Research Article

Accelerating Convergence in LMS Adaptive Filters Using Particle Swarm Optimization: A Hybrid Approach for Real-Time Signal Processing

Muhammed Davud[1a]

[1]Department of Software Engineering, Istinye University, 34010, Istanbul, Türkiye

muhammed.davud@istinye.edu.tr

Abstract Adaptive filtering is essential for control systems, system identification, and noise cancellation, especially when using Least Mean Square (LMS) algorithms. Although LMS-based approaches are popular because they are straightforward and efficient, they frequently have sluggish convergence and numerical instability. This study offers a hybrid framework that integrates Particle Swarm Optimization (PSO) with LMS variations, such as ZA-LLMS, RZA-LLMS, ZA-VSS-LMS, and RZA-VSS-LMS, in order to overcome these issues. PSO outperforms conventional methods in terms of mean square error (MSE) performance and convergence speed by dynamically modifying weight coefficients. The proposed system was evaluated utilizing synthetic noise models, such as Additive White Gaussian Noise (AWGN) and Colored Gaussian Sequence (CGS), in addition to MRI scan restoration. The results show that the PSO-enhanced LMS versions reduce the number of required iterations by up to 67% while improving filtering accuracy. Under Gaussian noise, the PSO-RZA-VSS-LMS approach remarkably obtained a Peak Signal-to-Noise Ratio (PSNR) of 24.58 dB while its non-PSO equivalent only obtained 20.51 dB. In a similar line, PSO-RZA-LLMS attained 17.12 dB for Salt & Pepper noise, at 15.32 dB, surpassing the baseline RZA-LLMS. These results show the flexibility of the PSO-driven approach toward different noise distributions. Apart from raising filtering accuracy, PSO integration significantly accelerates convergence without sacrificing signal integrity. The results show that the proposed method offers a practical and efficient replacement for real-time adaptive filtering systems including medical imaging, speech processing, and high-speed communication networks.

Keywords: Adaptive filtering, Hybrid Optimization, LMS, Particle Swarm, Image Denoising.

I. INTRODUCTION

With a wide range of applications, adaptive filtering (AF) is a well-established and extensively researched topic [1]–[4]. The fundamental component of an adaptive filter is a linear filter with movable parameters that are updated dynamically through optimization. Because of its adaptability, AF can be set up in a variety of ways to meet the needs and demands of diverse applications. The Finite Impulse Response (FIR) filter is a popular setup in which a set of tunable coefficients determines the output by means of a weighted sum of the input samples [5]. The Infinite Impulse Response (IIR) filter is another method that combines feedback into its construction. With fewer coefficients than FIR filters, IIR filters can computationally be more efficient while nevertheless attaining comparable filtering characteristics [6]. This efficiency makes them especially valuable in real-time applications and situations with limited memory capacity. Apart from these basic setups, various other adaptive filtering methods provide special signal-processing capacity. One type of block adaptive filter divides the input signal into blocks and independently updates the filter coefficients for each block. This method allows more exact adaptation and tracking of changes, hence it is particularly useful for managing signals whose attributes change with time [7]. Another fascinating configuration, whereby a filter bank divides the input signal into many frequency subbands, is the subband adaptive filter. When different frequency components require distinct filtering techniques in applications like audio coding and noise cancellation, this approach is very helpful since an adaptive filter handles each subband individually [8]. Still another sometimes used model is the adaptive Radial Basis Function (RBF). Particularly effective for nonlinear and non-Gaussian signals, it bases the output on an array of RBFs. Common uses of this approach in systems identification, pattern recognition, and time-series forecasting [9]. Designed particularly to remove undesirable noise from a target signal, the Adaptive Noise Canceller (ANC) is a specific use of adaptive filtering [10]. The ANC uses two inputs: the main input has the intended signal mixed with noise, while the reference input has a noise component linked with the noise in the main input. An adaptive filter then continuously adjusts to lower the correlation between the projected noise and the primary input. By eliminating this estimated noise from the main input, the ANC can dynamically adapt to match changing noise levels, thereby enhancing signal

Accelerating Convergence in LMS Adaptive Filters Using Particle Swarm Optimization...

ECJSE

clarity and intelligibility. Usually, the convergence speed and the ability to obtain the lowest Mean Square Error (MSE) define the performance of an AF method. One of the fundamental strategies for optimizing AF performance is the Wiener filter, which assumes prior knowledge of the input signal [11]. Nonetheless, some adaptive filtering techniques, such as Least Mean Square (LMS), are often applied despite not requiring this prior knowledge because of their simplicity and ease of implementation [12]. The typical LMS technique has limits notwithstanding these benefits. When the input sequence lacks enough excitation, it can suffer from unbounded parameter estimations, which cause numerical instability and poor performance due to excessively high prediction errors [13]. Several variants of the LMS method have been proposed to handle these problems. The Leaky LMS (LLMS) model is one such development, since it reduces parameter drift, thereby stabilizing the system [14]. Proposed to improve sparsity control and reduce update complexity, the Zero-Attracting LLMS (ZA-LLMS) method is a further development [15]. Convergence speed and steady-state performance studies reveal that ZA-LLMS outperforms both conventional LLMS and ZA-LMS algorithms [16]. The Reweighted Zero-Attracting LLMS (RZA-LLMS) algorithm, introduced in [17], shows even greater performance in adaptive filtering applications, further enhancing the efficiency of ZA-LLMS.

In a different track, it is recognized that the Mean Square Error (MSE) is directly related to the adaptation step size in terms of performance, and that it determines the convergence rate to optimal weights. Usually, a small step size is recommended to attain a low final excess MSE, but this results in sluggish convergence, which is, in many circumstances, unsatisfactory. Variable Step Size (VSS) adaptation is one method for achieving greater performance than the LMS algorithm, which uses a constant fixed step size. In the Variable Step Size LMS (VSS-LMS) method, the step size is large when the LMS weights are suboptimal, and it decreases as the weights approach optimality. For changing the step size, several criteria are available, including the squared estimation error [18], the gradient of the squared estimation error with respect to the step [19], and the autocorrelation of subsequent estimation errors [20]. In [21], the authors introduced the Zero-Attracting VSS-LMS (ZA-VSS-LMS) algorithm, which outperforms the standard VSS-LMS algorithm. To minimize the update formula, they added the ℓ_1 norm of the filter coefficient vector in the ZA-VSS-LMS cost function, particularly when the majority of coefficients are zero. Nevertheless, the ZA-VSS-LMS algorithm's shrinkage does not distinguish between zero taps and non-zero taps. In [21], the Reweighted Zero-Attracting VSS-LMS (RZA-VSS-LMS) method is presented to address this issue.

Although the LMS method was utilized to generate a reference current because of its ease of implementation, there was a trade-off between its convergence rate and minimum MSE [22]. In the past decade, convex AF combinations have emerged as a way to avoid this trade-off [23]–[25]. The combination of convex AFs has improved both stationary and non-stationary performance [25]. Typically, two AFs are used to construct the convex combination: one provides rapid convergence with a high steady-state output error, while the other has slow convergence and a small MSE [26]. This strategy offers a solution to the trade-off problem by merging the faster-convergent and lower-MSE features of the two filtering techniques [25]. Particle Swarm Optimization (PSO) [27] is a swarm intelligence-based metaheuristic optimization method that can be applied in adaptive filters (AFs). To identify the best solution inside a search space, the PSO technique has been extensively used in several disciplines, including signal processing, control systems, and machine learning [28], [29]. In signal processing, PSO has been applied to create adaptive filters for addressing noise reduction concerns [30]. Furthermore, the integration of LMS with PSO has been suggested in [31] to improve its performance.

This work presents hybrid Adaptive Noise Canceller (ANC) algorithms using the Particle Swarm Optimization (PSO) approach to improve adaptive filtering performance. To enhance adaptive filtering performance, we propose the PSO-ZA-LLMS, PSO-RZA-LLMS, PSO-ZA-VSS-LMS, and PSO-RZA-VSS-LMS adaptive filtering techniques. While ZA-LLMS, RZA-LLMS, ZA-VSS-LMS, and RZA-VSS-LMS are used to address the local convergence issue that may result from employing PSO, these algorithms utilize PSO during the iterative process to seek the best solution. In terms of both convergence speed and performance, experimental results demonstrate that the proposed systems, PSO-RZA-LLMS and PSO-RZA-VSS-LMS, outperform the conventional RZA-LLMS and RZA-VSS-LMS filters.

The paper is organized as follows. Section II provides a review of different models of LMS adaptive filters, including LMS, LLMS, ZA-LLMS, RZA-LLMS, VSS-LMS, ZA-VSS-LMS, and RZA-VSS-LMS algorithms. In Section III, Particle Swarm Optimization (PSO) is reviewed as an optimization technique that can be used in conjunction with adaptive filters. Section IV presents the proposed method, which is a combination of PSO and the weighted zero-attracting VSS-LMS (RZA-VSS-LMS) algorithm. The simulation results and discussion are presented in Section V, where the performance of the proposed method is compared with other traditional adaptive filter algorithms. The application of the proposed algorithms to 2D images is explored in Section VI. Section VII provides an in-depth analysis of the proposed algorithms, evaluating their effectiveness and limitations while discussing potential future research directions. Finally, in Section VIII, the paper concludes by summarizing the main findings and contributions of the proposed method, and by highlighting potential directions for future research in this area.

