# An Automated Approach to Post-Flight Simulation Analysis and Validation Using Stochastic Optimization Methods

**Murray B. Anderson, Ph.D.**

Director, Advanced Technologies and Studies Department
Sverdrup Technology Inc./TEAS Group
murray.anderson@eglin.af.mil

**Abstract**

With the advent of streamlined acquisition processes, such as simulation-based acquisition, the need for superior weapon simulation capability is critical to developing launch and jettison envelops. Given the limited test assets allocated to envelop development, certification agencies are today being asked by new weapons programs to develop the same large envelops produced 10 years ago with only 10 percent of the assets. An additional complication is that weapon systems are becoming increasingly complex: non-axisymmetric, unstable or nearly unstable at launch, employed from weapons bays, employed at very high angles of attack. Given these complexities, it is becoming increasing common for weapons to exhibit non-linear and/or highly coupled behavior during launch or jettison events. For the engineer, having a system that behaves in a non-linear fashion greatly complicates post-flight analysis. It is critical, therefore, to have tools available that can optimize a large number of independent variables to produce the desired output. This paper introduces one such tool and shows its application to sample flight test data.

## Introduction

Verification of launch/jettison performance predictions requires implementation of accurate mathematical models of the weapon aerodynamics, the aircraft interference flowfield contributions, the ejector performance, the flight control system, and knowledge of the actual flight conditions at launch. A number of parameter estimation methods, such as maximum likelihood, linear regression, and Kalman filter, among others, have been applied to flight data[1,2,3] to examine the fidelity of these models. Iterative gradient-based procedures are usually invoked, beginning with initial estimates of the parameters, to modify the parameter estimates until convergence is achieved. With noise included in the measurements, uniqueness of the parameter set is not guaranteed for a minimum error solution, since, in general, different combinations of parameters may produce solutions having equivalent, least-squares differences relative to the data. The complications introduced by including non-linear and highly-coupled aerodynamic behavior into the optimization process make gradient-based methods even less attractive because these complications almost guarantee the presence of local optima. Stochastic optimization techniques are free from the limits imposed by gradient-based methods. Since transition rules are not deterministic but stochastic, techniques that use probability theory to decide which part of the solution space to sample next have the ability to broadly sample the optimization space of interest and develop a non-linear multi-dimensional mapping of the optimization space. Probably the most popular stochastic optimization scheme in use today is genetic algorithms. Although popular for years for control system studies (Norris and Crossley[4], McGookin[5], Homaifar and McCormick[6]), genetic algorithms have really not penetrated the flight test and simulation validation/verification community until recently[7,8]. Genetic algorithms (GAs) will not be described in this paper, but it is important to know the type of algorithm used. For this study an elitist Pareto GA is used to operate on two goals (defined below). The population size was 100, the crossover probability was 90% and the mutation rate was 0.5%. Creep mutation was also used at a rate of 5%, meaning that 5% of the variables could creep at any time.

## Definition of Optimization Goals

In order to get a simulation to match flight test data, it is important that the goals be properly defined. During the separation event, whether jettison or active launch, there are two main objectives when attempting to get a simulation to match flight test data: minimize position errors, minimize attitude errors. Performance estimates for the attitude and position goals are based on determining the root-mean-square (RMS) difference between the flight test telemetry data and the current simulation attempt at reproducing the flight test data.

## Interference Flow Field and Freestream Aerodynamic Data

Rather than use actual flight test data for this study, a conventional six-degree-of-freedom (6DOF) generated "simulated" flight test data using defined interference flow field and freestream aerodynamic models. There are two major advantages to this approach. First, since this is a process study to explore the performance of an optimization algorithm, knowing the "right" answers if of critical importance to evaluating the algorithm's performance during the solution process. Second, there are no public-release issues involved when using "simulated" flight test data. The "simulated" flight test data used for this demonstration were, however, based on real flight test experience for a real weapon system. From this point forward, these data will be referred to as flight test data.

The aircraft interference data was modeled as force and moment increments which decayed exponentially to zero after the weapon traversed 30 feet from the launch point. The freestream aerodynamic data were statically unstable in pitch and yaw, and neutrally stable in roll.

