energies

Article

Optimal Scheduling of Non-Convex Cogeneration Units Using Exponentially Varying Whale Optimization Algorithm

Vinay Kumar Jadoun [1], G. Rahul Prashanth [1], Siddharth Suhas Joshi [1], Anshul Agarwal [2], Hasmat Malik [3,*], Majed A. Alotaibi [4,5] and Abdulaziz Almutairi [6]

1 Department of Electrical & Electronics Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal 576104, India; vjadounmnit@gmail.com (V.K.J.); rahulprashanth.g@gmail.com (G.R.P.); siddhu0408.sj@gmail.com (S.S.J.)
2 Department of Electrical Engineering, National Institute of Technology, Delhi 110040, India; anshul@nitdelhi.ac.in
3 BEARS, University Town, NUS Campus, Singapore 138602, Singapore
4 Department of Electrical Engineering, College of Engineering, King Saud University, Riyadh 11421, Saudi Arabia; majedalotaibi@ksu.edu.sa
5 Saudi Electricity Company Chair in Power System Reliability and Security, King Saud University, Riyadh 11421, Saudi Arabia
6 Department of Electrical Engineering, College of Engineering, Majmaah University, Al Majma'ah 11952, Saudi Arabia; ad.almutiri@me.edu.sa
* Correspondence: hasmat.malik@gmail.com

Citation: Jadoun, V.K.; Prashanth, G.R.; Joshi, S.S.; Agarwal, A.; Malik, H.; Alotaibi, A.M.; Almutairi, A. Optimal Scheduling of Non-Convex Cogeneration Units Using Exponentially Varying Whale Optimization Algorithm. *Energies* **2021**, *14*, 1008. https://doi.org/10.3390/en14041008

Academic Editor: Nand Kishor Meena
Received: 22 December 2020
Accepted: 8 February 2021
Published: 15 February 2021

Abstract: This paper proposes an Exponentially Varying Whale Optimization Algorithm (EVWOA) to solve the single-objective non-convex Cogeneration Units problem. This problem seeks to evaluate the optimal output of the generator unit to minimize a CHP system's fuel costs. The nonlinear and non-convex characteristics of the objective function demands a powerful optimization technique. The traditional Whale Optimization Algorithm (WOA) is improved by incorporating four different acceleration functions to fine-tune its performance during exploration and exploitation phases. Among the four variants of the proposed WOA, the emphasis is laid on the EVWOA which uses the exponentially varying acceleration function (EVAF). The proposed EVWOA is tested on six different small-scale to large-scale systems. The results obtained for these six test systems, followed by a statistical study highlight the supremacy of EVWOA for finding the best optimal solution and the convergence traits.

Keywords: Combined Heat and Power Unit (CHP); Non-convex Optimal Economic Scheduling; Cogeneration plants; Whale Optimization Algorithm (WOA); Exponentially Varying Acceleration Functions; Meta-heuristic Optimization

1. Introduction

Cogeneration units are a single entity with the ability to produce thermal and electrical energy. The ever-growing need for electrical and heat energy has contributed to an increase in demand for these systems. There are manifold advantages of these schemes. In comparison to coal-based thermal power plants and boilers, some of them have higher productivity, economic benefits, lower environmental emissions, etc. These CHP systems have a performance of about 80 to 85% [1], in comparison to the lower values of around 30 to 40% [2] of coal-based power plants. In CHP systems, by using waste heat that is supplied to different regions as per their needs, they boost the aggregate efficacy of the system. Figure 1 shows a sample sketch of the Heat Recovery Steam Generator, which is one of the configurations of operating a CHP unit. Holding all these advantages of cogeneration systems in mind, having the system run at the highest possible efficiency is of utmost importance. This is made possible by knowing the system's best dispatch schedule. It leads to a CHPED conundrum in which, considering all the operational constraints, electrical and heat energy outputs of each generator unit are computed, such that the fuel cost is the lowest.

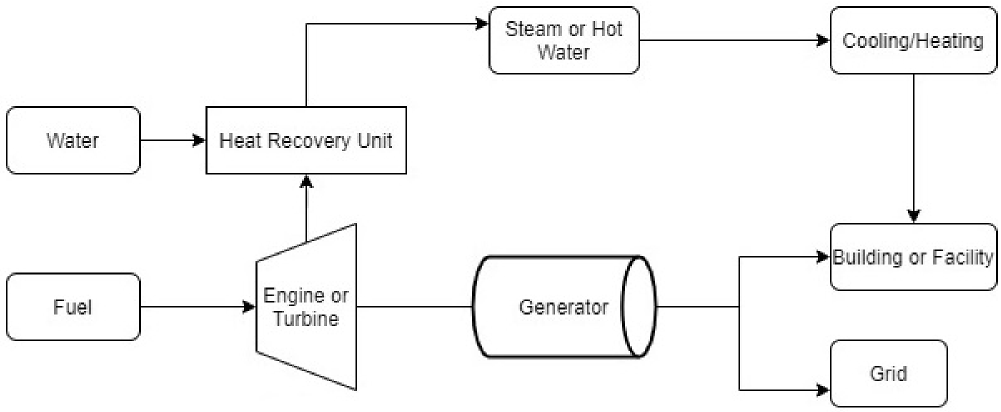

Figure 1. Heat Recovery Steam Generation—One of the configurations of operation of CHP units.

This CHPED conundrum becomes complicated and cumbersome due to the non-convex and nonlinear structure of cost function and considering various operational difficulties. Therefore, to compute the best optimal solution, a strong optimization technique is required. The technique should be able to prevent local trapping in the optimal solution to achieve this best optimal solution and therefore pursue a global best solution by having a better equilibrium between the exploration and exploitation phases. To solve the above-mentioned conundrum, several various methodologies have been used in numerous studies.

Figure 2 shows a summary of the various optimization techniques. Some of the conventional mathematical approaches used in numerous studies include Lagrangian Relaxation [3], Dual and Quadratic Programming [4], and Bender's decomposition [5], etc. However, these methods are unable to find the required solution adequately because they are unable to work successfully with the non-convexities and hard constraints.

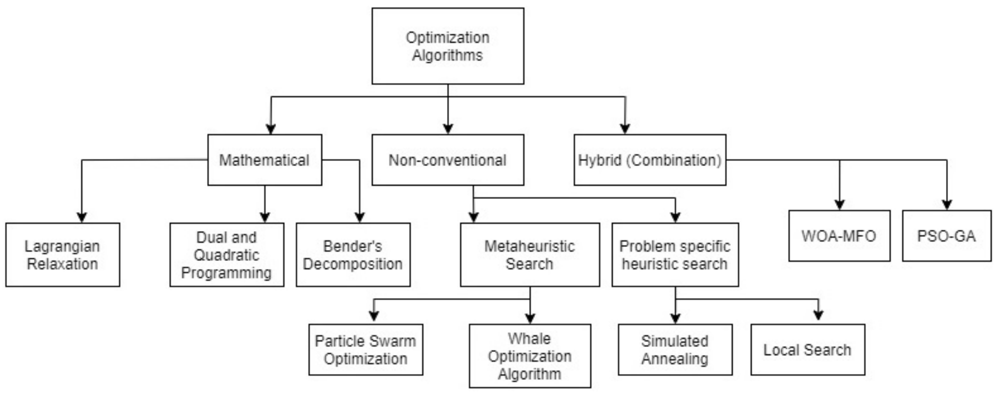

Figure 2. Summary of various types of optimization algorithms.

To deal with the problem of not being able to handle the complexities, researchers have now started using meta-heuristic methods and have found a rational solution. In earlier literature, some of the methods used were: Ant Colony Optimization (ACO) [6], Gravitational Search Algorithm (GSA) [7], Krill Herd Algorithm (KHA) [8], Particle Swarm Optimization (PSO) [9] and its variants such as Selective PSO [10], Time-Varying Acceleration Coefficients PSO (TVAC-PSO) [9] etc. A Self-Adaptive Real coded GA (SARGA) has been proposed in [11]. A penalty factor-based constraint handling is used by the writers to solve the conundrum. Harmony Search, which is based on an analogy of music improvisation process has been used in [12]. An Improvised HS (IHS) technique has also been suggested by the authors to solve the problem. To find a solution to the CHPED conundrum, the authors in [13] used the Teaching-Learning-Based Optimization (TLBO). TLBO is based on a teacher's instructional style and its influence on the pupils. To further the developments, the authors have also proposed a modification known as Oppositional TLBO. Differential evolution (DE) is implemented in [14], which is centered on the adaptation mechanism during species evolution. DE was used in association with a modern mutation operator in [15]. To solve the problem, the authors considered the valve-point effect and added a Gaussian mutation. To address the CHPED problem, a bio-inspired algorithm known as Group Search Optimization (GSO) was used in [16] for solving the problem. It is centered on animals seeking each other while coexisting together. In [17], its modification, referred to as Opposition GSO (OGSO) was used. Several limitations were considered, such as prohibited operating zones and valve-point loading of electrical generators. Cuckoo Search (CS) in [18] has been incorporated for solving the CHPED conundrum. The cuckoo birds' reproductive mechanism is the foundation for it. Because it has a reduced number of control variables, the algorithm was simple to implement. While solving the conundrum, the authors have considered power losses in the framework. Another method used for solving the problem is Bee Colony Optimization (BCO). It sought to duplicate the form of food hunting followed by the honeybees. It was used in [19] and consideration was given to valve-point loading and power losses in the system. Researchers have made use of another bio-inspired approach called Invasive Weed Optimization (IWO) in [20]. The growth trend of unwelcome and invasive weeds at a site forms the foundation of the technique. The Exchange Market Algorithm (EMA) has drawn researchers' interest as one of the best methods of optimization and is used in [21]. The methodology is based on terms from the stock exchange. This includes the role of elite partners in the sale and purchase of market shares. The conduct of these elite stakeholders is analyzed and measured, which forms the basis of the algorithm. Valve-point loading and power losses in the framework have been considered. The algorithm based on Grey Wolf (GWO) was used in [22]. It is focused on the wolves' social actions and hunting practices. Several limitations have been considered when solving the problem, such as valve-point loading, system power losses, ramp-rate limits.

Mirjalili recently implemented a similar technique called the Whale Optimization Algorithm (WOA) [23]. This mimics how when living beneath the surface of the water, humpback whales behave and hunt. This approach has drawn the interest of the researchers effectively and has even been implemented in [24]. The methodology has proved its worth based on its capability of delivering a solution to different types of test systems, with stronger solutions, lesser computational time, and good convergence characteristics ranging from small to large scales. However, the simple WOA is not capable of offering a promising solution when dealing with the large-scale multi-model problem [25]. The key problem is poor convergence speed [26] and premature convergence [27]. This is because of not having a proper balance between local exploration and global exploitation in basic WOA. In previous literature, several researchers have made numerous modifications to boost the efficiency of WOA by better balancing the process of discovery and exploitation. Hongping et al. added a new inertia weight control function similar to PSO in papers [28,29] and obtained an improved WOA (IWOA). This control function is used to change the effect of the best solution currently available. Kaveh et al. recommended enhanced WOA (EWOA) to improve the speed of convergence, reliability, and solution accuracy [30]. In addition, Majdi et. al. proposed Hybrid WOA to solve the issue of function selection [31]. In Hybrid WOA, to improve the best solution in each iteration, the Simulated Annealing (SA) algorithm is paired with the WOA algorithm. Improving the exploitation phase is the key goal of integrating the Simulated Annealing (SA) algorithm with WOA. Reference [25] suggested WOA focused on the Levy Flight Trajectory to solve the issue of global optimization. To ensure a better negotiation between the exploration and exploitation of the WOA, the suggested approach is very useful. Diego et al., in [32], introduced enhanced Chaotic WOA (CWOA) for solving solar photovoltaic cell parameter estimation. Chaotic maps are used for calculation in this CWOA method and the internal parameters of the method are automatically chosen. Aziz et. Mohamed Abd El Aziz et al. to achieve the optimum multi-level thresholds for image segmentation in [33], suggested a hybrid approach consisting of WOA and Moth-Flame Optimization (MFO). A new MLP training method based on the recently developed WOA for the optimization of relation weights in neural networks is proposed in Reference [34]. Mohamed Abdel-Basset et al. presented a framework for hybrid whale optimization with a local search approach to address the issue of permutation flow shop scheduling in [35]. Kaur et al., in [26], by considering multiple chaotic maps, the chaotic WOA (CWOA) approach was suggested. For fine-tuning the key parameters of the WOA system, these chaotic maps are very useful, for maintaining the exploration and exploitation of the proposed CWOA method. A hybrid approach consisting of Support Vector Machines and WOA for solving the challenge of detecting spammers in the issue of online social networks is proposed in Reference [36]. An improved WOA (IWOA) for the correct extraction of the parameters of various solar photovoltaic model problems is proposed in Reference [27]. Two prey-seeking techniques have been implemented by this IWOA approach to effectively preserve the proper balancing between exploration and exploitation to boost WOA efficiency. In addition, Jadhav et al., recommended a hybrid solution consisting of a grey wolf optimizer with whale optimization to solve the optimization problem of data clustering [37]. To solve the 0–1 knapsack problem with various scales, Mohamed Abdel-Basset et al. suggested an improved WOA (IWOA) [38].

