Efficient Hybrid Optimization Solution for the Economic Dispatch with Nonsmooth Cost Function

Irina Ciornei, *Student Member, IEEE,* and Elias Kyriakides, *Member, IEEE*

Abstract-- **This paper proposes a novel global optimization technique to solve the nonconvex economic load dispatch (NCELD) problem. The foraging strategy of the pachycondyla apicalis ant (API) is hybridized with a genetic algorithm (GA) strategy to incorporate key features of both API and GA and form a relatively simple but robust algorithm, entitled GAAPI. The novel algorithm proposed in this paper combines the downhill behavior of API (a key characteristic of optimization algorithms) and a good spreading in the solution space of the GA search strategy (a guarantee to avoid being trapped in local optima). The feasibility of the proposed method is tested for three different test systems having different size and complexity. The results are calculated in terms of solution quality and computational efficiency; it is shown that the proposed GAAPI is capable of obtaining highly robust, quality solutions in a reasonable computational time.**

Index Terms-- **Economic dispatch, ant colony, API, genetic algorithm, robust search, nonconvex optimization.**

I. INTRODUCTION

ONE of the most important optimization and scheduling problems for generating companies is the economic load dispatch (ELD) due to its major variable cost component in electric utility bills. Any reduction in the generation cost benefits directly and significantly both the generation companies and the consumers. Economic dispatch aims at allocating the load demand to the committed generating units in the most economic or profitable way, while respecting second by second the physical constraints of the power network. Therefore, the system dispatcher needs to take into consideration system parameters such as the heat rate curves of generators, 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 requirements have to be continuously respected. Mathematically, the problem may be stated as a minimization problem [1], when the objective is to minimize the total cost of supplying the load, or a maximization problem [2], when the objective is to maximize the profit of the generating company.

Previous efforts on solving the ELD problem have employed various mathematical programming methods and optimization techniques. Traditionally, the ELD problem is solved using deterministic methods based on Lagrange multipliers, gradient algorithms [1], linear programming [3], quadratic programming [4], or dynamic programming (DP) [5]. All these classical dispatch algorithms (besides DP) require monotonically increasing incremental cost curves, so that the derivatives of the cost functions exist. In other words, the problem is solved by standard nonlinear programming techniques, minimizing a convex function over a convex set. This convex minimization problem is guaranteed to have a unique local minimum, which is also the global minimum. However, the input-output characteristics of the real units are highly nonlinear because of factors such as valve-point loading, multiple fuel usage, prohibited operating zones, nonlinear power flow equality constraints, switched shunt devices, transformer taps, and phase shifters. Furthermore, for a large scale system, the conventional methods have oscillatory problems resulting in a longer solution time [6]. As far as DP is concerned this method does not impose any restrictions on the cost curve shape but it suffers from the "curse of dimensionality" [7], especially when the number of system variables increases.

The complexity of the actual ELD problems has led to the use of metaheuristic methods in an attempt to reach the global optimum solution even when including the non-convex characteristics of the problem. Methods like genetic algorithm (GA) [8]-[11], (improved) evolutionary programming (IEP/EP) [7], [12]-[15], simulated annealing (SA) [16]-[17], neural network [18]-[20], fuzzy logic [21], tabu search [22], and particle swarm optimization techniques (PSO) [6], [23]-[26] are only a few examples that solve the NCELD problem. These methods have the advantage of not having restrictions on the generation cost function or generally on the type of the system. On the other hand, in certain cases they prove to be efficient in finding near global optimal solutions within a reasonable computational time.

Many of the above mentioned optimization techniques suffer from premature convergence, particularly for complex functions having multiple minima [25]. Therefore, hybrid solutions were proposed to overcome these issues. Such hybrid techniques include GA combined with SA [27], EP

I. Ciornei and E. Kyriakides are with the Department of Electrical and Computer Engineering and the KIOS Research Center for Intelligent Systems and Networks, University of Cyprus, 1678 Nicosia, Cyprus (e-mail: eep5ci1@ucy.ac.cy and elias@ucy.ac.cy).

with sequential quadratic programming (SQP) [28], or a neural network (NN) with PSO [29].

This paper proposes a novel hybrid stochastic method to solve the nonconvex economic dispatch problem. This method, termed GAAPI, is shown to be a fast and robust solution that combines an ant colony approach (API) with a genetic algorithm. Three test systems are used to validate its effectiveness and applicability for solving the nonlinear economic dispatch problem. Section II of this paper presents the mathematical formulation of the nonconvex economic load dispatch problem emphasizing the objective function to be optimized and the constraints imposed by the power system operation. Section III describes in detail the proposed GAAPI method used to solve the NCELD problem. Section IV shows the applicability of the proposed method for three different benchmark systems by a detailed analysis of the results. The last section of the paper is allocated for final conclusions.

