applied sciences

MDPI

Article

Chaotic Evolutionary Programming for an Engineering Optimization Problem

Nirbhow Jap Singh [1], Shakti Singh [1], Vikram Chopra [1], Mohd Asim Aftab [1,*], S. M. Suhail Hussain [2] and Taha Selim Ustun [2]

[1] Electrical and Instrumentation Engineering Department, Thapar Institute of Engineering and Technology, Patiala 147001, India; nirbhow@thapar.edu (N.J.S.); shakti.singh@thapar.edu (S.S.); vikram.chopra@thapar.edu (V.C.)

[2] Fukushima Renewable Energy Institute, AIST (FREA), National Institute of Advanced Industrial Science and Technology (AIST), Koriyama 963-0298, Japan; suhail.hussain@aist.go.jp (S.M.S.H.); selim.ustun@aist.go.jp (T.S.U.)

* Correspondence: asim.aftab@thapar.edu; Tel.: +91-995-649-8473

Abstract: The aim of the current paper is to present a mimetic algorithm called the chaotic evolutionary programming Powell's pattern search (CEPPS) algorithm for the solution of the multi-fuel economic load dispatch problem. In the CEPPS algorithm, the exploration process is maintained by chaotic evolutionary programming, whereas exploitation is taken care off by a pattern search. The proposed CEPPS has two variants based on the Gauss map and the tent map. Seven generalized benchmark test functions and six cases of the multi-fuel economic load dispatch problem are considered for the performance analysis. It is observed from the analysis that the CEPPS solution procedure based on the tent map exhibits superiority to obtain an excellent solution and better convergence characteristics than traditional chaotic evolutionary programming. Further, the performance investigation for the considered economic load dispatch shows that the Gauss map CEPPS solution procedure performs better than the tent map based CEPPS to obtain the solution of the multi-fuel economic dispatch problem.

Keywords: chaotic evolutionary programming; Gauss map; Powell's pattern search; robustness test; tent map

check for updates

Citation: Singh, N. J.; Singh, S.; Chopra, V.; Aftab, M.A.; Hussain, S.M.S.; Ustun, T.S. Chaotic Evolutionary Programming for an Engineering Optimization Problem. *Appl. Sci.* **2021**, *11*, 2717. https://doi.org/10.3390/app11062717

Academic Editor: Giancarlo Mauri and Pierluigi Siano

Received: 15 February 2021
Accepted: 15 March 2021
Published: 18 March 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The electric power system is a complex engineering system. The planning, operation and control of the interconnected electric power system is a challenging task. Economic load dispatch (ELD) of the power system is one such task, which means planning, scheduling, and operating generators in an economical manner. The transmission losses form an inherent part of the economic operation of the power system. Today's ELD problem posses nonlinear behavior, due to imposed equality and inequality constraints [1]. The ELD of a multi-fuel system involves power generators that utilize more than one type of fuel. Depending on the power demand, the generator can switch the fuel type [2]. The ELD problem has been identified as a multimodal problem, which is a challenge to solve. Since the practical problems are multimodal in nature, the gradient approaches are not suitable for them. In this respect, the solution of nonlinear economic dispatch problems has been effectively obtained using random search algorithms irrespective of the shape of the solution hyperspace. Although these heuristic methods provide a faster and reasonable solution, these do not ensure a global optimal solution in a finite time. The complex dispatch and scheduling problems require effective and efficient optimization algorithms for a beneficial solution [3]. In effect, the global optimization algorithms have been extensively used as a solution procedure for ELD problem having a multi-fuel generator.

In the previous decade, global optimization techniques such as genetic algorithm or simulated annealing, which is a form of probabilistic heuristic algorithm, have been used to solve the ELD problem. The other widely accepted stochastic search algorithms are evolutionary programming (EP) [4], particle swarm optimization (PSO) [5], genetic algorithm (GA) [6], simulated annealing (SA), etc. The population based algorithms, also called evolutionary algorithms (EAs), have been widely employed to solve the practical constrained ELD optimization problems. The derivative free mechanism, parallel processing nature, fast convergence rate, performance independent of the hyperspace, etc., are the key factors for the superiority of these methods.

In the 1990s, EP dominated the field of optimization algorithms. The features of the EP mechanism are: firstly, the utilization of real-valued variables and parameters; secondly, mutation and selection are the sole operators, i.e., EP uses a single evolution operator. The computational resources required by EP are much less compared to other EAs; hence, it may result in a smaller computational time. The maturity phase of EP consists of classical EP [7], self-adaptive EP [8], fast EP [9], scaled Gaussian mutation EP [10], EP with the mean of Gaussian and Cauchy mutations with an empirical learning rate [7], EP with Gaussian mutation with empirical learning [11] and EP with the better of Gaussian and Cauchy mutations with an empirical learning rate [12]. The objective of all these modifications is to elevate the limitations of multimodal problem solving [12]. Liu et al. [13] showed empirically that EP with cooperative convolution cab be used to solve large-scale problems with superior performance. Further, a fast EP algorithm based on a Levy probability distribution based mutation operator has shown a performance advantage [14]. It has been observed from the literature survey that the modifications of EP are random number or mutation operator oriented.

