IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 18 March 2024, accepted 8 April 2024, date of publication 11 April 2024, date of current version 19 April 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3387704

TOPICAL REVIEW

A Comparative Review of Current Optimization Algorithms for Maximizing Overcurrent Relay Selectivity and Speed

SETHEMBISO NONJABULO LANGAZANE AND AKSHAY KUMAR SAHA, (Senior Member, IEEE)
Discipline of Electrical, Electronics, and Computer Engineering, University of KwaZulu-Natal, Durban 4041, South Africa
Corresponding author: Akshay Kumar Saha (saha@ukzn.ac.za)

ABSTRACT An exponential growth and complexity in diverse distribution systems have contributed to protection coordination challenges. Initially, protection coordination schemes were achieved by means of conventional techniques; however, the utilisation of such methods is based on trial-and-error principles and laborious. Consequently, current studies have adopted the utilisation of particle swarm optimization, artificial intelligence models, and genetic algorithms to optimise overcurrent relay selectivity and operational speed. Particle swarm optimization, artificial intelligence, and genetic algorithms are optimization techniques that at times converges prematurely due to poor selection of control parameters and lack of optimal values, which results in increased computational time. Therefore, this paper presents a comprehensive review of recent developments in terms of parametric sensitivity analysis, selection of artificial intelligence models based on data availability, and the likelihood of solving overcurrent relay coordination problems. The reviewed literature shows that particle swarm optimization performance is greatly influenced by inertia weight and swarm size, while the number of iterations has insignificant effect. The findings also indicate that crossover rate, mutation probability, and population size affect genetic algorithms behaviour. Artificial intelligence models lack sensitivity study for parametric tuning, that is, number of hidden layers, membership functions, epsilon in support vector machine, and number of fuzzy rules affects the models' performance.

INDEX TERMS Adaptive neuro-fuzzy inference system, artificial intelligence, artificial neural networks, control parameters, genetic algorithms, overcurrent relay, particle swarm optimization, power system protection, protection coordination, selectivity, sensitivity analysis, speed.

ABBREVIATIONS

ABC	Artificial Bee Colony.
AC	Alternating Current.
AI	Artificial Intelligence.
ANN	Artificial Neural Network.
ANFIS	Adaptive Neuro-Fuzzy Inference System.
CTI	Coordination Time Interval.
DE	Differential Evolution.
FO	Fractional Order.
GA	Genetic Algorithms.
IDMT	Inverse Definite Minimum Time.
IEC	International Electrotechnical Commission.
MAPSO	Modified Adaptive Particle Swarm Optimization.
MFs	Membership Functions.
OBL	Opposition Based Learning.
PSO	Particle Swarm Optimization.
PSM	Plug Setting Multiplier.
TMS	Time Multiplier Setting.

The associate editor coordinating the review of this manuscript and approving it for publication was Arturo Conde.

I. INTRODUCTION

During last decades, power systems engineers and researchers relied on traditional optimization methods for coordinating overcurrent relay. The drawback of the techniques is that solutions are based on trial-and-error, and the process is laborious as well as time-consuming [1], [2]. Therefore, [3] and [4] advocated the importance of adopting evolutionary algorithms to overcome disadvantages presented by traditional optimization techniques. Authors in [5] also mentioned the need to utilise meta-heuristic algorithms to eliminate drawbacks presented by conventional techniques. Particle Swarm Optimization (PSO) and Genetic Algorithms (GA) have emerged as efficient and effective techniques for handling protection coordination problems. However, tuning particle swarm optimization and genetic algorithms control parameters to attain optimum overcurrent relay settings is a long-standing problem [6]. A comparative study conducted in [7] demonstrated that GA algorithm fails to converge efficiently due to premature convergence caused by loss of diversity which results in population deteriorating to local optima.

In order to improve converging speed and succeed premature convergence of genetic algorithms, the concept of altering crossover and mutation probability has been employed in [5], [8]. The significance of choosing crossover and mutation probability in such a way that GA behaviour is robust and efficient has been documented in research work [7], [8], [9], [10], [11]. Rojas et al. [8] performed a basic genetic algorithm sensitivity study where one genetic operator was varied at a time to analyse and study system behaviour, and it was discovered that mutation rate, crossover type, and population size have negligible effect on genetic algorithm performance while crossover probability has substantial impact [8]. In [11], author urged that mutation probability must accelerates as population diversity increases [11]. Another sensitivity study conducted in [12] and [13] showed combination of crossover and mutation probabilities yields optimal global solution [12], [13]. Nevertheless, the best optimum solution is attained when integrating a low mutation probability with a high crossover rate. A review supplying dynamic method for selecting mutation and crossover probabilities is presented in [9], and a further study in [10] reviewed the application of GA control parameters, pointing out the pros and cons of genetic operators whereas in [5], a review of control parameters was presented stating past, present, and future methods of assessing genetic algorithms.

In contrast, a study comparing PSO algorithm behaviour with other meta-heuristic methods shows that PSO algorithms manage to attain global minima with fewer number of iterations [14]. However, PSO algorithm presented limitations such as failing to fly pass local optima region hence, not succeeding to reach global optimal solutions. Therefore, research work in [14] and [15] presented alternative solutions for improving PSO algorithm performance significantly. An experimental study [15] established the introduction of weight term into velocity update equation to enhance particles' search capability by stabilising local and global search. Swarm size and inertia weight are extremely associated with algorithms' converging prematurely. Kennedy et al. [16] categorised velocity equation into three segments for successful controlling of particles' previous velocity to current momentum. Reference [17], studied the behaviour of PSO control parameters, it was discovered that both cognitive and social components contained in the velocity update equation must not be greater than 4 [17]. A further study in [18] presented hypothetical assessment when both cognitive and social components exceed 4, the particles experienced higher oscillations. These analysis operates as guidance and helps to improve swarm search ability and avoid premature convergence [17], [18]. Another work in [19] investigated the performance of PSO algorithm by using design experiments for decreasing simulations runs; nonetheless, it does not allow individual parametric sensitivity analyses thus, not much was gained from the study [3], [19].

Chui et al. [20] reduced Artificial Neural Network (ANN) ineffectiveness by introducing radial basis function which is a training strategy designed to predict time series by adaptive studying algorithm and response surface methodology. Nevertheless, radial basis function network involves determining the optimal core parameters and kernel function width. Further study in [21] utilised ANN to identify faulty power line and location of the faulty from the source, thereafter, a backtracking algorithm was adopted to coordinate the primary and backup relay such that selectivity is maximised [21]. The findings were compared with other studies of ANNs presented in [22] and [23] which revealed it performs better with hybrid algorithm. In [24], adaptive neuro-fuzzy inference systems (ANFIS) managed to outperform the ANNs due to its advantageous utilization of hybrid model between neural network and fuzzy logic. Consequently, this paper intends to present a parametric sensitivity analysis for particle swarm optimizer and genetic algorithms, a comprehensive review of artificial intelligence models, and a comparative evaluation with respect to convergence speed and fitness values. Table 1 highlights the gaps in the existing research and the contributions of the proposed paper. The contributions of the paper are as follows:

a) A comprehensive review of evolutionary algorithms with respect to parametric sensitivity analysis based on convergence speed and fitness function values.
b) Analysis of overcurrent relay behaviour based on various literature to evaluate whether protection selectivity and operational speed is accomplished.
c) To review the application of artificial intelligence models in overcurrent protection coordination and the concept of adaptive control parameters.

Selectivity and speed studies are presented in section II. An overview of protection philosophy is presented in section III. Section IV provides algorithms that optimises overcurrent relay coordination. Selectivity and speed analysis in terms of convergence rate are presented in section V. Artificial intelligence models for optimizing overcurrent relays are provided in section VI. Lastly, recommendations for future studies as well as conclusive remarks, are presented in section VII and section VIII, respectively.

TABLE 1. Summary of literature survey.

Algorithm	Current approach for tuning parameters	Proposed approach
Particle swarm optimization	In [15], authors suggested the use of larger inertia weight in the beginning, thereafter, slowly decreasing to minimal value. However, the time-varying based inertia weight variation may not lead to a global optimal solution. Ref. [19] investigated the performance of PSO algorithm by using design experiments for decreasing simulations runs; nonetheless, it does not allow individual parametric sensitivity analyses thus, not much was gained from the study.	Parametric sensitivity analysis is done for self-adaptive control parameters such as self-adaptive mutation with dynamic non-linear inertia weight, evolutionary state-based acceleration coefficients, and adaptive velocity update to further improve PSO algorithm performance.
Genetic algorithm	Reference [12] and [13] adopted a combinational strategy that tunes both mutation and crossover probabilities simultaneously to observe algorithm behaviour. Similarly, authors in [8] varied crossover and mutation probabilities at the same time. The approaches only focused on crossover and mutation probabilities which are not the only genetic operators that affect the behaviour of GA algorithm. Due to the variation of parameters simultaneously, it was difficult to determine poor performing parameters.	This paper reviews current analysis of crossover probability, mutation probability and population size. Moreover, sensitivity analysis to evaluate parameters that significantly affects algorithms' performance is presented. Parametric analysis that changes on parameter at a time while keeping others constant is proposed. Analyses are performed based on convergence speed, selectivity, and fitness function.
Artificial intelligence	Authors in [24] utilised ANFIS structure generating adaptive pickup current and time multiplier setting parameters based on fuzzy rules to provide input/output pairs. An improvement in performance was achieved when ANFIS model used seven numbers of Gaussian-type membership functions (MFs) and more precise findings were obtained using five numbers of triangular-type MFs [24]. Nonetheless, the use of least-squares estimation in training leads to complex information extraction in ANFIS relays in comparison to simple and straightforward fuzzy logic overcurrent relays.	As a result, this paper outlines the effect of model architecture, quantity of *If-Then* rules, data processing speed, membership function, and the anticipated output. Comparative study is presented for AI models with respect to overcurrent relay coordination.

II. SELECTIVITY AND SPEED STUDIES

The concept of relay selectivity measures system reliability and effectiveness by ensuring continuous power supply even in instances when abnormal conditions occur in certain protected zones. It plays an important role in segregating faulty section while leaving the healthy part intact and functional for continual supply of power to end-users. In a study attempting to investigate the factors affecting selectivity and operational speed of relays in [25], Sorrentino et al. proved that it is not possible to achieve selectivity for mesh or ring distribution network with more than two equivalent power sources due to identical currents seen by the relay. Author in [26] and [27] improved protection selectivity as well as speed of operation by using time grading margins of 0.2 and 0.3 seconds, respectively. Distribution system constraints such as safety factor, circuit breaker interrupting time, maximum fault currents, and load currents were taken into consideration which resulted in protection miscoordination and longer fault clearing time. According to [28] relay selectivity in a radial distribution network made up of one feeder and unidirectional power flow can be attained by precise time grading of the overcurrent relay, whereas other configurations such as ring and mesh networks, the selectivity is not adequate for system protection; thus, directional overcurrent relay is favourable for such systems [28]. In [29], it was explicitly stated that majority of distribution network adopts time grading method to achieve selectivity. However, both techniques result in higher tripping times which have detrimental effects on power systems expensive equipment and safety of personnel [28], [29]. Depending on the type of distribution network setup, heuristic, hybrid, and artificial intelligence techniques are preferred to maximise selectivity of relays and speed of operation since these work in conjunction to optimise protection on the network. Furthermore, these optimization techniques mitigate iterative trial and error as well as laborious process which is utilised in conventional methods [30]. Therefore, there is a necessity to review optimization techniques for coordinating overcurrent relays such that maximum selectivity and optimised operational speed is obtained and to advance converging speed and circumvent premature convergence of some algorithms.

III. OVERVIEW OF PROTECTION PHILOSOPHY

In a power distribution system, the reliability of electrical protection system is as critical as selectivity, sensitivity, and speed since these protection characteristic parameters are dependent such that an increase in one parameter leads to the other decreasing [3], [30], [31]. For this reason, protection quality was improved in [32] through the utilisation of analysis tools to evaluate all protection system characteristics. Similarly, authors in [31] studied energy distribution redundancy and reliability, sensitivity, speed, and selectivity is not included in [31]. Nevertheless, the main attributes of insignificant protection selectivity and speed of overcurrent protection were not addressed. System disturbances triggered by unanticipated faults, load transients, and maloperation of electrical equipment require protection devices to respond timely, speedily, and selectively [30]. Protective system comprises circuit breakers, relays, and other circuit interrupters to isolate misoperational equipment. Circuit breakers function to isolate abnormally operating part of the distribution network when triggered by the overcurrent relay, which senses, localises a fault and issues a trip command to the circuit breaker for promptly discrimination of faulty section [30], [33]. Protection system must satisfy the predominate objectives as follows [30], [33], [34]:

- To minimise duration of the fault.
- To protect customers' apparatus and continual power supply.
- Reduction of overhaul outages to slightest section of the system.
- Improvement of system performance, stability, and reliability.
- Deisolation of defective power lines, transformers, and other apparatus timeously.

