DETERMINATION OF REVOLUTIONARY REQUIREMENTS BOUNDARIES FOR A HIGH-SPEED, AIRBREATHING PROPULSION SYSTEM

Peter M. Hollingsworth* & Dimitri N. Mavris†
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta GA 30332-0150

Abstract

Modern propulsion system design and selection for future air-vehicle systems is an inherently uncertain process. The long lead-times in the development of new propulsion systems produce significant levels amount of risk for the propulsion system manufacturer. Additionally, this long lead-time allows a tremendous amount of program inertia to build up as the development process progresses. This inertia prohibits the propulsion system manufacturer from reacting to "catastrophic" changes in the system requirements. It can be shown that there exist certain regions in the system requirements hyper-space where a small change in a given requirement or requirements requires a completely different solution. Additionally, because of the inherent security associated with evolving current designs; there exists in the engineering community a reluctance to develop truly new and revolutionary technologies and systems. Therefore it is of interest to develop a method by which the location of catastrophic boundaries can be discerned. The method chosen to investigate the requirements hyper-space for supersonic cruise propulsion systems is a genetic algorithm (GA). The GA was used to determine both individual and combined technology limit boundaries and to determine the effect of technology infusion on these boundaries.

Introduction

Every complex system is constructed of smaller subsystems; these subsystems are in turn constructed of smaller subsystems and/or components. In the case of an aerospace vehicle, major subsystems include the propulsion subsystem, airframe subsystem, avionics subsystem, etc. The aerospace system is itself a subsystem of a larger system, such as a commercial airline. Each system or subsystem can be described as a unique function of a set of state variables (SV), also called a state vector. The members of the state vector can made up of variables that are independently set by the engineer, are functions of higher level variables, and/or are environmental noise. The independent state variables plus the higher level independent variables that control multiple state variables are called the control variables (CV).[1] The benefit in using this formulation is that there are typically only a few CVs that define the overall performance of a system, while the total number of state variables may be in the thousands.[1] Further, the control vector may or may not define a unique response. The requirements for a system are in effect the control vector for that system. They are inherently independent of each other, at least at the particular system level in question. They also serve to define a large number of traditional design variables that ultimately define the performance of the system.

This paper investigates the requirements region for supersonic cruise propulsion systems, specifically the catastrophic boundary for a ramjet propulsion system. Potential requirements to be investigated include:

- Required TSFC/ISP
- Required T/W
- Required Specific Thrust
- Required Maximum Mach Number
- Required Cruise Mach Number
- Fuel Type
- Geometry

* Graduate Research Assistant, Student Member AIAA
† Boeing Professor, Associate Fellow AIAA

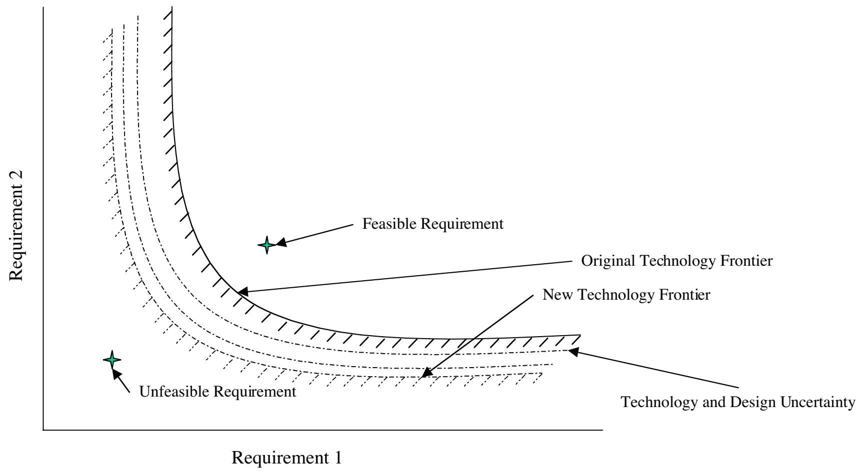

Figure 1: Notional Effect of Technology Infusion, and Uncertainty on a Single Requirements Space Catastrophic Boundary

Notional results for a 2-dimensional requirements hyper-space, incorporating both uncertainty in technology limits and in the requirements are shown in Figure 1.

