GENETIC ALGORITHMS FOR ROBUST PARAMETER OPTIMIZATION

Nicolo A/L Belavendram, nbelmah@gmail.com; drnic@utar.edu.my
Faculty of Science and Engineering, Universti Tunku Abdul Rahman, Kuala Lumpur.

Introduction

Parameter optimization[1] can be achieved by many methods such as Monte-Carlo, Full factorial design and fractional factorial designs. While Monte-Carlo methods provide good solutions it is very computer intensive. Full and fractional factorial designs provide elegant optimization methods a very repeatable manner. Genetic Algorithms[2] (GA) are fairly recent in this respect but afford a novel method of parameter optimization.

In GA there is a pool of individuals each with its own specific phenotypic trait expressed as a "genetic chromosome". Different genes enable individuals with different fitness levels to reproduce according natural reproductive gene theory. This reproductive theory is well established in terms of Selection, Crossover and Mutation of reproducing genes. The resulting child generation of individuals has a better fitness level as is true in natural selection, namely evolution. Populations then evolve towards the fittest individuals.

Such a mechanism has a parallel application in parameter optimization. Factors in a parameter design can be expressed as a genetic analogue in a pool of sub-optimal random solutions. Allowing this pool of sub-optimal solutions to evolve over several generations produces fitter generations. The solutions converge to an optima meeting some required or pre-defined engineering criteria.

In this paper, several aspects of Genetic Algorithms are studied using a seven factor non-linear equation for a Wheatstone Bridge (Figure 1) as the equation to be optimized. There is also a comparative discussion on the Monte-Carlo, Full factorial design and fractional factorial designs.

Parameter Optimization

Parameter Optimization is an important field of engineering that seeks to find some optimum (i.e. minimum or maximum) point in the design space.

Consider the Wheatstone Bridge circuit. The objective is to measure the value of the unknown resistor, y in such a way that small variations in variables (e.g. resistors A, B, C, D, F and voltage E) do not affect the reading of the value y. In other words, it is not just sufficient to determine the value of y but do so with a high signal-to-noise ratio. Such concepts have been termed robust engineering by Phadke[3].

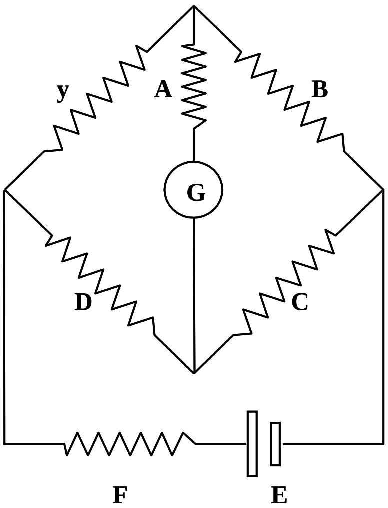

Figure 1 Wheatstone Bridge.

The equation for the unknown resistor is

$$y = \frac{BD}{C} - \frac{G(AD + AC + BD)(BC + BD + BF + CF)}{C^2 E}$$

Genetic Algorithm

For parameter optimization, the selection of the Fitness Function poses some difficulty. If it is required to attain a specified target value with the smallest variability, then both the mean and variance need to be considered. The mean could be targeted at 2 kΩ. The variance however, is not so straight forward. Choosing to evaluate variance directly is not a good method since variance is a function of the size of numbers. Therefore, the quantity $\omega = \frac{y}{\sigma}$ is used. This however, distinguishes $-y$ from $+y$ which is not necessary for the present case. Hence, one can use the function $\eta = \left(\frac{y}{\sigma}\right)^2$. Additionally, one can use the function $\eta = 10\log\left(\frac{y}{\sigma}\right)^2$ to improve additivity. With an arbitrary target of y =2 kΩ and η = 45 dB, it is possible to optimize $f = (y-2)(\eta-45)$ the resulting surface response is a saddle. Clearly, it is not possible to find a suitable minimum on a saddle.

However, the function $F = (y-2)^2(\eta-45)^2$ produces a surface response as shown in Figure 2. Although the region of interest is fairly flat (which is a good thing to have) it is possible to determine the minima. In this paper, this fitness function F is used.

