Emperor Penguin Optimization Algorithm for solving Multi-Fuel Non-Convex Economic Load Dispatch Problems

M Balasubbareddy[1], Divyanshi Dwivedi[2]

Department of Electrical and Electronics Engineering, Chaitanya Bharathi Institute of technology, Hyderabad

Abstract

This paper presents an effective use of Emperor Penguin Optimization Algorithm (EPO) which basically mimics the huddle behavior of emperor penguin. This algorithm dwelt to solve the economic load dispatch problems considering multi-fuel generation fuel cost function and emission as objective functions which are to be optimized while satisfying the equality and in-equality constraints. The proposed algorithm has been examined and analyzed on IEEE 30-bus test system to validate its performance and effectiveness in comparison to other existing algorithms.

Keywords: *Emperor Penguin Optimization Algorithm (EPO), multi-fuel non-convex cost function, multi-fuel emission function, Economic Load Dispatch Problems*

1. Introduction

Economic load dispatch problem deals to utilize the power system economically by minimizing the generation fuel cost with consideration of operating and physical constraints. Basically in a generating plant multiple fuels including coal, oil, diesel and natural gas are present simultaneously for maintaining continuity in generating electrical power at lower cost as compared to the generating stations possessing only a single fuel. Considering multi-fuels, this paper solves the economic load dispatch problems by optimizing the non-convex cost function and emission functions.

Recently, many meta-heuristics algorithms have been developed by researchers which can efficiently solve the economic dispatch problem with proper handling the non-linear, constraint bounded functions as compared to the classical approaches[1]-[3] which are used in last decade. These heuristic algorithms including Genetic Algorithm (GA) [4], Evolutionary Programming (EP) [5], Biogeography Based Optimization (BBO) [6], Moth Flame Optimization (MFO) [7], Criss-Cross Algorithm (CCA) [8], Fruit Fly Algorithm (FFA) [9], Ant Colony Optimization (ACO) [10], Ant Lion Optimization (ALO) algorithm [11], and many more had proven their robustness in solving the economic load dispatch problems.

Similarly, a recently developed algorithm named as Emperor Penguin Optimization (EPO) Algorithm has been considered which mimics the huddle mechanism of emperor penguin. As it is already proven in Ref. [12] that, EPO efficiently gives global best solution in less computation time. Thus, this paper proposes the implementation of EPO for solving the economic load dispatch problems of standard IEEE-30 bus system. Obtained results validate the performance of considered algorithm in comparison to the other algorithms.

2. Problem Formulation

Optimization Problem deals to solve the steady state problem of electric power systems through minimizing the objective functions with the consideration of constraints simultaneously. Mathematically OPF is represented by:

Min $F_a(x,y) \qquad \forall a = 1,2,.......p$

Subject to: $k(x,y) = 0$,

$l(x,y) \leq 0$

where, 'k' and 'l' are the equality and inequality constraints respectively, 'x' is the state vector of dependent variables and 'y' is the control vector of system and p is the total number of objectives functions.

The state vector may be represented by:

$$x^T = [P_{G,1}, V_{l,1}....V_{l,NL}, Q_{G,1}....Q_{G,NG}, S_{l,1}....S_{l,NT}]$$

The control vector may be represented by:

$$y^T = [P_{G,2}.....P_{G,NB}, V_{G,1}....V_{G,NB}, Q_{SH,1}....Q_{SH,NC}, T_1....T_T]$$

where $P_{G,1}$ is the real power, $V_{l,1}$ is the load bus voltage, , $Q_{G,1}$ is the reactive power of generator, $S_{l,1}$ is the apparent power of generator $V_{G,1}$ is the generator voltage of slack bus. NL, NG, NT, NC and T are the total number of PQ buses, PV buses, transmission lines, shunt compensators and off-nominal tap transformers respectively.

