Proceedings of CONF-MCEE 2026 Symposium: Advances in Sustainable Aviation and Aerospace Vehicle Automation
DOI: 10.54254/2755-2721/2026.KA29035

Design of an FIR Low-Pass Digital Filter Based on the Sine Function PSO Algorithm

Mingze Wei

School of Electronic Engineering, North China University of Water Resources and Electric Power, Zhengzhou, China
202319131@stu.ncwu.edu.cn

Abstract. The Particle Swarm Optimization (PSO) algorithm offers the advantages of fast convergence speed and a simple structure for digital filter design. However, it is prone to premature convergence, often settling in local optima. Improved PSO algorithms are being continuously developed. Starting from the sine adjustment of the acceleration coefficient in the PSO algorithm speed update formula, a Sine Function Particle Swarm Optimization (SFPSO) algorithm was implemented and used to design finite impulse response (FIR) low-pass digital filters. Simultaneously, it was compared with the standard PSO, Adaptive Particle Swarm Optimization (APSO), and Opposition-Based Particle Swarm Optimization (OPSO) algorithms. The research results show that the FIR low-pass digital filter designed by SFPSO algorithm performed the best in convergence speed and accuracy; the transition zone was the steepest; the passband ripple (0.4046 dB) was very close to the optimal value of the four algorithms (0.4021 dB), with the lowest passband standard deviation, therefore the passband performance was excellent; the stopband ripple was only lower than APSO, the stopband attenuation is only lower than OPSO, and the stopband high-frequency filtering performance is good.

Keywords: sine function, particle swarm optimization, FIR digital filter

1. Introduction

Digital filters are divided into FIR and Infinite Impulse Response (IIR) digital filters based on their impulse response length. An FIR filter is a linear phase filter that can achieve strict linear phase calculation, and because it is more conducive to programming and hardware implementation without feedback, it is widely used in many fields such as signal acquisition and processing [1].

The frequency sampling and window function methods are the main design methods for FIR digital filters, both of which have the advantages of simple design and easy implementation. However, when faced with complex signals, their designed filters often exhibit problems, for example, slow convergence speed and low convergence accuracy [2], and therefore are rarely directly applied in practical filter design. To enhance the performance of FIR digital filters, industry scholars have begun employing intelligent algorithms to aid in optimizing their design. Common intelligent algorithms include the genetic algorithm [3], Artificial Bee Colony [4], Particle Swarm Optimization (PSO), and Differential Evolution Algorithm [5]. These intelligent algorithms have

© 2025 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2026.KA29035

shown good results when applied to the optimization of FIR digital filters, thereby improving the performance of the filters. Nevertheless, each algorithm possesses its own distinct merits and limitations. In comparison, PSO has fewer parameter settings, faster convergence speed, smaller computational complexity, stronger search ability, and a simpler and easier-to-implement structure when optimizing the design of digital filters. Therefore, it is highly favored by scholars in the field.

The PSO algorithm was proposed by Kenney and Eberhart in 1995 [6], and later, new inertia weights were added to the PSO algorithm by SHI Y et al [7]. The PSO algorithm with added inertia weights was referred to as the standard PSO algorithm. Because the search process of the PSO algorithm has a certain directionality, which is different from the random selection strategy of other algorithms, its performance in solving specific problems is significantly better than that of other algorithms. It has been widely used in many fields, such as communication network design [8], circuit design [9], antenna design [10], reservoir water level control [11], and prediction models [12]. However, the standard PSO algorithm still has the drawback of easily getting stuck in local optima. Therefore, some improved PSO algorithms have been continuously developed and applied, such as the APSO algorithm and the OPSO algorithm [13-14].

Improving the PSO algorithm and applying it to filter design has always been a popular research topic in recent years. The sine function value has the characteristic of periodic fluctuations. It is theoretically feasible to improve the PSO algorithm using a sine function and apply it to filter design. The most widely used SFPSO algorithm introduces a sine function to dynamically adjust the inertia weight ω in the velocity update equation. However, there are few reports on the application of the SFPSO in filter design. Only Shao et al introduced sine trigonometric functions as parameter factors into the position update formula of the PSO algorithm to design FIR low-pass digital filters [14]. Compared with the OPSO algorithm, the results showed that the convergence speed and accuracy of the filters with trigonometric function factors introduced in the PSO position update equation were slower and lower. In view of this, this study starts with the sine adjustment of the acceleration coefficient in the PSO algorithm speed update formula, implements an SFPSO algorithm, and uses it to design FIR lowpass digital filters. It was compared with the standard PSO, adaptive PSO, and reverse learning PSO algorithms, providing a theoretical reference for the application of the SFPSO algorithm in the optimization design of FIR low-pass digital filters.