II. Nonconvex Economic Load Dispatch Problem

The practical NCELD problem refers to generator nonlinearities such as valve point loading effects that are present in the generation cost function, and/or transmission losses, prohibited operating zones and ramp rate limits as nonlinear constraints. All the above mentioned nonlinearities are modeled in this paper.

A. Optimization function

In the classical ELD the main objective is to minimize the total generation cost from traditional generating units. The objective function can be either *smooth* or *nonsmooth*. A *smooth* thermal generation function is a quadratic approximation of the incremental cost curves that could include the operation maintenance cost, and is of the form,

$$F_t = \sum_{i=1}^{n} a_i + b_i P_i + c_i P_i^2 . \tag{1}$$

A *nonsmooth* thermal generation function incorporates sinusoidal components that represent the valve point loading effect produced by opening the steam admission valve,

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

where F_t is the total generation cost, the terms a_i, b_i, and c_i are the fuel-cost coefficients of unit i, and e_i, and f_i are the sinusoidal term coefficients that model the valve-point effect of unit i; P_i is the output power of the ith unit, and n is the number of generators in the system.

B. Constraints

1) Balance constraint

The total electric power generation has to meet the total electric power demand and the real power losses in the transmission lines. Hence,

$$P_{Loss} + P_D - \sum_{i=1}^{n} P_i = 0 , \tag{3}$$

where P_D is the load power demand, P_{Loss} represents the transmission losses, and P_i is the output power of unit i.

2) Transmission constraints

The transmission power losses (P_{Loss}) can be computed through a power flow computation. The power flow equations for a power system are given by,

$$Q_i - Q_{Di} - |V_i| \sum_{j=1}^{NB} |V_j| \left[G_{ij} \sin(\delta_i - \delta_j) - B_{ij} \cos(\delta_i - \delta_j) \right] = 0 \tag{4}$$

$$P_i - P_{Di} - |V_i| \sum_{j=1}^{NB} |V_j| \left[G_{ij} \cos(\delta_i - \delta_j) + B_{ij} \sin(\delta_i - \delta_j) \right] = 0 \tag{5}$$

where $i = 1,..., NB$; NB is the number of buses; Q_i is the reactive power generated at the ith bus; P_{Di} and Q_{Di} are the ith bus real and reactive power demand, respectively; G_{ij} and B_{ij} are the transmission line conductance and susceptance respectively between bus i and bus j; $|V_i|$ and $|V_j|$ are the voltage magnitudes at bus i and bus j respectively; and δ_i and δ_j are the voltage angles at bus i and bus j respectively.

The power flow solution gives all the bus voltage magnitudes and angles. The real power losses in the transmission system can thus be computed as,

$$P_{Loss} = \sum_{k=1}^{NL} g_k \left[V_i^2 + V_j^2 - 2V_i V_j \cos(\delta_i - \delta_j) \right] \tag{6}$$

where NL is the number of transmission lines and g_k is the conductance of the kth line that connects bus i to bus j.

A common practice is to express the total transmission losses either as a quadratic function of the power output of generating units (known as *Kron's loss formula*), or through a simplified formula [1]. The Kron's loss formula is,

$$P_{Loss} = \sum_{i=1}^{n} \sum_{k=1}^{n} P_i B_{ik} P_k + \sum_{i=1}^{n} B_{i0} P_i + B_{00} \tag{7}$$

where n denotes the number of generators in the system, and B_{ik}, B_{i0}, and B_{00} are the so called "B-coefficients" which are assumed to be constant. A detailed method to determine the B-coefficients from the power flow equations is given in [30]. Reasonable accuracy can be expected when the actual operating conditions are close to the case at which these coefficients were computed. To determine the coefficients for a new case study, a power flow program must be run in advance.

Security limits are part of the transmission constraints and refer to the secure operation of the power system, i.e. the apparent power flow through the transmission line (S_l) is restricted by its upper limit. Hence,

$$S_l \leq S_l^{\max} , l = 1,..., NL, \tag{8}$$

where NL represents the number of lines in the system.

3) Generation limit constraints

For stable operation, the real power output of each generator is restricted by lower and upper limits as follows,

$$P_i^{\min} \leq P_i \leq P_i^{\max} , \tag{9}$$

where P_i^{min} and P_i^{max} are the lower and the upper bound of generation of the power unit i.

4) Ramp rate limits

Increasing or decreasing the output generation of each unit is restricted to an amount of power over a time interval due to

the physical limitations of each unit. The generator ramp rate limits change the effective real power operating limits as follows:

$$\max(P_i^{\min}, P_i^0 - DR_i) \le P_i \le \min(P_i^{\max}, P_i^0 + UR_i), \quad (10)$$

where, P_i^0 is the previous output power of generator i; DR_i and UR_i are the down and up-rate limit respectively of the ith generator (MW/time-period).

