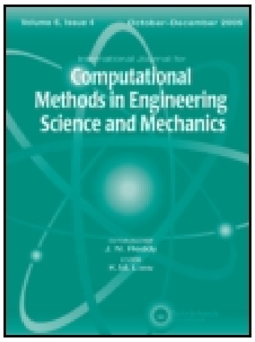

Taylor & Francis
Taylor & Francis Group

International Journal for Computational Methods in Engineering Science and Mechanics

ISSN: 1550-2287 (Print) 1550-2295 (Online) Journal homepage: www.tandfonline.com/journals/ucme20

Directional Search Genetic Algorithm Applications to Economic Dispatch of Thermal Units

Adhinarayanan Theerthamalai & Sydulu Maheswarapu

To cite this article: Adhinarayanan Theerthamalai & Sydulu Maheswarapu (2008) Directional Search Genetic Algorithm Applications to Economic Dispatch of Thermal Units, International Journal for Computational Methods in Engineering Science and Mechanics, 9:4, 211-216, DOI: 10.1080/15502280802069954

To link to this article: https://doi.org/10.1080/15502280802069954

Published online: 02 Jun 2008.

International Journal for Computational Methods in Engineering Science and Mechanics, 9:211–216, 2008

ISSN: 1550–2287 print / 1550–2295 online
DOI: 10.1080/15502280802069954

Directional Search Genetic Algorithm Applications to Economic Dispatch of Thermal Units

Adhinarayanan Theerthamalai and Sydulu Maheswarapu
National Institute of Technology, Andhrapradesh, India

This paper proposes a novel directional search genetic algorithm (DSGA) for solving the Economic Dispatch (ED) problem with fixed boundary conditions. In this paper, a new population generation technique has been employed by using Lambda Limits updation technique. The incremental cost of the generating unit is taken as the decision variable. The feasibility of the proposed method is demonstrated for three different systems, and is compared with various methods available in literature. The proposed algorithm works well for both small- and large-scale generating units and it remarkably reduces the computation time, number of iterations taken to converge, and population size.

Keywords Economic Dispatch (ED), Incremental cost, Lambda Limits updation technique, Directional Search Genetic Algorithm (DSGA), fixed boundary conditions

1. INTRODUCTION

The economic dispatch (ED) problem is an optimization problem and its objective is to reduce the total generation cost of units, while satisfying constraints. Previous efforts on solving ED problems have employed various mathematical programming methods and optimization techniques. These conventional methods include traditional lambda-iteration method, the base point and participation factors method, and the gradient method [1, 2]. For a large-scale generating system, the conventional method has an oscillatory problem resulting in a longer solution time. Unlike traditional algorithms, dynamic programming (DP) imposes no restrictions on the nature of the cost curves and therefore it can solve ED problems with inherently nonlinear and discontinuous cost curves. This method, however, suffers from the "curse of dimensionality" or local optimality [1, 2].

In order to make the numerical methods more convenient for solving ED problems, artificial intelligence techniques, such as the Hopfield neural networks, have been successfully employed to solve ED problems for units with piecewise quadratic fuel cost functions and prohibited zones [3, 4]. However an unsuitable sigmoidal function adopted in the Hopfield model may suffer from excessive numerical iterations, resulting in huge calculations. In the past decade, a global optimization technique known as genetic algorithm (GA) or simulated annealing (SA), which is a form of probabilistic heuristic algorithm, has been successfully used to solve power optimization problems such as feeder reconfiguration and capacitor placement in a distribution system [1, 9, 11]. The GA method is usually faster than SA method because the GA has parallel search techniques, which emulate natural genetic operations. Due to its high potential for global optimization, GA has received a great deal of attention in solving ED problems. Though the GA methods have been employed successfully to solve complex optimization problems, recent research has identified some deficiencies in GA performance. This degradation in efficiency is apparent in applications with highly epistatic objective functions (i.e., where the parameters being optimized are highly correlated) the crossover and mutation operations cannot ensure better fitness offspring because chromosomes in the population have similar structures and their average fitness is high toward the end of the evolutionary process [12–14]. Moreover, the premature convergence of GA degrades its performance and reduces its search capability that leads to a higher probability toward obtaining a local optimum [13]. In this paper, an improved version of genetic algorithm (GA) is proposed. It is named as Directional Search Genetic Algorithm (DSGA) because, in this approach, instead of searching for final best fit chromosome in the entire random search space, a direction is followed to reach the solution. Hence the total search space is narrowed down to a single route. This results in reduction of total execution time, number of iterations and population size, which are the salient features of the proposed method.

