COMBINATORIAL MULTIOBJECTIVE OPTIMIZATION USING GENETIC ALGORITHMS

Final Summary of Research
conducted under NASA Cooperative Agreement NCC-1-01042

William A. Crossley, Associate Professor
Eric T. Martin, Graduate Research Assistant

Purdue University
School of Aeronautics and Astronautics
315 N. Grant Street
West Lafayette, Indiana 49707-2023

COMBINATORIAL MULTIOBJECTIVE OPTIMIZATION USING GENETIC ALGORITHMS

Final Summary of Research conducted under NASA Cooperative Agreement NCC-1-01042

Submitted by:
William A. Crossley, Associate Professor
Purdue University, School of Aeronautics and Astronautics
315 N. Grant Street, West Lafayette, IN 47907-2023

ABSTRACT

The research proposed in this document investigated multiobjective optimization approaches based upon the Genetic Algorithm (GA). Several versions of the GA have been adopted for multiobjective design, but, prior to this research, there had not been significant comparisons of the most popular strategies. The research effort first generalized the two-branch tournament genetic algorithm in to an *N*-branch genetic algorithm, then the *N*-branch GA was compared with a version of the popular Multi-Objective Genetic Algorithm (MOGA). Because the genetic algorithm is well suited to combinatorial (mixed discrete / continuous) optimization problems, the GA can be used in the conceptual phase of design to combine selection (discrete variable) and sizing (continuous variable) tasks. Using a multiobjective formulation for the design of a 50-passenger aircraft to meet the competing objectives of minimizing takeoff gross weight and minimizing trip time, the GA generated a range of tradeoff designs that illustrate which aircraft features change from a low-weight, slow trip-time aircraft design to a heavy-weight, short trip-time aircraft design. Given the objective formulation and analysis methods used, the results of this study identify where turboprop-powered aircraft and turbofan-powered aircraft become more desirable for the 50-seat passenger application. This aircraft design application also begins to suggest how a combinatorial multiobjective optimization technique could be used to assist in the design of morphing aircraft.

RESEARCH DESCRIPTION

DEVELOPMENT OF THE *N*-BRANCH TOURNAMENT GA

The *N*-branch tournament GA is a generalization of the two-branch tournament,[1] which was developed to solve two-objective optimization problems. The *N*-branch tournament method uses the selection operator to perform multiobjective design, rather than formulating a single fitness function like the more popular Multi-Objective Genetic Algorithm (MOGA). With the *N*-branch approach, selection is organized so that designs compete once on each fitness function. To accomplish this, the entire population is copied into a "pot", from which N_{obj} individuals are randomly selected without replacement to compete on the first fitness function. The best performing individual of these designs is added to the "parent pool". The competition using the first fitness is repeated until the pot is empty. At this point, the parent pool will be partially full, containing only the individuals that were selected based upon the first fitness function. The pot is then refilled with the original population and the process is repeated for all other objectives. After the final branch has been completed, the parent pool is full. A flowchart of this process for two objectives appears in Fig. 1. The fitness functions require no user-defined scaling or weighting coefficients for the multiple objectives, and each individual fitness function is evaluated irrespective of other individuals.

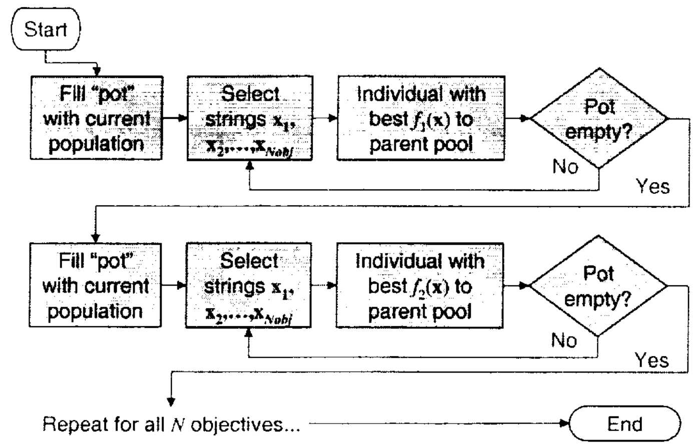

Fig. 1. Flowchart of the N-branch tournament selection.

