

Journal of Asian Scientific Research

Online ISSN 2223-1331

Vol 1 No 5

September, 2011



Asian Economic and Social Society
www.aessweb.com



Solution Of Multi – Objective Optimization Power System Problems Using Hybrid Algorithm

Abstract

Author

S.Jaganathan

Assistant professor , Electrical Engineering RVS College of Engg and Tech Coimbatore, Tamilnadu, India
E-mail: jaganathangct@gmail.com

S.Palaniswami

Professor, Electrical Engineering Government College of Tech Coimbatore, Tamilnadu, India E-mail: joeget81@yahoo.com

Key words:

Fast Bacterial Foraging algorithms, Hybrid Technique, Sequential Quadratic Programming.

Generally the two most common approaches to solve multiple objectives are: combine them into a single objective function and obtain a single solution, obtain set of non-dominated Pareto optimal solutions. Thus there is a need to bridge the gap between single solutions and Pareto optimal sets. The Pareto set includes all rational choices, among which the decision maker has to select the final solution by trading the objectives against each other. The search is then not for one optimal solution but for a set of solution that are optimal in a border sense. There are a number of techniques to search the solution for Pareto optimal solutions. The objective of this search is to achieve this balance, by introducing two practical methods that reduce the Pareto optimal set to achieve a smaller set called the “pruned pareto set”.

Multiple, often conflicting objectives arise naturally in most real-world optimization scenarios. As Fast Bacterial Foraging algorithms possess several characteristics that are desirable for this type of problem, this class of search strategies has been used for multi-objective optimization for more than a decade. Meanwhile Fast Bacterial Foraging algorithms multi-objective optimization has become established as a separate sub discipline combining the fields of Fast Bacterial Foraging computation and classical multiple criteria decision making.

A new hybrid technique algorithm is presented for the solution of the comprehensive model of real world problems. This method is developed in such a way that a simple Fast Bacterial Foraging algorithms is applied as a base level search, which can give a good direction to the optimal global region and a local search Sequential Quadratic Programming (SQP) is used as a fine tuning to determine the optimal solution at the end.

Introduction

This paper gives an overview of Fast Bacterial Foraging multi-objective optimization Sequential Quadratic Programming. On the one hand, basic principles of multi-objective optimization and Fast Bacterial Foraging algorithms are presented, and various algorithmic concepts such as chemotaxis, cell to cell communication, reproduction and elimination and dispersal are discussed. Many (perhaps most) real-world design problems are, in fact, Multiobjective optimization problems in which the designer seeks to optimize simultaneously several performance attributes of the design and an improvement in one objective is often only gained at the cost of deteriorations in other objectives trade-offs are

necessary. There are two standard methods for treating Multiobjective problems, if a traditional optimization algorithm which minimizes a single objective is to be employed.

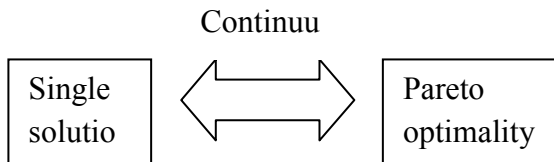
Multi-Objective Optimization Problems

Multiple objective problems are solved using a variety of different approaches [S.Jaganathan¹] [S.Jaganathan³]. Often the multi-objective is combined into a single objective so that optimization and mathematical methods can be used. In general, for a problem with n objective functions, the multi-objective formulation can be as follows

$$\begin{aligned} &\text{Minimize/maximize } f_i(x) \text{ for } i=1, 2, 3 \dots n \\ &\text{Subject to} \\ &G_j(x) \leq 0 \quad j=1, 2 \dots J \\ &H_k(x) = 0 \quad k=1, 2 \dots K \end{aligned}$$

There are n objectives and p variables so $f(x)$ is an n dimensional vector and x is a p dimensional vector corresponding to p decisions or variables, solutions to a multi-objective optimization problem are often mathematically expressed in terms of non-dominated or superior points. X is defined as the set of feasible solutions or feasible decision alternatives. Thus, in a maximization problem x is non-dominated in X . Then the optimal solutions to a multi objective optimization problem are in the set of non dominated solutions N [C.M Fonseca][Miroslav M.Begovic], and they are usually known as pareto optimal set.

