The Journal of the Astronautical Sciences, Vol. 46, No. 3, July-September 1998, pp. 267–282

Optimal Interplanetary Spacecraft Trajectories via a Pareto Genetic Algorithm

John W. Hartmann,[1] Victoria L. Coverstone-Carroll,[1] and Steven N. Williams[2]

Abstract

A Pareto genetic algorithm is applied to the optimization of low-thrust interplanetary spacecraft trajectories. A multiobjective, nondominated sorting genetic algorithm is developed following existing methodologies. A hybridized scheme is designed integrating the nondominated sorting genetic algorithm with a calculus-of-variations-based trajectory optimization algorithm. "Families" of Pareto optimal trajectories are generated for the cases of Earth-Mars flyby and rendezvous trajectories. A novel trajectory type generated by the genetic algorithm is expanded to develop a series of versatile, high-performance Earth-Mars rendezvous trajectories.

Introduction

Low-Thrust Trajectory Optimization

In recent years, pressure to reduce the costs of interplanetary missions has led to a heightened emphasis on designing missions with shorter flight times, smaller launch vehicles, and simpler flight systems. These requirements have renewed interest in low-thrust propulsion systems due to their high propellant efficiencies; however, the need to optimize the continuous thrust profiles inherent to these systems presents new challenges to trajectory designers. Trajectory optimization techniques currently in use need only model a series of discrete events: launch time, planetary flyby times (including characteristics of the flyby trajectory), and any deep space maneuvers that may be required. This is by no means a trivial exercise when complicated multiple gravity-assist trajectories are considered, but the additional requirement of optimizing a continuous thrust profile severely strains, and in many cases exceeds the capability of traditional techniques.

[1]University of Illinois at Urbana-Champaign, Urbana, IL 61801-2935.
[2]Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109.

Project Overview

At the Jet Propulsion Laboratory (JPL), current working methods for low-thrust trajectory optimization are calculus-of-variations based. This formulation typically has difficulty with longer, more complicated trajectories, and can expend large amounts of time executing inefficient searches. An on-going joint research venture being conducted between JPL and the University of Illinois at Urbana-Champaign's Computational Astrodynamics Research Laboratory (UIUC CARL) is investigating alternative methodologies in an attempt to discover new, more robust, and more efficient optimization algorithms. Automation of the optimization process using stochastic search techniques (e.g., simulated annealing and genetic algorithms) to drive the current optimization software is a major topic of research.

In this paper, a Pareto genetic algorithm is developed following an existing method outlined in Srinivas and Deb [1]. This particular method approaches the problem of locating Pareto optimal solutions by implementing the concept of nondominated sorting. The algorithm is then used to perform a multiobjective optimization for low-thrust orbit transfers, expanding the previous work of Rauwolf and Coverstone-Carroll [2].

The optimization techniques discussed address the problem of low-thrust trajectory optimization, and provide for the generation of optimal sets or "families" of these trajectories. Generation of a trajectory family is accomplished by sorting the population according to an individual's Pareto optimality. Such sets illustrate the relationships between different flight characteristics and provide compromise solutions when maximum performance (i.e., mass delivered to destination) is one of several requirements. Additionally, the new method provides increased robustness through its inherent separation of objectives and eliminates the objective conflict [3] that arises from classical techniques involving scalarization of multiple objectives. Objective conflict occurs when improper assignment of weighting factors results in a bias towards certain objectives. The Pareto genetic algorithm is then combined with existing JPL trajectory optimization software in a hybridized manner to produce a more informed search, along with familiar and usable results.

Pareto Optimization

Multiobjective Optimization

Multiobjective optimization, as its name implies, differs from single-objective optimization in that the intent is to optimize a system with more than one objective. As in single-objective optimization, the objective(s) may have any number of equality or inequality constraints imposed upon them. This can be represented mathematically as [4],

$$\begin{array}{lll} \text{Minimize/Maximize:} & f_i(\mathbf{x}) & i = 1, 2, \ldots, N \\ \text{subject to:} & g_j(\mathbf{x}) \le 0 & j = 1, 2, \ldots, F \\ & h_k(\mathbf{x}) = 0 & k = 1, 2, \ldots, K \end{array} \tag{1}$$

Rather than searching for the solution which yields the globally maximal (or minimal) value for a single objective, the "best" solution is found by simultaneously optimizing several objectives. These types of optimization problems have traditionally been solved by averaging each objective with a weighting factor, then combining the objectives into a single scalar objective. Such reduction techniques eliminate the need for a more complex multiobjective algorithm, but introduce new parameters in the form of weighting factors [3]. Individual solutions are highly sensitive to the magnitudes of these weighting factors. The user must become familiar with the exact relationship between objectives in order to determine the proper weighting values that will yield the desired result. Determination of the correct weighting factors can, in fact, become an optimization process of its own.

In addition, the result of the optimization will be a single solution rather than a set of compromise solutions illustrating the relationship between different objectives. This is acceptable for cases in which optimality of all objectives coincide in the same solution, but in most cases where adjustments beneficial to one objective are detrimental to others, such a technique provides only a single point on what may be an expansive "front" of possible solutions. In the case of genetic algorithms (GAs) in particular, such a formulation fails to take advantage of the population-based nature of the technique. It therefore becomes desirable to develop a more robust multiobjective algorithm, capable of identifying the trades between objectives and able to make better use of the population-based GA to produce sets of optimal solutions.

Genetic Algorithms

Genetic algorithms are an optimization procedure based on natural selection and Darwinian genetics [5]. These algorithms differ from the more traditional optimization techniques in that they work with a coding of the parameter set (usually binary) instead of the parameters themselves. GAs search from a population of points instead of a single point, use only the objective function evaluation instead of derivatives or other auxiliary knowledge, and use probabilistic transition rules instead of deterministic rules. A population of individual solutions is evolved over a series of generational cycles, each undergoing alterations to their respective parameter set and objective function value ("fitness").

