Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

An Enhanced PSO-Based PI Coefficient Tuning Method for a Fast Three-Phase PLL

Hesam Salimi
Arak University of Technology https://orcid.org/0000-0002-7937-9222
Behnam Salimi
Arak University of Technology https://orcid.org/0009-0007-6843-1500
Ehsan Nazemorroaya
ehsan.nzm74@gmail.com
University of Kashan https://orcid.org/0000-0002-3657-7595
Amir Arzhang
National University of Skills https://orcid.org/0009-0004-9815-8128

Research Article

Keywords: Phase-Locked Loop (PLL), Particle Swarm Optimization (PSO), Synchronous Reference Frame (SRF), PI Controller Tuning, Grid Synchronization, Dynamic Response

Posted Date: November 19th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11775072/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

An Enhanced PSO-Based PI Coefficient Tuning Method for a Fast Three-Phase PLL

Hesam Salimi [1], Behnam Salimi [1], Ehsan Nazemorroaya [2,*] and Amir Arzhang [3]

[1] Department of Electrical Engineering, Arak University of Technology, Arak, Iran
[2] Department of Electrical and Computer Engineering, University of Kashan, Kashan, Iran
[3] Department of Electrical Engineering, National University of Skills (NUS), Tehran, Iran
[*] Correspondence: ehsan.nzm74@gmail.com

Abstract: In many power electronic systems and grid-connected applications, the accurate estimation of voltage/current phase and frequency plays a pivotal role in system stability and proper operation. Phase-Locked Loops (PLLs) are among the most widely used synchronization methods due to their simple structure, high reliability, and ease of implementation in embedded systems. However, the slow dynamic response and the sensitivity of the PI controller gains in the Synchronous Reference Frame PLL (SRF-PLL) are considered significant challenges. Unlike conventional tuning strategies, this research work introduces an advanced Particle Swarm Optimization (PSO) framework specifically tailored for determining the optimal PI gains in three-phase PLLs. By redefining the objective function to strictly penalize instability and steady-state deviations, the proposed method concurrently enhances transient response and robustness. Moreover, by incorporating dynamic constraints and adopting a dynamic particle-weighting mechanism, the convergence speed of the algorithm is enhanced while the overall computational burden is reduced. Simulation results indicate that, in comparison with conventional PLLs and state-of-the-art enhanced structures reported in the literature, the proposed method achieves a shorter settling time, effective suppression of transient oscillations, and zero steady-state error in angle and frequency estimation. Furthermore, the system response time is reduced from approximately 0.5 s to the range of 0.004–0.01 s, while loop stability is preserved under disturbed operating conditions. With its simple structure and low memory requirements, the proposed method is suitable for implementation on lower-performance microcontrollers.

Keywords: Phase-Locked Loop (PLL); Particle Swarm Optimization (PSO); Synchronous Reference Frame (SRF); PI Controller Tuning; Grid Synchronization; Dynamic Response

1. Introduction

In modern power systems, precise estimation of grid phase and frequency is not merely a routine operational requirement but a critical prerequisite for ensuring the stable and reliable operation of a wide range of power electronic devices. With the rapid expansion of renewable energy sources, smart inverters, variable-speed drives, and sensitive control equipment, the dependence on fast and accurate measurement of key system parameters has increased more than ever. Devices such as D-STATCOMs [1], active power filters [2], photovoltaic converters [3], energy storage systems [4], uninterruptible power supplies [5], and various AC drives [6] require precise synchronization with the grid voltage for proper operation. Even small errors in estimating the phase angle or frequency variations can result in unwanted circulating currents, power quality deterioration, increased electrical stress on components, and potentially, system-wide instability.

The increasing penetration of non-linear loads and inverter-based distributed resources has significantly distanced modern grid structures from their traditional counterparts. The presence of harmonics, three-phase unbalancing, voltage sags, rapid frequency fluctuations, voltage flicker, and transient disturbances has transformed the grid into a dynamic, complex, and highly variable system [7]. Under such conditions, grid parameter estimation methods must possess not only high speed and precision but also a substantial degree of stability and robustness against disturbances. Particularly in inverter-based systems, where the grid short-circuit capacity is low and the dynamic behavior is highly sensitive to minor fluctuations, an inadequate angle estimator can significantly impair the performance of the entire control system [8].

Among the numerous methods available for angle and frequency estimation, Phase-Locked Loops (PLLs) [9] have been adopted as the standard solution in many power electronic applications due to their structural

simplicity, high reliability, fast dynamic response, and ease of implementation in digital controllers. Fundamentally, a PLL is a closed-loop feedback system that continuously compares the grid reference signal with an internally generated signal, aiming to drive the phase difference to zero. Typically, this structure consists of three main components [10]: a phase detector to extract the instantaneous phase error, a loop filter to smooth and process the error, and a voltage-controlled oscillator or an angle observer to generate a signal synchronized with the grid. The precise and rapid performance of the PLL is directly dependent on the architecture of the phase detector and the design of the loop filter. Even slight variations in the coefficients of this stage can cause oscillations, slow response, and potentially system instability.