2. PROBLEM FORMULATION

The economic dispatch problem minimizes the following cost function associated with dispatchable units.

$$\textit{Minimize}: \quad F_T = \sum_{i=1}^{k} F_i(P_i) = \sum_{i=1}^{k} (a_i + b_i P_i + c_i P_i^2) \tag{1}$$

Address correspondence to Adhinarayanan Theerthamalai, National Institute of Technology, Electrical Engineering, NIT Warangal, Andhrapradesh 506004, India. E-mail: adhi_71@yahoo.co.in

where F_T is the total fuel cost in \$/hr, a_i, b_i and c_i are coefficients of fuel cost functions, P_i is power generation of unit-i in MW, F_i (P_i) is the generation cost of P_i\$/hr, k is the total no. of units.

Subject to constraints:

(i) *Equality Constraints*:

$$\sum_{i=1}^{k} P_i = P_D \tag{2}$$

where P_D is the System Load Demand in MW.

(ii) *Inequality Constraints*:

$$P_{i\,\min} \leq P_i \leq P_{i\,\max} \tag{3}$$

where P_{imin} is the minimum generation of unit-i in MW and P_{imax} represents the maximum generation of unit-i in MW.

3. SOLUTION METHODOLOGY

3.1. Description of Directional Search Genetic Algorithm [DSGA]

DSGA is an improved version of genetic algorithm (GA) and is more efficient for problems with fixed boundary conditions. In this method, instead of searching for final best fit chromosome in the entire random search space, a direction is followed to reach the solution. Hence the total search space is narrowed down to a single route. This results in reduction of total execution time, number of iterations taken to converge and population size. Hence DSGA can overcome the disadvantages of conventional GA. In the conventional GA, initial population is generated randomly. This may contain a smaller portion of best fit chromosomes and a large portion of worst fit chromosomes. Then this population is evaluated and processed through various genetic operators of GA to generate a new population and this process is carried out until global optimum point is reached. As the worst fit chromosomes are very far from the final best fit solution, performing evaluation and genetic operations on them needs a large execution time.

To overcome this drawback, a new Directional Search Genetic Algorithm (DSGA) is proposed. In DSGA, while generating the initial population itself a directional procedure is followed. In the initial population generation, the first chromosome is generated randomly and evaluated. Using this information of first chromosome the next chromosome generation is directed. Hence it will have better fitness value compared to the previous chromosome. In this way all chromosomes are generated using the information of their immediate previous chromosomes and using the direction that they have to follow. This is done until the entire population is filled. This population acts as the basic population. Then this population is processed through various genetic operations as in the case of conventional GA. As a large portion of the basic new population contains best fit chromosomes, it converges rapidly and needs less population size compared to conventional GA. The salient features of DSGA are low population sizes, fewer number of iterations to converge and small execution time. The DSGA approach for ED with fixed boundary condition is explained below.

TABLE 1
Unit data for the 3 on-line units

Unit	a_i (\$/hr)	b_i (\$/MWhr)	c_i (\$/MW^{2}hr)	P_{min} MW	P_{max} MW
1	561	7.92	0.001562	150	600
2	310	7.85	0.00194	100	400
3	78	7.97	0.00482	50	200

3.2. Demonstration of Directional Search Genetic Algorithm [DSGA] for Three Unit Test System

Consider a 3 units data as shown in equations 4, 5, 6 and in Table 1.

$$\frac{dF_1}{dP_1} = 7.92 + 0.003124P_1 = \alpha_1 + \beta_1 P_1 \tag{4}$$

$$\frac{dF_2}{dP_2} = 7.85 + 0.00388P_2 = \alpha_2 + \beta_2 P_2 \tag{5}$$

$$\frac{dF_3}{dP_3} = 7.97 + 0.00964P_3 = \alpha_3 + \beta_3 P_3 \tag{6}$$

From the above given data, the incremental cost at each P_{limit} is calculated as below:

$$\left.\frac{dF_1}{dP_1}\right|_{P_{1\min}} = 7.92 + 0.003124P_{1\min} = 8.388\$/\text{MWh}$$

$$\left.\frac{dF_1}{dP_1}\right|_{P_{1\max}} = 7.92 + 0.003124P_{1\max} = 9.794\$/\text{MWh}$$

$$\lambda_{1\min} = 8.388\$/\text{MWh} \qquad \lambda_{1\max} = 9.794\$/\text{MWh}$$

