



**SCRAMJET FUEL INJECTION ARRAY OPTIMIZATION
UTILIZING MIXED VARIABLE PATTERN SEARCH WITH
KRIGING SURROGATES**

THESIS

Bryan Sparkman, Captain, USAF

AFIT/GOR/ENS/08-19

**DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY**

Wright-Patterson Air Force Base, Ohio

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AFIT/GOR/ENS/08-19

**SCRAMJET FUEL INJECTION ARRAY OPTIMIZATION
UTILIZING MIXED VARIABLE PATTERN SEARCH WITH
KRIGING SURROGATES**

THESIS
Bryan Sparkman
Captain, USAF

AFIT/GOR/ENS/08-19

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense or the United States Government.

AFIT/GOR/ENS/08-19

**SCRAMJET FUEL INJECTION ARRAY
OPTIMIZATION UTILIZING MIXED VARIABLE
PATTERN SEARCH WITH KRIGING SURROGATES**

THESIS

Presented to the Faculty
Department of Operational Sciences
Graduate School of Engineering and Management
Air Force Institute of Technology
Air University
Air Education and Training Command
in Partial Fulfillment of the Requirements for the
Degree of Master of Science in Operations Research

Bryan Sparkman, B.S.

Captain, USAF

March 2008

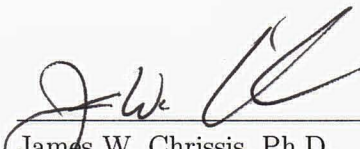
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AFIT/GOR/ENS/08-19

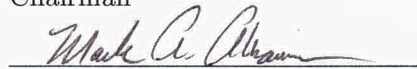
SCRAMJET FUEL INJECTION ARRAY
OPTIMIZATION UTILIZING MIXED VARIABLE
PATTERN SEARCH WITH KRIGING SURROGATES

Bryan Sparkman, B.S.
Captain, USAF

Approved:


James W. Chrissis, Ph.D.
Chairman

26 Mar 08
Date


LtCol Mark A. Abramson, Ph.D.
Member

26 MARCH 2008
Date

Abstract

Fuel-air mixing analysis of scramjet aircraft is often performed through experimental research or Computational Fluid Dynamics (CFD) algorithms. Design optimization with these approaches is often impossible under a limited budget due to their high cost per run. This investigation uses JETPEN, a known inexpensive analysis tool, to build upon a previous case study of scramjet design optimization. Mixed Variable Pattern Search (MVPS) is compared to evolutionary algorithms in the optimization of two scramjet designs. The first revisits the previously studied approach and compares the quality of MVPS to prior results. The second applies MVPS to a new scramjet design in support of the Hypersonic International Flight Research Experimentation (HiFIRE). The results demonstrate the superiority of MVPS over evolutionary algorithms and paves the way for design optimization with more expensive approaches.

Acknowledgements

I would like to thank the members of my thesis committee, LtCol Mark Abramson and Dr. James Chrissis. Their extraordinary knowledge and expertise provided an invaluable base of knowledge. Also I would like to thank Dr. Mark Gruber of the Air Force Research Lab, who provided a working copy of JETPEN and the HiFIRE application of this thesis. His guidance and support was greatly appreciated. Lastly, I thank the L^AT_EX crew for helping me preserve my sanity each night as we worked past 3am.

Bryan Sparkman

Table of Contents

	Page
Abstract	iv
Acknowledgements	v
List of Figures	ix
List of Tables	xi
1. Introduction	1-1
1.1 Background	1-1
1.1.1 Hypersonic International Flight Research Experimentation	1-2
1.1.2 Scramjet Engines	1-4
1.1.3 Numerical Models	1-6
1.1.4 Optimization	1-7
1.2 Purpose of Research	1-9
2. Relevant Literature	2-1
2.1 Overview	2-1
2.2 Transverse Fuel Injection	2-1
2.2.1 Injection Angle	2-3
2.2.2 Array Design	2-4
2.2.3 Injectant Dynamic Pressure	2-5
2.3 Other Injection Techniques	2-6
2.3.1 Ramp Injection	2-6
2.3.2 Pylon Injection	2-7
2.4 JETPEN Numerical Simulation	2-8

	Page
2.4.1	Mach Disk Estimation 2-8
2.4.2	Governing Behavior 2-10
2.4.3	Validation 2-12
2.5	Simulation Optimization 2-13
2.5.1	Evolutionary Algorithms 2-15
2.5.2	Micro-Genetic Algorithms 2-17
2.5.3	Response Surface Methods 2-18
2.6	Pattern Search Methods 2-21
2.6.1	Generalized Pattern Search 2-22
2.6.2	Mixed Variable Pattern Search 2-26
2.6.3	Convergence Results 2-29
2.6.4	Kriging Surrogates 2-30
2.7	Summary 2-33
3.	Problem Approach 3-1
3.1	Design Variables 3-1
3.2	Dependent Variables 3-3
3.3	Design Evaluation 3-4
3.4	Performance Measures 3-5
3.4.1	Jet Penetration 3-6
3.4.2	Plume Expansion 3-6
3.4.3	Fuel Concentration Decay 3-7
3.4.4	Performance Measure Estimation 3-8
3.5	Constraints 3-9
3.5.1	Payne’s Design 3-9
3.5.2	HiFIRE Design 3-10
3.6	Summary 3-10

	Page
4. Optimization	4-1
4.1 Problem Statement	4-1
4.1.1 Design and Response Vectors	4-1
4.1.2 Original Design Problem Statement	4-3
4.1.3 HiFIRE Problem Statement	4-3
4.2 Objective Function Form	4-4
4.3 Algorithm Performance Assessment	4-6
4.4 Mixed Variable General Pattern Search Application	4-7
4.5 Genetic Algorithm Application	4-8
4.6 JETPEN Monte-Carlo Sampling	4-9
4.7 Summary	4-11
5. Results	5-1
5.1 Previous Design Re-Optimization Results	5-1
5.2 Relaxation of Injector Restriction	5-6
5.3 HiFIRE Optimization: 1 Variable	5-8
5.4 HiFIRE Optimization: 2 Variables	5-11
5.5 Summary	5-13
6. Conclusions and Future Recommendations	6-1
6.1 Final Design Evaluation	6-1
6.2 Genetic Algorithms	6-2
6.3 Future Applications of MVPS	6-2
6.4 Future Recommendations	6-3
Bibliography	BIB-1
Appendix A. HiFIRE Mission Parameters	A-1
Appendix B. Dependent Variables	B-1

List of Figures

Figure		Page
1.1.	Dimensions of the NASA X-43A	1-2
1.2.	AFRL X-51 Concept Vehicle	1-3
1.3.	Scramjet Engine	1-5
2.1.	Transverse Fuel Injection	2-2
2.2.	Wall Fuel Injection	2-4
2.3.	Ramp Fuel Injection	2-7
2.4.	Central Pylon Fuel Injection	2-7
2.5.	Merged Adjacent Plumes	2-13
2.6.	Comparison of Predicted vs. Experimental Data	2-14
2.7.	Predicted vs. Measured Plume Cross-section	2-14
2.8.	Evolutionary Strategy	2-15
2.9.	Simple Genetic Algorithm	2-16
2.10.	Micro-Genetic Algorithm	2-17
2.11.	Gradient-Based RSM	2-21
2.12.	GPS Poll Step and Mesh Update	2-25
2.13.	Basic GPS Algorithm	2-25
2.14.	MVPS Algorithm	2-29
2.15.	MVPS Full Iteration	2-30
3.1.	Fuel Injection Array Cross-Section	3-2
3.2.	Flowchart of Design Evaluation Process	3-4
3.3.	Jet Penetration	3-6
3.4.	Plume Expansion	3-7
3.5.	Fuel Concentration Decay	3-8

Figure		Page
3.6.	Typical JETPEN Plume Penetration Data	3-9
4.1.	JETPEN crashes at $N = 10$	4-10
4.2.	Local JETPEN behavior ($\vec{x} \pm 0.5$)	4-11
5.1.	Genetic Algorithm Average Performance Across 5 samples . .	5-4
5.2.	Mixed Variable Pattern Search Performance	5-4
5.3.	MVPS and GA comparisons	5-6
5.4.	Algorithm Convergence Comparisons	5-8
5.5.	Algorithm Performance	5-13

List of Tables

Table		Page
1.1.	Typical takeoff weight fraction breakdowns of current systems	1-2
2.1.	Parameter Descriptions	2-12
2.2.	Parameter Values for Experimental Data Comparisons	2-13
3.1.	Design Variables	3-2
4.1.	Payne’s Performance Measures	4-2
4.2.	Tuning parameters for Genetic Algorithms	4-8
4.3.	Monte Carlo Results	4-9
5.1.	Payne’s Design Re-Optimization Results	5-2
5.2.	Best Designs and Associated Responses	5-3
5.3.	Genetic Algorithm Comparisons	5-3
5.4.	Incumbent Improvement Breakdown by Algorithm Step	5-5
5.5.	AFRL Design Re-Optimization Results	5-7
5.6.	Optimization Results	5-7
5.7.	Incumbent Improvement Breakdown by Algorithm Step	5-8
5.8.	AFRL Design: Baseline 15 ^o Results	5-9
5.9.	HiFIRE Design Optimization Results	5-10
5.10.	$f_1(\vec{Y})$ Improvement Over Baseline (Best F)	5-10
5.11.	HiFIRE Design Optimization Results	5-10
5.12.	$f_2(\vec{Y})$ Improvement Over Baseline (Best F)	5-11
5.13.	HiFIRE Design Optimization Results	5-11
5.14.	$f_3(\vec{Y})$ Improvement Over Baseline (Best F)	5-12
5.15.	Surrogate Performance by Objective Function	5-12
5.16.	Designs from 2-Variable Optimization	5-13
5.17.	2-Variable Optimization Improvement Over Baseline	5-13
A.1.	HiFIRE Combustion Parameters	A-1
A.2.	HiFIRE Flowpath Parameters	A-1
A.3.	HiFIRE Fuel Flow Conditions	A-1
A.4.	HiFIRE Air Flow Conditions	A-2

SCRAMJET FUEL INJECTION ARRAY OPTIMIZATION UTILIZING MIXED VARIABLE PATTERN SEARCH WITH KRIGING SURROGATES

1. Introduction

1.1 *Background*

For over five decades hypersonic air-breathing propulsion has been the focus of much research in the aerospace engineering community. The goal of this research is to develop hypersonic air-breathing vehicles capable of flight in the range of Mach 6–12, and beyond. Advanced versions of these aircraft are envisioned to takeoff from conventional runways, accelerate to hypersonic speeds, and enter low-earth orbit. Prior to 2002 the only means of hypersonic flight, let alone low-earth orbit, was by rocket propulsion.

Rocket propulsion systems must carry their own oxidizer, which comprises most of the total vehicle weight. This large weight penalty results in higher payload delivery costs and reduced efficiency. Since rocket-based systems must travel through the atmosphere, this situation has been likened to bringing a canteen full of water to a fish [36]. The promise of hypersonic air-breathing propulsion lies in the ability to burn fuel utilizing oxygen from the atmosphere, thus avoiding the weight penalty incurred by rocket systems. The typical takeoff weight fraction (TWF) of air-breathing aircraft and rocket systems are compared in Table 1.1 [36]. The elimination of the oxidizer weight penalty allows hypersonic air-breathing aircraft to devote larger weight fractions to payload and support systems. In turn this should translate into efficiency and durability not achievable by rocket systems.

Table 1.1 Typical takeoff weight fraction breakdowns of current systems

TWF	Rocket	Aircraft
Oxygen	65%	0%
Fuel	24%	30%
Empty	7%	55%
Payload	4%	15%

Despite decades of research, only recently have the most basic flight characteristics of these exotic aircraft come to fruition. Engineers at Australia’s University of Queensland are credited with the first successful flight test of an air-breathing hypersonic vehicle in July of 2002. In March and November of 2004, NASA successfully tested the X-43A flight test vehicle as part of their Hyper-X program, shown in Figure 1.1 [52]. The X-43A demonstrated sustained speeds of Mach 6.8 and Mach 9.6. These vehicles were the culmination of years of research by both organizations.

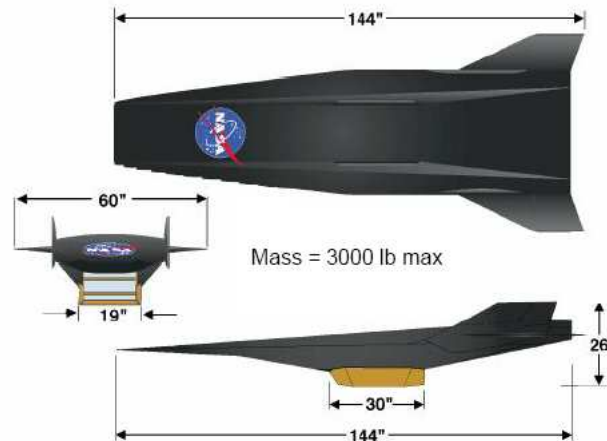


Figure 1.1 Dimensions of the NASA X-43A

1.1.1 Hypersonic International Flight Research Experimentation

The Hypersonic International Flight Research Experimentation program, or HiFIRE, is a collaborative effort between the U.S. Air Force, the Australian Defense Force, and NASA. While not a direct follow-on from the Hyper-X program,

the purpose of HiFIRE is to develop and demonstrate critical hypersonic technologies. Research objectives of the HiFIRE program include “boundary layer transition (BLT), turbulent separated shock boundary layer interaction (SBLI), and optical measurement of mass capture (OMC) in a duct” [38]. These experiments are intended to improve the overall knowledge of poorly understood hypersonic phenomena. Flight tests of the HiFIRE program are scheduled to begin in 2008.

The X-51A flight test vehicle is another hypersonic aircraft that is currently under development and is conceptually shown in Figure 1.2. The X-51A could potentially evolve into an air-launched expendable hypersonic cruise missile for potential use against time-sensitive and hardened targets. Unlike the X-41A, the X-51A uses JP-7 as fuel. The JP-7 is “cracked” into smaller, lighter fuels by the high operating temperatures of the engine. This unique approach allows the X-51A to use its fuel to both cool and power its engine, while avoiding exotic cryogenic fuels such as hydrogen [43].



Figure 1.2 AFRL X-51 Concept Vehicle

1.1.2 Scramjet Engines

Hypersonic flight cannot be achieved by conventional turbojet or ramjet engines. These engines operate by compressing and reducing the speed of a subsonic or supersonic airflow prior to combustion. This deceleration and compression translates into an increase in the overall pressure, temperature, and density of the airstream. At speeds of Mach 6 and greater the temperatures and heat transfer rates generated are high enough to incinerate most known materials [36]. These temperatures also result in dissociation of the combustion materials, resulting in large chemical energy losses [36].

The solution to this problem is to only partially compress the airstream, thus keeping the internal flow of the engine at supersonic speeds. This type of engine is known as a supersonic combustion ramjet, or scramjet. Scramjet engines, similar to ramjets, have no moving parts and use the forward movement of the vehicle and shape of the engine and vehicle to achieve compression [36]. A major drawback is that scramjet engines cannot produce thrust at a standstill, required some other mode (usually a rocket) to accelerate up to the “takeover speed”.

Fuel injection within a scramjet engine is a difficult task to accomplish. The major challenges are “accomplishing stable, efficient mixing and combustion in a supersonic flow within a burner of reasonable size” [36]. The subsonic flows within turbojets and ramjets allow sufficient time for fuel to mix in appropriate ratios. However, at supersonic speeds encountered within scramjets the residence time of any fluid particle inside the engine is on order of 10^{-3} seconds [56]. This requires appropriate penetration and mixing of fuel to occur in extremely short time spans [67].

Sustaining combustion is another difficulty encountered in scramjet engines. The high speed airflow through the engine requires the flame within the combustor to move extremely fast. Fuel auto-ignition and internal cavities have been methods used to solve the problems with flame suffocation and sustainment [22]. Combustors

designed to have an internal temperature high enough to auto-ignite the fuel/air mixture are sensitive to flight conditions and must be restricted to narrow altitude and speed ranges. Low-pressure zones created by ramps and cavities internal to the combustor have been effective at flame holding, but they also may inhibit the flow and contribute to pressure losses.

Compounding these difficulties is the fact that airflow at hypersonic speeds has inviscid fluid properties [15]. These properties manifest themselves as a series of normal and oblique shock waves emanating from the vehicle surfaces [37]. These shock waves dominate the flow properties at the high Reynolds numbers present in hypersonic flows and may interact with each other and the boundary layer to create effects non-existent at lower speeds [37].

A schematic of a theorized scramjet engine is shown in Figure 1.3 [19]. In this graphic, the oncoming air is compressed and deflected by a series of shockwaves emanating from ramps on the forwards section of the aircraft. Upon entering the engine, the air is further compressed, fuel is injected, and the mixture of air and fuel is burned in the combustor. The expanding air exits the engine as the aft section of the aircraft is used as an additional expansion surface for the exhaust.

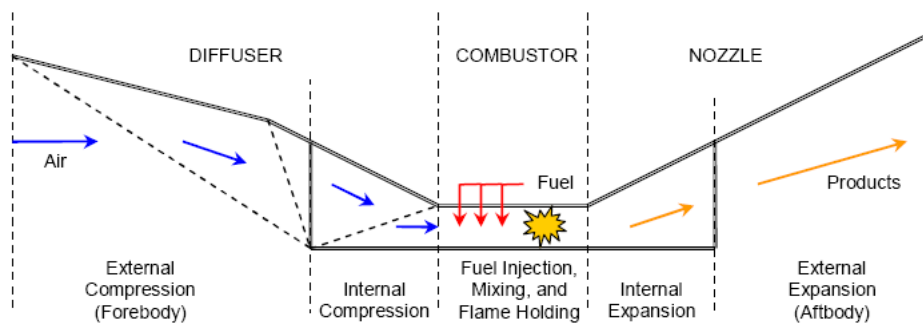


Figure 1.3 Scramjet Engine

1.1.3 Numerical Models

To approximate the complex flowfield associated with scramjet engines, researchers have traditionally turned to discretization of the Navier-Stokes and Euler equations, as well as their derivatives [37]. The Navier-Stokes and Euler equations form the foundation of the numerical modeling approach called Computational Fluid Dynamics (CFD). The Navier-Stokes equations describe the fluid motion of gases and liquids, taking into account fluid momentum, viscosity, and other important factors [31]. Removing the viscous portions of these equations yields the Euler equations, which are more applicable to hypersonic flight. These equations and their derivatives are applied within a numerical model along with appropriate assumptions. Discretization of the solutions at a finite number of points is necessary, since solutions to these systems of differential equations do not exist in closed form and can only be approximated [31].

The simplest CFD models are 1-D versions that can be quickly solved to sufficient accuracy with a stiff ordinary differential equation solver using backward or central finite differences. Higher order 2-D and 3-D models are far more computationally expensive, requiring supercomputers, until only recently. Recent advances in computing have shown efficient performance with 3-D CFD models using massively parallel hardware platforms of 128–512 processors [70].

The scope of this investigation is restricted to 1-D numerical models. Access to high fidelity CFD algorithms, supercomputers, or massively parallel hardware platforms was not available, due to time and funding constraints. Furthermore, the results must be directly comparable to previous work with 1-D models. This previous work by Payne [56] used the code JETPEN, developed by Schetz and Billig at the Johns Hopkins University Applied Physics Lab [59]. JETPEN is a 1-D numerical model of the complex flow field resulting from the transverse injection of a gas jet from a wall into a supersonic or hypersonic cross flow [17]. This model effectively

simulates the gross features of the transverse method of fuel injection into a scramjet combustor for analysis with reliable accuracy at a reasonable computational cost.

The level of reliability and low computational cost of JETPEN allowed it to be one of the first computational models studied for optimization [56]. The highly interactive and discontinuous nature of hypersonic flows makes identification of optimal engineering designs difficult. Payne [56] successfully incorporated JETPEN into several optimization methodologies in an attempt to identify an optimal design for the Air Force Research Lab (AFRL), Propulsion Directorate. His study showed that improved designs could be returned by both Evolutionary Algorithms (EVAs) and Response Surface Methodology (RSM) with a relatively small number of function evaluations, when compared to other methods, such as sequential quadratic programming [56].

1.1.4 Optimization

Since Payne's work [56], advances in computing have allowed optimization techniques to be applied to higher dimensional 2-D and 3-D numerical scramjet models. The use of EVAs and RSM is found throughout the literature. Both have consistently returned improved designs when applied to scramjet injection array, inlet, combustor, and vehicle designs. However, to date no provably convergent algorithms have been applied to scramjet optimization problems.

Evolutionary algorithms are local search algorithms that fall into the category of global search heuristics. They are used with the purpose of identifying globally optimal solutions to difficult problems. These algorithms utilize user-defined functions that are based on the genetic principles of inheritance, mutation, and selection. Design variables are modeled as genes, and the initial population consists of randomly generated points within the design space. After fitness evaluation of these points, the best solutions are selected and "crossed" via a defined routine, producing a new generation of solutions. This new generation of solutions is then evaluated for fit-

ness and the process repeats. Within this cyclic process are random components that simulate genetic mutation.

The difficulty with evolutionary algorithms is in their lack of convergence theory and computational efficiency. While popular in the literature and generally accepted as convergent to the global optimal, there is no mathematical proof to date that proves this assumption. Further compounding this is that they are generally computationally inefficient, requiring many generations of solutions to converge, if they converge at all. In order to compensate for this, several studies have investigated GAS with a population size of 25 or fewer with good results [56]. Several search augmentations, such as gradient-based methods, have also been blended with GAS. Within the context of multi-dimensional CFD models, GAS would likely come at a computationally prohibitive cost for many studies, due to the excessive number of CFD design evaluations required for convergence.

