



Genetic Algorithms, with Inheritance, versus Gradient Optimizers, and GA/Gradient Hybrids

Jeff Finckenor

*NASA/Marshall Space Flight Center/Structural Development
Branch ED52, Huntsville, AL 35812 USA,
Jeff.Finckenor@msfc.nasa.gov*

Abstract

This study compares gradient solvers, Genetic Algorithms (GA's), and two kinds of GA-gradient hybrids.

Gradient optimization methods can rapidly converge to an optimum solution. However, the solution is often a local optimum, particularly if the function is noisy. The local optimum may be far from the global optimum and depends on the user input starting point. Gradient solvers are also unable to perform integer optimization.

GA's are effective at finding global solutions, but require many function evaluations. The GA is an integer optimizer and can lose resolution available to gradient optimizers when operating on a continuous function.

The first hybrid uses the final GA solution as a starting point for the gradient solver. The second hybrid uses each GA individual as a starting point.

The representative problem is a skin-stringer construction cylinder. It is a fairly noisy design space with several discontinuities.

Results compare the weight of the final solution against the function calls required. The most efficient solver is a hybrid with the GA selecting starting points and a fairly small population of 50-100 individuals.

Introduction

The goal of numerical optimization is to provide ideal designs or solutions. In engineering, many applications involve intensive computational effort for a single analysis. Because time and computer resources are limited it is important for optimization methods to provide good solutions using the fewest function evaluations/analyses. The focus problem of this study, a stringer stiffened cylinder, is covered in detail in Finckenor1 and Finckenor2.

The gradient optimizer is ADS, Vanderplaats3. Two gradient methods are used. The Method of Feasible Directions (MFD) Vanderplaats4, with the



Bounded Polynomial Interpolation One-Dimensional Search option. This method is identified as 047, indicating the ADS control flags: Strategy, ISTRAT=0; Optimizer, IOPT=4; and 1D Search, IONED=7. This combination of ADS options provides consistently good, valid solutions with the fewest function calls for this problem.

The second gradient method uses Sequential Unconstrained Minimization with exterior penalty functions. The optimization method is the Davidon-Fletcher-Powell Variable Metric Unconstrained Minimization. The 1D search is the Golden Section followed by polynomial interpolation. This is identified as 122: ISTRAT=1; IOPT=2; IONED=2. This variation generally gives slightly better results than 047, but requires many more function calls.

The GA used is called JGA, Finckenor5. It is based on the simple GA, Goldberg6, uses two competitor tournament selection, two point crossover, and includes inheritance.

The simple GA represents each design variable as a range of integers. The integers are coded into binary and concatenated into a bitstring. Each bitstring is an "individual" and represents a specific design. The initial population is randomly generated and each design is evaluated. The best designs are selected and combined in crossover. A small chance of mutation is included to help insure a diverse population. The new individuals are evaluated and the process repeats until an ending criteria is met.

To save function calls, JGA checks each individual to see if an identical bitstring has already been evaluated in that generation. This has little effect on a diverse population, but can significantly reduce run time when the population begins to converge. When there is a non-zero chance for inheritance, the checking of previous individuals is not performed.

The first hybrid uses the GA solution to provide the starting point for a single gradient optimization. The rationale is that the GA operates well at finding the global solution, and the gradient solution is good at finding the precise optimum point, if you can start near it. For smooth functions, where the peaks and valleys of the objective function are sufficiently larger than the GA resolution, this should work well.

The second hybrid, referred to here as the True Hybrid, uses each member of a GA population as a starting point. This has the effect of driving the GA search to find the optimum gradient starting point, rather than the optimum point. For noisy functions this should be very effective.

Another Type of Hybrid

Another kind of hybrid is presented by Foster and Dulikravich, Foster7, who use gradient methods to apply equality boundary constraints on the GA. Their application is 3-D aerodynamic shape optimization. Each individual is moved along the gradients of the violated constraints until they are active, but not severely violated. Using this method, the global abilities of the GA can



solve problems to which they could not otherwise be applied. This hybrid is competitive with gradient methods in terms of CPU time.

Focus Problem

The skin-stringer cylinder design to be used as the focus problem is similar to a barrel of the intertank of the Space Shuttle External Tank. It is made of 2219-T87 aluminum, has a radius of 165.5", a length of 48", and uses external hat stringers of constant thickness.