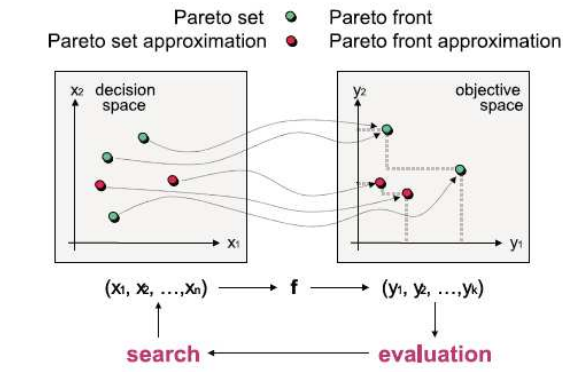
Figure.1. Multi objective optimization general block diagram.



The scenario considered in this paper involves an arbitrary optimization problem with k objectives, which are, without loss of generality, all to be maximized and all equally important, i.e., no additional knowledge about the problem is available [S.Jaganathan²] [C.A.C.Coello]. We assume that a solution to this problem can be described in terms of a decision vector (x_1, x_2, \dots, x_n) in the decision space X . A function $f : X \rightarrow Y$ evaluates the quality of a specific solution by assigning it an objective vector (y_1, y_2, \dots, y_k) in the objective space Y . Now, let us suppose that the objective space is a subset of the real numbers, i.e., $Y \subseteq \mathbb{R}^k$, and that the goal of the optimization is to maximize the single objective. In such a single-objective optimization problem, a solution $x_1 \in X$ is better than another solution $x_2 \in X$ if $y_1 > y_2$ where $y_1 = f(x_1)$ and $y_2 = f(x_2)$. Although several optimal solutions may exist in decision space, they are all mapped to the same objective vector, i.e., there exists only a single optimum in objective space. In the case of a vector-valued evaluation function f with $Y \subseteq \mathbb{R}^k$ and $k > 1$, the situation of comparing two solutions x_1 and x_2 is more complex. Following the well known concept of Pareto dominance, an objective vector y_1 is said to dominate another objective vectors y_2 ($y_1 \succ y_2$) if no component of y_1 is smaller than the corresponding component of y_2 and at least one component is greater. Accordingly, we can say that a solution x_1 is better to another solution x_2 , i.e., x_1 dominates x_2 ($x_1 \succ x_2$), if $f(x_1)$ dominates $f(x_2)$. Here, optimal

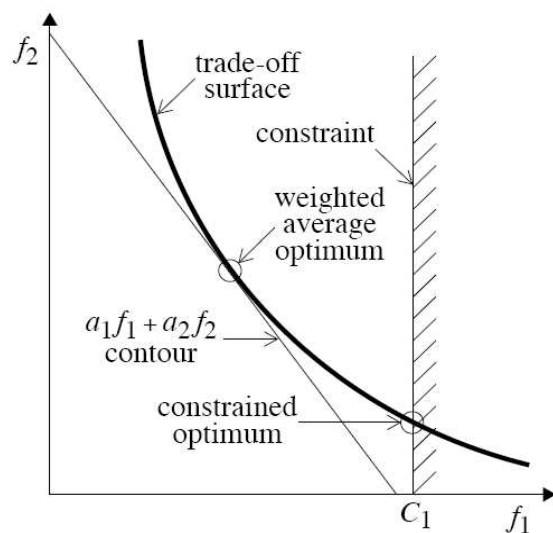
solutions, i.e., solutions not dominated by any other solution, may be mapped to different objective vectors. In other words: there may exist several optimal objective vectors representing different trade-offs between the objectives.

Figure 2. Illustration of a general multi-objective optimization problem



Eventually a single solution must be chosen, but it is self-evident that the designer will make a better informed decision if the trade-off surface between the conflicting objectives can be inspected before this choice is made. By using suitably adapted stochastic optimization methods it is possible to reveal the trade-off surface of a multi-objective optimization problem in a single run.

