EFFICIENT SEQUENTIAL RELIABILITY-BASED DESIGN OPTIMIZATION WITH ADAPTIVE KRIGING INVERSE RELIABILITY ANALYSIS

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF AERONAUTICS &
ASTRONAUTICS
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Richard Walter Fenrich
December 2018

This dissertation is online at: http://purl.stanford.edu/sn120rw8191

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Juan Alonso, Primary Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Gianluca Iaccarino

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Mykel Kochenderfer

Approved for the Stanford University Committee on Graduate Studies.

Patricia J. Gumport, Vice Provost for Graduate Education

This signature page was generated electronically upon submission of this dissertation in electronic format. An original signed hard copy of the signature page is on file in University Archives.

Preface

In engineering, the reliability of a design is a primary quantity of interest that encompasses both performance and safety requirements. Standard engineering systems such as aircraft, cars, buildings, and bridges are all subject to the whims of nature and their users and as such require a high degree of reliability, with potentially disastrous consequences otherwise. Traditional approaches for design build in margin by using safety factors or conservative material properties. However such methods are at best limited in scope and at worst *ad hoc* and may lead to overly conservative designs with decreased performance as well as under conservative designs with insufficient reliability. In reliability-based design optimization (RBDO) the uncertainties in a design's operating environment, material properties etc. are directly taken into account to generate a design that achieves a desired reliability target. However the solution of RBDO problems, particularly in complex high-dimensional applications (10–100 variables) such as those often found in aerospace engineering, remains challenging due to the need to estimate small probabilities of failure while performing a nonlinear optimization with quantities of interest that are typically expensive to evaluate (hours to days).

In this thesis, new methods for reliability-based design optimization are presented. In particular the Adaptive Kriging Inverse Reliability Analysis (AKIRA) algorithm and a multifidelity sequential RBDO algorithm are introduced and demonstrated on a complex multidisciplinary supersonic nozzle design problem. AKIRA adaptively refines kriging models of the design problem's quantities of interest while estimating failure levels for a prescribed system probability of failure. The algorithm demonstrates competitive performance with other reliability analysis algorithms while also benefiting from the solution of the inverse reliability analysis problem during RBDO. The proposed sequential RBDO algorithm mitigates the cost of solving the RBDO problem by decoupling the optimization and reliability analyses, thereby reducing its solution to a series of deterministic optimizations. The method is motivated by anchored decomposition for function approximation, has guaranteed convergence inherited from trust region methods, and is shown in certain cases to be a generalization of existing sequential RBDO methods. It also derives enhanced efficiency by incorporating lower-fidelity models when available. The final demonstration of the proposed algorithms on an industrial-type problem, the supersonic nozzle, shows that the solution of RBDO problems for complex realistic engineering applications is well within reach.

Acknowledgments

First and foremost I would like to recognize my Lord and the *great things He has done for me (Ps 126:3)*. My time at Stanford University has truly been blessed and I am grateful for its challenges and joyful moments.

My advisor Juan Alonso has been a source of academic inspiration throughout my PhD. I am indebted to his encouragement and guidance, and the freedom to investigate my own research interests. Prof. Gianluca Iaccarino has provided a welcoming and stimulating intellectual environment for the discussion of uncertainty quantification and reliability-based design optimization. I am grateful for his guidance. Prof. Mykel Kochenderfer has provided valuable and discerning feedback and suggestions throughout my PhD. I am thankful for our discussions, as well as his interesting classes. I have also had the opportunity to interact with a variety of other world-class experts while at Stanford. Prof. Michael Saunders' kindness and introduction to numerical optimization has been invaluable and much appreciated. Prof. Gerd Infanger's introduction to the field of stochastic optimization helped chart the course for the remainder of my PhD. I would also like to acknowledge Prof. Cantwell who has taught me everything I know about nozzles and has always been generous with time for my research questions.

The faculty and staff of the Stanford Aeronautics & Astronautics Department have been supportive and instrumental in ensuring that one day I would graduate. I am especially thankful for the patience and support of Patrick Ferguson, Robin Murphy, Jenny Scholes, Corinna Haussecker, and Ralph Levine. I would also like to acknowledge the support of the Vice Provost for Graduate Education Stanford Graduate Fellowship and the DARPA Enabling Quantification of Uncertainty in Physical Systems program for their financial support, which enabled me to study questions of interest and importance for engineering systems.

I am indebted to the work of Dr. Victorien Menier and Dr. Philip Avery for their aid in building a robust coupled computational model for a supersonic nozzle system. Without their help this dissertation would remain a solely academic exercise. I am also thankful for the support of the Aerospace Design Lab members who have provided a source of encouragement and a welcoming environment for pursuing challenging research questions. My collaborations with Zachary del Rosario have been a motivational source of energy for changing the world one publication at a time.

I would be remiss if I did not acknowledge the many outside of my academic bubble who helped sustain me during my time at Stanford. Michael and Katie Barako, Joe Katz, Allison Yau, Roong Cheacharoen, Emilio Botero, and too many more to name have made my time at Stanford bright. Fr. Nathan, Fr. Xavier, and Lourdes Alonso have provided a source of joy and a reminder of the true meaning of life. My family too has been an endless source of encouragement and inspiration; they have given me a love for learning and taught me the importance of perseverance during challenging times. Words are unable to express my gratitude and love for my parents, sisters, grandparents, aunts, uncles, and cousins who have inspired me and reminded me what truly matters.

Finally my wife Colleen has been an unwavering support and fount of love and encouragement in all my endeavors, however difficult or challenging. I am eternally grateful for her patience, care, and love, and for prompting me to excel both academically and as a human being.

Richard W. Fenrich
Stanford, CA
November 15, 2018

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This dissertation concerns the subject of reliability, a topic which in various forms seems to have occupied human beings from the earliest of times. Perhaps the literary greats, the likes of Shakespeare, Shelley, Keats, Dickinson and others, were most vividly aware of the concept of reliability and how the world through time, death, or human folly somehow always manages to thwart the ideal of absolute reliance on someone or something[1]. Whether it be "nature's changing course untrimm'd" [116] or "thou, O uncontrollable!" in reference to the West Wind [120] (perhaps early references to aleatory uncertainty), the concept of reliability or lack thereof has drawn much interest. Indeed, leave it to a poet to endow the English language with the word reliability in the first place. Coleridge penned the word in a letter concerning his friend Robert Southey in 1816 [113].

The human desire to control reliability then is perhaps unsurprising and the remainder of this dissertation discusses this problem from the point of view of engineering design. In engineering, the profession that deals with the application of scientific knowledge to the solution of practical problems [78], the concept of reliability is particularly important as one typically desires a solution that reliably remains a valid solution to a particular problem. Problems of design are poignant since an engineering design must reliably meet performance and safety requirements. Noteworthy examples include aircraft, cars, skyscrapers, and bridges, all of which have disastrous consequences when designed with insufficient reliability. Thus the consideration of reliability in engineering design, or its complement failure, is important when formulating and solving such design problems. Such problems can be formulated mathematically, and the reliability of a design can be rigorously guaranteed when characterized appropriately. This dissertation proposes new methods for the formulation and solution of these so-called *reliability-based design optimization* (RBDO) problems and demonstrates them on applications from structural and aerospace engineering.

Why do engineers need to consider reliability in the first place? Paraphrasing the eloquent words of countless poets, uncertainties are an everpresent trait of reality. The strength of steel is not just

[1] The only exceptions appear to be God, death, and as Americans fondly say, taxes.

a single number conveniently written in the appendix of a textbook nor is the air temperature at 30,000 feet above sea level a constant governed by the 1970 Standard Atmosphere model. Material properties will vary by location and batch and weather conditions will vary with time and place; they are inherently uncertain. Such uncertainties are often called *aleatory*, from the Latin word *alea* meaning rolling of the dice. Another class of uncertainties also exists, *epistemic* uncertainties coming from the Greek word $\varepsilon\pi\iota\sigma\tau\eta\mu\eta$ (*episteme*) meaning knowledge. Therefore an epistemic uncertainty is one characterized by lack of knowledge and thus theoretically could be reduced (unlike aleatory uncertainties) given enough time, money, or investigation. A classical example in engineering design is model-form uncertainty, *i.e.* the choice of model fidelity which could always be increased to yield more accurate design analysis with greater computational expense. The prescription of an uncertainty as epistemic or aleatory has been argued to depend on the situation and context of a given problem [76]. This dissertation however only considers aleatory uncertainties.

As a result, the mere presence of uncertainty in a design's operating environment and the materials it is constructed from, among others, motivates the consideration of reliability in design problems. Traditional methods for engineering design build margin into a design to boost its reliability — common methods include safety factors, partial safety factors, and basis values — however such methods do not directly control a design's reliability and often operate in a deterministic fashion without considering uncertainty. The amount of margin built into a design by traditional methods is often unknown and as such can lead to both overly conservative designs with decreased performance as well as under conservative designs with insufficient reliability. Furthermore the presence of uncertainties means that the optimal design on paper is never the optimal design in practice because the uncertainties were never directly taken into account. Indeed the representation of reality distilled into the formulation of a deterministic design problem is frozen in a particular state and thus remains ignorant of the neverending fluctuations present in reality.

In this dissertation uncertainties are modeled with the mathematical theory of probability for use in design optimization, however the characterization of uncertainties is not a primary concern of this dissertation. Rather the focus is on algorithm development for reliability-based design optimization (RBDO) once uncertainties have been identified. Uncertainties are modeled with analytic distributions and are assumed to be independent *with the recognition* that more complex distributions and correlations between uncertainties can be handled with appropriate transformations with a few caveats [110, 79] or alternate methods. Such caveats include the nontrivial possibility that the transformation does not exist or that the transformation leads to a more nonlinear problem. However the treatment of more involved probabilistic modeling in RBDO problems, including the characterization of uncertainties in the estimated parameters of probability distributions, is left for future dissertations.

Provided the uncertainties have been properly characterized for a given design, the task at hand then is to solve a design problem that takes into account the reliability requirements of the design

given the input uncertainties. Figure 1.1 illustrates the basic problem of reliability-based design optimization (RBDO). Solving RBDO problems is challenging for several reasons:

1. The presence of uncertainties requires probabilistic analysis, *i.e.* the calculations of means, variances, and especially small probabilities of failure which are computationally expensive.

2. The dimension of the problem may be high, *i.e.* a large number of uncertainties and design variables may be present.

3. The quantities of interest in the problem are often nonlinear.

4. The quantities of interest in the problem are often computationally expensive to evaluate.

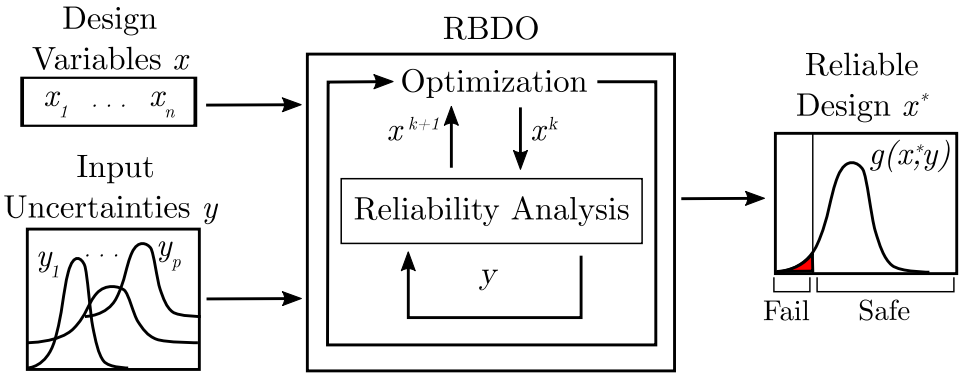


Figure 1.1: Schematic of reliability-based design optimization (RBDO). A design problem is characterized by design variables x and random parameters y which lead to random system responses, or *limit states*, $g(x, y)$. The goal is to design the system by choosing x such that the limit states reliably meet performance targets, *i.e.* $g(x, y)$ can only fail a small percentage $\bar{p}_f$ of the time which can be written as $P[g(x, y) \leq \bar{z}] \leq \bar{p}_f$ where $\bar{z}$ is the limit state level at which failure occurs. The RBDO algorithm solves an optimization problem while calculating the reliability of the limit states as necessary. An optimal design x^* is returned for which the limit states reliably meet performance targets.

For example, chapter 5 of this dissertation presents the RBDO of a supersonic nozzle system (see Figure 1.2). A nozzle is the aft end of an aircraft engine that is responsible for generating the thrust force that propels an aircraft forward. Since the nozzle channels the hot engine exhaust, it is subject to high temperatures and pressures. Naturally there are a variety of failure modes: failure due to excessive temperature or stress, as well as failure to generate enough thrust. In addition, a variety of uncertainties (up to 40 or more) are present in the nozzle system and include material properties, temperatures and pressures etc. As a nozzle designer, the goal is to modify the shape of the nozzle so that failure only occurs a small percentage of the time (say once every one million critical climb segments during flight) while ensuring that the nozzle is cost-effective. However to

design the nozzle one must simulate the nozzle, and the accurate analysis of the nozzle system requires the modeling of the aerodynamics, heat transfer, and structure of the nozzle. A single coupled analysis may take eight hours of compute time on an average laptop[2]. Thus the nozzle problem embodies each of the challenging aspects listed above for a realistic industrial-strength RBDO problem. Chapter 5 not only demonstrates the algorithms described in this dissertation on the nozzle problem, it also compares them to a standard deterministic design method and shows that solving the RBDO problem is both tractable and more effective at controlling reliability.

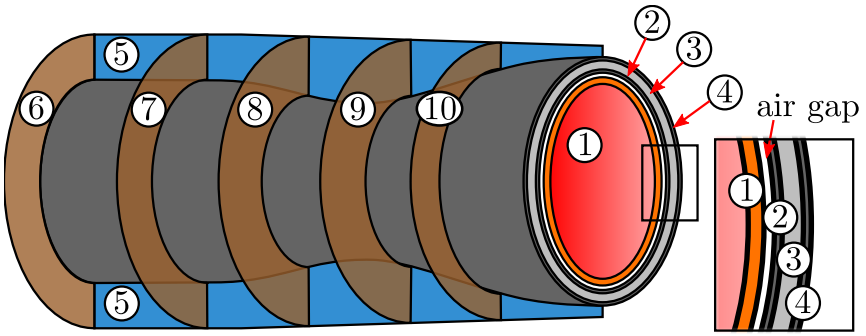


Figure 1.2: Schematic for the supersonic nozzle design problem. The problem requires the determination of the optimal shape of the innermost nozzle wall (component 1), as well as the thicknesses of each wall layer (components 1–4) and the air gap, the thicknesses of the stringers (components 5), and the location and thicknesses of the baffles (components 6–10). Wall layer and stringer thicknesses are allowed to vary along the length of the nozzle. Since the exhaust gases exiting the nozzle are at high pressure and temperature, assessing reliability with respect to thermal and structural failure is critical. In addition, the thrust force of the nozzle must reliably meet or exceed a certain performance requirement. Reliability of the nozzle is affected by uncertainties in the environment such as inlet pressures and temperatures, as well as the variability of material properties. Chapter 5 contains a full description of the nozzle problem and shows results for reliable nozzle designs.

Existing methods for solving RBDO problems have focused on increasing the efficiency of the solution process through several means; the sole goal in each case is to reduce the number of function evaluations needed to obtain a solution. One common tactic for reducing the cost of RBDO problems is to use efficient methods for *reliability analysis*, that is the calculation of small probabilities of failure, by focusing on the region of failure for the design. Chapter 2 introduces the concept of reliability and reviews methods for efficient calculation of probabilities of failure. Another common tactic is to reformulate the design problem or solve a series of approximate design problems which are easier to solve. Chapter 4 introduces and reviews several classes of RBDO problem-solving methods. Finally another tactic for simplifying high-dimensional problems is to employ dimension reduction techniques. The use of dimension reduction is not extensively covered in this dissertation, however one method for motivating dimension reduction, Sobol' sensitivity indices, is used and is described

[2] 3D RANS aerodynamic analysis and FEM thermal and structural analyses on 4 cores.

in detail in Appendix E.

The remainder of this dissertation reviews the existing state of the art in reliability-based design optimization and presents improvements to reliability analysis and the solution of RBDO problems. The focus is on enabling RBDO for complex, high-dimensional realistic design problems. To this end, the supersonic nozzle problem presented and solved in chapter 5 is the motivating problem and final demonstration of the techniques developed in this dissertation. These techniques include a new algorithm called the Adaptive Kriging Inverse Reliability Analysis (AKIRA) proposed in chapter 3 and an efficient sequential RBDO method derived from a new angle using anchored decomposition in chapter 4. The additional infusion of lower-fidelity models yields a multifidelity RBDO method and further enhances efficiency. A detailed description of the contributions of this dissertation is presented in the next section.

Chapter 2 introduces the mathematical definition of reliability and reviews methods for reliability analysis. Methods can be broadly separated into four categories: sampling (section 2.2), most probable point (MPP, section 2.3), response surface (section 2.4), and hybrid methods which combine aspects of the first three (section 2.5). To date most RBDO problems have been solved using MPP-based methods, which determine the point that contributes the most to the failure probability of the design and uses information at that point to characterize the design and make decisions for improving it [77, 128, 8, 41, 3, 40, 5, 4, 14, 117, 67]. The shortcoming of such methods is primarily the loss of accuracy when multiple MPPs exist or the design is highly nonlinear [115, 128, 117, 83]. Chapter 2 also discusses the differences between the forward reliability analysis and the inverse reliability analysis problems in section 2.1.5 and why the inverse problem is preferable for use in RBDO. Finally the calculation of design sensitivities for use in optimization is reviewed in section 2.6.

Chapter 3 proposes a new algorithm for the reliability analysis of an engineering system called the Adaptive Kriging Inverse Reliability Analysis (AKIRA). AKIRA constructs kriging response surfaces for the quantities of interest in a RBDO problem (section 3.2) and solves a system-level inverse reliability analysis for use in RBDO (section 3.4). Since kriging response surfaces contain a metric estimating their inaccuracy, *i.e.* the *kriging variance*, they can be adaptively refined in regions deemed to be most important for the estimation of failure probability. Section 3.3 presents a new pointwise heuristic for refinement which is more efficient than other pointwise metrics. Existing kriging-based reliability analysis methods primarily differ in their method of refinement and are reviewed in sections 3.1 and 3.3. Section 3.5 reviews the calculation of design sensitivities using AKIRA and section 3.6 presents a new method based on cokriging for the estimation of design sensitivities when function evaluations are expensive. Finally, AKIRA is demonstrated in section 3.7 on several common structural test problems and the nozzle problem and is shown to be either more efficient or more accurate than two well-accepted reliability analysis techniques.

Chapter 4 introduces the field of reliability-based design optimization and proposes a multifidelity sequential method using anchored decomposition for efficient RBDO. Traditional methods for

accounting for uncertainties such as safety factors, partial safety factors, and basis values are first reviewed in section 4.2. Next each of the three main classes of RBDO methods, double-loop, sequential, and unilevel, are reviewed in section 4.3. Section 4.3.1 reviews double-loop methods which nest a reliability analysis inside of an outer optimization loop [128, 8, 41, 42, 98, 67]. Section 4.3.2 reviews sequential methods which separate the reliability analyses and the optimization so they can be performed sequentially [132, 111, 40, 4, 14, 140, 5]. As a result sequential methods are typically more efficient than double-loop methods but require careful management of the sequencing in order to ensure convergence. Section 4.3.3 reviews unilevel methods which decompose the reliability analysis into a set of conditions which can then be directly added as constraints to an optimization problem [77, 3, 117].

An overview of trust region methods is then given in section 4.4 as they form the foundation for convergence of the proposed multifidelity sequential approach. Finally, the proposed sequential RBDO algorithm is presented in section 4.5. It is motivated by the use of anchored decomposition and incorporates lower-fidelity models when available for enhanced convergence. It also inherits the guaranteed convergence properties of trust region methods. The AKIRA algorithm presented in chapter 3 is embedded within it to solve the inverse reliability analysis. In certain cases the approach is shown to be the same as other existing sequential RBDO methods. Section 4.6 demonstrates the approach on two structural RBDO test problems and compares results with those from literature.

Finally chapter 5 demonstrates the proposed multifidelity sequential RBDO method with AKIRA on the complex, coupled supersonic nozzle problem introduced above. Sections 5.1 and 5.2 introduce the nozzle system and section 5.3 describes the hierarchy of multifidelity computational models. Sections 5.4 and 5.5 describe the problem's parameterization and formal RBDO problem formulations. Finally section 5.6 summarizes the RBDO approach taken for the nozzle problem and section 5.7 presents the optimal nozzle designs with comparison to a standard deterministic design practice using safety factors and basis values to build margin into the design. The results demonstrate the benefits of pursuing the solution of a RBDO problem as well as the tractability of doing so given well-applied dimension reduction techniques and efficient reliability analysis and optimization methods. In addition, the results shed light on the design of a nozzle system and highlight the importance of using high-fidelity models in design.

Chapter 6 adds to an already lengthy dissertation by summarizing the primary contributions as well as outlining interesting directions for future development in the field of reliability-based design optimization.

1.1 Contributions

The contributions of this dissertation lie primarily in the field of reliability-based design optimization (RBDO) which concerns the intersection of the fields of reliability analysis and design optimization:

1. **A new algorithm for inverse reliability analysis for use in RBDO is proposed.** The Adaptive Kriging Inverse Reliability Analysis (AKIRA) algorithm described in chapter 3 is the first inverse reliability analysis using a kriging-based response surface approach to take multiple limit states into account. The algorithm demonstrates competitive performance with other reliability analysis algorithms while benefiting from the solution of the inverse problem during RBDO. A new heuristic is also proposed for the adaptive updating of kriging models used in reliability analyses which improves upon other existing pointwise heuristic criteria. Finally, the efficient calculation of design sensitivities is derived for use with AKIRA, including the proposal of an adaptive cokriging approximation method when evaluation of such sensitivities is expensive.

2. **The use of the directional simulation method as a reliability analysis technique is further enhanced for use in RBDO.** A method for approximating gradients of probability of failure via a Riemann sum using directional simulation is proposed in section 2.6.1. In addition the information obtained from the directional simulation method is used in new ways — not only to estimate probabilities of failure, but also to provide an approximation of the limit state surfaces and aid in the adaptive refinement of kriging models and efficient reliability analysis and gradient calculation during the AKIRA algorithm in sections 3.3.2 and 3.5.

3. **A new sequential RBDO algorithm is proposed.** The algorithm described in section 4.5 avoids an expensive double-loop procedure by managing anchored decomposition approximations within a trust region method. The additional expansion of the method to incorporate a set of bifidelity models further increases efficiency. The algorithm is demonstrated using the AKIRA algorithm for reliability analysis on a series of academic problems and a more complex nozzle design problem (chaper 5) and is shown to be competitive with existing sequential RBDO methods. The sequential RBDO algorithm is also shown in certain cases to be a generalization of existing sequential RBDO methods and to lead to optimal designs which meet reliability constraints for just an order of magnitude increase in cost compared to a similar deterministic design optimization.

4. **Reliability-based design optimization is applied to a large-scale complex multidisciplinary aerospace design problem, namely the design and optimization of a supersonic nozzle.** The coupled aerodynamic, thermal, and structural nozzle system is optimized using the proposed AKIRA and sequential RBDO algorithms, thereby demonstrating the capability of solving industrial-type problems with high-fidelity models in a reasonable amount of time. The results presented in chapter 5 motivate the benefits of RBDO over standard practice deterministic design and are informative for engineering judgements concerning reliable nozzle design.

1.2 Notation

A unifying notation is difficult to prescribe due to the convergence of multiple disciplines, each with its own standard notation, including but not limited to statistics, calculus, and optimization. The following descriptions are guidelines for the notation in this dissertation; exceptions are made clear from the context.

General Guidelines

In general scalars are denoted by lowercase Greek letters, *e.g.* γ. The primary exceptions are the probability of failure p_f and N, which is used to denote the number of samples at various times. Vectors are denoted by lowercase Roman letters, *e.g.* x, of which a component is denoted with a subscript, *e.g.* x_i. Matrices are denoted by uppercase Roman letters, *e.g.* A, of which a component is denoted A_{ij}. Scalar quantities of interest may also be denoted by upper case letters, *e.g.* F for thrust and T for temperature, however context will make the distinction clear.

For statistical quantities, $E[\cdot]$ is used for expectation, $\mathrm{Var}(\cdot)$ and $\mathrm{Cov}(\cdot,\cdot)$ for variance and covariance, and $P[\cdot]$ for probability. In contrast to standard statistical notation, uppercase Roman letters are not used to denote random variables. A probability distribution function for random variable u is denoted $f_u(u)$ and the cumulative distribution function is $F_u(u)$. A sample of a variable is denoted with a superscript in parentheses, *e.g.* $u^{(i)}$ is a sample of the vector u. Estimates are denoted with a hat notation, *e.g.* $\hat{p}_f$ is the estimate of the probability of failure p_f.

Reoccurring Variables and Functions

Several sets of variables and functions reoccur throughout this dissertation. They are listed here:

$x \in \mathbb{R}^n$: the vector of deterministic design variables

$y \in \mathbb{R}^p$: the vector of random variables/parameters

$u \in \mathbb{R}^p$: the vector of standard normal variables obtained via transformation T from y,

$$u = T(y)$$

$g(x, y) \in \mathbb{R}^m$: the limit state function(s) defining the region of failure

$g(x, u) \in \mathbb{R}^m$: the limit state function(s) as a function of u instead of y

$\hat{g}(x, y)$: an approximation of the limit state $g(x, y)$

$z(x) \in \mathbb{R}^m$: a level set of the limit state for a given design x, defined by

$$z(x) = g(x, y)$$

$\bar{z} \in \mathbb{R}^m$: the vector of failure thresholds for the limit states, *i.e.* failure for the ith limit state occurs when

$$g_i(x, y) \leq \bar{z}_i$$

$m(y) \in \mathbb{R}$: the mean of a kriging model evaluated at point y

$s(y) \in \mathbb{R}$: the standard deviation of a kriging model evaluated at point y

$\Phi(u)$: the standard normal cumulative distribution function (CDF)

$\phi(u)$: the standard normal probability distribution function (pdf)

$\mathbb{1}_{\{\textbf{Boolean expression}\}}(u)$: the indicator function defined as

$$\mathbb{1}_{\{\text{Boolean expression}\}}(u) = \begin{cases} 0 & \text{if Boolean expression is false} \\ 1 & \text{if Boolean expression is true} \end{cases}$$

Chapter 2

Reliability

The reliability of any designed system and its estimation, prediction and enhancement is of utmost importance to the engineering profession. Engineers would simply not be relied upon if they were unable to produce reliable designs. Indeed, the interest in the reliability of manmade objects is not new (medieval guilds required lengthy training periods at least in part to ensure quality products [113]), particularly when the livelihood of human beings is at stake. As a result, a reliable design is often synonymous with a safe design. However, reliable designs usually encompass more than safety considerations; they are designs that consistently perform well.

The study of reliability is built upon the mathematics of probability and was initially motivated by the introduction of mass production [113]. Although probability theory was introduced by Pascal and Fermat in the 1600s, it was not until the advent of mass production in the late 1800s and early 1900s that probability theory was applied to problem of estimating the reliability of a design. Saleh and Marais argue that the catalyst leading to the development of reliability engineering was the vacuum tube which was the primary contributor to failures in electronic equipment in the early 1900s and crucial to wartime strategy in the ensuing decades [113]. Since the more formal advent of reliability engineering in the 1950s, more specific disciplines arose such as the community of structural reliability [38]. Finally, the design and engineering of complex multidisciplinary systems in the 1970s, most notably nuclear power plants and aerospace vehicles, fueled a growing interest in system reliability [113]. Nevertheless the accurate estimation of system reliabilities when failure events are rare has proved difficult for complex systems which often require expensive computer simulations and feature a large number of design variables and uncertainties[1].

[1]Indeed, the research presented in this thesis was primarily funded through the DARPA EQUiPS (Enabling Quantification of Uncertainty in Physical Systems) program which completed in 2018. The goal of the program was to address this very difficulty, to "provide a rigorous mathematical framework and advanced tools for propagating and managing uncertainty in the modeling and design of complex physical and engineering systems," (DARPA EQUiPS mission statement, https://www.darpa.mil/program/equips, September 2018) with a particular emphasis on the design of complex systems such as those found in aerospace applications. The existence and goals of the program highlight the importance of solving reliability analysis and design problems for realistic high-dimensional applications.

This chapter is about *reliability analysis* or the mathematics behind predicting the reliability of an engineered system. First the mathematics of reliability analysis are presented in section 2.1 including the commonly used standard normal space and the differences between the forward and inverse reliability analysis problems. Section 2.2 introduces random sampling methods for reliability analysis, including the technique of directional simulation which is extensively used later in the AKIRA algorithm presented in chapter 3. Section 2.3, section 2.4, and section 2.5 introduce most probable point (MPP) methods, response surface methods, and hybrid methods for reliability analysis. A deeper look at kriging-based response surface methods for reliability analysis is provided in chapter 3 since the AKIRA algorithm uses kriging response surfaces. Finally section 2.6 presents and derives design sensitivities for probabilities of failure obtained from reliability analyses in preparation for use with reliability-based design optimization.

2.1 Introduction

The first step in reliability analysis is to identify the system and its subsystems. The second step is to identify the metrics and their limits from which system and subsystem failure will be gauged. Finally, models or experiments can be constructed which can be used together with a reliability analysis method to estimate system or subsystem reliabilities according to the chosen metrics.

Most often, the reliability of the overall system is the primary quantity of interest, which is naturally affected by the reliability of the system's subsystems. For example, for the supersonic nozzle system introduced in chapter 1, engineers may consider a variety of metrics such as maximum temperature and stress in the nozzle's wall, fatigue life, and nozzle thrust among others during the nozzle's analysis and design. Some quantities are clearly related to the structural failure of the nozzle, others, such as thrust, are related to performance specifications. For each metric a limit can be determined, beyond which the nozzle system is considered failed. For example, the nozzle's material will have a maximum service temperature and yield strength, and the performance specification will require a minimum thrust at a given operating condition. Accurate and efficient reliability analysis methods enable the reliability of such a complex system to be quantified, and ultimately used in design optimization.

We represent the metrics of a system by $g(x, y) \in \mathbb{R}^m$ for m metrics, where $x \in \mathbb{R}^n$ denotes the system's n deterministic variables and $y \in \mathbb{R}^p$ denotes the system's p random variables. Each $g_i(x, y)$ for $i = 1 \dots m$ is called a *limit state*. Note that since y is random, each limit state $g_i(x, y)$ is random as well. The limit at which failure occurs is denoted as $\bar{z} \in \mathbb{R}^m$ and thus a limit state $g_i(x, y)$ separates a system's performance space into a failure region F_i and a safe region S_i at the boundary defined by $\bar{z}_i$. The failure and safe regions are defined as

$$F_i \equiv \{(x, y) \in \mathbb{R}^n \times \mathbb{R}^p \mid g_i(x, y) \leq \bar{z}_i\},$$

$$S_i \equiv \{(x, y) \in \mathbb{R}^n \times \mathbb{R}^p \mid g_i(x, y) > \bar{z}_i\}.$$

Traditionally, the limit states $g(x, y) \in \mathbb{R}$ are defined such that the limit state level at which failure occurs is $\bar{z} = 0$.

For simplicity, consider a system with a single limit state $g(x, y)$, *i.e.* a system that has a single failure mode. Furthermore let failure occur for a limit state level of $\bar{z}$. To calculate the reliability of the system we can first estimate its probability of failure p_f. The reliability of the system is then defined as $1-p_f$. The probability of failure p_f can be written in many equivalent forms each of which provides a different perspective relevant to its calculation as well as an introduction to notation:

$$\begin{aligned} p_f(x) &\equiv F_g(\bar{z}) && \text{(2.1a)} \\ &= P\left[g(x, y) \leq \bar{z}\right] && \text{(2.1b)} \\ &= \int_{g(x,y)\leq\bar{z}} f_y(y)\mathrm{d}y && \text{(2.1c)} \\ &= \int \mathbb{1}_{g(x,y)\leq\bar{z}}\mathrm{dP}_y, && \text{(2.1d)} \end{aligned}$$

where $p_f(x)$ remains a function of the deterministic variables x, in other words, the probability of failure is different for different systems. $F_g(\bar{z})$ is the cumulative distribution function of $g(x, y)$ whose definition is given in the following statement. $P[\cdot]$ represents probability. (2.1c) shows the probability of failure definition as a volume integral over the failure set F, and (2.1d) makes use of the indicator function $\mathbb{1}_{g(x,y)\leq\bar{z}}$ which by definition is 1 when $g(x, y) \leq \bar{z}$ and 0 otherwise.

Despite the convenient definition of the probability of failure p_f, unfortunately the analytic solution of the typically multidimensional integral in (2.1) is in general very difficult, if not impossible to obtain. The problem is further exacerbated since the shape of the failure region is also typically unknown *a priori*. Section 2.1.2 summarizes a variety of methods for estimating p_f.

2.1.1 Multiple Limit States

When multiple limit states are present in a system, the specification of the failure region can become more complicated. There are two common cases: in a *series system* the failure of one subsystem leads to a system failure, whereas in a *parallel system* system failure only occurs once all subsystems have failed. In both cases a simple expression can be given for the system's probability of failure. In a *combined system* where subsystems interact in both series and parallel there is no such equally neat mathematical description, although such a description can be given for any system using Boolean operations on indicator functions [72].

In a series system (*i.e.* failure of one subsystem results in system failure), the probability of

failure is

$$\begin{aligned} p_f(x) &= P\left[\bigcup_{i=1}^{n} \{(x,y) \in \mathbb{R}^n \times \mathbb{R}^p : g_i(x,y) \leq \bar{z}_i\}\right] \\ &= P\left[\min_i \{g_i(x,y) - \bar{z}_i\} \leq 0\right]. \end{aligned} \tag{2.2}$$

In a parallel system (*i.e.* system failure occurs only when all components fail), the probability of failure is

$$\begin{aligned} p_f(x) &= P\left[\bigcap_{i=1}^{n} \{(x,y) \in \mathbb{R}^n \times \mathbb{R}^p : g_i(x,y) \leq \bar{z}_i\}\right] \\ &= P\left[\max_i \{g_i(x,y) - \bar{z}_i\} \leq 0\right]. \end{aligned} \tag{2.3}$$

2.1.2 Survey of Reliability Analysis Methods

In general, reliability analysis methods typically fall into one of three classes: sampling methods, most probable point (MPP) methods, and response surface methods.

In sampling methods, described in more detail in section 2.2, the quantities of interest or limit states are directly sampled to obtain an estimate of reliability. The simplest approach is simple Monte Carlo which can be augmented by importance sampling [97, 99] to increase efficiency. Other sampling methods such as directional simulation [20] and its adaptation for importance sampling [37] as well as subset simulation [13] are alternative sampling methods that have been applied to reliability analysis. However, while the approaches above increase the efficiency of obtaining an accurate reliability estimate they are often still too expensive to be used for systems modeled using expensive computer codes.

In most probable point methods, described in more detail in section 2.3, the failure point in the random space which has the highest likelihood of failure is searched for and then used to obtain an estimate of the reliability. The classical first-order reliability analysis (FORM) and second-order reliability analysis (SORM) methods are examples of MPP methods [38] where the limit state separating the safe and failure regions is approximated as a hyperplane or a quadratic surface, respectively. MPP methods can benefit from analytic sensitivities, provided that sensitivities for quantities of interest can be efficiently obtained [4, 8], however they are inaccurate, especially for higher dimensions, when the limit state is not linear or quadratic [115].

Response surface methods have the advantage of providing a cheap estimate of reliability via sampling or other methods from a response surface model, provided that the model is accurate in regions that contribute to the reliability estimate. Section 2.4 surveys a few common methods. Polynomial regression [23, 63], artificial neural networks [23, 63], support vector machines [87], and kriging [18, 42, 71], among other response surfaces have been investigated in the literature. Adaptive construction of response surfaces has also been considered and been shown to drastically reduce the

number of exact quantity of interest simulations compared to sampling methods, particularly for use with a kriging response surface where an estimate of model inaccuracy is available [71]. In addition, hybrid methods have also been proposed to mitigate inaccuracies due to incorrect response surfaces (see section 2.5); for example Li et al's importance sampling-based approach relies only on exact data in important regions of the random space and on the response surface elsewhere [81].

2.1.3 The Standard Normal Space

In many reliability analysis methods, the integration for probability of failure in (2.1) is performed in the *standard normal space*, which is the space where the problem's random variables have been transformed to zero mean, unit variance, uncorrelated random normal variables. This space is often denoted as the U-space corresponding to the standard normal variables u, in contrast to the original space of the physically meaningful random variables y.

The primary benefit of performing the integration in the standard normal space is a very convenient interpretation: in the standard normal space, the closest point on the limit state ($g(u) = 0$) to the origin is the point of maximum likelihood of failure [115, 121], also referred to as the most probable point of failure (MPP). Consequently many algorithms for estimating the probability of failure p_f implement iterative techniques for finding the MPP(s) and approximating the limit state around these points [38].

A variety of transformations have been proposed for transforming correlated non-normal variables y into standard normal variables $u \sim \mathcal{N}(0, 1)$ such as the Rosenblatt [110] and Nataf [79] transformations. Ditlevsen and Madsen provide a good overview of different methods [38]. When y are uncorrelated, the Rosenblatt and Nataf transformations are the same and given by

$$u_i = \Phi^{-1}(F_i(y_i)), \tag{2.4}$$

where $\Phi(\cdot)$ is the standard normal cumulative distribution function and $F_i(\cdot)$ is the cumulative distribution function for $y_i \sim Y_i$.

Rosenblatt Transform

The Rosenblatt transform [110] can be used when the joint probability distribution function for the random variables is specified (a typically rare situation) or when conditional distributions are specified for the random variables. A set of correlated random variables $y \sim Y$ can be transformed to a set of uniform uncorrelated variables $z \sim \mathcal{U}(0, 1)$ in the following manner beginning with the

marginal distribution of the first variable [110]:

$$z_1 = P[Y_1 \leq y_1], \tag{2.5a}$$
$$z_2 = P[Y_2 \leq y_2 \mid Y_1 = y_1], \tag{2.5b}$$
$$\vdots \tag{2.5c}$$
$$z_p = P[Y_p \leq y_p \mid Y_1 = y_1, \ldots, Y_{p-1} = y_{p-1}]. \tag{2.5d}$$

A transformation from the set of uniform uncorrelated variables z to the standard normal variables u is easily made with standard normal quantile function $u = \Phi^{-1}(z)$. Note that in most situations the transformation proposed in (2.5) must be performed numerically.

Nataf Transform

The Nataf transform [38, 79] can be used when the marginal distributions for the correlated random variables y are specified together with a linear correlation matrix R specifying dependencies between variables defined as

$$R_{ij} = E\left[\left(\frac{y_i - \mu_i}{\sigma_i}\right)\left(\frac{y_j - \mu_j}{\sigma_j}\right)\right], \tag{2.6}$$

where μ_i is the mean of y_i and σ_i is the standard deviation of y_i.

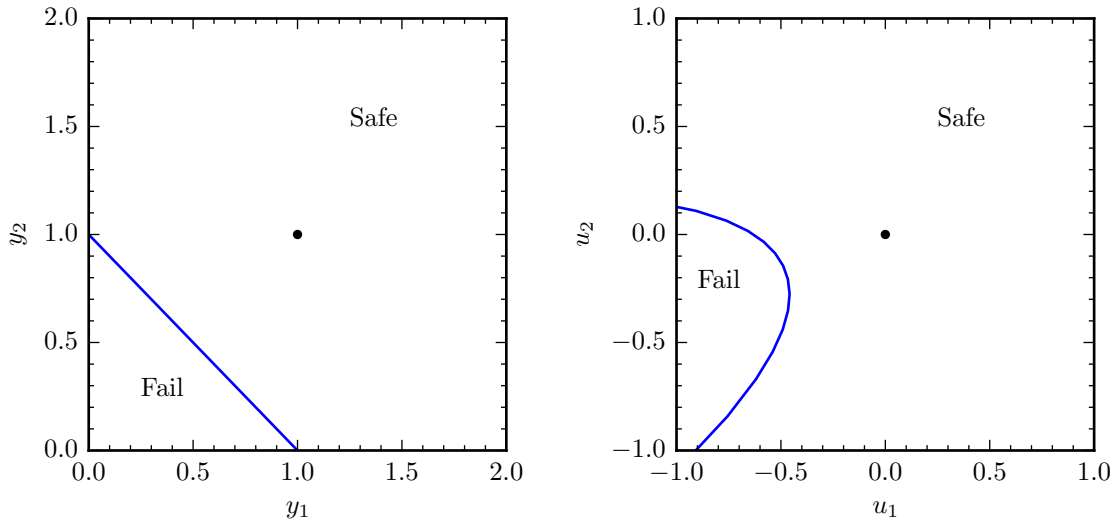
First the marginal transformation from (2.4) is applied to $y \sim Y$, which yields a set of (still correlated) variables $z \sim Z$. Z is assumed to be a standard normal vector with correlation matrix R_0. A set of standard normal variables is then typically obtained via

$$u = L^\top z, \tag{2.7}$$

where L is the lower triangular Cholesky factor or similar square root for R_0^{-1} [79]. The normal correlation matrix R_0 must be calculated so as to be consistent with the prescribed correlation matrix R. The following integral equation is solved to determine the elements $R_{0,ij}$ of R_0:

$$\begin{aligned} R_{ij} &= E\left[\left(\frac{y_i - \mu_i}{\sigma_i}\right)\left(\frac{y_j - \mu_j}{\sigma_j}\right)\right] \\ &= E\left[\left(\frac{F_i^{-1}(\Phi(z_i)) - \mu_i}{\sigma_i}\right)\left(\frac{F_j^{-1}(\Phi(z_j)) - \mu_j}{\sigma_j}\right)\right] \\ &= \frac{1}{\sigma_i \sigma_j}\int\int \left(F_i^{-1}(\Phi(z_i)) - \mu_i\right)\left(F_j^{-1}(\Phi(z_j)) - \mu_j\right)\varphi_2(z_i, z_j, R_{0,ij})\mathrm{d}z_i\mathrm{d}z_j, \end{aligned} \tag{2.8}$$

where $\varphi_2(z_i, z_j, R_{0,ij})$ is the bivariate standard normal probability density function with correlation $R_{0,ij}$. The solution of (2.8) is typically obtained numerically (when possible) although approximations exist [38].



(a) Linear limit state $g(y)$

(b) Transformed limit state $g(u)$

Figure 2.1: Nonlinearities introduced by the Nataf transformation from the physical y space (left) to the standard normal space u (right) for two correlated variables $y_1, y_2 \sim \Gamma(1)$ (equivalent to an exponential distribution) with linear correlation $R_{12} = 0.05$. Note how the initially linear limit state becomes nonlinear. The black dot marks the mean of the random variables in both the physical space and standard normal space.

Despite the popularity of the method, the Nataf transform may be impossible, for example in the situations where (2.8) does not have a solution due to large R_{ij} or yields a non-positive definite correlation matrix R_0 [38, 79]. Lebrun and Dutfoy recast the transform in terms of cupola theory and show the Nataf transformation is equivalent to the choice of a normal cupola for the joint distribution of y [79]. They highlight some of the limitations of the Nataf transformation including failure to accurately model tail dependencies when the normal cupola is used as well as lack of a solution of (2.8) under certain combinations of marginal distributions with the specification of a linear correlation matrix [79].

2.1.4 The Reliability Index β

The geometric or Hasofer-Lind reliability index β is often used in reliability analysis and RBDO problems instead of the probability of failure p_f. It derives from the structural reliability analysis literature [38] and measures the minimum distance from the mean to the limit state surface $g(x, y) = \bar{z}$ where failure first occurs in the standard normal space:

$$\beta = \min_u \{\sqrt{u^\top u} \mid g(x, u) = \bar{z}\}. \tag{2.9}$$

When a first-order approximation to the limit state is made (see section 2.3.1) the approximate failure probability $\hat{p}_f$ is related to β by

$$\hat{p}_f = \Phi(-\beta), \tag{2.10}$$

where $\Phi(\cdot)$ is the standard normal CDF. (2.10) is exact for a linear limit state, however it may incur large errors in $\hat{p}_f$ for nonlinear limit states, especially in higher dimensions [115]. A typical target value of β for design problems is 3 which corresponds to a failure probability $p_f = 1.35 \times 10^{-3}$ for a limit state $g(x, u)$ linear in u.