To solve constrained problems with a GA, fitness functions often use exterior penalty functions. The N-branch tournament uses this idea with a slight modification to allow for objectives of different magnitudes. The first fitness function, f_1, is computed using Equation 1. Equation 2 then computes a penalty-scaling factor for subsequent fitness functions. All other fitness functions are calculated with Equation 3. Using a penalty-scaling factor in this way penalizes all objectives with the same percentage that penalized the first objective.

$$f_1(\mathbf{x}) = \phi_1(\mathbf{x}) + \sum_{j=1}^{J} c_j \max\left[0, g_j(\mathbf{x})\right] + \sum_{k=1}^{K} c_k \left|h_k(\mathbf{x})\right| \tag{1}$$

$$P^* = 1 + \left|\frac{f_1 - \phi_1}{\phi_1}\right| \tag{2}$$

$$f_i = P^* \phi_i \quad i = 2, \ldots, N_{obj} \tag{3}$$

In a two-branch tournament, half of the parents survive based upon their performance in the first objective, and half survive based on the second. If parent couples were chosen from the parent pool randomly without replacement, half of the children (on average) will have parents selected based on different fitness functions, one fourth will have parents both selected based upon f_1, and one fourth will have parents both selected based upon f_2. For a three-branch tournament, two thirds of the children will have mixed-objective parents (on average), and only one ninth of the children will have parents both selected based on any given fitness function. Because most of the population will have parents that won tournaments on separate fitness functions, the compromise region of the Pareto front will be favored with less emphasis on the "ends" where good performance on only one objective is desired.

An improvement to the two-branch tournament that encourages designs at the ends of the Pareto front is the parent-mixing factor.[2] This improvement was easily adapted to the N-branch tournament. The parent mixing factor, α, determines what fraction of parents surviving each branch will crossover with parents surviving another branch. If $\alpha = 0$, there is no mixing, and parents surviving the f_1 branch will only mate with other f_1 survivors. If $\alpha = 1$, f_1 survivors mate only with survivors of other branches. Higher values of α encourage compromise designs that are good in all objectives; smaller values encourage good performance in only one objective near the ends of the Pareto front. This study used the value $\alpha = 1/9$. This value was found to perform well in an empirical study during development of the N-branch tournament. It should be noted that for all necessary parental pairings to occur, population size must be a multiple of $2 \times N_{obj} / \alpha$. If $\alpha = 1/9$, then population size must be a multiple of 54 for a three-objective problem.

Once parents are selected and paired for mating, uniform crossover is used to generate children. After each generation, the set of feasible non-inferior designs is stored as an approximation to the Pareto set. This stored set is updated as new designs dominate previously stored designs.

COMPARISON TO MOGA

The Multi-Objective Genetic Algorithm (MOGA) is a Pareto-based method initially developed by Fonseca and Fleming.[3] MOGA is quite popular, and several researchers have used this approach and / or developed their own versions of this approach. (See for example, Refs. 4, and 5) MOGA generates selection pressure by replacing the vector of objective functions with a single rank-based fitness function. The problem is then treated as a single fitness minimization problem with the rank providing the single fitness value. Individuals in the population receive a rank value based upon their degree of non-inferiority or non-dominance. Here, inferiority is evaluated only with respect to the current population, so it is not necessary for an individual to be in the actual Pareto set to be considered non-inferior. For a given population, the individuals that are non-inferior are assigned the lowest (best) rank and individuals that are highly dominated (dominated by many individuals) are assigned the highest (worst) rank. A fitness value based on this rank is then assigned to each individual.

Kurapati, Azarm, and Wu[6] have developed improved constraint handling techniques for MOGA. The aim of this approach is to improve MOGA's effectiveness and potentially reduce computational cost by evaluating objectives and constraints separately. This version of MOGA was used for the investigations presented here. The procedure is as follows:

Step 1: Evaluate constraints for all individuals

Step 2: Assign a poor rank ($0.95 \times N_{pop}$) to infeasible individuals

Step 3: Assign a moderate rank ($0.5 \times N_{pop}$) to feasible individuals

Step 4: Evaluate objectives of feasible individuals

Step 5: Assign good rank (1) to feasible, non-inferior individuals

Step 6: Assign fitness values to all individuals based on their rank values with penalties based upon the degree and number of constraint violations.

