IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 9, 2021, accepted March 6, 2021, date of publication March 17, 2021, date of current version March 25, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3066329

Recent Methodology-Based Gradient-Based Optimizer for Economic Load Dispatch Problem

SANCHARI DEB[1], DIAA SALAMA ABDELMINAAM[2,3], MOKHTAR SAID[4], AND ESSAM H. HOUSSEIN[5]

[1]VTT Technical Research Centre, 02100 Espoo, Finland
[2]Faculty of Computers and Artificial Intelligence, Benha University, Benha 12311, Egypt
[3]Faculty of Computers Science, Misr International University, Cairo 11311, Egypt
[4]Electrical Engineering Department, Faculty of Engineering, Fayoum University, Fayoum 63514, Egypt
[5]Faculty of Computers and Information, Minia University, Minia 61519, Egypt

Corresponding author: Diaa Salama Abdelminaam (diaa.salama@fci.bu.edu.eg)

This work was supported in part by the Misr International University, under Grant DSA28211231302952.

ABSTRACT Economic load dispatch (ELD) in power system problems involves scheduling the power generating units to minimize cost and satisfy system constraints. Although previous works propose solutions to reduce CO2 emission and production cost, an optimal allocation needs to be considered on both cost and emission—leading to combined economic and emission dispatch (CEED). Metaheuristic optimization algorithms perform relatively well on ELD problems. The gradient-based optimizer (GBO) is a new metaheuristic algorithm inspired by Newton's method that integrates both the gradient search rule and local escaping operator. The GBO maintains a good balance between exploration and exploitation. Also, the possibility of the GBO getting stuck in local optima and premature convergence is rare. This paper tests the performance of GBO in solving ELD and CEED problems. We test the performance of GBO on ELD for various scenarios such as ELD with transmission losses, CEED and CEED with valve point effect. The experimental results revealed that GBO has been obtained better results compared to eight other metaheuristic algorithms such as Slime mould algorithm (SMA), Elephant herding optimization (EHO), Monarch butterfly optimization (MBO), Moth search algorithm (MSA), Earthworm optimization algorithm (EWA), Artificial Bee Colony (ABC) Algorithm, Tunicate Swarm Algorithm (TSA) and Chimp Optimization Algorithm (ChOA). Therefore, the simulation results showed the competitive performance of GBO as compared to other benchmark algorithms.

INDEX TERMS Gradient-based optimizer (GBO), economic load dispatch (ELD), combined economic and emission dispatch (CEED), metaheuristics, optimization.

ABBREVIATIONS

ELD	Economic Load Dispatch
PSO	Particle Swarm Optimization
QBA	Quantum Bat Algorithm
MBA	Mine Blast Algorithm
MFO	Moth Flame Optimizer
ALO	Ant Lion Optimization
SSA	Salp Swarm Algorithm
GA	Genetic Algorithm
ADFA	Ameliorated Dragonfly Algorithm
GBO	Gradient-Based Optimizer
ACTO	Aggrandized CTO
MODE	Multi-objective DE
SMA	Slime mould algorithm
MBO	Monarch butterfly optimization
CEED	Economic and Emission Dispatch
TCO	Termite Colony Optimization
ACS	Artificial Cooperative Search
SHO	Spotted Hyena Optimizer
GWO	Grey Wolf Optimizer
ACCS	Adaptive Charged System Search
ISFS	Improved Stochastic Fractal Search
LFA	Lighting Flash Algorithm
MSSA	Modified Social Spider Algorithm
TSA	Tunicate Swarm Algorithm
NSGA	Non Dominated Sorting GA
PDE	Pareto DE

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

EHO	Elephant herding optimization
EWA	Earthworm optimization algorithm
CTO	Class Topper Optimization
BA	Bat Algorithm
FA	Firefly Algorithm
WOA	Whale Optimization Algorithm
AEFA	Artificial Electric Field Algorithm
ABC	Artificial Bee Colony
MFO	Moth Flame Optimization
SOS	Symbiotic Organism Search
FPA	Flower Pollination Algorithm
ChOA	Chimp Optimization Algorithm
DE	Differential evolution
WCA	Water Cycle Algorithm
MSA	Moth search algorithm
TLBO	Teaching Learning Based Optimization

I. INTRODUCTION

Engineers of 21^{st} century are curious about the increasing complexity of the societal and technological challenges such as the parameter extraction problem in photovoltaic (PV) [1] and the problem of the Economic Load Dispatch (ELD) [2]. The ELD involves involves minimizing production costs by allocating power produced by each power system unit economically [3]. Although some solutions are proposed to reduce emissions and production costs [4] by considering both cost and emission optimally, leading to combined economic and emission dispatch (CEED), the ELD and CEED solutions require efficient optimization algorithms.

Several metaheuristic optimization algorithms are proposed to solve a wide range of real-life problems. For example, nature-inspired algorithms mimic the biological, physical, or environmental processes [5]–[7]. The metaheuristic algorithms' versatility and gradient-free features consider black-box problems in addition to the theoretical developments and significant advantages. The resulting search space is not limited, making the algorithms scalable for solving different problems. Real problems are solved more effectively since solutions are not restricted to locally optimal approaches. The metaheuristic algorithms are applied in various fields and proved helpful [8]–[12].

Several metaheuristic optimization algorithms are proposed to solve a wide range of real-life problems. For example, nature-inspired algorithms mimic the biological, physical, or environmental processes [5], [6]. Nature-inspired algorithms are designed to mimic the biological, physical, or environmental processes they are modeled after [7], [8]. The metaheuristic algorithms' versatility and gradient-free features consider black-box problems in addition to the theoretical developments and significant advantages. The resulting search space is not limited, making the algorithms scalable for solving different problems. Real problems are solved more effectively since solutions are not restricted to locally optimal approaches. The metaheuristic algorithms are applied in various fields and proved helpful [9], [10]. Metaheuristics have proved to be useful in various problems, including this one [11]–[13].

The Economic Load Dispatch (ELD) plays an important role in powering the electrical loads and reducing emissions with the guarantee of meeting the equality and inequality constraints [14]–[17]. Metaheuristic optimization algorithms (MHs) performs relatively well on the ELD problem. This work tests the performance of Gradient Based Optimizer (GBO) in solving ELD and CEED. GBO is a new metaheuristic inspired by the gradient based Newton method involving Gradient search rule (GSR) and local escaping operator (GEO). GBO has good balance between exploration and exploitation.

This work uses a new met-heuristic algorithm called Gradient-Based Optimizer (GBO) [18] was developed by Ahmadianfar *et al.* in 2020, which was one of the most promising algorithms for solving different variants of ELD. GBO is a metaheuristic inspired by the gradient based Newton method involving Gradient search rule (GSR) and local escaping operator (GEO). To evaluate various characteristics of the GBO, 28 mathematical test functions were first used and then six engineering problems were optimized by the GBO. Moreover, the exploitative, exploratory, and local optima avoidance of GBO was also investigated using unimodal, multi-modal and composition problems. Finally, the results show that GBO was capable of finding excellent solutions compared to other well-regarded optimizers. Also, the possibility of getting stuck in local optima and premature convergence is rare in GBO. The performance of GBO on ELD is tested for various scenarios such as ELD with transmission losses, CEED, CEED with valve point effect and for various test networks. The performance of GBO is compared with compared with eight other metaheuristic algorithms such as Slime mould algorithm (SMA) [19], Elephant herding optimization (EHO) [20], Monarch butterfly optimization (MBO) [21], Moth search algorithm (MSA) [22], Earthworm optimization algorithm (EWA) [23], Artificial Bee Colony (ABC) Algorithm [24], Tunicate Swarm Algorithm (TSA) [25] and Chimp Optimization Algorithm (ChOA) [26].

The rest of the paper is organized as follows. Section III elaborates the ELD problem, then an overview for the Gradient-Based Optimizer (GBO) is presented in Section IV. The obtained findings and discussion is introduced in Section V. Finally, Section VI concludes the work.

II. RELATED WORK

An overview of metaheuristics used for solving the different variants of ELD is as shown in Table 1. In [27], authors have used CTO and ACTO to solve ELD as well as CEED. It was observed that CTO performs better than other metaheuristics such as TLBO, DE, GA, PSO etc. Dynamic ELD was solved by hybrid PSO TCO [28]. The hybridization of PSO and TCO favoured faster convergence and produced better quality of solutions. The authors performed ELD considering renewable resources [29]. A modified version of BA was used for solving the problem. In [30], a quantum inspired BA was solved for using ELD. It was observed that the premature convergence was avoided by the modified quantum inspired BA. Dynamic ELD considering renewable sources was solved by multiswarm PSO [31]. Quantum inspired BA was used for solving ELD considering valve point effect [32]. In [33], the authors utilized ACO to solve different variants of ELD. Non-convex ELD was solved by ACS in [17]. The authors have hybridized BA and FA for solving ELD and CEED in [34]. A self-adaptive version of Jaya algorithm was used for solving ELD in [34]. It was observed that the modified version of Jaya algorithm performed better that the basis Jaya algorithm and TLBO in solving the ELD problem. In [35], the authors proposed MBS for solving CEED with valve point effect. A drift mechanism in the self-adaptive version of PSO was introduced and used for solving the ELD problem [36]. In [37], the authors proposed a hybrid ED SHO for solving ELD.

In the same context, in [38], [39], WOA, and MFO were used to solve different variants of ELD. In [40], ELD with valve point effect was solved by using modified GWO. In [41], ELD was solved by an improved version of PSO with inertia weights factor. In [42], [43], AEF, and ALO were used for solving ELD for a small-scale power system. In [44] adaptive charged system search algorithm is applied to solve EED of power systems. In [45], [46] the authors have utilized improved version of PSO for solving ELD. Authors solved the ELD problem for 24-hour load pattern by GWO [47]. An enhanced version of BA for solving ELD was proposed in [48], [49]. In [50], modified ABC algorithm was used for solving non-smooth dynamic ELD. A modified version of cultural algorithm having a local search component for solving ELD and CEED was proposed in [51]. A chaotic firefly algorithm having mutation operator is used for solving ELD in case of large scale power system having valve point effect and multiple fuel options [52].

In [53], the dynamic ELD problem was solved by modified TLBO. In [54], smooth as well as non-smooth ELD was solved by WCA. The ELD problem in presence of wind power was attacked by hybrid BA [55]. In [56]–[58], multi-area ELD was solved by SSA, FSA, and MFO algorithm. In [59], the authors used an improved version of TLBO for solving ELD problem considering distributed generation. A modified version of GWO was used for solving non-convex ELD for current power system scenario in [60]. In [61], dynamic ELD was solved by hybrid GA PSO. LFA and SSA was used for solving large scale dynamic ELD in [62], [63]. In [64], GWO was used for solving ELD and CEED with valve point effect. An improved version of DFA was used for solving ELD considering demand response and renewable resources [65]. In [66]–[68], chaotic bat, modified social spider and FPA algorithm was used for solving ELD. In [69] Levy flight Moth-Flame optimizer is proposed to solve the ELD. A novel parallel hurricane algorithm was used to solve ELD as well as CEED in [70]. An improved version of DE was used to solve ELD with and without valve point effect in [71].

From Table 1, it is observed that metaheuristics such as PSO, GA, BA, GWO are widely used by researchers in solving different variants of ELD. Despite the availability and use of different metaheuristics for solving ELD, researchers are still proposing new and novel algorithms for its solution. The prime motivation behind this is the No Free Lunch (NFL) theorem [72]. NFL theorem states that a single algorithm does not perform equally well on all the optimization problems. Hence, it is justified to propose new more efficient algorithms and improve the existing algorithms.