Among various PLL types, the Synchronous Reference Frame PLL (SRF-PLL) [11] has become one of the most widely adopted methods due to its relatively high stability, low sensitivity to harmonics, and disturbance rejection capabilities utilizing the dq0 transformation. In this method, the grid voltage is mapped into the dq synchronous reference frame, and the phase error is extracted directly from the q-axis component [12]. While this architecture outperforms many PLL variants, the precise tuning of the PI controller gains within the loop filter remains one of the most critical and challenging design stages. This controller must achieve a balance among synchronization speed, loop stability, steady-state error, and disturbance rejection. If coefficients are chosen incorrectly, oscillations, increased settling time, steady-state errors, or even synchronization failure may occur. Although traditional heuristic rules, most notably Ziegler–Nichols [13], are favored for their straightforward implementation, they are inherently limited by their dependence on linear approximations. Consequently, these techniques frequently fail to provide adequate robustness when applied to systems exhibiting significant non-linearities and dynamic grid disturbances. Adaptive and self-tuning methods, while effective in certain conditions, are often unsuitable for low-cost inverters due to high computational complexity [14, 15]. Intelligent methods like fuzzy control or neural networks [16] also present limitations regarding operating point dependency, tuning difficulty, and noise sensitivity [17].

Recently, nature-inspired computational techniques have emerged as superior alternatives for control parameter synthesis. These meta-heuristic strategies are particularly advantageous for handling the complex constraints of modern power electronics, offering a reliable solution where analytical derivation is impractical [18]. Unlike classical gradient-based methods, these algorithms do not require a precise mathematical model of the system. Consequently, they offer significantly more reliable performance, particularly in controllers characterized by non-linear dynamics, multi-objective criteria, and susceptibility to extensive disturbances [19, 20]. Evolutionary algorithms such as the genetic algorithm, Particle Swarm Optimization (PSO), imperialist competitive algorithm, and ant colony optimization, along with nature-inspired methods like the grey wolf optimizer or whale optimization algorithm, each employ a unique approach for exploring the search space [21]. However, in practice, fundamental differences are observed among them. For instance, the genetic algorithm relies on mechanisms such as mutation and crossover and can deliver satisfactory results in problems with small to medium-sized search spaces [22]. However, its slow convergence speed and high computational complexity render it less suitable for real-time applications, such as PLL controllers in grid-connected inverters [23]. Similarly, while ant colony optimization has proven successful in routing, scheduling, and discrete optimization tasks, it exhibits limited efficiency in continuous domains—such as PI controller tuning—and requires precise and sensitive initialization of parameters. Newer algorithms like the grey wolf optimizer and the whale optimization algorithm exhibit stable search dynamics. However, they often suffer from slow convergence in the final stages, requiring increased iterations to reach the optimum.

Consequently, PSO has gained considerable popularity due to its structural simplicity, minimal tuning parameters, and computational efficiency suitable for digital implementation [24]. Inspired by swarm behavior, PSO effectively balances global exploration and local exploitation. In including PI controller tuning for PLLs, it demonstrates superior convergence speed compared to the genetic algorithm and the imperialist competitive algorithm [25]. Particularly in power systems, where computational speed and processing capability are of critical importance, the low computational burden of PSO renders it well-suited for implementation in embedded systems and digital signal processors. Nevertheless, the standard PSO is not free from limitations. One well-known limitation of this algorithm is premature convergence to local minima, particularly when the objective function is multimodal or the search space is large and rugged [26]. Moreover, oscillatory behavior of particles in the vicinity of the optimum may increase convergence time or even lead to algorithmic instability [27]. Furthermore, improper tuning of inertia and weighting parameters results in either overly aggressive or sluggish behavior, both of which

are detrimental to PLL PI control. Consequently, advanced variants—such as adaptive inertia PSO, Hybrid genetic algorithm-PSO, PSO with advanced population topologies, and comprehensive learning PSO—have been developed to enable more accurate and stable exploration with greater precision and stability. These enhanced variants preserve the rapid convergence of PSO while improving its capability to escape local minima, thereby enabling a faster and more reliable tuning process of the PI controller in PLL applications.

Given the inherently nonlinear and highly sensitive nature of PLLs and their strong dependence on grid operating conditions, the optimization algorithm must be capable of selecting control parameters without inducing oscillations, such that stability, steady-state error, response time, and disturbance rejection are simultaneously improved under varying grid conditions. Comparative analysis indicates that PSO—particularly its enhanced variants—achieves a superior balance of speed, accuracy, stability, and implementation simplicity compared to genetic algorithm, imperialist competitive algorithm, grey wolf optimizer, and whale optimization algorithm. Consequently, PSO has emerged as a premier choice for designing rapid, robust controllers for SRF-PLL applications [28].

In this paper, a comprehensive cost function is introduced that incorporates transient performance metrics (rise time, settling time, and overshoot), steady-state criteria (frequency error and steady-state error), and system stability indices, enabling an optimal multi-objective tuning of the PI controller within the SRF-PLL framework. Furthermore, analytical stability constraints based on the Routh–Hurwitz criterion and the closed-loop model are imposed to ensure that the optimized coefficients remain within the dynamic stability region. In addition, the adoption of a dynamic weighting mechanism and adaptive search range adjustment allows the PSO algorithm to simultaneously integrate global exploration and local exploitation, thereby avoiding entrapment in suboptimal solutions.

