We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,300

Open access books available

172,000

International authors and editors

190M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Parameters Determination for Optimum Design by Evolutionary Algorithm

Wen-Jye Shyr

*Department of Industrial Education and Technology,
National Changhua University of Education, Changhua 500,
Taiwan, R. O. C.*

1. Introduction

The finding a maximum or a minimum function problem under some constraint conditions are called optimization problem. Almost every engineering design problem can be formulated as optimum problems. Solving the optimum problem requires the computation of the global maxima or minima of the objection function. Many heuristic intelligent algorithms had been developed and adapted to several optimal design problems. Heuristic algorithms have merit that they can search the global optimum with a higher probability than deterministic ones. Evolutionary algorithms are stochastic search methods that mimic the metaphor of natural biological evolution and/or the social behavior of species (Shyr, 2008). Obviously, in order to reach this goal makes the search process complicate and the selection of an optimum technique critical. It is a challenge for engineers to design efficient and cost-effect systems without compromising the integrity of the system. The conventional design process depends on the designer's intuition, experience, and skill.

There are several kinds of numerical optimization methods such as neural network, gradient-based search, genetic algorithm, etc (Rao, 1979). Neural network can simulate the relation between the input and output. But it needs samples for training the network first. Sometimes the gradient-based search is fast and efficient, but it is easy to get stuck in local extreme. Compared with them, the genetic algorithm has its special characteristics. No sample is needed for the implementation of genetic algorithm and the most important is that genetic algorithm can derive a global optimum by mutation and crossover technique so as to avoid being trapped in local optima. It is considered as a technique the most suitable for combinatorial optimization design.

The concept of Genetic Algorithm (GA) was first established by Holland (Holland, 1975), based on the mechanism of nature selection and evolutionary genetics. The purpose of the genetic algorithm is to find a better function via some simulation artificial operation process, which includes evaluation, selection, crossover and mutation. Genetic algorithm is used in optimum design because of its efficient optimum capabilities. The genetic algorithm is an efficient tool in the field of engineering education (Bütün, 2005).

2. Optimum design problem formulation

The aim of the optimum design course is to find the best possible combination of solutions, which can be termed as design parameters to maximize or minimize an optimization function.

Source: Convergence and Hybrid Information Technologies, Book edited by: Marius Crisan,
ISBN 978-953-307-068-1, pp. 426, March 2010, INTECH, Croatia, downloaded from SCIYO.COM

It is generally assumed in this course that various preliminary analyses have been completed and a detailed design of a concept or a sub problem needs to be carried out. Students should bear in mind that a considerable number of analyses usually have to be performed before reaching this stage of the design optimization problem and it must be stressed because the optimum solution will only be as good as the formulation. Once the problem is properly formulated, good software is usually available to solve it. In this paper, the optimum design software is supported by a genetic algorithm for undergraduate students.

Fig. 1 shows the formulation procedure for the design optimization problems that involve translating a descriptive statement of the problem into a well defined mathematical statement. Detailed steps of formulation procedure are as follows (Arora, 2004):

Step 1: Project/problem statement

The formulation process begins by developing a descriptive statement for the project/problem, which is usually done by the project's owner/sponsor. The statement describes the overall objectives of the project and the requirements to be met.

Step 2: Data and information collection

To develop a mathematical formulation of the problem, students need to gather material properties, performance requirement, resource limits, and other relevant information. Some of the design data and expressions may depend on design variables that are identified in the next step.

Step 3: Identification/definition of design variables

The next step in the formulation process is to identify a set of variables that describe the system, called design variables. The design variables should be independent of each other as far as possible.

Step 4: Identification of a criterion to be optimized

There can be many feasible designs for a system and some are better than others. To compare different designs, it must have a criterion. The criterion must be a scalar function whose numerical value can be obtained once a design is specified. Such a criterion is usually called an objective function for the optimum design problem, which needs to be maximized or minimized depending on problem requirements.

Step 5: Identification of constraints

All restrictions placed on a design are collectively called constraints. The final step in the formulation process is to identify all constraints and develop expressions for them. All these and other constraints must depend on the design variables, since only then do their values change with different trial designs.