The generalized reliability index β_g was proposed to overcome the shortcomings of the geometric reliability index and is defined as

$$\beta_g \equiv \Phi^{-1}(p_f), \tag{2.11}$$

where $\Phi^{-1}(\cdot)$ is the inverse standard normal CDF and p_f is the failure probability [38]. The use of the generalized reliability index β_g *does not* imply a normal assumption on the input random variables y for the limit state $g(x, y)$; rather the inverse standard normal CDF is used a mathematical tool to extend the concept of the geometric reliability index to more general problems.

2.1.5 Forward vs. Inverse Problems

The forward reliability analysis problem of estimating the probability of failure p_f given limit states $g(x, y)$ and corresponding failure levels $\bar{z}$ is given by

$$p_f = F(\bar{z}), \tag{2.12}$$

where F is the cumulative distribution function for random variables y. The direct solution of the reliability analysis problem via (2.12) is known as the *reliability index approach*. The solution of (2.12) is straightforward with sampling methods and other well established techniques such as FORM. The forward analysis leads to *chance constraints* in reliability-based design optimization written as

$$p_f(x) < \bar{p}_f, \tag{2.13}$$

where $\bar{p}_f$ is an upper limit on the probability of failure $p_f(x)$.

However, in certain cases solving the inverse reliability analysis problem which is called the *performance measure approach* is more advantageous. In this method, a response level $z = g(x, y)$ is calculated for a prescribed probability of failure $\bar{p}_f$ instead as

$$z = F^{-1}(\bar{p}_f; y), \tag{2.14}$$

where F^{-1} is the quantile function corresponding to limit state $g(x, y)$ for prescribed probability of

failure $\bar{p}_f$. The notation makes clear that the quantile function is also dependent on the choice of random variables y. The inverse reliability analysis leads to *quantile constraints* in reliability-based design optimization:

$$z(x) > \bar{z}, \tag{2.15}$$

where $\bar{z}$ is a lower limit on the limit state level $z(x)$.

The use of the inverse problem is preferable over the forward problem in reliability-based design optimization for several reasons:

Robustness. A solution of the forward reliability analysis from (2.1) for p_f can easily yield $p_f = 0$ or $p_f = 1$, *i.e.* the design either is either always safe or always fails [128]. Such a scenario introduces numerical noise into the optimization, thereby interfering with gradient calculations and algorithm convergence etc. The problem is further exacerbated when approximate estimates $\hat{p}_f$ of the probability of failure are used as such estimates may not be sufficiently resolved for extremely small values of p_f and will therefore encounter the same situation. On the other hand, a solution to the inverse problem always exists, both analytically and numerically, for a prescribed probability of failure p_f.

Efficiency. A forward reliability analysis may require a large number of function evaluations to obtain a converged estimate $\hat{p}_f$, particularly when $p_f \ll \bar{p}_f$, the required reliability target. In contrast, an inverse reliability analysis explicitly returns limit state levels corresponding to the desired probability of failure; as a result it may be more efficient when the failure probability is low [41].

Ease of solution. The inverse reliability analysis is easier to solve for certain types of reliability analysis methods such as the most probable point methods [3, 128]. In such situations, the use of the inverse reliability analysis may increase the efficiency and robustness of the reliability analysis. Note however that the solution of the inverse problem may be substantially more complicated in other situations, notably the multiple limit state case (see chapter 3).

Suitability for RBDO. Finally, the solution of the inverse problem may be better suited for certain reliability-based design optimization (RBDO) algorithms. In chapter 4, a new sequential method for RBDO is introduced which uses anchored decomposition. The use of an inverse reliability analysis in this sequential method leads to a convenient decoupling of the reliability and optimization problems which in turn enhances the efficiency of the RBDO method.

Clearly the inverse reliability analysis has several strengths compared with the forward reliability analysis. The primary difficulty of solving the inverse reliability analysis arises when multiple limit states are present, in which case (2.14) may have no unique solution if a system-level probability of failure $\bar{p}_f$ is prescribed. For example, a high stress and a low temperature in an aerospace design problem such as the supersonic nozzle problem introduced earlier may result in the same probability

of failure as a low stress and a high temperature. Clearly, there may be two (or more) solutions to the inverse problem.

As a result, the inverse reliability analysis has primarily been applied to problems with a single limit state or those where multiple limit states can be considered as a single limit state. The use of the inverse reliability analysis for MPP methods is well-established [3, 128]. Bichon et al make an extension of the kriging-based response surface method EGRA to the inverse reliability analysis for a single limit state [17]. Li considers the inverse reliability analysis for both single and multiple limit states in the context of polynomial response surfaces and FORM [80] and identifies the problem with non-unique solutions for the inverse problem. To avoid the non-uniqueness issue in the inverse problem, Li proposes artificially constructing dependencies between the limit states. For example β_i, the reliability index for limit state $g_i(x, y)$ ($i = 2 \dots m$ for m limit states) is considered to be a fixed ratio k_i of β_1, *i.e.* $\beta_i = k_i \beta_1$. In this way, the multiple limit state problem is reduced to a single limit state problem in terms of β_1. Unfortunately when used in RBDO, this method will not lead to a correct solution if k_i are picked incorrectly.

2.2 Random Sampling Methods

The most basic method for the calculation of the probability of failure integral in (2.1) is simple Monte Carlo, however large numbers of limit state function evaluations are required to estimate the small probabilities of failure required in reliability analyses. Variance reduction methods such as Latin Hypercube Sampling (LHS) and especially importance sampling etc. can be very helpful but may still exceed the available computational budget when limit state functions are expensive to evaluate as in the case of coupled multiphysics simulations. Here four sampling methods are reviewed that have found application in reliability analysis: simple Monte Carlo, importance sampling, directional simulation, and subset simulation.

2.2.1 Simple Monte Carlo

The simplest and most robust method is simple Monte Carlo which is attractive due to its independence of convergence rate from dimension and its applicability for non-smooth functions[2]. Its primary limitation lies in its slow convergence rate which necessitates large numbers of samples to accurately estimate small probabilities of failure. By drawing N independent random samples for y, an unbiased estimate of p_f can be obtained as

$$\hat{p}_f(x) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}_{g(x,y) \leq \bar{z}}(y^{(i)}), \tag{2.16}$$

[2]An excellent introduction to the Monte Carlo method and its many varieties is provided by Owen [96].

where $\mathbb{1}_{g(x,y)\leq\bar{z}}$ is the indicator function for failure in the domain, *i.e.* taking the value 1 in the failure region and 0 otherwise. The variance of the estimator is obtained as

$$\begin{aligned}
\sigma_f^2(x) &= \mathrm{Var}\left(\frac{1}{N}\sum_{i=1}^{N}\mathbb{1}_{g(x,y)\leq\bar{z}}(y^{(i)})\right) \\
&= \frac{1}{N}\mathrm{Var}\left(\mathbb{1}_{g(x,y)\leq\bar{z}}(y)\right) \\
&= \frac{1}{N}\left(E[\mathbb{1}^2_{g(x,y)\leq z}(y)] - E[\mathbb{1}_{g(x,y)\leq\bar{z}}(y)]^2\right) \\
&= \frac{1}{N}\left(E[\mathbb{1}_{g(x,y)\leq z}(y)] - E[\mathbb{1}_{g(x,y)\leq\bar{z}}(y)]^2\right) \\
&= \frac{1}{N}(p_f - p_f^2) \\
&= \frac{p_f}{N}(1-p_f),
\end{aligned} \tag{2.17}$$

where we make use of the fact that $E[\mathbb{1}^2_{g(x,y)\leq\bar{z}}(y)] = E[\mathbb{1}_{g(x,y)\leq\bar{z}}(y)]$. A plug-in estimate $\hat{p}_f$ from (2.16) is commonly used to substitute for p_f in (2.17) to estimate σ_f^2. The coefficient of variation of the estimator can be estimated as

$$\hat{CoV}(x) = \sqrt{\frac{1-\hat{p}_f}{N\hat{p}_f}}, \tag{2.18}$$

which shows that the coefficient of variation of the estimator (and likewise root mean square error) converges at the relatively slow rate of $\sqrt{N}$. For example, if a probability of 10^{-6} is to be estimated with a target coefficient of variation of 0.1, 10^8 samples will be required, a number much too large for expensive multiphysics simulations. On the other hand, the coefficient of variation in (2.18) is independent of the complexity of $g(x,y)$, which could be discontinuous or highly nonlinear, and is also independent of the dimension of y, which could be very large.

The central limit theorem can be used to establish a confidence interval for the probability of failure estimate as

$$\hat{p}_f \pm z_{\alpha/2}\sqrt{\frac{\hat{p}_f(1-\hat{p}_f)}{N}}, \tag{2.19}$$

where $z_{\alpha/2}$ is the standard normal variate for a confidence interval $1-\alpha$. For a 95% confidence interval, $z_{\alpha/2} \approx 1.96$. In the case that $\hat{p}_f$ is zero (*i.e.* the failure region was not sampled), the confidence interval can be estimated as $[0, {}^3/_N]$ with 95% confidence [96], however in such situations using importance sampling or other methods is preferable.

2.2.2 Importance Sampling

Importance sampling can be used to adjust a random sampling method so that more samples fall in the important region of interest. To adapt importance sampling for the probability of failure

estimate, (2.1) can be rewritten as

$$\begin{aligned} p_f(x) &= P[g(x,y) \leq \bar{z}] \\ &= \int_{g(x,y)\leq \bar{z}} f_y(y)\mathrm{d}y \\ &= \int_{g(x,y)\leq \bar{z}} \frac{f_y(y)}{h_y(y)} h_y(y)\mathrm{d}y, \end{aligned} \tag{2.20}$$

where $h_y(y)$ is the biasing distribution and $f_y(y)/h_y(y)$ is the likelihood ratio. By sampling from the biasing distribution and calculating the likelihood ratio, a Monte Carlo estimator based on (2.20) can be constructed as

$$\hat{p}_f(x) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{g(x,y)\leq \bar{z}}(y^{(i)}) \frac{f_y(y^{(i)})}{h_y(y^{(i)})}. \tag{2.21}$$

The estimator is unbiased and its variance is given by

$$\begin{aligned} \sigma_f^2(x) &= \mathrm{Var}\left(\frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{g(x,y)\leq \bar{z}}(y^{(i)}) \frac{f_y(y^{(i)})}{h_y(y^{(i)})}\right) \\ &= \frac{1}{N}\mathrm{Var}\left(\mathbb{1}_{g(x,y)\leq \bar{z}}(y)\frac{f_y(y)}{h_y(y)}\right) \\ &= \frac{1}{N}\left(E_h\left[\left(\mathbb{1}_{g(x,y)\leq \bar{z}}(y)\frac{f_y(y)}{h_y(y)}\right)^2\right] - p_f^2\right) \\ &= \frac{1}{N}\left(E_h\left[\mathbb{1}_{g(x,y)\leq \bar{z}}(y)\frac{f_y^2(y)}{h_y^2(y)}\right] - p_f^2\right) \\ &= \frac{1}{N}\left(\int_{g(x,y)\leq \bar{z}} \frac{f_y^2(y)}{h_y(y)}\mathrm{d}y + p_f^2\int_{g(x,y)\leq \bar{z}} h_y(y)\mathrm{d}y - 2p_f\int_{g(x,y)\leq \bar{z}} f_y(y)\mathrm{d}y\right) \\ &= \frac{1}{N}\left(\int_{g(x,y)\leq \bar{z}} \frac{f_y^2(y)}{h_y(y)} + p_f^2 h_y(y) - 2p_f f_y(y)\mathrm{d}y\right) \\ &= \frac{1}{N}\left(\int_{g(x,y)\leq \bar{z}} \frac{(f_y(y) - p_f h_y(y))^2}{h_y(y)}\mathrm{d}y\right), \end{aligned} \tag{2.22}$$

where $E_h[\cdot]$ denotes expectation with respect to density $h_y(y)$. An estimator for the variance can be obtained using the original samples sampled from the biasing distribution $h_y(y)$ as

$$\hat{\sigma}_f^2 = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{g(x,y)\leq \bar{z}}(y^{(i)}) \left(\frac{f_y(y^{(i)})}{h_y(y^{(i)})} - \hat{p}_f\right)^2. \tag{2.23}$$

Thus a confidence interval can be constructed as

$$\hat{p}_f \pm z_{\alpha/2}\frac{\hat{\sigma}_f}{\sqrt{N}}, \tag{2.24}$$

where $z_{\alpha/2}$ is the standard normal variate for a confidence interval $1 - \alpha$.

The difficulty of importance sampling lies in first determining the important region(s), and next determining a good biasing distribution for these region(s) [96]. There are several types of importance sampling which have been employed for reliability estimation including cross-entropy [81] and choosing a normal distribution for $h_y(y)$ centered at the design point(s) [115]. Ang et al consider a kernel-based adaptive method for importance sampling [10]. A survey of several techniques applied to reliability analysis can be found in [97] and [99]. Although importance sampling can drastically decrease the number of samples required for an accurate estimate of p_f compared to Monte Carlo, use of importance sampling alone still typically exceeds the computational budget for reliability analyses with complex analyses. It is often used as a component of other reliability analysis methods, in particular response surface methods [19, 26, 37]. Note that a transformation to the standard normal space is not necessary for the implementation of importance sampling.

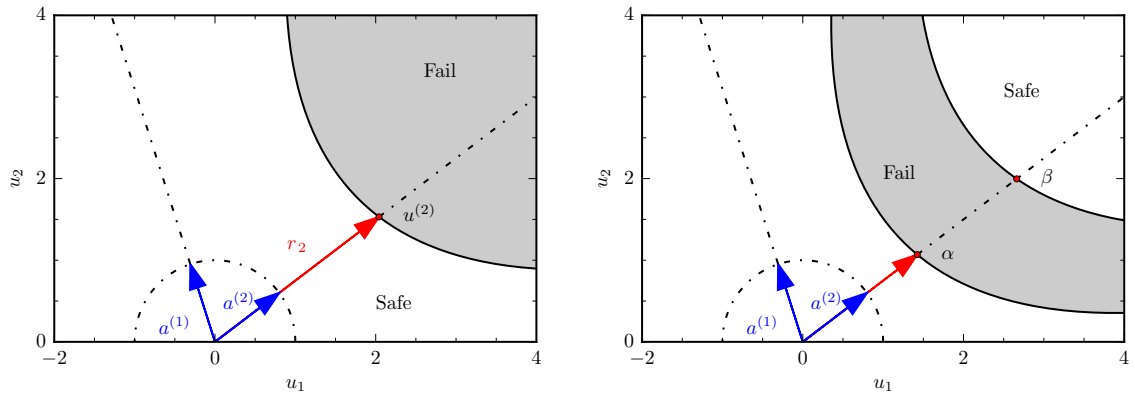
2.2.3 Directional Simulation

The method of directional simulation was originally introduced by Deák for the integration of the multivariate normal distribution [35] and subsequently investigated by Ditlevsen, Bjerager, and others for application to reliability analysis [38, 20, 88, 39, 91]. The key idea is to sample random directions and perform a univariate integration along each direction to obtain the conditional probability of failure given that direction. An estimator of the total probability of failure $\hat{p}_f$ can then be constructed by combining all simulated directions. Directional simulation is usually performed in the standard normal space (although it is not limited as such [88]) since doing so affords a closed-form analytic formula for the conditional probabilities of failure. The univariate integration is typically approximated by only finding the most probable point of failure for each direction, which is typically determined via a linesearch. The derivation of directional simulation below primarily follows Bjerager [20] and is accompanied by Figure 2.2.

Let each $u \in \mathbb{R}^p$ be a realization of the standard normal random vector U in the standard normal space expressed as $u = Ra$ where $a \in \mathbb{R}^p$ is a realization of the random unit vector A constrained to the surface of the unit sphere $S_p(0,1) \in \mathbb{R}^p$ and $R^2 \sim \chi^2_p$; both are independent. The probability of failure p_f from (2.1) can then be written as

$$\begin{aligned} p_f(x) &= P[g(x,u) \leq \bar{z}] \\ &= P[g(x,u) \leq \bar{z} \mid A]P[A] \\ &= \int_{S_p(0,1)} P[g(x,RA) \leq \bar{z} \mid A = a] f_a(a) da \\ &= \int_{S_p(0,1)} P[g(x,Ra) \leq \bar{z}] f_a(a) da. \end{aligned} \tag{2.25}$$

The conditional probability $P[g(x,Ra) \leq \bar{z}]$ in (2.25) can be approximated assuming that $g(x,Ra) \leq$



(a) Limit state $g(x,u) = \bar{z}$ (solid line) is shown in the standard normal space separating the safe and failure regions. Uniform random vectors $a^{(1)}$ and $a^{(2)}$ are simulated on the unit sphere and define random directions in the standard normal space. These directions are then searched along until the limit state is discovered. Here failure first occurs along direction $a^{(2)}$ at a distance r_2 from the origin, but does not occur along direction $a^{(1)}$.

(b) The linesearch along a simulated direction may not always have a unique root as shown in this example where the limit state defines a band of failure separating two safe regions. For example, direction $a^{(2)}$ intersects the limit state $g(x,u) = \bar{z}$ twice and has two roots labeled α and β. In such cases, it is important to at least find the root closest to the origin (α) as this has a larger impact on the probability of failure p_f.

Figure 2.2: Schematic of directional simulation and its limitations.

$\bar{z}$ on an interval $R \in [r, \infty)$. By determining the location of the limit state surface at distance r from the origin along direction a, the conditional probability can thus be written as $1 - \chi_p^2(r^2)$ where $\chi_p^2(\cdot)$ is the chi-squared cumulative distribution function with p degrees of freedom. Thus an unbiased estimator for p_f can be constructed by sampling i.i.d. random directions a:

$$\begin{aligned} \hat{p}_f(x) &= \frac{1}{N} \sum_{i=1}^{N} P[g(x, Ra^{(i)}) \leq \bar{z}] \\ &= \frac{1}{N} \sum_{i=1}^{N} \left(1 - \chi_p^2(r_i^2)\right). \end{aligned} \tag{2.26}$$

An estimate for the variance of the estimator is

$$\hat{\sigma}_f^2(x) = \frac{1}{N(N-1)} \sum_{i=1}^{N} \left(1 - \chi_p^2(r_i^2) - \hat{p}_f\right)^2. \tag{2.27}$$

Finally, we note as usual for large N a confidence interval for $\hat{p}_f$ can be established as

$$\hat{p}_f \pm z_{\alpha/2} \hat{\sigma}_f, \tag{2.28}$$

Table 2.1: Comparison of directional simulation when only the root closest to the origin is found (DS) and simple Monte Carlo (MC) for several problems (Rastrigin with $u \in \mathbb{R}^2$, cantilever beam with $u \in \mathbb{R}^4$, and short column with $u \in \mathbb{R}^4$) at various limit state levels z. Probabilities of failure are reported with their coefficient of variation in parentheses. Directional simulation provides excellent probability of failure estimates in certain cases (see the short column results), and reasonable but conservative results in other cases, in particular for very nonlinear limit states (see Rastrigin and cantilever beam results).

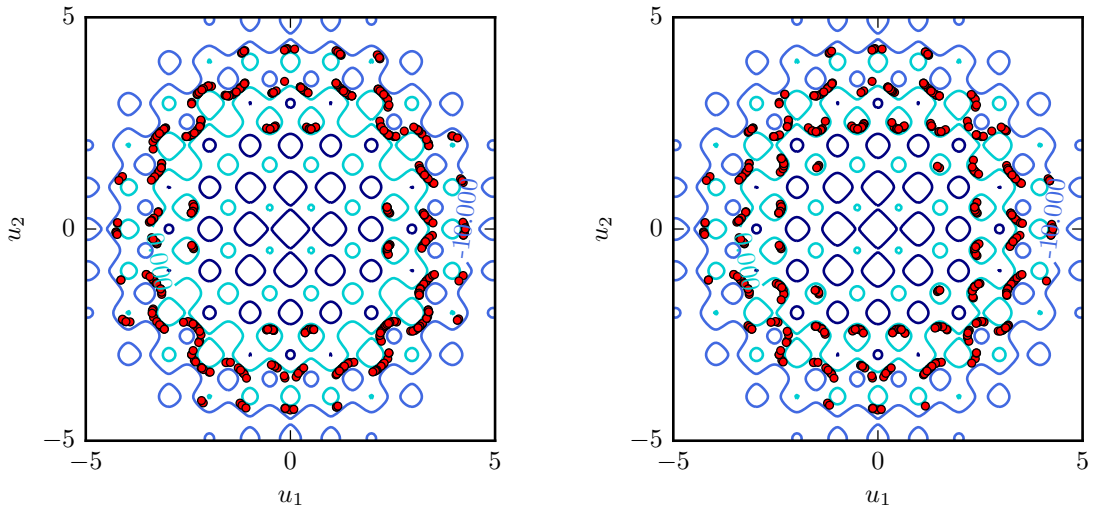
Problem	z	$\hat{p}_{f,MC}$	$\hat{p}_{f,DS}$
Rastrigin (2D)	-4.0453	10^{-2} (0.001)	1.900×10^{-2} (0.01)
	-12.8593	10^{-4} (0.01)	1.394×10^{-4} (0.01)
	-22.2623	10^{-6} (0.1)	1.386×10^{-6} (0.01)
Cantilever Beam	$[0.00, 0.05]^\top$	0.9765×10^{-2} (0.02)	2.0722×10^{-2} (0.01)
	$[-0.10, -0.01]^\top$	0.9400×10^{-4} (0.02)	1.6154×10^{-4} (0.01)
Short Column	0.2156	10^{-2} (0.001)	1.003×10^{-2} (0.01)
	-0.0802	10^{-4} (0.01)	0.991×10^{-4} (0.01)
	-0.3796	10^{-6} (0.1)	0.994×10^{-6} (0.01)

where $z_{\alpha/2}$ is the standard normal variate for a confidence interval $1 - \alpha$.

The primary difficulty with directional simulation lies in the determination of the minimum distance r from the origin to the limit state surface defined by level $\bar{z}$ along direction a which is typically determined by a line search. Efficient rootfinding methods such as Brent's method [103] can be used to determine a root once bracketed. For highly nonlinear limit states (for example the Rastrigin function shown in figure 2.3), the bracketing procedure requires care and may become expensive if it is to be robust. Univariate optimization methods may be susceptible to local minima, however efficient global optimization methods [75] or a simple pattern search, albeit more expensive, can be effective for bracketing and/or rootfinding in such cases.

As the regions of the limit state closest to the origin contribute the most to the probability of failure, neglecting more distant points on the limit state (if any) at worst leads to a more conservative estimate of p_f. For this reason, directional simulation works well for so-called star-shaped safe regions [38], as each point on the boundary of such regions is accessible by a ray from the origin and will contribute to the probability of failure estimate. For spherical limit states, only a single sample is required to obtain the exact failure probability.

Table 2.1 compares probabilities of failure obtained using simple Monte Carlo and directional simulation (using only the first root closest to the origin) for three problems and variety of probabilities of failure. Directional simulation performs very well for the short column problem but gives conservative estimates for the cantilever beam problem and highly nonlinear Rastrigin function. Figure 2.3 compares the limit state surface identified during a directional simulation procedure when roots are correctly and incorrectly identified for the highly nonlinear Rastrigin function in 2 dimensions.



(a) Insufficient rootfinding

(b) Sufficient rootfinding

Figure 2.3: The highly nonlinear Rastrigin limit state function in the bivariate standard normal space is shown. Failure occurs for $g(u) \leq -4.0453$ which yields a probability of failure of 1% (Monte Carlo simulation, $CoV = 0.001$). The first 1000 points used in the directional simulation integration are shown as red dots. The insufficient bracketing used in the directional simulation on the left misses the smaller failure regions closer to the origin and actually underpredicts the failure probability as 0.84% ($CoV = 0.01$). However when a robust bracketing method is used (shown on the right), the correct failure regions are identified and directional simulation overpredicts the failure probability as expected as 1.90% ($CoV = 0.01$). In this particular example a pattern search with 100 equally spaced points was found to be sufficient for robust integration of the nonlinear function. Efficiency could be increased with other univariate global optimization methods. When directional simulation using a linesearch that finds multiple roots is used (not shown), the estimated failure probability is 1.01% ($CoV = 0.01$), which matches very well with the Monte Carlo estimate as expected.

Although directional simulation can perform quite well for problems of small dimensions, it suffers a similar fate as Monte Carlo sampling in high-dimensional spaces where many of the sampled directions do not encounter the limit state. As a result, importance sampling techniques have also been applied to directional simulation in an effort to increase its convergence rate in higher dimensions. Bjerager considered the effects of a mixed sampling density and derives several sampling densities for specific cases [20]. Ditlevsen et al also considered directional importance sampling and derived densities for an approximating halfspace and a convex polyhedral set [37]. Moarefzadeh and Melchers proposed a directional importance sampling method in the physical y space as opposed to the standard normal space [88]. They considered using a multivariate normal joint probability distribution with the same covariance matrix as the original joint pdf for the sampling density. Papaioannou proposed an intuitively appealing adaptive directional importance sampling method that

centers sampling distributions at important directions from a deterministic point set [99]. Shayanfar et al considered a cross-entropy adaptive directional importance sampling method that centers samples around an important direction [119].

2.2.4 Subset Simulation

In the subset simulation method introduced by Au and Beck in 2001 [13], the probability of failure p_f is estimated by a product of larger conditional probabilities which are more easily obtained. Each conditional probability corresponds to an intermediate failure region and is estimated in sequence using a Markov chain Monte Carlo simulation technique. In this way the true failure region is reached after a certain number of steps.

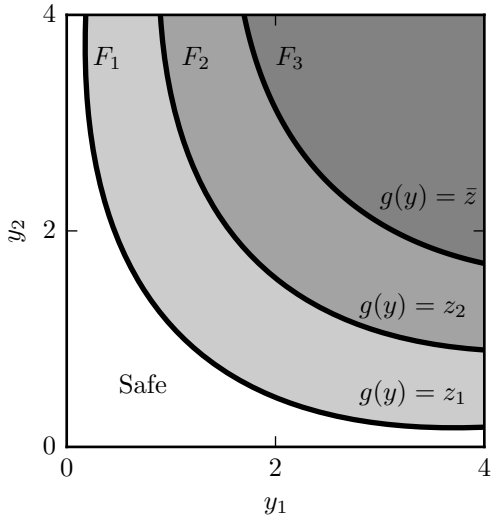


Figure 2.4: Three failure regions F_1 through F_3 to be used in subset simulation. By sampling for failure on region F_1 first and then estimating the conditional probability of failure on subsequent more distant failure regions (F_2 and then F_3), an estimate of the failure probability for failure region F_3 can be obtained corresponding to limit state level $\bar{z}$.

In particular, consider a sequence of failure regions $F_1 \subset F_2 \subset \cdots \subset F_s = F$ where $F \equiv \{(x, y) \in \mathbb{R}^n \times \mathbb{R}^p \mid g(x, y) \leq \bar{z}\}$ is the true failure region we are interested in estimating defined by limit state(s) $g(x, y)$. Let the event of belonging to a given region F_i be denoted by F_i as well, *i.e.* $P[(x, y) \in F_i] = P[F_i]$. Then the definition of conditional probability can be used to separate

the failure probability as

$$\begin{aligned} p_f(x) &= P[F] = P[F_s] \\ &= P\left[\bigcap_{i=1}^{s} F_i\right] \\ &= P\left[F_s \mid \bigcap_{i=1}^{s-1} F_i\right] P\left[\bigcap_{i=1}^{s-1} F_i\right] \\ &= P[F_s \mid F_{s-1}]\, P\left[\bigcap_{i=1}^{s-1} F_i\right] \\ &= \dots \\ &= P[F_1] \prod_{i=1}^{s-1} P[F_{i+1} \mid F_i]\,. \end{aligned} \tag{2.29}$$

Thus by estimating $P[F_1]$ and the conditional probabilities $P[F_{i+1} \mid F_i]$ for $i = 1, \dots, s-1$, an estimate of p_f can be constructed. By choosing appropriate failure regions F_i, the estimated probability of failure for each failure region can be large, around 0.1, and thus easily estimated however the constructed probability of failure $\hat{p}_f$ for the true failure region F can be quite small. $P[F_1]$ can be initially estimated using Monte Carlo sampling and the calculation of the remaining conditional probabilities can be efficiently performed using a Markov chain Monte Carlo method based on the Metropolis algorithm [13].

Au and Beck show that the total number of samples required to obtain an estimate $\hat{p}_f$ with a specified coefficient of variation is proportional to $|\log p_f|^r$ where $r \leq 3$ [13]. This compares very favorably to the required number of samples proportional to $1/p_f$ for Monte Carlo simulation.

2.3 MPP Methods

Traditional most probable point (MPP) methods such as the First Order Reliability Method (FORM) and Second Order Reliability Method (SORM) iteratively approximate the limit state using a first order or second order Taylor series expansion while searching for the point on the limit state with maximum likelihood of failure, *i.e.* the most probable point of failure (MPP) [38]. As a result, MPP methods rely on the standard normal space and therefore a transformation of variables $u = T(y)$ as discussed in section 2.1.3. The description of MPP methods is facilitated by the reliability index β introduced in section 2.1.4. Recall that the geometric reliability index β is the distance $\|u\|_2$ between the origin and the MPP in the standard normal space [38, 121]. For some applications multiple MPPs may occur.

Almost all MPP methods rely on approximation of the limit state in the vicinity of the MPP(s). Thus, despite their widespread use [4, 5, 8, 38, 47, 106, 121, 128], they can suffer from poor accuracy

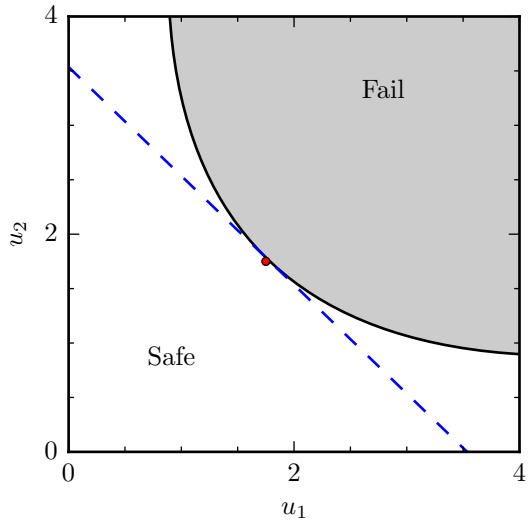


Figure 2.5: First order reliability method (FORM). The most probable point (MPP) of failure is identified by the red dot on the limit state $g(x, u) = \bar{z}$ (solid black line). The FORM method approximates the limit state by a hyperplane at the MPP, shown as the dashed blue line.

if multiple MPPs are present or if not all the MPPs are identified. In addition, if the limit state is not represented well by a linear or quadratic function at the MPP then the failure probability estimated from FORM or SORM will be inaccurate. Even small discrepancies in the vicinity of the MPP can cause large inaccuracies in the estimated probability of failure, particularly as the number of dimensions increases [115].

2.3.1 FORM

In the First Order Reliability Method (FORM), the limit state $g(x, u)$ is approximated using a hyperplane at the most probable point (MPP). The probability of failure is estimated as

$$\hat{p}_f = \Phi(-\beta), \tag{2.30}$$

where β is the geometric reliability index defined as the distance of the MPP from the origin in the standard normal space. When failure occurs at the origin, then β is given by $\beta = \|u_{MPP}\|_2$ instead, indicating failure at the mean of the random variables.

When a forward reliability analysis (reliability index approach) is performed, the following optimization problem is solved to determine a MPP:

$$\begin{aligned} \underset{u}{\text{minimize}} \quad & u^\top u \\ \text{s.t.} \quad & g(x,u) = \bar{z}, \end{aligned} \tag{2.31}$$

where $\bar{z}$ is the specified limit state level(s) corresponding to failure.

To perform an inverse reliability analysis (performance measure approach), the following optimization problem is solved instead:

$$\begin{aligned} \underset{u}{\text{minimize}} \quad & g(x,u) \\ \text{s.t.} \quad & u^\top u = \beta^2, \end{aligned} \tag{2.32}$$

where the optimal objective value $g(x, u_{MPP})$ is the limit state level $\bar{z}$ corresponding to the prescribed reliability index β.

Although the FORM traditionally performs the MPP optimization using the exact limit state function $g(x,u)$, a variety of limit state approximations have also been considered in an effort to reduce the computational cost associated with solving the optimization problems in 2.31 and 2.32. These include the Advance Mean Value (AMV) method where the limit state is linearized in either the physical space $g(x,y)$ or the standard normal space $g(x,u)$ and the AMV+ method where the limit state linearizations are updated during the optimization [47]. The AMV+ method has shown the most consistent savings in computational cost for comparable accuracy to traditional FORM.

In addition to applying general optimization algorithms to the problems in (2.31) and (2.32), a variety of specialized methods have been developed. Such methods exploit the structure of the MPP search problem and use techniques such as arc searches, local adaptive approximations of the limit state and steepest descent to find the MPP. Youn et al's Hybrid Mean Value (HMV) and HMV+ methods [135] and Du et al's search algorithm [41] are two such examples.

2.4 Response Surface Methods

In response surface methods, the limit state $g(x,y)$ or transformed limit state $g(x,u)$ is approximated with a response surface from which the probability of failure is estimated via (2.1) usually using a sampling method such as Monte Carlo or importance sampling. Polynomial regression, kriging, artificial neural networks (ANN), support vector regression, and other response surfaces have been used in reliability analysis with varying degrees of success. Several comparative studies have also been performed [55, 63, 23, 138], however most studies consider academic problems limited to a handful of random variables.

Fox uses a design of experiments to construct a quadratic response surface for reliability estimation in a problem with 3 random variables [54]. In a later paper, he compares Monte Carlo, Latin Hypercube Sampling (LHS), the Advanced Mean Value method (AMV), SORM, and a quadratic polynomial response surface for reliability estimation [55]. The SORM and quadratic response surface perform reasonably well when compared with Monte Carlo and LHS and the author notes that AMV and SORM are unable to provide an estimation of their accuracy. Gomes and Awruch compare linear, quadratic, and cubic polynomial response surfaces, ANN, Monte Carlo, and the FORM method for simple structural problems [63]. They find neural networks may be able to reduce the number of limit state function evaluations while obtaining comparable reliability estimates as other methods. Bucher and Most compare polynomial response surfaces, ANN, radial basis functions (RBF), and moving least squares for reliability analysis of structures with up to 6 random variables [23]. They find that RBF and ANN converge the fastest in terms of error in $\hat{p}_f$. Kriging [138] and support vector machines [87] have also been used to construct response surfaces for reliability estimation.

The primary drawback of response surface methods is possible inaccuracies in the approximated limit state which affect the calculation of an accurate estimate $\hat{p}_f$. Adaptive refinement techniques can be used to combat this issue: Li et al adaptively refine the estimate of p_f by explicitly replacing all approximate limit state function evaluations by exact evaluations near the limit state [81]. Grooteman does the same but restricts replacement to important regions of the limit state only while also adaptively refining the response surface [66]. Bichon et al instead adaptively refine an approximation of the limit state in a large area of the design space prior to performing a reliability analysis [19].

Normalization

Scaling the sample data $(y^{(i)}, g^{(i)})$ in preparation for fitting a response surface is common practice. It may also help ameliorate the fitting process for response surfaces and is highly recommended for kriging and artificial neural network response surfaces. Given a set of samples $g^{(i)}$ corresponding to $(x, y^{(i)})$, response surface data can be scaled, for example using a max-min scaling:

$$\tilde{g}^{(i)} = \frac{g^{(i)} - \min_j(g^{(j)})}{\max_j(g^{(j)}) - \min_j(g^{(j)})}.$$

Data $y^{(i)}$ can be similarly scaled. The scaled data $(\tilde{y}^{(i)}, \tilde{g}^{(i)})$ can then be used to construct the response surface.

2.4.1 Polynomial

Linear regression is a common method for building response surfaces using polynomials with standard regression techniques. Typically linear or quadratic polynomials are used. A linear polynomial for

$y \in \mathbb{R}^p$ requires at least $p+1$ samples and a full quadratic polynomial with cross terms requires at least $(p^2+3p)/2+1$ samples for construction. The full quadratic polynomial response surface with cross terms can be written as

$$\hat{g}(y) = a + \sum_{i=1}^{p} b_i y_i + \sum_{i=1}^{p} c_i y_i^2 + \sum_{i=1}^{p} \sum_{j>i}^{p} d_{ij} y_i y_j, \tag{2.33}$$

where coefficients a, b_i, c_i, and d_{ij} can be estimated using ordinary least squares. The set of samples $(y^{(i)}, g^{(i)})$ used in the regression typically come from a design of experiments, where a factorial, Box-Muller or other sampling plan such as Latin Hypercube Sampling is implemented. To mitigate the explosion of number of cross-terms when higher-order polynomials are used it is also common to invoke the so-called *sparsity of effects* principle [90]. This principle states that a system is usually dominated by main effects and low order interactions, thus implying that higher order effects and interaction terms can be ignored.

A variety of adaptive polynomial response surface techniques have also been considered [24, 63] where the polynomial response surface is updated with new samples, say for example during a most probable point search in the standard normal space (see section 2.3.1).

Polynomial Chaos

The polynomial chaos expansion (PCE) is a classical uncertainty quantification method which approximates a function by a series of mutually orthogonal polynomials [58]. It is, essentially, polynomial regression. Optimal convergence of the PCE is obtained when the orthogonal polynomials whose weighting functions correspond to the probability density functions of the random variables in the approximated function are used [134]. A limit state $g(y)$ can be approximated as

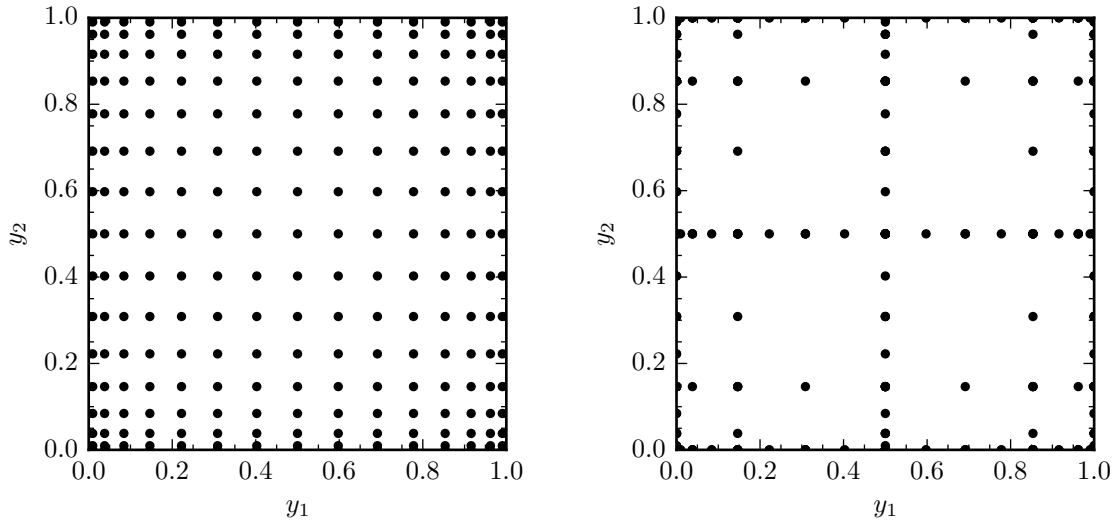
$$\hat{g}(y) = \sum_{i=1}^{t} a_i \Psi_i(y), \tag{2.34}$$

where $\Psi_i(y)$ is the ith basis function, a_i is the ith coefficient, and the expansion is approximate since it has been truncated to include t terms. The difficulty of efficiently constructing the approximation in (2.34) lies in the calculation of the coefficients a_i. In non-intrusive polynomial chaos the coefficients a_i can be estimated using a pseudo-spectral projection process

$$a_i = \frac{1}{\langle \Psi_i^2(y) \rangle} \int_{\mathbb{R}^p} g(y) \Psi_i(y) w(y) \mathrm{d}y, \tag{2.35}$$

where the inner product $\langle \Psi_i(y) \rangle$ can be calculated *a priori* and $w(y)$ is the weight function corresponding to polynomial basis $\Psi_i(y)$ [46]. The multidimensional integral over $\mathbb{R}^p$, the space spanned by all random variables in y, can be approximated using quadrature, of which several techniques have been proposed including tensor product quadrature, cubature, and sparse grids [133]. Figure

2.6 compares a tensor grid with its corresponding sparse grid in two dimensions. For a small number of variables, full tensor product quadrature works well, but for higher dimensions, full tensor product quadrature is too expensive and a sparse grid approach is more tractable although less accurate for non-smooth functions [133].



(a) Tensor grid.

(b) Sparse grid.

Figure 2.6: A full tensor product grid using Clenshaw-Curtis quadrature (15 quadrature points along each variable y_i) is compared with its corresponding nested sparse grid. Note the reduction in number of quadrature points required by the sparse grid leading to more efficient integration and solution of polynomial coefficients.

The benefits of the generalized polynomial chaos expansion include an analytical estimate of the mean and variance of $g(y)$ in terms of the coefficients a_i and adaptivity and error estimates for the approximation. Details can be found in Xiu and Karniadakis [134] regarding optimal bases and calculation of coefficients a_i. Once the expansion is constructed, any sampling method can be used to estimate the probability of failure p_f. Note however that although polynomial chaos provides a general approximation to $g(y)$ including its mean and variance, the approximation $\hat{g}(y)$ is not tuned for the tails of the distribution and may lead to inaccurate estimates of probability of failure.

2.4.2 Kriging

Kriging[3] is a widely used probabilistic method in geostatistics, engineering and data science for interpolation, prediction, and function approximation given a limited set of data. In kriging an initially unknown response $g(y) : y \in \mathbb{R}^p \rightarrow \mathbb{R}$ is modeled as a random field $Z(y)$ due to lack of

[3]The name is derived from Daniel Krige for whom G. Matheron named the method in 1963.

knowledge about the response and is refined as information becomes available. Since the response is modeled as a random field, both a mean $m(y)$ and variance $s^2(y)$ can be calculated for any point y. Thus a major benefit of using a kriging response surface is the metric $s(y)$ which can be used in refinement and is often interpreted as a measure of uncertainty in the kriging model (although not necessarily the true response itself).

One of the central ideas of kriging is to assume a relationship between a set of known data $(y^{(i)}, g(y^{(i)}))$ for $i = 1, \ldots, N$ samples and then estimate information at locations where no data is available. In kriging the relationship is encoded in a *correlogram*, or under stronger assumptions a *covariance kernel* $K(y^{(i)}, y^{(j)})$ which defines the covariance between response values at points $y^{(i)}$ and $y^{(j)}$. Gaussian process regression refers to kriging under the assumption of a Gaussian random field and typically under the additional assumption of a known and constant mean of zero. The following treatment generally follows Chiles and Delfiner (Chapter 3) [29].

A general kriging model can be written as

$$Z^*(y) = \sum_{i=1}^{N} \lambda_i(y) Z(y^{(i)}) + \lambda_0(y), \tag{2.36}$$

where scalars λ_i define a linear combination of the sample response values $Z(y^{(i)}) = g(y^{(i)})$ yielding the random estimator $Z^*(y)$. Note that the weights λ_i depend on the input data y whereas each $Z(y^{(i)})$ does not and is *conditioned* on an already sampled data pair $(y^{(i)}, g(y^{(i)}))$. Here, as is usual, the $*$ superscript denotes estimated data. (2.36) can be concisely written as

$$Z^*(y) = \lambda^\top Z + \lambda_0, \tag{2.37}$$

where $\lambda = [\lambda_1 \ldots \lambda_N]^\top$ and $Z = [Z(y^{(1)}) \ldots Z(y^{(N)})]^\top$.

The derivation of the system of kriging equations from which the coefficients λ_i can be calculated is widely available (see for example [29, 33]) and is not reproduced here. However, it is worth mentioning that the process in general is to first eliminate the bias of the kriging estimator $Z^*(y)$ and then minimize the mean square error of the estimator. Thus kriging is often said to give the Best Linear Unbiased Predictor/Estimator (BLUP/BLUE) for a set of data [29].