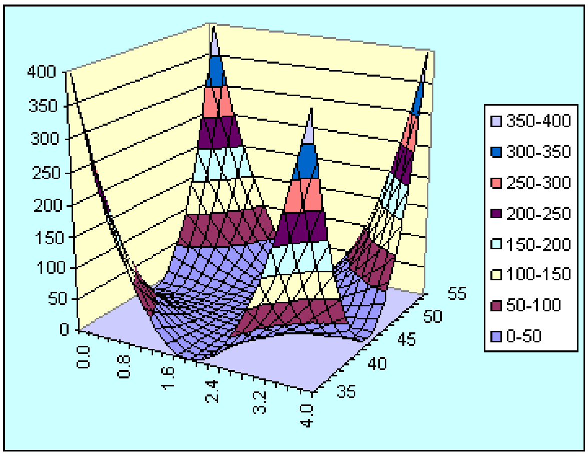

Figure 2 Response Surface of Fitness Function.

The GA method of optimization begins with the expression of engineering factors in some binary code although other codes are also possible[4]. For the Wheatstone bridge, the following factor representation is possible. We take the range of factor values i to be tried and set up eight values according to the binary digits 000, 001, 010, 011, 100, 101, 110, 111. The factor values are set according to $\phi_i = \phi_{\min} + (i-1)\left(\frac{\phi_{\max} - \phi_{\min}}{7}\right)$. The eight levels of ϕ are selected randomly. The response was then calculated for the combination of factor levels. Partial differentials were used to calculate the variances. For a factor ϕ, $\Delta y = \frac{\delta y}{\delta \phi}\Delta\phi$, so that the variance σ^2 of j variables is $\sum_1^j \frac{\delta y}{\delta \phi_j}\Delta\phi_j$ and η can be calculated as above.

For a combination, $F(A_{000}; B_{010}; \ldots, G_{111})$ etc. corresponding to $F(A_1; B_2; \ldots, G_8)$, y, σ^2, η and lastly, the Fitness function is calculated. Indviduals are selected, crossed and mutated to form the next generation. This reproduction is contiued until a specificied criterion (e.g. number of generations, tolerance of target value, etc.) is reached. At this point, the optimum gene pool is used as a solution.

Results

Tracing the Fitness Function of a population overs generations shows that the population comes closer to the solution. Figure 3 compares the optimum solution to the unoptimized solution with a typical variance reduction in the breaktrhough region of 100 times!

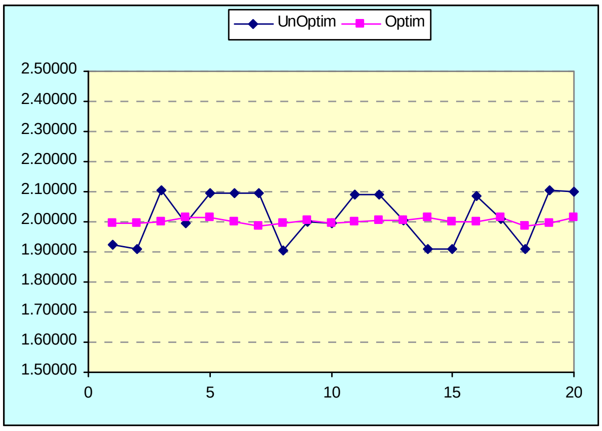

Figure 3 Optimum - Unoptimum results.

Discussion

Approaching as a full factorial experiment, for seven variables, each at eight levels, the number of permulations is 5, 764, 801, typically taking about 1.5 hours on a Pentium IV processor. Fractional factorials based on orthogonal arrays may be used to study the problem as seven variables, each at three levels. While elegant in itself, fractional factorials do not provide a fine comb search. However, it allows the analysis of variance and hence a better understanding of the contribution of a variable to the problem.

Genetic Algorithms provide a radically different approach to finding the solution. In particular, the Selection, Crossover and Mutaiton algorithms provide a superior method of escaping local maxima or minima.

References

Belavendram, *Quality by Design*, 1995, Prentice-Hall, ISBN-10: 0131863622.

http://www.obitko.com/tutorials/genetic-algorithms/index.php

Phadke, Madhav, *Quality Engineering Using Robust Design*, 1989 Prentice Hall.

Randy Haupt and Douglas Werner, *Genetic Algorithms in Electromagnetics*, 2007 Wiley-Interscience.