A. OVERCURRENT PROTECTION

Overcurrent relay provides protection against excessive current magnitudes in distribution networks due to malfunctioning of the system [30], [34]. These extreme current magnitudes can be used to signify the existence of faulty conditions and help activates the operation of protection devices accordingly, which vary in system complexity and design requirements [30], [33], [34]. Auxiliary devices such as current transformers and voltage transformers act as protection and metering components that measure current and voltage levels which serves as input signals into overcurrent relay [34]. The received signals are analysed and compared to the predetermined values. Distribution networks are typically safeguarded by overcurrent relays against extreme currents due to added advantage of being cost-effective and thus preferred on distribution level in comparison to distance and differential protection [29].

At times, primary protective devices are bound to maloperation in the distribution network due to breakdown in circuit breaker trip mechanism, inadequate trip coil voltage, or faulty protection relay [30]. Consequently, backup protection provides a second line of defence to avoid catastrophic damage to the scheme by detecting and clearing abnormalities from the system. However, in order to enforce sequential operation of primary and backup protection, optimum protection coordination between relays is essential. Kudkelwar et al. [35] formulated overcurrent relay operating time mathematically through the objective function as follows [13].

$$Min\,(C) = \sum\nolimits_{d=1}^{k} W_o \times S_{DJ} \tag{1}$$

where C the objective function, k the total number of relays installed, W_o the weight factor, and S_{DJ} is the functioning time of D_{th} main protection at its assigned zone J. The weight factor W_o can be set at approximately one due to its equivalent shorth line length and weight [35]. With the objective of maximising protection coordination between upstream and downstream relay, standard inverse time characteristic curve is employed in [30]. According to IEC 60255-151:2009 [33], inverse definite minimum time class for overcurrent protection scheme adopt the standard inverse characteristic equation as follows:

$$T_{op} = \left(\frac{0.14}{PSM^{0.02} - 1}\right) \times TMS \tag{2}$$

where T_{op} the relay operating time, TMS the time multiplier setting, and PSM the plug setting multiplier. It was stated in [33] that T_{op} values are dependent on utmost fault current level, IDMT curve type, and the operating time of downstream relays whereas PSM contains the ratio of maximum fault current I_F to pick-up current I_P [33], [37]. The overcurrent relay issue trip command when $I_F \geq I_P$ but retain normal operating state if $I_F < I_P$ [36] under TMS values ranging between $0.01 < TMS < 1.0$ seconds [37]. When maximum fault current I_F is equal to pickup current I_P, overcurrent relay prompts the operation of the circuit breaker to isolate the faulty section. According to [34] I_P settings for phase-to-phase faults can be selected between 50% and 200% in steps of 25% whereas authors in [37] suggested 10% to 70% in steps of 10% is sufficient for earth leakage faults. Therefore, I_P is the product of current transformer secondary current and current setting. Current setting can be defined as the adjustment of tappings on the relay coil to obtain the desired relay pickup current. The more current setting the relay has, the greater current the relay needs to send the trip command [36], [37]. References [31] and [32] presented two common approaches to compute the pickup current. The first method stipulates that the pickup current is twice the maximum load current, or it must be one-third of the minimum fault current at the nearest busbar [31]. Second method proposes that pickup current must be selected between 125% of the maximum load current and 2/3 times of minimum fault current [32].

It is challenging to execute non-linear standard inverse characteristics like (1) directly due to it exponential expression in the order of 0.02. It is imperative to indicate that former researchers and scholars have prevented this inverse definite minimum time characteristic in their work [38], [39]. Due to this constraint, Amin et al. [40] proposed the exploitation of artificial intelligence-based optimization techniques, particularly neural networks by using universal function approximation capabilities. Reference [30] and [40] utilised evolutionary algorithm and artificial intelligence models, respectively, to accomplish optimal protection coordination and assure precise sequential operation of upstream and downstream relays which is achieved by coordination time interval (CTI) [30], [31]. Under normal system conditions, the backup protection is not active until CTI exceeds the predetermined value [30]. Once the CTI is exceeded, the backup relay must operate within coordination constraints as formulated in the following equation [30], [31]:

$$T_{backup} - T_{main} \geq CTI \tag{3}$$

where T_{backup} the backup protection operating time, T_{main} the time of operation for main protection. The backup relay T_{backup} must function within the coordination constraints of with coordination time interval set between 0.1 seconds and 0.2 seconds for microprocessor-based relay, while electromagnetic relay utilises 0.3 seconds to 0.4 seconds [31], [32].

IV. ALGORITHMS TO OPTIMISE OVERCURRENT RELAY COORDINATION

Scholars have been exploring different algorithms to find solutions to complex and multi-dimensional problems, and taking into consideration factors such as problem complexity, computational strength, time availability, and understanding of function behaviour, an optimization algorithm may be chosen. Nature is commonly utilised as an inspiration in establishing and solving sophisticated optimization techniques such as hybrid algorithms which are utilised to enhance converging behaviour of meta-heuristics.

In this section, algorithms for optimizing overcurrent relays and their developments are discussed.

A. PARTICLE SWARM OPTIMIZATION

To evaluate distinctive optimization techniques that will yield superior performance throughout optimization problems, a statistical approach demanding unfeasibly great number of simulations may be required, which is not practical; however, greater certainty can be set on techniques that constantly achieve improved results than others. Particle swarm optimization has the capability to utilise its memory for updating particles' location which other nature-inspired algorithms lack this feature. In contrast with other meta-heuristic algorithms such as ant colony optimization [41] and firefly algorithm [42], studies have proven PSO robustness and efficiency in attaining optimal solutions [14], [41], [42]. Nonetheless, PSO at times experiences a condition that leads it to converge prematurely to local solution due to incapability of particles to fly from local minima. Poor setting and selection of PSO control parameters may contribute significantly to unsatisfactory performance and premature convergence [30]. Additionally, PSO algorithm may be inefficient as it needs higher function evaluations to obtain optimum solutions in hyperdimensional problems [43]. Consequently, new PSO variants are developed to address algorithms' premature convergence and improve converging speed.

Table 2 depicts some state-of-the-art PSO algorithm variants. An earlier study modified PSO algorithm to find solutions to multi-objective [34] and discrete [45] optimization problems. A number of PSO algorithm variations have been introduced either by incorporating theories of PSO algorithm with other meta-heuristic techniques or by developing novel mechanisms. In [46], PSO was coupled with differential evolution (DE) for constraint handling by using feasibility principles to obtain best personal solution of individual particles in the swarm. Furthermore, a study in [15] introduced the weight expression into velocity update equation to improve search capability of particles by stabilising search abilities. Reference [16] categorised velocity update equation into three segments to control the effects of particles' past velocity to prevailing speed. In a study aiming to learn the performance of PSO algorithm particles [17], it was proposed that acceleration coefficients in velocity update equation must not exceed 4. A further analysis in [18] demonstrated that when acceleration coefficients exceed 4, the particle tends to experience higher oscillations. The literature reviewed serves as a guide and assists in enhancing swarm search abilities and avoiding converging prematurely [17], [18]. Another work in [19] examined PSO algorithm behaviour using design experiments, this method decreased the number of runs; nonetheless, it does not allow analyses of individual parameter, thus not mush was discovered [19].

Constraint handling technique problems were highlighted in [47] as crucial in effective performance of PSO, it was clearly stated that constraint handling approaches provide essential information with regards to solution feasibility [47]. Therefore, PSO algorithm modifications are necessary to handle constrained optimization problems. Reference [6] proposed adaptive constraint handling approach that prefers any feasible solution over unfeasible ones, and among two feasible solutions, the one with best objective function values is favoured. It was observed in [48] that by choosing feasible solutions only, the algorithm position itself to favour feasible

TABLE 2. Some state-of-the-art PSO variants.

PSO Variant	Description
CVI – PSO [49]	In CVI-PSO, constraints are handled by attracting problem solutions to the feasible search regions through interval arithmetic that normalises overall violations and the subsequent objective function is executed by means of basic lexicographic approach.
PSO – DE [46]	In an attempt to minimise premature convergence in the original PSO, differential evolution (DE) was introduced to handle constraints through the utilisation of feasibility principles to best individual solutions of particles in the region.
APSO [50]	An accelerated PSO was introduced to accelerate algorithm convergence to transverse into global best position by velocity vector.
PSOGSA [51]	PSOGSA was developed to integrate optimization intelligence of gravitational search algorithm (GSA) in local search with the robust capability PSO in social learning global best solution mathematically.
DPSO [52]	By incorporating deterministic (DPSO), a significant reduction in CPU time cost was achieved by eradicating stochasticity in PSO. In [53] and [54], the technique has been effective in determining single and multi-objective functions, respectively.
FPSOGSA [55]	A concept of integrating fractional calculus within the mathematical model of canonical PSO algorithm and coupled with traditional GSA to further improve optimization characteristics such as convergence rate.
PSO – TVAC [56]	Acceleration coefficients are varied with respect to time such that cognitive coefficient is lowered while social coefficient is accelerated as the search advances. Re-initialisation approach is enforced to particles' velocity in such a manner that static performance is prevented.
MAPSO [30]	A modified PSO with self-adaptive inertia weight, and acceleration coefficients based on feedback control parameters generated by the fitness of individual particles.
ALC – PSO [57]	Inspired by aging leaders and challengers, the swarm leader is assigned with a propagating age and a life expectancy, and particles are motivated to protest each other for leadership when swarm leader has aged. Therefore, diversity is facilitated, and algorithms' efficiency is enhanced.
PSO – DFCM [58]	In PSO-DFCM, premature convergence was prevented, and algorithms' performance significantly improved by introducing a damping factor and collaborative mechanism into particle swarm optimizer. Local exploitation and global exploration of particles are regulated by damping factor, whereas the cooperative mechanism updates particles' velocity through the utilisation of global and local best swarm learning systems.

solutions and the particles moves towards feasible regions with optimal solution.

The literature highlights PSO algorithm premature convergence and other factors contributing to unsatisfactory behaviour of the algorithm [14], [41], [42], [43], other authors [46], [47], [48], [49], [50], [51], [52], [53], [54], [55], [56], [57], [58] introduce variants to optimise the performance of PSO. However, setting particle swarm optimizer parameters to yield optimal overcurrent relay settings is a long-standing problem. The subsequent paragraph presents an overview of PSO algorithm background and its sensitivity parameters.

In PSO, individual particle initially transverse through the multidimensional search space at spontaneous velocity and its prevailing position in the i-th dimension is indicated by $s_i^{(k)}$ where k is the iteration number, and i the individual particle. Each particle learns its best position and its own experience indicated by $pbest_i^{(k)}$, and the whole swarm experience is expressed by $gbest^{(k)}$. At each iteration, the particle velocity $v_i^{(k)}$ changes with current velocity and position from the $pbest_i^{(k)}$ solution and $gbest^{(k)}$ solution. Therefore, the $v_i^{(k)}$ and $s_i^{(k)}$ updated in accordance with the subsequent equation [59]

$$v_i^{(k+1)} = v_i^{(k)} + c_1 rand_1^{(k)} \left(pbest_i^{(k)} - s_i^{(k)}\right) + c_2 rand_2^{(k)} \left(gbest^{(k)} - s_i^{(k)}\right); i = 1 \ to \ N \quad (4)$$

$$s_i^{(k+1)} = s_i^{(k)} + v_i^{(k+1)}; i = 1 \ to \ N \quad (5)$$

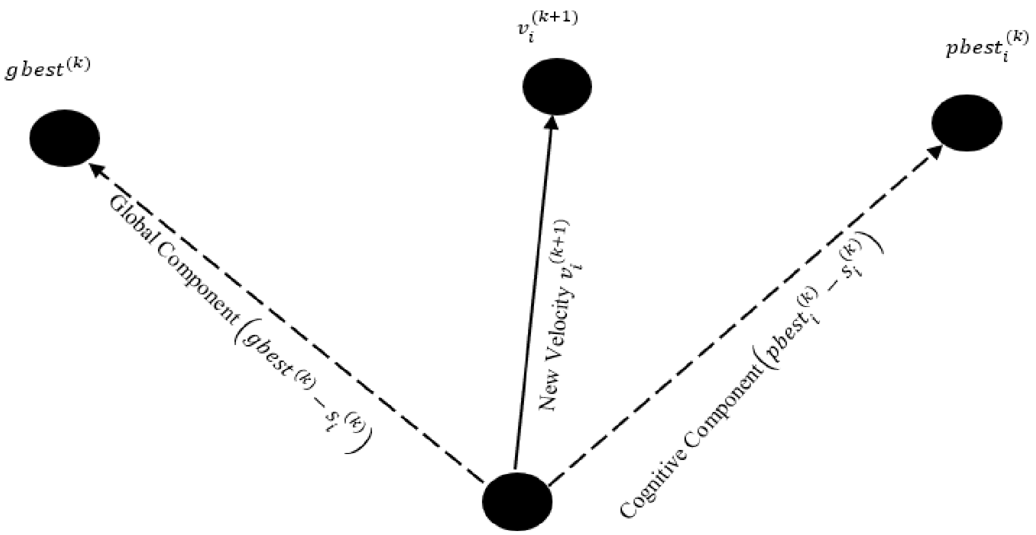

FIGURE 1. Particle velocity and position update in a two-dimensional search space [6].

where N the swarm size, $rand_1^{(k)}$ and $rand_2^{(k)}$ are numbers produced randomly every k iteration ranging between 0 and 1. Acceleration coefficients c_1 and c_2, named as cognitive and social components, separately, are positive constants [59].