Another method of scramjet design optimization found in the literature is Response Surface Methodology (RSM). Several optimization studies have used RSM as the foundation for analysis. RSM attempts to iteratively fit a local statistical model to an orthogonal set of sample points within the feasible design space. Typically, the fitted statistical models are first or second-order, which lend themselves well to optimization. However, for the optimal solutions of these models to lie close to the global optimizer, the original orthogonal sampling of points must be within the neighborhood of the global optimum. If not, added computational cost of sequential experiments is necessary to identify the correct neighborhood. Payne [56] showed that RSM techniques returned near-optimal solutions with an order of magnitude fewer function evaluations when compared to other techniques. The drawback to this is that it took hundreds of function evaluations to identify the neighborhood of the optimal region on which to center the RSM design.

The derivative-free class of Generalized Pattern Search (GPS) algorithms has been shown to be stationary point-convergent under a set of mild assumptions [68].

GPS has been extended into the domain of mixed variables, or the domain containing both categorical and continuous variables, without sacrificing convergence properties. Sriver [61] applied a modification of the Mixed Variable Pattern Search (MVPS) algorithm to stochastic response models using Ranking-and-Selection. Dunlap [28] investigated different forms of surrogate fitting within Sriver's algorithm such as Kriging and Nadaraya-Watson kernel regression. Currently, MVPS is the only provably convergent algorithm for the class of optimization problems that involves mixed variables.

1.2 Purpose of Research

The purpose of this research is to apply a rigorous optimization algorithm to the engineering design problem of scramjet injection array design. The 1-D scramjet analysis model used in this investigation has not previously been studied in this context. Other studies involving RSM and local search heuristics will provide a basis of comparison for the performance of MVPS. Ideally, the preliminary design identified by the optimization will be further analyzed using higher order CFD models. Unfortunately, time and monetary restrictions preclude this as an option in this investigation. Subsequent high order CFD analysis is proposed as follow-on work.

This investigation applies MVPS and an updated version of JETPEN to Payne's previous scramjet design and a new design from the HiFIRE program. Payne's scramjet design is used to compare the performance of MVPS to previously applied optimization techniques. The best performing optimization techniques on Payne's design are carried forward for use in optimization of the HiFIRE design. Recommendations are made for future designs and research based upon the results. The injection array design, decision variables, constraints, and supporting materials are provided by AFRL/RZAS.

The rest of this document is divided into five different chapters. Chapter 2 focuses on the underlying physics of scramjet fuel injection and discusses JETPEN estimation techniques for transverse fuel injection. Additionally, several optimization techniques, with emphasis on MVPS, are covered in Chapter 2. Chapter 3 details the design variables for the two scramjet as well as the measures of performance by which they will be judged. Chapter 4 outlines the specifics of each optimization technique and details how the performance of each technique is assessed. Chapter 5 contains the results of this research, with conclusions and future recommendations discussed in Chapter 6.

2. Relevant Literature

2.1 Overview

The intent of this chapter is not to develop a comprehensive review of the topics at hand, but to develop a basic working knowledge. First, the various methods of fuel injection are discussed with emphasis on transverse injection techniques. Particular attention is paid to developing a basic understanding of the underlying physics. Second, a well-known numerical simulation used in prior studies for scramjet analysis is presented and discussed. Finally, an overview of relevant optimization techniques and their use in literature is presented, followed with a more detailed description of the MVPS optimization algorithm, which is a main focus of this study.

2.2 Transverse Fuel Injection

Efficient fuel injection is a difficult problem in scramjet engine design. The primary purpose and challenge of any scramjet fuel injection system is to achieve proper penetration and mixing of fuel and air within a reasonably sized combustion or mixing chamber. At supersonic speeds common within scramjet engines, the residence time of any air and fuel particle within the engine is extremely short. Thus mixing must take place in an equally short timespan. Proper mixing is generally considered to be that mass ratio of air to fuel in which the mixture is chemically balanced, called the *stoichiometric ratio*. Plainly put, the stoichiometric ratio is the mass ratio of air to fuel where there is just enough available oxygen to completely burn all of the available fuel. This section focuses on this study's primary fuel injection technique: transverse fuel injection. Many other techniques are investigated throughout the literature; however, since they are not considered in this study, they are only briefly discussed.

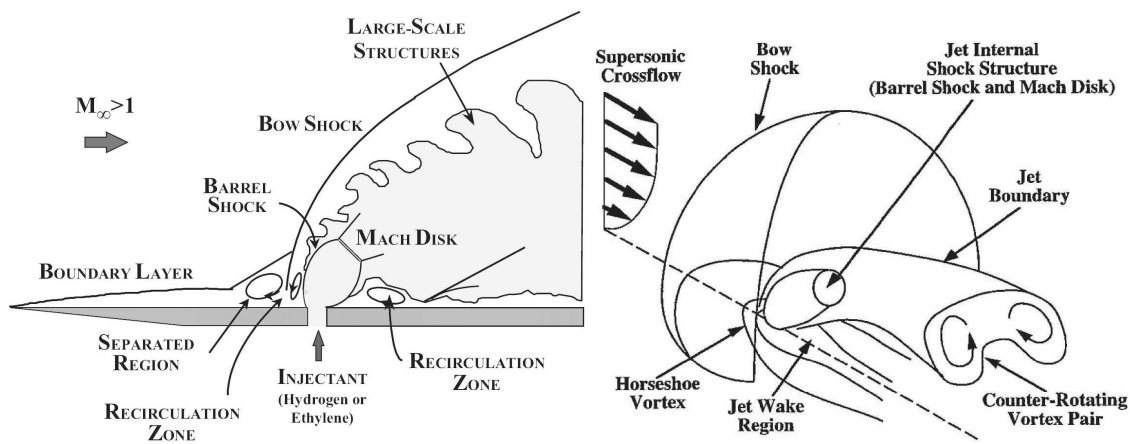


Figure 2.1 Transverse Fuel Injection

Transverse fuel injection involves fuel injected from a flush wall port into a crossflow. A visual representation and the associated complex flowfield is shown in Figure 2.1 [32] [14]. The sonic or supersonic fuel jet acts as an obstruction to the crossflow and a system of strong shocks develops. The fuel jet exits the injection port and begins to be turned by the crossflow. Often the fuel jet is underexpanded and begins to increase in diameter as it enters the freestream. The jet remains mostly intact until a normal shock, called the Mach disk, forms in the jet and turbulent mixing begins. As mixing begins two counter-rotating vortices form and the plume takes on a horseshoe shape. The shockwaves induce significant freestream dynamic pressure losses in the combustor [55]. Freestream higher freestream dynamic pressures are essential to scramjet engine efficiency and freestream dynamic pressure losses adversely affect the resulting performance of the scramjet engine.

Several studies of sonic liquid and gas jets transversely injected into a high-speed cross-flow from ports of various shapes and configurations are found in the literature. The most basic case is that of a single port centrally located along the combustor that injects a sonic gaseous fuel normal to the crossflow. Several studies, such as those by McClinton *et al.* [48], and Srinivasan [60], investigate the effects of different modifications to this basic design as a means for improving mixing. These modifications involved varying the injection angle, number and shape of the injection

array, and injectant dynamic pressure, to name a few. In essence, the need to balance mixing and pressure losses clearly exists for any transverse injection array.

2.2.1 Injection Angle

The angle of injection can greatly impact fuel mixing and engine performance. Normal injection of fuel into the crossflow as shown in Figure 2.1 can produce reasonable mixing [54]. However, this jet acts as a significant obstruction to the crossflow and the path of the fuel jet must be turned 90° to the crossflow. Again, these combined effects contribute to significant pressure losses. Ortwerth [54] references research done at the NASA Langley Research Center in developing the working formula for determining mixing efficiency (η_m):

$$\eta_m = 1.01 + 0.176 \ln \frac{X}{X_\varphi} \quad (2.1)$$

$$S = 60W \quad (2.2)$$

$$\varphi = \frac{H}{W}. \quad (2.3)$$

Efficiency, η_m , defined at values of φ are:

$$\eta_m = \begin{cases} \frac{\varphi_{mix}}{\varphi}, & \text{if } \varphi \leq 1 \\ \varphi_{mix}, & \text{if } \varphi > 1 \end{cases} \quad (2.4)$$

$$X_\varphi = \begin{cases} 0.179e^{1.72}, & \text{if } \varphi \leq 1 \\ 3.333e^{-1.204}, & \text{if } \varphi > 1. \end{cases} \quad (2.5)$$

These working equations apply for normal injection techniques in a rectangular combustor of length X , height H , and width W , using injector spacing S and aspect ratio φ [54].

In contrast to normal injection, wall injection introduces fuel parallel to the crossflow behind a step in the combustor, as in Figure 2.2. This type of injection minimizes dynamic pressure losses while providing film cooling to the combustion

walls [55]. Wall injection relies completely on turbulent effects for mixing, a comparatively slow process that typically results in prohibitively long combustion chambers [36]. The working mixing equation for parallel injection techniques cited by Ortwerth [54] is,

$$\eta_m = \frac{X}{X_\varphi} \quad (2.6)$$

where all the assumptions and variables are as defined previously.

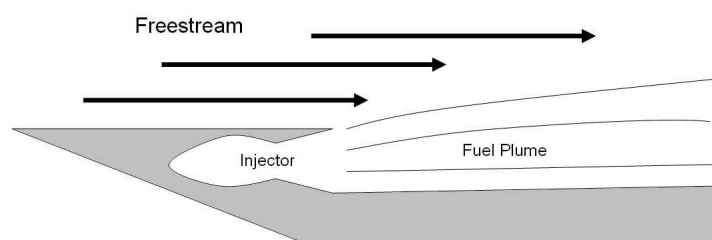


Figure 2.2 Wall Fuel Injection

Injection angles other than 90° have been investigated by McClinton [48] who concluded that decreasing the injection angle increased penetration of the fuel plume into the freestream. Low injection angles were studied by Mays, Thomas, and Schetz [47] and found to be highly influenced by injectant dynamic pressure. In particular, they demonstrated injection at 15° with equivalent penetration and mixing as that of normal injection. Ortwerth recommends linear interpolation between the two mixing rules defined in (2.6) and (2.1) for injection angles between 0° and 90° [54]. In general, the rule of thumb has been that penetration and mixing of fuel into the crossflow is fairly indifferent to injection angles greater than 15° [55].

2.2.2 Array Design

Several studies have been performed on the effect of utilizing multiple laterally-spaced injectors instead of a single centrally located injector. Morgan [49] compared the efficiencies of a single central injector to four laterally spaced injectors. Two sets of two injectors were placed across from each other on parallel walls of the combustor

and the injector diameters were chosen so that the mass flow rate of fuel through the four would be comparable to that of the single injector. This multiple-injector design showed significantly improved mixing properties.

In contrast, some studies involving a large number of small-diameter injectors showed a significant lack of mixing. When the diameter of the holes increased and the number of holes decreased, improved mixing resulted [55]. This finding is in agreement with multiple sources cited by Kutschenreuter [40], where penetration and mixing was found to improve as injector diameter increased, the result of increased flow per injector. A rule of thumb proposed by Anderson [10] suggests that the injection ports should be separated by twice the required penetration distance.

2.2.3 Injectant Dynamic Pressure

Experiments by Schetz and Billig [59] determined that the ratio of injectant to freestream dynamic pressures (\bar{q}) greatly influences mixing and penetration. This ratio is given by

$$\bar{q} = \frac{\rho_{fuel} V_{fuel}^2}{\rho_{air} V_{air}^2},$$

where ρ_{air} and ρ_{fuel} are the densities of fuel and air, respectively, and V_{fuel} and V_{air} are the respective velocities of fuel and air [40]. They found that penetration of the fuel plume into the freestream increased with \bar{q} by forcing the Mach disk farther into the freestream. Experimental data referenced by Kutschenreuter [40] demonstrates that penetration from normal injection of a lateral array of circular injectors correlates well with the expression,

$$\left(\frac{Y}{d^*}\right)^2 = 16N\bar{q},$$

where Y is the Schlieren-determined penetration height, d^* is the injector diameter, and N is the total number of injectors [40].

This suggests that penetration may be increased by increasing either the velocity or density of the fuel stream. An increase in velocity has the negative effect of allowing less time for mixing with the freestream, as well as requiring sharper turning of the jet by the freestream. An additional negative effect is higher induced pressure losses in the freestream [59]. Increases in fuel density have the negative effect of increased resistance to mixing with the freestream. Thus, it can be concluded that the mixing rate is inversely proportional to \bar{q} [72].

2.3 Other Injection Techniques

Several other fuel injection strategies other than transverse injection have been studied in the literature. It is important to note that the JETPEN software is currently incapable of predicting the resulting flowfield of these types of injection. Two common techniques are briefly presented here for the purpose of completeness.

2.3.1 Ramp Injection

Injection of fuel at the base of a low degree ramp, as shown in Figure 2.3, is another extensively researched technique. The fuel is injected *axially*, or with the direction of the freestream air, and is dependent on “vortex shedding from the corners, step-type recirculation behind the after surfaces, and impingement of the reflected ramp shock just after the injector” for mixing [40]. Experimental data has shown that the swept-ramp type injection when combined with downstream normal transverse injection, as shown in Figure 2.3, has mixing efficiency roughly equivalent to that of normal transverse injection [40]. The presence of wide in-stream structures provides improved flame holding characteristics not seen in pure transverse injection. Consequently, these in-stream ramps contribute to pressure and momentum losses in the combustor.

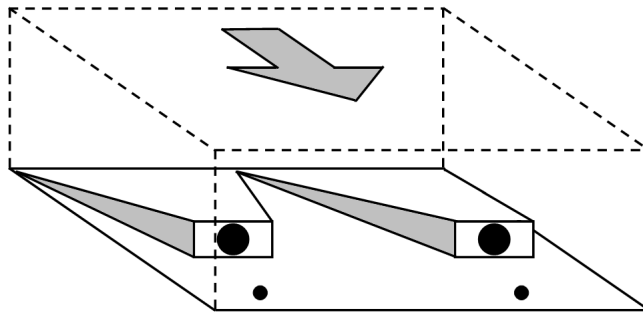


Figure 2.3 Ramp Fuel Injection

2.3.2 Pylon Injection

Though referred to by many names in the literature, pylon injection is essentially injection behind a tall, narrow in-stream body, such as shown in Figure 2.4. Injection may be axial, normal, or at some other angle relative to the freestream. Many shapes and angles of injection have been investigated. Vinogradov *et. al.* [71] experimented with gaseous fuel injection far upstream behind a swept, thin pylon with a various cross sectional pylon shapes. The results showed much improved mixing and penetration, improved flame holding, and a lack of pressure losses and pronounced edge shocks. These results are not typical of earlier work referenced by Paull and Stalker [55], where an advantageous system of shocks from the pylon helped improve mixing but at the sacrifice of pressure losses.

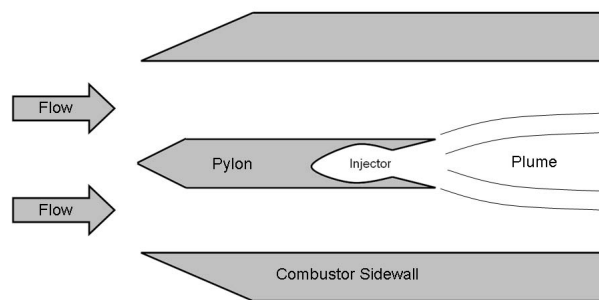


Figure 2.4 Central Pylon Fuel Injection

2.4 *JETPEN Numerical Simulation*

Originally developed at Johns Hopkins University, JETPEN is an analysis tool for the injection of gaseous jets into a supersonic crossflow. It assumes injection is from evenly spaced flush circular wall ports into a high-speed crossflow. The purpose of JETPEN is to analyze the general characteristics of the resulting complex flowfield at a minimal computational cost with reasonable accuracy. JETPEN estimates the trajectory of the plume, location and size of the Mach disk, and the downstream average characteristics of the flowfield. At the time of JETPEN's development alternative CFD based codes could estimate the same measures but were computationally and monetarily expensive [17].

JETPEN is limited to only these estimates of the flow properties. It cannot estimate the pressure losses or complex plume behavior resulting from such flows. As a result, it is limited to use as a preliminary design tool. Detailed analysis using higher fidelity CFD codes is necessary to estimate the overall design performance. However, the accuracy and low computational cost of JETPEN makes it an attractive tool to investigate large design regions or compare a large number of competing designs.

2.4.1 *Mach Disk Estimation*

Mach disk location and size estimation are important parameters in determining the penetration of the jet into the freestream. JETPEN makes several assumptions in determining the Mach disk. The jet is assumed to be intact and up to the Mach disk flow is assumed to be *isentropic*, meaning no energy is added to the jet from the freestream and no energy is lost in the jet due to friction or dissipation. No mixing of the fuel and air is allowed to occur downstream until after the Mach disk, which is justified by the extremely short axial distance from injection to the Mach disk. This is especially true for the common underexpanded case, where the pressure of

the jet exceeds that of the surrounding freestream, and the jet expands outward as it enters the freestream. An “effective back-pressure” of the freestream on the jet is modeled with an average of the freestream static pressure and predictions for the pressure on inclined bodies based on Newtonian impact theory [17]. The Mach disk center location, empirically calculated in [16], is given by

$$\frac{y_1}{d_j^*} = M_j^{\frac{1}{4}} \left(\frac{p_j^*}{p_{eb}} \right)^{\frac{1}{2}} \quad (2.7)$$

$$\frac{x_1}{y_1} = 1.25 \left[1 - \exp \left(-\frac{M_a}{M_j} \right) \right]. \quad (2.8)$$

Here x_1 and y_1 represent the Mach disk center coordinates. M_j and M_a are the Mach numbers of the fuel jet and freestream respectively, and p_j and p_{eb} are the jet total pressure and effective back-pressure, respectively. The Mach disk area, or equivalently the area of the fuel plume, also empirically modeled in [17], is

$$\left(\frac{A_1}{A_j} \right)^{\frac{1}{2}} = 1 + 1.45 \ln \left(\frac{p_j}{p_{eb}} \right) \left[1 - \exp \left(\frac{-0.322y_1}{d_j^*} \right) \right] \quad (2.9)$$

where A_1 is the plume cross-sectional area just before the Mach disk, and A_j is the jet area as it exits the injector. Mach number just after the Mach disk can be modeled [17] when coupled with the mass continuity principle, as

$$M_2 \sqrt{1 + \frac{\gamma - 1}{2} M_2^2} = M_j \sqrt{1 + \frac{\gamma - 1}{2} M_j^2} \left(\frac{p_j}{p_{eb}} \right) \left(\frac{d_2}{d_j^*} \right)^{-2}, \quad (2.10)$$

where M_2 is the Mach number of the fuel plume just after the Mach disk. γ is the ratio of specific heats, d_2 is the diameter of the plume just after the Mach disk, and d_j^* is the injector diameter. The angle of plume trajectory δ_1 is also empirically estimated by

$$\delta_1 = \delta_j - \left(\frac{q_a}{q_j} \right)^{\frac{1}{4}} \left(\frac{180}{\pi} \right) \sin(\delta_j), \quad (2.11)$$

where δ_j is the fuel injection angle, and q_a and q_j are the respective dynamic pressures of the main flow and of the fuel jet. These relationships are sufficient to empirically estimate the size and location of the Mach disk, as well as the plume speed and direction immediately after the Mach disk.

2.4.2 Governing Behavior

The fluid dynamics principles of mass continuity, momentum conservation, and energy conservation model the jet plume behavior within the primary stream. Momentum conservation is broken into both the normal and streamwise components. Species conservation of the interacting gasses is also modeled, which neglects any effects of dissociation or combustion of the gases. Additionally, an entrainment relation derived for high-speed flow is added [17]. The explicit relationships modeled in JETPEN are as follows:

Mass Continuity:

$$\frac{\frac{d(\dot{m})}{\dot{m}_j}}{d\left(\frac{s}{d_j^*}\right)} = \frac{\left(\frac{d}{d_j^*}\right)}{\pi} \left(\frac{p_a}{p_j^*}\right) \left(\frac{U}{U_j^*} - \frac{U_a}{U_j^*}\right) E^* \quad (2.12)$$

Normal Momentum Conservation:

$$\frac{d\delta}{d\left(\frac{s}{d_j^*}\right)} = -\frac{\frac{2}{\pi} C_D \left(\frac{q_a}{q_j^*}\right) \left(\frac{d}{d_j^*}\right) \sin^2(\delta)}{\left(\frac{p}{p_j^*}\right) \left(\frac{U}{U_j^*}\right)^2 \left(\frac{d}{d_j^*}\right)^2} - \frac{\left(\frac{U_a}{U_j^*}\right) \sin(\delta) \left[\frac{d\left(\frac{\dot{m}}{\dot{m}_j}\right)}{d\left(\frac{s}{d_j^*}\right)}\right]}{\left(\frac{p}{p_j^*}\right) \left(\frac{U}{U_j^*}\right)^2 \left(\frac{d}{d_j^*}\right)^2}, \quad (2.13)$$

$$C_D = \begin{cases} 1.06 + \frac{1.14}{[M_a \sin(\delta)]^3}, & \text{if } M_a \sin(\delta) \geq 1.0 \\ 1.20 + [M_a \sin(\delta)]^{3.5}, & \text{if } M_a \sin(\delta) < 1.0 \end{cases} \quad (2.14)$$

Streamwise Momentum Conservation:

$$\frac{d\left(\frac{U}{U_j^*}\right)}{d\left(\frac{s}{d_j^*}\right)} = \frac{\left[\left(\frac{U_a}{U_j^*}\right) \cos(\delta) - \left(\frac{U}{U_j^*}\right)\right] \left[\frac{d\left(\frac{\dot{m}}{\dot{m}_j}\right)}{d\left(\frac{s}{d_j^*}\right)}\right]}{\left(\frac{\dot{m}}{\dot{m}_j}\right)} - \frac{\left(\frac{d}{d_j^*}\right)^2 \left(\frac{q_a}{q_j^*}\right) \left[\frac{dp_{avg}}{d\left(\frac{s}{d_j^*}\right)}\right]}{\left(\frac{\dot{m}}{\dot{m}_j^*}\right)} \quad (2.15)$$

Species Conservation:

$$\alpha = \frac{\dot{m}_j}{\dot{m}} \quad (2.16)$$

Energy Conservation:

$$T_T = \frac{\alpha c_{pj} T_{Tj} + (1 - \alpha) c_{pa} T_{Ta}}{c_p} \quad (2.17)$$

Entrainment:

$$E^* = \frac{0.4}{\left(\frac{U_2}{U_a}\right)^{0.6}} \left(\frac{s}{d_j^*}\right)^{1.37} \quad (2.18)$$

Table 2.1 defines the parameters used in (2.12)–(2.18). The restriction $s/d_j^* \leq 10$ is placed on the entrainment model due to the lack of experimental data beyond this point [17]. The model itself is an extension of an empirically developed low-speed model to high-speed flow [17]. As shown in Figure 2.5, each plume is modeled as having an expanding circular cross-section and entrainment is allowed to occur only at the exposed area. When adjacent plumes begin to touch, this area is reduced and JETPEN only allows for entrainment to occur at the plume area in contact with the freestream and not at the area where adjacent plumes merge. This area is now essentially a series of curved circular arcs, as shown in Figure 2.5.