The applied loads are 410482 lb axial compression, 9.152E7 in-lb bending moment, 98372 lb shear force, a slight crush pressure of -.087 psi, a safety factor of 1.4 with a skin buckling safety factor of 1.0 (as long as 1.4 is maintained for all other failure modes).

Failure modes are: skin buckling; column/shell buckling; stringer crippling; and material stress. The design variables are: total stringer height, h ; length of the top flange of the hat, l_2 ; stringer thickness, tst ; number of stringers, Nst ; and skin thickness, t .

GA Parameters

The GA stopping condition is 25 generations without an improvement. A constant probability of mutation of 0.1% was used for all cases, regardless of population size.

A penalty function of $10 + G[i]*1000$ is added to the GA fitness (cylinder weight) when constraints are violated, where G is the array of constraints which are negative when satisfied. Adding 10 to the penalty insures there is a large enough difference between the optimum design and a design with a barely violated constraint (a slight positive number), to drive the GA to a valid optimum. The penalty function must be large enough to insure a valid design when the population converges, but small enough that good features of early designs that have some constraint violations are not lost.

Three lengths of bit string are used, designated as high, medium and low resolution. The number of bits, the resolution, and the range for each variable are shown in Table I.

Design Variable	RESOLUTION						Design Variable Range
	Low		Medium		High		
	bits	inches	bits	inches	bits	inches	
h	4	.26667	6	.06349	10	.00391	1.0" - 5.0"
l2	4	.26667	6	.06349	10	.00391	1.0" - 5.0"
tst	3	.03214	5	.00726	8	.00088	.025"-.250"
Nst	8	1*	8	1*	8	1*	100 - 355
t	3	.03214	5	.00726	8	.00088	.025"-.250"

* Resolution provides an integer number of stringers

Table I: Design Variable Resolution



Baseline Solution

Since the focus problem is a real engineering design problem, the true global minimum is not known. To develop a working number for the global minimum, two excessively large, high resolution True Hybrid cases were run. Both the 047 and 122 gradient optimizers were used with a GA population of 10000 individuals.

The 047 case gave the better result after 43 generations over 4 days on one processor of a 4 processor Sun Sparc 10 514 MP workstation. The final weight was 502.6 lb, which is assumed to be the global minimum. The optimum design variable values are: $h=2.1728$; $l_2=1$; $tst=.025$; $Nst=170$; and $t=.0612$.

Gradient With Random Starts

Without the GA, one method of trying to find the global optimum with a gradient optimizer is to select a series of random starting points. Figure 1 shows a histogram of the results from 10000 random starts for both gradient methods. The X axis indicates how far above the minimum weight the solutions are. That is, 10 indicates a weight of $502.6 + 50.26$ lb. The left vertical axis is the number of solutions in each 1% (5.026 lb) increment. The right vertical axis shows the cumulative distribution. Both the 047 and 122 methods give the same general results, with the 122 option providing slightly better results.

This indicates that for this problem, several random starts gives only reasonable confidence of being within 20% of the global optimum. Vast amounts of random starts are needed to assure being close to the global minimum. From this data it can be concluded that for noisy functions, a gradient solver is ineffective at finding the global minimum.

Genetic Algorithm

A series of GA's were run, with the results in Figure 2. The final solution and function calls required are shown as a function of population size. "Lo", "Med", and "Hi" indicate the low, medium and high resolution bitstrings.

The required function calls are nearly linear. Each run took about the same number of generations, so the function calls increase directly with the population size. The required calls also increase with bitstring resolution, since more evaluations are required to converge over the larger design space.

Since these results are from individual runs, not averages, it is important to look at the behavior of the curve rather than values of individual cases. The high resolution GA gives consistent results above a population of 600. The low and medium GA's give consistent results above 1000 members.

Since the GA has discrete design points it is unlikely to be able to solve for

the global minimum of a continuous function. The low resolution GA can only come within 32% of the minimum for the discrete set of values closest to each optimum variable. It can find other design solutions that are closer. The medium and high resolution GA's can only come within 0.9% and 0.06% of the global solution. When run, the high and medium resolution cases provide a result within 8% of the global, and the small GA comes within 14%.