III. ECONOMIC LOAD DISPATCH PROBLEM

ELD is one of the prime and complex problems of modern power system planning and operation. The objective of ELD is to maximize the economic welfare of the power system subject to certain operational constraints thereby optimally allocating each production units and reducing the net fuel cost consumption. The different variants of ELD is elaborated in this section.

A. ECONOMIC LOAD DISPATCH (ELD) WITH LOSSES

The mathematical formulation of ELD with losses is explained in this section. For operating n generators, the overall fuel cost is:

$$\mathrm{Min}(F) = F_1\,(P_1) + \cdots F_n\,(P_n) \tag{1}$$

where F is the overall fuel cost, F_1 is the fuel cost of 1st generator and F_n is the fuel cost of nth generator.

The fuel cost function is further approximated in quadratic form as:

$$\mathrm{Min}(F) = \sum_{k=1}^{n} F_i\,(P_i) = \sum_{k=1}^{n} a_k P_k^2 + b_k P_k + c_k \tag{2}$$

where a, b, c are the weight constants of the fuel cost The minimization of fuel cost is performed subject to constraints given by equation (3) and (5)

$$\sum_{k=1}^{n} P_k - P_D - P_L = 0 \tag{3}$$

where P_D represents net demand of the network and P_L represents the transmission losses of the network

$$P_L = \sum_{i=1}^{n} \sum_{j=1}^{n} P_i B_{ij} P_j \tag{4}$$

where B_{ij} is the loss coefficient, P_i is the power generated at ith generator, and P_j is the power generated at jth generator

$$P_k^{min} \le P_k \le P_k^{max} \tag{5}$$

B. COMBINED ECONOMIC LOAD DISPATCH (CEED)

Stress has been laid on reduction of emission also in addition to production cost. Hence, optimal allocation is done considering both cost and emission leading to Combined Economic

TABLE 1. Recent research studies on ELD.

Ref	Year	Type	Algorithm
[27]	2020	ELD, ELD with valve point effect, CEED	CTO
[28]	2020	Dynamic ELD	PSO TCO
[29]	2020	ELD considering renewable sources	Modified BA
[30]	2020	ELD with losses	QBA
[31]	2020	Dynamic ELD considering renewable sources	PSO
[32]	2020	ELD with valve point effect	QBA
[33]	2020	ELD with losses	PSO
[17]	2019	Non convex ELD	ACS
[34]	2019	CEED	Hybrid FA BA
[73]	2019	ELD	Jaya Algorithm
[35]	2018	ELD with valve point effect, CEED	MBA
[36]	2017	ELD with valve point effect, ramp rates and prohibited zones	Improved PSO
[37]	2018	Convex and non-convex ELD	SHO
[38]	2018	ELD with ramp rates and prohibited zones	WOA
[39]	2018	Convex, non-convex and dynamic ELD	MFO
[40]	2019	ELD with valve point effect	Hybrid GWO
[41]	2019	ELD with losses	Improved PSO
[42]	2019	ELD with losses	AEFA
[43]	2017	Non- convex ELD	ALO
[44]	2018	ELD with losses	ACSS
[45]	2018	ELD with losses	Adaptive PSO
[46]	2017	ELD with losses	Enhanced PSO
[47]	2020	ELD with losses	GWO
[48]	2017	ELD with losses	Enhanced BA
[49]	2019	CEED	QBA
[50]	2018	Dynamic ELD, CEED	Modified ABC
[51]	2018	CEED	Enhanced Cultural algorithm
[52]	2018	Dynamic CEED	Chaotic FA
[53]	2020	Dynamic CEED	TLBO
[54]	2017	Smooth and non-smooth ELD	WCA
[55]	2018	ELD with renewable resources	Hybrid BA
[56]	2020	Multi-area ELD	SSA
[74]	2019	Multi-area ELD	Hybrid Jaya TLBO
[57]	2019	Multi-area ELD	ISFS
[75]	2018	CEED with renewable sources	MOEA/D
[58]	2017	Multi-area ELD	MFO
[59]	2019	CEED	Improved TLBO
[60]	2018	ELD with losses	Modified GWO
[61]	2018	Dynamic ELD	GA PSO
[62]	2017	ELD with valve point effect	LFA
[63]	2017	Dynamic ELD with valve point effect	SOS
[64]	2016	Non convex ELD	GWO
[65]	2019	ELD considering renewable energy sources	ADFA
[66]	2016	ELD with losses	Chaotic BA
[67]	2016	ELD with losses, valve point and ramp rates	MSSA
[68]	2016	ELD, CEED	FPA
[14]	2016	ELD	MFO
[70]	2018	ELD, CEED	Hurricane algorithm
[71]	2016	ELD with and without valve point	Improved DE

and Emission Dispatch (CEED). In CEED, both economic and emission dispatch are taken into consideration.

The emission dispatch problem is concerned with minimization of the gases from power plants. Mathematically, the emission factor is given by:

$$Min(E) = \sum_{k=1}^{n} E_i(P_i) = \sum_{k=1}^{n} \alpha_k P_k^2 + \beta_k P_k + \gamma_k \tag{6}$$

The objective function of the CEED problem is:

$$objective\,function = Min\left(\sum_{k=1}^{n} E_i(P_i) + h_e \sum_{k=1}^{n} F_i(P_i)\right) \tag{7}$$

where h_e is the price penalty factor as in equation 8:

$$h_e = \frac{F_i(P_{imax})}{E_i(P_{imax})} \tag{8}$$

The optimization is performed subject to constraints given by equations (3) and (5).

a: CEED WITH VALVE POINT EFFECT:

In modern era, steam turbines have multiple valves causing valve point effect. The valves make the cost function nonlinear as in equation 9:

$$Min(F) = \sum_{k=1}^{n} F_i = \sum_{k=1}^{n} a_k P_k^2 + b_k P_k + c_k + |e_k \sin(f_k \times (P_{kmin} - P_k))| \tag{9}$$

where e_k and f_k are the coefficients reflecting valve point effect of kth generator. The optimization is concerned with minimization of both cost and emission subject to constraints as in equations (3) and (5).

IV. GRADIENT-BASED OPTIMIZER (GBO)

Researchers have invented the GBO metaheuristic algorithm which mimics the population-based and gradient-based methods [18]. To explore the search space for a set of search metrics, Newton's approach is utilized. The main steps of GBO are as follows:

A. THE INITIALIZATION PROCESS

In GBO, the control parameters (α) and probability rate are used to balance and switch from the exploration to exploitation. Population and iterations numbers are related to the problem's complexity. In GBO, the vector of N vectors in D-dimensional space can be described. The initial vectors for the GBO are usually randomly generated in the D-dimensional search space.

$$X_n = X_{min} + rand(0, 1) \times (X_{max} - X_{min}) \tag{10}$$

where X_{min}, and X_{max} are the bounds of decision variables X, and $rand(0, 1)$ is a random number in [0, 1].

B. GRADIENT SEARCH RULE (GSR) PROCESS

In GBO algorithm, a significant factor ρ is employed to achieve balanced exploration of significant search space regions while still achieving near optimum and global points.The ρ is employed as follows:

$$\rho_1 = 2 \times \text{rand} \times \alpha - \alpha \tag{11}$$

$$\alpha = \left|\beta \times \sin\left(\frac{3\pi}{2} + \sin\left(\beta \times \frac{3\pi}{2}\right)\right)\right| \tag{12}$$

$$\beta = \beta_{\min} + (\beta_{\max} - \beta_{\min}) \times \left(1 - \left(\frac{m}{M}\right)^3\right)^2 \tag{13}$$

where; $\beta_{\min}$ is a constant value of 0.2 and βmax is a constant value of 1.2, while m represents the current iteration number, while M represents the total number of iterations. The parameter ρ_1 is responsible to balance the exploration and exploitation based on the sine function. This parameter value changes during optimization iterations, beginning at a large value to facilitate wider variety, then decreasing over the iterations to speed up convergence. The parameter value increases through defined iterations within a range. This increases diversity in solutions, and allows the algorithm to explore multiple solutions to the problem. Described above, the GSR can be calculated as follows:.

$$GSR = randn \times \rho_1 \times \frac{2\Delta x \times x_n}{(x_{\text{worst}} - x_{best} + \varepsilon)} \tag{14}$$

The GBO algorithm uses a random behavior to create a randomized exploration mechanism that includes finding local optima. In Equation (14), it is specified the random offset that deals the difference between the best solution (x_{best}) and a randomly selected solution x_{r1}^m The meaning Δx of the variable is altered by iterations due to the following equation. (17). Additionally another random number (randn) is included to allow for exploration as follows:

$$\Delta x = \text{rand}(1:N) \times |\,\text{step}\,| \tag{15}$$

$$\text{step} = \frac{\left(x_{\text{best}} - x_{r1}^m\right) + \delta}{2} \tag{16}$$

$$\delta = 2 \times \text{rand} \times \left(\left|\frac{x_{r1}^m + x_{r2}^m + x_{r3}^m + x_{r4}^m}{4} - x_n^m\right|\right) \tag{17}$$

where $rand(1 : N)$ is a random vector of N elements in the range of $\in$ [0, 1].

The four randomly selected integers are $r1$, $r2$, $r3$, and $r4$ such that ($r1 \neq r2 \neq r3 \neq r4 \neq n$). *step* represents a phase scale, which is quantified by x_{best} and x_{r1}^m.

To achieve convergence, directional movement is employed in order to converge across the solution field x_n. In order to provide a convenient local search tendency with a major effect on GBO convergence, the term DM uses the best vector from a set of candidate vectors and transfers the current vector (x_n) in the direction of the best vector ($x_{best} - x_n$) and is computed as follows:

$$DM = rand \times \rho_2 \times (x_{best} - x_n) \tag{18}$$

where, rand is a uniform distributed number within range $\in$ [0, 1], a function of two parameters, and ρ_2 is a random

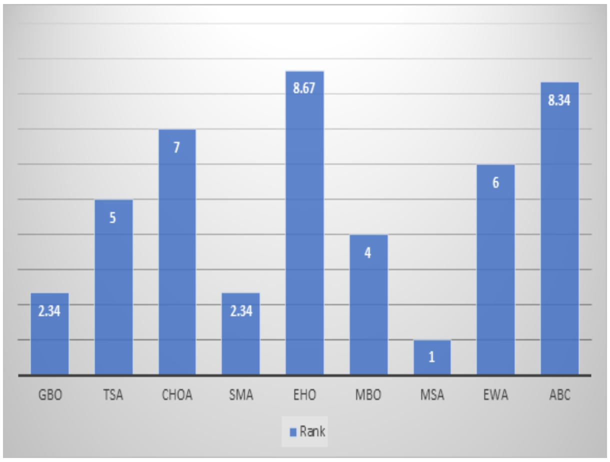

FIGURE 1. Friedman rank test for case 1.

TABLE 2. Parameter settings of GBO and all other competed algorithms.

Algorithms	Parameters setting
Common Settings	Population size: $N = 30$ Maximum iterations: $t_{max} = 1000$ Number of independent runs 30
ABC	$MR = 0.8$, colony size $= 40$, $MCN = 6000$
EWA	$\alpha = 0.98$, $\beta_0 = 0.1$, $\gamma = 0.9$
MSA	$\beta_0 = 1.5$, $S_{max} = 1$
MBO	$S_{max} = 1$, $BAR = 5/12$, $peri = 1.2$
EHO	$\alpha = 0.5$ and $\beta = 0.1$
SMA	$Z = 0.03$
ChOA	a decreases linearly from 2 to 0 (Default)
TSA	P_{min}=1 and P_{max}=4 (Default)
GBO	$FADs = 0.2, Pr = 0.5$

TABLE 3. Test cases.