The structure and content of the subsequent sections are organized as follows to clarify the research process: Section 2 presents the research methodology; specifically, Section 2.1 addresses the analysis of the SRF-PLL structure and operation. Then, in Section 2.2, a review of the PI controller tuned using the PSO algorithm is provided. Subsequently, Section 2.3 is dedicated to the definition and explanation of the cost function utilized for the optimization process, and Section 2.4 briefly outlines the steps and operating principles of the PSO algorithm. Section 3 provides a comprehensive analysis of the simulation outcomes, while Section 4 summarizes the final conclusions.

2. Methodology

2.1. SRF-PLL Analysis

The three-phase grid voltage signals are expressed as follows:

$$\begin{cases} v_a(t) = V_m \cos\theta \\ v_b(t) = V_m \cos(\theta - \frac{2\pi}{3}) \\ v_c(t) = V_m \cos(\theta - \frac{4\pi}{3}) \end{cases} \quad (1)$$

where $v_a(t)$, $v_b(t)$, and $v_c(t)$ denote the voltages of phases A, B, and C, respectively. Furthermore, V_m and θ represent the RMS voltage and the phase angle of the three-phase signal, respectively. Applying the Park transformation to (1) yields:

$$\begin{cases} v_d(t) = V_m \cos(\theta - \hat{\theta}) \\ v_q(t) = V_m \sin(\theta - \hat{\theta}) \end{cases} \quad (2)$$

where $\hat{\theta}$ denotes the angle estimated by the PLL, and the Park transformation is performed using this angle. According to the relationship $v_q(t) = V_m sin(\theta - \hat{\theta})$, the control loop aims to achieve $\theta = \hat{\theta}$ in (2), thereby ensuring accurate angle estimation.

Figure 1(a) illustrates the block diagram of the SRF-PLL. Given that the approximation $sin\ (\theta - \hat{\theta}) \cong (\theta - \hat{\theta})$ holds valid for small phase errors, the scheme in Figure 1(a) can be linearized as depicted in Figure 1(b). This linearized model provides valuable insights into the characteristics of the SRF-PLL. Accordingly, the voltage

amplitude V_m acts as a loop gain, thereby influencing the system's phase margin and gain margin. Moreover, the value of V_m directly impacts the dynamic response of the system.

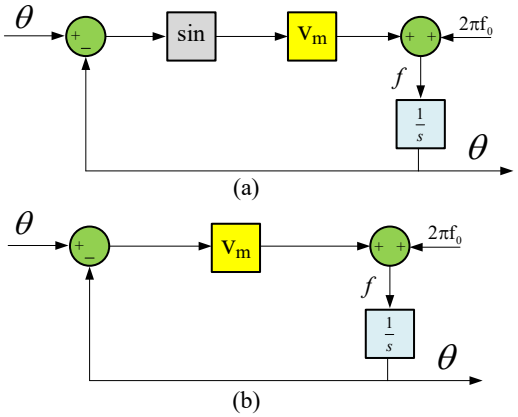


Figure 1. (a) Block diagram of SRF-PLL, (b) Simplified SRF-PLL.

The closed-loop transfer function of the system in Figure 1(b) can be written as follows:

$$G_{CL}(s) = \frac{V_m k_i + V_m k_p s}{s^2 + k_P V_m s + k_i V_m} \tag{3}$$

The coefficients tuned using the PSO algorithm require knowledge of the limit ranges of k_p and k_i, such that the closed-loop system is stable. To determine the mentioned variables and guarantee the stability margin, the Routh–Hurwitz method is used. Accordingly, if the characteristic coefficients in (3) are greater than zero, the system is stable.

2.2. PSO-PI Review

The PI controller constitutes the core of the SRF-PLL loop filter compensator and directly determines the dynamic behavior of the closed-loop system. In the continuous-time domain, the controller output is expressed as follows:

$$u(t) = k_p e(t) + k_i \int_0^t e(\tau) d\tau \tag{4}$$

where K_p and K_i are the proportional and integral coefficients, respectively. The controller transfer function in the Laplace domain is given by:

$$C(s) = k_p + \frac{K_i}{s} \tag{5}$$

This structure introduces a pole at the origin and a real zero, the location of which is determined by the ratio k_p/K_i. The proportional coefficient, K_p, governs the effective loop bandwidth and the tracking speed of the PLL, while the integral coefficient, K_i, eliminates the steady-state phase error and modifies the slope of the compensator response at low frequencies. Improper tuning of (K_p, K_i) may lead to insufficient damping, increased overshoot, or even instability, as the SRF-PLL dynamics are highly sensitive to the loop filter parameters.

Since the PLL dynamics vary with grid voltage amplitude, frequency disturbances, and harmonic distortion, traditional tuning methods (such as Ziegler–Nichols) offer limited reliability. Analytical methods based on linearized models require precise knowledge of system parameters and generally cannot guarantee stability across the entire operating range. Consequently, optimization-based tuning has gained significant attention as a reliable alternative for determining the controller coefficients under non-linear and time-varying conditions.