These results show a marked improvement in a reliable solution by the GA over the gradient solver, but at the cost of more function calls. Also, the final solution is still not particularly close to the global optimum.

Inheritance And The Pure GA

The high resolution GA was run for varying populations and chance of inheritance. When using inheritance, there is a user provided percent chance that an individual will not be evaluated, but will inherit its fitness from its parents. The new fitness is the average of the two parents, weighted by the number of bits received from each parent.

The number of function calls required decreases as chance of inheritance increases, as expected. Up to about 30% chance of inheritance, a larger population is required to achieve the same final solution. For larger chances of inheritance, the solution is noticeably degraded regardless of population size.

As inheritance increases, the required function calls decrease, but so does the quality of the solution.

Hybrid 1: GA Followed By A Single Gradient Optimization

The first hybrid option uses the GA solution as a starting point for the gradient optimizer. For these cases the 122 gradient optimizer was used since it gives generally better results than the 047 option, and a single gradient run will not significantly alter the total number of function calls already required by the GA. Figure 3 shows the results of the gradient run.

While the gradient optimizer does consistently improve the results of the GA, the final solutions are still not very close to the global optimum. The low resolution does see an improvement of approximately 20 lb, but is still only within 10% of the global. There are only small improvements for the medium and high resolution GA's.

Hybrid 2: The True Hybrid

The second hybrid option performs a gradient optimization for each GA population member in each generation. Solutions and number of function calls are shown in Figure 4. The function calls required are again generally linear with the higher resolution solutions needing a few more function calls.

Since the solution for each individual is so much better, especially in the



early generations, much smaller populations are needed. Populations as small as 20 give results within 0.25% of the global solution for the medium and small resolution GA's. Interestingly, the high resolution GA shows much more inconsistent results than the others for small populations. Once the population size reaches a minimum of 60, the worst solution is within 0.5% of the global. Above 80 members, the worst solution is within 0.2%.

This hybrid provides excellent results with small populations and no significant increase in function calls from the pure GA.

The True Hybrid is so much more effective since the best starting point is not particularly close to the optimum solution. Even a very good initial guess may not lead to the optimum using the gradient solver. From the optimum start point the stringer height increases 40%, the flange width is halved, and the other variables and the weight are cut to a third. The starting point has no active constraints, but both skin buckling and shell buckling become design drivers.

Inheritance And The True Hybrid

The effect of inheritance on selected True Hybrid cases is shown in Figures 5 and 6. The X-axis indicates the percent chance of inheritance. The legends indicate the population size (e.g. P20 has a population of 20), and the GA resolution (Lo, Med and Hi).

Note that the weight axis of Figure 5 is expanded only 1% above the global solution. The larger population solutions, P70 and P500, are relatively unaffected by inheritance up to 70%. The smaller populations begin showing erratic results above 30-40% inheritance. There is the expected downward trend of calls required as inheritance increases.

Inheritance has little effect on the large population of 500. Because of this, the interaction of population size and inheritance was studied and the results are shown in Figures 7 and 8. Weight is unaffected for populations as low as 200 and inheritance up to 70%. A population of 100 is more erratic, but is still within 0.5% of the global solution at 70% inheritance.

The function calls decrease with inheritance for all population sizes, but the population has a greater effect on the calls required. For example, a population of 300 with 60% inheritance takes more calls than a population of 200 with 20% inheritance.

The value of inheritance may be in the initial optimization of a given design. If the required population is unknown, good results can be achieved by running a large population with high inheritance. As the user becomes more familiar with the problem and can better determine the minimum population, inheritance can be reduced to provide better answers with fewer calls.

Conclusions

The weakness of gradient solvers for finding local minima has long been known. This study confirms that for a noisy, real world design, gradient



solvers are ineffective in finding a near optimal solution. Additionally, the gradient solver must treat the number of stringers as a continuous variable, instead of as an integer which it is in reality.

The GA is limited by the resolution of its bitstring as to how close to the true global optimum it can get. It does allow the number of stringers to be treated as an integer (by the correct selection of variable range and number of bits to represent it). For this problem the GA provided slightly more reliable results, but at the cost of many function calls.

Using the final GA solution as a gradient starting point provides only a marginal improvement over the GA results. However, for a smooth function in which the peaks and valleys are well within the resolution of the GA this should be very effective at finding the global minimum.