Figure 3. Types of Multi-objective optimum



Whichever of these approaches is used, the solution of the one objective problem so produced results in the identification of a single point on the trade-off surface, the position of which depends on the designer's preconceptions. In the following sections appropriate adaptations to standard Genetic Algorithm (GA) and

Simulated Annealing (SA) implementations will be discussed [E.Zitzler].

By using suitably adapted stochastic optimization methods it is possible to reveal the trade-off surface of a Multiobjective optimization problem in a single run. In the following sections appropriate adaptations to standard Bacterial foraging Algorithm (BFO) [S.Mishra] implementations will be discussed. Adapting any stochastic optimization algorithm to perform Multiobjective optimization will inevitably require a common change to the method of archiving. In Multiobjective optimization solutions lying on the trade-off surface (or Pareto front as it is also known) are sought. Any solution on the Pareto front can be identified formally by the fact that it is not dominated by any other possible solution.

Multiobjective And Bacterial Foraging

The fact that BFA search from population to population rather than from one individual solution to another makes them very well suited to performing Multiobjective optimization. It is easy to conceive of a population being evolved onto the trade-off surface by a suitably configured BFA.

In fact, with an appropriate archiving scheme in place, the only modification required to a single objective GA, in order to perform Multiobjective optimization, is in the selection scheme. As with single objective GAs, a wide variety of Multiobjective selection schemes have been devised. Three of the most widely used (and most easily implemented) will be described here.

Bacterial foraging algorithm, which is tailored for optimizing difficult numerical functions and based on metaphor of human social interaction. Its key concept is that potential solutions are flown through hyperspace and are accelerated towards better or more optimum solutions. Its paradigm can be implemented in simple form of computer codes and is computationally inexpensive in terms of both memory requirements and speed.

Implementation Of Fast Bacterial Swarming Algorithm With Multi-Objective Optimization

A. Fast Bacterial Swarming Algorithm

i) Principle of Swarming:

Conventional method: The swarming pattern of the cell-to-cell attraction and repulsion in bacterial foraging algorithm (BFA) has negative effects, which has been indicated and verified earlier. At the beginning of the optimization of the process, the bacteria are scattered into random locations in the optimization domain. Then the bacteria gather to the

local optima, and finally converge to the global optimum. By analyzing we know that where there are more bacteria, there has lower fitness value. Therefore, the bacteria in the local optima attract those in the global optimum and the convergence speed of the population is pulled down.

Novel method: Inspired by the swarming pattern in BFO [S.Jaganathan³], a novel principle of swarming for FBSA is introduced. It is assumed that the bacteria have the similar ability like birds to follow the best bacterium (bacterium with the best position in the previous chemotactic process) in the optimization domain. The position of each bacterium after every move (tumble or run) is updated according to

$$\Theta^{i(j+1,k,l)} = \Theta^{i(j,k,l)} + C_{cc} \times (\Theta^b(j,k,l) - \Theta^{i(j,k,l)}) \quad (39)$$

Where

$(\Theta^b(j,k,l))$ = the positions of best bacterium

C_{cc} = new parameter

In FBSA the bacteria in historically global worse positions abandon their former environment and quickly flock to the neighborhood of the probable global optimum.

ii) step length:

In BFA the step length C is a constant. If C is too large, the bacteria may miss the global optimum by swimming through it without stop; if C is too small, it takes a large amount of time to find the global optimum. So C influences both the accuracy and speed of the search. An adaptive step length $C(k, l)$ is defined [S.Jaganathan³] [S.Mishra]. The size of the step length is dynamically adjusted in the reproduction and elimination-dispersal process, which ensures the bacteria moving towards the global optimum quickly at the beginning, and converging to the global optimum accurately in the end.

$$C(k, l) = L_{red} \ln^{k+1}$$

Where

L_{red} = the initial size of the chemotactic step length

n = constant controlling the decreasing rate of the step length at the k_{th} reproduction loop in the l_{th} elimination-dispersal event.

B. Sequential Quadratic Programming

SQP used here is from the MATLAB toolbox. The MATLAB SQP implementation consists of three main

stages, which are discussed briefly in the following subsections

- Updating of the Hessian matrix of the
- Lagrangian function Quadratic programming (QP) problem solution
- Line search and merit function calculation

For, each iteration QP is solved to obtain the search direction. These are used to update the control variables. QP problem can be described as follows.

