged states in the case of (a) the third mode shape and (b) the MAC correlation values (using the first three modes).

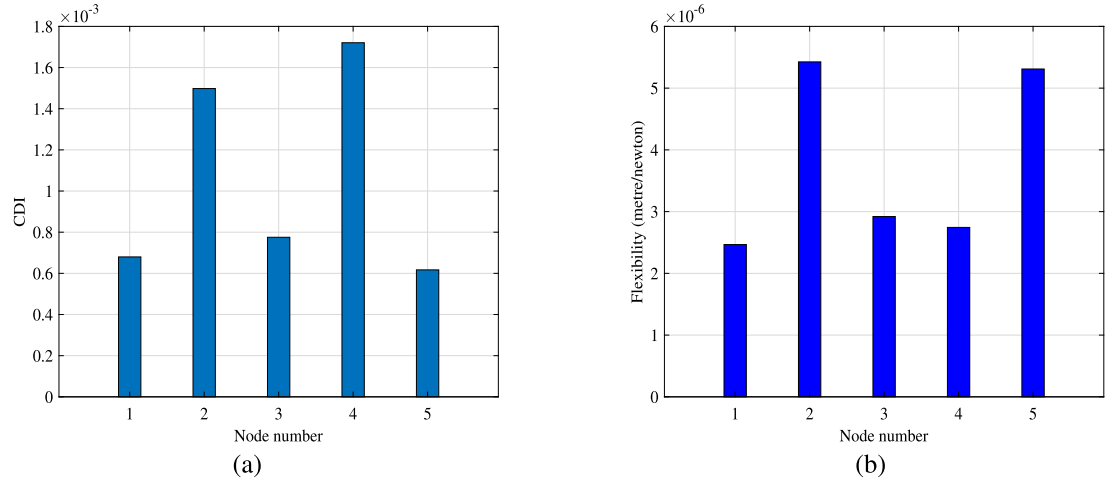

FIGURE 15. Damage localization based on the first three bending modes using (a) Curvature damage index (b) Change in flexibility.

Our initial step is to evaluate the variation in the natural frequency of the structure, which acts as a signal for the existence of any damage. By examining the values of the first three modes natural frequencies as listed in Table 3, we can detect any alterations in the structural properties as this parameter is extremely responsive to such changes. a significant shift is observed in the damaged state. Specifically, there is a deviation of 6.02% and 11.51% in the second and third natural frequencies, respectively, in comparison to the healthy state. As the shift exceeds the predefined threshold of 5% mentioned in Section II-C, the presence of damage is detected. While this method effectively detects the presence of damage in the structure, it does not provide information regarding the specific location of the damage.

Alternatively, we can use the change in mode shape method to calculate the similarity between the mode shapes of the healthy and damaged states. Fig. 14 (a) depicts the third mode shape for both the healthy and damaged states of the building model. The noticeable deviation between the two curves is evident, highlighting the impact of the damage on the structural behavior. Fig. 14 (b) shows the correlation between the first three mode shapes of the test structure in the healthy and damaged states. The Modal Assurance Criterion (MAC) is utilized to calculate this correlation. The correlation values among each mode within the first three modes and its counterpart for the healthy and damaged states are represented by the diagonal elements in the correlation matrix. We note that the third diagonal value is 77.02% that is lower than the MAC rejection level (80%). Accordingly, MAC based algorithm is able to identify the damage. In some cases the MAC-based algorithm may not successfully detect the presence of damage. This is justified by the fact that the MAC method may require modes of higher order beyond the third mode to effectively identify damage in the structure.

The third and fourth techniques (mode shape curvature and Flexibility methods) are sensitive to lower modes and applied on the monitoring server to localize the damage. Fig. 15 (a) illustrates the variation in curvature of the mode shape versus the position of the sensor node. Evidently, the most significant variation occurs at the second and fourth nodes, emphasising the existence of damage near to the second and fourth storeys. Also, Fig. 15 (b) shows the variation in Flexibility versus the location of sensor node. The flexibility method could locate the damage close to the second and fifth stories.