Table 1 summarizes the characteristics of various PI controller tuning methods, including genetic algorithm-based optimization, fuzzy controllers, neuro-fuzzy techniques, PSO-based tuning, and classical Ziegler–Nichols rules. The genetic algorithm requires extensive parameter tuning and entails a high computational burden. Fuzzy

and neuro-fuzzy controllers rely on subjective rule design or data-driven training, leading to increased implementation complexity. Despite their simplicity, classical Ziegler–Nichols rules typically result in excessive overshoot and insufficient phase margin in PLL systems. In contrast, the PSO algorithm requires fewer tuning parameters, necessitates no training data, and exhibits favorable convergence behavior toward global minima. Consequently, PSO is well-suited for real-time or embedded implementations where computational resources are limited, and fast, stable convergence of the PLL dynamics is required.

Table 1. Comparison between PI controller tuning methods.

Algorithm Name	Training Data Required?	Function Evaluation	Set Parameters	Reference Paper
Genetic algorithm	No	No	Many parameters need to be set	[29]
Fuzzy logic controller	No	Yes	Simple parameter setting	[30, 31]
PSO	No	No	Simple parameter setting	[32]
Neural fuzzy logic controller	Yes	Yes	Many parameters need to be set	[33]
Ziegler-Nichols	No	No	Simple mathematical calculation	[13]

2.3. Cost Function

To effectively calibrate the PI parameters, establishing a precise quantitative metric is essential. This metric serves as the guiding criterion for the minimization process. While indices like mean squared error are common, this work defines a composite objective function to address specific synchronization requirements. The cost function of the system in this paper is defined as follows:

$$F(x) = w1 * Mp + w2 * RT + w3 * ST + w4 \int e^2 \, dt + w5 * sser \tag{6}$$

where Mp, RT, ST, e, and $sser$ are the maximum overshoot, rise time, settling time, error, and steady-state error of the system step response, respectively. $w1$, $w2$, $w3$ and $w4$ are weighting coefficients. As can be seen in the cost function, the controller coefficients can be tuned based on the adjustment of weighting coefficients and the control strategy.

2.4. PSO Algorithm

The PSO algorithm is a powerful optimization technique inspired by the collective behavior of groups of living organisms, such as bird flocks and fish schools. In this algorithm, each particle represents a candidate solution to the optimization problem and is characterized by a position and velocity within the search space. At the beginning of the algorithm, the particles are randomly distributed across the search space, and each particle possesses an initial solution. Subsequently, each particle is evaluated based on an objective function to determine the quality of its solution. Each particle stores its best personal position so far, and the best position of the entire population is also continuously updated. In each iteration, particles move with a combination of movement towards the personal best position and the global best position, with the addition of a random component, to both explore new points in the search space and be guided towards optimal solutions. This process causes the particles to gradually approach optimal regions, and the best possible solution within the entire population is discovered. Equation (7) shows the velocity and position update mechanism of a particle in the PSO algorithm:

$$\begin{aligned} V_i^{n+1} &= \omega V_i^n + C_1 r_1 (P_i^n - X_i^n) + C_2 r_2 \left(P_g^n - X_i^n\right) \\ X_i^{n+1} &= X_i^n + V_i^{n+1}, \qquad i = 1, \dots, N \end{aligned} \tag{7}$$

where V_i^n and X_i^n denote the velocity and position of the i-th particle at iteration n, respectively. P_i^n represents the best position experienced by the ith particle up to iteration n, and P_g^n is the best position found by all particles up to iteration n. Furthermore, C_1 is the social rate, C_2 is the cognitive rate, n is the iteration number, r_1 and r_2 are

random variables in (0,1), and ω denotes the inertia weight. The PSO algorithm is employed to search for the optimal PI controller parameters, as illustrated in Figure 2. The parameters of the PSO algorithm are set as follows: population size (N) is 75, maximum number of iterations (I) is 100, weighting coefficients (C_1 and C_2) are 2, and the inertia constant (ω) is 1.