## Ejector Model

The ejector was modeled as a force versus displacement model that had the capability to distribute the load as desired between the forward and aft foot of the ejector. This type of model is very representative of actual hardware.

## Optimization Variables

There were 17 input variables to the simulation that the genetic algorithm was forced to include in the optimization process. The genetic algorithm also requires specification of the maximum and minimum bounds that each value can take, and the desired resolution of parameter. For this study, the variables and bounds found in Table 1 were used in the optimization process. Given these parameter bounds and resolution requirements, there are $2^{93}$ possible answers to this optimization problem.

Table 1. Ejector, Inertial, and Freestream Aerodynamic Factors

| Variable Name | Definition | Minimum | Maximum | Resolution | Bits |
|---|---|---|---|---|---|
| fwdfac | Multiplication factor governing amount of force applied to forward ejector foot | 0.2 | 1.0 | 0.025 | 5 |
| aftfac | Multiplication factor governing amount of force applied to aft ejector foot | 0.2 | 1.0 | 0.025 | 5 |
| xinerfac | roll inertia multiplier | 0.9 | 1.1 | 0.02 | 4 |
| yinerfac | pitch inertia multiplier | 0.9 | 1.1 | 0.02 | 4 |
| zinerfac | yaw inertia multiplier | 0.9 | 1.1 | 0.02 | 4 |
| caoff | freestream axial force coefficient offset (f.c.o) | -0.1 | 0.1 | 0.01 | 5 |
| cyoff | freestream side f.c.o | -0.2 | 0.2 | 0.01 | 6 |
| czoff | freestream vertical f.c.o. | -0.2 | 0.2 | 0.01 | 6 |
| cmoff | freestream pitching moment coefficient offset (m.c.o.) | -0.5 | 0.5 | 0.02 | 6 |
| cnoff | freestream yawing m.c.o. | -0.5 | 0.5 | 0.02 | 6 |
| cloff | freestream rolling m.c.o. | -0.5 | 0.5 | 0.02 | 6 |
| caintfac | interference axial force coefficient multiplier (f.c.m.) | 0.25 | 2.0 | 0.05 | 6 |
| cyintfac | interference side f.c.m. | 0.25 | 2.0 | 0.05 | 6 |
| czintfac | interference vertical f.c.m. | 0.25 | 2.0 | 0.05 | 6 |
| cmintfac | interference pitching moment coefficient multiplier (m.c.m.) | 0.25 | 2.0 | 0.05 | 6 |
| cnintfac | interference yawing m.c.m. | 0.25 | 2.0 | 0.05 | 6 |
| clintfac | interference rolling m.c.m. | 0.25 | 2.0 | 0.05 | 6 |

## Results

To better gauge how well the new post-flight analysis approach works, it will be compared directly against a "hill climbing" gradient optimization approach. Gradient methods have been the standard way to perform optimization problems such as this one for many years. However, for highly non-linear optimization problems, the advantages of this new approach will become readily apparent.

Figure 1 shows the Euler pitch angle history for the hill climber and the genetic algorithm compared to flight test data. The hill climber obviously got stuck in a local optima and did not compare nearly as well with the flight test data than the genetic algorithm. Forgetting the genetic algorithm comparison for a moment, it could be said that the hill climber captured the right "trend". Engineers often talk about trend comparisons with the actual comparisons are not as good as they would like.

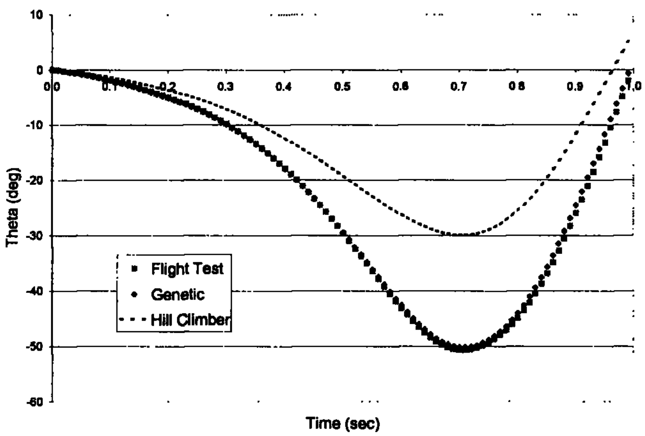


Figure 1. Euler Pitch Angle Comparison

The yaw angle history plot, Figure 2, shows much better performance by the hill climber, however, it is still not as good as the genetic algorithm. Both methods provide very reasonable yaw behavior for the vehicle.