Background

The study of the requirements hyper-space for complex systems requires the combination of ideas from several disciplines. These include methods of complex system construction and modeling, control and state variables, and catastrophe theory.

System Buildup

Most complex systems designs can be thought of in terms of build-ups of small/simpler subsystems. In turn these subsystems are themselves made up of further subsystems and components. The components are each defined by a set of state properties. It is the product of all of the component state properties that produces the final, unique system. This type of buildup is illustrated in Figure 2.

The whole process of identifying a complete system in this way is extremely intensive. While it is necessary for detailed design, it is overly time consuming and requires too much knowledge to be performed during the conceptual or even pre-conceptual design phase. However, it is possible to think of each subsystem of a given system level as a component with a unique set of properties. This reduced fidelity/complexity idea is illustrated in Figure 3. This simplified system decreases not only the total complexity of the problem, but also hides a large percentage of the unique states that exist in the full model.

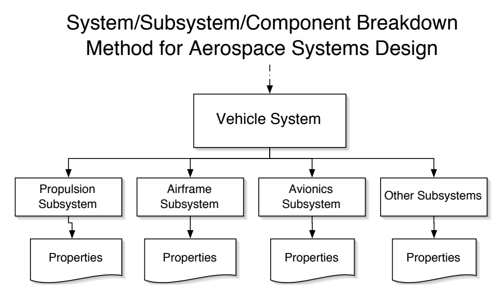

Figure 3: Simplified Multi-Level System Buildup Flow-chart

Requirements Space Investigation

Using this multilevel method allows the engineer to consider the requirements for a given subsystem psuedo-independently of the requirements placed upon the other subsystems. The challenge, therefore, becomes how to determine the requirements for a given subsystem. This is simplified by the fact that many of the requirements for a given subsystem are themselves functions of factors and responses at higher system levels, i.e. the requirements for a commercial aircraft propulsion system are set by the airframe, regulatory environment, airline economics, etc. Therefore, once the potential higher-order systems are identified, and their needs determined, it is possible to translate these needs into requirements for a given subsystem. The sub-

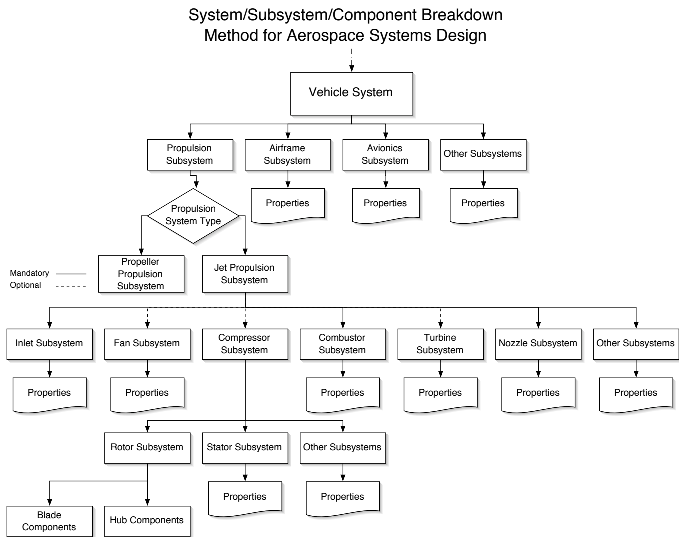

Figure 2: Multi-Level SystemSubsystemComponent Buildup Flow-chart

system of interest to this paper is the cruise propulsion system of a high speed, airbreathing, cruise vehicle. Because of the relatively constrained scope of such a system it is relatively straightforward to develop a set of constraining requirements. Some of these are given in the introduction. Additionally, for the early conceptual and pre-conceptual design phase the properties required to define such a system are relatively few, consisting of mainly efficiencies, pressures, temperatures, etc. An impediment to using this method is that since the requirements are not truly independent of the system levels above the current subsystem, there is a high likelihood that their values will change over time. It is, therefore, even more important that the requirements for subsystems be treated in a non-deterministic manner. Furthermore, there maybe a higher likelihood that catastrophic changes in the requirements may occur, increasing the risk level associated with a subsystem program. Therefore, a relatively comprehensive identification of these catastrophic boundaries is desired.