Similarly,

$$\lambda_{2\min} = 8.238\$/\text{MWh} \qquad \lambda_{2\max} = 9.402\$/\text{MWh}$$

$$\lambda_{3\min} = 8.452\$/\text{MWh} \qquad \lambda_{3\max} = 9.898\$/\text{MWh}$$

Out of 6 λ-values, the λ_{min} and λ_{max} are as: $\lambda_{min} = 8.238\$/\text{MWh}$; $\lambda_{max} = 9.898\$/\text{MWh}$. These two values would be acting as fixed boundary conditions for λ-value of ED problem. For a given demand PD, the corresponding λ-must lie within λ_{min} and λ_{max} values. This is a well-known fundamental observation for ED problem.

Let $P_D = 850$ MW be the specified demand and let us consider $\lambda_d = 0.4908$ is the decoded value of the first chromosome of GA, then the corresponding

$$\lambda_{act} = \lambda_{min} + \lambda_d(\lambda_{max} - \lambda_{min}) \tag{7}$$

$$\lambda_{act} = 8.238 + 0.4908(9.898 - 8.238)$$

$$\lambda_{act} = 9.05289\$/\text{MWh}$$

The power allocation of each generator can be determined as follows:

$$P_1 = \frac{\lambda_{act} - \alpha_1}{\beta_1} = \frac{9.05289 - 7.92}{0.003124} = 362.642MW \quad (8)$$

Similarly P_2 and P_3:

$P_2 = 310.024$ MW, $P_3 = 112.333$ MW
Total generation $= P_1 + P_2 + P_3 = 785$ MW
diff $= P_D - \text{Pgen} = 850 - 785 = 65$ MW

When *diff* is positive, it means that total generation is less than demand. Hence, λ has to be increased over and above the present λ_{act}. Then the future λ would be between $\lambda_{act} = 9.05289$/MWh and $\lambda_{max} = 9.898$/MWh. Hence, the λ_{min} can be replaced by λ_{act}, and two new boundary values of λ are identified in proper direction. In this way direction for next chromosome is decided.

$$\lambda_{min} = \lambda_{act} = 9.05289\$/\text{MWh}.$$

This new range of λ_{min} and λ_{max} is dynamically fixed with chromosome count. Let us consider $\lambda_d = 0.7888$ is the decoded value of the first chromosome of GA, then the corresponding λ_{act} can be calculated using equation (7).

$$\lambda_{act} = 8.238 + 0.6888(9.898 - 8.238) = 9.3814\$/\text{MWh}$$

The power allocation of each generator can be determined as follows:

$$P_1 = \frac{\lambda_{act} - \alpha_1}{\beta_1} = \frac{9.3814 - 7.92}{0.003124} = 467.797\,\text{MW} \quad (9)$$

Similarly P_2 and P_3:

$P_2 = 394.690$ MW, $P_3 = 146.410$ MW
Total generation $= P_1 + P_2 + P_3 = 785$ MW
diff $= P_D - \text{Pgen} = 850 - 1008.89 = -158.897$ MW

When *diff* is negative, it means that total generation is more than demand. Hence, λ has to be decreased below the present λ_{act}. Then the future λ would be between $\lambda_{min} = 8.238$/MWh and $\lambda_{max} = 9.3814$/MWh. Hence, the λ_{max} can be replaced by λ_{act}, and two new boundary values of λ are identified in proper direction.

Hence $\lambda_{max} = \lambda_{act} = 9.3814$ \$/MWh. Then generate the next chromosome randomly, decode it and calculate normalized λ_d using (10). Now calculate actual value using modified minimum and maximum limits of lambda using (5). Here no restriction is put on the random generation of next chromosome. The domain in which λ_{final} lies is adjusted by appropriately setting the values of λ_{min} and λ_{max}. This approach is very reliable and converges very fast, maybe before generation of one full length population chromosome.

3.3. Proposed Directional Search Genetic Algorithm [DSGA] for Solving Economic Dispatch Problem

The detailed algorithm for solving the economic dispatch problem using lambda based Directional Search Genetic Algorithm [DSGA] employed with Lambda Limits technique as given below.