Simple Kriging

In simple kriging, the mean for the random function $Z(y)$ is assumed to be known and zero. The following kriging system is then derived which allows determination of the unknown coefficients λ_i at a point y [29, 151-152]:

$$K\lambda = k_0, \tag{2.38}$$

where $K = K(Y, Y)$ is the $N \times N$ matrix of data-to-data covariances for $Y = [y^{(1)} \dots y^{(N)}]^\top$, λ is the vector of weights specific to point y, and $k_0 = K(Y, y)$ is the $N \times 1$ covariance matrix between the data and y. For example, the covariance matrix K takes the form

$$K = \begin{bmatrix} k(y^{(1)}, y^{(1)}) & \dots & k(y^{(1)}, y^{(N)}) \\ \vdots & & \vdots \\ k(y^{(N)}, y^{(1)}) & \dots & k(y^{(N)}, y^{(N)}) \end{bmatrix}, \tag{2.39}$$

where $k(y^{(i)}, y^{(j)})$ is the covariance kernel/function giving the covariance between points $y^{(i)}$ and $y^{(j)}$. The solution of (2.38) is easily implemented via Cholesky decomposition when K is positive definite.

The mean of the kriging estimator $m(y)$ follows from its definition as

$$\begin{aligned} m(y) = E[Z^*] &= \lambda^\top Z \\ &= k_0^\top K^{-1} Z. \end{aligned} \tag{2.40}$$

The kriging variance $s^2(y)$ derives from the minimum mean square error calculation [29] as

$$\begin{aligned} s^2(y) = \mathrm{Var}(Z^*)(y) &= K(y, y) - \lambda^\top k_0 \\ &= K(y, y) - k_0^\top K^{-1} k_0. \end{aligned} \tag{2.41}$$

(2.40) and (2.41) make no assumption on the distribution of the random functions $Z(y)$ but they can also be derived from a Gaussian process perspective. For simple kriging the expressions above for mean and variance are the same as those derived using conditional expectation under the assumption of a Gaussian process [108].

Universal Kriging

In universal kriging, the mean is not assumed to be known. Instead, it is written as a weighted sum of basis functions as

$$m(y) = E[Z^*] = \sum_{i=1}^{B} a_i f_i(y) = a^\top f(y), \tag{2.42}$$

where a_i are fixed coefficients for known basis functions $f_i(y)$ which can be determined through weighted least-squares. The universal kriging model can then be written as

$$Z^*(y) = m(y) + R(y), \tag{2.43}$$

where $m(y)$ is known as the *drift* and $R(y)$ is a zero-mean random function known as the *residual*. The kriging model variance can be derived through the same process as for simple kriging and leads

to a slightly larger and also no longer positive definite system of linear equations, with nevertheless can be readily solved for small to moderate dimensions [29].

Gaussian Process Regression

Gaussian process regression is recovered by taking $Z(y)$ to be a Gaussian random field in simple kriging. In this case, the simple kriging estimator $Z^*(y)$ is the same as the conditional expectation $E[Z(y) \mid Z(y^{(1)}), \ldots, Z(y^{(N)})]$. An estimate at point y is therefore normally distributed as $\mathcal{N}\left(m(y), s^2(y)\right)$.

Covariance Kernels

The covariance kernel $k(y^{(i)}, y^{(j)})$ encodes the spatial relationship in the data. In many cases an assumption of second order *stationarity* is made which implies that the mean and variance of the random field are invariant under translation. In this case the covariance kernel is dependent only on the difference between the two points $h = y^{(i)} - y^{(j)}$ and can be written as $k(h)$. When the covariance kernel is additionally invariant under rotations and reflections, it is said to be *isotropic*.

$k(y^{(i)}, y^{(j)})$ is typically parameterized using *hyperparameters* whose values are chosen to more accurately reflect the underlying data. A common method of choosing hyperparameters is Bayesian maximum likelihood estimation [108]. Typically, the following *log marginal likelihood* optimization problem is solved to determine the hyperparameters θ:

$$\begin{aligned} &\underset{\theta}{\text{maximize}} && \log\left(P[z \mid Y, \theta]\right) \\ &\text{s.t.} && l \leq h \leq u, \end{aligned} \tag{2.44}$$

where $Y = [y^{(1)}, \ldots, y^{(N)}]^\top$ is the array of sampled data and $z = [g(y^{(1)}), \ldots, g(y^{(N)})]^\top$ is the corresponding vector of obtained responses for N samples.

For a Gaussian process, the log marginal likelihood is

$$\log\left(P[z \mid Y, \theta]\right) = -\frac{1}{2} z^\top K^{-1} z - \frac{1}{2}\log|K| - \frac{N}{2}\log 2\pi, \tag{2.45}$$

where K is the $N \times N$ matrix of data-to-data covariances (see (2.39)) [108]. Since the log marginal likelihood is usually multimodal, robustly solving the optimization in (2.44) for a global optimum can be difficult, especially in higher dimensions. Restarting the optimization from multiple initial guesses for θ is recommended for gradient-based optimizers. Common choices for covariance kernels such as the widely used radial basis function and Matérn kernels are described in Appendix B.

2.4.3 Artificial Neural Networks

Artificial neural networks (ANN) draw inspiration from the structure and function of the brain and can be used to approximate a function via a weighted network of nodes called neurons [68]. A commonly used variety of artificial neural network known as the multilayer perceptron has been investigated for use in reliability analysis [63, 23]. A multilayer perceptron network is constructed of several layers, beginning with an input layer, followed by one or more hidden layers, and then ending with an output layer. Each layer consists of several neurons, and each neuron is linked with all neurons in the previous layer. Each neuron applies a transformation to its inputs to generate an output. The transformation T usually takes the form

$$T = f\left(\sum_{i=0}^{n} w_i y_i\right), \tag{2.46}$$

where w_i is the weight for corresponding input y_i and f is an activation function. For example, the hyperbolic tangent activation function $f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$ is commonly used. Neuron weights w_i are determined via back-propagation during which an optimization problem is solved to minimize the regularized squared error loss $\left\|\hat{g}(y^{(i)}) - g(y^{(i)})\right\|_2^2$ given pairs $(y^{(i)}, g(y^{(i)}))$ of training data [100, 68].

To obtain an accurate and generalized function approximation, the number of hidden layers and number of neurons per layer should be tuned (trial-and-error is a popular approach [23]) and a sufficient amount of training data should be used with a cross-validation method such as k-fold cross-validation.

2.5 Hybrid Methods

Hybrid methods for reliability analysis combine multiple approaches for reliability analysis. For example, the methods discussed below combine sampling and response surface methods to efficiently and accurately estimate probabilities of failure. The key idea is to rely on the response surface when it is accurate and fall back on random sampling when it is not.

2.5.1 Response Surface Sample Replacement

Li and Xiu draw attention to the fundamental flaw of the response surface approach for reliability estimation, namely that an inaccurate response surface for the limit state function yields an inaccurate probability of failure estimate [82]. A particularly poignant demonstration of this flaw is illustrated by considering the trivial limit state $g(y) = 0$ for all y, for which failure occurs when $g(y) \leq 0$; thus the probability of failure $p_f = 1$. Now consider the response surface $\hat{g}(y) = 1/N$, $N > 0$, which approximates $g(y)$ and indeed converges to $g(y)$ as $N \to \infty$. However any finite value of N will yield a response surface that predicts $\hat{p}_f = 0$ [82].

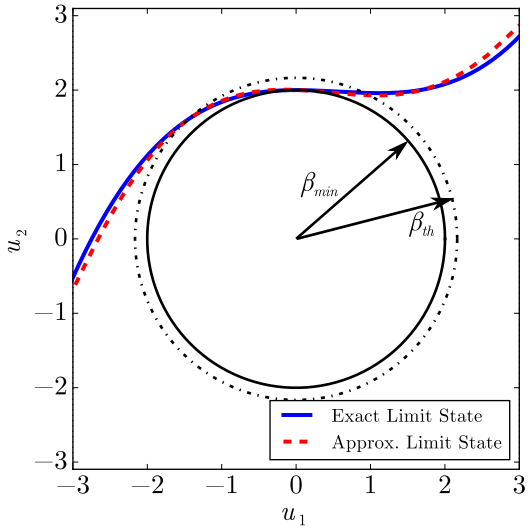
The key idea of Li and Xiu's hybrid method is to sample on the response surface $\hat{g}(y)$ and replace all samples that appear within a certain threshold of the approximated limit state surface with exact limit state evaluations $g(y)$. The threshold can neither be too large which would lead to an inefficient algorithm or too small which could lead to an inaccurate probability of failure estimate. Determination of the so-called critical threshold value can be determined given an estimate of the error of the response surface in the L^p norm [82], which in practice is often unavailable. The authors therefore additionally propose an alternate iterative algorithm based on the same idea but without the need to estimate the response surface error. The alternative method measures convergence of the probability of failure estimate $\hat{p}_f$ to determine termination. Unfortunately this does not necessarily circumvent response surface inaccuracies as it depends upon numerical convergence of the response surface. Li et al extend the method to importance sampling [81]. Application of the response surface sample replacement method is demonstrated to obtain accurate probabilities of failure around 10^{-6} and lower using several hundred to 1000 samples. The authors do not consider the cost of building the response surface or refining the response surface [81].

2.5.2 Adaptive Directional Importance Sampling

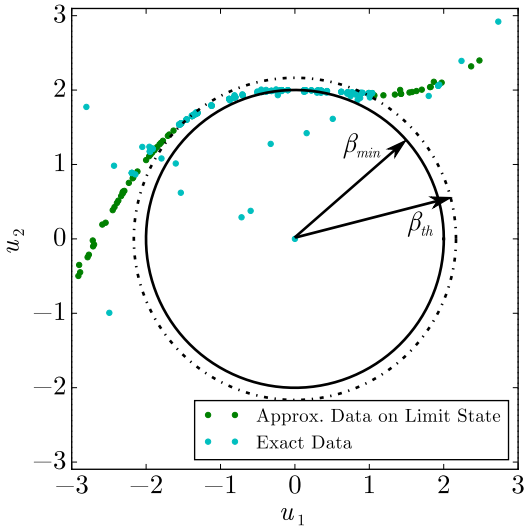
Grooteman introduced the Adaptive Directional Importance Sampling (ADIS) method [66] which is a hybrid response surface reliability analysis approach using directional simulation. ADIS begins with ordinary directional simulation on the exact limit state function. When enough samples have accumulated, a response surface is built for the limit state function and subsequent directional simulations use the response surface instead of the exact limit state function. However, when a directional simulation using the response surface predicts the limit state to be within a critical radius β_{th} of the origin, exact limit state function evaluations are performed instead (see Figure 2.7). These exact evaluations are then used to update the response surface. In this way, ADIS is both adaptive since the response surface becomes more accurate near important regions of the limit state and efficient in the sense that exact evaluations are only performed for potentially important regions which contribute the most to the failure probability p_f.

The critical radius β_{th} is updated such that the ratio γ of the portion of $\hat{p}_f$ obtained from exact evaluations to the portion of $\hat{p}_f$ obtained from approximate evaluations is greater than a certain threshold, say 0.4 (see (2.26)). The critical radius is determined as $\beta_{th} = \beta_{min} + \Delta\beta$ where β_{min} is the current most probable point (MPP) estimate, *i.e.* the minimum sampled distance between the origin and the limit state, and $\Delta\beta$ is an offset initially taken to be 0.1. Convergence of ADIS is determined by monitoring the estimated maximum relative error of $\hat{p}_f$ ε_{rel}. Using (2.28), the maximum relative error is estimated as

$$\varepsilon_{rel} = z_{\alpha/2}\frac{\hat{\sigma}_f}{\hat{p}_f}. \tag{2.47}$$



(a) Exact and approximated limit states.



(b) Approximate evaluations on limit state and all exact limit state function evaluations.

Figure 2.7: Example of ADIS showing the reliability analysis for a nonlinear limit state in the standard normal space, $u \in \mathbb{R}^2$. The response surface for the limit state is an artificial neural network (ANN). The circle inscribed by radius β_{min} marks the location of the most probable point (MPP) and β_{th} marks the radius of the sphere within which exact limit state evaluations will replace response surface evaluations.

When ε_{rel} decreases below a threshold, for example $1.96 * 0.1$ which corresponds to a 95% confidence interval for a coefficient of variation less than 0.1, the algorithm is reinitialized and all previous directional simulations using approximate evaluations are rerun using the updated response surface. The algorithm is converged when such a reinitialization does not result in any exact limit state evaluations. Grooteman demonstrates the ADIS algorithm using a polynomial response surface on a host of test problems showing it can provide fairly accurate probability of failure estimates (1.7% to 15% error) for a variety of test problems with as little as tens to hundreds of function evaluations [66].

2.6 Design Sensitivities

Obtaining design sensitivities from reliability analyses is important for the efficient application of the reliability-based design optimization (RBDO) methods discussed in chapter 4. In this section we discuss design sensitivities for both the forward and the inverse reliability analyses.

2.6.1 Gradient of Probability of Failure

The general case for the probability of failure gradient has already been considered by Uryas'ev [129]. Here we briefly present Uryas'ev's theorem rendered in the current notation, assuming limit state levels $\bar{z}_i = 0$ with no loss of generality.

Theorem 1. *Assume the following conditions are satisfied:*

1. *At the point x all limit states $g_i(x_0, y)$ for $i = 1, \ldots, m$ are active.*
2. *The failure set F is bounded in a neighborhood of x.*
3. *The limit state $g(x, y)$ has continuous partial derivatives $\nabla_x g(x, y)$ and $\nabla_y g(x, y)$.*
4. *The density function $f_y(x, y)$ has continuous partial derivatives $\nabla_x f_y(x, y)$ and $\nabla_y f_y(x, y)$.*
5. *A matrix function $H_l(x, y)$ exists where*

$$H_l(x, y)\nabla_y g_{1:l}(x, y) + \nabla_x g_{1:l}(x, y) = 0,$$

   where $g_{1:l}(x, y) \equiv [g_1(x, y), \ldots, g_l(x, y)]^\top$ for $l \leq m$.

6. *$H_l(x, y)$ has continuous partial derivative $\nabla_y H_l(x, y)$.*
7. *The gradient $\nabla_y g(x, y)$ is non-zero on the limit state boundary $\partial F(x) = \{y \in \mathbb{R}^p \mid g(x, y) = 0\}$.*
8. *The vectors $\nabla_y g(x, y)$ are linearly independent.*

Then, the gradient of the probability of failure $\nabla_x p_f$ is

$$\begin{aligned} \nabla_x p_f(x) = &\int_{g(x,y)\leq 0} \left[\nabla_x f_y(x, y) + div_y \left(f_y(x, y) H_l(x, y)\right)\right] dy \\ &- \sum_{i=l+1}^{m} \int_{g(x,y)=0} \frac{f_y(x, y)}{\|\nabla_y g_i(x, y)\|} \left[\nabla_x g_i(x, y) + H_l(x, y)\nabla_y g_i(x, y)\right] dS. \end{aligned} \tag{2.48}$$

A technical proof of theorem 1 can be found in [129]. A simpler alternative proof is sketched in [130]. The solution of the nonlinear system of equations to obtain $H_l(x, y)$ is also discussed in the above references. Theorem 1 is flexible in the sense that it covers a variety of limit state partitionings (sets $\{g_i(x, y) \in \mathbb{R}^m \mid i = 1, \ldots, l\}$ and $\{g_i(x, y) \in \mathbb{R}^m \mid i = l + 1, \ldots, m\}$ for a variety of l values) which result in equivalent but different expressions for the probability of failure gradient as the sum of a volume integral and/or multiple boundary integrals, assuming that the prerequisite conditions are satisfied.

For example, choosing $l = 0$ removes the matrix function $H_l(x, y)$ from (2.48) and the probability of failure gradient is

$$\nabla_x p_f(x) = \int_{g(x,y)\leq 0} \nabla_x f_y(x, y)\mathrm{d}y - \sum_{i=l+1}^{m} \int_{g(x,y)=0} \frac{f_y(x, y)}{\|\nabla_y g_i(x, y)\|} \nabla_x g_i(x, y)\mathrm{d}S. \tag{2.49}$$

When the density $f_y(x, y) = f_y(y)$ and only one limit state $g(x, y)$ is present (2.49) simplifies to

$$\nabla_x p_f(x) = -\int_{g(x,y)=0} \frac{f_y(y)}{\|\nabla_y g(x, y)\|} \nabla_x g(x, y)\mathrm{d}S. \tag{2.50}$$

Uryas'ev [129, 130] notes that the surface integral(s) in (2.48) and (2.50) are often difficult to evaluate numerically, thus it is advantageous to choose $l = m$ if possible, thereby removing the surface integral and leaving a volume integral which can be easily estimated, for example by Monte Carlo sampling. In this case

$$\nabla_x p_f(x) = \int_{g(x,y)\leq 0} \left[\nabla_x f_y(x, y) + \mathrm{div}_y \left(f_y(x, y)H_m(x, y)\right)\right] \mathrm{d}y. \tag{2.51}$$

However, it is shown below that for certain reliability analysis methods when the limit state boundary is known, such as directional simulation, the surface integral in (2.50) can be easily obtained.

Application to FORM

In the First Order Reliability Method (FORM, see section 2.3.1), the limit state $g(x, y)$ is transformed to the standard normal space and approximated as a hyperplane at the most probable point (MPP) of failure. Let $\hat{g}(x, u)$ be the transformed and linearized limit state. Under the assumptions of FORM, the gradient of the probability of failure is

$$\begin{aligned} \nabla_x p_f(x) &\approx -\int_{\hat{g}(x,u)=0} \frac{f_u(u)}{\|\nabla_u \hat{g}(x, u)\|} \nabla_x \hat{g}(x, u)\mathrm{d}S \\ &= -\frac{1}{\|\nabla_u \hat{g}(x, u)\|} \int_{\hat{g}(x,u)=0} f_u(u)\nabla_x \hat{g}(x, u)\mathrm{d}S, \end{aligned} \tag{2.52}$$

which follows since the gradient of the linearized limit state is constant. Furthermore assume that the MPP encompasses all information about the failure of the system. Then letting $\nabla_x \hat{g}(x, u) \approx \nabla_x \hat{g}(x, u_{MPP})$ yields

$$\nabla_x p_f(x) \approx -\frac{\nabla_x \hat{g}(x, u_{MPP})}{\|\nabla_u \hat{g}(x, u)\|} \int_{\hat{g}(x,u)=0} f_u(u)\mathrm{d}S, \tag{2.53}$$

To facilitate the calculation of the integral in (2.53) represent the linearized limit state as the hyperplane $\hat{g}(x, u) = a(x)^\top u + b(x)$ in the standard normal space. Noting that the density kernel $f_u(u)$ is rotationally symmetric, rotate the coordinate system such that the linear limit state is

aligned with one axis. This leads to $\hat{a} = [0, 0, \dots 1]^\top$ and $\hat{b} = [0, 0, \dots, \beta]^\top$ with no loss of generality where β is the geometric reliability index from section 2.1.4. Thus the limit state can be written as $\hat{g}(u, x) = u_n + \beta(x)$. The integral from (2.53) is derived as

$$\begin{aligned}
\int_{\hat{g}(x,u)=0} f_u(u)\mathrm{d}u &= \int_{\hat{g}(x,u)=0} (2\pi)^{-\frac{p}{2}} e^{\frac{1}{2}u^\top u}\mathrm{d}u \\
&= (2\pi)^{-\frac{p}{2}} \int_{\mathbb{R}^{p-1}} e^{-\frac{1}{2}u^\top u - \frac{1}{2}\beta^2}\mathrm{d}u \\
&= (2\pi)^{-\frac{p}{2}} (2\pi)^{\frac{p-1}{2}} e^{-\frac{1}{2}\beta^2} \\
&= \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\beta^2} \\
&= \varphi(\beta),
\end{aligned} \tag{2.54}$$

where $\varphi(\beta)$ is the standard normal density. Thus, for a limit state that is linear in the standard normal space where the most probable point is assumed to contain all design sensitivity information, the probability of failure gradient is

$$\nabla_x p_f(x) \approx -\frac{\nabla_x \hat{g}(x, u_{MPP})}{\|\nabla_u \hat{g}(x, u)\|} \varphi(\beta), \tag{2.55}$$

which is in agreement with the result presented by Allen and Maute [8] which is based on the use of FORM. The approximation in (2.55) is valid when the limit state is close to linear and the design sensitivity at the most probable point is representative among the design sensitivities in neighboring regions on the limit state surface.

Application to Directional Simulation

In directional simulation (see section 2.2.3), the limit state $g(x, y)$ is transformed to the standard normal space and an estimator of the probability of failure is constructed by simulating random directions and searching along them until failure occurs. Each directional simulation yields a direction $u^{(i)}$ and a radius $r^{(i)}$ giving the distance from the origin to the limit state along the ith random direction. In this way an approximation of the limit state surface is mapped out. N simulations can then be combined into an estimator for the probability of failure p_f as shown in (2.26).

When directional simulation is used, the probability of failure gradient can be obtained with little effort using the existing samples from the directional simulation. Let $u_i = r^{(i)}u^{(i)}$ be the point on the limit state $g(x, u) = \bar{z}$ corresponding to the ith sample and approximate the integral from

(2.50) using a Riemann sum over the limit state surface using the N simulations as

$$\begin{aligned}\nabla_x p_f(x) &= -\int_{g(x,u)=\bar{z}} \frac{f_u(u)}{\|\nabla_u g(x,u)\|}\nabla_x g(x,u)\mathrm{d}S \\ &\approx -\sum_{i=1}^{N} \frac{f_u(u_i)}{\|\nabla_u g(x,u_i)\|}\nabla_x g(x,u_i)\Delta S_i.\end{aligned} \tag{2.56}$$

Given i.i.d. directions $u^{(i)}$, an appropriate area element ΔS_i for each simulation, and assuming the conditions of theorem 1 leading to (2.50) are satisfied, the summation should converge as $N \to \infty$. The value to which it converges is limited by the restrictions of directional simulation; for a non-star-shaped limit state, the gradient will be approximate if only a single root is sought during the linesearch procedure.

The appropriate area element ΔS_i for the ith simulation is the fractional surface area of the hypersphere of radius $r^{(i)}$ with a correction γ_i accounting for the orientation of the limit state surface with respect to the origin as shown in Figure 2.8a:

$$\Delta S_i = \frac{1}{N} S_{p-1}(0, r^{(i)})\gamma_i, \tag{2.57}$$

where $S_n(0,r)$ is the surface area of the n-sphere centered at the origin of radius r given by

$$S_n(0,r) = \frac{2\pi^{\frac{n+1}{2}}}{\Gamma(\frac{n+1}{2})} r^n. \tag{2.58}$$

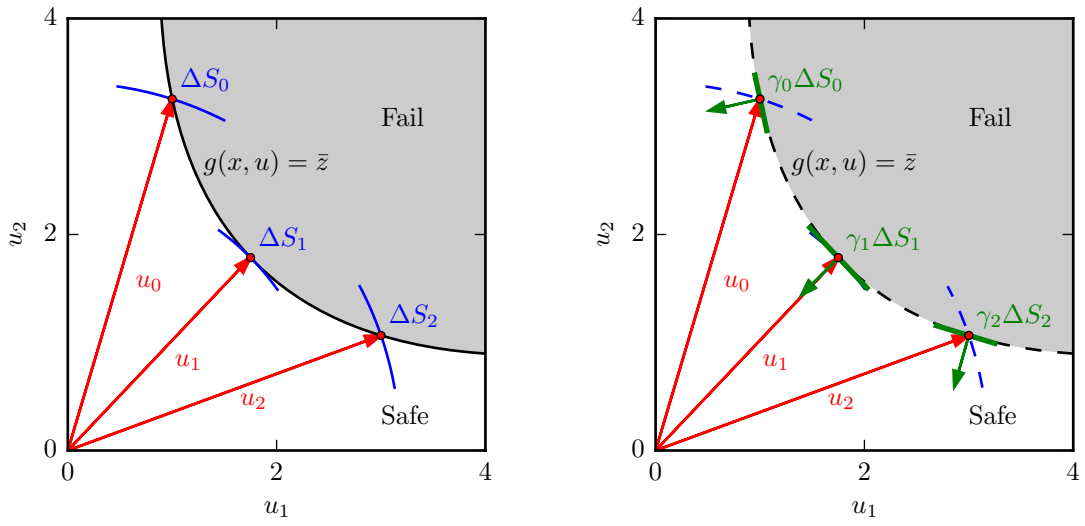
The correction term γ_i accounts for the tilting of the limit state surface away from the origin. Here we define γ_i such that ΔS_i is the projection of the fractional hypersphere surface area at u_i onto the plane tangent to the limit state surface at the same point as shown in Figure 2.8b:

$$\gamma_i = \left(\frac{u_i^\top \nabla_u g(x,u_i)}{\|u_i\| \, \|\nabla_u g(x,u_i)\|} \right)^{-1}. \tag{2.59}$$

Noting that $\|u_i\| = r^{(i)}$ and combining (2.56), (2.57), and (2.59), the probability of failure gradient can be approximated as

$$\nabla_x p_f(x) \approx -\frac{1}{N} \sum_{i=1}^{N} \frac{f_u(u_i) \cdot \nabla_x g(x,u_i) \cdot S_{p-1}(0, r^{(i)}) \cdot r^{(i)}}{u_i^\top \nabla_u g(x,u_i)}. \tag{2.60}$$

Figure 2.9 shows the convergence of the Riemann sum in (2.60) for the limit state derivatives in the standard cantilever beam test problem described later in section 3.7.2.



(a) Three simulated directions that have intercepted the limit state $g(x,u) = \bar{z}$ for $u \in \mathbb{R}^2$ are shown. The integrand at each point u_i simulated on the limit state is weighted by the corresponding fractional area of a hypersphere with radius $r^{(i)} = \|u_i\|$. In two dimensions the fractional area is given by ${}^{S_1(0,\|u_i\|)}/_N = {}^{2\pi\|u_i\|}/_N$. The diagram is shown without the γ_i correction term for clarity.

(b) Three simulated directions that have intercepted the limit state $g(x,u) = \bar{z}$ for $u \in \mathbb{R}^2$ are shown. Corrected area contributions to the Riemann sum $\gamma_i \Delta S_i$ are shown in green. Note that the fractional hypersphere area for u_1 approximates the limit state well, however the projection of the fractional hypersphere area onto the plane tangent to the limit state at points u_0 and u_2 provides a better estimate of area.

Figure 2.8: Riemann sum procedure used to approximate the gradient of the failure probability.

2.6.2 Gradient of Quantile Function

Gradients for the limit state levels $z \in \mathbb{R}^m$ with respect to deterministic design variables $x \in \mathbb{R}^n$ are also desired when an inverse reliability analysis is used in a reliability-based design optimization. Recall $z = z(x) = F_g^{-1}(p_f(x))$ where $F_g^{-1}(\cdot) : \mathbb{R} \to \mathbb{R}^m$ is the quantile function for the limit state(s) $g(x,y)$. The chain rule gives

$$\begin{aligned} D_x z &= D_x(F^{-1} \circ p_f)(x_0) \\ &= \left[D_{p_f} F^{-1}\right](p_f(x_0))\,[D_x p_f](x_0) \\ &= \left[D_{p_f} z\right](x_0)\,[D_x p_f](x_0), \end{aligned} \tag{2.61}$$

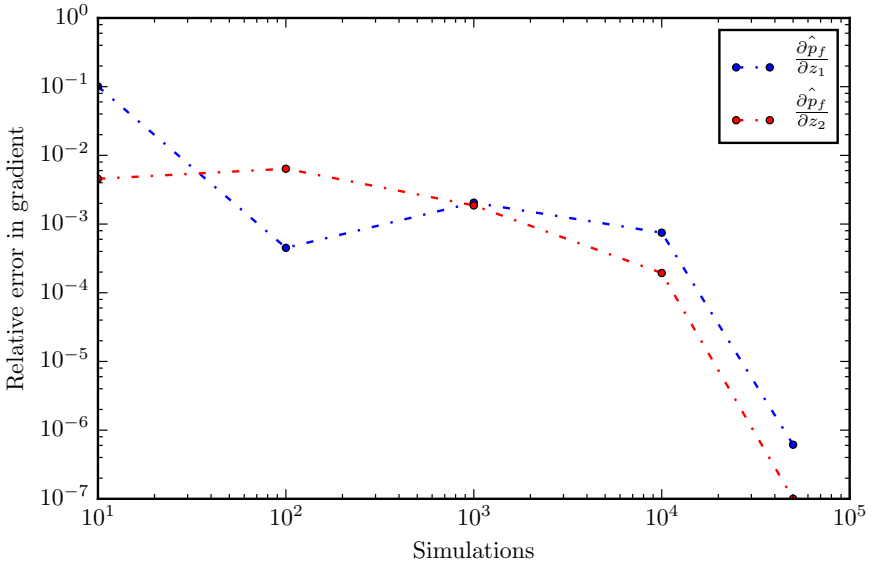


Figure 2.9: Convergence of the Riemann sum in (2.60) for the derivatives of p_f with respect to the two limit states (stress z_1 and displacement z_2) in the cantilever beam problem ($y \in \mathbb{R}^4$, see section 3.7.2). The relative error in the derivatives is measured with respect to a finite difference approximation. Note that even a coarse directional simulation (∼100 sampled directions) is able to provide a reasonable estimate of the derivatives (within 1% error) for this problem at this particular design point.

where the derivative matrices $D_{p_f} z$ and $D_x p_f$ are given by

$$D_{p_f} z(x_0) = \begin{bmatrix} \frac{\partial z_1}{\partial p_f} \\ \vdots \\ \frac{\partial z_m}{\partial p_f} \end{bmatrix}, \tag{2.62}$$

$$D_x p_f(x_0) = \begin{bmatrix} \frac{\partial p_f}{\partial x_1} & \cdots & \frac{\partial p_f}{\partial x_n} \end{bmatrix}, \tag{2.63}$$

yielding the derivative matrix

$$D_x z(x_0) = \begin{bmatrix} \frac{\partial z_1}{\partial p_f}\frac{\partial p_f}{\partial x_1} & \cdots & \frac{\partial z_1}{\partial p_f}\frac{\partial p_f}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial z_m}{\partial p_f}\frac{\partial p_f}{\partial x_1} & \cdots & \frac{\partial z_m}{\partial p_f}\frac{\partial p_f}{\partial x_n} \end{bmatrix}, \tag{2.64}$$

whose entries are succinctly written as

$$\frac{\partial z_i}{\partial x_j} = \frac{\partial z_i}{\partial p_f} \mid_{z=p_f(x_0)} \frac{\partial p_f}{\partial x_j} \mid_{x=x_0} . \tag{2.65}$$

The partials $^{\partial p_f}/_{\partial x_j}$ were obtained in section 2.6.1. The partials $^{\partial z_i}/_{\partial p_f}$ can be obtained easily if we assume the probability of failure p_f can be written as

$$p_f(z) = p_1(z_1) + \ldots + p_m(z_m). \tag{2.66}$$

In other words, each limit state is responsible for an independent contribution to the total failure probability p_f. Under such an assumption

$$\frac{\partial p_f}{\partial z_i} = \frac{\mathrm{d}p_i}{\mathrm{d}z_i}. \tag{2.67}$$

In this way we can define

$$\frac{\partial z_i}{\partial p_f} = \left(\frac{\mathrm{d}p_i}{\mathrm{d}z_i}\right)^{-1}. \tag{2.68}$$

In the case that $^{\mathrm{d}p_f}/_{\mathrm{d}z_i} = 0$ such as when the ith limit state does not contribute to the probability of failure, we define $^{\partial z_i}/_{\partial p_f} = 0$. Thus, the derivative of the quantile function $z(x)$ is the $m \times n$ matrix whose ijth entry is

$$\frac{\partial z_i}{\partial x_j} = \left(\frac{\mathrm{d}p_i}{\mathrm{d}z_i}\right)^{-1} \mid_{z=p_f(x_0)} \frac{\partial p_f}{\partial x_j} \mid_{x=x_0} . \tag{2.69}$$

2.6.3 Gradient Simplification Using x as Hyperparameters

Other authors have considered incorporating the deterministic variables x as hyperparameters for the random vector y which simplifies the probability of failure gradient calculation [124, 42], possibly at the expense of increasing the size of the random space. This is the case for example when x parameterizes the distributions of y such as when geometric uncertainties are considered and x corresponds to the mean of the geometric dimensions. Under these circumstances $y = y(x)$ and the probability of failure is given by

$$p_f(x) = \int_{g(x,y)\leq \bar{z}} f_y(x,y)\mathrm{d}y. \tag{2.70}$$

Since any deterministic variable x can be considered as a random uniform or normal variable with very small variance, (2.70) is generally valid. The derivative of the failure probability is

$$\begin{aligned}\frac{\partial p_f}{\partial x_i}(x) &= \frac{\partial}{\partial x_i}\int_{g(x,y)\leq\bar{z}} f_y(x,y)\mathrm{d}y \\ &= \int_{g(x,y)\leq\bar{z}} \frac{\partial f_y(x,y)}{\partial x_i}\mathrm{d}y \\ &= \int_{g(x,y)\leq\bar{z}} \frac{\frac{\partial f_y(x,y)}{\partial x_i}}{f_y(x,y)} f_y(x,y)\mathrm{d}y \\ &= E_y\left[\mathbb{1}_{g(x,y)\leq\bar{z}}\frac{\frac{\partial f_y(x,y)}{\partial x_i}}{f_y(x,y)}\right]\end{aligned} \tag{2.71}$$

of which an unbiased estimator exists. Note that the gradient of the probability distribution function $f_y(x,y)$ may be obtained analytically in many circumstances [79].

Chapter 3

Adaptive Kriging Inverse Reliability Analysis (AKIRA)

In this chapter a new method is introduced for the solution of the inverse system reliability analysis problem for multiple limit states while adaptively refining kriging models of the limit state functions. The Adaptive Kriging Inverse Reliability Analysis (AKIRA) method retains the benefits of an inverse reliability analysis method for use in reliability-based design optimization (RBDO) while also avoiding the limitations of MPP methods and sampling methods for reliability analysis (see chapter 2 for an in-depth discussion). This chapter also introduces a new adaptive refinement heuristic for kriging-based reliability analysis as well as an adaptive cokriging method for approximating limit state design sensitivities in cases where they are expensive to obtain.

Section 3.1 introduces the AKIRA algorithm and summarizes existing kriging-based reliability analysis methods. Details are provided as necessary in subsequent sections as the details of AKIRA are explained. Section 3.2 reviews the construction of kriging models for system reliability analysis problems. Section 3.3 surveys kriging model refinement techniques for reliability analysis and introduces a new heuristic for refinement. The new heuristic is simpler and more efficient than similar pointwise criteria. Section 3.4 formulates the optimization problem for the inverse reliability analysis — the heart of the AKIRA algorithm — and proposes solution procedures. Next, the estimation of design sensitivities is discussed for use in RBDO in section 3.5, followed by the proposal of a new adaptive cokriging method for the estimation of limit state sensitivities in section 3.6. The method can be used after the termination of AKIRA to estimate sensitivities and is useful in situations where function evaluations are expensive and/or the most probable point of failure (MPP) is not representative of the entire limit state's contributions to the probability of failure. Finally, section 3.7 demonstrates AKIRA on several problems of academic and engineering interest.

3.1 Introduction

AKIRA builds upon the work of existing kriging-based methods which solve the forward reliability analysis problem [18, 42, 44, 26, 71, 72]. In a kriging-based method for reliability analysis kriging response surfaces are constructed for each limit state and a probability of failure is estimated from the kriging models, typically via a sampling method, which bypasses the need for many costly evaluations of the original limit state functions. Kriging response surfaces are typically chosen since they provide an empirical error estimate, the kriging variance, which can be used as a measure of uncertainty for the response surface prediction of the underlying limit state. As a result, the use of adaptive refinement methods are well-suited for kriging response surfaces where the approximate kriging model is refined only in regions impacting the probability of failure. An additional benefit of the kriging response surface is its exact interpolation property for sampled data points, unlike a polynomial regression response surface.

The solution of the inverse reliability analysis problem for multiple limit states is the heart of the AKIRA method (see section 2.1.5 for a comparison of the forward and inverse reliability analysis approaches). There are many practical benefits to solving the inverse reliability analysis problem during a reliability-based design optimization such as: 1) robustness when the probability of failure p_f is estimated to be 0 or 1, and 2) no need to resolve probabilities of failure less than the prescribed design target $\bar{p}_f$. In addition, the numerical calculation of a limit state may be less robust in off-design conditions where the probability of failure is less than the target; thus solving the inverse problem for an *a priori* prescribed probability of failure $\bar{p}_f$ can be more robust in terms of limit state evaluations.

The inverse reliability analysis problem is cast and solved in AKIRA as a constrained least squares problem. This formulation ensures a local solution to the inverse problem *which is also consistent* with the original limit state levels. Although the solution unfortunately may not necessarily be unique for multiple limit states, it still exhibits desirable properties for reliability-based design optimization. The solution of the inverse problem can then be efficiently used in a sequential RBDO method (see chapter 4).

In this dissertation, directional simulation is used to integrate probabilities of failure over the kriging models for the limit states, although any sampling method may be used to obtain a probability of failure estimate. The choice of directional simulation is primarily motivated by the fact that an outcome of the directional simulation is a set of samples lying exactly on the limit state surface (see section 2.2.3). The samples can then be used in efficient gradient approximation procedures. An additional benefit of directional simulation is its enhanced convergence rate compared to Monte Carlo.

3.1.1 Survey of Kriging Reliability Analysis Methods

No kriging-based reliability analysis methods known to the author have been directly applied for the solution of the inverse reliability analysis problem with multiple limit states, although several have been proposed for the forward reliability analysis problem [18, 42, 44, 26, 71, 72] and the inverse reliability analysis problem with a single limit state [17, 95]. Of these, the Efficient Global Reliability Analysis (EGRA) method was among the first of kriging-based system-level reliability analysis methods [18]. EGRA proposed the use of an easily calculated Expected Feasibility function to select the next point to refine the kriging model and was later applied for the inverse reliability analysis problem with a single limit state [17]. The Active learning method combining Kriging and Monte Carlo Simulation (AK-MCS) proposed a different metric for refinement of the kriging model [44]. Methods using Stepwise Uncertainty Reduction (SUR) were proposed soon thereafter and were shown to outperform other methods relying on pointwise data alone such as EGRA and AK-MCS [16, 28]. However methods relying on SUR are more expensive since they solve an optimization problem to determine the optimal one-step lookahead policy for selecting the next point. A different approach for intelligently refining the kriging model was proposed in the Global Sensitivity Analysis for Surrogates (GSAS) method, where a global sensitivity analysis is used to guide model refinement [71]. The Efficient Kriging Surrogate modeling Approach (EKSA) also proposed a solution for forward reliability analysis with multiple limit states by using a composite limit state based on the singular values of the system responses in addition to individual limit state kriging models [72].

All kriging-based reliability analysis algorithms share the same general procedure:

1. Perform a space-filling design of experiments (DoE)
2. Construct one or more initial kriging models from the DoE results
3. Select a point or points to update the kriging model(s)
4. Evaluate the point(s) and update the kriging model(s)
5. Estimate the probability of failure from the kriging model(s)
6. Repeat from step 3 until the kriging model(s) and probability of failure estimate are converged

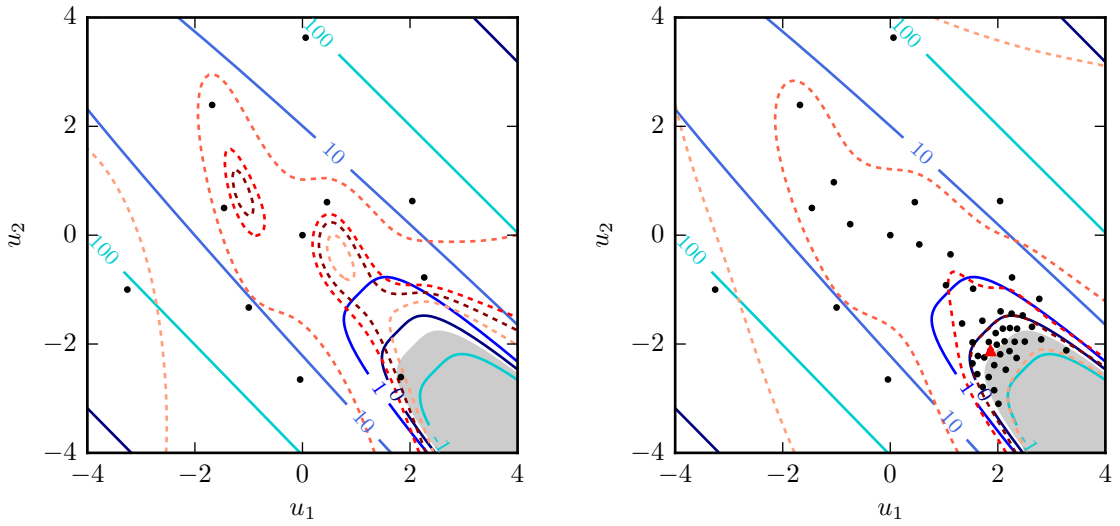
Kriging-based reliability analysis methods differ primarily in how they manage multiple limit states, how they refine the model(s), and how the probability of failure is estimated. Most methods have only been described for use with a single limit state. Many refinement methods originally selected a single "best" point prompting improvements to adaptive refinement algorithms where a batch of points is selected instead, which allows user simulations to be dispatched in parallel [42, 26]. Such details on the algorithms in the brief summary above are uncovered as necessary in the remainder of this chapter.

3.1.2 AKIRA Algorithm Overview

The Adaptive Kriging Inverse Reliability Analysis (AKIRA) contains many of the same features as a kriging-based forward reliability analysis algorithm but additionally includes a nonlinear optimization which is used to solve the inverse reliability analysis problem (see section 2.1.5), *i.e.* estimate limit state levels z for a prescribed probability of failure $\bar{p}_f$ and limit state failure levels $\bar{z}$:

$$\begin{aligned} \underset{z}{\text{minimize}} \quad & \phi(z) = \frac{1}{2} \left\| z - \bar{z} \right\|_2^2 \\ \text{s.t.} \quad & \hat{p}_f(z) = \bar{p}_f. \end{aligned} \tag{3.1}$$

The formulation and solution of the optimization problem in (3.1) is central to the AKIRA method and is described in detail in section 3.4. The remaining improvements to kriging-based reliability analysis presented in this chapter and included in AKIRA, namely an alternate adaptive refinement method for kriging-based reliability analysis as well as the use of directional simulation and estimation of design sensitivities, can be readily substituted by other existing techniques.



(a) Initial Monte Carlo samples.

(b) Limit state approximation after 39 iterations.

Figure 3.1: Illustration of AKIRA for a nonlinear limit state in $\mathbb{R}^2$ [21]. Exact limit state contours are shown as solid lines whereas contours for the kriging approximation are shown as dashed lines. The initial set of 10 Monte Carlo samples and a sample at the mean $u = [0, 0]^\top$ is augmented with 39 more samples that refine the kriging model of the limit state at the estimated failure region boundary corresponding to a prescribed probability of failure $\bar{p}_f = 10^{-3}$. The true failure region is shown shaded in gray. The most recently added sample is marked with a red triangle. Note how AKIRA predominantly refines the kriging model in the vicinity of the estimated limit state level $z = -0.3868$, but not in regions away from the failure boundary.

Algorithm 1 gives an overview of AKIRA and the remainder of the chapter provides details on each part. The limit states $g(u) : \mathbb{R}^p \to \mathbb{R}^m$ use vector notation where $g(u) \equiv [g_1(u) \dots g_m(u)]^\top$ and $g_i(u)$ is the ith limit state. The dependence of the limit state on the deterministic design variables x is dropped since the reliability analysis is performed for fixed x, *i.e.* a specific design. AKIRA begins with a standard space-filling design of experiments in the standard normal space (lines 1–3) from which an initial set of kriging surrogates $\hat{g}_i(u)$ are constructed, one for each limit state $g_i(u)$ (line 4). Each kriging model $\hat{g}_i(u)$ is associated with a mean $m_i(u)$ and a standard deviation $s_i(u)$. Vector notation is similarly used for the kriging response surfaces, means, and standard deviations; for example $\hat{g}(u) \equiv [\hat{g}_1(u) \dots \hat{g}_m(u)]^\top$.

An initial estimate is made for the limit state levels z^0 given the prescribed probability of failure $\bar{p}_f$ (line 5). Then an iterative loop (line 6) is entered where the kriging models are partially refined at each iteration assuming limit state levels z^k for the kth iteration (lines 19–22) and an optimization carried out to determine a new estimate z^{k+1} for the limit state levels (line 25), which may depend on the updating of a penalty parameter each iteration (line 24). The penalty parameter corresponds to the nonlinear equality constraint in the inverse reliability analysis optimization problem shown in (3.1). Consequently, at each iteration the kriging models become more informed of the failure regions corresponding to the current limit state level z^k and in addition the estimate of the limit state levels corresponding to the prescribed probability of failure $\bar{p}_f$ is refined.