5) Prohibited operating zones

Modern generators with valve point loading have many prohibited operating zones [6]. Therefore, in practical operation, adjusting the generation output P_i of unit i must avoid unit operation in the prohibited zones. The feasible operating zones of unit i can be described as follows:

$$P_i^{\min} \le P_i \le P_{i,1}^l$$
$$P_{i,j-1}^u \le P_i \le P_{i,j}^l, \; j = 2,3,...,n_i \quad (11)$$
$$P_{i,n_i}^u \le P_i \le P_i^{\max}$$

where n_i is the number of prohibited zones of unit i.

III. Implementation of GAAPI Method for the NCELD Problem

A. Background thoughts on GAAPI

This paper presents the solution of the NCELD problem considering valve point loading, ramp rate limits and prohibited operating zones and employing a hybrid method that incorporates favorable features of two powerful optimization algorithms: GA and API. The starting point (first nest position) of the GAAPI search is the solution obtained with Lagrange multipliers method for the smooth cost of generation. This solution is used as an initial approximation of the desired global optimum.

It is well known that metaheuristic algorithms like GAs, EP, SA and PSO are working well for small dimensional, less complex problems, but fail to locate the global optima for more complicated problems [26] as they have a very good search space covering, but a weak search capability around the global solution. On the other hand, API has a good "hill climbing behavior", but does not cover the solution space very well. Therefore, by incorporating these two algorithms into one technique, it is expected to create a method that combines their good features while overcoming their disadvantages. The GA used in the proposed algorithm is a simple real coded genetic algorithm (RCGA). The API algorithm and the RCGA that form the proposed GAAPI method are described in the remainder of this Section.

A first approach over the proposed algorithm starts with the shape of the cost of generation function with valve point effect loading (Fig.1). For the reduced case, when the transmission losses are ignored and only the balance constraint is considered, the quadratic approximation of the generation cost function can narrow the area where the global solution might be. The balance constraint becomes a straight line crossing the cost function at the point where the generation output equals the load demand. With more constraints taken into account, this geometrical delimitation is a bit more difficult to draw, especially when the transmission losses, prohibited operating

zones, and ramp rate limits must be considered. Therefore, if at time t_0 the losses can be computed and set as a constant value in the balance constraint equation, then this equation is linear, and consequently the quadratic approximation can be considered as a good starting point for the next, more accurate search.

The RCGA is used to reinforce the search whenever the search stagnates, causing saturation. The novel GAAPI method is found to perform efficiently for discontinuous and nonsmooth cost of generation, while premature convergence is avoided.
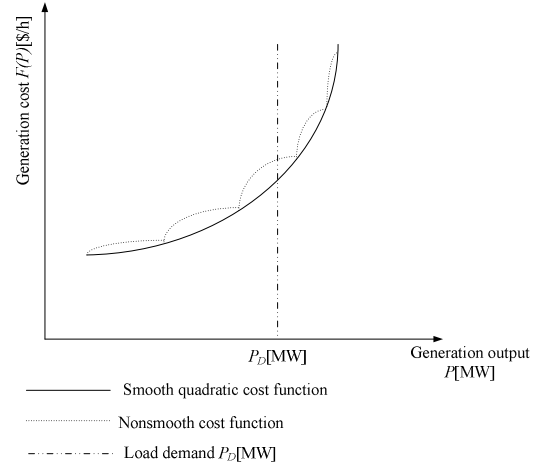

Fig. 1. Determining the first approximation of the ELD solution

The flowchart for this algorithm is given in Fig. 2 and the implementation steps are as follows:

1) Initialization: The first position of the nest is the solution obtained from a lambda iteration algorithm applied to the quadratic part of the generation cost function, and considering the transmission system constraints and the generator limits.

2) Exploitation with API. A detailed description of the search mechanism of API is given in part C of this Section.

2.1) Information sharing with RCGA: In order to keep diversity in the solution space, *information sharing i*s performed using a simple RCGA method. A random site is chosen in the memory of a randomly chosen ant, and it is replaced by the new RCGA solution. This can be seen as a form of communication. The RCGA procedure involves a population formed by all the best hunting sites in all ants' memory so far and the forgotten sites. The best solution obtained after one set of GA operations (selection, crossover, mutation), replaces the chosen site in the memory of the selected ants. This technique is applied before moving the nest in the best position so far. The RCGA contains the forgotten sites in order to keep diversity in the population.

3) Generation of new nest/ Nest movement: After initialization, only the best solution found since the last nest movement has the opportunity to be selected as a new nest to start the next iteration. The "hill climbing" property is not very strong in this case, and therefore a local minimum trapping is avoided.