STEP 1: Read generator data, P limits, power demand and GA parameters. Calculate minimum incremental cost λ_{min} and maximum incremental cost λ_{max} for the given set of units using P-limits.

STEP 2: Generate initial population of chromosome

2.1 Set chromosome count = 1.
2.2 Initially generate a chromosome randomly.
2.3 Decode the chromosome and determine Normalized value using

$$\lambda_d = \sum_{i=1}^{16}(d_i * 2^{-i}) \qquad d_i \in \{0, 1\} \quad (10)$$

2.4 Determine actual value of lambda using actual system incremental cost λ_{act} and calculate powers of all units subjected to P-limits.

$$\lambda_{act} = \lambda_{min} + \lambda_d(\lambda_{max} - \lambda_{min}) \quad (11)$$

2.5 Then calculate difference between demand and generation (*diff*) and decide the direction for next chromosome generation.

$$diff = P_D - \sum_{i=1}^{k} P_i \quad (12)$$

Use λ_{act} to set new boundary limits of minimum incremental cost (λ_{min}) or to maximum incremental cost (λ_{max}) depending on the direction decided.

$$If(diff > 0) \quad set \quad \lambda_{min} = \lambda_{act} \quad (13)$$

diff > 0 implies that P_{gen} is less than P_D and the new λ –value would be between λ_{act} and λ_{max}. Thus, set λ_{min} to λ_{act} by $\lambda_{min} = \lambda_{act}$.

$$If\,(diff < 0)\, set \lambda_{max} = \lambda_{act} \quad (14)$$

diff < 0 implies that P_{gen} is more than P_D and the new λ –value would be between λ_{min} and λ_{act}.
Thus, set λ_{max} to λ_{act} by $\lambda_{min} = \lambda_{act}$.
If (error < tolerance) go to step 14.

2.6 Then generate the next chromosome randomly, decode it and calculate normalized value and actual value.
2.7 Increment chromosome count and repeat the procedure for step 2.3 until the below-condition satisfied chromosome count > population size.

STEP 3: Set iteration count = 1

STEP 4: Calculate the fitness values of chromosomes.

$$Fit = \frac{1}{1 + \left[\frac{error}{P_D}\right]} \tag{15}$$

STEP 5: Sort the chromosomes and all their related data in the descending order of fitness.

STEP 6: Check if the error of first chromosome is less than tolerance. If yes, go to step 14.

STEP 7: Elitism is the process of selecting the better individuals, or more to the point, selecting an individual with a bias towards the better ones. Copy the percentage elitism (P_e%) chromosomes of old population to new population starting from the best ones from the top.

STEP 8: Perform crossover on selected parents and generate new child chromosomes; repeat it to get required number of chromosomes.

STEP 9: Add all the generated child chromosomes to new population.

STEP 10: Perform mutation on all chromosomes.

STEP 11: Replace old population with new population.

STEP 12: Then evaluate new population.

12.1 Decode the chromosomes and calculate actual value of lambda (λ_{act}).

12.2 Calculate generating Power output (P_i) of all units using equation (9).
Subjected to P_i Limits.

$$If \quad P_i < P_{i\,\min}, \quad set \quad P_i = P_{i\,\min}$$
$$If \quad P_i < P_{i\,\max}, \quad set \quad P_i = P_{i\,\max}$$

12.3 Calculate fitness values of all chromosomes using (15).

STEP 13: Increment iteration count. If iteration count < max. iteration, go to step 5; otherwise go to step 14.

STEP 14: Calculate the generating power output of all units, total fuel cost. Stop the calculation process.

TABLE 2
Optimal generation levels

Unit Load (MW)	Optimal power generation [MW] 750	1080	1140
1	346.204	517.495	562.81
2	296.789	399.989	399.98
3	107.006	162.5156	177.2
TC ($/hr)	7286.72	10338.77	10915.41

TABLE 3
Comparison of total generation cost of on-line units for 750 MW load

Methods	Total generation cost ($/hr)	Number of iterations to converge	Population size	Execution time in (sec.)
GA	7286.76	18	30	0.125
Proposed DSGA	7286.72	1	20	0.001

4. RESULTS

The above algorithm is tested on 3 units [2], 18 units [10], and 40 units [4] test system data. The effectiveness of the proposed approach is demonstrated by solving the test cases as shown below. Parameters used in directional search GA method:

(i) Number of Bits = 16
(ii) Maximum Population Size = 30
(iii) Elitism Probability = 0.15
(iv) Crossover Probability = 0.8
(v) Mutation Probability = 0.01

4.1. Test Case 1

There are 3 on-line units to supply various load demand as shown in the unit data of Table 1.

