A MULTI-AGENT GENETIC ALGORITHM FOR THE SOLUTION OF THE ECONOMIC DISPATCH PROBLEM

Irina CIORNEI[1], Elias KYRIAKIDES[2]

This paper presents the application and comparison of classical and heuristic methods which solve the economic dispatch (ED) problem, such as the Lagrange multipliers method (LM), genetic algorithm (GA) and simulated annealing (SA). An analytical and empirical comparison among these methods is performed with consideration to the problem formulation and its complexity and ruggedness. In addition, we propose an improved genetic algorithm with ant strategy (GAAPI) method in order to make the search more robust for economic dispatch problems that involve nonsmooth cost functions. The GAAPI method involves a multi-agent search in the solution space inspired from ant colony optimization techniques. For large systems, where many heuristic algorithms are time consuming, we propose a nearest neighbour (NNei) method to generate the starting solution for the GAAPI algorithm and to obtain near optimum global solution with a small computational time.

Keywords: economic dispatch, genetic algorithm, ant colony, nonconvex optimization, GAAPI

1. Introduction

The optimal allocation of the load demand to the committed generating units is a critical aspect of power system optimization in terms of the economics of each electric utility. This process, termed economic dispatch (ED), is often a computationally intensive task especially when the generating units in the system are characterized by nonsmooth cost functions. The economic dispatch problem is solved as either a minimization problem (minimize the fuel cost) or as a problem of profit maximization (especially in deregulated energy market environments). The power system dispatcher needs to take into consideration system parameters such as the heat rate curves of generators, minimum and maximum generation limits, and ramp rate limits to obtain the most economic schedule of generation. Further, constraints such as transmission line limits and power system spinning reserve have to be continuously respected.

[1] Ph.D. candidate, Department of Electrical and Computer Engineering, University of Cyprus, Cyprus (*Corresponding author)
[2] Lecturer, Department of Electrical and Computer Engineering, University of Cyprus, Cyprus

Traditionally, the economic dispatch problem is solved using methods based on LM, gradient methods [1-2], or dynamic programming (DP) [3]. These classical dispatch algorithms that use Lagrangian multipliers or gradient methods require monotonically increasing incremental cost curves. Unfortunately, the input-output characteristics of real generating units are highly nonlinear because of factors such as valve-point loading and rate limits. Despite the fact that DP does not impose any restrictions on the cost curve shape, this method suffers from the "curse of dimensionality" [4].

With the advent of deregulated power markets the problem of ED became more complex. The cost function is no longer smooth, market rules need to be considered and interconnection constraints have to be respected. Therefore, intelligent techniques are required to overcome the obstacles of the increased complexity and nonlinearity. In this respect, recent years brought about an increased research activity on economic dispatch methods based on intelligent optimization techniques such as neural networks, genetic algorithms [5-7], evolutionary based methods [8-9], or simulated annealing [10]. These intelligent-based methods improve the previous solution techniques by searching for the global minimum of the cost function. The first approaches in applying intelligent based algorithms to solve the economic dispatch problem used the classical formulation of the problem with a quadratic cost function without losses or with constant losses [11-12].

Heuristic methods such as GAs and SA do not guarantee to always converge to the global optimal solution, but they often provide fast and reasonable solutions (suboptimal or near globally optimal), especially for small problems. In this paper, an improved genetic algorithm with ant colony strategy (GAAPI) method is proposed. The method involves a multi-agent search and aims to a robust, guided search in the solution space of the nonsmooth ED problem. The test systems used in this study are: a three generator test system [1], a modified IEEE 30-bus (6 generators) test system and a modified IEEE 118-bus (40 generators) test system [13].

2. Problem description

In this work, the ED problem is to minimize the cost of generation (objective function). Further, a number of system constraints such as the balance constraint, ramp rate limits and transmission system limits should be respected at all times.

A. *Objective function:*

The objective function in the ED problem can be either *smooth* or *nonsmooth*. A smooth objective function is a quadratic approximation of the incremental cost curves that could include the operation maintenance cost and is of the form,

$$\min F_t = \sum_{i=1}^{n} a_i + b_i P_i + c_i P_i^2 . \qquad (1)$$

A nonsmooth function incorporates a number of extra factors such as the valve point loading effect, multiple fuel types, and prohibited operating zones and is of the form,

$$\min F_t = \sum_{i=1}^{n} a_i + b_i P_i + c_i P_i^2 + | e_i \cdot \sin(f_i (P_i^{\min} - P_i)) | \qquad (2)$$

Terms $a_{i,}$ $b_{i,}$ c_i $e_{i,}$ $f_{i,}$ are the cost function coefficients (constants), and P_i is the output power of unit *i*.

