A Note on Optimality vs. Stability - A Genetic Algorithm based approach.

Sourav Kundu

Department of Human and Mechanical Systems Engineering, Faculty of Engineering, Kanazawa University, 2-40-20 Kodatsuno, Kanazawa 920-8667, Japan. Email: sourav@kenroku.kanazawa-u.ac.jp

1. Abstract

Research on optimality criteria is founded on the underlying notion of "efficiency" which allows us to model the real world in an overly simple way than it really is. The concepts are good enough for research, but in the real world inefficient solutions may be preferred over theoretically more efficient ones, as they are more *stable.* When using well known optimization methods, we make intuitive assumptions that the model from which the solution is derived is absolutely error free. These assumptions, project the obtained solution as much more efficient than what they actually are - had we taken model errors into account. A pragmatic engineering definition of stability would be - "The likelihood for an optimal solution to degenerate into a dominated one, as the bounds get more optimistic". Thus, if one defines the minimal acceptable performance of a solution as P_{min}, then a solution that is most likely to degrade to P_{min} , as constraints are progressively relaxed, should be considered as the least stable of all. In the Multi Criteria Optimization (MCO) scenario, it is a common experience that many Pareto optimal solutions easily degrade to dominated (non Pareto) solutions, for very small changes in the constraints or boundary conditions. The reason that theoretical optimal solutions are often *unstable* compared to sub-optimal or partially dominated solutions is that they often occur on the boundary of the feasible region, that is when most of the equality and inequality constraints are active. Thus a small perturbation in any direction pushes the solution into infeasibility. In this paper a Genetic Algorithm (GA) based method is proposed that evaluates the fitness value of an individual considering the stability of the solution. Here a GA based multicriteria optimization method is presented where a Pareto solution is awarded a fitness value according to its stability criterion; together with its ability to dominate other previously found Pareto Solutions while constraints are relaxed.

1. Keywords

Optimization, Stability Criteria, Pareto Optimality, Genetic Algorithms

2. Problem Formulation

The general nonlinear multicriteria optimization problem is formulated as follows:

$$\frac{\min}{x \in \Re^N} \mathbf{f}(\mathbf{x})$$

under the inequality constraints

$$g_i(\mathbf{x}) \leq 0, i\_1,2,\ldots,m$$

and equality constraints

$$h_j(\mathbf{x}) = 0,\ j = 1,2,\ldots,l$$

where $\mathbf{x} = [x_1, x_2, \ldots x_N]^T$ is a column vector of N real-valued decision variables and

$$\mathbf{f}(\mathbf{x}) = \left[f_1(\mathbf{x}),\ f_2(\mathbf{x}), \ldots, f_K(\mathbf{x}) \right]^T$$

is a vector of K objective functions. For the $\mathbf{f}(\mathbf{x})$ vector, the problem is multicriteria, whereas for an $f(x)$ scalar, the problem is single criterion. The notation $\mathbf{x}_0$ for starting point, $\mathbf{x}^*$ for optimum, and $\mathbf{x}_p$ for the (current) point at p^{th} iteration will be generally used throughout this paper. The solution of the multicriteria optimization problem is to find the set of *Pareto* optimal solutions (non dominated solutions). If X is a set of feasible solutions, i.e., set of solutions which satisfy constraint conditions given above, the Pareto optimum will be defined as : A solution $\mathbf{x}^* \in X$ is Pareto optimal if and only if there exists no $\mathbf{x} \in X$ such that $f_k(\mathbf{x}) \le f_k(\mathbf{x}^*)$ for all k, $k = 1,2,...,K$ with $f_k(\mathbf{x}) < f_k(\mathbf{x}^*)$for at least one k, $1 \le k \le K$. The proposed genetic algorithm based method generates a solution and compares the objective functions considering the above Pareto optimality condition. The objective functions f_k for k = 1,2,...,K, are evaluated using the following formula:

$$f_k(\mathbf{x}) = f_k(\mathbf{x}) + r\sum_{i=1}^{m}\left[g_i(\mathbf{x}) + a\right]^2 + r\sum_{j=1}^{l} G_j\left[h_j(\mathbf{x}) + b\right]^2 ,.......\mathbf{for}\ \ k = 1,2,...K$$

where G_j is the Heaveside operator such that $G_j = 0$ for $h_j(\mathbf{x}) \le 0$ and $G_j=1$ for $h_j(\mathbf{x}) > 0$ and r is a positive multiplier which controls the magnitude of the penalty terms, or the penalty multiplier. a and b are the contraint relaxation terms which are used to check the stability of a Pareto solution under progressive relaxation of constraints. More on a and b will be explained later in the paper.

3. Solution Method: Pareto Set Updating

Throughout each generation of the GA a set of Pareto optimal solutions is maintained (updated each time after the fitness value of a solution is evaluated), all of which share a common single fitness value. All members of a Pareto set has the same shared fitness value. A new solution in a certain generation of the GA can fall in any of the three mutually exclusive and exhaustive categories:

[a.] It is a <u>new</u> Pareto optimal solution, and it <u>dominates</u> some (or all) of the Pareto optimal solutions found up till the immediately preceding GA evaluation run.
[b.] Although it is a <u>new</u> Pareto optimal solution it <u>does not dominate</u> any of the Pareto optimal solutions found up till the immediately preceding GA evaluation run.
[c.] It is <u>not</u> a Pareto optimal solution.