At the heart of JETPEN is a stiff Ordinary Differential Equation (ODE) solver. This solver marches downstream estimating solutions to the system of stiff equations in (2.12)–(2.18). The ODE solver uses a backward difference formula, also known as Gear’s method, to approximate a full Jacobian matrix and can choose a variable

Table 2.1 Parameter Descriptions

Parameter	Description
\dot{m}_j	Fuel mass flow rate
p_a	Main flow pressure
p_j	Fuel jet exit pressure
U_a	Main flow velocity
U_j	Fuel jet exit velocity
s	Arc length along the trajectory
E^*	Entrainment model response
δ	Angle of the plume trajectory
C_D	Drag coefficient
p_{avg}	Plume average pressure
α	Mass fraction of injected species
T_{T_a}	Main flow static temperature
T_{T_j}	Fuel jet exit static temperature
c_{pa}	Main flow specific heat capacity @ const. press.
c_{pj}	Fuel specific heat capacity @ const. press.

stepsize to control error. Error tolerances are set at 10^{-4} , initial stepsize at 0.5, and a limit of 4000 steps is enforced.

2.4.3 Validation

Billig and Schetz [17] compared JETPEN's results to available experimental data to verify its computations. Analysis is based on the decay α of fuel concentration in the plume across several key parameters. The values for key parameters of the experimental data are summarized in Table 2.2. Much of the experimental data is sparse in coverage of the experimental space and only reports the maximum concentration α_{max} , while JETPEN estimates the average concentration α_{avg} . In light of these limitations, JETPEN estimates the available data quite well. Figure 2.6 shows the JETPEN estimates compared to the experimental data at $M_a = 3.0$. The predicted values for α_{avg} lie approximately 40% below the experimental data for α_{max} [17]. Figure 2.7 shows the comparison of the predicted and measured plume cross-

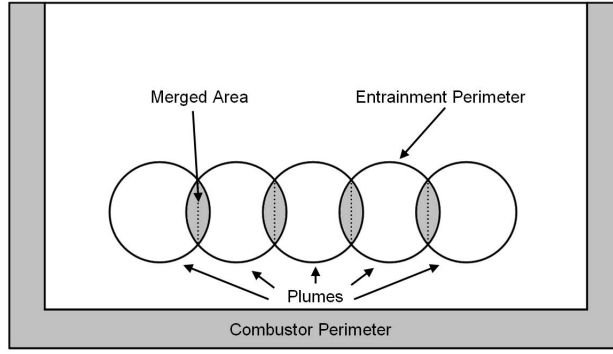


Figure 2.5 Merged Adjacent Plumes

section at $x/d_j^* = 80$, $M_a = 6.0$, $\delta_j = 15^\circ$, and $w/d_j^* = 9.0$ [17]. In general, JETPEN is accepted as sufficiently accurate in its predictions.

Table 2.2 Parameter Values for Experimental Data Comparisons

Parameter	Lower Value	Upper Value
M_a	1.4	6.0
M_j	1.0	1.7
w/d_j^*	6.25	∞
δ_j	15	90
\bar{q}	1.0	Large Values

2.5 Simulation Optimization

The optimization problem considered in this research is

$$\min_{x \in \Omega} f(x) \tag{2.19}$$

where $f : \mathbb{R}^n \cup \mathbb{Z} \rightarrow \mathbb{R} \cup \{\infty\}$ is considered computationally expensive to evaluate, $\Omega = \{x \in \mathbb{R}^n \cup \mathbb{Z} : \ell \leq x \leq u\}$ and $\ell, u \in \mathbb{R}^n$ with $\ell < u$. The objective function f will be treated as a “black box”, meaning only the response of the function is of interest and information on the inner workings of the function, such as derivative information, is either un-used or unavailable. The response of this function f may

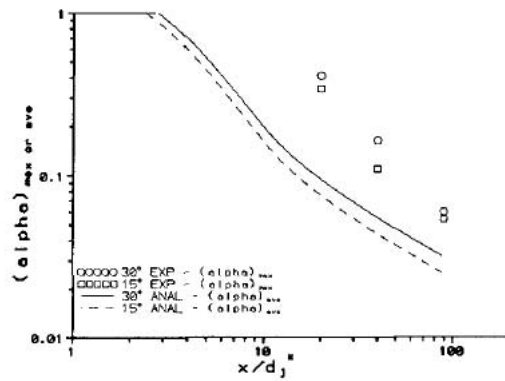


Figure 2.6 Comparison of Predicted vs. Experimental Data

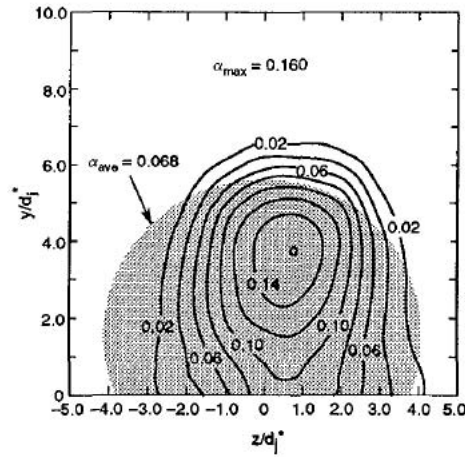


Figure 2.7 Predicted vs. Measured Plume Cross-section

exhibit properties to include nonsmoothness, discontinuity, and may fail to return a value for $x \in \Omega$.

In this research the simulation, namely JETPEN, is used to return function evaluations for f . JETPEN is the main analysis tool for the engineering design problem of scramjet fuel injection array design. The solution to this problem cannot be found empirically or through traditional nonlinear math programming techniques. Several familiar and new optimization techniques are applied to the problem (2.19).

2.5.1 Evolutionary Algorithms

Evolutionary algorithms are one of the most common optimization techniques applied to solve difficult engineering design optimization problems. Evolutionary algorithms are based on heredity and the Darwinian principle of “survival of the fittest.” The basic evolutionary strategy for continuous parameter optimization developed by Rechenberg and Schwefel [50] is outlined in Figure 2.8.

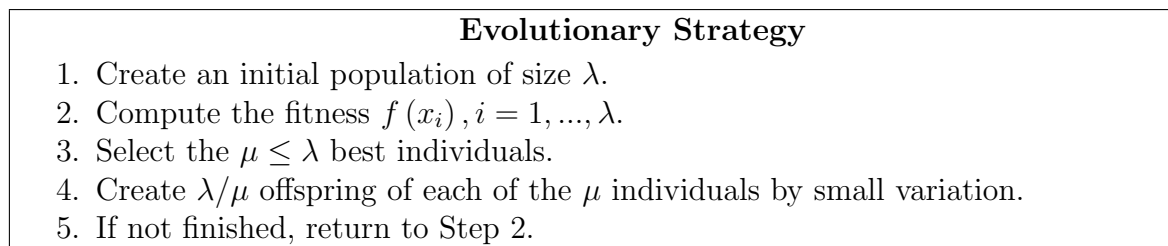


Figure 2.8 Evolutionary Strategy

A subclass of the evolutionary algorithms are genetic algorithms (GA). Genetic algorithms attempt to emulate reproduction through processes based on fitness, selection, recombination, genetic representation, and mutation [50]. A Simple Genetic Algorithm (SGA) first developed by Holland, is shown in Figure 2.9 [50]. Genetic representation is typically done through bitstring representation of a chromosome, where the positions on the bitstring represent the loci of the chromosome [50]. The value (allele) of a particular variable (gene) is held in its locus [50]. All the chromosomes make up the genotype, which in turn defines the phenotype [50].

Recombination, also called cross-over, combines two parent strings into a subsequent child or children that make up part of the next generation. *Mutation* randomly operates on any bit of the bitstring with a given probability [50]. In essence, genetic algorithms are parallel random searches with central control through the selection schedule, based on the average fitness of each generation [50].

Examples of genetic algorithm applications in scramjet research exist throughout the literature. Markell [46] applied a commercially available evolutionary algorithm in the optimization of a total vehicle design. He concluded that the optimizer

Simple Genetic Algorithm

1. Define a genetic representation of the problem, commonly a bitstring.
2. Create an initial population $P(0) = \{x_1^0, \dots, x_N^0\}$. Set $t = 0$.
3. Compute the average fitness $\bar{f}(t) = \sum_i^N f(x_i)/N$. Assign each individual its normalized fitness value $f(x_i)/\bar{f}(t)$.
4. Assign each x_i a probability $p(x_i, t)$ proportional to its normalized fitness. Using this distribution, select an even number of N vectors from $P(t)$. This gives the set of selected parents.
5. Pair all parents at random forming $N/2$ pairs. Apply crossover with a certain probability to each pair and other genetic operators, such as mutation, forming a new population $P(t + 1)$.
6. Set $t = t + 1$ and return to Step 3.

Figure 2.9 Simple Genetic Algorithm

successfully converged to the local area of the optimum, but gradient-based augmentation of the search strategy would have helped reduce the 40,000 required function evaluations [46].

Foster *et al.* [29] used a hybrid genetic algorithm with a gradient-based search to optimize several aerodynamic shapes in hypersonic flow based on Newtonian flow theory. They reported improved convergence over strictly evolutionary and empirical-based optimization methods. While not directly used in their optimization, CFD was used to verify the characteristics of the finalized designs [29].

Chernyavsky *et al.* [21] also applied gradient-based augmentation to a genetic algorithm in their optimization of a three-dimensional scramjet inlet. They used an algorithm developed by Rasheed [21] that only applies gradient-based augmentation in the final stages of optimization. This algorithm also includes several modifications for faster convergence in engineering design problems in continuous space [21]. Optimization was applied to a one-dimensional flow solver and converged after approximately 16,000 function evaluations with minimal improvement after roughly 1500 function evaluations [21]. This suggests that the gradient-based augmentation may not have been as effective as they originally hoped.

2.5.2 *Micro-Genetic Algorithms*

Krishnakumar [39] developed a genetic algorithm based around a small population size and the basic principles of evolutionary algorithms, called the *Micro-Genetic Algorithm* (μ GA). This approach showed superior convergence to the optimum local area over SGAs. It also proved superior in the presence of multi-modality and non-stationary function optimization, or functions that change over time [39].

His μ GA is outlined in Figure 2.10. Several key differences, aside from its small population size, exist that differentiate it from SGAs. Mutation is excluded from the algorithm under the assumption that enough diversity is introduced into the population through the random selection of strings at every new generation [39]. Cross-over is done deterministically instead of stochastically, as is typical in the SGA.

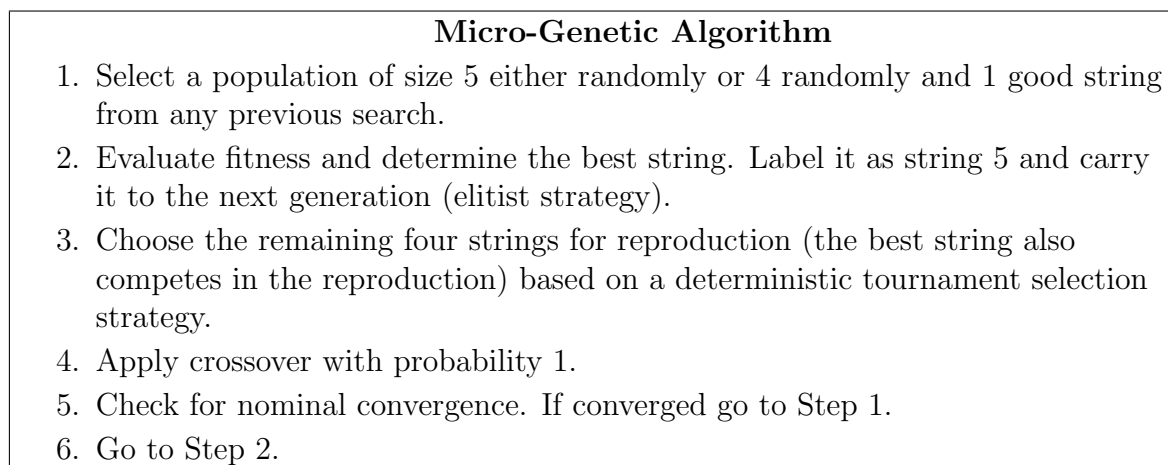


Figure 2.10 Micro-Genetic Algorithm

Krishnakumar [39] applied the μ GA to several test functions and found it significantly reduced the required function evaluations for convergence when compared against the SGA. Payne [56] reported nearly a four-fold reduction in required function evaluations, needing only 159 evaluations as compared to the SGA's 580. While Payne's work clearly proved its potential, it is the only known application of a μ GA to hypersonic design optimization. Applications of μ GA exist in other disciplines, and several variants have been introduced as a result.

2.5.3 Response Surface Methods

Another technique common in hypersonic design optimization is Response Surface Methodology (RSM). RSM “is a collection of statistical and mathematical techniques useful for developing, improving, and optimizing processes” [51]. These methods were first introduced by Box and and Wilson [18] in the 1950s and commonly involve the estimation of complex processes through the fitting of first and second-order regression polynomials. These fitted models use the response points from carefully constructed experimental designs which then can be optimized through basic techniques such as stationary point or ridge analysis.

The polynomial model

$$\hat{\mathbf{y}} = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n + \beta_{12} x_1 x_2 + \dots + \beta_{11} x_1^2 + \dots + \beta_{nn} x_n^2 + \varepsilon \quad (2.20)$$

$$\vec{\mathbf{b}} = (\beta_0, \beta_1, \dots, \beta_n, \beta_{12}, \dots, \beta_{11}, \dots, \beta_{nn}) \quad (2.21)$$

estimates the expected response $\hat{\mathbf{y}}$ from the experimental design points of the n experimental variables x_1, \dots, x_n . The vector $\vec{\mathbf{b}}$ is the vector of estimated model coefficients, X is the matrix of experimental design points, and ε is the error term. In general, coefficients of response surface models are estimated by the method of least squares, yielding

$$\vec{\mathbf{b}} = (X^T X)^{-1} X^T y, \quad (2.22)$$

where y is the vector of actual experimental responses. The method of least squares has the advantage of providing unbiased estimates for $\vec{\mathbf{b}}$, assuming model adequacy.

A common yet simple experimental design is the 2-level factorial design. Each of the k experimental variables is assigned two levels and run at each for a total of 2^k experimental design points. This design has the special property of orthogonality between the estimated effects. Thus, removal of one experimental factor has no impact on the ability to estimate the others. Two-level factorial experiments and their derivative, the 2-level fractional factorial, are very efficient for screening

experimental variables. A drawback of the 2-level factorial design is that it cannot estimate quadratic or higher order model terms. Three levels for each experimental variable are necessary for this estimation, which causes the necessary experimental runs to increase very quickly. For example, assigning 3 levels to a five-factor experiment requires $N = 3^5 = 243$ runs, compared to $N = 32$ runs for a 2-level 5-factor experiment.

A Central Composite Design (CCD) is a more efficient alternative to a 3-level factorial experiment. A CCD is a spherical, 5-level design that combines a 2-level factorial with center runs and axial runs for a total of $N = 2^n + 2n + 1$ experimental design points. The CCD has been proven to be a robust experimental technique especially when designed with rotatability [51]. A rotatable CCD is one where the prediction variance only depends on the distance, not the direction, from the design center. Here, the scaled prediction variance, $N \text{Var}[\hat{y}(\mathbf{x})]/\sigma^2$, is the same throughout the design region at a given distance from the design center, with σ^2 the process pure error estimate. Rotatability in the CCD requires [51]:

1. All odd moments through order 4 are zero.
2. The ratio of moments $[iiii]/[iijj] = 3$ for $(i \neq j)$

The first condition on rotatability is satisfied as long as the factorial portion of the CCD is a properly chosen full or fractional 2^k factorial design. The second condition on rotatability can be satisfied by:

$$\alpha = \sqrt[4]{N_F} \tag{2.23}$$

where α is the distance from design center of the axial points and N_F is the number of factorial points. The CCD is an efficient design involving a reasonable number of design points for the information returned, which is particularly useful in sequential experimentation [51].

Another alternative to both the 3-level factorial and CCD is the Box-Behnken design (BBD). The BBD is a 3-level spherical design that has experimental points centered on the “edges” of the associated 2-level factorial design. The result is a rotatable, or nearly-rotatable, spherical design that requires fewer runs than a CCD. For example for $k = 3$ experimental variables, the CCD requires $N = 14 + n_C$ runs, while the BBD requires $N = 12 + n_C$ runs [51].

These designs can be built and modified to achieve specific modeling objectives. The most common objective is the minimization of $Var(b_i)$, $i = 1, \dots, k$, or variance-optimality. Variance-optimality is achieved if $(X^T X)^{-1} = N^{-1} \mathbf{I}_n$ for a given design X , where \mathbf{I}_n is the identity matrix. Another common modeling objective is D-optimality, which maximizes the determinant of the moment matrix over all possible designs. Box and Draper [51] developed several methods for minimum design bias in the case of an underspecified model. These methods are beyond the scope of this investigation but have been successfully applied in the context of hypersonic design optimization [56].

The backbone of RSM optimization is the method of steepest descent. The method of steepest descent applies gradient-based process improvement to the information returned from experimental designs [51], and is outlined in Figure 2.11. Movement along the chosen path, Δ , is proportional to the largest regression coefficient, b_i ,

$$\Delta x_j = \frac{b_j}{b_i / \Delta x_i}, \quad \text{where typically: } \Delta x_i = 1. \quad (2.24)$$

At every subsequent first-order experiment, it is important to test for lack-of-fit to determine if any significant curvature exists in the experimental region. A CCD, BBD, or other spherical design is justified in the existence of significant lack-of-fit and minimal improvement along the selected path. This final experiment will identify the optimal point within the design region by stationary point or ridge analysis. The

stationary point, x_S , can be easily found by

$$x_S = -\frac{1}{2}\hat{\mathbf{B}}^{-1}\mathbf{b} \quad (2.25)$$

where $\hat{\mathbf{B}}$ is the symmetric $k \times k$ matrix of second-order terms, and \mathbf{b} is the vector of first-order terms.

<p>Gradient-Based RSM</p> <ol style="list-style-type: none">1. Fit a first-order model using an orthogonal design.2. Compute a path of steepest descent.3. Conduct experimental runs along the path.4. Run a second experiment at the point of approximate maximum (or minimum) response along the path. This design should again be a first-order model.5. Another direction of steepest descent is computed and more experiments are run. Eventually, the improvement will be diminished enough to warrant a higher-order experimental design, which is the final basis for optimality.

Figure 2.11 Gradient-Based RSM

Several examples of RSM application in hypersonic design optimization exist in the literature. Steffen [64] applied a design called the face-centered CCD, or three-level CCD, in the fuel injection array optimization of NASA's GTX multi-mode propulsion system. Computationally expensive CFD analysis was used as the response with four experimental variables being considered. This computational expense justified the use of RSM instead of more expensive evolutionary techniques. Despite the variance instability inherent in his design, Steffen reported good prediction capability with the second-order response model [64].

2.6 *Pattern Search Methods*

This section covers the methods, theoretical results, and application of Mixed Variable Pattern Search (MVPS). It is a derivative-free optimization technique that evaluates points on a conceptualized mesh. At each iteration, the search is done such that a descent direction will be found, if one exists, while ensuring that the

stepsize is not too small. Under the assumptions of continuous differentiability of the objective function and that all iterates of the algorithm lie in a compact set, certain limit points of the algorithm are shown to satisfy first-order optimality conditions [12], and a pseudo-second-order necessary condition [3]. Methods covered in this investigation are restricted to bound constrained optimization and application is performed through the NOMADm software graphical user interface. The NOMADm software package developed by Abramson [4] is a framework for the application of pattern search methods within MATLAB®.

2.6.1 Generalized Pattern Search

Generalized pattern search was introduced and studied by Torczon [68] for unconstrained optimization on continuously differentiable objective functions. For the purpose of this discussion, the bound constrained optimization problem is defined as in (2.19) where there is no noise, or discrete variables and the values for ℓ and u are permitted to take on values of $-\infty$ and ∞ , respectively, to allow variables to be constrained in only one direction. A “barrier” function, identical to those of Audet and Dennis [13] and Lewis and Torczon [41], $f_\Omega = f + \psi_\Omega$, is applied for each point to be evaluated, where $\Omega = \{x \in \mathbb{R}^n : \ell \leq x \leq u\}$. If a candidate point lies outside the bounded region, then $\psi_\Omega(x) = \infty$ and a function value of $f_\Omega(x) = \infty$ is returned. Otherwise, $\psi(x) = 0$ and $f_\Omega(x) = f(x)$.

The GPS algorithm generates a set of iterates of non-increasing function values. Each iteration has two main steps, an optional SEARCH and a local POLL. The intent is to find a point with a lower objective function value by evaluating the “barrier” function f_Ω at a finite number of points on a mesh defined around the current solution. The size of the mesh is then adjusted depending on whether or not an *improved mesh point* is found. This improved mesh point must be strictly lower in objective function value.