In this paper, an Exponentially Varying WOA (EVWOA) is proposed to improve and enhance the performance of the basic WOA in terms of convergence, maintaining the balancing between global exploration and local exploitation throughout the search process and improving its efficiency. The performance of basic WOA is improved by introducing an exponentially varying acceleration function. This helps the proposed EVWOA method for providing proper balancing of both exploration and exploitation phases throughout the search process. The solution in the proposed EVWOA converges faster, is robust and efficient, and obtains a better optimal solution without premature convergence. The proposed EVWOA technique is implemented in this paper for giving a better optimal solution than the traditional WOA along with different variants of WOA. The proposed WOA and its different variants have been tested on six different test systems ranging from a small-scale system of four units to a large-scale system comprising of 96 units with several related constraints. The results obtained by the proposed EVWOA method after 100 different unbiased trials are compared with the basic WOA, its variants, and different published recent algorithms. It can be observed that the proposed WOA method provides good quality solutions in terms of cost, robustness, feasible and obtain effective convergence characteristic.

The main highlights of the paper are as follows:

- Four different variants of WOA are considered for improving and enhancing the performance of the basic WOA. In the first variant of WOA, the acceleration function is randomly selected, and it is written as Randomly Varying WOA (RVWOA). In the second variant of WOA, the acceleration function is linearly varied and is called Linearly Varying WOA (LVWOA), the third variant is Sinusoidally Varying WOA (SVWOA), where the acceleration function is varied Sinusoidally. In addition, in the proposed variant, Exponentially Varying WOA (EVWOA), the exponentially varied acceleration function is used.
- All the four variants of WOA and basic WOA are tested on well-known and standard benchmark functions for performance evaluation and later on six small to large different CHP case studies.

- Simulation results generated by EVWOA after independent 100 different trials are compared with basic WOA, other remaining variants of WOA, and recently published results obtained by different latest methods. The comparison results show that the proposed EVWOA performs much better than other latest methods.

The paper is organized as: the problem formulation is explained in Section 2; a description of the basic WOA and its shortcomings is given in Section 3. This section also explains, in brief, the proposed Exponentially Varying WOA (EVWOA); Section 4 presents the simulation results and discussion; the conclusion is presented in Section 5.

2. Problem Formulation

The objective of the CHPED conundrum is to obtain the best output values for the interconnected electrical energy, heat energy, and CHP plants in a manner that the total fuel cost is minimum, and all the related constraints are satisfied, and energy demands are met.

This can be stated as [3]:

$$Min[Costfuel]$$

where

$$\text{Cost}_{fuel} = \left[\sum_{l=1}^{N_{EG}} C_l\left(EE_l^{EG}\right) + \sum_{m=1}^{N_{CG}} C_m(EE_m^{CG}, HE_m^{CG}) + \sum_{n=1}^{N_{HG}} C_n(HE_n^{HG})\right] \tag{1}$$

where N_{EG} are the number of electrical generator units, N_{CG} are the number of CHP units and N_{HG} are the number of heat generator units. EE and HE are the electrical and heat energy outputs of the units. $C_l\left(EE_l^{EG}\right)$, $C_m(EE_m^{CG}, HE_m^{CG})$ and $C_n(HE_n^{HG})$ are the fuel cost of the lth electrical, mth CHP and heat generator units for the generation of EE^{EG} MW and HE^{HG} MWth for an hour.

The equation of the electrical energy generators taking into account valve-point loading is given as [7]:

$$C_l\left(EE_l^{EG}\right) = \left[e_l\left(EE_l^{EG}\right)^2 + f_l(EE_l^{EG}) + g_l + \left|h_l \sin\left(i_l\left(EE_l^{EG_{min}} - EE\right)\right)\right|\right] \tag{2}$$

where e_l, f_l and g_l are the cost coefficients for the lth electrical energy generator unit and h_l and i_l are the coefficients for modeling the valve-point. The non-convex nature of the CHPED problem can be attributed to this sinusoidal term. Figure 3 shows graphically the effect of valve-point loading.

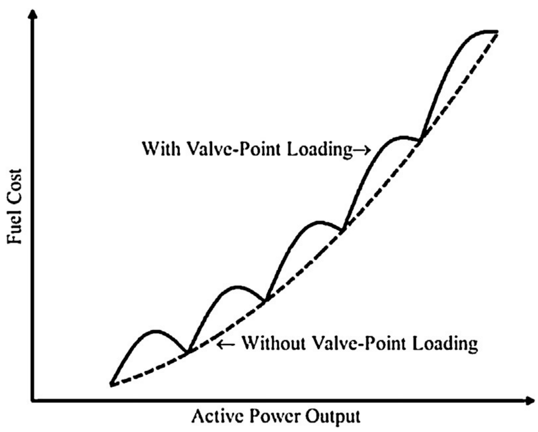

Figure 3. Valve-point loading effect of electrical energy generator units.

The fuel cost for the CHP plants is given by [6]:

$$C_m(EE_m^{CG}, HE_m^{CG}) = e_m(EE_m^{CG})^2 + f_m(EE_m^{CG}) + g_m + h_m(HE_m^{CG})^2 + i_m(HE_m^{CG}) + j_m HE_m^{CG}(EE_m^{CG})]\$/h \quad (3)$$

where e_m, f_m, g_m, h_m, i_m and j_m are the coefficients of the mth CHP unit.

The fuel cost for the heat energy generators is given by [7]:

$$C_n(HE_n^{HG}) = e_n(HE_n^{CG})^2 + f_n(HE_n^{CG}) + g_n\$/h \quad (4)$$

where e_n, f_n and g_n are the cost coefficients of the nth heat energy generator.

2.1. Constraints

The equality and inequality constraints, which form the energy balance equations are stated as [18]:

2.1.1. Equality Constraints

The electrical energy balance equation is stated as:

$$\sum_{l=1}^{N_{EG}} EE_l^{EG} + \sum_{m=1}^{N_{CG}} EE_m^{CG} = EE_{demand} + EE_{Loss} \quad (5)$$

The above Equation (5) signifies that the total electrical energy output from the electrical energy generators and the CHP units should be sufficient to satisfy the electrical energy demand, as well as certain energy losses that may arise in the system in due course of transmission from the plant to the consumer. This electrical energy loss is modeled using Kron's loss formula and is stated as below:

$$EE_{Loss} = \sum_{l=1}^{N_{EG}} \sum_{m=1}^{N_{EG}} EE_l^{EG} B_{lm} EE_m^{EG} + \sum_{l=1}^{N_{EG}} \sum_{m=1}^{N_{CG}} EE_l^{EG} B_{lm} EE_m^{CG} \quad (6)$$

$$+ \sum_{l=1}^{N_{CG}} \sum_{m=1}^{N_{CG}} EE_l^{CG} B_{lm} EE_m^{CG} + \sum_{l=1}^{N_{EG}} B_{0l} EE_l^{EG} + \sum_{l=1}^{N_{CG}} B_{0l} EE_m^{CG} + B_{00}$$

In addition, the heat energy balance equation is given as:

$$\sum_{m=1}^{N_{CG}} HE_m^{CG} + \sum_{n=1}^{N_{HG}} HE_n^{HG} = HE_{demand} \quad (7)$$

Equation (7) signifies that the total heat energy output from the CHP units and heat generator units should be sufficient to meet the thermal energy requirement.

2.1.2. Inequality Constraints

$$EE_l^{EG_{min}} \le EE_l^{EG} \le EE_l^{EG_{max}}\ ; \quad l = 1\ to\ N_{EG} \quad (8)$$

$$EE_m^{CG_{min}}\left(HE_m^{CG}\right) \le EE_m^{CG} \le EE_n^{CG_{max}}\left(HE_m^{CG}\right) \quad m = 1\ to\ N_{CG} \quad (9)$$

$$HE_m^{CG}\left(EE_m^{CG}\right) \le HE_m^{CG} \le HE_m^{CGmax}\left(EE_m^{CG}\right) \quad m = 1\ to\ N_{CG} \quad (10)$$

$$HE_n^{HG_{min}} \le H\ E_n^{HG} \le HE_n^{HG_{max}} \quad n = 1\ to\ N_{HG} \quad (11)$$

where $EE_l^{EG_{min}}$ and $EE_l^{EG_{max}}$ are lower and upper boundary values of the outputs of the lth electrical energy generator in MW, $EE_m^{CG_{min}}\left(HE_m^{CG}\right)$,$EE_n^{CG_{max}}\left(HE_m^{CG}\right)$,$HE_m^{CG}\left(EE_m^{CG}\right)$ and

$HE_m^{CGmax}\left(EE_m^{CG}\right)$ are linear inequalities, which define the feasible operating region of the mth CHP plant, and $HE_n^{HG_{min}}$ and $HE_n^{HG_{max}}$ are the lower and upper boundary values of the of the nth heat energy generator.

Equations (8) to (11) form the heat-power Feasible Operating Region (FOR) of the CHP units. The FOR, as the name suggests is the zone in which CHP units should have their heat and electrical energy outputs to operate. A sample sketch of this FOR is shown in Figure 4.

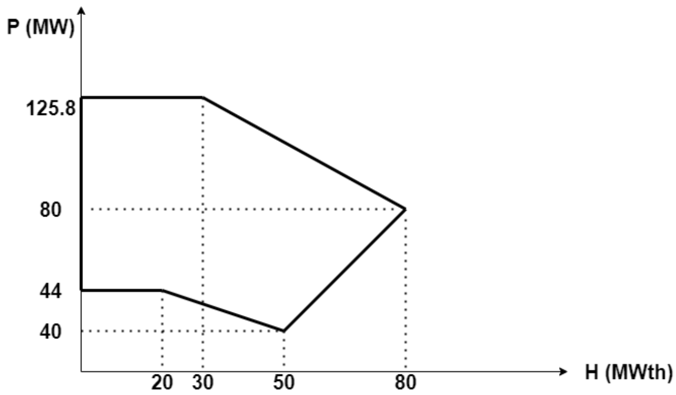

Figure 4. Heat-Power Feasible Operating Region (formed using Equations (8)–(11)).

3. Proposed Exponentially Varying Whale Optimization Algorithm (EVWOA)

This section discusses the equations which form the WOA and some limitations which hampers its performance. Later, an acceleration function term [ζ(itr)] is proposed, which is then appended in the characteristic equations of the WOA.

The WOA is an experience-based algorithm which is having a hunting strategy adopted by humpback whales. The peculiar bubble-net hunting pattern of these whales is the inspiration for the exploitation strategy. The characteristic equations of the WOA are given as: [23–27].

