Received: 16 August 2023 | Revised: 2 November 2023 | Accepted: 22 November 2023

DOI: 10.4218/etrij.2023-0335

ORIGINAL ARTICLE

ETRI Journal WILEY

Finite impulse response design based on two-level transpose Vedic multiplier for medical image noise reduction

Joghee Prasad[1] | **Arun Sekar Rajasekaran**[2] | **J. Ajayan**[2] | **Kambatty Bojan Gurumoorthy**[1]

[1]Department of ECE, KPR Institute of Engineering and Technology, Coimbatore, India

[2]Department of ECE, SR University, Warangal, Telangana, India

Correspondence
Arun Sekar Rajasekaran, Department of ECE, SR University, Warangal, 506371, Telangana, India.
Email: rarunsekar007@gmail.com

Abstract

Medical signal processing requires noise and interference-free inputs for precise segregation and classification operations. However, sensing and transmitting wireless media/devices generate noise that results in signal tampering in feature extractions. To address these issues, this article introduces a finite impulse response design based on a two-level transpose Vedic multiplier. The proposed architecture identifies the zero-noise impulse across the varying sensing intervals. In this process, the first level is the process of transpose array operations with equalization implemented to achieve zero noise at any sensed interval. This transpose occurs between successive array representations of the input with continuity. If the continuity is unavailable, then the noise interruption is considerable and results in signal tampering. The second level of the Vedic multiplier is to optimize the transpose speed for zero-noise segregation. This is performed independently for the zero- and nonzero-noise intervals. Finally, the finite impulse response is estimated as the sum of zero- and nonzero-noise inputs at any finite classification.

KEYWORDS

FIR, signal processing, Vedic multiplier, zero-noise segregation

1 | INTRODUCTION

Noise reduction is the process of removing or eliminating noise from signals. Noise reduction is important for improving the performance range of applications [1]. Finite impulse response (FIR) filters are applied to finite-time impulse responses. The FIR filter is commonly used to estimate the average amplitude of the signals and reduces the latency in the estimation process. FIR-based signal interference reduction methods have been used in signal processing systems [2, 3]. An analysis technique was implemented to analyze nonlinear audio signal effects. Audio effects contain impulsive noise that causes miscommunication between processing systems [4]. The identified effects produced response codes that eliminated the unwanted noise from the signals. An FIR filter is primarily used to modify signals based on noise [5]. The FIR-based method increased the accuracy of the noise reduction process.

Improved noise reduction based on FIR filters has also been used in signal processing systems. The FIR filters the response time of the signals using division and frequency

points. The filtered data provide optimal characteristics for the noise reduction and elimination processes. The improved method reduces the energy consumption ratio during the computational process [6, 7].

The Vedic multiplier reorganizes the conventional multiplication process. The Vedic multiplier is fast in the multiplication process and reduces the time-delay ratio in the computation process [8]. A Vedic multiplier was used for the FIR architecture filter variables at a high-speed ratio. Vedic multipliers are commonly used to develop digital signal processing (DSP) applications. A Vedic multiplier based on the FIR filter was used for the DSP. An FIR filter was used to analyze the key characteristics and frequency ratios of the signals [9]. The FIR filter optimizes the Vedic mathematics to perform the necessary tasks in processing systems. This method aims to identify the noise present in the signals. Additionally, this method also detects noise in complex arrays, which reduces the time-consumption ratio in DSP systems [10]. The Vedic multiplier achieves zero noise in the signals and improves the feasibility and significance range of the processing systems. An FIR channel configuration framework based on the Vedic multiplier algorithm was also used in the processing systems. The configuration framework segments the signals using an FIR channel that minimizes latency in subsequent processes. This configuration framework enhances the performance of processing systems [11, 12].

Vedic multipliers were used to calculate the variables and achieve saturated optimization in different computational fields. A Vedic multiplier is also used to achieve zero noise in signal-processing systems [13]. The actual goal was to obtain zero noise in the signals during the transaction and data-transfer processes. A novel approach based on a Vedic multiplier was used for signal processing [14]. The exact functions of the signals were calculated based on integrated circuits. The Vedic multiplier analyzes the digital signals and produces the necessary information for data processing. This novel approach reduces the noise ratio in signals and improves the functional capabilities of the processing applications [15]. An FIR filter-based method was used to achieve zero noise during signal processing. The FIR filter monitors the noise in the signals [16]. The FIR filter also detects the cause of the interruption present in the processing systems. The FIR-based method optimizes the response time of signals in signal processing systems. The FIR-based method estimates the total noise in signals, which increases the accuracy of signal classification and estimation processes [12, 16]. The contributions of this study are as follows:

1. Designing an FIR based on a two-level transpose Vedic multiplier for medical signal noise reduction with segregation;
2. Modifying the operations of conventional Vedic multiplier with two levels for nonzero equalization and optimization for finite signal classification; and
3. Performing a comparative analysis using definite metrics and methods for validating the proposed method's efficiency.

2 | RELATED WORKS

Guo and others [17] proposed a maximum correntropy FIR (MCFIR) filter for industrial applications. The MCFIR filter estimates the problems based on the severity and conditions of the process. A Gaussian filter was implemented to provide cost functions for the applications. The MCFIR filter increases the filtering performance range, which reduces the complexity of the computational process. The proposed filter improves the robustness of the systems.

Qiu and others [18] developed a new FIR filter for rapid single-flux-quantum technology. Stochastic computing (SC) was used in the filter to reduce the computational cost ratio of the system. The SC theory also minimizes the adders and multipliers during the filtering process. The SC is designed to identify the logic behind the target frequency. The developed FIR filter enhanced the feasibility and significance range of the rapid single-flux quantum.

Ramírez-Chavarría and Schoukens [19] introduced an artificial neural network (ANN)-based nonlinear FIR (NFIR) scheme. The main objective of this scheme was to identify nonlinear impulse responses in various applications. The ANN detects delayed inputs and solves issues that occur during the computation process. The ANN algorithm minimized the sensitivity ratio of the systems. The introduced NFIR scheme reduced the delay level in the finite-impulse identification process.

Carlini and others [20] proposed a new tool to predict and analyze the inertia in power systems. A minimum-variance harmonic FIR filter was used to filter the necessary data for the prediction process. The goal was the identification of sudden changes and issues occurring in power systems. The FIR filter monitored the inertial values and behavioral patterns of the system. Experimental results showed that the proposed method increased the accuracy of the inertia prediction and analysis processes.

Li and others [21] developed a fuzzy adaptive FIR Kalman filter (KF) algorithm for global navigation satellite systems. The developed method used an auxiliary optimization algorithm to improve the development range of systems. The KF was primarily used to reduce the noise level of the given inputs. Abnormal noise and activities were detected using fuzzy adaptive FIR KF, which reduced latency in subsequent processes. The developed

method improved the performance and accuracy of tracking elements in the global navigation satellite systems.

Wu and others [22] proposed an FIR filter-based fault estimation method for linear discrete time-varying (LDTV) systems. The actual conditions of the system were estimated using an FIR filter. A fault estimation algorithm was implemented to identify the causes of faults in the LDTV. A fault estimator was used to detect faults based on the task priorities. The proposed method increased the feasibility and robustness of LDTV systems.

Pu and others [23] introduced an expectation-maximization algorithm using an FIR filter for time-delayed output-error models. The main aim of this method was to identify random delays that were present in systems. It also estimated the parameters and time-delay ratios of the systems. Noise was identified in the system; this process reduced the computational cost range during the execution of tasks. The proposed method increased the accuracy of time-delay detection, which enhanced the feasibility of the systems.

Wang and others [24] designed a gradient-based iterative parameter-estimation method for FIR systems. The proposed method estimated parameters and noise in nonlinear systems. This reduced the problematic aspects presented in the systems and improved the significance level of nonlinear systems. The proposed method improved the overall computational efficiency and reliability range of FIR systems.

Jain and others [25] proposed a power-efficient, multichannel, low-pass filter for digital image processing. A cascade multiple accumulation FIR (CMFIR) was used to filter the delay and response times of the systems. The CMFIR also detected low-pass parameters that provided the necessary solutions to solve these issues. CMFIR filters enhance the functional capabilities of image processing systems. Compared with other methods, the proposed filter increased the effectiveness of the system.

Karthick and others [26] introduced a linear-phase FIR (LPFIR) filter using a water-strider optimization algorithm for a field-programmable gate array (FPGA). The LPFIR filter was primarily used to estimate the coefficients of the optimal filter. This evaluated the optimal coefficients for further processes. The introduced filter minimized pass ripple, which increased the significance ratio of the system. The introduced LPFIR filter improved the performance and feasibility of FPGA systems.