Recently, the application of non-linear dynamics has been suggested for the selection of algorithm control parameters [15] and search algorithm tuning [16,17]. A random sequence inheriting features of long periodicity and uniformity is suitable to enhance the search ability of the stochastic search algorithm. The random numbers generated by the chaotic map exhibit ergodicity, non-repetitiveness, and non-linearity and are dynamic in nature [18]. The random number sequence possesses an element of regularity and exhibits sensitive dependence on initial conditions [19]. Since different chaotic maps lead to diverse behavior, the chaos maps are potential alternatives to pseudo random sequences. Gandomi et al. [20] provided an in-depth analysis of different chaos maps as an alternative to conventional pseudo random numbers. The investigation of chaotic bat algorithm using generalized benchmark test functions leads to a conclusion that the use of chaos is advantageous. Similarly, chaotic practical swarm optimization [21], chaotic differential evolution [22], and crisscross differential evolution [23] have been used to solve engineering problems. Hui et al. [24] investigated the performance of chaos based multi-objective evolutionary algorithms and concluded that chaotic maps improved the performance of evolutionary algorithms to solve a problem. An advancement in the field of chaotic numbers involves the application of adaptive symmetry to create chaos [25] and digital chaotic systems [26]. The use of chaos map based random numbers in EP may be advantageous to ensure that the algorithm generates diverse solutions and potentially explores the multimodal objective landscape. An algorithm with these properties can result in better exploration and a better convergence rate [27].

On the basis of the above facts and arguments, it is necessary to empirically investigate the performance of the EP by incorporating a chaotic sequence. Therefore, a hybrid algorithm, chaotic evolutionary programming and pattern search (CEPPS), has been proposed by implementing the following steps:

1. Introduction of the chaotic sequence based population initialization process.
2. A chaotic mutation operator is proposed and employed.
3. A chaos guided tournament selection operator is considered to select better candidates.
4. The Powell's pattern search is applied to enhance the exploitation of the proposed algorithm.

The standard benchmark test function and a practical problem of ELD for a multi-fuel generator problem are used to analyze the performance of CEPPS. The results are compared with the results of available algorithms from the past.

The paper is organized into seven parts. Section 2 presents the formation of the multi-fuel ELD problem. Section 3 presents the mathematical foundation of evolutionary programming algorithms. Section 4 discusses the mathematical foundation of the chaotic EP algorithm. Section 5 presents the details of various generalized benchmark functions and the multi-fuel economic dispatch problem considered in the study. Section 6 presents the numerical results by CEPPS and comparisons with recently published work. Finally, Section 7 concludes the paper.

2. Economic Load Dispatch Problem

The economic load dispatch problem aims to minimize the power generation's cost while satisfying the constraints of expected load demand and the generator's operation. In the case of the multi-fuel load dispatch problem, the power generators have the option of multiple fuels, and each unit represents several piece-wise, quadratic functions reflecting the effect of fuel change. The multiple fuel options and valve point loading effect result in the multimodal and discontinuous nature of the problem. A multi-fuel ED problem is mathematically expressed as follows:

Minimize the operating cost:

$$F(P) = \sum_{j=1}^{N_g} (a_{jm} P_j^2 + b_{jm} P_j + c_{jm} + |d_{jm} \sin e_{jm} (P_{jm}^{min} - P_j)|) \quad (P_{jm}^{min} \leq P_j \leq P_{jm}^{max}) \tag{1}$$

where P_j is the generated real power and $P= [P_1, P_2, \ldots, P_{Ng}]^T$. N_g is the number of generators. a_{jm}, b_{jm}, c_{jm}, d_{jm}, and e_{jm} are the thermal generators' cost coefficients of the $j$thgenerator's m^{th} fuel option. ${P_{jm}}^{min}$ and ${P_{jm}}^{max}$ are the generator's lower and upper limits for the m^{th} fuel option.

The cost objective function is subject to:

(i) The power balance equality constraint:

$$\sum_{j=1}^{N_g} P_j - (P_D + P_L) = 0 \tag{2}$$

(ii) The generator operating limits:

$$P_j^{\min} \leq P_j \leq P_j^{\max} \qquad (j = 1, 2, \ldots, N_g) \tag{3}$$

(iii) The ramp rate limit.

- As generation increases:

$$P_j - P_j^0 \leq UR_j \qquad (j = 1, 2, \ldots, N_g) \tag{4}$$

- As generation decreases:

$$P_j^0 - P_j \leq DR_j \qquad (j = 1, 2, \ldots, N_g) \tag{5}$$

(iv) Prohibited operating zone constraint:

$$\begin{aligned} &P_j^{\min} \leq P_j \leq P_{j,1}^L \quad (j = 1, 2, \ldots, N_g) \\ &P_{j,i-1}^U \leq P_j \leq P_{j,i}^L \quad (i = 1, 2, \ldots, N_{zj}; j = 1, 2, \ldots, N_g) \\ &P_{j,Nzj}^U \leq P_j \leq P_j^{\max} \quad (j = 1, 2, \ldots, N_g) \end{aligned} \tag{6}$$

where P_D and P_L are the forecasted demand and transmission loss of the network, respectively. N_{zj} the is number of prohibited zones of the jth generator. P_j^0 is the previously generated power. UR_j and DR_j are the up-ramp limit and down-ramp limit of the jth generator. $P_{j,i}^L \; and \; P_{j,i}^U$ are the lower and upper range of the jth prohibited zone respectively of the jth generator

