Engineering, 2013, 5, 157-162
doi:10.4236/eng.2013.51b029 Published Online January 2013 (http://www.SciRP.org/journal/eng)

Differential Evolution Immunized Ant Colony Optimization Technique in Solving Economic Load Dispatch Problem

N. A. Rahmat, I. Musirin
Universiti Teknologi MARA, Malaysia
Email: azamrahmat@gmail.com, i_musirin@yahoo.co.uk

Received 2013

Abstract

Since the introduction of Ant Colony Optimization (ACO) technique in 1992, the algorithm starts to gain popularity due to its attractive features. However, several shortcomings such as slow convergence and stagnation motivate many researchers to stop further implementation of ACO. Therefore, in order to overcome these drawbacks, ACO is proposed to be combined with Differential Evolution (DE) and cloning process. This paper presents Differential Evolution Immunized Ant Colony Optimization (DEIANT) technique in solving economic load dispatch problem. The combination creates a new algorithm that will be termed as Differential Evolution Immunized Ant Colony Optimization (DEIANT). DEIANT was utilized to optimize economic load dispatch problem. A comparison was made between DEIANT and classical ACO to evaluate the performance of the new algorithm. In realizing the effectiveness of the proposed technique, IEEE 57-Bus Reliable Test System (RTS) has been used as the test specimen. Results obtained from the study revealed that the proposed DEIANT has superior computation time.

Keywords: Ant Colony Optimization (ACO), Differential Evolution (DE), Differential Evolution Immunized Ant Colony Optimization (DEIANT)

1. Introduction

In 1992, Marco Dorigo introduced a probabilistic algorithm known as Ant Colony Optimization (ACO) technique. In his PhD thesis, he described that ACO resembles the natural behavior of a colony of ant during their random expedition to find the best path between their nest and food source. The ant will deposit a chemical trace known as pheromone. The pheromone will act as the stimulant to attract more ants to utilize on the same path. Any less-traveled paths will be forgotten since their pheromone traces has been evaporated. Marco Dorigo employed this behavior into his research to solve the travelling salesmen problem (TSP) [1]. Since, ACO has attracted many researchers to employ the algorithm into their research. Mohd Rozeli Kalil et. al successfully implements ACO to gain maximum loadability in voltage control study [2]. Ashish Ahuja and Anil Pahwa [3] stated in their research that ACO significantly minimized the loss in a distribution system. Moreover, D. Nualhong et. al utilizes ACO in his research to solve the unit commitment problems [4]. However, further research on the algorithm indicates that ACO suffers from several shortcomings. H. B. Duan, et al stated in his research that ACO may experience stagnation due to its positive feedback strategy. The researcher also highlighted that the random selection strategy of ACO makes the algorithm sluggish [5]. Another researcher claimed that ACO has slow convergence rate [6-7].

Another attractive optimization technique is the Differential Evolution (DE). DE was created by Storn and Price in 1995 [8]. As a typical Evolutionary Algorithm (EA), DE was used to solve stochastic, non-differentiable, non-linear, multi-dimensional and population-based optimization problem [9]. The algorithm was also used to solve a Chebychev Polynomial Fitting Problem during the First International Contest on Evolutionary Computer (1st ICEO) in Nagoya. As a typical Evolution Algorithm, DE comprises of mutation, crossover, and selection process. In 1997, Storn and Price stated that DE is much better than Simulated Annealing and Genetic Algorithm [10]. As such, DE has become the candidate technique to optimize Neural Network Learning, Radio Network design, multiprocessor synthesis and gas transmission network design. Therefore, this paper presents Differential

Evolution Immunized Ant Colony Optimization (DEIANT) technique in solving economic load dispatch problem. The aim of the proposed technique is to alleviate the slow convergence, and stagnation experienced in the traditional ACO through the application of DE. The performance of DEIANT will be evaluated by using the algorithm to solve and optimize economic load dispatch problems. Performance evaluation of the proposed DEIANT revealed that the proposed technique is superior than the traditional ACO in terms of achieving optimal solution and computation time.