2. PSO algorithm and design of an FIR low-pass filter

2.1. Standard PSO algorithm

Within the conventional standard PSO algorithm, a particle corresponds to a potential solution in the search domain [7]. Particles have the option to depend on their personal experiences to identify the individual extreme value pbest, or to utilize the experience values of other particles in pursuit of the overall optimal value gbest. The inertia weight ω follows a linear modification approach, progressively diminishing with the rise in particle iterations. At the beginning of the iterative process, a higher ω value enhances the overall particle exploration, but as the iteration progresses, a lower ω value becomes more suitable for detailed local particle searches. The update formulas for particle speed and location in the standard PSO algorithm are outlined as follows:

$$v_{id}^{t+1} = \omega\, v_{id}^{t} + c_1 r_1 \left(pb_i^t - x_{id}^t\right) + c_2 r_2 \left(gb^t - x_{id}^t\right) \tag{1}$$

$$x_{id}^{t+1} = x_{id}^{t} + v_{id}^{t+1} \tag{2}$$

DOI: 10.54254/2755-2721/2026.KA29035

$$\omega_t = \omega_{max} - \frac{(\omega_{max}-\omega_{min})t}{t_{max}} \tag{3}$$

Within the equation, v_{id}^{t+1} represents the speed of particle i in the d-th dimension post the t+1th iteration, x_{id}^{t+1} denotes particle i's location in the d-th dimension following iteration t+1, with ω being the inertia weight, c_1 and c_2 are the acceleration coefficients, r_1 and r_2 are the figures within the range of [-1,1], pb_i^t represents the individual optimal value for the i-th particle after t iterations, gb^t represents the overall optimal value after t optimization iterations, ω_{max} is peak inertia weight, ω_{min} is the least inertia weight, t is the present iteration count, t_{max} is the highest Number of iterations.

The standard PSO algorithm is prone to premature convergence and gets stuck in local optima in practical applications. This is because the inertia weight ω follows a regular linear variation and acceleration coefficients c_1 and c_2 in the update formula are typically constant values, making this search strategy difficult to adapt to complex particle swarm optimization problems.

2.2. APSO algorithm

The inertia weight stands as a pivotal element in the PSO algorithm, with the standard PSO algorithm's inertia weight typically following a linear decline, leading to an unsatisfactory convergence outcome. The most commonly used APSO algorithm is the inertia weight ω changes automatically with the variation of particle iteration fitness value, thereby enhancing both the local and global search functions of the PSO algorithm [13,15]. The APSO algorithm uses the same velocity update formula as the standard PSO algorithm, but using the ω_t that automatically changes with t. The update of ω_t follows formulas (4) and (5).

$$\omega_{id}(t) = \omega_{min} + (\omega_{max} - \omega_{min})\frac{f(\mathrm{x}_{id}(t)) - f_{min}^t}{f_{average}^t - f_{min}^t} \quad f(\mathrm{x}_{id}(t)) \le f_{average}^t \tag{4}$$

$$\omega_{id}(t) = \omega_{\max} \quad f(\mathrm{x}_{id}(t)) > f_{average}^t \tag{5}$$

Within the equation, $f(\mathrm{x}_{id}(t))$ represents the fitness level of the i-th particle post t iterations, f_{min}^t represents the minimum fitness value attained by any particle at the t-th iteration, $f_{average}^t$ represents the average fitness value computed over the entire particle swarm at the t-th iteration.