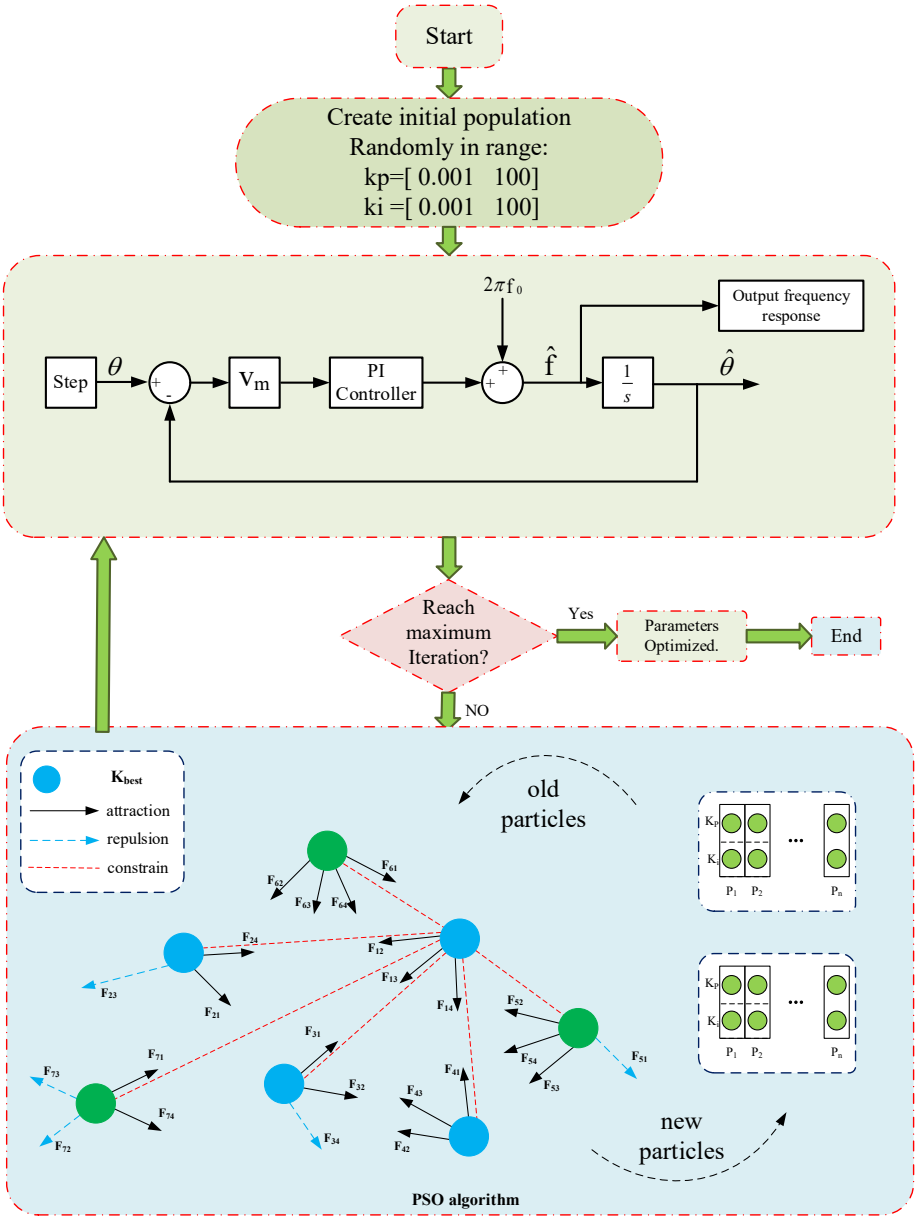


Figure 2. Proposed PSO-based PI controller tuning flowchart.

The inertia weight function regulates the dynamic velocity of the particles and balances global and local search behaviors. Employing a large inertia weight at the early stages of the algorithm, followed by a gradual reduction, promotes global exploration initially and local exploitation in the later stages. In this paper, the inertia weight function is defined as follows:

$$\omega = \omega_{max} - \frac{(\omega_{max} - \omega_{min}).iter}{iter_{max}} \tag{8}$$

where ω_{max} and ω_{min} are the initial and final weights, respectively. $iter$ denotes the current iteration number, and $iter_{max}$ represents the maximum number of iterations.

3. Simulation Results

In this section, the proposed method for tuning the PLL parameters is simulated and analyzed. Figure 3 illustrates the stability of the tuned controller based on the Bode plot. As observed, the gain margin is infinite, and the phase margin is 179.18. Consequently, the tuned PLL system is stable.

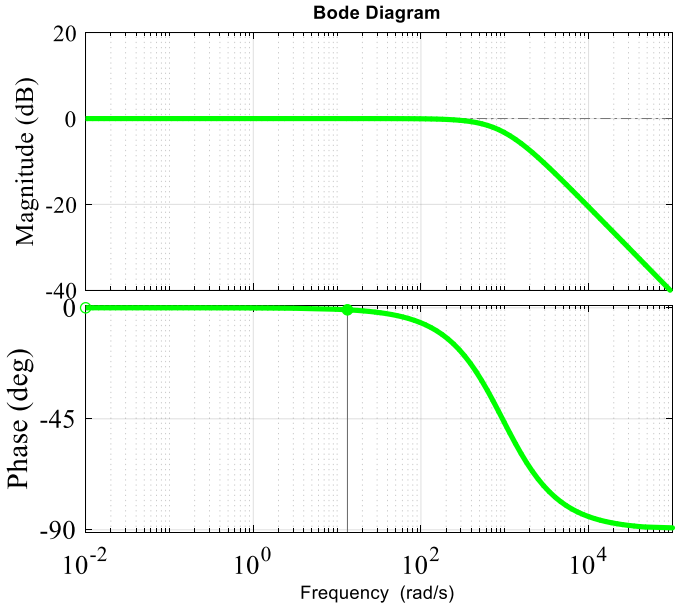


Figure 3. The Bode diagram of the proposed PLL transfer function.

Figure 4 compares the frequency step response of the proposed method with the method presented in [34]. As observed, the settling time in [34] is 0.5 s, whereas it is 0.0045 s in the proposed method. Furthermore, in [34], the overshoot is 45%, while in the proposed method, it is zero.