4) Termination: Two termination criteria are used: (a) the number of iterations reaches an upper limit, or (b) there is no improvement in the solution after a successive number of iterations.
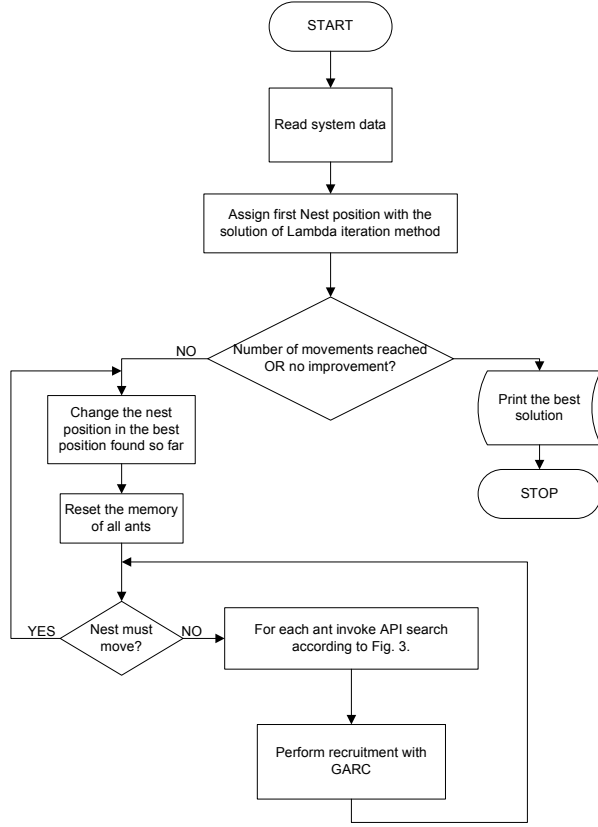
Fig. 2. GAAPI flowchart

the Guatemalan border. Their colonies comprise around 20 to 100 ants and they have a foraging strategy that can be described in short 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. An ant has the tendency to go back to the last successful hunting site using the same path. To follow this path it uses visual landmarks. When a prey has been captured at a given hunting site, the next exploration performed by the ant always starts from that hunting site. The nest movement is quite a complex process relying on ants specialized in the search of a new site and is based on a recruitment mechanism, called tandem running, in which each ant leads another one.

The artificial search mechanism of API described by Monmarché [34] can be summarized as shown in Fig. 3.
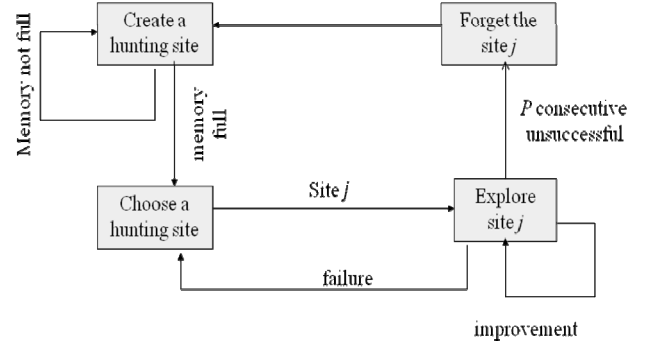

Fig. 3. Search mechanism of ants as used in the proposed method

B. Overview over RCGA

The main idea behind GAs is to improve a set of candidate solutions for a problem by using several genetic operators inspired from genetic evolution mechanisms observed in real life. Usually, the genetic operators used are *selection, crossover,* and *mutation*. The *selection* operator makes sure that the best member from a population survives. *Crossover* generates two new individuals (offspring) from two parent solutions, based on certain rules such as mixing them with a given probability. *Mutation* takes an individual and randomly changes a part of it with a certain probability [29].

In this paper, a real-coded genetic algorithm [31] was adopted, considering the difficulties of binary representation when referring to a continuous search space with large dimensions. Therefore, the decision variable is represented by a real number within its lower and upper bounds.

The RCGA operators were set as follows: (1) *Blend crossover* operator (BLX-α) [32] with a probability of 0.3, and a value of α set to 0.366 [33]; a *uniform mutation* with a mutation probability set to 0.35; *Elitism*: the best two individuals are retained with no modifications in the population of the next generation, such that the strongest genes up to this point are not lost.

C. The search strategy of API

The API (a short for apicalis) algorithm [34] is based on the natural behavior of pachycondyla apicalis ants described in [35]. This type of ants lives in the Mexican tropical forest near

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 hunting site in the small neighbourhood of the current ant's center, save it to its memory, and use it as the next hunting site. Otherwise, a randomly chosen site from 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 same site until an unsuccessful search occurs; otherwise, the ant will select an alternative site 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 a predefined number of exploitation iterations is reached or there is no improvement after a number of iterations. The sites which were explored unsuccessfully a predefined number of consecutive iterations will be erased from the memory of the ant. For the proposed GAAPI algorithm, these erased sites will be selected to form the RCGA population in the current nest movement loop.