2.3. OPSO algorithm

The OPSO algorithm introduces a reverse learning mechanism, whose basic idea is to obtain the reverse solution of the current particle velocity and position, and compare the fitness of the original solution and the reverse solution, selecting the better solution to participate in the next generation optimization [14,16]. Implementing reverse solutions enhances the variety within the particle population, thereby effectively avoiding the premature convergence of the particle population. The equation for updating position in the reverse solution is outlined below:

$$x_2 = A + B - x_1 \tag{6}$$

Within the equation, x_2 is defined as the inverse solution, A signifies the highest x value in the search area, B indicates the lowest x value in the search area, x_1 denotes the present particle

DOI: 10.54254/2755-2721/2026.KA29035

position's value.

2.4. Design of SFPSO algorithm

The sine function is introduced as a parameter factor into the acceleration coefficients c_1 and c_2 of the velocity update formula of the PSO algorithm, utilizing the fluctuation characteristics of the sine function value to expand the velocity vibration range of each particle, which also indirectly expands the search range of the particle positions. c_1 and c_2 fluctuate sinusoidally in a constant interval but are in opposing phases. With an increase in c_1 and a decrease in c_2, algorithms tend to be developed to enable particles to rely on their experience for the local fine search pbest. With a decrease in c_1 and an increase in c_2, the algorithm tends to explore ways to make particles globally search for the best position of the population. The formula for updating the speed of the SFPSO algorithm reads:

$$v_{id}^{t+1} = \omega\, v_{id}^{t} + c_1^t r_1 \left(pb_i^t - x_{id}^t\right) + c_2^t r_2 \left(gb^t - x_{id}^t\right) \tag{7}$$

$$c_1^t = c_1\left(1 + \mathrm{a}\ \sin\frac{2\pi t}{T_{max}}\right) \tag{8}$$

$$c_2^t = c_2\left(1 - \mathrm{a}\ \sin\frac{2\pi t}{T_{max}}\right) \tag{9}$$

In the formula, c_1 and c_2 is the acceleration coefficients, t denotes the count of present iterations, T_{max} indicates the highest iteration count, a is the sine amplitude coefficient. The introduction of sine factor can balance exploration and development through the periodic fluctuations of sine, thereby improving algorithm efficiency, and the sine coefficient can directly affect the extent of sine factor's impact on c value.

The simple implementation process of the SFPSO algorithm:

Step 1: Initiate each particle's position and speed in the swarm at random.

Step 2: Determine the fitness metrics for all particles.

Step 3: Revise the individual ideal position pbest and the overall optimal position gbest.

Step 4: Update the acceleration coefficient for each particle as per equations (8) and (9), and then update all particle velocities and positions for the next iteration according to Equations (7) and (2).

Step 5: Upon meeting the termination criteria, the calculation stops. If the termination criteria remain unfulfilled, repeat steps 3 through 5 until the termination condition is met.

2.5. Design of an FIR low pass digital filter

The primary function of an FIR low-pass digital filter is to permit frequencies below a specified cutoff to pass while suppressing those above it. The system function $H(z)$ for an FIR digital filter is expressed as equation (10) [14]. In the PSO optimization design of the filter in this study, the filter coefficient h(n) constitutes the optimization variable of the PSO algorithm, and each particle position represents a candidate H(z). The amplitude frequency response of each particle at each position is calculated using this formula, and then the fitness of the evaluated particles is calculated to ultimately achieve the optimization for the filter design.

$$H\left(z\right) = \sum_{n=0}^{N-1} h\left(n\right) z^{-n} \tag{10}$$

DOI: 10.54254/2755-2721/2026.KA29035

In the formula, filter order is N, the coefficient of the filter is h(n), (N-1) is the number of taps in the digital filter, 0≤n≤N-1.

2.6. Fitness evaluation

To verify the impact of different improved PSO algorithms on the performance of the designed FIR low-pass digital filter, the minimum mean square error function was used as the fitness function, and the cumulative mean square error formula is shown in formula (11) [14]. The formula combines amplitude frequency response, passband ripple, and stopband ripple. By calculating the weighted approximation error between passband ripple and stopband ripple, a set of coefficient combinations for the optimal FIR digital filter that minimizes the J value is obtained. This function is used as the fitness function for the FIR low-pass digital filter.

$$J = \sum abs[abs\left(|H\left(\omega\right)| - 1\right) - \delta p] + G(\omega)\sum[abs(|H\left(\omega\right)|) - \delta s] \tag{11}$$

δp is the passband ripple, δs is the stopband ripple, G (ω) is the weighting coefficient, the value of G(ω) is equal to the ratio of passband ripple to stopband ripple, H(ω) is the amplitude frequency response function of the FIR filter.