Case	Description	Test system	Demand (MW)
1	ELD	6	700
			1000
			1200
2	CEED	6	700
			1000
			1200
3	CEED with valve point effect	10	2000

parameter employed to adjust phase size of each vector agent, besides the ρ_2 parameter considers of significant parameters of the GBO exploration process. This is how the ρ_2 parameter is computed:

$$\rho_2 = 2 \times rand \times \alpha - \alpha. \tag{19}$$

Finally, based on the words GSR and DM, Eq. 20 and 21 are modified based on the current vector location (x_n^m).

$$X1_n^m = x_n^m - GSR + DM \tag{20}$$

where, $X1_n^m$ is the modified vector resulting from modifying $X1_n^m$. According to (13), (18), the transformation of $X1_n^m$ can be defined as:.

$$X1_n^m = x_n^m - randn \times \rho_1 \times \frac{2\Delta x \times x_n^m}{\left(yp_n^m - yq_n^m + \varepsilon\right)} + randn \times \rho_2 \times \left(x_{\text{best}} - x_n^m\right) \tag{21}$$

TABLE 4. Statistical results for fitness function of case 1.

Demand (MW)	Algorithm	Best	Average	Worst	SD
700	GBO	8685.549188	24919.83349	262153.549	47322.75726
	TSA	210758.0329	13246569.28	46985262.44	11671889.45
	ChOA	2853186.277	151906508.7	860192164	174654055
	SMA	8706.33765	10228.4272	14496.01421	1435.411121
	EHO	2973884.734	133421320.8	460410140.1	127091384
	MBO	9305.924329	411336.7757	10995801.76	1999680.718
	MSA	8407.954306	8802.07288	9124.64661	166.1269287
	EWA	213216.419	34600771.85	157758914.7	39613470.25
	ABC	895842.5443	50562707.72	193652436.8	41521859.27
1000	GBO	12643.35353	33618.03327	352137.1764	62598.53743
	TSA	528150.5477	21742441.55	67121464.62	20059279.53
	ChOA	4780211.892	86022589.34	225405206.5	68112191.62
	SMA	12334.7719	14060.36459	26255.76917	2744.191485
	EHO	276971.101	46260786.71	282058559.4	58248102.28
	MBO	13447.83996	33788.89758	165898.9357	28621.36091
	MSA	12135.486	12304.2484	12727.55517	115.870386
	EWA	42655.58313	24992802.66	179592008.1	43120730.13
	ABC	1738136.181	37391815.75	137622278	34046433.01
1200	GBO	14905.29219	26346.61414	196633.6921	33442.94929
	TSA	206685.1467	16788728.29	119598234	23703490.08
	ChOA	142680.7814	105558363	350408704.5	94088147.73
	SMA	14928.6678	16517.4336	32838.98118	3534.804878
	EHO	105334722.3	1626274160	7859331183	1845294906
	MBO	15283.91213	724059550.9	21721034896	3965695530
	MSA	14849.22045	14939.65519	15013.28197	43.71791147
	EWA	70678.7321	90693915.36	967534926.7	187625885.6
	ABC	4662407.156	66969975.67	218065260.7	53093537.28

TABLE 5. Best costs of case 1 at various demand setting in $ per hour.

Algorithm	700 MW	1000 MW	1200 MW
GBO	8596.2688	12288.39245	14865.88322
TSA	8797.414047	12375.42954	14933.35395
ChOA	8440.90984	12341.01675	14925.73293
SMA	8688.39929	12279.57551	14907.55555
EHO	9569.329425	14049.35905	16540.28986
MBO	9318.537106	14685.03214	16771.76311
MSA	9837.746036	14243.00936	17382.97919
EWA	9276.574594	14227.62604	17497.59526
ABC	10504.18364	12958.16337	15719.35108

where yp_n^m, yq_n^m are equal to $y_n + \Delta x$, and $y_n - \Delta x$, y_n vector is equal to the average of two vectors: current solution x_n, and z_{n+1}.

$$z_{n+1} = x_n - randn \times \frac{2\Delta x \times x_n}{(x_{\text{worst}} - x_{\text{best}} + \varepsilon)} \tag{22}$$

although x_n is current solution vector, $randn$ is a random solution vector of dimension n, x_{worst} and x_{best} represent best and worst solutions, and Δx is given by equation.(15).

Taking the previous formula and substituting current solution vector x_n^m with new solution vector x_{best} in it, we get current solution vector $X2_n^m$.

$$X2_n^m = x_{\text{best}} - randn \times \rho_1 \times \frac{2\Delta x \times x_n^m}{\left(yp_n^m - yq_n^m + \varepsilon\right)} + randn \times \rho_2 \times \left(x_{r1}^m - x_{r2}^m\right) \tag{23}$$

The GBO algorithm aims to improve discovery and exploitation by making each more profitable than the next using using Eq. (21). To help the global search on the discovery process and to boostEq. (23) is used to boost the exploitation process of the local search. Finally, a new version of the solution is found as follows

$$x_n^{m+1} = r_a \times \left(r_b \times X1_n^m + (1 - r_b) \times X2_n^m\right) + (1 - r_a) \times X3_n^m \tag{24}$$

TABLE 6. Allocation vector at best fitness function using all algorithms for case 1 at 700 MW load demand.

GBO	TSA	ChOA	SMA	EHO	MBO	MSA	EWA	ABC
189.5616586	149.44244	341.3132267	185.7135056	66.44209243	73.62897905	56.83489394	70	301.8958987
101.689967	86.8705366	50	50.51695339	69.56621478	90.6969419	59.94476851	84.53333333	25.12490159
147.8302654	146.154003	113.5773156	298.9869879	83.98385487	99.15531604	71.87398307	112.0322222	174.6556292
89.61369411	55.910796	50	50.00032437	113.4233433	100.2939772	83.1495189	127	171.7820666
98.97821595	155.7067605	72.44884018	79.08102815	116.4156123	150	125.3403864	132	375.4694066
84.96718005	120	83.61343611	50.00012025	262.0815035	200	314.0113024	187	23.02519716

TABLE 7. Allocation vector at best fitness function using all algorithms for case 1 at 1000 MW load demand.

GBO	TSA	ChOA	SMA	EHO	MBO	MSA	EWA	ABC
499.8308274	500	500	479.141796	70.91256905	50	50.21399887	68.00014638	115.3298661
189.2454428	54.91450856	51.09100804	50.75277544	101.3029624	97	65.54577834	74.00006839	168.0112459
161.6041441	137.880252	300	197.539091	127.8586154	100	134.9641294	84.21154656	343.9736182
50.07806391	58.84231778	60.65076508	50.00010544	134.3265124	101	147.1481687	182.9998604	124.4705483
50.24321015	200	56.56901973	185.5525677	175.5065586	179	206.5396705	217.9996438	202.8838002
70.11253675	72.2482933	54.43694594	61.00932017	413.4851986	495.0938002	418.907567	397.9997587	75.79661423

TABLE 8. Allocation vector at best fitness function using all algorithms for case 1 at 1200 MW load demand.

GBO	TSA	ChOA	SMA	EHO	MBO	MSA	EWA	ABC
446.0656723	490.0641928	500	457.4740366	111.5133965	87.80868946	55.0227177	84	373.4243039
136.1821761	192.3807862	200	199.9998079	119.7265688	120	110.6578847	91.27257153	275.1957334
250.6336747	300	300	272.1978148	169.237804	150	151.6459541	164.1219416	304.1841199
122.7149098	97.0175366	76.00801196	55.420549	185.26913	200	166.9030794	193.2239142	285.5176576
197.7869545	69.4396344	93.65269981	199.9933025	263.0320301	300	254.1660547	245.3686708	71.02010523
82.55237513	84.4404838	63.64575047	50.30025955	387.7545655	380.8939961	495.4057489	481.2169083	69.52165244

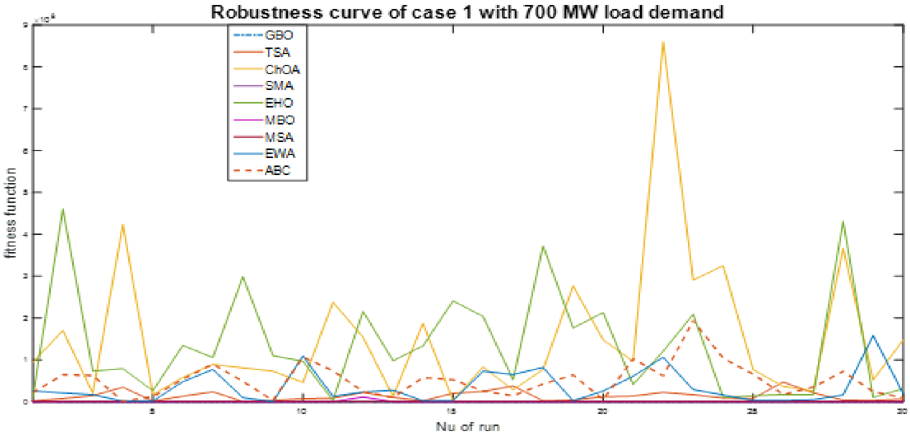

FIGURE 2. Robustness curves for all algorithms for case 1 at 700 MW load demand.

where r_a, and r_b are random numbers determined in range [0, 1], and $X3_n^m$ is defined as:

$$X3_n^m = X_n^{m+1} - \rho 1 \times (X2_n^m - X1_n^m) \tag{25}$$

C. THE LOCAL ESCAPING OPERATOR (LEO) PROCESS

The LEO is implemented to add extra power to an optimization algorithm by helping to solve tricky engineering problems. The LEO operator helps the algorithm to quickly switch out of local optima points to speed up the convergence of the algorithm. To build a new solution with a superior efficiency, the LEO operator targets (X_{LEO}^m), by many solutions (X_{best} best solution, the solutions $X1_n^m, X1_n^m$ are randomly selected from population, $X_{r1}^m, X1_{r2}^m$ randomly generated solutions), so that the current solution can effectively be modified, the procedure is performed based on a scheme that is as follows:.

If rand < pr

$$X_{LEO}^m = \begin{cases} X_n^{m+1} + f_1 \left(u_1 x_{\text{best}} - u_2 x_k^m\right) \\ +f_2 \rho_1 \left(u_3 \left(X2_n^m - X1_n^m\right)\right) \\ + u_2 \left(x_{r1}^m - x_{r2}^m\right)/2, & \text{if rand} < 0.5 \\ X_n^{m+1} + f_1 \left(u_1 x_{\text{best}} - u_2 x_k^m\right) \\ +f_2 \rho_1 \left(u_3 \left(X2_n^m - X1_n^m\right)\right) \\ + u_2 \left(x_{r1}^m - x_{r2}^m\right)/2, & \text{otherwise} \end{cases} \tag{26}$$

End

where pr is a probability value, $pr = 0.5$, the values f_1, and f_2 are uniform distribution random numbers $\in [-1, 1]$,

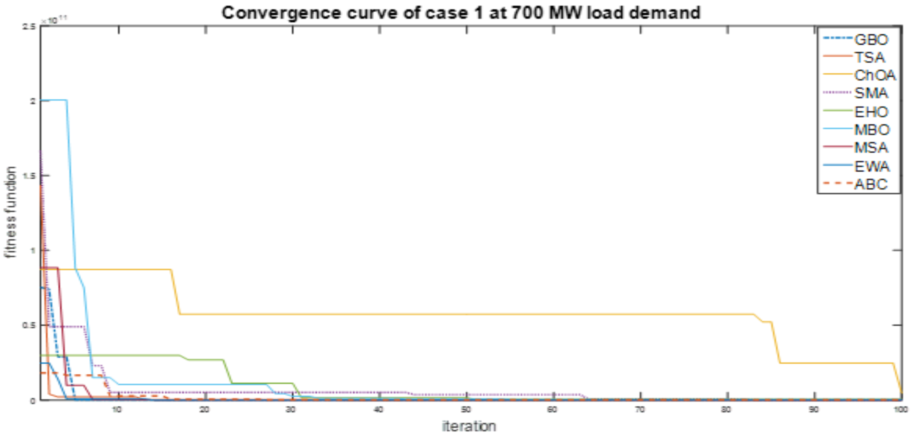

FIGURE 3. Convergence curves for all algorithms for case 1 at 700 MW load demand.