B. Constraints:

The constraints can be evaluated as either *hard constraints* or *soft constraints*. Hard constraints are the ones that must be respected continuously in order to keep the system in a safe operating state. Soft constraints are constraints that are desirable to be respected but which may or may not affect the system performance (such as emission constraints and spinning reserve constraints).

The hard constraints considered in this paper include the balance constraint (with and without losses),

$$\sum_{i=1}^{n} P_i - P_D - P_L = 0 , \qquad (3)$$

generation limit constraints,

$$P_i^{\max} \le P_i \le P_i^{\min} , \qquad (4)$$

transmission constraints (security limits and losses), and ramp rate limits. P_D is the load power demand, P_L are the transmission losses and P_i^{min}, P_i^{max} are the minimum and maximum output powers of unit *i* respectively.

Transmission power losses can be computed through a power flow computation, but a common practice is to express the total transmission losses as a quadratic function of the power outputs of generating units either through *Kron's loss formula*, or through a simplified formula [2]. The B coefficients are assumed to be constant, and reasonable accuracy can be expected when the actual operating conditions are close to the case at which these coefficients where computed.

3. Description and development of the proposed method

The main idea behind GAs is to improve a set of candidate solutions for the problem by using several genetic operators inspired from genetic evolution mechanisms observed in real life. Genetic operators are the variation mechanisms that generate new candidate solutions, similar to the parents (solutions from a previous generation), but including some differences. Usually, genetic operators such as: *selection*, *crossover* and *mutation* are in charge of this task. The selection operator makes sure that the best member from a population survives. Crossover takes two parents and mixes them up with a given probability so that new individuals are generated. Mutation takes an individual and randomly changes a part of it with a certain probability.

Pachycondyla apicalis ants have been studied in the Mexican tropical forest near the Guatemalan border. Colonies of these ants comprise around 20 to 100 ants. The foraging strategy of such ants can be characterized as follows. First, these ants create their hunting sites which are distributed relatively uniformly around their nest within a radius of approximately 10 m. In this way, using a small mosaic of areas, the ants cover a rather large region around the nest. Second, the ants will intensify their searches around some selected sites for prey. In this foraging process, these ants communicate with each other using visual landmarks rather than pheromone trails. After capturing their prey, the ants will move to a new nest based on a recruitment mechanism called "tandem running" to begin a new cycle of foraging. Based on the natural behaviour of pachycondyla apicalis, Monmarche *et al.* proposed an API algorithm (short for apicalis) for the solution of optimization problems [14]. However, further research shows that API has poor use of the memory that generally characterizes ant colony systems [15].

To eliminate the shortcomings and the inadequate robustness of the global search ability of the API algorithm, a GAAPI algorithm that incorporates some favourable features of API and GA algorithms is proposed in this paper. To facilitate the understanding and description of the method, the steps of the proposed algorithm are outlined in Table 1, and described below.

1) Generation of New Nest: After initialization, only the best solution found since the last nest move has the opportunity to be selected as a new nest to start the next iteration. The "hill climb" property is not very strong in this case, so the entrapment in local minima is avoided.

2) Exploitation: The main differences of the proposed GAAPI and the API algorithm lie in the following aspects: initially, each ant checks its memory. If the number of hunting sites in its memory is less than a predefined number, it will generate a new one in the small neighbourhood of the current ant center, save it to its memory, and use it as the next hunting site. Otherwise, one of its memorized sites is selected as the hunting site. The ant then performs a local search around the neighbourhood of this hunting site. If this local exploitation is successful, the ant will repeat its exploration around the site until an unsuccessful search occurs; otherwise, the ant will select an alternative one among its memorized sites. This process will be repeated until a termination criterion is satisfied. The termination criterion used in this phase is that the procedure will stop automatically once the number of successive unsuccessful explorations reaches a predefined value or there is no improvement after a number of iterations.

3) Information Sharing: As described previously, the available API algorithm makes poor use of the memory that generally characterizes ant colony systems. To compensate for such a shortcoming in an API algorithm, an information-sharing mechanism is proposed so as to increase the use of the information gathered from the latest searched solutions, and to speed up the solution process. In essence, it is proposed that after each ant has extensively exploited a hunting site, one member of the memorized sites for every ant will be replaced by the solution that survived from a GA that has as initial population the best site of each ant in this cycle. For the selection of the replaced sites, the roulette wheel selection scheme is used [6].

Table 1

GAAPI algorithm

1) ***Initialization:*** set the algorithm parameters.
2) ***Generation of new nest*** (exploration): Choose randomly the initial nest location
3) ***Exploitation***
    3.1) Search intensification:
  while *NestPatience* before moving
   for each ant *ai*,
    while *AntPatience*
     if *ai* has less than *p* hunting sites in its memory, then *create* a new site in the neighborhood of *N* and exploit this new site;
     else if the previous site exploitation is successful, exploit the same site again;
      else explore a randomly selected site (among its p sites in memory).
      endif
     endif
    3.2. Remove from ants memories all sites that have been explored more than Plocal(ai) consecutive times
    3.3. Information sharing:
    Probabilistically replace a site in the memory of the ant
    endwhile *AntPatience*
   endfor
    GA to determine new *Ant Centers*
  endwhile *NestPatience*
   3.4. Nest movement: if more that NestPatience iterations have been performed then change the nest location and reset the memories of all ants
4) ***Termination test:*** Go to (3) or stop if a stopping criterion is satisfied

For the economic dispatch solution of the IEEE 118 bus test system, a NNei strategy is added to the proposed GAAPI method. The NNei strategy is used to find an initial nest center using the solution obtained from LM for the quadratic part of the cost function. This reduces the computational time of the search procedure. The method incorporating the additional NNei strategy is termed Fast-GAAPI method.

4. Test cases and analysis of results

The algorithms (LM, SA, GA and GAAPI) were implemented on a 3.6 GHz personal computer using MATLAB 7.4. In order to verify and compare the performance of each algorithm in practical applications, three sample networks having nonconvex solution spaces are used as test systems. For this system the transmission losses are calculated by Kron's loss formula.

A. *3 generator test system*

The first test system consists of three generators with nonconvex cost of generation and having a load demand of 850 MW [1], [4]. Table 2 shows the best generation dispatch and the minimum cost achieved for the three test methods applied to the nonconvex system. The minimum cost is achieved using the GAAPI method that was proposed in this paper. SA was the fastest among the three methods and GA was the slowest.

Table 2

Test case results for the 3-generator system

Unit power output [MW]	SA	GA	GAAPI
P1	365.0082	402.828539	393.1699
P2	129.5031	196.438952	122.2264
P3	355.4914	250.732522	334.6037
Total generation [MW]	850.0027	850.000013	850.000
Total generation cost [$/h]	8409.6766	8235.1542	**8234.0838**
CPU time [sec]	47.96875	421.5312	**298.0781**

B. *6 generator test system*

The second test system consists of six generators and a load demand of 1263 MW [4]. The analysis is given for two cases: when the generation cost function is convex (Table 3) and for the case when the generation cost function is nonconvex (Table 4). For both convex and nonconvex cost functions the minimum generation cost is achieved using the GAAPI method as illustrated in Tables 3 and 4.

C. *40 generator test system*

The third test system consists of forty generators with a nonconvex cost of generation and having a load demand of 10500 MW [16]. Table 5 shows the results obtained using the optimization methods. In this case, the fast genetic algorithm with API strategy (Fast-GAAPI) is used, to speed up the optimization process. This method resulted in the best solution among the methods tested. The

Fast-GAAPI algorithm is approximately 2.6 times faster than the GA algorithm and 2.3 times faster than the GAAPI method.

Table 3

Test case results for the 6-generator system with convex cost functions

Unit power output [MW]	LM	SA	GA	GAAPI
P1	448.5503	448.5501	441.9695	449.0586
P2	174.0915	174.0913	178.9703	140.0420
P3	264.2750	264.2748	267.5925	264.8572
P4	139.9208	139.9206	136.0149	138.7214
P5	166.2886	166.2884	160.0929	166.2875
P6	87.9659	87.9657	96.4542	88.3631
Total generation [MW]	1281.092	1281.0909	1281.094	1281.0485
Losses [MW]	18.0924	18.0910	18.0946	18.0485
Total generation cost [$/h]	15518.19	15519.18305	15520.03	**15518.0786**
CPU time [sec]	-	39.9687	538.125	**177.453**

Table 4

Test case results for the 6-generator system with nonconvex cost functions

Unit power output [MW]	SA	GA	GAAPI
P1	448.5503	447.056645	435.4418
P2	174.0914	176.433270	173.3131
P3	264.2750	289.671932	261.6615
P4	139.9208	138.224723	138.4678
P5	166.2886	144.161847	173.5029
P6	87.9659	85.520172	98.7820
Total generation [MW]	1281.0922	1281.3700	1281.1694
Losses [MW]	18.0922	18.3700	18.1694
Total generation cost [$/h]	15560.30409	15563.500166	**15521.0549**
CPU time [sec]	**37.4375**	410.593	**200.2187**