3. Results and analysis

3.1. Parameter design

According to the equiripple criterion, an FIR low-pass digital filter was designed. Standard PSO, APSO, OPSO and SFPSO algorithms were used to optimize the filter. The filter was designed with an order of 20, sampling frequency of 1000 Hz, 128 samples, normalized passband and stopband cutoff frequencies of 0.4 and 0.6, respectively. The PSO parameters were as follows: particle count of 50, maximum iteration of 200, and a simulation run 10 times for statistical calculation and analysis. The performance of the filter was evaluated by convergence speed, final convergence accuracy, passband ripple, stopband ripple, and stopband attenuation.

3.2. Convergence performance analysis

The convergence curves of the FIR digital low-pass filters designed using the four PSO algorithms were compared in Figure 1. As shown in Figure 1, the filters designed using the four PSO algorithms all have high convergence speeds, and the convergence accuracy of the FIR filters designed by the three optimized PSO algorithms is better than that of the standard PSO. Overall, SFPSO has the fastest convergence speed and the highest final convergence accuracy; the convergence speed of OPSO in the early stage is almost as fast as that of SFPSO, and the final convergence accuracy is slightly worse than that of SFPSO but better than that of APSO and standard PSO; the final convergence accuracy of standard PSO is the worst.

DOI: 10.54254/2755-2721/2026.KA29035

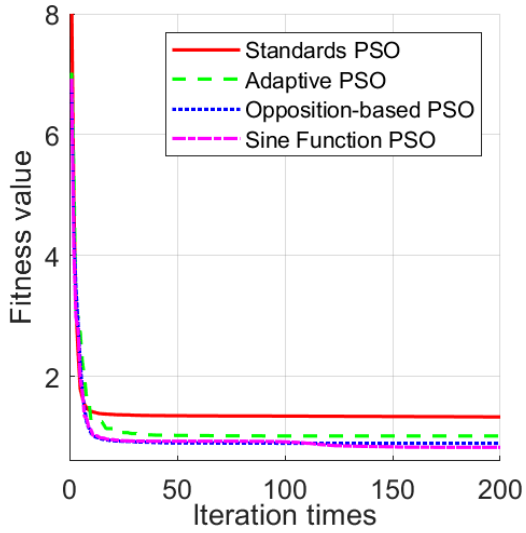


Figure 1. The convergence curves of the four PSO algorithms

3.3. Frequency response analysis of FIR filters designed with four PSO algorithms

Analyzing the amplitude ripple characteristics of the frequency response is a common method for evaluating the performance of FIR filters. The frequency response curves of the FIR filters designed using four PSO algorithms were shown in Figure 2-3. Figure 2 shows that, within the entire passband frequency range, the passband ripple of the three improved PSO algorithms is closer to the ideal 0 dB than that of the standard PSO, indicating that the low-pass filtering effect of the filters designed by the three improved PSO algorithms is optimized compared to the standard PSO. The passband ripple of the filter designed by the SFPSO algorithm is almost overlapped with that of the filter designed by the APSO and OPSO algorithms, which is difficult to distinguish. The passband curve of the standard PSO-designed filter is the worst among the four PSO algorithm-designed filters. It can be seen that within the transition band, the attenuation of the SFPSO transition band is the steepest, followed by OPSO and APSO, whereas the attenuation of the standard PSO transition band is the slowest.