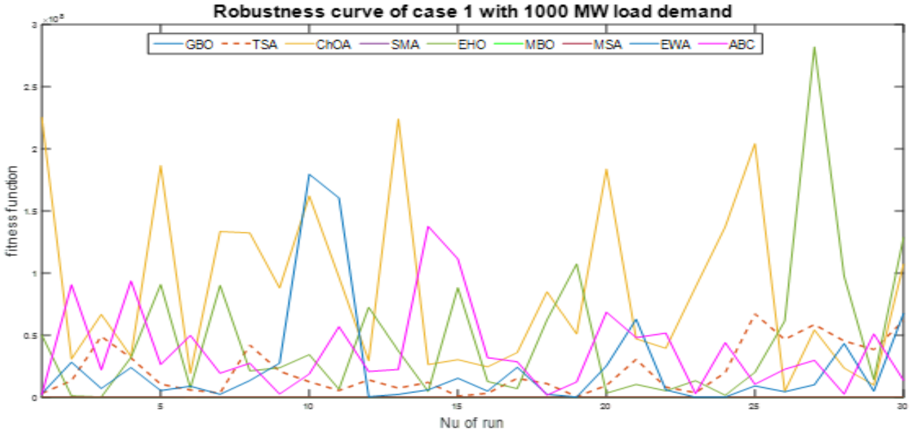

FIGURE 4. Robustness curves for all algorithms for case 1 at 1000 MW load demand.

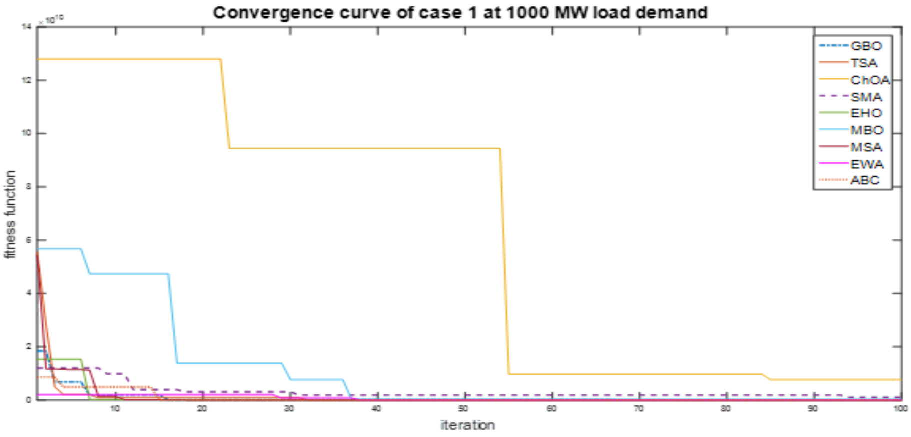

FIGURE 5. Convergence curves for all algorithms for case 1 at 1000 MW load demand.

and u_1, u_2, u_3 are random values generated as following:

$$u_1 = \begin{cases} 2 \times \text{rand} & \text{if } \mu_1 < 0.5 \\ 1 & \text{otherwise} \end{cases} \tag{27}$$

$$u_2 = \begin{cases} \text{rand} & \text{if } \mu_1 < 0.5 \\ 1 & \text{otherwise} \end{cases} \tag{28}$$

$$u_3 = \begin{cases} \text{rand} & \text{if } \mu_1 < 0.5 \\ 1 & \text{otherwise} \end{cases} \tag{29}$$

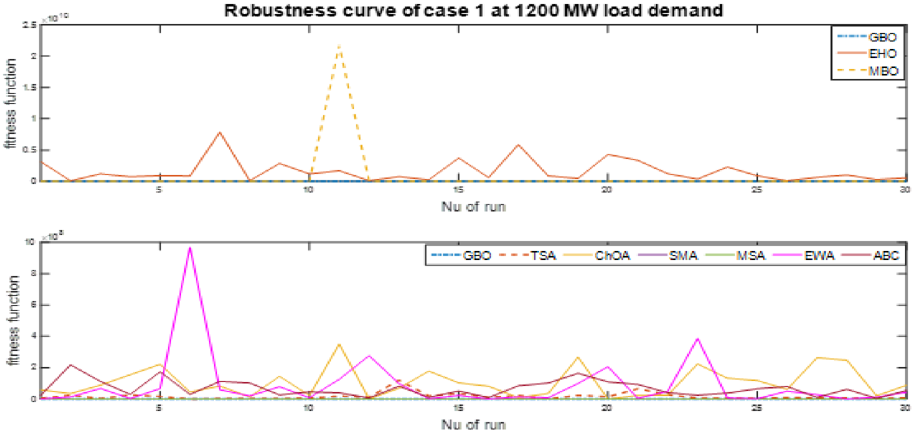

FIGURE 6. Robustness curves for all algorithms for case 1 at 1200 MW load demand.

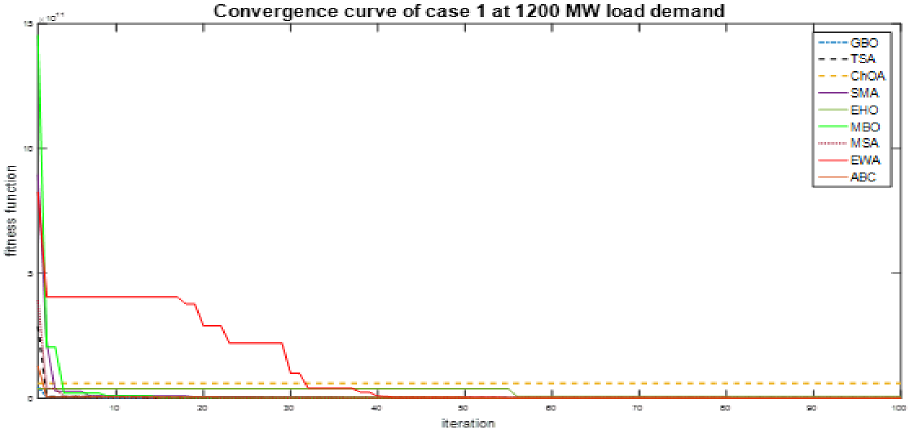

FIGURE 7. Convergence curves for all algorithms for case 1 at 1200 MW load demand.

where *rand* is a random number between 0 and 1, while mu_1 is an arbitrary value from the interval [0, 1].

The previous equations for u_1, u_2, u_3, can be explained as follow:

$$u_1 = L_1 \times 2 \times rand + (1 - L_1) \tag{30}$$

$$u_2 = L_1 \times rand + (1 - L_1) \tag{31}$$

$$u_3 = L_1 \times rand + (1 - L_1) \tag{32}$$

where L_1 is a binary parameter take value 0, 1, such as if parameter $\mu_1 < 0.5$, then value of $L_1 = 1$, otherwise $L_1 = 0$. where the solution x_k^m is generated as follow:

$$x_k^m = \begin{cases} x_{\text{rand}} & \text{if } \mu_2 < 0.5 \\ x_p^m & \text{otherwise} \end{cases} \tag{33}$$

x_{rand} is a random generated solution according to following formula:

$$x_{\text{rand}} = X_{\min} + \text{rand}(0, 1) \times (X_{\max} - X_{\min}) \tag{34}$$

and x_p^m is a random selected solution from population, μ_2 is a random number $\in$ [0, 1]. For more details about GBO see [18].

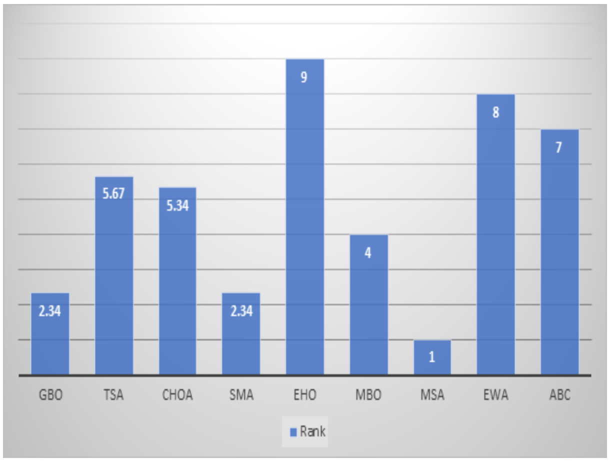

FIGURE 8. Friedman rank test for case 2.

V. EXPERIMENTAL RESULTS AND NUMERICAL ANALYSIS

The performance of GBO on different variants of GBO is compared with other metaheuristics such as; Slime mould algorithm (SMA) [19], Elephant herding optimization (EHO)

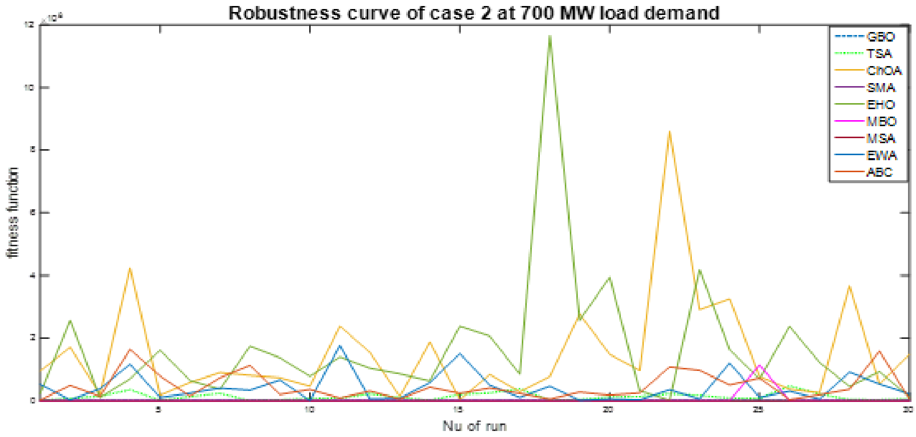

FIGURE 9. Robustness curves for all algorithms for case 2 at 700 MW load demand.

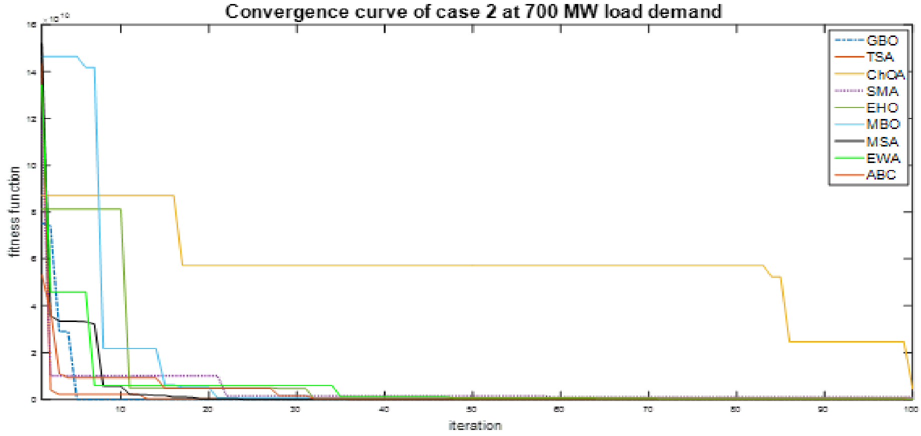

FIGURE 10. Convergence curves for all algorithms for case 2 at 700 MW load demand.