Catastrophic requirements boundaries are curves in the requirements space, where a small change in one or more of the system or subsystem requirements produces a discontinuous change in the resulting system state. These are caused by limits in the capabilities of technologies that make up the system or subsystem. Since these limits identify the points at which the system changes appreciably, and development risk is the highest, identifying them is of great interest. The most straightforward manner to approach this identification is to study the topology of the entire requirements hyper-space; this was performed by the authors for a notional rapid response strike system.[2] There are, however, significant drawbacks to this approach.

Combinatorial Aspects

One of the most significant drawbacks is that the computational power necessary to determine the shape of the potential system responses is inherently high. Most requirement spaces are hyperspacial, i.e. very often there are more than three requirements that posses uncertainty. Additionally, if the user wants to get a true feel for the exact response hypersurfaces, the number of points in each dimension that must be evaluated is relatively high. Since generally it cannot be assumed that the hypersurface behave in a polynomial manner, i.e. the typical structured models used for the Response Surface Methods (RSM), the number of points, at any given level, that must be evaluated is an increasing function of the resolution desired. Therefore, if the engineer wishes a relatively high fidelity the number of observation points, per dimension, can easily exceed ten or fifteen. This poses a problem as the total number of function calls is given as a power function of the number of observations in each dimension. This is shown in Equation 1 for a uniform number of levels in each dimension.

$$Fcalls = obs^{dimen} \tag{1}$$

For a 5 dimensional hyper-space and 10 evaluation levels in each dimension this is 100,000 calls. For a

6 dimensional space, which is entirely probable, and 16 evaluation levels, the number of function calls required grows to over 16 million. Additionally, each different system for which the response is being evaluated must be run separately; further increasing the number of function calls. If each function call takes 1 second to evaluate on a modern CPU, the total time to evaluate the 6 dimensional, 16 level requirements hyper-space for two competing systems is over 1 year. While parallelizing the computations can decrease the actual time, the CPU cost can quickly become prohibitive. Therefore a less intensive method of determining the catastrophic boundaries needs to be developed.

One of the benefits of design is that it is a compilation of multiple systems/disciplines for which either the physics and/or technology set the performance limits. It is, therefore, possible to determine the physical and technological limits of potential system beforehand, i.e, the overall pressure ratio of a gas turbine engine is limited to 75 for a given level of technology. When the response limits are reached, any increase in the severity of the requirements produces a non-feasible system. With these predetermined limits it is possible to determine the catastrophic surfaces in the requirements hyperspace.

There are many potential ways to perform this. One of the most promising techniques is the use of genetic algorithms. Roth et al., have demonstrated the use of genetic algorithms in a highly combinatorial technology evaluation space.[3] In this case 40 technologies were evaluated to determine their system level benefit when infused into a baseline system. Roth et al., have demonstrated the ability to determine a front of technology combinations of different levels for customer "desirements." The extension of this capability to the arena of system requirements is clearly feasible. Additionally, when evaluating future technologies and systems, there is an inherent uncertainty as to their true capabilities. This uncertainty needs to be incorporated. Because of the nature of genetic algorithms, the incorporation of uncertainty into the process is relatively easy. An initial implementation involving technology risk was demonstrated by Roth et al.[4] In this instance the effect that a new technology had on the performance of a system was considered an uncertain quantity. The incorporation of uncertainty in technology boundaries would be handled in a similar manner.

One of the other benefits of using a GA is that all of the technology limits can be evaluated during one run. In this case the resulting limit is either an amalgamation of the most stringent limits, or the most stringent limit. When comparing multiple systems or evaluating the limits on a single system, these resulting technology limits can provide a "requirements Pareto front", i.e. the maximum limit of requirements that the given technologies will allow.