Ganatra and Vithalani [27] developed a variable step-size, variable tap length delayed error-normalized least-mean-square noise algorithm. The developed algorithm is primarily used for noise reduction in FPGA systems. It filters the adaptive structures based on a boost multiplier. The boost multiplier improves the performance of the noise-reduction process. The developed method improved the sustainability and mobility range of FPGA systems.

Zhang and others [28] designed a Lorentzian optimization method to recover sparse signals from two types of impulsive noise. The main objective of this method was to solve optimization problems. It is used as a smoothing strategy to provide effective services to systems. The proposed method reduced optimization problems that occurred during signal recovery. It also increased the accuracy of sparse signal recovery from noisy parameters.

MejiaCruz and Davis [29] proposed an event reconstructing adaptive spectral evaluation approach for the noise-removal process. The proposed approach was commonly used to estimate noise in structural acceleration signals. A spectral abstraction technique was used to remove the unwanted noise from the signals. The proposed approach increased the localization accuracy and improved the performance of the filtering process.

The methods discussed above rely on the impulse response to address the severity and nonlinearity in multiplier systems, as also shown in the literature [17, 19, 25]. Other methods, as also discussed in the literature [18, 24, 27], rely on exclusive filters for noise suppression and optimization of multiplier operations. Differential noise suppression and multilevel operations were facilitated by the proposals in the literature [23, 28]. The aforementioned methods and techniques did not focus adequately on tamper resistance and equalization for noise-control sustainability. This key feature motivated the development of the proposed architecture.

From the summary listed above, it is noted that the FIR design depends on stagewise optimization to improve noise suppression. However, conventional Vedic architectures rely on limited architectures to improve the FIR swiftness across various stages. In conventional optimization processes, self-interference cancellation and zero-noise response are mandatory to prevent continuous-to-discrete interval segregation. However, with limited hardware constraints, the overhead increases without preclassification of the intervals. Handling multiple channels is mandatory to improve the signal and prevent noise overlapping for better FIR designs. These features motivated us to design a novel multiplier with a two-level transpose using a conventional Vedic multiplier.

3 | FIR DESIGN BASED ON TWO-LEVEL TRANSPOSE VEDIC MULTIPLIER

In medical signal processing, obtaining reputable and precise information is essential for the protection and categorization of signals [30]. However, noise and

intervention from sensing and wireless transmission devices help manipulate signals and thwart feature extraction processes. To address these issues, a novel approach based on an FIR architecture and on FIR design based on two-level transpose Vedic multiplier (FIR-TTVM) is introduced herein. An overview of the FIR-TTVM is shown in Figure 1.

The proposed FIR-TTVM architecture determines and terminates noise impulses during different sensing intervals. To achieve this goal, this architecture consists of two processing levels. At the first level, this procedure involves transposing the array operations with equalization, which helps achieve zero noise at any sensed interval. Transpose operations occur between successive array representations of the input signal while ensuring continuity. This continuity is important because (when disrupted) it results in higher noise levels and signal tampering. By meticulously controlling the transpose operations, this architecture effectively reduces or eliminates noise during sensing intervals. The second level of the Vedic multiplier is responsible for optimizing the transpose speed, specifically for achieving zero-noise segregation. From a technical perspective, the proposed architecture describes interval classification, its mapping toward transpose/interference, and the FIR design. In particular, finite-interval-based grouping is exceptional for handling noise suppression across various intervals. Preclassification of intervals is optimal for differentiating the required filtering and grouping instances. These constructive aspects form the technical contributions of the proposed architecture with less overhead and a conventional design.

The input signal is extracted for interval continuity verification. This procedure helps secure a smooth and uninterrupted flow of data between successive intervals. This architecture efficiently determined and reduced unnecessary voice noise interruptions by demonstrating and managing continuity in the acquired signal intervals. This step primarily helps identify precise signal processing. This input is then used to check the interval continuity and is used as the input for an additional zero-noise-checking procedure. The process for determining the signal input is described by the following equations:

$$\eta\left(\frac{x}{y}\right) = (a, x, y), \tag{1a}$$

$$\eta\left(\frac{x}{y}\right) = \left\{ \begin{array}{c|c} 0 & a < 1 \\ (a-1)/(y-1) & 1 \le a \le x \\ (x-y)/(x-1) & x \le a \le y \\ 0 & y \ge a \end{array} \right\}, \tag{1b}$$

where η denotes the extracted input signals, a represents the precise information of the acquired signal, x denotes the time intervals of the signals acquired, and y denotes the noise impulses. After determining the input signals, an interval-checking process was performed. This checking process helps evaluate each successive interval of the acquired input signal to secure its continuity and featureless transition between intervals. Any interruptions or inconsistencies in the interval between them are indicated as noise or interference. An interval continuity-checking process is necessary to determine and separate intervals with zero noise. This acquired nonzero noise helps in the subsequent optimization of the transpose speed of the precise segregation and categorization of the signals. The process of obtaining the interval continuity of the acquired input signals is explained by the following equation:

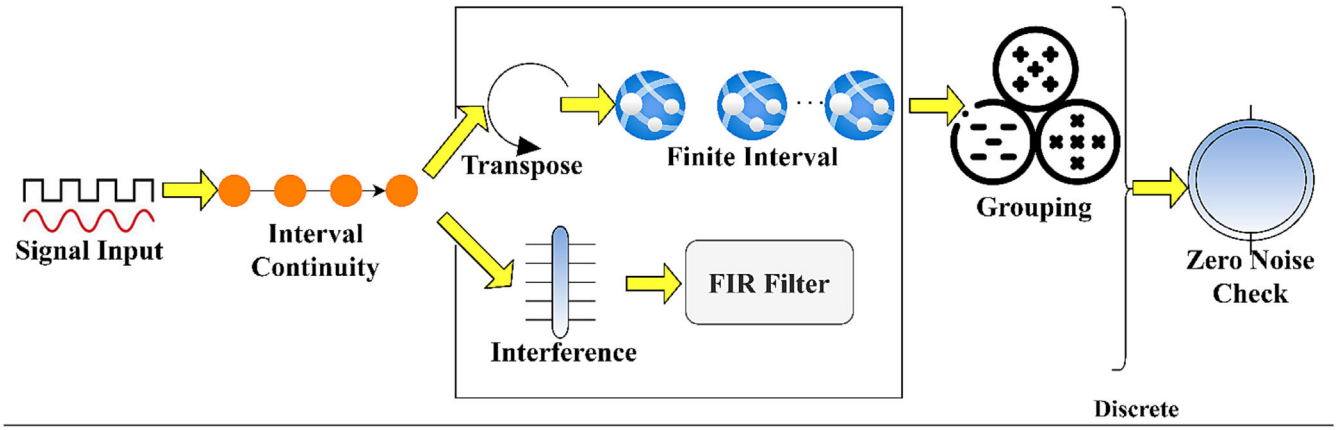

FIGURE 1 Overview of the finite impulse response (FIR) two-level transpose Vedic multiplier.

$$\left.\begin{array}{c}\bar{x}(+)\bar{y}=(x_1, x_2, x_3)(+)(y_1, y_2, y_3)=\{x_1+y_1, x_2+y_2, x_3+y_3\}\\ \bar{x}(-)\bar{y}=(x_1, x_2, x_3)(-)(y_1, y_2, y_3)=\{x_1-y_1, x_2-y_2, x_3-y_3\}\\ \bar{x}(\times)\bar{y}=(x_1, x_2, x_3)(\times)(y_1, y_2, y_3)=\{x_1\times y_1, x_2\times y_2, x_3\times y_3\}\\ \bar{x}(/)\bar{y}=(x_1, x_2, x_3)(/)(y_1, y_2, y_3)=\{x_1/y_1, x_2/y_2, x_3/y_3\}\\ \text{such that,}\\ \bar{x}=\{Vy_1, Vy_2, Vy_3\}\end{array}\right\}, \quad (2)$$

where V denotes the interval continuity of the acquired signals. In this continuity-checking process, the continuous and discrete intervals were determined. In this determination operation, if there are no interruptions or gaps in the signals in which the data points are connected, a smooth interval appears to exist. Discrete signals were considered if there were any interruptions in the data points during the connection. In addition, if there is no flow of precise information in the data, it is represented as a discrete interval of signals. The interval continuity check process is illustrated in Figure 2.