II. VARIOUS MODELS OF LMS

This paper uses uppercase boldface letters to show matrices, while lowercase boldface letters are used to show vectors. The transpose operator is shown as $(\cdot)^T$, and the inverse operator is shown as $(\cdot)^{-1}$. The operator $\| . \|_1$ stands for the $\ell 1$ norm, and $tr(\cdot)$ stands for the trace operator. Also, $E[\cdot]$ is the symbol for the expectation operator. The basic ANC filter considered in

this paper is depicted in Figure 1. The input signal that serves as the primary input to the ANC system is the noisy signal $d(\omega)$, which is depicted as:

$$d(\omega) = s(\omega) + v(\omega), \tag{1}$$

where the components of the input signal $d(\omega)$ are the noise-free signal $s(\omega)$ and the added noise signal $v(\omega)$, which are assumed to be uncorrelated in time. The reference input signal $x(\omega)$ provided to the adaptive filter is a noise signal that is correlated with $v(\omega)$. By processing $x(\omega)$, the adaptive filter calculates an estimation of the noise signal as:

$$y(\omega) = \theta^T(\omega)x(\omega), \tag{2}$$

where $\theta = [z_0, z_1, \cdots, z_{N-1}]^T$ is the filter coefficient vector with length N taps, $\mathbf{x}(\omega) = [x(\omega), x(\omega-1), \cdots, x(\omega-N+1)]^T$. Therefore, the error signal is calculated as:

$$e(\omega) = d(\omega) - y(\omega). \tag{3}$$
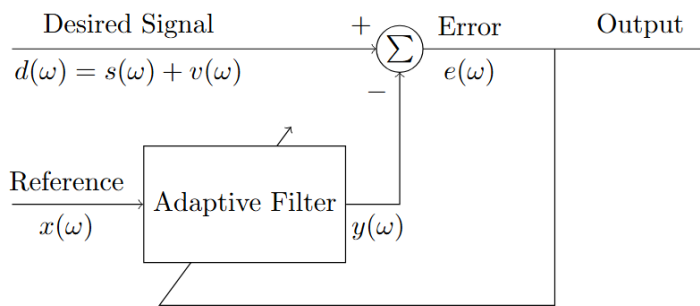
Figure 1. Block diagram of the basic Adaptive Noise Canceller (ANC).

A. LEAST MEAN SQUARE (LMS)

The objective of LMS-type filters is to gradually estimate the unknown coefficient vector using the reference signal $x(\omega)$ and the target signal $d(\omega)$. The estimated coefficient vector of the adaptive filter is indicated as $\theta(\omega)$ for each iteration. The typical LMS defines the cost function $L_1(\omega)$ as:

$$L_1(\omega) = 0.5\, e^2(\omega), \tag{4}$$

where

$$e(\omega) = d(\omega) - y(\omega) = d(\omega) - \theta^T(\omega)x(\omega), \tag{5}$$

is the instantaneous error. The filter coefficient vector is then updated by

$$\theta(\omega+1) = \theta(\omega) - \nabla\frac{\partial L_1(\omega)}{\partial\theta(\omega)} = \theta(\omega) + \nabla e(\omega)x(\omega), \tag{6}$$

where ∇ is the step size that controls how the LMS algorithm gets to the steady state. Considering that R stands for the covariance matrix of $\mathbf{x}(\omega)$, and λ_{max} stands for its largest eigenvalue of $\boldsymbol{R}$. The LMS algorithm's generally known convergence criterion is

$$0 < \nabla < \frac{1}{\lambda_{max}}, \tag{7}$$

and with the premise of independence, the steady-state excess MSE is

$$P_{\text{ex}}(\infty) = \lim_{n\to\infty} E\left[\left((\theta(\omega)-\theta)^T x(\omega)\right)^2\right] = \frac{\eta}{2-\eta}P_0, \tag{8}$$

where P_0 is the power of observation noise and calculated as:

$$P_0 = E[v^2(\omega)], \tag{9}$$

and

$$\eta = \text{tr}(R(I - \nabla R)^{-1}). \tag{10}$$

B. LEAKY LMS (LLMS)

The LLMS algorithm redefine the cost function $L_1(\omega)$ shown in (4) as $L_2(\omega)$, and

$$L_2(\omega) = 0.5e^2(\omega) + \gamma\theta^T(\omega)\theta(\omega), \tag{11}$$

where γ is the leakage factor, which is a positive parameter. Therefore, the minimum of $L_2(\omega)$ is found using a recursive process as:

$$\theta(\omega + 1) = \theta(\omega) - \nabla\frac{\partial L_2(\omega)}{\partial\theta(\omega)} = (1 - \nabla\gamma)\theta(\omega) + \nabla e(\omega)x(\omega). \tag{12}$$

C. ZERO-ATTRACTING LLMS (ZA-LLMS)

The ZA-LLMS algorithm redefine the cost function $L_1(\omega)$ shown in (4) as $L_3(\omega)$, and

$$L_3(\omega) = 0.5e^2(\omega) + \gamma\theta^T(\omega)\theta(\omega) + \gamma_3'|\theta(\omega)|, \tag{13}$$

where γ_3' is a positive constant, and $|\theta(\omega)| = \sqrt{\sum_{i=1}^N z_i^2}$. Then, the ZA-LLMS algorithm's update equation becomes

$$\theta(\omega + 1) = \theta(\omega) - \nabla\frac{\partial L_3(\omega)}{\partial\theta(\omega)} = (1 - \nabla\gamma)\theta(\omega) + \nabla e(\omega)x(\omega) - \rho\,\text{sgn}[\theta(\omega)], \tag{14}$$

where $\rho = \nabla\gamma_3'$, and $sgn(.)$ is a component-wise sign function defined by

$$\text{sgn}(x) = \frac{x}{|x|}, \; x \neq 0; \; 0, \; x = 0. \tag{15}$$

D. REWEIGHTED ZERO-ATTRACTING LLMS (RZA-LLMS)

The RZA-LLMS algorithm define the cost function $L_4(\omega)$ as:

$$L_4(\omega) = 0.5\,e^2(\omega) + \gamma\theta^T(\omega)\theta(\omega) + \gamma_4'\sum_{i=1}^N \log\left(1 + \frac{|z_i|}{\zeta_4'}\right), \tag{16}$$

where γ_4' and ζ_4' are positive constants. Then, the RZA-LLMS algorithm's update equation becomes

$$\theta(\omega + 1) = \theta(\omega) - \nabla\frac{\partial L_4(\omega)}{\partial\theta(\omega)} = (1 - \nabla\gamma)\theta(\omega) + \nabla e(\omega)x(\omega) - \rho_4\frac{\text{sgn}[\theta(\omega)]}{1 + \zeta_4|\theta(\omega)|}, \tag{17}$$

where $\rho_4 = (\nabla\gamma_4'/\zeta_4')$ is the zero-attracting parameter $\zeta_4 = (1/\zeta_4')$, and sgn$(.)$ is the same as in (15). Upon comparing (12) and (17), it is evident that the RZA-LLMS algorithm contains an additional term, i.e., $-\rho_4(sgn[\theta(\omega)])/(1 + \zeta_4|\theta(\omega)|)$, which always pulls the tap coefficients towards zero. This term, known as a zero-attractor, is governed by ρ_4 to accelerate convergence when the preponderance of system coefficients is zero.

E. VARIABLE STEP SIZE LMS (VSS-LMS)

A different algorithm that uses variable step-size has been proposed in [18] and called variable-step-size LMS (VSS-LMS). In this algorithm, the step size (i.e. ∇) is recalculated in each iteration as:

$$\nabla'(\omega + 1) = \alpha_5\nabla'(\omega) + \kappa_5 e^2(\omega), \tag{18}$$

where $0 < \alpha_5 < 1$, and $\kappa_5 > 0$. Therefore

$$\nabla(\omega) = \nabla_{max} \; (\nabla'(\omega + 1) > \nabla_{max}), \; \nabla_{min} \; (\nabla'(\omega + 1) < \nabla_{min}), \; \nabla'(\omega + 1) \; \text{(otherwise)}, \tag{19}$$

where $0 < \nabla_{min} < \nabla_{max}$, and $\nabla'(0)$ has no restrictions (although ∇_{max} could be better choice, as indicated in [18]). Equation (18) makes it clear that the step-size is always positive and is defined by the prediction error e(ω), α_5, and κ_5. A higher step-size and quicker tracking are often produced by a significant initial prediction error. Reduced misadjustment results from a reduction in step size as the prediction error becomes less. According to [18], ∇_{max} is chosen in a manner that assures a limited mean-square error (MSE) as:

$$\nabla_{max} \le \frac{2}{3\,\text{tr}(E[\boldsymbol{R}])}. \tag{20}$$

F. ZERO-ATTRACTING VSS-LMS (ZA-VSS-LMS)

In this algorithm, the $\ell 1$ norm of the filter coefficient vector is added to the square of e in (4) to be as follows:

$$L_6(\omega) = 0.5e^2(\omega) + \varphi_6\|\theta(\omega)\|_1, \tag{21}$$

where φ_6 is a positive constant. Therefore, θ is updated as follows:

$$\theta(\omega+1) = \theta(\omega) - \frac{\nabla(\omega)}{2}\frac{\partial L_6(\omega)}{\partial\theta(\omega)} = \theta(\omega) + \nabla(\omega)e(\omega)x(\omega) - \beta_6(\omega)f\big(\theta(\omega)\big) \tag{22}$$

where $\beta_6(\omega) = \varphi_6\nabla(\omega)$ and $f\big(\theta(\omega)\big)$ is the sign function given in (15). When comparing (6) and (22), it is evident that (22) has an additional term $\big(-\beta_6(\omega)f\big(\theta(\omega)\big)\big)$. The tap coefficients are always pushed towards zero by this additional term. When the majority of the coefficients in θ are zero, the zero-attractor in (22) accelerates the convergence process. The method is known as the zero-attracting VSS-LMS (ZA-VSS-LMS) because the intensity of the zero-attractor is adjusted by $\beta_6(\omega)$.