2.1. Objective Functions

In this paper, two objective functions are minimized, which are mathematically expressed below:

a. Multi-fuel non-convex cost objective

Usually, a generating station possesses different types of fuel including coal, fossil fuel, oil and gas for generation. Considering system with multi-fuel, the non-convex generation fuel cost function can be represented as shown in **Figure 1.**

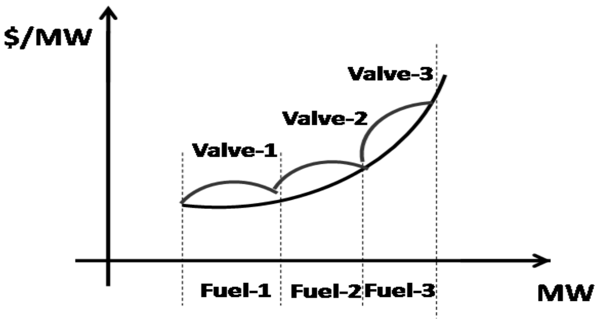

Figure 1. Representation of Multi-fuel generation fuel cost function

The multi-fuel cost function can be formulated as:

$$F_1 = \left(\sum_{i=1}^{2} F_i(P_i) \right) + \left(\sum_{i=3}^{NG} F_i(P_i) \right) \qquad (1)$$

$$F_i(P_i) = \begin{cases} a_{i1}P_i^2 + b_{i1}P_i + c_{i1} + \left| e_{i1} \times \sin\left(f_{i1} \times \left(P_i^{\min} - P_i \right) \right) \right| & ; \; P_i^{\min} \le P_i \le P_i^1 \\ a_{i2}P_i^2 + b_{i2}P_i + c_{i2} + \left| e_{i2} \times \sin\left(f_{i2} \times \left(P_i^{\min} - P_i \right) \right) \right| & ; \; P_i^1 \le P_i \le P_i^2 \\ \ldots \\ a_{ik}P_i^2 + b_{ik}P_i + c_{ik} + \left| e_{ik} \times \sin\left(f_{ik} \times \left(P_i^{\min} - P_i \right) \right) \right| & ; \; P_i^{k-1} \le P_i \le P_i^{\max} \end{cases}$$

where $a_{ik}, b_{ik}, c_{ik}, e_{ik}, f_{ik}$ are the fuel cost-coefficients of the i^{th} unit with valve-point effects for fuel type k.

b. Multi-fuel emission objective

The emission for multi-fuel generating units can be defined as

$$F_2 = \left(\sum_{i=1}^{2} E_i(P_i) \right) + \left(\sum_{i=3}^{N_G} E_i(P_i) \right) \qquad (2)$$

$$E_i(P_i) = \begin{cases} \alpha_{i1} + \beta_{i1}P_{G_i} + \gamma_{i1}P_{G_i}^2 + \xi_{i1}\exp^{\left(\lambda_{i1}P_{G_i}\right)} & ; \; P_i^{\min} \le P_i \le P_i^1 \\ \alpha_{i2} + \beta_{i2}P_{G_i} + \gamma_{i2}P_{G_i}^2 + \xi_{i2}\exp^{\left(\lambda_{i2}P_{G_i}\right)} & ; \; P_i^1 \le P_i \le P_i^2 \\ \ldots \\ \alpha_{ik} + \beta_{ik}P_{G_i} + \gamma_{ik}P_{G_i}^2 + \xi_{ik}\exp^{\left(\lambda_{ik}P_{G_i}\right)} & ; \; P_i^{k-1} \le P_i \le P_i^{\max} \end{cases}$$

where $\alpha_{ik}, \beta_{ik}, \gamma_{ik}, \xi_{ik}$ and λ_{ik} are emission coefficients of the i^{th} generator for fuel type k.

2.2. Constraints

The equality and in-equality constraints are as follows:

a. Equality constraints

$$\sum_{m=1}^{NGB} P_{G,m} - P_D - P_L = 0 \, , \quad \sum_{m=1}^{NGB} Q_{G,m} - Q_D - Q_L = 0$$

