Applications of Genetic Methods to NASA Design and Operations Problems

Philip D. Laird
NASA Ames Research Center
Moffett Field, CA

APPLICATIONS OF GENETIC METHODS TO NASA DESIGN AND OPERATIONS PROBLEMS

Philip D. Laird
NASA Ames Research Center
Mail Stop 269-2
Moffett Field, CA 94035-1000

We review four recent NASA-funded applications in which evolutionary/genetic methods are important. In the process we survey

- the kinds of problems being solved today with these methods;
- techniques and tools used;
- problems encountered; and
- areas where research is needed.

The presentation slides are annotated briefly at the top of each page.

PROBLEM I - ROTORCRAFT DESIGN

This study applies a simple genetic algorithm to the choice of both the configuration and the sizing parameters in the design of rotorcraft (helicopters and VTOL vehicles).

Source:

W. Crossley, J. Regulski, V. Wells, D. Laananen, "Incorporating Genetic Algorithms and Sizing Codes for Conceptual Design of Rotorcraft", American Helicopter Society Vertical Lift Aircraft Design Conference, Jan., 1995.

(Work performed under a grant from NASA Ames Research Center.)

General Problem:

Combination of both configuration design and sizing parameters in the design of rotorcraft (helicopters and tilt-wing/rotor VTOL craft).

ROTORCRAFT DESIGN (Cont.)

In standard aircraft design methodology, the configuration is determined heuristically based on experience. Then the parameters (sizes, dimensions, etc.) are calculated using optimization programs (sizing codes). In this study the configuration itself is part of the design problem, and the sizing codes are used as part of the genetic fitness function.

The Usual Approach to A/C Design:

- **Configure selection based on "experience"**
- **Sizing codes used to optimize parameter values**
- **Carpet plot to visualize influence of three key parameters on the solution space, to adjust for non-mathematical constraints.**

Tools:

- **PC platform**
- **HESCOMP, VASCOMP2 "sizing" codes.**

Reasons for Using GAs:

- **Ability to handle discrete and continuous parameters and search a highly discontinuous landscape**
- **Ability to search more configuration selections than an engineer's bias would normally allow.**

ROTORCRAFT DESIGN (Cont.)

Two design problems were attacked: a helicopter design and a VTOL design. Both entailed discrete and continuous parameters. Fitness was the vehicle gross weight as computed by the sizing code. Each problem was run five times to assess premature convergence and the number of optimal designs.

Approach:

- **Bit string coding for discrete variables (e.g., single main rotor vs. tandem rotors), integer variables (e.g., num. blades, engines), and continuous variables (disk loading, rotor solidity factors, ...).**
- **Fixed mission defined for each craft.**
- **Fitness:**
  Optimum gross weight returned by HESCOMP or VASCOMP2.
- **G-bit improvement adjustment, elitism for best-of-generation individual.**
- **Five runs of each problem.**
- **Compare results with human designs.**

Shown below is the encoding for the helicopter design, comprising 15 bits. In most problems such a compact encoding would mean that one could exhaustively evaluate all design possibilities, but the cost of running the sizing programs is prohibitive here.

Encoding For Helicopter Design:

Variable (units)	Min.	Max.	Resolution	String Length (bits)
Tip Speed (ft/sec)	700	735	5.0	3
Disk Loading (lb/ft^2)	11.5	15.0	0.5	3
Wing Loading (lb/ft^2)	250	320	10	3
Blade Loading (C_t/σ)	.085	.120	.005	3
Num. Engines	2	3	1	1
Num. Blades	3	6	1	2

ROTORCRAFT DESIGN (Cont.)

For the helicopter design problem, convergence to optimal was rapid, showing no indication of premature convergence. The designs were not directly realizable because not all constraints were included in the fitness evaluation. For the VTOL design, the GA converged prematurely on the second generation; but the runs produced four different designs of weight comparable to the human designs. Interestingly, in both cases the design configurations differed from those chosen by the human designers of the aircraft.

Results:

(Helicopter)

- **Rapid convergence (about 9 generations), no premature convergence.**
- **All runs gave similar designs. About 4 hours/run.**
- **Designs tended to be unrealizable and expensive because of missing constraints.**
- **Recommended tandem-rotor design not used by original (human) design team.**

(VTOL)

- **Converged prematurely on second generation, but runs yielded 4 very different designs. About 10 hours/run.**
- **Predicted gross weight below NASA design, but within 15%.**
- **All designs preferred tilt wing to tilt rotor (NASA chose tilt rotor), but differed significantly in the parameter values.**

ROTORCRAFT DESIGN (Cont.)

Note that the difficulty of representing all the constraints of a real design problem is always present, but is especially apparent in the use of genetic methods since this is all the algorithms have to go on. Nevertheless, this work shows that even simple GAs are effective for both configuration and sizing, and hence can be used to test fixed biases toward certain configurations.