3. Evolutionary Programming

The evolution process of EP has two steps: (i) mutate the current population; (ii) select best one out of the current solution and the mutated solution. A real-valued vector (x_i, η_i) is used for each individual in the population. Here, x_i is the decision variable, and η_i is the associated strategy parameter. The generation of a new solution x_i^k at kth iteration is based on the mutation operator. The selection operation decides the survival of a solution in the future generation population. Mathematically, this concept is illustrated as below:

$$x_{ij}^{k+1} = x_{ij}^k + \eta_{ij}^k N_j(0,1) \qquad (i = 1, 2, \ldots, N_P, j = 1, 2, \ldots, N_D) \tag{7}$$

$$\eta_{ij}^{k+1} = \eta_{ij}^k exp(\chi_1 N(0,1) + \chi_2 N_j(0,1)) \quad (i = 1, 2, \ldots, N_P, j = 1, 2, \ldots, N_D) \tag{8}$$

where $N_j(0,1)$ is the normally distributed random number with mean zero and one as the standard deviation, generated for the jth component. N_p is the population size and N_D represents the components for $i$the individual. The parameters χ_1 and χ_2 are given by $\left(\sqrt{2N_D - 1}\right)^{-1}$ and $\left(\sqrt{2\sqrt{N_D - 1}}\right)^{-1}$ [28].

The strategy parameter η_{ij}^k and offspring x_{ij}^k are updated using the repetitious process mentioned above along with the selection operation to decide the parents for the $k + 1^{th}$ generation. The process repeats until the termination criteria are satisfied.

4. Proposed Algorithm

The proposed algorithm blends the chaotic evolutionary programming (CEP) approach and Powell's pattern search (PS), to solve the various benchmark test problems. CEP aims at the exploration, whereas PS focuses on the exploitation of the search area around the solution located by CEP. The process is explained in the following subsections.

4.1. Chaotic Evolutionary Programming

In chaotic evolutionary programming, a pre-selected chaotic sequence is used to replace the conventional random number generator. The Gauss map and the tent map are used in the iterative process. Therefore, in an N_D-dimensional search space, an ith individual vector (x_i, η_i) is a possible solution of the problem. Mathematically, the evolution concept of chaotic EP to generate offspring is illustrated as follows:

$$x_{ij}^{t+1} = x_{ij}^k + \eta_{ij}\phi_j(0,1) \quad (i = 1, 2, \ldots, N_P, j = 1, 2, \ldots, N_D) \tag{9}$$

$$\eta_{ij}^{t+1} = \eta_{ij}^k exp(\chi^{t+1}\phi(0,1) + \chi\phi_j(0,1)) \quad (i = 1, 2, \ldots, N_P, j = 1, 2, \ldots, N_D) \tag{10}$$

where $\phi_j(0,1) \in [0,1]$ is a chaos generated random number for the jth individual.

In this work, the Gauss map and the tent map sequence are used as these have the advantage of a uniform distribution and are one-dimensional maps. These maps generate numbers that help the algorithm converge faster.

A Gauss map based chaotic sequence is represented as [20]:

$$\phi_{n+1} = \begin{cases} 0 & ; \phi_n < 0 \\ \frac{1}{\phi_n} - [\frac{1}{\phi_n}] & ; otherwise \end{cases} \tag{11}$$

and the tent map chaotic sequence follows [20]:

$$\phi_{n+1} = \begin{cases} \frac{\phi_n}{0.7} & ; \phi_n < 0.7 \\ \frac{10}{3}(1 - \phi_n) & ; otherwise \end{cases} \tag{12}$$

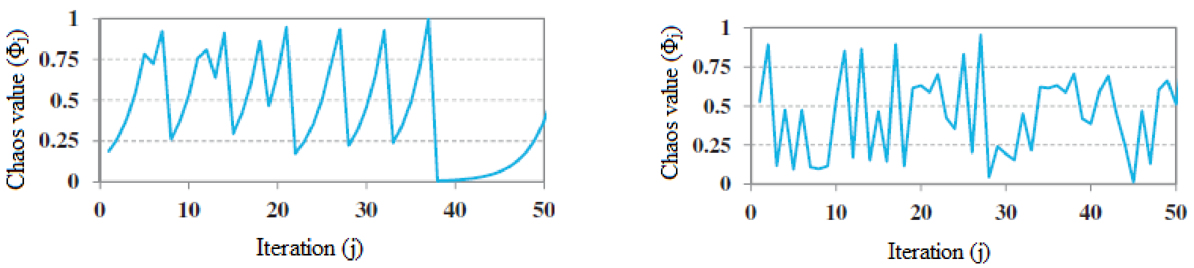
The time series plot of the tent map and the Gauss map are shown in Figure 1a,b, respectively. The selected maps generates chaos numbers that are well in the acceptable range of the EP algorithm. The correctness of the range of chaotic sequences is another deciding factor for the selection of the chaotic map [24]. After the offspring generation at the kth generation, the combined population (parents and offspring) compete with each other to survive in the $(k+1)$th generation. For the selection, an individual's score ζ_i in the stochastic competition is given by:

$$\zeta = \sum_{n=1}^{N_P} w_n \qquad (i = 1, 2, \ldots, N_P) \tag{13}$$

with:

$$w_n = \begin{cases} 1 & ; \phi_1 < \frac{f_m}{f_i + f_t} \\ 0 & ; otherwise \end{cases} \tag{14}$$

where f_m is the fitness of the mth randomly selected competitor in the combined population; $m = int(2L\phi_2 + 1)$; f_i is the fitness value of the ith individual; $\phi_1, \phi_2 \in [0, 1]$ are random numbers generated by the chaotic sequence.