C. LIMITATIONS AND FUTURE WORK

Our structure health monitoring system is designed to be scalable, accommodating structures of varying sizes and complexities through modular sensor networks and flexible data processing algorithms, ensuring effectiveness regardless of structure size. However, our test was performed on a lab-size scale. Some practical challenges may affect the system's performance. Two examples of these are the ESP-NOW protocol can achieve a range of up to 220 meters, However, this range can vary significantly based on environmental factors. Also, as we use ESP as a hub of short-range communication, it limits the number of leaf nodes it can communicate with concurrently. ESP-NOW has a limit on the number of peers it can manage. For example, the ESP32 typically supports up to 20 peers by default [46]. The other one is the limitation associated with battery powered devices. The leaf node is powered by a battery that with its full capacity can operate the node for approximately 15 hours [13]. Given that it can be charged by an attached solar cell, the mentioned period is enough for the battery to be fully charged. However, this may depend on other factors including weather condition and battery status. One of the future research directions is to investigate other short-range communications of longer transmission range and low power consumption, eg. LoRaWan communication protocol.

VII. CONCLUSION

This paper presents the design, implementation, and operation of a hybrid IoT-based SHM system where the damage detection stage takes place in the fog layer and the damage identification stage, if required, takes place in the cloud-based monitoring server. The system consists of leaf nodes which, synchronously based on GPS timing signal, collect the acceleration signals form MEMS accelerometers directly attached to the target structure. These signals are transmitted to an on-site central node using short range communication protocol. the central node processor applies damage detection algorithms on the collected data. If a damage is detected, the central node relays the acceleration signals to a cloud based server using cellular internet. We provided details about the design of the different nodes, the algorithm implementations and the server architecture. Different practical experiments were conducted to assess the effectiveness of the proposed system. The results shows that the energy consumption due to the intensive calculations required for the damage detection algorithm implementation on the microcontroller is less than that required when the central node continuously transmits the data to the remote server without participating in the damage detection process. Also, this approach is cost-effective as the cellular internet, which is a metered connection, is only used in the case of the presence of a damage. The result shows that, despite the computational power limitation of the microcontroller, it was able to detect the damage with performance that is comparable to these take place in powerful workstation.

REFERENCES

[1] W. Ostachowicz and A. Güemes, *New Trends in Structural Health Monitoring*, vol. 542. Cham, Switzerland: Springer, 2013.

[2] J. J. Moughty and J. R. Casas, "A state of the art review of modal-based damage detection in bridges: Development, challenges, and solutions," *Appl. Sci.*, vol. 7, no. 5, p. 510, May 2017.

[3] M. Muttillo, L. D. Battista, T. de Rubeis, and I. Nardi, "Structural health continuous monitoring of buildings—A modal parameters identification system," in *Proc. 4th Int. Conf. Smart Sustain. Technol. (SpliTech)*, Jun. 2019, pp. 1–4.

[4] M. Muttillo, V. Stornelli, R. Alaggio, R. Paolucci, L. Di Battista, T. de Rubeis, and G. Ferri, "Structural health monitoring: An IoT sensor system for structural damage indicator evaluation," *Sensors*, vol. 20, no. 17, p. 4908, Aug. 2020.

[5] S. N. Pakzad, G. L. Fenves, S. Kim, and D. E. Culler, "Design and implementation of scalable wireless sensor network for structural monitoring," *J. Infrastruct. Syst.*, vol. 14, no. 1, pp. 89–101, Mar. 2008.

[6] A. Sabato, C. Niezrecki, and G. Fortino, "Wireless MEMS-based accelerometer sensor boards for structural vibration monitoring: A review," *IEEE Sensors J.*, vol. 17, no. 2, pp. 226–235, Jan. 2017, doi: 10.1109/JSEN.2016.2630008.

[7] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, 4th Quart., 2015.

[8] C. A. Tokognon, B. Gao, G. Y. Tian, and Y. Yan, "Structural health monitoring framework based on Internet of Things: A survey," *IEEE Internet Things J.*, vol. 4, no. 3, pp. 619–635, Jun. 2017.

[9] M. Abdelraheem, M. Abdelhafeez, and A. Nassr, "IoT-based interdigital capacitance sensing system for damage detection in CFRP-concrete structures," *IEEE Access*, vol. 9, pp. 138658–138667, 2021.

[10] A. Moreno-Gomez, C. A. Perez-Ramirez, A. Dominguez-Gonzalez, M. Valtierra-Rodriguez, O. Chavez-Alegria, and J. P. Amezquita-Sanchez, "Sensors used in structural health monitoring," *Arch. Comput. Methods Eng.*, vol. 25, no. 4, pp. 901–918.