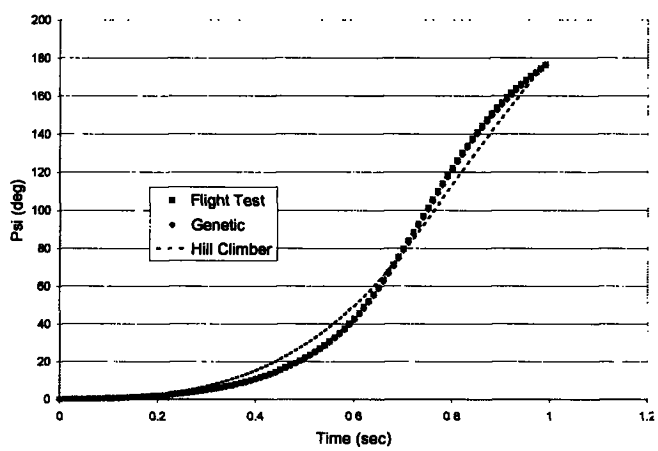


Figure 2. Euler Yaw Angle Comparison

Figure 3 shows the roll angle comparison. The hill climber missed the initial roll to the right and lacked overall trend performance. The genetic algorithm mimicked the flight test data very well. Even the fairly complex roll motion (right-left-right) is captured nearly exactly.

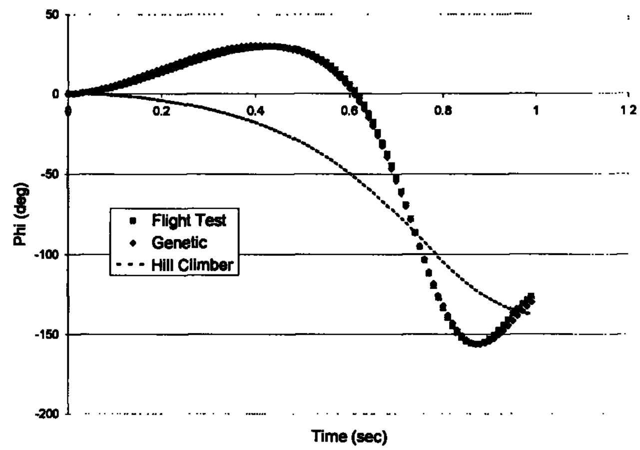


Figure 3. Euler Roll Angle Comparison

The trajectory of the vehicle was captured very well by the genetic algorithm. The downrange comparison (Figure 4) shows that either method works well, but since downrange is typically the least important trajectory direction for safety of flight it is best to look at lateral and vertical vehicle movement relative to the launch point.

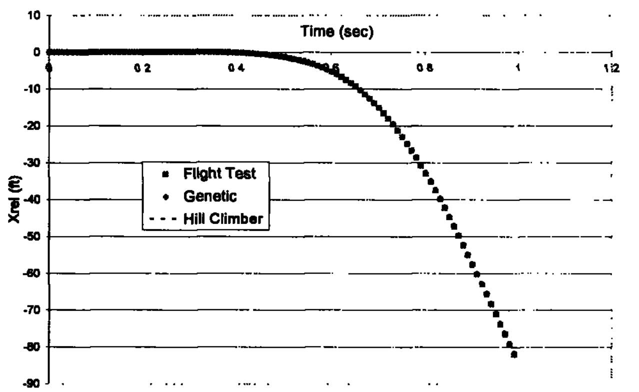


Figure 4. Downrange Comparison

In the altitude and crossrange histories (Figure 5 and Figure 6) the strength of the genetic algorithm becomes readily apparent. Both lateral and vertical vehicle motion is captured nearly perfectly by the genetic algorithm. The hill climber actually performed fairly well, capturing both trends and magnitudes reasonably, but when compared to the genetic algorithm the hill climber does not look very good.