2. Differential Evolution Immunized Ant Colony Optimization Formulation

The capability to achieve fast convergence has been identified as the attractive feature of DE [10]. This feature will be used to compensate the stagnation of ACO algorithm. The modification of ACO algorithm will be focusing on the pheromone updating rule which is subjected to cloning, mutation, crossover, and selection process. Figure 1 depicts the overall structure and processes of Differential Evolution Immunized Ant Colony Optimization (DEIANT) algorithm. DEIANT is configured from the combination of original ACO, combined with DE and cloning process. ACO search agent will initialize random tours and deposit pheromone layers. The algorithm will update the pheromone level at each iteration. Cloning process will increase the pheromone amount by generating the exact copies of the pheromone layer. To enhance the pheromone layer, DE will mutate and crossover the pheromone layer, thus increasing the diversity of the pheromone. ACO will evaluate the fitness of the pheromone layer.

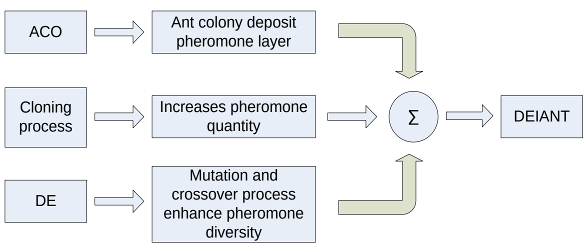

Figure 1. Structural diagram of DEIANT

2.1. Initialization

DEIANT consist of several classical ACO parameters. Therefore, similar to ACO technique, the initial number of ant, nodes, pheromone decay parameter, α, and the initial pheromone level, τ_0 will be heuristically initialized. The maximum distance travelled by each ant, d_{max}, is subjected to a constraint in order to avoid large computation time. d_{max} is obtained by calculating the longest distance travelled by the ants. Each ant will tour and select the next unvisited node until all nodes have been visited.

2.2. Apply State Transition Rule

The next unvisited node is chosen according to the state transition rule, s. Each node can only be visited once. The ant that was positioned at node(r) will go to the next node(s).

$$P_k(r,s) = \begin{cases} \dfrac{[\tau(r,s)]\cdot[\eta(r.s)^{\beta}]}{\sum \mu\varepsilon J_{k(r)}[\tau(r,u)]\cdot[\eta(r.u)^{\beta}]} \end{cases} \tag{1}$$

2.3. Apply Local Updating Rule

Altering the level of pheromone deposition is done during the construction of solution. The amount of pheromone is either increased or decreased to vary the attractiveness of the route via the *evaporation rate, ρ*. Dorigo stated that parameter ρ is needed to avoid the algorithm converge pre-maturely. The parameter ρ act as a multiplier and is set between 0 to 1. In this research, the evaporation rate *is* set to 0.7, meaning that at every iteration; the pheromone will be reduced by 0.7.

2.4. Pheromone Cloning

The cloning process from Artificial Immune System is implemented into DEIANT. The pheromone level will be duplicated to increase pheromone population and diversity. Figure 2 below depicts the cloning of the original pheromone level.

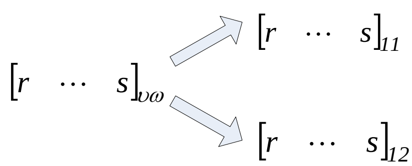

Figure 2. Pheromone cloning process

Where υ represents the set of pheromone that will be cloned, and ω represents the cloned pheromone set. Both υ and ω can be written as $\upsilon = 1,\ldots,V$ and $\omega = 1,\ldots W$, respectively.

2.5. Pheromone Mutation

The pheromone mutation process is used to enhance the pheromone layer over the visited node during ant exploration. The Gaussian Distribution Equation was utilized to mutate the pheromone level as in (2):

$$X_{i+m,j} = X_{i,j} + N\left(0, \beta \cdot \left(X_{j\,max} - X_{j\,min}\right) \cdot \frac{f_i}{f_{max}}\right) \quad (2)$$

Where

$X_{i+m,j}$: Pheromone mutation function

$X_{j\,min}$: Smallest visited node

$X_{j\,max}$: Largest visited node

f_i : Distance travelled by ant

f_{max} : Longest distance travelled by ant

2.6. Crossover

Crossover operation is implemented to emphasize the diversification vector of the pheromone trail. The mutated pheromone trail and the original one will merge together by applying a binomial distribution known as the crossover operation. In this algorithm, the original and mutated pheromone level will be resorted within the same matrix. The crossover matrix, X_{gi} is sorted in descending order, as shown below:

$$X_{gi} = \begin{bmatrix} n & \cdots & n-1 \\ \vdots & \ddots & \vdots \\ n-1 & \cdots & n-m \end{bmatrix} \quad (3)$$

Where n is the highest pheromone level for the tour, and n-m is the lowest pheromone level.