Convergence of the algorithm is determined by assessing the epistemic uncertainty in the kriging model's probability of failure prediction π_f (lines 7–12, 16), the error in estimated limit state levels z^k ε_z between iterations (lines 13, 17), and the satisfaction of the the $\hat{p}_f = \bar{p}_f$ constraint (lines 14–15). Typical values of the tolerances τ_z, τ_{p_f} and τ_p are 10^{-2}, 10^{-6}, and 0.1, respectively. The two-sided confidence interval α may be as low as 50% or as large as 99%. In situations where the kriging model is a good predictor of the underlying limit states, a smaller interval is desirable and will lead to faster convergence.

Note that the kriging model may not be updated each iteration if the current limit state levels do not correspond to the prescribed probability of failure (line 15); instead a new attempt at determining the correct limit state levels is made (lines 24–25). When the probability of failure constraint is satisfied or little progress has been made towards the constraint satisfaction (line 15), the kriging model is updated again (lines 16–22). While unfortunately there is no proof of convergence due to the epistemic nature of the kriging models, the framework outlined in algorithm 1 has successfully converged on a variety of reliability test problems as well as realistic engineering applications.

Algorithm 1: Adaptive Kriging Inverse Reliability Analysis (AKIRA)

Data: prescribed probability of failure $\bar{p}_f$, exact limit state functions $g(u)$ and failure levels $\bar{z}$, penalty ρ, tolerances τ_z, τ_{p_f}, and τ_p, maximum iterations k_{max}, number of points r to add to kriging models each iteration, standard normal variate $z_{\alpha/2}$ and confidence interval α used in kriging model convergence estimation

Result: limit state levels z, kriging models of limit states $\hat{g}(u)$

$u^{(j)}$ for $j = 1 \dots r^0 \leftarrow$ Generate a set of space-filling samples;
foreach $u^{(j)}$ **do**
    Evaluate exact limit state functions $g(u^{(j)})$;
$\hat{g}^0(u) \leftarrow$ Build initial kriging models of limit states using $(u^{(j)}, g(u^{(j)}))$;
$z^0 \leftarrow$ Estimate initial limit state levels;
for $k \leftarrow 0$ **to** k_{max} **do**
    $\hat{p}_f \leftarrow$ Estimate failure probability using kriging means $m^k(u)$ and limit state levels z^k;
    $\hat{p}_f^- \leftarrow$ Estimate failure probability using $m^k(u) - z_{\alpha/2}s^k(u)$ and limit state levels z^k;
    $\hat{p}_f^+ \leftarrow$ Estimate failure probability using $m^k(u) + z_{\alpha/2}s^k(u)$ and limit state levels z^k;
    $\pi_f^k \leftarrow$ Calculate $\hat{p}_f$ convergence ratio: $\frac{\max\left(\hat{p}_f - \hat{p}_f^+, \hat{p}_f^- - \hat{p}_f\right)}{\hat{p}_f}$;
    if $\pi_f^k > \pi_f^{k-1} > \pi_f^{k-2} > \dots$ **then**
        return since kriging model is ill-conditioned;
    $\varepsilon_z \leftarrow$ Estimate max error in z between iterations: $\max_i\{|\frac{z_i^k - z_i^{k-1}}{1+|z_i^k|}|\}$;
    $v^k \leftarrow$ Calculate constraint violation: $\|\hat{p}_f - \bar{p}_f\|$;
    if $v^k \leq \tau_{p_f}$ **or** $|\frac{v^k - v^{k-1}}{v^k}| < 0.1$ **then**
        if $\pi_f^k \leq \tau_p$ **then**
            if $\varepsilon_z < \tau_z$ **then**
                return z, $\hat{g}(u)$;
        $u^{(j)}$ for $j = 1 \dots r \leftarrow$ Select r points for adaptive refinement of kriging model;
        foreach $u^{(j)}$ **do**
            Evaluate exact limit state functions $g(u^{(j)})$;
        $\hat{g}^{k+1}(u) \leftarrow$ Update kriging models of limit states using $(u^{(j)}, g(u^{(j)}))$;
    else
        $\rho \leftarrow$ Increase penalty to, say, 10ρ;
    $z^{k+1} \leftarrow$ Solve inverse reliability analysis for new limit state levels;

3.2 Model Construction

3.2.1 Individual or Composite?

Most kriging-based system reliability analysis methods fall into one of three categories: an *individual limit state* method where a separate kriging model is built and refined separately for each limit state, a *composite limit state* method where a single kriging model is built for all limit states, and an *individual limit state with composite learning* method where a separate kriging model is built for each limit state but each is refined using a composite learning function [72]. When individual kriging models are built, each model may be refined individually but correlations between limit states are not taken into account, which can lead to a less efficient method requiring more model refinement overall between the models. In general, refining a group of individual models together using a single composite learning criterion appears to be more efficient than individual refinement using separate criteria as certain limit states may be more important than others [18]. When a composite limit state kriging model is used instead, correlations between limit states are inherently taken into account but inaccuracies at limit state intersections may occur due to the smoothing property of the kriging model.

Most existing system reliability analysis methods use a composite limit state approach. However, in the EKSA method, the authors propose a hybrid approach where both individual limit state kriging models and a single kriging model based on the Singular Value Decomposition (SVD) of the matrix of limit state responses at training points is constructed [72]. A hybrid limit state model is constructed by choosing between the SVD-kriging model or individual limit state kriging model at different locations, depending on which model is more accurate. Since the SVD-kriging model captures correlations between limit states, the EKSA method addresses the shortcomings of using a purely individual limit state or composite limit state-type approach.

In AKIRA an individual limit state with composite learning approach is taken. Each limit state is modeled using an independent kriging response surface. Then a composite learning function (see section 3.3) is used to select the next shared training point(s) for each kriging model based on information from each model. Since an individual limit state approach is used, accurately modeling the intersections of limit states poses no issues for the kriging models. The use of a composite learning function also ensures that limit state evaluations are not wasted on limit states that do not need to be refined.

3.2.2 Initial Construction

Loeppky et al argue in favor of the $N = 10p$ rule for the initial design of experiments (DoE) where N is the number of initial samples and p is the dimension of the random space [84]. They note that the rule leads to reasonable initial kriging models for "tractable" functions. For more difficult functions their results suggest increasing N yields little benefit in terms of increasing approximation

accuracy. Bichon recommended $\frac{(p+1)(p+2)}{2}$ initial samples for an initial DoE spanning $\pm 5\sigma$ of the random input variable space during use in the EGRA algorithm [18].

In the standard normal space the furthest limit state from the origin corresponding to a specified probability of failure p_f is the hypersphere of radius r_{max}:

$$r_{max} = \left((\chi_p^2)^{-1} \left(1 - p_f\right) \right)^{\frac{1}{2}}, \tag{3.2}$$

where $(\chi_p^2)^{-1}$ is the inverse cumulative chi-squared distribution function with p degrees of freedom and p is the dimension of the standard normal space. The derivation of (3.2) follows from the probability of failure estimator for integration via directional simulation from (2.26). Thus when the probability of failure is prescribed as $\bar{p}_f$, the furthest distance to search from the origin in the standard normal space for a point corresponding to the limit state level z leading to failure with probability $\bar{p}_f$ is r_{max}.

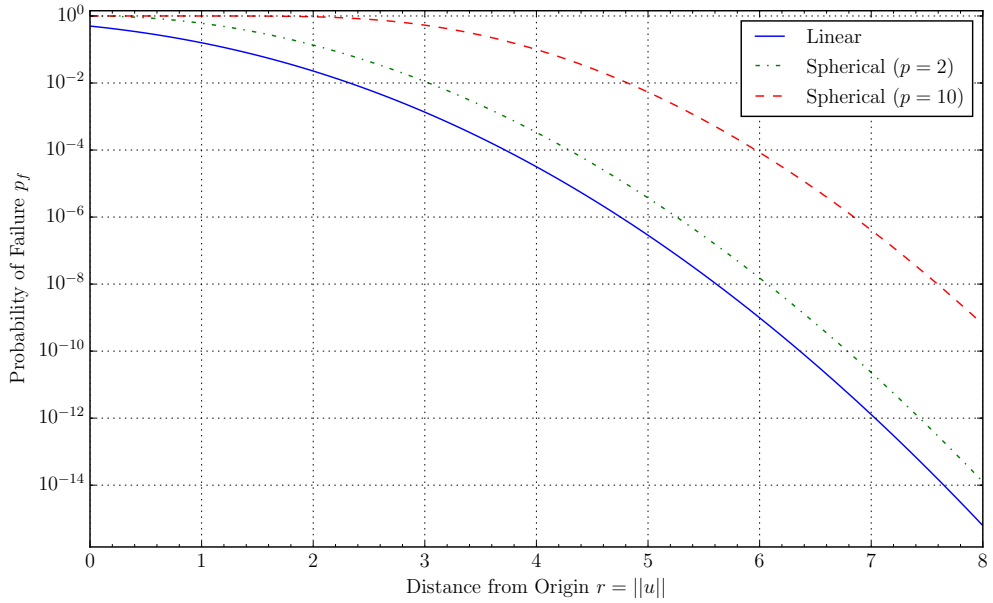


Figure 3.2: Comparison of the probabilities of failure for linear and spherical limit states in the standard normal space as a function of distance from origin $r = \|u\|$. The linear limit state distance corresponds to the case of the geometric reliability index β. Note the 100% chance of failure for the spherical limit states and the 50% chance of failure for a linear limit state when $r = 0$. Note also that a spherical limit state in higher dimensions (compare $p = 10$ to $p = 2$) induces a larger probability of failure for equivalent radius.

Figure 3.2 compares distances for several simple limit states. For example, in the standard normal space a linear limit state yielding probability of failure p_f is a distance $\beta = \Phi^{-1}(p_f)$ from

the origin, which follows from the definition of the reliability index β (see section 2.1.4). Thus for $p_f = 10^{-4}$, $\beta = 3.72$ which is less than $r_{max} = 4.29$ for the hyperspherical failure surface in two dimensions. When the number of dimensions increases to 20, $r_{max} = 7.24$.

The above discussion shows that for an inverse reliability analysis the samples used to construct the initial kriging model need only be space-filling within the p-sphere of radius r_{max}. A variety of techniques can be used to construct a set of space-filling samples. Traditional techniques such as Latin Hypercube Sampling (LHS) or Sobol' sequences can also be used if combined with acceptance-rejection sampling, *i.e.* samples that fall outside of the sphere of radius r_{max} can be rejected.

3.3 Model Refinement

Selecting the next training point(s) is a crucial component of an effective adaptive kriging-based reliability analysis. Typically the search for the training point(s) is formulated as an optimization problem, where the objective is a criterion measuring how "good" a candidate training point is. However, even if an effective criterion is chosen, the solution of the problem is typically non-trivial. Most search spaces are multimodal and highly nonlinear requiring the use of efficient global optimization techniques or sampling methods to estimate the optimum.

A good algorithm for selecting the next point(s) to refine the kriging model in a reliability analysis method will select points that have a large impact on the probability of failure estimate $\hat{p}_f$. Note that at any point u, the kriging model prediction at u has epistemic uncertainty $\hat{g}(u) \sim \mathcal{N}(m(u), s^2(u))$. Therefore the probability of a selected point u corresponding to a point that actually resides in the failure region (where $g(u) \leq \bar{z}$) can be estimated as

$$\begin{aligned} \pi(u) &= P\left[\hat{g}(u) \leq \bar{z}\right] \\ &= P\left[\frac{\hat{g}(u) - m(u)}{s(u)} \leq \frac{\bar{z} - m(u)}{s(u)}\right] \\ &= \Phi\left(\frac{\bar{z} - m(u)}{s(u)}\right), \end{aligned} \tag{3.3}$$

where $\Phi(\cdot)$ is the standard normal cumulative distribution function. $\pi(u)$ can also be interpreted as the probability of misclassification [44] and has also been labeled the *excursion probability* [28] and *probabilistic classification function* [43].

3.3.1 Existing Methods

Methods for adaptive refinement of kriging models can be roughly divided into two categories: point-based methods and global methods. In a point-based method, the decision to select a point for refinement is based on information available at that point only, namely the mean $m(u)$ and standard deviation $s(u)$ of the kriging model. In contrast, a global method assesses the impact of

selecting a point on nearby points and/or on a global quantity of interest. As a result global methods can more efficiently refine a kriging model, although they may be more expensive.

Point-based Methods

In a point-based method, information from the kriging model at a single point u, *i.e.* its mean $m(u)$ and standard deviation $s(u)$, is used to calculate a score for the point. The point with the best score is then chosen as the next training point. Three methods are discussed below.

In EGRA, maximization of the Expected Feasibility function $EF(u)$ is used to pick the next training point in a Gaussian process giving limit state level z [18], where

$$EF(u) \equiv \int_{z-\epsilon}^{z+\epsilon} (\epsilon - |g - z|)\, \hat{g}(g) dg, \tag{3.4}$$

where g is a realization of the normally distributed $\hat{g}(u)$. Bichon recommends taking $\epsilon = 2s(u)$ and terminating when the scaled maximum $EF(u)$ falls below a certain value, for example 10^{-3}. The Expected Feasibility function is inspired from contour estimation and quantifies how likely a point u yields a value $g(u)$ close to the desired limit state level z. (3.4) is expressible in simple analytic form in terms of the standard normal variate when a Gaussian Process is used (see the derivation in [18, p. 145]), and is therefore easy to calculate.

In AK-MCS, minimization of the learning function $U(u)$ is used to select the next training point [44], where

$$U(u) \equiv \frac{|m(u) - z|}{s(u)}, \tag{3.5}$$

where $U(u)$ has the interpretation of the distance measured in kriging standard deviations between the prediction and estimated limit state. As the integrand of the excursion probability $\pi(u)$ defined in (3.3), it is readily interpreted as a reliability index. The authors recommend terminating when $\min_u\{U(u)\} \geq 2$, *i.e.* when there is a 2.3% chance of making an error on the sign of $g(u)$.

In the EKSA method, the authors propose choosing the point with the maximum probability of error of its sign to be the next training point [72]. Although this is similar in spirit to the AK-MCS U function, it is complicated by EKSA's use of an SVD-based kriging model which requires the integral of a correlated multivariate Gaussian. In general, the authors propose integration via sampling. The authors propose terminating the algorithm when the maximum probability of sign error falls below 10^{-4}.

Global Methods

In spite of the simplicity of EGRA, AK-MCS, and similar point-based methods they are "short-sighted" since they only use local information from the posterior marginal distribution to select the next training point, as well as for assessing convergence. In other words, point-based methods

do not consider the effect of a training point choice on its neighborhood, and they also do not consider the convergence of the primary quantity of interest, the failure probability. As a result, several authors have pointed out this leads to inefficiencies as training points are selected that do not affect the reliability analysis, sometimes even after the probability of failure estimate has converged [71, 72, 26]. Two criteria which overcome this difficulty are reviewed here.

In the GSAS method, the authors propose using a global sensitivity analysis to estimate the sensitivity of the probability of failure estimate $\hat{p}_f$ with respect to candidate training points and choosing the point with the highest sensitivity [71]. Since performing the global sensitivity analysis is expensive, they first screen candidate points using the AK-MCS U-function and keep points that have a probability of sign error greater than 0.1%, which corresponds to $U(u) \leq 3.1$. They then pick the point with the highest sensitivity from the remaining candidate points.

Bect et al introduced the concept of Stepwise Uncertainty Reduction (SUR) for probability of failure estimation [16, 26, 28]. Such a strategy seeks to choose a point $u^{(s+1)}$ or points which minimize the average of a future measure of uncertainty $H_{s+1}(u)$ in a quantity of interest given the selection of those points. In particular, a SUR strategy solves the following problem for a given kriging model constructed from s samples:

$$\underset{u^{(s+1)}}{\text{minimize}} \quad E_s[H_{s+1}(u) \mid u^{(s+1)}]. \tag{3.6}$$

Since the outcome of the kriging model at $u^{(s+1)}$ is random, the measure of uncertainty $H_{s+1}(u)$ is also random. The conditional expectation given the $s+1$st sample $E_s[H_{s+1}(u) \mid u^{(s+1)}]$ averages the effect of choosing $u^{(s+1)}$ on the uncertainty measure.

One proposed SUR criterion is the J_s^α criterion which is defined as the average variance of the probability of failure estimate given the $s+1$st sample $u^{(s+1)}$ [26]:

$$J_s^\alpha(u^{(s+1)}) \equiv \mathrm{E}_s\left[\mathrm{Var}_{s+1}\left(\hat{p}_f\right) \mid u^{(s+1)}\right]. \tag{3.7}$$

The variation in the estimated excursion set $S \equiv \{u \in \mathbb{R}^p : \hat{g}(u) > z\}$, also called the safe set, can also be used to assess the accuracy of the probability of failure estimation. Note that the safe set S can also be written as in terms of its indicator function $S = \{u \in \mathbb{R}^p : \mathbb{1}_{\hat{g}(u)>z}\}$. The variance of the random variable $\mathbb{1}_{\hat{g}(u)>z}$ is simply $\pi(u)(1-\pi(u))$ where $\pi(u)$ is defined in (3.3). Therefore another criterion J_s^Γ can be defined as the average of the integrated variance of the indicator of the safe set given the $s+1$st sample $u^{(s+1)}$ [26]:

$$J_s^\Gamma(u^{(s+1)}) \equiv \mathrm{E}_s\left[\int \pi_{s+1}(u)(1-\pi_{s+1}(u))dP_u \mid u^{(s+1)}\right]. \tag{3.8}$$

In summary, the J_s^α criterion estimates how much the volume of the failure region will vary on average if u is selected as the $s+1$st point and the J_s^Γ criterion estimates how much the probability

of misclassification will vary on average if u is selected as the $s+1$st point. Although Chevalier et al derive efficient methods for calculating both the J_s^{α} and J_s^{Γ} SUR criteria [26], the calculation of SUR criteria is more expensive than point-based refinement criteria due to the need to perform nested integrations to calculate the criterion at each candidate point. However, when function evaluations are expensive the additional effort required to select an optimal next point may be well worth the expense. In fact, a SUR criterion is derived and used in the adaptive cokriging algorithm introduced later for the approximation of design sensitivities of the limit state (see section 3.6).

A similar optimal one-step lookahead method for adaptive refinement of a kriging model is the Robust Maximum Improvement for Level-set Estimation (RMILE) introduced by Zanette et al [137]. The RMILE method applied for probability of failure estimation seeks to choose a point $u^{(s+1)}$ that yields the maximum expected improvement in the volume of a sub-level set I_s defined by

$$I_s = \{u : \pi_s(u) > 1 - \delta\} \tag{3.9}$$

for a given level of confidence δ. The RMILE method solves the following problem for a given kriging model constructed from s samples:

$$\underset{u^{(s+1)}}{\text{maximize}} \quad E_s[|I_{s+1}| - |I_s| \mid u^{(s+1)}], \tag{3.10}$$

where $|\cdot|$ denotes the volume of the set. The authors ensure robustness for the problem in (3.10) by falling back on maximization of variance when all points yield sufficiently small expected improvements. They consider solving (3.10) via discretization on a grid and derive an analytic expression which can be used to efficiently solve (3.10) on a fixed grid.

Convergence Criteria

Termination criteria is important for any adaptive refinement method since additional function evaluations may be costly and lead to marginal, if any, improvements in the probability of failure estimate or the estimated limit state levels. Reliability analysis methods using a point-based refinement method tend to rely on a tolerance for the maximum or minimum value of the learning function [18, 44]. When the learning function value reaches the tolerance, refinement is stopped. However, since point-based refinement methods do not directly measure the impact on the quantity of interest, the probability of failure estimate $\hat{p}_f$, such a tolerance does not necessarily dictate convergence of the kriging model or $\hat{p}_f$. On the other hand, $\hat{p}_f$ may have already converged and subsequent iterations of refinement may be wasted effort. Below several termination criteria are described.

One suggested conservative termination criterion adopted by GSAS and EKSA is to terminate when the estimated maximum percent error in the failure probability ε_{max} is below a certain threshold. Assuming $\hat{p}_f$ is a Monte Carlo estimate, the samples are partitioned into two groups: those

where the chance of making an error on the sign of the shifted limit state $m(x) - z$ is low (group 1 samples) and the remaining samples (group 2 samples) [71, 72]. The maximum percent error is then given by

$$\varepsilon_{max} = \max_{N \in [0, N_2]} \left(\frac{|N_2 - N|}{N_1 + N} \times 100\% \right), \tag{3.11}$$

where N_1 is the number of samples in group 1 and N_2 is the number of samples in group 2 [71, p. 505].

GSAS considers a second termination criterion where $\hat{p}_f$ is considered to be a stochastic response with random multivariate input variables $\hat{g}(u) \sim \mathcal{N}(m(u), S(u))$ where $S(u)$ is the covariance matrix between samples in the kriging model [71]. The authors sample from the correlated random input variables and propagate the samples through a Monte Carlo estimator to build a histogram of $\hat{p}_f$ and the error in $\hat{p}_f$, which they then fit a distribution to and estimate 1% and 99% quantiles. The maximum of either quantile is taken as the maximum error. Since the additional sampling is expensive for large numbers of samples, they propose performing the sampling only for a subset of samples, those with U function value ≤ 3.1 (see section 3.3).

Dubourg et al consider a termination criterion based on the maximum range of the probability of failure estimate due to epistemic uncertainty in the kriging model which is similar to the method used in AKIRA and described in the following section [42].

3.3.2 A New Heuristic for Refinement

Although SUR-based refinement criteria are optimal, they incur additional computational cost. In many applications, it may be preferable or advantageous to use a refinement criterion that is sub-optimal but faster to compute. The expected feasibility function of EGRA and the U function of AK-MCS are both candidates, but they suffer from several shortcomings [18, 16, 26, 71] including the selection of points that have little to no impact on the probability of failure estimate $\hat{p}_f$ as well as the continued selection of points even after $\hat{p}_f$ is converged. In addition, AK-MCS and other methods sometimes lead to the selection of points that are very close in space, when points that are better spaced can lead to a better-conditioned kriging model and $\hat{p}_f$.

An effective heuristic for model refinement during reliability analysis selects points that both have a large impact on the probability of failure estimate $\hat{p}_f$ and reduce the epistemic uncertainty of the kriging model in regions contributing greatly to $\hat{p}_f$. Here a new point-based heuristic for adaptive refinement of kriging models for reliability analysis is introduced. It addresses both desired criteria and is based on ideas from directional simulation (see section 2.2.3). First an appropriate convergence criterion for assessing when the kriging model is sufficiently converged is discussed.

Termination Criterion

Kriging model refinement should be stopped when the kriging model is sufficiently accurate that it provides a consistent estimate of the probability of failure. Note however that this does not imply the probability of failure estimate $\hat{p}_f$ is accurate. If the kriging model does not represent the original limit state due to limit state discontinuities, improper choice of kriging covariance function, insufficient initial random sampling or other issues, $\hat{p}_f$ may be inaccurate. Nevertheless, we define a criterion for termination based on the decrease in epistemic uncertainty of the kriging model with regard to its effect on the probability of failure estimate $\hat{p}_f$.

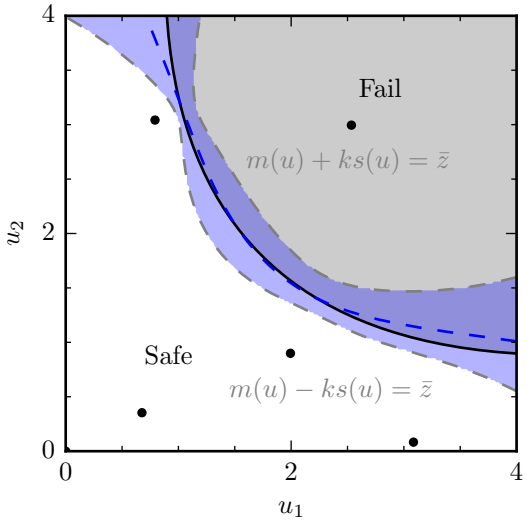


Figure 3.3: Upper and lower empirical 95% confidence bounds ($k = 1.96$) on a kriging model with mean $m(u)$ and standard deviation $s(u)$ are shown in the standard normal space in $\mathbb{R}^2$. The true limit state is shown in the solid black line. The probability of failure can be estimated with respect to the mean contour and each bound. If the kriging model is accurate in the areas of the limit state that contribute to the probability of failure then each estimated probability of failure will be close in value. Such a metric is encoded in the probability convergence ratio π_f given in (3.14).

Let the following failure subsets F^{-1}, F^0, and F^{+1} be defined similarly to [42] as

$$F^i = \{u \in \Omega \mid m(u) + iz_{\alpha/2}s(u) \leq z\}, \tag{3.12}$$

where $i \in \{-1, 0, +1\}$ and $z_{\alpha/2} = \Phi^{-1}(0.95) \approx 1.65$ or similar, in the case of a one-sided 95% confidence interval on the kriging estimate at point u. Figure 3.3 shows the bounds for a particular kriging approximation to a convex limit state in $\mathbb{R}^2$. Note that $F^{+1} \subseteq F^0 \subseteq F^{-1}$ since the kriging standard deviation $s(u) \geq 0$. The following relationship exists between the probabilities of failure

measured with respect to each failure subset:

$$P[u \in F^{+1}] \leq P[u \in F^{0}] \leq P[u \in F^{-1}], \tag{3.13a}$$

$$\hat{p}_f^{+} \leq \hat{p}_f \leq \hat{p}_f^{-}, \tag{3.13b}$$

where a simpler notation $\hat{p}_f^{+}$, etc. is introduced and the previously used $\hat{p}_f \equiv P[u \in F^{0}]$, *i.e.* the probability of failure estimate when the kriging mean is taken as the limit state. Then the following probability of failure convergence ratio π_f may be used to estimate termination of refinement:

$$\pi_f = \frac{\max\left(\hat{p}_f - \hat{p}_f^{+}, \hat{p}_f^{-} - \hat{p}_f\right)}{\hat{p}_f} \leq \tau_p. \tag{3.14}$$

When $\hat{p}_f = 0$, the termination criterion is undefined but causes no issues when used within an inverse reliability analysis since the sought $\hat{p}_f$ will be non-zero. In such cases, the termination criterion indicates that the kriging model has not converged. The tolerance τ_p can be selected to be around 0.1 or so. The selection of τ_p depends not only on the variance of the probability of failure estimate but also on whether the kriging covariance kernel has a nugget, *i.e.* a known noise term σ^2 added to its diagonal, in which case $s(u)$ is non-zero at kriging training points. Both will increase the value of the refinement criterion. To address variance in $\hat{p}_f$, previous authors have recommended using the 95% confidence bound of $\hat{p}_f^{+}$ and $\hat{p}_f^{-}$ instead of their mean values [42].

Single-point Refinement Criterion

The termination criterion given by the probability of failure convergence ratio π_f in (3.14) motivates a refinement criterion that addresses the difference between the probability of failure estimate using the mean of the kriging model and the estimates using a lower and upper confidence interval for the kriging model. Such a criterion indirectly addresses variance in the kriging model ($\pi_f = 0$ implies zero variance in the kriging model at all points relevant to the probability of failure calculation) and at the same time directly addresses variability in the quantity of interest, the probability of failure estimate $\hat{p}_f$.

While the termination criterion $\pi_f \leq \tau_p$ is easily calculated, the determination of an appropriate refinement criterion is less obvious, particularly with regard to determining a preference for selecting one point over another. SUR methods address the selection issue by forecasting which points will contribute most to the reduction in uncertainty but they are more expensive.

Ideas from the technique of directional simulation prove useful here (see section 2.2.3). Recall in directional simulation that a sampling of random directions is made in the standard normal space and a search performed along each direction from the origin to determine the first point of failure, given that direction. The estimator of the probability of failure $\hat{p}_f$ is approximated as a sum of conditional probabilities, which are easily calculated in the standard normal space knowing the

distance r from the origin to the limit state surface as previously shown:

$$\begin{aligned}\hat{p}_f(x) &= \frac{1}{N}\sum_{i=1}^{N} P[g(x, Ra^{(i)}) \leq z \mid a^{(i)}] \\ &= \frac{1}{N}\sum_{i=1}^{N} 1 - \chi_n^2((r^{(i)})^2),\end{aligned} \tag{2.26}$$

where $r^{(i)}$ is the distance from the origin to the limit state surface along the ith sampled direction.

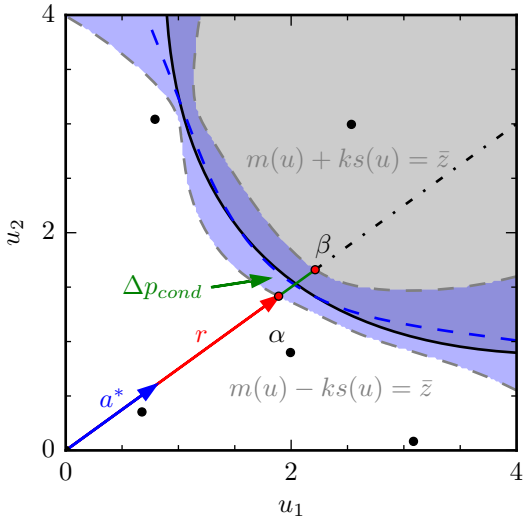


Figure 3.4: Upper and lower empirical 95% confidence bounds ($k = 1.96$) on a kriging model with mean $m(u)$ and standard deviation $s(u)$ are shown in the standard normal space in $\mathbb{R}^2$. The true limit state is shown in the solid black line. To select a new point that will increase the kriging model accuracy in areas contributing to the probability of failure, directions are first sampled in the random space and the conditional probability discrepancy Δp_{cond} between the upper and lower kriging model bounds is calculated. In the plot above, Δp_{cond} corresponds to the integrated conditional probability content between points α and β. The direction $a^{(i)}$ with the largest Δp_{cond} is selected for a linesearch from which a point will be chosen to be added to the kriging model.

Consider performing directional simulation for the estimation of the probability of failure convergence ratio π_f. Three simulations must be performed, one using the limit state model $m(u) - z_{\alpha/2}s(u)$, one using $m(u)$ and one using $m(u) + z_{\alpha/2}s(u)$. In such a situation one could consider using the same

set of directions $a^{(i)}$ for $i = 1 \dots N$, *i.e.* common random numbers. Then, $\hat{p}_f^- - \hat{p}_f^+$ is given by

$$\begin{aligned} \hat{p}_f^- - \hat{p}_f^+ &= \frac{1}{N} \sum_{i=1}^{N} \left(\chi_n^2((r_+^{(i)})^2) - \chi_n^2((r_-^{(i)})^2) \right) \\ &= \frac{1}{N} \sum_{i=1}^{N} \Delta p_{cond}^{(i)}, \end{aligned} \tag{3.15}$$

where $\Delta p_{cond}^{(i)}$ is the conditional probability content along direction $a^{(i)}$ between $r_-^{(i)}$ and $r_+^{(i)}$ denoted as points α and β in Figure 3.4. $\Delta p_{cond}^{(i)}$ represents the discrepancy in failure probability due to the kriging model's epistemic uncertainty given direction $a^{(i)}$. A large value of $\Delta p_{cond}^{(i)}$ implies the ith direction is one where the kriging model has a (relatively) large epistemic uncertainty and may have a (relatively) large effect on $\hat{p}_f$. Note that controlling the values of $\Delta p_{cond}^{(i)}$ directly influences the convergence of π_f. As a result, a naive algorithm for ensuring convergence with respect to π_f is to simply select the point $r_-^{(i)}$ or $r_+^{(i)}$ that corresponds to the maximum $\Delta p_{cond}^{(i)}$.

However such a naive approach can be slow to converge and also susceptible to numerical instabilities once many points which are close in distance and/or value have been used to train the kriging model. In addition, the refinement of the kriging model can benefit from the selection of a variety of intermediate points lying on the line segment between $r_+^{(i)}$ and $r_-^{(i)}$. As such, we propose first selecting the direction $a^{(i)}$ whose conditional probability discrepancy content is greatest and then choosing a point along that direction according to its proximity to other points already in the trained model and its contribution to the probability of failure.

First, select the direction whose probability discrepancy content is greatest as

$$\underset{a^{(i)}}{\text{argmax}} \quad \Delta p_{cond}^{(i)}. \tag{3.16}$$

Let r_+ and r_- denote the corresponding endpoints of the line segment along direction $a^{(i)}$. Any point between r_+ and r_- can be written as

$$r(\alpha) = \alpha r_+ + (1 - \alpha) r_- \tag{3.17}$$

for $\alpha \in [0, 1]$. Then perform the univariate and potentially multimodal optimization for α where $u^{(i)}(\alpha) = r(\alpha) a^{(i)}$ and $u^{(j)}$ is the jth training point in the kriging model:

$$\begin{aligned} \underset{\alpha}{\text{maximize}} \quad & w(\alpha) \text{min}_j \left\{ \text{dist} \left(u^{(i)}(\alpha), u^{(j)} \right) \right\} \\ \text{s.t.} \quad & 0 \leq \alpha \leq 1, \end{aligned} \tag{3.18}$$

where the maximization objective is the minimum distance between the prospective point $u^{(i)}$ and each training point in the kriging model, weighted by $w(\alpha)$. Such an objective ensures that training

points are well spaced. The weight $w(\alpha)$ is taken to be $w(\alpha) = p_{cond}(r(\alpha)) = 1 - \chi_p^2(r^2(\alpha))$, *i.e.* the conditional probability of failure of $u^{(i)}$ given direction $a^{(i)}$. Its purpose is to ensure points are not selected in regions that do not contribute much to the probability of failure. The minimization in (3.18) is most easily and robustly carried out via evaluation of the objective on a grid spanning $\alpha = 0$ to $\alpha = 1$.

Multi-point Refinement Criterion

The selection of multiple points for adaptive refinement of the kriging model can be advantageous when limit state evaluations can be run in parallel. Although a multiple point selection strategy may increase the total number of limit state evaluations, the number of iterations of the adaptive refinement procedure can decrease thereby yielding a performance benefit overall in terms of time to convergence. In addition, there is rarely a single best point for selection and addition to the kriging model; typically many points could be selected that would yield similar improvements in the kriging model over the course of the adaptive refinement process.

A variety of multiple point adaptive refinement techniques have been explored including those based on k-means clustering [42] and SUR criteria [26]. Here an approach similar to the heuristic optimization discussed in [28] is used. Suppose N points are to be selected at each iteration for model refinement according to the available computational resources. Instead of solving a multivariate optimization problem for the N currently unknown choices, we solve a suboptimal sequence of univariate optimization problems by selecting a point using the single point refinement criterion, updating the kriging model using an estimate of the selected point, and then selecting the next point etc. until N points have been selected. The multipoint selection algorithm for adaptive refinement is outlined in Algorithm 2.

When a single point $u^{(i)}$ is selected, the kriging model provides an estimate of its value $g(u^{(i)})$ as a distribution with a mean $m(u^{(i)})$ and variance $s^2(u^{(i)})$. In SUR-based criteria (see section 3.3.1), this uncertainty is taken into account when selecting the next point $u^{(i+1)}$ in a multipoint criteria. However, in Algorithm 2 this uncertainty is only partially taken into account. Instead, the next point $u^{(i+1)}$ is selected assuming the previous point's value (which is currently unknown) is in fact equal to its mean, *i.e.* $g(u^{(i)}) = m(u^{(i)})$.

In fact, the updated kriging variance given a newly selected point $u^{(i)}$ is independent of the response value $g(u^{(i)})$ at $u^{(i)}$ (barring reoptimization of kernel hyperparameters) and only the updated kriging mean is dependent on the response value [27]. Thus, if the kriging mean is indeed equal to the response value at $u^{(i)}$, then no additional value is added to the kriging mean prediction. Inspection of Algorithm 2 shows that refinement is driven by the deviation of the kriging model from its mean; as such the multiple point selection under the predictive mean assumption above is reasonably expected to yield benefits over a single-point criterion.

Algorithm 2: Kriging adaptive refinement with new heuristic

Data: kriging models of limit states $\hat{g}_i(u)$ and limit state levels z_i for $i = 1, \ldots, m$; set of directions $a^{(j)}$ for $j = 1, \ldots, N$ from directional simulation; number of points t to select

Result: t next points $u^{(k)}$, $k = 1, \ldots, t$, for kriging model refinement

for $i \leftarrow 1$ **to** m **do**
  $\tilde{g}_i(u) \leftarrow$ Save an alternate copy of the kriging model $\hat{g}_i(u)$;
for $k \leftarrow 1$ **to** t **do**
  for $j \leftarrow 1$ **to** N **do**
    $r_0 \leftarrow$ Perform directional simulation linesearch for direction $a^{(j)}$ using $\tilde{m}_i(u)$;
    $r_- \leftarrow$ Perform linesearch for direction $a^{(j)}$ using $\tilde{m}_i(u) - z_{\alpha/2}\tilde{s}_i(u)$;
    $r_+ \leftarrow$ Perform linesearch for direction $a^{(j)}$ using $\tilde{m}_i(u) + z_{\alpha/2}\tilde{s}_i(u)$;
    $\Delta p^{(j)}_{cond} \leftarrow$ Calculate conditional probability discrepancy from (3.15);
  $a^* \leftarrow$ Determine most important direction: $\text{argmax}_{a^{(j)}}\{\Delta p^{(j)}_{cond}\}$;
  $u^{(j)} \leftarrow$ Perform linesearch along direction a^* ((3.18));
  for $i \leftarrow 1$ **to** m **do**
    $\tilde{g}_i(u) \leftarrow$ Update kriging model with mean value of $u^{(j)}$;
return $u^{(k)}$ *for* $k = 1 \ldots t$;

3.3.3 Extension to Multiple Limit States

Multiple limit states are easily accommodated using Algorithm 2 since it generally relies upon the technique of directional simulation for adaptive refinement and directional simulation is readily applied to series, parallel, or mixed systems (see chapter 2 and section 2.2.3 in particular).

3.3.4 Demonstration

Comparison to Other Refinement Criterion

The new adaptive refinement heuristic proposed in this section is compared with Monte Carlo and the Efficient Global Reliability Analysis (EGRA) method [19] for the estimation of probability of failure on three structural reliability test problems: the short column problem [19] with $y \in \mathbb{R}^4$, the steel column problem [77] with $y \in \mathbb{R}^9$, and the cantilever beam problem [132] with $y \in \mathbb{R}^4$. Each problem is described in detail in Appendix A. For each problem the new heuristic is either more efficient or more accurate than EGRA in estimating the probability of failure.

Table 3.1: The adaptive refinement heuristic for kriging models proposed in this section is compared with Monte Carlo and EGRA [19] for three structural reliability test problems. In EGRA, a probability of failure is estimated from the kriging model using multimodal adaptive importance sampling. The new heuristic is used to refine a kriging model from which probability of failure is estimated using directional simulation. The average and coefficient of variation (in parentheses) of probability of failure estimate $\hat{p}_f$ and number of simulations N is shown for each method. Both EGRA and the new heuristic are much more efficient than Monte Carlo. Furthermore the new heuristic yields more accurate kriging models which generally require less function evaluations than EGRA for probability of failure estimation.

Problem	Method	z	$\hat{p}_f$	N
Short Column	Monte Carlo	-0.3796	10^{-6} (0.1)	10^8
	EGRA		1.1965×10^{-6} (0.01)	51.0 (0.12)
	New Heuristic		0.9737×10^{-6} (0.09)	66.6 (0.18)
Steel Column	Monte Carlo	-0.2562	10^{-6}(0.1)	10^8
	EGRA		1.2136×10^{-6} (0.13)	163.4 (0.17)
	New Heuristic		1.0494×10^{-6} (0.21)	70.8 (0.06)
Cantilever	Monte Carlo	$[-0.18, -0.17]^\top$	1.01×10^{-6}	10^8
	EGRA		1.1658×10^{-6} (0.22)	66.7 (0.15)
	New Heuristic		1.0680×10^{-6} (0.07)	30.1 (0.09)

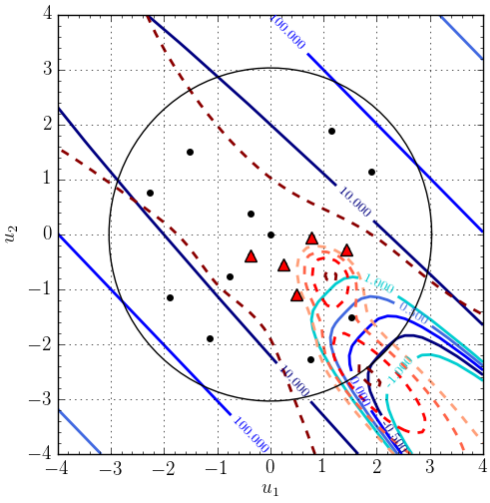
Comparison of Single-point and Multi-point Refinement

Figures 3.5 and 3.6 compare the single-point and multi-point refinement algorithm as embedded in the AKIRA algorithm for the highly nonlinear limit state problem described in Appendix A.5. In this particular case, the AKIRA algorithm requires 75 iterations for convergence using the single-point adaptive refinement strategy but only 17 iterations for convergence using the multi-point strategy which results in a reduction of approximately 4 in serial computation time when limit state evaluations are run in parallel. The multi-point refinement strategy requires 96 function evaluations total compared to 86 for the single-point criterion. Both refinement methods converge to the same estimated limit state levels with 5 digits of accuracy.

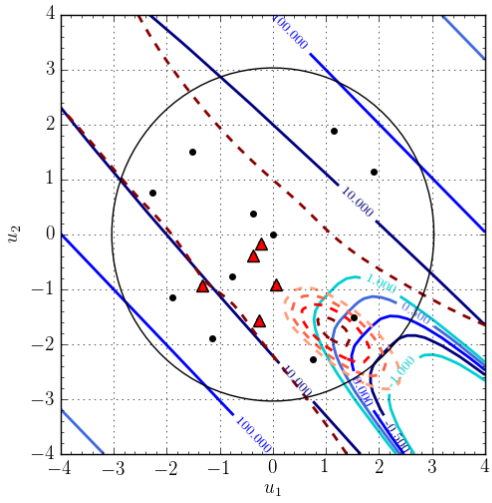
3.4 Solving the Inverse Reliability Analysis

When only one limit state function is present, obtaining the limit state level $z \in \mathbb{R}$ that results in a prescribed probability of failure $\bar{p}_f$ is a simple 1D optimization problem with a non-decreasing function $p_f(z)$ (see Figure 3.7). In this case standard rootfinding methods methods can be used to efficiently determine the unique solution z^*.