After each generation, the feasible, non-inferior designs are stored in an approximate Pareto set. This set is updated as new designs are found that dominate previously stored designs; the new, non-dominated designs are added and the now dominated designs are removed from the approximate set.

The MOGA approach and the N-branch tournament GA were used to solve several test problems to assess the relative strengths and weaknesses of the two methods. For these comparisons there were two general problem types: "highly constrained" and "adjacent minima". Significant detail of the test problems and more comprehensive discussion of the results are presented in Ref. 7.

The test problems included simple quadratic functions that easily illustrate both highly constrained and adjacent minima problems, a multiobjective version of Golinski's speed reducer problem, and a combinatorial multiobjective version of the ten-bar truss structural optimization benchmark problem. For the quadratic problems and Golinski's speed reducer, all variables are continuous variables. The combinatorial ten-bar truss problem seeks to minimize the weight of the truss, the maximum displacement of the truss and the cost of the truss. Because both the discrete material type and continuous cross-sectional area of each truss element are treated as design variable, the ten-bar truss lends itself well as a benchmark combinatorial multiobjective problem.

For this suite of test problems, which included continuous and combinatorial multiobjective optimization problems, both approaches adequately provided an approximation to the Pareto front. This suggests that both methods are effective tools for multiobjective optimization with three or more objectives. As with many aspects of the GA, the choice for the best approach appears to be problem dependent.

The results of the simple constrained and 10-bar truss problems suggest that N-branch GA may be better suited to solving highly constrained problems. For the simple constrained problem, N-branch GA found more designs near the constraint boundary than the MOGA. For the 10-bar truss problem, *N*-branch was able to find many more low-weight designs, where the stress constraints are active, than the MOGA was able to find.

The results of the adjacent minima problem favor the MOGA approach. While the *N*-branch tournament approach found more total non-dominated designs than the MOGA approach, the MOGA approach provides a much better resolution of the true shape of the Pareto front. *N*-branch GA is unable to find designs in the middle portion of the front. This is due to the selection mechanics of the *N*-branch tournament. Designs that perform well on one of the adjacent minima will also perform well on the other adjacent minimum. Because these designs perform well in two objectives, they are likely to be selected as parents twice as often as designs performing well on the remaining, non-adjacent objective. For many engineering problems, it may not be known if two or more objectives are adjacent *a priori*.

Golinski's speed reducer is both highly constrained and has two adjacent minima. It is not clear which approach is preferred for this type of problem. The MOGA approach finds many more points on the Pareto front than the *N*-branch GA. However, the *N*-branch GA finds better solutions near the extremes of the Pareto set where constraints are active. If the aim is to find the broadest range of solutions, then the N-branch would be favored; however, if the aim is to find a large number of solutions, then MOGA is favored.

MULTIOBJECTIVE DESIGN TO INVESTIGATE AIRCRAFT DESIGN

The two-branch version of the *N*-branch GA was used to generate solutions to an aircraft design problem posed as a combinatorial multiobjective optimization problem. This approach is envisioned for use at the onset of the conceptual design phase, when an aircraft mission, requirements, and design objectives are being formulated and engineering designers are seeking aircraft concepts that satisfy these needs. To help illustrate this approach for aircraft conceptual design, a 50-seat commuter aircraft problem was developed. The well-known aircraft sizing code, FLOPS,[8] provided predictions of aircraft performance, size and weight for this study. More detail about the specific problem formulation and a more comprehensive discussion of the results can be found in Ref. [9].

The 50-seat commuter aircraft problem can be posed as a multiobjective optimization problem. For this work, the objectives were meant to reflect desires of the commuter airline and the aircraft's passengers, but must also be measurable. Minimizing design gross weight is an objective, this traditional aircraft design objective serves as a surrogate for aircraft acquisition cost. Reducing aircraft total trip time also provides an objective; this reflects a passenger's desire.