2.8 Selection

Trial pheromone trail, ρ_t is the product of the crossover process. The algorithm will now select the required trail according to this rule:

$$\rho_t = \begin{cases} \rho_t, if \;\; pheromone\, level, \rho \le \alpha \\ \rho, otherwise \end{cases} \quad (4)$$

Where α is the pheromone decay parameter.

Basically, the selection process will specify the acceptable condition to accept the pheromone level. The trail with a higher pheromone level will be selected. The trail with fewer pheromone levels will be discarded.

2.9 Control Variable Calculation

The control variable x, is computed by applying the following equation:

$$x = \frac{d}{d_{max}} \cdot x_{max} \quad (5)$$

Where:

d : distance for every ants tour

d_{max} : maximum distance for every ants tour

x_{max} : maximum of x

Variable x will become the multiplier to calculate the objective function. In this research, variable x is multiplied with the cost function (7).

2.10 Global Updating Rule

After all ants have finished performing their random exploration, the best ant of the colony will be selected. The best ant is carrying the data of the best tour, and thus, the data will be stored. The following equation is applied to update the pheromone level globally:

$$\tau(r,s) \leftarrow (1-\alpha)\tau(r,s) + \alpha.\Delta\tau(r,s) \quad (6)$$

The globally best tour will be selected as the first node of the next iteration.

2.11 End Condition

The DEACO algorithm will halt the iteration when the maximum number of iteration (t_{max}) has been reached and all ants have completed their tours. If a better path is discovered, it will be used as the reference. Only one ant will find the optimal path.

3. Economic Load Dispatch Formulation

Economic load dispatch is the operation of determining the optimal output that must be produced by the generation facilities to feasibly satisfy the load demand [12]. Therefore, the key objective of economic load dispatch is to reduce the operating cost of each generating unit in the power system. The operation cost can be calculated by utilizing equation (7):

$$cost = \sum_{i}^{Ng} F_i(P_i) \quad (7)$$

Where Ng is the number of generating unit. $F_i(P_i)$ is the cost function, and P_i is the real power output of the unit I, measured in MW. $F_i(P_i)$ is usually approximated by a quadratic function:

$$F_i(P_i) = a_i P_i^2 + b_i P_i + c \quad (8)$$

Where a_i, b_i, and c_i are the cost coefficients of generator i. The above equation is subjected to both the equality and inequality constraints.

The system power loss can be calculated by using the power flow equation below (19):

$$P_L = \sum_i^N \sum_j^N P_i B_{ij} P_j + \sum_i^N B_{oi} P_i + B_{oo} \quad (9)$$

Where B_{ij}, B_{oi}, and B_{oo} are the *B-loss* coefficient.
The cost function is given by equation (26):

$$C_n = a + bP_n + cP_n^2 \quad (10)$$

The followings are the generators' operating costs for IEEE 57-Bus System. These equations are derived from equation (10):

$$C_1 = 400 + 7.0P_1 + 0.0070P_1^2$$
$$C_2 = 200 + 10.0P_2 + 0.0095P_2^2$$
$$C_3 = 220 + 8.5P_3 + 0.0090P_3^2$$
$$C_6 = 200 + 11.0P_6 + 0.0090P_6^2$$
$$C_8 = 240 + 10.5P_8 + 0.0080P_8^2$$
$$C_9 = 200 + 12.0P_9 + 0.0075P_9^2$$
$$C_{12} = 180 + 10.0P_{12} + 0.0068P_{12}^2 \quad (11)$$

Where C_1, C_2, C_3, C_6, C_8, $C_{9\ and}$ C_{12} are the operating cost functions for generator 1, generator 2, generator 3, generator 6, generator 8, generator 9, and generator 12 respectively.
The total operating cost, C_T can be formulated as:

$$C_T = C_1 + C_2 + C_3 + C_6 + C_8 + C_9 + C_{12} \quad (12)$$

The total operating cost is the objective function for this research. The objective is to cut-down the total operating cost, while preserving the system constraints within the permissible limits. While minimizing the operating cost, it was initially estimated that the power loss will be reduced as well. Power loss is the marginal difference between the power produced, and the power sold to the consumer [13].