The True Hybrid is by far the most effective at finding the best results for the fewest function calls. Very small populations, 10 to 40, provide weight results much better than either the gradient solver or GA. Medium populations of 40 to 100 provide excellent results. Populations in excess of 100 are extremely reliable, even for the small resolution GA. The two types of solvers provide a synergy in which each can make up for the flaws of the other. A small population, coarse GA can be combined with a moderately effective gradient solver to reliably find designs near the global optimum, with a relatively small number of function calls. Very useful in a practical sense is the True Hybrid's ability to work with mixed mode optimization. The GA optimizes integers, such as the number of stringers, while the gradient solver finds precise solutions for the continuous variables.

Inheritance reduces the required number of function calls for a given population size, but using the correct population size without inheritance gives better results with fewer function calls. The greatest utility of inheritance may be for a situation in which only one optimization run can be made, using a large population with a high inheritance.

References

1. Finckenor, J., Rogers, P., Otte, N. CORSS: Cylinder Optimization of Rings, Skin and Stringers, NASA TP 3457, 1994.
2. Finckenor, J., Bevill, M. CORSSSTOL: Cylinder Optimization of Rings, Skin and Stringers With Tolerance Sensitivity, NASA TP 3551, 1995.
3. Vanderplaats, G.N. ADS - A Fortran Program for Automated Design Synthesis, NASA CR 172460, 1984
4. Vanderplaats, G.N. *Numerical Optimization Techniques for Engineering Design: With Applications*, McGraw-Hill Book Company, New York, 1984.



5. Finckenor, J. JGA, unpublished, NASA/MSFC/ED52, 1995
6. Goldberg, D. *Genetic Algorithms in Search, Optimization & Machine Learning*; Addison-Wesley Publishing Company, Reading, Mass.; 1989
7. Foster, N.F. & Dulikravich, G.S. Three-Dimensional Aerodynamic Shape Optimization Using Genetic and Gradient Search Algorithms, *Journal of Spacecraft and Rockets*, Jan-Feb 1997, Vol. 34 No.1, 36-42.

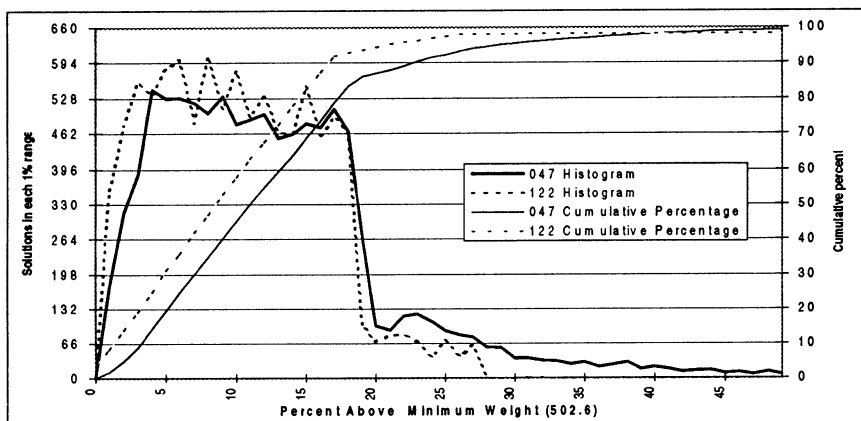


Figure 1. Gradient Optimizer - Solution Distribution from 10000 Random Starts

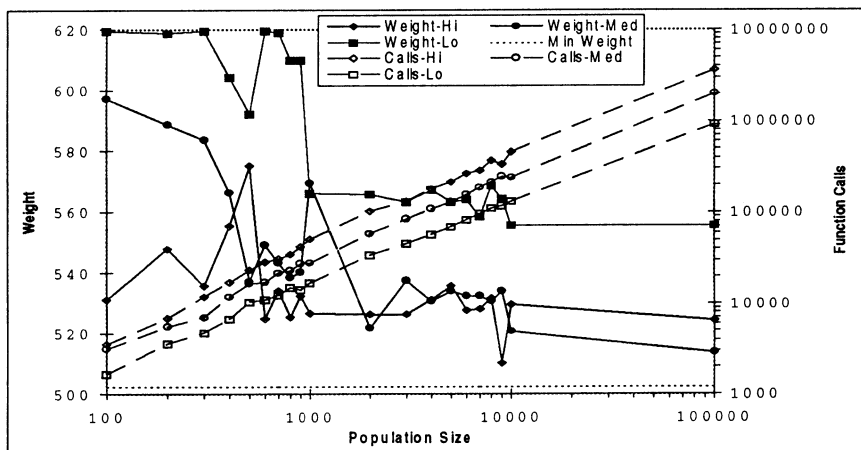
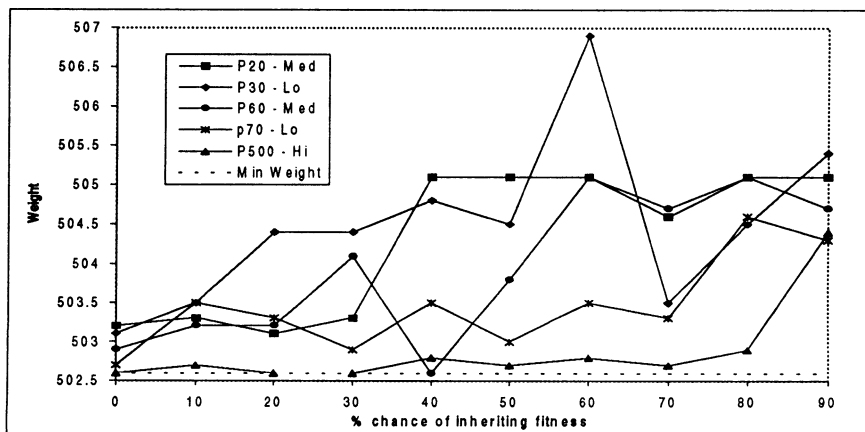
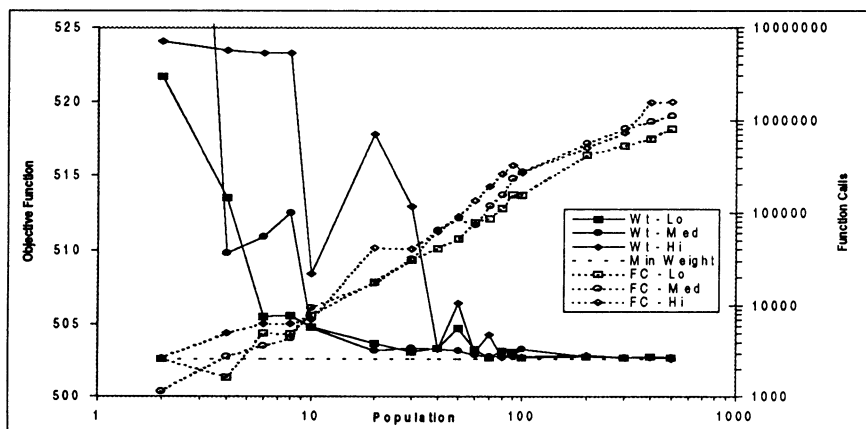
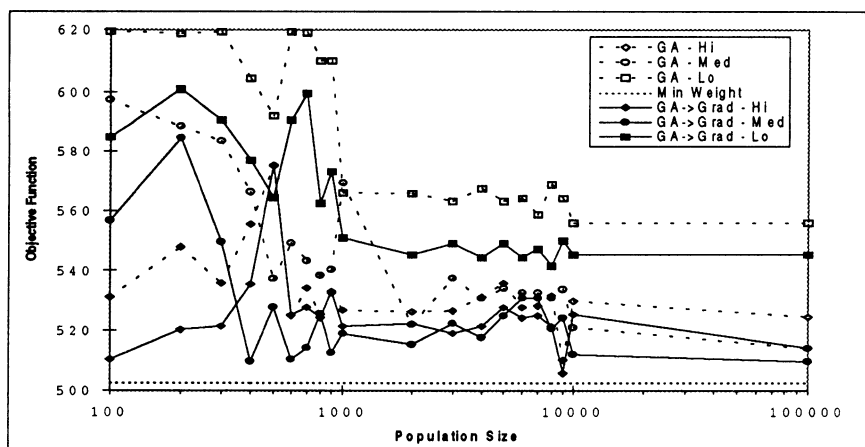