Table 1: Notional High-Speed, Cruise Propulsion System Requirements/Control Variables

Requirements	Lowerbound	UpperBound
Mach#	2.0	6.0
I_{SP} (sec)	500	4500
Fn_{SP} (lblbsec)	50	250
Inlet Area (ft^2)	0.1	10
Fuel Type	H_2	JP

Implementation

In order to validate the existence of technology driven, catastrophic boundaries in the requirements hyper-space at the subsystem level, and validate the idea of using a genetic algorithm to determine the location of these boundaries, an high-level environment must be used. One of the problems with many propulsion system cycle codes is that they require significant knowledge of the system to operate. Further, propulsion system cycle analysis generally involves analyzing many different operating conditions. Therefore, in order to scope the problem to a size appropriate for the undertaking it was decided that only the cruise propulsion system of a high speed cruise vehicle would be studied, i.e. it would be assumed, for simplicities sake, that the cruise cycle would be different from the boost cycle. In a real world environment one would want to consider both single and variable cycle boost/cruise systems; however, there is no theoretical reason that the ideas and methods used in this paper could not be expanded to include these systems and their associated requirements. The notional requirements for the high-speed, cruise propulsion system studied in this paper are listed in Table 1.

For the purposes of this study only one system, a ramjet, was evaluated. However, the procedure would be essentially identical for additional systems. The code used to perform the cycle analysis on the ramjet was RAMSCRAM.[5] RAMSCRAM is a quasi-one dimensional ramjet/scramjet/rocket cycle analysis code, which includes an equilibrium gas chemistry model. Since it allows for varying levels of fidelity it is relatively straight-forward to analyze a propulsion system at the pre-conceptual level.

Of course without some sort of limits on the capabilities of the cycle, at least one solution can be

Table 2: Technological Limits Imposed Upon the High-speed, Cruise Propulsion System

Technology Limit	Value	Unit
η_{inlet}	0.80 - 0.95[6]	
$\eta_{combustor}$	0.95	
η_{nozzle}	0.99	
T4	4500 & 5000	° R
T3	2900	° R

Table 3: Input/Output Status of Requirements & Limit Variables

Requirement	Status	Variable Name
I_{SP}	Output	ISP
Mach Number	Input	AMO
Fn $_{SP}$	Output	SPF
Fuel Type	Input	multiple
Inlet Area	Input	ADES1
Technology Limit	**Status**	**Variable Name**
η_{inlet}	Input	AKD1
$\eta_{combustor}$	Input	AKD3
η_{nozzle}	Input	CS
T4	Output	T4
T3	Output	T3

found at any combination of the requirements. The analytical model for RAMSCRAM includes limits such as conservation of energy. While this limits the space significantly, more limits must be imposed to determine the true viable range of the system in the requirements hyperspace. The technological limits imposed are presented in Table 2.

Because of the implementation of RAMSCRAM, some of the CVs were outputs, and many of the limit variables were inputs. This forces the user to use the code in an inverted manner. Table 3 lists both the control and technology limit variables and their status as either inputs or outputs. Furthermore, it was necessary to allow for the variation of a number of state, or design variables, to vary in order to ensure that each function call produced a physical result. These variables are listed in Table 4, on page 5.

In order to verify the capability of the GA in determining the most stringent combination of technology limits, the GA was run individually for each technology limit and subsequently for all of the limits combined.

Results

The investigation both into the existence and the validity of using a GA to determine the location of the catastrophic boundaries was performed using individual technology limits and the combined technology limits. Both of these methods saw significant computational improvement over using the grid method.

Investigation of Individual Limits

The results of the individual technology limit searches are presented separately for the hydrogen and JP fuel ramjet systems. This is done in part because of the difficulty in showing highly dimensional requirements hyper-spaces. The reason for splitting the fuel type requirement up is that it is a categorical requirement. In this study the ramjet is either JP or H_2 fueled, and is not allowed to be a mixture of the two.

JP Fueled Ramjet

The GA successfully determined the presence of individual technology limits for the JP fueled ramjet. These limits were determined in a four-dimensional hyper-space. Therefore, visualization of these limits is a difficult matter. In order to minimize the amount of data presented to the reader, only a slice of the space is shown. Through reduction of the data, it was determined that the inlet area and specific thrust presented themselves as more correlated; therefore, the axis chosen for presentation were Mach and ISP. The slice was taken at an inlet area of one square foot, and a specific thrust of 70 (lbf/lbm/sec). The results are shown in Figure 4, on page 6.