Figure 1 depicts how particles' velocity, cognitive component, and global component affect particles' search capabilities for best optimal solutions. In [26], the particles' velocity update equation is grouped into three segments, the first part of the equation denotes particles' momentum which includes the influence of the past velocity on current velocity, second segments represent cognitive constant which indicates particles' pull velocity toward its individual *pbest* while the last segment demonstrates *gbest* or social collaboration between particles, as depicted in Fig. 1. Subsequent calculation of particles' new velocity and position, $pbest_i^{(k)}$ and $gbest^{(k)}$ are updated according to the following equations:

$$pbest_i^{(k)} = \begin{cases} s_i^{(k)}, & if\ f\left(s_i^{(k)}\right) < f\left(pbest_i^{(k)}\right) \\ pbest_i^{(k)}, & if\ f\left(s_i^{(k)}\right) \geq f\left(pbest_i^{(k)}\right) \end{cases} \tag{6}$$

$$gbest^{(k)} = \begin{cases} s_i^{(k)}, & if\ f\left(s_i^{(k)}\right) < f\left(gbest^{(k)}\right) \\ gbest^{(k)}, & if\ f\left(s_i^{(k)}\right) \geq f\left(gbest^{(k)}\right) \end{cases} \tag{7}$$

where f is the fitness function of PSO algorithm. Normally, the velocity of the particle is constant to the range $[-v_{max}, v_{max}]$ to reduce the likelihood of particles traversing out of feasible search area [59]. It has been observed that setting higher v_{max} causes particles to transverse past optima solution, whereas lower v_{max} values minimise particles' exploitation abilities thus, particles get trapped in local minima solution [45], [59].

1) PSO CONTROL PARAMETERS

The PSO algorithm discussed in the former section has control parameters that govern algorithms' performance and have an influence on the whole search abilities of the technique. These parameters include inertia weight, swarm size, acceleration coefficients, number of iterations, and velocity clamping-limit, are described as:

- **Inertia weight:** Due to limitations presented by v_{max}, authors in [15] suggest incorporating weight term into velocity update equation to improve particles' exploring capability by balancing the $pbest_i^{(k)}$ and $gbest^{(k)}$ search [15]. The inertia weight w is the scaling factor associated with the velocity iteration during the last time step and helps to enhance PSO algorithm converging rate. In accordance with the alteration proposed in [54], inertia weight is included into (4) as follows:

$$v_i^{(k+1)} = wv_i^{(k)} + c_1 rand_1^{(k)}\left(pbest_i^{(k)} - s_i^{(k)}\right) + c_2 rand_2^{(k)}\left(gbest^{(k)} - s_i^{(k)}\right);\ i = 1\ to\ N \tag{8}$$

A higher inertia weight value promotes exploration, whereas a lower value facilitates exploitation which maximise local search ability of PSO algorithm. An earlier study presented in [15] demonstrated significant improvement in PSO algorithm performance with inertia weight set between 0.9 to 1.2. Presently, researchers have favoured the use of linearly descending inertia weight which was original executed in [60], the w value was regulated between 0.9 to 0.4 of which yields better results. The subsequent weighting function is employed in linearly decreasing inertia weight [60]:

$$w = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} \times iter \tag{9}$$

where w_{max} the maximum inertia weight, w_{min} the minimum inertia weight, *iter* is the current iteration, and $iter_{max}$ the maximum iterations. Reference [59] conducted a comprehensive sensitivity analysis by evaluating inertia weight values between $0.8 - 1.2$, and it was discovered that bigger values facilitate global search whereas, small values encourage local search [15], [19].

- **Acceleration coefficients:** The two positive constants, c_1 and c_2 related to velocity of traversing particles into optimum regions and its individual position, these components regulate search abilities and period taken to accomplish feasible solutions by individual particles. Authors in [59] set both c_1 and c_2 to 2 and observed substantial advancement in terms of algorithms' behaviour whereas, altering c_1 and c_2 values resulted in particles moving to infeasible regions [59]. For higher c_1 and c_2 parameters, particle transverse pass feasible regions and at lower values, particles stagnate local solution due to getting trapped in the local regions before moving towards optimum solutions [59]. Consequently, c_1 and c_2 have been set to 2 since the beginning of PSO algorithm [15], [60].
- **Number of iterations:** A study conducted in [61] showed that bigger maximum number of iterations $iter_{max}$ results in longer computational effort, and it was observed that the chosen $iter_{max}$ value directly affects the likelihood of PSO algorithm succeeding to global solutions [61]. Furthermore, poor selection of $iter_{max}$ values lead to algorithm converging prematurely. Smaller $iter_{max}$ values lessen algorithms' probability of reaching global solution whereas, for larger $iter_{max}$, convergence speed improves at the expense of computing time [61], [62].
- **Swarm size:** Typically, swarm size, N, selection is based on the optimization problem application and complexity. It performs an imperative part in the behaviour of PSO and affects diversity of the population as it controls number of particles traversing toward optimum space [54], [58]. Authors in [63] specified that swarm size, N, selected between 5 and 10 particles is precise estimate; however, utilising swarm size with the range 10 and 50 particles is commonly favoured to solve optimization complex problems [63]. For larger population size, particles learn to traverse additional search space and algorithm improves performance but at computational efforts [58], [63].
- **Velocity clamping limit:** [64] explicitly stipulates that $[-v_{max}, v_{max}]$ bounds particles' velocity with $v_{max} = \delta \times (x_{max} - x_{\min})$, where x_{max} and x_{min} are design variables (minimum and maximum), and δ the clamping constant set at $[0.1 - 1.0]$. The v_{max} value regulates the smoothness of the constraining changes in velocity [65]. Higher v_{max} value facilitates global exploration; nevertheless, very high v_{max} value leads to particles flying past global optima which results in premature convergence [64]. On the contrary, smaller v_{max} value promotes local search; however, a very small v_{max} results in particles demanding more iterations to obtain global optimum solution [64], [65].

Tuning control parameters to solve optimization problems still remains an issue. The aforementioned paragraphs discuss setting control parameters based on theoretical and mathematical assumptions, which at time leads to premature convergence of the algorithm. Harrison et al. [66] utilised function analysis of variance to compute original PSO control parameters i.e., inertia weight w, cognitive c_1 and social c_2 parameters. It was seen that w provides the greatest sensitivity, thus the most crucial parameter to be tuned as it affects the probability of particles' momentum in reaching global optima. The optimization problem complexity to be solved serve as a guide to select reasonable parameter values and by testing the proposed control parameters on a common benchmark function, as done in [6]. Reference [6] conducted control parameter sensitivity analysis and introduced the unique adapting control parameters to circumvent algorithm converging prematurely. As extensively discussed, some scholars tend to utilise the published values in the literature while others prefer the use of fine-tuned static values. Nonetheless, these methods were found inefficient and ineffective due to control parameters being time dependent. Consequently, new novel self-adapting and dynamic PSO approaches are developed and currently favoured to solve modern optimization problems.

B. DEVELOPMENTS ON PSO CONTROL PARAMETERS

Numerous variances of PSO algorithm based on self-adapting strategies have been developed using various methods as the feedback parameter to keep record of algorithms' condition and make adjustments in accordance with transient states. The algorithm advancements have been categorized in the subsequent subsection.

1) INERTIA WEIGHT ADAPTATION STRATEGIES

A novel self-adaptive mutation with dynamic non-linear inertia weight was proposed in [67] to improve searching momentum by using the average particle spacing approach. The average particle spacing $S(t)$ describes dispersion degree of the swarm among individual particles as follows [67]:

$$S(t) = \frac{1}{NH} \sum_{i=1}^{N} \sqrt{\sum_{d}^{D} \left(x_{id}^{k} - x_{id}^{-k} \right)^2} \tag{10}$$

where H is the diagonal maximum length of the search area, N and D are swarm size and space dimension, separately. x_{id}^{k} and x_{id}^{-k} are coordinate average value of the i-th particle in k-th iteration, and it was reported that smaller average spacing results in more concentrated swarm and poorer species diversity [67]. For altering the nonlinear inertia weight $w(t)$ as swarm diversity transverse, a new nonlinear dynamic approach based on average particle spacing was adopted as defined:

$$w(t) = 1/\left(1 + e^{-10(S(t)-0.5)}\right) \tag{11}$$

TABLE 3. Impact of different acceleration coefficient values on PSO performance.

Acceleration coefficient value	Impact on PSO performance
$c_1 = c_2 = 0$ [69]	Particles transverse with constant speed towards the search space as its velocity is independent from the effect of $pbest$ and $gbest$. The position update is entirely dependent on the inertia weight and swarm is attracted to the $gbest$ position.
$c_1 > 0$ and $c_2 = 0$ [70]	Only the cognitive component influences particles' velocity such that the particle travels toward global search space based on individual experience.
$c_1 = 0$ and $c_2 > 0$ [70]	The cognitive component does not influence particle momentum, whereas social component directly facilitates exploitative behaviour and displays quicker convergence as particles rely on the experience of the greatest particle in the whole swarm.
$c_1 = c_2 > 0$ [69]	Particles travels towards median of $pbest$ and $gbest$ solution.
$c_1 > c_2$ [69]	Cognitive acceleration that facilitates exploration behaviour.
$c_1 < c_2$ [69]	Social acceleration promotes exploitation search.

Another effective group fitness variance strategy employed in [68] was also utilised in [67] to accomplish a self-adaptive adjustment of global and local search ability.

Nagra et al. [71] presented outstanding advancement in the performance of PSO algorithm when the self-adaptive inertia weight with gradient-based local search was utilised to balance exploitation and exploration capabilities [71]. A ranked-based inertia weight system was proposed in [72] and [73] to improve converging speed by means allocating ranks to particle based on its fitness value. However, the strategy yields more efficient results with the increasing iterations which may increment computational time. Reference [73] proposed cosine inertia weight approach in the form of variable-period cosine function ranging between $[0, \pi]$ to meet the multi-stage inertia weight requirements. At the range $[0, \pi/6]$, the cosine function preserves a larger value $\left(\leq \sqrt{3}/2\right)$ and traverse slowly whereas the cosine function declines rapidly at $[\pi/6, 5\pi/6]$. In the range $[5\pi/6, \pi]$, the cosine function keeps a smaller value $\left(\leq -\sqrt{3}/2\right)$ and changes slowly. Due to inconsistence of original cosine function, an iterative term $I(t)$ was introduced into the cosine function to modify the period and $w_{cos}(t)$ was rescaled in the range $[w_{fin}, w_{ini}]$ and denoted by the following equation [73]:

$$w_{cos}(t) = \frac{(w_{ini} + w_{fin})}{2} + \frac{(w_{ini} - w_{fin})}{2} \times cos\left(\frac{I(t)\pi}{t_{max}}\right) \tag{12}$$

$$I(t+1) = I(t) + a, I(1) = 0 \tag{13}$$

$$a = \begin{cases} a_1, & \frac{I(t) \leq t_{max}}{6} \\ a_2, & \frac{t_{max}}{6} < I(t) \leq \frac{5t_{max}}{6} \\ a_3, & \frac{t_{max}}{6} < I(t) \leq t_{max} \end{cases} \tag{14}$$

where a the constant for adjusting $w_{cos}(t)$, the values of a_1, a_2, and a_3 control the length of each stage in $w_{cos}(t)$. The requirements of cosine inertia weight $w_{cos}(t)$ limits the phase $(I(t)\pi/t_{max})$ in the range $[0, \pi]$ and $(I(t)\pi/t_{max})$ increases from 0 to π while t moves from 0 to t_{max}. Consequently, $\left(\frac{1}{6a_1} + \frac{2}{3a_2} + \frac{1}{6a_3} = 1\right)$ must be satisfied [73]. It should be noted that cosine inertia weight approach demonstrated significant improvement, but more experiments need to be performed on practical engineering fields.