3. Fundamentals of genetic algorithm

Genetic algorithm is a numerical optimization technique. More specifically, they are parameter search procedures based upon the mechanics of natural genetics. They combine a Darwinian survival-of-the-fittest strategy with a random. This technique has gained popularity in recent years as a robust optimization tool for a variety of problems in engineering and various problems. The genetic algorithm uses only the function values in the search process to make progress toward a solution without regard to how the functions are evaluated. Continuity of differentiability of the problem functions is neither required nor

Fig. 1. The formulation procedure for design optimization problems

used in calculations of the genetic algorithm. Therefore, the genetic algorithm is very general and can be applied to all kinds of optimum design problems. In addition, the genetic algorithm determines global optimum solutions as opposed to the local solutions determined by a continuous variable optimization algorithm. The algorithm is easy to use and program since it does not require use of gradients of cost function.

The flow chart of genetic algorithm is given in Fig. 2. The implementation of the genetic algorithm described in this section as follows (Goldberg, 1989; Pan et al., 1995):

Step 1: Initialization

Algorithm is started with a set of solutions (represented by chromosomes) called population. A set of chromosomes is randomly generated. A chromosome is composed of genes.

Step 2: Evaluation

For every chromosome, its fitness value is calculated. Check every chromosome's fitness value one by one. Compared with the present best fitness value, if one chromosome can give better fitness, renew the values of the defined vector and variable with this chromosome and its fitness value. Otherwise, keep their values unchanged.

Fig. 2. The flow chart of genetic algorithm

Step 3: Selection

Within the algorithm, population selection is based on the principle of survival of the fittest, which is based on the Darwinian's concept of "Natural Selection" (Dawkins, 1976). A random number generator is used to generate random numbers whose values are between 0 and 1.

Step 4: Crossover

The crossover operation is one of the most important operations in genetic algorithms. The basic idea is to combine some genes from different chromosomes. It is the recombination of bit strings by copying the segments from pairs of chromosomes.

Step 5: Mutation

Some useful genes are not generated in the initial step. This difficulty can be overcome by using the mutation technique. The basic mutation operator randomly generates a number as the crossover position and then changes the value of this gene randomly.

Step 6: Stopping criteria

Steps 2-5 are repeated till the predefined number of generations has been reached. The optimal solution can be generated after termination.

4. The teaching method

The lectures are held in the optimum design laboratory. The teacher explains the conceptions. The examples of genetic algorithm are executed and projected demonstrating the behavior of different optimum design problem. Time is left for the students to run some examples with different parameters realizing an interactive learning process.

The optimum design laboratory serves active problem solving tightly attached to the theoretical material. Each student works on his own computer and solves the optimum design problems by himself. The teacher sets up the problem and gives a permanent guidance. The students build up genetic algorithm programs or combine the programs from given software. The results are then evaluated. Besides worked out examples further programs are also provided. The examination is held in the optimum design laboratory and consists of solving an optimum problem which is assigned by the teacher. The solution is accepted only if the program works in a right way. Genetic algorithm is available during the examination programs.

Interested students may use their knowledge in further optimum design projects. It has to be mentioned that nowadays students are attracted much more by projects connected to optimal control. Students may be better interested in optimum design projects if these are connected somehow to optimal control.

5. Test functions and simulation results

There are three examples examined in this section. Example 1 examined a single variable, and example 2 examined two variables. Three variables are examined in example 3.

Example 1

The objective function of optimum engineering design problems is described as follows (Lindfield & Penny, 1995):

Step 1: Project/problem statement

A manufacturer wishes to produce a wall mounting container which consists of a hemisphere surmounted by a cylinder of fixed height.

Step 2: Data and information collection

The data and information are given as follows: The height of the cylinder is fixed but the common radius of the cylinder and hemisphere may be varied between 2 and 4. The manufacturer wishes to find the radius value which maximizes the volume of the container.

Step 3: Identification/definition of design variables

The two design variables are defined as follows: r as the common radius of the cylinder and hemisphere and h as the height of the cylinder.

Step 4: Identification of a criterion to be optimized

We can formulate this as an optimization problem by taking r as the common radius of the cylinder and hemisphere and h as the height of the cylinder. Taking $h=2$ units leads to the objective function

$$\text{Maximize } v = \frac{(2*\pi*r^3)}{3} + 2*\pi*r^2 \quad (1)$$

Step 5: Identification of constraints

The radius of the cylinder r is between 2 to 4. The defining parameters are as follows: population size=10, probability of crossover=0.6, probability of mutation=0.005 and the generations=20. The simulation of search space is depicted in Fig. 3 and shows the search space performance via genetic algorithm. The simulation result in genetic algorithm is depicted as follows: common radius of the cylinder r is equal to 4 and the objective function v is equal to 234.5723.