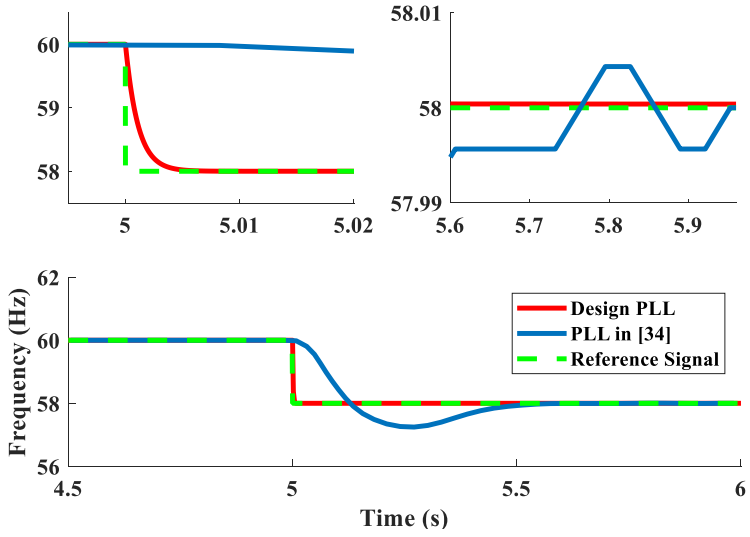


Figure 4. Comparison of the frequency step response of the proposed method and the reference [34].

Figure 5 illustrates the comparison of the angle estimated by the proposed method and that of [34]. In the proposed method, the steady-state error of the angle estimation is zero, whereas the method in [34] exhibits a steady-state error.

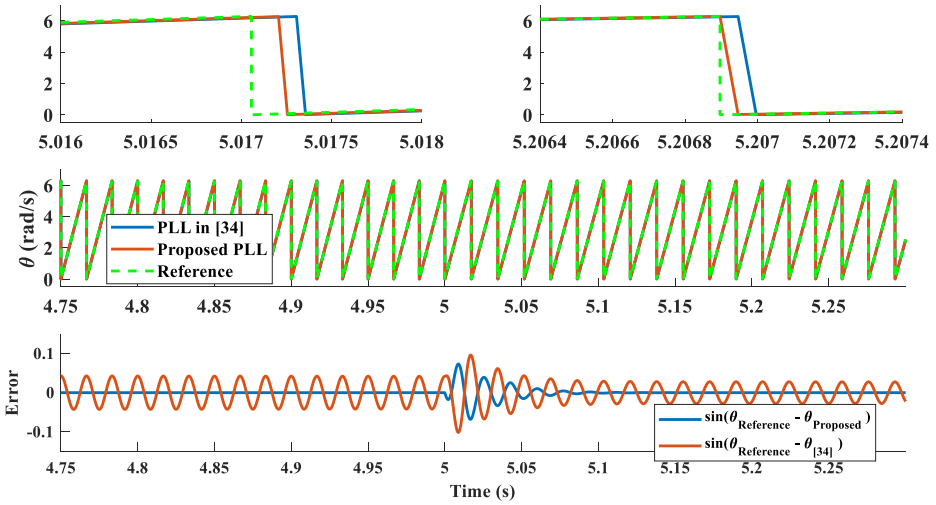


Figure 5. Comparison of estimated angle in the proposed method and [34] due to frequency step.

The authors in [35] presented two methods for tuning the PLL in fast and very fast modes. Figure 6 illustrates the comparison between [35] and the proposed method. As observed, the maximum overshoot/undershoot of qPLL and FF-qPLL are 40% and 10%, respectively, whereas the maximum overshoot/undershoot of the proposed method is zero. Furthermore, the settling times of qPLL and FF-qPLL are 0.15 s and 0.12 s, respectively, while the settling time of the proposed method is 0.05 s.

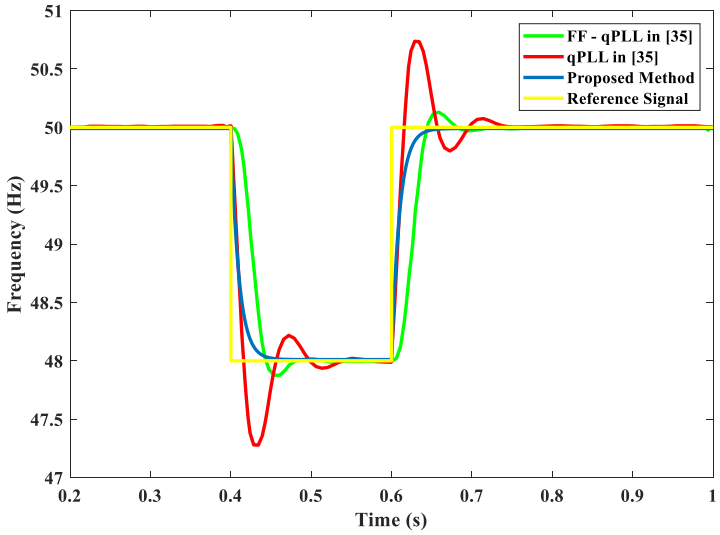


Figure 6. Comparison of the frequency step response of [35] and the proposed method.