3.1. Encircling Prey

The location of the prey is traced by the leader whale and is then encircled by the other whales. Initially, a leader whale is selected among the group and the others follow it. This phenomenon is given by [23]:

$$\vec{Dist} = \left|\vec{\beta} \times \vec{Zbest}^*(itr) - \vec{Z}(itr)\right| \tag{12}$$

$$\vec{Z}(itr+1) = \vec{Zbest}^*(itr) - \vec{\alpha} \times \vec{Dist} \tag{13}$$

Here, $\vec{\alpha}$ and $\vec{\beta}$ are the coefficient vectors, itr is the present iteration, while $\vec{Z}$ and $\vec{Zbest}^*$ are the position and best position vectors respectively, which are updated in each cycle. The coefficient vectors are given as:

$$\vec{\alpha} = 2\vec{\tau} \times \vec{rand} - \vec{\tau} \tag{14}$$

$$\vec{\beta} = 2 \times \vec{rand} \tag{15}$$

where $\vec{\tau}$ varies from 2 to 0 in a decreasing manner, while $\vec{rand}$ is a random value in [0,1].

In basic WOA, as per Equation (13), the updated position is regulated by multiplication of coefficient vector ($\vec{\alpha}$) and distance of the ith whale to the prey ($\vec{Dist}$) which is subtracted from current best position, which is further dependent upon coefficient vector ($\vec{\alpha}$), where $\vec{\alpha}$ is calculated from Equation (14). From Equation (12), $\vec{Dist}$ is dependent upon coefficient vector ($\vec{\beta}$) and from Equation (15), it can be said that $\vec{\beta}$ is dependent upon $\vec{rand}$ which is a random vector. It can be observed that $\vec{\alpha}$ is dependent upon $\vec{\tau}$ which is decreasing linearly with respect from 2 to 0 as per the iterations. This may cause uncontrolled updated positions during the iterations. It may also create an unbalance between local exploration and global exploitation in the search space and further it may result in poor convergence. For improving this, ref. [39,40] introduced an inertia weight into WOA similar to the PSO to fine-tune the influence of the current best solution. According to references [39,40], in the proposed WOA, an acceleration function (ζ) is introduced.

In proposed EVWOA, Equations (12) and (13) for Encircling prey are modified as and referred from [28,29]:

$$\vec{Dist} = \left|\vec{\beta} \times \zeta(itr) \times \vec{Zbest}^*(itr) - \vec{Z}(itr)\right| \tag{16}$$

$$\vec{Z}(itr+1) = \zeta(itr) \times \vec{Zbest}^*(t) - \vec{\alpha} \times \vec{Dist} \tag{17}$$

3.2. Exploitation Phase

As stated earlier, the exploitation phase is based on the bubble-net hunting pattern of the whales. Here, two different techniques are used to mimic this behavior.

3.2.1. Shrinking Encircling

Shrinking encircling is obtained when $\vec{\tau}$ reduces to 0 from 2 in Equation (14). Hence as $\vec{\tau}$ decreases, a lesser variation is seen in the value of $\vec{\alpha}$, which lies in $[-\vec{\tau}, \vec{\tau}]$.

3.2.2. Spiral Updating Position

In this method, it is assumed that the whale and the prey are present at (Z,W) and (Z_{best}, W_{best}). The following equation defines the helical movement of the whale towards the prey [24]:

$$\vec{Z}(itr+1) = \vec{Dist}' times^{\gamma\eta} \times \cos(2\pi\eta) + \vec{Zbest}^*(itr) \tag{18}$$

$\vec{Dist}'$ is the distance between the ith whale and the prey. It is written as:

$$\vec{Dist}' = \left|\vec{Zbest}^*(itr) - \vec{Z}(itr)\right| \tag{19}$$

$'\gamma'$ lies between $[-1,1]$ and is fixed valued and helps to model the spiral shape of the path.

By assuming that around the prey, whales either swim in a spiral path or a shrinking encircling path, an equal probability is assumed for selecting either of the paths. This is stated as [25]:

$$\vec{Z}(itr+1) = \vec{Zbest}^*(itr) - \vec{\alpha} \times \vec{Dist},\ prob \leq 0.5 \tag{20}$$

$$\vec{Z}(itr+1) = \vec{Dist}' \times e^{\gamma\eta} \times \cos(2\pi\eta) + \vec{Zbest}^*(itr),\ prob \geq 0.5$$

where *prob* is the chance of selecting either path.

In the proposed EVWOA, the above equations are modified as:

$$\vec{Z}(itr+1) = \vec{Dist}' \times e^{\gamma\eta} \times \cos(2\pi\eta) + \zeta(itr) \times \vec{Zbest}^*(itr) \tag{21}$$

$$\vec{Dist}' = \left|\zeta(itr) \times \vec{Zbest}^*(itr) - \vec{Z}(itr)\right| \tag{22}$$

$$\vec{Z}(itr+1) = \zeta(itr) \times \vec{Zbest}^*(itr) - \vec{\alpha} \times \vec{Dist},\ \ prob < 0.5 \tag{23}$$

$$\vec{Z}(itr+1) = \vec{Dist}' \times e^{\gamma\eta} \times \cos(2\pi\eta) + \zeta(itr) \times \vec{Zbest}^*(itr),\ \ prob \geq 0.5$$

3.3. Exploration Phase

This phase comprises searching for the prey, in which the value of $\vec{\alpha}$ can be changed for locating the prey. Whenever $|\alpha| > 1$, the current search agent is to be moved away from the leader agent. The location of an agent in this phase is improved by erratically selecting a particle. This phase is described by the following [26]:

$$\vec{Dist} = \left|\vec{\beta} \times \vec{Z_{rand}} - \vec{Z}\right| \quad (24)$$

$$\vec{Z}(itr+1) = \vec{Z_{rand}} - \vec{\alpha} \times \vec{Dist} \quad (25)$$

In the proposed EVWOA, these equations are varied as [28,29]:

$$\vec{Dist} = \left|\zeta(itr) \times \vec{\beta} \times \vec{Z_{rand}} - \vec{Z}\right| \quad (26)$$

$$\vec{Z}(t+1) = \zeta(itr).\vec{Z_{rand}} - \vec{\alpha} \times \vec{Dist} \quad (27)$$

3.4. Choice of Acceleration Function

Four different variants of WOA are considered by incorporating different acceleration functions for improving and enhancing the WOA performance. These different variants are as follows:

- Randomly Varying WOA (RVWOA)
- Linearly Varying WOA (LVWOA)
- Sinusoidally Varying WOA (SVWOA)
- Exponentially Varying WOA (EVWOA)

The detailed description of these above-mentioned different variants of WOA is as explained below:

1. Randomly Varying WOA (RVWOA): In the first variant of WOA, acceleration function is randomly selected between [0,1] as in Reference [29].
2. Linearly Varying WOA (LVWOA): In the second variant of WOA, acceleration function is linearly varied and decreasing from 0.9 to 0.1 as in References [29,39,40]. The expression is written below:

$$\zeta(itr) = \zeta_{min} + \frac{(\zeta_{max} - \zeta_{min}) \times (itr_{max} - itr)}{itr_{max}} \quad (28)$$

3. Sinusoidally Varying WOA (SVWOA): In this variant, the acceleration function (ζ) is varied sinusoidally from 0.9 to 0.1 as in [39]. The mathematical expression is given below:

$$\zeta(itr) = \zeta_{min} + (\zeta_{max} - \zeta_{min})cos^2\left(\frac{\theta}{2}\right); 0 \le \theta \le \pi \quad (29)$$

where θ = X * itr + Y and the coefficients X and Y are calculated by Equations (30) and (31) and iteration (itr) is varied from itrmin to itrmax.

$$X = \pi/(itr_{max} - itr_{min}) \quad (30)$$

$$Y = -\pi \times itr_{min}/(itr_{\max -} \; itr_{min}) \quad (31)$$

4. Exponentially Varying WOA (EVWOA): In this proposed variant, the acceleration function (ζ) is varied exponentially as in [40]. The exponential variation of ζ is calculated by the following relation:

$$\zeta(itr) = \exp(-l \ln k_w) \quad (32)$$

where $l = \frac{itr}{itr_{max}}$, $itr_{min} \le itr \le itr_{max}$ and k_w is the ratio of maximum minimum bounds of the acceleration function.

The linear, sinusoidal, and exponential variations of acceleration function with iterations are shown in Figure 5. It can be analyzed from the figure that ζ-linear is decreasing linearly. By this, there may be a chance of local trapping or premature convergence during the iterations. Moreover, in the case of ζ-sinusoidal, during the first half (50%) of iterations, ζ is maintained higher for wider exploration, and in the second half, ζ is lower for better exploitation during the computation process. However, when ζ-exponential is considered, initially exploration is wider and exploitation capability is improved by a sharp fall in ζ during the iterations. This will improve and efficiently maintain the exploration and exploitation phase of the proposed method. Hence, it is favorable that ζ exponential is selected for the best performance for the WOA. Therefore, further in this paper, the emphasis is laid on the EVWOA.

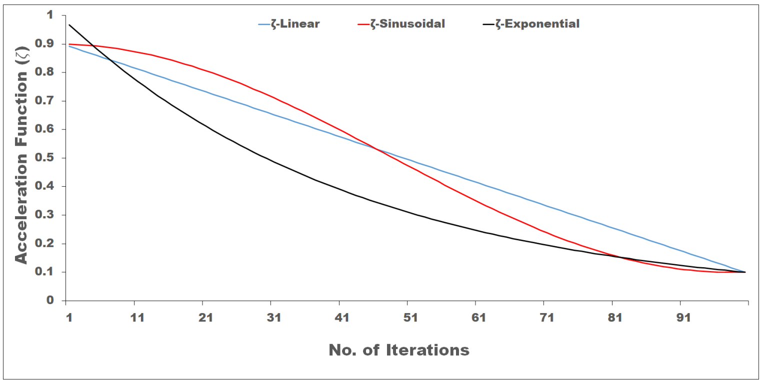

Figure 5. Linear, Sinusoidal and Exponential variation of acceleration function (ζ) with respect to iterations.

The flowchart of the Exponentially Varying WOA (EVWOA) is shown in Figure 6. The flowchart in a concise manner explains the various phases of the EVWOA, right from the selection of the EVAF up to displaying the results of the CHPED problem.

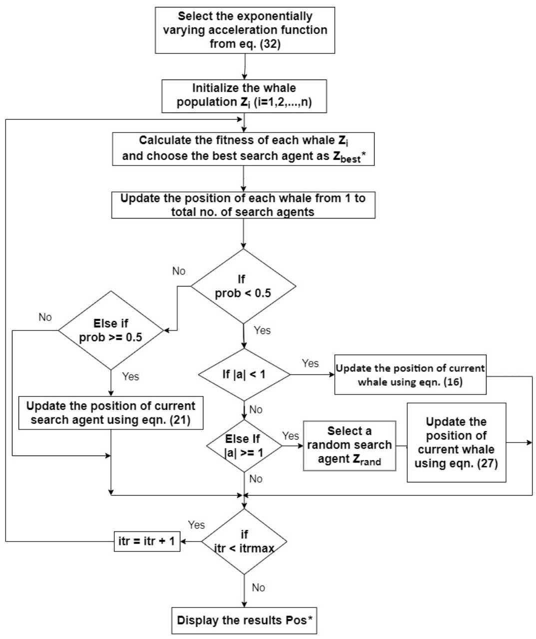

Figure 6. Flowchart of Exponentially Varying Whale Optimization Algorithm (EVWOA).

Figures 7 and 8 show a sample 2D representation of the exploration and the exploitation phases of the proposed EVWOA. These figures help understand the movement of a whale graphically.

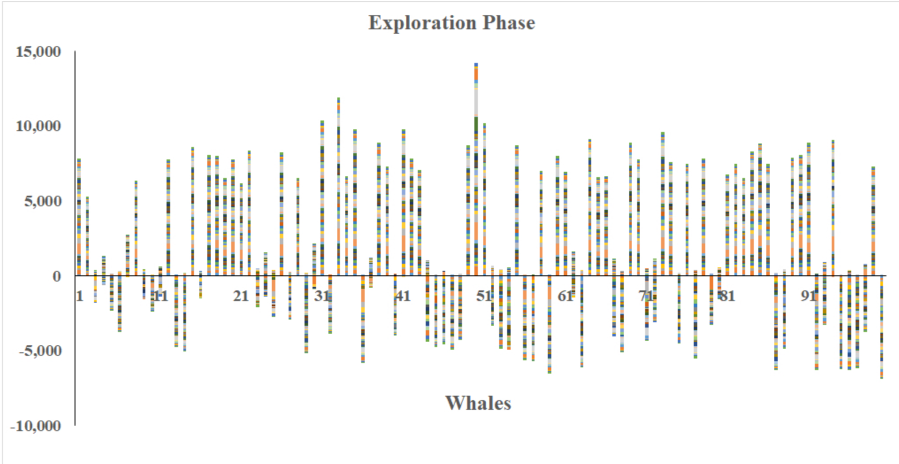

Figure 7. Exploration phase of the proposed EVWOA.