Because FLOPS predicts the gross weight, size, and performance of an aircraft to complete a given design mission, the range was not directly treated as a design objective. Rather, a fixed distance was used for the design mission and one set of designs representing the tradeoff between low takeoff gross weight and low total trip time is generated. By changing the design mission range and finding a new multiobjective solution set, the third objective of maximizing design range can also be investigated, although this third objective is limited to discrete values of range.

The design variables for the 50-seat commuter problem incorporate continuous sizing variables and discrete selection variables that describe the layout and geometry of the aircraft. Two continuous variables describing the design mission are also included. Discrete selection variables for the 50-seat aircraft include one variable describing the number and placement of the engines, a second variable describing the number of seats abreast in the cabin, and a third variable describing the tail arrangement.

Runs were conducted using design ranges of 500-nmi, 1,000-nmi, and 2,000-nmi. A design range of 1,000 nmi is comparable to existing 50-seat commuter aircraft. The 500-nmi and 2,000-nmi ranges are half and twice the standard 50-seat commuter aircraft design range. Superposing the 500-, 1000- and 2000-nmi range results onto one plot allows examination of the effect of range. Figure 2 presents the results from all three ranges on one objective space plot.

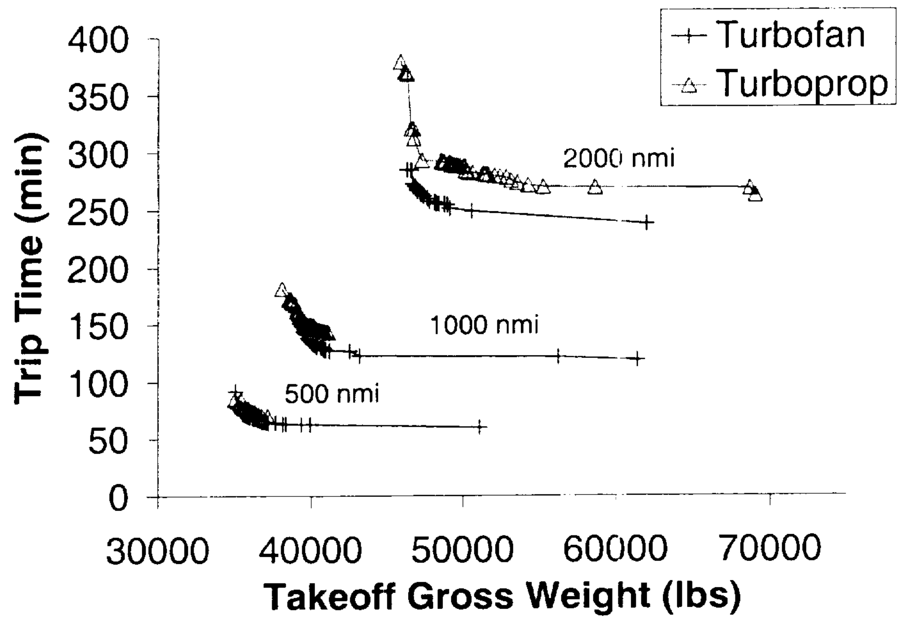

Fig. 2. Superposed approximate Pareto fronts for all 50-seat commuter design mission ranges.

As the design mission range increases, the results show the expected increase in both takeoff gross weight and trip time. For the longest design mission range (2000 nmi), the Pareto front for the turbofans appears to dominate the front for the turboprops, while the shortest design mission range (500 nmi) shows that turboprop designs dominate the turbofan designs for some portion of the front. Also, the distance between the two fronts associated with the 2000 nmi range is much larger than either the 500- or 1000-nmi range results. This may be somewhat intuitive, because the turbofan-powered aircraft will have the opportunity to climb to altitudes at which their advantage in speed significantly impacts trip time and their best cruise Mach number provides for a lower fuel weight.

Using a multiobjective genetic algorithm to investigate the conceptual design space for aircraft is an innovative way to provide design engineers with information about important tradeoffs between the various objectives under consideration. In turn, the engineers can make better decisions about the aircraft concept and size that best meets the design team's interpretation of the design requirements.