(**a**) Tent map (**b**) Gauss map

Figure 1. Time series plot of the tent and Gauss maps, respectively [20].

4.2. Powell's Pattern Search Method

Powell's method is a numerical technique based on a direct search to obtain the solution of the problem in hand. If the quality of the solution improves, the newly generated solution will be a success. The pattern-move accelerates the search process in an ascertained direction. The PS is represented mathematically as follows:

Initialize N_D-dimensional linearly independent search direction S:

$$S_{ij} = \begin{cases} 1; i = j \\ 0; otherwise \end{cases} \quad (i, j = 1, 2, \ldots, N_D) \tag{15}$$

In each direction, a unidirectional search is performed using x_i as the best point described as follows:

$$x_j = \begin{cases} x_j + \lambda_i^* S_{ij} & ; f(x_j + \lambda_i^* S_{ij}) < f(x_j) \\ x_j & ; otherwise \end{cases} \tag{16}$$

where $\lambda_i^* \in [\lambda_i^{min}, \lambda_i^{max}]$ is a randomly generated step size.

A pattern search direction is given by following equation:

$$S_{ij} = x_j - x_{j-N} \qquad (j = 1, 2, \ldots, N_D) \tag{17}$$

5. Simulation Test Problems

In order to prove the capability and efficacy of the CEPPS solution approach, the generalized benchmark test functions, as well as standard real-world problems related to power system operation are undertaken. The Gauss and tent map sequences are applied to investigate their behavior while implementing CEP and CEPPS. Using the Gauss map and the tent sequence, the CEP method is termed CEP-1 and CEP-2, respectively. A similar notation is used for the CEPPS procedures. The generalized test functions are taken from [29] and are described in the following subsection.

5.1. Generalized Test Functions

The standard test functions considered to prove the ability to solve optimization problems are non-differentiable, non-separable, discontinuous, and multimodal in nature and are described below:

1. Griewank function: This is described mathematically as:

$$F_1(x) = \sum_{i=1}^{n} \frac{x_i^2}{4000} - \prod cos(\frac{x_i}{\sqrt{i}}) \tag{18}$$

subject to $(-600 \leq x_i \leq 600)$.

The above function is multimodal, non-separable, differentiable, scalable, and continuous in nature. The global minimum is $f(x^*) = 0,\ x^* = f(0, \ldots, 0)$.

2. Rastrigin's function: This is described mathematically as:

$$F_2(x) = \sum_{i=1}^{n} [x_i^2 - 10cos(2\pi x_i)] \tag{19}$$

subject to $(-10 \leq x_i \leq 10)$.

The above function is non-differentiable and highly multimodal. The global minimum is located at $x^* = f(0, \ldots, 0),\ f(x^*) = 0$.

3. Rosenbrock's function: This is described mathematically as:

$$F_3(x) = \sum_{i=1}^{n-1} [(x_i - 1)^2 + 100(x_{i+1} - x_i^2)^2] \tag{20}$$

subject to $(-30 \leq x_i \leq 30)$.

The global minimum is located at $x^* = f(1, \ldots, 1),\ f^* = 0$.

4. Schwefel 2.22 function: This is described mathematically as:

$$F_4(x) = \sum_{i=1}^{n} |x_i| + \prod_{i=1}^{n} |x_i| \tag{21}$$

subject to $(-10 \leq x_i \leq 10)$.

5. Sphere function: This is one of the simplest of De Jong's functions. It is described mathematically as:

$$F_5(x) = \sum_{i=1}^{n} x_i^2 \tag{22}$$

subject to $(-10 \leq x_i \leq 10) \qquad (i = 1, 2, \ldots, n)$.

The above function is differentiable, continuous, scalable, separable, and unimodal in behavior. The global minimum is $x^* = f(0, \ldots, 0),\ f(x^*) = 0$.

6. Step function: This is described mathematically as:

$$F_6(x) = \sum_{i=1}^{n} \lfloor |x_i| \rfloor \tag{23}$$

subject to $\quad (-10 \leq x_i \leq 10)$.

The above function is unimodal, discontinuous, separable, non-differentiable, and scalable. The global minimum is $x^* = f(0, \ldots, 0) = 0,\ f(x^*) = 0$.

7. Step 2 function: This is described mathematically as:

$$F_7(x) = \sum_{i=1}^{n} \lfloor |x_i + 0.5| \rfloor \tag{24}$$

subject to $\quad (-10 \leq x_i \leq 10)$

The above function is uni-modal, separable, discontinuous, non-differentiable, and scalable in nature. The global minimum is $x^* = f(0.5, \ldots, 0.5) = 0,\ f(x^*) = 0$.

5.2. Multi-Fuel Economic Load Dispatch Problem

The practical problem considered is the ELD problem having multi-fuel generators. The generators have additional associated complexities such as prohibited operating zone, valve point loading, etc., as shown in Table 1. In total, six cases of the multi-fuel economic dispatch problem, listed in Table 1, are used in this study.