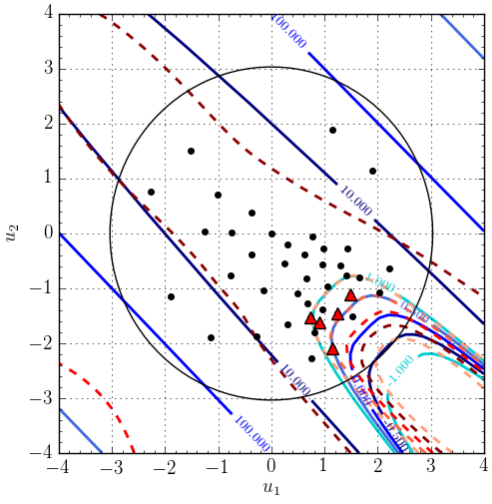
When multiple limit state functions are present in a system reliability analysis, the problem becomes more complicated since multiple sets of limit state levels can result in the same probability of failure p_f, *i.e.* the solution of the inverse problem is not unique. Previous inverse reliability



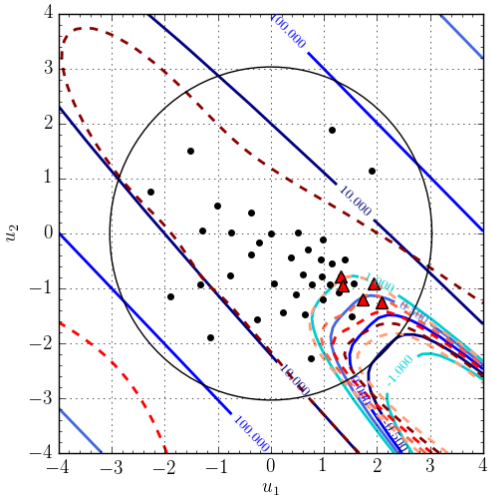
(a) Single-point, $k = 5$



(b) Multi-point, $k = 1$

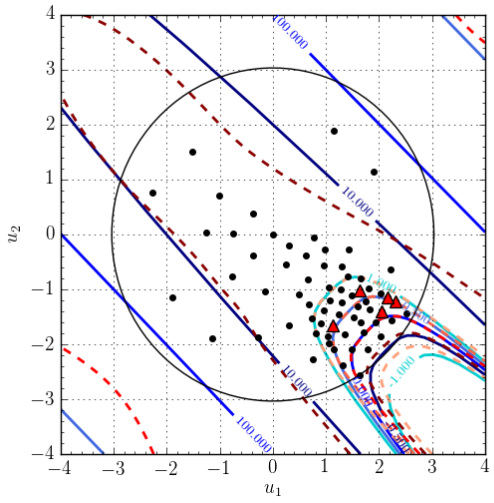


(c) Single-point, $k = 30$

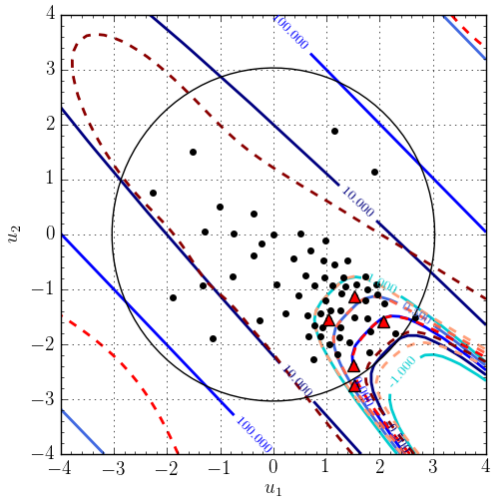


(d) Multi-point, $k = 6$

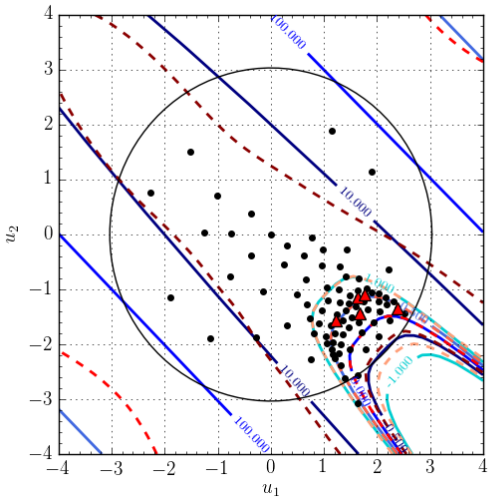
Figure 3.5: Single-point vs. multi-point adaptive kriging refinement in AKIRA for the highly non-linear limit state problem (section A.5). The kriging approximation is shown in dashed lines and the exact limit state contours are shown in solid lines. Five points are added per iteration for the multi-point criterion; the most recent training points for both the single-point and multi-point refinement strategies are shown in red triangles; older training points are shown as black dots. Iterations 1 and 6 of the AKIRA algorithm using the multi-point adaptive refinement heuristic are compared with iterations 5 and 30 of the AKIRA algorithm using the single-point adaptive refinement heuristic. Note the slightly better spaced training points using the single-point refinement criteria, but overall similar refinement. AKIRA with multi-point refinement requires much fewer iterations for convergence.



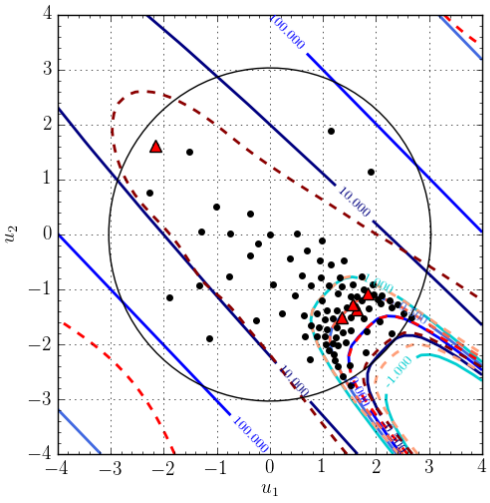
(a) Single-point, $k = 55$



(b) Multi-point, $k = 11$

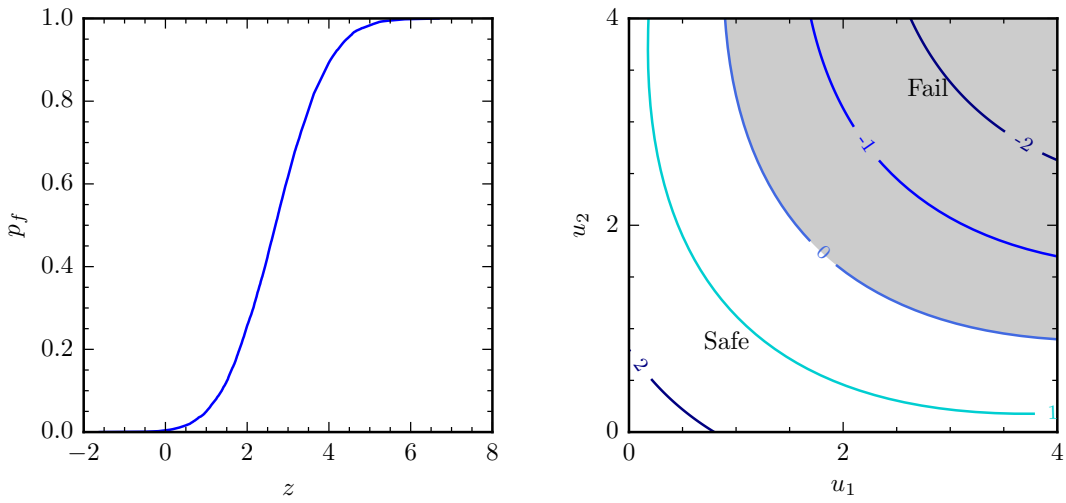


(c) Single-point, $k = 75$



(d) Multi-point, $k = 16$

Figure 3.6: Single-point vs. multi-point adaptive kriging refinement in AKIRA for the highly nonlinear limit state problem (section A.5). The kriging approximation is shown in dashed lines and the exact limit state contours are shown in solid lines. Five points are added per iteration for the multi-point criterion; the most recent training points for both the single-point and multi-point refinement strategies are shown in red triangles; older training points are shown as black dots. Iterations 11 and 16 of the AKIRA algorithm using the multi-point adaptive refinement heuristic are compared with iterations 55 and final iteration 75 of the AKIRA algorithm using the single-point adaptive refinement heuristic. Note the slightly better spaced training points using the single-point refinement criteria, but overall similar refinement. AKIRA with multi-point refinement requires much fewer iterations for convergence.



(a) Root-finding using the CDF.

(b) Corresponding limit state level sets.

Figure 3.7: Determination of a single limit state level $z \in \mathbb{R}$ for a prescribed probability of failure $\bar{p}_f$ is a simple problem due to the non-decreasing cumulative distribution function.

analysis methods for systems have enforced a unique solution by requiring each limit state level to be described by different univariate functions of the same shared variable. For example, Li proposed letting reliability indices $\beta_1 = k_1\beta, \ldots, \beta_m = k_m\beta$, where $k_i > 0$ is interpreted as a ratio of each reliability index β_i to β (this is the inverse problem formulated in terms of reliability index β as opposed to limit state level z) [80]. Then only a single unique β need be determined during a rootfinding procedure. Of course the solution of the inverse problem depends on the prescribed ratios k_i.

For use in reliability-based design optimization, the chosen set of limit state levels $z \in \mathbb{R}^m$ should be representative of the true limit state levels $\bar{z} \in \mathbb{R}^m$ in the sense that the most critical limit states for the system should be reflected in the chosen limit state levels z. As such, we desire an ideally unique set of limit state levels z that are both close to the true limit state levels $\bar{z}$ and correspond to a prescribed probability of failure $\bar{p}_f$. One way of writing the problem is as a least squares problem with a nonlinear equality constraint:

$$\begin{aligned} \underset{z}{\text{minimize}} \quad & \phi(z) = \frac{1}{2}\left\|z - \bar{z}\right\|_2^2 \\ \text{s.t.} \quad & \hat{p}_f(z) = \bar{p}_f, \end{aligned} \tag{3.1}$$

where $\hat{p}_f(z)$ is the probability of failure estimate obtained from the kriging models of the limit states $\hat{g}(u)$. Assuming that the limit state kriging models are well-scaled, a scaled version of the

optimization problem is

$$\begin{aligned} \underset{z}{\text{minimize}} \quad & \phi(z) = \frac{1}{2} \|z - \bar{z}\|_2^2 \\ \text{s.t.} \quad & c(z) = \frac{\hat{p}_f(z)}{\bar{p}_f} - 1 = 0. \end{aligned} \tag{3.19}$$

Unfortunately, there is no guarantee of uniqueness of the optimal solution z^* from solving (3.19) due to the nonconvex equality constraint $c(z)$. A potentially redeeming characteristic of $\hat{p}_f(z)$ and therefore $c(z)$ is that they are nonincreasing along each coordinate z_i, however there is no such similar property for an arbitrary z.

Nevertheless, (3.19) has several nice properties, namely 1) it is formulated as a least squares problem for which efficient methods are available for solution (see section 3.4.1) and 2) the optimal solution z^* is reliable in the sense that $z_i^* < \bar{z}_i$ only when $z_j^* \leq \bar{z}_j$ for all $j \neq i$. In other words z^* is *consistent* with $\bar{z}$ such that the estimated limit state level z_i^* only predicts failure when all other (active) levels also predict failure. Empirical evidence for this property can be found in section 3.7.3. A short proof follows for the second property:

Proof. Let $z^* = [z_1^*, \ldots, z_m^*]^\top$ be the optimal solution of (3.19) and $c(z)$ be defined as in (3.19) with continuous partial derivatives. Assume all limit state levels are active, that is they contribute to the probability of failure. Now let $z_1^* < \bar{z}_1$ and all $z_j^* > \bar{z}_i$ for $i = 2, \ldots, m$. Consider limit state levels z_1 and z_i for $i \neq 1$. Since $c(z)$ is nonincreasing and continuous in each z_i, there exist two small quantities δ_1 and δ_i where $0 < \delta_1 < \bar{z}_1 - z_1^*$ and $\bar{z}_i - z_i^* < \delta_i < 0$ such that $c(z^* + \delta_1 e_1 + \delta_i e_i) = 0$, *i.e.* feasibility is maintained. $e_i = [0, \ldots, 1, \ldots, 0]^\top$ is the m-vector of zeros with 1 in the ith place. It follows that $\phi(z^* + \delta_1 e_1 + \delta_i e_i) < \phi(z^*)$ and therefore z^* is not the optimal solution of (3.19). □

3.4.1 Solving the Optimization for Limit State Levels

The presence of the nonlinear equality constraint $c(z) = 0$ unfortunately prevents the problem from being convex for general $\hat{p}_f$, which otherwise would guarantee a unique global solution. Nevertheless a solution can be obtained relatively efficiently using a general nonlinear equality-constrained optimization algorithm, such as sequential quadratic programming (see Appendix C.4) or augmented Lagrangian methods. Since the gradient and Hessian of both the objective $f(z)$ and constraint $c(z)$ are available (details provided in section 3.4.2), efficient Newton-type methods can be used.

On the other hand, (3.19) can be reformulated by exploiting the fact that it is an equality-constrained least squares problem and rewriting it as an unconstrained least squares problem, thereby allowing the use of a potentially more efficient algorithm for least squares. De Meersman has previously described a procedure for reformulating a least squares problem with nonlinear equality constraints as an unconstrained least squares problem [34]. The key idea is to rewrite an *exact penalty function* for the original constrained least squares problem as an equivalent sum of squares, thereby enabling solution as an unconstrained least squares problem. The method for doing so is

to introduce a set of new variables and some linear mappings and massage the equations until an unconstrained least squares problem can be written.

Both an augmented Lagrangian approach for solving (3.19) and reformulating and solving it as an unconstrained least squares problem require the specification of a penalty ρ. In both cases, a sufficiently large penalty ρ will give the optimal solution to the original constrained least squares problem in (3.19) for the limit state levels. As in standard augmented Lagrangian or penalty methods, an initial guess is made for ρ and the penalty is increased in subsequent iterations if the constraint violation exceeds a certain tolerance. As the AKIRA algorithm is iterative, at each iteration the probability of failure constraint feasibility is checked and the penalty updated if necessary. An initial penalty value of 100 and constant multiplicative update factor of 10 when the equality constraint is violated appear to work well in practice for a variety of problems. The use of the penalty is reflected in the AKIRA algorithm shown in Algorithm 1.

3.4.2 Efficient Gradient Calculation

The efficient solution of (3.19) requires the gradients of the probability of failure estimate $\hat{p}_f$ with respect to the levels z. The sensitivities from section 2.6 can be adapted; for example for a single limit state $g(x,u) : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}$, the gradient $\nabla_z p_f$ can be written as a surface integral:

$$\nabla_z p_f(x) = -\int_{g(x,u)=z} \frac{f_u(u)}{\|\nabla_u g(x,u)\|} \mathrm{d}S \tag{3.20}$$

since $\nabla_z g(x,u) = \nabla_z z = 1$. Of course, the integral in (3.20) can also be written as a volume integral or combination of volume and surface integrals in accordance with Uryas'ev's theorem (see section 2.6).

Using Directional Simulation

In certain cases, it may be easier to directly differentiate the estimator as in the case when directional simulation is used to estimate the probability of failure (see section 2.2.3).

First, the gradient calculation is generalized to multiple limit states using the assumptions from section 2.6.2, namely, let $p_f(z)$ be separable and represented as

$$p_f(z) = p_1(z_1) + \cdots + p_m(z_m). \tag{2.66}$$

In other words, each limit state is responsible for an independent contribution $p_i(z_i)$ to the total failure probability p_f, which is the case when using directional simulation to find single roots from

which to estimate p_f. The derivatives of the estimator are then obtained as

$$\begin{aligned}\frac{\partial \hat{p}_f}{\partial z_j} = \frac{\mathrm{d}\hat{p}_j}{\mathrm{d}z_j} &= \frac{\mathrm{d}}{\mathrm{d}z_j}\left[\frac{1}{N}\sum_{i=1}^{N}\mathbb{1}_{ij}\left(1-\chi_p^2(r_i^2)\right)\right] \\ &= -\frac{2}{N}\sum_{i=1}^{N}\mathbb{1}_{ij}\chi_p^{2\prime}(r_i^2)r_i\frac{\mathrm{d}r_i}{\mathrm{d}z_j},\end{aligned} \tag{3.21}$$

where there are N simulated directions for $i = 1, \ldots, N$, each with a corresponding r_i measuring the distance to the nearest failure surface. $\mathbb{1}_{ij}$ is 1 when the ith direction corresponds to failure for the jth limit state, and 0 otherwise. $\chi_p^{2\prime}(r_i^2)$ is the first derivative of the chi-squared cumulative distribution function with p degrees of freedom evaluated at r_i^2. Note again that ${}^{\partial r_i}/_{\partial z_j} = 0$ when the ith distance to the failure surface r_i does not correspond to the jth limit state.

${}^{\mathrm{d}r_i}/_{\mathrm{d}z_j}$ is the reciprocal of the directional derivative of $z_j = g_j(x, u)$ in the radial direction given by r_i:

$$\frac{\mathrm{d}r_i}{\mathrm{d}z_j} = \left(\frac{\mathrm{d}z_j}{\mathrm{d}r_i}\right)^{-1} = \left(\nabla_u z_j^\top \frac{u}{r_i}\right)^{-1}, \tag{3.22}$$

which requires the gradient of the limit state $\nabla_u g_j(x, u)$ to be available.

Efficient optimization methods also make use of second derivative information. The Hessian of the directional simulation estimator $\hat{p}_f$ is derived as

$$\frac{\partial^2 \hat{p}_f}{\partial z_j \partial z_k} = \frac{-2}{N}\sum_{i=1}^{N}\mathbb{1}_{ij}\left(\chi_p^{2\prime\prime}2r_i^2\frac{\partial r_i}{\partial z_j}\frac{\partial r_i}{\partial z_k} + \chi_p^{2\prime}\frac{\partial r_i}{\partial z_j}\frac{\partial r_i}{\partial z_k} + \chi_p^{2\prime}r_i\frac{\partial^2 r_i}{\partial z_j \partial z_k}\right), \tag{3.23}$$

where the first and second derivative of the chi-squared cumulative distribution function with p degrees of freedom $\chi_p^{2\prime}$ and $\chi_p^{2\prime\prime}$ are evaluated at r_i^2. Note again that since ${}^{\partial r_i}/_{\partial z_j} = 0$, the ith distance to the failure surface r_i does not correspond to the jth limit state level z_j resulting in a diagonal Hessian:

$$\begin{aligned}\frac{\partial^2 \hat{p}_f}{\partial z_j^2} &= \frac{-2}{N}\sum_{i=1}^{N}\mathbb{1}_{ij}\left(\chi_p^{2\prime\prime}2r_i^2\left(\frac{\mathrm{d}r_i}{\mathrm{d}z_j}\right)^2 + \chi_p^{2\prime}\left(\frac{\mathrm{d}r_i}{\mathrm{d}z_j}\right)^2 + \chi_p^{2\prime}r_i\frac{\mathrm{d}^2 r_i}{\mathrm{d}z_j^2}\right) \quad j = k \\ \frac{\partial^2 \hat{p}_f}{\partial z_j \partial z_k} &= 0. \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad j \neq k\end{aligned} \tag{3.24}$$

3.5 Design Sensitivities

Design sensitivities were introduced in section 2.6 for both the forward and inverse reliability analysis. The primary sensitivity of interest for design is $\nabla_x z$, or the derivative of the quantile function $z(x)$ with respect to the set of deterministic design variables x. As a reminder, the quantile sensitivity is

the $m \times n$ matrix whose ijth entry is

$$\frac{\partial z_i}{\partial x_j} = \left(\frac{\mathrm{d}p_i}{\mathrm{d}z_i}\right)^{-1} |_{z=p_f(x_0)} \frac{\partial p_f}{\partial x_j} |_{x=x_0}, \tag{2.69}$$

where the assumption was made that p_f can be decomposed as a sum $p_f = \sum_i^m p_i(z_i)$. In other words the probability of failure can be decomposed into a sum of contributions from each limit state.

3.5.1 Calculation Using Directional Simulation

When directional simulation is used as the technique for integrating the probability of failure p_f (see section 2.2.3), the sensitivities in (2.69) can be obtained fairly easily. Two methods are described below; method I is based on (2.69) and method II is a more robust approximation method.

In both methods only the design variable sensitivity $\nabla_x g(x,u)$ is potentially difficult to obtain, especially as it must be evaluated at each of N points on the limit state surface in the standard normal space $u^{(i)}$ for $i = 1, \ldots, N$, albeit for a fixed x. The following section describes an adaptive cokriging approach that can be used to aid in the estimation of $\nabla_x g(x,u)$ when gradient estimation is expensive.

Method I

(2.69) may be used for gradient estimation. The first component was derived in (3.21) and will not be discussed further here. The second component ($\partial p_f / \partial x_j$) is the sensitivity for the forward reliability analysis which was derived in section 2.6.1 for the case of directional simulation. As a reminder the sensitivity when directional simulation is used is

$$\nabla_x p_f(x) \approx -\frac{1}{N} \sum_{i=1}^{N} \frac{f_u(u^{(i)}) \cdot \nabla_x g(x, u^{(i)}) \cdot S_{p-1}(0, r^{(i)}) \cdot r^{(i)}}{u^{(i)\top} \nabla_u g(x, u^{(i)})}, \tag{2.60}$$

where N is the number of simulations that intercept a limit state, $u^{(i)}$ is the point of interception for the ith simulation, f_u is the standard normal density, $\nabla_x g(x,u)$ is the gradient of the limit state with respect to the deterministic design variables x, $\nabla_u g(x,u)$ is the gradient of the limit state with respect to the standard normal variables u, S_{p-1} is the surface area of the p-sphere of radius $r^{(i)}$, and $r^{(i)} = \left\|u^{(i)}\right\|$.

Method II

An alternative method for estimating the design sensitivities is considered due to the limitations of the chain rule derivation when $\mathrm{d}p_i / \mathrm{d}z_i$ approaches zero. This is inevitably the case when multiple limit states are present. In such situations derivatives estimated with method I can be incorrect when a limit state contributes only a small amount to the system failure probability. Instead, note that

the design sensitivity $\nabla_x z(x,u)$ for a given design x is a function of random variables u subject to the constraint $g(x,u) = z$. In directional simulation a random sample of N points on the constraint $g(x,u) = z$ has been obtained, furthermore the likelihood of each point $u^{(i)}$ is given by $f_u(u^{(i)})$, the standard normal density function. As a result, the design sensitivities may be estimated as

$$\nabla_x z(x) \approx \frac{\sum_{i=1}^{N} \nabla_x z(x,u^{(i)}) f_y(u^{(i)})}{\sum_{i=1}^{N} f_y(u^{(i)})}. \tag{3.25}$$

3.6 Adaptive Cokriging Model for Limit State Sensitivities

In many cases the limit state sensitivity $\nabla_x z(x,u) = \nabla_x g(x,u)$ is difficult or expensive to obtain. In this section an adaptive cokriging model is proposed for efficient estimation of $\nabla_x g(x,u)$ for a given design x. The following cokriging approach takes advantage of the information obtained during the AKIRA algorithm when directional simulation is used for probability of failure integration, namely

1. An accurate model of $g(x,u)$ has already been obtained in the vicinity of the limit state surface $g(x,u) = z$.
2. A set of points $u^{(i)}$ for $i = 1,\ldots,N$ has been obtained which reside on the limit state surface $g(x,u) = z$.

3.6.1 Cokriging

In cokriging, kriging (see section 2.4.2) is extended to the multivariate case by modeling the covariances between responses in addition to the covariances between the inputs for each response. In doing so, a response that has little data but is well correlated with another more well-informed response can rely on its correlation to make more accurate predictions with less uncertainty. There are many varieties of cokriging for different applications (see Chapter 5 of Chiles and Delfiner [29] for an introduction), however here we focus on the use of *simple bivariate cokriging*. *Simple* implies the assumption of a known and zero mean prior for each response, which is in agreement with the AKIRA kriging model $\hat{g}(x,u)$ which is built using normalized data. *Bivariate* implies that only two responses are correlated during the cokriging process, the *primary* response which typically has less data and is to be estimated, and the *secondary* response which typically has more data and informs the prediction for the primary response.

A general cokriging model can be written as

$$Z^{**}(u) = \sum_{i=1}^{r} \sum_{j=1}^{N_i} \lambda_{ij}(u) Z_i(u^{(ij)}) + \lambda_0, \tag{3.26}$$

where the r responses for $i = 1,\ldots,r$ are considered to be modeled by the random functions $Z_i(u) : u \in \mathbb{R}^p \to \mathbb{R}$, and the weights $\lambda_{ij}(u)$ define a linear combination of the random functions

$Z_i(u^{(ij)})$. Note that the weights λ_{ij} depend on the input data u, whereas the random functions do not and are *conditioned* on already sampled data pairs $(u^{(ij)}, Z_i(u^{(ij)}))$. Note also that each response's sampled data need not be collocated, *i.e.* data can be sampled for each response at different locations and in different quantities.

Adopting a vector notation leads to the following succinct representation of the cokriging estimator at point u:

$$Z^{**}(u) = \sum_{i=1}^{r} \lambda_i^\top Z_i + \lambda_0, \tag{3.27}$$

where $\lambda_i = [\lambda_{i0}, \ldots, \lambda_{iN_i}]^\top$ and $Z_i = [Z_i(u^{(i1)}), \ldots, Z_i(u^{(iN_i)})]^\top$.

In general the same steps as for kriging are followed in the derivation of the cokriging system equations which allow determination of the coefficients λ_i, namely that the bias of the estimator is first eliminated and then the mean square error of the estimator is minimized, leading to its moniker of Best Linear Unbiased Estimator [29]. These derivations are widely available and not reproduced here for brevity.

Simple Cokriging

In simple cokriging, the mean for each random function $Z_i(u)$ (*i.e.* response) is considered to be known and zero. The following cokriging system is then derived which allows determination of the unknown coefficients λ_i at a point u_0:

$$\sum_j K_{ij}\lambda_j = k_{i0} \text{ for } i = 1, \ldots, r, \tag{3.28}$$

where $K_{ij} = K(U^{(i)}, U^{(j)})$ is the $N_i \times N_j$ matrix of data-to-data covariances between the ith and jth responses and $U^{(i)} = [u^{(i1)}, \ldots, u^{(iN_i)}]^\top$ is the matrix of sampled data for the ith response. $k_{i0} = K(U^{(i)}, u_0)$ is similarly the $N_i \times 1$ covariance matrix between the known data $U^{(i)}$ for the ith response and the point u_0 whose response is estimated. In a bivariate cokriging model with two responses, the simple cokriging system in matrix notation is

$$\begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} k_{10} \\ k_{20} \end{bmatrix}, \tag{3.29}$$

which is succinctly written as $K\lambda = k_0$. Comparing the simple bivariate cokriging system in (3.29) with the simple kriging sytem from (2.38), we note that cokriging system considers cross-covariances (K_{12} and K_{21}) in addition to two univariate kriging covariance matrices (K_{11} and K_{22}). The solution of (3.29) may be obtained if K is positive definite via Cholesky decomposition.

The mean of the cokriging estimator $m(u_0)$ at u_0 for simple cokriging follows simply from (3.27)

as

$$m(u) = E[Z^{**}] = \sum_{i=1}^{r} \lambda_i^\top Z_i. \tag{3.30}$$

The cokriging variance $s^2(u_0)$ derives from the minimum mean square error calculation [29] as

$$s^2(u) = \mathrm{Var}(Z^{**})(u) = K_{00} - \sum_{i=1}^{r} \lambda_i^\top K_{i0}. \tag{3.31}$$

Multivariate Matérn Covariance Kernel

The selection of a covariance function for modeling cross-covariances between random functions Z_i and Z_j is non-trivial, especially since it is desirable that such functions yield a positive definite covariance matrix K. One common covariance model is the proportional covariance model where all direct and cross-covariances are assumed to be proportional to the same covariance kernel, *i.e.* $K_{ij}(h) \propto K(h)$ [29]. Here h represents the separation between two points, implying a *stationarity* assumption, namely that the mean and variance of the random fields Z_i and Z_j are invariant under translations (second order stationarity).

In the bivariate cokriging model the proportional covariance model is

$$\begin{aligned} K_{11}(h) &= \sigma_1^2 K(h) \\ K_{22}(h) &= \sigma_2^2 K(h) \\ K_{12}(h) &= K_{21}(h) = \rho\sigma_1\sigma_2 K(h), \end{aligned} \tag{3.32}$$

where $-1 < \rho < 1$. In the multivariate Matérn covariance kernel $K(h)$ is taken to be a univariate Matérn covariance function which is described in Appendix B.2.

In order to ensure a positive definite covariance matrix, Gneiting et al derived several conditions on the hyperparameters of the Matérn covariance functions (namely smoothness parameters ν_{ij}, scale parameters a_{ij} and variance parameters σ_i^2 and ρ_{ij}) [62]. Interested readers are referred to their paper [62] for the appropriate conditions which should be checked during kernel construction and hyperparameter optimization.

3.6.2 Model Construction

For a given limit state $g(x_0, u) : u \in \mathbb{R}^p \to \mathbb{R}$, a model can be constructed for the partial derivative $\partial g/\partial x_i(x_0, u) : u \in \mathbb{R}^p \to \mathbb{R}$ using cokriging. The primary response (to be estimated) is taken to be the partial derivative $\partial g/\partial x_i(x_0, u)$ and the secondary response is taken to be the limit state response $g(x_0, u)$ for which a very accurate representation was previously obtained through the AKIRA algorithm.

The accuracy of the assumption of a correlation between a function and its gradient depends on the case. If there is a strong correlation then cokriging will make estimation of the the primary response very efficient, however if there is no correlation then no benefit is gained by cokriging and the system will degenerate to univariate kriging with an appropriate choice of kernel hyperparameters ($K_{12} = K_{21} = 0$ in (3.29)). Take for example, the artificially modified limit state function with 3 deterministic design variables and 2 random variables [21]:

$$g(x,u) = x_1(u_1 - u_2)^2 - (u_1 + u_2)x_2 + x_3. \tag{3.33}$$

The limit state in (3.33) is well-correlated with its partial derivative $\partial g/\partial x_3$ in the standard normal space (via a translation) but less well-correlated with its partial derivative $\partial g/\partial x_1$ due to the quadratic nonlinearity in u.

Instead of constructing a series of independent bivariate cokriging models, one for each partial derivative, one could consider solving a larger cokriging system where all partial derivatives and the limit state are considered correlated. In this case correlations between partial derivatives could be exploited by the cokriging model to yield potentially more efficient estimates of $\nabla_x g(u,x)$, however for large numbers of design variables the cokriging system matrix K will become large. A more significant problem is perhaps the accurate and efficient determination of covariance kernel hyperparameters which will grow quadratically with the $n+1$ responses in the cokriging system for $x \in \mathbb{R}^n$.

Thus in the general case, a total of mn simple bivariate cokriging models are constructed for m limit states $g_i(x_0,u)$ and their n partial derivatives $\partial g_i/\partial x_j(x_0,u)$. Each cokriging model is initially trained using the N samples obtained from AKIRA for $g(x,u)$ and a single sample for $\nabla_x g(x,u)$ located at the most probable point (MPP) of failure. The multivariate Matérn covariance kernel is used and the kernel's variance hyperparameters σ_1^2, σ_2^2, and ρ are optimized via maximum log marginal likelihood (see section 2.4.2). The length scales for all covariance kernels are taken to be the same as those in refined kriging model for $g(x,u)$ from AKIRA.

3.6.3 Model Refinement

Each simple bivariate cokriging model can be refined using a Stepwise Uncertainty Reduction (SUR) strategy [16, 26] which was introduced in section 3.3.1. As a reminder a SUR strategy seeks to choose a point $u^{(s+1)}$ or points which minimize the average of a future measure of uncertainty $H_{s+1}(u)$ in a quantity of interest given the selection of those points.

In the current problem, the quantity of interest is a design sensitivity $\partial p_f/\partial x_i(x)$ which can be approximated by a finite weighted sum of limit state sensitivities $\partial g/\partial x_i(x,u^{(j)})$ for $j = 1,\ldots,N$.

For example, when directional simulation is used, the ith partial of p_f can be written as

$$\frac{\partial p_f}{\partial x_i}(x) \approx \sum_{j=1}^{N} \gamma_j(u^{(j)}) \frac{\partial g}{\partial x_i}(x, u^{(j)}), \tag{3.34}$$

which was derived in section 2.6.1. The derivative is further approximated by replacing the limit state sensitivity by its cokriging model $\hat{\partial g}/\partial x_i(x)$

$$\frac{\partial p_f}{\partial x_i}(x) \approx \sum_{j=1}^{N} \gamma_j(u^{(j)}) \frac{\hat{\partial g}}{\partial x_i}(x, u^{(j)}). \tag{3.35}$$

SUR Uncertainty Measure

At this point to simplify notation we consider a general quantity of interest $f \in \mathbb{R}$ which is defined as a weighted sum of N cokriging model evaluations $c(u)$, where the cokriging model has been constructed from s samples $u^{(1)}, \ldots, u^{(s)}$ in the form

$$f = \sum_{i=1}^{N} \gamma_i c(u^{(i)}), \tag{3.36}$$

where the correspondence with (3.35) is clear.

One appropriate measure of uncertainty $H_{s+1}(u)$ for the quantity of interest f is its variance taking $u^{(s+1)}$ into account given by

$$\begin{aligned} \mathrm{Var}_{s+1}(f) &= \mathrm{Var}_{s+1}\left(\sum_{i=1}^{N} \gamma_i c(u^{(i)})\right) \\ &= \sum_{i=1}^{N} \gamma_i^2 \mathrm{Var}_{s+1}\left(c(u^{(i)})\right) + 2\sum_{i=1}^{N}\sum_{j=i+1}^{N} \gamma_i\gamma_j \mathrm{Cov}_{s+1}\left(c(u^{(i)}), c(u^{(i)})\right), \end{aligned} \tag{3.37}$$

which shows that the variance and the covariances of each of N points with each other must be calculated from the cokriging model. The conditional expectation of $\mathrm{Var}_{s+1}(f)$ requires a univariate integral over the possible values of $c(u^{(s+1)})$ and can be approximated using Gauss-Hermite quadrature as

$$E_s[\mathrm{Var}_{s+1}(f) \mid u^{(s+1)}] \approx \sum_{q=1}^{Q} w^{(q)} \mathrm{Var}_{s+1}^{(q)}(f), \tag{3.38}$$

where $w^{(q)}$ are the quadrature weights for the Q univariate quadrature points and $\mathrm{Var}_{s+1}^{(q)}(f)$ is the variance of f when the $s+1$st sample is evaluated at the qth quadrature point.

Extension to Multiple Quantities of Interest

In the section above a single quantity of interest f was considered for refinement. If however n quantities of interest (each estimated using separate cokriging models) are considered for refinement and a single point $u^{(s+1)}$ is to be picked at which to evaluate each of them, we choose the point corresponding to the quantity with the maximum optimized SUR criterion from (3.6). Such a strategy is only valid when all quantities of interest are comparable, such as when each cokriging model is constructed using normalized data. The rationale of this approach is to choose the best sample according to the SUR strategy for the quantity of interest with largest uncertainty.

Termination Criterion

A satisfactory termination criterion for cokriging model refinement is one that ensures the uncertainty in the quantity of interest is small. Unfortunately, it is difficult to ensure the cokriging model is sufficiently accurate without exhaustive model refinement due to the assumptions made in the choice of covariance kernel. Nevertheless, a bound on the maximum coefficient of variation for the quantities of interest $\partial p_f / \partial x_i(x, u)$ is useful:

$$\max_i \left\{ \frac{\sqrt{\mathrm{Var}\left(\frac{\partial \hat{p}_f}{\partial x_i}\right)}}{E\left[\frac{\partial \hat{p}_f}{\partial x_i}\right]} \right\} \leq \tau_c, \tag{3.39}$$

where τ_c is taken to be 0.1 or so.

3.6.4 A Practical Algorithm

The procedure outlined in this section for the construction and adaptive update of a cokriging model for limit state sensitivities $\nabla_x g(x, u)$ is nontrivial and can become quite expensive when the number of design variables x and limit states are large. Algorithm 3 outlines a practical algorithm.

Algorithm 3: Adaptive SUR cokriging approximation for limit state sensitivities

Data: N samples $u^{(1)}, \ldots, u^{(N)}$ on the limit state surface obtained via directional simulation, refined kriging model for the limit states $\hat{g}_j(x,u)$ for $j = 1, \ldots, m$, Q quadrature points, convergence tolerance τ_c

Result: Refined cokriging models of limit state sensitivities $\frac{\hat{\partial} g_j}{\partial x_k}(x,u)$

$u_{MPP} \leftarrow$ Determine the most probable point (MPP) of failure: $\text{argmin}_{u^{(i)}} \left\| u^{(i)} \right\|$;

$\nabla_x g(x, u_{MPP}) \leftarrow$ Evaluate design sensitivities at the MPP;

for $j \leftarrow 1$ **to** m **do** `// each limit state` g_j

  for $k \leftarrow 1$ **to** n **do** `// each design variable` x_k

    $\frac{\hat{\partial} g_j}{\partial x_k}(x,u) \leftarrow$ Build a bivariate cokriging model for each response and partial derivative;

$w^{(q)}$, $p^{(p)}$, $q = 1, \ldots, Q \leftarrow$ Calculate Gauss Hermite quadrature weights and points;

Select $M < N$ representative points from $u^{(1)}, \ldots, u^{(N)}$;

for $s \leftarrow 1$ **to** s_{max} **do**

  if $max_i \left\{ \frac{\sqrt{Var\left(\frac{\partial \hat{p}_f}{\partial x_i}\right)}}{E\left[\frac{\partial \hat{p}_f}{\partial x_i}\right]} \right\} \leq \tau_c$ **then** `// model is converged`

    return $\frac{\hat{\partial} g_j}{\partial x_k}(x,u)$;

  for $i \leftarrow 1$ **to** M **do** `// each prospective point` $u^{(i)}$

    for $q \leftarrow 1$ **to** Q **do** `// each quadrature point` $p^{(q)}$

      for $j \leftarrow 1$ **to** m, $k \leftarrow 1$ **to** n **do** `// each cokriging model`

        $r_{ijk}^{(q)} \leftarrow m_{jk}(u^{(i)}) + p^{(q)} \sigma_{jk}(u^{(i)})$;

        $\frac{\tilde{\partial} g_j}{\partial x_k}(x,u) \leftarrow$ Temporarily update cokriging model with $(\frac{\hat{\partial} g_j}{\partial x_k}(u), u^{(i)}, r_{ijk}^{(q)})$;

        $V_{ijk}^{(q)} \leftarrow \text{Var}(\frac{\tilde{\partial} g_j}{\partial x_k}(u))$;

      $S_{ijk} \leftarrow$ Calculate SUR criterion: $\sum_{q=1}^{Q} w^{(q)} V_{ijk}^{(q)}$;

  $u^{(s+1)} \leftarrow$ Select next sample as optimum of: $\max_{j,k}\{\min_i\{S_{ijk}\}\}$;

  $\nabla_x g(x, u^{(s+1)}) \leftarrow$ Evaluate design senstivities at $u^{(s+1)}$;

  for $j \leftarrow 1$ **to** m, $k \leftarrow 1$ **to** n **do** `// each cokriging model`

    $\frac{\hat{\partial} g_j}{\partial x_k}(u) \leftarrow$ Update cokriging model with $(u^{(s+1)}, \frac{\partial g_j}{\partial x_k}(u^{(s+1)}))$;

3.6.5 Demonstration

Table 3.2 compares the design sensitivities $\nabla_x \hat{p}_f$ calculated using several methods after the AKIRA algorithm successfully converged for the following modified convex limit state with 3 deterministic

Table 3.2: Comparison of methods for approximating design sensitivities $\nabla_x \hat{p}_f$ for the simple modified convex limit state in (3.33). For this problem, the adaptive cokriging approach yields acceptable derivatives with two gradient evaluations and near-exact derivatives with six. Note also that merely using the design sensitivities at the most probable point (MPP) does not necessarily lead to accurate derivative estimation.

Method	$\frac{\partial \hat{p}_f}{\partial x_1}$	$\frac{\partial \hat{p}_f}{\partial x_2}$	$\frac{\partial \hat{p}_f}{\partial x_3}$
Exact	-0.03658	0.09278	-0.02889
One sample at MPP	-4.2848×10^{-7}	0.08770	-0.02889
Adaptive cokriging with 2 samples	-0.01254	-0.0890	-0.02889
Adaptive cokriging with 6 samples	-0.03679	0.09278	-0.02889

design variables $x \in \mathbb{R}^3$ [21]:

$$g(x,u) = x_1(u_1 - u_2)^2 - (u_1 + u_2)x_2 + x_3. \tag{3.33}$$

In this case the cokriging approach yields an acceptable derivative approximation with only two gradient evaluations and a near-exact approximation with six evaluations. Derivative estimation based solely on the gradient at the most probable point (MPP) may be prone to error as shown even in this simple example. The derivative for x_1 is completely wrong when calculated only using the MPP.

3.7 Demonstration

In this section, AKIRA is demonstrated in detail on several test problems including a highly nonlinear limit state with $y \in \mathbb{R}^2$, a cantilever beam problem with two limit states with $y \in \mathbb{R}^4$, and several other structural reliability test problems. Comparisons are made with Monte Carlo and inverse FORM when possible.

3.7.1 2D Example: Highly Nonlinear Limit State

Grandhi and Wang [65] introduced the highly nonlinear limit state

$$g(y) = 2.5 + 0.00463(y_1 + y_2 - 20)^2 - 0.2357(y_1 - y_2), \tag{3.40}$$

where $y_1, y_2 \sim \mathcal{N}(10, 3^2)$ and are uncorrelated. Using a response surface method to solve the reliability analysis for the limit state in (3.40) is challenging due to its nonlinearity.

Here we apply AKIRA for the solution of the inverse reliability analysis with a prescribed probability of failure $\bar{p}_f = 10^{-2}$. Four Monte Carlo samples are used in the initial sampling stage in addition to a sample at the mean of the random variables. Samples are adaptively selected in

Table 3.3: Random parameters in the cantilever beam problem.

Name	Variable	Distribution	Mean	CoV
Yield strength	Y	Normal	40,000	0.05
Elastic modulus	E	Normal	2.9×10^7	0.05
Horizontal load	H	Normal	500	0.2
Vertical load	V	Normal	1,000	0.1

batches of five using the heuristic from section 3.3.2 using $z_{\alpha/2} = 0.67$ (75% one-sided confidence interval) and added to the kriging model each iteration. Figure 3.8 compares the exact limit state contours and kriging model contours at the beginning, end, and two intermediate iterations during the AKIRA algorithm. AKIRA converges in 9 iterations when the $\hat{p}_f$ convergence ratio π_f falls below 0.05 leading to 45 total function evaluations.

The convergence of AKIRA is shown in Figure 3.9. The estimated limit state level from AKIRA is $\hat{z} = 0.4371$ which compares well with level $z_{MC} = 0.4654$ obtained from a Monte Carlo simulation with 10^6 samples ($CoV = 0.01$) on the exact limit state function. The -6.1% difference between the two levels is attributable to two sources of error: inaccuracies in the kriging model in the vicinity of the limit state and differences between the directional simulation and Monte Carlo sampling methods. Postprocessing using directional simulation with common random numbers (*i.e.* the same set of directions used during AKIRA) on the exact limit state for the final level estimate shows $\hat{p}_f = 0.996 \times 10^{-2}$ ($CoV = 0.1$), thus we conclude that the kriging model is very accurate. Instead, the main contribution to the source of error in $\hat{z}$ is the use of uniform directional simulation. A more accurate estimate of $\hat{z}$ can be made by using a more accurate integration method or searching for multiple roots in directional simulation.