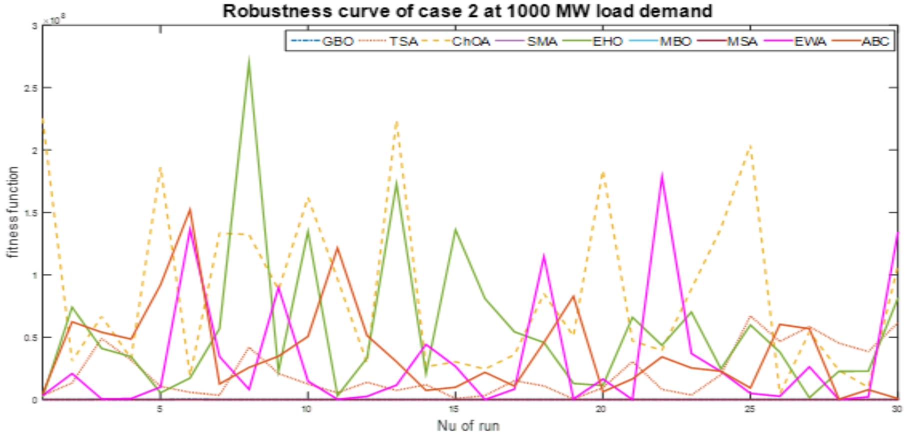

FIGURE 11. Robustness curves for all algorithms for case 2 at 1000 MW load demand.

[20], Monarch butterfly optimization (MBO) [21], Moth search algorithm (MSA) [22], Earthworm optimization algorithm (EWA) [23], Artificial Bee Colony (ABC) Algorithm [24], Tunicate Swarm Algorithm (TSA) [25] and Chimp Optimization Algorithm (ChOA) [26]. The comparison results are reported in this section.

The obtained results of the proposed GBO as well as the considered competitor algorithms to solve the Economic

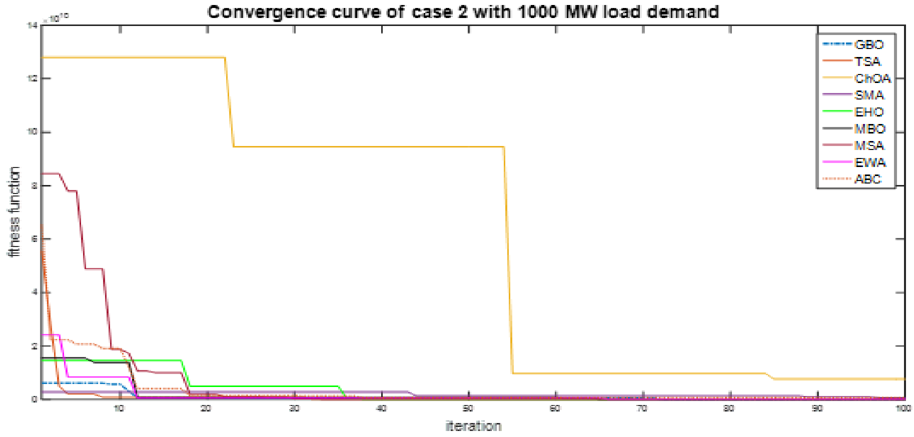

FIGURE 12. Convergence curves for all algorithms for case 2 at 1000 MW load demand.

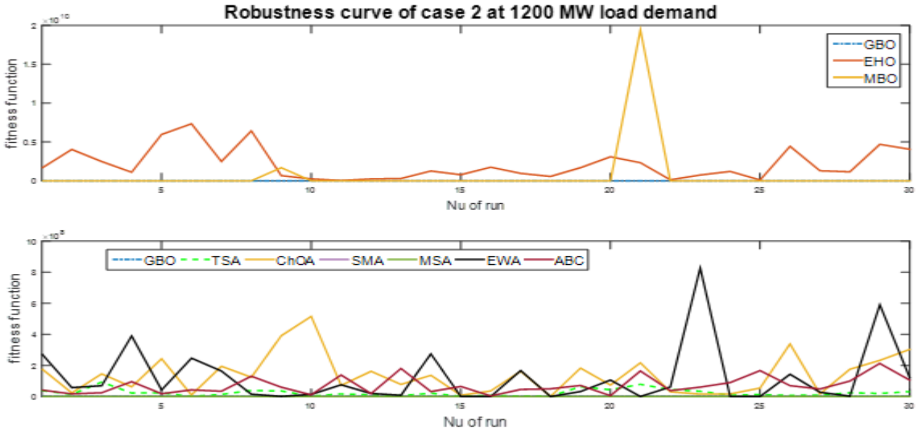

FIGURE 13. Robustness curves for all algorithms for case 2 at 1200 MW load demand.

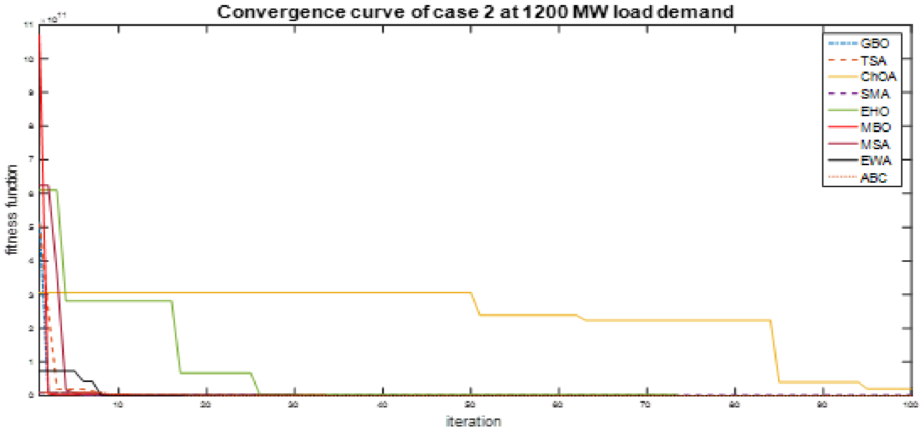

FIGURE 14. Convergence curves for all algorithms for case 2 at 1200 MW load demand.

Load Dispatch (ELD) problems.All the experiments are implemented by coded through MATLAB R2016a and executed on Intel(R) Core i7 CPU- 2.80 GHz with 8 GB RAM and operating system (Windows 10).

A. PARAMETER SETTINGS

For fair comparison, the algorithms are tested with the following settings: population size $N = 30$, maximum iterations $t_{max} = 1000$, and number of independent runs is 30 for all the

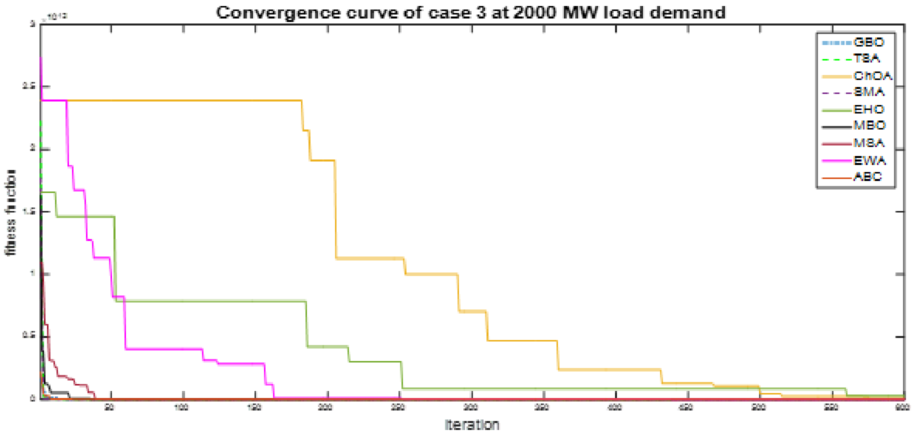

FIGURE 15. Convergence curves for all algorithms for case 3 at 2000 MW load demand.

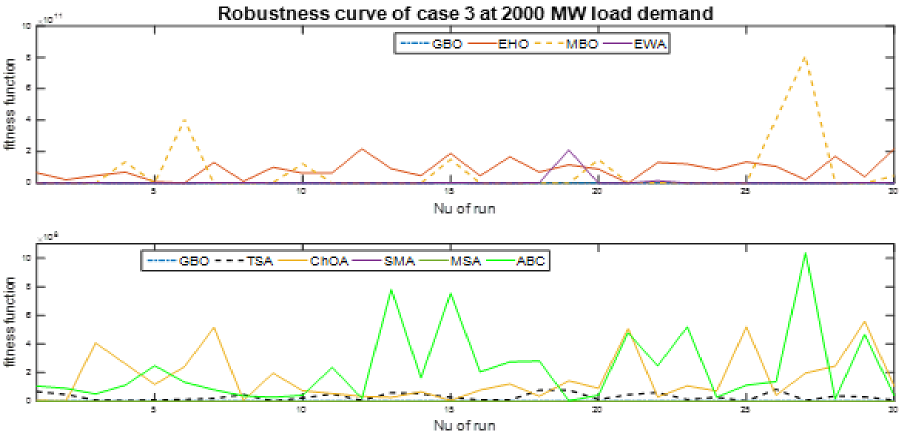

FIGURE 16. Robustness curves for all algorithms for case 3 at 2000 MW load demand.

TABLE 9. Statistical results for fitness function of case 2.

Demand (MW)	Algorithm	Best	Average	Worst	SD
700	GBO	14017.1375	30846.5297	267814.481	47239.5965
	TSA	216788.13	13252541.9	46991553.2	11671864.1
	ChOA	2858544.31	151912122	860197559	174653982
	SMA	13951.6739	16079.3504	18092.9311	1054.53798
	EHO	1844902.97	164473687	1165285104	216677362
	MBO	15143.5719	3852192.32	113785874	20763225.2
	MSA	13549.522	14522.2371	15628.6549	597.038018
	EWA	340877.537	42110221.9	176051262	46198481.3
	ABC	1782940.05	45222544.3	163747197	44115528.8
1000	GBO	22252.1745	43654.5268	361955.166	62544.648
	TSA	538440.142	21752609.8	67131455.5	20059070.3
	ChOA	4790838.24	86032963.1	225415259	68112142.6
	SMA	22039.7803	24264.563	33331.9981	2366.63301
	EHO	1523376.33	55351077	269667029	58057247.8
	MBO	24656.5505	61653.7154	237739.235	45456.4532
	MSA	21635.2641	22262.0805	23689.2616	429.887368
	EWA	35945.2227	31790728	178571263	48104438.3
	ABC	395848.98	38713524.5	152276546	36307752.4
1200	GBO	28256.3199	39401.2656	149954.967	26615.5129
	TSA	1025742.79	24601430	93651200.7	22939102.8
	ChOA	675391.413	140396231	516094443	126893506
	SMA	28365.4891	29494.5948	34556.3202	1328.22808
	EHO	31304760.5	2105542396	7345510175	2021393472
	MBO	28807.6531	703414998	1.9417E+10	3547827656
	MSA	28035.8981	28445.6287	29340.6699	336.625021
	EWA	47305.3256	124648523	827830270	190917552
	ABC	4054724.14	71689028.2	213489243	56300578.6

experiments. Table 2 reported the parameters setting of each algorithm.

TABLE 10. Costs at the best fitness function for case 2 at all various demand.