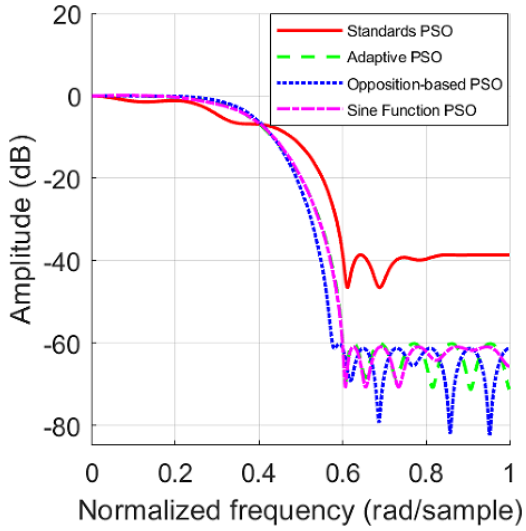


Figure 2. FIR low-pass digital filter amplitude response diagram

DOI: 10.54254/2755-2721/2026.KA29035

As shown in Figure 3, within the stopband, the stopband attenuation of the three improved PSO-designed filters is generally greater than 60dB, indicating that all three improved PSO algorithms meet the basic requirements for designing high-quality filters. The stopband attenuation value of the filter designed with standard PSO is around 40dB, and the stopband signal suppression effect is general.

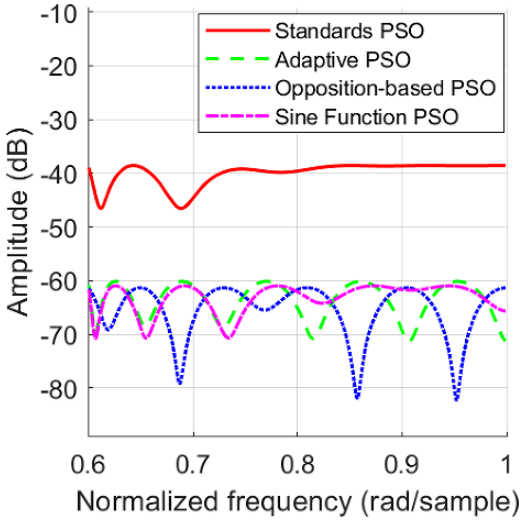


Figure 3. Attenuation diagram of the amplitude response of an FIR low-pass digital filter

3.4. Performance comparison of filters designed with four PSO algorithms

The performance statistics of the FIR low-pass digital filters designed using four PSO algorithms was compared in Table 1. According to Table 1, the filter designed with standard PSO has the minimum stopband attenuation and the largest passband ripple, indicating that the overall performance of the filter is the worst. The filter designed with APSO has the minimum stopband ripple, the largest passband standard deviation, and good stopband attenuation, resulting in excellent stopband performance; however, the large fluctuations in passband performance lead to poor stability. The filter designed with OPSO has the smallest passband ripple and the largest stopband attenuation, but its stopband standard deviation is the largest; the comprehensive performance is that the filter has the best passband flatness and the strongest stopband high-frequency signal suppression ability, but its stopband performance fluctuates greatly. The filter designed by SFPSO algorithm has a relatively low standard deviation in both passband and stopband, especially the passband standard deviation is much lower than other algorithms, indicating good repeatability of its design results; the passband ripple (0.4046 dB) is very close to the optimal value (0.4021 dB), showing excellent performance; the stopband ripple is only lower than APSO, and the stopband attenuation is only lower than OPSO, demonstrating excellent high-frequency suppression capability.

DOI: 10.54254/2755-2721/2026.KA29035

Table 1. Performance comparison of filters designed with four PSO algorithms

Filter performance index	Algorithm name			
	Standards PSO	APSO	OPSO	SFPSO
Passband ripple(dB)	0.497530	0.429626	0.402058	0.404570
Standard deviation of stopband ripple	0.071279	0.175455	0.162089	0.061077
Stopband ripple(dB)	0.009289	0.003836	0.007128	0.004822
Standard deviation of stopband ripple	0.003210	0.004418	0.008978	0.005961
Optimal stopband attenuation(dB)	38.63	60.14	61.35	60.98

4. Conclusion

The designed SFPSO algorithm with sinusoidal amplitude coefficient was applied to the optimization design of an FIR low-pass digital filter and compared with standard PSO, APSO and OPSO. The experimental results show that the FIR low-pass digital filter designed by SFPSO algorithm performs the best in convergence speed and accuracy; the transition zone is the steepest; the passband ripple (0.4046 dB) is very close to the optimal value of the four algorithms (0.4021 dB), with the lowest passband standard deviation and excellent passband performance; the stopband ripple is only lower than APSO, the stopband attenuation is only lower than OPSO, and the stopband high-frequency filtering performance is good. On the whole, the FIR low-pass digital filter designed with the SFPSO algorithm performs well in terms of convergence speed, final convergence accuracy, passband flatness, stopband suppression ability and operational stability. However, the passband ripple of the FIR low-pass digital filter designed with the SFPSO algorithm is higher than that of OPSO, the stopband ripple is higher than that of APSO, and the stopband attenuation is lower than that of OPSO; further optimization is still possible. The next study should be to further optimize the SFPSO by combining the introduction of a nonlinear decreasing sine amplitude, combining other local search algorithms, and introducing flight mechanisms to replace some random parameters.