The interval continuity is verified using ys in a$\in x$. In the successive intervals (as in the case of consecutive x), $y\_(x-1)$ is estimated to identify discreteness. The interference suppression is validated as $(x+y)$ or $(x-y)$ depending on V. Therefore, $(x+y)$ is the suppression that refers to continuity whereas the other one $(x-y)$ refers to the discrete interval for segregation (refer to Figure 2). Based on this continuity, further grouping of nonzero-noise signals was performed using the Vedic multiplier. The process of determining the continuous and discrete intervals of the signals is described by the following equations:

$$Z(\tilde{x}, \tilde{y})=\sqrt{\frac{1}{2}(x_1-y_1)^2+(x_2-y_2)^2+(x_3-y_3)^2}, \quad (3a)$$

$$(x_1-y_1)^2=\{(Vy_1+Vx_1)+(Vx_1-Vy_2)\}, \quad (3b)$$

where $x=0, 1, 2, 3, \ldots, n$, $y=0, 1, 2, 3, \ldots, n$, and Z denotes the type of interval continuity of the acquired signals. If the interval is discrete, interference is determined. Interference occurs if there are interruptions between signals. These are extracted from the discrete intervals for further grouping of nonzero noise. To acquire the interference from the signals, this process analyzes the frequency patterns of the data points within discrete intervals. By comparing these characteristics with expected signal intervals, the process separates the interference present in the signal. After determining the interference, appropriate filtering is used in the reduction procedure. The process of identifying the interference from the acquired signals is explained by the following equations:

$$\overline{Q}=\left[\tilde{a}_{ij}\right]_{n\times j}, \quad (4a)$$

$$\overline{Q}_{ij}=\tilde{x}_{ij}\times x, \quad (4b)$$

$$a_j=(j=1, 2, 3, \ldots, j), \quad (4c)$$

$$\overline{Q}=\left\{\overline{Q}_{ij}, i=1, 2, \ldots, n; j=1, 2, 3, \ldots, j\right\}, \quad (4d)$$

where $x_i=(i=1, 2, 3, \ldots, n)$, $i=1, 2, \ldots, n; j=1, 2, \ldots, j$, Q represents the interference, i denotes the frequency of the acquired signal extracted for the checking procedure, and j represents the determined pattern of the signals. After identifying the interferences, they are used in the process of the Vedic multiplier to determine the finite intervals. The Vedic multiplier incorporates the interference information to optimize the transpose speed of the acquired signals. In addition, this multiplier considers the filter

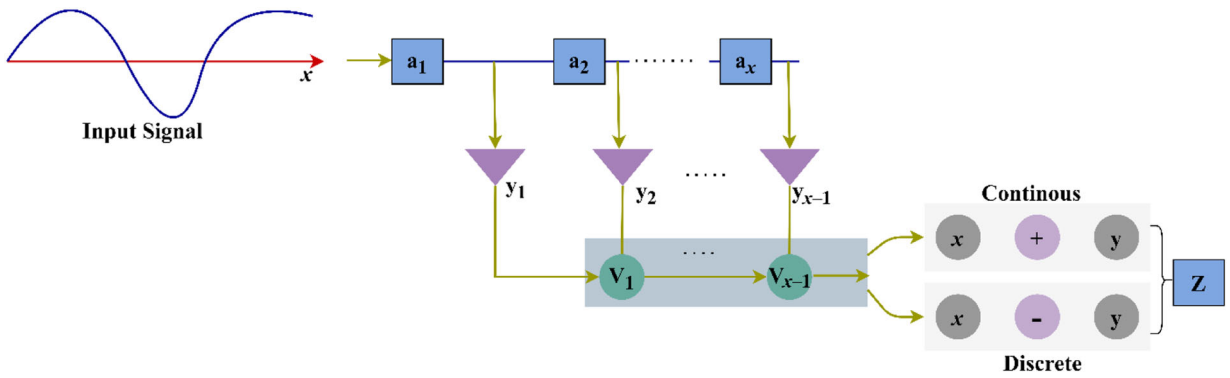

FIGURE 2 Interval continuity check process.

process in the optimization procedure to determine zero-noise outcomes. For the precise determination and classification of finite intervals, these interferences are used in the subsequent process; this process is expressed by the following equations:

$$\tilde{P}_{ij} = \left(x_{ij}, y_{ij}, z_{ij}\right), \tag{5a}$$

$$x_{ij} = \sum\nolimits_{ij}\{x_{ij}\} - \left\{y_{ij}\right\} = \frac{1}{Z}\sum\nolimits_{ij}^{Z} y_{ij}, z_{ij} - \sum\nolimits_{ij}\{x_{ij}\}, \tag{5b}$$

$$\tilde{Q}_{ij} = \sum\nolimits_{ij}\left(x_{ij} \times y_{ij}\right)^{-1} = \frac{1}{4}\left\{\sum\nolimits_{ij}\left[P_{ij}\left(x_{ij} + y_{ij}\right)\right]\right\}^{-2}, \tag{5c}$$

$$\widehat{Q} = \left[\tilde{T}_{ij}\right]_{i\times j}, \widehat{P} = \left[\tilde{R}_{ij}\right]_{i\times j}, \tilde{Q}_{ij} = \tilde{P}_{ij}(\cdot)\widehat{Z}, \tag{5d}$$

where P represents the interference information to be used in the subsequent Vedic multiplier process. The interference is reduced using the FIR filter procedure by checking the status of the interval continuity. The identified interference was effectively reduced by using the FIR filter procedure. This filter was used to remove particular frequency components associated with interference while maintaining the essential features of the input signal. This filter helps reduce unwanted noise and interference, thus resulting in a more reliable signal for subsequent signal processing. The process of the FIR filter used to reduce interference is explained by the following equation:

$$\left.\begin{aligned} \alpha(n) &= \begin{cases} 0 & t \le \alpha \le 1 \\ t & \text{otherwise} \end{cases} \\ \begin{bmatrix} \alpha_0(n+\Delta n) \\ \alpha_1(n+\Delta n) \end{bmatrix} &= \begin{bmatrix} 1-\alpha\Delta n & \alpha\Delta n \\ \alpha\Delta n & 1-\alpha\Delta n \end{bmatrix} \cdot \begin{bmatrix} \alpha_0(n) \\ \alpha_1(n) \end{bmatrix} \\ \alpha(n) &= \frac{\alpha}{\alpha+\Delta n} + \frac{t}{\alpha+\Delta n} \\ \alpha(n) &= \frac{\alpha}{\alpha+\Delta n} \end{aligned}\right\}, \tag{6}$$

where α is represented by the unnecessary noise disruptions (in the form of interference) in the signal. Additionally, the information about the FIR filter in reducing interference was used in the Vedic multiplier procedure. This filtering process also helps the assessment of the status of the acquired signal intervals. This process also ensures that the filtered output maintains smooth transitions between successive data points. Algorithm 1 presents the interval classification steps.

Algorithm 1 Interval Classification

Input: η, x

for $a \in \eta$ do{

compute $\eta\left(\frac{x}{y}\right) = \frac{(a-1)}{(y-1)}\ \forall\ 1 \le a \le x$

if{$(x \le a)||(a \le y)$}{

$\eta\left(\frac{x}{y}\right) = \frac{(x-y)}{(x-1)}$

identify $V \in \frac{(a-1)}{(y-1)}\,||\,V \in \frac{(x-y)}{(x-1)}$

if$\left\{V \in \frac{(a-1)}{(y-1)}\right\}${

compute $\bar{x} = \{Vy_1,\ Vy_2,\ \ldots,\ Vy_n\},\ \forall\ a \in \eta$

if $\{V = (x+y)\}${

compute $Z(\bar{x},\ \bar{y})\forall(x-y)^2$

The sequence is continous

else

$V = x - y$ and

$Q = \left[a_{ij}\right]_{n\times j}\ \forall\ \begin{matrix} i = 1,\ 2,\ \ldots,\ n \text{ and} \\ j = 1, 2,\ \ldots,\ i \end{matrix}$

The sequence is discrete

} end if

} end if

Update $\bar{x} = \{Vy_n\}\ \forall\ V \in (x+y)$

$x = \{Vy_{n-1}\}\ \forall\ V \in (x-y)$

} end if

} end for

By conserving the interval continuity of the input signal, the FIR filter prevents signal disruptions by establishing precise signal processing. Hence, the determined outcomes helped the Vedic multiplier operation to produce zero-noise outputs. The process of using an FIR filter to check the interval continuity of the acquired signal can be explained by the following equations:

$$\alpha_2(n) = \frac{\lambda_1\alpha_1}{\lambda_1\alpha_2 + \lambda_2\alpha_1 + \lambda_1\lambda_2} + \frac{\lambda_2(\lambda_1 + \alpha_2 - \alpha_1)}{(\lambda_2+\alpha_2)[(\lambda_1+\alpha_1) - (\lambda_2+\alpha_2)]}, \tag{7a}$$

$$\alpha_2(\alpha) = \frac{\lambda_2\alpha_2}{\lambda_1\alpha_2 + \lambda_2\alpha_1 + \alpha_1\alpha_2}, \tag{7b}$$

where λ is represented as the FIR filter operation in reducing the interferences. If the interval is continuous, then the transpose array operations are determined using a Vedic multiplier. The transposing process for zero noise is shown in Figure 3.