2) ACCELERATION COEFFICIENT

Since the beginning of PSO algorithm, the cognitive and social parameters have been set positive and constant [14], [35]. As depicted in Table 3, researchers observed the impact of various acceleration coefficient values on algorithms' behaviour by changing either cognitive or social component and in some instances, both parameters were varied. Shirazi et al. [69] experimented acceleration coefficient by means of dynamic change such that when c_1 increases, c_2 decreases and vice versa. Similarly, Maharana et al. [70] tested $c_1 > 0$ while setting c_2 to zero and conversely. However, both approaches are time-dependent only and have proven to be inefficient at times. An approach that considers particles' evolutionary state was proposed in [16], it allows evolutionary state to command acceleration coefficient into converging towards the best optimum solution in the hyperdimensional space. Particles are forced into performing in a predefined manner to enhance searching abilities of the swarm by the acceleration coefficient equations that follows [6]:

$$c_1^k = \begin{cases} \frac{(c_{max} - c_{min}) \times (iter_{max} - k)}{iter_{max}} \\ +c_{min} \quad if\ 0 \leq ES_i^k \geq 0.5 \\ c_{max} - \frac{(c_{max} - c_{min}) \times (iter_{max} - k)}{iter_{max}} \quad if \\ 0.5 \leq ES_i^k \geq 1.0 \end{cases} \tag{15}$$

$$c_2^k = \begin{cases} c_{max} - \frac{(c_{max} - c_{min}) \times (iter_{max} - k)}{iter_{max}} \\ if\ 0 \leq ES_i^k \geq 0.5 \\ \frac{(c_{max} - c_{min}) \times (iter_{max} - k)}{iter_{max}} \\ +c_{min} \quad if\ 0.5 \leq ES_i^k \geq 1.0 \end{cases} \tag{16}$$

$0 \leq ES_i^k \geq 0.5$ refers to low evolutionary state and global search is promoted by permitting more global search at the beginning and towards the end, local search is facilitated. Larger evolutionary state value $0.5 \leq ES_i^k \geq 1.0$ promotes exploitive search in early iteration stages, allowing particles to traverse towards global solution and developing gradually, further exploration is facilitated at later stages of the iterations with two constants c_{min} and c_{max} set at 0 and 4.0, respectively [6]. The self-adapting acceleration coefficient was validated on benchmark engineering constrained problems which proven to be effective, it managed to promote convergence speed and overcome premature convergence [27].

C. GENETIC ALGORITHM

Contrary to PSO, Genetic Algorithm (GA) search functions' solution space by adopting survival of the fittest strategy, whereas PSO algorithm inspiration emerges from social behaviour of animal and birds [3], [18]. In GA, mutated chromosomes improve algorithms' solution by maintaining a certain probability of population diversity with a great percentage of genetic parameter selection to support new population [30]. Roulette wheel selection approach assigns selection likelihood to individual chromosomes based on their fitness function values [74]. The numbers are generated randomly to correlates cumulative probability for computing new population selection [15], [74]. This approach presented drawbacks such as prematurely converging to local optima due to the supremacy of individual chromosomes that steadily succeed in the competitiveness and are selected as parents. The probability $P_i(t+1)$ for individual chromosome i is stated in the equation below, where $f_i(t)$ the chromosome i fitness function, and n denotes population size [74].

$$P_i(t+1) = \frac{f_i(t)}{\frac{1}{n}\sum_{j=0}^{n} f_i(t)} \tag{17}$$

Due to limitations encountered on roulette wheel approach, variations such as ranking method, scaling technique, and tournament selection were developed to permit negativity and minimization on genetic algorithms [74], [75]. In the ranking-based selection method, the probability of individual chromosome P_i is allocated based on the success of individual solution i when all the solutions are mapped based on their fitness values to permit minimisation. Chromosomes constituting of larger fitness values have a greater likelihood of succeeding to the subsequent generation. The randomly generated number in a range [0, 1] contributes to reproduction of a next population n_{keep} of optimal solutions. Probability of individual chromosome P_i can be determined as follows [74], [75].

$$P_i = \frac{n_{keep} - i + 1}{\sum_{i=1}^{n_{keep}} i} \tag{18}$$

Similarly, to PSO algorithm, genetic algorithm consists of control parameters that require tuning such that premature convergence is avoided, and algorithms' performance is maximised. Eberhart et al. [76] noticed a property that affected GA algorithm performance that was crossover or recombination. Michalewics [77], on the hand, studied genetic operators and found that mutation have significant impact on the algorithm convergence. Population size also plays a role in behaviour of the algorithm and computational effort [75].

- **Crossover:** Recombination draws two chromosomes from the reproduced population sets and employs crossover. Ahmed [78] proposed sequential constructive crossover operator that builds offspring from selective parents based on parents' structure qualities using better edges approach. However, the technique presented the drawback of algorithm not utilizing any local search method for enhancing quality due to the small population size [78]. A cycle crossover that counts bits circularly from parents and their current position was proposed in [79], but the produced offspring had identical characteristics as parents. Simple or single-point crossover produces a random number r from a recurring allocation and generates two new individuals (x_i' and y_i') corresponding to the equations that follows [76].

$$x_i' = \begin{cases} x_i & if\ i < r \\ y_i & otherwise \end{cases} \tag{19}$$

$$y_i' = \begin{cases} y_i & if\ i < r \\ x_i & otherwise \end{cases} \tag{20}$$

  Recombination presents a new neighbourhood for supplemental implementation within the hyperdimension, which are indicated by either parent assembly [74], [75]. Consequently, the probability of attaining better performing offspring is substantially risen. In [62], it was seen that higher crossover rate leads to introduction of new population quickly, whereas very high recombination likelihood results to structure discarding rapidly before selection produces improvements [62]. Smaller recombination probability leads to stagnating search due to insufficient exploration [62].
- **Mutation:** Establishes heterogeneousness into the population by extending the search space for genetic algorithm to assess and mitigate faster convergence prior to whole search area exploration [74], [75]. An incremental in mutation probability leads to population discovering beyond current search region of variable area which may results in impairment of population by changing surviving feasible solutions. Therefore, smaller mutation probability advocated [77]. Uniform mutation chooses one variable j randomly and equate it into a constant number $U(a_i, b_i)$ where a_i and b_i are lower and upper bound, separately [77].

$$x_i' = \begin{cases} U(a_i, b_i) & if\ i < j \\ x_i & otherwise \end{cases} \tag{21}$$

- **Population size:** As stated in [75], the group of chromosomes known as population affects GA algorithm behaviour. It was explicitly asserted that small population size results in algorithm performing poorly due to inadequate trail size for exploring hyperplane [75]. Contrarily, larger population size prevents algorithm from converging prematurely by permitting more particles to occupy search space but at computational expense [75], [76]. According to [81], any values between 10-50 is a precise choice however, in other study [80], any value between 25-250 produces efficient solutions.

1) DEVELOPMENTS ON GA CONTROL PARAMETERS

In 2019, Hassanat et al. [9] undertook a comprehensive study to review GA parameter selection and proposed the change of crossover and mutation rate dynamically. The study adopted a deterministic method to decrease crossover probability from 100% to 0% and increase mutation probability linearly from 0% to 100%, and vice versa. However, the approach lacks diversity, operates on smaller population size only, and requires larger number of mutations. On the other hand, Akter et al. [82] suggested a new crossover operator made up of two crossover points selection and new offspring reproduction by comparing cost between two parents; the approach needs to be validated through benchmark problems and compared with other adaptive as well as self-adaptive methods. A further study in [83] detailed improvements in genetic algorithms and proposed adaptive GA by modifying important genetic operators, that is, crossover and mutation probabilities. The adaptive crossover probability continuously adjusts the probability with respect to fitness function value of individuals in the population, it adapts such that for individuals made up of smallest and largest fitness functions, the crossover operates with specific probability to accommodate changes. The crossover probability P_c is adjusted according to the following equation:

$$P_c = \begin{cases} k_1 \dfrac{f_{max} - f_c}{f_{max} - f_{min}}, & if\ f_c \neq f_{max}, f_{min} \\ k_2, & if\ f_c = f_{min} \\ k_3, & if\ f_c = f_{max} \end{cases} \tag{22}$$

where P_c the crossover probability, f_c consists of higher fitness in the first two parents of crossover operation, f_{max} and f_{min} are maximum and minimum fitness, respectively. k_1, k_2, k_3 are constants ranging between 0 - 1 and $k_2 > k_3$.

In another study [84], it was empirically depicted that by generating mutation probability based on chromosome rank in the population, quicker convergence was obtained. Besides population size, mutation probability plays a significant role in general algorithms' performance.

The motivation for implementing a rank-based adaptive mutation was to overcome insufficient genetic information in the initial population and loss of such information during optimization process. It assigns the fittest chromosome a rank N in a population of N individuals which ranges between $[1, N]$ depending on their fitness function values. Mutation probability P_m adapts the rank of chromosomes r as follows:

$$P_m = p_{max}\left(1 - \frac{r-1}{N-1}\right) \tag{23}$$

The best chromosome has zero mutation probability and the poorer consists of the maximum probability p_{max}, meaning P_m distributes linearly between 0 and p_{max}. Nevertheless, if one or more chromosomes obtain identical fitness, ranks are allocated randomly to them, and the mutation probability remains unaffected by the asymmetry of fitness distribution [84]. By applying multi-population with self-adaptive mutation scheme [85], GA algorithm overcomes premature convergence when coupled with overlapping and subpopulation convergence. Overlapping evaluated between two subpopulations which are determined by the comparison of finest solutions distance of the subpopulation, and if the search radius of the best individuals of subpopulation is less than the threshold value, then individuals associated with subpopulation are removed. Subsequent overlapping search, the overall subpopulations experience convergence process to evaluate converging robustness.

Another adaptive mechanism to set mutation probability dynamically is proposed in [86], it controls the utilisation of population entropy toward the end of epoch (consecutive generations number) by computing the variation of the current entropy from the preceding $k+1$ epoch H_{k-1} denoted as $\Delta H_k = H_k - H_{k-1}$. A comparison between the change $\Delta \mathrm{H_k}$ and the one evaluated in the previous epoch $\Delta H_{k-1} = H_{k-1} - H_{k-2}$ is performed in such a way that when the change in entropic decreases at least by factor ε, loss of diversity is signalized which then triggers the mutation probability by means of including a constant factor (α). Otherwise, the mutation probability P_m is reduced by the subtraction of the constant factor α value, P_m value prevents overshooting by keeping the value on the interval of $[P_{mLB}, P_{mUP}]$; where the lower bound is set at 0.001 and the upper bound set at 0.1 [86]. In [87], mutation probability was modified based on stochastic Manhattan learning algorithm, whereas in [88] P_m was changed by means of fitness frequency distribution. Lastly, [89] utilised the entropy value for modification of mutation operator instead mutation probability values, as reviewed in the aforementioned subsection.

V. SELECTIVITY AND SPEED ANALYSIS

In an attempt to discover control parameters that influence overcurrent relay selectivity and operational speed, a sensitivity analysis of GA and PSO algorithms was performed in [30]. With a swarm size ranging between 10 to 500 particles, inertia weight set at 0.9, acceleration coefficient set at 2 and maximum velocity set at 50, it was observed that an increase in swarm size caused PSO algorithm performance to be more efficient but at computational time expense [30]. Authors in [59] utilised swarm size set between 20 and 160 particle and noticed that swarm size have minimal effects on the performance of particle swarm optimizer. Another study in [63] proposed that choosing swarm size must be done based on variables number. Nevertheless, the response of overcurrent relay with respect to smaller swarm size was unsatisfactory, the relays took long to operates which resulted in violation of protection principle to discriminate faults speedily. A comparative study conducted in [90] showed that GA converges quicker than PSO algorithm meaning the relay was more selective and operated speedily when configured with GA algorithm; however, the contributary factors to PSO algorithm poor convergence were not clearly indicated, as [30] suspected inertia weight. It was seen that larger inertia weight 0.8 – 1.2 was unsuccessful in optimizing protection coordination hence, selectivity and operational speed were neglected [30] which opposes the study conducted in [59] that found inertia weight in this range facilitates global search. Poor selectivity and speed are evident that PSO converged prematurely due to larger inertia weight and failed to fulfill its goal of reducing iteration number and improving particles' exploration and exploitation abilities. Furthermore, a decreasing inertia weight 0.9 – 0.4 permits particles to transition from exploration mode to exploitative mode to produce global solutions, as claimed by [91] and [92].

Parveen [93] proposed a new hybrid optimization technique particle swarm optimization gravitational search algorithm to coordinate relays, both GA and PSO algorithms were utilised for comparison purposes which showed the latter yields infeasible results. The hybrid algorithm showed its superiority by obtaining more sensitive time multiplier setting values which signifies speed relay operation and protection selectivity [93]. However, control parameters impact in algorithms' performance and the relay selectivity as well as operational speed was not considered [93]. A detailed sensitivity analysis is presented in [6] considers the effect of velocity clamping-limit on the PSO algorithm behaviour, it studied distinct clamping constants [0.0, 1.0], taking into consideration local search enhancer, velocity clamping-limit decreasing technique, active penalty scheme, and reinitialization methods. The attained results revealed that the clamping-limits have least effect on the algorithms' behaviour which might be due to the problems' nature as the velocity is normally set to range dynamically [6]. Contrary to [6], Barrera et al. [94] experimented different polynomial functions that lessen maximum velocity and developed a parabola function to positions particles near a localized search was proven successfully. The number of iteration selection is problem-nature dependent, and complexity dependent as larger values increase computational time, while smaller value reduces the probability of obtaining global solution [30]. It was reported that protection relay remained selective throughout the variation of iterations and the operational speed was minimised [30], the overall overcurrent relay response showed that larger iterations fail to enhance PSO performance due to algorithms' ability to regulate search period and not particles movement in search area [30].