b. Inequality Constraints

(i). Generator constraints

$V_{G,m}^{min} \leq V_{G,m} \leq V_{G,m}^{max}$ and

$Q_{G,m}^{min} \leq Q_{G,m} \leq Q_{G,m}^{max} \qquad \forall m \in NG$

(ii). Voltage at bus and discrete transformer tap settings

$V_{G,m}^{min} \leq V_{G,m} \leq V_{G,m}^{max}$ and

$T_{m}^{min} \leq T_{m} \leq T_{m}^{max} \qquad \forall m \in T$

(iii). Active power generation limits

$P_{G,m}^{min} \leq P_{G,m} \leq P_{G,m}^{max} \qquad \forall m \in NG$

(iv). Reactive power supply by the capacitor banks

$Q_{SH,m}^{min} \leq Q_{SH,m} \leq Q_{SH,m}^{max} \qquad \forall m \in NC$

(v). Transmission line loadings

$S_{l,m} \leq S_{l,m}^{max} \qquad \forall m \in NT$

3. Emperor Penguin Optimization Algorithm

In Antarctic winter, these emperor penguins congregate together to keep them warm as shown in **Figure 2,** and this huddling mechanism is considered for developing the algorithm.

Figure 2. Huddling mechanism of Emperor Penguin

Basically, this huddling mechanism of emperor penguins occurs in four phases:

• Huddling boundary for emperor penguins will be generated.

• Temperature around huddle will be calculated.

• Distance between the emperor penguins will be determined.

• Effective mover will be relocated.

3.1. Steps for mathematically modeling of huddling mechanism of emperor penguins

3.1.1. Generation of huddle boundary

Initially huddle boundary is generated randomly in accordance to the wind flow around it. Usually, emperor penguins huddle themselves in a polygon shaped grid.

Let us assume ϕ as the velocity of wind and Ψ be the gradient.

$$\Psi = \nabla\phi \tag{3}$$

Complex potential can be generated by combining λ with it.

$$F_P = \phi + j\lambda \tag{4}$$

where F_P is an analytical function on the polygon plane.

3.1.2. Temperature around the huddle

The temperature profile is accountable for exploration and exploitation in this algorithm. It is considered that when radius of polygon is greater than 1 then temperature T is assumed to be equal to 0 otherwise T is assumed to be equal to 1. The temperature can be formulated as:

$$T_P = \left(T - \frac{Max\_iter}{t - Max\_iter}\right) \tag{5}$$

$$\begin{cases} 0, \text{ if } r > 1 \\ 1, \text{ if } r < 1 \end{cases}$$

$T =$

where, t and Max_iter are the current and maximum number of iterations and r is the radius of polygon.

3.1.3. Distance between the emperor penguins

Initially best optimal solution (emperor penguin) is computed and then its distance with other emperor penguin is calculated. The other emperor penguins will update their positions according to current best optimal solution which is mathematically defined as follows:

$$\vec{D}_{EP} = Abs(S(\vec{M}).\vec{P}(t) - \vec{N}.\vec{P}_{EP}(t)) \quad (6)$$

where, $\vec{D}_{EP}$ is the distance between the best and other emperor penguin, $\vec{M}$ and $\vec{N}$ helps in avoiding the collision among emperor penguins, $\vec{P}$ and $\vec{P}_{EP}$ are the position vectors of the best and other emperor penguin respectively and S is responsible to move emperor penguins towards the best emperor penguin.

$$S(\vec{M}) = \left(\sqrt{f.e^{-t/l} - e^{-t}}\right)^2 \quad (7)$$

where, f and l maintains exploration and exploitation and their values is in the range of [2,3] and [1.5,2] respectively.

$$\vec{M} = (A \times (T_P) + P_G(Accuracy) \times R()) - T_P \quad (8)$$

$$\vec{N} = R() \quad (9)$$

$$P_G(Accuracy) = Abs(\vec{P} - \vec{P}_{EP}) \quad (10)$$

where, A maintains gap between the emperor penguins to avoid collision which is considered to be equal to 2, $P_G(Accuracy)$ is the grid accuracy obtained by comparing the difference between emperor penguins and $R()$ is the random function which generates value between [0,1].

3.1.4. Relocation of effective mover

As, other emperor penguins follow the best selected emperor penguin in a given search space, thus to update the next position following equation is considered:

$$\vec{P}_{EP}(t+1) = \vec{P}(t) - \vec{M}.\vec{D}_{EP} \quad (11)$$

where, $\vec{P}_{EP}(t+1)$ represents the next updated position of emperor penguin.

3.2. Pseudo code and flowchart for the EPO algorithm

The steps of SHO are summarized as follows:

START

Step 1: Read bus data, line data and generation data of considered power system.
Step 2: Initialize the parameters of the proposed algorithm such as number of spotted hyenas and maximum number of iterations.

Step 3: Initialize vectors T_P, $\vec{M}$, $\vec{N}$, S and r.

Step 4: Calculate the fitness of each search agents and identify the first best search agent $\vec{P}$.
while(t<Max_iter)
for (each $\vec{P}_{EP}$)
Step 5: Calculate T_P using Eq. (5).
Step 6: Calculate the distance between the best and other search agents using Eqs. (6) - (10).
end for

Step 7: Update the position of other search agents using Eq. (11).
Step 8: Check the boundary limit and then adjust it.
Step 9: Update the fitness function accordance to the updated position and identify the best optimal solution.
Step 10:t=t+1
end while

Step 11: Return best optimum value obtained.
END

4. Results and Analysis

In Ref. [12], it is already proven that EPO algorithm provides better results as compared to other algorithms discussed in literature. It is also observed that EPO handles various types of constraints very efficiently and provides better solutions. The EPO algorithm has been verified on IEEE-30 bus system by solving the economic dispatch problems. Generally, this system consists of 6 generators which are located on the buses 1, 2, 5, 8, 11 and 13, four tap changing transformers installed between the buses 6-9,6-10, 4-12 and 27-28 and two shunt capacitors installed at buses 10 and 24. Relevant data is taken from Ref. [13].

4.1. Minimization of multi-fuel non-convex cost function

The formulated objective function given in Eq. (1) is optimized using EPO algorithm. The corresponding results are tabulated in **Table 1**, from which it can be seen that the generation fuel cost obtained using EPO algorithm is 647.7310 \$/h which is less than the 667.1426 \$/h obtained by UDTPSO. Furthermore, obtained result is compared with other algorithms in **Table 2.**

Table 1. Comparison of formulated multi-fuel generation fuel cost function

Variables	Multi- fuel generation fuel cost, \$/h	
	UDTPSO [14]	EPO
PG1, MW	139.9829	139.98271
PG2, MW	50	54.98827
PG5, MW	25.79842	23.89929

PG8, MW	25	31.88881
PG11, MW	19.93026	20.84097
PG13, MW	30	18.72352
VG1, p.u.	1.048757	1.07749
VG2, p.u.	1.029419	1.06254
VG5, p.u.	1.007824	1.03514
VG8, p.u.	1.012906	1.04826
VG11, p.u.	1.012993	1.00894
VG13, p.u.	1.027766	1.04193
Tap 6-9, p.u.	0.980952	0.94993
Tap 6-10, p.u.	0.955207	1.02891
Tap 4-12, p.u.	0.960662	0.99242
Tap 28-27, p.u.	1.011175	0.99477
Qc 10, p.u.	21.17225	14.44321
Qc 24, p.u.	16.25719	10.78569
Generation fuel cost, $/h	**667.1426**	**647.7310**
Emission (ton/h)	0.227909356	0.23456

Table 2. Summary and validation of results for multi-fuel generation fuel cost objective minimization

S. No.	Algorithms	Multi-fuel generation fuel cost $/h
1	BBO [6]	647.7645
2	EP [5]	649.67
3	MDE [15]	648.356
4	DE [14]	650.822
5	EPO	**647.731**

Further we can also justify the performance of proposed algorithm by considering the convergence curve shown in **Figure 3.** We can clearly observe that the convergence starts with lesser value and final result obtained very early as compared to UDTPSO.