The mesh is conceptually constructed via a set of n_D *positive spanning directions*; *i.e.*, a set of directions such that any vector in \mathbb{R}^n can be expressed as a nonnegative linear combination of directions in the set [25]. A *positive basis* is a positive spanning set such that removing any vector in the set would make it no longer a positive spanning set, having between $n + 1$ and $2n$ directions. For this discussion, the $n \times n_D$ matrix D , whose columns are the n_D positive spanning directions, is required to be constructed as

$$D = GZ, \tag{2.26}$$

where G is a real nonsingular $n \times n$ generating matrix (often chosen as the identity matrix), and Z is an $n \times n_D$ full-rank integer matrix. Thus, each direction $d_j \in D$ can be represented by $d_j = Gz_j$ with integer vector $z_j \in \mathbb{Z}^n$, for $j = 1, 2, \dots, n_D$. The mesh M_k at iteration k [13] is given by

$$M_k = \bigcup_{x \in S_k} \{x + \Delta_k Dz : z \in \mathbb{Z}^{n_D}\}, \tag{2.27}$$

where S_k is the set of points at which f_Ω has previously been evaluated prior to iteration k , and the mesh size parameter $\Delta_k > 0$ controls the mesh fineness. This definition ensures all previous iterates lie on the current mesh, and is consistent with that of Audet and Dennis [13].

The optional SEARCH step evaluates f_Ω at a finite number of points (including none), in an attempt to find an improved mesh point. It may employ any technique such as a few iterations of a heuristic, fitting and optimizing a surrogate function, random sampling, or any other finite technique. The SEARCH step contributes nothing to the convergence theory (see Section 2.6.3) but is completely flexible in its employment by the user. Each of the evaluated points in the SEARCH step are required to lie on the current mesh.

If the SEARCH step fails to find an improved point, then the POLL step is invoked. This step is deliberate in its construction and is necessary for the convergence

theory. In the POLL step, f_Ω is evaluated at mesh points adjacent to the incumbent solution x_k . This set of points, called the POLL set, is given by:

$$P_k(x_k) = \{x_k + \Delta_k d : d \in D_k \subseteq D\} \in M_k, \quad (2.28)$$

where D_k is itself a positive spanning set whose columns are taken from D . The mesh is controlled via the parameter w_k . If the SEARCH or POLL steps return an improved mesh point, then the current iterate is immediately terminated and a new iteration centered around the new incumbent, x_{k+1} , is started. The mesh is then either retained or coarsened. If no improved mesh point is found at either step, then the incumbent is declared a *local mesh optimizer*, and the mesh is then refined and a new iterate begun. The term w_k determines the fineness of the mesh at each iteration with $w^- \leq w_k \leq w^+$ for fixed integers $w^+ \geq 0$ and $w^- \leq -1$. If the mesh is coarsened, then $w_k \in \{0, 1, \dots, w^+\}$, otherwise $w_k \in \{w^-, w^- + 1, \dots, -1\}$. The mesh size Δ_k is then determined by

$$\Delta_{k+1} = \tau^{w_k} \Delta_k. \quad (2.29)$$

An example of three consecutive POLL steps is shown in Figure 2.12. In this case, $x_k \in \mathbb{R}^2$, $D_k = D = \{e_1, e_2, -e_1, -e_2\}$, $\tau = 2$, and $w_k = -1$. The SEARCH step is omitted for simplicity. At iteration k , the incumbent x_k and POLL set is marked by the dots. Notice that each of the POLL points lies in the 4 standard coordinate directions. Assume that no improved mesh point is found. The incumbent is declared a mesh local optimizer and is updated, $x_{k+1} = x_k$. The mesh is now refined with $w_{k+1} = -1$, giving $\Delta_{k+1} = \tau^{w_k} \Delta_k = 2^{-1}(1) = 0.5$. The next iteration, $k + 1$, builds a finer mesh around the incumbent. Again, the POLL points are in the standard coordinate directions but are now half the original distance from the incumbent. Assume that each of the POLL set points are not improved mesh points. Again, the incumbent is declared a mesh local optimizer and is updated $x_{k+2} = x_{k+1}$. The mesh

is refined as $\Delta_{k+2} = \tau^{w_k} \Delta_{k+1} = 2^{-1}(0.5) = 0.25$. A new, finer mesh is constructed around the incumbent and polling takes place again.

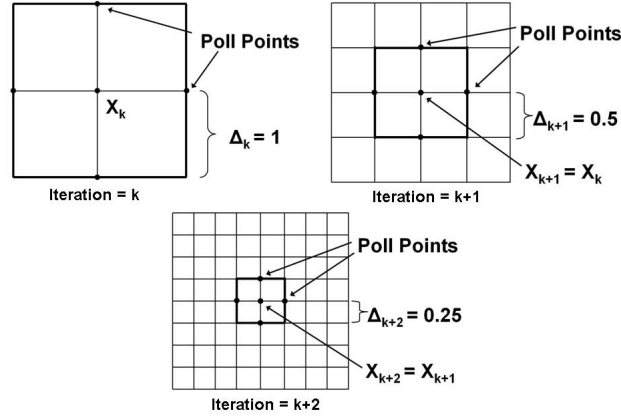


Figure 2.12 GPS Poll Step and Mesh Update

In this fashion the algorithm iterates until the stopping criteria $\Delta_k < \varepsilon$ is reached, where ε is a user defined mesh-size tolerance. A basic algorithm for GPS is outlined in Figure 2.13.

Generalized Pattern Search (GPS)

Initialization : Evaluate the set of initial points S_0 . Define $x_0 \in S_0$ such that $f_\Omega(x_0) \leq f_\Omega(y) \leq \infty, \forall y \in S_0$. Define $\Delta_0 > 0$, M_k as in (2.12), and D as a positive spanning set of n_D directions.

1. **SEARCH step:** Search for an improved mesh point via a finite strategy; *i.e.*, $\ell \leq x_{k+1} \leq u$ such that $f_\Omega(x_{k+1}) < f_\Omega(x_k)$.
2. **POLL step:** If SEARCH step is unsuccessful, evaluate f_Ω at points in the poll set $P_k(x_k)$ defined in (2.28) until an improved x_{k+1} is found or until all points in $P_k(x_k)$ are evaluated.
3. **Update:** If the SEARCH or POLL steps find an improved mesh point, then update x_{k+1} and set $\Delta_{k+1} \geq \Delta_k$ as in (2.29). Otherwise, set $x_{k+1} = x_k$ and set $\Delta_{k+1} < \Delta_k$ as in (2.29).

Figure 2.13 Basic GPS Algorithm

2.6.2 Mixed Variable Pattern Search

Often in engineering design problems a variable cannot take on a continuous value, but may be represented by a finite set of numbers. These discrete variables must be handled differently than their continuous counterparts. Branch and bound techniques may be applicable for variables that take on discrete integer values which have some inherent meaning. Often the variables are categorical, such as material type, color, or shape, where the assigned numerical value may hold no inherent meaning. Problems involving both continuous and categorical variables are called *mixed variable optimization problems*.

Audet and Dennis extended GPS to the domain of mixed variables [12]. Mixed variable pattern search is similar to GPS but requires additional function evaluations to account for the discrete variables. The MVPS POLL step evaluates f_Ω at the set of points defined in GPS, but also evaluates f_Ω at a user-defined list of discrete neighbors of the current iterate. Additionally, an EXTENDED POLL step is performed around each discrete neighbor whose objective function value falls within a user-specified amount from that of the current incumbent. Much of the following discussion and definitions that follow come from Abramson [2].

To define the set of discrete neighbors, let x be partitioned into its continuous and discrete components; *i.e.*, $x = (x^c, x^d)$, where $x^c \in \Omega^c$ and $x^d \in \Omega^d$, with $\Omega = \Omega^c \times \Omega^d$. The sets Ω^c and Ω^d represent the continuous and discrete domains, respectively. The set of discrete neighbors is constructed from a set-valued function, $\mathcal{N}: \Omega \rightarrow 2^\Omega$ where Ω represents the entire feasible region and 2^Ω is the power set, containing all possible subsets of Ω . The finite set of discrete neighbors of a point x_k is denoted by $\mathcal{N}(x_k)$ [12]. Thus, a point y is a discrete neighbor of point x_k if $y \in \mathcal{N}(x_k)$, where $\mathcal{N}(x_k)$ is defined by the user. A common choice of discrete neighbors for integer variables is

$$\mathcal{N}(x_k) = \{(x_k^c, y^d) : y^d \in \Omega^d, \|y^d - x_k^d\|_1 \leq 1\}. \quad (2.30)$$

While this definition is applicable to this research, it is more restrictive than that of the general case, where the continuous variables, as well as the bounds ℓ and u , may change depending on the combination of discrete variables. All discrete neighbors are required to lie on the mesh defined by the current iterate; thus, $\mathcal{N}(x_k) \subseteq M_k$ with $\mathcal{N}(x_k)$ finite.

To define the current mesh M_k , the matrix D^i is constructed like D in (2.26) and has the same properties as in the GPS algorithm, with the exception that D^i denotes the positive spanning directions for the i th combination of discrete variable values, $i = 1, 2, \dots, i_{max}$, where i_{max} is the total number of different discrete variable settings. The mesh is the direct product of Ω^d with the union of meshes for each possible combination of categorical variable settings; namely,

$$M_k = \Omega^d \times \bigcup_{i=1}^{i_{max}} \{x_k^c + \Delta_k D^i z \in \Omega^c : z \in \mathbb{Z}^{|\mathcal{D}^i|}\}, \quad (2.31)$$

where $|\mathcal{D}^i|$ is the cardinality of \mathcal{D}^i . The mesh size parameter Δ_k retains the same restrictions as in GPS. The poll set for the continuous variables is then defined

$$P_k(x_k) = \{(x_k^c + \Delta_k d, x_k^d) \in \Omega : d \in \mathcal{D}_k^i \subseteq \mathcal{D}^i\} \quad (2.32)$$

where $\mathcal{D}_k^i \subseteq \mathcal{D}^i$ is the set of positive spanning directions for iterate k at the i th discrete variable combination. It is important to note that the values of the discrete variables do not change from those of the current incumbent, x_k , during this portion of the POLL step. If no improvement is found in $P_k(x_k)$, then the discrete neighbors of x_k , *i.e.* $y \in \mathcal{N}(x_k)$, are evaluated.

If the POLL step fails to find a new incumbent, then the EXTENDED POLL step is executed. The EXTENDED POLL step initiates a POLL step in the continuous variables for all neighbors in $\mathcal{N}(x_k)$ whose function value was sufficiently close to the incumbent function value, *i.e.* the EXTENDED POLL is initiated for each discrete

neighbor that satisfies $f_{\Omega}(x_k) \leq f_{\Omega}(y) \leq f_{\Omega}(x_k) + \xi_k$. The parameter $\xi_k \geq \xi$, for some fixed $\xi > 0$, is called the *extended poll trigger*, and is typically set to a percentage of the objective function value, such as $\xi_k = \max\{0.05f(x_k), \xi\}$. This subset of discrete neighbors $\mathcal{N}_k^{\xi_k}$ is defined by

$$\mathcal{N}_k^{\xi_k} = \{y \in \mathcal{N}(x_k) : f_{\Omega}(x_k) \leq f_{\Omega}(y) \leq f_{\Omega}(x_k) + \xi_k\}. \quad (2.33)$$

The set of EXTENDED POLL centers forms the sequence $\{y_k^j\}_{j=1}^{J_k}$, which begins with $y_k^0 = y_k \in \mathcal{N}(x_k)$ and ends with $z_k = y_k^{J_k}$, where J_k is finite under some mild assumptions. The EXTENDED POLL endpoint z_k occurs when either $f_{\Omega}(z_k^c + \Delta_k^m d, z_k^d) < f_{\Omega}(x_k)$, or when $f_{\Omega}(x_k) < f_{\Omega}(z_k^c + \Delta_k^m d, z_k^d)$ for all $d \in D_k(z_k)$. Thus the entire set of EXTENDED POLL points is given by

$$\chi_k(\xi_k) = \bigcup_{y \in \mathcal{N}_k^{\xi_k}} \bigcup_{j=1}^{J_k} P_k(y_k^j). \quad (2.34)$$

The mesh updating in MVPS is the same as in GPS. The set of trial points at each iteration of MVPS is $T_k = S_k \cup P_k(x_k) \cup \mathcal{N}(x_k) \cup \chi_k(\xi_k)$, where S_k is the set of points evaluated in the SEARCH step. A point x_k is considered to be a mesh local optimizer if $f_{\Omega}(x_k) \leq f_{\Omega}(y) \forall y \in T_k$.

The MVPS algorithm is shown in Figure 2.14, and an example of one iteration is shown in Figure 2.15. The problem shown in Figure 2.15 has one discrete and two continuous variables. The incumbent is x_k and $P_k(x_k) = \{a, b, c\}$. No improved mesh point is found in $P_k(x_k)$ and the discrete neighbors of x_k , $\mathcal{N}(x_k) = \{y_1, y_2\}$, are evaluated with $f_{\Omega}(x_k) < f_{\Omega}(y_1) < f_{\Omega}(x_k) + \xi_k < f_{\Omega}(y_2)$. Since $\mathcal{N}_k^{\xi_k}(x_k) = \{y_1\}$, the algorithm next evaluates $P_k(y_1) = \{d, e, f\}$ with no improved mesh point and the mesh is refined.

Mixed Variable Pattern Search (MVPS)

Initialization : Evaluate the set of initial points S_0 . Define $x_0 \in S_0$ such that $f_\Omega(x_0) \leq f_\Omega(y) \leq \infty, \forall y \in S_0$. Define $\Delta_0 > 0$, M_k as in (2.31), and D^i as a positive spanning set of n_{D^i} directions.

1. **SEARCH step:** Search for an improved mesh point via a finite strategy; *i.e.*, $x_{k+1} \in \Omega$ such that $f_\Omega(x_{k+1}) < f_\Omega(x_k)$.
2. **POLL step:** If SEARCH step is unsuccessful evaluate f_Ω at points in $P_k(x_k) \cup \mathcal{N}(x_k)$ until an improved mesh point is found or all points are exhausted.
3. **EXTENDED POLL step:** Perform a poll at each $x_k \in \chi_k(\xi_k)$ until an improved mesh point is found or until the set is exhausted.
4. **Update:** If the SEARCH, POLL, or EXTENDED POLL steps find an improved mesh point, then update x_{k+1} and set $\Delta_{k+1} \geq \Delta_k$ as in (2.29). Otherwise, set $x_{k+1} = x_k$ and set $\Delta_{k+1} < \Delta_k$ as in (2.29).

Figure 2.14 MVPS Algorithm

2.6.3 Convergence Results

Torczon [68] proved that a subsequence of GPS iterates converge to a point \hat{x} satisfying $\nabla f(\hat{x}) = 0$ if the objective function is continuously differentiable in the neighborhood of the level set $\{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$, with $x_0 \in \mathbb{R}^n$ the initial iterate. Lewis and Torczon expanded this result to bound [41] and linear constrained [42] problems and showed that a subsequence of iterates converges to a point satisfying the first-order KKT optimality conditions. Audet and Dennis [12] extended these results to problems with less well-behaved objective functions using Clarke nonsmooth calculus [23]. Abramson [3] proved some limited second-order results, which eliminate strict local maximizers and an entire class of saddle points from convergence consideration. Audet and Dennis [12] developed MVPS for bound constrained problems, which was extended to problems with general linear constraints by Abramson [2]. MVPS has also been extended to mixed variable problems with nonlinear constraints [2, 5].

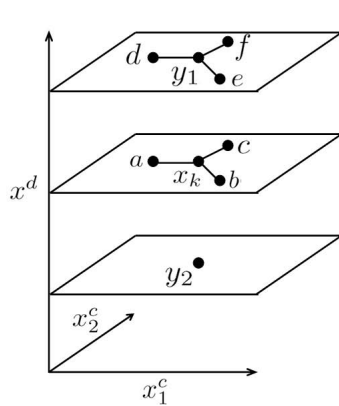


Figure 2.15 MVPS Full Iteration

2.6.4 Kriging Surrogates

While many options are available for use at the SEARCH step, in practice many engineering optimization problems employ surrogates. Surrogates fit a model to the set of previously evaluated points that is fairly easy to optimize. This optimum is then evaluated as a means for finding improvement prior to the POLL and EXTENDED POLL steps. A surrogate may be something as simple as a regression model similar to that of RSM. Often, more complex, but still easily optimized, surrogates are used.

Kriging surrogates are one such type of these more complex surrogates. Kriging attempts to interpolate the response $f(x)$ by fitting a regression model, \mathcal{F} , and a random function, z , to the objective function response at each evaluated point. The predicted response is modeled as

$$\hat{y} = \mathcal{F}(\beta, x) + z(x). \quad (2.35)$$

Each response is deterministic in the sense that repeated evaluations of the same design return exactly the same value. The regression model consists of a linear combination of p functions, chosen by the user, with corresponding regression coefficients

β

$$\begin{aligned}
\mathcal{F} &= \beta_1 f_1(x) + \dots + \beta_p f_p(x) \\
&= [f_1(x) \dots f_p(x)] \beta \\
&= f(x)^T \beta.
\end{aligned} \tag{2.36}$$

The random function is assumed to have mean zero and variance σ^2 which models the process variance of the response. “An interpretation of the model is that deviations from the regression model, though the response is deterministic, may resemble a suitably chosen stochastic process z ” [44]. Covariance between different design sites (x and w , where $x \neq w$) is assumed to be a function of the designs and a parameter Θ ,

$$E[z(x)z(w)] = \sigma^2 \mathcal{R}(\Theta, x, w). \tag{2.37}$$

The matrix $R = [R_{ij}]$ is the matrix of correlations between the set S of m previously visited design sites, $S = [s_1 \dots s_m]$, with each element $R_{ij} = \mathcal{R}(\Theta, s_i, s_j)$. A new design site, x , has a correlation vector $r(x)$

$$r(x) = [\mathcal{R}(\Theta, x, s_1), \dots, \mathcal{R}(\Theta, x, s_m)]. \tag{2.38}$$

While not shown for the sake of brevity [44], the derivation for the unbiased least squares solution for β with respect to R is

$$\beta^* = (F^T R^{-1} F)^{-1} F^T Y, \tag{2.39}$$

where $Y = [y(s_1) \dots y(s_m)]$ is the vector of responses at the m design sites. The $m \times p$ matrix F is the matrix where $F_{ij} = f_j(s_i)$, or equivalently $F = [f(s_1) \dots f(s_m)]^T$. Actual computation of β is performed by QR factorization on F instead of computing an inverse. This is particularly helpful in the case where F is over-determined and/or

near-singular. The result is that the predicted value at a visited design site equals the response at that design site, $\hat{y}(x) = y(x)$.

Regression polynomials of order 0, 1, and 2 are typically used for f_1, \dots, f_p , and are studied in this investigation. The corresponding values of p are 1, $n + 1$, and $\frac{1}{2}(n + 1)(n + 2)$, respectively. The explicit models are

Constant (Order 0):

$$f_1(x) = 1,$$

Linear (Order 1):

$$f_1(x) = 1, f_2(x) = x_1, \dots, f_{n+1}(x) = x_n,$$

Quadratic (Order 2):

$$\begin{aligned} f_1(x) &= 1, \\ f_2(x) &= x_1, \dots, f_{n+1}(x) = x_n, \\ f_{n+2}(x) &= x_1^2, f_{n+3}(x) = x_1x_2, \dots, f_{2n+1}(x) = x_1x_n, \\ f_{2n+2}(x) &= x_2^2, f_{2n+3}(x) = x_2x_3, \dots, f_{3n}(x) = x_2x_n, \\ &\dots, f_{\frac{1}{2}(n+1)(n+2)}(x) = x_n^2 \end{aligned} \tag{2.40}$$

While several options exist for correlation models [44], this investigation restricts the model to only the Gaussian correlation model; namely

$$\mathcal{R} = \prod_{j=1}^m \mathcal{R}(\Theta, d_j), \quad d_j = x - w_j \tag{2.41}$$

where:

$$\mathcal{R} = \exp \Theta_j d_j^2, \quad j = 1, \dots, m. \tag{2.42}$$

The Gaussian correlation model is most often used in practice and is considered to mimic the underlying function behavior better than other correlation models as the number of design sites increases [44]. In reality, the appropriate choice of correlation

model is problem-specific. Many problems may exhibit anisotropic behavior, where different correlations exist in different directions of the model space [44].

2.7 Summary

This chapter covered the physics, modeling, and optimization approaches relevant to scramjet injection array design optimization. The challenging flow environment, heat transfer, flame propagation, and mixing characteristics combine to make hypersonic design optimization a true challenge. CFD and JETPEN provide the vehicle that allows for accurate approximation and modeling of this environment. Familiar optimization techniques provide a back-drop for the first application of provably convergent algorithms.

3. Problem Approach

This chapter outlines the process by which the scramjet injection arrays are designed and evaluated. The evaluation process exclusively uses the JETPEN simulation software, as well as a series of other short programs written by the author and Payne [56]. Care is taken to ensure the results of this investigation are comparable to the previous results found by Payne [56]. The design variables and evaluation methods are nearly the same as those used by Payne [56]. It is important to note that the version of JETPEN used in this study is different than that used by Payne. While the specific improvements are not known, the difference in estimations between the two versions has been found by the author to be fairly small, less than 2%. This should not have any significant impact on the ability to compare results.

3.1 Design Variables

Many potential variables exist for the problem of scramjet fuel injection array design. For the purpose of this investigation the design variables must be limited to only those which are direct inputs into JETPEN. JETPEN requires 15 input parameters from the scramjet design and flow-field to perform its analysis. Many of the input variables are fixed and are determined from the physical scramjet design and mission parameters. The mission and physical design parameters are determined by designs from AFRL/RZAS and outlined in Appendix A. The remaining inputs associated with the injection array design are the only inputs directly considered for determining each design, shown in Table 3.1. For the HiFIRE design, the only inputs are δ_j and N ; P_{T_j} and T_{T_j} are determined from the mission parameters.