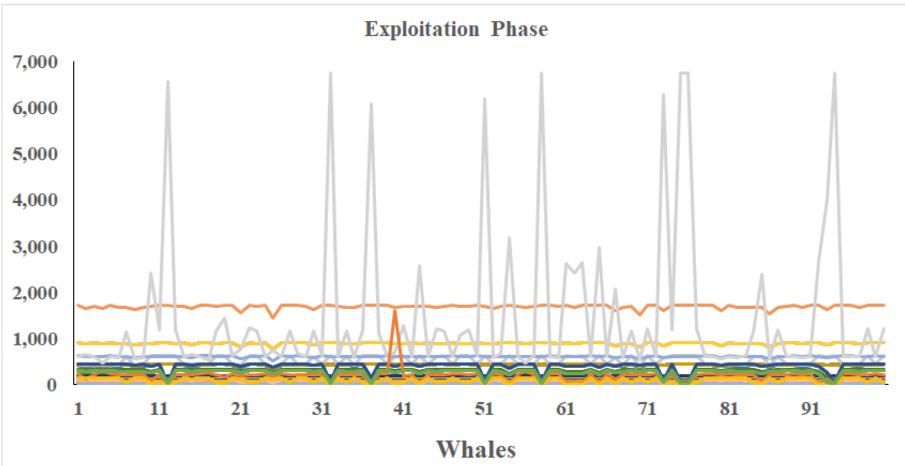

Figure 8. Exploitation phase of the proposed EVWOA.

For the above Figures 7 and 8, the population of whales considered was 100 whales and number of generators considered were 60.

3.5. Testing of the WOA Variants on Well-Known Benchmark Functions

The proposed EVWOA is tested on ten different well-known benchmark functions. The benchmark functions are defined in Table 1 and their optimum and bound limits are known to be a priori. The functions used are Ackley, Generalized Griewank, Generalized Rastrigin, Generalized Rosenbrock, Schwefel 1.2, Generalized Schwefel 2.26, Sphere, Six-hump Camel-back, and Goldstein-Price. The data for the benchmark functions are referred from [41].

The parameters for EVWOA are: Population of whales = 100 and the Maximum number of iterations (itrmax) = 1000. The results obtained using proposed EVWOA after 100 different trials are compared with some of the well-known optimization algorithms in Table 2.

MATLAB ver. 2020a is used to simulate these benchmark functions. The computer specifications are Processor—Intel i5 7th Generation @ 2.50 GHz, RAM – 8 GB, and Storage Capacity—1TB.

Table 1. Description of the Benchmark Functions.

Function	Expression	D	Search Space
Ackley	$20exp(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}x_i^2}) - exp(\frac{1}{D}\sum_{i=1}^{D}cos(2\pi x_i))$ $+20 + exp(1)$	30	$(-32, 32)^D$
Generalized Griewank	$\frac{1}{4000}\sum_{i=1}^{D}x_i^2 - \prod_{i=1}^{D}cos(\frac{x_i}{\sqrt{i}}) + 1$	30	$(-600, 600)^D$
Generalized Rastrigin	$\sum_{i=1}^{D}x_i^2 - 10\,cos(2\pi x_i) + 10$	30	$(-5.12, 5.12)^D$
Generalized Rosenbrock	$\sum_{i=1}^{D-1}100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2$	30	$(-30, 30)^D$
Schwefel 1.2	$\sum_{i=1}^{D}(\sum_{j=1}^{i}x_j)^2$	30	$(-100, 100)^D$
Schwefel 2.26	$418.9829 * D - \sum_{i=1}^{D}x_i sin\sqrt{x_i}$	30	$(-500, 500)^D$
Sphere	$\sum_{i=1}^{D}x_i^2$	30	$(-100, 100)^D$
Six-hump Camel-back	$4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	2	$(-5, 5)^D$
Goldstein-Price	$1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2$ $+3x_2^2) \times 30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2-$ $36x_1x_2 + 27x_2^2)$	2	$(-5, 5)^D$

Table 2. Comparison of the Results of Benchmark Functions.

F	PSO [42]	HS [42]	EFWA [41]	IFWA CSO [41]	RV WOA	LV WOA	SV WOA	EV WOA
F1	2.00×10^{-2}	2.94×10^{-3}	-	-	0	0	0	0
F2	5.50×10^{-2}	5.00×10^{-1}	9.64×10^{-2}	0	0	0	0	0
F3	7.28×10^{2}	4.27×10^{-2}	4.02×10^{-1}	0	0	0	0	0
F4	1.70×10^{3}	7.64×10^{1}	1.01×10^{2}	3.02×10^{1}	0	0	0	0
F5	2.90×10^{3}	3.66×10^{3}	2.36×10^{-1}	5.30×10^{-4}	0	0	0	0
F6	-	-	-1.12×10^{4}	-1.26×10^{4}	0	0	0	0
F7	5.00×10^{-2}	5.14×10^{-4}	-	-	0	0	0	0
F8	-	-	-1.03	-1.03	0	0	0	0
F9	-	-	3.00	3.00	3.00	3.00	3.00	0

From Table 2 it can be observed that for the non-shifted functions F1, F2, F3, F5, and F7, the proposed EVWOA can obtain the local minima of an absolute zero, compared to some of the other algorithms considered for comparison. For the shifted functions F4, F6, F8, and F9, the performance of the proposed EVWOA is found to be satisfactory compared to the other algorithms, where except in F9, EVWOA obtains the minima of 0, especially when compared with FWA variants [41], PSO [42] and HS [42].

4. Results and Discussions

In this paper, six different case studies have been considered and solved using WOA and all the different variants of the proposed WOA, i.e., RVWOA, LVWOA, SVWOA, and EVWOA. A summary of the six different case studies is presented in Table 3.

Table 3. Case Studies Data for solving CHPED Problem.

Case Study	N_{EG}	N_{CG}	N_{HG}	Total	EE_{Demand}	HE_{Demand}
1	1	2	1	4	200	115
2	1	3	1	5	300	150
3	4	2	1	7	600	150
4	13	6	5	24	2350	1250
5	26	12	10	48	4700	2500
6	52	24	20	96	9400	5000

For solving the proposed problem, WOA and different proposed variants of WOA are implemented in MATLAB ver. 2020a on PC with 8 GB RAM, 1 TB storing capacity, and an Intel i5 processor. Each case study is simulated 100 different times, and the schedule obtained for the best cost using EVWOA is presented in this paper. For showing the algorithm's robustness, the best, mean and worst cost, along with the standard deviation (STD) of cost and computation time is also presented. These factors would establish the algorithm's superiority compared to the basic WOA and other latest techniques in the literature. For each of the case studies, the population of whales and iterations, both are considered to be 100.

4.1. Case Study 1

This case study data has been referred from [3] and is given in Appendix A. The electrical and thermal energy requirement of the whole system is 200MW and 115MWth. This system is considered to be a small-scale system. The optimal schedule obtained by the proposed EVWOA for the best cost is presented in Table 4.

Table 4. Optimal schedule for the best cost obtained for Case study 1.

P1	P2	P3	H2	H3	H4
0	159.99	39.99	0	115	0

The results obtained by the proposed EVWOA after 100 different trials are presented in Table 5. It can be observed from Table 5 that the results obtained by WOA, and other variants of WOA are giving much better results as compared with recently published methods such as Firefly Algorithm (FA) [43], Genetic Algorithm (GA) [44], Differential Evolution (DE) [15] and Grey Wolf Algorithm (GWO) [22].

Table 5. Comparison of quality solution for Case Study 1.

Technique	Best ($/h) Cost	Mean ($/h) Cost	Worst ($/h) Cost	STD	Time(s)
FA [43]	9257.1	-	-	-	-
GA [44]	9267.2	-	-	-	-
DE [15]	9236.14	-	-	-	1.0674
GWO [22]	9257.07	-	-	-	1.33
WOA	9089.446	9108.160	9496.51	0.019422	0.23
RVWOA	9089.440	9089.489	9089.498	0.000001	0.27
LVWOA	9089.432	9089.488	9089.497	0.000001	0.24
SVWOA	9089.429	9089.486	9089.497	0.000002	0.22
EVWOA	9089.420	9089.441	9089.442	0.000001	0.26

It can also be analyzed from Table 5 that among WOA and different variants of WOA, the basic WOA is improved by adding different acceleration functions, the quality of solution in terms of best cost (minimum cost), mean cost (average cost) and Standard

Deviation (STD) is also improved. From Table 5, proposed EVWOA gives minimum best cost, minimum average cost, and minimum STD than WOA and other different variants of WOA. This establishes the robustness of the proposed EVWOA. The average CPU time taken by the proposed EVWOA is less than DE [15] and GWO [22]. It shows that the proposed EVWOA converges faster with less CPU time. Figure 9 compares the best cost of all the above-mentioned algorithms in a graphical format. Among recently published methods i.e., DE [15], GWO [22], FA [43] and GA [44], DE [15] has the lowest cost of \$9,236.14/h. By comparing the minimum cost obtained by DE [15] and the proposed EVWOA, EVWOA gives an hourly saving of \$146.72, which means that \$1,285,267 could be saved annually. This leads to the fact that the proposed EVWOA is capable of obtaining better quality solutions compared to other methods in terms of the best and average fuel cost and is also an efficient optimization algorithm to solve dispatch problems.

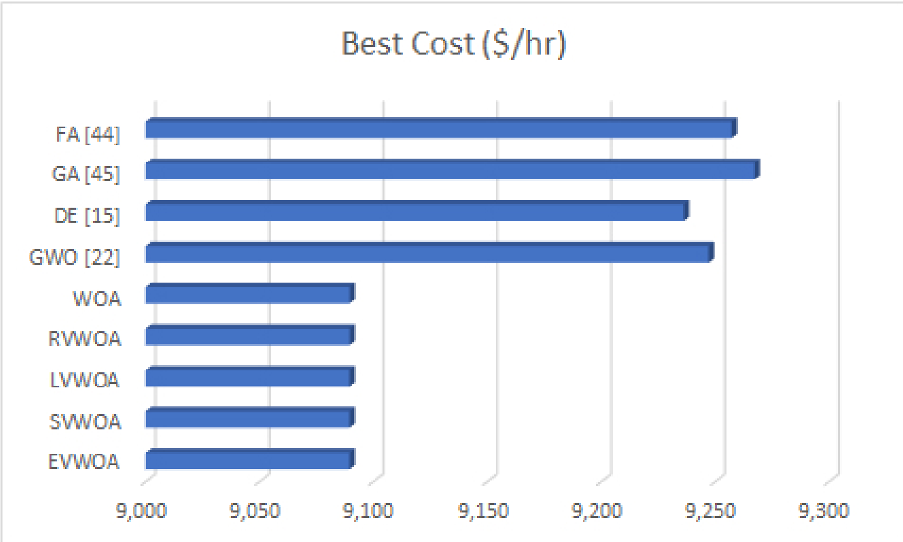

Figure 9. Best Cost Comparison for Case Study 1.

4.2. Case Study 2

This case study data has been taken from [45] and is given in Appendix B. The optimal generator output schedule obtained by the proposed EVWOA is given in Table 6.

Table 6. Optimal schedule for the best cost obtained for Case study 2.

P1	P2	P3	P4	H2	H3	H4	H5
135	40.77	19.23	105	73.64	36.73	0	39.63

The best optimal solution generated by WOA and different variants of WOA after independence 100 trials are compared with the cost obtained with Firefly Algorithm (FA) [43], Harmony Search (HS) [45], Invasive Weed Optimization (IWO) [20], Cuckoo Optimization Algorithm (COA) [46] and Exchange Market Algorithm (EMA) [21] in Table 7. It is observed from this Table 7 that WOA and other different proposed variants of WOA provide best minimum best cost as compared with latest published methods i.e., FA [43], HS [45], IWO [20], COA [46] and EMA [21].