D. Feasible solution mechanism

There are two ways to handle constraints in a constrained optimization problem. One is to use a penalty fitness function (optimization function) that aggregates the objective function with the constraint functions penalized [18]-[19], [24]-[26]. However, the penalty parameter must be carefully chosen to distinguish between feasible and infeasible solutions.

Sometimes this parameter training is a difficult task even when the problem is very well known. The other option is to generate only feasible solutions and work only with feasible solutions during the search process. The second procedure was adopted in this paper and is briefly described below.

First an initial solution, $P = (P_1, P_2, ..., P_i, ...P_n)$ is generated between limits, as can be seen below:

$$P_i = P_i^{min} + rand * (P_i^{max} - P_i^{min}), \qquad (12)$$

where P is a vector containing the output value of each generating unit P_i, and *rand* is a normal distribution function that generates random numbers between 0 and 1.

If the balance constraint considering losses as shown in (3) is not satisfied, a random generator is chosen as slack from the pool of the n generators, and fixes its output to meet the balance constraint. If its limits are exceeded, then another random slack generator is chosen from the (n-1) pool. If all the generators were checked and no one can cover the difference to meet the balance, then two generators will be chosen as slack and share the difference, and so on. When a generator is in prohibited zone, then its output is fixed to the closest feasible bound.

IV. NUMERICAL RESULTS AND ANALYSIS

A. Benchmark Systems

In order to validate the proposed GAAPI method that solves the NCELD problem, three test systems were used: 1) a three generator test system without losses and with valve point effect for a load demand of 850 MW [23], [25]; 2) a modified IEEE 30 bus test system with six generators with (a) a quadratic cost of generation [6], [25]-[26], and (b) valve point loading, transmission losses and prohibited operating zones for a load demand of 1263 MW; and 3) a 15-generator test system with quadratic cost of generation, transmission losses, ramp rate limits and prohibited operating zones for a load demand of 2630 MW [24]-[26]. The input data of all test systems can be found on the references mentioned above.

B. Convergence test

The convergence behavior of the GAAPI algorithm was tested in order to determine how fast the proposed algorithm drops under the best average cost of generation reported so far and to prove that the algorithm convergence is not steep, therefore avoiding local minima trapping. Figure 4 shows the convergence behavior of GAAPI for the test system with 6 generators and nonsmooth fuel cost. It can be noticed that the solution drops quickly (only after 10 iterations) under the average best solution reported so far and smoothly decreases in time trying to gather better solutions, as near as possible to the global.

C. Robustness test

Due to the random process that heuristic algorithms involve, the robustness test is carried out over a number of independent trials. For all three test systems the GAAPI found the best average reported so far and shows its superiority by reaching a better average than the best reported so far, as can be seen in Tables I and II. The average solution indicates the consistency

of the best solution over the independent trials, always satisfying the equality and inequality constraints. For the smooth 6-generator test system, GAAPI gives comparable results with PSO or LM methods, and better average than GA.
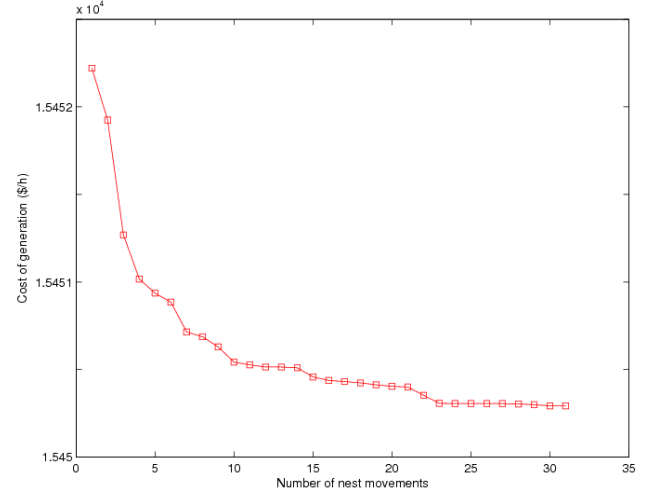

Fig. 4. Convergence characteristics of the GAAPI algorithm (6 - generator test system)

TABLE I
COMPARISON OF AVERAGE RESULTS FOR THE NONSMOOTH TEST SYSTEMS

Test system	Cost of generation ($/h)		
	GA binary	PSO	GAAPI
3-generator	8237.6	8237.2 (GARC)	8235.08
6-generator	16296.63	15855.29	15631.66
15-generator	33228.00	33039.00	32735.06