Minimize the following

$$\frac{1}{2} d_k^T H_k d_k + \nabla f(x_k)^T d_k$$

Subject to the following:

$$[\nabla g(x_k)]^T d_k + g_i(x_k) = 0 \quad i = 1, \dots, m_e$$

$$[\nabla g(x_k)]^T d_k + g_i(x_k) \leq 0 \quad i = 1, \dots, m_e + 1, \dots, m$$

where

H_k Hessian matrix of the Lagrangian function defined by

$$L(x, \lambda) = f(x_k) + \lambda^T g_i(x) \text{ at } x = x_k$$

d_k basis for a search direction at iteration k ;

$f(x)$ objective function;

$g(x)$ constraints;

m_e number of equality constraints;

m number of constraints;

Any non differentiable parameters are approximated using finite differences mentioned above equation. SQP used in this paper consists of three main stages as follows.

Hybrid Algorithm for Multi-Objective Optimization

Step 1: Randomly initialize the position of each bacterium in the domain, set the position and fitness value of the best bacterium as $\theta^b(j, k, l)$ and $J_{\min}(j, k, l)$ respectively.

FOR (elimination-dispersal loop $l=1: N_{ed}$)
 FOR (reproduction loop $k=1: N_{re}$)

FOR (chemotactic loop $j=1: N_c$)
 FOR (each bacterium $i=1: S$)
 Calculate $J^i(j, k, l)$ and set it as J_{last} ;

Step 2:

Tumble: Generate a random angle ϕ belongs to $(0, 2\pi)$ and move to the direction by a unit walk; the new position $\theta^i(j+1, k, l)$ is calculated. Calculate $J^i(j+1, k, l)$ and set it as J_{current} .

Swarm:

Update $\theta^i(j+1, k, l)$.
 Recalculate $J^i(j+1, k, l)$;

IF ($J_{\text{current}} < J_{\text{last}}$)

WHILE ($J_{r+1}^i(j+1, k, l) < J_r^i(j+1, k, l)$ and $r < N_s$)

Set $J_r^i(j+1, k, l)$ as J_{last} ;

Run:

Update $\theta^i(j+1, k, l)$
 Set $J_{r+1}^i(j+1, k, l)$ as J_{current} ;

Swarm:

Update $\theta^i(j+1, k, l)$
 Recalculate $J_{r+1}^i(j+1, k, l)$;

END WHILE

END IF

END FOR (bacterium)

Calculate $J_{\min}(j+1, k, l)$

END FOR (chemotaxis)

Step 3:

Sum:

Evaluate the sum of the fitness value J_{health}^i for the i_{th} bacterium.

Sort:

Sort J_{health}^i in an ascending order of fitness values;

Step 4:

Split and eliminate:

Select the best half S_r bacteria to split and the other bacteria are eliminated.

Update:

Update the step length $C(k, l)$.
 END FOR (reproduction)

Step 5:

Disperse:

Disperse certain bacteria to random places in the optimization domain with probability P_{ed} .

Update:

Update the step length $C(k, l)$
 END FOR (elimination-dispersal)
 END

Use the answer from above as starting points of SQP. Here the FBSA solution result is taken as initial solution of SQP solution after that the multi objective solution is achieved.

Update:

Updating of the Hessian matrix with help of global values.

Step 6:

Initiate the values of d_k basis of given values.

Step 7:

Estimate the values of m_e and m values from basis of d_k .

Step 8:

Estimate the Lagrangian function Quadratic programming (QP) problem solution.

Step 9:

Finally estimate Line search and merit function calculation.

Step 10:

With help of above results find out the best global solution.

Step 11:

If it's optimal solution is achieved means, the program will stopped.

Case Studies and Discussion

I have selected case study from power system problem and particularly Multi-objective optimal power flow problem with FACTS devices and the objectives function are transmission losses, voltage profile optimization, real and reactive power dispatch and minimization of cost (generation cost, cost of reactive power devices).

This optimization problem consists of set of constraints also included (Set of power flow equations, real power limits, voltage limits, Tap-setting limits, reactive power limits and line flow limits). The section presents the results of simulation of FACTS devices included in IEEE 30 bus system to evaluate the proposed method also present a comparison with other methods, Bacterial foraging algorithms and Particle swarm optimization.