Fig. 3. The search space vs. generation in genetic algorithm (Example 1)

Example 2

The objective function of the optimum design with two variables is described as follows:

$$f(x, y) = x^2 + y^2 \quad (2)$$

where $[x, y] = [1 \ 3; 0 \ 3]$. The population size equals to 20, the probability of crossover is 0.4, the proportion of mutation is 0.05 and the generations are 200.

The solution of this maximization problem is found to be $[x, y] = [3, 3]$, and the maximum value of $f(x, y) = 18$. The simulation of objective function versus generations is depicted in Fig. 4.

Example 3

The objective function of the optimum design with three variables is described as follows:

$$f(x, y, z) = xyz - xy^2z^2 + xy^2z \quad (3)$$

where: $3 \leq x \leq 10$, $2 \leq y \leq 20$, $3 \leq z \leq 7$. The population size equals to 20, the probability of crossover is 0.5, the proportion of mutation is 0.05 and the generations are 200.

For a complicated multivariable function $f(x, y, z)$, it is usually difficult to obtain the global minimum. The genetic algorithm shows its effectiveness to this kind of optimization problem. The solution of this maximization problem is found to be $x = 9.804$, $y = 19.946$, $z = 6.628$, and the minimum value of $f(x, y, z) = -144200$. The simulation of objective function versus generations is depicted in Fig. 5.

Fig. 4. The simulation of objective function versus generations (Example 2)

Fig. 5. The simulation of objective function versus generations (Example 3)

These above examples show that the genetic algorithm is effective for solving the optimum design problem.

6. Parameters identification

The identification of suitable adaptive antenna parameters can be carried out using information from optimizing interference cancellation that can easily be retrieved. The identification of adaptive antenna parameters, as shown in Fig. 6 is performed using the genetic algorithm. The fitness function (Hsu et al., 2005) can be written as follows:

$$AF_n(\theta) = \frac{1}{N} \sum_{n=1}^N \alpha \cos[(n-0.5)\psi + \beta_n] \quad (4)$$

where N =number of elements, α =constant amplitude weight for all elements, β_n =phase shifter weight at element n , $\psi = kd \sin \theta = \frac{2\pi}{\lambda} d \sin \theta$, θ = an incidence angle of interfering signal or desired signal, an interfering signal with wavelength λ impinges on any two adjacent sensor elements by a distance d and from a direction θ with respect to array normal. Using the optimization technique, the fitness function cannot exist the imaginary part. As the equation (4) only the real part is left, it is available for searching the optimal solutions using optimization technique.

Fig. 6. Diagram of an adaptive linear array designed by phase-only perturbations using genetic algorithms

The values assigned to the necessary variables used by genetic algorithms are as follows: population size P equals to 20; the selected survival rate p_s equals 0.5; the probability of crossover is 0.5; the proportion of mutation is 0.05. In addition, in this problem, we assume a linear antenna array composing of 20 isotropic elements. So, $N = 10$. The distance d of any two adjacent elements is half of λ . The value of β_n is set between $-\pi$ and π . The unit of β_n is rad. The fitness function is the square of $AF_n(\theta)$ in equation (4). The value of α is constant and set between 0.1 and 1. In this case, let α equal 1. Pattern nulling skill can be used to suppress

multiple interferences (including noises). This skill is able to do the cancellation of multiple interferences for different incident directions. One example is given as follows. The interfering signal’s direction is from 40⁰ with respect to the array normal. The pattern nulling is derived in the 40⁰ interfering direction. The genetic algorithm stops after 250 generations. The result is listed in Table 1. The convergent map is shown in Fig. 7, and the radiation pattern is shown in Fig. 8. This article focuses on the interference cancellation in order to increase the Signal Interference Ratio (SIR). As we know, when we optimized the signal to interference ratio, the noise can be ignored always. Actually, the main purpose of our paper is to propose a method of adjustable canceling interfering signal in the mobile communication. By phase-only perturbation method, the nulling design of an adaptive antenna has been studied by the approach of the genetic algorithm. The excellent nulling results have been derived. In this example, whatever direction the desired signal is from, the SIR has 40 dB at least shown in Fig. 8. For the nulling design, the interferences can be suppressed effectively.