Figure 2. Genetic Algorithm - Objective and Calls vs Population





266 Computer Aided Optimum Design of Structures V

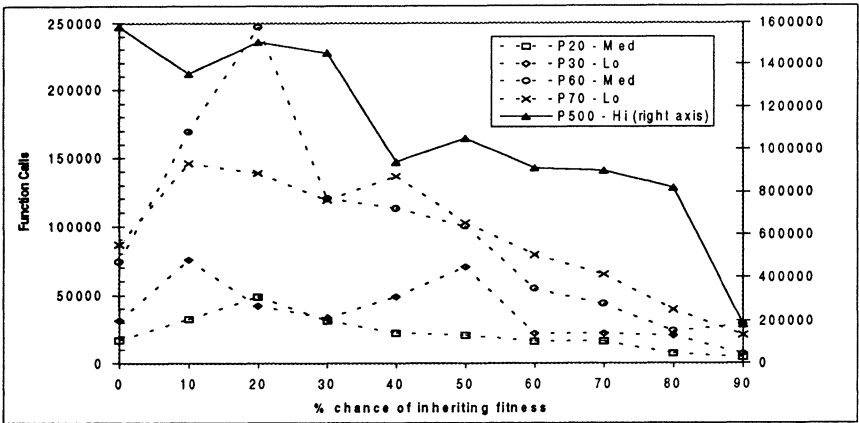


Figure 6. Selected True Hybrids - Effect of Inheritance on Function Calls

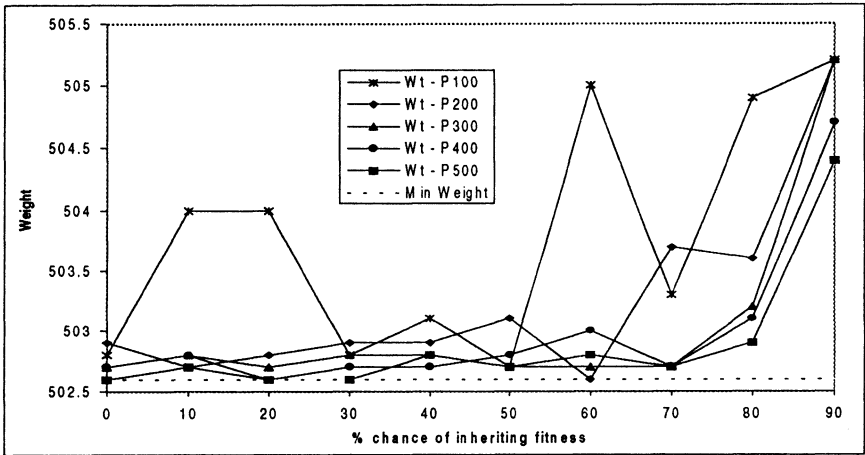


Figure 7. True Hybrid - Effect of Population and Inheritance on Weight

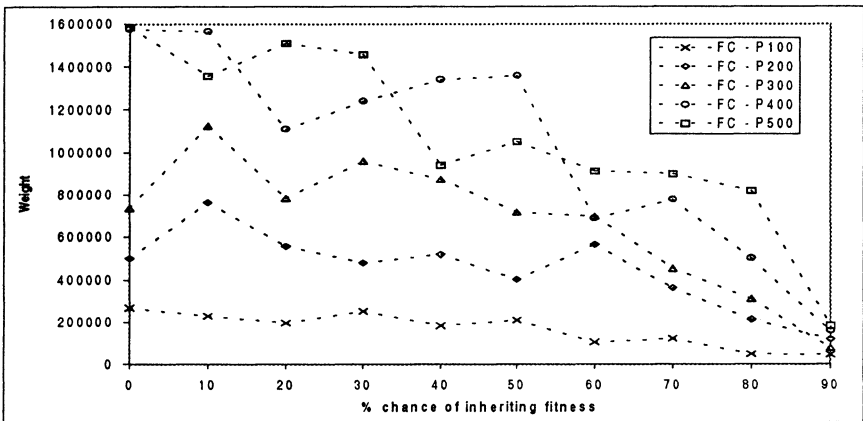


Figure 8. True Hybrid - Effect of Population and Inheritance on Calls