G. REWEIGHTED ZERO-ATTRACTING VSS-LMS (RZA-VSS-LMS)

The shrinkage mechanism of the ZA-VSS-LMS algorithm handles zero and non-zero taps equally, which has a negative impact on performance for less sparse systems. When working with less sparse systems, weighting the ZA term in (22) may enhance the algorithm's performance [19]. Equation (4)'s cost function L_1 is redefined as:

$$L_7(\omega) = 0.5\,e^2(\omega) + \varphi_7\sum_{i=1}^{N}\log\left(1+\frac{|z_i|}{\zeta_7}\right), \tag{23}$$

where φ_7 and ζ_7 are positive constants. Then, the same as before, by applying the gradient method we get

$$\theta(\omega+1) = \theta(\omega) + \nabla(\omega)e(\omega)x(\omega) - \rho_7(\omega)\frac{\text{sgn}[\theta(\omega)]}{1+\zeta_7'|\theta(\omega)|}, \tag{24}$$

where $\rho_7(\omega) = (\nabla(\omega)\varphi_7)/(\zeta_7)$, and $\zeta_7' = 1/\zeta_7$. The RZA-VSS-LMS algorithm introduces a weighted zero-attracting effect that only affects the taps whose magnitudes are comparable to $1/\zeta_7'$, while taps with much greater magnitudes experience little shrinkage. This property leads to a reduction in bias compared to other algorithms.

III. Particle Swarm Optimization (PSO)

In the conventional PSO model, an individual is represented as a particle in a D-dimensional space, where its location and velocity are denoted by $C_i = (C_{i1}, C_{i2}, \ldots, C_{iD})$ and $S_i = (S_{i1}, S_{i2}, \ldots, S_{iD})$, respectively. The movement of each particle is governed by:

$$S_{id} = \epsilon \cdot S_{id} + c_1 \cdot \text{rand}() \cdot (U_{id} - C_{id}) + c_2 \cdot \text{rand}() \cdot \big(U_g - C_{id}\big), \tag{25}$$

$$C_{id} = C_{id} + S_{id}. \tag{26}$$

The inertia weight ϵ controls the trade-off between global exploration and local refinement, while the acceleration coefficients c_1 and c_2 determine how much influence the particle's personal best and global best have on velocity updates. In the given set of equations, c_1 and c_2 are positive constants, typically both set to 2 in this study [32], and rand() is a random function within the range [0, 1]. The vector $U_i = (U_{i1}, U_{i2}, \ldots, U_{iD})$ represents the best prior position of particle i, known as $pbest$, which corresponds to the highest fitness value. Similarly, $gbest$ indicates the position of the best particle in the entire population, defined by the vector $U_g = \big(U_{g1}, U_{g2}, \ldots, U_{gD}\big)$. Here, C_{id}, S_{id}, and U_{id} refer to the dth dimension of the vectors C_i, S_i, and U_i, respectively. The inertia weight ϵ is a crucial parameter that accelerates the Particle Swarm Optimization (PSO) convergence speed. It is dynamically adjusted using the formula [33]:

$$\epsilon = \epsilon_{max} - \frac{iter \cdot (\epsilon_{max} - \epsilon_{min})}{iter_{PSO}}, \tag{27}$$

where ϵ_{max} and ϵ_{min} are typically set to 0.9 and 0.4, respectively. The iteration number iter represents the current iteration, and $iter_{PSO}$ is the maximum number of iterations, and set to 200.

A. COMPARISON WITH OTHER OPTIMIZATION TECHNIQUES

The proposed PSO-based LMS algorithm and several commonly used optimization techniques in adaptive filtering, including

Genetic Algorithms (GA), Artificial Bee Colony (ABC), Differential Evolution (DE), Artificial Neural Networks (ANN), and Reinforcement Learning (RL), are compared below in terms of their key strengths and limitations.

- **Genetic Algorithms (GA):** Natural selection serves as the inspiration for evolutionary optimization methods known as genetic algorithms (GA). They are frequently employed in adaptive filtering for global optimization, particularly in situations involving a vast and intricate search space. Even while GA is reliable and capable of solving non-linear and non-convex optimization problems, it is typically more computationally costly and has a slower rate of convergence than PSO, especially in real-time applications [34], [35].
- **Artificial Bee Colony (ABC):** Artificial Bee Colony (ABC) is a swarm intelligence-based optimization algorithm inspired by the foraging behavior of honeybees. It is known for its simplicity and effectiveness in solving complex optimization problems. ABC is highly effective in exploring the search space and can handle multi-modal optimization problems better than PSO. However, ABC can be slower in convergence compared to PSO, especially in high-dimensional problems [36], [37].
- **Differential Evolution (DE):** Differential Evolution (DE) is a population-based optimization method that evolves solutions over generations using crossover, mutation, and selection procedures. DE is renowned for its quick convergence and great efficacy in resolving continuous optimization issues. Nevertheless, DE necessitates meticulous adjustment of its parameters (such as crossover rate and mutation factor), which might be difficult in adaptive filtering applications [38], [39].
- **Artificial Neural Networks (ANN):** Deep learning and other optimization methods based on Artificial Neural Networks (ANN) are being utilized more and more in adaptive filtering for challenging signal processing problems. ANN is useful for tasks like system identification and noise suppression since it can represent extremely non-linear systems. However, ANN is less appropriate for real-time applications than PSO since it needs a lot of data for training and is computationally costly [40], [41].
- **Reinforcement Learning (RL):** A machine learning method called reinforcement learning (RL) teaches an agent to make decisions by interacting with its surroundings. It has been used for dynamic optimization in adaptive filtering. Real-time adaptive filtering can benefit from RL's high degree of adaptability and ability to manage dynamic settings. In contrast to PSO, RL is more complicated to implement and demands a large amount of processing power [42], [43].
- **Convex Combination Adaptive Filters:** Convex combination adaptive filters improve performance in both stationary and non-stationary environments by combining two or more adaptive filters. These filters are appropriate for dynamic contexts because they may strike a balance between minimal steady-state error and quick convergence. However, they can be computationally costly and necessitate careful combination parameter tweaking [44], [45].
- **Bayesian Adaptive Filtering:** Bayesian adaptive filtering estimates a system's state using probabilistic models. It is especially helpful in applications where users already know something about the system. Bayesian techniques can provide probabilistic estimates of the filter parameters and are very good at managing uncertainty. But Bayesian techniques are computationally demanding and necessitate system information, which is not always available [46], [47].

As shown in Table 1, the suggested PSO-based LMS algorithm strikes a balance between computational efficiency, convergence speed, and ease of implementation, making it particularly well-suited for real-time adaptive filtering applications, even though each optimization technique has advantages of its own.

A. INTEGRATION OF PSO WITH LMS

The goal of integrating PSO with the LMS approach is to optimize the filter coefficients in a dynamic manner. The filter coefficients are modeled as particles in a multi-dimensional optimization space using the PSO framework in this method. In order to converge on an ideal set of weights that minimize the Mean Square Error (MSE) between the intended and actual output of the filter, these particles iterate through positions and velocities. Every particle is a potential solution for the filter coefficients, and the weight space may be explored effectively thanks to the iterative process of updating locations and velocities. The Mean Square Error (MSE), which is minimized during the optimization process, is used as the fitness function to evaluate the PSO's performance. The MSE can be written as follows:

$$MSE = 0.5\, e^2(\omega), \tag{28}$$

where $d(\omega)$ is the target signal and $y(\omega) = \theta^T(\omega)x(\omega)$ is the output of the LMS filter. The error signal is represented by $e(\omega)$, which is defined as $e(\omega) = d(\omega) - y(\omega)$.

IV. PROPOSED METHOD

By helping to choose the best weights for each iteration, the PSO optimizer is incorporated into the ZA-LLMS, RZA-LLMS, ZA-VSS-LMS, and RZA-VSS-LMS algorithms, which is a revolutionary method that accelerates the convergence process. PSO is specifically used to identify the optimal weights for each adaptive filter that minimize the cost function.

Table 1. Comparison of Various Optimization Techniques in Adaptive Filtering: Strengths, Weaknesses, and Applicability.

Optimization Technique	Strengths	Weaknesses	Applicability
PSO-based LMS	• Fast convergence. • Easy to implement. • Suitable for real-time systems.	• May get stuck in local minima. • Requires tuning of hyperparameters.	• Real-time signal processing. • Noise cancellation. • System identification.
Genetic Algorithms (GA)	• Robust for global optimization. • Handles non-linear problems.	• Computationally expensive. • Slower convergence.	• Large search spaces. • Non-convex optimization problems.
Artificial Bee Colony (ABC)	• Effective in exploring search space. • Handles multi-modal problems.	• Slower convergence in high-dimensional problems.	• Complex optimization problems. • Multi-modal optimization.
Differential Evolution (DE)	• Fast convergence. • Effective for continuous optimization.	• Requires careful parameter tuning	• Continuous optimization problems. • High-dimensional problems.
Artificial Neural Networks (ANN)	• Models non-linear systems. • Effective for complex tasks.	• Requires large datasets. • Computationally expensive.	• Noise cancellation. • System identification. • Non-linear signal processing.
Reinforcement Learning (RL)	• Highly adaptive. • Suitable for dynamic environments.	• Complex to implement. • Requires significant computational resources.	• Dynamic environments. • Real-time adaptive filtering.
Convex Combination Adaptive Filters	• Balances fast convergence and low MSE. • Suitable for dynamic environments.	• Requires careful tuning of combination parameters. • Computationally expensive.	• Dynamic environments. • Real-time signal processing.
Bayesian Adaptive Filtering	• Handles uncertainty well. • Provides probabilistic estimates.	• Computationally intensive. • Requires prior knowledge of the system.	• Systems with prior knowledge. • Probabilistic signal processing.

This optimization process is repeated for each iteration until the adaptive filters converge to their optimal weights. A comprehensive flowchart of the proposed method is depicted in Figure 2. The flowchart shows the steps involved in the PSO-based optimization of the adaptive filters, including the initialization of the PSO algorithm, the calculation of the fitness function, the updating of the particle velocities and positions, and the termination criterion. The flowchart denotes four identified points inside dashed circles, which are referred to as point 1, point 2, point 3, and point 4 in this context. It is evident that the proposed algorithm's complete iteration involves the processes from point 1 to point 3, which comprises two consecutive phases. The first phase, from point 1 to point 2, entails applying PSO before each iteration of the AF algorithm. The second phase, from point 2 to point 3, involves executing one iteration of the AF algorithm on the optimized data. Consequently, the value of θ at each point is as follows:

- Point 1: The value from the preceding iteration, or θ, starting value.
- Point 2: The first value of θ or the value obtained in the last iteration PSO-optimized.
- Point 3: The value of θ derived from one full optimization then adaption iteration.

- Point 4: The last value of θ (or, the filter) applied to data filtering.
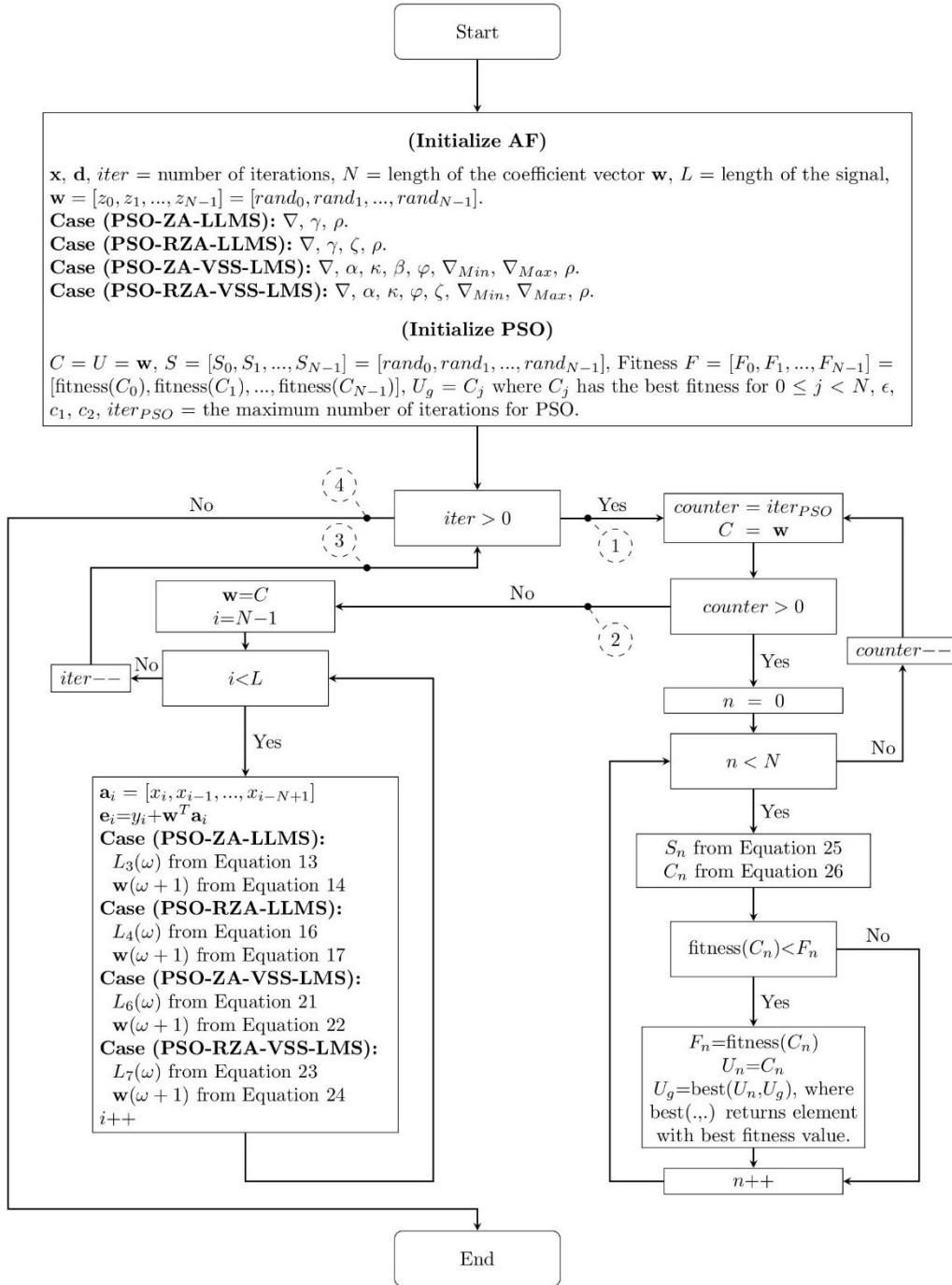
Figure 2. Flow chart for proposed algorithms.

V. Simulation Results and Discussion

The suggested PSO-based algorithms are thoroughly compared in this part with those of the ZA-LLMS, RZA-LLMS, ZA-VSS-LMS, and RZA-VSS-LMS algorithms. Two different noise categories were used in order to verify the validity and robustness of the acquired data by means of comprehensive examinations. Characterised by a zero mean, a variance of 0.551, and a Signal-to-Noise Ratio (SNR) of 25, the first category is Additive White Gaussian Noise (AWGN). The second category consists in the use of Correlated Gaussian Sequence (CGS), commonly known as Colored Noise, which offers a more delicate and complex noise profile. In this part of the experiments, it is assumed that the input signal follows a white Gaussian distribution with a zero mean and a variance of one. In accordance with the symbols used in the equations in this paper, Table 2 presents a summary of the selected values from the literature, which are used as fixed values for models without PSO

optimization and as initial values for models utilizing PSO. It also includes the parameter ranges used to constrain the optimization process.

Table 2. Suggested Ranges and Optimal Values for Parameters.

Parameter	Description	Suggested Range	Optimal Value
∇	Step size for LMS algorithms [18], [19]	[0.001, 0.1]	0.01
γ	Leakage factor for LLMS [14]	[0.001, 0.01]	0.002
γ_3	Zero-attracting parameter for ZA-LLMS [15]	[0.01, 0.1]	0.1
γ_4	Reweighted zero-attracting parameter for RZA-LLMS [17]	[0.01, 0.1]	0.01
ζ_4	Reweighted zero-attracting constant for RZA-LLMS [17]	[0.1, 1.0]	0.1
α_5	Decay factor for VSS-LMS [18]	[0.9, 0.99]	0.9
κ_5	Step size adjustment factor for VSS-LMS [18]	[0.01, 0.1]	0.05
∇_{min}	Minimum step size for VSS-LMS [18]	[0.001, 0.01]	0.01
∇_{max}	Maximum step size for VSS-LMS [18]	[0.01, 0.1]	0.04
φ_6	Zero-attracting parameter for ZA-VSS-LMS [21]	[0.01, 0.1]	0.1
φ_7	Reweighted zero-attracting parameter for RZA-VSS-LMS [21]	[0.001, 0.1]	0.001
ζ_7	Reweighted zero-attracting constant for RZA-VSS-LMS [21]	[0.1, 1.0]	0.1

The convergence behavior of the proposed hybrid approach, namely PSO-ZA-LLMS and PSO-RZA-LLMS, is visually depicted in Figure 3. The figure illustrates the performance of the algorithms in two distinct noise environments, namely AWGN as depicted in Figure 3a, and CGS noise as depicted in Figure 3b. Notably, the RZA-LLMS algorithm exhibits superior efficacy by effectively manipulating the weight coefficients, thereby mitigating bias to a greater extent when compared to the ZA-LLMS algorithm. Remarkably, both the PSO-ZA-LLMS and PSO-RZA-LLMS algorithms yield MSE values that are comparable to, or even lower than, those achieved by the ZA-LLMS and RZA-LLMS algorithms, respectively, in both noise environments. Furthermore, the PSO-ZA-LLMS and PSO-RZA-LLMS approaches exhibit accelerated convergence rates and significantly reduced MSE values in comparison to the ZA-LLMS and RZA-LLMS methods. Specifically, the PSO-ZA-LLMS and PSO-RZA-LLMS algorithms achieve steady-state performance at an early stage of the learning process, facilitated by the ability of PSO to enhance the weight coefficients and smoothly guide the MSE towards local minima with heightened efficiency. Conversely, both the ZA-LLMS and RZA-LLMS algorithms require significantly longer periods to reach the steady state. Moreover, the incorporation of PSO in the ZA-LLMS and RZA-LLMS algorithms enables a notable reduction in the computational time, resulting in approximately 400 fewer iterations compared to their non-PSO counterparts, namely ZA-LLMS and RZA-LLMS.

In order to establish the effectiveness of the proposed hybrid approach, it has been employed to assess the performance of the ZA-VSS-LMS and RZA-VSS-LMS algorithms. A similar trend is observed in Figure 4. Moreover, compared to their equivalents, the suggested PSO-ZA-VSS-LMS and PSO-RZA-VSS-LMS algorithms show remarkable traits like faster convergence rates and lower MSE values. This result is evidence of the strength of the hybrid method since it efficiently covers the capacity to understand several models, maximize their learning process, and interact smoothly with other algorithms.
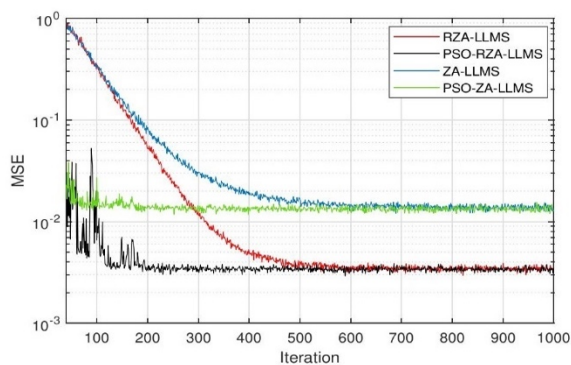
(a) AWGN.
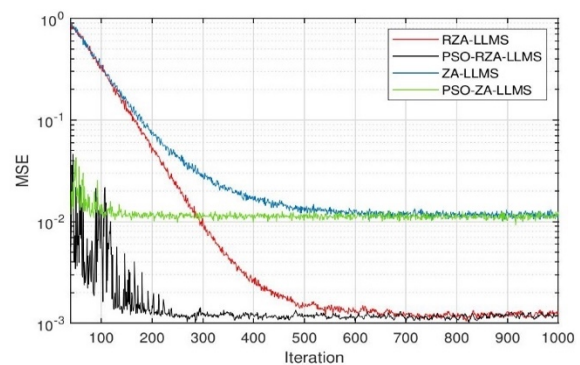
(b) CGS.

Figure 3: MSE to Iteration for ZA-LLMS, RZA-LLMS, PSO-ZA-LLMS, and PSO-RZA-LLMS using (a) Additive White Gaussian Noise (AWGN), and (b) Correlated Gaussian Sequence (CGS) noise.
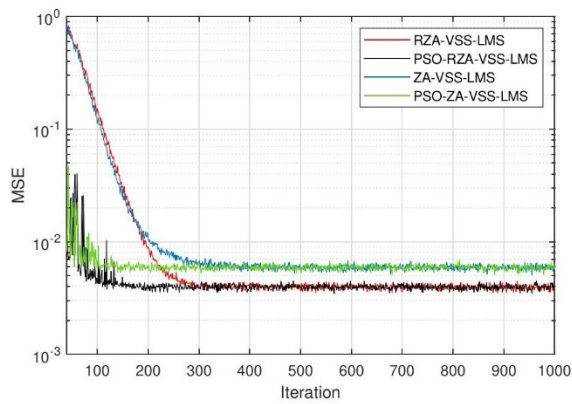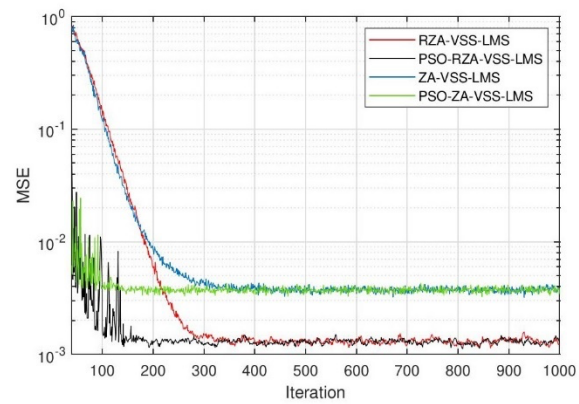
(a) AWGN. **(b) CGS.**

Figure 4: MSE to Iteration for ZA-VSS-LMS, RZA-VSS-LMS, PSO-ZA-VSS-LMS, and RZA-VSS-LMS using (a) Additive White Gaussian Noise (AWGN), and (b) Correlated Gaussian Sequence (CGS) noise.

Table 3 shows the convergence iterations of the eight techniques over several SNR values. The results reveal that PSO causes a clear acceleration in the convergence of all algorithms and over all cases.

Table 3: Comparison of the convergence iterations and MSE of the 8 algorithms across various SNR values for AWGN.

SNR	10		15		20		25		30	
Measurement	MSE	Iter. #	MSE	Iter. #	MSE	Iter. #	MSE	Iter. #	MSE	Iter. #
ZA-LLMS [15]	0.131	597	0.046	607	0.019	643	0.013	663	0.012	711
PSO-ZA-LLMS	0.11	137	0.043	154	0.019	170	0.012	190	0.011	194
RZA-LLMS [17]	0.102	398	0.037	434	0.0093	459	0.0024	556	0.0013	602
PSO-RZA-LLMS	0.098	149	0.034	152	0.0089	192	0.0024	272	0.001	312
ZA-VSS-LMS [21]	0.134	201	0.057	222	0.01	283	0.0049	297	0.0021	314
PSO-ZA-VSS-LMS	0.125	97	0.055	140	0.01	189	0.003	195	0.0012	205
RZA-VSS-LMS [21]	0.102	261	0.046	284	0.011	317	0.005	324	0.0034	404
PSO-RZA-VSS-LMS	0.101	90	0.037	113	0.009	138	0.0042	179	0.0019	212

Figure 5 shows, at various SNR levels, the ratio of iterations needed to achieve MSE convergence in the PSO-optimized algorithms (PSO-ZA-LLMS, PSO-RZA-LLMS, PSO-ZA-VSS-LMS, RZA-VSS-LMS, respectively) compared to non-PSO versions (ZA-LLMS, RZA-LLMS, ZA-VSS-LMS, RZA-VSS-LMS, respectively). The results indicate that the use of PSO optimization accelerates convergence in all cases, with varying ratios. For instance, at SNR=10, PSO-ZA-LLMS requires only 23% of the iterations needed by ZA-LLMS to converge. Although Figure 5 shows that PSO-ZA-VSS-LMS and PSO-RZA-VSS-LMS require higher percentages compared to PSO-ZA-LLMS and PSO-RZA-LLMS, Table 3 demonstrates that they still require fewer iterations to converge.

VI. EVALUATION OF THE 2-D LMS-BASED ALGORITHMS IN 2D IMAGES
A. VALIDATION AND EXPERIMENTAL DESIGN
In order to evaluate the performance of the proposed technique on MRI scans, we applied several noise types to the original images presented in Figure 6. These noise variations are described as follows:
- **Gaussian Noise:** White noise with a mean of 0.3 and variance of 0.01.
- **Localvar Noise:** Zero-mean Gaussian noise with a local variance of 0.3.
- **Poisson Noise:** Representing photon-based noise commonly found in medical and astronomical imaging.
- **Salt & Pepper Noise:** Impulsive noise with a density of 0.3, simulating errors in data transmission or sensor malfunction.
- **Speckle Noise:** Often encountered in radar and ultrasonic imaging systems, a multiplicative noise model with a variance of 0.05.

We used the Peak Signal-to-Noise Ratio (PSNR) to evaluate the image restoration and noise lowering efficacy. Expressed in decibels, PSNR is a widely used statistic measuring the ratio between the noise present and the maximum achievable power of an image. A greater PSNR value denotes improved recovered image quality relative to original one.
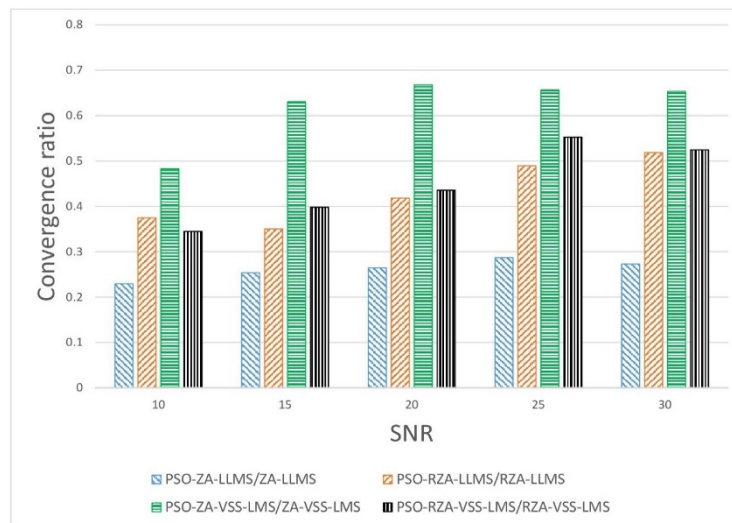
Figure 5: Comparison of convergence rates between algorithms with and without PSO optimization technique for AWGN.
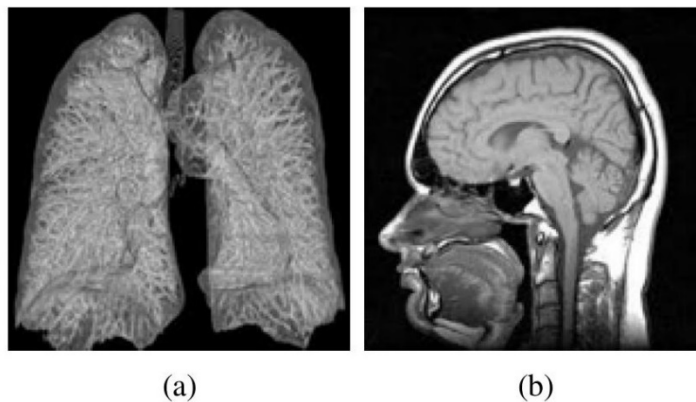
Figure 6: Original images used in the experiments.

B. RESULTS

Tables 4 and 5 show the Peak PSNR values for the images restored by both the suggested and conventional techniques. The outcomes allow one to make the following important observations:

- **Improvement with PSO-Enhanced Algorithms:** In terms of PSNR, the algorithms improved with Particle Swarm Optimization (PSO) routinely exceeded their conventional equivalents. For images damaged with Gaussian noise, for example, the RZA-LLMS-PSO technique attained a PSNR value of 18.0922 dB, as shown in Table 4, compared to 15.7893 dB acquired using the regular RZA-LLMS approach. Similarly, Table 5 shows that the PSO-augmented RZA-VSS-LMS algorithm produced a PSNR of 24.5831 dB under Gaussian noise conditions, which is substantially higher than the 20.5074 dB achieved by the non-PSO RZA-VSS-LMS approach. The PSO-RZA-VSS-LMS method outperformed others in preserving fine anatomical details for Speckle noise reduction, particularly in difficult areas like tissue borders and parenchymal textures in MRI data. The PSO-enhanced version, as seen in Table 5, got a PSNR of 24.56 dB, well above the baseline RZA-VSS-LMS, which got 20.95 dB, a 17.2% increase in signal quality. This improvement shows the capacity of the method to keep significant structural characteristics while lowering multiplicative artefacts. Visual outcomes (Table 7) show the clarity enhancement even more; PSO-RZA-VSS-LMS generates sharper edges and more homogenous textures in denoised photos. These quantitative and qualitative improvements highlight the efficiency of the model and especially fit it for diagnostic imaging uses since high-fidelity restoration of small structures is vital.

- **Algorithm Robustness Across Different Noise Variants:** The proposed algorithms demonstrated considerable robustness when applied to various noise types. For example, the ZA-VSS-LMS-PSO algorithm delivered notable PSNR values for both Poisson noise (16.9356 dB) and Speckle noise (16.4370 dB), as indicated in Table 4. This suggests that the algorithms are adept at managing both additive and multiplicative noise, positioning them as highly adaptable solutions for a diverse range of practical applications.
- **Visual Quality Enhancement:** Further confirmation of the effectiveness of the proposed algorithms is found in the visual quality improvements seen in the restored images, as presented in Tables 6 and 7. The images processed with PSO-enhanced algorithms exhibited sharper details and fewer distortion artifacts compared to those restored by the traditional methods. For example, as shown in Table 6, the image restored using the RZA-LLMS-PSO algorithm under Salt & Pepper noise conditions displays a significantly higher quality than the image restored using the conventional RZA-LLMS approach.

Table 4: PSNR values for the tested algorithms for the image in Figure 6a.

Noise Type	Gaussian	Localvar	Poisson	Salt & Pepper	Speckle
ZA-LLMS	15.9843	15.2826	15.1810	15.7862	15.5390
ZA-LLMS-PSO	16.2832	16.0262	15.7550	16.2376	15.9215
RZA-LLMS	15.7893	16.0250	15.9567	15.3292	16.0247
RZA-LLMS-PSO	18.0922	16.5063	16.6082	17.1189	16.6440
ZA-VSS-LMS	15.9636	16.2342	16.3827	15.7617	16.2120
ZA-VSS-LMS-PSO	16.4213	16.5370	16.9356	16.4390	16.4370
RZA-VSS-LMS	15.8754	16.1019	16.3433	15.6311	15.9219
RZA-VSS-LMS-PSO	16.4203	16.4362	16.7356	16.4395	16.4361

Table 5: PSNR values for the tested algorithms for the image in Figure 6b.

Noise Type	Gaussian	Localvar	Poisson	Salt & Pepper	Speckle
ZA-LLMS	18.4685	16.6679	16.7260	16.3003	16.4937
ZA-LLMS-PSO	18.5152	16.8115	16.7854	16.4749	16.5749
RZA-LLMS	18.4056	17.5042	17.3549	18.1443	17.5370
RZA-LLMS-PSO	20.2756	20.7498	20.7560	20.2489	20.6871
ZA-VSS-LMS	20.5119	20.9539	20.9358	20.7327	20.9812
ZA-VSS-LMS-PSO	24.5807	24.5587	24.5602	24.5490	21.2305
RZA-VSS-LMS	20.5074	21.0273	20.9358	20.1662	20.9545
RZA-VSS-LMS-PSO	24.5831	24.5609	24.5618	24.5513	24.5596

VII. ANALYSIS, LIMITATIONS, AND FUTURE WORKS

A. ANALYSIS AND DISCUSSION

The proposed PSO-based LMS adaptive filtering approach introduces a novel framework for optimizing filter coefficients, thereby improving convergence speed and reducing Mean Square Error (MSE). Compared to conventional LMS methods, the integration of PSO dynamically adjusts weights, preventing stagnation in local minima and enhancing adaptation efficiency. The experimental results validate the effectiveness of the proposed algorithms, particularly in synthetic noise environments (AWGN and CGS) and real-world MRI scans denoising.

Although the results show a notable increase in filtering accuracy and convergence rate, the following remarks draw attention to important aspects controlling performance:

- **Impact of PSO Optimization:** The adaptive exploration of the weight space of the PSO-enhanced LMS algorithms shows exceptional convergence characteristics. This advantage is particularly evident in cases where traditional LMS algorithms require extensive iterations to stabilize. As shown in Table 8, suboptimal PSO parameters can increase MSE by 29-75% and iterations by 26-63%, confirming the need for careful parameter selection.
- **Performance in Different Noise Conditions:** The proposed method demonstrates robust performance under Gaussian and colored Gaussian noise; however, further analysis is required to assess its effectiveness in handling non-Gaussian noise distributions and real-world signal variations.

B. LIMITATIONS OF THE STUDY

Despite the demonstrated improvements, the study has certain limitations that must be acknowledged:

- **Hyperparameter Sensitivity:** The choice of PSO parameters (e.g., inertia weight, number of particles, acceleration coefficients) significantly influences performance. A suboptimal configuration may lead to slower convergence or premature stagnation. As shown in Table 8, suboptimal PSO parameters can increase MSE by 29-75% and iterations by 26-63%, confirming the need for careful parameter selection.
- **Risk of Local Minima:** Although PSO mitigates local minima issues better than standard gradient-based approaches, it is still susceptible to convergence stagnation in complex search spaces.

Table 6: Retrieved images using the proposed and the compared algorithms for the image in Figure 6a.
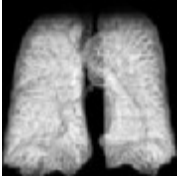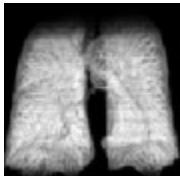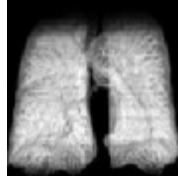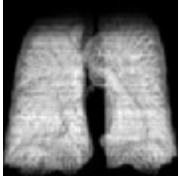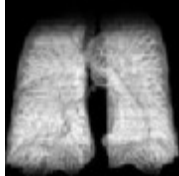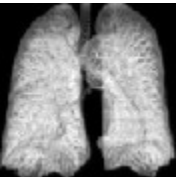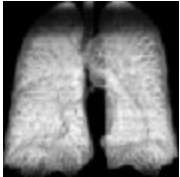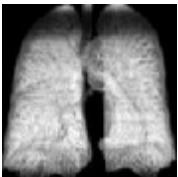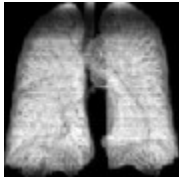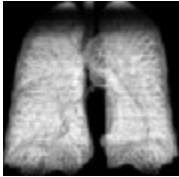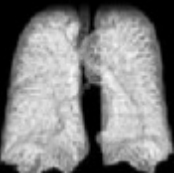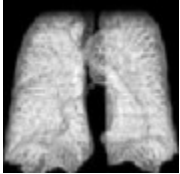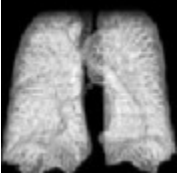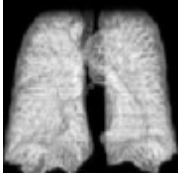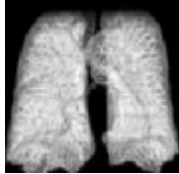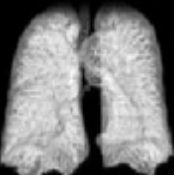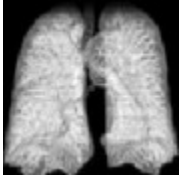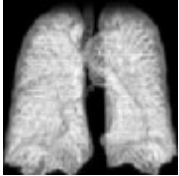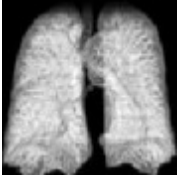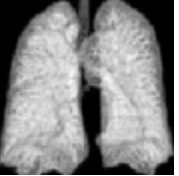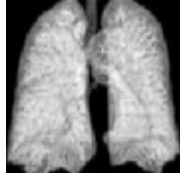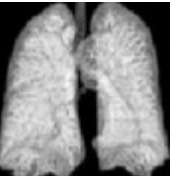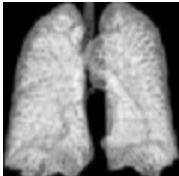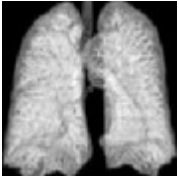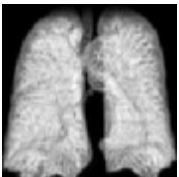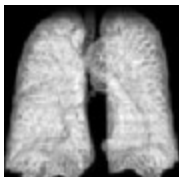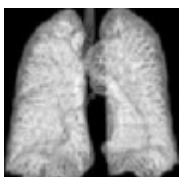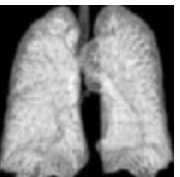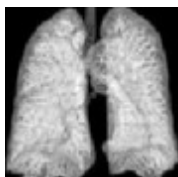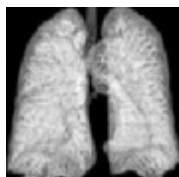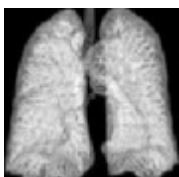
Technique	Gaussian	Localvar	Poisson	Salt&Pepper	Speckle
Noised Images					
ZA-LLMS					
ZA-LLMS-PSO					
RZA-LLMS					
RZA-LLMS-PSO					
ZA-VSS-LMS					
ZA-VSS-LMS-PSO					
RZA-VSS-LMS					
RZA-VSS-LMS-PSO					

Table 7: Retrieved images using the proposed and the compared algorithms for the image in Figure 6b
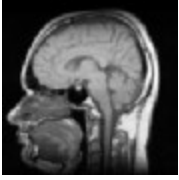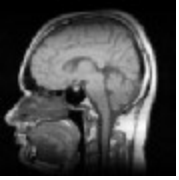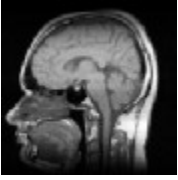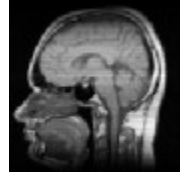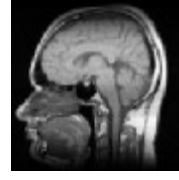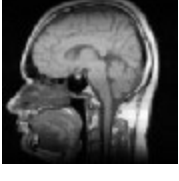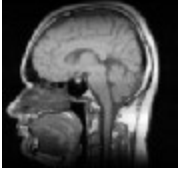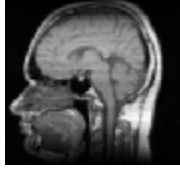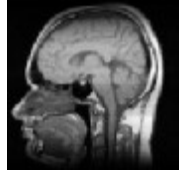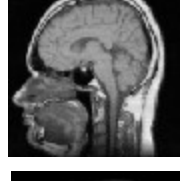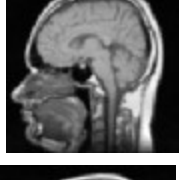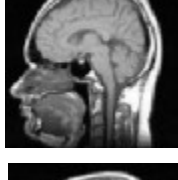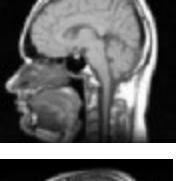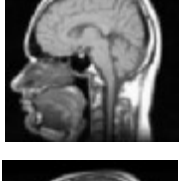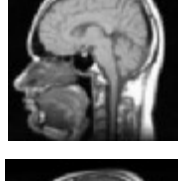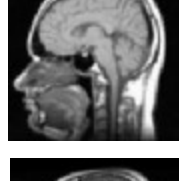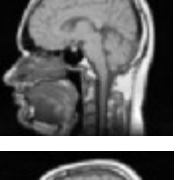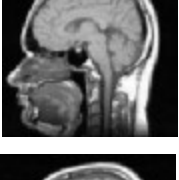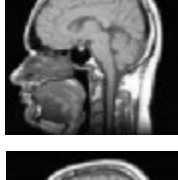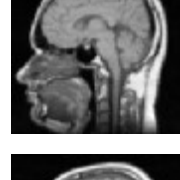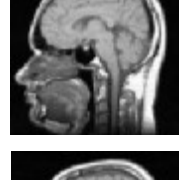
Technique	Gaussian	Localvar	Poisson	Salt&Pepper	Speckle
Noised Images					
ZA-LLMS					
ZA-LLMS-PSO					
RZA-LLMS					
RZA-LLMS-PSO					
ZA-VSS-LMS					
ZA-VSS-LMS-PSO					
RZA-VSS-LMS					
RZA-VSS-LMS-PSO					

Table 8: Ablation study evaluating the impact of individual PSO components on the performance of LMS-variant algorithms under AWGN conditions (SNR = 25 dB).

Algorithm	Configuration	MSE (Steady-State)	Iterations to Converge	% Faster vs. Non-PSO
PSO-ZA-LLMS	Full PSO	0.019	170	63%
	Fixed Inertia ($\epsilon = 0.5$)	0.022	230	50%
	No Personal Best ($c1 = 0$)	0.025	290	37%
	No Global Best ($c2 = 0$)	0.027	310	33%
PSO-RZA-LLMS	Full PSO	0.0024	190	66%
	Fixed Inertia ($\epsilon = 0.5$)	0.0031	240	57%
	No Personal Best ($c1 = 0$)	0.0038	300	46%
	No Global Best ($c2 = 0$)	0.0042	310	44%
PSO-ZA-VSS-LMS	Full PSO	0.003	195	34%
	Fixed Inertia ($\epsilon = 0.5$)	0.0039	250	16%
	No Personal Best ($c1 = 0$)	0.0045	320	-6%
	No Global Best ($c2 = 0$)	0.0051	340	-12%
PSO-RZA-VSS-LMS	Full PSO	0.0042	179	45%
	Fixed Inertia ($\epsilon = 0.5$)	0.0049	220	32%
	No Personal Best ($c1 = 0$)	0.0057	280	13%
	No Global Best ($c2 = 0$)	0.0063	300	7%

C. FUTURE RESEARCH DIRECTIONS

To further improve the proposed PSO-based LMS filtering approach, several potential research directions are identified:

- **Benchmarking Against Alternative Optimization Techniques:** While a theoretical comparison of PSO with Genetic Algorithms (GA), Differential Evolution (DE), Artificial Bee Colony (ABC), Artificial Neural Networks (ANN), and Reinforcement Learning (RL) has been provided, future work will include experimental benchmarking to validate performance trade-offs in different filtering tasks.

- **Advanced Hyperparameter Optimization:** Future research could include advanced hyperparameter tuning methods—such as Grid Search, Random Search, or Bayesian Optimization—to automatically identify ideal PSO parameters (e.g., inertia weight, acceleration coefficients), hence improving the robustness and generalization of the PSO-based adaptive filters. This would increase convergence consistency across datasets and lessen dependence on manual parameter selection.

- **Evaluation with Non-Gaussian Noise and Real-World Signals:** Broadening the study to encompass non-Gaussian noise distributions, larger real-world signal datasets, and dynamic telecommunication data would improve the robustness and generalizability of the proposed method.

- **Integration with Deep Learning Architectures:** The effectiveness of PSO-based LMS filtering in deep learning applications, such as Transformer-based denoising, LSTM-based adaptive filtering, and CNN-based signal restoration, should be investigated.

- **Ablation Studies to Assess PSO Contributions:** Conducting ablation studies by removing specific PSO components (e.g., adaptive weight selection, particle velocity updates) can provide deeper insights into the individual contributions of PSO to LMS performance.

- **Real-Time Implementation and Hardware Acceleration:** Implementing the PSO-LMS approach on FPGA, DSP, or GPU platforms can enable real-time deployment in latency-sensitive applications, such as speech enhancement and medical signal processing.

By tackling these topics, next research can improve PSO-based LMS filtering methods, increase their relevance in more general fields, and raise their performance in practical adaptive filtering applications.

VIII. CONCLUSION

This paper introduced a hybrid adaptive filtering framework that combines Particle Swarm Optimization (PSO) with various LMS-based algorithms, including ZA-LLMS, RZA-LLMS, ZA-VSS-LMS, and RZA-VSS-LMS. The integration of PSO significantly improved convergence speed and reduced Mean Square Error (MSE) compared to traditional LMS-based methods.

Experimental evaluations demonstrated the robustness of the proposed approach in both synthetic and real-world scenarios. In simulations with Additive White Gaussian Noise (AWGN) and Colored Gaussian Sequence (CGS), the PSO-enhanced methods outperformed conventional LMS variants, achieving up to a 67% reduction in iterations while maintaining lower MSE. Notably, the PSO-RZA-VSS-LMS algorithm achieved a PSNR of 24.58 dB under Gaussian noise conditions, surpassing its non-PSO counterpart at 20.51 dB. Similarly, under Salt & Pepper noise conditions, PSO-RZA-LLMS recorded 17.12 dB, compared to 15.32 dB with the standard RZA-LLMS, demonstrating superior denoising performance.

Further validation on 2D MRI images confirmed the adaptability of the proposed approach to image restoration tasks. Across multiple noise types, including Localvar noise, Poisson noise, Salt & Pepper noise, and Speckle noise, the PSO-augmented algorithms consistently outperformed their non-PSO counterparts in terms of PSNR and convergence speed. The experimental results revealed that PSO optimization not only enhanced filtering accuracy but also reduced computational complexity, with some PSO-enhanced methods requiring nearly 400 fewer iterations compared to their standard versions.

Despite these promising results, the study has certain limitations. While PSO enhances weight optimization, it may encounter local minima issues in high-dimensional filtering tasks, and its generalizability to broader signal processing applications, such as EEG analysis and telecommunication systems, remains an open research question. Additionally, the sensitivity of PSO to hyperparameter selection requires further investigation to ensure robust performance across different scenarios.

For future research, several directions are proposed. First, a comprehensive benchmarking of PSO-based LMS against alternative optimization techniques, such as Genetic Algorithms (GA) and Differential Evolution (DE), would provide deeper insights into its competitive advantages. Second, grid search and Bayesian optimization among other automated hyperparameter tuning techniques might improve model stability. Third, the flexibility of the suggested method would be enhanced by include real-world signal processing activities and non-Gaussian noise distributions in evaluation. Finally, investigating deep learning integrations with PSO-based LMS, such Transformer-based filtering or LSTM-based adaptive filtering, could open fresh opportunities for signal augmentation in challenging settings.

The outcomes of this work show the great possibilities of PSO-driven adaptive filtering as an effective substitute for real-time signal processing uses including dynamic noise cancellation, speech enhancement, and medical imaging. Future developments in hybrid learning methods and optimization strategies will help to improve and increase the relevance of the suggested strategy.

Authors' Contributions
In this study, author prepared and wrote the study.

Competing Interests
The authors declare that they have no competing interests.