Nulling direction at 40 ⁰	
$\beta_1 = -0.364$	$\beta_2 = -2.229$
$\beta_3 = -1.394$	$\beta_4 = 0.195$
$\beta_5 = -2.977$	$\beta_6 = 0.402$
$\beta_7 = 1.149$	$\beta_8 = 0.622$
$\beta_9 = -2.016$	$\beta_{10} = 0.088$

Table 1. The weight vector $[\beta_n]$ for pattern nulling in the interfering directions

Fig. 7. Relative interference power getting convergent pattern for nulling direction at 40⁰

7. Conclusions

Genetic algorithm is global stochastic method based on the mechanism of nature selection and evolutionary genetics. Using genetic algorithm, the extreme value of a function is very easy to be solved as these examples. In this paper, genetic algorithm for identifying adaptive antenna parameter was introduced. Pattern nulling design of adaptive antenna by phase-

Fig. 8. Adaptive array pattern for single interfering source at 40°

only perturbations using genetic algorithms is proposed and achieved. This proposed evolutionary algorithm was then successfully applied to the test problems and the adaptive antenna parameter identification, and then proved to be a very promising optimization algorithm for using in optimization problems.

The optimum design problem formulation and the teaching method have been overviewed. Genetic algorithm based optimum design laboratory supported understanding of optimum design problem for engineering education. Studying the optimum design course the students get the experience of solving the optimum design problems by themselves.

8. References

- Arora, J. S. (2004). Introduction to optimum design, Elsevier Academic Press. 2nd.,
- Bütün, E. (2005). Teaching genetic algorithms in electrical engineering education: a problem-based learning approach, International Journal of Electrical Engineering Education, Vol. 42, Issue 3, pp.223-234.
- Dawkins, R. (1976). The selfish gene, Oxford University Press.
- Goldberg, D. E. (1989). Genetic algorithms in search optimization and machine learning, Addison- Wesley Publishing Company.
- Holland, J. H. (1975). Adaptation in natural and artificial systems, Ann Arbor: The University of Michigan Press.
- Hsu, C. H., Shyr, W. J. and Kuo, K. K. (2005). Optimizing interference cancellation of adaptive linear array by phase-only perturbations using genetic algorithms, Lecture Notes in Artificial Intelligence (LNAI), Vol. 3681, pp. 561-567.
- Lindfield, G. and Penny, J. (1995). Numerical methods using MATLAB, Prentice-Hall, Inc.
- Pan, J. S., McInnes, F. R. and Jack, M. A. (1995). Codebook design genetic algorithms, IEE Electronics Letters, Vol. 31, No.17, pp.1418-1419.
- Rao, S. S. (1979). Optimization theory and applications, 2nd.
- Shyr, W. J. (2008). Introduction and comparison of three evolutionary-based Intelligent algorithms for optimal design, Proceeding of International Conference on Convergence and Hybrid Information Technology (ICCIT 2008), Vol.2, pp.879- 884.

Convergence and Hybrid Information Technologies

Edited by Marius Crisan

ISBN 978-953-307-068-1

Hard cover, 426 pages

Publisher InTech

Published online 01, March, 2010

Published in print edition March, 2010

Starting a journey on the new path of converging information technologies is the aim of the present book. Extended on 27 chapters, the book provides the reader with some leading-edge research results regarding algorithms and information models, software frameworks, multimedia, information security, communication networks, and applications. Information technologies are only at the dawn of a massive transformation and adaptation to the complex demands of the new upcoming information society. It is not possible to achieve a thorough view of the field in one book. Nonetheless, the editor hopes that the book can at least offer the first step into the convergence domain of information technologies, and the reader will find it instructive and stimulating.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Wen-Jye Shyr (2010). Parameters Determination for Optimum Design by Evolutionary Algorithm, Convergence and Hybrid Information Technologies, Marius Crisan (Ed.), ISBN: 978-953-307-068-1, InTech, Available from: <http://www.intechopen.com/books/convergence-and-hybrid-information-technologies/parameters-determination-for-optimum-design-by-evolutionary-algorithm>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen