JOURNAL OF AIRCRAFT
Vol. 36, No. 1, January–February 1999

Multidisciplinary Conceptual Design Optimization of Space Transportation Systems

Lawrence F. Rowell* and Robert D. Braun*
NASA Langley Research Center, Hampton, Virginia 23681-0001
John R. Olds†
Georgia Institute of Technology, Atlanta, Georgia 30332-0150
and
Resit Unal‡
Old Dominion University, Norfolk, Virginia 23529

This paper presents a recent history of progress both in disciplinary modeling and in optimization methods and frameworks for space transportation systems conceptual design and analysis. The disciplinary models and process typically used for space transportation analyses are identified, including physics-based and empirical models. The diverse characteristics of these disciplinary models require equally diverse integration and optimization approaches to enable implementation of automated, multidisciplinary design systems. Two general approaches are described for integrating these disciplinary models into computational frameworks for automated vehicle synthesis and optimization. Several optimization approaches are discussed including parameter, gradient-based, stochastic, and collaborative methods. Representative examples are given of multidisciplinary applications of optimization methods to the launch vehicle conceptual design problem. A primary goal for the future is to enable a space transportation design-to-cost capability.

Introduction

THE design of large, complex, aerospace systems, such as Earth-to-orbit (ETO) launch vehicles, requires making appropriate compromises to achieve balance among many coupled objectives such as safety, high performance, simple operations, and low cost. The earlier in the design process that these compromises can be understood, the greater the potential for reduction of technical, schedule, and cost risks. Conceptual design is intended to reveal trends and allow relative comparisons among alternatives early in the process while design flexibility exists, and before a large percentage of life-cycle costs are committed. The difficulties are that early, conceptual design is characterized by a low level of system definition and that, at this stage of definition, the relationships among design objectives and the conceptual design parameters are often not well modeled or understood. To improve results during this design phase, at least two emphases must be pursued: 1) improvement of disciplinary analysis modeling and tools (engineering codes) that capture, with sufficient fidelity, the major relationships among design variables and system objectives; and 2) development of methods for coordinating the engineering analyses and optimizing the total launch vehicle system.[1] This second objective is significantly expanded by the application of multidisciplinary design optimization (MDO). When compared to the typical one-variable-at-a-time (OVAT) trade study process, MDO offers both improved understanding for complex engineering problems, by exploiting interactions among the design disciplines, and higher productivity, because feasible solutions are identified more quickly.

Significant progress has been made in the past decade in both of these areas for space transportation conceptual design. The sections that follow first give an overview of the principal disciplines involved and those areas of greatest needs for modeling. Next, a brief history is given of the vehicle synthesis and optimization frameworks within which these disciplines have been incorporated. Finally, a discussion of optimization methods and examples of their application will be presented.

Space Transportation Conceptual Design Problem

Many interesting design problems can be described in the broad context of space transportation. Two examples demonstrating the breadth of design and optimization objectives might be the design of a low-cost, reusable, ETO launch vehicle or a highly reliable Mars lander. The first, because of its reusability and cost objectives, would need adequate models to produce performance, operations, and cost estimates from vehicle design features. The second, because of the precision interplanetary and atmospheric flight requirements, would need models to produce performance and reliability estimates from vehicle design characteristics. While the disciplinary modeling needs of these two problems might be different, they have similarities in their requirements for a diverse set of optimization methods. In this paper, only the launch vehicle design process will be discussed. This example is used to illustrate the diversity of design disciplines involved, the characteristics of the associated engineering analysis codes, and their implications for the application of MDO methods.

To distinguish, in a simplistic way, launch vehicle design issues from aircraft design issues, several observations can be made. Commercial airlines, the largest customer for aircraft manufacturers, are critically concerned with operating cost, and, to a lesser degree, production costs. New fleet introductions build on a wealth of design heritage, and design decisions

Received Oct. 7, 1997; revision received April 1, 1998; accepted for publication April 28, 1998.

*Senior Researcher, Vehicle Analysis Branch. Senior Member AIAA.

†Assistant Professor, School of Aerospace Engineering. Senior Member AIAA.

‡Associate Professor, Engineering Management Department. Member AIAA.

are based on business models reflecting production runs of hundreds of aircraft to estimate passenger ticket prices. The aircraft performance and configuration are optimized to expected cruise conditions, and the flight environment is well understood.

While current government investments in launch vehicle design and technology are intended to enable a viable, reusable, launch vehicle business, such a business does not yet exist, and the production of future launch vehicles will continue to be small, on the order of tens rather than hundreds, for the next decade or two while new industries develop around the decreasing costs of space launch. Thus, each new introduction typically has little design heritage, incorporates first-time flight technologies, and has as its customer a NASA or Department of Defense customer focused on specific mission performance rather than a profit-loss business model. With performance as the goal, rather than a low per-seat ticket price, design and trade studies have, in the past, optimized toward production costs and weight, and, to a much lesser degree, operations costs. Reusable vehicles will encounter wide loading and environmental conditions, from the launch and high dynamic pressure ascent to the entry and high heating conditions in hypersonic flight through supersonic, transonic, and subsonic speeds. The aerothermodynamics of hypersonic flight are still the subject of experimental testing, making it difficult to accurately tailor entry configuration and trajectories to control heating while still providing adequate range and cross-range margin for acquiring the landing site. It is against this backdrop that the disciplinary analysis models for launch vehicle design have evolved and have been incorporated into the conceptual design process.

At the beginning of conceptual design, often only the mission requirements are known, but, in some cases, additional information regarding vehicle concept, operational approach, and subsystem technologies may also be available. In Fig. 1, the ovals on the left represent design decisions, or options, to be evaluated, and the rectangles represent the analyses that might be conducted, depending on the specific issues being addressed. This simplified notion of the conceptual design process for launch vehicles illustrates the variety of disciplines (subproblems) making up the system-level design problem. The process includes the following.

1) Specification of the mission requirements, e.g., payload size, mass, destination, environmental constraints, on-orbit operations.

2) Selection of a vehicle approach, e.g., rocket or air-breather, winged or ballistic, piloted or automated, single or multiple stages, expendable or reusable.

3) Selection of associated operational scenarios, e.g., assembly, launch, recovery, refurbishment.

4) Selection of technologies, e.g., structural materials, thermal protection system, avionics, propulsion.

5) Creation of a physical layout and surface geometry that will contain the payload, subsystems, and airborne support equipment.

6) Estimation of the ascent and entry aerodynamics, e.g., subsonic, transonic, supersonic, hypersonic.

7) Calculation of trajectories and the resultant flight environments.

8) Execution of structural, controls, heating, and propulsion analyses based on the flight environment.

9) Estimation of the vehicle weights, dimensions, and center of gravity based on layout, environment, and technology selection.

10) Analysis of operations and maintainability requirements based on operational scenario, vehicle configuration, and technologies.

11) Estimation of life-cycle costs, e.g., design, development, test, evaluation, production, operations, disposal.

12) Feedback of these results for optimization and modification of the overall system to meet mission requirements and design objectives.

The conceptual design process is highly coupled (nonhierarchical), and significant data exchange and iteration are often required among disciplines and disciplinary tools. Thus, the number of design and coupling variables present in the full problem can be prohibitively large given current computing limitations for analysis and optimization. Traditionally, the vehicle design process of Fig. 1 is decoupled into three smaller, more manageable, multidisciplinary problems (Table 1 footnote). Two problems address vehicle performance: 1) an ascent problem primarily involving trajectory, weights and sizing, and propulsion analyses; and 2) an entry problem emphasizing geometry, aerodynamics, trajectory, heating, structures, and controls. Once the configuration is designed to meet both ascent and entry performance requirements, a third problem, referred to here as the economics problem, may bring the more empirical disciplines, such as operations and cost analyses, into the

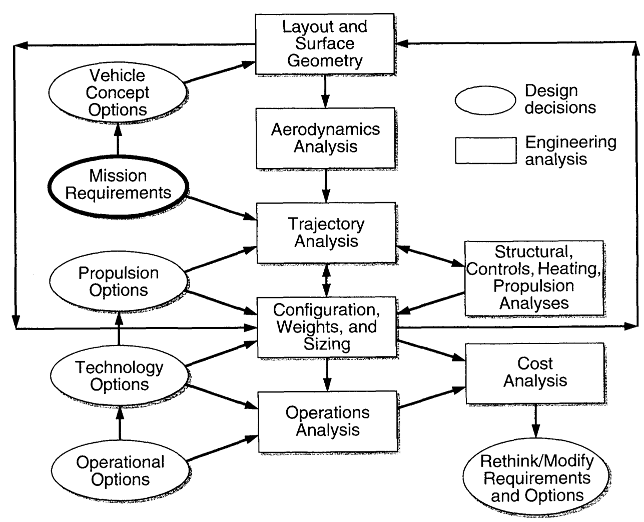

Fig. 1 Launch vehicle conceptual design process.

Table 1 Characteristics of representative launch vehicle conceptual design codes

Discipline[a]	Variable/constraint number and type	Code type
Geometry/packaging	Few-to-many variables, continuous/discrete	Interactive, graphical, commercial
Aerodynamics	Few variables, continuous	Interactive
Trajectory	Many, continuous, dense/sparse	Design-oriented
Weights and sizing	Few-to-many variables, continuous/discrete	Design-oriented
Heating	Few, continuous/discrete	Interactive
Structures	Many, continuous/discrete	Batch, commercial
Controls	Many, continuous	Batch, commercial
Propulsion	Few, continuous/discrete	Design-oriented, regression equations
Operations	Many, discrete	Highly interactive, regression equations
Cost	Many, continuous/discrete	Highly interactive, regression equations

[a]Multidisciplinary problems: ascent problem = trajectory, weights and sizing, propulsion (plus aero and heating if airbreathers). Entry problem = geometry, aero, trajectory, heating, structures, and controls. Economics problem = operations, costs (plus selected other disciplines).

design problem. In the past, optimization studies within any of these three simplified problems have used a selected, manageable subset of the disciplines for application of MDO methods.

Modeling Needs

Physics-based engineering models, such as those used for aerodynamics, structures, and trajectory analysis, have been employed and validated over decades and can provide a level of fidelity limited primarily by the computational power and design time available. As computing speed increases, conceptual designers have begun to bring many of these higher-fidelity, physics-based tools forward to the very early stages of design. However, state-of-the-art modeling in several other launch vehicle analysis areas falls short of sufficiently defining the dependencies between design objectives and conceptual design variables.[1] For example, weights and sizing, operations, and cost analyses are empirical tools heavily dependent on extrapolation from sparse databases of previous vehicle developments. For vehicle concepts significantly different from past developments or for advanced technologies where there has been no hardware development, such data do not exist. However, these disciplines are critical to developing life-cycle cost-estimation capabilities. As a result, efforts are underway to develop innovative modeling approaches that improve the state of the art in these areas.

Weights and sizing estimation are critical to almost every aspect of the vehicle design problem; yet, for the conceptual design of complex systems such as ETO vehicles, it is fraught with uncertainty. Statistical approaches, e.g., Monte Carlo simulation, that allow these uncertainties to be modeled are being developed and evaluated for application to launch vehicle design.[2] Using this approach, weight estimates can be produced for various probability distributions and for a specified percentile of cumulative probability. Thus, competing vehicle designs can be compared on the basis of weight risk rather than on simple point estimates of weight. Information of this form is a more useful input to follow-on cost analyses than are point-design weight estimates. Unfortunately, probabilistic uncertainty distributions of component-level weight are difficult to determine and validate.

Operational support requirements estimation, similarly characterized by uncertainty, has traditionally been data intensive and required substantial design information. New models are being developed[3] for conceptual-level design studies by linking support requirements to the concept through its reliability and maintainability (R&M) characteristics. These support requirements are estimated based on analysis of comparability of the new concept's systems and technologies to known aircraft and Space Shuttle systems. Having estimated R&M requirements, discrete event simulations are then developed to model proposed operational scenarios and assess the impact of various resource constraints on flight rates, facility usage, and other resource metrics. Minimizing these resources using the discrete event simulation model is a difficult optimization problem involving discrete variables and a simulation domain that does not allow explicit calculation of the objective function nor application of constraints, thus rendering common methods inadequate.

A third area of considerable uncertainty, cost estimation, has not historically been applied until the end of the serial design process, after configuration and technology decisions are made. However, to achieve cost-effective designs, cost analyses must be incorporated into the design process directly. Efforts to do so[4,5] show promise. Currently, cost-estimation tools are in hand for design, development, test, and evaluation (DDT&E) costs, but other aspects of costs, such as operations, manufacturability, or advanced technology development, are not well modeled. Research[6] into life-cycle cost models, as well as cost and revenue business modeling using discounted cash-flow techniques, has been initiated. Initial results indicate significant insight can be gained by integrating life-cycle cost analysis into the design process.

These empirical models, though showing promise, must be treated with caution. While demonstrations of optimization approaches incorporating these models into the design process alongside traditional engineering models are being pursued with good success, the bottom line is: if you can't model it, you can't optimize it.

Computational Frameworks

To perform multidisciplinary vehicle synthesis and optimization on conceptual space transportation systems, designers rely on computer-based simulation and analysis tools. The characteristics of the individual engineering models are diverse, requiring a variety of optimization approaches capable of treating discrete or continuous design variables, few or many coupling variables and constraints, single solution or multiple minima, and simple or computationally intensive analyses. The number and characteristics of the analysis tools employed determine whether this synthesis and optimization process will be conducted manually by a team of disciplinary experts or automatically via a computational framework. For many complex problems, such automation (and the formulation of the problem[7,8]) can significantly reduce design time and ensure data consistency between individual disciplines. Incorporation of engineering models into an automated computational framework typically takes one of two forms. In one form, a stand-alone, monolithic, synthesis tool, i.e., compiled as a single executable code, is developed that contains internal modules or subroutines to accomplish each disciplinary analysis. In the second form, each discipline is represented by a separate code or codes that are loosely integrated into a framework for vehicle synthesis and optimization.

The monolithic approach is typically self-contained and is run on a single computing platform by a single user who is reasonably knowledgeable in each analysis discipline included. Data exchange between modules is handled internally. To prevent the code from becoming unwieldy, high-fidelity, disciplinary analyses are often replaced with smaller and faster approximate methods. For example, a complete finite element analysis of wing loads and weights could be replaced with a simple table lookup, a response surface equation fit to historical data, or computational data generated off-line. The notion of using off-line data generated by detailed analysis tools (and disciplinary experts) to periodically recalibrate a monolithic tool's approximate model has been suggested. Such a technique is often called variable complexity.[9]

ODIN,[10] AVID,[11] and PICTOS[12] are three early representatives of monolithic synthesis tools for launch vehicles. While having the advantage of being fast and executable by a single designer, monolithic synthesis codes tend to exclude the disciplinary experts from the design process and can quickly become outdated without continued support and improvement. In addition, these systems can become difficult to extend or modify. More recent examples of monolithic launch vehicle synthesis codes receiving continued development include a code for hypersonic airbreathing transatmospheric launch vehicles, HOLIST.[13,14] HOLIST is derived from previous monolithic synthesis tools, but adds improved engine, weight, and aerodynamic prediction capabilities as well as an improved user interface. FASTPASS,[15] a synthesis tool for advanced launch systems, includes a three-degree-of-freedom trajectory optimization module and a flexible propulsion module. The hypersonic vehicle optimization code (HAVOC) synthesis code[16] contains internal modules for structures, performance, aerodynamics, propulsion, and life-cycle costing. Another example is the integration of POST (a widely used trajectory optimization tool) and CONSIZ (a weight-estimation tool). The integrated POST–CONSIZ tool has been used for direct optimization of vehicle mass properties, e.g., minimum dry weight, and, with the additional inclusion of a cost model, the vehicle development costs.[4]

The second computational framework approach maintains each discipline as a separate code or codes. This form closely resembles the environment typical of many advanced design organizations. These stand-alone tools develop and evolve naturally within their discipline, and each discipline maintains an expertise in their operation. Such tools typically reflect a level of analysis fidelity exceeding the approximations found in monolithic synthesis codes and are often operated independently by disciplinary experts for single-discipline analysis. In this approach, the computational framework automates the task of executing the contributing codes and exchanging coupling data until the design is converged. The disciplinary experts remain involved to set up and validate initial input files for their disciplinary code, establish ranges of acceptability on key input variables, and suggest alternative solutions for their discipline. However, within bounds specified by the disciplinary experts, the computational framework automates the process of entering input data, executing each of the contributing codes, extracting the required output data, and passing coupling variables on to the next code in the process.

Automated design frameworks are under development that support loose coupling of codes and retain the high fidelity and collaborative nature of this design environment, while potentially introducing the speed, data handling, and automated execution advantages of a monolithic synthesis code. Encouraging progress in the development of such generic, automated, computing frameworks for conceptual design, e.g., FIDO,[17] IMAGE,[18] has heightened the interest in a design framework specifically for conceptual launch vehicle design. Commercial frameworks, such as Engineous' iSight,[19] are offering generic integration utilities to allow easier integration of diverse, stand-alone codes with a choice of optimizers.

This approach has a number of advantages relative to a monolithic synthesis code, such as collaborative design, computing platform and geographical location independence, higher-fidelity vehicle solutions, and involvement of the disciplinary experts. However, execution can be slower and setup and checkout may be tedious. In addition, the contributing disciplinary tools, unless developed to be design oriented, may be highly user-interactive, nonrobust, poorly suited to the calculation of gradient information, or of commercial origin where no source code is available for local modification (see Table 1). Automated execution of such codes requires significant pre- and postprocessing capability.

Both of the computational frameworks discussed in the previous text automate the process of exchanging data and ensuring consistency between various engineering disciplines, and this has enabled application of optimization methods to system-level design. By combining the optimization and synthesis capabilities, designers can quickly explore a design space, evaluate numerous design alternatives, and determine the best design to pursue. New MDO methods are being developed[20–23] that are specifically tailored to complex problems involving several engineering disciplines, and, as discussed next, some of these methods make use of the frameworks just described.

Optimization Methods

The optimization methods discussed here may be divided into three main groups: 1) parameter methods based on design of experiments (DOE) techniques, 2) gradient or calculus-based methods, and 3) stochastic methods such as genetic algorithms and simulated annealing. In this section, a summary is given of successful applications of various methods to one of the three decoupled multidisciplinary launch vehicle design problems identified in Table 1. In many of these cases, the test problems have been solved by more than one MDO method to compare accuracy of the solution, computational resources required, and level of human involvement for setup and execution. Discussion of the detailed problem statements and results and their underlying methods, too lengthy to be incorporated here, can be found in the individual study papers referenced.

Parameter Methods

The engineering codes used in launch vehicle design are typically stand-alone, interactive analyses run by disciplinary experts. Therefore, one approach to system optimization across disciplines is to use methods that do not require software integration or automation. This consideration, and a need to accommodate both discrete and continuous design variables, led to application of parameter methods, including response surface methods, to build polynomial approximation models representing the relationships between system characteristics and design variables.[24–29] These approximation models are then used for MDO and sensitivity analysis.

In the response surface method approach, disciplinary experts, each using their independent codes, produce the disciplinary–feasible design solutions at several statistically selected combinations of design variables within the design space, and a polynomial surface is fit to the results. The design points are usually selected by DOE methods such as Taguchi's orthogonal arrays, central composite designs, and saturated designs. Data generated by the individual engineering codes are collected from the experimental cases and then input to a utility program that calculates the surface approximation equation. Numerical optimization, usually a gradient method, is then performed on these approximation surfaces. The solution point can be some combination of values of the design variables not studied among the originally selected point designs.

To use a parameter method, it is not necessary to have an automated computing framework or monolithic synthesis tool. For a small number of variables (less than 10), the required number of overall vehicle point designs is often small, and these designs may be generated by traditional, manual iteration among disciplines and used to generate the vehicle-level response surface equation. An alternate approach is to generate, in parallel, a response surface for each discipline in the problem. The resulting set of discipline-level response surfaces can be implemented as the representative disciplinary modules of a monolithic synthesis tool and used in subsequent convergence and optimization studies.[30]

Some advantages of parameter methods are 1) disciplinary analysis integration is not required (so interactive codes present no special problem), 2) discrete and continuous design variables may be accommodated, 3) sensitivity information over the entire design space may be inferred from the approximation surface, and 4) constraints can be modified without running additional cases. Also, once the response surface model is validated, the approximation surface can provide a surrogate model for coupling with other applications. Disadvantages of this approach are 1) approximations to some systems may be poor, 2) the approximations yield only a near-optimal solution, 3) as the number of variables grows, this approach becomes unmanageable, and 4) human involvement in selecting and running the experimental cases is greater than for an existing monolithic synthesis code.

Figure 2 shows a conical, vertical takeoff, horizontal landing, single-stage-to-orbit (SSTO) vehicle concept designed to deliver 10,000 lb to a polar, low-Earth orbit.[25] Propulsion is provided by ejector scramjet, rocket-based, combined-cycle (RBCC) engines. Table 2 gives results of a design study performed using both the central composite design and the Taguchi robust design approaches. The ascent problem for this vehicle involved five, nonhierarchically coupled engineering disciplines: trajectory, weights and sizing, heating, aerodynamics, and propulsion. Each discipline was represented by a separate, stand-alone analysis code. The required vehicle-level point designs were converged manually among the codes. No automated framework was used. The study objective was twofold. First, determine the design variable values that resulted

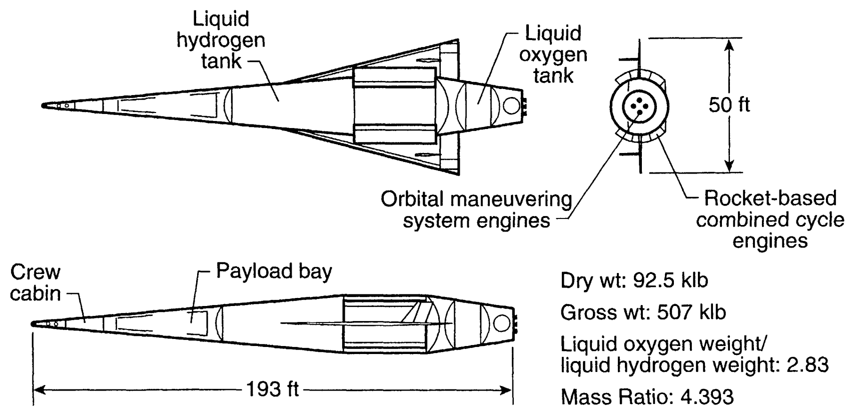

Fig. 2 Rocket-based, combined cycle, SSTO vehicle concept.

Table 2 Rocket-based, combined cycle study results using parameter methods

Design variable	Response surface optimum	Robust design
T/W_0	1.27	1.2
Rocket mode transition Mach number	14.6	12
Cowl wraparound angle	180 deg	180 deg
Predicted dry weight (second-order model)	89,660 lb	N/A
Verification dry weight (point design)	91,578 lb	92,498 lb

in the lowest dry-weight vehicle. Second, given uncertainties in the expected engine weight, airframe weight, and scramjet engine specific impulse (I_{sp}), determine the design variable values that resulted in a near-optimal, robust design. Here, robustness refers to a concept that is insensitive to adverse changes in the uncertainty variables.

Given the three, system-level design variables of Table 2 and three uncertainty variables, 39 different vehicle point designs (forming a central composite design experimental array) were generated and used to fit a 22-term, second-order, response surface of the design space as a function of the design and noise variables. The design variable values that minimize dry weight are shown in the center column of Table 2. In this case, noise variables were set to their nominal values and bounds were imposed on the design variables. Initial thrust-to-weight ratio (T/W_0) was limited to 1.2–1.4, rocket-mode transition Mach number to 12–18, and engine cowl wraparound angle (the circumferential angle of the annular cowl as viewed from the nose) to 180–360 deg.

The results of the Taguchi method/robust design exercise are listed in the right column of Table 2. The three design variables were discretized to three evenly spaced values and placed in an eight-row, orthogonal array (L_8), and the three noise (uncertainty) variables were each discretized to two values (nominal and degraded) and placed in a four-row, orthogonal array (L_4). The intersecting 32 point designs were used to construct signal-to-noise ratios (SNRs) for each combination of discretized design variable values. Maximizing Taguchi's SNR is equivalent to finding a design that nearly minimizes dry weight while not being overly sensitive to adverse changes in the noise variables. Note that, compared with the optimum design, the robust design reduces vehicle takeoff T/W_0 to 1.2 (thereby reducing engine weight and sensitivity to engine weight growth), and reduces the rocket-mode transition Mach number to 12 (thereby reducing the sensitivity to possible scramjet I_{sp} degradation). With these changes, the robust design is just 920 lb heavier than the optimum design.

The primary significance of this example lies in the fact that, even in the absence of an automated computing framework to integrate the five disciplinary tools, a substantial amount of information was able to be derived from a carefully chosen set of point designs using parameter methods (including a dry weight minimum solution and a robust design solution). Such a technique for MDO could be introduced into a wide range of existing design environments with little or no cultural change in the way vehicle point designs are determined.

Several other researchers have reported success in applying parameter MDO methods to space vehicle design problems. Among them, Bush and Unal[26] used Taguchi methods on the design of a lunar aerobrake with discrete variables; Engelund et al.[27] used second-order response surface methods for optimizing the aerodynamic configuration of an all-rocket SSTO; Unal et al.[28] used saturated experimental arrays (D-optimal) to optimize the dry weight of a dual-fuel, rocket-powered, SSTO launch vehicle; and Galati and Elkins[29] used Taguchi methods to examine thrust augmentation for SSTO launch vehicles.

While the parameter approach is a significant advance over OVAT analyses, there are design and analysis problems where the disciplinary analyses have already been or can be readily integrated into a monolithic synthesis code without requiring approximation. In these cases, traditional gradient methods for numerical optimization can be more efficient in terms of both human and computational resources.

Gradient Methods

Gradient, or calculus-based, numerical optimization methods are widely used in individual disciplinary analyses, such as trajectory or structural analysis programs, and in multidisciplinary monolithic synthesis codes where several disciplines are represented within a single program. Gradient methods calculate derivatives of the objective function with respect to design variables, either analytically or by finite differences, to find a path to the minimum solution. Because gradient methods typically require many function calls, i.e., vehicle point designs, to determine a vehicle-level optimum, they are often paired with a fast, automated computing framework for determining converged vehicle information. In most cases, the popular choice has been a monolithic synthesis code.

Advantages of this approach are 1) a mathematically rigorous optimal solution may be found; 2) large numbers of variables and constraints can be handled; 3) a consistent vehicle model across disciplines may be guaranteed; and 4) once developed, these codes can be executed with little human involvement while they run to completion.

Disadvantages of this approach are 1) these methods do not accommodate discrete variables or nonsmooth design spaces because discontinuous derivatives result; 2) sensitivity information is only known along the solution path, rather than across the design space; 3) the sheer size of these codes may make them impractical as surrogate models for other applications; 4) substantial human effort is often required to merge disciplines into a single code; and 5) many disciplines, because of their complexity or interactive nature, do not lend themselves to integration (see the discussion of system sensitivity analysis later in this section). The last three disadvantages are related more to the framework needed for the gradient method than to the method itself.

Figure 3 shows results of a study[31] to minimize total vehicle dry weight of an SSTO, rocket vehicle by varying 40 continuous design variables (shown next), defining vehicle configuration, propulsion system, and trajectory profile while meeting 13 constraints on trajectory, vehicle, and propulsion configuration features. Design variables = 1) gross liftoff weight (GLOW), lb; 2) S_{ref}, ft^2; 3) landed weight, lb; 4) base diameter, ft; 5) launch azimuth, deg; 6) vehicle liftoff T/W; 7) % liquid hydrogen, mode 1; 8) mode 2 mixture ratio; 9) nozzle area ratio, retracted; 10) nozzle area ratio, extended; 11) propulsion mode transition Mach number; 12) nozzle transition Mach number; 13) F_z boundary duration, s; 14) total trajectory time, s; and 15–40) set of pitch angles, deg. Constraints = 1) terminal altitude, ft; 2) terminal velocity, ft/s; 3) terminal flight path angle, deg; 4) terminal inclination, deg; 5) dynamic pressure, lb/ft^2; 6) F_z magnitude, lb; 7) nozzle extension ratio; 8) angle of attack, deg; 9) pitch rate, deg/s; 10) GLOW compatibility, lb; 11) S_{ref} compatibility, ft^2; 12) landed weight compatibility, lb; and 13) base diameter compatibility, ft. This problem was implemented in the POST–CONSIZ synthesis code using several gradient-based optimization approaches. While some implementation strategies showed better computational performance than others, each produced fast, accurate, hands-off results and achieved feasible designs while reducing the dry weight relative to a baseline solution obtained previously by a response surface method approach. Similar results have been presented in Refs. 32–34.

Other examples of successful application of the gradient method to the space transportation problem include Ref. 4, where the POST–CONSIZ synthesis code for solution of the ascent problem was extended to an economics problem by also incorporating DDT&E costs. This work was extended still further in Ref. 5, where the objective of the study was to minimize total vehicle development cost, taking into consideration the costs and savings potential of an advanced technology maturation program addressing nine core technologies. In this implementation, the representative DDT&E equations were replaced with a commercial cost-estimating model running on a separate platform and communicating over an Internet link. Figure 4 shows how a given technology budget would be allocated to bring all technologies to the same technology readiness level (TRL) of 6 (TRL ranges from 1 for an idea to 9 for flight-proven hardware). Figure 5 shows the optimized allocation of technology funding which, while equal to the funding of Fig. 4, results in a 15% reduction in vehicle development costs. This is because some subsystems, e.g., the thermal protection system, have a relatively higher contribution to total development cost and also have a higher development cost sensitivity to TRL.

System Sensitivity Analysis

For reasons of execution speed, gradient methods were characterized earlier as best suited for use with a monolithic synthesis code; however, a subclass of gradient methods known as system sensitivity analysis (SSA) does not require explicit analysis integration. Instead, its characteristics lend themselves well to loosely integrated computing frameworks. For highly coupled design problems, convergence of a design traditionally requires time-consuming internal iteration among disciplines. To reduce optimization time, SSA[35] takes advantage of the implicit function theorem to decompose the larger problem into a set of parallel disciplines for the calculation of gradients. Each discipline simultaneously calculates its local derivatives with respect to design variables and interdisciplinary coupling variables. Local derivatives from all disciplines are combined into a single, linear matrix equation (the global sensitivity equation), which may be solved to simultaneously generate all total derivatives for a numerical optimizer. As a result, no iteration between disciplines is required during gradient calculation. Because the disciplines determine their local derivatives independently and in parallel, this method is also well suited to a design environment in which the design tools have not yet been integrated into a computational framework. Application of the SSA method to the dual-fuel, rocket, SSTO vehicle of Fig. 3 is given in Ref. 36. In this example, no computing framework was used, and each analysis was executed manually in a stand-alone environment. The optimization results compared well with those obtained with the integrated POST–CONSIZ code.[31]

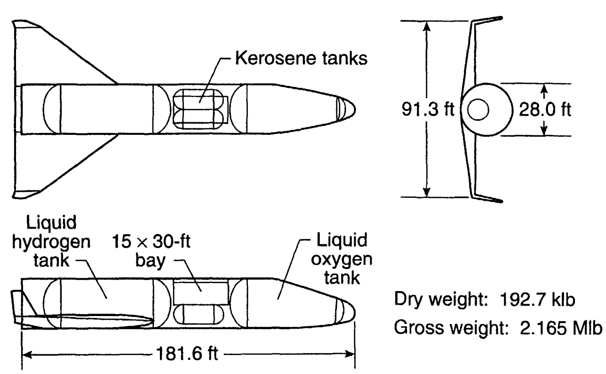

Fig. 3 Rocket, SSTO vehicle concept.

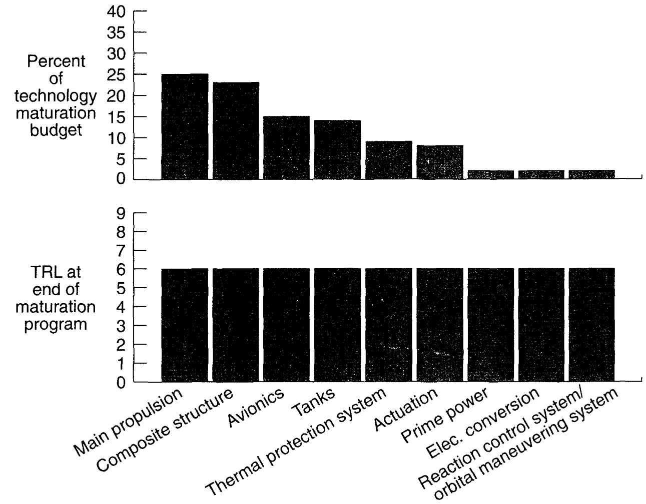

Fig. 4 Technology investment strategy when all technologies matured to technology readiness level 6.

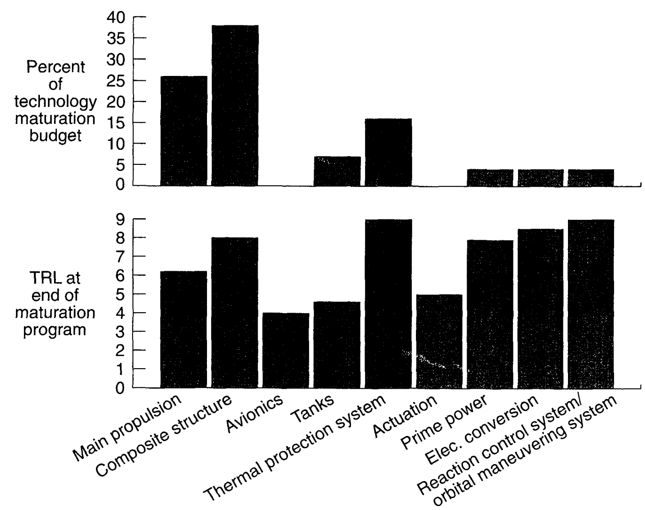

Fig. 5 Technology investment strategy optimized by gradient methods.