4. Results and Discussion

The DEIANT algorithm was developed by using MATLAB. The algorithm was used to optimize economic load dispatch problem. The research was conducted on IEEE-57 bus system. The system contains seven generating units. The load was varied to assess the performance of the proposed algorithm. The objectives are to minimize the total generating cost, to reduce transmission loss, and to reduce the computation time. Comparisons were made between DEIANT and the original ACO. Table 1 tabulates the simulation results of classical ACO algorithm. Table 2 tabulates the simulation results of the newly developed DEIANT algorithm.
Note that the load, designated by Q_d at Bus 10 was varied between 0 to 25MVAR to see the performance of DEIANT in solving economic load dispatch problem. The comparison of DEIANT and classical ACO undoubtedly points out that the new algorithm successfully outperforms the classical ACO. Varying the load gives less impact to the performance of DEIANT. For example, when Q_d is set to 20MVAR, DEIANT generates the total operating cost of 41855 \$/h. The cost is economical than the one calculated by classical ACO (41862 \$/h), giving a price drop of 0.0167%. If the load was set to 15MVAR, the idea is the same; DEIANT only generates lower power loss.

Table 1. Simulation results of classical ACO

Q_d (MVAR)	P_1 (MW)	P_2 (MW)	P_3 (MW)	P_6 (MW)	P_8 (MW)	P_9 (MW)	P_{12} (MW)	P_{loss} (MW)	Total Cost (\$/h)	Time (s)
0	137.1095	89.8488	45.1114	75.8669	463.3333	100	358.7308	19.2007	41842	30.63371
5	138.5126	96.3783	45.3754	74.8029	458.1759	96.2979	360.593	19.336	41856	70.29794
10	138.4993	96.3443	45.3752	74.7286	458.1097	96.6145	360.495	19.3666	41858	52.4811
15	138.4077	96.3373	45.377	74.6806	458.0548	96.9501	360.4142	19.4217	41860	67.40478
20	138.4437	96.2948	45.3759	74.5882	457.9824	97.268	360.3066	19.4597	41862	34.64456
25	138.4127	96.2995	45.366	74.7954	457.8748	97.4865	360.121	19.5559	41866	53.20833

Table 2. Simulation results of DEIANT

Q_d (MVAR)	P_1 (MW)	P_2 (MW)	P_3 (MW)	P_6 (MW)	P_8 (MW)	P_9 (MW)	P_{12} (MW)	P_{loss} (MW)	Total Cost ($/h)	Time (s)
0	137.3725	90.9657	45.045	74.2499	463.3627	100	358.7523	18.9481	41832	4.317901
5	137.349	91.0764	45.0431	74.2626	463.3531	100	358.6874	18.9716	41832	4.539834
10	138.7015	97.2722	45.324	73.6004	458.3919	96.009	360.6952	19.1941	41851	4.854118
15	138.6771	97.2475	45.3171	73.5275	458.3209	96.3482	360.5982	19.2362	41853	4.778885
20	138.6519	97.2263	45.31	73.4555	458.25	96.6919	360.5016	19.2872	41855	5.184511
25	138.626	97.2089	45.3028	73.3846	458.1796	97.04	360.4055	19.3474	41857	5.328133

Moreover, at 20MVAR load adjustment, the classical ACO computed power loss of 19.4597MW. However, DEIANT effectively reduces it to 19.2872MW, with a significant difference of 172.5kW. Furthermore, DEIANT possesses faster computation time than the classical ACO. For instance, classical ACO requires about 35 seconds to optimize the objective function, but DEIANT optimize the objective function in just five seconds. This implies that DEIANT can achieve optimal solution at a faster rate, although the algorithm is more complex and sophisticated than the classical ACO.