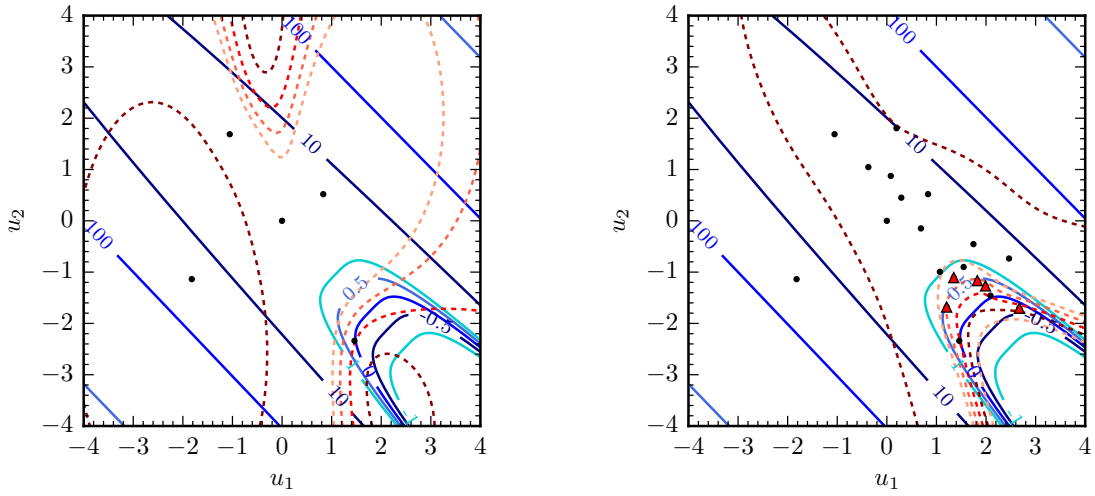
3.7.2 4D Example with Multiple Limit States: Cantilever Beam

The cantilever beam problem is illustrated in Figure 3.10 [132, 126]. Its four random variables $y = [Y, E, H, V]^\top$ are summarized in Table 3.3. For the purposes of demonstrating AKIRA the beam geometry's geometry parameterized by $x = [w, t]^\top$ is fixed with $w = 2.6041$, $t = 3.6746$ and $L = 100$ inches. Two limit states are considered: one related to stress and one related to displacement. The stress S in the beam is given by

$$S(x, y) = \frac{600}{wt^2}V + \frac{600}{w^2t}H, \tag{3.41}$$

whereas the displacement of the beam D is given by

$$D(x, y) = \frac{4L^3}{Ewt}\sqrt{\left(\frac{V}{t^2}\right)^2 + \left(\frac{H}{w^2}\right)^2}. \tag{3.42}$$



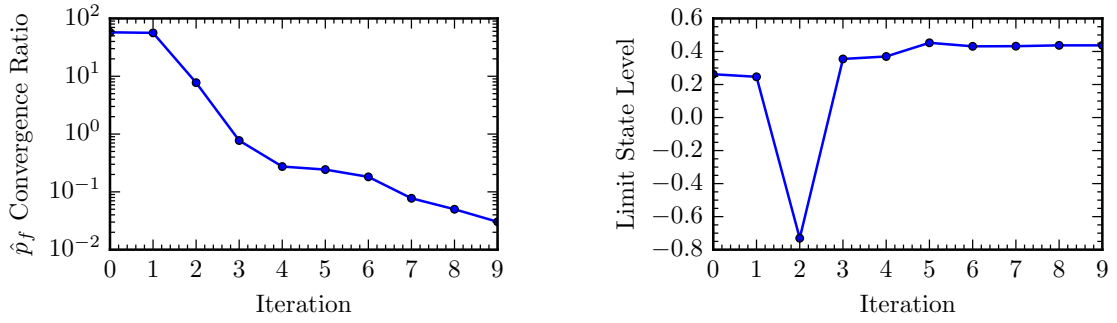
(a) $k = 0$

(b) $k = 4$

(c) $k = 7$

(d) $k = 9$

Figure 3.8: AKIRA's kriging approximation (dashed lines) of the highly nonlinear limit state (solid lines) defined in (3.40) at the beginning ($k = 0$) of the algorithm and at iterations 4, 7, and final iteration 9. The prescribed probability of failure $\bar{p}_f$ is 10^{-2} which corresponds to a limit state level $z = 0.4654$ (estimated from 10^6 Monte Carlo samples). AKIRA terminates after 9 iterations with estimated $\hat{z} = 0.4371$. The multipoint adaptive point selection strategy from section 3.3.2 is employed with batches of 5 points selected per iteration. The last 5 samples added to the kriging model are denoted using red triangles; other training samples are shown with black dots. A small initial Monte Carlo sample of 4 points is used to illustrate AKIRA's recovery from a poor initial kriging model in the first iteration. Note how AKIRA chooses to refine the kriging model in the region that contributes most to the probability of failure; samples are also relatively spaced out.



(a) Convergence of π_f

(b) Convergence of $\hat{z}$

Figure 3.9: Convergence of AKIRA for the nonlinear limit state defined in (3.40) occurs in 9 iterations and 45 total function evaluations. The convergence of π_f, the $\hat{p}_f$ convergence ratio defined in (3.14) in subplot 3.9a illustrates the convergence of AKIRA's kriging model in the regions important to the probability of failure calculation. This is qualitatively shown in Figure 3.8 as well. A tolerance of 0.05 is used for π_f. Convergence of the estimated limit state level $\hat{z}$ in subplot 3.9b shows that by iteration 3, AKIRA already has decent estimate of the solution to the inverse reliability analysis problem.

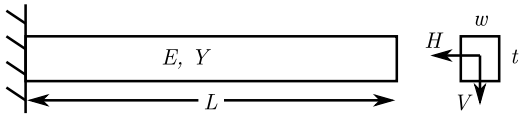


Figure 3.10: Schematic for the proposed cantilever beam problem subject to a lateral and vertical load [132, 126]. Elastic modulus E, yield strength Y, and horizontal and vertical loading H and V are uncertain. $L = 100$ inches and w and t are deterministic design variables.

Failure occurs when the stress S exceeds the yield stress Y or the displacement D exceeds $D_0 = 2.2535$ inches. The limit states are normalized such that a value less than zero corresponds to failure. The normalized limit states $g_s(x, y)$ and $g_D(x, y)$ are given by

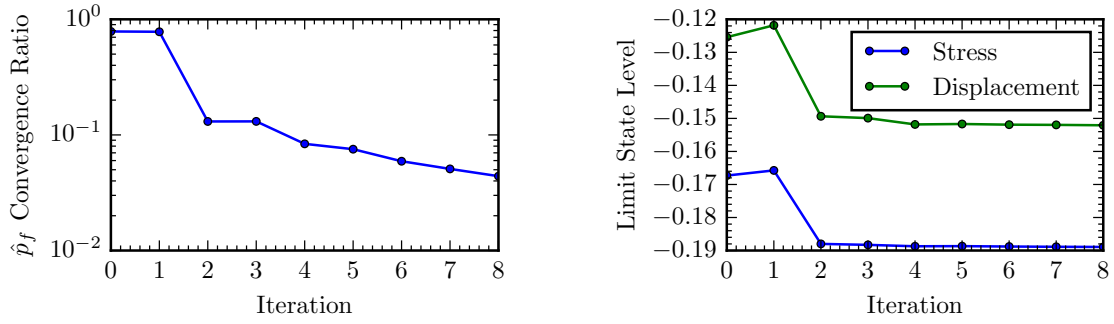
$$g_S(x, y) = 1 - \frac{S(x, y)}{Y}, \tag{3.43}$$

$$g_D(x, y) = 1 - \frac{D(x, y)}{D_0}. \tag{3.44}$$

We apply AKIRA for the solution of the inverse reliability analysis with a prescribed probability of failure $\bar{p}_f = 10^{-6}$. 8 Monte Carlo samples are used in the initial sampling stage in addition to a sample at the mean of the random variables. Samples are adaptively selected in batches of five using the heuristic from section 3.3.2 using $z_{\alpha/2} = 0.67$ (75% one-sided confidence interval) and added to the kriging model each iteration. AKIRA converges in 8 iterations when the $\hat{p}_f$ convergence ratio π_f falls below 0.05 leading to 47 total function evaluations. Inspection of the AKIRA analysis

shows failure occurs most often when the loads are high, yield strength is low, and elastic modulus is slightly below average. Both limit states contribute nearly equally to the probability of failure with the stress limit state garnering 182 failure samples and the displacement limit state garnering 215 failure samples out of 3000 directional simulations.

The convergence of AKIRA is shown in Figure 3.11. The estimated limit state levels from AKIRA are $\hat{z}_S = -0.1889$ for the stress and $\hat{z}_D = -0.1521$ for the displacement. A Monte Carlo simulation with 10^8 samples ($CoV = 0.1$) using $\hat{z}_S$ and $\hat{z}_D$ from AKIRA estimates the probability of failure as 9.5×10^{-7}, thus we conclude the AKIRA estimate of the limit state levels is fairly accurate.



(a) Convergence of π_f

(b) Convergence of $\hat{z}$

Figure 3.11: Convergence of AKIRA for the cantilever beam problem with limit states defined in equations 3.43 and 3.44. Convergence occurs after 8 iterations and 47 total function evaluations for each limit state. The convergence of π_f, the $\hat{p}_f$ convergence ratio defined in (3.14) in subplot 3.11a illustrates the convergence of AKIRA's kriging model in the regions important to the probability of failure calculation. A tolerance of 0.1 is used for π_f. Convergence of the estimated limit state level $\hat{z}$ in subplot 3.11b shows that by iteration 4, AKIRA already has a good estimate of the solution to the inverse reliability analysis problem.

3.7.3 14D Example with Multiple Limit States: Supersonic Nozzle

The supersonic nozzle problem is described in detail in chapter 5. Here we content ourselves with a cursory overview for the purpose of demonstrating AKIRA. Figure 3.12 shows the nozzle geometry which is axisymmetric only for illustrative purposes. Three quantities of interest and 14 random parameters have been identified as important contributors to the performance and failure states of the nozzle (see section 5.5.3). These are the thrust F which is primarily affected by variability in pressures and temperatures, the temperature ratio in the inner load layer T (component 2 in Figure 3.12) which is primarily affected by variability in material thermal properties such as thermal conductivity as well as temperatures, and the structural failure criterion S in the thermal layer which is primarily affected by variability in material thermal properties and temperatures.

AKIRA is applied for the solution of the inverse reliability analysis with a prescribed probability

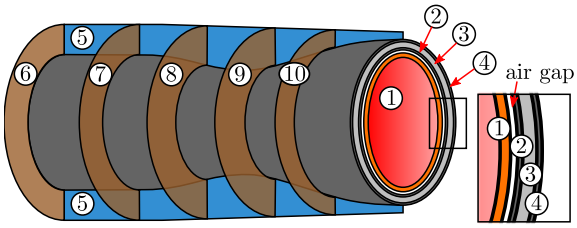


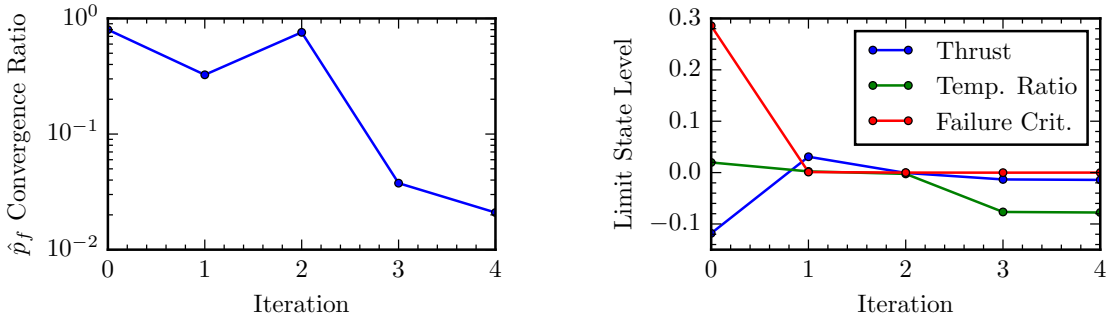
Figure 3.12: Schematic for the supersonic nozzle design problem. 14 random parameters have been identified as important including the inlet and atmospheric pressures and temperatures, various material properties such as thermal conductivities and the heat transfer coefficient from the external nozzle wall to the ambient. Three quantities of interest are considered to be the critical design limiting quantities: the thrust, the temperature ratio in the inner load layer (2), and the structural failure criterion in the thermal layer (1). A full description of the supersonic nozzle design problem can be found in chapter 5.

of failure $\bar{p}_f = 10^{-6}$. 70 samples generated from a Sobol' sequence are used in the initial sampling stage in addition to a sample at the mean of the random variables. Samples are adaptively selected in batches of eight using the heuristic from section 3.3.2 using $z_{\alpha/2} = 0.67$ (75% one-sided confidence interval) and added to the kriging model each iteration. AKIRA converges in 4 iterations when the $\hat{p}_f$ convergence ratio π_f falls below 0.1 leading to 95 total function evaluations. Inspection of the AKIRA analysis shows failure occurs most often when the inlet stagnation pressure is high, the heat transfer coefficient is low, thermal conductivity of the air gap is high, and the max service temperature in the inner load layer is low (low/high is measured with respect to the mean). Under such conditions, both stress and temperature increase in the nozzle wall layers, leading to a greater chance of failure. The thrust and the inner load layer temperature ratio quantities of interest contribute to the system failure of the nozzle whereas the failure criterion in the thermal layer is not an active limit state for the chosen design.

The convergence of AKIRA is shown in Figure 3.13. The estimated limit state levels (for the normalized quantities of interest; ≤ 0 is failure) from AKIRA are $\hat{z}_F = -0.0142$ for the thrust, $\hat{z}_T = -0.0777$ for the inner load layer temperature ratio and $\hat{z}_S = 0.0000$ for the structural failure criterion. Note for this particular design, the nozzle system has failed and the property of consistency discussed in section 3.4 is met (all limit state levels returned by AKIRA correspond to failure or non-failure), even for the inactive limit state $\hat{z}_S$. A Monte Carlo simulation was not performed for comparison due to the expense of running the nozzle problem.

3.7.4 Comparison of AKIRA Performance on Select Problems

Table 3.4 compares the performance of AKIRA with Monte Carlo and FORM using the performance measure approach (see section 2.3.1) when possible on several common structural reliability analysis test problems. For problems with multiple limit states, a direct comparison cannot be made since



(a) Convergence of π_f

(b) Convergence of $\hat{z}$

Figure 3.13: Convergence of AKIRA for the supersonic nozzle problem with three normalized limit states: thrust F, inner load layer temperature ratio T, and thermal layer structural failure criterion S, described in detail in chapter 5. Convergence occurs after 4 iterations and 95 total function evaluations for each limit state. The convergence of π_f, the $\hat{p}_f$ convergence ratio defined in (3.14) in subplot 3.13a illustrates the convergence of AKIRA's kriging model in the regions important to the probability of failure calculation. A tolerance of 0.1 is used for π_f. Convergence of the estimated limit state level $\hat{z}$ in subplot 3.13b demonstrates the consistency property described in section 3.4 where limit state levels chosen by AKIRA (iterations 1–4) either all correspond to failure or non-failure (in this case $>$ or ≤ 0).

other methods work for a single limit state. Instead, the limit state levels obtained with AKIRA are used in a Monte Carlo forward reliability analysis to verify they correspond to the prescribed probability of failure $\bar{p}_f$.

The full problem definitions for the short column and the steel column problems can be found in Appendix A. The short column problem has four random variables and features a short column with rectangular cross-section subject to two bending moments and an axial load [77]. The steel column problem has nine random variables and features a steel column subject to a variety of loads and geometric uncertainties [77]. Table 3.4 shows that AKIRA is superior to FORM in terms of accuracy on nonlinear limit states and also that AKIRA solves the inverse problem in relatively few function evaluations.

Table 3.4: AKIRA is compared with Monte Carlo and FORM for a prescribed probability of failure $\bar{p}_f = 10^{-6}$. The average estimated limit state level $\hat{z}$ and average number of limit state evaluations N is reported from 20 independent AKIRA analyses. Coefficients of variation are reported in parentheses. Data marked with – is unavailable; for FORM and Monte Carlo this is due to lack of a means to solve the inverse problem for multiple limit states. Instead, a Monte Carlo forward reliability analysis is run for the average limit state levels estimated by AKIRA to verify AKIRA accurately identifies limit state levels for the prescribed probability of failure. The AKIRA algorithm uses the multipoint adaptive point selection strategy from section 3.3.2 with batches of 5 points selected per iteration.

Problem	Method	$\hat{z}$ / $\hat{p}_f$	N
Short Column	Monte Carlo	$\hat{z} = -0.3796$	10^8
	Inverse FORM	$\hat{z} = -0.3048$	25
	AKIRA	$\hat{z} = -0.3677$ (0.024)	73 (0.19)
Steel Column	Monte Carlo	$\hat{z} = -0.2562$	10^8
	Inverse FORM	$\hat{z} = -0.3158$	65
	AKIRA	$\hat{z} = -0.2524$ (0.081)	72 (0.05)
Cantilever	Monte Carlo	$\hat{p}_f = 1.012 \times 10^{-6}$ (0.03)	10^8
	Inverse FORM	—	—
	AKIRA	$\hat{z} = [-0.1894(0.014), -0.1524(0.011)]^\top$	54 (0.13)

Chapter 4

Reliability-Based Design Optimization (RBDO)

Many problems in engineering as well as physics, economics and myriad other disciplines can be formulated as optimization problems, that is, as the minimization or maximization of some quantity or quantities such as safety, energy, or profit. In engineering the word "optimization" is often used interchangeably with "design", the intent being that a good design has a lower cost, higher reliability, or some combination thereof than other designs. Chapters 1 and 2 motivated the importance of reliability in engineering systems and as such, the incorporation of reliability into design optimization is the logical next step thereby leading to *reliability-based design optimization* (RBDO). The use of RBDO enables engineers and designers to ensure that systems perform reliably and maintain sufficient standards of safety. The application areas are virtually limitless and include among others, electronics, finances, structures, and aerospace.

This chapter introduces RBDO and proposes a new multifidelity sequential method for use with AKIRA. First an introduction to deterministic and reliability-based design optimization is given in section 4.1, including the reformulation of the RBDO problem using the performance measure approach. Next, traditional methods for building margin into a design to account for uncertainties such as safety factors and basis values are discussed in section 4.2. Then existing classes of RBDO methods and classical trust region methods are reviewed in sections 4.3 and 4.4. Finally a sequential RBDO algorithm which takes advantage of anchored decomposition, multifidelity models, and trust region methods is presented in section 4.5. A demonstration of the algorithm on two common structural reliability problems is given in section 4.6. A review of optimization theory may be helpful and can be found in Appendix C.

4.1 Introduction

4.1.1 Deterministic Optimization

First let us take a deterministic standpoint for the purpose of introducing optimization. Often the most difficult task in design optimization is choosing the appropriate mathematical description for the physical design problem at hand. First and foremost are the design variables $x \in \mathbb{R}^n$. The design variables are the "knobs" which can be turned to tune the design; for example geometric dimensions or process parameters. The *objective* $f(x) : \mathbb{R}^n \to \mathbb{R}$ is the quantity to be minimized or maximized; for example cost or weight. The *constraints* $g(x) : \mathbb{R}^n \to \mathbb{R}^m$ prescribe limits on characteristics of the design; for example a maximum allowable temperature or stress in a structural component. Constraints may also encompass geometric limitations such as size and shape as well as bounds on the design variables x. A typical deterministic nonlinear optimization problem with nonlinear inequality constraints can be written as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{s.t.} \quad & g(x) \geq 0, \end{aligned} \tag{4.1}$$

where the constraints $g(x)$ may include nonlinear constraints, linear constraints, and/or bounds on the variables x. On occasion the presence of linear constraints and variable bounds is emphasized by explicitly writing them out as shown in the following problem statement:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{s.t.} \quad & g(x) \geq 0 \\ & Ax \geq b \\ & l \leq x \leq u. \end{aligned} \tag{4.2}$$

The problem in (4.2) shows a set of nonlinear constraints $g(x)$, linear constraints defined by the matrix A and vector b, and lower and upper bounds l and u on the design variables x. Linear constraints are common in design optimization problems where simple linear relationships often exist between x_i as a result of the chosen design parameterization. Such constraints can be used to enforce reasonable designs and furthermore can be directly incorporated when an active set optimization algorithm is used [61], ensuring that the linear constraints are satisfied for all iterates *i.e.* that only reasonable designs are considered during the optimization.

Naturally the solution of the optimization problem in (4.1) yields an optimal design insofar as the objective and constraint functions accurately model the true behavior of the system being designed, and in addition that all variables and requirements have been taken into account. As a result, proper problem specification and design variable, objective, and constraint identification are key to

obtaining a useful optimal design. *If any part of the design problem is misspecified or overlooked, the result of the optimization is likely to be useless.* The supersonic nozzle design problem introduced and solved in chapter 5 showcases a methodical procedure for determining a proper optimization problem in the context of engineering design.

Unfortunately the optimization problems stated in (4.1) and (4.2) are only approximations of any real design problem. As discussed in chapter 1, in addition to a dependence on x, design problems are also characterized by uncertainties, which are parameterized here by the random variables y. Reliability-based design optimization seeks to directly account for the uncertainties y in order to meet reliability requirements for a design.

4.1.2 RBDO Problem Formulation

A RBDO problem can be written as a stochastic optimization problem over deterministic variables $x \in \mathbb{R}^n$ and random variables $y \in \mathbb{R}^p$, where one or more of the objectives or constraints seek a small probability of failure, *i.e.* a reliable design. A typical form of the RBDO problem is shown in (4.3) for the objective $f(x,y) \in \mathbb{R}$ and limit states $g(x,y) \in \mathbb{R}^m$:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & E[f(x,y)] \\ \text{s.t.} \quad & P\left[g_i(x,y) \leq \bar{z}_i\right] \leq \bar{p}_{f,i} \quad i = 1,\ldots,m \\ & Ax \leq b, \end{aligned} \tag{4.3}$$

where x are deterministic design variables which may be independent from y or may be distributional parameters for the random variables y such as a mean. The objective function $f(x,y)$ is a random function and its expectation $E[\cdot]$ is taken as the optimization objective. Each limit state $g_i(x,y)$ is constrained to only violate its performance bound $\bar{z}_i$ with at most probability $\bar{p}_{f,i}$. Finally a set of linear deterministic constraints defined by the matrix A and vector b are included which specify simple relationships between the deterministic variables as well as their bounds.

The RBDO problem in (4.3) constrains the failure probability of each limit state independently. However in most engineering systems the *system failure probability*, *i.e.* the chance that the system as a whole fails instead of a single component, is more important. System reliability was introduced in chapter 2 on reliability. For example recall that in a series system (*i.e.* failure of one subsystem results in system failure), the probability of failure is given by

$$\begin{aligned} p_f(x) &= P\left[\bigcup_{i=1}^{n} \left\{(x,y) : g_i(x,y) \leq \bar{z}_i\right\}\right] \\ &= P\left[\min_i \left\{g_i(x,y) - \bar{z}_i\right\} \leq 0\right]. \end{aligned} \tag{2.2}$$

For a series system designed for a prescribed system failure probability $\bar{p}_f$, a typical form of the

RBDO problem is given by

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & E[f(x,y)] \\ \text{s.t.} \quad & P\left[\min_i \left\{g_i(x,y) - \bar{z}_i\right\} \leq 0\right] \leq \bar{p}_f \\ & Ax \leq b. \end{aligned} \tag{4.4}$$

A similar formulation can be written for parallel or mixed systems.

The RBDO problems in (4.3) and (4.4) are typically difficult to solve for several reasons. First, they require resolution of stochastic quantities such as expectations and more challenging, probabilities of failure, which are typically small numbers and expensive to resolve accurately. Second, x and y are typically high-dimensional ($n, p > 10$) and $f(x,y)$ and $g(x,y)$ are also often nonlinear in a realistic application which increases both the computational expense of the probability integration as well as the computational expense of solving the optimization problem. Finally, in many realistic applications of engineering interest, the objective and/or limit state functions $f(x,y)$ and $g(x,y)$ are computationally expensive to evaluate. For example in the domain of aerospace engineering a single coupled analysis (*i.e.* one function evaluation of $f(x,y)$ and $g(x,y)$) may require hours or days of compute time[1]. As a result, many methods for solving RBDO problems have focused on two areas: increasing the efficiency of the solution procedure, and when possible also increasing the accuracy of the solution. Section 4.3 surveys existing RBDO methods.

4.1.3 RBDO Reformulation with Inverse Reliability Analysis

The benefits of using an inverse reliability analysis (*i.e.* the *performance measure approach*) instead of a forward reliablity analysis (*i.e.* the *reliability index approach*) were introduced in section 2.1.5 and have been advocated by many [128, 3, 40, 136, 5, 83]. As a reminder, a forward reliability analysis involves calculating a probability of failure p_f for a set of limit state functions $g(x,y)$ with an associated set of failure levels $\bar{z}$. The RBDO formulations in equations 4.3 and 4.4 use a reliability index approach since the problem formulation uses chance constraints, *i.e.* constraints of the form $p_f(x) \leq \bar{p}_f$.

On the other hand, the use of an inverse reliability analysis gives rise to quantile constraints in the RBDO problem formulation, *i.e.* constraints of the form $z(x) > \bar{z}$. An inverse reliability analysis solves for the limit state levels $z(x)$ that yield a prescribed probability of failure $\bar{p}_f$. Using a performance measure approach can yield a more robust optimization problem, avoid the cost of evaluating probabilities smaller than $\bar{p}_f$ and is potentially easier to solve for certain types of reliability analysis methods (see section 2.1.5 for a full description).

[1]For example, the supersonic nozzle problem presented in chapter 5 requires approximately 20–24 core hours (number of processor units times duration of computation) for a one-way coupled high-fidelity RANS aerodynamic analysis and FEM thermal and structural analyses. The majority of the cost is in the RANS calculation.

The RBDO problem with an independent constraint on each component from (4.3) can be rewritten using a performance measure approach as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & E[f(x,y)] \\ \text{s.t.} \quad & z_i(x;\bar{p}_{f,i},y) > \bar{z}_i \quad i=1,\ldots,m \\ & Ax \leq b. \end{aligned} \tag{4.5}$$

The series system RBDO problem from (4.4) can be rewritten using a performance measure approach as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & E[f(x,y)] \\ \text{s.t.} \quad & z_i(x;\bar{p}_f,y) > \bar{z}_i \quad i=1,\ldots,m \\ & Ax \leq b. \end{aligned} \tag{4.6}$$

The only difference between the performance measure approach formulations in (4.5) and (4.6) is that individual component failure probabilities $\bar{p}_{f,i}$ are prescribed in (4.5) and a system failure probability $\bar{p}_f$ is prescribed in (4.6). For both, a constraint is associated with each limit state, where $z_i(x;\bar{p}_f,y)$ or $z_i(x;\bar{p}_{f,i},y)$ is the limit state level resulting from an inverse reliability analysis. The right-hand side of each constraint $\bar{z}_i$ is the failure level for the ith limit state. Note that the componentwise formulation requires m inverse reliability analyses which can become expensive for many limit states or a system with coupled limit states whereas the system formulation requires a single inverse reliability analysis. However, the componentwise formulation has typically been used due to the lack of an inverse reliability analysis that can solve for multiple limit states [140]. The AKIRA algorithm introduced in chapter 3 overcomes this challenge and allows the system formulation to be solved instead.

4.2 Traditional Methods for Accounting for Uncertainties

Acknowledging that there are uncertainties in the both the model and environment (*i.e.* choice of design variables, random parameters etc.) and properly identifying them is a critical first step in the design process [76]. Indeed the deterministic optimization problem from (4.1) is an approximation of the more general RBDO problem that accounts for uncertainties from (4.4). Nevertheless, a good deterministic design should also somehow take uncertainties into account. To do so, engineers often build margin into a design to address uncertainties, whether they be unimagined loading scenarios, rare manufacturing defects, subpar material properties, or other situations typically unaccounted for in a deterministic mathematical model. Such margins take a variety of forms and can be applied in different ways. Here, several common techniques used to apply margin in a deterministic fashion to a design problem are recounted including safety factors, partial safety factors, and basis values (a statistical method for estimating conservative material properties). It is standard practice to apply

multiple types of margin in design; for example Acar et al demonstrate that the standard practice use of basis values, safety factors, and certification tests in aircraft design lead to a compounding effect of margin in design, ultimately resulting in an estimated probability of failure (per flight) of approximately 10^{-7} [2].

4.2.1 Safety Factors

A common approach for building margin into a design is to use a safety factor [107, 139, 89, 38]. The safety factor γ is the ratio of resistance to load, therefore values greater than 1 build margin into a design. Consider the following constraint which governs failure for a load $g(x, y)$ (for example: stress) and resistance $\bar{z}(y)$ (for example: yield strength):

$$g(x, y) < \bar{z}(y). \tag{4.7}$$

The safety factor $\gamma = \gamma(y)$ is defined for a given design x as

$$\gamma = \gamma(y; x) = \frac{\bar{z}(y)}{g(x, y)}, \tag{4.8}$$

where it is clear that the safety factor is itself a random quantity with a dependence on y. However for ease of use a deterministic value of the safety factor $\tilde{\gamma}$ is often specified during design. Then the stochastic load-resistance constraint from (4.7) is replaced with a deterministic constraint, typically using the mean of the random parameters y:

$$\tilde{\gamma} g(x, E[y]) < \bar{z}(E[y]). \tag{4.9}$$

One option for a deterministic safety factor is the central safety factor [56, 38] given by

$$\gamma_c = \frac{E_y[\bar{z}(y)]}{E_y[g(x, y)]}. \tag{4.10}$$

However, the central safety factor is not generally used in conventional design as it is preferable to use a safety factor assuming a high load and low resistance [56]. Thus, a more pragmatic safety factor is the general safety factor [38] defined in terms of the fractiles of $g(x, y)$ and $\bar{z}(y)$:

$$\gamma_{p,q} = \frac{\bar{z}_p}{g_q}, \tag{4.11}$$

where

$$P[\bar{z}(y) < \bar{z}_p] = p \tag{4.12a}$$

$$P[g(x, y) < g_q] = q. \tag{4.12b}$$

Typical values for p and q are 0.05 and 0.98 [38].

As shown by the safety factor's dependence on y in (4.8), the value of γ should be informed by the uncertainties present in the problem. The value of the safety factor is also dependent on the definition of the resistance [38]. For example, the constraint $g(x,y)^2 < \bar{z}(y)^2$ defines the same load-resistance relationship as (4.7) but requires a safety factor of γ^2 instead of γ to be equivalent to (4.9). Typical values of γ range from 1.2 to 1.5 or higher depending on the application and discipline. In aerospace engineering, the standard value of 1.5 is commonly used in structural applications[2]. Its origin is debatable, but likely arose from a combination of factors ranging from the ultimate to yield strength ratio of alloyed aluminum to flight experience, design codes, and engineering judgement around the 1930s [139].

We emphasize again that safety factors should only be used when there is statistical evidence to back up their value for the relevant application [38]. Unfortunately this seems not to be the case in general engineering education: several safety factor methods such as the Pugsley method [89] or Likel method [73] taught in engineering courses and used to calculate a safety factor are based on qualitative judgements of the uncertainties in the problem (accuracy of material properties, loads, stress calculation, manufacturing quality etc.), and the impact of design failure on humans and the economy. Such methods may appear to agree with typical values found in engineering codes, however they lack objectivity and mathematical rigor.

Probabilistic Sufficiency Factor

Qu and Haftka reintroduced a closely related concept to the safety factor for design, the *probabilistic sufficiency factor* [104]. First note that a chance constraint can be formulated in terms of the safety factor γ as

$$P[\gamma(y;x) < 1] \leq \bar{p}_f. \tag{4.13}$$

The probabilistic sufficiency factor p_{sf} is defined implicitly as

$$P[\gamma(y;x) < p_{sf}] = \bar{p}_f. \tag{4.14}$$

In other words, p_{sf} is the safety factor that yields the prescribed probability of failure $\bar{p}_f$. Qu and Haftka argue that recasting the design problem in terms of probabilistic sufficiency factors gives a more intuitive interpretation of how design changes impact reliability. For example when $p_{sf} = \alpha$ for $0 < \alpha < 1$, it is immediately clear that decreasing the load $g(x,y)$ by $100\alpha\%$ or increasing the resistance by $100(1/\alpha - 1)\%$ will approximately satisfy the reliability requirement corresponding to $\bar{p}_f$ [104].

[2] Although it should be used with care in light of the comments above. As noted by Shanley in 1961: "The 1.5 factor of safety...is not 'sacred'." [118]

4.2.2 Partial Safety Factors

The concept of partial safety factors is used widely in a variety of engineering codes, particularly within the structural reliability community [38, 11]. Let y be the random variables representing the uncertainty in the design problem. In particular a chosen deterministic x_i may correspond to the mean of the random variable y_i; thus y_i describes the variability of the deterministic variable x_i. y_i may also be unrelated to the design variables and for instance represent a random parameter in the design instead.

The idea of partial safety factors is to build margin into a deterministic design by analyzing the "corners" of the design space to ensure failure does not occur. For example each random input variable to the design can be fixed at either a code-specified lower characteristic value or upper characteristic value for example $x_i/\gamma_{m,i}$ or $x_i\gamma_{f,i}$ where $\gamma_{m,i} > 1$ and $\gamma_{f,i} > 1$. Each combination of characteristic values for the input variables can then be analyzed to approximate a reliable design. Figure 4.1 illustrates the partial safety factor concept for a 2D design space consisting of design variables x_1 and x_2.

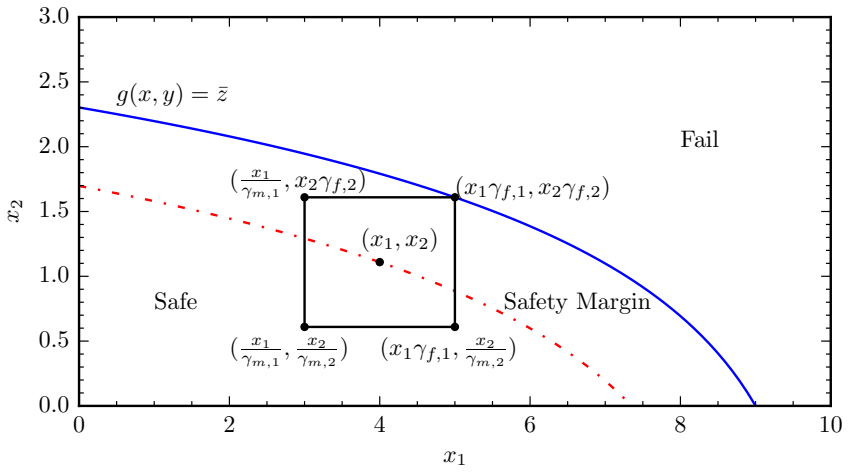


Figure 4.1: Illustration of the partial safety factor concept for a limit state $g(x, y) = \bar{z}$ in two dimensions with two design variables x_1 and x_2 that define the means of two random parameters y_1 and y_2. The partial safety factors $\gamma_{m,1}$, $\gamma_{m,2}$, $\gamma_{f,1}$, and $\gamma_{f,2}$ define a box around the current design point (x_1, x_2) that encompasses most of the variability in the design at the design point. Any design for which the four corners of the box (in two dimensions) remain inside the safe region has sufficient reliability according to the code defining the partial safety factors. Note that partial safety factors only guarantee increased reliability when the safe region is convex.

Analyzing every combination of characteristic values may be expensive; for n variables there are 2^n combinations of characteristic values. As a result usually only a subset is analyzed. The subset choice may be aided by physical trends in a given application such as monotonic relationships

between an input variable and a response (for example, increasing thickness will yield increasing strength, etc.) [38].

To be useful in assessing reliability, the partial safety factors $\gamma_{m,i}$ and $\gamma_{f,i}$ should be informed by both the application and variability in the problem. In structural design codes, the values of partial safety factors are indeed standardized and calibrated for a variety of different structures and problems [38]. However, generalization of the partial safety factors for a variety of cases over the entire design domain can lead to unnecessarily conservative designs [11]. In addition, although partial safety factors may ensure a specified probability of failure for the cases used in calibration, in general they are only an approximation for other similar designs. For application to complex systems, non-standard designs, or designs which lack statistically-informed and calibrated partial safety factors the partial safety factor method leads to uncertain margin in the design.

4.2.3 Basis Values

Basis values are a common technique used in aerostructural design to build in margin to a design to account for variability in material properties. A *basis value* b is simply a lower bound on the $100(1-p)$ percentile of a true material property y with confidence c [30]. The reason for specifying a confidence bound for a material property's percentile is that in practice the true distribution, and therefore percentile, of a material property is unknown since it is only approximated by the samples used to determine the material property statistics. Thus distribution parameters estimated from a finite number of material characterization tests are intrinsically random variables. As a result, a basis value is a conservative underestimate of the likely true material property value. Note however that although basis values are often taken as conservative deterministic material properties, they are in fact random variables.

Mathematically a basis value is defined such that the basis value will be less than a fraction p of the unknown population y with probability c. In the case of a normally distributed random variable, the basis value b takes the form

$$b = \bar{y} - k_{p,c}(N)s(y), \tag{4.15}$$

where $\bar{y}$ is the sample average and $s(y)$ is the sample standard deviation of y. The "knock-down" factor $k_{p,c}(N)$ depends on the percentile $100p$, degree of confidence c, and number of samples N used to estimate the mean and standard deviation of the distribution. The two most commonly used basis values are A and B basis values [30] and are defined in Table 4.1 with specified values of p and c.

When *perfect information* is assumed, *i.e.* the distribution of the material property is presumably known exactly via an infinite number of test samples, a basis value is simply the $100(1-p)$th percentile of the material property. Otherwise, the basis value can be calculated by solving

$$P[L_{1-p} > \bar{y} - k_{p,c}s(y)] = c \tag{4.16}$$

Table 4.1: Common basis values used in practice.

Basis	p	c	Purpose
A	0.99	0.95	critical component
B	0.90	0.95	redundant component

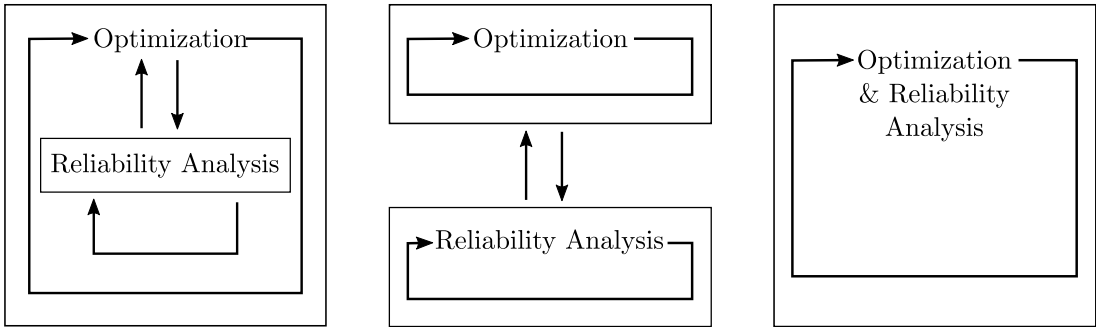


Figure 4.2: Schematic showing the three main classes of reliability-based design optimization methods. *Left:* In a traditional double-loop method the reliability analysis is nested inside of an outer optimization loop, which is typically an expensive process requiring many function evaluations. *Middle:* In a sequential method the optimization and reliability analyses are decoupled and serially iterated until convergence. *Right:* In a unilevel method the reliability analysis is decomposed into a set of conditions, for example equivalent KKT conditions, which are appended to the optimization problem as constraints. No iteration is necessary.

for $k_{p,c}(N)$ where L_{1-p} is the lower $100(1-p)$ percentile of the estimated distribution of y. Takano outlines a solution method for (4.16) [127] for a general distribution. Alternately, A and B basis values are tabulated for normal, lognormal, and Weibull distributions in [30].

4.3 Survey of RBDO Methods

RBDO methods are typically separated into three categories, *double-loop*, *sequential*, and *unilevel* methods [3]. Figure 4.2 gives a brief schematic of each method and they are described in more detail in subsequent sections. In a double-loop method the reliability analysis is nested within an outer optimization. In a sequential method the reliability analyses and optimizations are decoupled and a sequence of reliability analyses and deterministic optimizations are solved. Finally in a unilevel method, the reliability analysis is embedded directly into the optimization as a set of constraints and additional design variables; for example FORM may be reformulated in terms of its equivalent KKT conditions. Typically double-loop methods are the most expensive due to their nested structure, whereas sequential methods and unilevel methods are much more efficient [3, 40].

Another method for making the solution of RBDO problems more efficient is to use surrogate-based optimization directly for the probabilistic quantities. In this method, a design of experiments

is first conducted and reliability analyses are conducted for each sample. Then a response surface is constructed for the probabilistic constraints, for example kriging. Deterministic optimization is then performed over the response surfaces. This is the approach used by Acar for optimizing the reliability of an aircraft's wing and tail system [1]. A similar approach is used by Missoum et al with support vector machine and kriging surrogates for the design of an airfoil under flutter [87]. Qu and Haftka also consider surrogate-based RBDO using the probabilistic sufficiency factor since it is typically easier to approximate with a response surface than a probability of failure [104].

Several authors have also investigated the use of evolutionary or genetic algorithms in RBDO. Hao et al present a double-loop method using particle swarm optimization [67]. They identify clusters of promising designs with a low-fidelity model and then determine an optimal design from a kriging response surface built over samples from a high-fidelity model. Alternately Lobato et al consider the use of the differential evolution and multiobjective differential evolution algorithms for RBDO [83]. They take probabilistic constraints into account via a penalty function and generate Pareto frontiers for the objective and the reliability. Naturally the use of such algorithms is more expensive then even a standard double-loop RBDO method, however the benefits for practical design include Pareto frontiers and the generation of different options for designs with similar reliability.

There is also considerable interest in incorporating multifidelity methods into RBDO, at least beyond response surface methods and surrogate-based optimization, however the field is still growing [101]. Hao et al present a multifidelity RBDO method used in the design of stiffened shells and while it appears to work well for their application, there is no guarantee of convergence for their disparate models of differing fidelity [67].

4.3.1 Double-loop RBDO

Reliability-based design optimization has been traditionally approached using a *double-loop* strategy (also called *nested*) in which an outer loop runs the overall optimization and inner loops solve the reliability analysis problem for each probabilistic constraint [128]. Although the double-loop procedure is intuitive it can become prohibitively expensive – every time the optimization algorithm in the outer loop needs to query a probabilistic constraint a new reliability analysis must be performed. In addition, gradient-based optimization requires sensitivity analyses. However, the reliability analysis and associated sensitivity analysis themselves are usually expensive. For example in a sampling-based method a large number of function evaluations may be required to resolve a probability of failure. In a most probable point (MPP) method the reliability analysis is also itself formulated as an optimization problem.

Most double-loop RBDO methods use MPP-based methods such as FORM or SORM for the reliability analysis (see section 2.3.1) [128, 8, 41, 67]. The primary reason for doing so is that MPP-based methods are relatively inexpensive compared to other methods, they have a well-developed theoretical background, and sensitivities can be easily obtained at the MPP as a byproduct of

the MPP search. A notable limitation for most MPP-based methods is their inability to find to multiple MPPs in more complex problems [117]. Another limitation of the First Order Reliability Method (FORM) is its approximation of the limit state with a hyperplane which will yield incorrect reliability estimates for nonlinear limit states [128, 83]. On the other hand the use of the Second Order Reliability Method (SORM) can yield better limit state approximations but requires second derivative information which is typically not available for limit states. Section 2.3.1 discusses MPP reliability analysis methods in more detail.

Several strategies have been explored in order to ameliorate the cost associated with double-loop RBDO while still maintaining the double-loop structure. Two notable examples are the use of response surfaces [42, 67, 98] and the augmented reliability space [42].

Replacing exact limit state functions with a response surface during the reliability analysis in the inner loop can reduce the cost of the double-loop procedure. A popular method is to use kriging surrogates for the limit states [42, 67, 98] and then estimate probabilities of failure directly from the kriging surrogates. The choice of the kriging surrogate (as opposed to polynomial, neural network etc.) is typically motivated by the ability of the surrogate to provide information, the kriging variance, that can be used to adaptively refine the surrogate. Section 2.4.2 and chapter 3 discuss the use of kriging surrogates for reliability analysis in more detail.

Dubourg et al also propose an interesting double-loop RBDO concept using the *augmented reliability space* [42]. The augmented reliability space combines the space of design variables $x \in \mathbb{R}^n$ and random parameters $y \in \mathbb{R}^p$; the design variables x are said to augment the uncertainty in the random variables y. Dubourg et al then build a global kriging model for each limit state over the augmented reliability space and optimize on the surrogates using a double-loop RBDO method. They use a quasi-SQP optimization algorithm and subset simulation for the reliability analysis. At each step of the SQP algorithm the kriging models are refined as necessary to meet an accuracy requirement. Although the method is a double-loop algorithm it derives efficiency from the use of the augmented reliability space and the kriging response surface and compares well to equivalent FORM-based double-loop RBDO.

4.3.2 Sequential RBDO

Sequential methods (also called *decoupled* or *serial-loop*[3]) for RBDO were introduced to reduce the inefficiencies with a double-loop procedure. The key idea is to separate the solution of the RBDO problem into a series of reliability analyses and deterministic optimizations. The separation is typically made by approximating probabilistic constraints at a point, for example the most probable point (MPP) of failure. At each iteration in the series, the approximation is updated.

Most sequential RBDO methods use MPP methods for the reliability analysis [4, 14, 40, 5]. In

[3]The word *single-loop* has also been used in the literature to refer to sequential RBDO methods. Here we side with the majority and reserve single-loop for unilevel RBDO.

this case the separation of the original RBDO problem is typically obtained via the approximation of the location of the MPP during each deterministic optimization. For example Agarwal et al considered the approximation of the MPP with a first order Taylor series [4, 3] and Ba-abbad et al considered the approximation of the MPP with a scaling of the reliability index [14].

Approximation of the limit states instead of the MPP can also lead to a decoupling of the reliability analyses and optimization. For example Wu et al propose approximating limit states by fixing the MPPs at each iteration and shifting the limit state value to promote constraint feasibility [132]. In contrast, Du and Chen propose the Sequential Optimization and Reliability Assessment (SORA) method. Here they similarly consider approximation of the limit state failure boundaries by fixing the MPPs each iteration, but they shift instances of the problem's random variables instead [40]. When limit states are nonlinear, Du and Chen's approach may be more efficient as it retains the nonlinearity of the limit states in its approximation.

On the other hand, Zou and Mahadevan consider directly approximating chance constraints with a first order Taylor series at the current design point [140]. Their approach differs from others in that it uses a reliability index approach (forward reliability analysis) instead of performance measure approach (inverse reliability analysis). The use of a forward reliability analysis allows a greater variety of reliability analysis methods to be used. In addition the authors note that by doing so, RBDO can be performed with system failure probabilities. We note that the AKIRA method presented in chapter 3 also enables RBDO with system failure probabilities and at the same time yields the benefits of an inverse reliability analysis.

Royset et al present another interesting alternative for achieving a sequential RBDO method using semi-infinite optimization (a semi-infinite optimization problem is one with a finite number of variables but an infinite number of constraints) [111]. They present a reformulation of three types of RBDO problems into deterministic semi-infinite optimization problems. The sequential RBDO method therefore consists of a series of reliability analyses and deterministic semi-infinite optimization problems which can be solved using the Polak-He algorithm. The semi-infinite optimization procedure consists of replacing each chance constraint with functions that match the minimum limit state within a subset of the random space. By controlling the size of the subsets the sequence of deterministic optimizations and reliability analyses is directed towards an optimum, although the authors do not give a convergence proof. The proposed method is exact for limit state functions that are affine in the standard normal space for a given design x.

Many of the above approaches are shown to work well on a variety of test problems, however they lack a proof of global convergence [40, 4, 140]. Incorporating trust region methods is a simple next step to ensure convergence (see section 4.4) [5]. In addition, most methods concentrate on RBDO for individual limit state reliability as opposed to system reliability. The remainder of this section presents details for some of the sequential RBDO methods mentioned above.

Shift in Limit State

Wu et al proposed a sequential RBDO method where the deterministic approximation of the limit states includes a shift to promote convergence towards feasibility for the probabilistic constraints [132]. Consider a single limit state $g(x,u) \in \mathbb{R}$. Wu et al proposed replacing $g(x,u)$ in each deterministic optimization with $g(x,u_{MPP})$ where u_{MPP} is the MPP corresponding to the limit state shifted by a *design shift factor* s. s is determined from

$$P[g(x^k,u)+s \leq \bar{z}] = \bar{p}_f, \tag{4.17}$$

where $\bar{z}$ is the failure threshold for limit state $g(x,u)$, $\bar{p}_f$ is the desired probability of failure, and x^k are the deterministic design variables corresponding to the kth iteration. The effect of s is to shift the mean value of $g(x,u)$ but not otherwise modify the shape of the distribution. s may be readily calculated by reformulating (4.17) in quantile form and using a MPP method or other reliability analysis method. The corresponding quantile constraint for use during the deterministic subproblem is

$$g(x,u_{MPP}) \geq \bar{z}, \tag{4.18}$$

where u_{MPP} is the MPP corresponding to the shifted limit state. When $s=0$, the current design meets the reliability requirement exactly. Application to multiple limit states requires treating each limit state separately and finding the MPP for each limit state. Shortcomings of the proposed method include the application to deterministic design variables only, lack of a convergence proof, and reliance on MPP methods, in addition to difficulties with approximating nonlinear limit states with a simple shift.

Shift in Variables: SORA

Du and Chen proposed the Sequential Optimization and Reliability Assessment (SORA) approach for RBDO where the deterministic approximation of the limit states includes a shift in the design variables as opposed to the limit state [40]. In this way limit state nonlinearities can be better taken into account. In addition to the shift in design variables s, the random parameters u are replaced with their MPP estimates u_{MPP} from an inverse reliability analysis with prescribed failure probability $\bar{p}_f$. Thus the corresponding quantile constraint in a SORA formulation for the single limit state $g(x,u) \in \mathbb{R}$ is

$$g(x-s,u_{MPP}) \geq \bar{z}, \tag{4.19}$$

where u_{MPP} is the location in the standard normal space of the MPP for the limit state. The shift $s \in \mathbb{R}^n$ is defined as

$$s_i = \begin{cases} x_{c,i} - u_{MPP,j} & \text{if design variable } x_i \text{ corresponds to the mean of } u_j, \\ 0 & \text{otherwise,} \end{cases} \tag{4.20}$$

where $x_{c,i}$ is the ith component of the initial design variables at the start of the deterministic subproblem. The effect of the shift is to push the deterministic variables corresponding to a distribution mean a distance s_i away from the MPP during the solution of the deterministic subproblem, thereby encouraging convergence towards a feasible design. Clearly, SORA is able to handle uncertainties in the design variables. However for application to multiple constraints, it requires finding the MPP for each limit state. Due to its formulation, it is usually not applied to system reliability problems. SORA shows improved performance compared to double-loop methods and appears to be one of the more robust RBDO algorithms [11].

SORA for Series Systems

Ba-abbad et al consider an extension of SORA where they demonstrate application to system reliability problems by considering the optimal apportionment of failure probability among different limit states [14]. To do so they consider the failure probability contribution from each limit state to be an additional design variable in the optimization problem. Their modified RBDO problem is

$$\begin{aligned} \underset{x,\bar{\beta}_i}{\text{minimize}} \quad & f(x) \\ \text{s.t.} \quad & g_i(x, u_{MPP,i}(\bar{\beta}_i)) \geq \bar{z}_i \quad i = 1 \dots m \\ & \sum_{i=1}^{m} \Phi(-\bar{\beta}_i) \leq \bar{p}_f, \end{aligned} \tag{4.21}$$

where $\bar{\beta}_i$ is the minimum allowable reliability index for the ith limit state, $u_{MPP,i}$ is the location in the standard normal space of the most probable point (MPP) of failure for the ith limit state, and $\bar{p}_f$ is the prescribed probability of failure. The lack of a shift s on the design variables implies they are deterministic, although this is not a necessary assumption.

Since evaluating the constraints $g_i(x, u_{MPP,i}(\bar{\beta}_i))$ requires an inverse reliability analysis for a prescribed value of $\bar{\beta}_i$, Ba-abbad et al approximate the location of the MPP as

$$u_{MPP,i}(\bar{\beta}_i) \approx \frac{\bar{\beta}_i}{\beta_{c,i}} u_{MPP,i}(\beta_{c,i}), \tag{4.22}$$

where $\beta_{c,i}$ is the reliability index corresponding to a specific design x_c. Since the approximation is valid only for designs close to x_c, the authors use a trust region method to manage the optimization

which enables provable convergence (see section 4.4).

Ba-abbad et al show their modified SORA algorithm can generate improved designs over methods that treat each limit state separately. By considering the system failure probability some limit states may have a higher chance of failure than others, but the objective can be reduced more. Note however that the proposed approach still requires finding the MPP for each limit state, and in addition increases the size of the optimization problem by the inclusion of additional variables. The method also naturally relies on MPP methods which may be inaccurate for nonlinear limit states, especially in higher dimensions. Finally Ba-abbad et al make use of Ditlevsen's first order upper bound which approximates the upper bound on system failure probability as

$$p_f \approx \sum_i^m p_{f,i}, \tag{4.23}$$

where p_f is the system failure probability, and $p_{f,i}$ is the failure probability associated with the ith limit state. Note that the upper bound will result in a conservative (larger) estimate of system failure probability if the failure regions of different limit states have significant overlap.

4.3.3 Unilevel RBDO

In a *unilevel* method (also called *unified*, *single-shot*, or *one-level*)[4], the reliability analysis is formulated in such a way that it is embedded directly into the optimization problem statement. Such an approach can readily be taken for reliability analysis methods that are formulated themselves as optimization problems, namely for most probable point (MPP) based methods such as FORM and SORM. The optimization problem solved in the MPP-based reliability analysis is replaced with its KKT conditions (see section C.1) which themselves replace the probabilistic constraints in the RBDO problem.

Kuschel and Rackwitz derived the forms of equivalent unilevel optimizations for RBDO problems using a FORM reliability analysis for both minimum cost and maximum reliability problems [77]. Their results on structural test problems show the superior efficiency of unilevel RBDO methods compared to double-loop methods. Later, Agarwal presented a unilevel method for RBDO using FORM and solving the inverse reliability problem [3]. Agarwal shows that in order for the unilevel problem to be equivalent to the original double-loop RBDO problem the limit state should be pseudoconvex in its random parameters y, *i.e.* that the limit state increases along all directions with a positive directional derivative. Due to the inclusion of the KKT conditions in the optimization problem, the size of the unilevel RBDO problem is increased (the original design variables plus Lagrange multipliers and MPP locations are now included as design variables in addition in the constraints form the KKT conditions). However, solving the problem can still be more efficient than

[4]The use of the phrase "single-loop" has also been used to describe unilevel RBDO methods, however this appears to be in the minority as single-loop predominantly refers to a sequential RBDO method.

a double-loop method [77, 3].

Shan and Wang also present a unilevel method for RBDO [117]. They recast the RBDO problem as an equivalent deterministic optimization problem over the so-called Reliable Design Space (RDS), the space where all the reliability constraints are satisfied. This space is naturally a subset of the feasible space for the deterministic optimization obtained by fixing the random variables in probabilistic constraints at their mean values. However, identifying the reliable design space can be non-trivial. Shan and Wang propose using the KKT conditions for the MPP search problem (i.e. FORM) to derive an expression for points in the reliable design space as a function of partial derivatives of the limit state functions. Then the analytic form of the reliable design space is plugged into the analytic formulae for the limit states and massaged to obtain a mathematical description of the limit state in the reliable design space. As such, the method does not apply to implicitly defined limit states. However, when such manipulations can be done the result is a deterministic optimization problem which can be efficiently solved once and for all.

4.4 Trust Region Methods

The key idea in a trust region method is to solve a series of less expensive optimization subproblems within a trust region using a locally accurate model such as a corrected surrogate model or a low-fidelity model. The *trust region* represents the region of confidence in the accuracy of the local model. As a result, it constrains how far away from the initial point or *trust region center* to search for a solution to the subproblem. Once the solution of the subproblem is obtained, a decision is made to accept or reject the solution as the next iterate and shrink, expand or retain the trust region size. Convergence to a local minimum of the original optimization problem is guaranteed when the locally accurate model is first-order consistent (matches function value and gradients with the original objective and constraints) and the trust region size and location is updated appropriately[5] [6, 31]. Figure 4.3 illustrates a trust region method applied to a simple nonlinear 1D optimization problem.

Trust region methods are frequently used in multifidelity optimization and a *trust region model management* (TRMM) framework can be used to manage multiple models of different fidelity to ensure convergence to an optimum of the highest-fidelity model [101]. By relying on a lower-fidelity model during the solution of the trust region subproblems, considerable increases in efficiency may be achieved [48], particularly when the objective and constraints are expensive to evaluate and there is a good correlation between the low- and high-fidelity models. An overview of a basic trust region optimization algorithm for two model fidelities is given in Algorithm 4.

There are many guidelines for constructing a successful trust region optimization algorithm and the remainder of this section discusses available choices and practicalities. Algorithm 4 showcases

[5] A sufficient decrease condition and appropriate trust region logic is required. In addition the standard assumptions on continuous derivatives and a bounded objective and Hessian are required to ensure convergence.

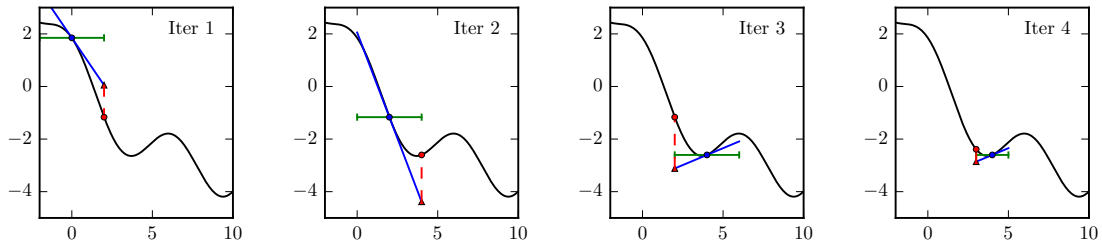


Figure 4.3: Schematic showing the idea behind a trust region optimization method. The first four iterations of a simple 1D minimization for nonlinear function $f(x)$ (solid black line) are shown using a trust region method. The trust region center is marked with a blue dot and the extent (size) of the trust region is shown with the green bar. A low-fidelity model, in this case a first-order Taylor series expansion at the trust region center, is shown in blue. At each iteration a minimization is performed on the low-fidelity model within the trust region. At the conclusion of each subproblem the high-fidelity model is evaluated at the optimum of the low-fidelity model (red dot) and compared with the low-fidelity objective (red triangle). Since progress is made with the high-fidelity objective in the first two iterations, the trust region location is updated. However in iteration 3, the low-fidelity model is a poor predictor of the high-fidelity model and as a result the subproblem optimum is rejected and the trust region size is shrunk as shown in iteration 4. As shown in this simple example, by relying on optimization over a (corrected) low-fidelity model convergence to a local minimum of the high-fidelity model can be achieved.

some of the choices such as the type of correction for the low-fidelity model, the form of the subproblem to be solved, and trust region acceptance/rejection and size update logic. In addition we note that the accuracy and correlation of the low-fidelity model, as well as the selection of the initial trust region size can have a large impact on the efficiency of the algorithm (although theoretically it should not impact whether convergence is ultimately achieved [31]). In a practical implementation, a minimum allowable trust region size should also be specified. Finally careful management of constraint infeasibilities is required as the trust region may shrink such that no feasible point is contained within the trust region. In such a situation, the inclusion of a *feasibility restoration phase* where infeasibilities are minimized until the subproblem becomes feasible again is necessary [31]. Such a phase is tacitly assumed in the subproblem solution step of algorithm 4.

Algorithm 4: Simple bifidelity trust region optimization algorithm

Data: initial design x^0, initial trust region size Δ^0

Result: optimal design x^*

$f_{hi}(x^0)$, $g_{hi}(x^0) \leftarrow$ Evaluate high-fidelity objective and constraints at trust region center x^0;

for $k \leftarrow 0$ **to** k_{max} **do**

  $\nabla_x f_{hi}(x^k)$, $\nabla_x g_{hi}(x^k) \leftarrow$ Evaluate high-fidelity objective and constraint gradients at trust region center x^k;

  if *converged* **then**

    return x^k;

  Calculate $f_{lo}(x^k)$, $g_{lo}(x^k)$, $\nabla_x f_{lo}(x^k)$, $\nabla_x g_{lo}(x^k)$;

  Construct corrected low-fidelity model $\hat{f}^k(x)$, $\hat{g}^k(x)$;

  $x^* \leftarrow$ Solve approximate subproblem:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \hat{f}^k(x) \\ \text{s.t.} \quad & \hat{g}^k(x) \geq 0 \\ & \left\| x - x^k \right\| \leq \Delta^k \end{aligned}$$

  $f_{hi}(x^*)$, $g_{hi}(x^*) \leftarrow$ Evaluate high-fidelity objective and constraints at subproblem optimum x^*;

  accepted $\leftarrow$ Decide whether x^* should be accepted or rejected as the next iterate;

  if x^* *accepted* **then**

    $x^{k+1} \leftarrow x^*$;

  else

    $x^{k+1} \leftarrow x^k$;

  $\Delta^{k+1} \leftarrow$ Update the trust region size (shrink, expand, or retain?);

4.4.1 Application to Multifidelity Models

A major motivation for using trust region methods in design optimization is their straightforward application to different fidelity models. Alexandrov was among the first to generalize classical trust region methods to multifidelity models [6]. In doing so she took advantage of the global convergence properties of trust region methods and applied them to a general model management framework [7].

In many physics-based applications, a variety of computational models are available, some more accurate (and usually more computationally expensive) than others. For example, in computational fluid dynamics available models range from the simplest area-averaged 1D Navier Stokes equation

(easily solved in seconds) to the numerical solution of the Euler equations (minutes), Reynolds-averaged Navier Stokes (RANS) equations (hours), Large Eddy Simulation (LES) (days), and Direct Numerical Simulation (DNS) (weeks). By leveraging lower-fidelity models in a multifidelity trust region optimization like that outlined above in Algorithm 4 much of the optimization work can be offloaded to faster but less accurate models, and the expensive high-fidelity models need only be brought in periodically to ensure convergence. In chapter 5, the design of a supersonic nozzle using a variety of multifidelity models is explored in both a deterministic and reliability-based design optimization context.

4.4.2 Subproblem Formulation

A variety of approximate subproblems can be formulated. For example in SQP trust region methods, the objective of the approximate subproblems takes the form of a quadratic approximation to the Lagrangian and the constraints are linearized. Augmented Lagrangian trust region methods use a subproblem featuring an augmented Lagrangian objective and no constraints; other modifications add linearized constraints in an attempt to enhance convergence [48]. Eldred and Dunlavy advocate a *direct surrogate* approach where the original objective and constraints are replaced with surrogate functions thereby avoiding the need for penalty and multiplier updates and leading to greater simplicity and potentially increased efficiency in the case of a predictive surrogate model [48]. The direct inclusion of the typically nonlinear surrogate constraints can also be more helpful at avoiding infeasible iterates. The original problem and the approximate direct surrogate subproblem are

Original Formulation:		kth Subproblem Formulation:	
$\underset{x}{\text{minimize}}$	$f(x)$	$\underset{x}{\text{minimize}}$	$\hat{f}^k(x)$
s.t.	$g(x) \geq 0,$	s.t.	$\hat{g}^k(x) \geq 0$
			$\left\|x - x_c^k\right\| \leq \Delta^k,$

(4.24)

where $f(x)$ denotes the objective function and $g(x)$ denotes the nonlinear constraints. $\hat{f}^k(x)$ and $\hat{g}^k(x)$ denote the corrected surrogate functions for the objective and constraints. Note the additional restriction on the design space through the use of a trust region constraint where the center of the trust region is x_c^k for the kth iteration and it has size Δ^k. A typical choice for the norm is the ℓ_∞ norm due to its easy implementation during solution of the subproblems.

4.4.3 Low-fidelity Correction

For provable convergence to a local minimum, the low-fidelity model's objective and constraints must be corrected to match both the function values and gradients of the high-fidelity model at the trust region center x_c at the start of each subproblem. First order and second order multiplicative,

additive, and mixed corrections have been investigated in the literature [48, 49]. A typical procedure for deriving a model correction includes first choosing the form of the correction δ and then approximating it at the trust region center, for example via a Taylor series which ensures first or second order consistency. The following derivations closely follow Eldred et al [49].

Additive Correction

Let $\delta_a(x)$ be an additive correction defined as

$$\delta_a(x) = f_{hi}(x) - f_{lo}(x). \tag{4.25}$$

Let $\hat{\delta}_a$ be the second order Taylor series approximation of $\delta_a(x)$ at the trust region center x_c given by

$$\hat{\delta}_a = \delta_a(x_c) + \nabla\delta_a(x_c)^\top(x - x_c) + \frac{1}{2}(x - x_c)^\top\nabla^2\delta_a(x_c)(x - x_c), \tag{4.26}$$

where $\delta_a(x_c)$ is defined in (4.25) and $\nabla\delta_a(x)$ and $\nabla^2\delta_a(x)$ are obtained from the differentiation of (4.25). Thus the low-fidelity model with the second order additive correction $\hat{f}_a$ can be written as

$$\hat{f}_a(x) = f_{lo}(x) + \delta_a(x_c) + \nabla\delta_a(x_c)^\top(x - x_c) + \frac{1}{2}(x - x_c)^\top\nabla^2\delta_a(x_c)(x - x_c) \tag{4.27}$$

for a given trust region center x_c.

Multiplicative Correction

A multiplicative correction $\delta_m(x)$ is defined as

$$\delta_m(x) = \frac{f_{hi}(x)}{f_{lo}(x)}. \tag{4.28}$$

Following the same steps as for the additive correction yields the low-fidelity model with a second order multiplicative correction $\hat{f}_m(x)$ as

$$\hat{f}_m(x) = f_{lo}(x)\left(\delta_m(x_c) + \nabla\delta_m(x_c)^\top(x - x_c) + \frac{1}{2}(x - x_c)^\top\nabla^2\delta_m(x_c)(x - x_c)\right). \tag{4.29}$$

Two-point Correction

Applying a correction that combines the additive and multiplicative corrections may be desirable as doing so affords an additional degree of freedom. For example, in addition to achieving first order consistency at the trust region center, a function value can be matched at another point, say the previous iterate. The low-fidelity model with a multipoint correction can be written as

$$\hat{f}(x) = \gamma\hat{f}_a(x) + (1 - \gamma)\hat{f}_m(x), \tag{4.30}$$

where γ is a scalar weight. Simple algebraic manipulations show that

$$\gamma = \frac{f_{hi}(x_p) - \hat{f}_m(x_p)}{\hat{f}_a(x_p) - \hat{f}_m(x_p)} \tag{4.31}$$

enforces a matched function value at the point x_p.

Second Order Information

In most situations second order information is not available and thus calculating $\nabla^2\delta_a(x)$ and $\nabla^2\delta_m(x)$ exactly may be prohibitively expensive. In such situations quasi-Newton approximations may be useful for enhancing the accuracy of the correction. Such approximations update an approximate Hessian with previously obtained gradient information. For example, the common Broyden-Fletcher-Goldfarb-Shanno (BFGS) update is

$$H_{k+1} = H_k - \frac{H_k s_k s_k^\top H_k}{s_k^\top H_k s_k} + \frac{y_k y_k^\top}{y_k^\top s_k}, \tag{4.32}$$

where H_k is the approximation of the Hessian at iteration k, $s_k = x^{k+1} - x^k$ is the step in design variables between iterations, and $y_k = \nabla f(x^{k+1}) - \nabla f(x^k)$. The Symmetric Rank 1 update may also be used. Eldred et al recommend initializing the Hessian approximation with a scaled version of the identity matrix and skipping the Hessian update when $y_k^\top s_k$ is close to zero to avoid numerical difficulties [49]. In the case that the iterate x^{k+1} is rejected by the trust region method, Bryson and Rumpfkeil recommend still updating the Hessian approximation with information at the rejected point, but using a reversed step instead, *i.e.* $s_k = x^k - x^{k+1}$ [22].

4.4.4 Iterate Acceptance

At the conclusion of each subproblem a decision is made to accept or reject the new iterate pending on whether sufficient progress has been made toward the solution. This decision has a direct impact on the convergence of the trust region algorithm and can stall or otherwise lead to lack of convergence if not managed appropriately. Two common techniques for gauging iterate acceptance for a general constrained optimization problem are using a trust region ratio or using a filter.

Trust Region Ratio

The trust region ratio from trust region methods for unconstrained optimization can be used for constrained problems provided a suitable merit function is used. The trust region ratio η^k for the kth iterate is

$$\eta^k = \frac{\Phi_{hi}(x_c) - \Phi_{hi}(x^*)}{\Phi_{lo}(x_c) - \Phi_{lo}(x^*)}, \tag{4.33}$$

where $\Phi_i(\cdot)$ is the merit function for model i. The trust region ratio η^k in (4.33) measures the ratio of realized improvement in the high-fidelity model over the predicted improvement from the low-fidelity model. A value of η^k close to one shows the low-fidelity model is a good predictor of the high-fidelity model in the current trust region Δ^k whereas a value of η^k less than zero implies the high-fidelity model has actually regressed in the past iteration and the candidate optimum x^* should be rejected.

In trust region methods for unconstrained optimization the merit function $\Phi(\cdot)$ is the objective function [109], however in constrained optimization constraint feasibility must also be taken into account. Penalty or augmented Lagrangian functions can be used for this purpose [109, 48]. For example, a typical choice of penalty merit function is

$$\Phi(x) \equiv f(x) + \frac{1}{2}\rho g^-(x)^\top g^-(x) \tag{4.34}$$

where $f(x)$ is the objective function, ρ is a positive penalty parameter, and $g^-(x)$ is the set of active constraints (the $-$ superscript denotes retention of the negative portion of the constraints $g(x) \geq 0$ only, implying such constraints are active). Note that the optimal value of the penalty parameter ρ is *a priori* unknown and must be estimated. A standard procedure is to tie increases of the penalty parameter to the outer iteration count of the trust region method [48]. More information on the augmented Lagrangian function can be found in Appendix C.10 and requires, in addition to the estimation of a penalty parameter ρ, an estimate of the Lagrange multipliers λ for the optimization problem.

With a suitable selection of merit function and the trust region ratio defined in (4.33), typical logic for accepting or rejecting the current iterate proceeds as follows: if the trust region ratio $\eta \leq 0$ then reject the iterate ($x^{k+1} \leftarrow x^k$), otherwise accept the iterate ($x^{k+1} \leftarrow x^*$).

Filter

Filter methods are an alternative for managing the convergence of an optimization algorithm and were first introduced by Fletcher in 1996 [53]. They borrow the concept of Pareto dominance from multiobjective optimization and apply it to the multiobjective goal in constrained optimization of decreasing both the objective and constraint infeasibility. Filter methods have been successfully applied in SQP trust region optimization [52] as well as unconstrained optimization [64] and are competitive with other trust region methods [31].

Consider a set of pairs $(f(x), v(x))$ for different points x where $f(x)$ is the objective value and $v(x)$ is a measure of the constraint violation at x (*e.g.* $\|g^-(x)\|$, for some chosen norm; ℓ_1, ℓ_2, and ℓ_∞ norms are common choices). For two points x and y, point x is considered to *dominate* point y if $f(x) < f(y)$ and $v(x) < v(y)$. A *filter* is a set of objective-constraint violation pairs such that each point is not dominated by any other point in the filter. Figure 4.4 illustrates the concept of a

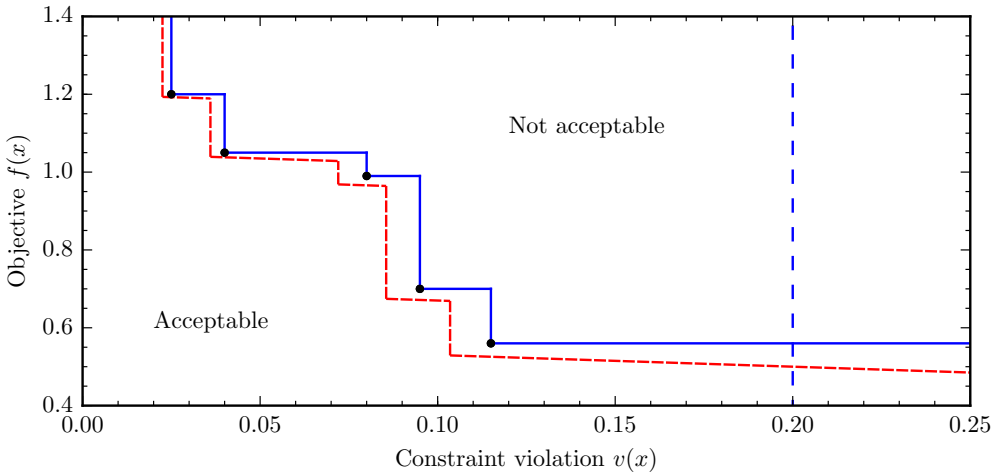


Figure 4.4: Illustration of a filter used in trust region optimization methods. Iterates are considered for acceptance via a filter consisting of objective-constraint violation pairs $(f(x), v(x))$. In the example here, five points (black dots) have been included in the filter (solid blue lines). A point is acceptable to the filter if it is not dominated by any points in the filter, *i.e.* it lies to the lower left of the filter. Fletcher et al introduced an envelope for the filter (exaggerated on the plot and shown in dashed red lines) to help ensure global convergence [52]. The filter is typically initialized with a lower limit on constraint violation, shown here as the dashed blue line at $v(x) = 0.20$.

filter. With appropriate updating of the filter and enforcement of acceptance/rejection conditions based on the filter, a globally convergent trust region algorithm can be obtained [52].

To obtain a globally convergent filter SQP trust region method, Fletcher et al consider several modifications: 1) a sloping filter envelope, 2) a sufficient reduction criterion, and 3) a feasibility restoration phase [53].

A sloping envelope on the basic filter can be used to ensure convergence to infeasible points is avoided. In particular, Fletcher et al [53] consider an iterate x^{k+1} with objective f^{k+1} and constraint violation v^{k+1} to be acceptable if

$$v^{k+1} \leq \alpha v^f, \text{ or } f^{k+1} \leq f^f - \gamma v^{k+1} \tag{4.35}$$

for all f^f and v^f in the filter. $0 < \alpha, \gamma < 1$ are constants with typical values $\gamma \approx 10^{-5}$ and $\alpha = 1 - \gamma$. The sloping envelope for the filter is compared with a standard filter in Figure 4.4.

A *sufficient reduction* criterion is also required to ensure that the objective value converges. The criterion (called a *switching condition*) specifies that the realized reduction in objective must be greater than a certain percentage of the predicted reduction in objective when the goal of the subproblem is to reduce the objective as opposed to decreasing constraint violation. Thus when

$f_{lo}(x^*) < f_{lo}(x_c)$, the switching condition

$$f_{hi}(x_c) - f_{hi}(x^*) \geq \sigma \left(f_{lo}(x_c) - f_{lo}(x^*) \right) \tag{4.36}$$

should be checked for $0 < \sigma < 1$ where one can observe the similarity with an unconstrained trust region ratio. A typical value of σ is 0.1.

Finally, Fletcher et al consider a *feasibility restoration* phase which is entered when the subproblem solution is infeasible. A typical reason for an infeasible solution to a subproblem is a trust region that is too small and thereby excludes any feasible region for the problem's constraints. In such a situation Fletcher et al successively minimize the constraint violations until the subproblem becomes feasible and the trust region algorithm can proceed.

4.4.5 Trust Region Update

The decision to shrink, expand or retain a trust region's size between iterations is closely coupled with the decision to accept or reject a trust region which was discussed in section 4.4.4. Typically, a merit function $\Phi(\cdot)$ which accounts for the objective and constraints is used to calculate a trust region ratio. As a reminder, the trust region ratio η^k for the kth iterate is

$$\eta^k = \frac{\Phi_{hi}(x_c) - \Phi_{hi}(x^*)}{\Phi_{lo}(x_c) - \Phi_{lo}(x^*)}. \tag{4.33}$$

The value of η^k can be used to make a decision about updating the trust region. When $\eta^k < 0$, the low-fidelity and high-fidelity models do not match well, and the merit function of the high-fidelity model has actually increased if the low-fidelity model predicted a decrease. Therefore shrinking the trust region is appropriate. On the other hand, when $\eta^k \approx 1$ the models match well and expanding the trust region is appropriate. For $0 < \eta^k < 1$, the trust region may be shrunk or expanded, depending on proximity to 0 or 1. Finally for $\eta^k > 1$, it may be most appropriate to retain the trust region size even though the models do not match well since the high-fidelity merit function actually decreased more than expected. Algorithm 5 outlines typical trust region update logic. Note that the trust region size has an upper limit Δ_{max} and is also limited in expansion when the optimal point x^* lies in the interior of the trust region Δ^k. Such safeguards have been found to be beneficial in practice.

Algorithm 5: Basic trust region update logic

Data: *rejected* flag, trust region ratio η^k, current trust region size Δ^k, maximum allowable size Δ_{max}

Result: new trust region size Δ^{k+1}

if *step rejected* **then**
  $\Delta^{k+1} \leftarrow 0.25\Delta^k$;
else if $\eta^k < 0.25$ **then** // poor fit
  $\Delta^{k+1} \leftarrow 0.25\Delta^k$;
else if $\eta^k < 0.75$ **then** // good fit
  $\Delta^{k+1} \leftarrow \min\{1.5 \left\| x_c - x^* \right\|, \Delta^k\}$;
else if $\eta^k < 1.25$ **then** // excellent fit
  $\Delta^{k+1} \leftarrow 1.5 \left\| x_c - x^* \right\|$;
else // okay fit but good improvement
  $\Delta^{k+1} \leftarrow \min\{1.5 \left\| x_c - x^* \right\|, \Delta^k\}$;
if $\Delta^{k+1} > \Delta_{max}$ **then**
  $\Delta^{k+1} \leftarrow \Delta_{max}$;
return Δ^{k+1};

4.5 Sequential RBDO via Anchored Decomposition

Section 4.3.2 presented a variety of sequential RBDO strategies each of which decomposed the RBDO problem into a series of deterministic optimizations. In this section *anchored decomposition* is used to separate the deterministic design variables x and random parameters y, thereby enabling deterministic approximation of chance and quantile constraints. Lower-fidelity models are also incorporated into the constraint approximation, which leads to a multifidelity approach for sequential RBDO.

First the ANalysis Of VAriance (ANOVA) decomposition is introduced, which has close ties with the anchored decomposition but is generally more expensive to calculate. Next anchored decomposition is introduced followed by its multifidelity adaptation and application to chance and quantile constraints. Finally practicalities for the use of the anchored decomposition approximation in RBDO are addressed and a comparison is made with existing sequential RBDO methods.

4.5.1 ANOVA Decomposition

The ANalyis Of VAriance (ANOVA) decomposition has a long pedigree with widespread use in the design of experiments and global sensitivity analysis literature [12, 105, 123]. It considers the

decomposition of the square integrable function $f(x)$ for $x \in \mathbb{R}^n$, whose domain is the unit cube $[0,1]^n$. $f(x)$ can be decomposed as

$$f(x) = f(x_1, \ldots, x_n) = f_0 + \sum_{i=1}^{n} f_i(x_i) + \sum_{1 \le i \le j}^{n} f_{ij}(x_i, x_j) + \ldots + f_{12\ldots n}(x_1, \ldots, x_n). \tag{4.37}$$

where $f_{i_1 \ldots i_s}$, $1 \le s \le n$, are defined by the relationship

$$\int_0^1 f_{i_1 \ldots i_s} \mathrm{d}x_k = 0 \text{ for } k = i_1, \ldots, i_s. \tag{4.38}$$

The relationship in (4.37) uniquely defines the $f_{i_1 \ldots i_s}$ [122]. For example, integrate (4.37) over the unit cube to obtain

$$\int_0^1 f(x) \mathrm{d}x = f_0. \tag{4.39}$$

Similarly integrate over all variables except x_i:

$$\int_0^1 f(x) \prod_{k \ne i} \mathrm{d}x_k = f_0 + f_i(x_i). \tag{4.40}$$

Next, integrate over all variables except x_i and x_j:

$$\int_0^1 f(x) \prod_{k \ne i,j} \mathrm{d}x_k = f_0 + f_i(x_i) + f_i(x_j) + f_{ij}(x_i, x_j). \tag{4.41}$$

The process can be repeated until all summands in (4.37) are defined. Note that due to (4.38), all the terms $f_{i_1 \ldots i_s}$ are orthogonal. Unfortunately due to the integration over $f(x)$, their calculation may be expensive.

The truncated decomposition from (4.37) can be used as an approximation for $f(x)$. For example, ignoring all but the first order terms yields:

$$f(x) \approx \hat{f}(x) = f(x_1, \ldots, x_n) = f_0 + \sum_{i=1}^{n} f_i(x_i). \tag{4.42}$$

The error δ incurred by such an approximation is defined as

$$\delta \equiv \frac{1}{D} \int \left(f(x) - \hat{f}(x) \right)^2 \mathrm{d}x, \tag{4.43}$$

where $D = \int f^2(x) \mathrm{d}x - f_0^2$ is the total variance of $f(x)$. When $\hat{f}(x)$ is selected to be the zeroth order truncated ANOVA decomposition, *i.e.* $\hat{f}(x) = f_0$, then $\delta = 1$. Thus an approximation with error $\delta \ll 1$ is deemed to be good. Sobol' derives the error in terms of the Sobol' indices $S_{i_1 \ldots i_s}$ [122] which are described in detail in Appendix E. The error for a truncated ANOVA decomposition of

order $L < n$ is given by

$$\begin{aligned}\delta &= 1 - \sum_{s=1}^{L} \sum_{i_1<\cdots<i_s}^{n} S_{i_1\ldots i_s} \\ &= \sum_{s=L+1}^{n} \sum_{i_1<\cdots<i_s}^{n} S_{i_1\ldots i_s},\end{aligned} \tag{4.44}$$

which follows since the sum of all the Sobol' indices is 1. Thus the error is equal to the sum of the Sobol' indices of order $L+1$ and higher. Recall the Sobol' indices are global sensitivity indices and measure the contribution of input variables x_i and their combinations to the variability of a function $f(x)$ (refer to Appendix E). For the first order approximation in (4.42), if first order effects dominate $f(x)$ then $S_{i_1\ldots i_s}$ will be small for $s > 1$, and the approximation will be good. However the use of the truncated ANOVA decomposition for function approximation is limited due to the need to estimate the terms $f_i(x_i)$ etc. via integration. Performing accurate integrations for high-dimensional functions may be excessively costly.

4.5.2 Anchored Decomposition

The anchored decomposition is more convenient than the ANOVA decomposition for function approximation due to its direct construction from $f(x)$ rather than from integrals over $f(x)$ which may be difficult to obtain. Unlike the ANOVA decomposition, which integrates out dependence on variables x_i, the anchored decomposition removes dependence on variables x_i by freezing the value of the variables at an anchor a. The anchored decomposition (also referred to as cut-HDMR for High Dimensional Model Reduction [105]) for a function $f(x)$ with respect to the anchor a has the same form as the ANOVA decomposition:

$$f(x) = f(x_1, \ldots, x_n) = f_0 + \sum_{i=1}^{n} f_i(x_i) + \sum_{1\le i\le j}^{n} f_{ij}(x_i, x_j) + \cdots + f_{12\ldots n}(x_1, \ldots, x_n). \tag{4.37}$$

Let $f^{i_1 i_2 \ldots i_s}$ denote the function $f(x)$ with all input variables x_i whose $i \notin \{i_1 i_2 \ldots i_s\}$ are fixed at the anchor a_i, $1 \le s \le n$. For example $f^i(x_i) = f(a_1, a_2, \ldots a_{i-1}, x_i, a_{i+1}, \ldots a_n)$. The constituent

functions from (4.37) are then defined as [105]:

$$\begin{aligned}
f_0 &= f(a), \\
f_i(x_i) &= f^i(x_i) - f_0, \\
f_{ij}(x_i, x_j) &= f^{ij}(x_i, x_j) - f_i(x_i) - f_j(x_j) - f_0, \\
&\dots \\
f_{i_1 \dots i_s}(x_{i_1}, \dots, x_{i_s}) &= f^{i_1 \dots i_s}(x_{i_1}, \dots, x_{i_s}) - \sum_{i_1 < \dots < i_{s-1} \subset \{i_1 \dots i_s\}} f_{i_1 \dots i_{s-1}}(x_{i_1}, \dots x_{i_{s-1}}) - \dots \\
&\quad - \sum_i f_i(x_i) - f_0.
\end{aligned}$$

With the exception of f_0, each term $f_{i_1,\dots,i_s}(x_{i_1}, \dots, x_{i_s})$ has the property that when any component x_i is set to the value of its corresponding anchor a_i, the function value is zero.

The truncated anchored decomposition can be similarly used as an approximation of $f(x)$. However, unlike the ANOVA decomposition, its error δ depends on the choice of the anchor a as well as the order of the approximation L. Wang showed that the ANOVA decomposition is optimal for a given order L with respect to the error measure δ defined in (4.43) [131]. As a result, the anchored decomposition cannot be globally more accurate than the ANOVA decomposition. Sobol' apparently derived a uniform error bound for the anchored decomposition of order L in terms of the partial derivatives of $f(x)$, assuming continuous mixed partial derivatives [131]. However in the general engineering design situation, calculation of the error bound may be difficult. Here we present Wang's result for the expected error of the first order anchored decomposition approximation, $L = 1$, assuming a uniformly distributed random anchor over $[0, 1]^n$ [131]:

$$E[\delta] = \sum_{s=2}^{n} b_s \sum_{i_1 \dots i_s} S_{i_1 \dots i_s}, \tag{4.45}$$

where $b_s = s^2 - s + 2$ and $S_{i_1 \dots i_s}$ are Sobol' indices (refer to Appendix E). (4.45) shows the expected error will be small if the Sobol' indices of order two and higher are small and if the dimension of the input variables is small. Unfortunately if the dimension is even moderately high, say $s = 10$, then $b_{10} = 92$ and a 10th order Sobol' index even as large as 0.01 may induce unacceptable error ($\delta \geq 0.92$). Of course, Wang's derivation is for a random anchor and measures a global error. In most cases, the anchor is selected to be the only point at which information about $f(x)$ has been obtained and only a locally accurate approximation of $f(x)$ is required.

In this dissertation, the use of anchored decomposition is primarily considered with separating two sets of variables, the deterministic design variables x and the random parameters y. Since there are only two sets of variables, the anchored decomposition truncated to the first order is used since the full anchored decomposition is only of the second order. Following (4.37) the first order

approximation of $f(x, y)$ is

$$f(x, y) \approx f(x_c, y) + f(x, y_c) - f(x_c, y_c), \tag{4.46}$$

where the anchor $a = (x_c, y_c)$. The decomposition in (4.46) contains a stochastic component $f(x_c, y)$, a deterministic component $f(x, y_c)$, and a constant $f(x_c, y_c)$.

4.5.3 Multifidelity Adaptation

The use of trust region methods motivates the infusion of multifidelity models into the anchored decomposition approximation of $f(x, y)$ from (4.46). The idea is to replace the terms in the approximation with lower-fidelity models $f_{lo}(x, y)$ and thereby rely on the trust region machinery introduced in section 4.4 to ensure convergence to an optimum of the high-fidelity model $f_{hi}(x, y)$. In particular the low-fidelity model must be corrected to ensure first order consistency, *i.e.* matching function values and gradients with the high-fidelity model at the center of the trust region. Thus the anchor for the deterministic design variables x is taken to be the trust region center x_c. Choices for the anchor of the random parameters y such as $E[y]$ or the most probable point of failure are discussed in section 4.5.6.

Consider the first order anchored decomposition approximation $\hat{f}(x, y)$ for a single model fidelity:

$$\hat{f}(x, y) = f(x_c, y) + f(x, y_c) - f(x_c, y_c). \tag{4.47}$$

Inspection of (4.47) shows that it is already first order consistent at the anchor (x_c, y_c) with respect to x and thus no correction term is needed.

When two model fidelities are available for a quantity of interest, $f_{lo}(x, y)$ and $f_{hi}(x, y)$, an approximation for $f_{hi}(x, y)$ can be written as

$$\hat{f}(x, y) = f_{lo}(x_c, y) + f_{lo}(x, y_c) - f_{lo}(x_c, y_c). \tag{4.48}$$

However, (4.48) is not first order consistent with the high-fidelity model at the trust region center. The additive, multiplicative, or mixed first or second order corrections described in section 4.4.3 can be applied to (4.48) to ensure first order consistency. To derive the form of the first order additive bifidelity correction first apply the first order correction from (4.25) to the low-fidelity model for the deterministic design variables x only. The corrected model is

$$\tilde{f}(x, y) = f_{lo}(x, y) + f_{hi}(x_c, y) - f_{lo}(x_c, y) + \left(\nabla_x f_{hi}(x_c, y) - \nabla_x f_{lo}(x_c, y)\right)^\top (x - x_c)\,. \tag{4.49}$$

Next apply the anchored decomposition approximation from (4.46) to the low-fidelity model. Canceling terms yields the following corrected low-fidelity model approximation $\hat{f}(x, y)$:

$$\hat{f}(x,y) = f_{hi}(x_c, y) + f_{lo}(x, y_c) - f_{lo}(x_c, y_c) + \left(\nabla_x f_{hi}(x_c, y) - \nabla_x f_{lo}(x_c, y)\right)^\top (x - x_c)\,. \tag{4.50}$$

Similar corrected models can be derived using the multiplicative or mixed first or second order corrections. The bifidelity function approximation in (4.50) can be substituted for the original function in the same way as the single-fidelity function approximation in (4.47) during the solution of an optimization subproblem within a trust region method. Note the additional requirement in the bifidelity case of calculating $f_{hi}(x_c, y)$ and the gradients $\nabla_x f_{hi}(x_c, y)$ and $\nabla_x f_{lo}(x_c, y)$ at the trust region center, all of which in fact are random. The use of (4.50) in chance and quantile constraints for RBDO is discussed further in sections 4.5.4 and 4.5.5.

4.5.4 Chance Constraints

Simplifying approximations for the chance constraints in a RBDO problem are derived here using the corrected multifidelity anchored decomposition approximations from (4.47) and (4.50). These approximations may be substituted for the original chance constraints in order to form a deterministic subproblem managed by a trust region optimization method. The following derivation uses a simplified notation where dependence on the fixed anchor points x_c and y_c has been dropped from function arguments. For example $g_{lo}(x)$ refers to $g_{lo}(x, y_c)$, $g_{lo}(y)$ refers to $g_{lo}(x_c, y)$, and g_{lo} refers to $g_{lo}(x_c, y_c)$ etc.