Advantages of the SSA method are 1) time-saving parallel execution of disciplines during gradient calculation, 2) time savings caused by the reuse of many of the local sensitivities between optimization steps, and 3) suitability to analysis environments lacking a monolithic synthesis code. Disadvantages include 1) additional complexity introduced by a matrix solution; 2) possible expansion of individual subproblems because of the large numbers of coupling variables; and 3) full analysis (with probable iteration) is still required at fixed point, nongradient solutions.

While the methods described thus far have proven advantageous to conceptual studies, these techniques are not applicable to some classes of problems. For example, optimization of interplanetary trajectories is difficult because the design space is discontinuous with localized minima. Conventional calculus-based and response surface methods are not effective in such domains, and, typically, exhaustive grid search methods would be employed. For domains of this type, or for problems involving discrete design variables, stochastic methods, such as genetic algorithms[37] or simulated annealing,[38] can be used more efficiently to span the design space.

Stochastic Methods

One of the better-known stochastic methods is genetic algorithms (GAs), which are designed to mimic evolutionary selection. Each individual design candidate is represented by a string that is a coded listing of design parameter values. The string is analogous to a chromosome with genes for the various parameters. The objective function is evaluated for an initial population of candidates, and these candidates compete to contribute new members to future populations.

The advantages of GAs are 1) discontinuous problems or discrete variables present no special problem to the method; 2) setup and execution are not difficult; and 3) only analysis function evaluation, not gradients, is required. The disadvantages are 1) a large number of function evaluations are typically required, 2) only near-optimum solutions are found, 3) no sensitivity information is developed, and 4) to be practical all disciplines must be integrated into a single code.

Examples of GA methods applied to space transportation include disciplinary optimization studies of interplanetary trajectories[39] and operation simulation models.[40] Operation simulation models[3] cannot be optimized in the traditional sense because the simulation domain prevents explicit problem formulation in terms of objectives and constraints, thus rendering common mathematical optimization methods inapplicable. However, a GA-based optimization methodology has been developed for use in conjunction with discrete event simulation models with stochastic components[41] to minimize fleet size and maintenance crew sizes for an operations scenario. In this study, the GA-based simulation optimization methodology produced a solution resulting in a 23% reduction of operations and support resources over the previous OVAT design approach.

GAs have also been applied to MDO studies characterized by discrete design parameters. In Ref. 42, the disciplines of cost and weights and sizing were integrated to evaluate the GA method for solution of a vehicle design problem characterized by discrete design parameters. Optimization was performed toward two objectives: minimum dry weight and minimum cost, by varying material type (three discrete values) for seven major vehicle structural components. Vehicle components = 1) hydrogen tank (L_{h_2}/tank), 2) liquid oxygen tank (L_{ox}/tank), 3) hydrocarbon fuel tank (L_{hc}/tank), 4) wing section structure (wing), 5) wing tip-fin section structure (tip-fin), 6) basic structure (basic), and 7) secondary structure (second). Material types = 1) aluminum (Al), 2) aluminum–lithium (Al–Li), and 3) composite material (comp.). The cost-estimating relations model the DDT&E costs as a function of both weight and technology readiness. Figure 6 shows the results from the minimum cost optimization as well as a comparison of weight and cost factors for both optimizations. One significant observation is that the minimum weight vehicle is often not the minimum cost vehicle, which is reasonable when the cost of technology maturation is considered.

Minimum Cost Selections

Lh2/Tnk	Lhc/Tnk	Lox/Tnk	Wing	Tip-fin	Basic	Second
Comp	Comp	Comp	Comp	Al	Al	Al-Li

Vehicle Cost and Weight Factors

Cost	Vehicle Weight, lb.
0.794	220334
0.795	185275

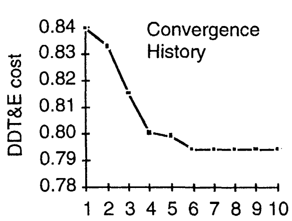

Fig. 6 Rocket vehicle study results using GA methods.

The simulated annealing (SA) algorithm, like GAs, employs randomized search techniques to solve multivariate optimization problems. An analogy to the statistical mechanics of annealing of solids provides the strategy for optimization.[38] SA is a general optimization tool that can be used to solve problems involving both continuous and discrete variables. Like GA, SA requires a large number of function evaluations to reach an optimal solution.[43] Research is being conducted to parallelize SA to accelerate the convergence of the algorithm to the global optima.[44] SA algorithms have advantages and disadvantages similar to those listed for GAs.

SA algorithms have been applied to parametric design of aircraft.[45] Using an iterated SA algorithm, eight design parameters were varied to minimize system weight, producing answers comparable with the other search methods studied.

Having briefly touched on a variety of optimization methods used for launch vehicle design, it is clear that the coordination of the many required disciplines, and the goal to bring the ascent, entry, and economics problems together, will require an efficient, flexible strategy that can accommodate many optimization methods and models. Research[46] has been conducted on a practical strategy for optimization that may allow the best features of the previous approaches to synergistically coexist.

Collaborative Optimization

In each of the approaches stated earlier, the strategy for communication and control among the optimizer and disciplinary codes was briefly stated. A developing strategy,[46,47] termed collaborative optimization, allows disciplinary codes to be executed in parallel, each with control of its own design variables and satisfying its own constraints. In this approach, a system-level optimizer minimizes the overall objective and coordinates the subproblem optimizers through negotiated agreement on coupling variables. This strategy models the practical aspects often employed by design teams, but has improved communications and coordination constructs. Collaborative optimization is a design architecture specifically created for large-scale, distributed-analysis applications. This decentralized design strategy allows domain-specific issues to be accommodated by disciplinary analysts, while requiring interdisciplinary decisions to be reached by consensus.

Table 3 Rocket vehicle study design variables and constraints using collaborative optimization

Optimization level	Design variables	Nonlinear constraints
System	23	3
Trajectory subspace	47	9
Propulsion subspace	4	1
Weights, sizing, cost subspace	21	3

Table 4 Comparison of three multidisciplinary optimization strategies for launch vehicle design

Optimization architecture	Function evaluations	Modification time, month	Communication requirements
Iterative loop gradient	10,482	4	66
Compatibility constraint gradient	3182	3	65
Collaborative	3125–24,840	1	23

Advantages of the architecture are that it 1) may not require either modification of codes or explicit integration into an automated computing framework, 2) allows subproblems to be optimized by the best-suited method, 3) allows for the addition or modification of subproblems, and 4) can efficiently accommodate a large number of variables. Consideration should be given to the fact that, as the number of disciplinary-specific design variables increases and the relative interdisciplinary coupling decreases, the performance of this approach improves.