The gains evidenced by the genetic algorithm are not, however, without cost. The hill climber found its answer

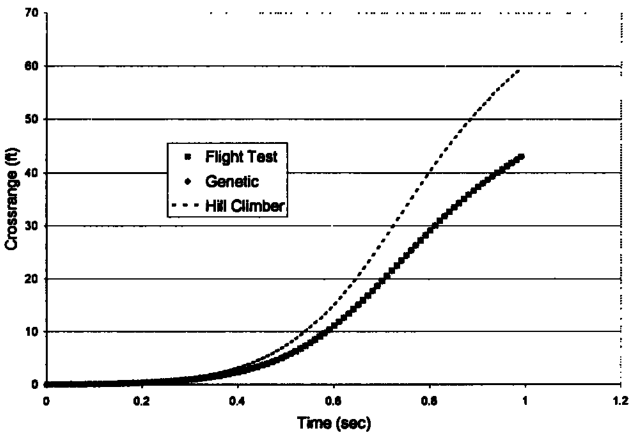


Figure 5. Crossrange Comparison

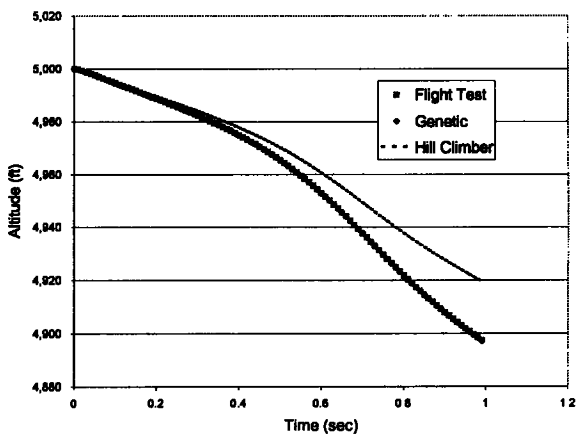


Figure 6. Altitude Comparison

in roughly 500 simulation runs (roughly 2 hours of CPU time). The genetic algorithm answer was generated in 50,000 simulation runs (500 generations with 100 members per generation).

A natural question to ask at this point is: what kind of parameter values did the two optimization approaches find?

Figure 7 shows the values found for each parameter compared to the "baseline" or truth model. One of the most noticeable differences is also one of the most important for a separation event: the ejector forces. The genetic algorithm found nearly perfect values for the ejector force factors. The hill climber had difficulty finding reasonable values. Both methods missed the freestream axial force coefficient offset and the hill climber missed the "normal" force coefficient badly. The performance of each method for the freestream moment coefficient offsets was mixed. Both methods missed the "true" negative pitching moment offset. The genetic algorithm was closest on the yawing moment offset, and the hill climber was nearly perfect on the rolling moment offset. It is interesting to note that although the hill climber worked so well in finding the rolling moment coefficient offset, the roll angle performance of the hill climber was not good (recall Figure 3). For strongly coupled yaw-roll behavior, it is necessary to find reasonable parameter values in more than one variable to capture complex motion.

The interference flow field contributions to the vehicle motion show generally better performance by the genetic algorithm when compared to the hill climber. Most noticeable is the "normal" force contribution difference, where the hill climber performed poorly. The genetic algorithm worked much better than the hill climber in pitching and yawing moment coefficients. The hill climber was closer to the "truth" model than the genetic algorithm for the rolling moment contribution. The hill climber worked better than the genetic algorithm in both the freestream and interference rolling moment contributions, but still the genetic algorithm worked better in overall roll motion. It is possible that the genetic algorithm's underestimation of the freestream rolling moment coefficient contribution could have been compensated by the overprediction of the interference flow field's contribution.