TABLE II
STATISTICAL RESULTS FOR THE SMOOTH TEST SYSTEM (6 GENERATOR TEST SYSTEM)

	Cost of generation ($/h)			
Method	Max	Min	Average	Standard deviation
GA binary	15519.87	15451.66	15469.21	15.73
GA	15524.00	15459.00	15469.00	Not available
PSO	15492.00	15450.00	15454.00	Not available
PSO-LRS	15455.00	15450.0	15454.00	Not available
GAAPI	15449.85	15449.78	15449.81	0.011

Table III and Fig. 5 give an insight to the robustness characteristics of the GAAPI algorithm in finding the best generation cost for the third benchmark system (15-generator test system).

TABLE III
STATISTICAL RESULTS FOR THE 15-GENERATOR TEST SYSTEM

	Cost of generation ($/h)			
Method	Max	Min	Average	Standard deviation
GA	33337.00	33113.00	33228.00	0.0061*
PSO	33331.00	32858.00	33039.00	0.007*
GAAPI	32756.01	32732.95	32735.06	0.011

* Standard deviation is computed over the evaluation value of the fitness function [6]
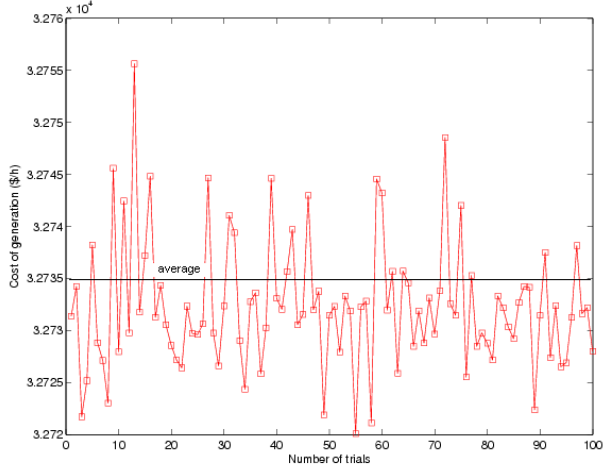
Fig. 5. GAAPI performance in finding the best solution in 100 independent trials (15-generator test system)

D. Comparison of the best solution

For all three test systems the best solution is compared to the best solutions reported in the literature so far. For the 3-generator test system, being a small test system the global solution is known [23], therefore what counts more is the average result obtained by GAAPI, proving its robustness in attaining the nearest point solution to the global. Table IV presents a comparison of the best results obtained for this test system with different methods.

TABLE IV
BEST SOLUTION OF THE 3- GENERATOR TEST SYSTEM FOR A SMOOTH COST FUNCTION

Unit output (MW)	GA	IEP	EP	MPSO	GAAPI
P_1	300.00	300.23	300.26	300.27	300.25
P_2	400.00	400.00	400.00	400.00	399.98
P_3	150.00	149.77	149.74	149.73	149.77
Total output	850	850	850	850	850
Generation cost ($/h)	8237.6	8234.09	8234.07	8234.07	8234.07

For the 6-generator test system with smooth and nonsmooth cost of generation, the best results reported so far are summarized in Tables V and VI, respectively. It was shown that in the case where the cost function is smooth, the GAAPI method, PSO and LM method have comparable results and better than the GA algorithm in terms of the best cost of generation. For the nonsmooth generation cost function, GAAPI proved its superiority against GA algorithms (binary or real coded).

A comparison of the best solutions obtained with different heuristic methods for the third test system examined in this paper (the 15-genarator test system), is given in Table VII. As can be seen, in all cases GAAPI proved to give comparable results or better in terms of the best solution obtained in a number of independent trials. However, its superiority is given by the average value of the generation cost, when for all three benchmark systems GAAPI gives the best average.

TABLE V
BEST SOLUTION OF THE 6-GENERATOR TEST SYSTEM (SMOOTH)

Unit output (MW)	LM	GA binary	GARC	PSO	NPSO-LRS	GAAPI
P_1	447.00	456.46	474.81	447.50	446.96	447.12
P_2	173.50	168.26	178.64	173.32	173.39	173.41
P_3	264.00	258.68	262.21	263.47	262.34	264.11
P_4	138.50	132.66	134.28	139.06	139.51	138.31
P_5	166.04	170.97	151.90	165.48	164.70	166.02
P_6	87.00	89.10	74.18	87.13	89.01	87.00
Losses	13.00	13.13	13.02	12.96	12.93	12.98
Total output	1276.00	1276.13	1276.03	1276.01	1275.94	1275.97
Generation cost ($/h)	15450.00	15451.66	15459.00	15450.00	15450.00	15449.78