Lessons and Issues:

- **Difficult to represent true hard and soft constraints in one fitness function.**
- **Even simple GAs are a good way to explore configuration and sizing problems, and to challenge fixed design biases.**

Related Work:

Bramlette and Bouchard, "GAs in Parametric Design of Aircraft,"*Handbook of Genetic Algorithms*, Davis, 1991

Dike and Smith, R. E., *X-31 Combat Utility Study of High Alpha Air Combat Tactics*, Contract Report, NASA Dryden Flight Research Center, 1994

PROBLEM II - GA APPLIED TO SPACE SHUTTLE THERMAL DATA MODELS

A simple GA was evaluated as an alternative to an existing thermal modeling program that was felt to be too slow.

Source:

J. Snyder, "Genetic Algorithm Applied to Shuttle Thermal Data Models," unpublished manuscript, Oct. 1994.

(Work performed by Client/Server Systems Branch (PT4), Johnson Space Flight Center, by Joe Snyder and Lui Wang).

General Problem:

Generate thermal profiles for shuttle sensors to serve as predictions for the next flight. Predictions are also compared to real-time data during flight to monitor potential problems.

Parameters in a thermal model are chosen to optimize agreement with historical data. Existing iterative methods were very slow and computationally expensive. GAs were evaluated as an alternative.

THERMAL MODELS (Cont.)

The task was to adjust the model parameters to agree with measured data from past flights. A total of eight parameters were encoded initially with sixteen bits. After initial checkouts with a reduced number of parameters and a subset of data, the full problem was attacked.

Approach:

- **Check coding/representation by using simple cases (e.g., one flight's data for one component).**
- **Evaluate the GA approach on a complex case: four components (two parameters each), four flights of data.**
- **Experiment with population size (125, 250 and 500), parameter ranges and resolution (16 bits per parameter), mutation and crossover rates, and selection methods (tournament, roulette-wheel).**
- **Following each run, plot the temperature error vs. generation for each parameter.**

Tools:

- **SPLICER package for GAs (a multi-platform SGA written at JSC and available through COSMIC).**

THERMAL MODELS (Cont.)

Initial results were discouraging: convergence to solutions was not occurring in reasonable times. After considerable experimentation with the control parameters of the GA, it turned out that an approach of successive refinement in the parameter values, significantly reducing the size of the parameter search space, led to successful convergence.

Results:

- **GAs did not improve the convergence time over the iterative method.**
- **Adjustments to genetic "knobs" had only minor influence on the convergence to low-error solutions.**
- **Critical was limiting the allowable range of values for the parameters (e.g., from 0-3 to 0-1). This enabled convergence to optimum solutions in two or three hours on a Sun workstation.**

THERMAL MODELS (Cont.)

Careful design of the search space is often very important for success with GAs, and this takes time. Also, a general technique like GAs, capable of solving a broad array of problems, is not expected to outperform a problem-specific solution such as the one here that had been developed over a long time.

Lessons and Issues:

- **Choosing parameters and representations for numerical parameters can be very difficult, even for knowledgeable and skilled technicians. Compare: 128 bits vs 15 bits for the rotorcraft experiments.**
- **The GA was being asked to improve on an existing approach designed for the problem. In such cases a generic method will not often yield improvement.**
- **Authors suggest series of runs in which the value constraints and resolutions of the parameters are modified.**

Related Work:

S. Colombano, "Goal-Directed Model Inversion," ***NIPS Workshop on Neural Networks and the Solution of Inverse Problems*****, Nov. 1994**

PROBLEM III - OPTIMIZING COCKPIT DISPLAYS

This study shows an innovative application of evolutionary algorithms to the design of multi-function displays in a "glass" cockpit.

Source:

Tica Technologies, Inc., *Optimizing Cockpit Display Configurations with a Genetic Algorithm System*, Contract Report, Dec. 1994.
(Work performed by L. Davis, B. Constantine, J. Kelly, S. Shieber and others under contract from NASA Ames Research Center.)

General Problem:

To organize in an optimal way the arrangement of information onto pages of an MFD (Multi-Function Display) in an advanced aviation cockpit.

OPTIMIZING COCKPIT DISPLAYS (Cont.)

MFDs are an alternative to cockpits where all dials, gauges, etc., are visible at all times. However, it creates difficult human factors problems in making sure that critical information is available when needed. Here genetic methods were studied as one alternative for designing pages of an MFD.

Multi-Function Displays:

- **Alternative to traditional cockpit displays consisting of fixed gauges, dials and other displays. No room in "glass cockpit."**
- **Not all information immediately visible. Human factors problem to determine the "best" arrangement of data on pages, and the access to those pages.**

Specific Problem:

- **Using a challenging mission for a Comanche attack helicopter, determine an optimal allocation of information to MFD pages as measured by number of page changes weighted by the importance of the information.**

Tools: MIDAS platform, proprietary GA software.