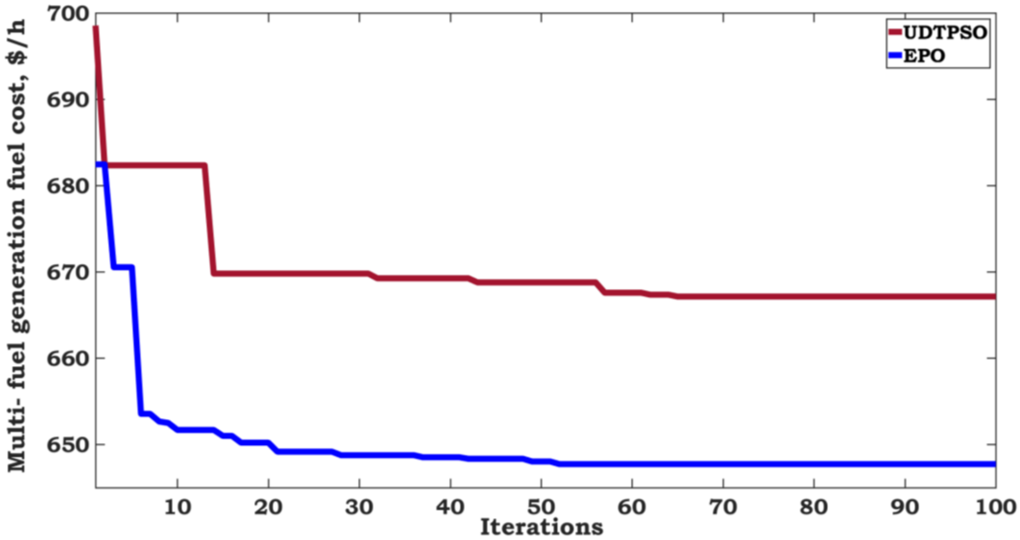

Figure 3. Convergence curve of multi-fuel generation fuel cost, $/h

4.2. Minimization of multi-fuel emission function

The formulated objective function given in Eq. (2) is optimized using EPO algorithm. The corresponding results are tabulated in **Table 3**, from which it can be seen that the emission obtained using EPO algorithm is 0.18621814 ton/h which is less than the ton/h obtained by UDTPSO and convergence curve is shown in **Figure 4.**

Table 3. Comparison of formulated multi-fuel emission function

Variables	Multi-fuel emission, ton/h	
	UDTPSO [14]	EPO
PG1, MW	104.8481	77.36788609
PG2, MW	50	54.96933413
PG5, MW	49	50
PG8, MW	25	35
PG11, MW	25	30
PG13, MW	35	40
VG1, p.u.	0.985061	1.029658984
VG2, p.u.	0.979254	0.968163671
VG5, p.u.	0.980714	1.000212381
VG8, p.u.	0.971643	1.0113926
VG11, p.u.	1.05	1.040342721
VG13, p.u.	1.009539	1.015997273
Tap 6-9, p.u.	0.981196	1.039926264
Tap 6-10, p.u.	0.9	1.069905056
Tap 4-12, p.u.	0.958114	0.965979296
Tap 28-27, p.u.	0.942157	0.998504736

Qc 10, p.u.	13.97391	27.4228281
Qc 24, p.u.	20.83055	6.474612997
Generation fuel cost, $/h	775.2209	828.5704231
Emission (ton/h)	**0.199144**	**0.18621814**