A cross section of the fuel injection array assumed by JETPEN is shown in Figure 3.1. JETPEN is only capable of performing analysis on this design, which contains a number N of evenly-spaced circular injectors of diameter d_j^* . The injectors are arranged laterally across the width (l) of a rectangular combustor of height h .

Table 3.1 Design Variables

\vec{x}_i	Description	Units
δ_j	Injection Angle	deg
N	Number of Injectors	
P_{T_j}	Jet Total Pressure	<i>psia</i>
T_{T_j}	Jet Total Temperature	<i>°R</i>

The parameter l_{nf} takes into account any space that cannot contain injectors due to manufacturing constraints. The non-dimensional spacing parameter w/d_j^* is the ratio of injector spacing (w) to injector diameter. While not explicitly handled by JETPEN, the number of injectors (N), as derived by Payne [56], can be found by

$$N = \frac{2l_f^2}{A_r Q \left(\frac{w}{d_j^*}\right)^2}, \quad (3.1)$$

$$A_r = \frac{l_f}{l}, \quad (3.2)$$

where Q is derived from the flow conditions under an ideal gas behavior assumption. In contrast to Payne's work, N is handled directly as a design variable, instead of being derived implicitly post-analysis from the value of w/d_j^* .

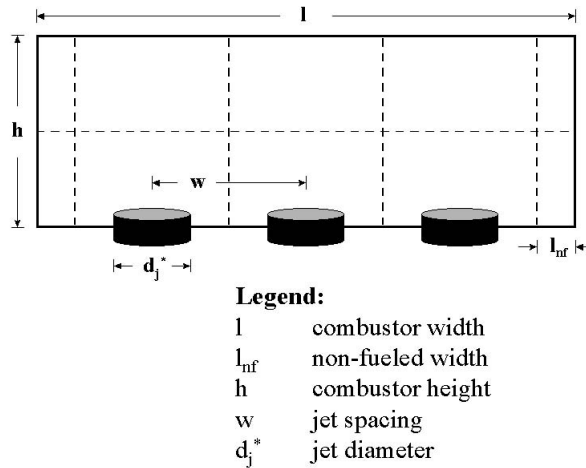


Figure 3.1 Fuel Injection Array Cross-Section

3.2 *Dependent Variables*

Several remaining JETPEN inputs are dependent on the values for the design parameters. For Payne’s design, injector diameter changes with the jet total pressure (P_{T_j}), jet total temperature (T_{T_j}), and the number of injectors (N) to keep the fuel mass flow rate constant. This value, as derived by Payne [56], can be computed as

$$d_j^* = \left[\frac{1}{2N} A_r Q \right]^{\frac{1}{2}}. \quad (3.3)$$

For the HiFIRE design the injector diameter is fixed at 0.125 inch.

The jet specific heat at constant pressure (c_{p_j}) and ratio of specific heats (γ_j) are inputs to JETPEN that are dependent on fuel molecular weight (w_j) and Mach number (M_j) of the injected fuel, as well as the design variables, jet total pressure (P_{T_j}) and total temperature (T_{T_j}). Fuel molecular weight is that of ethylene, and Mach number (M_j) is determined by the mission parameters. The inputs c_{p_j} and γ_j are determined by the method developed by Payne [56], utilizing a bisection routine with interpolating polynomials. The author modified this routine slightly to extrapolate the polynomials in the case where, after conversion to static conditions, the jet total pressure exceeded the bounds of the uppermost interpolating polynomial. This modification has implications on Payne’s results that are discussed in Section 4.6. Ideal gas laws are applied to calculate the dependent variables and these relationships can be seen in Appendix B.

Interpolating polynomials are developed from tabular data for ethylene. For the original design studied by Payne, the interpolating polynomials are developed from tabular data provided by AFRL/RZAS. The second design problem requires a different range of temperatures and pressures, and thus a different set of interpolating polynomials are used. This second set of polynomials is derived from tabular data readily available from the National Institute for Standards and Technology (NIST) website [53] for ethylene (ethene).

The HiFIRE design requires several calculations for the specific heat capacity (c_{pa}) and ratio of specific heats of air (γ_a). The temperatures and pressures inside the scramjet engine for this design fall into the range called “thermally perfect” by [36]. The relationship between γ_a and T_{T_a} is essentially linear in this range, and their values are determined according to this relationship via the same bisection routine.

3.3 Design Evaluation

Evaluation of a scramjet injection array design begins with writing design variables into a short text file, called DV.DAT. A pre-processing routine uses this to calculate the dependent variables, and two files, INPUT.DAT and CMBST.DAT, are built. The JETPEN input file, INPUT.DAT, contains all the input parameters needed for JETPEN to perform its analysis. The CMBST.DAT file contains the input parameters needed by the post-processor. A flowchart of the design evaluation process is shown in Figure 3.2.

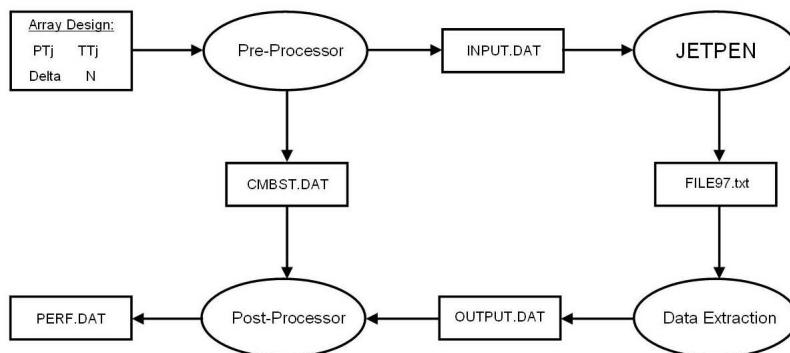


Figure 3.2 Flowchart of Design Evaluation Process

Upon completion of the pre-processing routine, JETPEN is run and several output files are built containing the resulting flowfield data. The text-file FILE97 contains the flowfield axial data needed to assess the combustor performance. In this investigation, the source code for JETPEN is unavailable and the axial data cannot be put separately into another file, as Payne did. As a result, a short routine

is necessary to extract and format the appropriate data. A post-processing routine uses the formatted data from the file OUTPUT.DAT and the input variables contained in CMBST.DAT to assess the design performance. The final output lies in the file PERF.DAT and contains the design variables and the performance measures.

3.4 *Performance Measures*

A single performance measure, such as total thrust generated or delivered specific impulse (I_{sp}), is best for design assessment. JETPEN is incapable of determining the generated thrust or I_{sp} , and thus the mixing properties estimated by JETPEN are the only performance measures available. It is important to note again that JETPEN is not capable of determining the dynamic pressure losses in the combustor as a result of mixing, and they must be estimated by means outside the scope of this investigation.

Keeping in line with Payne [56], the merit of a fuel injection array design is measured by the downstream distance where sufficient mixing occurs for combustion. The optimal array is the combination of δ_j , N , P_{T_j} , and T_{T_j} that yields the shortest downstream distance where appropriate mixing occurs. Shorter distances are preferable to longer distances, due to the extremely short residence time of the crossflow. The intent is to create a laterally even, properly mixed, block of air and fuel as soon as possible for combustion.

The output flowfield data from JETPEN are given at axial distances expressed as multiples of the jet diameter d_j^* . To find the actual distance, the output axial distances (x_i^*) must be multiplied by d_j^* . Non-dimensional performance measures are preferred, and the dimensional axial distances are normalized by the combustor height. The resulting axial distances outlined in the following subsections are found by

$$x_i = \frac{x_i^* d_j^*}{h}, \quad (3.4)$$

where $i = 1, 2, 3$ represents the respective performance measures.

3.4.1 Jet Penetration

The combustor design in this investigation has two injection arrays arranged across from each other on parallel walls. As a result, it is important for the fuel plumes of the facing arrays to merge, or equivalently, to penetrate to the combustor centerline. It is desirable for this to occur as quickly as possible to allow time for proper mixing. Had only one injection array been considered, then the fuel jet would need to penetrate the entire combustor height (h). Jet penetration height is defined as the vertical height (y) achieved by the fuel plume at axial distance x_i and is shown in Figure 3.3 [56].

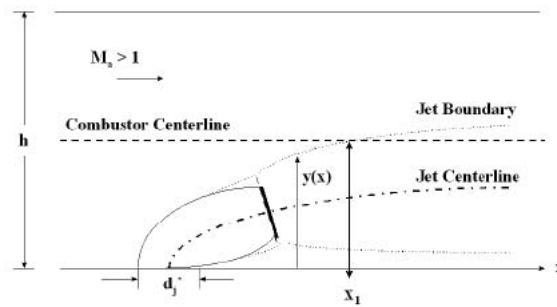


Figure 3.3 Jet Penetration

The jet penetration to the combustor centerline is considered satisfied at the first axial station where the fuel plume height is greater than half the combustor height. The performance measure y_1 [56] is the axial station where

$$\frac{y(x_i^*)}{d_j^*} \geq \frac{h/2}{d_j^*}. \quad (3.5)$$

3.4.2 Plume Expansion

The jet plume expansion rate, particularly the axial distance where adjacent plumes merge, shown in Figure 3.4, is another key indicator of mixing performance.

Satisfaction of this criterion [56] occurs at the axial station where

$$\frac{D(x)}{d_j^*} \geq \frac{w}{d_j^*}. \quad (3.6)$$

The performance measure y_2 is this axial station where the plume diameters ($D(x)$) of adjacent jets meet.

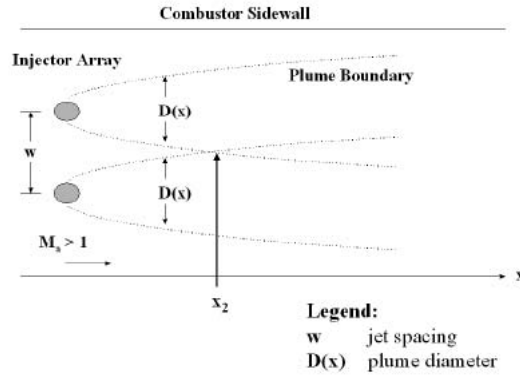


Figure 3.4 Plume Expansion

3.4.3 Fuel Concentration Decay

The decay of the fuel concentration in the plume is another important measure of mixing. Prior to the Mach disk the plume is completely comprised of fuel, and only at the Mach disk does the fuel begin to mix with the freestream [59]. The average concentration of fuel in the plume (α_{avg}) must decay to the stoichiometric ratio (f_{ST}) to maximize combustion efficiency. The final performance measure, y_3 , [56] is the axial distance where

$$\alpha_{avg}(x) \leq f_{ST}. \quad (3.7)$$

A typical plot of the fuel concentration decay versus the axial distance is shown in Figure 3.5.

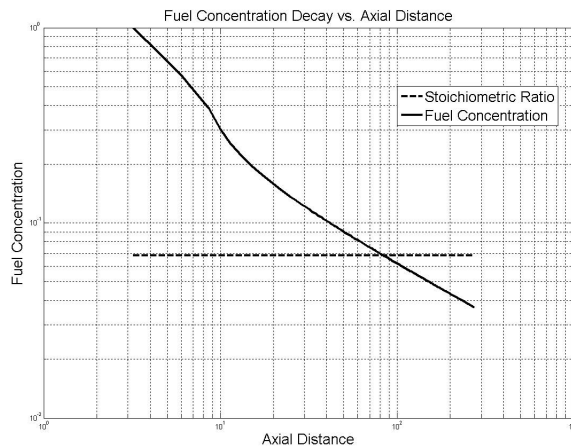


Figure 3.5 Fuel Concentration Decay

3.4.4 Performance Measure Estimation

JETPEN is incapable of direct evaluation of the performance measure criteria. The value of a performance measure is declared equal to the first axial station where its criterion is satisfied. This inherently over-estimates the performance measure and may not be the most accurate estimation method available. The discrete and relatively smooth nature of the output data may make model fitting and subsequent criterion solving a promising alternative. For example, the output data shown in Figure 3.6 resembles a square-root function. Fitting this model, or equivalently a linear model to the square of the data, and then solving for the penetration criterion may yield improved estimation. However, the output data from JETPEN is itself an estimate, and fitting an estimated model to estimated data is likely assuming fidelity that does not exist and compounding error. Therefore the conservative methods employed by Payne are applied throughout this study.

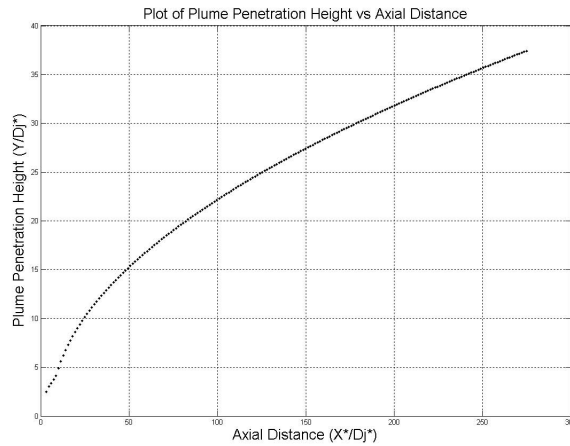


Figure 3.6 Typical JETPEN Plume Penetration Data

3.5 Constraints

3.5.1 Payne's Design

The constraints on the design variables P_{T_j} and T_{T_j} of the Payne's design are enforced as before to ensure validity of the AFRL interpolation polynomials, and to ensure that the fuel is in a gaseous state at injection. The previous lower restriction of 10° on δ_j is kept with the intent of avoiding the manufacturing difficulties of producing high jet total pressures at the injection point. The upper restriction of 70° on δ_j is expanded to 90° to allow for the possibility of full normal injection. It is known *a priori* from Payne's results that the optimal design does not likely lie in this added region and poses minimal risk in adversely influencing results.

The constraints on the number of injectors (N) is handled differently in this investigation. Payne restricted his investigation to a minimum of 3 and a maximum of 10 injectors. The upper restriction was based in part on the physical limitation fitting the injectors inside the combustor width. However, Payne's optimal design using the maximum of 10 injectors took up less than 20% of the available space. Secondly, Payne found that for large values of N , JETPEN would abort and fail to return data. This phenomena is further investigated in Section 4.6, but the end

result is that this upper restriction is no longer applicable. This investigation handles N differently in two applications. First, an attempt is made to reproduce Payne’s results using an upper limit of 10, followed by an application where the upper limit is 20. This increase should allow improved designs to be found, since the optimum found by Payne is at the boundary of this constraint.

The design bounds used in Payne’s problem, applied again in this study, are:

$$\begin{aligned}
 10^\circ &\leq \delta_j \leq 90^\circ \\
 20 \text{ psia} &\leq P_{T_j} \leq 650 \text{ psia} \\
 850 \text{ R}^\circ &\leq T_{T_j} \leq 1500 \text{ R}^\circ \\
 N &\in \{3, 4, \dots, N_{max}\} \\
 N_{max} &\in \{10, 20\}.
 \end{aligned}$$

3.5.2 *HiFIRE Design*

The HiFIRE design problem only considers δ_j and N as design variables. The constraint on δ_j is kept for the same reasons as the previous problem. Restrictions on N are set at a lower limit of 3 and an upper limit of 31. This upper limit is the largest feasible number of injectors that can physically be placed in the design. Specifics of the design parameters can be found in Appendix A. The constraints applied in the optimization of this problem are:

$$\begin{aligned}
 10^\circ &\leq \delta_j \leq 90^\circ \\
 N &\in \{3, 4, \dots, 31\}.
 \end{aligned}$$

3.6 *Summary*

This chapter outlines the design variables, performance measures, and constraints necessary to optimize the scramjet fuel injection array. Optimization of the design previously studied by Payne uses has four design variables δ_j , P_{T_j} , T_{T_j} , and N , where N is now an explicit design variable. The HiFIRE design considers only

two design variables: δ_j and N . Three performance measures based on gross mixing characteristics are used to assess each candidate design. A design evaluation routine using JETPEN is developed to evaluate each design delivered by the optimizer. Bound constraints are outlined for both scramjet designs.

4. Optimization

The intent of this chapter is to detail the optimization of the approach outlined in Chapter 3. Optimization is performed using the two primary methods outlined in Chapter 2: mixed variable pattern search and genetic algorithms. In the SEARCH step of MVPS, several Kriging surrogates are investigated including a base-case where no surrogates are used. Both classes of genetic algorithms from Section 2.5 are applied.

It is important to note up front that the intent of the author is not to provide a fair comparison of the two primary methods. Instead, the author focuses on making an overall useful comparison. The previous work done by Payne identifies several parameters used in the genetic algorithms that greatly improve performance. To “dumb-down” the algorithms in the spirit of providing a fair playing field would certainly betray his previous work. Instead, the genetic algorithms are applied at their tuned parameters and MVPS uses default values. Hence, if MVPS performs comparatively well to the genetic algorithms, then one may reasonably conclude that MVPS performs well on this class of problems. However, if the converse is true, then conclusions may be harder to draw. In either case, measures of algorithm performance must be fair and balanced for any conclusion to be made.

4.1 Problem Statement

4.1.1 Design and Response Vectors

Each candidate design is represented by a vector of the design variables. For Payne’s design this vector contains four variables and is expressed as

$$\vec{\mathbf{x}} = [\delta_j, N, P_{T_j}, T_{T_j}]^T .$$

The HiFIRE design only considers two variables and the design vector is expressed as

$$\vec{\mathbf{x}} = [\delta_j, N]^T.$$

These definitions are necessary for the statement of the optimization problem.

For Payne’s design, the three performance measures are assessed for each candidate design returned by the optimization algorithm. They are expressed as a vector of performance measures normalized by combustor height, $\vec{\mathbf{y}} = [y_1(\vec{\mathbf{x}}), y_2(\vec{\mathbf{x}}), y_3(\vec{\mathbf{x}})]$, as defined in Table 4.1. The notation $y(\vec{\mathbf{x}})$ is used to denote a specific performance measure y that corresponds to a specific design $\vec{\mathbf{x}}$.

Table 4.1 Payne’s Performance Measures

$\vec{\mathbf{y}}_i$	Description
y_1	Axial distance to combustor half-line penetration
y_2	Axial distance to adjacent plume merge
y_3	Axial distance to stoichiometric fuel concentration decay

The HiFIRE design is evaluated at three flight speeds. The vector of performance measures is assessed for each flight speed. The result is a 3×3 matrix of performance measures to flight speeds, is

$$\vec{\mathbf{Y}} = \begin{bmatrix} \vec{\mathbf{y}}_{M6} \\ \vec{\mathbf{y}}_{M7} \\ \vec{\mathbf{y}}_{M8} \end{bmatrix} = \begin{bmatrix} y_{1,M6}(\vec{\mathbf{x}}) & y_{2,M6}(\vec{\mathbf{x}}) & y_{3,M6}(\vec{\mathbf{x}}) \\ y_{1,M7}(\vec{\mathbf{x}}) & y_{2,M7}(\vec{\mathbf{x}}) & y_{3,M7}(\vec{\mathbf{x}}) \\ y_{1,M8}(\vec{\mathbf{x}}) & y_{2,M8}(\vec{\mathbf{x}}) & y_{3,M8}(\vec{\mathbf{x}}) \end{bmatrix}. \quad (4.1)$$

Again, the notation $y(\vec{\mathbf{x}})$ is used to denote a specific response resulting from a specific design, but is also extended across the flight conditions. For example, $y_{1,M6}(\vec{\mathbf{x}})$ represents the axial distance for the fuel plumes to penetrate to the combustor half-line, resulting from design $\vec{\mathbf{x}}$ at Mach 6 flight conditions.

4.1.2 Original Design Problem Statement

Optimization of the injection array design problem studied by Payne and outlined in Chapter 3 is summarized as

$$\text{minimize } F = f(\vec{\mathbf{y}})$$

subject to:

$$\begin{aligned} 10^\circ &\leq \delta_j \leq 90^\circ \\ 20 \text{ psia} &\leq P_{T_j} \leq 650 \text{ psia} \\ 850 \text{ R}^\circ &\leq T_{T_j} \leq 1500 \text{ R}^\circ \\ N &\in \{3, 4, \dots, N_{max}\} \\ N_{max} &\in \{10, 20\}. \end{aligned}$$

The solution to this problem is the design vector, $\vec{\mathbf{x}} = [\delta_j, N, P_{T_j}, T_{T_j}]^T$, that optimizes some measure of fuel injection array performance, $F = f(\vec{\mathbf{y}})$, as defined in Section 3.4, within the prescribed limitations.

4.1.3 HiFIRE Problem Statement

The HiFIRE design has several different mission objectives. The design is optimized across three different flight speeds of Mach 6, 7, and 8. The fuel mass flow rate, \dot{m} , is adjusted to maintain a constant fuel/air ratio at each of these flight conditions. The response $\vec{\mathbf{Y}}$ for a candidate design is the 3×3 matrix of responses to flight Mach numbers. The optimization problem takes the form:

$$\text{minimize } F = f(\vec{\mathbf{Y}})$$

subject to:

$$\begin{aligned} 10^\circ &\leq \delta_j \leq 90^\circ \\ N &\in \{3, 4, \dots, 31\}. \end{aligned}$$

4.2 Objective Function Form

Payne [56] investigated both single and multi-objective optimization of the three performance measures and found that the former worked best. Several objective functions were investigated in the single objective context, while only one multi-objective form was investigated. The single objective functions investigated by Payne are

$$f(\vec{y}) = y_1 \quad (4.2)$$

$$f(\vec{y}) = y_2 \quad (4.3)$$

$$f(\vec{y}) = y_3 \quad (4.4)$$

$$f(\vec{y}) = \|\vec{y}\|_1 \quad (4.5)$$

$$f(\vec{y}) = \|\vec{y}\|_2. \quad (4.6)$$

As could be expected, he found that optimizing only one performance measure yielded poorer designs in the remaining two. The 1-norm and 2-norm of all three measures consistently returned better designs, with the 2-norm only slightly outperforming the 1-norm. These normed versions of $f(\vec{y})$ typically returned values for y_1 , y_2 , and y_3 that were near or better than points generated by their individual optimization. Thus, the only objective function form applied in this study is (4.6).