It is interesting to note that the technology limit boundaries track almost identically at a constant ISP for the lower mach numbers; however the nozzle efficiency seems to decrease the ISP as the Mach number decreases. Additionally, because no work is done on the flow during compression, the temperature location of the temperature limits in the requirements hyper-space are solely a function of one CV, Mach number. Figure 4 shows that for a T4 limit of 4500 °R, the maximum Mach limit occurs around Mach 4.6. An increase in the technological capability of the engine to a T4 of 5000 °R, which may be provided through active cooling of the combustor and nozzle walls increase the maximum Mach number to greater than Mach 6. Furthermore, for this run the GA determined that there maybe a lower Mach limit of around 4.1, determined by the combustor pressure ratio. This maybe a spurious data point, produced by the mutation function of the GA during the last generations.

Table 4: Additional Design/State Variables Used in the Genetic Algorithm

State Variable	Range	Variable Name
Altitude	30,000 - 150,000 ft	ALT
Nozzle Velocity Coefficient	0.9 - 1.0	CV
Normal Shock Mach#	1.1 - 6.0	AMD2
Nozzle Exit Area	1 - 10 ft^2	ANOZZ
Equivalency Ratio	0.95 - 1.05	PHI

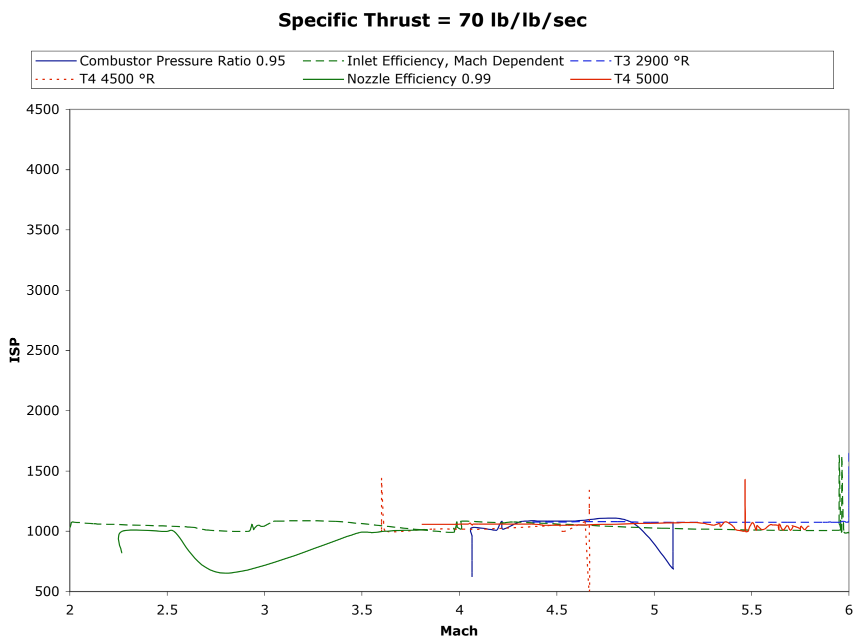

Figure 4: JP Powered Ramjet Requirements Space, Mach vs. ISP

H_2 Fueled Ramjet

Again the GA was successful in identifying the individual requirements boundaries for the hydrogen fueled ramjet. The maximum ISP range for the hydrogen ramjet was significantly higher than for the JP fueled ramjet. This is to be expected. If volumetric ISP had been included as one of the requirements the hydrogen ramjet would have fared differently. The results for the hydrogen fueled ramjet are presented in Figure 5, on page 6, again the slice was taken at an inlet area of one square foot, and a specific thrust of (70 lbf/lbm/sec).

Again the results were essentially similar with the nozzle efficiency limit decreasing the ISP boundary at lower Mach numbers. The T4 limit of 4500 °R produces a maximum Mach limit of around 4.1, with the limit increased to a T4 of 5000 °R, this increases to Mach 5.9. It is reasonable to assume the active cooling capabilities of the hydrogen fueled ramjet are greater than that of the JP fueled ramjet. Furthermore, it is of interest to note that the combustor efficiency limit did not produce a minimum Mach number limit, further suggesting that the results in the JP fueled ramjet case were an aberration. Additionally, in neither case did the T3 maximum temperature limit produce an effect in the requirements range studied.