Table 7. Comparison of quality solution for Case Study 2.

Technique	Best ($/h) Cost	Mean ($/h) Cost	Worst ($/h) Cost	STD	Time(s)
FA [43]	13,683.22	-	-	-	-
HS [45]	13,723.2	-	-	-	-
IWO [20]	13,683.65	-	-	-	1.0674
COA [46]	13,672.83	-	-	-	1.33
EMA [21]	13,672.84	-	-	-	1.33
WOA	13,672.8211	13,686.0696	13,738.7084	0.001269	0.179
RVWOA	13,672.7970	13,673.5999	13,715.5134	0.000403	0.178
LVWOA	13,672.7964	13,673.4296	13,694.7275	0.000311	0.161
SVWOA	13,672.7961	13,673.1225	13,681.8538	0.000073	0.162
EVWOA	13,672.7889	13,672.7911	13,672.8052	0.000000	0.229

It can also be observed from this Table 7 that the proposed EVWOA again obtained the best quality solution with almost 0 STD than WOA and RVWOA, LVWOA, and SVWOA. This shows that EVWOA is efficient, robust, and feasible. Figure 10 compares the best cost obtained by all the above-mentioned algorithms. From Table 7, COA [46] has the lowest hourly cost of $13,672.83/h. Therefore, by implementing EVWOA for this test system, hourly savings of $0.04 are possible, which amounts to a yearly saving of $350.40. It still proves that EVWOA can work more efficiently than other latest techniques.

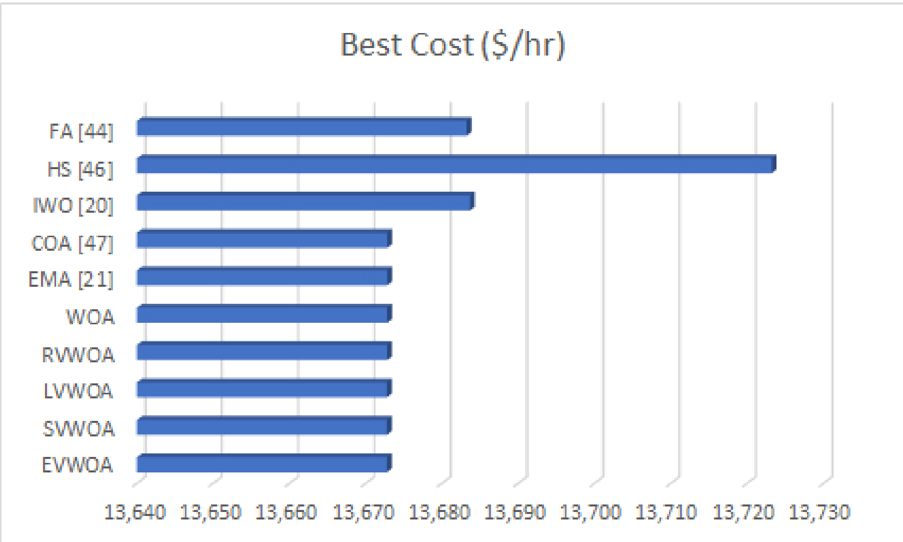

Figure 10. Best Cost Comparison for Case Study 2.

4.3. Case Study 3

The case study parameters have been taken from [47] and are also provided in Appendix C. Power system transmission losses are taken into account for this system.

The optimal generator schedule by EVWOA is provided in Table 8.

Table 8. Optimal schedule for the best cost obtained for Case study 3.

P1	P2	P3	P4	P5	P6	H5	H6	H7
58.81	98.54	112.67	209.82	81	40	0	95.18	54.82

The quality solution obtained by WOA and its proposed variants after 100 trials are compared with the best cost obtained using Teaching-Learning-Based Optimization (TLBO) [13], Gravitational Search Algorithm (GSA) [7] and Differential Evolution (DE) [48] in Table 9. This shows that the minimum cost of WOA and its proposed variants are much better than TLBO [13], GSA [7] and DE [48] with less CPU time.

Table 9. Comparison of quality solution for Case Study 3.

Technique	Best ($/h) Cost	Mean ($/h) Cost	Worst ($/h) Cost	STD	Time(s)
TLBO [13]	10,094.84	-	-	-	2.86
GSA [7]	9912.69	-	-	-	2.578
DE [48]	10,317	-	-	-	5.26
WOA	9798.9957	10,025.0678	10,445.6392	0.015587	0.84
RVWOA	9745.6136	9820.6379	10,052.4499	0.005443	0.764
LVWOA	9742.9177	9819.7254	10,048.6937	0.005436	0.637
SVWOA	9741.0869	9818.4077	10,030.7712	0.004869	0.643
EVWOA	9739.5049	9810.4882	10,003.2598	0.004716	0.767

From this comparison, it can be concluded that EVWOA provides minimum best cost and minimum average cost than the WOA and other variants of WOA i.e., RVWOA, LVWOA, and SVWOA. The standard deviation of EVWOA is also quite good and is about 0.005, meaning that in almost each trial best cost is achieved.

Figure 11 compares the best cost achieved by all the above algorithms in a graphical format. Among the other published algorithms TLBO [13], GSA [7] and DE [48], GSA [7] provides the lowest cost of $9,912.69/h. However, by using the proposed EVWOA an hourly saving of $173.18 is achieved, which leads to a yearly saving of $1,517,057, which is a huge amount. This high figure of saving re-establishes the superiority of the proposed EVWOA as it can provide a better solution with less CPU time for a system where system transmission losses and other constraints are considered.

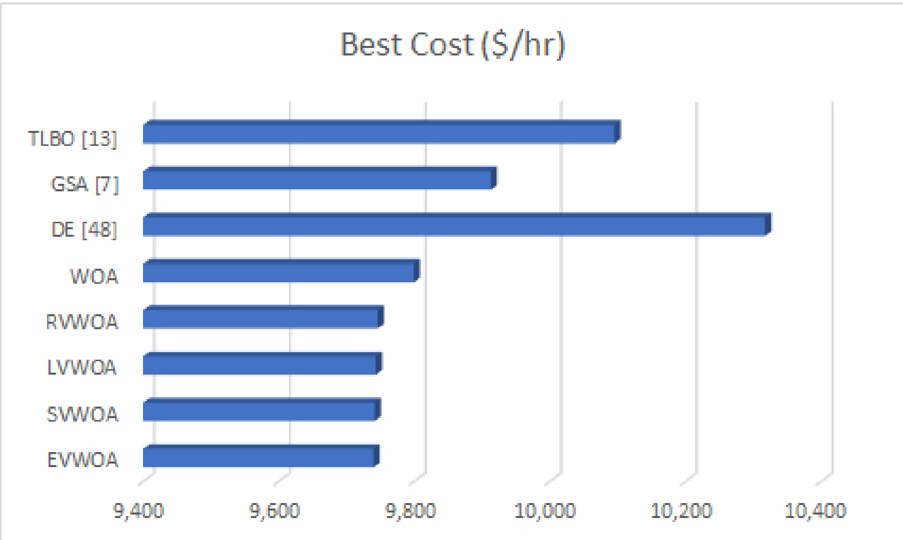

Figure 11. Best Cost Comparison for Case Study 3.

4.4. Case Study 4

The data for this test system is referred from [24] and is given in Appendix D. The schedule for the best cost obtained by EVWOA is provided in Table 10.

Table 10. Optimal schedule for the best cost obtained for Case study 4.

P1	628.32	P7	159.73	P13	55	P19	35	H19	45
P2	298.65	P8	60	P14	81	H14	180	H20	158.79
P3	286.85	P9	60	P15	40	H15	135.6	H21	60
P4	109.86	P10	40	P16	81	H16	180	H22	60
P5	109.86	P11	40	P17	40	H17	135.6	H23	120
P6	159.73	P12	55	P18	10	H18	55	H24	120

The solutions obtained by WOA and its variants after 100 trials are presented in Table 11. It can be observed from this table that the quality solution of WOA and different variants of WOA are much better than Oppositional Teaching-Learning-Based Optimization (OTLBO) [13], Exchange Market Algorithm (EMA) [21], Improved Group Search Optimization (IGSO) [49] and Civilized Swarm Optimization (CSO) [48].

Table 11. Optimal schedule for the best cost obtained for Case study 4.

Technique	Best ($/h) Cost	Mean ($/h) Cost	Worst ($/h) Cost	STD	Time(s)
OTLBO [13]	57,856.27	57,883.22	57,913.77	-	-
EMA [21]	57,825.48	57,832.74	57,841.15	-	-
IGSO [49]	58,049.02	58,156.52	58,219.14	-	-
CSO [21]	57,907.12	57,908.31	579,11.95	-	-
WOA	53413.4800	53,985.95	55,991.7200	0.011879	0.513
RVWOA	53,370.3314	53,766.9799	55,314.1664	0.005630	0.560
LVWOA	53,288.2511	53,737.1996	55,253.9625	0.005256	0.564
SVWOA	53,275.6137	53,730.6929	54,805.6054	0.004046	0.564
EVWOA	53,167.3683	53,373.1923	53,813.7479	0.002577	0.587

It can also be analyzed from this table that again proposed EVWOA gives minimum cost and minimum average cost with a lesser standard deviation of around 0.003 than WOA, RVWOA, LVWOA, and SVWOA. Figure 12 compares the best cost of all the above algorithms graphically.

EMA [21] among other published algorithms OTLBO [13], IGSO [49] and CSO [48] has the lowest best and average costs. When comparing the average cost obtained by EMA [21] and proposed EVWOA, it can be said that the proposed EVWOA provides an hourly saving of $4459.5447 and this leads to a yearly saving of $39,065,637.852 which is indeed a huge sum. This high figure is due to better handling of the related constraints while using EVWOA to solve the dispatch problem.

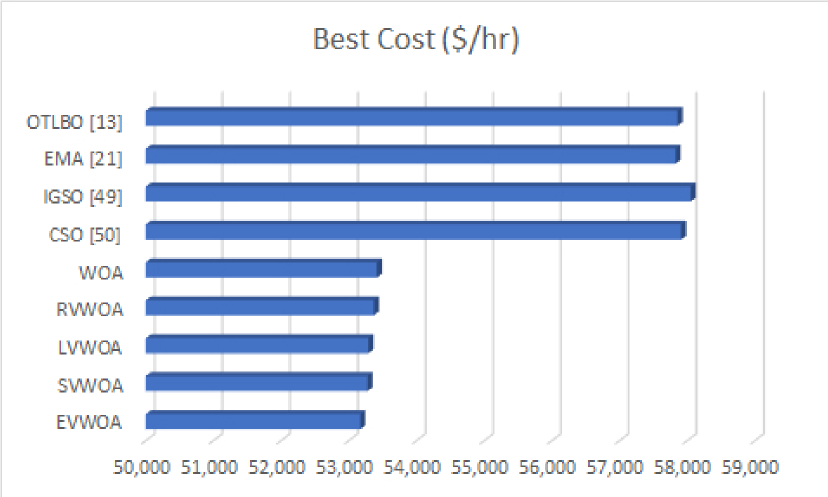

Figure 12. Best Cost Comparison for Case Study 4.

4.5. Case Study 5

In this case study, 48 units large test system is considered. The data is referred from [9] and is provided in Appendix E. The optimal generator outputs obtained by EVWOA are as shown in Table 12.

Table 12. Optimal schedule for the best cost obtained for Case study 5.