Table 5

Test case results for the 40-generator system with nonconvex cost functions

Method	SA	GA	GAAPI	FGAAPI
Total generation [MW]	10499.99	10499.99	10500.00	10500.00
Total generation cost [$/h]	128227.35	135 198,50	125770.85	**12535.96151**
CPU time [sec]	23.964	756.9	658.7	**284.07812**

5. Conclusion

A GAAPI method is developed and integrated with the API and GA procedures in order to form a powerful optimization tool for the ED problem with a nonconvex cost function. Three other methods (LM, SA, and GA) were implemented and tested by three sample networks having nonconvex solution spaces in order to compare the performance of existing methods to the performance of the proposed algorithm. It was shown that in the case where the cost functions are smooth, the GAAPI method and the LM method have comparable results and are better than the SA and GA methods in terms of the cost of generation. In the case of the nonsmooth cost functions, the proposed GAAPI method gives better results than the SA and GA methods. It was also shown that for the economic dispatch solution of the modified IEEE 118 bus test system, the computational time is significantly reduced when the NNei strategy is added to the proposed GAAPI method.

REFERENCES

[1]. *A. J. Wood and B. F. Wollenberg*, Power Generation Operation and Control, 2nd ed., Ed.Wiley, New York, 1996, pp. 37-45.

[2]. *H. Saadat*, Power System Analysis, 2nd ed., Singapore: Ed. Mc. Graw –Hill, 2004 pp. 268-289.

[3]. *D. W. Ross and S. Kim*, "Dynamic economic dispatch of generation" in IEEE Trans. On Power Apparatus and Systems, **vol. 6**, Nov./Dec. 1980, pp. 2060-2068.

[4]. *N. Sinha, R. Chakraberti, P. K. Chattopadhyay*, "Evolutionary programming techniques for economic load dispatch", in IEEE Trans. On Evol. Computation, **vol. 7**, no.1, Feb. 2003, pp. 83-94

[5]. *I. G. Damousis, A. G. Bakirtzis and P.S. Dokopoulos*, "Network-constrained economic dispatch using real-coded genetic algorithm," in IEEE Trans. on Power Systems, **vol. 18**, no.1, Feb. 2003, pp. 198-205.

[6]. *D. E. Goldberg*, Genetic Algorithms in Search, Optimization and Machine Learning, Reading, MA: Addison–Wesley ed., 1989.

[7]. *A. G. Bakirtzis, P. N. Biskas, C. E. Zoumas, and V. Petridis*, "Optimal power flow by enhance genetic algorithm," IEEE Trans. Power Systems, **vol. 17**, May 2002, pp. 229–236.

[8]. *K. P. Wong, J. Yuryevich*, "Evolutionary programming based algorithm for environmentally-constrained economic dispatch," in IEEE Trans. Power Syst., **vol. 13**, May 1998, pp. 301–306.

[9]. *H. T. Yang, P. C. Yang, and C. L. Huang*, "Evolutionary programming based economic dispatch for units with nonsmooth fuel cost functions," IEEE Trans. Power Syst., **vol. 11**, Feb. 1996, pp. 112–118.

[10]. *K. P. Wong, Y. W. Wong*, "Thermal generator scheduling using hybrid genetic/simulated annealing approach," in IEE Proceeding C, vol. 142, July 1995, pp. 372–380.

[11]. *P. Attaviriyanupap, J. Hasegawa*, "A fuzzy-optimization approach to dynamic economic dispatch considering uncertainties," IEEE Trans. on Power Systems, **vol. 19**, no.3, Aug. 2004, pp. 1299-1307.

[12]. *M. Yoshimi, K. S. Swamp, Y. Izui*, "Optimal economic power dispatch using genetic algorithms", Proceedings of the Second International Forum on Applications of Neural Networks to Power Systems, *1993.*

[13]. *R. Christie, "*Power system test archive", 1999, August. [Online] http://www.ee.washington.edu/research/pstca

[14]. *N. Monmarche, G. Venturini, and M. Slimane*, "On how pachcondyla apicalis ants suggest a new search algorithm," Fut. Gen. Comput. Syst., **vol. 16**, 2000, pp. 937–946.

[15]. *J. Dreo and P. Siarry*, "A new ant colony algorithm using the heterarchical concept aimed at optimization of multiminima continuous functions," Lecture Notes Comput. Sci., **vol. 2463**, 2002, pp. 216–221.

[16]. Z. L. Gaing, "Particle Swarm Optimization to solving the economic dispatch considering the generator constraints", IEEE Trans. on Power Systems, **vol. 18**, no.3, Aug. 2003, pp. 1187-1195.