The genetic algorithm used in this paper employs the standard GA operators of selection, crossover, and mutation to perform the evolutionary search. Selection is the process of choosing two solutions on which crossover of information may occur. Crossover is used to create new solution strings or "children" from existing "parent" strings. A string is a binary representation of all the independent variables. Through crossover, beneficial information encoded in the parents has a chance to propagate to future solutions. The role of mutation is mainly that of protecting the population of solutions against irrevocable losses of potentially helpful information that may be lost in weak individuals. Mutation also protects and provides diversity within the population.

"Niching" or "fitness sharing" is a supplemental strategy that is also included to promote diversity. In this procedure, solutions are penalized according to their degree of similarity to others in the population in an attempt to distribute members over large portions of the search space and avoid premature convergence to a single solution. The population is examined on a per individual basis to determine the number of other members that reside within the neighborhood of the current examinee. The size of this neighborhood is determined by the niching control parameter, σ_{share}. A penalty value is associated with each individual residing within this neighborhood. The magnitude of the penalty is governed by a sharing function whose value falls off as proximity to the examinee decreases by the power α, the second of two niching parameters. A value of $\alpha = 1$ corresponds to a linear reduction in magnitude, with the effective penalty approaching zero as individual proximity approaches the neighborhood boundary, σ_{share}. The number of individuals residing within this defined neighborhood and their corresponding penalties are tallied, and the current examinee's fitness is penalized by an amount proportional to that number at the end of each generational cycle. For a more complete discussion of genetic algorithms and niching, see Goldberg [5].

Pareto Optimality

Pareto optimization is the principle of optimizing multiple competing objectives. The problem is essentially one of finding optimal solutions based on criteria that have inverse relationships. Edwin Dean [6] succinctly described the problem:

> *A Pareto optimal solution is not unique, but is a member of a set of such points which are considered equally good in terms of the vector objective. This space may be viewed as a space of compromise solutions in which each objective could be improved upon, but if it was, it would be improved at the expense of at least one other objective.*

Another way of stating this would be to say that a solution is Pareto optimal, or "nondominated," for a given set of objectives if there is no other existing solution that is superior to that solution in *all* objectives. If a solution exists that is superior in *some* objectives, then that solution may constitute a point on a front of Pareto optimal solutions. Take for example the problem of minimizing both coordinates for a set of points, {(0,5), (1,3), (2,4)}. Point 1 is dominated by points 2 and 3 in its second coordinate; however, it dominates points 2 and 3 with respect to the first coordinate, and therefore is nondominated. Point 2 is dominated neither by point 1 nor point 2 since it is superior to these two with regard to its second coordinate, and thus is also nondominated. Point 3, while superior to point 1 in its second coordinate, is dominated with respect to *both* coordinates by point 2, therefore it is a dominated individual and not Pareto optimal.

The Pareto Genetic Algorithm

The algorithm devised in this study is based on the concept of nondominated sorting originally conceptualized by Goldberg [5] and developed by Srinivas and Deb [1] as the nondominated sorting genetic algorithm (NSGA). The NSGA is one of several comparable methods for Pareto optimization [7, 8], but is unique in its use of nondominated sorting. It is not an entirely new optimization algorithm, but rather a modification to the fitness evaluation procedures that exist in standard genetic algorithms. NSGA is in some sense a supplement to a genetic algorithm that allows for a more effective means of multiobjective optimization.

The NSGA uses the concept of nondominance to sort through a population of possible solutions, assigning each member to a Pareto optimal "front" according to their level of nondominance. The process consists of two iterated steps. The population is first sorted according to the problem objectives, and those individuals that are nondominated with respect to these objectives are assigned an artificial fitness associated with their level of Pareto optimality. For the first iteration of the nondominated sorting routine, the fitness value assigned establishes nondominated members as possessing the highest level of Pareto optimality in the population at large. This initial fitness is arbitrarily chosen to be a value equal to the size of the population.

After assignment to a front, fitness sharing (or niching) is applied to these individuals where the artificial fitness is adjusted according to the solutions' proximity to one another in an attempt to evenly distribute individuals across the Pareto front. After sharing, the minimal fitness in the current front is determined. This fitness is then slightly reduced and used as the initial artificial fitness for the next front in the next iteration of the sorting routine. During this iteration and all subsequent ones, individuals previously tagged as nondominated and assigned to a specific front are excluded from examination, and the remaining subpopulation is examined to determine the next subgroup of nondominated solutions. The nondominated sorting process continues this iterative evaluation of subpopulations until all individuals in the population are assigned to a specific front and given a fitness value.

In order to further illustrate the fitness assignment process that takes place inside the algorithm, let us once again examine the simple example of minimizing both coordinates for the set of points, {(0,5), (1,3), (2,4)}. As was previously determined, points 1 and 2 are readily distinguished as nondominated solutions. This being the case, they can be separated from the population as rank 1 individuals. Given a population size of 3 for this simple example, the initial artificial fitness value is also 3. Points 1 and 2 are thus initially given fitness values of 3.0.

Niching is then applied within the current front in an attempt to distribute individuals along the front. If solutions within a front encroach upon one another's immediate vicinity, they are penalized. This vicinity is defined by the σ_{share} parameter. Points 1 and 2 are separated by a Euclidean distance of 2.236. If σ_{share} is less than or equal to this value, each individual will be penalized. Since typical values for σ_{share} are usually much smaller than this, these individuals would most likely not be penalized. If this is the case, their fitness values would remain at 3.0 and they would be removed from the population at large since the nondominated sorting is not yet complete. To finish the sorting, the current artificial fitness level is reduced by some arbitrarily small amount (say 1% of the lowest fitness in rank 1) and the remaining population is examined to determine the individuals that comprise the next Pareto front. In this example, only one solution remains. Point 3 thus becomes the sole rank 2 individual and is assigned a fitness value of $(3.0 - 0.03) = 2.97$.

The benefits of incorporating a Pareto search algorithm in the trajectory optimization process are twofold: i) elimination of the problems encountered in classical multiobjective optimization methodologies such as objective conflict, and ii) development of a Pareto optimal front of solutions, providing an array of compromise solutions. When applied to the population-based genetic algorithm, these Pareto concepts should enable automatic generation of Pareto optimal solutions. In the context of spacecraft trajectory optimization discussed in this study, a Pareto genetic algorithm should provide the mission designer the capability of generating "families" of optimal trajectories illustrating the trades between defined objectives.

Verification of Algorithm Performance

A set of test functions comprised of those found in the literature as well as some of original design were used to test the NSGA. Six diagnostic test functions in all were used. Three of these functions were taken from the Srinivas and Deb test suite to test replication of performance, while three new functions were designed to test previously undemonstrated capabilities [1] required for the trajectory optimization work in this study: optimization of problems with more than two objectives, and with objectives of both minimizing and maximizing type. In order to facilitate comparison, values for main control parameters were assigned the same values as those used in the original NSGA study [1,9].

The test functions considered verified that the version of nondominated sorting algorithm developed in this study performed accurately. Tests to identify Pareto optimal regions for more than two objectives, as well as those to verify the NSGA's ability to handle combinations of optimization types were successful. For further details of these verification tests, see Hartmann [9].

NSGA/SEPTOP Hybridization

The algorithm formulated for the application of trajectory optimization does not consist solely of the Pareto genetic algorithm coupled with a function evaluation subroutine. It is a hybridization of the nondominated sorting genetic algorithm and the calculus-of-variations-based trajectory optimizer, SEPTOP [10]. The NSGA is used as a driver for the SEPTOP software, essentially automating the optimization process by acting as a kind of "smart" user.

Brief Description of SEPTOP

SEPTOP (Solar Electric Propulsion Trajectory Optimization Program) is a preliminary mission planning tool that uses a two-body, sun-centered, low-thrust solar-electric propulsion model. Classical calculus-of-variations (COV) is used to obtain a maximum final (burnout) mass, resulting in a two-point boundary value problem (TPBVP). The user provides initial estimates for costates (Lagrange multipliers) and the state and costate differential equations are forward integrated. Terminal constraints on the states and costates that result from the COV formulation must be satisfied. The convergence of SEPTOP to an optimal trajectory can be highly dependent on the user's initial guess and the relative difficulty of the mission. As the number of intermediate planetary flybys and total number of revolutions about the sun is increased, the user's initial guess must move closer to their converged values for the TPBVP to be successfully solved.

The Nondominated Sorting "Memetic" Algorithm

The NSGA evolves populations of individuals representing possible trajectories, with input parameters for SEPTOP being encoded as each individual's genotype. These input parameters include the Lagrange multiplier values associated with a given trajectory, as well as the total time of flight. Individual fitness is evaluated through a call to SEPTOP using the input parameters encoded within. SEPTOP is run for a set number of iterations, effectively executing a localized search for each member in an attempt to better identify any basins of attractions that might exist in the individual's immediate vicinity. The improved fitness (if any improvement was seen) is returned to the NSGA in the form of an objective vector containing the values for each objective in the multiobjective optimization. The individual is thus assigned a new fitness. The new SEPTOP input parameters associated with the improved solution are *not* returned as the individual's new parameter set, i.e. genotype. This is done in an attempt to maintain a greater amount of diversity in the population's gene pool.

The procedures just catalogued describe what might be termed a "memetic" algorithm—a genetic algorithm coupled with a nongenetic localized search algorithm to improve individual characteristics [11]. This terminology is derived from the study of memetics that theorizes that ideas can evolve in ways analogous to biological evolution [12]. The algorithm used in this study might then be labeled the nondominated sorting *memetic* algorithm (NSMA). Furthermore, the algorithm described is one employing a "Baldwinian learning strategy" [13]. The specific characteristics returned from the improved individual constitute the type of learning strategy implemented. In a Baldwinian learning strategy, only the improved fitness value is recorded.

Within this structure, the SEPTOP software is programmed to return the following four parameters: mass delivered, time of flight, number of heliocentric revolutions, and SEPTOP convergence error (a normalized representation of the degree of convergence). Three of the parameters (mass delivered, time of flight, and number of heliocentric revolutions) are modeled as objectives, and the fourth (SEPTOP convergence error) is modeled as one of three constraints. The additional two constraints limit population members through user-specification of trajectory structure, establishing upper and lower bounds on the allowable number of heliocentric revolutions for a given trajectory. This added control enables the user to implement any prior understanding of the "physics" of the problem to narrow the search space—which might otherwise be overly expansive—to a region that one expects the solution to fall within. These constraints are then applied at the end of the evaluation procedure using appended penalty terms.

The trade relationship between the two objectives, mass delivered and time of flight, is the key focus of this study. The addition of the third objective of maximizing heliocentric revolutions is employed as a strategy for obtaining viable solutions. When the program is initiated, the solution space is populated almost exclusively with individuals far from convergence (i.e., trajectories whose final states do not closely match the final boundary conditions). Beginning with a fixed initial mass, many trajectories exist which may be evaluated as Pareto optimal despite their degree of convergence due to very short flight times and/or minimal thrusting. These are uninteresting solutions, but may dominate the solution space without additional control. Using the SEPTOP convergence error term as a constraint is the supposed safeguard against such erroneous results; however the nature of this parameter is such that decreasing amounts of information are gained from it as a solution moves further from convergence. As a result, using the convergence error as a constraint provides little guidance at the beginning stages of the search. In order to increase the selection pressure placed on individuals with more appropriate characteristics and avoid undesirable premature convergence, a third objective of maximizing heliocentric revolutions is added. This provides an objective inversely related to time of flight, and produces a more balanced Pareto optimization.

Results

Control Parameters

Two test cases were run to provide proof-of-concept for the hybridized method. The control parameters for the genetic algorithm used for all trajectory optimizations discussed are listed in Table 1. Selection and crossover operators were implemented with no mutation in order to maintain consistency with previous testing. Niching parameters were calculated based upon the methods provided in Deb and Goldberg [14], setting q, the induced number of niches, to 15—the same value proportional to population size as in the diagnostic test functions. Existing methodologies for establishing σ_{share} are only rough guides however, since they require prior knowledge of the search space for each optimization problem. Guidance for population sizing and maximum generation determination for hybridized methods was found to be nonexistent at the time of this study, therefore population sizing and number of generations run were determined experimentally through both trial and error and limits on processing capacity. A thorough study of the effects of population size was not conducted. A population size of 150 was used as it provided a good compromise between quality of results and length of time to convergence.

A Delta II 7925 launch vehicle was used for both test cases along with a single 30-cm xenon engine for spacecraft propulsion. A 3.0 kW solar array was used for the flyby case, and a 5.2 kW solar array for the rendezvous case. For further details concerning engine characteristics and solar array modeling, see Williams and Coverstone-Carroll [15]. All simulations were run on a Sun Ultra 10 workstation with a 333 MHz ULTRA Sparc IIi processor. Run times for the Earth-Mars flyby and Earth-Mars rendezvous were approximately 16 hours and 240 hours respectively, as clocked by the UNIX *time* command.

Earth-Mars Flyby

The first case was an Earth-Mars flyby trajectory. Trajectories were constrained between 0.0 to 1.0 heliocentric revolutions. The time of flight parameter was bound between approximately two and twenty months, a time span predicted to be large enough to encompass the designated trajectory types. Launch date was fixed at September 1, 2005, a date close to the optimum ballistic launch date.

TABLE 1. Control Parameters for Pareto Trajectory Optimizations

Control Parameter	Parameter Value
Population Size	150
String Length	80
Probability of crossover	1.0
Probability of mutation	0.0
σ_{share}	0.033
α	2.0
Number of parameters	8
Number of objectives	3
Number of constraints	1

The genetic search was executed for 20 generations, resulting in the family of Pareto optimal trajectories illustrated in Fig. 1. For the purpose of this study, the term "family" refers to a grouping of solutions in the Pareto space, each related to one another through a continuous Pareto curve. Trajectories along this curve may have very different sequences of thrust and coast arcs. Recall that for an individual to be Pareto optimal means that no other individual has a lower cost in all three objective criteria (mass, time of flight and heliocentric revolutions.) The solid line shown on Fig. 1 indicates the actual Pareto curve, generated by iterative calls to SEPTOP. Individuals representing converged Pareto optimal trajectories are indicated with an $\times$. Solutions for which trajectory plots have been included (Fig. 3) are additionally labeled by individual number. Of the 150 members of the population, 112 were determined to be converged Pareto optimal trajectories. These 112 solutions reside in an evenly distributed manner on the actual Pareto curve, demonstrating the NSMA's ability to discover the Pareto relationships for a given low-thrust trajectory optimization problem.

The addition of the heliocentric revolutions objective, while an effective strategy for producing viable solutions, also allows for the generation of solutions which are not Pareto optimal with respect to the trade study strictly between mass delivered and time of flight. This third objective allows for the generation of trajectories with maximal revolutions, but inferior performance and times of flight. It can be seen from Fig. 2 that several of the trajectories generated are non-Pareto optimal with respect to the performance vs. transfer time trade. They inhabit the far right end of the time of flight spectrum (transfer time $>$ 1.25 yrs). At this point on the graph, performance falls off as transfer time continues to increase. Trajectories in

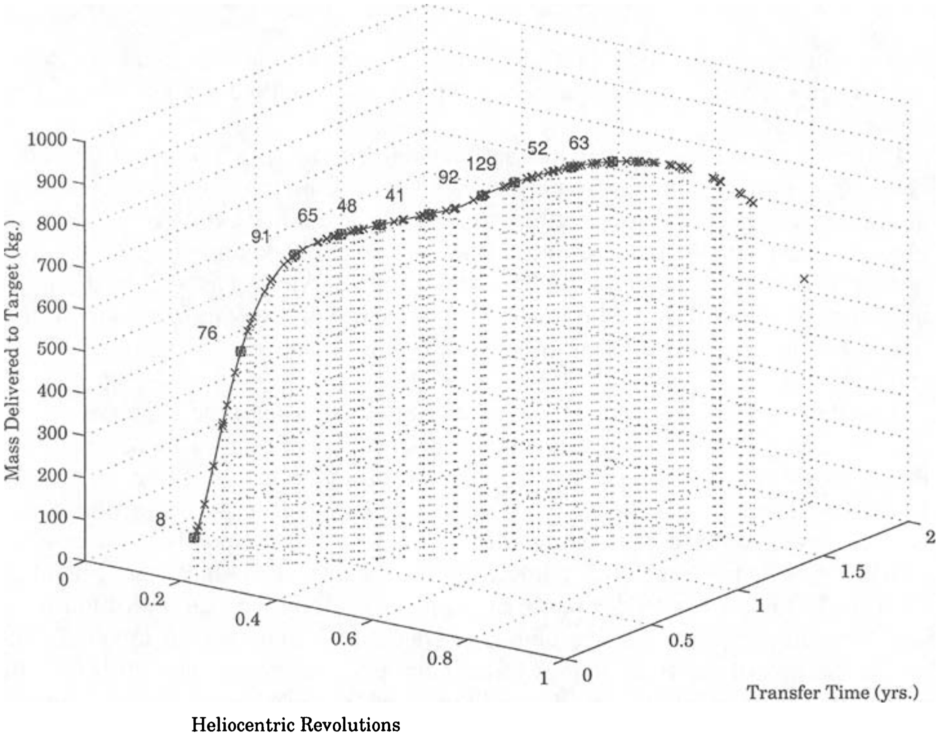

FIG. 1. Pareto Front for Earth-Mars Flyby Trajectory Case at Generation 20.