TABLE VI
BEST SOLUTION OF THE 6-GENERATOR TEST SYSTEM (NONSMOOTH)

Unit output (MW)	GA binary	GARC	GAAPI
P_1	408.24	495.09	499.98
P_2	194.09	150.45	199.89
P_3	263.42	223.11	225.75
P_4	138.93	149.40	124.95
P_5	155.39	147.94	150.19
P_6	115.72	109.72	74.97
Losses	12.89	12.07	13.13
Total output	1275.83	1275.70	1276.13
Gneration cost ($/h)	15938.08	15634.70	15607.47

TABLE VII
BEST SOLUTION OF THE 15-GENERATOR TEST SYSTEM

Unit output (MW)	PSO	SOH_PSO	GAAPI
P_1	455.00	455.00	454.70
P_2	380.00	380.00	380.00
P_3	130.00	130.00	130.00
P_4	129.28	130.00	129.53
P_5	164.77	170.00	170.00
P_6	460.00	459.96	460.00
P_7	424.52	430.00	429.71
P_8	60.00	117.53	75.35
P_9	25.00	77.90	34.96
P_{10}	160.00	119.54	160.00
P_{11}	80.00	54.50	79.75
P_{12}	72.62	80.00	80.00
P_{13}	25.00	25.00	34.21
P_{14}	44.83	17.86	21.14
P_{15}	49.42	15.00	21.02
Losses	30.49	32.28	30.36
Total output	2660.44	2662.29	2660.36
Generation cost ($/h)	32798.69	32751.39	32732.95

V. CONCLUSIONS

This paper presents a novel algorithm, named GAAPI, emerging from the hybridization process of GA and API strategies, to solve the nonconvex economic load dispatch problem. It was proven that starting from the solution obtained for the quadratic form of the generation cost function (Lagrange multipliers method), the search space can be reduced, and implicitly the computational effort can be reduced. The strategy for handling the constraints is to always generate feasible solutions and work only with these feasible solutions during the search process. Compared to the penalty method, this strategy has the advantage of not dealing with

other parameter settings that complicate user ability to use the method.

The proposed algorithm is relatively simple to implement and manage, and proved to always find comparable or even better solutions in a number of independent trials. The GAAPI has provided the global solution always satisfying the constraints and proved its superiority in robustness by its high probability to reach the global or quasi-global solution, especially in nonconvex formulation. GAAPI converges smoothly to the global, avoiding fast convergence that may lead to local optima.

VI. References

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

[2] M. Shahidehpour, H. Yamin and Z. Li, *Market Operations in Electric Power System*, New York: John Wiley and Sons Inc., 2002.

[3] Z. Li and M. Shahidehpour, "Generation scheduling with thermal stress constraints," *IEEE Trans. on Power Systems*, vol. 18, pp. 1402 – 1409, Nov, 2003.

[4] S. Coelho and V.C. Mariani, "Combining of chaotic differential evolution and quadratic programming for economic dispatch optimization with valve-point effect," *IEEE Trans. on Power Systems*, vol. 21, pp. 989 – 996, May 2006.

[5] D. W. Ross and S. Kim, "Dynamic economic dispatch of generation," *IEEE Trans. on Power Apparatus and Systems*, vol. PAS-99, no. 6, pp. 2060-2068, Nov./Dec. 1980.

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

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

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

[9] C. L. Chiang, "Improved genetic algorithm for power economic dispatch of units with valve-point effects and multiple fuels," *IEEE Trans. on Power Systems*, vol. 20, no. 4, pp. 1690-1699, Nov. 2005.

[10] K. S. Swarup and S. Yamashiro, "Unit commitment solution methodologies using genetic algorithm," *IEEE Trans. Power Syst.*, vol. 17, pp. 87–91, Feb. 2002.

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

[12] G. Bakare, U. O. Aliyu, G. K. Venayagamoorthy, and Y. K. Shu'aibu, "Genetic Algorithms Based Economic Dispatch with Application to Coordination of Nigerian Thermal Power Plants," *IEEE PES General Meeting*, San Francisco, CA, USA, June 2005.

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

[14] 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, pp. 112–118, Feb. 1996.

[15] P. Venkatesh, R. Gnannadassa, E. Pandimeena, G. Ravi, R. Chakrabarti, and S. Choudhary, "A, improved evolutionary programming based economic load dispatch of generators with prohibited operating zones," *J. Inst. Eng. (India)*, vol. 86, pt. EL, pp. 39–44, Jun.2005.

[16] K. S. Swaroop and P. R. Kumar, "A new evolutionary computation technique for economic dispatch with security constraints," *Int. J. Elect. Power Energy Syst.*, vol. 28, pp. 273–283, 2006.