For every new solution in a certain generation, first a fitness value is assigned according to the method described in [Osyczka and Kundu, 1995]. Then we deal with the three different categories mentioned above in three separate ways, whereby a fitness value is returned for the GA selection to be implemented. In this research, tournament selection is implemented.

- For category [a] solutions the raw fitness value (stage 1) is first calculated by <u>adding</u> the distance value, to the shared common fitness value of the Pareto optimal solutions found till the immediately preceding GA evaluation runs and the Pareto set is <u>updated by removing</u> those old solutions that this new Pareto solution dominates. At this point the the values of the terms a and b are progressively changed to relax the constriants and then the newly found solution is again evaluated and compared with the old Pareto set to see if it still dominates some or all of solutions in the old Pareto set. Ranges and steps for a and b are predetermined and amount of constraint relaxation is preset. If the new solution still dominates some or all of the old Pareto solutions, it is taken to be a stable solution and a proportionate final fitness value (stage 2) is returned to the GA.
- For category [b] solutions the raw fitness value (stage 1) is first calculated by <u>adding</u> the distance value to the shared common fitness value of the Pareto optimal solutions found till the immediately preceding GA evaluation run and the Pareto set is <u>updated by adding</u> this new solution to it. At this point the the values of the terms a and b are changed to relax the constraints and then the newly found solution is again evaluated and compared with the old Pareto set to check if it is still a Pareto solution (that is member of the set). Ranges and steps for a and b are predetermined and the amount of constraint relaxation is

preset. If the new solution is still a Pareto solution, it is taken to be a stable solution and a proportionate final fitness value (stage 2) is returned to the GA. This fitness value is then assigned to all the members of the updated Pareto set.

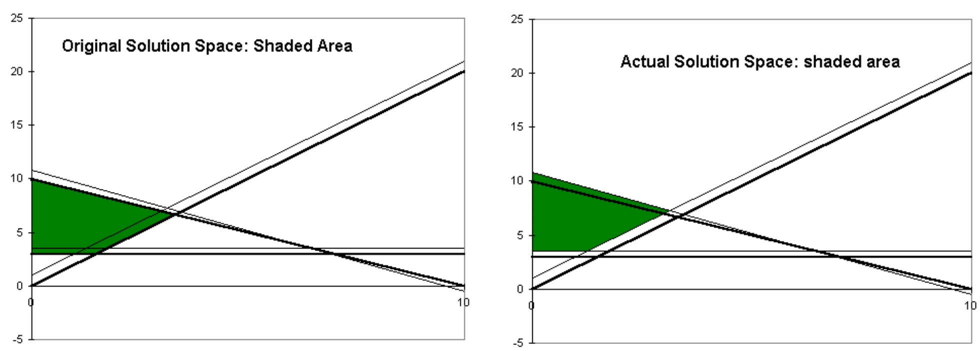
- For category [c] solutions the fitness returned is simply the distance value <u>subtracted</u> from the shared common fitness value of the Pareto Optimal solutions found till the immediately preceding GA evaluation run. The constraint relaxation stability check is not performed in this case.

4. Numerical Examples

An optimization problem of a multiple disc brake and a network optimization problem has been solved with the method outlined in this paper. Due to space restrictions, detail implementation and solutions will be presented in the full paper.

5. Conclusions

In real world optimization, for implementing real solutions, stability is one of the most essential aspect of the solution. The stability of a system is defined here as the likelihood of a theoretical optimal solution with most or all of the constraints being active - that is when solution is on boundary of feasible region - to fall into infeasible region for some stochastic changes in real values of the variables in the real world. For example, let us take a simple case from [Ignizio, 1998] where *original model* is: **Max**(z_1); **Max**(z_2), subject to: $x+y \leq 10$, $2x-y \leq 0$ and $y \geq 3$. The *real world model* could be something like: **Max**(z_1); **Max**(z_2), subject to: $1.08x+0.9537y \leq 10.3$, $1.6x-0.8y \leq -0.8$ and $0.95y \geq 3.325$. The *assumed* versus *actual* solution space will then be represented by the following figure.



Assuming that the boundary of the solution space in the first quadrant, represents the efficient solutions (efficient frontier), it is clear that the solutions are not efficient in the actual model. Some points in the original efficient frontier are actually dominated while others are actually infeasible. This example clearly illustrates that efficient solutions (Pareto solutions in case of Multicriteria problems and optimal solution in case of Single criteria problems) are invariably least stable of the solutions as generated by either conventional or unconventional methods. [Ignizio, 1998] . Thus the Genetic Algorithm based method outlined in this paper, with stability checking, attempts to yield real world solutions rather than producing only theoretically correct optimal solutions.

6. References

[1] Osyczka,A and Kundu,S.: 1995. A new method to solve generalized multicriteria optimization problems using the simple genetic algorithm. *Structural Optimization.* Vol. **10**, No. 2, pp. 94-99.

[2] Ignizio,J.P.: 1998. Illusions of Efficiency; Delusions of Optimality, MOPGP 1998 Conference. (*Presentation slides).*