The genetic operators, i.e., crossover and mutation probability displayed significant impact on the performance of GA algorithm [30], the fitness value increased proportionately to crossover and mutation probability. The findings demonstrated that an increase in crossover and mutation probability resulted in overcurrent relays response time was increased and the CTI was exceeded on other relays. With crossover probability set at 30%, the yield time multiplier setting was 2.30 seconds which was optimal value and selectivity was maximised [30]. A crossover probability of 80% obtained optimum time multiplier value of 4.98 seconds which indicated slower response of overcurrent relay. In a subsequent study on the linearly increasing crossover probabilities [0.3, 0.9], Bikirli and Kut [80] reported that a higher crossover value (0.9) leads to dominant individuals with best fitness function values getting lost in the hyperplane space. Further experiments conducted in [30] utilised a uniform single-point crossover value of 0.3 while varying mutation probability from 0.02 – 0.3 to observe the behaviour of GA algorithm with respect to overcurrent selectivity and speed of operation. A behaviour similarly to increasing the crossover probability was observed, increasing mutation rate improved algorithm fitness and aids to avoid premature convergence by generating distinct chromosomes. It introduced diversity and preservation which are predominately the primary purpose of mutation, and the relays were more selective and operated promptly when required. Another genetic operator that influences the behaviour GA algorithm is population size as alluded in aforementioned section. A sensitivity analysis performed in [30] set population size at 10 – 500 particles, it was proven through plots that larger population size results in GA algorithm performing robustly and efficient at computational efforts expense, which agrees with [80]. Overcurrent relays operated speedily when population size was set at 500 and when the population was set at 10, the relays took longer to operate with CTI higher than the stipulated value which violated one of protection philosophies to discriminate fault promptly [30].

Both PSO and GA algorithms parameters were analysed and reviewed in terms of their selection influence on the behaviour of overcurrent relay selectivity and speed. From analysis of PSO algorithm, it can be seen that parameters such as swarm size, inertia weight, acceleration coefficient, and number of iterations influences the performance of the algorithm. Therefore, selection of the operating parameters plays an essential role in the response of overcurrent relays. Sensitivity analysis of GA algorithm revealed that crossover rate, mutation probability and population size possess a direct influence on the behaviour of genetic algorithms. Slower convergence and higher fitness function was experienced when larger crossover, mutation, and population size were chosen whereas smaller values yielded faster convergence and optimised fitness function.

Selectivity and speed analysis can be substantiated by the performance comparison of GA with PSO algorithm. A comparative study conducted in [95] highlighted that GA algorithm presents slower convergence behaviour than PSO algorithm. Consequently, the authors introduced a hybrid optimization algorithm by combining PSO and GA by studying the results of one algorithm as an input for the other. An observation of interest in [30] showed that PSO convergence speed was slightly faster than GA; however, GA algorithm fitness curves were smoother due to fewer alterations and efficient exploitation at the beginning of the search.

The experimental work [96] proved that PSO yields global best solution with at least 100 particles which agrees with the sensitivity study in [30]. Furthermore, Beielstein et al. [19] investigated the behaviour of PSO with GA algorithm and found that PSO yields best fitness in fewer iterations and overall performed efficient and effective.

Although PSO algorithm converging prematurely and reliance on the preliminary control parameter settings problems were addressed in earlier sections, the algorithm performance can be further improved through hybridization with other algorithms. Eberhart and Kennedy [16], [101] conducted the first experiment to train Artificial Neural Network (ANN) using PSO algorithm, which successfully feedforward multilayer perceptron ANN to arrange Fisher's Iris dataset. Authors in [102] utilised an ANN with 60 input, 12 hidden nodes, and 2 input nodes to train a feedforward multilayer perceptron, which resulted in successful application of PSO to train ANN with minimal error and maximised performance. ANNs are used to solve protection problems in [103] and a nonlinear signal transformation-based ANN was suggested to optimize differential protection scheme. A significant percentage error was encountered in [104] when the ANN was integrated with the discrete wavelet transform algorithm, the error was a result of the quantity of transient data utilised to train ANN algorithm.

Baran et al. [105] solved overcurrent coordination problem through ANNs for smart distribution systems, the work was further modified in [106] to maximise the effectiveness of the artificial intelligence protection strategy. In [107], Opposition Based Learning (OBL) is hybridized to enhance cuckoo optimization techniques known as CH-EOBCCS, the algorithm updates the population by means of OBL strategy with chaotic type of cuckoo search technique. However, a further hybridization [108] OBL with GSA (OBL-GSA) outperformed traditional methods and successfully solved overcurrent relay problem. A modified form of class topper optimization with the concept of OBL and Fractional Order (FO) was introduced in [28] to enhance exploitation and exploration capabilities, which demonstrated it superiority through optimum selectivity and optimised operational speed of overcurrent relays. Nevertheless, an incremental in computational effort and complexity was observed, another observation of interest was the failure of the algorithm to coordinate small distribution network [28]. Some of the favoured algorithms are tabulated in Table 4 with their merits/ demerits.

A comprehensive review and analysis of optimization techniques with respect to relay selectivity and speed revealed the following:

- The tuning of GA genetic operators, i.e., population size, crossover rate, and mutation probability have the substantial effect on the algorithms' performance and overall response of overcurrent relays. Parametric sensitivity revealed that varying one parameter while keeping the other constant helps to identify operators responsible for the model failure.
- Similarly, PSO algorithm sensitivity analysis revealed that inertia weight and swarm size have greater effect on the algorithms' performance, whereas number of iterations presents an insignificant impact on convergence speed and fitness function.
- A comparative review between PSO and GA algorithm showed that PSO managed to perform efficiently and effectively, the maximised selectivity and optimised speed were attained. Although optimal PSO performance was obtained, the algorithm is sensitive and depends on initial settings.

VI. ARTIFICIAL INTELLIGENCE MODELS FOR OPTIMIZING OVERCURRENT RELAY

In this section, the literature of preceding case studies are analysed for appropriate selection of artificial intelligence models based on data accessibility and the possibility of solving optimization problem.

A. ARTIFICIAL NEURAL NETWORK

In ANNs, transfer functions determine the connection between the input and output nodes in a system by incorporating a non-linearity level and exploiting ANNs merits of insensitivity to noise data, which leads to excellent capability for generalization [109]. Reference [20] trained ANN by using multi-layer feedforward back propagation neural network to decrease the predictable percentage error between virtual and target parameters through searching for the greatest compilation of relative weights [20]. The mechanism tends to be extensively favoured as it presents effective function and has the capability of learning in the user absence. However, in certain circumstances, it results in increased computational time due to an intrinsic failure to comprehend the output produced by ANN algorithm [20], [109]. Chui et al. [110] reduced this optimization ineffectiveness by introducing radial basis function which is a training strategy designed to predict time series by adaptive studying algorithm and response surface methodology [110]. Nevertheless, radial basis function network involves determining the optimal core parameters and kernel function width [110].

Backpropagation learning trains the weight of ANNs based on error probability attained in previous iterations, but to accomplish faster and efficient fine-tuning, algorithms of the second order need to be exploited for instructions [111]. Consequently, Levenberg Marquardt algorithm aids in tuning of backpropagation neural network of smaller and medium sized configurations as it improves convergence rate [111]. In another work [112], Levenberg Marquardt algorithm

TABLE 4. Selected state-of-the-art algorithm for the comparative study.

Algorithm	Description
Artificial Bee Colony (ABC) [97]	ABC adopts the concept of self-organization and labour division inspired by behaviour of a colony of bees. The constraints are handled through employed bees, onlooker bees, and scouts.
Class Topper Optimization (CTP) [98]	CTP imitates from student learning behaviour to improve performance. The algorithms' premature convergence was circumvented by introducing chaotic search class topper optimization which uses a logic map in the initial stage and varying sigmoid functions towards updated stage to maintain diversity in the search space.
Adaptive Fuzzy Directional Bat Algorithm (AFDBA) [99]	Calculates optimum overcurrent relay settings in any distribution network topology without requiring initial tuning of parameters. It relies on the evaluation of bat algorithm and fuzzy inference system to determine essential parameters on a real-time basis. Due to AFDBA dependence on fuzzy inference system, a higher computational time is incurred which leads to a slower relay response.
Cascade Forward Neural Network (CFNN) [100]	CFNN modelling includes cascade neural networks by associating neurons number and learning rates in the testing as well as learning processes. Its merit is the ability to maximise the weight at iteration phase to accomplish appropriate error value based on the set goal. Nonetheless, high infiltration of distributed generation affects the sampling rate reliability.
ANN Levenbergs Marquardt Algorithm [99]	Ref. [99] explored the ability of ANN to predict overcurrent relay miscoordination time interval between main and backup relay operating time by assigning the output of the ANN to curve fitting technique. Quicker relay operating time and less miscoordination time were attained which proves the robustness of the algorithm.

was utilised to teach multilayer perceptron and optimized structure was obtained which agrees with work in [113]. Karupiah et al. [114], further adopted the utilization of Levenberg Marquardt ANNs for relay coordination problem and recommended a new efficient relay operational speed with capabilities of forecasting the likelihood of protection miscoordination in the distribution network; however, the proposed solution was not validated experimentally. A typical architecture of ANN, as can be seen in Fig. 2, comprises input neurons ($x_1 \ldots . x_n$) with given respective weights ($W_1 \ldots . W_n$) and the addition of bias (b) to all inputs [109]. Activation function (F) determines the association between the weighted input nodes and neural network output by incorporating non-linearity level required in majority of ANN applications [109].

A further study in [21] utilised ANN to identify faulty power lines and location of the fault from the source, thereafter, a backtracking method was adopted to provide coordination between primary and secondary relay such that selectivity is maximised [21]. The findings were associated with other studies of ANNs studied in [22] and [23] which revealed it performs better with the hybrid algorithm. Reference [115] hybridised feedforward neural network with support vector machine to detect faults and estimate fault location, respectively. Improved findings were demonstrated, and the protection settings managed to adapt to system changes, unlike the traditional ANN that stagnates to changes. A study [116] attempting to estimate protection miscoordination time of relay operations utilised ANN, which managed to obtain minimal solution for medium size of radial system; however, miscoordination was produced.

B. ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM

To achieve optimal protection settings performance, [24] developed adaptive neural-fuzzy inference system (ANFIS) structure generating adaptive I_p and TMS parameters. It produces suitable input/output mapping and membership functions (MFs) based on fuzzy *If-Then* rules to provide input/output pairs [24], as can be seen in Fig. 3. The ANFIS technique was implemented on the normal inverse overcurrent relay characteristics, and further experiments are required to prove algorithms' effectiveness as well as its superiority in comparison with other competitive algorithms [24]. In ANFIS, every node i is equivalent with node function $O_i^1 = \mu A_i(x)$, where x the input to node i and A_i denotes the linguistic label related to node function. $\mu A_i(x)$ maintains interval $0 \leq \mu A_i(x) \leq 1$ irrespective of bell function (24) or Gaussian function (25).

$$\mu A_i(x) = \frac{1}{1 + \left[\left(\frac{x-c_i}{a_i}\right)^2\right] b_i} \tag{24}$$

$$\mu A_i(x) = exp\left[-\left(\frac{x-c_i}{a_i}\right)^2\right] \tag{25}$$

where a_i, b_i, and c_i are parameter sets that adjust accordingly with varying bell-shaped function thus displaying different forms of MFs on A_i label. A circle node $w_i = \mu A_i(x)\,\mu B_i(y)$ signifies the firing strength of a particular rule and square node i with a function $O_i^4 = \bar{w}_i(p_i x + q_i y + r_i)$ where w_i the output of previous layer and p_i, q_i, and r_i are consequent parameter set. An improvement in performance was achieved

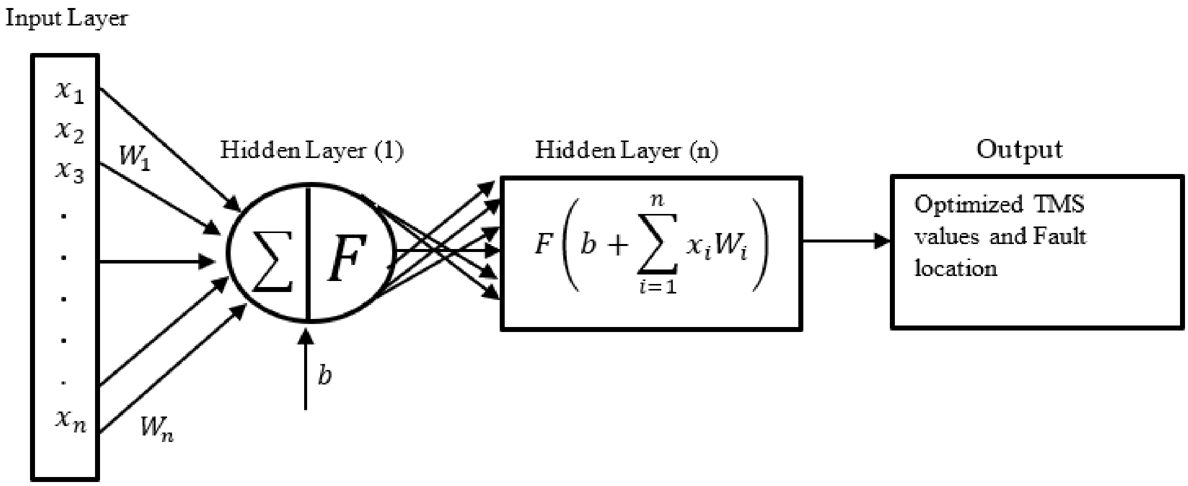

FIGURE 2. Typical ANN architecture for determining optimised TMS and Fault location.

when ANFIS model used seven numbers of Gaussian-type MFs and more precise findings were obtained using five numbers of triangular-type MFs [24]. It was observed that ANFIS relays yield more precise results under varying load currents and TMS values. Nevertheless, the use of least-squares estimation in training leads to complex information extraction in ANFIS relays in comparison to simple and straightforward fuzzy logic overcurrent relays.