From these results, it is clearly shown that the proposed DSGA method converges faster than standard GA. The proposed

TABLE 4
Unit data for the 18 on-line units

Unit	a_i ($/hr)	b_i ($/MWhr)	c_i ($/MW^{2}hr)	P_{min} MW	P_{max} MW
1	85.74158	22.45526	0.602842	7	15.00
2	85.74158	22.45526	0.602842	7	45.00
3	108.98377	22.45526	0.214263	13	25.00
4	49.06263	26.75263	0.077837	16	25.00
5	49.06263	26.75263	0.077837	16	25.00
6	677.7300	80.39345	0.734763	3	14.75
7	677.7300	80.39345	0.734763	3	14.75
8	44.39000	13.19474	0.514474	3	12.28
9	44.39000	13.19474	0.514474	3	12.28
10	44.39000	13.19474	0.514474	3	12.28
11	44.39000	13.19474	0.514474	3	12.28
12	574.9603	56.70947	0.657079	3	24.00
13	820.3776	84.67579	1.236474	3	16.20
14	603.0237	59.59026	0.394571	3	36.20
15	567.9363	56.70947	0.420789	3	45.00
16	567.9363	55.96500	0.420789	3	37.00
17	567.9363	55.96500	0.420789	3	45.00
18	820.3776	84.67579	1.236474	3	16.20

TABLE 5
Optimal generation levels

Unit Load (MW)	Optimal power generation [MW] 346.576	368.237	411.559
1	15.00	15.00	15.00
2	45.00	45.00	45.00
3	25.00	25.00	25.00
4	25.00	25.00	25.00
5	25.00	25.00	25.00
6	3.00	4.66	13.70
7	3.00	4.66	13.70
8	12.28	12.28	12.28
9	12.28	12.28	12.28
10	12.28	12.28	12.28
11	12.28	12.28	12.28
12	20.72	23.24	24.00
13	3.00	3.00	6.412
14	30.86	35.05	36.20
15	32.36	36.29	45.00
16	33.24	37.00	37.00
17	33.24	37.18	45.00
18	3.00	3.00	6.412
TC [$/hr]	23855.27	25710.44	29731.39

method takes single iteration to converge whereas the standard GA takes 18 iterations to converge. It is observed that the proposed method performs better than the standard GA in terms of computational burden. Please see Tables 2 and 3.

4.2. Test Case 2

There are 18 on-line units to supply various load demand. The unit characteristics are described in Table 4.

From Table 5, the total fuel cost for various demand has shown using the proposed DSGA method. The proposed method takes a single iteration to converge whereas the standard GA takes 2 iterations to converge. It is observed in Table 6 that the proposed method performs better than the standard GA in terms of computational burden, too.

TABLE 6
Comparison of total generation cost of on-line units for 346.576 MW load

Methods	Total generation cost ($/hr)	Number of iterations to converge	Population size	Execution time in sec.
GA	23857.54	8	60	0.235
Proposed DSGA	23855.27	2	21	0.047

4.3. Test Case 3

Case 3 considers the system that consists of 40 units in the realistic Taipower system that is large-scale and a mixed generating system where coal-fired, oil-fired, gas-fired, diesel, and combined cycle are all present [4]; please see Tables 7 and 8. The load demand of the system is 9500 MW.