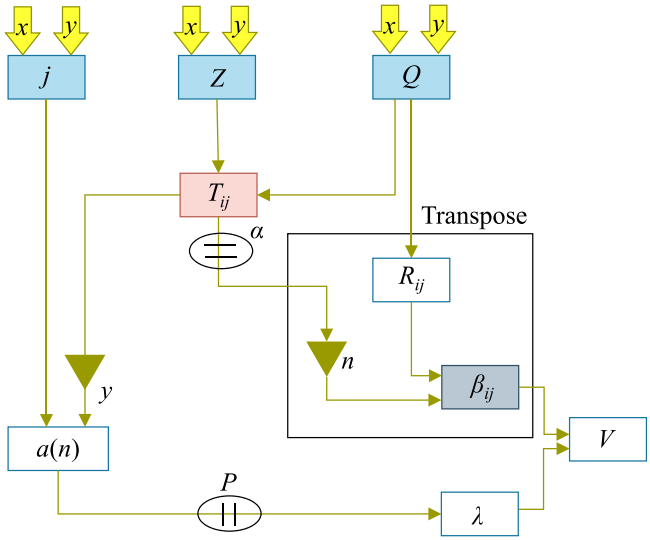

FIGURE 3 Transpose process illustration.

A transpose was performed to identify V from T_{ij} and (Z, Q). The validation was performed with Z to transpose T_{ij} to identify α. If the transpose is not validated by α, then Q is used for the transposition of R_{ij} instead of T_{ij}. The modified transposes are y and n for recurring P for detecting $\lambda \in V$ (refer to Figure 3). This transposition occurs between successive representations of the input with continuity. A Vedic multiplier was used to optimize the transpose speed and achieve zero-noise segregation within a continuous interval. By performing efficient transpose operations between successive array representations of the input signals, this architecture ensured that continuous intervals were free from noise interruptions. These outcomes also aid in precise signal classification and feature extraction of the acquired input signals. The process of determining the occurrence of a transposition between successive array operations is explained by the following equations:

$$\alpha = \frac{1}{(\lambda_1+\alpha_1)(\lambda_2+\alpha_2)}, \tag{8a}$$

$$\beta = \frac{1}{(\lambda_1+\alpha_1)[(\lambda_2+\alpha_2)-(\lambda_1+\alpha_1)]}, \tag{8b}$$

$$\gamma = \frac{1}{(\lambda_2+\alpha_2)[(\lambda_1+\alpha_1)-(\lambda_2+\alpha_2)]}, \tag{8c}$$

$$\begin{aligned}\alpha(n) &= \sum\nolimits_{n=1}\alpha(n) = \frac{\alpha_1\alpha_2}{\lambda_1\alpha_2+\lambda_2\alpha_1+\alpha_1\alpha_2} \\ &= \frac{-\lambda_2(\lambda_2+\alpha_1-\alpha_2)}{(\lambda_2+\alpha_1)[(\lambda_2+\alpha_2)-(\lambda_2+\alpha_1)]},\end{aligned} \tag{8d}$$

where β denotes the determined transpose outcome and γ represents the successive array operations. For the detection of the transpose between successive array operations, a Vedic multiplier was used. To speed up the operation, information about the interference and FIR filter operation was used. After the extraction of the transpose, the finite interval of the signal was estimated. This procedure helped maintain the integrity and accuracy of continuous intervals during signal processing operations. This process can be explained by the following equation:

$$\begin{gathered}\pi(\alpha n = i|\alpha_{n-1} = i, \alpha_{n-2} = n\ldots) = \alpha(\alpha n = i|\alpha_{n-1} = i) \\ \alpha_{ij} = \pi(\alpha n = j|\alpha_{n-1} = i) \\ \text{where} 1 \le i,j \le N \\ \alpha_{ij} > 0 \forall j,i \\ \sum\nolimits_{j=1}\alpha_{ij} = 1 \\ \pi_i = \alpha(\Delta n = i) \\ \text{where} 1 \le i \le n \\ P_n = \frac{Q}{P+Z} \\ P = 1 - 1/\left[\gamma\left(1+\frac{1-Q}{Z}\times\frac{\beta_o}{P_o}\right)\right]\end{gathered}, \tag{9}$$

where π represents the speeding procedure of the Vedic multiplier operation. The finite interval was determined from the transpose using the Vedic multiplier. By utilizing the processed transposed data information, this process determined the exact bounds of finite intervals in the acquired input signals. These finite intervals represent noise- and interference-free signals, thus enabling precise signal processing and categorization.

The Vedic multiplier was used to determine finite intervals from the transposed data with the help of interference information and the FIR filter process. This finite-interval detection process (which used a Vedic multiplier) can be explained using the following equations:

$$\begin{aligned}G[\alpha_i] &= \sum\nolimits_{n=1}^{\infty}\alpha.\pi_i(\alpha) \\ &= \sum\nolimits_{n=1}^{\infty}\alpha.\alpha_{ij}(1-\alpha_{ij}) \\ &= (1-\alpha_{ii})\frac{\partial}{\partial\alpha_{ij}}\left(\sum\nolimits_{n=1}\alpha_{ij}^n\right) \\ &= (1-\alpha_{ii})\frac{\partial}{\partial\alpha_{ij}}\left(\frac{\alpha_{ii}}{1-\alpha_{ii}}\right) \\ &= \frac{1}{(1-\alpha_{ii})} \\ \beta_j &= \beta_j(G) \\ &= \alpha(\lambda_t = \alpha_n \,|\, \beta_t = j\},\end{aligned} \tag{10}$$

$$\left.\begin{array}{c}\prod_{n=1}^{N}\alpha_n\sum_{i=1}\beta_n(i)=1\\ \prod_{n=1}^{N}\alpha_n.Q(\lambda|\gamma)=1\\ Q(\lambda|\gamma)=\dfrac{1}{\prod_{n=1}^{N}\alpha_n}\\ \sum_{n=1}Q(\lambda|\gamma)=\sum_{n=1}^{N=1}\dfrac{1}{\prod_{n=1}^{N}\alpha_n}\\ =-\sum_{i=1}\alpha(n)\\ \beta n(i)=1,\quad 1\le i\le n\\ \tilde{\beta}n(i)=\sum_{i=1}(i),\ 1\le i\le n\end{array}\right\},\tag{11}$$

where G is a finite interval determined from the transposed data. These finite interval signals were then grouped to identify the nonzero-noise intervals. The second Vedic operation is illustrated in Figure 4.

The second operation of the Vedic multiplier optimizes the speed of the first layer. This optimization is required for γ (under j and α) for array transposition. Transposition is required for T_{ij} and R_{ij} across Z to identify G or P. G or P is grouped after the τ operation to achieve zero-noise throughout for Q and γ (Figure 4). This grouping process helps targeted noise elimination and further optimization of the transpose speed to improve signal accuracy. This was performed independently for zero- and nonzero-noise intervals grouped by the transpose operator. The process of grouping the finite intervals can be explained by the following equations:

$$\tau_i=\gamma_1-(i)=\frac{\alpha_1(i)\beta_1(i)}{\sum_{i=1}\alpha_1(i)\beta_1(i)}=\frac{\alpha_1(i)\beta_1(i)}{\sum_{i=1}\alpha_t(i)},\tag{12a}$$

$$\alpha_{ij}=\frac{\sum_{t=1}^{T=1}\alpha_t(i,j)}{\sum_{i=1}\gamma_t(i)}=\frac{\sum_{t=1}^{T=1}\alpha_t(i)\beta_t(j)}{\alpha_t(i)\beta_t(i)},\tag{12b}$$

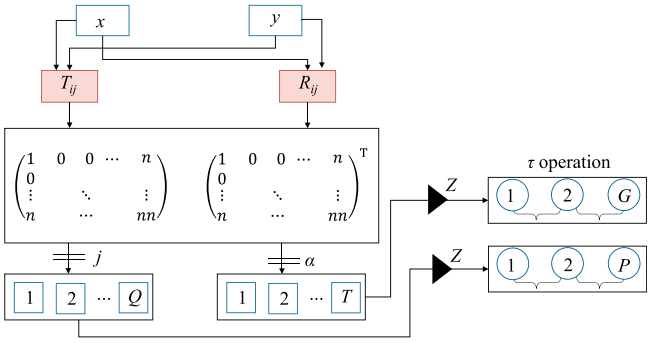

FIGURE 4 Second Vedic operation process.

$$\left.\begin{array}{c}Z(i,j)=\sum_{t=1}(\alpha_t-\alpha_{ij})^T(\alpha_n-\alpha_{ij})\\ G(n,N)\coloneqq\{\alpha_{i1},\ldots,\alpha_{i2-1}\},\{\alpha_{i2},\ldots,\alpha_{i2-1}\},\ldots\ldots,\{\alpha_{in},\ldots,\alpha_n\}\\ Z[\alpha(n,t)]=\sum_{j=1}Z(ij,ij+1-1)\\ Z[\beta|\alpha,Q]=\dfrac{Z[\alpha,\beta|Q]}{Z[\alpha|Q]}\\ Z\{\beta_i|\alpha,Q\}=\sum_{n=1}\gamma(\alpha,\beta)\end{array}\right\},\tag{13a}$$

$$\tau(a)=\begin{cases}0 & a\le x\\ \dfrac{1}{2}+\dfrac{1}{2}a\dfrac{\tau}{y-x}\left(a-\dfrac{y+x}{2}\right) & x<a<y\\ 1 & y\le a\le z\\ \dfrac{1}{2}+\dfrac{1}{2}a\dfrac{\tau}{x-z} & z<a<x\\ 0 & a\ge z\end{cases},$$

where τ is represented as the grouping of the acquired finite intervals. From this, nonzero intervals were determined. The final estimation of the FIR was obtained by associating the outputs from both the zero and nonzero intervals determined during signal processing. By adding these outputs, the architecture effectively established a noise-free signal by eliminating unwanted noise interruptions. The grouping process is as follows.