In [117], an ANFIS architecture was included into the development of protective relay model and the structure of individual relay was associated with the system state. Consequently, the protective relay demonstrated better performance due to less complicated selection of network and the amount of topological state was extremely slightest. Nevertheless, the approach utilised a single inverse relay characteristics and other characteristic curves were not taken into consideration [117]. Further study [23] suggested fine tuning of membership-functions due to highly non-linear mapping and self-adaptive nature of tuning MFs of ANFIS. The following was drawn from different literature reviewed:

The model architecture, quantity of *If-Then* rules, data processing speed, and the anticipated output are dependent on the quantity of inputs for the model.

Membership function plays an essential role in model performance; however, the absence of a standard technique for selection of suitable membership function results in stagnant performance.

Adaptive neuro-fuzzy inference systems managed to outperform the artificial neural network due to its advantageous utilization of hybrid model between neural network and fuzzy logic [24].

VII. RECOMMENDATIONS FOR FUTURE STUDIES

Optimising overcurrent relay selectivity and speed by means of optimization techniques has demonstrated greater progress in recent research and implementation; however, the challenge of algorithms' converging prematurely still remains. The poor parametric selection and absence of models to serve as a guide for choosing critical parameters contribute significantly to unsatisfactory performance. Some of the recommendations for future studies to tackle these challenges are as follows:

- The problems of protection coordination in modern distribution networks are tackled adequately by multiple meta-heuristic techniques, particularly evolutionary algorithms with major improvements such as adaptive inertia weight, crossover, mutation, and variation acceleration coefficients. Nonetheless, it is worth determining a standard range for coefficients such that divergence risks are eliminated. Furthermore, the hybridization of metaheuristic methods with other algorithms improves selectivity and speed but the likelihood of premature convergence remnants which leads to stagnant behaviour of the algorithm. To mitigate stagnating performance, the selection of control parameters must be standardised based on problem complexity and a comprehensive sensitivity analysis needs to be conducted.
- In artificial intelligence models, the selection of input variables plays an essential role in determining the models' performance and precision of outcomes. Consequently, it is commended that more research interest into discovering optimum input variable combination. Similarly, the filtration of data during pre-processing phase needs to be reliable and more accurate for subsequent utilisation as input.
- Another observation of interest, artificial intelligence model parameters, that is, number of hidden layers, membership functions, epsilon in support vector machine, and number of fuzzy rules affects the models' performance and the obtained results. However, these parameters lack optimal values and are varied in different case studies which have drawbacks such as longer computation time. It is advisable to perform a sensitivity study for parametric tuning using respective

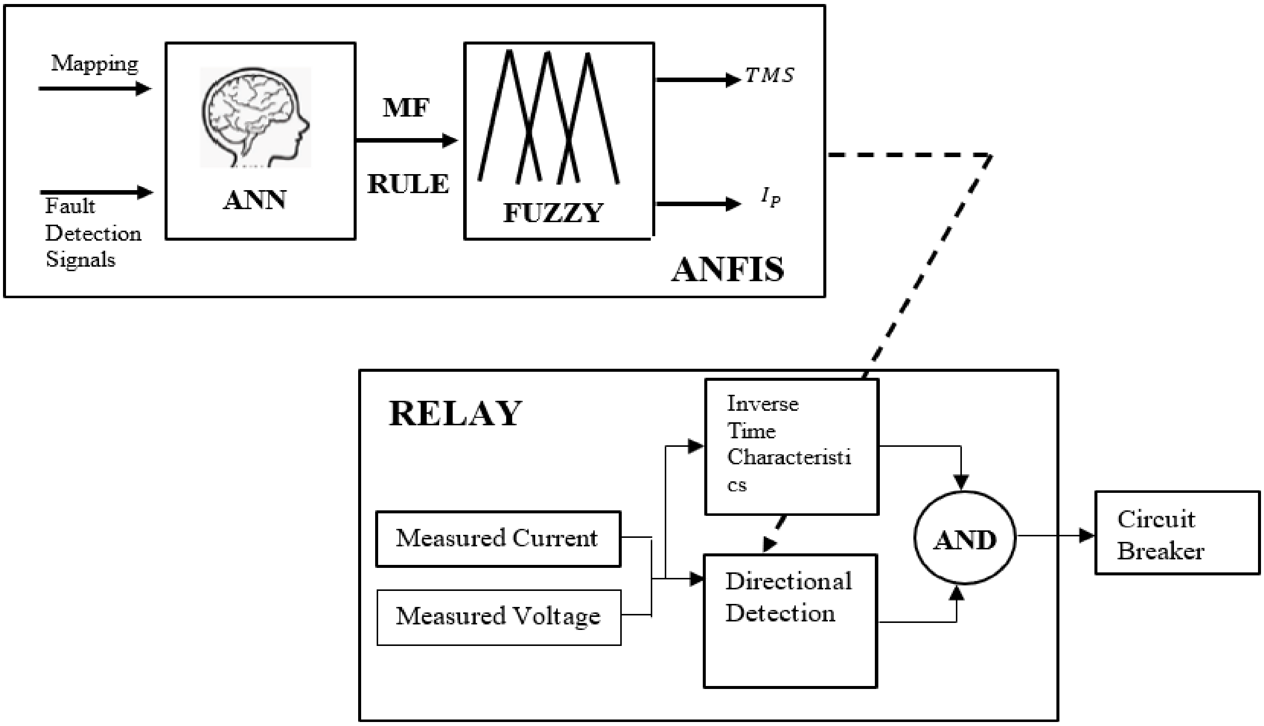

FIGURE 3. Flow diagram depicting adaptive relaying using ANFIS.

algorithms and eliminate the utilisation of a trial-and-error approach.

- Majority of literature reviewed [24], [114], [115] and etc, adopt a maximum of two artificial intelligence models and performs a comparative study between two models which fails to prove models' robustness and effectiveness in optimizing overcurrent relay coordination. Therefore, it is advocated to employ more than two various artificial intelligence models in a case study for more accurate results.
- Future studies to entail the effects of renewable energy sources on overcurrent relay coordination problems. There is a lack of research papers focusing on renewable energy association with overcurrent relay selectivity and sensitivity studies.

VIII. CONCLUSION

The objective of this paper was to review optimization techniques that maximize overcurrent relay selectivity and operational speed. Due to the disadvantages of traditional methods, evolutionary and artificial intelligence techniques are favoured to solve overcurrent relay coordination problems in distribution networks. However, literature revealed that tuning control parameters to yield optimum overcurrent relay settings is a long-standing problem. The comparative study performed in this paper found that control parameters definitely influence algorithms' behaviour which ultimately affects overcurrent relay discrimination time. The varying of one parameter at a time while keeping others constant was useful in identifying parameters contributing to poor protection settings.

A theoretical review conducted presented numerous variations of PSO algorithm based on self-adapting strategies by means of various methods as the feedback parameter to keep track of algorithms' condition and make adjustments in accordance with transient states. Inertia weight adaptation strategies was presented to accomplish self-adaptive tuning of global search capabilities, but the approach lacks practical experiments to prove its effectiveness. Similarly, acceleration coefficient was experimented by means of dynamic change which is time-dependent and proven to be inefficient at times.

There are developments in terms of tuning GA control parameters, adaptive crossover probability continuously adjusted likelihoods with respect to fitness function value of individuals in the population. Insufficient genetic information in the initial population and loss of such information during optimization process was mitigated by implementing a rank-based adaptive mutation. The comparison of PSO and GA algorithm depicts that genetic algorithm converges slower than particle swarm optimizer. This means particle swarm optimizer maximised overcurrent relay settings and achieved optimum protection coordination in distribution networks. Although optimal PSO performance was obtained, the algorithm is sensitive and depends on initial settings.

Artificial intelligence models have emerged as faster, most accurate, and more efficient solutions for protection coordination problems. Adaptive neuro-fuzzy inference systems managed to outperform the artificial neural network due to its advantageous utilization of hybrid model between neural network and fuzzy logic. Nonetheless, ANN and ANFIS performed efficiently in comparison to both evolutionary algorithms, that is, particle swarm optimization and genetic algorithms. Challenges currently faced in the research domain and recommendations were detailed with a main focus on optimization techniques for maximising overcurrent relay selectivity and speed.

REFERENCES

[1] N. I. Nkhasi, "Real-time modelling and simulation of distribution system protection with and without renewable distributed generation," M.S. thesis, Dept. Elect., Electron., Comput. Eng., Univ. KwaZulu-Natal, Durban, South Africa, 2017.

[2] M. H. Hussain, S. R. A. Rahim, and I. Musirin, "Optimal overcurrent relay coordination: A review," *Proc. Eng.*, vol. 53, pp. 332–336, Jan. 2013.

[3] A. R. Haron, A. Mahamed, and H. Shareef, "Coordination of overcurrent, directional and differential relays for the protection of microgrid system," in *Proc. Int. Conf. Electr. Eng. Informat.*, vol. 11, 2013, pp. 366–373.

[4] F. B. Bottura, W. M. S. Bernardes, M. Oleskovicz, and E. N. Asada, "Setting directional overcurrent protection parameters using hybrid GA optimizer," *Electr. Power Syst. Res.*, vol. 143, pp. 400–408, Feb. 2017.

[5] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: Past, present, and future," *Multimedia Tools Appl.*, vol. 80, no. 5, pp. 8091–8126, Feb. 2021.

[6] M. Isiet and M. Gadala, "Sensitivity analysis of control parameters in particle swarm optimization," *J. Comput. Sci.*, vol. 41, Mar. 2020, Art. no. 101086.

[7] J. Moirangthem, K. R. Krishnanand, S. S. Dash, and R. Ramaswami, "Adaptive differential evolution algorithm for solving non-linear coordination problem of directional overcurrent relays," *IET Gener., Transmiss. Distrib.*, vol. 7, no. 4, pp. 329–336, Apr. 2013.

[8] I. Rojas, J. Gonzalez, H. Pomares, J. J. Merelo, P. A. Castillo, and G. Romero, "Statistical analysis of the main parameters involved in the design of a genetic algorithm," *IEEE Trans. Syst., Man Cybern. C, Appl. Rev.*, vol. 32, no. 1, pp. 31–37, Feb. 2002.

[9] A. Hassanat, K. Almohammadi, and E. Abunawas, "Choosing mutation and crossover ratios for genetic algorithms—A review with a new dynamic approach," *Information*, vol. 10, no. 12, pp. 390–399, 2019.

[10] C. K. H. Lee, "A review of applications of genetic algorithms in operations management," *Eng. Appl. Artif. Intell.*, vol. 76, pp. 1–12, Nov. 2018.

[11] P. Charbonneau, "An introduction to genetic algorithms for numerical optimization," Nat. Center Atmos. Res., Boulder, CO, USA, Tech. Rep. 450, Mar. 2002, pp. 4–7.

[12] M. I. G. Arenas, "Statistical analysis of parameter setting in real-coded evolutionary algorithms," in *Proc. Int. Conf. Parallel Problem-Solving Nature*, Berlin, Germany, Sep. 2010, pp. 452–461.

[13] L. Nünez-Letamendia, "Fitting the control parameters of a genetic algorithm: An application to technical trading systems design," *Eur. J. Oper. Res.*, vol. 179, no. 3, pp. 847–868, Jun. 2007.

[14] X. Hu, R. C. Eberhart, and Y. Shi, "Engineering optimization with particle swarm," in *Proc. IEEE Swarm Intell. Symp.*, Indianapolis, IN, USA, Apr. 2003, pp. 53–57.

[15] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *Proc. IEEE Int. Conf. Evol. Comput. World Congr. Comput. Intell.*, May 1998, pp. 69–73.