[17] K.P. Wong and C.C. Fung, "Simulated annealing based economic dispatch algorithm," *IEE Proceedings on Generation, Transmission and Distribution*, vol. 40, pp. 509 – 515, Nov 1993.

[18] C. T. Su and C. T. Lin, "New approach with a Hopfield modeling framework to economic dispatch," *IEEE Trans. on Power Systems*, vol. 15, no. 2, pp. 541-545, May 2000.

[19] A. Y. Abdelaziz, S. F. Mekhamer, M. Badr, M. Z. Kamh, "Economic dispatch using an enhanced Hopfield neural network," *Electric Power Components and Systems*, Vol. 36, pages 719 – 732, July 2008.

[20] A. Mohammadi and M.H. Varahram,. "Using neural network for solving of on-line economic dispatch problem," *International Conference on Intelligent Agents for Modelling, Control and Automation*, pp. 81 – 87, Nov. 2006.

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

[22] W.M. Lin; F.S.. Cheng; M.T. Tsay, "An improved tabu search for economic dispatch with multiple minima," *IEEE Trans. on Power Systems*, vol. 17, no. 1, pp. 108-112, Feb. 2002.

[23] J.-B. Park, K.-S. Lee, J.-R. Shin, and K. Y. Lee, "A particle swarm optimization for economic dispatch with nonsmooth cost functions," *IEEE Trans. Power Syst.*, vol. 20, no. 1, pp. 34–42, Feb. 2005.

[24] D. N. Jeyakumar, T. Jayabarathi, and T. Raghunathan, "Particle swarm optimization for various types of economic dispatch problems," *Int. J. Elect. Power Energy Syst.*, vol. 28, no. 1, pp. 36–42, 2006.

[25] K. T. Chaturvedi, M. Pandit, and L. Srivastava,"Self-organizing hierarchical particle swarm optimization for nonconvex economic dispatch," *IEEE Trans. Power Syst.*, vol. 23, no. 3, pp. 1079–1087, Aug. 2008.

[26] A. I. Selvakumar, and K. Thanushkodi, "A new Particle swarm optimization solution to nonconvex economic dispatch problems," *IEEE Trans. on Power Systems*, vol. 22, no. 1, pp. 42-51, Feb. 2007.

[27] K. P. Wong and Y. W. Wong, "Thermal generator scheduling using hybrid genetic/simulated annealing approach," *IEE Proceedings on Generation, Transmission and Distribution*, vol. 142, pp. 372–380, July 1995.

[28] P. Attaviriyanupap, H. Kita, E. Tanaka, and J. Hasegawa, "A hybrid EP and SQP for dynamic economic dispatch with nonsmooth fuel cost function," *IEEE Trans. on Power Systems*, vol. 17, no. 2, pp. 411-416, May 2002.

[29] C. M. Huang and F. L. Wang, "An RBF network with OLS and EPSO algorithms for real-time power dispatch," *IEEE Trans. on Power Systems*, vol. 22, no. 1, pp. 96-104, Feb. 2007.

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

[31] D. Goldberg, *Genetic algorithms in search, optimization, and machine learning*. Reading, MA: Addison-Wesley, 1989.

[32] Z. Michalewicz, *Genetic algorithms + Data Structures = Evolution Programs*. Berlin, Germany: Springer-Verlag, 1994.

[33] M. Takahashi and H. Kita, "A crossover operator using independent component analysis for real-coded genetic algorithms," *Congress on Evolutionary Computation*, vol. 1, pp. 643-649, May 2001.

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

[35] D. Fresneau, *Biologie et comportement social d'une fourmi ponérine néotropicale (Pachycondyla apicalis)*, Ph.D. Thesis, Laboratoire d'Ethologie Expérimentale et Comparée, Université de Paris XIII, France, 1994.

VII. Biographies

Irina Ciornei (S'07) is a Ph.D. student in the Department of Electrical and Computer Engineering at the University of Cyprus and researcher of the KIOS Research Center for Intelligent Systems and Networks. Her research interests include economic dispatch of generation, computational intelligence optimization methods, and energy generation from renewable sources.

Elias Kyriakides (S'00, M'02) is from Nicosia, Cyprus. He received the B.Sc. degree from the Illinois Institute of Technology in Chicago, Illinois in 2000, and the M.Sc. and Ph.D. degrees from Arizona State University in Tempe, Arizona in 2001 and 2003 respectively, all in Electrical Engineering. He is currently an Assistant Professor in the Department of Electrical and Computer Engineering at the University of Cyprus, and a founding member of the KIOS Research Center for Intelligent Systems and Networks. His research interests include modeling of electric machines, security and reliability of the power system network, optimization of power system operation techniques, and energy generation from renewable sources.