Algorithm 2 Grouping Process

Input: α, j
for$\{V\in(x+y)\{$
compute $\alpha(n)$ $u$sing equation $(8d)$
$L1$:
compute $\pi\ (\alpha n=i|a_{n-1}=i)$ using equation (9)
if $\{G[\alpha_i]=\alpha,\pi_i(\alpha)\}$then
compute P using equation (9)
if$\{(x<a)\|(a<y)\}$ and $\{(Z<a)\|(a<x)\}\{$
perform $\tau(a)\forall\ V\in(x+y)$
update $G(n,N)$
else if $\{(a\le x)\|a\ge z)\}\{$
$V\in(x-y)$
discard $\tau\ (a)$
end if d
update $Z\{\beta_i|\alpha_iQ\}$ $u$sing equation $(13\ a)$
if $\{\tau(a)==\tau(\pi)\}$
End
else
Goto $L1$;
end if
Update $a-\frac{(y-x)}{2}\ \forall\ V\in(x=y)$
}end if
end for

The process of determining nonzero intervals from the previous operation is explained by the following equation:

$$\left.\begin{aligned}\left|\frac{\widehat{x}}{x_0}-1\right| &= \left|\frac{x_1}{x_0}-1+\frac{\widehat{x}-x_i}{x_1}\frac{x_1}{x_0}\right| \\ &= \alpha\beta-\left|\frac{\widehat{x}}{x_0}-1\right|(1+\alpha\beta) \\ Q(T=0) &= \alpha\left(\left|\frac{\widehat{x}}{x_0}-1\right| \le \left|\frac{\widehat{x}}{x_1}-1\right|\right) \\ &= \left(\alpha\beta-\left|\frac{\widehat{x}}{x_1}-1\right|(1+\alpha\beta) \le \left|\frac{\widehat{x}}{x_1}-1\right|\right) \\ &= \left(\left|\frac{\widehat{x}}{x_1}-1\right| \ge \alpha\beta\right) \\ U &= 1\left\{\left|\frac{\widehat{x}}{x_0}-1\right| > \left|\frac{\widehat{x}}{x_1}-1\right|\right\}\end{aligned}\right\}, \quad (14)$$

where U represents the zero-noise check interval. This process helps detect interference using discrete interval continuity. The filter design is shown in Figure 5.

The filtering process incorporates R_{ij} and T_{ij} for noise reduction. The noise reduction process d was pursued for $\beta \in T_{ij}$ and π was performed. The variations ($\pi \in \tau$) were suppressed using all (i, j) values across the ($x + y$), ($x - y$), and ($x \times y$) operations. This was required to remove additional noise interruptions across different signal inputs (Figure 5). Therefore, an FIR filter was used to reduce the interference; hence, this information was used in the Vedic multiplier process. The transpose was detected between successive array operations, and the finite intervals were then determined and grouped. Based on this classification, nonzero intervals were extracted. The Q and α values were validated for a sample input signal with $x = 50$ s, as shown in Figure 6.

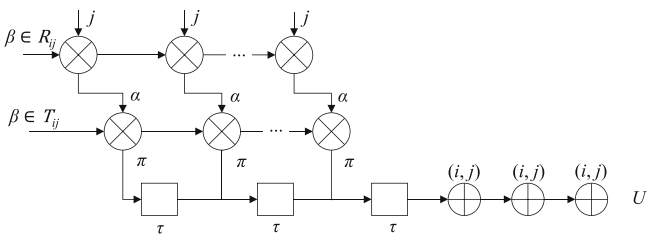

FIGURE 5 Filter design using τ.

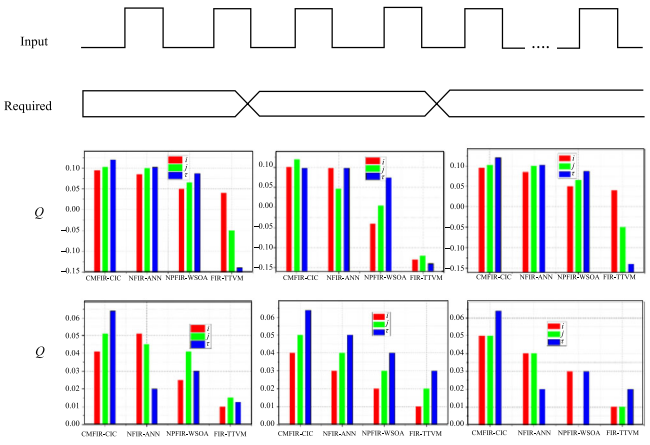

FIGURE 6 Q and α analysis.

Q and α are the identified intervals of the required output signals to achieve high-filtered outputs. Considering the $(x+y)$ and $(x-y)$ for multiple T_{ij} and R_{ij}, this is induced for j and Z. Based on the second level of multiplier operation, the τ value was required for further noise suppression. This is pursued for (x,y) inputs for new π. Thus, the multiplier transpose induces the changes between successive intervals for suppression of Q and α (Figure 6).

4 | RESULTS AND DISCUSSION

This section presents an assessment of the proposed method using the classification, noise, delay, and interrupt rate metrics. The results and discussion of the previous and pursued analyses are provided in Yamanashi and others [31]. These data provide a Verilog-based logical FIR design. Based on this information, a nine-layer process with a current density of 10 kA/cm^2 was endorsed in this study for analysis. The CONNECT open data library was used to verify the nine-layer process using Verilog software. The time interval was varied between 10 and 100 s, and the normalized frequency ranged between 0.5 and 3 Hz for the assessment. The CMFIR-CIC [25], NFIR-ANN [19], and LPFIR-water-strider optimization algorithm [26] methods were introduced in Section 2.

4.1 | Classification

The classification process is effective in this method with the help of a precise interval continuity-checking procedure. After extracting the interval continuity, continuous and discrete intervals were determined. If the interval was continuous, a transpose determination was performed. If the interval was discrete, the interference and its reduction processes were performed.

The finite interval was identified and grouped. After grouping the finite intervals, nonzero-noise intervals were extracted based on the outcome of the Vedic multiplier process. The Vedic multiplier incorporated the interference information to optimize the transpose speed of the acquired signals. The interference and FIR filter information were then used in the Vedic multiplier to speed up

the identification operation of nonzero-noise intervals. The nonzero-noise outcome helps the subsequent optimization of the transpose speed of the precise segregation and categorization of the signals (Figure 7).

4.2 | Noise

Noise was reduced using this method by the Vedic multiplier in the classification procedure. After acquiring the transpose and interference from the interval continuity, the grouping process of the nonzero intervals was performed. Using a Vedic multiplier, the accuracy of the process was enhanced by mitigating unnecessary noise and interruptions. From the input signal, the interval continuity was determined for a further classification process in which the finite interval was identified. Based on this, noise and zero-noise intervals were extracted using a Vedic multiplier. A Vedic multiplier was used to optimize the transpose speed and achieve zero-noise segregation within a continuous interval. The FIR filter helped reduce unwanted noise and interference; this resulted in a more reliable signal for subsequent signal processing. By conserving the interval continuity of the input signal, the FIR filter prevented signal disruptions by establishing precise signal processing. Hence, with all the aforementioned features of the Vedic multiplier, noise was reduced during this process (Figure 8).