P1	616.01	P16	360	31	10	H34	135.6
P2	0	P17	179.99	P32	35	H35	180
P3	0	P18	179.99	P33	81	H36	135.6
P4	60	P19	179.99	P34	40	H37	55
P5	60	P20	179.99	P35	81	H38	45
P6	60	P21	179.99	P36	40	H39	176.41
P7	60	P22	179.99	P37	10	H40	60
P8	60	P23	119.99	P38	35	H41	60
P9	60	P24	119.99	H27	179.98	H42	120
P10	40	P25	119.99	H28	135.6	H43	120
P11	40	P26	120	H29	144.74	H44	176.46
P12	55	P27	81	H30	135.6	H45	60
P13	55	P28	40	H31	55	H46	60
P14	679.99	P29	81	H32	45	H47	120
P15	359.99	P30	40	H33	180	H48	120

The cost obtained by WOA and its variants are compared with the costs obtained using Cuckoo Optimization Algorithm (COA) [46], Crisscross Optimization Algorithm (CSO) [50], Modified Particle Swarm Optimization (MPSO) [51] and Group Search Optimization (GSO) [52] in Table 13. It can be said that the quality solution, i.e., minimum cost, average cost, and STD obtained by WOA, RVWOA, LVWOA, SVWOA, and EVWOA are much better than the solution presented by COA [46], CSO [50], MPSO [51] and GSO [52]. It shows that WOA and its variants perform superior to other established methods with less CPU time.

Table 13. Optimal schedule for the best cost obtained for Case study 5.

Technique	Best ($/h) Cost	Mean ($/h) Cost	Worst ($/h) Cost	STD	Time(s)
COA [46]	116,789.92	116,835.55	117,068.27	-	-
CSO [50]	115,967.72	115,995.88	116,047.22	-	-
MPSO [51]	116,465.54	116,471.36	116,482.44	-	-
GSO [52]	116,457.96	116,463.65	116,473.22	-	-
WOA	99,136.8303	107,026.2565	116,304.7858	0.033216	0.471
RVWOA	99,136.8303	107,026.2565	116,304.7858	0.033216	0.471
LVWOA	97,958.8952	104,552.0051	111,761.6686	0.026202	0.559
SVWOA	97,774.6398	104,342.2406	111,088.5884	0.023165	0.540
EVWOA	92,887.876940	96,657.722614	105,615.720811	0.021991	0.617

It can also be observed from Table 13 that minimum cost, average cost, and STD obtained by the proposed EVWOA outperforms WOA, RVWOA, LVWOA, and SVWOA. It implies that the proposed EVWOA dominates all other techniques. Figure 13 compares the best cost of all the above algorithms graphically. GSO [52] has better best minimum cost and average costs among COA [46], CSO [50] and MPSO [51]. When comparing the average cost of $116,463.65/h obtained by GSO [52] and EVWOA's average cost of $96,657.72/h, it can be observed that EVWOA gives an hourly saving of $19,805.93 and the yearly savings are $173,499,947 which is a very huge amount. This again is attributed to better constraint handling employed while solving with EVWOA.

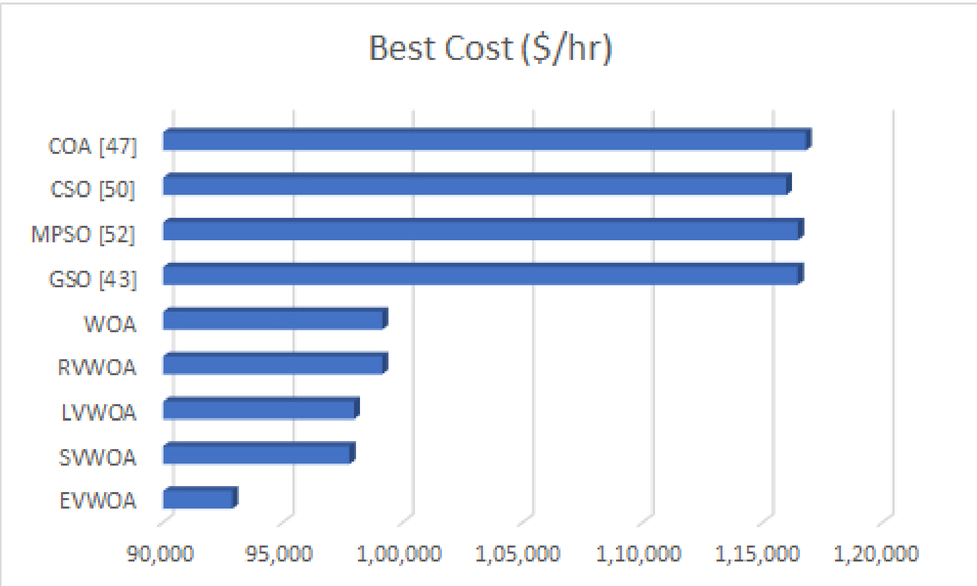

Figure 13. Best Cost Comparison for Case Study 5.

4.6. Case Study 6

This is also considered to be a large system consisting of 96 units. This case study is included in this work to highlight the superiority of EVWOA in efficiently handling large test systems. The data is taken from [24] and is included in Appendix F. The optimal generator outputs using EVWOA are presented in Table 14.

Table 14. Optimal schedule for the best cost obtained for Case study 6.

P 1	630.39	P 21	109.69	P 41	294.73	P 61	81	H 57	0.014	H 77	0.014
P 2	294.73	P 22	109.91	P 42	294.73	P 62	40	H 58	0.014	H 78	0.014
P 3	294.73	P 23	40	P 43	109.91	P 63	10	H 59	0.014	H 79	0.014
P 4	109.91	P 24	40	P 44	109.91	P 64	35	H 60	0.014	H 80	0.014
P 5	109.91	P 25	55	P 45	109.91	P 65	81	H 61	0.014	H 81	0.014
P 6	109.91	P 26	55	P 46	108.22	P 66	40	H 62	0.014	H 82	2499.70
P 7	109.91	P 27	630.39	P 47	109.69	P 67	81	H 63	0.014	H 83	0.014
P 8	109.91	P 28	294.73	P 48	86.79	P 68	40	H 64	0.014	H 84	0.014
P 9	109.91	P 29	294.73	P 49	40	P 69	10	H 65	0.014	H 85	0.014
P 10	40	P 30	109.91	P 50	40	P 70	35	H 66	0.014	H 86	0.014
P 11	40	P 31	109.91	P 51	55	P 71	81	H 67	0.014	H 87	2499.70
P 12	55	P 32	109.91	P 52	55	P 72	40	H 68	0.014	H 88	0.014
P 13	55	P 33	109.91	P 53	81	P 73	81	H 69	0.014	H 89	0.014
P 14	630.39	P 34	109.91	P 54	40	P 74	40	H 70	0.014	H 90	0.014
P 15	294.73	P 35	109.91	P 55	81	P 75	10	H 71	0.014	H 91	0.014
P 16	294.73	P 36	40	P 56	40	P 76	35	H 72	0.014	H 92	0.014
P 17	109.91	P 37	40	P 57	10	H 53	0.014	H 73	0.014	H 93	0.014
P 18	109.91	P 38	55	P 58	35	H 54	0.014	H 74	0.014	H 94	0.014
P 19	109.91	P 39	55	P 59	81	H 55	0.014	H 75	0.014	H 95	0.014
P 20	109.91	P 40	630.39	P 60	40	H 56	0.014	H 76	0.014	H 96	0.014

The quality solution obtained by WOA and its different variants after 100 trials are presented in Table 15 and is compared with Real Coded Genetic Algorithm with improved Muhlenbein mutation (RCGA-IMM) [53]and TVAC-PSO [9]. It can be observed that WOA and its variants perform superior to RCGA-IMM [53] and TVAC-PSO [9] with less CPU time.

Table 15. Optimal schedule for the best cost obtained for Case study 6.

Technique	Best ($/h) Cost	Mean ($/h) Cost	Worst ($/h) Cost	STD	Time(s)
RCGA-IMM [53]	239,896.4082	-	-	-	280.47
TVAC-PSO [9]	239,139.5018	-	-	-	198.25
WOA	166,598.2166	172,476.1561	190,501.0311	0.019690	2.555
RVWOA	166,017.3783	170,507.9580	182,953.4631	0.018518	1.922
LVWOA	165,290.1180	170,003.9993	176,234.0878	0.014683	1.749
SVWOA	165,224.8771	168,008.8607	173,527.3772	0.010526	1.922
EVWOA	164,691.7101	167,690.1798	171,696.4346	0.008756	2.600

Again, it can be said by analyzing the results in Table 15 that proposed EVWOA performs better than WOA, RVWOA, LVWOA, and SVWOA. This shows that by introducing exponentially varying acceleration function in basic WOA, the proposed method performs well and during the computation process, the exploration and exploitation are properly balanced for getting the best optimal quality solution.

Figure 14 compares the best cost of all the above algorithms graphically. The best minimum cost of EVWOA $164,691.71/h is compared to the best minimum cost of TVAC-

PSO [9], i.e., $239,139.5018/h. The hourly savings are calculated and found by EVWOA to be $74,447.7918, which when seen yearly, amounts to $652,162,656. These high figures prove the worthiness of EVWOA for obtaining a good solution, with good standard deviation and a good computation time compared to other different published algorithms. The main objective of including this case study in this paper is to highlight the supremacy of the EVWOA algorithm compared to others. Lesser minimum cost and average cost obtained by EVWOA prove the point. The standard deviation is also very low, i.e., 0.009.

Figure 14. Best Cost Comparison for Case Study 6.

4.6.1. Discussion about Convergence Characteristics

The convergence characteristics obtained by WOA, RVWOA, LVWOA, SVWOA, and EVWOA for best cost vs no. of iterations for this large test system during sample trial are shown in Figure 15. It can be observed from this figure that initially basic WOA explores the search area very well but in the latter part of the iterations, it traps in local optima. However, by incorporating randomly varying acceleration function in basic WOA, during initial iterations, exploration of RVWOA is not proper but in the latter half exploitation is improved but it is not able to get an optimal solution. Next, LVWOA is not performing better because there is an unbalance between local exploration and global exploitation. It can be observed from Figure 15 that during some initial iterations it explores well, but after that, it remains constant till the end of iterations. In SVWOA, a sinusoidally varying acceleration function is used. The basic tendency of the sinusoidal function is that for the initial half, it provides sufficient search space for exploration, and later half this search space is reduced for exploitation. It can be seen from this figure that the exploration of SVWOA is good but still, the exploitation potential remains weak. For improving the efficiency and maintaining the balance between exploration and exploitation, an exponentially varying acceleration function is used in the proposed EVWOA. It can be observed from Figure 15. that by adding exponentially varying acceleration function, EVWOA well explores the search area in a better way during the initial half and in the remaining half, the proposed method getting the best optimal solution because of better exploitation by avoiding the local optima. This proves that the proposed EVWOA obtains better quality solutions by properly managing the balance between exploration and exploitation throughout the search process.

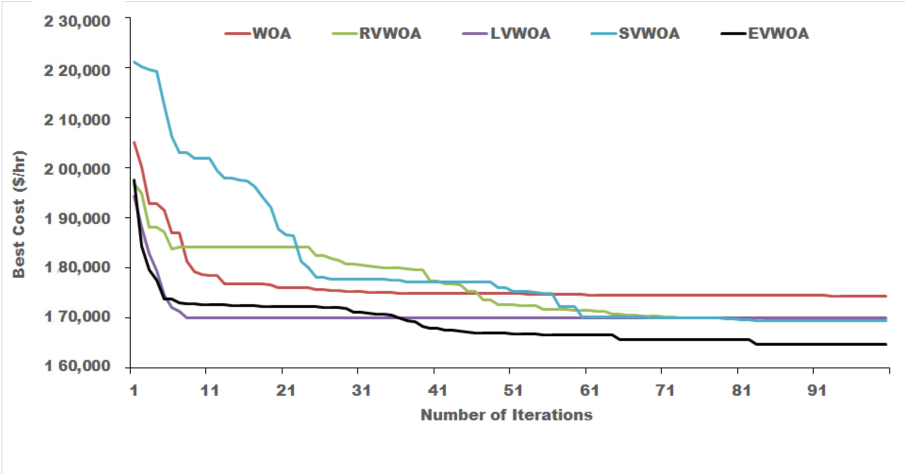

Figure 15. Convergence Curve for Case Study 6 obtained by WOA, RVWOA, LVWOA, SVWOA and EVWOA.

