

Power loss reduction by chaotic based predator-prey brain storm optimization algorithm

Kanagasabai Lenin

Department of Electrical and Electronics Engineering, Prasad V. Potluri Siddhartha Institute of Technology, India

Article Info

Article history:

Received Jul 19, 2019

Revised Feb 19, 2020

Accepted Mar 3, 2020

Keywords:

Chaotic predator-prey brain storm optimization algorithm
Optimal reactive power
Transmission loss

ABSTRACT

In this paper chaotic predator-prey brain storm optimization (CPB) algorithm is proposed to solve optimal reactive power problem. In this work predator-prey brain storm optimization position cluster centers to perform as predators, consequently it will move towards better and better positions, while the remaining ideas perform as preys; hence get away from their adjacent predators. In the projected CPB algorithm chaotic theory has been applied in the modeling of the algorithm. In the proposed algorithm main properties of chaotic such as ergodicity and irregularity used to make the algorithm to jump out of the local optimum as well as to determine optimal parameters CPB algorithm has been tested in standard IEEE 57 bus test system and simulation results show the projected algorithm reduced the real power loss considerably.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Kanagasabai Lenin,
Department of Electrical and Electronics Engineering,
Prasad V.Potluri Siddhartha Institute of Technology,
Kanuru, Vijayawada, Andhra Pradesh-520007, India.
Email: gklenin@gmail.com

1. INTRODUCTION

The main objective of optimal reactive power problem is to minimize the real power loss and bus voltage deviation. To till date various methodologies has been applied to solve the optimal reactive power problem. The key aspect of solving reactive power problem is to reduce the real power loss. Previously many types of mathematical methodologies [1-6] have been utilized to solve the reactive power problem, but they lack in handling the constraints to reach global optimization solution. In the next level various types of evolutionary algorithms [7-15] has been applied to solve the reactive power problem. This paper proposes chaotic predator-prey brain storm optimization (CPB) algorithm to solve optimal reactive power problem. Brain storm optimization algorithm commonly uses grouping, replacing, creating, crossing, and selecting operators to generate new-fangled ideas which grounded on the present ideas, in order to perk up the ideas in all generation in order to reach the optimal solution. K-mean clustering method is utilized to group the N ideas into M clusters in the grouping operator. In this work predator-prey brain storm optimization position cluster centers to perform as predators, consequently it will move towards better and better positions, while the remaining ideas perform as preys; hence get away from their adjacent predators.

Finally cluster centers can maintain the most excellent individuals of the swarm and moving in the direction of the global best position, but at the same time the prey operation avert the swarm from getting trapped into local optimum solution. In the projected CPB algorithm chaotic theory has been applied in the modeling of the algorithm. In the proposed algorithm main properties of chaotic such as ergodicity and irregularity used to make the algorithm to jump out of the local optimum as well as to determine optimal parameters. Proposed CPB algorithm has been tested in standard IEEE 57 bus test system and simulation results show the projected algorithm reduced the real power loss effectively.

Journal homepage: <http://ijape.iaescore.com>

2. PROBLEM FORMULATION

Reduction real power loss is the key goal of the work and the objective function has been written as follows:

$$F = P_L = \sum_{k \in \text{Nbr}} g_k (V_i^2 + V_j^2 - 2V_i V_j \cos \theta_{ij}) \quad (1)$$

Voltage deviation mathematically written as,

$$F = P_L + \omega_v \times \text{Voltage Deviation} \quad (2)$$

$$\text{Voltage Deviation} = \sum_{i=1}^{N_{pq}} |V_i - 1| \quad (3)$$

Constraint (equality);

$$P_G = P_D + P_L \quad (4)$$

Constraints (inequality);

$$P_{\text{gslack}}^{\min} \leq P_{\text{gslack}} \leq P_{\text{gslack}}^{\max} \quad (5)$$