5. Conclusion

This study demonstrates the development of a new algorithm termed as Differential Evolution Immunized Ant Colony Optimization (DEIANT). Through the combination of DE, ACO and cloning process, this new algorithm has successfully overcome the drawbacks of classical ACO algorithm. The good performance of DEIANT was verified by optimizing economic load dispatch problem. Comparisons with classical ACO imply that DEIANT effectively cuts-down the operating cost while reducing the power loss. The optimization process was performed with a faster speed than the classical ACO. Future development will be focusing on the modification to increase the convergence speed of the algorithm.

6. Acknowledgement

The authors would like to acknowledge The Research Management Institute (RMI) UiTM, Shah Alam and Ministry of Higher Education Malaysia (MOHE) for the financial support of this research. This research is supported MOHE under the Fundamental Research Grant Scheme (FRGS) with project code: 600-RMI/ST/FRGS 5/3/Fst (163/2010).

REFERENCES

[1] Ying-Tung Hsiao, Cheng-LongChuang and Cheng- Chih Chien, “Ant Colony Optimization for Best Path Planning,” *International Symposium on Communications and Information Technologies 2004 (ISCIT 2004)*, Sapporo, Japan, 26-29 October 2004, pp. 109-113.

[2] Mohd Rozely Kalil, Ismail Musirin, Muhammad Murtadha Othman, “Maximum Loadability in Voltage Control Study Using Ant Colony Optimization Technique”, *IEEE First International Power and Energy Conference (PECon2007)*, 28-29 Nov. 2006, pp. 240-245.

[3] Ashish Ahuja and Anil Pahwa, “Using Ant Colony Optimization for Loss Minimization in Distribution Networks”, *37th Annual North American Power Symposium, 2005*, 23-25 Oct. 2005, pp. 470- 474.

[4] D. Nualhong, *et al.*, "Diversity Control Approach to Ant Colony Optimization for Unit Commitment Problem," in *TENCON 2004. 2004 IEEE Region 10 Conference*, 2004, pp. 488-491 Vol. 3.

[5] H.B. Duan and D.B. Wang, “a novel improved ant colony algorithm with fast global optimization and its simulation,” *Information and Control*, vol.33, pp. 241-244, April 2004.

[6] Linda Slimani and Tarek Bouktir, “Economic Power Dispatch of Power System with Pollution Control using Multiobjective Ant Colony Optimization”, *International Journal of* ***Computational Intelligence Research (IJCIR)*** 2007**,** Vol. 3, No. 2, pp. 145-153.

[7] R. Bhavani, G. Sudha Sadasivam, and R. Kumaran, "A novel parallel hybrid K-means-DE-ACO clustering approach for genomic clustering using MapReduce," in *Information and Communication Technologies (WICT), 2011 World Congress on*, 2011, pp. 132-137.

[8] R. Storn and K. Price, “Differential Evolution – A Simple and Efficient Adaptive Scheme for Global Optimization Over Continuous Spaces”, Technical Report TR-95-012, ICSI, March 1995.

[9] K.P. Wong and Z.Y. Dong, “Differential Evolution, an AlternativeApproach to Evolutionary Algorithm”, in K.Y. Lee edt.

Intelligent Optimization and Control for Power Systems, IEEE Publishing, invited chapter, Nov. 2005.

[10] Storn R., Price K.: ‘Differential Evolution – A Simple and Efficient Adaptive Scheme For Global Optimization Over Continuous Space”, Journal of Global Optimization, 1997.

[11] N. A. Rahmat, I. Musirin (2012). Differential Evolution Ant Colony Optimization Technique (DEACO) In Solving Economic Load Dispatch Problem. *IEEE Internation Power Engineering and Optimization.*

[12] S. M. V. Pandian and K. Thanushkodi, "Solving Economic Load Dispatch Problem Considering Transmission Losses by Hybrid EP-EPSO Algorithm for Solving Both Smooth and Non-Smooth Cost Function," *International Journal of Computer and Electrical Engineering*, vol. 2, 2010

[13] M. Basu, “Artificial Immune System for Dynamic Economic dispatch,” Electrical Power and Energy System, vol. 33, pp. 131-136, 7 June 2010