5. Conclusions

The optimal scheduling of non-convex CHP units is a highly complex, non-convex, and hard constrained nonlinear optimization problem. In this paper, an Exponentially Varying WOA (EVWOA) method is proposed and implemented to enhance and improve the performance of basic WOA in terms of convergence, maintaining the balance between global exploration and exploitation throughout the search process, and improving its efficiency. This variation is done by the introduction of an exponentially varying acceleration function in the characteristic equations of the original WOA.

This proposition, as well as the other variations of WOA, namely RVWOA, LVWOA, and SVWOA, are tested on several well-known standard benchmark functions to test their ability and effectiveness in finding the solution. The obtained results are compared to some of the existing optimization algorithms in the literature. It is found that all the variations perform satisfactorily and can get the desired results.

Next, the proposed EVWOA is tested on six small to large different test systems of CHPED problem consisting of several related operational constraints. The simulation results provided by the proposed EVWOA after 100 independent trials are compared with basic WOA, other remaining variants of WOA, and recently published results obtained by different methods. The comparison results show that the proposed EVWOA performs much better than other latest published methods and can obtain a much better optimal quality solution in terms of best minimum cost, mean of cost, and STD of cost in less CPU time. It can also be said that the proposed EVWOA is providing promising results with an efficient constraint handling method for solving the CHPED problem.

Apart from the conventional single-objective CHPED problem, the proposed EVWOA can be used for solving various other power system optimization problems. Some of them to be named are the multi-objective CHP economic emission dispatch (CHPEED), combining economic load dispatch consisting of conventional electrical generator units with renewable energy sources, multi-area economic dispatch, dynamic load dispatch, etc.

Author Contributions: In this research work, all authors have contributed equally in term of technical as well as non-technical point of view. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported through the project numberRSP-2020/278, King Saud University, Riyadh, Saudi Arabia.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Case Study 1 details are referred from Reference [3].

Electrical energy generator units data is as per Table A1.

Table A1. Electrical Energy Generator unit data for Case Study 1.

Unit	a ($/MW2)	b ($/MW)	c ($)	Pmin (MW)	Pmax (MW)
1	0	50	0	0	150

Cogeneration unit data is as per Table A2.

Table A2. Cogeneration unit data for Case Study 1

Unit	a ($/MW2)	b ($/MW)	c ($)	d ($/MWth2)	e ($/MWth)	f ($/MW.MWth)	FOR (P,H)
2	0.0345	14.5	2650	0.030	1.200	0.031	[98.8, 0], [81, 104.8], [215, 180], [247, 0]
3	0.0435	36.0	1250	0.027	0.600	0.011	[44, 0], [44, 15.9], [40, 75], [110.2, 135.6], [125.8, 32.4], [125.8, 0]

Heat energy generator unit data is as per Table A3.

Table A3. Heat energy generator unit data for Case Study 1.

Unit	a ($/MW2)	b ($/MW)	c ($)	Hmin (MWth)	Hmax (MWth)
4	0	23.4	0	0	2695.2

Appendix B

Case Study 2 details are taken from [47].

Electrical energy generator data is as per Table A4.

Table A4. Electrical Energy Generator unit data for Case Study 2.

Unit	h ($/MW3)	a ($/MW2)	b ($/MW)	c ($)	Pmin (MW)	Pmax (MW)
1	0.000115	0.00172	7.6997	254.8863	0	35

Cogeneration unit data is as per Table A5.

Table A5. Cogeneration unit data for Case Study 2.

Unit	a ($/MW2)	b ($/MW)	c ($)	d ($/MWth2)	e ($/MWth)	f ($/MW.MWth)	FOR (P,H)
2	0.0435	36.0	1250	0.027	0.600	0.011	[44, 0], [44, 15.9], [40, 75], [110.2, 135.6], [125.8, 32.4], [125.8, 0]
3	0.1035	34.5	2650	0.025	2.203	0.051	[20, 0], [10, 40], [45, 55], [60, 0]
4	0.072	20	1565	0.02	2.340	0.04	[35, 0], [35, 20], [90, 45], [90, 25], [105, 0]

Heat energy generator data is as per Table A6.

Table A6. Heat energy generator unit data for Case Study 2.

Unit	a ($/MW2)	b ($/MW)	c ($)	Hmin (MWth)	Hmax (MWth)
5	0.038	2.0109	950	0	60

Appendix C

Details of Case Study 3 are referred from ref. [45].

Electrical energy generator data is as per Table A7.

Table A7. Electrical Energy Generator unit data for Case Study 3.

Unit	a ($/MW2)	b ($/MW)	c ($)	d ($)	e (rad/MW)	Pmin (MW)	Pmax (MW)
1	0.008	2	25	100	0.042	10	75
2	0.003	1.8	10	140	0.04	20	125
3	0.0012	2.1	100	160	0.038	30	175
4	0.001	2	120	180	0.037	40	250

Cogeneration unit data is as per Table A8.

Table A8. Cogeneration unit data for Case Study 3.

Unit	a ($/MW2)	b ($/MW)	c ($)	d ($/MWth2)	e ($/MWth)	f ($/MW.MWth)	FOR (P,H)
5	0.0345	14.5	2650	0.030	1.200	0.031	[98.8, 0], [81, 104.8], [215, 180], [247, 0]
6	0.0435	36.0	1250	0.027	0.600	0.011	[44, 0], [44, 15.9], [40, 75], [110.2, 135.6], [125.8, 32.4], [125.8, 0]

Heat energy generator data is as per Table A9.

Table A9. Heat energy generator unit data for Case Study 3.

Unit	a ($/MW2)	b ($/MW)	c ($)	Hmin (MWth)	Hmax (MWth)
7	0.038	2.0109	950	0	2695.2

Transmission line losses for electrical generator units are given by:

$$
\begin{aligned}
B &= \begin{bmatrix} 49 & 14 & 15 & 15 & 20 & 25 \\ 14 & 45 & 16 & 20 & 18 & 19 \\ 15 & 16 & 39 & 10 & 12 & 15 \\ 15 & 20 & 10 & 40 & 14 & 11 \\ 20 & 18 & 12 & 14 & 35 & 17 \\ 25 & 19 & 15 & 11 & 17 & 39 \end{bmatrix} * 10^{-7} \\
B_0 &= [-0.3908\ -0.1297\ 0.7047\ 0.0591\ 0.2161\ -0.6635] * 10^{-3} \\
B_{00} &= 0.056
\end{aligned}
\tag{A1}
$$

Appendix D

Data of Case Study 4 are referred from ref. [24].
Electrical energy generator data is as per Table A10.

Table A10. Electrical Energy Generator unit data for Case Study 4.

Unit	a ($/MW2)	b ($/MW)	c ($)	d ($)	e (rad/MW)	Pmin (MW)	Pmax (MW)
1	0.00028	8.1	550	300	0.035	0	680
2,3	0.00056	8.1	309	200	0.042	0	360
4,5,6,7,8,9	0.00324	7.74	240	150	0.063	60	180
10,11	0.00284	8.6	126	100	0.084	40	120
12,13	0.00284	8.6	126	100	0.084	55	120

Cogeneration unit data is as per Table A11.

Table A11. Cogeneration unit data for Case Study 4.

Unit	a ($/MW2)	b ($/MW)	c ($)	d ($/MWth2)	e ($/MWth)	f ($/MW.MWth)	FOR (P,H)
14,16	0.0345	14.5	2650	0.030	1.200	0.031	[98.8, 0], [81, 104.8], [215, 180], [247, 0]
15,17	0.0435	36.0	1250	0.027	0.600	0.011	[44, 0], [44, 15.9], [40, 75], [110.2, 135.6], [125.8, 32.4], [125.8, 0]
18	0.1035	34.5	2650	0.025	2.203	0.051	[20, 0], [10, 40], [45, 55], [60, 0]
19	0.072	20	1565	0.02	2.340	0.04	[35, 0], [35, 20], [90, 45], [90, 25], [105, 0]

Heat energy generator data is as per Table A12.

Table A12. Heat energy generator unit data for Case Study 4.

Unit	a ($/MW2)	b ($/MW)	c ($)	Hmin (MWth)	Hmax (MWth)
20	0.038	2.0109	950	0	2695.2
21,22	0.038	2.0109	950	0	60
23,24	0.052	3.0651	480	0	120

Appendix E

Details of Case Study 5 are taken from [51].

Electrical energy generator data is as per Table A13.

Table A13. Electrical Energy Generator unit data for Case Study 5.

Unit	a ($/MW2)	b ($/MW)	c ($)	d ($)	e (rad/MW)	Pmin (MW)	Pmax (MW)
1,14	0.00028	8.1	550	300	0.035	0	680
2,3,15,16	0.00056	8.1	309	200	0.042	0	360
4,5,6,7,8,9, 17,18,19,20, 21,22	0.00324	7.74	240	150	0.063	60	180
10,11,23,24	0.00284	8.6	126	100	0.084	40	120
12,13,25,26	0.00284	8.6	126	100	0.084	55	120

Cogeneration unit data is as per Table A14.

Table A14. Cogeneration unit data for Case Study 5.

Unit	a ($/MW2)	b ($/MW)	c ($)	d ($/MWth2)	e ($/MWth)	f ($/MW.MWth)	FOR (P,H)
27,29, 33,35	0.0345	14.5	2650	0.030	1.200	0.031	[98.8, 0], [81, 104.8], [215, 180], [247, 0]
28,30, 34,36	0.0435	36.0	1250	0.027	0.600	0.011	[44, 0], [44, 15.9], [40, 75], [110.2, 135.6], [125.8, 32.4], [125.8, 0]
31,37	0.1035	34.5	2650	0.025	2.203	0.051	[20, 0], [10, 40], [45, 55], [60, 0]
32,38	0.072	20	1565	0.02	2.340	0.04	[35, 0], [35, 20], [90, 45], [90, 25], [105, 0]

Heat energy generator data is as per Table A15.

Table A15. Heat energy generator unit data for Case Study 5.

Unit	a ($/MW2)	b ($/MW)	c ($)	Hmin (MWth)	Hmax (MWth)
39,44	0.038	2.0109	950	0	2695.2
40,41,45,46	0.038	2.0109	950	0	60
42,43,47,48	0.052	3.0651	480	0	120

Appendix F

Case Study 6 data are referred from ref. [24].

Electrical energy generator data is as per Table A16.

Table A16. Electrical Energy Generator unit data for Case Study 6.

Unit	a ($/MW2)	b ($/MW)	c ($)	d ($)	e (rad/MW)	Pmin (MW)	Pmax (MW)
1,14,27,40	0.00028	8.1	550	300	0.035	0	680
2,3,15,16, 28,29,41,42	0.00056	8.1	309	200	0.042	0	360
4,5,6,7,8,9, 17,18,19,20, 21,22,30,31, 32,33,34,35, 43,44,45,46, 47,48	0.00324	7.74	240	150	0.063	60	180
10,11,23,24, 36,37,49,50	0.00284	8.6	126	100	0.084	40	120
12,13,25,26, 38,39,51,52	0.00284	8.6	126	100	0.084	55	120

Cogeneration unit data is as per Table A17.

Table A17. Cogeneration unit data for Case Study 6.

Unit	a ($/MW2)	b ($/MW)	c ($)	d ($/MWth2)	e ($/MWth)	f ($MW.MWth)	FOR (P,H)
27,29, 33,35	0.0345	14.5	2650	0.030	1.200	0.031	[98.8, 0], [81, 104.8], [215, 180], [247, 0]
28,30, 34,36	0.0435	36.0	1250	0.027	0.600	0.011	[44, 0], [44, 15.9], [40, 75], [110.2, 135.6], [125.8, 32.4], [125.8, 0]
31,37	0.1035	34.5	2650	0.025	2.203	0.051	[20, 0], [10, 40], [45, 55], [60, 0]
32,38	0.072	20	1565	0.02	2.340	0.04	[35, 0], [35, 20], [90, 45], [90, 25], [105, 0]

Heat energy generator data is as per Table A18.

Table A18. Heat energy generator unit data for Case Study 6.