$$Q_{gi}^{\min} \leq Q_{gi} \leq Q_{gi}^{\max}, i \in N_g \quad (6)$$

$$V_i^{\min} \leq V_i \leq V_i^{\max}, i \in N \quad (7)$$

$$T_i^{\min} \leq T_i \leq T_i^{\max}, i \in N_T \quad (8)$$

$$Q_c^{\min} \leq Q_c \leq Q_c^{\max}, i \in N_C \quad (9)$$

3. CHAOTIC PREDATOR-PREY BRAIN STORM OPTIMIZATION ALGORITHM

Inside the searching space a set of “ N_c ” ideas are arbitrarily engendered. Brain storm optimization algorithm population (BSO) population is defined as, $X = \{x_i = [x_{i1}, \dots, x_{id}] | x_i \in A, 1 \leq i \leq N_c\}$ in this x_i symbolize the i th idea of the Brain storm optimization algorithm population, $A = R^D$ indicate the idea in solution space, N_c - population size. Preliminary population $X(0)$ and the n th iteration population denoted as $X(n)$. Fitness value $f(x_i)$ is computed for evaluated idea. Brain storm optimization algorithm [16, 17] commonly uses grouping, replacing, creating, crossing, and selecting operators to generate new-fangled ideas which grounded on the present ideas, in order to perk up the ideas in all generation in order to reach the optimal solution. K-mean clustering method is utilized to group the N ideas into M clusters in the grouping operator. In order to engender new-fangled idea $y_i = [y_{i1}, y_{i2}, \dots, y_{id}]$, ($1 \leq i \leq N_c$). Brain storm optimization algorithm population first verify whether to generate the new-fangled idea y_i based on one or two chosen clusters. New-fangled idea is generated by:

$$y_{i,d} = x_d + \xi_d \times N(\mu, \sigma)_d \quad (10)$$

$$x_d = \begin{cases} x_{i,d} & \text{"1" cluster} \\ \omega_1 x_{i1,d} + \omega_2 x_{i2,d} & \text{2cluster} \end{cases} \quad (11)$$

$$\xi = \text{logsig} \left(\frac{0.50 \times \text{iteration}_{\text{maximum}} - i}{k} \right) \times \text{random} (0,1) \quad (12)$$

Once the new-fangled idea y_i has been formed, a crossover between new-fangled one and the previous one is conducted [16, 17]. Through crossover, x'_i, y'_i are engendered together both the previous and newly formed one are computed then the previous one is swap by the most excellent one. For “ N_c ” time’s new-fangled idea is created creating for completion of one generation. Once end criterion satisfied then Brain storm optimization algorithm procedure stops, or else it go to the subsequent generations to replicate the grouping, replacing, creating, crossing, and select procedure [17].

In this work predator-prey brain storm optimization position cluster centers to perform as predators, consequently it will move towards better and better positions, while the remaining ideas perform as preys; hence get away from their adjacent predators. Finally cluster centers can maintain the most excellent

individuals of the swarm and moving in the direction of the global most excellent position, but at the same time the prey operation avert the swarm from getting trapped into local optimum solution. Then, the (10) can be replaced by:

$$y_{predator,i,d} = x_d + \xi_d \times N(\mu, \sigma)_d + \omega_{predator}(x_{gbest,d} - x_d) \quad (13)$$

$$y_{prey,i,d} = x_d + \xi_d \times N(\mu, \sigma)_d - P_a \text{sgn}(x_{center,d} - x_d) e^{-b|x_{center,d} - x_d|} \quad (14)$$

“ P ” is a binary variable which determine about the status of the prey; flee or not; $\omega_{predator}$ -weight factor of the predator operator; a , b -factors used to measure the complexity of fleeing.

$$a = x_{span} \quad (15)$$

$$b = \frac{100}{x_{span}} \quad (16)$$

In the projected CPB algorithm chaotic theory has been applied in the modeling of the algorithm. In the proposed algorithm main properties of chaotic such as ergodicity and irregularity used to make the algorithm to jump out of the local optimum as well as to determine optimal parameters.

$$ch_{n+1} = 4ch_n(1 - ch_n) \quad (17)$$

At each generation end, chaotic search will be introduced to the exploration in the neighborhood of the present best solution to prefer superior solution for subsequent generation. Through this when local best is reached then stopping will be avoided and also, reaching the optimal solution time will be reduced.

Step a : Parameters are initialized.