4.3 | Delay

The delay in the process was smaller in the zero-noise interval detection process with the aid of precise interference and mitigation information. Transposition was performed when the interval continuity was continuous. This transposition occurred between successive representations of the input with continuity. For the detection of the transpose outcome between successive array operations, a Vedic multiplier was used; to speed up the

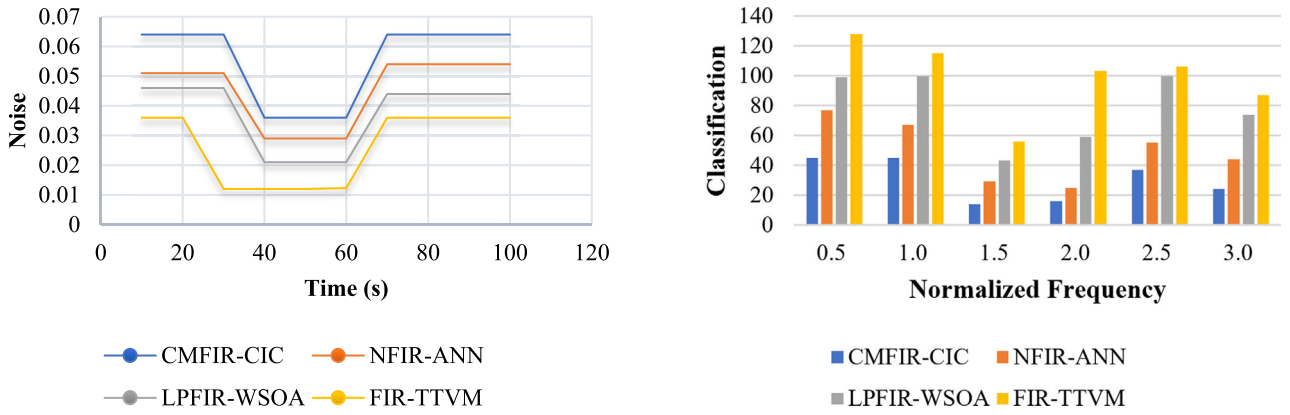

FIGURE 7 Classifications analysis. ANN, artificial neural network; CMFIR, cascade multiple accumulation finite impulse response; FIR, finite impulse response; LPFIR, linear-phase finite impulse response; NFIR, nonlinear finite impulse response; TTVM, two-level transpose Vedic multiplier; WSOA, water-strider optimization algorithm.

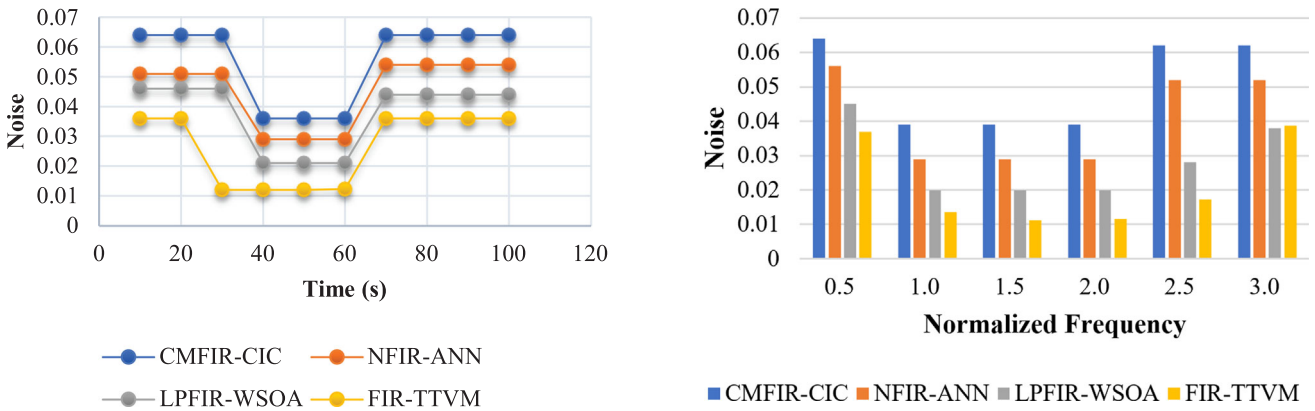

FIGURE 8 Noise analysis. ANN, artificial neural network; CMFIR, cascade multiple accumulation finite impulse response; FIR, finite impulse response; LPFIR, linear-phase finite impulse response; NFIR, nonlinear finite impulse response; TTVM, two-level transpose Vedic multiplier; WSOA, water-strider optimization algorithm.

operation, information about the interference and FIR filter operation was used. After the extraction of the transpose, the finite interval of the signal was estimated. This procedure helped maintain the integrity and accuracy of continuous intervals during signal processing operations. In addition, it helped reduce the delay between processes while it estimated zero-noise intervals. Utilizing the processed transposed data information, this process determined the exact bounds of finite intervals in the acquired input signals (Figure 9).

4.4 | Interrupt rate

The interrupt rate was lowered by mitigation in this process using the FIR filter after the evaluation of the interval continuity. Interference occurred if the interval continuity was discrete. This interference was reduced by using the FIR filter and by checking whether there was intervals continuity. This filter was used to remove particular frequency components associated with interference while maintaining the essential features of the input

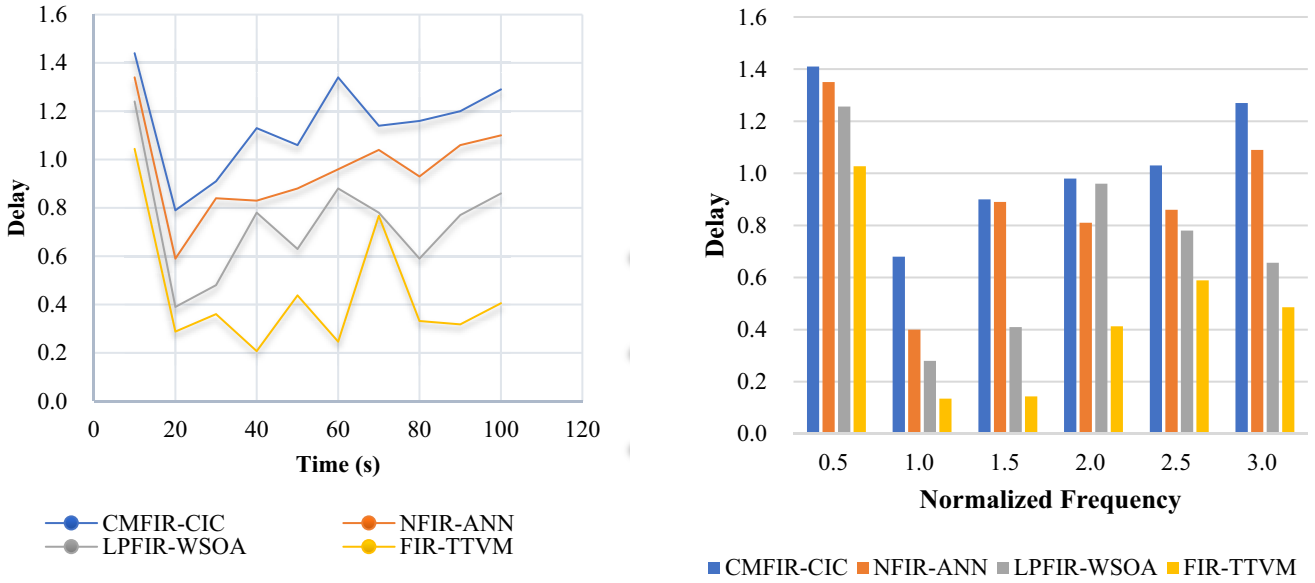

FIGURE 9 Delay analyses. ANN, artificial neural network; CMFIR, cascade multiple accumulation finite impulse response; FIR, finite impulse response; LPFIR, linear-phase finite impulse response; NFIR, nonlinear finite impulse response; TTVM, two-level transpose Vedic multiplier; WSOA, water-strider optimization algorithm.