TABLE 7
Optimal generation of 40 units system

Unit	Optimal generation using DSGA
1	80.000
2	120.000
3	190.000
4	42.000
5	42.000
6	140.000
7	300.000
8	300.000
9	300.000
10	152.494
11	171.571
12	171.984
13	267.480
14	393.084
15	395.175
16	395.175
17	395.175
18	500.000
19	500.000
20	550.000
21	550.000
22	550.000
23	550.000
24	550.000
25	550.000
26	550.000
27	550.000
28	10.952
29	10.952
30	10.952
31	20.000
32	20.000
33	20.000
34	20.000
35	18.000
36	18.000
37	20.000
38	25.000
39	25.000
40	25.000
Fuel cost ($/hr)	128424.2600

TABLE 8
Comparison of total generation cost of on-line units

No. of units	Load (MW)	Total cost (k$/h) Proposed Method	Classical Method [9]	Hopfield Neural Network [PHN] [9]
40	9500	128.4	128.4	129.1

From the results in tables 7 and 8, it is clearly shown that the proposed method converges faster than the classical method [9] and PHN [9]. The proposed method takes single iteration to converge whereas the PHN takes 9 iterations to converge. It is observed that the proposed method performs better than the PHN in the above case and it produces the optimal generation cost.

5. CONCLUSIONS

This paper presents a novel Directional Search Genetic Algorithm (DSGA) with lambda updation technique for solving the economic dispatch of thermal units. In contrast to other conventional methods, this approach gives a promising value of power for providing improved economic dispatch. The proposed method reduces the computational burden and is either matching or superior to many of available techniques of economic dispatch and it can be applied to various problems in power systems having fixed boundaries. The method yields a true optimal solution for large-scale systems without any convergence problems, which has been illustrated through three test cases.

REFERENCES

1. B.H. Chowdhury and S. Rahman, A Review of Recent Advances in Economic Dispatch, IEEE Trans. Power Syst., vol. 5(4), pp. 1248–1259, 1990.
2. A.J. Wood and B.F. Wollenberg, Power Generation, Operation, and Control, 2nd ed., John Wiley and Sons, New York 1996.
3. R. Naresh, J. Dubey, and J. Sharma, Two-phase Neural Network Based Modeling Framework of Constrained Economic Load Dispatch, IEE Proc.-Gener., Trans., Distrib. vol. 151, pp. 373–378, 2004.
4. P.H. Chen and H.C. Chang, Large-scale Economic Dispatch by Genetic Algorithm, IEEE Trans. Power Syst., vol. 10, pp. 1919–1926, 1995.
5. K.P. Wong and C.C. Fung, Simulated Annealing Based Economic Dispatch Algorithm, IEE Proc., Gener., Transm. Distrib., vol. 140, pp. 509–515, 1993.
6. K.P. Wong and Y.W. Wong, Genetic and Generic/simulated Annealing Approaches to Economic Dispatch, IEE Proc., Gener., Transm. Distrib., vol. 141, pp. 507–513, 1994.
7. W.M. Lin, F.S. Cheng, and M.T. Tsay, Nonconvex Economic Dispatch by Integrated Artificial Intelligence, IEEE Trans. Power Syst., vol. 16(2), pp. 307–311, 2001.
8. T. Su and C.T. Lin, New Approach with Hopfield Modeling Framework to Economic Dispatch, IEEE Trans. Power Syst., vol. 15(2), pp. 541–545, 2000.
9. T. Yalcinoz and M.J. Short, Large-scale Economic Dispatch Using an Improved Hopfield Neural Network, IEE Proc., Gener., Transm. Distrib., vol. 144, pp. 181–185, 1997.
10. I. G. Damousis, A. G. Bakirtzis, and P. S. Dokopoulos, Network Constrained Economic Dispatch Using Real-Coded Genetic Algorithm, IEEE Trans. Power Syst., vol. 18(1), pp. 198–205, 2003.
11. G. B. Sheble and K. Brittig, Refined Genetic Algorithm Economic Dispatch Example, IEEE Trans. Power Syst., vol. 10, pp. 117–124, Feb. 1995.
12. Zwe-Lee Gaing, Particle Swarm Optimisation to Solving the Economic Dispatch Considering the Generator Constraints, IEEE Trans. Power Syst., vol. 18, pp. 1187–1195, Aug. 2003.
13. Jong-Bae Park, Ki-Song Lee, Joong-Rin Shin, and Kwang Y. Lee, A Particle Swarm Optimisation for Economic Dispatch with Nonsmooth Cost Functions, IEEE Trans. Power Syst., vol. 20, pp. 34–42, Feb. 2005.
14. V. Bhuvaneshwaran, R. Langari, and C. Temponi, A Genetic Algorithm and Fuzzy Approach (GAFA) for Constrained Nonlinear Optimization in Design, International Journal for Computational Methods in Engineering Science and Mechanics, vol. 7(3), pp. 173–189, 2006.