Step b : Assessment of all ideas, then record the most excellent one as the global most excellent idea. In the interim, by k -means clustering algorithm, cluster the N_c ideas into M clusters; subsequently grade the ideas in each cluster and record the most excellent idea as cluster center in every cluster.

Step c : Comparison will be done with Probability to replace the cluster center, when arbitrary value between 0 and 1 is smaller, and then arbitrarily choose a cluster center to be swap by an arbitrarily engendered idea; or else, not anything.

Step d : Comparison will be done with probability to select one cluster, when arbitrary value between 0 and 1 is smaller, subsequently choose one cluster; or else, pick two clusters.

Step e : Comparison will be done with probability to select the center of the one selected when arbitrary value between 0 and 1 is smaller, subsequently choose cluster center and go to step f; or else, choose further ideas and move to step g.

Step f : With reference to $y_{predator,i,d} = x_d + \xi_d \times N(\mu, \sigma)_d + \omega_{predator}(x_{gbest,d} - x_d)$ and the most excellent idea, modernize the cluster center (s), and subsequently move to step h.

Step g : With reference to $y_{prey,i,d} = x_d + \xi_d \times N(\mu, \sigma)_d - P_a \text{sgn}(x_{center,d} - x_d) e^{-b|x_{center,d} - x_d|}$ modernize the ideas with propensity of stirring away from the adjoining cluster centers.

Step h : Recently engendered idea crossovers with the current idea to engender two more ideas. Then Compare the four ideas, and the most excellent one will be retained and recorded as the new-fangled individual.

Step i : In the region of the most excellent solution parameters carry out the chaotic exploration with reference to $ch_{n+1} = 4ch_n(1 - ch_n)$ subsequent to alter the parameters ranges into (0, 1). Amongst the produced series of ideas, choose the most excellent one and employ it to swap the previous finest idea.

Step j : When “ N_c ” ideas have been modernized, then go to step k. or else move to step d.

Step k : After assessing the N_c ideas, modernize the cluster center.

Step l : When present number of iterations is less than maximum number of iterations, then move to step b or else the algorithm is stopped and the most excellent idea is determined as the most excellent solution.

4. SIMULATION STUDY

Proposed CPB algorithm has been tested in IEEE 57 bus system [18]. Table 1 shows the constraints of control variables, Table 2 shows the limits of reactive power generators and comparison results are

presented in Table 3. Figure 1 shows the comparison of real power loss and Figure 2 indicate about the real power loss reduction in percentage.

Table 1. Constraints of control variables of IEEE 57 system

Variables type	Minimum value (PU)	Maximum value (PU)
Generator voltage	0.95	1.1
Transformer tap	0.9	1.1
VAR source	0	0.20

Table 2. Constrains of reactive power generators of IEEE 57 system

Variables	Q Minimum (PU)	Q Maximum (PU)
1	-140	200
2	-17	50
3	-10	60
6	-8	25
8	-140	200
9	-3	9
12	-150	155