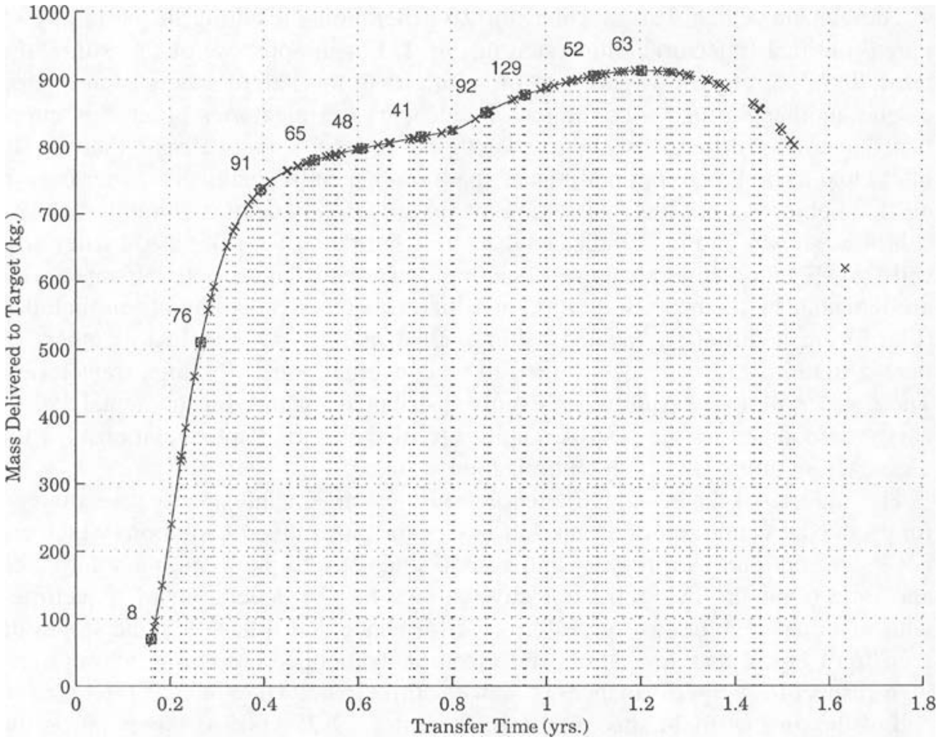

FIG. 2. Mass Delivered Vs. Time of Flight for Earth-Mars Flyby Pareto Optimal Trajectories.

this region are of little use since alternatives exist which deliver greater payloads in less time. These solutions, however, are easily identified for exclusion from final consideration.

Figures 1 and 2 reveal that the optimization produced a family of solutions defining a Pareto optimal front that exists as a smooth curve in the three-dimensional objective space rather than a surface of Pareto optimal points. Figure 3 illustrates the indexed individuals, whose corresponding trajectories span the entire family of solutions along the Pareto curve. As is the case for all trajectory plots to follow, solid line segments represent thrusting arcs, and dashed line segments coast arcs. From these three figures, some interesting observations can be made.

The Pareto front is rooted at individual 8. As transfer time increases, mass delivered rises sharply, with gains in performance tapering off as more significant thrust arcs begin to appear in the vicinity of individual 91. It can be seen from Figs. 1 and 3 that the first portion of the Pareto front—the section of the curve approximately between individuals 8 and 41—corresponds to trajectories consisting of no more than 0.5 heliocentric revolutions, with flight paths bound by Earth and Mars orbits. Performance again picks up following the inflection point at 0.5 revolutions (0.8 yrs), as outbound trajectories transition to inbound. The second portion of the front acquires maximum performance at individual 63 with a delivered mass of 913.96 kg. Beyond this point, performance decreases sharply, and individuals are no longer nondominated with respect to the objectives of mass delivered and time of flight.

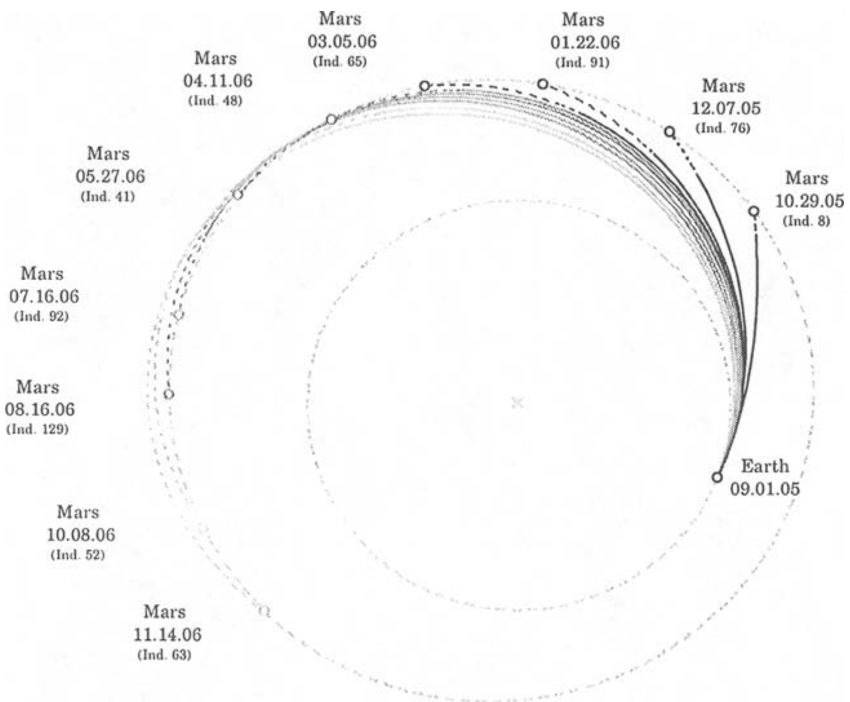

FIG. 3. Earth-Mars Flyby Trajectories for Indexed Individuals.

Earth-Mars Rendezvous

The second set of results are those for the optimization of Earth-Mars rendezvous trajectories. Trajectories were restricted to 0.5 to 3.0 heliocentric revolutions, flight time bound between approximately 10 months and 3.5 years, and launch date again fixed at September 1, 2005. Figure 4 illustrates the population of converged Pareto optimal trajectories at generation 30. In this case, 91 of the 150 population members were nondominated.

Three distinct groupings of Pareto optimal individuals were identified. These were again defined by curves in the three-dimensional objective space, with an outlying individual (individual 147) indicating a fourth area of Pareto optimality. Figure 5 reveals the largest and most evenly distributed family in the Pareto space—bound by individuals 44 and 36—to be the largely dominating subgroup. All but one of the trajectories comprising the other three families are not dominated by members of this group with respect to the mass delivered vs. time of flight trade. As in the first trajectory case, these individuals are a small subpopulation which can be easily identified for exclusion from final consideration since alternative solutions exist which deliver greater payloads in less time.