[11] G. Langfelder and A. Tocchio, "Microelectromechanical systems integrating motion and displacement sensors," in *Smart Sensors and MEMs*. Amsterdam, The Netherlands: Elsevier, 2018, pp. 395–428.

[12] M. Hassan, A. Nassr, U. S. Mohammed, and M. AbdelRaheem, "An IoT based structural health monitoring system for critical infrastructures," in *Proc. IEEE Global Conf. Artif. Intell. Internet Things (GCAIoT)*, Dec. 2021, pp. 130–135.

[13] M. AbdelRaheem, M. Hassan, U. S. Mohammed, and A. A. Nassr, "Design and implementation of a synchronized IoT-based structural health monitoring system," *Internet Things*, vol. 20, Nov. 2022, Art. no. 100639.

[14] L. Dai, M.-D. Cui, and X.-X. Cheng, "Structural-health-monitoring-oriented finite element model for a specially shaped steel arch bridge and its application," *Math. Comput. Appl.*, vol. 28, no. 2, p. 33, Feb. 2023.

[15] A. Bassil, X. Wang, X. Chapeleau, E. Niederleithinger, O. Abraham, and D. Leduc, "Distributed fiber optics sensing and coda wave interferometry techniques for damage monitoring in concrete structures," *Sensors*, vol. 19, no. 2, p. 356, Jan. 2019.

[16] M. Abdelhafeez, A. A. Nassr, and M. Abdelraheem, "Capacitance-based technique for detection of reinforcement bars in concrete structures," *IEEE Sensors J.*, vol. 21, no. 6, pp. 7713–7724, Mar. 2021.