The Newton-Raphson load flow technique is used to evaluate line flow calculation. In this case study, the results obtained by proposed method for large scale system were presented. In this example, the proposed algorithm was implemented in MATLAB and executed on Pentium IV personal computer.

A. Comparison of Results

i) Voltage profile optimization

To show the effectiveness of Hybrid approach formulated and it's used to solve the voltage profile optimization and check the load ability of system subjected to power flow equations under various operating conditions. The proposed method gives better Voltage profile under various operating conditions.

The quality of solution based on better Voltage magnitude irrespective of operating conditions. So the proposed method enhances better voltage profile optimization. The lower and upper limits of all buses except slack buses were taken 0.95 to 1.1 respectively.

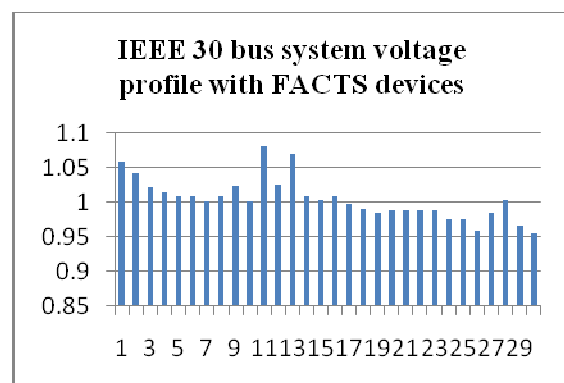
ii) Quality of Solution.

The proposed and new methodology algorithm is used to evaluate the correct location of FACTS devices and solution OPF, based on variations in load demand. The proposed Hybrid algorithm shows better solution with considerable computational time. The quality of solution based on generation cost, transmission losses and optimal location of FACTS devices.

The proposed method finding minimal value of generation cost is 801.0316 with FACTS devices and considerable reduces in transmission losses 9.173. The proposed method converges level very high, with in considerable time its gives better global solution and number iteration also very less. Therefore computational time very less compared other techniques.

The IEEE 30-bus test system also is used to verify the effectiveness of the proposed algorithm. It has six generators at buses 1, 2, 5, 8, 11, and 13 and four transformer with off-nominal tap ratio in lines 6-9, 6-10, 4-12, 27-28 and also nine buses for the reactive power compensation. The voltage profile optimization results are shown in figure.4.

Figure.4 IEEE 30 bus system Voltage profile with FACTS Devices



Conclusion

Optimization problems involving multiple objectives are common. In this context, Fast Bacterial Foraging algorithms computation represents a valuable tool, in particular. Flexibility is important if the underlying model is not fixed and may change or needs further refinement. The advantage of Fast Bacterial Foraging algorithms is that they have minimum requirements regarding the problem formulation; objectives can be easily added, removed, or modified. Moreover, due the fact that they operate on a set of solution Fast Bacterial Foraging algorithms are well-suited to generate Pareto set approximations. This is reflected by the rapidly increasing interest in the field of Fast Bacterial Foraging algorithms multi-objective optimization. Finally, it has been demonstrated in various applications that Fast Bacterial Foraging algorithms are able to tackle highly complex problems and therefore they can be seen as an approach complementary to traditional methods such as integer linear programming.

The performance of developed Hybrid Technique algorithm has been tested various power system problems. The algorithm has accurately and reliably converged to the global optimum solution. The algorithm is also capable of producing more favourable for real world problems. Therefore, the proposed approach can be used to improve quality obtained by other existing technique.