Investigation of the Requirements Pareto Front

In order to determine if the GA was capable of determining the requirements Pareto front for the ramjet vehicle, which searched for all of the technology limits simultaneously. The results of this are presented in Figure 6 on page

The GA was generally succesful in finding the Pareto limit for the ramjet in the requirements hyper-space. However, the trends associated with the nozzle efficiency that were visible at the lower Mach numbers in both Figures 4 & 5, are not present in Figure 6. It seems that the GA did not produce any results below Mach 3.75 for the JP fueled ramjet and below Mach 4.8 for the hydrogen fueled ramjet. Further study is required to properly ascertain the reasons for this; however, it is likely that fewer cases as closely satisfied the GA goals in the Pareto front analysis, and that most of them occurred at the higher Mach number, particularly the Mach limit. Even with this problem, the GA proved quite capable of determining the requirements Pareto front.

Computational Benefits

One of the primary reasons for using a GA in the determination of the requirements boundaries is that it is computationally more efficient that solving for solutions across a grid. While the grid computational effort grows with a multiple of the number of levels and a power of the number of variables, the GA effort is generally only a multiple of both the number of levels and variables, which determine the populations size. For this study the GA was set up to use five bits to represent each input variable. This creates a total of 32 levels for each variable. There were eleven input variables. The population was set to three times the sum of all of the bit length for all of the variables. The total number of generations was one third the total population. Several runs were performed, which showed that the population had stabilized in less generations than were used. Additionally, RAMSCRAM took approximated 0.05 CPU seconds to perform one case on the single CPU Apple Powerbook G4 800 on which it was run. With these values we can determine the computational effort required for both the grid and GA methods. Table 5, on page 8, shows comparison of the computational effort for each method.

The resulting savings using the GA makes a problem that is computationally unmanageable and provides a solution that in either is quick enough to be used in pre-conceptual and conceptual design.

Conclusions

The use of a reduced order complex system buildup coupled with a genetic algorithm to determine the catastrophic boundaries in the requirements hyperspace for a propulsion subsystem. Not only was the GA able to determine individual technology based boundaries, but it was also able to determine the combined technology boundaries, and the associated "requirements Pareto front." Further effort needs to be made to ensure that a GA can robustly determine these boundaries, verify the inclusion of probabilistic boundaries using non-uniform type distributions, and develop a straight-forward and relatively simple method of visualizing the results obtained from the requirements analysis.