Reason for Using GAs:

To evaluate GAs as one alternative for solving the problem.

OPTIMIZING COCKPIT DISPLAYS (Cont.)

Using a fixed mission as a criterion, fitness of a given assignment of information to pages was measured by the number of button presses, weighted by a factor measuring the importance of the information. The trick in solving the problem with GAs was to reduce it to a problem of partitioning a weighted directed graph, and to refine an initial solution found with the genetic method into a locally optimal solution by applying the Kerhighan-Lin algorithm.

Approach:

- **Fitness = expected number of button presses on mission, weighted by criticality factor. (Random allocation requires about 35 button presses on a mission.)**
- **Transform the page layout problem to a weighted, directed-graph partitioning problem; total weight of edges crossing partitions measures (raw) fitness.**
- **Use GA (evolutionary algorithm) to construct an initial solution to the graph partitioning problem (NP hard). Then use Kernighan-Lin (local improvement) algorithm to improve the solution.**
- **Compare results to those of humans and of a program using the pure Kernighan-Lin algorithm.**

OPTIMIZING COCKPIT DISPLAYS (Cont.)

Investigators compared the human solution for the same problem with that produced by the K-L algorithm alone and that produced by the GA plus K-L.

Results:

- **GA result was best: 22% better than human solution, 10 to 18% better than the pure KL algorithm.**
- **Run-time performance and resource requirements were not reported.**
- **Crossover was not used because it yielded no improvement on the results.**

OPTIMIZING COCKPIT DISPLAYS (Cont.)

Success in this case rested on several combined factors: clever representations, expertise in combinatorial techniques, and the lack of a known effective algorithm for an NP-hard problem.

Lessons and Issues:

- **GA is especially useful when no known existing algorithm exists to solve a "hard" problem.**
- **Key to success here was the clever reduction to a combinatorial optimization problem, plus the expertise of the team in genetic and graph-theoretic methods.**
- **Other applications to automated layout and network wiring.**

Related Work:

D. Orvosh and L. Davis, "Shall We Repair? Genetic Algorithms, Combinatorial Optimization, and Feasibility Constraints," ***Proceedings of the Fifth International Conference on Genetic Algorithms*****, 1993.**

C. Kosak, J. Marks, and S. Shieber, "Automating the Layout of Network Diagrams with Specified Visual Organization," ***IEEE Transactions on Systems, Man and Cybernetics*****, 1993.**

PROBLEM IV - GENETIC CONFIGURATION DESIGN

Instead of attacking an individual problem, Roston develops a design methodology applicable to a broad family of design problems. Besides the problem presented below he has applied his genetic design methodology to a diverse array of other problems.

<u>Source</u>:

Gerald P. Roston, *A Genetic Methodology for Configuration Design*, Ph.D Thesis, Department of Mechanical Engineering and the Robotics Institute, Carnegie Mellon University, Dec. 1994.

(Work supported by NASA contracts with CMU.)

<u>General Problem</u>:

To develop a genetic design (GD) methodology applicable to a broad range of application areas.

Specific problem illustrated here: to design a frame-walker rover and its controller capable of negotiating a variety of terrains.

Other applications described: stepping-stone roller vehicles, planar linkage mechanisms, truss bridges.

GENETIC DESIGN (Cont.)

The main tool in this approach is genetic programming (Koza). A design space is defined by context-free grammar. GP is used to search for an optimal design, taking advantage of the structural flexibility of its rules. Because GP runs take so long, a preliminary evaluation of the run parameters is conducted using a "meta"-GA.

Tools/Technology:

- **Genetic programming (with strong typing)**
- **Formal grammar representation of design**
- **"Meta"-GA to help select eight of thirteen numerical parameters for GP algorithm.**

Reason for Using GP:

- **Ability to search a space of variable structures and simultaneously co-evolve a controller program**
- **More domain-independent than other GA methods.**

GENETIC DESIGN (Cont.)

The design of a frame-walker rover requires selecting the number of frames, the number of legs per frame, the size and spacing of those frames, and a control program for driving the vehicle.

<u>Frame Walker</u>:
A simple, statically-stable rover able to negotiate rugged terrain.

- **Two or more frames, each with two or more legs.**
- **Leg positions fixed relative to the frame but differing in width and separation.**
- **Moves by lifting frame 1, moving forward by some amount, lowering, and repeating this for frames 2, ...**
- **Unable to proceed if frame fails to find terrain support beneath at least two legs.**

vehicle* -> *frames* C *controller
frames* -> *frames frame* | *frame
frame* -> F *number legs
legs* -> *leg legs* | *leg
***leg* -> L *number number* [gap distance, pad width]**
controller* -> *sensor* | *number* | *func
func* -> *conditional-func* | *math-func
etc.