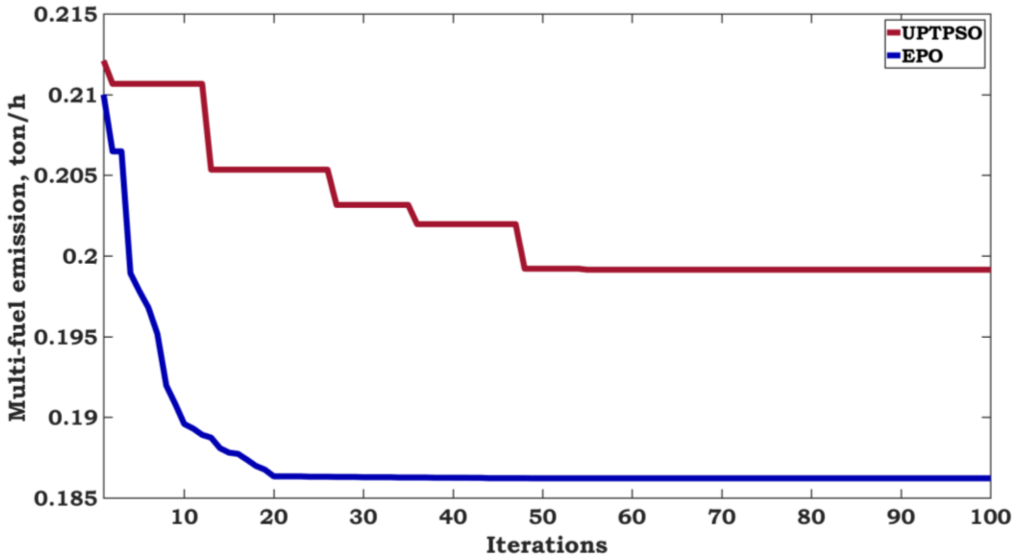

Figure 4. Convergence curve of multi-fuel emission, ton/h

5. Conclusion

In this paper, a novel algorithm is implemented for solving the economic dispatch problem named as Emperor Penguin Optimization (EPO) algorithm which is being developed by considering the natural huddle mechanism of emperor penguins. This algorithm is used to optimize the generation fuel cost and emission for the IEEE-30 bus system possessing multi-fuels as input to it and it has been observed that EPO algorithm yields better solutions as compared to other existing algorithms. We can also conclude that generation fuel cost and emission got minimized when generating plants possess multi-fuels as compared to the generating plants possessing only single fuel. EPO also results in better convergence and attain optimal values in less number of iteration. Thus, this proves the effectiveness and robustness of proposed algorithm for multi-fuel system. In future, this algorithm can be implemented to solve the multi-objective problems.

References

[1] *El-Hawary, J.A.M.M.E., Adapa, R, "A review of selected optimal power flow literature to 1993. Part i: nonlinear and quadratic programming approaches", IEEE transactions on power systems, Vol. 14, (1999A), No. 1.*

[2] *El-Hawary, J.A.M.M.E., Adapa, R., "A review of selected optimal power flow literature to 1993. Part ii: newton, linear programming and interior point methods", IEEE Transactions on Power Systems, 14(1), (1999b) 105–111.*

[3] *David I. Sun, Bruce Ashley, Brian Brewer, A. Hughes, William Tinney, "Optimal power flow by newton approach, IEEE Transaction on Power Apparatus and Systems", Vol. PAS-103, (1984) No.10.*
[4] *D. Devaraj, B. Yegnanarayana, "Genetic-algorithm-based optimal power flow for security enhancement", IEE Proc.-Gener. Transm. Distrib., Vol. 152, No. 6, (2005).*
[5] *H. Pulluri, R. Naresh, V. Sharma, "An enhanced self-adaptive differential evolution based solution methodology for multi objective optimal power flow", Appl. Soft Comput. 54 (2017) 229–245.*
[6] *A. Bhattacharya, P.K. Chattopadhyay, "Applications of biogeography-based optimization to solve different optimal power flow problems", IET generation, transmission and distribution (2010).*
[7] *Seyedali Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm", Knowledge-Based Systems, Volume 89, (2015), 228-249.*
[8] *An-bo Meng, Yu-cheng Chen, Hao Yin, Si-zhe Chen, Crisscross Optimization Algorithm and its application, Elsevier, Knowledge-Based Systems, Volume 67, September 2014, Pages 218-229.*
[9] *Bo Xing, Wen-Jing Gao, "Fruit Fly Optimization Algorithm", Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms, (2013), 167-170.*
[10] *Yun-He Hou, Yao-Wu Wu, Li-Juan Lu, Xin-Yin Xion, "Generalized ant colony optimization for economic dispatch of power systems", International Conference on Power System Technology, (2002).*
[11] *Divyanshi Dwivedi, M Balasubbareddy, "Optimal Power Flow using Hybrid Ant Lion Optimization Algorithm", Pramana Research Journal, Volume 9, Issue 2, (2019).*
[12] *Gaurav Dhiman, Vijay Kumar, "Emperor Penguin Optimizer: A Bio-inspired Algorithm for Engineering Problems', Knowledge-Based Systems, (2018).*
[13] *M. A. Abido, "Optimal Power Flow Using Tabu Search Algorithm", Electric Power Components and Systems Volume 30, (2002), Issue 5.*
[14] *Chintalapudi V.Suresh, Sirigiri Sivanagaraju, "Analysis and effect of multi-fuel and practical constraints on economic load dispatch in the presence of Unified Power Flow Controller using UDTPSO", Ain Shams Engineering Journal, Volume 6, Issue 3, (2015).*
[15] *Samir Sayah, Khaled Zehar, "Modified differential evolution algorithm for optimal power flow with non-smooth cost functions", Energy Conversion and Management, Volume 49, Issue 11, (2008), 3036-3042.*