The multi-objective approach investigated by Payne merely generated the basic Pareto-optimal points by sequential optimization of each performance measure. These points required an excessive number of function evaluations and showed minimal to no improvement over the single objective techniques [56].

Three different objective function forms are applied for the new HiFIRE design problem. A logical extension of Payne's work is to take the Frobenius norm of the

response matrix. This can be expressed as

$$f_1(\vec{\mathbf{Y}}) = \|\vec{\mathbf{Y}}\|_{F=Frobenius} = \sum_{i=1}^3 \sum_{j=1}^3 \vec{\mathbf{Y}}_{i,j}^2. \quad (4.7)$$

Another objective function form of interest is the maximum element of the response matrix; *i.e.*

$$f_2(\vec{\mathbf{Y}}) = \max_{i,j} [\vec{\mathbf{Y}}]. \quad (4.8)$$

One potential weakness of (4.7) is that the norm may be dominated by a single large value. For example, if $\vec{\mathbf{Y}}_1$ and $\vec{\mathbf{Y}}_2$ are given by

$$\vec{\mathbf{Y}}_1 = \begin{bmatrix} 2.12 & 0.23 & 2.12 \\ 1.95 & 0.23 & 1.95 \\ 1.77 & 0.23 & 1.77 \end{bmatrix}, \vec{\mathbf{Y}}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 4.81 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

then

$$\|\vec{\mathbf{Y}}_1\|_F = 4.81 = \|\vec{\mathbf{Y}}_2\|_F.$$

The design corresponding to response $\vec{\mathbf{Y}}_1$ is superior to that of $\vec{\mathbf{Y}}_2$ since the longest mixing response occurs at less than half that of $\vec{\mathbf{Y}}_2$, but (4.7) would show no preference to either. Optimizing with respect to the largest response element ensures that candidate designs avoid this potential situation.

The final objective function form attempts to minimize the overall fuel decay distances (y_3) subject to additional constraints on combustor half-line penetration (y_1) and adjacent plume merge distances (y_2). The objective function is the 2-norm of the fuel concentration decay distances across the flight Mach numbers. The response from each candidate design is the vector of the normalized axial distances required for the fuel plume to decay to the stoichiometric ratio at each flight number. This response is denoted $\vec{\mathbf{Y}}_{3,:} = [y_{3,M_6}(\vec{\mathbf{x}}) \ y_{3,M_7}(\vec{\mathbf{x}}) \ y_{3,M_8}(\vec{\mathbf{x}})]$. AFRL desired the combustor half-line penetration distance and adjacent plume merge distance to be

less than 16 jet diameters downstream. The problem statement becomes:

$$\text{minimize } F = f_3(\vec{Y}) = \|\vec{Y}_{3,:}\|_2$$

subject to:

$$\begin{aligned} 10^\circ &\leq \delta_j \leq 90^\circ \\ N &\in \{3, 4, \dots, 31\} \\ y_{1,i} &\leq 16d_j^* \\ y_{2,i} &\leq 16d_j^* \end{aligned}$$

for $i = M_6, M_7, M_8$.

4.3 Algorithm Performance Assessment

Wolpert and Macready [73] suggest algorithm performance measures be based on the path of points visited by the algorithm. Other measures, such as computation time, are more subjective and dependent on machine speed and programming proficiency. The time ordered unique set of m visited points d_m is the basis for the comparisons made in this investigation, where the set d_m is defined by

$$d_m = [(d_1^{\vec{x}}, d_1^{\mathbf{Y}}), \dots, (d_m^{\vec{x}}, d_m^{\mathbf{Y}})]. \quad (4.9)$$

The design vector \vec{x} corresponds to $d_i^{\vec{x}}$, and $d_i^{\mathbf{Y}}$ is the performance measure value produced by the design [73]. Overall algorithm performance is assessed by the best value achieved, γ , at the m th visited point. Average best value across several samples, γ_{avg} , is necessary for the genetic algorithms due to their random elements. This average best value is not applicable to MVPS because it is deterministic. To ensure comparable results to Payne's work, the algorithms are also assessed on the required function evaluations to where improvement failed to exceed 10^{-4} .

4.4 *Mixed Variable General Pattern Search Application*

The base case application of MVPS is the foundation for the surrogate-based optimizations. In this base case, the SEARCH step is not used and the NOMADm standard defaults are applied for the EXTENDED POLL trigger, mesh refinement factor w_k , etc. All other surrogate-based applications build upon this. A CCD is applied as the set of initial points for the several surrogate types.

The SEARCH step, when applied, is performed with the Design and Analysis of Computer Experiments (DACE[®]) software package developed by Lophaven et. al. [44, 45] at the Technical University of Denmark. This software package fits a Kriging approximation model to the responses at the design points visited by MVPS. This surrogate is then optimized by the MATLAB[®] FMINCON toolbox and a single optimum is returned. Zero, first, and second-order regression models are applied in the Kriging surrogates.

At the first application of the Kriging surrogate, the values for Θ are optimized in DACE[®] within specified bounds to obtain a maximum likelihood estimate:

$$\min_{\Theta} \psi(\Theta) \equiv |R|^{\frac{1}{m}}.$$

In this application this is the only time that Θ is optimized. The specified bounds on Θ are determined in NOMADm prior to the actual Θ optimization.

In the base case where no SEARCH is performed, the initial point is at the center of the continuous variable space. The number of injectors is kept at 6 for all initial points. The discrete neighbors used in the POLL step of MVPS are defined as in (2.30). This can be thought of as a “one-up, one-down” scheme. If the current incumbent has 6 injectors, then the discrete neighbors evaluated in the POLL step have the same continuous values as x_k , but have 5 and 7 injectors respectively. If $x_k^d = 3$, then only one neighbor is evaluated with 4 injectors. If $x_k^d = N_{max}$, then it is evaluated similarly. All MVPS optimization methods are applied to Payne’s original

design problem, with the best performing methods selected for use in subsequent optimization problems.

4.5 Genetic Algorithm Application

Both the SGA and the μ GA are used to optimize the engineering design problem outlined in Chapter 3. The genetic algorithms are written for maximization; thus it is necessary that the fitness function is taken as the negative of the objective function.

Several “tuning” parameters identified by Payne [56] are used by the genetic algorithms and listed in Table 4.2. *Elitism*, which carries the best individual into the next generation, is invoked and *uniform crossover* performs the function with a uniform distribution. Only one child is created per crossover in an attempt to reduce function evaluations, and the mutation probability (only applicable in SGA) is the reciprocal of the population size. Again, improvement less than 10^{-4} is the threshold considered for significant improvement, which is consistent with thresholds used by Payne [56] and Markell [46]. Each algorithm’s final value is reported as the last value achieved with improvement greater than this threshold.

Table 4.2 Tuning parameters for Genetic Algorithms

Parameter	SGA	μ GA
Population Size	32	5
Max. Generations	25	50
Mutation Prob.	0.03125	NA
# Children	1	1
Elitism	Yes	Yes
Crossover Distribution	Uniform	Uniform

The SGA and μ GA are employed in optimization of the original design problem investigated by Payne. Use in subsequent optimization problems is based on performance.

4.6 JETPEN Monte-Carlo Sampling

Payne reported that JETPEN would abort at values of $N \gg 10$. Prior to the modification discussed in Section 3.2, certain values for T_{T_j} and P_{T_j} would cause the dependent variable calculation to fail for values of c_{p_j} and γ_j outside the bounds of the interpolation polynomials. This effectively prevented the pre-processing routine from closing and is one potential source of JETPEN's undesirable behavior. The fix in Section 3.2 eliminates this source, however, in practice JETPEN crashes are now seen more often than reported by Payne [56]. This investigation uses an updated version of JETPEN developed in 2005. Crashes of JETPEN are seen at nearly every value of N in this new version. It is known that JETPEN will crash in the atypical case where the sum of the injector diameters exceeds the combustor width, namely:

$$\sum_{i=1}^N d_j^* = N d_j^* > w_{combustor}. \quad (4.10)$$

However, this never happens in practice. To characterize this behavior JETPEN was run at 10,000 Monte Carlo points. These points in the design region were chosen at 1250 randomly generated points for each value of $3 \leq N \leq 10$. The results conclusively showed that increased values for N and δ_j increase the crash probability. However, these values alone are not sufficient to determine JETPEN crashes with certainty. The number of failures at each level of N are shown in Table 4.3 and highlight its influence.

Table 4.3 Monte Carlo Results

N	3	4	5	6	7	8	9	10
Crashes	2	13	17	39	46	70	82	72
Successes	1248	1237	1233	1211	1204	1180	1168	1178

Figure 4.1 shows the distribution of JETPEN crash points against T_{T_j} , P_{T_j} , and δ_j at $N = 10$. The tight grouping of points at high values of δ clearly show its influence. Values of T_{T_j} and P_{T_j} seem to have some influence but are much less

pronounced. However, among the region of the crash points, there are many points that do not cause JETPEN to crash. To investigate further, JETPEN was run at 500 Monte Carlo points sampled on a small scale, centered on a randomly selected crash point. These runs are shown in Figure 4.2. The banded regions clearly identify P_{T_j} as the most influential factor, with T_{T_j} having lesser influence. Thus, all design variables have at least some influence on whether or not JETPEN will crash.

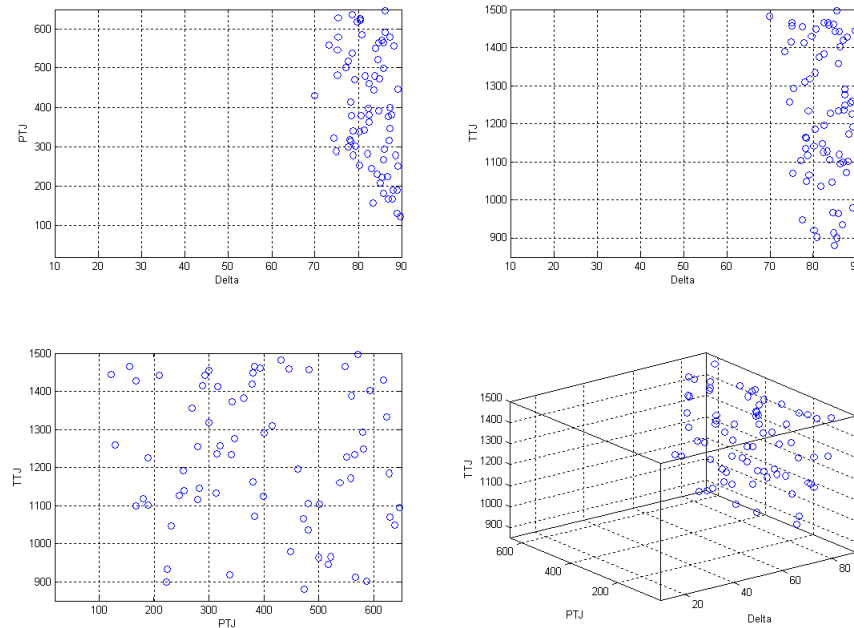


Figure 4.1 JETPEN crashes at $N = 10$

The genetic algorithms cannot inherently handle JETPEN crashes and thus a work-around is necessary. Utilizing the knowledge gleaned from the Monte Carlo runs, a computationally inexpensive routine is used to return a function value to the genetic algorithm in case of a crash. If JETPEN crashes, the values of P_{T_j} and T_{T_j} are decreased by 0.25, and JETPEN is re-run. The intent here is to get outside of the bands seen in Figure 4.2 and return the function value of a similar design. If P_{T_j} and T_{T_j} are adjusted 10 times consecutively without success, then a comparatively large function value of 10 is returned. This work-around is not necessary for the

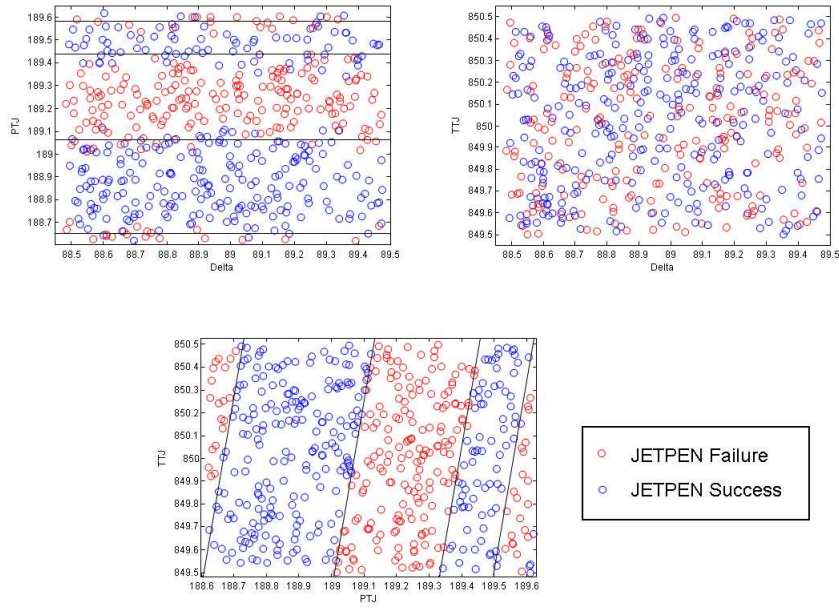


Figure 4.2 Local JETPEN behavior ($\bar{x} \pm 0.5$)

MVPS methods since they are designed to handle cases where a function value is not returned. If JETPEN crashes then a value of ∞ is returned to the optimizer, NOMADM. If JETPEN crashes at one of the initial points, then the initial surrogate is reduced to a zero-order Kriging polynomial.

4.7 Summary

This chapter outlined the application of each optimization technique and discussed the potentially problematic behavior of JETPEN. The MVPS techniques use the default settings of the NOMADM and Kriging surrogates use the DACE[®] software. The genetic algorithms are applied at the “tuned” parameters used by Payne [56]. The vectors of performance measures are defined and several forms of objective functions are developed. Finally, an analysis using 10,500 runs of JETPEN is performed in an attempt to characterize the input parameters that cause it to crash.

5. Results

The re-optimization of Payne’s problem closely matched his results and conclusively demonstrated the superiority of MVPS over genetic algorithms as applied in this context. Repeated applications of the genetic algorithms on this problem highlighted their inherent variability and an analysis of unique design sites showed a need for the use of a *cache*. Results where the upper restriction on N was lifted showed no significant improvement and suggests that the problem is fairly indifferent to increases in the number of injectors beyond 10. Optimization of the HiFIRE design injection angle supports recommendations by Ortwerth [54] and Mays *et al.* [47]. Optimal designs had injection angles near 30° and showed approximately 1% improvement over 15° injection. Optimization of the HiFIRE design considering both injection angle and number of injectors was inconclusive, which is attributed to a modeling deficiency in JETPEN.

5.1 Previous Design Re-Optimization Results

Table 5.1 details the results from the re-optimization of Payne’s original problem and Table 5.2 show the associated designs and mixing performance measures. The designs found by all methods closely resemble those found by Payne [56]. The only significant difference is that nearly all of Payne’s designs chose 10 injectors, while typically fewer were favored in this investigation. A plausible explanation for this is that the models in the current version of JETPEN have likely been improved.

The SGA returned the best 2-norm objective function value but required an excessive 545 function evaluations. Comparatively, the μ GA returned a 2-norm objective function value within 0.6% of this value with only 119 function evaluations. If design sites and their responses are stored in a *cache*, then previously evaluated designs need not be re-evaluated and function evaluations are reduced. Use of a

cache reduced the function evaluations of the SGA by approximately 5%, but the function evaluations for the μ GA were nearly cut in half.

The results of five runs of each genetic algorithm are shown in Table 5.3. Figure 5.1 shows the average best found solution and the average generational fitness across the five runs. It is clear that the best performance of each algorithm is fairly atypical, and significant run-to-run variability exists in the genetic algorithms. The average performance across the five runs is significantly poorer than that of the best single run. Most significantly, the μ GAs required about 75% fewer function evaluations to obtain the same quality solution as the SGAs.

Table 5.1 Payne's Design Re-Optimization Results

Optimization Results				
Simple Genetic Algorithm				
	Ftn Evals No Cache	Ftn Evals With Cache	Cache Hits	F
Best F	545	518	27	2.952
Avg of 5	503	409	24	3.012
Micro Genetic Algorithm				
	Ftn Evals No Cache	Ftn Evals With Cache	Cache Hits	F
Best F	119	64	55	2.969
Avg of 5	161	88	73	3.001
MVPS				
Surrogate	Ftn Evals No Cache	Ftn Evals With Cache	Cache Hits	F
None	129	104	25	3.014
Krig 0	104	92	12	2.958
Krig 1	154	121	33	2.974
Krig 2	76	53	23	2.964

The MVPS algorithm, when used with surrogates, conclusively outperformed the genetic algorithms. Convergence plots of all MVPS methods are shown in Figure 5.2. These plots show that MVPS found a better quality solution than the genetic algorithms on average. The plots also show that the addition of surrogates to MVPS

Table 5.2 Best Designs and Associated Responses

Results		Design Variables				Perf. Measures		
Method	Best F	δ_j	N	P_{T_j}	T_{T_j}	y_1	y_2	y_3
SGA	2.952	24.166	9	637.00	850.56	2.243	0.369	1.884
μ GA	2.969	12.759	9	608.49	851.47	2.260	0.344	1.893
MVPS	3.014	30.250	7	650	854.00	2.247	0.545	1.934
Krig 0	2.958	19.617	10	650	850	2.259	0.313	1.885
Krig 1	2.974	18.000	7	641.64	850	2.223	0.513	1.908
Krig 2	2.964	18.126	8	650	850	2.234	0.435	1.899

Table 5.3 Genetic Algorithm Comparisons

Simple Genetic Algorithm				
Run	Ftn Evals (No Cache)	Ftn Evals (With Cache)	Cache hits	F
1	778	736	42	3.024
2	194	183	11	3.020
3	545	518	27	2.952
4	399	375	24	2.990
5	244	231	13	3.073
Average	432	409	24	3.012
Micro-Genetic Algorithm				
Run	Ftn Evals (No Cache)	Ftn Evals (With Cache)	Cache hits	F
1	119	64	55	2.969
2	215	126	89	3.080
3	167	86	81	3.028
4	242	146	96	3.057
5	116	59	57	2.982
Average	172	96	72	3.023

significantly sped up convergence and reduced the number of function evaluations. Both zero-order and second-order Kriging surrogates performed well, requiring 92 and 53 function evaluations, respectively, to come within 0.5% of the SGA's best objective function value.

An analysis of surrogate performance is shown in Table 5.4. These results are based on improvement that occurred after evaluation of the 15 initial CCD points.

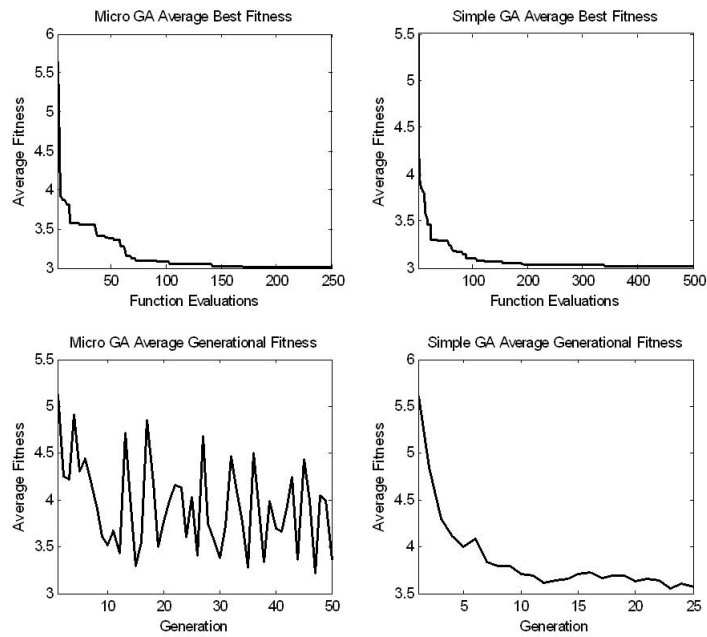


Figure 5.1 Genetic Algorithm Average Performance Across 5 samples

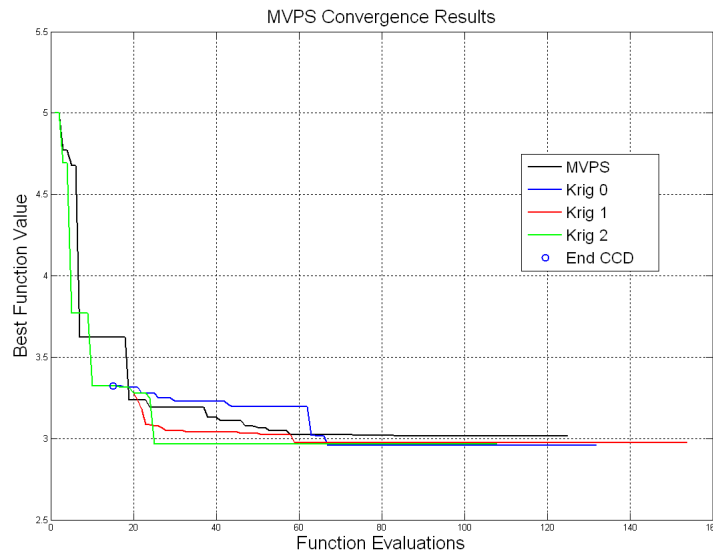


Figure 5.2 Mixed Variable Pattern Search Performance

For the base case where no initial CCD was performed, the improvement is based on the objective function value returned by the initial point at the center of the design

region. All initial points had a discrete value corresponding to 6 injectors. These results show that when surrogates are added they comprise the majority of incumbent improvement. Most of this improvement came on the first few applications of the surrogates. The computational cost in terms of function evaluations is also significantly reduced. It is interesting to note that, as the order of the regression polynomial increases, so does the improvement percentage attributed to the model. Figure 5.3 shows that MVPS with the best surrogates dominate the expected performance of the genetic algorithms. In other words, at nearly every function evaluation, MVPS with surrogates had found a better solution than the genetic algorithms. In conclusion, the MVPS methods converge faster to quality solutions than the genetic algorithms on average.