Table 3. Simulation results of IEEE-57 system

Control variables	Base case	MPSO [19]	PSO [19]	CGA [19]	AGA [19]	CPB
VG 1	1.040	1.093	1.083	0.968	1.027	1.019
VG 2	1.010	1.086	1.071	1.049	1.011	1.012
VG 3	0.985	1.056	1.055	1.056	1.033	1.018
VG 6	0.980	1.038	1.036	0.987	1.001	1.009
VG 8	1.005	1.066	1.059	1.022	1.051	1.010
VG 9	0.980	1.054	1.048	0.991	1.051	1.029
VG 12	1.015	1.054	1.046	1.004	1.057	1.031
Tap 19	0.970	0.975	0.987	0.920	1.030	0.914
Tap 20	0.978	0.982	0.983	0.920	1.020	0.913
Tap 31	1.043	0.975	0.981	0.970	1.060	0.917
Tap 35	1.000	1.025	1.003	NR*	NR*	1.010
Tap 36	1.000	1.002	0.985	NR*	NR*	1.020
Tap 37	1.043	1.007	1.009	0.900	0.990	1.012
Tap 41	0.967	0.994	1.007	0.910	1.100	0.919
Tap 46	0.975	1.013	1.018	1.100	0.980	1.023
Tap 54	0.955	0.988	0.986	0.940	1.010	0.931
Tap 58	0.955	0.979	0.992	0.950	1.080	0.930
Tap 59	0.900	0.983	0.990	1.030	0.940	0.945
Tap 65	0.930	1.015	0.997	1.090	0.950	1.056
Tap 66	0.895	0.975	0.984	0.900	1.050	0.912
Tap 71	0.958	1.020	0.990	0.900	0.950	1.020
Tap 73	0.958	1.001	0.988	1.000	1.010	1.023
Tap 76	0.980	0.979	0.980	0.960	0.940	0.932
Tap 80	0.940	1.002	1.017	1.000	1.000	1.016
QC 18	0.1	0.179	0.131	0.084	0.016	0.133
QC 25	0.059	0.176	0.144	0.008	0.015	0.144
QC 53	0.063	0.141	0.162	0.053	0.038	0.103
PG (MW)	1278.6	1274.4	1274.8	1276	1275	1272.68
QG (Mvar)	321.08	272.27	276.58	309.1	304.4	272.57
Reduction in PLoss (%)	0	15.4	14.1	9.2	11.6	25.42
Total PLoss (Mw)	27.8	23.51	23.51	25.24	24.56	20.732

Note: NR* -Not reported

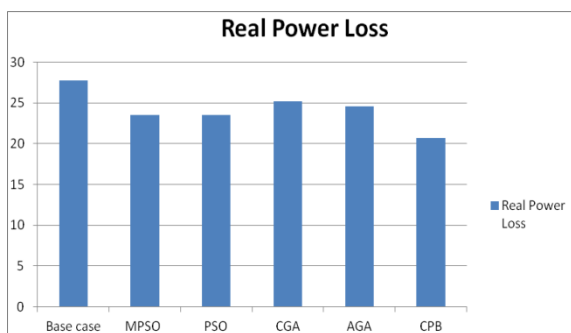


Figure 1. Comparison of real power loss

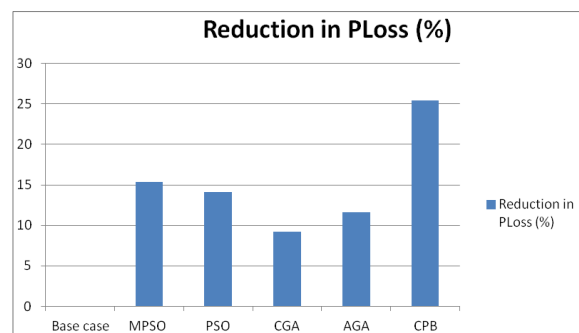


Figure 2. Real power loss reduction in percentage

5. CONCLUSION

Chaotic predator-prey brain storm optimization (CPB) algorithm successfully solved the optimal reactive power problem. Predator-prey brain storm optimization position cluster centers to perform as predators, consequently it will move towards better and better positions, while the remaining ideas perform as preys; hence get away from their adjacent predators. In the projected chaotic predator-prey brain storm optimization algorithm chaotic theory has been applied in the modeling of the algorithm. In the proposed algorithm main properties of chaotic such as ergodicity and irregularity used to make the algorithm to jump out of the local optimum as well as to determine optimal parameters. Proposed CPB algorithm has been tested in standard IEEE 57 bus test system and simulation results show the projected algorithm reduced the real power loss efficiently.