References

- C.A.C.Coello (199)** "A Comprehensive Survey of Evolutionary Based Multiobjective Optimization Technique" Knowledge and Information system, vol.1, no.3, 9, pp.269-308.
- C.M Fonseca and P.J. Fleming (1995)** "An overview of evolutionary algorithm Multiobjective Optimization", evolutionary computation, vol.3, no.1, pp.1-16.
- E.Zitzler and L.Thiele (1998)** "An Evolutionary Algorithm for Multiobjective Optimization: The Strength Pareto Approach" Swiss Federal Institute of Technology, TIK-report no.43.
- M.Z.EL-Sadek, A. Ahmed, M. A. Mohammed (2007)** " Advanced Modeling of FACTS in Newton Power Flow", Journal of Engineering Sciences, Assuit University, Vol. 35,No. 6, pp. 1467-1479.
- Miroslav M.Begovic, Branislav Radibratovic, and Frank C.Lambert (2004)** "On Multiobjective Volt-Var compensator in power systems" Hawaii international conference on system sciences.
- S.Jaganathan, S.Palaniswami, R.Mithunraj and G.Maharajavignesh (2011)** "Application of Multi-objective optimization in Reactive power planning problem using a Ant colony Algorithm" European Journal of Scientific Research Vol 2, No.51, pp.241-253.
- S.Jaganathan,S.Palaniswami, C.Sasikumar and R.Muthukumaran (2010)** "Multi Objective Optimization for Transmission Network Expansion Planning using Modified Bacterial Foraging Technique" International Journal of Computer Applications Vol.9, No.3, pp.0975 – 8887.
- S.Jaganathan, S.Palaniswami, B,Rajesh, K.Senthilkumaravel (2011)** "Application of Multi-objective Technique to Incorporate UPFC in Optimal Power Flow Using Modified Bacterial Foraging Technique", International Journal of Computer Applications, Vol.13, No.2, pp.0975 – 8887.
- S. Mishra and C. N. Bhende (2007)** "Bacterial Foraging Technique-Based Optimized Active Power Filter for Load Compensation" IEEE Transactions on power delivery, Vol. 22, No. 1, pp. 457-465

Aims and Scope

The Journal of Asian Scientific Research is a monthly, peer-reviewed international research journal which deals with empirical as well as theoretical issues. The editors welcome papers in all the major issues including:

- Research Articles and Reviews Article
- Scientific Commentaries in the Fields of Applied and Theoretical Sciences
- Biology
- Chemistry
- Physics
- Zoology
- Medical Studies
- Environmental Sciences
- Geology
- Engineering
- Short Communications
- Computer Science
- Social Sciences
- Natural Sciences
- Technological Sciences
- Linguistics
- Medicine
- Industrial
- Mathematics
- Statistics
- All other Applied and Theoretical Sciences

Indexed / abstracted in

- J-Gate
- New Jour
- ULRICHS WEB Global Serials Directory

Instructions for Authors

All articles are subject to a double blind peer-reviewed process. Manuscripts are invited from academicians, researchers and practitioners for publication consider in all areas related to applied sciences. The each manuscript must include 200 to 400 words abstract. Articles are accepted only in MS-Word format.

Authors are requested to prepare their articles by following the guidelines given below:

Author(s) introduction: Should be on a separate page consisting of full name, position, affiliation, and e-mail address

Font Size: 12 points with double spacing

1. Heading 16 point font sizes
2. Sub Heading 14 point font sizes
3. Text 12 point font sizes

Format: Times New Roman

Length: Maximum 7,000 words

Title: Should be impressive and not lengthy

Abstract: Should be 200 to 400 words

Keywords and JEL code: Most appropriate keywords and code should be used. JEL Code

Equations, Tables and Diagrams: Should be clear & accurate and on the right place

Reference: Should be completed according to the following guidelines

Arrow, K. (1970) "The Organization of Economic Activity: Issues Pertinent to the Choice of Market Versus Non-market Allocations" in Public Expenditure and Policy Analysis by R.H. Havenman and J. Margolis, Eds., Markham: Chicago, pp. 67-81

Benabou, Roland (1994) "Education, Income Distribution, and Growth: The Local Connection". NBER working paper number 4798

Berglas, E. (1976) "Distribution of tastes and skills and the provision of local public goods". Journal of Public Economics Vol. 6, No.2, pp.409-423.

Edgeworth, F.Y. (1881) Mathematical Psychics, Kegan Paul: London.

Mas-Colell, A and J. Silvestre (1991) "A Note on Cost-Share Equilibrium and Owner- Consumers" Journal of Economic Theory Vol.54, No.1,pp. 204-14.

Appendix: At the end of the paper

Author(s) pays a publication fee USD \$ 100 for each accepted article