Beginning with individual 44 (Fig. 6) from the largest family with a delivery mass of 685.28 kg, performance increases rapidly with increasing transfer time. This performance plateaus as a coast arc appears in the trajectories neighboring individual 73, and delivery masses reach approximately 862 kg. Only very small improvements in mass delivered are seen on the section of the curve between individuals 73 and 72. A second coast arc appears producing a burn-coast-burn-coast sequence for these solutions. In the vicinity of individual 72, an additional

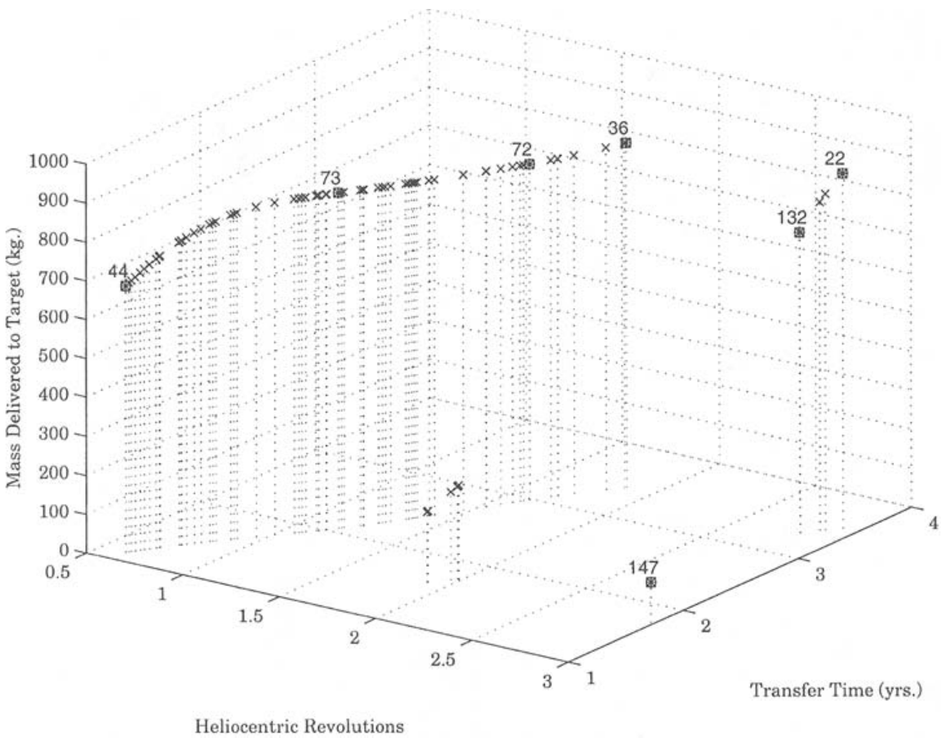

FIG. 4. Pareto Fronts for Earth-Mars Rendezvous Trajectory Case at Generation 30.

burn is appended creating burn-coast-burn-coast-burn trajectories and performance again rises. Performance for this subgroup of solutions approaches its maximum as trajectories transition back to a burn-coast-burn structure at individual 36 with a mass delivered of 884.10 kg.

The small family of solutions whose population resides between individuals 132 and 22 is also worthy of discussion due to the high performance and novel trajectory structure (Fig. 7) existent in its members. These solutions begin by taking an inward direction and spend some time performing heliocentric revolutions within or just outside of Earth orbit before spiraling out to Mars. In these subjects, two revolutions are made to increase the spacecraft's orbit inclination to more closely match that of Mars before the trajectory progresses outward. A significant increase in performance is seen as flight times increase. The minimum mass delivered for this subset of the population is 773.33 kg (individual 132) with a maximum of 911.78 kg (individual 22)—the highest payload delivery in the population.

Earth-Mars Rendezvous On-Demand

The high performance obtained by members of this Pareto family of novel solutions prompted further investigation into the potential of such a trajectory class. The control parameters for the trajectory with the highest performance (individual 22 with 911.78 kg) were extracted, and the SEPTOP software run to compute the performance for trajectories of this type over a range of launch dates

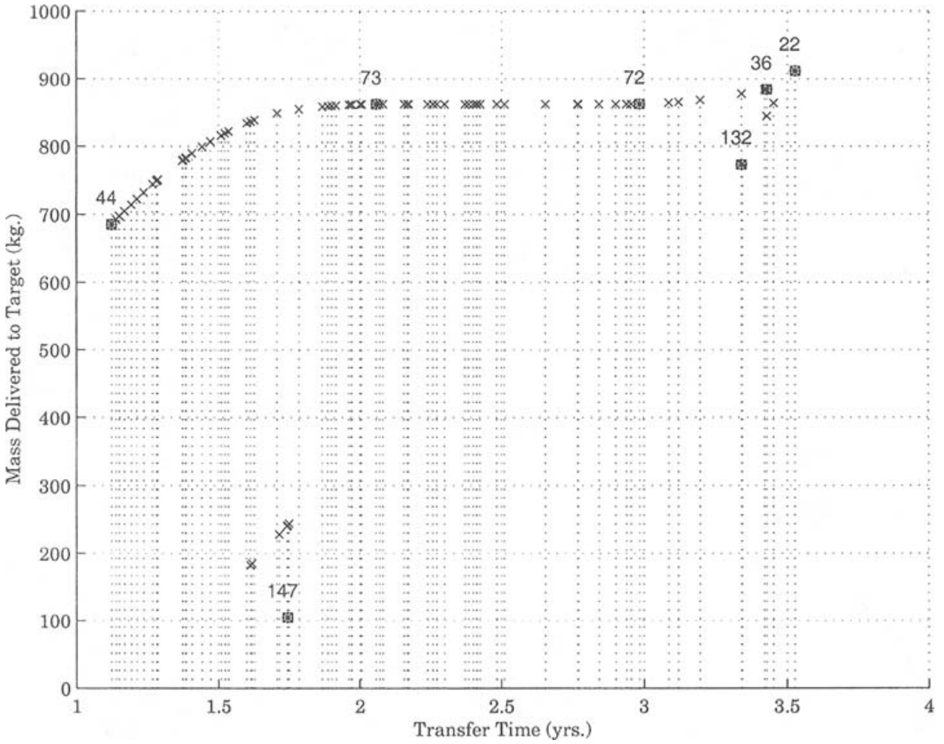

FIG. 5. Mass Delivered Vs. Time of Flight for Earth-Mars Rendezvous Pareto Optimal Trajectories.

spanning approximately one Martian synodic period. For purposes of comparison, the same control parameters were also used to generate a series of trajectories over the same range of launch dates with transfer times of 1.5, 2.5, and 3.0 years.