Table 5.4 Incumbent Improvement Breakdown by Algorithm Step

Improvement by Step				
Method	MVPS	Krig 0	Krig 1	Krig 2
Search	NA	52.05%	64.23%	78.19%
Poll	94.77%	23.86%	10.08%	3.08%
N Poll	5.23%	24.09%	12.47%	18.73%
Ext Poll	0.00%	0.00%	13.21%	0.00%

The designs and performance measures returned by all methods line up well with most of Payne’s previous results. He reported that μ GAs required the fewest function evaluations compared to other methods he studied. He concluded that low injection angles, maximum injectant pressure, minimum injectant temperature, and a maximum number of fuel injectors returned overall superior mixing characteristics [56]. All of these are similar to the designs in Table 5.2, with the exception of number of fuel injectors. Smaller numbers of fuel injectors were preferred in this study, with the maximum number of 10 only having been chosen by one optimization method. This discrepancy can be explained through the improved modeling in the updated version of JETPEN used in this study.

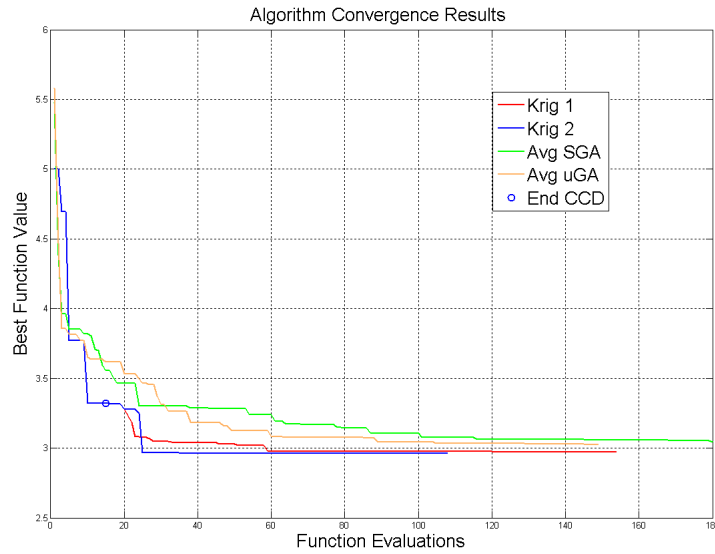


Figure 5.3 MVPS and GA comparisons

5.2 Relaxation of Injector Restriction

The optimization results showed that the μ GA and MVPS with zero and second-order polynomial Kriging performed the best in their respective classes. These methods were selected for the next optimization problem where the upper limit on N is expanded. The restriction $3 \leq N \leq 10$ is changed to $3 \leq N \leq 20$ to investigate areas not possible with the older version of JETPEN. The results showed only minimal improvement in this expanded area, but again demonstrated the superiority of the pattern search methods.

The performance of the three selected methods are shown in Table 5.5 and in Figure 5.4. Designs corresponding to Table 5.5 are shown in Table 5.6. The best of two μ GA runs took more function evaluations and returned an inferior solution as compared to MVPS. MVPS with zero and second-order Kriging polynomials required 61.9% and 13.6% fewer function evaluations, respectively, than the μ GA. MVPS independently converged to the same solution, which closely resembled the design from the previous optimization. The μ GA returned a significantly different design than

the prior optimization. The best designs of this phase showed an improvement of only 0.2% over the previous best objective function value. By performance measure, y_1 increased by 1.7%, y_2 decreased by 4.8%, and y_3 decreased by 0.8% over the best values attained in Section 5.1. It can be inferred that responses are fairly insensitive to injector additions beyond 10.

Table 5.5 AFRL Design Re-Optimization Results

Optimization Results				
Micro Genetic Algorithm				
	Ftn Evals (No Cache)	Ftn Evals (With Cache)	Cache Hits	F
Best Run	237	118	119	2.9644
MVPS				
Surrogate	Ftn Evals (No Cache)	Ftn Evals (With Cache)	Cache Hits	F
Krig 0	52	45	7	2.9472
Krig 2	102	85	17	2.9472

Table 5.6 Optimization Results

Results		Design Variables				Perf. Measures		
Method	Best F	δ_j	N	P_{T_j}	T_{T_j}	y_1	y_2	y_3
μ GA	2.964	18.299	16	624.97	858.39	2.279	0.192	1.886
Krig 0	2.947	20.749	11	650	850	2.260	0.298	1.868
Krig 2	2.947	20.263	11	650	850	2.260	0.298	1.868

Surrogate performance is shown in Table 5.7. The EXTENDED POLL was never triggered in either MVPS run. The second-order Kriging surrogate did not account for as much improvement as it did in the previous run. This is attributed to a crash of JETPEN in the initial points. The first surrogate was run as a zero-order model since the second-order model requires 15 valid design sites and only 14 were available. Overall, the surrogates were still the most significant source of improvement by the algorithm, and again this improvement came during the first few surrogate applications.

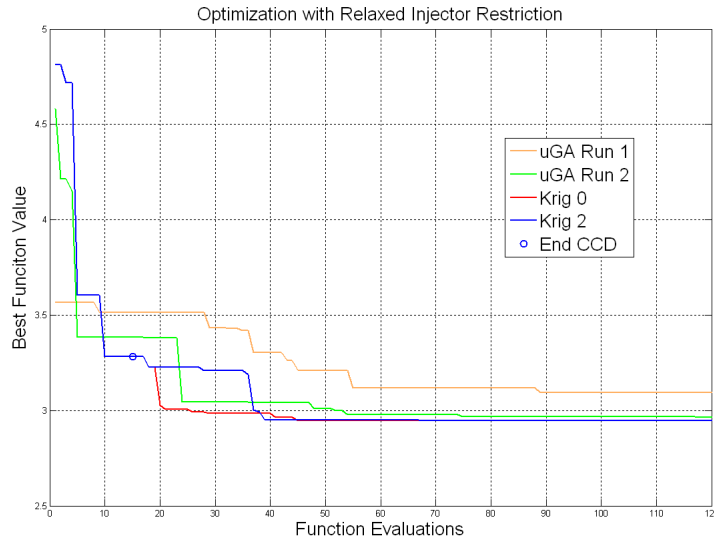


Figure 5.4 Algorithm Convergence Comparisons

Table 5.7 Incumbent Improvement Breakdown by Algorithm Step

Improvement by Step		
Method	Krig 0	Krig 2
Search	65.71%	68.38%
Poll	22.40%	25.63%
N Poll	11.89%	6.00%
Ext Poll	0.00%	0.00%

5.3 HiFIRE Optimization: 1 Variable

The first optimization attempt for the HiFIRE design problem considered only a single variable, δ_j . The original design specified two parallel arrays of four injectors each; thus, N was fixed at four. Sources from the literature suggest that 15° injection is likely to be near optimal. The responses at 15° injection are shown in Table 5.8. MVPS with zero and second-order Kriging polynomials are applied to each of the three objective functions outlined in Section 4.2. A CCD in one variable is not possible so three initial points of 15° , 45° , and 75° are evaluated for use in the initial zero-order Kriging surrogate. Six initial points, 10° , 15° , 30° , 50° , 70° , and 90° , are used for the

initial second-order Kriging surrogate. In two of the three function forms, the first incumbent solution was the same for both models. In the optimization of $F_2(\vec{\mathbf{Y}})$, an initial point (30°) essentially landed on the optimum. This led to unsuccessful surrogates at every SEARCH.

Table 5.8 AFRL Design: Baseline 15° Results

15° Injection Results			
Obj Ftn	$f_1(\vec{\mathbf{Y}})$	$f_2(\vec{\mathbf{Y}})$	$f_1(\vec{\mathbf{Y}})$
F	6.211	2.605	4.037
15° Injection Response $\vec{\mathbf{Y}}$			
OBJ	y_1	y_2	y_3
Mach 6	2.09	2.09	2.61
Mach 7	1.92	1.92	2.27
Mach 8	1.75	1.75	2.09

Results from this single variable case follow the observation from the literature. Results for the individual objective function forms can be found in Tables 5.9–5.14. Optimal values for δ_j were found between 25° and 31° . Overall improvement was on the order of 1% over the baseline 15° injection.

Only the Frobenius norm objective function, $f_1(\vec{\mathbf{Y}})$, found improvement over 15° injection in all nine responses. However, this function form proved to be the most computationally expensive, with 43 and 49 function evaluations required by zero and second-order Kriging surrogates, respectively. The second objective function, $f_2(\vec{\mathbf{Y}})$, was uninteresting because y_{3,M_6} - the axial distance for stoichiometric fuel concentration decay in the plume at Mach 6 - was the maximum element at every iteration for which the injection angle was below 70° . In other words, the problem could have been equivalently expressed as

$$\min_{x \in \Omega} f_2(\vec{\mathbf{Y}}) = \min_{x \in \Omega} \left[\max_{i,j} \vec{\mathbf{Y}} \right] = \min_{x \in \Omega} [y_{3,M_6}]. \quad (5.1)$$

As could be expected with this definition, the response value for y_{3,M_6} shows the largest improvement over the baseline for this measure. Computational cost was

comparatively smaller, requiring 26 and 34 function evaluations, respectively. Again, MVPS with second-order Kriging surrogates had one initial point land essentially at the optimum point. In this case, the surrogate never found an improved solution, and all improvement was found on the POLL step.

Table 5.9 HiFIRE Design Optimization Results

$f_1(\vec{Y})$: Minimize Frobenius Norm of Response						
	0th Order Kriging Polys			2nd Order Kriging Polys		
	Best F	# Evals	δ_j	Best F	# Evals	δ_j
		6.147	43	25.907	6.141	49
	Response \vec{Y}			Response \vec{Y}		
OBJ	y_1	y_2	y_3	y_1	y_2	y_3
Mach 6	2.075	2.075	2.588	2.073	2.073	2.586
Mach 7	1.902	1.902	2.244	1.900	1.900	2.242
Mach 8	1.724	1.724	2.066	1.722	1.722	2.064

Table 5.10 $f_1(\vec{Y})$ Improvement Over Baseline (Best F)

Improvement over Baseline			
Minimize $F = f_1(\vec{Y})$			
OBJ	y_1	y_2	y_3
Mach 6	0.95%	0.95%	0.74%
Mach 7	1.28%	1.28%	1.07%
Mach 8	1.57%	1.57%	1.30%

Table 5.11 HiFIRE Design Optimization Results

$f_2(\vec{Y})$: Minimize Maximum Response Element						
	0th Order Kriging Polys			2nd Order Kriging Polys		
	Best F	# Evals	δ_j	Best F	# Evals	δ_j
		2.581	26	30.601	2.581	34
	Response \vec{Y}			Response \vec{Y}		
OBJ	y_1	y_2	y_3	y_1	y_2	y_3
Mach 6	2.240	2.240	2.582	2.240	2.240	2.581
Mach 7	1.895	1.895	2.408	1.895	1.895	2.407
Mach 8	1.716	1.716	2.058	1.716	1.716	2.058

Table 5.12 $f_2(\vec{Y})$ Improvement Over Baseline (Best F)

Improvement over Baseline			
Minimize $F = f_2(\vec{Y})$			
OBJ	y_1	y_2	y_3
Mach 6	-7.00%	-7.00%	0.92%
Mach 7	1.56%	1.56%	-6.23%
Mach 8	1.90%	1.90%	1.57%

Optimization of the stoichiometric fuel concentration decay distances, subject to constraints on y_1 and y_2 , failed to converge to a feasible point. No injection angles were found meeting the criteria for y_1 and y_2 . Consistent violations occurred only at Mach 6 flight conditions. These restrictions were subsequently dropped, and an optimum injection angle was found very near that of the Frobenius norm objective function, f_1 . Both optimization methods converged to nearly the same solution, with the second-order Kriging model finding only a slightly better solution. Surrogate performance is shown in Table 5.15.

Table 5.13 HiFIRE Design Optimization Results

$f_3(\vec{Y})$: Minimize Fuel Concentration Decay Distances						
	0th Order Kriging Polys			2nd Order Kriging Polys		
	Best F	# Evals	δ_j	Best F	# Evals	δ_j
		3.999	22	26.282	4.002	37
	Response \vec{Y}			Response \vec{Y}		
OBJ	y_1	y_2	y_3	y_1	y_2	y_3
Mach 6	2.245	2.245	2.587	2.076	2.076	2.589
Mach 7	1.901	1.901	2.243	1.904	1.904	2.245
Mach 8	1.723	1.723	2.065	1.725	1.725	2.067

5.4 HiFIRE Optimization: 2 Variables

The two-variable optimization of this problem, outlined in Section 4.1.3, returned unfavorable designs. Both injection angle (δ_j) and number of injectors (N) were considered for optimization. The final design chosen from each objective func-

Table 5.14 $f_3(\vec{Y})$ Improvement Over Baseline (Best F)

Improvement over Baseline			
Minimize $F = f_3(\vec{Y})$			
OBJ	y_1	y_2	y_3
Mach 6	-7.27%	-7.27%	0.71%
Mach 7	1.22%	1.22%	1.02%
Mach 8	1.49%	1.49%	1.23%

Table 5.15 Surrogate Performance by Objective Function

Surrogate Performance						
	$F = f_1(\vec{Y})$		$F = f_2(\vec{Y})$		$F = f_3(\vec{Y})$	
OBJ	Krig 0	Krig 2	Krig 0	Krig 2	Krig 0	Krig 2
Search	82.11%	75.39%	57.20%	0.00%	65.157%	74.25%
Poll	17.89%	24.61%	42.80%	100.00%	34.85%	25.75%

tion optimization is shown in Table 5.16. Each final design included a prohibitive number of injection ports. Paull and Stalker [55] cite experiments where poor mixing occurred with a large number of small injection ports. In these experiments mixing improved as the number of ports decreased. The final designs in Table 5.16 are unfavorable for this reason. This suggests that JETPEN has a modeling inadequacy for these designs. Performance measures for these design are assessed by JETPEN as significantly better in y_2 and y_3 , but mildly worse in y_1 as shown in Table 5.17.

Again, the quality of these estimates is deceptive; it is known that these designs result in poor mixing characteristics. It is interesting however, that the optimal injection angles are very similar to the designs of the single variable optimization. This suggests that the optimal injection angle may not be sensitive to the number of injectors.

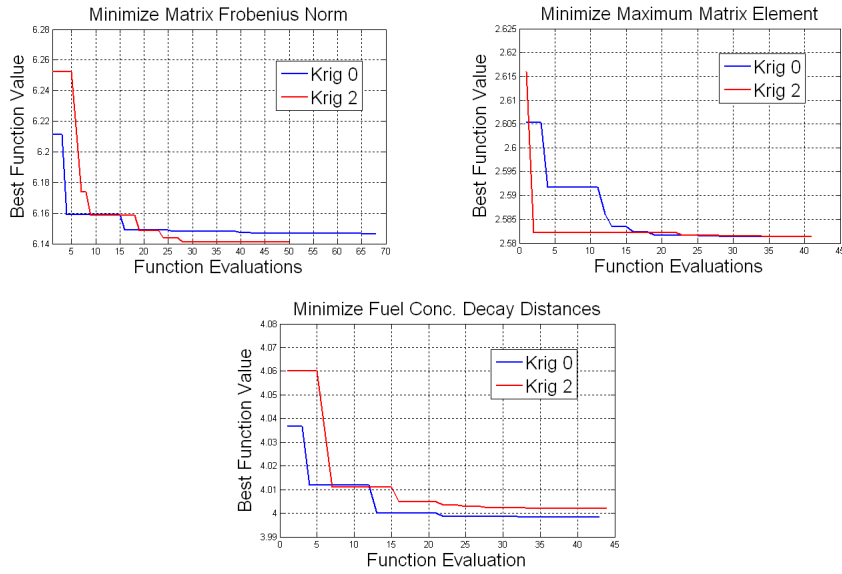


Figure 5.5 Algorithm Performance

Table 5.16 Designs from 2-Variable Optimization

OBJ	δ_j	Single Array N	Total N
$f_1(\vec{Y})$	21.86	28	56
$f_2(\vec{Y})$	32.77	31	62
$f_3(\vec{Y})$	32.77	31	62

Table 5.17 2-Variable Optimization Improvement Over Baseline

Improvement Over 15° Baseline			
	y_1	y_2	y_3
Mach 6	-1.50%	88.66%	18.46%
Mach 7	-1.40%	87.73%	13.88%
Mach 8	-1.62%	86.66%	14.99%

5.5 Summary

Re-optimization of the design studied by Payne identified several computationally less expensive techniques. Overall, MVPS with Kriging surrogates dominated the performance of the genetic algorithms, finding better solutions in less function evaluations. The best designs found by both this study and Payne's were essentially

the same, and the performance of the genetic algorithms were also similar in both studies. Expanding the possible number of injectors found little improvement over Payne's best designs.

Application of MVPS with Kriging surrogates to the HiFIRE design found minimal improvement over the baseline 15° injection angle. Fixing $N = 4$ and optimizing with respect to δ_j found optimal injection angles between 25° and 30° . Improvement of these designs was only about 1%. Optimizing both N and δ_j produced designs known to exhibit poor mixing. Very large values for N were favored in each optimization, but values for δ_j remained between 21° and 32° .

6. Conclusions and Future Recommendations

The optimization techniques applied in this research were effective in identifying optimal or near-optimal designs that are supported by the literature. The applied methods significantly reduced computational expense and show potential for future application with higher precision scramjet models. Several improvements are suggested here for further reducing computational cost and improving accuracy.

6.1 Final Design Evaluation

The final designs from each problem can be considered optimal or near-optimal within the bounds of this study. Results from re-optimizing the previous design problem found designs similar to those of the previous study [56]. Expanded investigation confirmed the quality of these solutions. The final design returned by the HiFIRE design optimization closely matches the recommendations in the literature and the views of researchers. This design, and similar designs, are candidates for follow-on analysis with higher fidelity methods for the HiFIRE program. A thorough CFD analysis should be performed on any candidate design prior to production. More realistically, several designs should be analyzed through CFD with a final design chosen on the basis of some other performance measures, such as thrust or I_{sp} .

The optimal HiFIRE injection angle near 30° is supported by both the literature and engineers at AFRL. Ortwerth [54] considers 15° to be the lower injection limit for good mixing performance. Mays *et al.* [47], as well as engineers at AFRL, consider optimal injection to be around 30° . Payne [56] found that mixing is fairly insensitive to injection angle when the angle is low. As could be expected, the results of this investigation showed a difference on the order of 1% between the two. This provides more evidence that the responses are fairly indifferent to changes at low injection angles. This should provide greater production flexibility for the HiFIRE program, assuming the results are confirmed through other means.

Low angle injection ports are difficult and costly to manufacture. It is very difficult to precisely drill a hole in a flat surface at very low angles, thus higher angles are preferable from a production point-of-view. The results of this study show that AFRL may pursue higher angles of injection at reduced production costs without a sacrifice in performance. Optimization of the number of injectors must be done with another, higher precision, analysis tool. Production of a 62-port injection array inside a 1"×4" combustor would be extremely difficult, if not impossible. Combine this difficulty with the known poor mixing properties of this design, and the drawbacks are clear.

6.2 Genetic Algorithms

MVPS with Kriging surrogates is superior to the genetic algorithms applied in this study. Genetic algorithms applied to a mesh have been developed [33, 34], allowing these versions to have convergence theory based on the ideas of Torczon [68] and Dennis and Schnabel [27] that gave GPS and its variants provable convergence. An interesting follow on study may combine these genetic algorithms with newer versions of MVPS to optimize designs using a more computationally expensive CFD code.

6.3 Future Applications of MVPS

This study conclusively demonstrates the quality of MVPS with Kriging surrogates, as applied in this study to hypersonic design optimization. The required number of function evaluations was greatly reduced by the application of MVPS without a sacrifice in final objective function value. When compared to other optimization techniques such as μ GAs, MVPS augmented with surrogates reduced the required number of function evaluations by as much as 55.5%.

The addition of Kriging surrogates to MVPS improved convergence and reduced computational cost when compared to MVPS alone. It is interesting to note that as the degree of Kriging polynomials increased, so did the total percentage of incumbent improvement attributed to them, as shown in Table 5.4. These surrogates comprised as much as 78% of the incumbent improvement after the initial CCD points, with the POLL step dominating the remaining improvement. Non-surrogate MVPS runs showed that the POLL step alone accounted for nearly 95% of the overall improvement.

6.4 *Future Recommendations*

JETPEN's modeling inadequacy discussed in Section 5.4 is a potential source of future investigation. If JETPEN is to be applied for future optimizations, then this problem must either be fixed or taken into consideration. JETPEN's failure to return axial data for some candidate designs, discussed in Section 4.6, is another potential future fix. Future versions of JETPEN may include additional modeling for other types of fuel injection and non-circular injection ports. Future applications could use a higher fidelity CFD model with JETPEN as a surrogate, with a Kriging surrogate applied to the error between the two. Hopefully, the results of this investigation will lead to optimizations involving more advanced analysis methods other than JETPEN, such as SRGULL. It was used in developing the NASA X-43A and could provide higher quality results than JETPEN.

Future investigations may also apply different surrogates or different correlation models to Kriging surrogates. Other surrogates, such as kernel regression developed by Nadaraya and Watson or radial basis functions may provide better surrogates for this class of problems. For Kriging, other correlation models such as cubic splines, linear models, and spherical models can be investigated. Any of these models may perform better, based on an analysis of the underlying system behavior.