Reference 48 describes an application of the collaborative method to an ascent/economics problem similar to the gradient application of Ref. 4. The design problem, characterized by 95 design variables and 16 constraints, is well posed for the collaborative architecture. Table 3 shows the partitioning of the design variables and constraints at the system level and among the four disciplines of trajectory, propulsion, weights and sizing, and DDT&E cost. Table 4 shows the performance comparison between the gradient[31] and collaborative[48] methods for this design problem. Further experience with this method is required to more fully understand its potential applications.

Vision for Space Transportation Conceptual Design

Traditionally, the vehicle design process of Fig. 1 has been decoupled into three smaller, more manageable multidisciplinary problems as noted earlier (Table 1 footnote). The long-range goals of MDO research for launch vehicle design are, first, to couple all of the disciplines within each of the three smaller problems and then to coordinate the three problems into a single system optimization problem. Many activities are underway in both government and industry to produce such a capability for life-cycle cost estimates at the conceptual design phase. If this design loop can be closed, then not only can traditional design-for-performance objectives be optimized, but also methods for cost-optimized design may evolve. At least two major emphases must be pursued to achieve this goal: 1) improvement of the fidelity of conceptual design models and 2) development of computational frameworks and optimization methods that handle the diverse characteristics of the disciplinary subproblems and effectively exploit synergies among the disciplines to improve design results. Significant among the modeling improvements must be the treatment of risk and uncertainty in each of the disciplines. Modeling improvements will be driven by research scattered throughout the transportation design community. Methods research is being conducted at a steady pace in academia, industry, and government.[23]

In the near term, designers will continue to make improvements to monolithic synthesis tools that will improve their capabilities and depth of analysis detail. Similarly, research into loosely integrated computing frameworks will likely demonstrate that high-fidelity, stand-alone, disciplinary codes can be integrated into an automated design environment.[46] Thus, the ultimate goal of a high-fidelity, collaborative conceptual design environment that is fast and highly automated will be approached from two directions.

In the far term, advances in computing speed, network bandwidth, data visualization, optimization methods, computing frameworks, and disciplinary models will likely enable virtual design environments, in which widely distributed members of a conceptual design team are able to quickly share and use high-fidelity data from a variety of sources. Such a capability will allow vehicle designers to rapidly consider a number of fully converged design alternatives in a very short time without sacrificing design detail, thus improving the quality of conceptual space vehicle design.

Summary

Application of several MDO methods to aspects of the launch vehicle conceptual design problem have demonstrated significant improvements in study results and human productivity. Even considering the diverse characteristics of these multidisciplinary design problems, whether involving continuous, discrete, or large numbers of design variables, most can be addressed by one or more existing MDO methods. In addition, progress in the development of automated computational frameworks offers the potential for efficiently coordinating disciplines in a large-scale, multidisciplinary design environment.

Launch vehicle analyses based on empirical models, such as operations, weights and sizing, and costs, are not as mature as models of the traditional engineering disciplines, and thus, must see continued development and validation to approach similar levels of credibility. Demonstration applications using these empirical models in the design process together with traditional engineering disciplines have been successful.

An important goal of the aerospace community is to develop the capability to design and optimize systems for life-cycle cost estimation. Continued work is required both in the development of systems analysis models and in the application of optimization methods and frameworks to achieve these goals.

References

[1]Rowell, L. F., "Design Challenges for Tomorrow's Manned Launch Systems," AIAA Paper 93-1068, Feb. 1993.

[2]Monroe, R. W., Lepsch, R. A., and Unal, R., "Weight Estimating Risk Analysis: Incorporating Uncertainty at Conceptual Design for a Launch Vehicle," *Proceedings of the 1995 ASME National Conference* (Arlington, VA), American Society of Mechanical Engineers, New York, 1995, pp. 25–32.

[3]Morris, W. D., White, N. H., Davis, W. T., and Ebeling, C. E., "Defining Support Requirements During Conceptual Design of Reusable Launch Vehicles," AIAA Paper 95-3619, Sept. 1995.

[4]Moore, A. A., Braun, R. D., and Powell, R. W., "The Infusion of Cost into the Multidisciplinary Design of Space Transportation Systems," *Proceedings of the 17th Annual Conference of the International Society of Parametric Analysts*, ISPA, Washington, DC, 1995, pp. 164–182.

[5]Moore, A. A., Braun, R. D., Powell, R. W., and Qualls, G. D., "Determination of Optimal Launch Vehicle Technology Investment Strategies During Conceptual Design," AIAA Paper 96-4091, Sept. 1996.

[6]Olds, J. R., and Lee, H., "Application of a New Economic Analysis Tool to a Two-Stage-to-Orbit RBCC Launch Vehicle Design," AIAA Paper 96-4092, Sept. 1996.

[7]Cramer, E. J., Dennis, J. E., Jr., Frank, P. D., Lewis, R. M., and Shubin, G. R., "Problem Formulation for Multidisciplinary Optimization," *SIAM Journal of Optimization*, Vol. 4, No. 4, 1994, pp. 754–776.

[8]Haftka, R. T., Sobieszczanski-Sobieski, J., and Padula, S. L., "On Options for Interdisciplinary Analysis and Design Optimization," *Structural Optimization*, Vol. 4, 1992, pp. 65–74.

[9]Giunta, A. A., Balabanov, V., Haim, D., Grossman, B., Mason, W. H., Watson, L. T., and Haftka, R. T., "Wing Design for a High-Speed, Civil Transport Using a Design of Experiments Methodology," AIAA Paper 96-4001, Sept. 1996.

[10]Gloat, C. R., and Hague, D. S., "ODIN: Optimal Design Integration System," Aerophysics Research Corp., NASA CR-2492, Hamp-

ton, VA, Feb. 1975.

[11]Wilhite, A., "The Aerospace Vehicle Interactive Design System," NASA Langley Research Center, NASA TM-81957, Hampton, VA, Feb. 1981.

[12]Krieger, R. J., and Mockapetris, L. J., "Interactive Design Technique for Arbitrarily Shaped Missile Configurations," AIAA Paper 84-0390, Jan. 1984.

[13]Alberico, J. F., "The Development of an Interactive Computer Tool for Synthesis and Optimization of Hypersonic Airbreathing Vehicles," AIAA Paper 92-5076, Dec. 1992.

[14]Lockwood, M. K., Petley, D. H., Hunt, J. L., and Martin, J. G., "Airbreathing Hypersonic Vehicle Design and Analysis Methods," AIAA Paper 96-0381, Jan. 1996.

[15]Szedula, J. A., "FASTPASS: A Tool for Launch Vehicle Synthesis," *Proceedings of the AIAA/NASA/ISSMO 6th Symposium on Multidisciplinary Analysis and Optimization* (Bellevue, WA), AIAA, Reston, VA, 1996, pp. 655–663.

[16]Ardema, M. D., Bowles, J. V., and Whittaker, T., "Near-Optimal Propulsion System Operation for an Airbreathing Launch Vehicle," *Journal of Spacecraft and Rockets*, Vol. 32, No. 6, 1995, pp. 951–956.