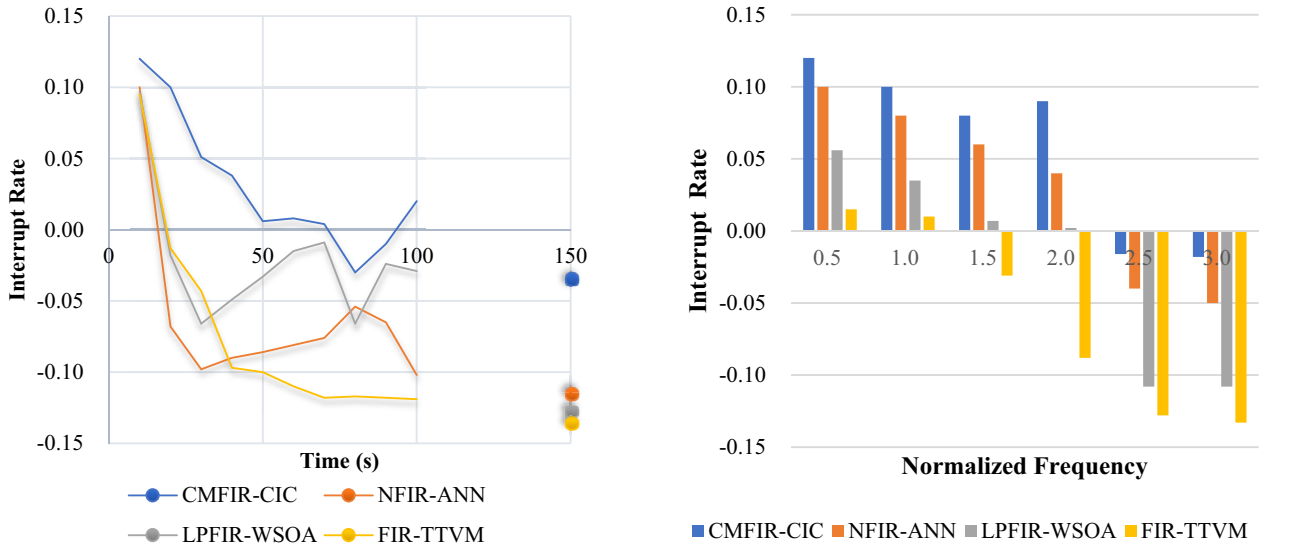

FIGURE 10 Interrupt rate analysis. ANN, artificial neural network; CMFIR, cascade multiple accumulation finite impulse response; FIR, finite impulse response; LPFIR, linear-phase finite impulse response; NFIR, nonlinear finite impulse response; TTVM, two-level transpose Vedic multiplier; WSOA, water-strider optimization algorithm.

TABLE 1 Difference improvements between the proposed and existing methods.

Metrics	CMFIR-CIC	NFIR-ANN	LPFIR-WSOA	FIR-TTVM
Time				
Classifications	28	40	70	96
Noise	0.064	0.054	0.044	0.036
Delay	1.29	1.1	0.86	0.405
Interrupt rate	0.02	−0.102	−0.029	−0.119
Normalized frequency				
Classifications	24	44	74	87
Noise	0.062	0.052	0.038	0.0387
Delay	1.27	1.09	0.657	0.486
Interrupt rate	−0.018	−0.05	−0.108	−0.133

Abbreviations: ANN, artificial neural network; CIC, cascaded integrator comb; CMFIR, cascade multiple accumulation finite impulse response; FIR, finite impulse response; LPFIR, linear-phase finite impulse response; NFIR, nonlinear finite impulse response; TTVM, two-level transpose Vedic multiplier; WSOA, water-strider optimization algorithm.

signal. This filter helped reduce unwanted noise and interference; this resulted in a more reliable signal for further signal processing. To acquire the interference from the signals, this process analyzed the frequency patterns of the data points within discrete intervals. By comparing these characteristics with expected signal intervals, the process separated the interference present in the signal. After identifying the interferences, they are used in the process of the Vedic multiplier to determine the finite intervals. This information was used in the Vedic multiplier to accelerate the operation (Figure 10). Table 1 presents the benchmark differences between the proposed architecture and existing methods numerically.

The proposed TTVM improved classification by 8.68% and reduced noise, delay, and interrupt rates by 9%, 10.47%, and 8.2%, respectively, at various times. The proposed architecture improved classification by 8.2% and reduced noise, delay, and interruptions by 11.9%, 8.66%, and 7.43%, respectively, at the various normalized frequencies.

5 | CONCLUSION

In this study, a two-level transpose Vedic multiplier was employed for the FIR design to handle noise in medical image signals. This optimization procedure was performed independently for intervals with zero noise and those with nonzero noise, which were grouped from the previous transpose levels. This architecture improved the accuracy of signal segregation and classification by modifying the transposition speed for various noise intervals. Therefore, the FIR-TTVM associated the outputs from both zero-noise and nonzero-noise intervals to evaluate the FIR. This aggregation of outputs effectively maintained the reputable signal while effectively reduced the impact of noise and interference. By employing a TTVM, the proposed approach effectively determined and managed noise impulses; this resulted in more precise and reliable medical signal analysis and recognition. Based on the comparative analysis, the proposed method was observed to improve classification by 8.68% and reduce noise by 9% for varying time intervals. However, the proposed method was stuck in the nondifferentiation phase between continuity and discrete intervals. This is particularly true for zero-noise equalization for which further optimization is required. Considering the need for optimization, finite derivative-based incorporations are planned to be assimilated between the equalization processes.

ACKNOWLEDGMENTS

We acknowledge the receipt of this ETRI Journal submission. Thank you for your consideration of our manuscript for publication.

CONFLICT OF INTEREST STATEMENT

The authors have no conflicts of interest to declare. All co-authors have seen and agree with the contents of the manuscript and there is no financial interest to report. We certify that the submission is original work and is not under review at any other publication.

ORCID

Arun Sekar Rajasekaran https://orcid.org/0000-0002-3531-1904

REFERENCES

1. W. Qin, M. T. Gamba, E. Falletti, and F. Dovis, *An assessment of impact of adaptive notch filters for interference removal on the signal processing stages of a GNSS receiver*, IEEE Trans. Aerosp. Electron. Syst. **56** (2020), no. 5, 4067–4082, DOI 10.1109/taes.2020.2990148.