An alternative optimization algorithm for possible future application is Mesh Adaptive Direct Search (MADS). It is a generalization of pattern search for non-linear constraints. MADS has also been extended to the mixed variable domain [6] and stochastic responses [61], and may be a desirable alternative to GPS in problems with known discontinuities, poor behavior, and non-linear constraints. The HiFIRE problem that minimized one set of performance measures, subject to constraints on the other performance measures, is better suited for MADS. The constraints on the other performance measures are essentially non-linear constraints. It may be possible to take a more thorough multi-objective formulation of this problem, and possibly generate the Pareto front.

Other initial designs and poll methods may also be investigated. A CCD is typically not efficient in design problems with a large set of variables. A Latin hypercube or fractional orthogonal design is more efficient than a CCD for these problems. These can also be applied in the initial SEARCH step or can be used at the POLL step, giving greater flexibility to the user.

In the end, these results should warrant further investigation using higher performance CFD codes. MVPS gives a user much greater flexibility in the likely case where the number of function evaluations is limited by a fixed budget. This flexibility, combined with reduced computational costs, holds the potential to open a whole new realm of performance for future scramjet designs.

Bibliography

1. Aarts, E. and J.K. Lenstra. *Local Search in Combinatorial Optimization*. Princeton, NJ: Princeton University Press, 2003.
2. Abramson, M.A. *Pattern Search Algorithms for Mixed Variable General Constrained Optimization Problems*. Ph.D. thesis, Department of Computational and Applied Mathematics, Rice University, August 2002.
3. Abramson, M.A. “Second-Order Behavior of Pattern Search”. *SIAM Journal on Optimization*, 16(2):315–330, 2005.
4. Abramson, M.A. “NOMADm Optimization Software”. <http://www.ait.edu/en/ENC/Faculty/MAbramson/NOMADm.html>. 2007.
5. Abramson, M.A., C. Audet, and J.E. Dennis, Jr. “Filter Pattern Search Algorithms for Mixed Variable Constrained Optimization Problems”. *Pacific Journal of Optimization*, 3(3): 477–500, 2007.
6. Abramson, M.A., C. Audet, J.W. Chrissis, and J.G. Walston. “Mesh Adaptive Direct Search Algorithms for Mixed Variable Optimization.” GERAD Technical Report 2007–47. *Optimization Letters*, to appear.
7. Ali, M. and A.K.M. Sadrul Islam. “Study on main flow and fuel injector configurations for Scramjet applications.” *International Journal of Heat and Mass Transfer* 49: 3634–3644 (2006).
8. Anderson, C.D. “Development and Testing of an Integrated Liquid-Fuel-Injector/Plasma-Igniter for Scramjets.” M.S. Thesis. Virginia Polytechnic Institute and State University, Blacksburg VA, February 2004.
9. Anderson, G.Y., P. Reagon, P. Gooderum, and W.R. Russin. “Experimental Investigation Concept for Scramjet Application of a Swept-Strut Fuel-Injector,” Hampton, VA: Langley Research Center. NASA TN D-8454, 1977.
10. Anderson, J.D., Jr. *Hypersonic and High Temperature Gas Dynamics*, McGraw-Hill, New York, 1989.
11. Auneau, I., G. Pascal, and P. Duveau. “Design and Optimization Methods for Scramjet Inlets.” *ALAA 6th International Aerospace Planes and Hypersonics Technologies Conference*. April 3–7 1995.
12. Audet, C. and J.E. Dennis, Jr. “Pattern Search Algorithms for Mixed Variable Programming”. *SIAM Journal on Optimization*, 11(3):573-594, 2000.
13. Audet, C. and J.E. Dennis, Jr. “Analysis of Generalized Pattern Searches”. *SIAM Journal on Optimization*, 13(3):889–903, 2003.

14. Ben-Yakar, A., G.M. Mungal, and R.K. Hanson. "Time Evolution and Mixing Characteristics of Hydrogen and Ethylene Transverse Jets in Supersonic Cross-flows." *Physics of Fluids*, 18: 2006
15. Bertin, J.J. *Hypersonic Aerothermodynamics*. Washington DC: American Institute fo Aeronautics and Astronautics, Inc., 1994.
16. Billig, F.S., R.C. Orth, and M. Lasky. "A Unified Analysis of Gaseous Jet Penetration," *AIAA Journal*, 9:1048–1058 (1971).
17. Billig, F.S. and J.A. Schetz. "Analysis of Penetration and Mixing of Gas Jets in Supersonic Cross Flow," 4th *International Aerospace Planes Conference*. December 1-4 1992.
18. Box, G.E.P. and K.B. Wilson. "On the Experimental Attainment of Optimal Conditions," *Journal of the Royal Statistical Society*, B(13): 1–38, 1951.
19. Brewer, K.M. *Exergy Methods for the Mission-Level Analysis and Optimization of Generic Hypersonic Vehicles* M.S. Thesis. Virginia Polytechnic Institute and State University, Blacksburg VA, May 2006.
20. Brown, M. , N.R. Mudford, A.J. Neely, and T. Ray. "Robust Design Optimization of Two-Dimensional Scramjet Inlets." *14th AIAA/AHI Space Planes and Hypersonic Systems and Technologies Conference*. November 6–9 2006.
21. Chernyavsky, B., V. Stepanov, K. Rasheed, M. Blaize, and D. Knight. "3-D Hypersonic Inlet Optimization Using a Genetic Algorithm." *AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*. July 13–15 1998.
22. Choi, J., F. Ma, and V. Yang. "Dynamics Combustion Characteristics in Scramjet Combustors with Transverse Fuel Injection." *41st AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit* July 10–13 2005.
23. Clarke, F.H. *Optimization and Nonsmooth Analysis*. Wiley, New York, 1983.
24. Curran, E.T. and S.N.B. Murthy. *Scramjet Propulsion*. AIAA Progress in Astronautics and Aeronautics, v 189, ed. 2000.
25. Davis, C. "Theory of Positive Linear Dependence". *American Journal of Mathematics*, 76(4): 733–746, 1954.
26. Dash, S.M., R. Ungewitter, J. Ott, and J. Papp. "Computational Modeling Advances Supporting Hypersonic Scramjet Design." *14th AIAA/AHI Space Planes and Hypersonic Systems and Technologies Conference*. November 6–9 2006.
27. Dennis, J.J. and R.B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall, 1983.
28. Dunlap, J.E. *On the Use of Surrogate Functions for Mixed Variable Optimization of Simulated Systems*. MS thesis, AFIT/GOR/ENS/05. Graduate School of

Operational Sciences, Air Force Institute of Technology (AU), Wright Patterson AFB, OH, March 2005.

29. Foster, N.F., G. Dulikravich, and J. Bowles. “Three-Dimensional Aerodynamic Shape Optimization Using Genetic Evolution and Gradient Search Algorithms.” *AIAA 34th Aerospace Sciences Meeting and Exhibit*. January 15–18 1996.
30. Gantovnik, V.B., C. Anderson-Cook, Z. Gurdal, and L. Watson. “A Genetic Algorithm with Memory for Mixed Discrete-Continuous Design Optimization.” *Computers and Structures*. 81: 2003–2009 (August 2003).
31. Griebel, M., T. Dornseifer, and T. Neunhoffer. *Numerical Simulation in Fluid Dynamics: a Practical Introduction*. SIAM, 1998.
32. Gruber, M.R., A.S. Nejad, T.H. Chen, and J.C. Dutton. “Mixing and penetration studies of sonic jets in a Mach 2 freestream.” *Journal of Propulsion & Power*, 11(2): 315–323, 1995.
33. Hart, W.E. “A Generalized Stationary Point Convergence Theory for Evolutionary Algorithms.” In *Proceedings of the International Conference on Genetic Algorithms* 1997.
34. Hart, W.E. “A Stationary Point Convergence Theory for Evolutionary Algorithms.” In *Foundations of Genetic Algorithms*, 4: 325-342, 1997.
35. He Hao, S.T. and J. Yu. “Three-Dimensional Simulation of Transverse Injection in a Supersonic Flow by the CESE Method.” *41st Aerospace Sciences Meeting and Exhibit* January 6–9 2003.
36. Heiser, W.H. and D.T. Pratt. *Hypersonic Airbreathing Propulsion*. Washington DC: American Institute of Aeronautics and Astronautics, Inc., 1994.
37. Hirschel, E.H. *Basics of Aerothermodynamics*. Reston,VA: American Institute of Aeronautics and Astronautics, Inc., 2005.
38. Kimmel, R. L. , D. Adamczak, D. Gaitonde, A. Rougeux, and J.R. Hayes. “HIFiRE-1 Boundary Layer Transition Experiment Design.” *45th AIAA Aerospace Sciences Meeting and Exhibit* January 8–11 2007.
39. Krishnakumar, K. “Micro-Genetic Algorithms for Stationary and Non-Stationary Function Optimization,” SPIE: Intelligent Control and Adaptive Systems, Vol. 1196, Philadelphia, PA, 1989.
40. Kutschenreuter, P. “Supersonic Flow Combustors,” in *Scramjet Propulsion*, Curran, E.T. and S.N.B. Murthy, ed. 2000. AIAA Progress in Astronautics and Aeronautics, v 189.
41. Lewis, R. M. and V. Torczon. “Pattern Search Methods for Bound Constrained Minimization”. *SIAM Journal on Optimization*, 9(4):1082–1099, 1999.

42. Lewis, R. M. and V. Torczon. "Pattern Search Methods for Linearly Constrained Minimization". *SIAM Journal on Optimization*, 10(3):917-941, 2000.
43. Little, G. "Mach 20 or Bust," *Air & Space Magazine*, September 2007.
44. Lophaven, S. N., H.B. Nielsen, and J. Søndergaard. "Aspects of the MATLAB Toolbox DACE". Report IMM-REP-2002-13, Informatics and Mathematical Modelling, Danish Technical University, 2002.
45. Lophaven, S. N., H. B Nielsen, and J. Søndergaard. "DACE Surrogate Models". <http://www2.imm.dtu.dk/hbn/dace>.
46. Markell, K.C. *Exergy Methods for the Generic Analysis and Optimization of Hypersonic Vehicle Concepts*. MS thesis. Virginia Polytechnic Institute and State University, February 2005.
47. Mays, R.B., R.H. Thomas, and J.A. Schetz. "Low Angle Injection Into a Supersonic Flow," 25th AIAA/ASME/SAE/ASEE *Joint Propulsion Conference*. July 10-12 1989.
48. McClinton, C.R. *The Effect of Injection Angle on the Interaction Between Sonic Secondary Jets and a Supersonic Free Stream*. Technical Report NASA TN D-6669, Hampton, VA: NASA Langley Research Center, February 1972.
49. Morgan, R. G. "Supersonic Combustion with Transverse, Circular Wall Jets, Shock Tunnel Studies for Scramjet Phenomena," Supplement 5, NASA CR 182096, 1990, pp. 20-39.
50. Muhlenbein, H. "Genetic Algorithms," in *Local Search in Combinatorial Optimization*, first ed. E. Aarts and J.K. Lenstra, Eds. Princeton: Princeton University Press, 2003, pp.137.
51. Myers, R. H. and D. Montgomery. *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. New York: John Wiley & Sons.
52. "NASA HyperX Program Demonstrates Scramjet Technologies" NASA Facts. NASA. FS20041098LaRC. October 10, 2004.
53. NIST Standard Reference Database Number 69, <http://webbook.nist.gov/chemistry/fluid/>, June 2005.
54. Ortwerth, P. J. "Scramjet Flowpath Integration," in *Scramjet Propulsion*, Curran, E.T. and S.N.B. Murthy, ed. 2000. AIAA Progress in Astronautics and Aeronautics, v 189.
55. Paull, A. and R.J. Stalker. "Scramjet Testing in the T3 and T4 Hypersonic Impulse Facilities," in *Scramjet Propulsion*, Curran, E.T. and S.N.B. Murthy, ed. 2000. AIAA Progress in Astronautics and Aeronautics, v 189.

56. Payne, M.D. *A Variable-Complexity Modeling Approach to Scramjet Fuel Injection Array Design Optimization*. M.S. Thesis, AFIT/GOR/ENS/98. Graduate School of Operational Sciences, Air Force Institute of Technology (AU), Wright Patterson AFB, OH, March 1998.
57. Pichitlamken, J., and B.L. Nelson. "Selection-of-the-best procedures for optimization via simulation." In *Proceedings of the 2001 Winter Simulation Conference*, B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, Eds., Institute of Electrical and Electronics Engineers, pp. 401-407.
58. Rowan, S.A. and A. Paull. "Performance of a Scramjet Combustor with Combined Normal and Tangential Fuel Injection." *Journal of Propulsion and Power*. 22: 1334–1338 (November-December 2006).
59. Schetz, J.A. and F.S. Billig. "Penetration of Gaseous Jets Injected Into a Supersonic Stream," *Journal of Spacecraft and Rockets*, 3 (11):16581665 (November 1966).
60. Srinivasan, R. *Improved Understanding and Control of High-Speed Jet Interaction Flows*. PhD dissertation. Texas A&M University, College Station TX, 2005.
61. Sriver, T.A. *Pattern Search Ranking and Selection Algorithms for Mixed-Variable Optimization of Stochastic Systems*. PhD thesis. Air Force Institute of Technology, Wright Patterson AFB, OH, 2004.
62. Starkey, R.P. "Scramjet Optimization for Maximum Off-Design Performance," 40th AIAA/ASME/SAE/ASEE *Joint Propulsion Conference and Exhibit*. July 11–14 2004.
63. Steffen, C.J. Jr., R.B. Bond, and J.R. Edwards. "Three Dimensional CFD Analysis of the GTX Combustor." *Combustion, Airbreathing Propulsion, Propulsion Systems Hazards, and Modelling and Simulation Subcommittees Joint Meeting sponsored by the Joint Army-Navy-NASA-Air Force*. April 8-12 2002.
64. Steffen, C.J. Jr. "Fuel Injector Design Optimization for an Annular Scramjet Geometry." *41st Aerospace Sciences Meeting and Exhibit*. January 6-9 2003.
65. Takita, K., H. Nakane, and G. Masuya. "Optimization of double plasma jet torches in a scramjet combustor." *Proceedings of the Combustion Institute* 31: 2513-2520 (2007).
66. Tharaldson, D. *Optimization of a Multi-Echelon Repair System via Generalized Pattern Search with Ranking and Selection: A Computational Study*. MS thesis, AFIT/GOR/ENS/06. Graduate School of Operational Sciences, Air Force Institute of Technology (AU), Wright Patterson AFB, OH, March 2006.
67. Thomas, R.H., J.A. Schetz, and F.S. Billig. "Gaseous Injection in High Speed Flow." *Ninth International Symposium on Air Breathing Engines*. September 3–8 1989.

68. Torczon, V. "On the Convergence of Pattern Search Algorithms". *SIAM Journal on Optimization*, 7(1):1-25, 1997.
69. Ungewitter, R.J., J.D. Ott, V. Ahuja, and S.M. Dash. "CFD Capabilities for Hypersonic Scramjet Propulsive Flowpath Design." *40th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*. July 11–14 2004.
70. Ungewitter, R.J., J.D. Ott, and S.M. Dash. "Advanced Modeling Methods for Hypersonic Scramjet Evaluation," *42nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*. July 9–12 2006.
71. Vinogradov, V.A., Y.M. Shikhman, and C. Segal. "A Review of Fuel Pre-injection in Supersonic, Chemically Reacting Flows," *Applied Mechanics Reviews*, 60: 139–148 (July 2007).
72. Wilson, M., R. Bowersox, and D. Glawe. "The Role of Downstream Ramps on Penetration and Mixing Enhancement for Supersonic Injection Flows." *33rd AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*. July 7–9 1997.
73. Wolpert, D.H. and W.G. Macready. "No Free Lunch Theorems for Optimization," *IEEE Transactions on Evolutionary Computation*, 1: 67–82 (April 1997).
74. Xu Xu, Xu Dajun. "Optimization Design for Scramjet and Analysis of its Operation Performance." *Acta Astronautica* 57: 390–403 (March 2005).

Appendix A. HiFIRE Mission Parameters

Combustion Parameters			
Parameter	Description	Value	Units
Fuel	Ethylene		
m_w	Fuel Molecular Weight	28	
f_{ST}	Stoichiometric Fuel/Air Ratio	0.0678	
ϕ	Equivalence Ratio	1	
N	Total # Fuel Injectors	8	
d_j^*	Injector Diameter	0.125	Inch
A_j	Fuel Injector Area	0.098175	Inch ²
T_{T_j}	Jet Total Temperature	540	°Rankine
R	Ideal Gas Constant	0.07092	$\frac{BTU}{lbm-R}$

Table A.1 HiFIRE Combustion Parameters

Flowpath Parameters			
Parameter	Description	Value	Units
l	Combustor Width	4	Inch
l_f	Fueled Combustor Width	4	Inch
l_{nf}	Unfueled Combustor Width	0	Inch
h	Combustor Height	1	Inch
A_c	Combustor Area	4	Inch ²

Table A.2 HiFIRE Flowpath Parameters

Fuel Flow Conditions							
Flight M	\dot{m}_j	M_j	T_{S_j}	P_{S_j}	P_{T_j}	γ_j	c_{p_j}
6	0.405	1	490	109.1	193.4	1.202	0.431
7	0.343	1	489	92.0	163.5	1.209	0.423
8	0.285	1	487	76.1	135.5	1.216	0.416

Table A.3 HiFIRE Fuel Flow Conditions

Air Flow Conditions								
Flight M	\dot{m}_a	M_a	T_{S_a}	P_{S_a}	P_{T_a}	T_{T_a}	γ_a	c_{p_a}
6	5.97	2.41	1408	26.0	390.8	2730	1.323	0.277
7	5.06	2.86	1456	18.9	595.7	3368	1.321	0.278
8	4.21	3.25	1633	14.7	880.6	4334	1.313	0.283

Table A.4 HiFIRE Air Flow Conditions

Appendix B. Dependent Variables

Injector Area:

$$A_j = N\pi \left(\frac{d_j^*}{2}\right)^2 \quad (\text{B.1})$$

Fuel Static Temperature:

$$T_{S_j} = \frac{T_{T_j}}{1 + \left(\frac{\gamma_j - 1}{2}\right)} \quad (\text{B.2})$$

Fuel Static Pressure:

$$P_{S_j} = \frac{\dot{m}_j R_j T_{S_j}}{M_j A_j \sqrt{32.2 \gamma_j R_j T_{S_j}}} \quad (\text{B.3})$$

Fuel Total Pressure:

$$P_{T_j} = P_{S_j} \left[1 + \frac{1}{2} (\gamma_j - 1) M_j^2\right]^{\frac{\gamma_j}{\gamma_j - 1}} \quad (\text{B.4})$$

Air Static Temperature:

$$T_{S_a} = \frac{32.2 \gamma_a}{R_a} \left(\frac{A_c P_{S_a} M_a}{\dot{m}_a}\right)^2 \quad (\text{B.5})$$

Air Total Temperature:

$$T_{T_a} = T_{S_a} \left[1 + \frac{1}{2} (\gamma_a - 1) M_a^2\right] \quad (\text{B.6})$$

Air Total Pressure:

$$P_{T_a} = P_{S_a} \left[1 + \frac{1}{2} (\gamma_a - 1) M_a^2\right]^{\frac{\gamma_a}{\gamma_a - 1}} \quad (\text{B.7})$$

Ratio of Specific Heats:

$$\gamma = \frac{C_p}{C_p - R} \quad (\text{B.8})$$

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE (DD-MM-YYYY) 28-03-2008		2. REPORT TYPE Master's Thesis		3. DATES COVERED (From — To) Mar 2007 — Mar 2008		
4. TITLE AND SUBTITLE SCRAMJET FUEL INJECTION ARRAY OPTIMIZATION UTILIZING MIXED VARIABLE PATTERN SEARCH WITH KRIGING SURROGATES			5a. CONTRACT NUMBER			
			5b. GRANT NUMBER			
			5c. PROGRAM ELEMENT NUMBER			
			6. AUTHOR(S) Sparkman, Bryan Capt, USAF			
			5d. PROJECT NUMBER			
			5e. TASK NUMBER			
5f. WORK UNIT NUMBER						
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way WPAFB OH 45433-7765				8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GOR/ENS/08-19		
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) AFRL/RZAS, Bldg 18 Attn: Dr. Mark R. Gruber 1950 5th Street WPAFB OH 45433-7765 DSN: 785-4539 e-mail: mark.gruber@wpafb.af.mil				10. SPONSOR/MONITOR'S ACRONYM(S)		
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)		
12. DISTRIBUTION / AVAILABILITY STATEMENT Approval for public release; distribution is unlimited.						
13. SUPPLEMENTARY NOTES						
14. ABSTRACT Fuel-air mixing analysis of scramjet aircraft is often performed through experimental research or computational fluid dynamics (CFD) algorithms. Design optimization with these approaches is often impossible under a limited budget due to their high cost per run. This investigation uses JETPEN, a known inexpensive analysis tool, to build upon a previous case study of scramjet design optimization. Mixed Variable Pattern Search (MVPS) is compared to evolutionary algorithms in the optimization of two scramjet designs. The first revisits the previously studied approach and compares the quality of MVPS to prior results. The second applies MVPS to a new scramjet design in support of the Hypersonic International Flight Research Experimentation (HiFIRE). The results demonstrate the superiority of MVPS over evolutionary algorithms and paves the way for design optimization with more expensive approaches.						
15. SUBJECT TERMS mixed variable optimization, pattern search, Kriging, surrogate functions, HiFIRE, scramjets, scramjet fuel injection, JETPEN, evolutionary algorithms, hypersonic design optimization						
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON	
a. REPORT	b. ABSTRACT	c. THIS PAGE			James W. Chrissis, PhD, Associate Professor (ENS)	
U	U	U	UU	105	19b. TELEPHONE NUMBER (include area code) (937) 785-3636, ext 4606; e-mail: james.chrissis@afit.edu	