Algorithm	700 MW		1000 MW		1200 MW	
	Fuel ($ per hour)	emission (lb)	fuel ($ per hour)	Emission (lb)	fuel ($ per hour)	emission(lb)
GBO	8596.2688	6.24E+03	12170.42256	10139.75021	14900.25503	13790.0282
TSA	8797.414047	7002.552521	12375.42954	9261.482867	14877.97606	14962.83278
ChOA	8440.90984	5802.992618	12341.01675	15457.44634	14976.34025	16905.31524
SMA	8463.442192	5164.536366	12211.26945	11771.83064	14932.41298	12709.73616
EHO	9243.25197	9526.058727	13524.17541	23872.4555	17106.28264	52020.11236
MBO	9436.142766	13536.43573	14492.31598	47609.4206	17197.50699	51619.05094
MSA	9745.611902	19188.66611	13840.51253	3.17E+04	17370.46043	52585.57961
EWA	10237.18289	13630.94618	13778.05316	2.56E+04	17148.55275	51219.41855
ABC	8858.421948	14448.08295	12191.25218	12506.13166	15026.75598	17521.29827

B. COMPARISON OF GBO WITH TSA AND CHOA

The comparison results of GBO with TSA and ChOA is reported in this section. The analysis is done for the test cases as shown in Table3. The ELD problem is solved for all the test cases reported in Table 3. Table 4 reports the best, worst, average fitness function obtained by the three aforesaid algorithms. Table 5 reports the best fuel cost for case 1 at various demand setting. The superior performance of GBO is clear from the obtained result. Figure 1 shows the Friedman rank of the three algorithms. It is observed that GBO has obtained the best rank followed by TSA for case 1. Table 6 to Table 8 reports the allocation vector obtained by

TABLE 11. Allocation vector at best fitness function using all algorithms for case 2 at 700 MW load demand.

GBO	TSA	ChOA	SMA	EHO	MBO	MSA	EWA	ABC
189.5616586	149.442441	341.3132267	378.9366313	79.54701954	68	52.05532709	72.00000805	244.1834959
101.689967	86.8705366	50	75.37433756	90.23498241	99.99842184	67.85545034	89.05319703	39.35143406
147.8302654	146.1540035	113.5773156	80.21206995	98.13130859	100.0014739	89.10876811	128	327.7581268
89.61369411	55.9107966	50	50.06590784	101.4400667	100.0163097	96.5103462	132.2194799	74.63261663
98.97821595	155.7067605	72.44884018	55.95878995	165.099417	110.9367445	108.0121203	148.9834814	37.74249925
84.96718005	120	83.61343611	69.66723407	178.5420941	235.3278444	297.7656245	216	70.17299783

TABLE 12. Allocation vector at best fitness function using all algorithms for case 2 at 1000 MW load demand.

GBO	TSA	ChOA	SMA	EHO	MBO	MSA	EWA	ABC
461.570441	500	500	434.4797434	97.09276335	82	67.3961078	69.07238424	380.6523155
123.2116856	54.91450856	51.09100804	50.35862217	101.8463667	85	83.58519529	87.14055458	89.76147203
176.6255197	137.8802528	300	217.8071943	133.4393152	100	157.1225566	127.6017913	225.350198
78.60560553	58.84231778	60.65076508	105.8636912	149.3393125	120	160.1194502	178.9058122	144.2181793
98.92455763	200	56.56901973	139.1466943	236.6763081	148	189.8577173	248.7546264	101.8788624
83.35778802	72.248293	54.43694594	76.44769788	306.9697441	487.5702429	366.3652719	314.9569037	82.37420716

TABLE 13. Allocation vector at best fitness function using all algorithms for case 2 at 1200 MW load demand.

GBO	TSA	ChOA	SMA	EHO	MBO	MSA	EWA	ABC
459.0589771	500	500	499.488136	111.2833768	67.550198	75.67767094	78.20754461	413.2126295
198.6849562	200	66.92036408	188.298177	117.8809823	120	98.87255155	88.07865235	235.2901806
251.5009682	268.4404202	273.6642867	206.7031703	134.9003054	166.2354635	138.8659866	173.7696489	267.0830323
54.82964292	89.67677903	150	53.47086713	146.9153126	200	142.0625883	182	46.04408805
199.7553995	122.6330863	180.0481543	198.3139709	225.0190675	200	278.5784557	231.0071992	111.6957993
71.47096762	52.51861086	65.0296636	88.25018247	498.2686034	481.3641117	499.9933777	480.749139	162.6237884

TABLE 14. Statistical results for fitness function of case 3.

Demand (MW)	Algorithm	Best cost	Average cost	Worst cost	SD
2000	GBO	219799.1059	238094.2967	313512.3747	20453.30556
	TSA	526049.3651	30314739.91	82394729.03	25862437.39
	ChOA	14724151.54	161954969.3	574814303.1	165602544.3
	SMA	221048.2919	233852.8048	255144.2964	8369.574912
	EHO	2708607.303	88773458687	2.19E+11	62069182459
	MBO	256541.9032	74521649305	8.09E+11	1.76E+11
	MSA	221757.381	228824.0022	236631.2423	3869.245477
	EWA	16986001.28	8655562888	2.11E+11	38288367019
	ABC	2689228.652	225008720.1	1036718216	258946901.8

TABLE 15. Costs at the best fitness function for case 3 at 2000 MW load demand.

Demand (MW)	Algorithm	Fuel cost with valve ($ per hour)	Fuel cost without valve ($ per hour)	Emission (lb)
2000	GBO	113454.2075	113192.2176	4123.37858
	TSA	115635.895	115429.6047	4083.769908
	ChOA	116718.3433	116499.79	4128.714921
	SMA	113420.5547	113211.8298	4364.964345
	EHO	114819.3837	114622.1343	4431.78109
	MBO	115876.0282	115698.5327	4233.946211
	MSA	114456.837	114266.9468	4161.150569
	EWA	116486.9316	116292.6838	4154.508192
	ABC	154779.744	154528.2801	13149.51024

the aforesaid algorithm for demand 700 MW, 1000 MW, and 1200 MW respectively. Figure 2, Figure 4, Figure 6 shows the robustness curve of case 1 for demand 700 MW, 1000 MW, and 1200 MW respectively. It is observed that GBO produces uniform solutions for different runs as compared to other algorithms. Figure 3, Figure 5, Figure 7 shows the convergence curve for case 1 in case of different demand.

The best, worst, average value of fitness function for case 2 is reported in Table 9. It is observed that GBO performs well as compared to other algorithms. Table 10 reports the fuel cost and emission for case 2 obtained by the three algorithms. It is observed that GBO maintains a good balance between cost and emission. Figure 8 shows the Friedman rank obtained for case 2. It is observed that GBO has obtained the best rank followed by ChOA. Table 11, Table 13, and Table 13 reports the allocation vector for demand 700 MW, 1000 MW, and 1200 MW respectively. Further, the convergence and robustness of GBO is compared with TSA and ChOA. Figure 9, Figure 11, Figure 13 shows the robustness curve of case 2 with load demand 700 MW, 1000 MW, and 1200 MW respectively. Figure 10, Figure 12, Figure 14 shows the convergence curve of case 2 with load demand 700 MW, 1000 MW, and 1200 MW respectively. It is observed that in case of GBO a faster convergence towards the optima is favoured.

The best, worst, average value of fitness function for case 3 is as shown in Table 14. The superior performance of GBO is prominent from the results reported in Table 14. Table 15 reports the fuel cost and emission for case 3 obtained by the three algorithms with and without valve point effect. Table 16 reports the allocation vector for case 3. Further, the convergence and robustness of GBO is compared with TSA and ChOA. Figure 15 and Figure 16 shows the convergence and robustness curve for case 3 respectively.

C. COMPARISON OF GBO WITH CTO AND ITS VARIANTS

The performance of GBO is compared with CTO and its variants in solving case 3 of Table 3. The results of CTO and its variants are taken from ref [27] and the ELD problem is solved by GBO with the same general parameter setting as

TABLE 16. Allocation vector at best fitness function using all algorithms for case 3 at 2000 MW load demand.

GBO	TSA	ChOA	SMA	EHO	MBO	MSA	EWA	ABC
53.42746261	33.698925	26.89673845	54.86291659	16.96206125	20	28.99571835	28	102.3078875
77.15682841	57.2314775	47.20099907	71.91549278	28.82349317	40.81027727	79.99999985	34	4.023651967
113.5029422	77.7232758	74.61307679	119.9897713	119.808397	50	101.5859835	87	122.1298362
83.323102	37.2058037	64.49420468	129.9999199	125.8574639	97	111.9962796	102.7166666	198.7076242
136.8829177	160	160	159.2858516	136.9372426	120	121.9511674	145.0000001	136.1017189
165.8333622	213.980094	240	83.50735483	184.514659	240	204.2708322	230	215.4571441
220.0906134	300	191.6427769	299.4084763	289.7231679	267	286.639065	294.999999	1082.577336
303.3529174	294.4096915	340	327.189747	320.1530829	340	300	305	184.9053026
466.2480961	470	470	463.5176487	424.8566276	440	376.9793254	430	461.0022716
465.0401509	440.6356112	470	373.6755744	435.0080636	470	469.9939046	436	4.691590207

TABLE 17. Comparison of GBO with CTO and its variants.

Algorithm	Cost ($ per hour)	Emission (lb)
CTO	113385	4007.2
ACTO	134100	4002.52
GBO	113192.21757	4123.37858

TABLE 18. Comparison of GBO with DE, PDE and MODE.

Algorithm	Cost ($ per hour)	Emission (lb)
DE	111500	4581
MODE	113480	4124.90
PDE	113510	4111.40
GBO	113192.21757	4123.37858

TABLE 19. Comparison of GBO with NSGA-II and SPEA-2.

Algorithm	Cost ($ per hour)	Emission (lb)
NSGA II	113540	4130.20
SPEA 2	113520	4109.10
GBO	113192.21757	4123.37858

in ref [27]. The comparison results are as shown in Table 17. It is observed GBO is the second-best performing algorithm.

D. COMPARISON OF GBO WITH DE AND ITS VARIANTS

The performance of GBO is compared with DE and its variants in solving case 3 of Table 3. The results of DE and its variants are taken from ref [76] and the ELD problem is solved by GBO with the same general parameter setting as in ref [76]. The comparison results are as shown in Table 18. It is observed that GBO performs better than PDE and MODE in terms of cost. And, GBO performs better than DE and MODE in terms of emission.

E. COMPARISON OF GBO WITH NSGA-II AND SPEA-2

The performance of GBO is compared with NSGA II and SPEA 2 in solving case 3 of Table 3. The results of NSGA-II and SPEA-2 are taken from ref [76] and the ELD problem is solved by GBO with the same general parameter setting as in ref [76]. The comparison results are as shown in Table 19. It is observed that GBO performs better than SPEA 2 and NSGA II in terms of cost. Also, GBO performs better than NSGA II in terms of emission.

VI. CONCLUSION AND FUTURE WORK

ELD is one of the complex problems of power system. This work tests the performance of gradient based optimization (GBO) in solving different variants of ELD such as ELD with losses, CEED, and CEED considering valve point effect. GBO is a metaheuristic inspired by the gradient based Newton method involving Gradient search rule (GSR) and local escaping operator (GEO). GBO has good balance between exploration and exploitation. Also, the possibility of getting stuck in local optima and premature convergence is rare in GBO. The performance of GBO is compared with eight other metaheuristic algorithms such as Slime mould algorithm (SMA), Elephant herding optimization (EHO), Monarch butterfly optimization (MBO), Moth search algorithm (MSA), Earthworm optimization algorithm (EWA), Artificial Bee Colony (ABC) Algorithm, Tunicate Swarm Algorithm (TSA) and Chimp Optimization Algorithm (ChOA). In addition, GBO is evaluated against other existing studies in the literature such as Differential evolution (DE), Class Topper Optimization (CTO), Non Dominated Sorting GA (NSGA-II), and Strength pareto evolutionary algorithm 2 (SPEA-2) for different demands. It is observed that GBO performs relatively well as compared to the aforesaid algorithms. Further, it is seen that GBO has good balance between exploration and exploitation and the possibility of getting stuck in local optima and premature convergence is rare in GBO. Our future work will focus on:

- Performance of GBO on dynamic ELD considering renewable resources
- Performance of GBO on other power system problems such as unit commitment, optimal load flow
- Hybridization of GBO with other metaheuristics for solving power system optimization problems

In the future studies, the GBO can be a good candidate to solve the problems in renewable energy for instance solar cell systems. Due to the great performance of the GBO, future work may extend to solve various single and multi-objective optimization problems in different fields.