[17] J. Zhang, G. Tian, A. Marindra, A. Sunny, and A. Zhao, "A review of passive RFID tag antenna-based sensors and systems for structural health monitoring applications," *Sensors*, vol. 17, no. 2, p. 265, Jan. 2017.
[18] A. Gomez-Cabrera and P. J. Escamilla-Ambrosio, "Review of machine-learning techniques applied to structural health monitoring systems for building and bridge structures," *Appl. Sci.*, vol. 12, no. 21, p. 10754, Oct. 2022.
[19] H. B. Bisheh and G. G. Amiri, "Structural damage detection based on variational mode decomposition and kernel PCA-based support vector machine," *Eng. Struct.*, vol. 278, Mar. 2023, Art. no. 115565.
[20] C. Rainieri and G. Fabbrocino, *Operational Modal Analysis of Civil Engineering Structures*, vol. 142. New York, NY, USA: Springer, 2014, p. 143.
[21] F. B. Zahid, Z. C. Ong, and S. Y. Khoo, "A review of operational modal analysis techniques for in-service modal identification," *J. Brazilian Soc. Mech. Sci. Eng.*, vol. 42, no. 8, pp. 1–8, Aug. 2020.
[22] P. Welch, "The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms," *IEEE Trans. Audio Electroacoustics*, vol. AE-15, no. 2, pp. 70–73, Jun. 1967.
[23] S. S. Haykin, *Adaptive Filter Theory*. London, U.K.: Pearson, 2002.
[24] L. Hogben, *Handbook of Linear Algebra*. Boca Raton, FL, USA: CRC Press, 2006.
[25] J. Demmel and W. Kahan, "Accurate singular values of bidiagonal matrices," *SIAM J. Sci. Stat. Comput.*, vol. 11, no. 5, pp. 873–912, Sep. 1990.
[26] G. E. Forsythe and P. Henrici, "The cyclic Jacobi method for computing the principal values of a complex matrix," *Trans. Amer. Math. Soc.*, vol. 94, no. 1, pp. 1–23, Jan. 1960.
[27] H. F. Kaiser, "The JK method: A procedure for finding the eigenvectors and eigenvalues of a real symmetric matrix," *Comput. J.*, vol. 15, no. 3, pp. 271–273, Mar. 1972.
[28] M. Gu and S. C. Eisenstat, "A divide-and-conquer algorithm for the bidiagonal SVD," *SIAM J. Matrix Anal. Appl.*, vol. 16, no. 1, pp. 79–92, Jan. 1995.
[29] W. Yang and Z. Liu, "Accelerating householder bidiagonalization with ARM NEON technology," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, Dec. 2012, pp. 1–4.
[30] M. Alessandrini, G. Biagetti, P. Crippa, L. Falaschetti, L. Manoni, and C. Turchetti, "Singular value decomposition in embedded systems based on ARM cortex-M architecture," *Electronics*, vol. 10, no. 1, p. 34, Dec. 2020.
[31] A. Liutkus, "Scale-space peak picking," Ph.D. thesis, Nat. Inst. Res. Comput. Sci. Control, Inria Nancy-Grand EST, Villers-lès-Nancy, France, 2015.
[32] M. Pastor, M. Binda, and T. Harč arik, "Modal assurance criterion," *Proc. Eng.*, vol. 48, pp. 543–548, Jan. 2012.
[33] C. Rainieri and G. Fabbrocino, *Operational Modal Analysis of Civil Engineering Structures: An Introduction and Guide for Applications*. Cham, Switzerland: Springer, 2014.
[34] D. Brigante, C. Rainieri, and G. Fabbrocino, "The role of the modal assurance criterion in the interpretation and validation of models for seismic analysis of architectural complexes," *Proc. Eng.*, vol. 199, pp. 3404–3409, Jan. 2017.
[35] F. Frigui, J. P. Faye, C. Martin, O. Dalverny, F. Peres, and S. Judenherc, "Global methodology for damage detection and localization in civil engineering structures," *Eng. Struct.*, vol. 171, pp. 686–695, Sep. 2018.
[36] O. S. Salawu, "Detection of structural damage through changes in frequency: A review," *Eng. Struct.*, vol. 19, no. 9, pp. 718–723, Sep. 1997.
[37] R. Frans, Y. Arfiadi, and H. Parung, "Comparative study of mode shapes curvature and damage locating vector methods for damage detection of structures," *Proc. Eng.*, vol. 171, pp. 1263–1271, Jan. 2017.
[38] A. Z. Hosseinzadeh, G. G. Amiri, S. A. S. Razzaghi, K. Y. Koo, and S. H. Sung, "Structural damage detection using sparse sensors installation by optimization procedure based on the modal flexibility matrix," *J. Sound Vibrat.*, vol. 381, pp. 65–82, Oct. 2016.
[39] V. Dawari and G. Vesmawala, "Structural damage identification using modal curvature differences," *IOSR J. Mech. Civil Eng.*, vol. 4, pp. 33–38, Feb. 2013.
[40] (Jul. 16, 2022). *U-Blox, Neo-6 U-Blox 6 GPS Modules Data Sheet*. [Online]. Available: https://content.u-blox.com/sites/default/files/products/documents/NEO-6DataSheet%28GPS.G6-HW-09005%29.pdf
[41] M. AbdelRaheem, M. Hassan, and H. Selim, "A lightweight sampling time error correction technique for micro phasor measurement units," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–8, 2022.
[42] A. Devices. (May 20, 2024). *Technical Articles: Measuring AC Accelerations: To Calibrate or not to Calibrate*. [Online]. Available: https://www.analog.com/en/resources/technical-articles/measuring-ac-accelerations-to-calibrate-or-not-to-calibrate.html
[43] *STM32F405/415, STM32F407/417, STM32F427/437 and STM32F429/439 Advanced Arm-Based 32-Bit MCUs Reference Manual, Rev 19*, STMicroelectronics, Geneva, Switzerland, Feb. 2021.
[44] *Reference Design LTE Module Series, Rev.A*, Beijing, China, Jan. 2018.
[45] *ESP-NOW User Guide, Rev. 1.0*, Espressif, Beijing, China, Jul. 2016.
[46] *Espressif, ESP-NOW: Espressif's Wireless Communication Protocol*. Accessed: Jul. 22, 2023. [Online]. Available: https://www.espressif.com/en/news/ESP-NOW
[47] U-Blox. *GPS-Based Timing Considerations With U-Blox 6 GPS Receivers Application Note*. Jul. 22, 2023. [Online]. Available: https://futuranet.it/futurashop/image/catalog/data/Download/u-blox6ReceiverDescriptionProtocolSpec.pdf
[48] *Molamodel, Mola Structural Model Kit 1*. Accessed: Jul. 22, 2023. [Online]. Available: https://molamodel.com/collections/products/products/mola-structural-kit-1?variant=39951027339443
[49] *Quanser, Shake table I-40*. Accessed: Jul. 22, 2023. [Online]. Available: https://www.quanser.com/products/shake-table-40/

MUHAMMAD HASSAN (Student Member, IEEE) received the B.Sc. degree in electrical engineering from Assiut University, Assiut, Egypt, in 2018. He has been a Teaching and Research Assistant with the Electrical Engineering Department, Assiut University, since 2020. His current research interests include embedded systems design, the Internet of Things networks, cryptography, and structural health monitoring systems.