[16] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, vol. 6, Nov./Dec. 1995, pp. 1942–1948.

[17] J. Kennedy, "The behavior of particles," in *Proc. Int. Conf. Evol. Program.*, San Diego, CA, USA, Mar. 1998, pp. 581–589.

[18] M. Juneja and S. Tiwari, "Reduced order modeling of triple link inverted pendulum using particle swarm optimization algorithm," *Int. J. Adv. Electron. Electr. Eng.*, vol. 3, no. 3, pp. 77–82, 2014.

[19] T. Beielstein, K. E. Parsopoulos, and M. N. Vrahatis, "Tuning PSO parameters through sensitivity analysis," in *Proc. Int. Conf. Evol. Program.*, Dortmund, Germany, Jan. 2002, pp. 1–6.

[20] D. Svozil, V. Kvasnicka, and J. Pospichal, "Introduction to multi-layer feed-forward neural networks," *Chemometric Intell. Lab. Syst.*, vol. 39, no. 1, pp. 43–62, Nov. 1997.

[21] H. Zayandehroodi, A. Mohamed, H. Shareef, and M. Farhoodnea, "A novel neural network and backtracking based protection coordination scheme for distribution system with distributed generation," *Int. J. Electr. Power Energy Syst.*, vol. 43, no. 1, pp. 868–879, Dec. 2012.

[22] N. Rezaei and M.-R. Haghifam, "Protection scheme for a distribution system with distributed generation using neural networks," *Int. J. Electr. Power Energy Syst.*, vol. 30, no. 4, pp. 235–241, May 2008.

[23] S. A. M. Javadian, M.-R. Haghifam, and N. Rezaei, "A fault location and protection scheme for distribution systems in presence of dg using MLP neural networks," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, Iran, Jul. 2009, pp. 1–8.

[24] M. J. Daryani, A. E. Karkevandi, and O. Usta, "A hybrid approach for microgrid protection system based on neural network and fuzzy logic," in *Proc. IEEE PES Innov. Smart Grid Technol. Eur. (ISGT-Eur.)*, Sep. 2019, pp. 1–5.

[25] E. Sorrentino and N. Gupta, "Summary of useful concepts about the coordination of directional overcurrent protections," *CSEE J. Power Energy Syst.*, vol. 5, no. 3, pp. 382–390, Sep. 2019.

[26] E. Sorrentino and M. Navas, "Two improvements related to overcurrent functions for bus protection in distribution systems," *IEEE Trans. Power Del.*, vol. 30, no. 3, pp. 1634–1635, Jun. 2015.

[27] C.-H. Lee and C.-R. Chen, "Using genetic algorithm for overcurrent relay coordination in industrial power system," in *Proc. Int. Conf. Intell. Syst. Appl. Power Syst.*, Nov. 2007, pp. 1–5.

[28] P. K. Choudhary and D. K. Das, "Optimal coordination of over-current relay in a power distribution network using opposition based learning fractional order class topper optimization (OBL-FOCTO) algorithm," *Appl. Soft Comput.*, vol. 113, Dec. 2021, Art. no. 107916.

[29] T. Keil and J. Jager, "Advanced coordination method for overcurrent protection relays using nonstandard tripping characteristics," *IEEE Trans. Power Del.*, vol. 23, no. 1, pp. 52–57, Jan. 2008.

[30] S. N. Langazane and A. K. Saha, "Effects of particle swarm optimization and genetic algorithm control parameters on overcurrent relay selectivity and speed," *IEEE Access*, vol. 10, pp. 4550–4567, 2022.

[31] E. O. Schweitzer, D. Whitehead, H. J. A. Ferrer, D. A. Tziouvaras, D. A. Costello, and D. S. Escobedo, "Line protection: Redundancy, reliability, and affordability," in *Proc. 64th Annu. Conf. Protective Relay Engineers*, Spokane, WA, USA, Apr. 2011, pp. 1–24.

[32] H. J. Altuve, K. Zimmerman, and D. Tziouvaras, "Maximizing line protection reliability, speed, and sensitivity," in *Proc. 69th Annu. Conf. Protective Relay Eng. (CPRE)*, Spokane, WA, USA, Apr. 2016, p. 128.

[33] B. Ram and N. D. Vishwakarma, *Power System Protection and Switchgear*. New Delhi, India: McGraw-Hill, 2011.

[34] J. M. Gers and E. J. Holmes, *Protection of Electricity Distribution Networks*. London, U.K.: The Institution of Electrical Engineers, 1998.

[35] S. Kudkelwar and D. Sarkar, "Online implementation of time augmentation of over current relay coordination using water cycle algorithm," *Social Netw. Appl. Sci.*, vol. 1, no. 12, pp. 16–28, Dec. 2019.

[36] M. Rojnić, R. Prenc, H. Bulat, and D. Franković, "A comprehensive assessment of fundamental overcurrent relay operation optimization function and its constraints," *Energies*, vol. 15, no. 4, p. 1271, Feb. 2022.

[37] L. Huchel and H. H. Zeineldin, "Planning the coordination of directional overcurrent relays for distribution systems considering DG," *IEEE Trans. Smart Grid*, vol. 7, no. 3, pp. 1642–1649, May 2016.

[38] V. Maheshwari, B. D. Devulapalli, and A. K. Saxena, "FPGA-based digital overcurrent relay with concurrent sense-process-communicate cycles," *Int. J. Electr. Power Energy Syst.*, vol. 55, pp. 66–73, Feb. 2014.

[39] C.-R. Chen, C.-H. Lee, and C.-J. Chang, "Optimal overcurrent relay coordination in power distribution system using a new approach," *Int. J. Electr. Power Energy Syst.*, vol. 45, no. 1, pp. 217–222, Feb. 2013.

[40] H. Amin, "Piecewise linear approximation applied to the nonlinear function of a neural network," *IEEE Proc. Circuits., Devices Syst.*, vol. 144, no. 6, pp. 313–337, Dec. 1997.

[41] M. Y. Shih, C. A. C. Salazar, and A. C. Enríquez, "Adaptive directional overcurrent relay coordination using ant colony optimisation," *IET Gener., Transmiss. Distrib.*, vol. 9, no. 14, pp. 2040–2049, Nov. 2015.

[42] R. Benabid, M. Zellagui, A. Chaghi, and M. Boudour, "Application of firefly algorithm for optimal directional overcurrent relays coordination in the presence of IFCL," *Int. J. Intell. Syst. Appl.*, vol. 6, no. 2, pp. 44–53, Jan. 2014.

[43] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE Trans. Evol. Comput.*, vol. 10, no. 3, pp. 281–295, Jun. 2006.

[44] X. Hu and R. Eberhart, "Multi-objective optimization using dynamic neighborhood particle swarm optimization," in *Proc. Congr. Evol. Comput.*, Honolulu, HI, USA, Jun. 2002, pp. 1667–1681.

[45] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. Comput. Cybern. Simulation*, Orlando, FL, USA, Oct. 1997, pp. 4104–4109.

[46] H. A. Tresnadi, T. A. Nugroho, M. Abdillah, and N. I. Pertiwi, "Coordination of directional overcurrent relay using PSO-DE algorithm," in *Proc. Int. Conf. Technol. Policy Energy Electr. Power (ICT-PEP)*, Sep. 2021, pp. 412–417.

[47] C. A. C. Coello, "Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art," *Comput. Methods Appl. Mech. Eng.*, vol. 191, nos. 11–12, pp. 1245–1287, Jan. 2002.

[48] E. Mezura-Montes and C. A. C. Coello, "An empirical study about the usefulness of evolution strategies to solve constrained optimization problems," *Int. J. Gen. Syst.*, vol. 37, no. 4, pp. 443–473, Aug. 2008.

[49] I. Mazhoud, K. Hadj-Hamou, J. Bigeon, and P. Joyeux, "Particle swarm optimization for solving engineering problems: A new constraint-handling mechanism," *Eng. Appl. Artif. Intell.*, vol. 26, no. 4, pp. 1263–1273, Apr. 2013.

[50] P. A. Shah, A. S. Nanoty, and V. N. Rajput, "Comparative analysis of different optimization methods for optimal coordination of directional overcurrent relays," in *Proc. Int. Conf. Electr., Electron., Optim. Techn. (ICEEOT)*, Mar. 2016, pp. 3927–3931.

[51] S. Mirjalili, G.-G. Wang, and L. D. S. Coelho, "Binary optimization using hybrid particle swarm optimization and gravitational search algorithm," *Neural Comput. Appl.*, vol. 25, no. 6, pp. 1423–1435, Nov. 2014.

[52] E. F. Campana, G. Liuzzi, S. Lucidi, D. Peri, V. Piccialli, and A. Pinto, "New global optimization methods for ship design problems," *Optim. Eng.*, vol. 10, no. 4, pp. 533–555, Dec. 2009.

[53] A. Serani, C. Leotardi, U. Iemma, E. F. Campana, G. Fasano, and M. Diez, "Parameter selection in synchronous and asynchronous deterministic particle swarm optimization for ship hydrodynamics problems," *Appl. Soft Comput.*, vol. 49, pp. 313–334, Dec. 2016.

[54] R. Pellegrini, A. Serani, C. Leotardi, U. Iemma, E. F. Campana, and M. Diez, "Formulation and parameter selection of multi-objective deterministic particle swarm for simulation-based optimization," *Appl. Soft Comput.*, vol. 58, pp. 714–731, Sep. 2017.

[55] Y. Muhammad, M. A. Z. Raja, M. A. A. Shah, S. E. Awan, F. Ullah, N. I. Chaudhary, K. M. Cheema, A. H. Milyani, and C.-M. Shu, "Optimal coordination of directional overcurrent relays using hybrid fractional computing with gravitational search strategy," *Energy Rep.*, vol. 7, pp. 7504–7519, Nov. 2021.

[56] K. T. Chaturvedi, M. Pandit, and L. Srivastava, "Particle swarm optimization with time varying acceleration coefficients for non-convex economic power dispatch," *Int. J. Electr. Power Energy Syst.*, vol. 31, no. 6, pp. 249–257, Jul. 2009.

[57] W.-N. Chen, J. Zhang, Y. Lin, N. Chen, Z.-H. Zhan, H. S. Chung, Y. Li, and Y.-H. Shi, "Particle swarm optimization with an aging leader and challengers," *IEEE Trans. Evol. Comput.*, vol. 17, no. 2, pp. 241–258, Apr. 2013.

[58] M. He, M. Liu, R. Wang, X. Jiang, B. Liu, and H. Zhou, "Particle swarm optimization with damping factor and cooperative mechanism," *Appl. Soft Comput.*, vol. 76, pp. 45–52, Mar. 2019.

[59] Y. Shi and R. C. Eberhart, "Empirical study of particle swarm optimization," in *Proc. Congr. Evol. Comput. (CEC)*, Washington, DC, USA, Jul. 1999, pp. 1945–1950.

[60] R. C. Eberhart and Y. Shi, "Comparing inertia weights and constriction factors in particle swarm optimization," in *Proc. Congr. Evol. Comput. (CEC)*, La Jolla, CA, USA, Jun. 2000, pp. 84–88.

[61] J. Xin, G. Chen, and Y. Hai, "A particle swarm optimizer with multi-stage linearly-decreasing inertia weight," in *Proc. Int. Joint Conf. Comput. Sci. Optim.*, vol. 1, Xian, China, Apr. 2009, pp. 505–508.

[62] A. P. Engelbrecht, *Computational Intelligence: An Introduction*, 2nd ed. Hoboken, NJ, USA: Wiley, 2007.

[63] J. S. Arora, *Introduction to Optimum Design*, 4th ed. New York, NY, USA: Academic, 2017.

[64] A. Banks, J. Vincent, and C. Anyakoha, "A review of particle swarm optimization. Part I: Background and development," *Natural Comput.*, vol. 6, no. 4, pp. 467–484, Dec. 2007.

[65] F. Vandenbergh and A. Engelbrecht, "A study of particle swarm optimization particle trajectories," *Inf. Sci.*, vol. 176, no. 8, pp. 937–971, Apr. 2006.

[66] K. R. Harrison, B. M. Ombuki-Berman, and A. P. Engelbrecht, "An analysis of control parameter importance in the particle swarm optimization algorithm," in *Proc. Int. Conf. Swarm Intell.* Cham, Switzerland: Springer, 2019, pp. 93–105.

[67] H. T. Liang and F. H. Kang, "Adaptive mutation particle swarm algorithm with dynamic nonlinear changed inertia weight," *Optik*, vol. 127, no. 19, pp. 8036–8042, Oct. 2016.

[68] C. Ze, D. Mengnan, Y. Tiankai, and H. Lijie, "Extraction of solar cell model parameters based on self-adaptive chaos particle swarm optimization algorithm," *Trans. China Electrotech. Soc.*, vol. 29, no. 9, pp. 245–252, 2014.

[69] M. Z. Shirazi, T. Pamulapati, R. Mallipeddi, and K. C. Veluvolu, "Particle swarm optimization with ensemble of inertia weight strategies," in *Proc. Interface Conf. Swarm Intel.* Cham, Switzerland: Springer, 2017, pp. 140–147.