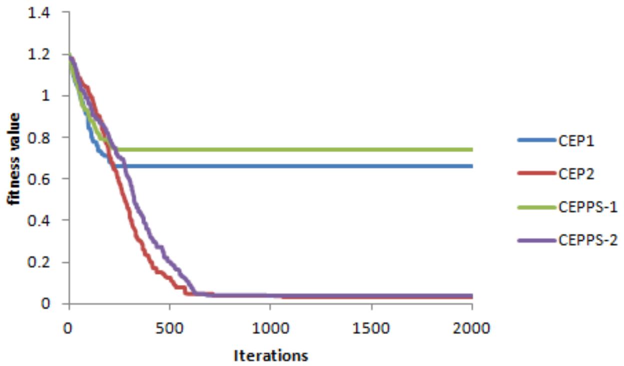
Table 1. Multi-fuel system undertaken for the study, $P_D = 2700$ MW [30].

Case	Valve Point Loading	Ramp Rate	Prohibited Operating Zone	Transmission Loss
1	×	×	×	×
2	✓	×	×	×
3	×	×	✓	×
4	×	×	×	✓
5	✓	×	✓	✓
6	×	×	✓	✓

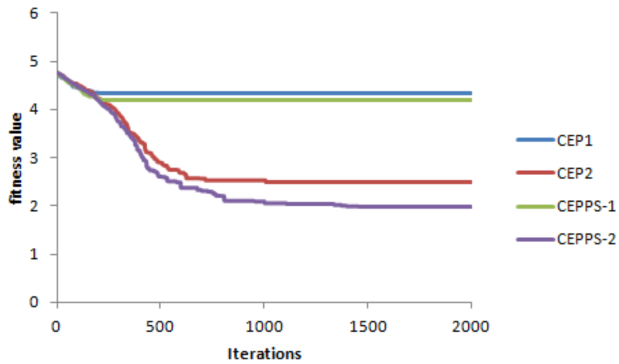
6. Results and Discussion

The standard test functions were considered to analyze the ability of the CEPPS solution approach. The results obtained by the proposed CEPPS variants were compared with that obtained from CEP. To compare the performance, seven standard test functions were solved using CEP and CEPPS variants. Table 2 shows the worst (W), average (A), and best (B) results for each test function after 30 trials.

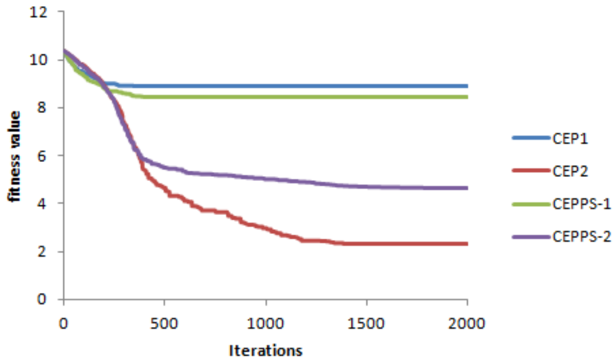
The comparison of the convergence plots for the investigated standard test functions is shown in Figure 2 using the log scale. Figure 2a represents the convergence behavior of Griewank's function, which exhibits that CEPPS-1 and CEP-1 converge quickly in the starting phase, but result in premature convergence. However, CEPPS-2 and CEP-2 result in much better solutions, although initially, these were slow to converge. Finally, CEPPS-2 obtains the best solution compared to the others.



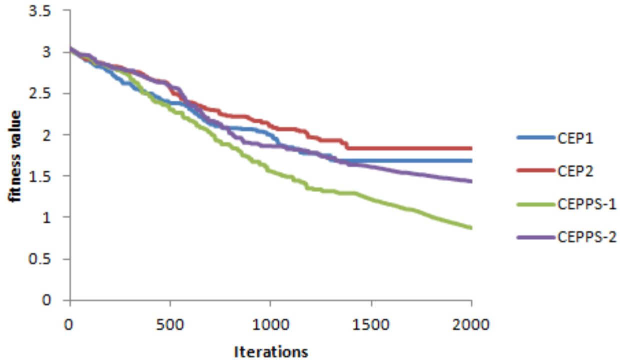
(**a**) Griewank function



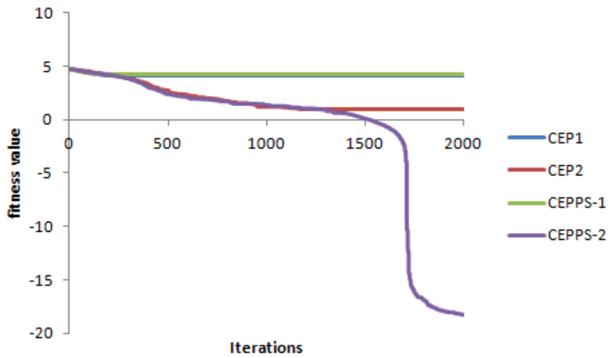
(**b**) Rastrigin function



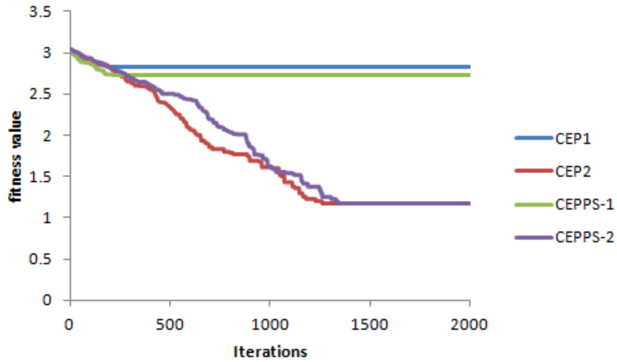
(**c**) Rosenbrock function



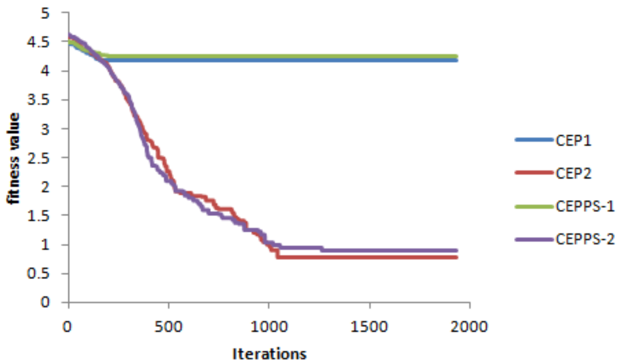
(**d**) Schwefel's 2.22 function



(**e**) Sphere function



(**f**) Step function



(**g**) Step 2 function

Figure 2. Convergence behaviors of the different test functions.

The convergence behavior of Rastrigin's function is depicted in Figure 2b. It shows the same trend as in the case of Griewank's function. However, for the Rastrigin function, it is found that CEPPS-2 performs significantly better than CEP-2 and provides a better solution.

The convergence behavior in Figure 2c shows that in the case of Rosenbrock's function, both CEPPS-1 and CEP-1 result in premature convergence, while the performance of CEP-2 is significantly better than CEPPS-2. Furthermore, the behavior of Schwefel's function in Figure 2d shows that CEPPS-1 results in the best solution of the problem, whereas both CEP-1 and CEP-2 are the worst performers. The result of the sphere function in Figure 2e shows that the CEPPS-2 algorithm has the best convergence rate. Furthermore, the convergence behavior comparison in Figure 2f,g for the step and Step 2 function, respectively, indicates that CEP-2 and CEPPS-2 have better performance than the others. Even CEP-2 performs better than CEPPS-2 for the Step 2 function.

Thus, the convergence behavior plots show that the CEP-2 and CEPPS-2 variants perform better than CEP-1 and CEPPS-1. Their performance is judged either on the basis of the convergence rate or solution quality. It is also observed that CEPPS-1 performs better on some of the standard test problems, whereas CEP-2 and CEPPS-2 show supremacy for other standard test functions.

Table 2. Performance analysis of the fitness value of generalized benchmark test functions [29].

Test Function	Fitness	CEP-1	CEP-2	CEPPS-1	CEPPS-2
Griewank function	Worst	4.61	1.08	11.05	9.29×10^{-1}
	Average	4.61	1.08	11.05	9.29×10^{-1}
	Best	4.61	1.08	5.48	0.01×10^{-1}
Rastrigin function	Worst	21,893.57	305.44	40,041.02	97.11
	Average	21,893.57	305.44	40,041.02	61.70
	Best	21,893.57	305.44	15,691.13	61.70
Rosenbrock function	Worst	7.54×10^{8}	222.46	1.00×10^{10}	43,304.03
	Average	7.54×10^{8}	22.36	1.00×10^{10}	7300.95
	Best	7.54×10^{8}	22.36	2.91×10^{8}	7300.95
Schwefel's 2.22 function	Worst	48.16	68.45	7.57	27.71
	Average	48.16	68.45	7.57	27.71
	Best	48.16	68.45	7.57	27.71
Sphere function	Worst	13,094.38	8.23	50,471.00	4.79×10^{-19}
	Average	13,094.38	8.23	50471.00	9.40×10^{-20}
	Best	13,094.38	8.23	18,642.14	9.40×10^{-20}
Step function	Worst	670.00	37.00	969.00	28.00
	Average	670.00	37.00	969.00	28.00
	Best	670.00	15.00	532.00	15.00
Step 2 function	Worst	15,349.00	6.00	39,277.00	8.00
	Average	15,349.00	6.00	39,277.00	5.00
	Best	15,349.00	6.00	17,381.00	5.00

The system depicted in Table 1 was solved for ELD by applying CEPPS-2. This system was considered for six different cases. In all the cases, the population size N_P were fixed at 50, and the number of iterations IT^{max} was fixed at 2000. The cost comparison was performed with other published work and is presented in Table 3. The results of biogeography based optimization (BBO), composite PSO (CPSO), GA with mutation update (CGA-MU), differential evolution (DE), the improved gravitational search algorithm (IGA), the improved genetic algorithm with multiplier update (IGA-MU), the enhanced augmented Hopfield neural network (ELHN), the hybrid of DE and BBO (DEBBO), krill herd optimization (KHA), the quadratic programming augmented Hopfield neural network (QP-ALHN), PSO, and synergic predator pey optimization (SPPO) were considered for the comparison of the solution quality.

The comparison of the results show that in Case 1, CEPPS-1 and CEPPS-2 had a lesser cost in comparison to all others, and both obtained a solution with a cost of 623.75 \$/h. Similarly, in Case 2, the cost for the generation schedule by CEPPS variants was better than the others except CPSO, where it was comparable. The cost was a little bit higher by 0.05 \$/h and 0.06 \$/h than CPSO. Further, in Cases 4 and 5, the generation cost for both CEPPS variants was better than the other contenders. Moreover, in these two cases, CEPPS-1 had the advantage of obtaining a lesser generation cost schedule. Lastly, in Case 6, CEPPS-2 obtained a better quality generation schedule than SPPO, whereas CEPPS-1 was able to obtain a somewhat deteriorated solution. To summarize, both CEPPS variants obtained a power generation schedule either with lesser than or comparable generation cost as the others. In addition to this, the generation cost comparison between CEPPS-1 and CEPPS-2 shows that CEPPS-1 was able to obtain a generation schedule with a lesser cost.