References

[1] Liu, H (2023) Research and Design of FIR Filter Circuit Based on FPGA. University of Electronic Science and Technology of China.
[2] Shao, P. (2023) FIR Low Pass Digital Filter Basedon Multi-strategy Discrete Artificial Bee Colony Algorithm. Computer science, 50(S1), 227-231.
[3] Fotsing, K., Noura, A., Ntsama, E. P. and Simo, Thierry. (2025) Optimized adaptive filtering using genetic algorithm and signal quality index for practical motion artefact reduction in ambulatory ECG. Engineering Research Express, 7 (3), 035316-035316.
[4] Houssam, H., Awos, K. and Khalid, A. H. (2023) Optimized FIR Filter Using Genetic Algorithms: A Case Study of ECG Signals Filter Optimization. BioMedInformatics, 3(4), 1197-1215.
[5] Lian, L. and Tian, Z. D. (2022) FIR digital filter design based on improved artificial bee colony algorithm. Soft Computing, 26 (24), 13489-13507.
[6] Kennedy, J. and Eberhart, R. C. (1995) Particle swarm optimization. Proceedings of IEEE International Conference on Neural Networks, 1942-1948.
[7] Shi, Y. (1998). A modified particle swarm optimizer. Evolutionary Computation Proceedings, 69-73.
[8] Zi, Y, Z., Tao, Dong., Jie, Yin., Yue, Xu., Zongyi, Luo., Hao, J. and Jing, Wu. (2025) A Particle Swarm Optimization-Based Queue Scheduling and Optimization Mechanism for Large-Scale Low-Earth-Orbit Satellite Communication Networks. Sensors, 25 (4), 1069-1069.

DOI: 10.54254/2755-2721/2026.KA29035

[9] Syed, A. S., Shahzad, I., Jeongju, Park., Sekyung, H. and Woo, Y. K. (2025) Field Data Retrieval for Electric Vehicles and Estimating Equivalent Circuit Model Parameters via Particle Swarm Optimization. Technologies, 13 (3), 91-91.
[10] Jakkuluri, V. K. and S, M. S. (2024) Optimizing inset-fed rectangular micro strip patch antenna by improved particle swarm optimization and simulated annealing. Network (Bristol, England), 36 (3), 31-31.
[11] Oana, N. F. and Catalin, P. (2025) Combined Particle Swarm Optimization and Reinforcement Learning for Water Level Control in a Reservoir. Sensors, 25 (16), 5055-5055.
[12] Zuhan, Liu., Yuanhao, H., Zihai, F., Siqi, X., Lili, W. and Cuihong, Bao. (2025) Improved prediction model for daily PM2.5 concentrations with particle swarm optimization and BP neural network. Scientific Reports, 15 (1), 32050-32050.
[13] Qing, Y., Xin, Z., Rui, W., Wenlong, L., Tie, L. and Wenxiao, C. (2024) Control of thermal uniformity in microwave heating process by BPNN and adaptive particle swarm optimization. Heliyon, 10 (21), e37971-e37971.
[14] Shao, P., Wu, Z. J., Peng, H., Wang, Y. L. and Zhou, X. Y. (2017) FIR Low-pass Digital Filter Design Using Improved PSO Algorithms. Computer science, 44(S1), 136-138+156.
[15] Miao, S. W., He, Z. J. and Li, R. B. (2025) Terrain Matching Method Based on Adaptive Particle Swarm Optimization. NAVIGATION AND CONTROL, 24 (01), 42-52.
[16] Tizhoosh, H. R. (2005) Opposition-Based Learning: A New Scheme for Machine Intelligence. 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on Proceedings of Computational Intelligence for Modeling. Control and Automation, 695-701.