References

[1] M. E. sayed M. Sakr and M. A. M. Hassan, Satellite tracking control system using optimal variable coefficients controllers based on evolutionary optimization techniques, El-Cezeri Journal of Science and Engineering, vol. 10, no. 2, pp. 326–348, 2023.

[2] M. K. Derdiman, Ayrık pso algoritması ile sehim kısıtı altında İki doğrultudaki kirişli döşemelerin güvenilirlik tabanlı optimizasyonu,'' El-Cezeri Journal of Science and Engineering, vol. 9, no. 1, pp. 49–64, 2022.

[3] Özdemir, S. "Oztürk, O. Şengül, and F. Kuncan, ''Position control of the suspended pendulum system with particle swarm optimization algorithm, El-Cezeri Journal of Science and Engineering, vol. 9, no. 2, pp. 669–679, 2022.

[4] Y. Bai, X. Gong, Q. Lu, Y. Song, W. Zhu, S. Xue, D. Wang, Z. Peng, and Z. Zhang, Application of adaptive filtering algorithm to the stability problem for double crystal monochromator. part i: Typical filtering algorithms, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, vol. 1048, p. 167924, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0168900222012165

[5] 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, Systems, and Signal Processing, vol. 41, no. 9, pp. 5254–5282, Sep 2022. [Online]. Available: https://doi.org/10.1007/s00034-022-02034-2

[6] B. Durmuş, Infinite impulse response system identification using average differential evolution algorithm with local search, Neural Computing and Applications, vol. 34, no. 1, pp. 375–390, Jan 2022. [Online]. Available: https://doi.org/10.1007/s00521-021-06399-4

[7] G. Clark, S. Mitra, and S. Parker, Block implementation of adaptive digital filters, IEEE Transactions on Circuits and Systems, vol. 28, no. 6, pp. 584–592, 1981.

[8] Y. Yu, Z. Huang, H. He, Y. Zakharov, and R. C. de Lamare, Sparsity-aware robust normalized subband adaptive filtering algorithms with alternating optimization of parameters, IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 69, no. 9, pp. 3934–3938, 2022.

[9] G. Boidi, M. R. Da Silva, F. J. Profito, and I. F. Machado, Using machine learning radial basis function (rbf) method for predicting lubricated friction on textured and porous surfaces, Surface Topography: Metrology and Properties, vol. 8, no. 4, p. 044002, 2020.

[10] A. Singh, Adaptive noise cancellation, Dept. of Electronics & Communication, Netaji Subhas Institute of Technology, vol. 1, 2001.

[11] H. Kolivand, K. A. Akintoye, S. Asadianfam, and M. S. Rahim, Improved methods for finger vein identification using composite median-wiener filter and hierarchical centroid features extraction, Multimedia Tools and Applications, pp. 1–32, 2023.

[12] S. M. Wilson, M. K. Bohn, A. Madsen, T. Hundhausen, and K. Adeli, Lms-based continuous reference percentiles for 14 laboratory parameters in the caliper cohort of healthy children and adolescents,Clinical Chemistry and Laboratory Medicine (CCLM), 2023.

[13] B. Singh, M. Kandpal, and I. Hussain, Control of grid tied smart pv-dstatcom system using an adaptive technique, IEEE transactions on smart grid, vol. 9, no. 5, pp. 3986–3993, 2016.

[14] K. Mayyas and T. Aboulnasr, Leaky lms algorithm: Mse analysis for gaussian data, IEEE Transactions on Signal Processing, vol. 45, no. 4, pp. 927–934, 1997.

[15] M. S. Salman, Sparse leaky-lms algorithm for system identification and its convergence analysis, International Journal of Adaptive Control and Signal Processing, vol. 28, no. 10, pp. 1065–1072, 2014. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/acs.2428

[16] Y. Li and M. Hamamura, Zero-attracting variable-step-size least mean square algorithms for adaptive sparse channel estimation, International Journal of Adaptive Control and Signal Processing, vol. 29, no. 9, pp. 1189–1206, 2015. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/acs.2536

[17] M. S. Salman, A. A. Hameed, C. Turan, and B. Karlik, A new sparse convex combination of za-llms and rza-llms algorithms, in 2015 23nd Signal Processing and Communications Applications Conference (SIU), 2015, pp. 711–714.

[18] R. Kwong and E. Johnston, ''A variable step size lms algorithm, IEEE Transactions on Signal Processing, vol. 40, no. 7, pp. 1633–1642, 1992.

[19] V. Mathews and Z. Xie, A stochastic gradient adaptive filter with gradient adaptive step size, IEEE Transactions on Signal Processing, vol. 41, no. 6, pp. 2075–2087, 1993.

[20] T. Aboulnasr and K. Mayyas, A robust variable step-size lms-type algorithm: analysis and simulations, IEEE Transactions on Signal Processing, vol. 45, no. 3, pp. 631–639, 1997.

[21] M. S. Salman, M. N. S. Jahromi, A. Hocanin, and O. Kukrer, A zero-attracting variable step-size lms algorithm for sparse system identification, in 2012 IX International Symposium on Telecommunications (BIHTEL), 2012, pp. 1–4.

[22] W. Jia, S. Kong, T. Cai, M. Li, Y. Jin, P. Wang, and Z. Dai, Steady-state performance analysis of the arctangent lms algorithm with gaussian input, IEEE Transactions on Circuits and Systems II: Express Briefs, 2023.

[23] H. Ferro, A combination of adaptive filters based on competitive learning principles, Available at SSRN 4345598, 2023.

[24] Y. He, J. Wei, Y. He, X. Rong, W. Guo, F. Wang, Y. Wang, and J. Liu, A process strategy planning of additive-subtractive hybrid manufacturing based multi-dimensional manufacturability evaluation of geometry feature, Journal of Manufacturing Systems, vol. 67, pp. 296–314, 2023.

[25] Á. A. Vázquez, J. G. Avalos, G. Sánchez, J. C. Sánchez, and H. Pérez, A comparative survey of convex combination of adaptive filters, IETE Journal of Research, vol. 69, no. 2, pp. 940–950, 2023.

[26] M. Martinez-Ramon, J. Arenas-García, A. Navia-Vázquez, and A. R. Figueiras-Vidal, An adaptive combination of adaptive filters for plant identification, in 2002 14th International Conference on Digital Signal Processing Proceedings. DSP 2002 (Cat. No. 02TH8628), vol. 2. IEEE, 2002, pp. 1195–1198.

[27] R. C. Eberhart and Y. Shi, Comparison between genetic algorithms and particle swarm optimization, in Evolutionary Programming VII, V. W. Porto, N. Saravanan, D. Waagen, and A. E. Eiben, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 611–616.

[28] A. K. Mahapatra, N. Panda, and B. K. Pattanayak, Hybrid PSO (SGPSO) with the incorporation of discretization operator for training RBF neural network and optimal feature selection, Arabian Journal for Science and Engineering, vol. 48, no. 8, pp. 9991–10 019, Aug. 2023.

[29] F. Yun, H. Dong, C. Liang, T. Weimin, and T. Chao, Feature selection of XLPE cable condition diagnosis based on PSO-SVM, Arabian Journal for Science and Engineering, vol. 48, no. 5, pp. 5953–5963, May 2023.

[30] M. Saber, M. E. Ghoneim, and S. Kumar, Survey on design of digital fir filters using optimization models, Journal of Artificial Intelligence and Metaheuristics, vol. 2, no. 1, pp. 16–26, 2022.

[31] D. Krusienski and W. Jenkins, A particle swarm optimization-least mean squares algorithm for adaptive filtering, in Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers, 2004., vol. 1, 2004, pp. 241–245 Vol.1.

[32] M. Chen, Y. Wang, P. Li, and H. Fu, Research on an improved pso algorithm with dual self-adaptation and dual variation, in 2022 IEEE International Conference on Mechatronics and Automation (ICMA), 2022, pp. 646–650.

[33] J. Xin, G. Chen, and Y. Hai, A particle swarm optimizer with multi-stage linearly-decreasing inertia weight, in 2009 International Joint Conference on Computational Sciences and Optimization, vol. 1, 2009, pp. 505–508.

[34] K. Deb, Multi-objective optimisation using evolutionary algorithms: an introduction, in Multi-objective evolutionary optimisation for product design and manufacturing. Springer, 2011, pp. 3–34.

[35] B. Alhijawi and A. Awajan, Genetic algorithms: Theory, genetic operators, solutions, and applications, Evolutionary Intelligence, vol. 17, no. 3, pp. 1245–1256, 2024.

[36] Q.-W. Chai, L. Kong, J.-S. Pan, and W.-M. Zheng, A novel discrete artificial bee colony algorithm combined with adaptive filtering to extract fetal electrocardiogram signals, Expert Systems with Applications, vol. 247, p. 123173, 2024.

[37] J. C. Bansal, H. Sharma, and S. S. Jadon, Artificial bee colony algorithm: a survey, International Journal of Advanced Intelligence Paradigms, vol. 5, no. 1-2, pp. 123–159, 2013.

[38] K. Price, R. M. Storn, and J. A. Lampinen, Differential evolution: a practical approach to global optimization, Springer Science & Business Media, 2006.

[39] E. Reyes-Davila, E. H. Haro, A. Casas-Ordaz, D. Oliva, and O. Avalos, Differential evolution: a survey on their operators and variants, Archives of Computational Methods in Engineering, pp. 1–30, 2024.

[40] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S.-Y. Chang, and T. Sainath, Deep learning for audio signal processing, IEEE Journal of Selected Topics in Signal Processing, vol. 13, no. 2, pp. 206–219, 2019.

[41] A. Younesi, M. Ansari, M. Fazli, A. Ejlali, M. Shafique, and J. Henkel, A comprehensive survey of convolutions in deep learning: Applications, challenges, and future trends, IEEE Access, vol. 12, pp. 41 180–41 218, 2024.