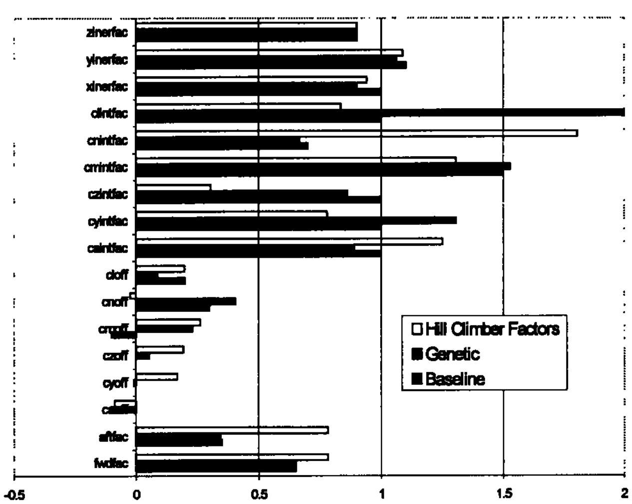


Figure 7. Factor Comparison

Finally, the inertia factors found by both methods were close to the "truth" model.

It is interesting that there is such a dramatic difference in vehicle motion given the genetic factors versus the hill climber factors. Yet for complex aerodynamics, and given a vehicle that is unstable at launch, differences early in a trajectory can dramatically influence motion a second after launch. It is also interesting that some of the genetic algorithm factors can be in error significantly (like interference flow field rolling moment coefficient factor) and yet the overall vehicle motion is captured nearly perfectly. This would indicate that many parameter combinations are capable of producing reasonable trajectories. For a complex non-linear aerodynamic optimization problem this should be the expected result.

## Conclusions

The genetic algorithm is a dramatic improvement over the popular hill climbing approach, but the parameter values it obtains can still be in error. There is a possibility that parameter combinations can compensate somewhat for each other, leading to finding a "good" local optima rather than the global optima. Given the way a genetic algorithm operates, it is inevitable that more generations would have eventually produced the "global" optima. But at what cost? The solution found by the genetic algorithm in 500 generations worked well for each of the six degrees of freedom, with parameter values that are not unreasonable for a flight test program. Were this a real flight test comparison, such good agreement between the simulation and the flight test data would be sufficient to proceed with future missions with increased confidence. The hill climber results, although generally capturing the correct trends, would not inspire nearly as much confidence in the simulation.

## References


1. Iliff, K. W., "Parameter Estimation for Flight Vehicles," AIAA Journal of Guidance, Control, and Dynamics, Vol. 12, No. 5, Sept-Oct 1989, pp. 609-622.
2. Chapman, G.T. and D.B. Kirk, "A Method for Extracting Aerodynamic Coefficients from Free-Flight Data," AIAA Journal, Vol. 8, No. 4, April 1970.
3. Welsh, C.J. and W.R. Lawrence, "Motion Analysis Procedure for Asymmetric Vehicles," AEDC-TR 75-153, May 1976, AIAA 3rd Atmospheric Flight Mechanics Conference, June 1976.
4. Norris, S.R., and Crossley, W.A., "Pareto-Optimal Controller Gains Generated by a Genetic Algorithm," AIAA-98-1010, AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, 1998.
5. McGookin, E.W., Murray-Smith, D.J., Li, Y., and Fossen, T.I., "Parameter Optimisation of a Non-linear Tanker Control System Using Genetic Algorithms," Proceedings of the 2nd IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems, Glasgow, United Kingdom, 1997.
6. Homaifar, A., McCormack, E.D., "Full Design of Fuzzy Controllers Using Genetic Algorithms," NASA-N93-19452 06-80, The NASA Center for Aerospace Research at North Carolina A&T State University, 1992.
7. Anderson, M.B., McCurdy, R.E., "Weapon Drag Coefficient Determination Using Genetic Algorithm", AIAA paper 94-3468, presented at the 1994 Atmospheric Flight Mechanics Conference, Scottsdale, AZ., August 1994.
8. Anderson, M.B., and Lawrence, W.R., "Launch Conditions and Aerodynamic Data Extraction By An Elitist Pareto Genetic Algorithm", AIAA Paper 96-3361, presented at the AIAA Atmospheric Flight Mechanics Conference, San Diego, CA, July 1996.