Table 3. Test Power System 1, comparison of economic load dispatch (ELD) ($P_D = 2700$ MW). BBO, biogeography based optimization; DE, differential evolution; ELHN, enhanced augmented Hopfield neural network; IGA, improved gravitational search algorithm; KHA, krill herd optimization; QP-ALHN, quadratic programming augmented Hopfield neural network.

Algorithm	Cost (\$/h)					
	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6
BBO [31]	624.51	–	–	–	–	–
CPSO [32]	–	623.82	–	–	–	–
CGA-MU [33]	623.80	624.71	–	–	–	–
DE [33]	623.80	624.46	–	–	–	–
DEBBO [34]	624.51	–	–	–	–	–
ELHN [35]	624.51	–	–	–	–	–
IGA [30]	624.51	–	–	–	–	–
IGA-MU [30]	623.80	624.51	–	–	–	–
KHA [36]	624.51	–	–	–	–	–
PSO [33]	623.80	624.24	–	–	–	–
QP-ALHN [37]	623.80	–	624.32	–	–	–
SPPO	623.80	623.82	624.32	700.29	700.77	700.48
CEPPS-1	623.75	623.87	623.76	699.70	699.54	704.94
CEPPS-2	623.75	623.88	623.77	699.77	699.73	700.60

7. Conclusions

In this manuscript, chaotic evolutionary programming and pattern search have been proposed as a solution for the economic load dispatch problem of multi-fuel power generators. The CEPPS employs achaotic map based stochastic population initialization. Secondly, the chaos based mutation and selection operators have been proposed. In order to enhance the exploration capability, Powell's pattern is introduced in the search process under stochastic control. The well accepted chaos map, viz. the Gauss map and the tent map based CEPPS variants, were analyzed for performance. Computer simulations have been performed on generalized test functions and a load dispatch problem to verify the effectiveness of the CEPPS variants. The numerical results of the generalized benchmark problems reveal that tent map based CEPPS has a better search capability to find the optimal solution in the majority of the test problems. The experimentation on the ELD problem of multi-fuel generators clearly indicates that CEPPS variants have the ability to provide very competitive results in terms of generation cost as compared with the existing literature. Lastly, it has been observed that the Gauss map based CEPPS variants have an advantage in providing a better solution for multi-fuel systems.