[70] H. S. Maharana and S. K. Dash, "Comparative optimization analysis of ramp rate constriction factor based PSO and electro magnetism based PSO for economic load dispatch in electric power system," in *Proc. Int. Conf. Appl. Mach. Learn. (ICAML)*, May 2019, pp. 63–68.

[71] A. A. Nagra, F. Han, Q. H. Ling, M. Abubaker, F. Ahmad, S. Mehta, and A. T. Apasiba, "Hybrid self-inertia weight adaptive particle swarm optimisation with local search using C4.5 decision tree classifier for feature selection problems," *Connection Sci.*, vol. 32, no. 1, pp. 16–36, Jan. 2020.

[72] A. Faryal, M. Yesir, S. Marium, K. Samina, and A. Iftikhar, "Rank-based self-adaptive inertia weight scheme to enhance the performance of novel binary particle swarm optimization," in *Proc. Int. Sci. Conf. Sinteza*, Sep. 2021, pp. 63–69.

[73] J. Zhang, J. Sheng, J. Lu, and L. Shen, "UCPSO: A uniform initialized particle swarm optimization algorithm with cosine inertia weight," *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–18, Mar. 2021.

[74] J. H. Holland, *Adaptation in Natural and Articial Systems: An Introductory Analysis with Applications to Biology, Control, and Articial Intelligent*. Cambridge, MA, USA: MIT Press, 1992.

[75] R. L. Hauptand and S. E. Haupt, *Practical Genetic Algorithms*. Hoboken, NJ, USA: Wiley, 2004.

[76] X. Hu and R. Eberhart, "Solving constrained nonlinear optimization problems with particle swarm optimization," in *Proc. 6th World Multiconf. Syst., Cybern. Inform.*, Orlando, FL, USA, Jul. 2002, pp. 203–206.

[77] Z. Michalewicz, *Genetic Algorithms C Data Structures D Evolution Programs (AI Series)* (AI Series). New York, NY, USA: Springer-Verlag, 1994.

[78] H. Z. Ahmed, "Genetic algorithm for the traveling salesman problem using sequential constructive crossover operator," in *Proc. Int. J. Biometrics Bioinf. (IJBB)*, vol. 3, 2010, p. 96.

[79] K. P. Ghadle and Y. M. Muley, "Travelling salesman problem with MATLAB programming," *Int. J. Adv. Appl. Math. Mech.*, vol. 2, no. 3, pp. 258–266, 2015.

[80] G. Bakrl, D. Birant, and A. Kut, "An incremental genetic algorithm for classification and sensitivity analysis of its parameters," *Expert Syst. Appl.*, vol. 38, no. 3, pp. 2609–2620, Mar. 2011.

[81] F. G. Lobo, C. F. Lima, and Z. Michalewicz, *Parameter Setting in EvoLutionary Algorithms*, vol. 54. Berlin, Germany: Springer, Mar. 2007, p. 118.

[82] S. Akter, N. Nahar, M. ShahadatHossain, and K. Andersson, "A new crossover technique to improve genetic algorithm and its application to TSP," in *Proc. Int. Conf. Electr., Comput. Commun. Eng. (ECCE)*, Feb. 2019, pp. 1–6.

[83] S. C. Chu, J. C. W. Lin, and J. Li, *Genetic and Evolutionary Computing*. Singapore: Springer, 2022.

[84] A. Basak, "A rank based adaptive mutation in genetic algorithm," 2021, *arXiv:2104.08842*.

[85] I. Korejo, S. Yang, K. Brohi, and Z. U. A. Khuhro, "Multi-population methods with adaptive mutation for multi-modal optimization problems," *Int. J. SCAI*, vol. 2, no. 2, pp. 1–19, 2013.

[86] N. Stark, G.F. Minetti, and C. Salto, "A new strategy for adapting the mutation probability in genetic algorithms," in *Proc. XVIII Congreso Argentino de Ciencias de la Computación*, Mar. 2012, pp. 1–9.
[87] D. Thierens, "Adaptive mutation rate control schemes in genetic algorithms," in *Proc. Congr. Evol. Comput. (CEC)*, May 2002, pp. 980–985.
[88] S. Blum, R. Puisa, J. Riedel, and M. Wintermantel, "Adaptive mutation strategies for evolutionary algorithms," in *Proc. Annu. Conf., EVEN Weimarer Optimierungsund Stochastiktage*, vol. 2, 2001, pp. 501–511.
[89] W. Z. Jian, "Design of mutation operator based on information entropy," in *Proc. 2nd Int. Conf. Intell. Comput. Technol. Autom.*, 2009, pp. 264–266.
[90] S. D. Saldarriaga-Zuluaga, J. M. López-Lezama, and N. Muñoz-Galeano, "Hybrid harmony search algorithm applied to the optimal coordination of overcurrent relays in distribution networks with distributed generation," *Appl. Sci.*, vol. 11, no. 19, p. 9207, Oct. 2021.
[91] R. Poli, "Analysis of the publications on the applications of particle swarm optimisation," *J. Artif. Evol. Appl.*, vol. 2008, pp. 1–10, Feb. 2008.
[92] M. Arumugam, M. Rao, and A. Chandramohan, "A new and improved version of particle swarm optimization algorithm with globallocal best parameters," *Knowl. Inf. Syst.*, vol. 16, no. 3, pp. 324–350, 2008.
[93] R. Perveen, "Hybrid optimal coordination of overcurrent relay and distance relay for offshore wind farm," in *Proc. 2nd Global Conf. Advancement Technol. (GCAT)*, Oct. 2021, pp. 1–6.
[94] J. Barrera, O. Á. Bajo, J. J. Flores, and C. A. C. Coello, "Limiting the velocity in the particle swarm optimization algorithm," *Computación y Sistemas*, vol. 20, no. 4, pp. 635–645, Dec. 2016.
[95] B. Yang, Y. Chen, and Z. Zhao, "A hybrid evolutionary algorithm by combination of PSO and GA for unconstrained and constrained optimization problems," in *Proc. IEEE Int. Conf. Control Autom.*, May 2007, pp. 166–170.
[96] S. T. P. Srinivas and S. K. Swarup, "Application of improved invasive weed optimization technique for optimally setting directional overcurrent relays in power systems," *Appl. Soft Comput.*, vol. 79, pp. 1–13, Jun. 2019.
[97] B. Akay and D. Karaboga, "Artificial bee colony algorithm for large-scale problems and engineering design optimization," *J. Intell. Manuf.*, vol. 23, no. 4, pp. 1001–1014, Aug. 2012.
[98] D. Acharya and D. K. Das, "An efficient optimizer for optimal overcurrent relay coordination in power distribution system," *Expert Syst. Appl.*, vol. 199, Aug. 2022, Art. no. 116858.
[99] F. C. Sampaio, F. L. Tofoli, L. S. Melo, G. C. Barroso, R. F. Sampaio, and R. P. S. Leão, "Adaptive fuzzy directional bat algorithm for the optimal coordination of protection systems based on directional overcurrent relays," *Electric Power Syst. Res.*, vol. 211, Oct. 2022, Art. no. 108619.
[100] M. N. Alam, "Adaptive protection coordination scheme using numerical directional overcurrent relays," *IEEE Trans. Ind. Informat.*, vol. 15, no. 1, pp. 64–73, Jan. 2019.
[101] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. 6th Int. Symp. Micro Mach. Hum. Sci.*, Oct. 1995, pp. 39–43.
[102] R. C. Eberhart and X. Hu, "Human tremor analysis using particle swarm optimization," in *Proc. Congr. Evol. Comput. (CEC)*, Washington, WA, USA, Jul. 2002, pp. 1927–1930.
[103] M. Mishra, R. R. Panigrahi, and P. K. Rout, "A combined mathematical morphology and extreme learning machine techniques-based approach to micro-grid protection," *Ain Shams Eng. J.*, vol. 10, pp. 307–318, Jun. 2019.
[104] J. J. Q. Yu, Y. Hou, A. Y. S. Lam, and V. O. K. Li, "Intelligent fault detection scheme for microgrids with wavelet-based deep neural networks," *IEEE Trans. Smart Grid*, vol. 10, no. 2, pp. 1694–1703, Mar. 2019.
[105] M. E. Baran, H. Hooshyar, Z. Shen, and A. Huang, "Accommodating high PV penetration on distribution feeders," *IEEE Trans. Smart Grid*, vol. 3, no. 2, pp. 1039–1046, Jun. 2012.
[106] M. Bakkar, S. Bogarra, F. Córcoles, A. Aboelhassan, S. Wang, and J. Iglesias, "Artificial intelligence-based protection for smart grids," *Energies*, vol. 15, no. 13, p. 4933, Jul. 2022.
[107] N. A. Alawad and B. H. Abed-Alguni, "Discrete island-based cuckoo search with highly disruptive polynomial mutation and opposition-based learning strategy for scheduling of workflow applications in cloud environments," *Arabian J. Sci. Eng.*, vol. 46, no. 4, pp. 3213–3233, Apr. 2021.
[108] J. Radosavljević and M. Jevtić, "Hybrid GSA-SQP algorithm for optimal coordination of directional overcurrent relays," *IET Gener., Transmiss. Distrib.*, vol. 10, no. 8, pp. 1928–1937, May 2016.
[109] F. G. Roba, D. S. Arya, and N. K. Goel, "Streamflow forecasting using artificial neural network," *Water Energy Int.*, vol. 57, pp. 30–37, Jun. 2000.
[110] C. C. Chiu, D. F. Cook, J. J. Pignatiello, and A. D. Whittaker, "Design of a radial basis function neural network with a radius modification algorithm using response surface methodology," *J. Intell. Manuf.*, vol. 8, pp. 117–124, Mar. 1997.
[111] N. M. Nawi, A. Khan, and M. Z. Rehman, "A new Levenberg Marquardt based back propagation algorithm trained with cuckoo search," *Proc. Technol.*, vol. 11, pp. 18–23, Jan. 2013.
[112] H. Yonaba, F. Anctil, and V. Fortin, "Comparing sigmoid transfer functions for neural network multistep ahead streamflow forecasting," *J. Hydrolog. Eng.*, vol. 15, no. 4, pp. 275–283, Apr. 2010.
[113] Y. Tan and A. Van Cauwenberghe, "Neural-network-based d-step-ahead predictors for nonlinear systems with time delay," *Eng. Appl. Artif. Intell.*, vol. 12, no. 1, pp. 21–35, Feb. 1999.
[114] S. Karupiah, M. Hussain, I. Musirin, and S. R. A. Rahim, "Prediction of overcurrent relay miscoordination time using urtificial neural network," *Indonesian J. Electr. Eng. Comp. Sc.*, vol. 14, no. 1, pp. 319–326, 2019.
[115] H. Lin, K. Sun, Z. Tan, C. Liu, J. M. Guerrero, and J. C. Vasquez, "Adaptive protection combined with machine learning for microgrids," *IET Gener., Transmiss. Distrib.*, vol. 13, no. 6, pp. 770–779, Mar. 2019.
[116] H. Saberi and T. Amraee, "Coordination of directional over-current relays in active distribution networks using generalised benders decomposition," *IET Gener., Transmiss. Distrib.*, vol. 11, no. 16, pp. 4078–4086, Nov. 2017.
[117] K. S. M. H. Ibrahim, Y. F. Huang, A. N. Ahmed, C. H. Koo, and A. El-Shafie, "A review of the hybrid artificial intelligence and optimization modelling of hydrological streamflow forecasting," *Alexandria Eng. J.*, vol. 61, no. 1, pp. 279–303, Jan. 2022.

SETHEMBISO NONJABULO LANGAZANE was born in Margate, KwaZulu-Natal, South Africa. She received the B.Sc. and M.Sc. degrees in electrical engineering from the University of KwaZulu-Natal, where she is currently pursuing the Ph.D. degree in electrical engineering. Her research interests include power systems protection and artificial intelligence.

AKSHAY KUMAR SAHA (Senior Member, IEEE) is currently a Professor and an Academic Leader Research and Higher Degrees with the School of Engineering, University of KwaZulu-Natal, Durban, South Africa. His research works are related to advances of power systems in various areas, including engineering education. He has published more than 60 articles in top-tier international journals and over 100 international conference papers in relevant areas. He is also a Registered Professional Engineer with the Engineering Council of South Africa and a fellow of South African Institute of Electrical Engineers, South African Academy of Engineering. He is also a Senior Member of SAIMC, a member of Academy of Science of South Africa (ASSAf), and an Individual Member Cigre. He is also a member of IEEE HKN. He was awarded the Best Lecturer Electrical Engineering, in 2013 and 2014, (2016–2019) by the School of Engineering, Research Excellence Award, in 2015 and 2021, and Top-30 Publishing Research Award, in 2020 and 2022, by the University of KwaZulu-Natal. He is also acting as an editorial board member of a number of top-tier international journals.