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest.

CREDIT AUTHOR STATEMENT

All authors contributed equally to this paper, where; Sanchari Deb: Formal analysis, Resources, Writing - review & editing.Diaa Salama AbdElminaam: Resources,Methodology, Conceptualization, Data curation, Writing - review & editing. Mokhtar Said: Software, Formal analysis, Resources, Writing - original draft.Essam H. Houssein: Supervision, Methodology, Conceptualization, Formal analysis, Writing - review & editing. All authors read and approved the final paper.

REFERENCES

[1] A. A. K. Ismaeel, E. H. Houssein, D. Oliva, and M. Said, "Gradient-based optimizer for parameter extraction in photovoltaic models," *IEEE Access*, vol. 9, pp. 13403–13416, 2021.

[2] M. H. Hassan, E. H. Houssein, M. A. Mahdy, and S. Kamel, "An improved manta ray foraging optimizer for cost-effective emission dispatch problems," *Eng. Appl. Artif. Intell.*, vol. 100, Apr. 2021, Art. no. 104155.

[3] A. Sheta, H. Faris, M. Braik, and S. Mirjalili, "Nature-inspired metaheuristics search algorithms for solving the economic load dispatch problem of power system: A comparison study," in *Applied Nature-Inspired Computing: Algorithms and Case Studies*. Singapore: Springer, 2020, pp. 199–230.

[4] Y. A. Gherbi, H. Bouzeboudja, and F. Z. Gherbi, "The combined economic environmental dispatch using new hybrid Metaheuristic," *Energy*, vol. 115, pp. 468–477, Nov. 2016.

[5] A. LaTorre, S. Muelas, and J.-M. Peña, "A comprehensive comparison of large scale global optimizers," *Inf. Sci.*, vol. 316, pp. 517–549, Sep. 2015.

[6] M. N. A. Wahab, S. Nefti-Meziani, and A. Atyabi, "A comprehensive review of swarm optimization algorithms," *PLoS ONE*, vol. 10, no. 5, May 2015, Art. no. e0122827.

[7] E. H. Houssein, Y. Mina, and E. Aboul, "Nature-inspired algorithms: A comprehensive review," in *Hybrid Computational Intelligence: Research and Applications*. Boca Raton, FL, USA: CRC Press, 2019, p. 1.

[8] G.-G. Wang, A. H. Gandomi, A. H. Alavi, and D. Gong, "A comprehensive review of krill herd algorithm: Variants, hybrids and applications," *Artif. Intell. Rev.*, vol. 51, no. 1, pp. 119–148, Jan. 2019.

[9] E. H. Houssein, M. R. Saad, F. A. Hashim, H. Shaban, and M. Hassaballah, "Lévy flight distribution: A new Metaheuristic algorithm for solving engineering optimization problems," *Eng. Appl. Artif. Intell.*, vol. 94, Sep. 2020, Art. no. 103731.

[10] F. A. Hashim, E. H. Houssein, M. S. Mabrouk, W. Al-Atabany, and S. Mirjalili, "Henry gas solubility optimization: A novel physics-based algorithm," *Future Gener. Comput. Syst.*, vol. 101, pp. 646–667, Dec. 2019.

[11] F. A. Hashim, K. Hussain, E. H. Houssein, M. S. Mabrouk, and W. Al-Atabany, "Archimedes optimization algorithm: A new metaheuristic algorithm for solving optimization problems," *Appl. Intell.*, vol. 51, pp. 1531–1551, Sep. 2020.

[12] E. H. Houssein, A. G. Gad, Y. M. Wazery, and P. N. Suganthan, "Task scheduling in cloud computing based on meta-heuristics: Review, taxonomy, open challenges, and future trends," *Swarm Evol. Comput.*, vol. 62, Apr. 2021, Art. no. 100841.

[13] E. H. Houssein, M. A. Mahdy, M. G. Eldin, D. Shebl, W. M. Mohamed, and M. Abdel-Aty, "Optimizing quantum cloning circuit parameters based on adaptive guided differential evolution algorithm," *J. Adv. Res.*, early access, Oct. 17, 2020, doi: 10.1016/j.jare.2020.10.001.

[14] J. N. Kuk, R. A. Gonçalves, L. M. Pavelski, S. M. G. S. Venske, C. P. D. Almeida, and A. T. R. Pozo, "An empirical analysis of constraint handling on evolutionary multi-objective algorithms for the environmental/economic load dispatch problem," *Expert Syst. Appl.*, vol. 165, Mar. 2021, Art. no. 113774.

[15] D. Singh and J. S. Dhillon, "Ameliorated grey wolf optimization for economic load dispatch problem," *Energy*, vol. 169, pp. 398–419, Feb. 2019.

[16] F. Mohammadi and H. Abdi, "A modified crow search algorithm (MCSA) for solving economic load dispatch problem," *Appl. Soft Comput.*, vol. 71, pp. 51–65, Oct. 2018.

[17] S. H. A. Kaboli and A. K. Alqallaf, "Solving non-convex economic load dispatch problem via artificial cooperative search algorithm," *Expert Syst. Appl.*, vol. 128, pp. 14–27, Aug. 2019.

[18] I. Ahmadianfar, O. Bozorg-Haddad, and X. Chu, "Gradient-based optimizer: A new Metaheuristic optimization algorithm," *Inf. Sci.*, vol. 540, pp. 131–159, Nov. 2020.

[19] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime mould algorithm: A new method for stochastic optimization," *Future Gener. Comput. Syst.*, vol. 111, pp. 300–323, Oct. 2020.

[20] G.-G. Wang, S. Deb, and L. D. S. Coelho, "Elephant herding optimization," in *Proc. 3rd Int. Symp. Comput. Bus. Intell. (ISCBI)*, Dec. 2015, pp. 1–5.

[21] G.-G. Wang, S. Deb, and Z. Cui, "Monarch butterfly optimization," *Neural Comput. Appl.*, vol. 31, no. 7, pp. 1995–2014, 2019.

[22] G.-G. Wang, "Moth search algorithm: A bio-inspired Metaheuristic algorithm for global optimization problems," *Memetic Comput.*, vol. 10, no. 2, pp. 151–164, Jun. 2018.

[23] G. G. Wang, S. Deb, and L. D. S. Coelho, "Earthworm optimisation algorithm: A bio-inspired Metaheuristic algorithm for global optimisation problems," *Int. J. Bio-Inspired Comput.*, vol. 12, no. 1, p. 1, 2018.

[24] D. Karaboga and B. Basturk, "An artificial bee colony (ABC) algorithm for numeric function optimization," in *Proc. IEEE Swarm Intell. Symp.*, 2006, pp. 181–184.

[25] S. Kaur, L. K. Awasthi, A. L. Sangal, and G. Dhiman, "Tunicate swarm algorithm: A new bio-inspired based metaheuristic paradigm for global optimization," *Eng. Appl. Artif. Intell.*, vol. 90, Apr. 2020, Art. no. 103541.

[26] M. Khishe and M. R. Mosavi, "Chimp optimization algorithm," *Expert Syst. Appl.*, vol. 149, Jul. 2020, Art. no. 113338.

[27] A. Srivastava and D. K. Das, "A new aggrandized class topper optimization algorithm to solve economic load dispatch problem in a power system," *IEEE Trans. Cybern.*, early access, Nov. 6, 2020, doi: 10.1109/TCYB.2020.3024607.

[28] D. Santra, A. Mukherjee, K. Sarker, and S. Mondal, "Dynamic economic dispatch using hybrid metaheuristics," *J. Electr. Syst. Inf. Technol.*, vol. 7, no. 1, p. 3, Dec. 2020.

[29] F. Tariq, S. Alelyani, G. Abbas, A. Qahmash, and M. R. Hussain, "Solving renewables-integrated economic load dispatch problem by variant of metaheuristic bat-inspired algorithm," *Energies*, vol. 13, no. 23, p. 6225, Nov. 2020.

[30] F. X. Rugema, G. Yan, S. Mugemanyi, Q. Jia, S. Zhang, and C. Bananeza, "A cauchy-Gaussian quantum-behaved bat algorithm applied to solve the economic load dispatch problem," *IEEE Access*, vol. 9, pp. 3207–3228, 2021.

[31] R. Keswani, H. K. Verma, and S. K. Sharma, "Dynamic economic load dispatch considering renewable energy sources using multiswarm statistical particle swarm optimization," in *Proc. IEEE Int. Conf. Comput., Power Commun. Technol. (GUCON)*, Oct. 2020, pp. 405–410.

[32] P. Vasant, F. P. Mahdi, J. A. Marmolejo-Saucedo, I. Litvinchev, R. R. Aguilar, and J. Watada, "Quantum-behaved bat algorithm for solving the economic load dispatch problem considering a valve-point effect," *Int. J. Appl. Metaheuristic Comput.*, vol. 11, no. 3, pp. 41–57, Jul. 2020.

[33] A. Srivastava and S. Singh, "Implementation of ant colony optimization in economic load dispatch problem," in *Proc. 7th Int. Conf. Signal Process. Integr. Netw. (SPIN)*, Feb. 2020, pp. 1018–1024.

[34] Y. A. Gherbi, F. Lakdja, H. Bouzeboudja, and F. Z. Gherbi, "Hybridization of two metaheuristics for solving the combined economic and emission dispatch problem," *Neural Comput. Appl.*, vol. 31, no. 12, pp. 8547–8559, Dec. 2019.

[35] E. S. Ali and S. M. Abd Elazim, "Mine blast algorithm for environmental economic load dispatch with valve loading effect," *Neural Comput. Appl.*, vol. 30, no. 1, pp. 261–270, Jul. 2018.

[36] W. T. Elsayed, Y. G. Hegazy, M. S. El-bages, and F. M. Bendary, "Improved random drift particle swarm optimization with self-adaptive mechanism for solving the power economic dispatch problem," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1017–1026, Jun. 2017.

[37] G. Dhiman, S. Guo, and S. Kaur, "ED-SHO: A framework for solving nonlinear economic load power dispatch problem using spotted hyena optimizer," *Mod. Phys. Lett. A*, vol. 33, no. 40, Dec. 2018, Art. no. 1850239.

[38] A. Kumar, V. Bhalla, P. Kumar, T. Bhardwaj, and N. Jangir, "Whale optimization algorithm for constrained economic load dispatch problems—A cost optimization," in *Ambient Communications and Computer Systems*. Singapore: Springer, 2018, pp. 353–366.

[39] A. Bhadoria, V. K. Kamboj, M. Sharma, and S. Bath, "A solution to non-convex/convex and dynamic economic load dispatch problem using moth flame optimizer," *INAE Lett.*, vol. 3, no. 2, pp. 65–86, 2018.

[40] M. A. Al-Betar, M. A. Awadallah, and M. M. Krishan, "A non-convex economic load dispatch problem with valve loading effect using a hybrid grey wolf optimizer," *Neural Comput. Appl.*, vol. 32, no. 16, pp. 12127–12154, Aug. 2020.