ALI HUSSEIN received the Ph.D. degree in IoT security using blockchain from Assiut University, in 2021. He is currently an Assistant Professor with the Information Technology Department, Faculty of Information and Computers, Assiut University. His current research interests include the IoT applications, image processing, security, and computer vision.

AMR A. NASSR received the B.Sc. degree in civil engineering from Assiut University, Egypt, in 2004, and the M.A.Sc. and Ph.D. degrees in structural engineering from McMaster University, Canada, in 2008 and 2012, respectively. He is currently an Associate Professor with the Department of Civil Engineering, Ajman University and Assiut University. His research interests include structural health monitoring of structures and resiliency of critical infrastructures exposed to multiple hazards. He is a member of many professional societies, including the Professional Engineers of Ontario (PEO), American Society of Civil Engineers (ASCE), and Canadian Society of Civil Engineers (CSCE). He is also a Former NSERC Fellow, a Former JSPS Fellow, and a Holder of many awards, including the Queen Elizabeth II Award in Science and Technology and International Excellence Award.

RAID KAROUMI has been a Distinguished Academic with the KTH Royal Institute of Technology, with a long-standing tenure, since 1999. His research interests include the dynamics of vehicle-bridge interactions, structural health monitoring, and life-cycle analysis of bridges. He has significantly contributed to the field with over 100 scholarly works and has played a crucial role in monitoring numerous bridges in Sweden. His educational background culminates in a doctoral thesis on the analysis and modeling techniques for cable-stayed and suspension bridges under vehicular movement.

USAMA M. SAYED received the B.Sc. and M.Sc. degrees in electrical engineering from Assiut University, Assiut, Egypt, and the Ph.D. degree in electrical engineering from Czech Technical University in Prague, Czech Republic. From November 1999 to February 2000, he was a Research Assistant with the University of California at Santa Barbara (UCSB), USA. He is currently a Professor with the Faculty of Engineering, Assiut University. He was the Head of the Electrical Engineering Department and after that, he was the Vice Dean of Graduate Studies and Research with the Faculty of Engineering, Assiut University. He is also the Dean of the Faculty of Computers and Artificial Intelligence, Sphinx University, Assiut. He has authored or coauthored more than 150 scientific articles, three undergraduate textbooks, and some chapters in reference books. His research interests include signal processing, wireless communications technology, wireless networks, image coding, statistical signal processing, blind signal separation, and video coding. He was given several awards (Soliman Hozain Award of Engineering Science, in 2007, The Best paper Award in electrical Engineering from Assiut University, in 2005, The Best paper Award in the two conferences NRSC2018 and NRSC2019). He acts as a reviewer and a member of the editorial board of several scientific journals and conferences. He has been selected for the inclusion, in 2010, Edition of the USA-Marquis Who's Who in the World. He is also a Reviewer of *The Quality Assurance Journal* and Accreditation of Egyptian for Higher Education. He is also a Reviewer of the *The Quality Assurance Journal* of the Special Program in the Ministry of Higher Education in Egypt. He is a National Key Facilitator (NKF) in International Labor organization (ILO) in the entrepreneurship.

MOHAMED ABDELRAHEEM (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees in electrical engineering from Assiut University, Egypt, in 2004 and 2010, respectively, and the Ph.D. degree in electrical and computer engineering from Virginia Tech, USA, in 2015. He is currently an Associate Professor in electrical engineering with Assiut University. He is also the Head of the Information Technology Department, Egyptian-German College of Technology (EGC-TECH). His research interests include wireless networking, the IoT, and embedded systems. He is a PI and a Co-PI of a number of research projects from Egyptian funding entities like STIFA, NTRA, and ITIDA.