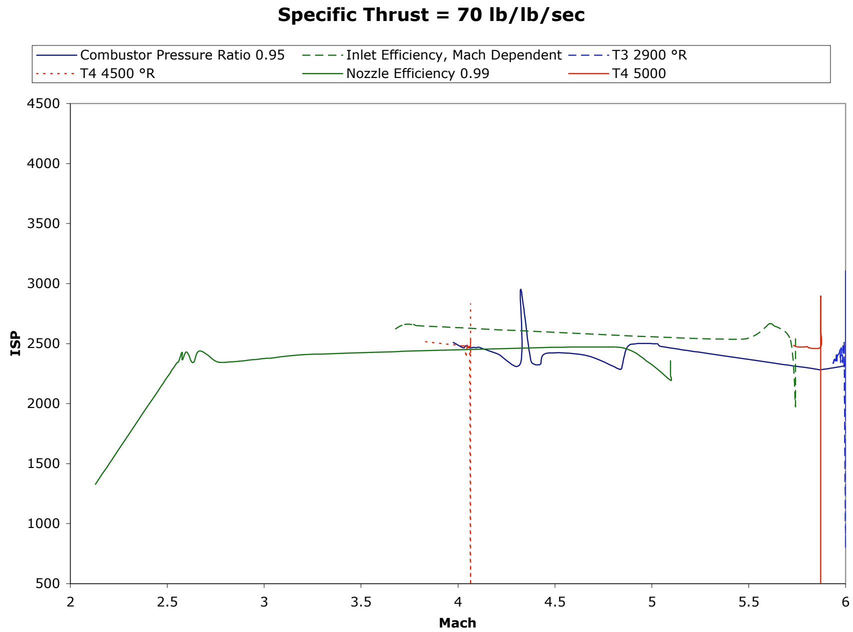

Figure 5: JP Powered Ramjet Requirements Space, Mach vs. ISP

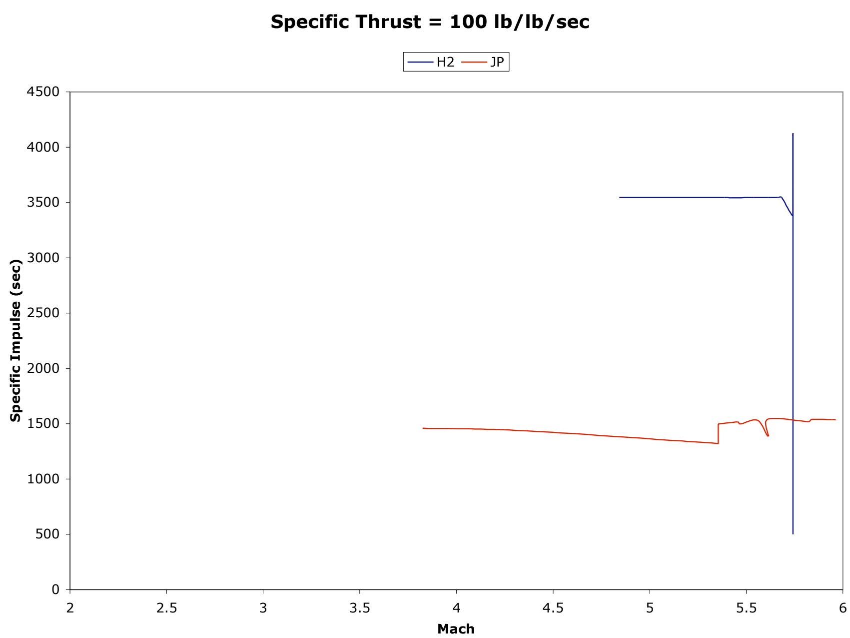

Figure 6: Ramjet Requirements Space, Combined Boundaries, Mach vs. ISP

Table 5: Comparison of the Computational Effort for Different Requirements Boundaries Discovery Methods

Method	Function Calls	Total Function Call CPU Time
Grid	3.6×10^{16}	57 million years
Individual GA	90,750	2.36 hours
Combined GA	9,075	7.6 minutes

Definitions & Abbreviations

- Catastrophic Boundary: A point or series of points in the control variable space where a slight change in one or more of the control variables produces a drastic change in the response.
- CV: Control Variable - Independent variables that control the response. Does not necessarily produce a unique response
- Desirement: Items, or characteristics that the customer desires to have but does not necessarily require to have.
- GA: Genetic Algorithm
- SV: State Variable - Dependent variable the determines the unique state of the response.
- Requirements Pareto Front: The catastrophic requirements boundaries for a system or multiple systems, that determines the curve of best requirements, i.e. any more stringent change in one requirement requires changes in the other requirements.
- Technology Limit: The maximum, or minimum, value of a property for a given subsystem that is allowed by either a given technology, or physical constraints, i.e. maximum temperature for a material or the second law of thermodynamics.

References

1. Sunders, P. T., *An Introduction to Catastrophe Theory*, Cambridge University Press, New York, 1980, Pp. 1-43.
2. Hollingsworth, Peter, Dimitri N. Mavris, "Identification of the Requirements Space Topology for a Rapid Response Strike System," Presented at the SAE Aerospace Congress and Exhibition, Seattle, WA, September 10-13, 2001.
3. Roth, Bryce, Brian German, Dimitri Mavris, Noel Macsotai, "Adaptive Selection of Engine Technology Solution Sets from a Large Combinatorial Space," Presented at the 37th Joint Propulsion Conference and Exhibit, Salt Lake City, UT, July 2001.
4. Roth, Bryce, Matthew Graham, Dimitri Mavris, Noel Macsotai, "Adaptive Selection of Aircraft Engine Technologies in the Presence of Risk," Presented at the 2002 ASME Turbo Expo, Amsterdam, Netherlands, June 3-6 2002.
5. *RAMSCRAM Users Manual*, Propulsion Systems Office, NASA Glenn Research Center.
6. Mahoney, John J., *Inlets for Supersonic Missiles*, American Institute of Aeronautics and Astronautics, Washington, 1990. Page 139.