[41] T. Yalcinoz and K. Rudion, "Economic load dispatch using an improved particle swarm optimization based on functional constriction factor and functional inertia weight," in *Proc. IEEE Int. Conf. Environ. Electr. Eng. IEEE Ind. Commercial Power Syst. Eur. (EEEIC / I CPS Europe)*, Jun. 2019, pp. 1–5.

[42] A. Yadav and N. Kumar, "Application of artificial electric field algorithm for economic load dispatch problem," in *Proc. Int. Conf. Soft Comput. Pattern Recognit.* Cham, Switzerland: Springer, 2019, pp. 71–79.

[43] V. K. Kamboj, A. Bhadoria, and S. K. Bath, "Solution of non-convex economic load dispatch problem for small-scale power systems using ant lion optimizer," *Neural Comput. Appl.*, vol. 28, no. 8, pp. 2181–2192, Aug. 2017.

[44] P. Zakian and A. Kaveh, "Economic dispatch of power systems using an adaptive charged system search algorithm," *Appl. Soft Comput.*, vol. 73, pp. 607–622, Dec. 2018.

[45] A. Lin and W. Sun, "Multi-leader comprehensive learning particle swarm optimization with adaptive mutation for economic load dispatch problems," *Energies*, vol. 12, no. 1, p. 116, Dec. 2018.

[46] Q. Qin, S. Cheng, X. Chu, X. Lei, and Y. Shi, "Solving non-convex/non-smooth economic load dispatch problems via an enhanced particle swarm optimization," *Appl. Soft Comput.*, vol. 59, pp. 229–242, Oct. 2017.

[47] K. Kadali, R. Loganathan, M. Veerasamy, and V. Jawalker, "Cost-effective dispatch using grey wolf optimization algorithm: Solution with diverse load pattern," in *Proc. Int. Conf. Syst., Comput., Autom. Netw. (ICSCAN)*, Jul. 2020, pp. 1–5.

[48] G. Pradhan and P. D. Dewangan, "Solving optimal load dispatch problem using enhanced BAT optimization algorithm," in *Proc. Innov. Power Adv. Comput. Technol. (I-PACT)*, Apr. 2017, pp. 1–6.

[49] F. P. Mahdi, P. Vasant, M. Abdullah-Al-Wadud, V. Kallimani, and J. Watada, "Quantum-behaved bat algorithm for many-objective combined economic emission dispatch problem using cubic criterion function," *Neural Comput. Appl.*, vol. 31, no. 10, pp. 5857–5869, Oct. 2019.

[50] I. Marouani, A. Boudjemline, T. Guesmi, and H. H. Abdallah, "A modified artificial bee colony for the non-smooth dynamic economic/environmental dispatch," *Eng., Technol. Appl. Sci. Res.*, vol. 8, no. 5, pp. 3321–3328, Oct. 2018.

[51] C. A. O. D. Freitas, R. C. L. D. Oliveira, D. J. A. D. Silva, J. C. Leite, and J. D. A. Brito, "Solution to economic–emission load dispatch by cultural algorithm combined with local search: Case study," *IEEE Access*, vol. 6, pp. 64023–64040, 2018.

[52] Y. Yang, B. Wei, H. Liu, Y. Zhang, J. Zhao, and E. Manla, "Chaos firefly algorithm with self-adaptation mutation mechanism for solving large-scale economic dispatch with valve-point effects and multiple fuel options," *IEEE Access*, vol. 6, pp. 45907–45922, 2018.

[53] B. M. Alshammari, "Teaching-Learning-Based optimization algorithm for the combined dynamic economic environmental dispatch problem," *Eng., Technol. Appl. Sci. Res.*, vol. 10, no. 6, pp. 6432–6437, Dec. 2020.

[54] A. Dihem, A. Salhi, D. Naimi, and A. Bensalem, "Solving smooth and non-smooth economic dispatch using water cycle algorithm," in *Proc. 5th Int. Conf. Electr. Eng.-Boumerdes (ICEE-B)*, Oct. 2017, pp. 1–6.

[55] H. Liang, Y. Liu, Y. Shen, F. Li, and Y. Man, "A hybrid bat algorithm for economic dispatch with random wind power," *IEEE Trans. Power Syst.*, vol. 33, no. 5, pp. 5052–5061, Sep. 2018.

[56] V. Chaudhary, H. M. Dubey, M. Pandit, and J. C. Bansal, "Multi-area economic dispatch with stochastic wind power using salp swarm algorithm," *Array*, vol. 8, Dec. 2020, Art. no. 100044.

[57] J. Lin and Z.-J. Wang, "Multi-area economic dispatch using an improved stochastic fractal search algorithm," *Energy*, vol. 166, pp. 47–58, Jan. 2019.

[58] M. A. Ali, H. M. Dubey, and M. Pandit, "Moth-flame optimization for multi area economic dispatch: A novel heuristic paradigm," in *Proc. Int. Conf. Energy, Commun., Data Analytics Soft Comput. (ICECDS)*, Aug. 2017, pp. 1068–1073.

[59] P. M. Joshi and H. K. Verma, "An improved TLBO based economic dispatch of power generation through distributed energy resources considering environmental constraints," *Sustain. Energy, Grids Netw.*, vol. 18, Jun. 2019, Art. no. 100207.

[60] J. Paramguru and S. K. Barik, "Modified grey wolf optimization applied to non-convex economic load dispatch in current power system scenario," in *Proc. Int. Conf. Recent Innov. Electr., Electron. Commun. Eng. (ICRIEECE)*, Jul. 2018, pp. 2704–2709.

[61] A. E. Fergougui, A. A. Ladjici, A. Benseddik, and Y. Amrane, "Dynamic economic dispatch using genetic and particle swarm optimization algorithm," in *Proc. 5th Int. Conf. Control, Decis. Inf. Technol. (CoDIT)*, Apr. 2018, pp. 1001–1005.

[62] M. Kheshti, X. Kang, Z. Bie, Z. Jiao, and X. Wang, "An effective lightning flash algorithm solution to large scale non-convex economic dispatch with valve-point and multiple fuel options on generation units," *Energy*, vol. 129, pp. 1–15, Jun. 2017.

[63] Y. Sonmez, H. T. Kahraman, M. K. Dosoglu, U. Guvenc, and S. Duman, "Symbiotic organisms search algorithm for dynamic economic dispatch with valve-point effects," *J. Experim. Theor. Artif. Intell.*, vol. 29, no. 3, pp. 495–515, May 2017.

[64] V. K. Kamboj, S. K. Bath, and J. S. Dhillon, "Solution of non-convex economic load dispatch problem using grey wolf optimizer," *Neural Comput. Appl.*, vol. 27, no. 5, pp. 1301–1316, Jul. 2016.

[65] V. Suresh, S. Sreejith, S. K. Sudabattula, and V. K. Kamboj, "Demand response-integrated economic dispatch incorporating renewable energy sources using ameliorated dragonfly algorithm," *Electr. Eng.*, vol. 101, no. 2, pp. 421–442, Jun. 2019.

[66] B. R. Adarsh, T. Raghunathan, T. Jayabarathi, and X.-S. Yang, "Economic dispatch using chaotic bat algorithm," *Energy*, vol. 96, pp. 666–675, Feb. 2016.

[67] W. T. Elsayed, Y. G. Hegazy, F. M. Bendary, and M. S. El-bages, "Modified social spider algorithm for solving the economic dispatch problem," *Eng. Sci. Technol., Int. J.*, vol. 19, no. 4, pp. 1672–1681, Dec. 2016.

[68] A. Y. Abdelaziz, E. S. Ali, and S. M. A. Elazim, "Implementation of flower pollination algorithm for solving economic load dispatch and combined economic emission dispatch problems in power systems," *Energy*, vol. 101, pp. 506–518, Apr. 2016.

[69] I. N. Trivedi, A. Kumar, A. H. Ranpariya, and P. Jangir, "Economic load dispatch problem with ramp rate limits and prohibited operating zones solve using levy flight moth-flame optimizer," in *Proc. Int. Conf. Energy Efficient Technol. Sustainability (ICEETS)*, Apr. 2016, pp. 442–447.

[70] R. M. Rizk-Allah, R. A. El-Sehiemy, and G.-G. Wang, "A novel parallel hurricane optimization algorithm for secure emission/economic load dispatch solution," *Appl. Soft Comput.*, vol. 63, pp. 206–222, Feb. 2018.

[71] D. Zou, S. Li, G.-G. Wang, Z. Li, and H. Ouyang, "An improved differential evolution algorithm for the economic load dispatch problems with or without valve-point effects," *Appl. Energy*, vol. 181, pp. 375–390, Nov. 2016.

[72] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, Apr. 1997.

[73] J.-T. Yu, C.-H. Kim, A. Wadood, T. Khurshaid, and S.-B. Rhee, "Jaya algorithm with self-adaptive multi-population and Lévy flights for solving economic load dispatch problems," *IEEE Access*, vol. 7, pp. 21372–21384, 2019.

[74] M. J. Mokarram, J. P. S. Catalao, J. Aghaei, M. Shafie-khah, and T. Niknam, "Hybrid optimization algorithm to solve the nonconvex multiarea economic dispatch problem," *IEEE Syst. J.*, vol. 13, no. 3, pp. 3400–3409, Sep. 2019.

[75] P. P. Biswas, P. N. Suganthan, B. Y. Qu, and G. A. J. Amaratunga, "Multiobjective economic-environmental power dispatch with stochastic wind-solar-small hydro power," *Energy*, vol. 150, pp. 1039–1057, May 2018.

[76] M. Basu, "Economic environmental dispatch using multi-objective differential evolution," *Appl. Soft Comput.*, vol. 11, no. 2, pp. 2845–2853, Mar. 2011.

SANCHARI DEB received the Master of Engineering degree in power systems and the Ph.D. degree from the Centre for Energy, Indian Institute of Technology, Guwahati, India. She is currently working as an ERCIM Fellow with the VTT Technical Research Centre, Finland. Her research interests include power systems, energy, electric vehicles, charging infrastructure, optimization, and evolutionary algorithms. She is a member of the IEEE PES.

DIAA SALAMA ABDELMINAAM received the Ph.D. degree in information system from the Faculty of Computers and Information, Menufia University, Egypt, in 2015. Since 2011, he has been an Assistance Professor with the Information Systems Department, Faculty of Computers and Information, Benha University, Egypt. He has worked on several research topics. He has contributed more than 40 technical articles in the areas of wireless networks, wireless network security, information security and Internet applications, cloud computing, mobile cloud computing, the Internet of Things, and machine learning in international journals, international conferences, local journals, and local conferences. He majors in cryptography, network security, the IoT, big data, cloud computing, and deep learning.

MOKHTAR SAID received the B.Sc. degree (Hons.) in electrical engineering, the M.Sc. degree, and the Ph.D. degree in renewable energy from the Faculty of Engineering, Fayoum University, Egypt, in June 2009, September 2013, and 2018, respectively. His major research interests include modeling and simulation of electrical systems, electrical drives control, and optimization of renewable energy systems.

ESSAM H. HOUSSEIN received the Ph.D. degree in Computer Science Wireless Networks based on Artificial Intelligence, in 2012. He is currently working as an Associate Professor with the Faculty of Computers and Information, Minia University, Egypt. He is also the Founder of the Computing and Artificial Intelligence Research Group (CAIRG), Egypt. He has more than 80 scientific research articles published in prestigious international journals in the topics of optimization, machine learning, image processing, and the IoT and its applications. His research interests include wireless sensor networks, the IoT, bioinformatics and biomedical, cloud computing, soft computing, image processing, artificial intelligence, data mining, optimization, and metaheuristics techniques. He serves as a Reviewer of more than 30 journals, such as Elsevier, Springer, and IEEE.