Results of this analysis are summarized in Fig. 8. The solid curves in the figure represent multiple revolution SEPTOP solutions (2 and 3 revolutions) possessing flight times ranging from 2.5 to 3.55 years. The dashed curve provides a comparison with a more typical solution: 1.5 years and less than 1 heliocentric revolution. These curves reveal a continuous period of launch dates for a flight time of 3.55 years, all with final spacecraft mass greater than 900 kg. Shorter flight times with very large (but not continuous) launch periods are also available, such as the 2.5 year curve which has a launch period close to one year with performance greater than 900 kg.

Conclusions

Based upon the results seen in this study, the hybridization of a Pareto genetic algorithm with a calculus-of-variations optimizer as a local improvement procedure proves an effective method for generating sets of optimal interplanetary low-thrust trajectories. Families of optimal trajectories were obtained in each test case, with family members related through continuous Pareto curves. Best results were obtained for simple, low-rev trajectories. As trajectory complexity increased, populations were distributed less evenly over apparent Pareto curves. These population distributions may improve, discovering new portions of Pareto

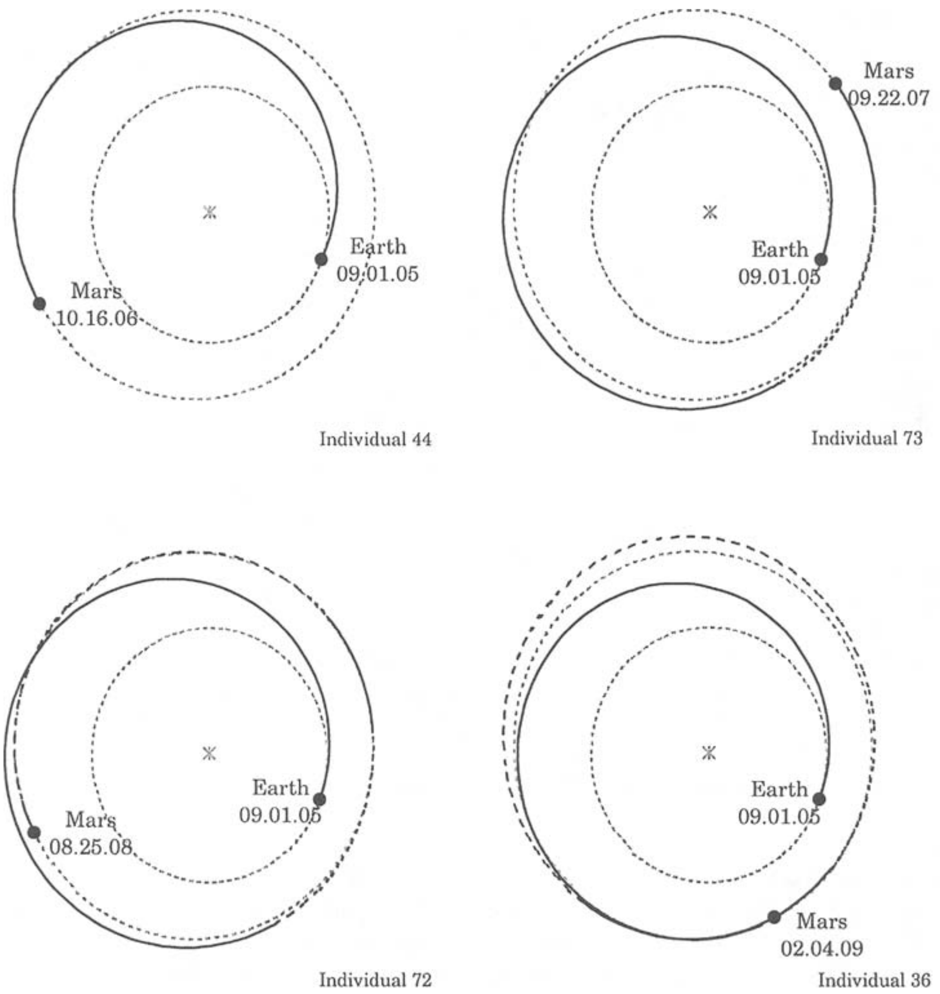

FIG. 6. Representative Trajectories for Earth-Mars Rendezvous Dominant Family.

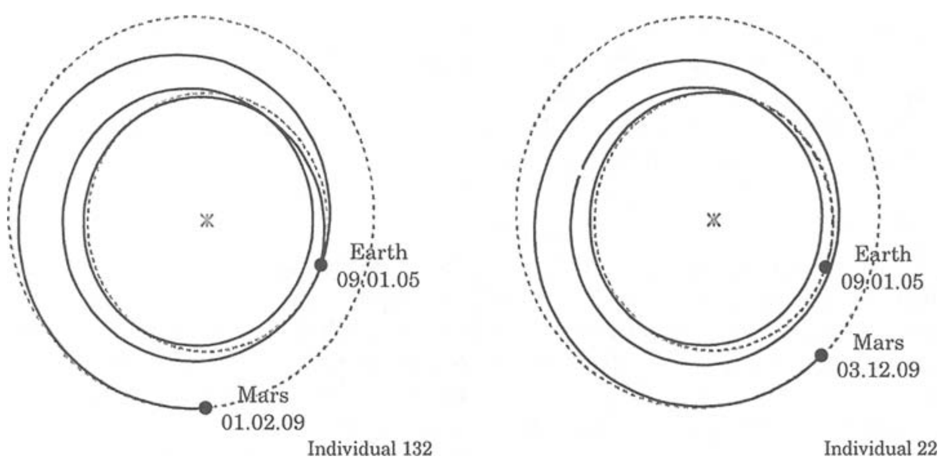

FIG. 7. Representative Trajectories for Earth-Mars Rendezvous Novel Family.