Author Contributions: All authors contributed equally to this work. All authors read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Dhillon, J.; Kothari, D. *Power System Optimization*; PHI Learning Pvt. Ltd.: New Delhi, India, 2004.
2. Singh, N.J.; Dhillon, J.; Kothari, D. Synergic predator-prey optimization for economic thermal power dispatch problem. *Appl. Soft Comput.* **2016**, *43*, 298–311. [CrossRef]
3. Singh, N.J.; Dhillon, J.; Kothari, D. Surrogate worth trade-off method for multi-objective thermal power load dispatch. *Energy* **2017**, *138*, 1112–1123. [CrossRef]
4. Jayabarathi, T.; Jayaprakash, K.; Jeyakumar, D.; Raghunathan, T. Evolutionary programming techniques for different kinds of economic dispatch problems. *Electr. Power Syst. Res.* **2005**, *73*, 169–176. [CrossRef]
5. Gaing, Z.-L. Particle swarm optimization to solving the economic dispatch considering the generator constraints. *IEEE Trans. Power Syst.* **2003**, *18*, 1187–1195. [CrossRef]
6. Chen, P.-H.; Chang, H.-C. Large-scale economic dispatch by genetic algorithm. *IEEE Trans. Power syst.* **1995**, *10*, 1919–1926. [CrossRef]
7. Chellapilla, K.; Fogel, D.B. Two new mutation operators for enhanced search and optimization in evolutionary programming. In *Optical Science, Engineering and Instrumentation*; International Society for Optics and Photonics: Bellingham, WA, USA, 1997; pp. 260–269.
8. Fogel, L.J.; Fogel, D.B. A preliminary investigation on extending evolutionary programming to include self-adaptation on finite state. *Informatica* **1994**, *18*, 387–398.
9. Yao, X.; Liu, Y. Fast evolutionary programming. *Evol. Program.* **1996**, *3*, 451–460.
10. Yang, H.-T.; Yang, P.-C.; Huang, C.-L. Evolutionary programming based economic dispatch for units with non-smooth fuel cost functions. *IEEE Trans. Power Syst.* **1996**, *11*, 112–118. [CrossRef]
11. Back, T.; Schwefel, H.P. An overview of evolutionary algorithms for parameter optimization. *Evolu. Comput.* **1993**, *1*, 1–23. [CrossRef]
12. Yao, X.; Liu, Y.; Lin, G. Evolutionary programming made faster. *IEEE Trans. Evolu. Comput.* **1999**, *3*, 82–102.
13. Liu, Y.; Yao, X.; Zhao, Q.; Higuchi, T. Scaling up fast evolutionary programming with cooperative co evolution. In Proceedings of the Congress on Evolutionary Computation, Seoul, Korea, 27–30 May 2001; Volume 2, pp. 1101–1108.
14. Lee, C.-Y.; Yao, X. Evolutionary programming using mutations based on the levy probability distribution. *IEEE Trans. Evol. Comput.* **2004**, *8*, 1–3. [CrossRef]
15. Thangaraj, R.; Pant, M.; Chelliah, T.R.; Abraham, A. Opposition based chaotic differential evolution algorithm for solving global optimization problems. In Proceedings of the Fourth World Congress on Nature and Biologically Inspired Computing, Mexico City, Mexico, 5–9 November 2012; pp. 1–15.
16. Dos Santos Coelho, L.; Mariani, V.C. Combining of chaotic differential evolution and quadratic programming for economic dispatch optimization with valve-point effect. *IEEE Trans. Power Syst.* **2006**, *21*, 989–996. [CrossRef]
17. He, D.; Dong, G.; Wang, F.; Mao, Z. Optimization of dynamic economic dispatch with valve-point effect using chaotic sequence based differential evolution algorithms. *Energy Conver. Manag.* **2011**, *52*, 1026–1032. [CrossRef]
18. Bharti, K.K.; Singh, P.K. Chaotic gradient artificial bee colony for text clustering. *Soft Comput.* **2016**, *20*, 1113–1126. [CrossRef]
19. Zhang, Z.; Wang, T.; Liu, X. Melt index prediction by aggregated rbf neural networks trained with chaotic theory. *Neuro Comput.* **2014**, *131*, 368–376. [CrossRef]
20. Gandomi, A.H.; Yang, X.-S. Chaotic bat algroithm. *J. Compt. Sci.* **2014**, *5*, 224–232. [CrossRef]
21. Chuang, L.-Y.; Hsiao, C.-J.; Yang, C.-H. Chaotic particle swarm optimization for data clustering. *Expert Syst. Appl.* **2011**, *38*, 62–66. [CrossRef]
22. Kumar, S.; Mandal, K.K.; Chakraborty, N. Optimal DG placement by multi-objective opposition based chaotic differential evolution for techno-economic analysis. *Appl. Soft Comput.* **2019**, *78*, 70–83 [CrossRef]
23. Kaur, M.; Dhillon, J.S.; Kothari, D.P. Crisscross differential evolution algorithm for constrained hydro thermal scheduling. *Appl. Soft. Comput.* **2020**, *93*, 1–19. [CrossRef]
24. Lu, H.; Wang, X.; Fei, Z.; Qiu, M. The Effects of Using Chaotic Map on Improving the Performance of Multi-objective Evolutionary Algorithms. *Math. Probl. Eng.* **2014**, *2014*, 924652. [CrossRef]
25. Tutueva, A.V.; Nepomuceno, E.G.; Karimov, A.I.; Andreev, V.S.; Butusov, D.N. Adaptive chaotic maps and their application to pseudo-random numbers generation. *Chaos Solitons Fractals* **2020**, *133*, 109615. [CrossRef]
26. Nepomuceno, E.G.; Lima, A.M.; Arias-García, J.; Perc, M.; Repnik, R. Minimal digital chaotic system. *Chaos Solitons Fractals* **2019**, *120*, 62–66. [CrossRef]
27. Chen, F.; Huang, G.; Fan, Y.; Liao, R. A nonlinear fractional programming approach for environmental economic power dispatch. *Int. J. Electr. Power Energy Syst.* **2016**, *78*, 463–469. [CrossRef]

28. Fogel, D.B. Applying evolutionary programming to selected traveling salesman problems. *Cybern. Syst.* **1993**, *24*, 27–36. [CrossRef]
29. Jamil, M.; Yang, X.-S. A literature survey of benchmark functions for global optimisation problems. *Int. J. Math Model. Numer. Optim.* **2013**, *4*, 150–194. [CrossRef]
30. Chiang, C.-L. Improved genetic algorithm for power economic dispatch of units with valve-point effects and multiple fuels. *IEEE Trans. Power Syst.* **2005**, *20*, 1690–1699. [CrossRef]
31. Bhattacharya, A.; Chattopadhyay, P.K. Solving complex economic load dispatch problems using bio geography-based optimization. *Expert Syst. Appl.* **2010**, *37*, 3605–3615. [CrossRef]
32. Park, J.-B.; Jeong, Y.-W.; Shin, J.-R.; Lee, K.Y. An improved particle swarm optimization for non convex economic dispatch problems. *IEEE Trans. Power Syst.* **2010**, *25*, 156–166. [CrossRef]
33. Barisal, A. Dynamic search space squeezing strategy based intelligent algorithm solutions to economic dispatch with multiple fuels. *Int. J. Electr. Power Energy Syst.* **2013**, *45*, 50–59. [CrossRef]
34. Bhattacharya, A.; Chattopadhyay, P.K. Hybrid differential evolution with bio geography-based optimization for solution of economic load dispatch. *IEEE Trans. Power Syst.* **2010**, *25*, 1955–1964. [CrossRef]
35. Vo, D.N.; Ongsakul, W. Economic dispatch with multiple fuel types by enhanced augmented Lagrange Hopfield network. *Appl. Energy* **2012**, *91*, 281–289. [CrossRef]
36. Mandal, B.; Roy, P.K.; Mandal, S. Economic load dispatch using krill herd algorithm. *Int. J. Electr. Power Energy Syst.* **2014**, *57*, 1–10. [CrossRef]
37. Dieu, V.N.; Schegner, P. Augmented lagrange hopfeld network initialized by quadratic programming for economic dispatch with piece wise quadratic cost functions and prohibited zones. *Appl. Soft Comput.* **2013**, *13*, 292–301. [CrossRef]