GENETIC DESIGN (Cont.)

Fitness of a given design is measured by a vector of four performance quantities. A distance measure in Pareto space serves as the scalar measure of fitness. In a series of increasingly challenging terrains, the qualities of the solutions were carefully evaluated and compared to expectations.

Approach:

- **Context-free grammar representing the configuration and the controller program.**
- **Multi-objective fitness function with a distance measure in Pareto space. Fitness features:**
  - **distance traveled before unable to proceed (larger is better)**
  - **number of frames (smaller is better)**
  - **average number of legs per frame (smaller is better)**
  - **average pad width (smaller is better)**
  - **number of steps required to traverse terrain (smaller is better).**
- **Crossover and mutation, population of 350, elitism, about three runs per experiment.**
- **Series of carefully graduated experiments in which expected results are compared with actual results over various terrain families (from simple periodic to fractal).**

GENETIC DESIGN (Cont.)

Among the interesting results: optimal controllers were found for simple problems where the optimal was known. On complex terrains the solutions were robust for random variations in the terrain specifics; and vehicles evolved against a fixed control program were outperformed by vehicles that co-evolved with their own controllers.

Results:

- **Run on a Sparc 10, probably using a Lisp implementation. Performance statistics not given, but executions measured in days.**
- **On simple terrains, found an optimal controller.**
- **On complex terrains, the optimal controller was not known, but solutions were robust for a range of terrain variations.**
- **A vehicle co-evolved with controller usually outperformed a vehicle evolved with a fixed controller program.**
- **Controller programs tended to be large and incomprehensible (common with GP).**

Lessons/Issues:
Configuration design with co-evolving controller is feasible and a fertile area for design research.

Related Work:

J. Koza, *Genetic Programming*, MIT Press, 1992.
T. Nyugen and T. Huang, "Evolvable 3D Modeling for Model-Based Object Recognition Systems," *Advances in Genetic Programming*, MIT Press, 1994.
"Electronic Homunculus Project," (work in progress evolving robotic controllers for Space Station Freedom), Johnson Space Center. Contact: Dennis Lawler (ER 221).

SUMMARY

The main uses of evolutionary/genetic methods in these and other NASA-related domains are:

- as a universal weak method;
- for automatic programming; and
- for designing and modifying structures.

We noted scant application of newer genetic techniques such as niching, steady-state breeding, sharing/crowding to control convergence, parallel-GAs, messy GAs, etc.

The difficulties most often encountered were:

- controlling the program (too many knobs);
- weakness of the theory for designing and guiding the process;
- tradeoffs between expressiveness and efficiency in the representation; and
- representing the true constraints of the problem with one global fitness function.

Further research is needed to develop tools that allow the user to visualize the evolution of the population and how effectively the search space is being explored. The bewildering set of tricks and tweaks that have decorated the research literature for years makes it difficult to decide on a good approach for a given problem; principled guidelines are needed; and a theory able to predict the computational resources required to solve a given problem would be a major boost to the engineering applications of genetic methods. Finally, careful studies of hybrid methods using genetic, neural, and fuzzy techniques are underway, but much remains to be done.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
January 1996

3. REPORT TYPE AND DATES COVERED
Conference Publication

4. TITLE AND SUBTITLE
Computational Intelligence and Its Impact on High-Performance Engineering Systems

5. FUNDING NUMBERS
505-63-50-17

6. AUTHOR(S)
Ahmed K. Noor, Compiler

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
NASA Langley Research Center
Hampton, VA 23681-0001

8. PERFORMING ORGANIZATION REPORT NUMBER
L-17554

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Washington, DC 20546-0001

University of Virginia Center for Computational Structures Technology
Hampton, VA 23681-0001

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
NASA CP-3323

11. SUPPLEMENTARY NOTES
Ahmed K. Noor: University of Virginia Center for Computational Structures Technology, Hampton, VA.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified–Unlimited
Subject Category 59

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
This document contains presentations from the joint UVA/NASA Workshop on Computational Intelligence held at the Virginia Consortium of Engineering and Science Universities, Hampton, Virginia, June 27–28, 1995. The presentations addressed activities in the areas of fuzzy logic, neural networks, and evolutionary computations. Workshop attendees represented NASA, the National Science Foundation, the Department of Energy, National Institute of Standards and Technology (NIST), the Jet Propulsion Laboratory, industry, and academia. The workshop objectives were to assess the state of technology in the computational intelligence area and to provide guidelines for future research.

14. SUBJECT TERMS
Computational intelligence; Fuzzy logic; Neural networks; Evolutionary computations

15. NUMBER OF PAGES
318

16. PRICE CODE
A14

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	