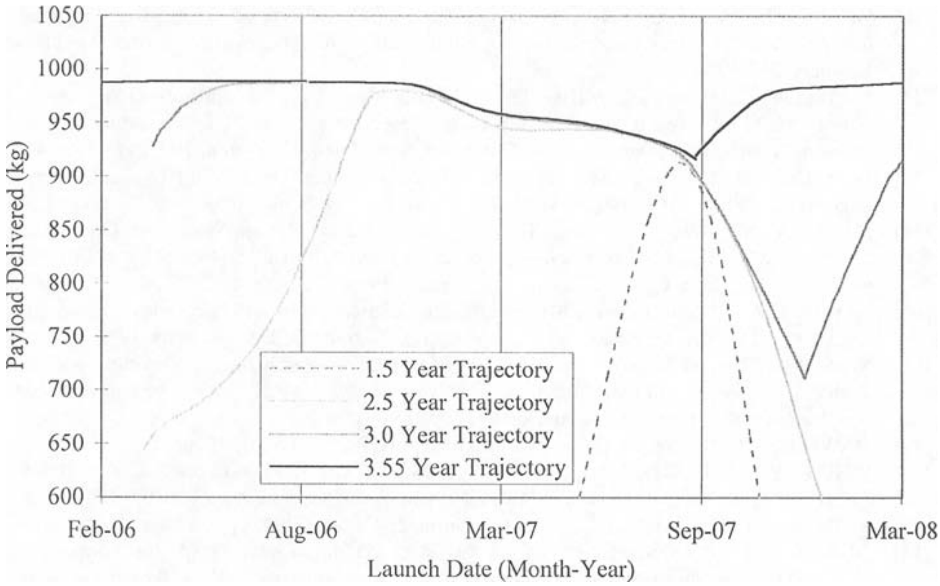

FIG. 8. Earth-Mars Rendezvous Performance for Various Flight Times.

curves or surfaces or filling in those partially populated at algorithm termination, with further generational cycles.

The algorithm also proved useful in producing novel trajectories. The new solutions discovered possessed both non-intuitive structures and very high performance. Unique trajectories found by the genetic search were used to generate a new and versatile trajectory class with a continuous period of launch dates and improved performance.

Acknowledgments

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. The authors would also like to thank D. Goldberg for initially bringing to our attention the algorithm implemented in this study, and for providing many useful leads, including a referral to K. Deb whose advice was critical to this study. Many thanks as well to W. Mason and J. Norr for their contributions to this project, and D. Carroll for developing the genetic algorithm software used in this study.

References

[1] SRINIVAS, N., and DEB, K. "Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms," *Evolutionary Computation,* Vol. 2, No. 3, 1995, pp. 221–248.

[2] RAUWOLF, G., and COVERSTONE-CARROLL, V. "Near-optimal Low-thrust Orbit Transfers Generated by a Genetic Algorithm," *Journal of Spacecraft and Rockets,* Vol. 33, No. 6, 1996, pp. 859–862.

[3] HANS, A. E. "Multicriteria Optimization for Highly Accurate Systems," *Multicriteria Optimization in Engineering and Sciences,* edited by W. Stadler, Vol. 19, Mathematical Concepts and Methods in Science and Engineering, Plenum Press, New York, 1988, pp. 309–352.

[4] RAO, S. S. *Optimization Theory and Application,* Wiley Eastern Limited, New Delhi, 1991.

[5] GOLDBERG, D. E. *Genetic Algorithms in Search Optimization and Machine Learning,* Addison-Wesley, Reading, MA, 1989.

[6] DEAN, E. B. "Multi-objective Optimization from the Perspective of Competitive Advantage" [online], NASA Langley Research Center, http://mijuno.larc.nasa.gov/dfc/mdo/moo.html [cited February 3, 1999].

[7] FONSECA, C. M., and FLEMING, P. J. "Genetic Algorithms for Multi-objective Optimization: Formula, Discussion, and Generalization," *Proceedings of the Fifth International Conference on Genetic Algorithms,* Morgan Kaufmann, San Mateo, California, 1993, pp. 416–423.

[8] SCHAFFER, J. D. "Some Experiments in Machine Learning Using Vector Evaluated Genetic Algorithms," Ph.D. Dissertation, Vanderbilt University, Nashville, Tennessee, May 1984.

[9] HARTMANN, J. W. "Low-thrust Trajectory Optimization Using Stochastic Optimization Methods," M. S. Thesis, Department of Aeronautical and Astronautical Engineering, University of Illinois at Urbana-Champaign, Illinois, January 1999.

[10] SAUER, C. G. "Optimization of Multiple Target Electric Propulsion Trajectories," AIAA Paper 73-205, AIAA 11th Aerospace Sciences Meeting, Washington, D.C., January 1973.

[11] MOSCATO, P., and NORMAN, M. G. "A 'Memetic' Approach for the Traveling Salesman Problem," *Proceedings of the International Conference on Parallel Computing and Transputer Applications,* IOS Press, Amsterdam, 1992, pp. 187–194.

[12] DAWKINS, R. *The Selfish Gene,* Oxford University Press, Oxford, 1976.

[13] WHITLEY, D., GORDON, S., and MATHIAS, K. "Lamarckian Evolution, The Baldwin Effect and Function Optimization," *Parallel Problem Solving from Nature-PPSN III,* edited by Y. Davidor, H. Schwefel, and R. Manner, Springer-Verlag, 1994, pp. 6–15.

[14] DEB, K., and GOLDBERG, D. E. "An Investigation of Niche and Species Formation in Genetic Function Optimization," *Proceedings of the Third International Conference on Genetic Algorithms,* Morgan Kaufmann, San Mateo, California, 1989, pp. 42–50.

[15] WILLIAMS, S., and COVERSTONE-CARROLL, V. "Benefits of Solar Electric Propulsion for the Next Generation of Planetary Exploration Missions," *Journal of the Astronautical Sciences,* Vol. 45, No. 2, April–June 1997, pp. 143–159.