REFERENCES

- [1] K. Y. Lee, Y. M. Park, and J. L. Ortiz, "Fuel-cost minimisation for both real-and reactive-power dispatches," in *IEEE Proceedings C-Generation, Transmission and Distribution*, vol. 131, no. 3, pp. 85-93, 1984.
- [2] N. I. Deeb and S. M. Shahidehpour, "An efficient technique for reactive power dispatch using a revised linear programming approach," *Electric Power System Research*, vol. 15, no. 2, pp. 121-134, 1998.
- [3] M. Bjelogrić, M. S. Calović, P. Ristanović, and B. S. Babić, "Application of Newton's optimal power flow in voltage/reactive power control," in *IEEE Transactions on Power Systems*, vol. 5, no. 4, pp. 1447-1454, 1990.
- [4] S. Granville, "Optimal reactive dispatch through interior point methods," in *IEEE Transactions on Power Systems*, vol. 9, no. 1, pp. 136-146, 1994.
- [5] N. Grudinin, "Reactive power optimization using successive quadratic programming method," in *IEEE Transactions on Power Systems*, vol. 13, no. 4, pp. 1219-1225, 1998.
- [6] R. Ng Shin Mei, M. H. Sulaiman, Z. Mustafa, and H. Daniyal, "Optimal reactive power dispatch solution by loss minimization using moth-flame optimization technique," *Appl. Soft Comput.*, vol. 59, pp. 210-222, 2017.
- [7] Gonggui Chen, Lilan Liu, Zhizhong Zhang, and Shanwai Huang, "Optimal reactive power dispatch by improved GSA-based algorithm with the novel strategies to handle constraints," *Appl. Soft Comput.*, vol. 50, pp. 58-70, 2017.
- [8] E. Naderi, H. Narimani, M. Fathi, and M. R. Narimani, "A novel fuzzy adaptive configuration of particle swarm optimization to solve large-scale optimal reactive power dispatch," *Appl. Soft Comput.*, vol. 53, pp. 441-456, 2017.
- [9] A. A. Heidari, R. A. Abbaspour, and A. R. Jordehi, "Gaussian bare-bones water cycle algorithm for optimal reactive power dispatch in electrical power systems," *Appl. Soft Comput.*, vol. 57, pp. 657-671, 2017.
- [10] M. Morgan, N. R. H. Abdulla, M. H. Sulaiman, M. Mustafa, and R. Samad, "Benchmark studies on optimal reactive power dispatch (ORPD) based multi-objective evolutionary programming (MOEP) using mutation based on adaptive mutation adapter (AMO) and polynomial mutation operator (PMO)," *Journal of Electrical Systems*, vol. 12, no. 1, pp. 121-132, 2016.
- [11] Rebecca Ng Shin Mei, Mohd Herwan Sulaiman, and Zuriani Mustafa, "Ant lion optimizer for optimal reactive power dispatch solution," *Journal of Electrical Systems, Special Issue AMPE2015*, pp. 68-74, 2015.
- [12] P. Anbarasan and T. Jayabarathi, "Optimal reactive power dispatch problem solved by symbiotic organism search algorithm," *2017 Innovations in Power and Advanced Computing Technologies (i-PACT)*, Vellore, 2017, pp. 1-8.
- [13] A. Gagliano and F. Nocera, "Analysis of the performances of electric energy storage in residential applications," *International Journal of Heat and Technology*, vol. 35, no. 1 SI, pp. S41-S48, Sep. 2017.
- [14] M. Caldera, P. Ungaro, G. Cammarata, and G. Puglisi, "Survey-based analysis of the electrical energy demand in Italian households," *Mathematical Modelling of Engineering Problems*, vol. 5, no. 3, pp. 217-224, 2018.
- [15] M. Basu, "Quasi-oppositional differential evolution for optimal reactive power dispatch," *Electrical Power and Energy Systems*, vol. 78, pp. 29-40, June 2016.
- [16] T. Yamada, T. Okuda, M. Abdullah, M. Awang, and A. Furukawa, "The leaf development process and its significance for reducing self-shading of a tropical pioneer tree species," *Oecologia*, vol. 125, no. 4, pp. 476-482, 2000.
- [17] M. El-Abd, "Cooperative coevolution using the brain storm optimization algorithm," *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, Athens, 2016, pp. 1-7, 2016.
- [18] IEEE, "The IEEE-test systems," 1993. [Online]. Available at: https://labs.ece.uw.edu/pstca/pf57/pg_tca57bus.htm.
- [19] A. N. Hussain, A. A. Abdullah, and O. M. Neda, "Modified particle swarm optimization for solution of reactive power dispatch," *Research J. of Applied Sciences, Engineering and Technology*, vol. 15, no. 8, pp. 316-327, 2018.