Unit	a ($/MW2)	b ($/MW)	c ($)	Hmin (MWth)	Hmax (MWth)
39,44	0.038	2.0109	950	0	2695.2
40,41,45,46	0.038	2.0109	950	0	60
42,43,47,48	0.052	3.0651	480	0	120

References

1. Keirstead, J.; Samsatli, N.; Shah, N.; Weber, C. The impact of CHP (combined heat and power) planning restrictions on the efficiency of urban energy systems. *Energy* **2012**, *41*, 93–103. [CrossRef]
2. Karki, S.; Kulkarni, M.; Mann, M.D.; Salehfar, H. TEfficiency improvements through combined heat and power for on-site distributed generation technologies. *Cogenerat. Distrib. Generat. J.* **2007**, *3*, 19–34. [CrossRef]
3. Guo, T.; Henwood, M.I.; Ooijen, M.V. An algorithm for combined heat and power economic dispatch. *IEEE Trans. Power Syst* **1996**, *11*, 1778–1784. [CrossRef]
4. Rooijers, F.J.; VanAmerongen, R.A.M. Static economic dispatch for co-generation systems. *IEEE Trans. Power Syst* **1994**, *9*, 1392–1398.
5. Abdol Mohammadi, H.R.; Kazemi, A.A Benders decomposition approach for a combined heat and power economic dispatch. *Energy Convers. Manag.* **2013**, *71*, 21–31. [CrossRef]
6. Song, Y.; Chou, C.; Stonham, T. Combined heat and power economic dispatch by improved ant colony search algorithm. *Electric Power Syst. Res.* **1999**, *52*, 115–121. [CrossRef]
7. Beigvand, S.D.; Abdi, H.; La Scala, M. Combined heat and power economic dispatch problem using gravitational search algorithm. *Electric Power Syst. Res.* **2016**, *133*, 160–172. [CrossRef]
8. Adhvaryyu, P.K.; Chattopadhyay, P.K.; Bhattacharjya, A. Application of bio-inspired krill herd algorithm to combined heat and power economic dispatch. In Proceedings of the 2014 IEEE Innovative Smart Grid Technologies-Asia (ISGTASIA), Kuala Lumpur, Malaysia, 20–23 May 2014; Volume 1, pp. 338–343.
9. Mohammadi-Ivatloo, B.; Moradi-Dalvand, M.; Rabiee, A. Combined heat and power economic dispatch problem solution using particle swarm optimization with time varying acceleration coefficients . *Electr. Power Sys. Res.* **2013**, *95*, 9–18. [CrossRef]
10. Ramesh, V. ; Jayabarathi, T.; Shrivastava, N.; Baska, A. A novel selective particle swarm optimization approach for combined heat and power economic dispatch. *Electr. Power Syst. Res.* **2009**, *37*, 1231–1240. [CrossRef]
11. Subbaraj, P.; Rengaraj, R.; Salivahanan, S. Enhancement of combined heat and power economic dispatch using self-adaptive real-coded genetic algorithm. *Appl. Energy* **2009**, *86*, 915–921. [CrossRef]
12. Khorram, E.; Jaberipour, M. Harmony search algorithm for solving combined heat and power economic dispatch problems. *Energy Convers. Manag.* **2011**, *52*, 1550–1554. [CrossRef]
13. Roy, P.K.; Paul, C.; Sultana, S. Oppositional teaching learning-based optimization approach for combined heat and power dispatch. *Int. J. Electr. Power Energy Syst.* **2014**, *57*, 392–403. [CrossRef]
14. Civicioglu, P.; Besdok, E. TA conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms. *Artif. Intell. Rev.* **2013**, *39*, 315–346. [CrossRef]
15. Jena, C.; Basu, M.; Panigrahi, C. Differential evolution with Gaussian mutation for combined heat and power economic dispatch. *Soft Comput.* **2016**, *20*, 681–688. [CrossRef]
16. He, S.; Wu, Q.H. Saunders, J. Group search optimizer: an optimization algorithm inspired by animal searching behaviour. *IEEE Trans. Evolut. Comput.* **2009**, *13*, 973–990.
17. Basu, M. Combined heat and power economic dispatch using opposition-based group search optimization. *Int. J. Electr. Power Energy Syst.* **2015**, *73*, 819–829. [CrossRef]
18. Nguyen, T.T.; Vo, D.N.; Dinh, B.H. Cuckoo search algorithm for combined heat and power economic dispatch. *Int. J. Electr. Power Energy Syst.* **2016**, *81*, 204–214. [CrossRef]
19. Basu, M. Bee colony optimization for combined heat and power economic dispatch. *Expert. Syst. Appl.* **2011**, *38*, 13527–13531. [CrossRef]
20. Jayabarathi, T.; Yazdani, A.; Ramesh, V.; Raghunathan, T. Combined heat and power economic dispatch problem using the invasive weed optimization algorithm. *Front. Energy* **2014**, *8*, 25–30. [CrossRef]
21. Ghorbani, N. Combined heat and power economic dispatch using exchange market algorithm. *Int. J. Electr. Power Energy Syst.* **2016**, *82*, 58–66. [CrossRef]
22. Jayakumar, N. Combined heat and power dispatch by grey wolf optimization. *Int. J. Energy Sect. Manag.* **2015**, *9*, 523–546. [CrossRef]
23. Seyedali, M.; Andrew, L. The Whale Optimization Algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67.
24. Nazari-Heris, M.; Mehdinejad, M.; Mohammadi-Ivatloo, B.; Babamalek-Gharehpetian, G. Combined heat and power economic dispatch problem solution by implementation of whale optimization method. *Neural Comput. Appl.* **2019**, *31*, 421–436. [CrossRef]
25. Ying, L.; Yongquan, Z.; Qifang, L. Levy, Flight Trajectory-Based Whale Optimization Algorithm for Global Optimization. *IEEE Access.* **2017**, *5*, 6168–6186.
26. Gaganpreet, K.; Sankalap, A. Chaotic whale optimization algorithm. *J. Computat. Des. Eng.* **2018**, *5*, 275–284.
27. Guojiang, X.; Jing, Z.; Dongyuan, S.; Yu, H. Parameter extraction of solar photovoltaic models using an improved whale optimization algorithm. *Energy Convers. Manag.* **2018**, *174*, 388–405.
28. Hongping, H.; Yanping, B.; Ting, X. A whale optimization algorithm with inertia weight. *WSEAS Transact. Comput.* **2016**, *15*, 319–326.
29. Hongping, H.; Yanping, B.; Ting, X. Improved whale optimization algorithms based on inertia weights and theirs applications. *Int. J. Circuits Syst. Signal Process.* **2017**, *11*, 12–26.

30. Kaveh, A.; Ilchi Ghazaan, M. Enhanced whale optimization algorithm for sizing optimization of skeletal structures. *Mech. Based Des. Struct. Mach.* **2017**, *45*, 345–365. [CrossRef]
31. Majdi, M.M; Seyedali, M. Hybrid Whale Optimization Algorithm with simulated annealing for feature selection. *Neuro Comput.* **2017**, *260*, 302–312.
32. Diego, O.; Mohamed, A.E.A.; Aboul, E.H.. Parameter estimation of photovoltaic cells using an improved chaotic whale optimization algorithm. *Appl. Energy* **2017**, *200*, 141–154.
33. Mohamed, A.E.A.; Ahmed, A.E.; Aboul Ella. H. Whale Optimization Algorithm and Moth-Flame Optimization for multilevel thresholding image segmentation. *Expert Syst. Appl.* **2017**, *83*, 242–256.
34. Ibrahim, A.; Hossam, F.; Seyedali, M. Optimizing connection weights in neural networks using the whale optimization algorithm. *Soft Comput.* **2018**, *22*, 1–15.
35. Mohamed, A.B.; Gunasekaran, M.; Doaa, E.S.; Seyedali, M. A hybrid whale optimization algorithm based on local search strategy for the permutation flow shop scheduling problem. *Future Generat. Comput. Syst.* **2018**, *85*, 129–145.
36. Ala' M.; Al-Zoubi, H.; Faris, Ja'far, A.; Mohammad, A.H. Evolving Support Vector Machines using Whale Optimization Algorithm for spam profiles detection on online social networks in different lingual contexts. *Knowl. Based Syst.* **2018**, *153*, 91–104. [CrossRef]
37. Amolkumar, N.; Gomathi, J. Hybridization of exponential grey wolf optimizer with whale optimization for data clustering. *Alexandria Eng. J.* **208**, *57*, 1569–1584.
38. Mohamed, A.-B.; Doaa, E.-S.; Arun, K.S. A modified nature inspired meta-heuristic whale optimization algorithm for solving 0–1 knapsack problem. *Int. J. Mach. Learn. Cyber* **2019**, *10*, 495–514.
39. Vinay, K.; Jadoun, N.; Gupta, K.R.; Niazi, A.S. Nonconvex Economic Dispatch Using Particle Swarm Optimization with Time Varying Operators. *Adv. Electric. Eng.* **2014**, *2014*, 13.
40. Jadoun, V.K.; Gupta, N.; Niazi, K.R.; Swarnakar, A. Dynamically Controlled Particle Swarm Optimization for Large Scale Non-Convex Economic Dispatch Problems. *Int. Transact. Electric. Energy Syst.* **2015**, *11*, 3060–3074. [CrossRef]
41. Pandey, V.C.; Jadoun, V.K.; Gupta, N.;Niazi, K.R.; Swarnakar, A. Improved Fireworks Algorithm with Chaotic Sequence Operator for Large-Scale Non-convex Economic Load Dispatch Problem. *Arabian J. Sci. Eng.* **2018**, *43*, 2919–2929. [CrossRef]
42. Ghorbani N.; Babaei E. Exchange market algorithm. *Appl. Soft Comput.* **2014**, *19*, 177–187. [CrossRef]
43. Yazdani, A.; Jayabarathi, T.; Ramesh, V.; Raghunathan, T. Combined heat and power economic dispatch problem using firefly algorithm. *Front. Energy* **2013**, *7*, 133–139. [CrossRef]
44. Song, Y.; Xuan, Q. Combined heat and power economic dispatch using genetic algorithm-based penalty function method. *Electr. Mach. Power Syst.* **1998**, *26*, 363–372. [CrossRef]
45. Vasebi, A.; Fesanghary, M.; Bathaee, S. Combined heat and power economic dispatch by harmony search algorithm. *Int. J. Electr. Power Energy Syst.* **2007**, *29*, 713–719. [CrossRef]
46. Mehdinejad, M.; Mohammadi-Ivatloo, B.; Dadashzadeh-Bonab, R. Energy production cost minimization in a combined heat and power generation systems using cuckoo optimization algorithm. *Energy Effic.* **2016**, *1*, 1–16. [CrossRef]
47. Heidarali, S.; Reza, D.B.; Navid, T.; Mehdi, M. Combined Heat and Power Economic Dispatch Solution Using Iterative Cultural Algorithm. *Int. Conf. Artif. Intell.* **2017**, 191–197.
48. Basu, M. Combined heat and power economic dispatch by using differential evolution. *Electr. Power Compon. Syst.* **2010**, *38*, 996–1004. [CrossRef]
49. Hagh, M.T.; Teimourzadeh, S.; Alipour, M.; Aliasghary, P. Improved group search optimization method for solving CHPED in large scale power systems. *Energy Convers. Manag.* **2014**, *80*, 446–456. [CrossRef]
50. Meng, A.; Mei, P.; Yin, H.; Peng, X.; Guo, Z. Crisscross optimization algorithm for solving combined heat and power economic dispatch problem. *Energy Convers. Manag.* **2015**, *105*, 1303–1317. [CrossRef]
51. Basu, M. Modified particle swarm optimization for non-smooth nonconvex combined heat and power economic dispatch. *Electr. Power Compon. Syst.* **2015**, *43*, 2146–2155. [CrossRef]
52. Basu, M. Group search optimization for combined heat and power economic dispatch. *Int. J. Electr. Power Energy Syst.* **2016**, *78*, 138–147. [CrossRef]
53. Haghrah, A.; Nazari-Heris, M.; Mohammadi-Ivatloo, B. Solving combined heat and power economic dispatch problem using real coded genetic algorithm with improved Mu¨hlenbein mutation. *Appl. Therm. Eng.* **2016**, *99*, 465–475. [CrossRef]