A multiobjective GA generates a large number of candidate designs in only one run of the code, with little computational effort. The GA can also consider a large number of design variables, allowing the GA to search a large design space. Because a GA can combine continuous with integer and discrete design variables, engineers can combine concept selection with aircraft sizing in the early stages of the design process, reducing the number of subjective decisions required. The GA can also suggest good concepts without any initial input concept from the design team.

For the 50-passenger commuter aircraft problem, it appears that the GA is able to generate reasonable results. The resulting approximate Pareto optimal designs provide information about which design variables change to provide the tradeoff between trip time and takeoff gross weight. The results generated here suggest that if trip time and takeoff gross weight are the two objectives for this problem, for most of the design space the turbofan-powered aircraft are better designs. The results also provide information to the designers about which design variables change to provide the tradeoffs between trip time and takeoff gross weight, and which variables do not contribute to this tradeoff. The information provided by this investigation of the design space exceeds that normally provided by a traditional selection, then sizing approach that would normally result in one concept or a small number of concepts.

REFERENCES

[1] Crossley, W. A., Cook, A. M., Fanjoy, D. W., Venkayya, V. B., "Using the Two-Branch Tournament Genetic Algorithm for Multiobjective Design," *AIAA Journal*, Feb. 1999, pp 261-275.

[2] Fanjoy, D. W., Beaver, A. K., and Crossley, W. A., 2 "Empirical Studies of Population Size and Parent Mixing for a Multiobjective Genetic Algorithm", AIAA Paper 2000-4892, Sep, 2000.

[3] Fonseca, C. M., and Fleming, P .J., "Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization," *Proceedings of the Fifth International Conference on Genetic Algorithms*, edited by S. Forrest, 1993, pp. 416-423.

[4] Obayashi, S., Tsukahara, T., and Nakamura, T., "Cascade Airfoil Design by Multiobjective Genetic Algorithms," 2nd IEE/IEEE Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications, Glasgow, UK, Sep 2-4, 1997.

[5] Cheng, F. Y., Li, D., "Multiobjective Optimization Design with Pareto Genetic Algorithm," *Journal of Structural Engineering*, Sep 1997, pp 1252-1261.

[6] Kurapati, A., Azarm, S. and Wu, J., "Constraint Handling in Multiobjective Genetic Algorithms," AIAA Paper 2000-4893, Sep, 2000.

[7] Martin, E. T., Hassan, R. A., and Crossley, W. A., "A Comparison of Multiobjective Genetic Algorithm Approaches for *N* Objectives," AIAA Paper 2002-5430, Sep. 2002.

[8] McCullers, L. A., "FLOPS Flight Optimization System, Release 5.94 User's Guide," NYMA, Inc., NASA Langley Research Center, Hampton, VA, Sep. 1998.

[9] Crossley, W. A., Martin, E. T., and Fanjoy, D. W., "A Multiobjective Investigation of 50-Seat Commuter Aircraft Using a Genetic Algorithm," AIAA Paper 2001-5247, 1st AIAA Aircraft Technology, Integration, and Operations Forum, Los Angeles, CA, Oct. 2001.

INVENTIONS

No inventions were developed as a result of this research.

BIBLIOGRAPHY

Publications resulting from this research are listed below.

Crossley, W. A., Martin, E. T., and Fanjoy, D. W., "A Multiobjective Investigation of 50-Seat Commuter Aircraft Using a Genetic Algorithm," AIAA Paper 2001-5247, 1st AIAA Aircraft Technology, Integration, and Operations Forum, Los Angeles, CA, Oct. 2001.

Martin, E. T., and Crossley, W. A., "Empirical Study of Selection Methods for Multiobjective Genetic Algorithm," AIAA Paper 2002-0177, 40th AIAA Aerospace Sciences Meeting and Exhibit, Jan. 2002.

Martin, E. T., Hassan, R. A., and Crossley, W. A., "A Comparison of Multiobjective Genetic Algorithm Approaches for *N* Objectives," AIAA-2002-5430, 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Atlanta, GA, Sep. 2002. (Revised version in preparation for submittal to *AIAA Journal*.)