4. Conclusion

In this research, an optimization framework based on an enhanced version of the PSO algorithm was proposed for tuning the PI controller parameters within a three-phase SRF-PLL structure, aimed at enhancing dynamic performance and increasing loop reliability under various operating conditions. In contrast to classical methods and linear model-based approaches, the proposed method facilitates the tuning of controller coefficients by accounting for the actual non-linear behavior of the PLL and utilizes a multi-objective cost function to simultaneously improve both transient and steady-state indices. The incorporation of analytical stability constraints and the utilization of a dynamic weighting mechanism resulted in a more stable and structured search process within the PSO algorithm, effectively mitigating the oscillations typically observed during the initial or

final stages of convergence. Numerical validations confirm that the suggested tuning protocol significantly outperforms conventional methods. Specifically, the optimized PLL exhibits rapid convergence in phase and frequency tracking. Moreover, the settling time is drastically minimized, ensuring superior dynamic behavior without compromising stability. Furthermore, comparative analysis with advanced structures such as qPLL and FF-qPLL, as well as existing optimization-based methods in the literature, revealed the distinct superiority of the presented method in terms of loop stability, steady-state error, and disturbance immunity. These improvements have been achieved without increasing computational complexity; consequently, the proposed method is considered a practical and reliable candidate for embedded applications and low-cost controllers.

References

1. Arya, S. R.; Patel, M. M.; Alam, S. J.; Srikakolapu, J.; Giri, A. K., Phase lock loop–based algorithms for DSTATCOM to mitigate load created power quality problems. *International transactions on electrical energy systems* **2020,** *30* (1), e12161.
2. Pazhanimuthu, C.; Baranilingesan, I.; Karthick, A., An improved control algorithm for series hybrid active power filter based on SOGI-PLL under dynamic load conditions. *Solid State Communications* **2021,** *333*, 114357.
3. Pranith, S.; Kumar, S.; Singh, B.; Bhatti, T., Improved Gaussian filter based solar PV-BES microgrid with PLL based islanding detection and seamless transfer control. *IEEE Transactions on Industrial Electronics* **2021,** *69* (6), 5815-5825.
4. Shang, L.; Dong, X.; Liu, C.; Gong, Z., Fast grid frequency and voltage control of battery energy storage system based on the amplitude-phase-locked-loop. *IEEE Transactions on Smart Grid* **2021,** *13* (2), 941-953.
5. Li, Q.; Yan, X.; Wang, Z.; Ma, Y.; Liao, W.; Ma, K.; Zhang, J.; Lei, E.; Jin, L.; Wen, B. In *An Uninterruptible Power Supply and Its Output Voltage Stability Control Method*, 2025 IEEE 12th International Symposium on Sensorless Control for Electrical Drives (SLED), IEEE: 2025; pp 1-7.
6. Salimi, H.; Zakipour, A.; Asadi, M., A novel analytical approach for time-response shaping of the pi controller in field oriented control of the permanent magnet synchronous motors. *Journal of Electrical and Computer Engineering Innovations (JECEI)* **2022,** *10* (2), 463-476.
7. Ali, Z. M.; Ćalasan, M.; Jurado, F.; Aleem, S. H. A., Complexities of power quality and harmonic-induced overheating in modern power grids studies: Challenges and solutions. *IEEE Access* **2024**.
8. Eskandari, M.; Savkin, A. V.; Alhelou, H. H.; Blaabjerg, F., Explicit impedance modeling and shaping of grid-connected converters via an enhanced PLL for stabilizing the weak grid connection. *IEEE Access* **2022,** *10*, 128874-128889.
9. Hsieh, G.-C.; Hung, J. C., Phase-locked loop techniques. A survey. *IEEE Transactions on industrial electronics* **1996,** *43* (6), 609-615.
10. Kroupa, V. F., *Phase lock loops and frequency synthesis*. John Wiley & Sons: 2003.
11. Rodríguez, P.; Pou, J.; Bergas, J.; Candela, J. I.; Burgos, R. P.; Boroyevich, D., Decoupled double synchronous reference frame PLL for power converters control. *IEEE Transactions on Power Electronics* **2007,** *22* (2), 584-592.
12. Priyanka, B.; Lavanya, M. In *Phase-locked loop (PLL) techniques for grid synchronization: A comprehensive review*, 2024 Second International Conference on Emerging Trends in Information Technology and Engineering (ICETITE), IEEE: 2024; pp 1-9.
13. Meshram, P.; Kanojiya, R. G. In *Tuning of PID controller using Ziegler-Nichols method for speed control of DC motor*, IEEE-international conference on advances in engineering, science and management (ICAESM-2012), IEEE: 2012; pp 117-122.
14. Vega, S.; Vasquez-Guevara, M.; Camacho, O., A Comparison of Adaptive PID, Adaptive Dual-PID and Adaptive Fractional PID Controllers for a Nonlinear System with Variable Parameters A Comparison of Adaptive PID, Adaptive Dual-PID and Adaptive Fractional PID Controllers for a Nonlinear System with V. *Res. Publ. Connect* **2024,** *1*, 166-177.
15. Nazemorroaya, E.; Shafieirad, M.; Hajatipour, M. In *Model Reference Adaptive Control for Nonlinear Systems in the Presence of Unknown External Disturbances*, 2022 30th International Conference on Electrical Engineering (ICEE), IEEE: 2022; pp 17-22.
16. Nazemorroaya, E.; Shafieirad, M.; Zahiripour, S. A., Mean square consensus of heterogeneous multi-agent systems in the presence of actuator faults. *Transactions of the Institute of Measurement and Control* **2023,** *45* (6), 1158-1168.
17. Salimi, B.; Zakipour, A.; Nazemorroaya, E., New approach of optimized SHE modulation using genetic algorithm for three-phase half-bridge converter. *Majlesi Journal of Mechatronic Systems* **2023,** *12* (1), 21-26.
18. Enad, M. H.; Hassan, R. F.; Mahmoud, A. A. K.; Humaidi, A. J., Performance evaluation of a 2DOF_PID controller using metaheuristic optimization algorithms. *Journal Européen des Systèmes Automatisés* **2024,** *57* (3), 709.
19. Yang, Y.; Gao, Y.; Wu, J.; Ding, Z.; Zhao, S., Improving PID controller performance in nonlinear oscillatory automatic generation control systems using a multi-objective marine predator algorithm with enhanced diversity. *Journal of Bionic Engineering* **2024,** *21* (5), 2497-2514.
20. Singh, A. P.; Katta, S. S.; Baghel, R.; Yadav, S., Graphene Nanoribbon-FET for Higher Drive Current using Machine Learning-Enhanced First Principles Analysis. *Electrical Engineering and Technology* **2025,** *1* (1), 1-14.
21. Mohamad Radzi, P. N. L.; Mekhilef, S.; Mohamed Shah, N.; Akhter, M. N.; Seyedmahmoudian, M.; Stojcevski, A., Optimizing solar power forecasting with metaheuristic algorithms and deep learning models for photovoltaic grid connected systems. *Scientific Reports* **2025,** *15* (1), 40045.
22. Arzhang, A.; Nazemorroaya, E. In *Genetic-Algorithm-Based Design of a Variable-Order Fractional PID Controller for Artificial Pancreas Systems*, 2026 34th International Conference on Electrical Engineering (ICEE), IEEE: 2026; pp 83-88.
23. Nassef, A. M.; Abdelkareem, M. A.; Maghrabie, H. M.; Baroutaji, A., Metaheuristic-based algorithms for optimizing fractional-order controllers—a recent, systematic, and comprehensive review. *Fractal and Fractional* **2023,** *7* (7), 553.