[17]Weston, R. P., Townsend, J. C., Eidson, T. M., and Gates, R. L., "A Distributed Computing Environment for Multidisciplinary Design," AIAA Paper 94-4372, Sept. 1994.

[18]Hale, M. A., and Craig, J. I., "Preliminary Development of Agent Technologies for a Design Integration Framework," AIAA Paper 94-4297, Sept. 1994.

[19]Tong, S. S., Powell, D., and Goel, S., "Integration of Artificial Intelligence and Numerical Optimization Techniques for the Design of Complex Aerospace Systems," AIAA Paper 92-1189, Feb. 1992.

[20]Sobieszczanski-Sobieski, J., "Multidisciplinary Design Optimization: An Emerging New Engineering Discipline," *Advances in Structural Optimization*, edited by J. Herskovitz, Kluwer, Dordrecht, The Netherlands, 1995, pp. 483–496.

[21]Olds, J. R., "The Suitability of Selected Multidisciplinary Design Techniques to Conceptual Aerospace Vehicle Design," AIAA Paper 92-4791, Sept. 1992.

[22]Sobieszczanski-Sobieski, J., and Haftka, R., "Multidisciplinary Aerospace Design Optimization: Survey of Recent Developments," AIAA Paper 96-0711, Jan. 1996.

[23]Alexandrov, N., and Hussaini, Y. (eds.), *Multidisciplinary Design Optimization: State of the Art*, SIAM Publications, Philadelphia, PA, 1997.

[24]Craig, J. A., "D-Optimal Design Method: Final Report and User's Manual," U.S. Air Force Contract F33615-78-C-3011, FZM-6777, General Dynamics, Fort Worth Div., TX, 1978.

[25]Olds, J. R., "Results of a Rocket-Based Combined-Cycle SSTO Design Using Parametric MDO Methods," Society of Automotive Engineers, Paper 94-1165, April 1994.

[26]Bush, L. B., and Unal, R., "Preliminary Structural Design of a Lunar Transfer Vehicle Aerobrake," AIAA Paper 92-1108, Feb. 1992.

[27]Engelund, W. C., Stanley, D. O., Lepsch, R. A., McMillin, M., and Unal, R., "Aerodynamic Configuration Design Using Response Surface Methodology Analysis," AIAA Paper 93-3967, Aug. 1993.

[28]Unal, R., Stanley, D. O., and Lepsch, R. A., "Parametric Modeling Using Saturated Experimental Designs," *Journal of Parametrics*, Vol. 16, No. 1, 1996, pp. 3–18.

[29]Galati, T., and Elkins, T., "DOE Approach to an SSTO Take-Off Thrust Augmentation Analysis," AIAA Paper 96-3115, July 1996.

[30]Way, D. W., and Olds, J. R., "Sirius: A New Launch Vehicle Option for Mega-LEO Constellation Deployment," AIAA Paper 97-3122, July 1997.

[31]Braun, R. D., Powell, R. W., Lepsch, R. A., Stanley, D. O., and Kroo, I. M., "Comparison of Two Multidisciplinary Optimization Strategies for Launch Vehicle Design," *Journal of Spacecraft and Rockets*, Vol. 32, No. 3, 1995, pp. 404–410.

[32]Beltrachi, T. J., "Decomposition Approach to Solving the All-Up Trajectory Optimization Problem," *Journal of Guidance, Control, and Dynamics*, Vol. 15, No. 3, 1992, pp. 707–716.

[33]Nguyen, H., "FONSIZE: A Trajectory Optimization and Vehicle Sizing Program," AIAA Paper 93-1100, Feb. 1993.

[34]Rahn, M., and Shöttle, U. M., "Decomposition Algorithm for Performance Optimization of a Launch Vehicle," *Journal of Spacecraft and Rockets*, Vol. 33, No. 2, 1996, pp. 214–221.

[35]Sobieszczanski-Sobieski, J., "Sensitivity Analysis and Multidisciplinary Optimization for Aircraft Design: Recent Advances and Results," *Journal of Aircraft*, Vol. 27, No. 12, 1990, pp. 993–1001.

[36]Olds, J. R., "System Sensitivity Analysis Applied to the Conceptual Design of a Dual-Fuel Rocket SSTO," AIAA Paper 94-4339, Sept. 1994.

[37]Goldberg, D., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison–Wesley, Reading, MA, 1989.

[38]Kirkpatrick, S., Gelatt, C. D., Jr., and Vecchi, M. P., "Optimization by Simulated Annealing," *Science*, Vol. 220, No. 4598, 1983, pp. 671–680.

[39]Gage, P. J., Braun, R. D., and Kroo, I. M., "Interplanetary Trajectory Optimization Using a Genetic Algorithm," *Journal of Astronautical Sciences*, Vol. 43, No. 1, 1995, pp. 59–76.

[40]Joshi, B. D., Morris, W. D., White, N. H., and Unal, R., "Optimization of Operations Resources Via Discrete Event Simulation Modeling," AIAA Paper 96-4181, Sept. 1996.

[41]Joshi, B. D., "Dynamic Scale Genetic Algorithm: An Enhanced Genetic Search for Discrete Optimization," Ph.D. Dissertation, Engineering Management Dept., Old Dominion Univ., Norfolk, VA, Dec. 1996.

[42]Unal, R., Braun, R. D., Moore, A. A., and Lepsch, R. A., "Design Optimization for Cost Using Genetic Algorithms," *Proceedings of the 17th Annual Conference of the International Society of Parametric Analysts*, ISPA, Washington, DC, 1995, pp. 183–191.

[43]Frank, P. D., Booker, A. J., Caudell, T. P., and Healy, M. J., "A Comparison of Optimization and Search Methods for Multidisciplinary Design," AIAA Paper 92-4827, Sept. 1992.

[44]Fleischer, M., "Cybernetic Optimization by Simulated Annealing: Accelerating Convergence by Parallel Processing and Probabilistic Feedback Control," *Journal of Heuristics*, Vol. 1, No. 2, 1996, pp. 225–246.

[45]Bramlette, M. F., and Cusic, R., "A Comparative Evaluation of Search Methods Applied to Parametric Design of Aircraft," *Proceedings of the 3rd International Conference on Genetic Algorithms*, edited by David J. Schaffer, Morgan Kaufmann, San Mateo, CA, 1989, pp. 213–218.

[46]Braun, R. D., "Collaborative Optimization: An Architecture for Large-Scale, Distributed Design," Ph.D. Dissertation, Dept. of Aeronautics and Astronautics, Stanford Univ., Stanford, CA, June 1996.

[47]Braun, R. D., and Kroo, I. M., "Development and Application of the Collaborative Optimization Architecture in a Multidisciplinary Design Environment," *Multidisciplinary Design Optimization: State of the Art*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1997, pp. 98–116.

[48]Braun, R. D., Moore, A. A., and Kroo, I. M., "Collaborative Approach to Launch Vehicle Design," *Journal of Spacecraft and Rockets*, Vol. 34, No. 4, 1997, pp. 478–485.