2. J. Wouters, P. Patrinos, F. Kloosterman, and A. Bertrand, *Multi-pattern recognition through maximization of signal-to-peak-interference ratio with application to neural spike sorting*, IEEE Trans. Signal Process. **68** (2020), no. 2020, 6240–6254, DOI 10.1109/tsp.2020.3033973.
3. Z. Xiang, G. Ou, and A. Rashidi, *Robust cascaded frequency filters to recognize rebar in GPR data with complex signal interference*, Autom. Constr. **124** (2021), 103593, DOI 10.1016/j.autcon.2021.103593.
4. C. Hellings, F. Askerbeyli, and W. Utschick, *Two-user SIMO interference channel with treating interference as noise: improper signaling versus time-sharing*, IEEE Trans. Signal Process. **68** (2020), 6467–6480, DOI 10.1109/tsp.2020.3027903.
5. J. Kim, *Signal processing and noise analysis on realistic radiation detector model*, Nucl. Inst. Methods Phys. Res. A **1038** (2022), 166931, DOI 10.1016/j.nima.2022.166931.
6. A. V. Lvov, V. A. Karaseva, O. A. Potapov, and A. M. Sokov, *Adaptive active wideband acoustic noise cancellation system with dynamic calibration*, Acoust Phys. **69** (2023), no. 3, 372–380, DOI 10.1134/s1063771022700026.
7. V. D. Christilda and A. Milton, *Speed, power and area efficient 2D fir digital filter using Vedic multiplier with predictor and reusable logic*, Analog Integr. Circuits Signal Process. **108** (2021), no. 2, 323–333, DOI 10.1007/s10470-021-01853-8.
8. P. Paliwal, J. B. Sharma, and V. Nath, *Comparative study of FFA architectures using different multiplier and adder topologies*, Microsyst. Technol. **26** (2019), no. 5, 1455–1462, DOI 10.1007/s00542-019-04678-8.
9. M. M. da Rosa, P. Ü. da Costa, E. A. C. da Costa, S. J. M. Almeida, G. Paim, and S. Bampi, *Design of a low power and robust VLSI power line interference canceler with optimized arithmetic operators*, Analog Integr. Circuits Signal Process. **112** (2022), no. 2, 247–261, DOI 10.1007/s10470-022-02050-x.
10. P. Chowdari and J. B. Seventline, *VLSI implementation of distributed arithmetic based block adaptive finite impulse response filter*, Mater. Today: Proc. **33** (2020), 3757–3762, DOI 10.1016/j.matpr.2020.06.206.
11. A. Carini, R. Forti, and S. Orcioni, *A polynomial multiple variance method for impulse response measurement*, SSRN Electron. J. **207** (2022), DOI 10.2139/ssrn.4258609.
12. J. Liu and J. Li, *Analytical performance of rank-one signal detection in subspace interference plus Gaussian noise*, IEEE Trans Aerosp. Electron. Syst. **56** (2020), no. 2, 1595–1601, DOI 10.1109/taes.2019.2950075.
13. P. A. Lopes, *Low-delay and low-cost sigma-delta adaptive controller for active noise control*, Circuits Syst. Signal Process. **41** (2022), no. 5, 2988–2999, DOI 10.1007/s00034-021-01913-4.
14. R. Walia and S. Ghosh, *Design of active noise control system using hybrid functional link artificial neural network and finite impulse response filters*, Neural Comput. Applic. **32** (2018), no. 7, 2257–2266, DOI 10.1007/s00521-018-3697-5.
15. J. Brunnström, T. Van Waterschoot, and M. Moonen, *Signal-to-interference-plus-noise ratio based optimization for sound zone control*, IEEE Open J. Signal Process. **4** (2023), 257–266, DOI 10.1109/ojsp.2023.3246398.
16. M. Qiu, Y.-C. Huang, and J. Yuan, *Discrete signaling and treating interference as noise for the Gaussian interference channel*, IEEE Trans. Inf. Theory **67** (2021), no. 11, 7253–7284, DOI 10.1109/tit.2021.3111551.
17. Y. Guo, X. Li, and Q. Meng, *An outlier robust finite impulse response filter with maximum correntropy*, IEEE Access. **9** (2021), 17030–17040, DOI 10.1109/access.2021.3053212.
18. R. Qiu, P. Qu, X. Zheng, and G. Tang, *Design of a finite impulse response filter for rapid single-flux-quantum signal processors based on stochastic computing*, Superconductivity **6** (2023), 100045, DOI 10.1016/j.supcon.2023.100045.
19. R. G. Ramírez-Chavarría and M. Schoukens, *Nonlinear finite impulse response estimation using regularized neural networks*, IFAC-PapersOnLine **54** (2021), no. 7, 174–179, DOI 10.1016/j.ifacol.2021.08.354.
20. E. M. Carlini, F. del Pizzo, G. M. Giannuzzi, D. Lauria, F. Mottola, and C. Pisani, *Online analysis and prediction of the inertia in power systems with renewable power generation based on a minimum variance harmonic finite impulse response filter*, Int. J. Electr. Power & Energy Syst. **131** (2021), 107042, DOI 10.1016/j.ijepes.2021.107042.
21. S. Li, M. Zhang, Y. Ji, Z. Zhang, R. Cao, B. Chen, H. Li, and Y. Yin, *Agricultural machinery GNSS/IMU-integrated navigation based on fuzzy adaptive finite impulse response Kalman filtering algorithm*, Comput. Electron. Agric. **191** (2021), 106524, DOI 10.1016/j.compag.2021.106524.
22. Y. Wu, Z. Mao, Y. Li, and S. Liu, *Finite impulse response filter based fault estimation with computational efficiency for linear discrete time-varying systems subject to multiplicative noise*, J. Frankl Instit. **359** (2022), no. 6, 2737–2754, DOI 10.1016/j.jfranklin.2022.01.044.
23. Y. Pu, Y. Yang, Y. Rong, and J. Chen, *Expectation maximization algorithm for time-delay output-error models based on finite impulse response method*, Int. J. Contr. Autom. Syst. **19** (2021), no. 12, 3914–3923, DOI 10.1007/s12555-021-0241-7.
24. X. Wang, Y. Rong, C. Wang, F. Ding, and T. Hayat, *Gradient-based iterative parameter estimation for a finite impulse response system with saturation nonlinearity*, Int. J. Contr. Autom. Syst. **20** (2022), no. 1, 73–83, DOI 10.1007/s12555-020-0872-0.
25. V. Jain, P. Chakrabarti, M. Mitolo, Z. Leonowicz, M. Jasinski, A. Vinogradov, and V. Bolshev, *A power-efficient multichannel low-pass filter based on the cascaded multiple accumulate finite impulse response (CMFIR) structure for digital image processing*, Circuits Syst. Signal Process. **41** (2022), no. 7, 3864–3881, DOI 10.1007/s00034-022-01960-5.
26. R. Karthick, A. Senthilselvi, P. Meenalochini, and S. Senthil Pandi, *Design and analysis of linear phase finite impulse response filter using water strider optimization algorithm in FPGA*, Circuits Syst. Signal Process. **41** (2022), no. 9, 5254–5282, DOI 10.1007/s00034-022-02034-2.
27. M. M. Ganatra and C. H. Vithalani, *FPGA design of a variable step-size variable tap length denlms filter with hybrid systolic-folding structure and compressor-based booth multiplier for noise reduction in ECG signal*, Circuits Syst. Signal Process. **41** (2022), no. 6, 3592–3622, DOI 10.1007/s00034-021-01933-0.
28. Y. Zhang, X. Zhu, A. Liu, and S. Yi, *A Lorentzian-IP norm regularization based algorithm for recovering sparse signals in two types of impulsive noise*, J. Comput. Appl. Math **430** (2023), 115251, DOI 10.1016/j.cam.2023.115251.
29. Y. MejiaCruz and B. T. Davis, *Event reconstructing adaptive spectral evaluation (ERASE) approach to removing noise in structural acceleration signals*, Exp. Tech. **47** (2022), no. 4, 827–837, DOI 10.1007/s40799-022-00598-x.
30. J. Prasad, D. Mohana Geetha, and K. Srinivasan, *Experimental setup of stretchable arid dry pad sensors for the signal*

acquisition FIR filter design using Vedic approach, Measurement **141** (2019), 209–216.

31. Y. Yamanashi, T. Kainuma, N. Yoshikawa, I. Kataeva, H. Akaike, A. Fujimaki, and M. Hidaka, *100 GHz demonstrations based on the single-flux-quantum cell library for the 10 kA/cm^2 Nb multi-layer process*, IEICE Trans. Electron. **93** (2010), no. 4, 440444.

AUTHOR BIOGRAPHIES

Joghee Prasad obtained his B.E. degree in Electronics and Communication Engineering from Sengunthar Engineering College, Tiruchengode, Tamil Nadu, India, in 2005 and his M.E. degree in Applied Electronics from Bannari Amman Institute of Technology, Sathyamangalam, Tamil Nadu, India, in 2008. He received his Ph.D. degree from the Faculty of Information and Communication Engineering at Anna University in 2020. He is currently an assistant professor in the Electronics and Communication Engineering Department at the KPR Institute of Engineering and Technology, Coimbatore, Tamil Nadu, India. He has more than 13 years of teaching experience. His primary research interests include VLSI, FPGA-based architectural design, and signal processing. He is a lifetime member of the ISTE and the International Association of Engineers.

Arun Sekar Rajasekaran received his Bachelor's degree in Electronics and Communication Engineering from Sri Ramakrishna Engineering College in 2008 and his Master's degree in VLSI Design and his Doctorate of Philosophy in Low Power VLSI design from Anna University, Chennai, Tamilnadu, India, in 2019. He is currently working as an associate professor in the Department of Electronics and Communication Engineering at SR University, Warangal, Telangana, India. Currently, he is advising four Ph.D. students at SR University. He has nearly 14 years of teaching experience and has an Anna University Guideship. He has published more than 27 papers in international conferences and 30 papers in indexed SCIE Journals, including IEEE Transactions on Industrial Informatics, Microprocessor and Microsystems, Computers and Electrical Engineering (Elsevier), IET Communications, Information and Security, and IEEE Access and Concurrency and Computation (Wiley). His areas of interest are low-power VLSI design, network security, blockchain, body area networks, fog computing, and image processing. He is a member of IEEE, ISTE, IETE (Fellow), ISRD, and IEANG.

J Ajayan received his B. Tech degree in Electronics and Communication Engineering from Kerala University in 2009, Kerala, India, and his M. Tech and Ph.D. degrees in Electronics and Communication Engineering from Karunya University, Coimbatore, Tamil Nadu, India, in 2012 and 2017, respectively. He is currently a professor at the Department of Electronics and Communication Engineering at SR University in Telangana, India. He has published more than 170 research articles (including 90 SCI articles) in various journals (including IEEE Transactions on Electron Devices) and international conferences. He has published three books and more than 20 chapters and has three patents. He is a reviewer for more than 30 journals from various publishers, including IEEE, IET, Elsevier, and Springer. His areas of interest include microelectronics, semiconductor devices, nanotechnology, RF integrated circuits, and photovoltaics.

Kambatty Bojan Gurumoorthy received his B.E. degree in Electronics and Communication Engineering from Sasurie College of Engineering, Erode, Tamil Nadu, India, in 2006. He obtained his M.E. degree in communication systems from Anna University, Coimbatore, Tamil Nadu, India, in 2011. In 2019, he completed his Ph.D. in Information and Communication Engineering at Anna University, Chennai, Tamil Nadu, India. He is currently working as an associate professor in the Department of Electronics and Communication Engineering at the KPR Institute of Engineering and Technology, Coimbatore, India. He has more than 11 years of teaching experience and 3 years of industry experience. His research interests include mobile ad hoc networks, wireless sensor networks, and wireless communications. He is a lifetime member of the ISTE.

How to cite this article: J. Prasad, A. S. Rajasekaran, J. Ajayan, and K. B. Gurumoorthy, *Finite impulse response design based on two-level transpose Vedic multiplier for medical image noise reduction*, ETRI Journal **46** (2024), 619–632. DOI 10.4218/etrij.2023-0335.