24. Gupta, M.; Tiwari, P.; Viral, R.; Shrivastava, A.; Zneid, B. A.; Hunko, I., Grid-connected PV inverter system control optimization using Grey Wolf optimized PID controller. *Scientific reports* **2025,** *15* (1), 28869.
25. Tuan, D. H.; Nguyen Ngoc Thanh, V.; Nguyen Chi, D.; Pham, V. H., Improving frequency control of multi-area interconnected hydro-thermal power system using PSO algorithm. *Applied Sciences* **2025,** *15* (6), 2898.
26. Tarekegn, D.; Tilahun, S.; Gemechu, T., A review on convergence analysis of particle swarm optimization. *International Journal of Swarm Intelligence Research (IJSIR)* **2023,** *14* (1), 1-34.
27. Abualigah, L., Particle Swarm Optimization: Advances, Applications, and Experimental Insights. *Computers, Materials & Continua* **2025,** *82* (2).
28. Nassef, A. M.; Abdelkareem, M. A.; Maghrabie, H. M.; Baroutaji, A., Review of metaheuristic optimization algorithms for power systems problems. *Sustainability* **2023,** *15* (12), 9434.
29. Civelek, Z.; Teke, M., Genetic algorithm-based optimization of PID controller parameters for DC motor speed control. *Journal of Computer & Electrical and Electronics Engineering Sciences* **2025,** *3* (2), 47-53.
30. Adnan, A. Q.; Hussain, M. K., A Comparative Study of Optimal Fuzzy Logic Controllers for Blade Pitch Angle in Horizontal-axis Wind Turbines. *International Journal of Renewable Energy Research (IJRER)* **2025,** *15* (3), 537-547.
31. Chotikunnan, P.; Khotakham, W.; Imura, P.; Chotikunnan, R.; Wongkamhang, A.; Thongpance, N., Comparative Analysis of PID-Driven Data-Based and PSO-Tuned Fuzzy Membership Functions for Robotic Manipulator Control. *Journal of Fuzzy Systems and Control* **2025,** *3* (3), 204-211.
32. Yenealem, M. G., Power flow management and control using PSO-PID and fuzzy logic controllers for autonomous solar and wind hybrid systems. *Cogent Engineering* **2025,** *12* (1), 2560977.
33. Kuang, M.; Hou, Q.; Wang, J.; Guo, J.; Wei, Z., GA-Synthesized Training Framework for Adaptive Neuro-Fuzzy PID Control in High-Precision SPAD Thermal Management. *Machines* **2025,** *13* (7), 624.
34. Rasheduzzaman, M.; Kimball, J. W., Modeling and tuning of an improved delayed-signal-cancellation PLL for microgrid application. *IEEE Transactions on Energy Conversion* **2018,** *34* (2), 712-721.
35. Liccardo, F.; Marino, P.; Raimondo, G., Robust and fast three-phase PLL tracking system. *IEEE Transactions on Industrial Electronics* **2010,** *58* (1), 221-231.