Recall a chance constraint is formulated as:

$$P[g(x,y) \le \bar{z}] \le \bar{p}_f, \tag{2.13}$$

where $\bar{z}$ are the limit state failure levels and $\bar{p}_f$ is the prescribed system failure probability. When two models of varying fidelity are available for the limit state $g(x, y)$, (2.13) may be approximated using (4.50):

$$\begin{aligned} P[g(x,y) \le \bar{z}] &\approx P[\hat{g}(x,y) \le \bar{z}] \\ &= P[g_{hi}(y) + g_{lo}(x) - g_{lo} + \left(\nabla_x g_{hi}(y) - \nabla_x g_{lo}(y)\right)^\top (x - x_c) \le \bar{z}] \\ &= P[g_{hi}(y) + \left(\nabla_x g_{hi}(y) - \nabla_x g_{lo}(y)\right)^\top (x - x_c) \le \bar{z} - g_{lo}(x) + g_{lo}]. \end{aligned} \tag{4.51}$$

Evaluating (4.51) requires an integration over the random space whose result varies with x. Such an approximation would appear to be prohibitively expensive as x changes throughout the solution of each subproblem and therefore requires multiple integrations. However notice that the random functions in the integrand in (4.51) are not functions of x due to the anchored decomposition approximation. As a result, if for example a sampling method such as Monte Carlo is used to perform

the integration, the sampling only need occur once for the design associated with the trust region center x_c. The samples $y^{(i)}$ and their responses $g_{hi}(x_c, y^{(i)})$, $\nabla_x g_{hi}(x_c, y^{(i)})$, and $\nabla_x g_{lo}(x_c, y^{(i)})$ can be saved and then used to efficiently calculate (4.51) for different values of x during the subproblem solution. In such a scenario, the cost of using the chance constraint approximation in (4.51) is a reliability analysis for $g_{hi}(x_c, y)$ at the trust region center x_c, sensitivity analyses for both $g_{hi}(x_c, y)$ and $g_{lo}(x_c, y)$ at the trust region center x_c, and then evaluation of the low-fidelity model $g_{lo}(x, y_c)$ during the subproblem solution. Thus the total cost of the subproblem solution is equivalent to a single reliability analysis and a single deterministic optimization.

When only a single-fidelity model is used ($g_{lo} = g_{hi} = g$), (4.51) requires no gradient calculations and simplifies to

$$P[\hat{g}(x, y) \leq \bar{z}] = P[g(y) \leq \bar{z} - g(x) + g] \leq \bar{p}_f, \tag{4.52}$$

which requires a reliability analysis for the limit state $g(x_c, y)$ at the trust region center and deterministic evaluations of the limit state $g(x, y_c)$ throughout the solution of the subproblem.

Aside from the shortcomings of using chance constraints in RBDO as compared to quantile constraints (refer to section 2.1.5 for an in-depth discussion), using the chance constraints from (4.51) or (4.52) in gradient-based RBDO requires estimation of $\nabla_x p_f(x)$. Although expressions were derived in section 2.6 it may be more convenient to use the quantile constraints shown in the next section.

4.5.5 Quantile Constraints

Simplifying approximations for quantile constraints in a RBDO problem are derived here using the corrected multifidelity anchored decomposition approximations from (4.47) and (4.50). In preparation for the derivation we emphasize the difference between the limit state function $g(x, y)$ and the limit state level $z(\bar{p}_f, x; y)$. The latter differs from the former in that it describes a level set of the limit state $g(x, y)$ for a prescribed probability of failure $\bar{p}_f$. In other words, the equation $g(x, y) = z(\bar{p}_f, x; y)$ defines a level set. The limit state level $z(\bar{p}_f, x; y)$ is obtained from an inverse reliability analysis (see section 2.1.5 and the AKIRA algorithm in chapter 3).

When multiple limit states are present $g(x, y)$ and $z(\bar{p}_f, x; y)$ represent a vector of values. Note that in general $g(x_0, y_0) \neq z(\bar{p}_f, x_0; y)$. The previous equation is only true for the limit state $g_i(x, y)$ which at the point (x_0, y_0) obtains the value $z_i(\bar{p}_f, x_0; y)$.

For notational convenience, let

$$z(\bar{p}_f, x; y) \equiv z(x), \tag{4.53}$$

where dependence on the fixed prescribed probability of failure $\bar{p}_f$ is understood. The procedure for deriving a simplified version of a quantile constraint is to first approximate the limit state level for the high-fidelity model with a corrected version of the low-fidelity model's limit state level as shown in (4.50). Then the low-fidelity limit state level is approximated with a corrected version of

the low-fidelity limit state as shown in (4.50).

Recall a quantile constraint for the high-fidelity model is formulated as

$$z_{hi}(x) \geq \bar{z}, \tag{2.15}$$

where $\bar{z}$ are the limit state failure levels and $z_{hi}(x)$ are the limit state levels for the high-fidelity model corresponding to design x such that failure occurs with probability $\bar{p}_f$. Replace $z_{hi}(x)$ with the corrected limit state levels for the low-fidelity model at the point x_c:

$$z_{hi}(x) \approx z_{lo}(x) - z_{lo}(x_c) + z_{hi}(x_c) + \left(\nabla_x z_{hi}(x_c) - \nabla_x z_{lo}(x_c)\right)^\top (x - x_c). \tag{4.54}$$

The approximation in (4.54) may be used in place of the original quantile constraint in (2.15) however it still requires recurring inverse reliability analyses for the low-fidelity model as well as associated sensitivity analyses during the solution of the subproblem which may be expensive. Instead $z_{lo}(x)$ can be further approximated by an anchored decomposition approximation of the low-fidelity model $g_{lo}(x, y_c)$. The approximation is the same as before, namely

$$z_{lo}(x) \approx g_{lo}(x, y_c) - g_{lo}(x_c, y_c) + z_{lo}(x_c) + \left(\nabla_x z_{lo}(x_c) - \nabla_x g_{lo}(x_c, y_c)\right)^\top (x - x_c). \tag{4.55}$$

Plugging (4.55) into (4.54) and simplifying yields the following approximation for $z_{hi}(x)$:

$$z_{hi}(x) \approx g_{lo}(x, y_c) - g_{lo}(x_c, y_c) + z_{hi}(x_c) + \left(\nabla_x z_{hi}(x_c) - \nabla_x g_{lo}(x_c, y_c)\right)^\top (x - x_c). \tag{4.56}$$

The approximation in (4.56) can be verified to satisfy first order consistency and may also be derived by skipping the intermediate step and approximating $z_{hi}(x)$ directly with $g_{lo}(x, y_c)$. Thus given a single high-fidelity inverse reliability analysis at the trust region center x_c and an associated sensitivity analysis, an approximate deterministic constraint can be written as

$$g_{lo}(x, y_c) - s - \nabla_x s^\top (x - x_c) \geq \bar{z}, \tag{4.57}$$

where s plays the role of an additive safety margin and is given by

$$s \equiv g_{lo}(x_c, y_c) - z_{hi}(x_c), \tag{4.58a}$$

$$\nabla_x s \equiv \nabla_x g_{lo}(x_c, y_c) - \nabla_x z_{hi}(x_c). \tag{4.58b}$$

4.5.6 Choice of Anchor Point

Although the obvious choice for the anchor of x is the trust region center x_c, the choice of anchor for the random variables y is less obvious. Most sequential RBDO methods choose to approximate the limit state at the most probable point of failure (MPP) or approximate the MPP itself. However

when multiple limit states are present doing so requires finding the MPP for each limit state. In addition if limit states are dependent, as in the case of coupled quantities of interest, then potentially m^2N or greater function evaluations are required where m is the number of limit states and N is the average number of evaluations required to find a MPP. Thus using a single shared anchor point can be advantageous for decreasing RBDO expense. Obvious choices include the mean or median. In this section we discuss the choice of the mean and MPP as anchor points for quantile constraints.

Mean

When the anchor point is chosen to be the mean, *i.e.* $y_c = E[y]$, the safety margin s in the approximate deterministic quantile constraint becomes

$$s = g_{lo}(x_c, E[y]) - z_{hi}(x_c). \tag{4.59}$$

In this case s determines the distance in limit state value between the anchor point and the levels obtained from the inverse reliability analysis at the trust region center.

Most Probable Point (MPP) of Failure

When the anchor point is chosen to be the most probable point (MPP) of failure, the safety margin s in the approximate deterministic quantile constraint becomes

$$s = g_{lo}(x_c, y_{MPP}) - z_{hi}(x_c). \tag{4.60}$$

In the case that only one model fidelity is considered, *i.e.* $g_{hi} = g_{lo} = g$ and $y_c = y_{MPP}$, then the safety margin s is zero, and likewise its gradient. Thus, the approximate deterministic quantile constraint simply becomes

$$g(x, y_{MPP}) \geq \bar{z}, \tag{4.61}$$

which follows since $g(x_c, y_{MPP}) = z(x_c)$ by construction, *i.e.* the MPP(s) reside on the limit state surface(s) determined by the inverse reliability analysis. Note that (4.61) is valid for any point y on the limit state surface defined by $z(x_c)$, not just y_{MPP}, although the MPP is a convenient choice with a largely developed theoretical background.

As in other sequential RBDO methods [132, 40, 14], when multiple limit states are present, it is more appropriate to select the MPP associated with each limit state as the anchor point for that particular limit state, instead of selecting the global MPP as the anchor point for each limit state approximation. In such a case, each limit state requires deterministic evaluation at its own MPP during an optimization subproblem. When many limit states are present and they are also dependent, evaluation of m limit states for each of m MPPs requires m^2 function evaluations for each set of constraint evaluations which may be quite expensive.

4.5.7 Algorithm

Algorithm 6 outlines the proposed sequential RBDO method using anchored decomposition and two model fidelities. The presented algorithm specifically addresses the standard RBDO problem from (4.4) with a deterministic or expectation objective and quantile constraints, however a reliability objective can be included in the same manner as the quantile constraints. Design variables x are deterministic but may parameterize the means of distributions of the random parameters y. In this case, the SORA shift in design variables from (4.20) is used to encourage convergence [40]. The AKIRA algorithm presented in chapter 3 is used for the system inverse reliability analysis each iteration. The associated gradient estimation was discussed in section 3.5. At each iteration a deterministic subproblem is solved using corrected anchored decomposition approximations of the low-fidelity model. The entire algorithm is embedded in a trust region method to ensure convergence to an optimum of the high-fidelity model. Both hard and soft convergence are assessed for termination and are described in detail in Appendix C. If the constraints are feasible and the norm of the reduced gradient is sufficiently small then hard convergence is achieved. If the previous three iterations have been accepted by the trust region method but progress has not been made towards decreasing the objective or constraint infeasibility then soft convergence is achieved.

Algorithm 6: Multifidelity sequential RBDO with anchored decomposition approximations

Data: initial design x_c^0, initial trust region size Δ^0, choice of anchor point (mean, MPP or other), choice of additive/multiplicative/mixed first or second order correction, limit state threshold $\bar{z}$

Result: optimal design x^*

Solve system inverse reliability analysis for $z_{hi}(x_c^0)$ using AKIRA;
Evaluate $f_{hi}(x_c^0)$;
for $k \leftarrow 0$ **to** k_{max} **do**
  Set anchor y_c^k for random parameters;
  Evaluate $\nabla_x z_{hi}(x_c^k)$;
  Evaluate $\nabla_x f_{hi}(x_c^k)$;
  if *converged* **then**
    return x_c^k;
  Evaluate f_{lo}, $\nabla_x f_{lo}$, g_{lo}, and $\nabla_x g_{lo}$ at anchor (x_c^k, y_c^k);
  Construct corrected low-fidelity model with SORA shift if necessary $\hat{f}(x), \hat{g}(x)$;
  $x^* \leftarrow$ Solve approximate deterministic subproblem:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \hat{f}(x) \\ \text{s.t.} \quad & \hat{g}(x) \geq \bar{z} \\ & \left\| x - x_c^k \right\| \leq \Delta^k \end{aligned}$$

  Solve system inverse reliability analysis for $z_{hi}(x^*)$ using AKIRA;
  Evaluate $f_{hi}(x^*)$;
  if *no progress made in last 3 iterations* **then**
    return x_c^k;
  Decide whether x^* should be accepted or rejected as the next iterate;
  if x^* *accepted* **then**
    $x_c^{k+1} \leftarrow x^*$;
  else
    $x_c^{k+1} \leftarrow x^k$;
  Update the trust region size (shrink, expand, or retain) for Δ^{k+1};

4.5.8 Comparison with Existing Methods

The sequential RBDO method presented in Algorithm 6 shares many similarities with other sequential RBDO methods. However it also has some notable differences from many methods:

Multifidelity optimization. A lower-fidelity model is used during the solution of the subproblem which potentially decreases the overall cost of the RBDO problem.

Trust Region Model Management. A trust region method is used to ensure convergence to an optimum of the high-fidelity model.

Anchored decomposition. The use of anchored decomposition motivates quantile constraint approximations for points other than the MPP. For example taking the mean as an anchor may have benefits for problems with coupled constraints and is helpful for system RBDO problems.

System inverse reliability analysis. Thanks to AKIRA a *system* RBDO problem can be solved using the more robust inverse reliability analysis.

As discussed in section 4.3.2, most sequential RBDO algorithms are MPP-based and/or use a MPP reliability analysis method. In addition the individual component RBDO problem is usually solved instead of the system RBDO problem. Finally, the use of multifidelity methods in RBDO is still a growing area of research.

As sequential methods often operate on the same principles, the presence of similarities is unsurprising. Here we consider the single-fidelity version of Algorithm 6 which is obtained by taking $f_{hi} = f_{lo}$ and $g_{hi} = g_{lo}$ since most algorithms use single-fidelity. When the anchor is taken as the MPP, Algorithm 6 is essentially the Sequential Optimization and Reliability Assessment (SORA) proposed by Du and Chen [40] (see section 4.3.2). Thus it may be called a SORA method. When the design variables are deterministic and do not parameterize the random variables, the anchor is taken as the MPP, and an additive correction is used for the limit state approximations then Algorithm 6 is also similar to Wu et al's proposed method with a shift in the limit state [132] (see section 4.3.2).

Algorithm 6 is also similar to Ba-abbad et al's proposed RBDO method [14] (see section 4.3.2) with the exception that the most probable point location y_{MPP} is fixed in Algorithm 6 during the solution of the subproblem. Both methods solve a series system RBDO problem. Ba-abbad et al do so by introducing an extra set of variables $\bar{\beta}_i$ into the optimization subproblem with the goal of determining the optimal distribution of probability of failure among different limit states. As a result they optimize the distribution of failure probability among limit states at the same time as optimizing deterministic variables x. In order to optimize the distribution of the failure probability, they allow the MPP for each limit state to change. On the other hand in the present approach, (4.57) shows an optimal distribution of probability of failure at the trust region center x_c is determined *prior* to solving the optimization subproblem. The optimal distribution is determined

Table 4.2: Variables in the short column problem.

Name	Variable	Distribution	Mean	CoV
Moment 1 (kN-m)	M_1	Normal	250	0.3
Moment 2 (kN-m)	M_2	Normal	125	0.3
Axial load (kN)	P	Normal	2500	0.2
Yield strength (MPa)	Y	Normal	40	0.1
Breadth (mm)	b	Normal	x_1	0.1
Height (mm)	h	Normal	x_2	0.1

via an inverse reliability analysis such as AKIRA which takes multiple limit states into account. Then the subproblem is solved assuming fixed MPPs from the inverse reliability analysis.

4.6 Demonstration

The algorithm presented in section 4.5.7 is demonstrated on two common structural reliability-based design optimization problems, the short column problem with a single limit state and 4–6 random variables, and the cantilever beam problem with two limit states and 4 random variables. Results for a realistic supersonic nozzle application are presented in chapter 5. The results presented use a filter method for the acceptance of trust region iterates (see section 4.4.4) with a convergence and constraint tolerance of 10^{-3}–10^{-4}. Various anchors and corrections are investigated. Comparisons with results from literature are made when possible. Finally AKIRA is used to solve the system inverse reliability analysis (chapter 3).

4.6.1 Short Column

Variations of the short column problem are commonly used in the RBDO literature to demonstrate reliability analysis and RBDO methods [77, 111, 3, 11, 42, 83]. The problem concerns a short column with rectangular cross-section subject to two bending moments M_1 and M_2 and an axial load P. In the version considered here [11], the loading and the column's yield strength Y are considered to be independent normal random variables. In addition the breadth b and height h of the column's cross-section are considered to be independent normal random variables; thus $y = [M_1, M_2, P, Y, b, h]^\top$. The deterministic design variables are the means of the geometric parameters; thus $x = [\mu_b, \mu_h]^\top$ with initial values of 500 mm each. Table 4.2 describes the variables in the problem.

The normalized limit state ($g(x, y) \leq 0$ corresponds to failure) which governs yielding in the column is

$$g(x, y) = 1 - \frac{4M_1}{bh^2Y} - \frac{4M_2}{b^2hY} - \left(\frac{P}{bhY}\right)^2. \tag{4.62}$$

The RBDO problem of interest is

$$\begin{aligned} \underset{x=[\mu_b,\mu_h]^\top \in \mathbb{R}^2}{\text{minimize}} \quad & f(x) = x_1 x_2 \\ \text{s.t.} \quad & p_f = P\left[g(x,y) \leq 0\right] \quad \leq 1.35 \times 10^{-3} \\ & 1/2 \leq {}^{x_1}/_{x_2} \leq 2, \end{aligned} \tag{4.63}$$

where the chance constraint with $p_f = 1.35 \times 10^{-3}$ corresponds to a system reliability index $\beta = 3$.

Table 4.3 compares the RBDO results for the short column problem. The proposed sequential method from section 4.5.7 performs comparably to SORA and is more efficient than double-loop RBDO with FORM even when the mean is used as the anchor point. As expected, the use of the mean as the anchor as opposed to the MPP leads to more function evaluations, at least for this problem with a single limit state. This is to be expected since the limit state behavior at the mean is less representative of the behavior in the distribution tails compared to the MPP. However using an improved low-fidelity correction in the trust region method can help offset the decrease in performance when the mean is used as the anchor. For example, the number of function evaluations dropped 8% when a two-point second order correction was used instead of a first order additive correction with a mean anchor.

Figure 4.5 compares the impact of anchor choice on convergence trajectory. When the MPP is used as the anchor only a single iteration is required to get close to the optimal solution, whereas when the mean is used as the anchor, several additional corrective steps must be taken for convergence. However, both methods converge to the same optimum. Figure 4.5b illustrates how the filter acceptance method used in the trust region method allows an increase in constraint violation if the objective is decreased. During the mean anchor optimization the initial step increases the constraint violation by a large amount, but as shown in Figure 4.5a makes progress towards the optimum regardless. The optimization then quickly recovers and converges to the optimum.

Figure 4.6 compares the convergence history for the MPP and mean anchor optimizations using a first order additive correction. The optimization using the MPP anchor converges in just 3 iterations, whereas the mean anchor optimization requires 5 iterations to declare convergence. The use of a two-point second order approximation can further help enhance optimizations using a mean anchor as shown in Table 4.3.

4.6.2 Cantilever Beam

The cantilever beam problem was introduced in section 3.7.2 to demonstrate the performance of AKIRA. Recall there are four independent normal random variables $y = [Y, E, H, V]^\top$ (see Table 3.3) which correspond to the yield stress, elastic modulus, and two random loads. The rectangular cross-section of the cantilever beam is parameterized by deterministic width w and thickness t. The design variables $x = [w, t]^\top$ with initial values $[2.35132, 3.32610]^\top$ [14]. Two limit states are considered,

Table 4.3: Comparison of RBDO results for the short column problem specified in (4.63). The problem is solved without geometric uncertainties, that is with deterministic $\mu_b = b$ and $\mu_h = h$, and with geometric uncertainties where μ_b and μ_h correspond to the means of normal distributions as specified in Table 4.2. Several items of interest are compared in this table: 1) the use of the mean or the MPP as the anchor point for the separable approximation of the limit state, 2) the use of additive 1st order or two-point 2nd order corrections for the limit state approximations, and 3) the results from the proposed method with results from SORA (a sequential method and a close cousin described in section 4.3.2) and inverse FORM with HMV (a popular double-loop method described in sections 4.3.1 and 2.3.1) from Aoues and Chateauneuf's benchmark paper [11]. $f(x^*)$ is the optimal value of the objective function, μ_b^* and μ_h^* are the optimal mean breadth and height of the column's cross section, and N is the number of function evaluations required during the optimization.

Method	$f(x^*)$	μ_b^*	μ_h^*	N
Without geometric uncertainties				
MPP anchor; additive 1st order correction	0.1918	0.3097	0.6194	115
Mean anchor; additive 1st order correction	0.1918	0.3114	0.6160	270
MPP anchor; two-point 2nd order correction	0.1918	0.3097	0.6194	115
Mean anchor; two-point 2nd order correction	0.1918	0.3097	0.6194	248
SORA [11]	0.1915	–	–	119
Inverse FORM w/ HMV [11]	0.1913	–	–	450
With geometric uncertainties				
MPP anchor; two-point 2nd order correction	0.2363	0.3606	0.6554	220
Mean anchor; two-point 2nd order correction	0.2357	0.3459	0.6813	459
SORA [11]	0.2372	–	–	252
Inverse FORM w/ HMV [11]	0.2372	–	–	570

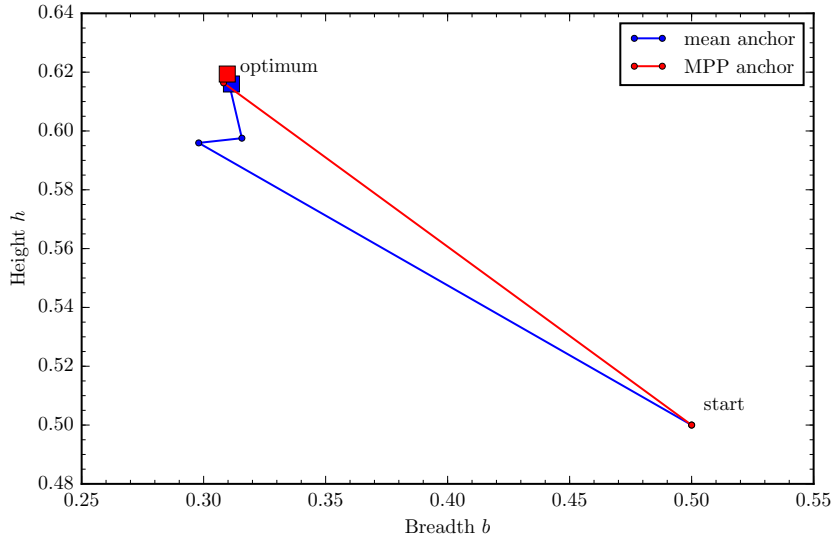
$g_S(x, y)$ which corresponds to failure by excessive stress, and $g_D(x, y)$ which corresponds to failure by excessive deflection. Both are normalized such that $g_i(x, y) \leq 0$ corresponds to failure.

The system RBDO problem follows Ba-abbad's formulation [14] and is given by

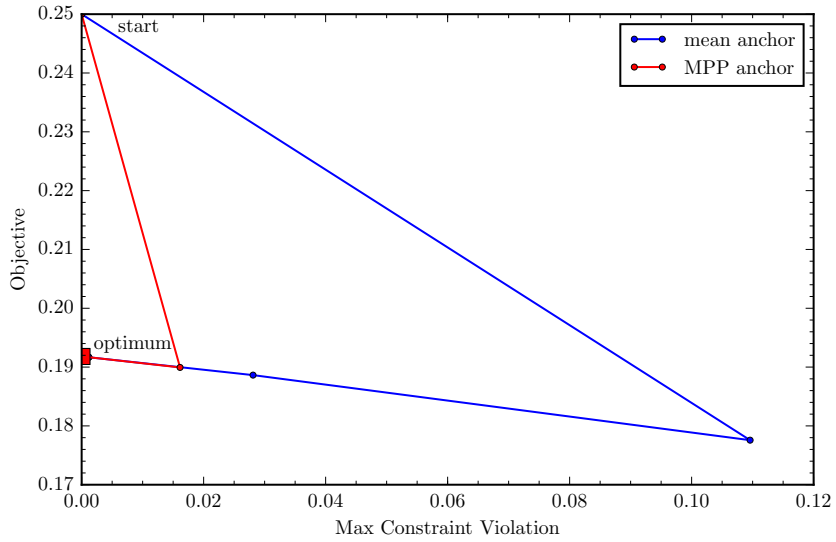
$$\begin{aligned} \underset{x=[w,t]^\top \in \mathbb{R}^2}{\text{minimize}} \quad & f(x) = wt \\ \text{s.t.} \quad & P\left[g(x, y) \leq 0\right] \quad \leq 1.35 \times 10^{-3} \\ & 1 \leq w \leq 4 \\ & 1 \leq t \leq 4, \end{aligned} \tag{4.64}$$

where $g(x, y) = \min_{S,D}\{g_S(x, y), g_D(x, y)\}$ which shows the cantilever beam is a series system, *i.e.* failure due to either excessive stress or displacement indicates system failure. The chance constraint with $p_f = 1.35 \times 10^{-3}$ corresponds to a system reliability index $\beta = 3$.

Table 4.4 compares the results of the cantilever beam optimization using MPP and mean anchors with the modified SORA approach proposed by Ba-abbad et al and described in section 4.3.2. Although Ba-abbad et al do not report the cost of their optimization, they use a sequential approach requiring two inverse reliability analyses each iteration, one for each limit state. AKIRA requires a

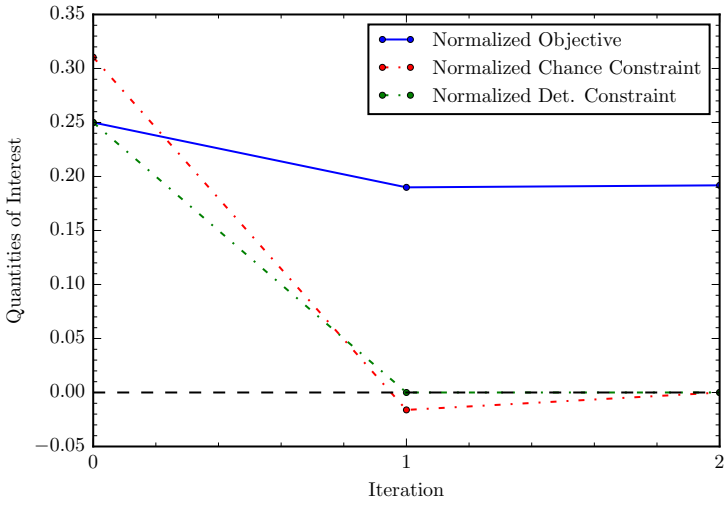


(a) Convergence of design variables

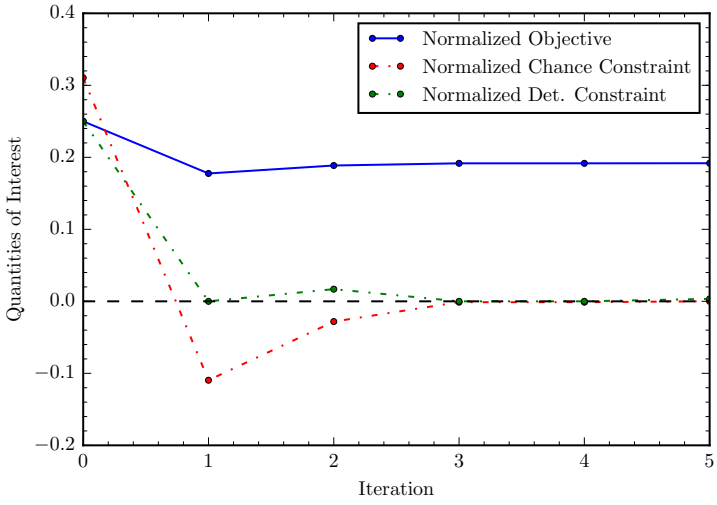


(b) Objective vs. constraint violation

Figure 4.5: Comparison of how the choice of anchor in the separable approximation of the limit state affects the RBDO process for the short column problem specified in (4.63). The design variables b and h are deterministic, *i.e.* geometric uncertainties are not considered. Results are shown for when the anchor is taken to be the mean of the 4 random variables and for when the anchor is taken to be the MPP. A first order additive correction is used for the limit state approximation and the filter method is used for acceptance in the trust region method.



(a) Convergence for MPP anchor



(b) Convergence for mean anchor

Figure 4.6: Comparison of convergence for the short column problem when the MPP is used as the anchor and when the mean of the random variables is used as the anchor. The design variables b and h are deterministic, *i.e.* geometric uncertainties are not considered. A first order additive correction is used for the limit state approximation and the filter method is used for acceptance in the trust region method.

single inverse reliability analysis, but the overall sequential optimization may require more iterations as Ba-abbad approximate the location of the MPP during the deterministic optimization. This is

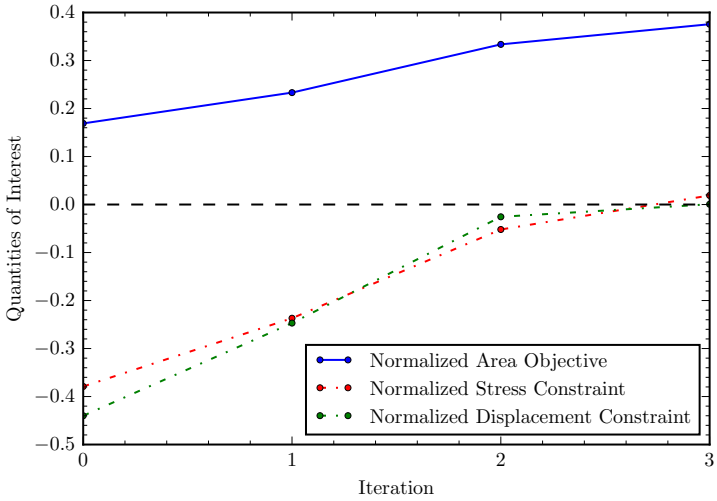
not done in the proposed sequential approach. Nevertheless, Table 4.4 shows that an optimal reliable design can be obtained in several hundred function evaluations, and much less when the MPP anchor is used.

Table 4.4 also shows that all optimizations converge to approximately the same design: $(w, t) = (2.62, 3.64)$. In fact the optimization using the MPP anchor declares convergence slightly outside of the optimum found by the other optimizations due to a slightly lax convergence tolerance of 10^{-3}. Nevertheless, the differences are small and can be ameliorated to a certain extent by using smaller convergence tolerances in the AKIRA algorithm and the RBDO algorithm. This issue thereby identifies one limitation of using AKIRA within a RBDO method — as a response surface method using sampling to obtain estimates of probabilities of failure, a certain amount of noise is present in the limit state level estimates used in the optimization. The noise can be decreased by refining the kriging model more and increasing the sample count, but only at the expense of more function evaluations. In practice, an optimization convergence tolerance of 10^{-3} has been found to be appropriate for RBDO using AKIRA with nominal settings. The difficulty with using a larger convergence tolerance, as shown in Table 4.4, is slightly premature convergence on occasion.

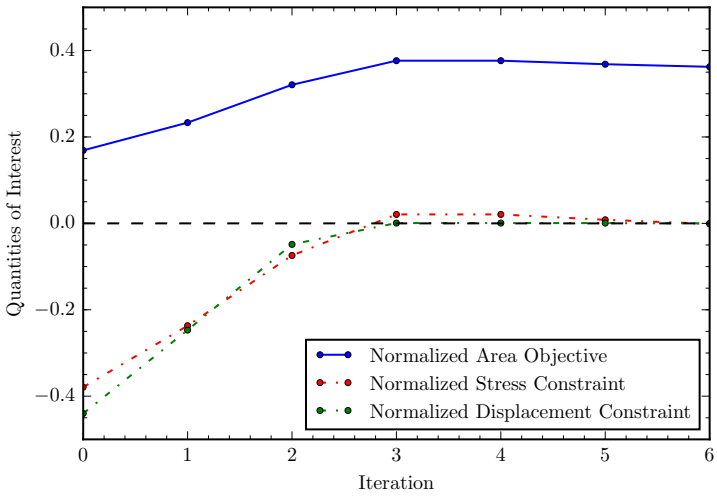
Table 4.4: Comparison of RBDO results for the cantilever beam problem specified in (4.64). Results for RBDO using a MPP anchor and a mean anchor are compared with Ba-abbad et al's modified SORA approach described in section 4.3.2 [14]. $f(x^*)$ is the optimal value of the objective function, w^* and t^* are the optimal width and thickness of the cantilever's cross section, and N is the number of function evaluations required during the optimization.

Method	$f(x^*)$	w^*	t^*	N
MPP anchor; two-point 2nd order correction	9.6809	2.6589	3.6410	179
Mean anchor; two-point 2nd order correction	9.5680	2.6219	3.6492	519
Modified SORA [14]	9.561	2.628	3.639	–

Figure 4.7 compares the convergence of the optimizations using the mean and MPP anchors. Convergence is declared after just 3 iterations for the optimization using the MPP anchor whereas the optimization using the mean anchor requires at least 6 iterations for convergence to the optimum. The results presented for the cantilever beam using the proposed sequential RBDO method with AKIRA instill confidence for the efficient reliability-based design optimization of a system with multiple limit states. In the next chapter, AKIRA and the sequential RBDO method are applied to the design of a supersonic nozzle system.



(a) Convergence for MPP anchor



(b) Convergence for mean anchor

Figure 4.7: Comparison of convergence for the cantilever beam problem when the MPP is used as the anchor and when the mean of the random variables is used as the anchor. A two-point second order correction is used for the limit state approximation and the filter method is used for acceptance in the trust region method.

Chapter 5

Supersonic Nozzle Demonstration

In this chapter the reliability-based design optimization (RBDO) of a military turbofan supersonic nozzle system is presented. The supersonic nozzle design problem was selected due to its inherent multiphysics nature requiring aerodynamic, thermal, and structural analyses, as well as the ample presence of uncertainties ranging from material properties to inlet and atmospheric conditions [51]. In addition the nozzle is an integral component of an aircraft's propulsion system and thus requires very high reliability. Together the complexity of the nozzle system and its importance to the overall reliability of an aircraft make it an ideal application for the demonstration and testing of RBDO methods on an industrial-type application.

We consider design for a critical top-of-climb design condition where temperatures and stresses in the nozzle system are the highest. The goal is to minimize the mass of the nozzle while maintaining a reliable nozzle system in terms of thrust generation and structural failure. 40 random variables were identified and characterized from experimental data, simulations, or expert judgment and define the uncertainties present in the design problem. The remainder of this chapter discusses the modeling and design choices for the supersonic nozzle problem, the RBDO approach, and the design optimization results. Section 5.1 introduces the nozzle system and section 5.2 describes the basic physics of supersonic nozzles. Then section 5.3 introduces the computational models used in the optimization, followed by the geometric and random variable parameterizations in section 5.4. The formal RBDO problem formulation and a standard practice deterministic optimization formulation are presented in section 5.5. Finally section 5.6 summarizes the RBDO approach and section 5.7 presents and compares results for the RBDO and deterministic optimizations for different fidelity models. Comments on nozzle design and the efficacy of RBDO for the nozzle design problem are given in section 5.8.

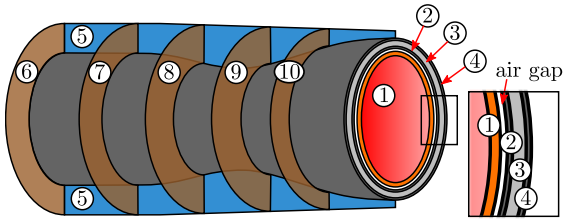


Figure 5.1: Schematic of nozzle components.

Number	Component Name
1	thermal layer
2	inner load layer
3	middle load layer
4	outer load layer
5	stringers
6-10	baffles 1-5

Table 5.1: Component numbering.

5.1 Nozzle Components

Figure 5.1 shows the primary components of the non-axisymmetric nozzle system. The nozzle structure adopted in this dissertation features a multi-layer wall which is attached to the surrounding aircraft substructure via a series of baffles and stringers. A brief description of each component follows. Detailed material properties for each component can be found in the appendix, section D.2.

Thermal Layer

The *thermal layer* (component 1) is the innermost wall layer and is in direct contact with the hot exhaust gases. It is built from a ceramic matrix composite, a material which possesses a high thermal stability as well as a low thermal conductivity. Its primary purpose is to inhibit the flow of heat into the surrounding aircraft structure. Small pinholes in the thermal layer allow the pressure to equilibrate on both sides of the thermal layer with a negligible increase in heat transfer through the thermal layer. As a result, the thermal layer does not take any pressure loads; instead these are transferred to the load layer (components 2-4). However the thermal stresses in the thermal layer are high due to its direct contact with the hot exhaust gases and must be taken into account.

Air Gap

The *air gap* separates the thermal layer from the load layers. Although the air gap may appear to be a seemingly trivial detail, it in fact acts as an additional thermal resistance inhibiting the flow of heat into the surrounding aircraft structure. The air gap is an effective thermal insulator because of the low thermal conductivity of air. Here we consider heat transfer across the air gap by conduction only; convection and radiation are neglected. In industrial applications, the nozzle may be actively cooled from the outside using the air gap or circumferential conduits thereby increasing the effectiveness of the thermal insulation of the hot interior nozzle wall.

Load Layer

The *load layer* (components 2–4) supports the thermal layer and absorbs any structural loads imposed on the nozzle. In particular it takes the pressure load from the interior nozzle flow. The load

layer is modeled as a sandwich composite comprised of a titanium honeycomb core and graphite-bismaleimide faceskins. The sandwich composite structure is a well-proven concept for lightweight but strong aerospace applications loaded in bending. The thermal layer is attached to the load layer via a series of standoffs (not shown) which cross the airgap and are not modeled.

Baffles and Stringers

A series of *baffles* (components 6–10) and *stringers* (components 5) support and attach the nozzle's load layer to the surrounding aircraft structure. Baffles extend in the plane normal to the nozzle's axis until intersection with a side wall or outer mould line. Stringers run along the length of the nozzle. The baffles are constructed from a sandwich composite structure similar to the load layer, whereas the stringers are considered to be manufactured from a graphite-bismaleimide composite layup only.

5.2 Supersonic Nozzle 101

The nozzle is the contoured duct at the aft end of an engine which is responsible for the production of the forward thrust force. The purpose of a nozzle is to convert the thermal energy from hot high-pressure combusted exhaust gases from the engine to kinetic energy, thereby generating a propulsive force. The shape of the nozzle determines how much thrust is generated for given inlet and atmospheric conditions. In particular the cross-sectional area ratio of the nozzle inlet, throat, and outlet, as well as the pressure ratio between the inlet and outlet are the governing quantities of interest in nozzle design. As a result the precise shape of the inner wall of the nozzle (in contact with the hot exhaust gases) is of great importance for nozzle designers. Figure 5.2 shows a schematic of a nozzle and illustrates several key parts.

Nozzles generally exhibit flow in a single direction (fore to aft) and for this reason they may with modest accuracy be analyzed using quasi-1D area-averaged flow. The remainder of this section assumes 1D area-averaged flow to explain how supersonic nozzles work.

5.2.1 Mass Conservation

In fact the relationship between the nozzle Mach number M and area A, and many related flow quantities at any streamwise station can be derived by considering the conservation of mass for area-averaged flow. The following derivation closely follows Cantwell [25] and considers all flow variables to be functions of the streamwise coordinate, although this is not explicitly shown. Consider the mass flow rate

$$\dot{m} = \rho U A, \tag{5.1}$$

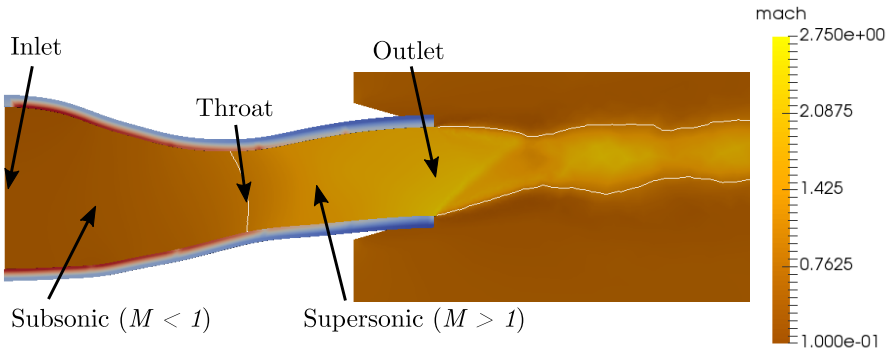


Figure 5.2: Diagram of a supersonic nozzle illustrating the inlet, outlet and point of minimum cross-sectional area, the throat. In a converging-diverging nozzle supersonic flow is achieved in the diverging portion of the nozzle if the inlet stagnation pressure to freestream pressure ratio is large enough. In such a situation the throat of the nozzle is choked and the Mach number M near the throat becomes 1. The white contours on the diagram show the sonic lines, where $M = 1$. The nozzle wall in the diagram is colored according to temperature (blue = cool, red = hot). Near the inlet the temperatures are the highest; as the fluid accelerates and expands it decreases in temperature.

where ρ is the density, U is the streamwise velocity, and A is the area of the nozzle. Applying the equation of state $P = \rho RT$ and definition of the Mach number $M = U/a$, where $a = \sqrt{\gamma RT}$ is the speed of sound, $\dot{m}$ can be written as

$$\dot{m} = \frac{P}{RT}\sqrt{\gamma RT}MA, \tag{5.2}$$

where P is the pressure, R is the specific gas constant, T is the temperature, and γ is the ratio of specific heats. Furthermore recall the isentropic relations for stagnation temperature and stagnation pressure:

$$T_t = T\left(1 + \frac{\gamma - 1}{2}M^2\right), \tag{5.3a}$$

$$P_t = P\left(1 + \frac{\gamma - 1}{2}M^2\right)^{\frac{\gamma}{\gamma-1}}, \tag{5.3b}$$

where T_t is the stagnation temperature and P_t is the stagnation pressure. By solving (5.3) for T and P and plugging into (5.2), the mass flow rate can be expressed as

$$\dot{m} = \frac{\gamma}{\left(\frac{\gamma+1}{2}\right)^{\frac{\gamma+1}{2(\gamma-1)}}}\left(\frac{P_t A}{\sqrt{\gamma RT}}\right) f(M), \tag{5.4}$$

where $f(M)$ is the area-Mach relation given by

$$f(M) = \left(\frac{\gamma+1}{2}\right)^{\frac{\gamma+1}{2(\gamma-1}} \frac{M}{\left(1+\frac{\gamma-1}{2}M^2\right)^{\frac{\gamma+1}{2(\gamma-1)}}}. \tag{5.5}$$

Equating the mass flow rates at any two points in the nozzle (*i.e.* $\dot{m}_1 = \dot{m}_2$) leads to the simple relationship between any two points in the nozzle for area-averaged flow

$$\frac{P_{t,1}A_1}{\sqrt{T_{t,1}}} f(M_1) = \frac{P_{t,2}A_2}{\sqrt{T_{t,2}}} f(M_2). \tag{5.6}$$

In an adiabatic flow, the stagnation temperature T_t is constant. In an isentropic flow, the stagnation pressure P_t is constant. In such a situation, *i.e.* the *ideal nozzle*, the mass flow rate relation simplifies to

$$A_1 f(M_1) = A_2 f(M_2), \tag{5.7}$$

which makes the relationship between area and Mach number apparent. For example suppose the Mach number is 1 at station 1 corresponding to A_1. The area-Mach function is at its maximum $f(M_1) = 1$, therefore we see that $A_2 \geq A_1$ for all other stations 2. As a result, the Mach number always becomes critical at the minimum area point of the nozzle first, *i.e.* the *throat*. In non-quasi-1D flow the critical point may be pushed slightly downstream of the throat in a real nozzle due to 3D flow effects.

5.2.2 Converging-Diverging Nozzle

A common nozzle shape, at least in military aircraft and spacecraft is the *converging-diverging* nozzle. In the converging-diverging nozzle the hot gases are accelerated through the converging section of the nozzle and, if the inlet stagnation pressure is high enough, reach the speed of sound near the *throat* of the nozzle, *i.e.* the streamwise point of minimum cross-sectional area. When this occurs the throat of the nozzle is said to be *choked* as it permits no greater increase in mass flow rate As the nozzle diverges after the throat, the gases continue to accelerate beyond the speed of sound, obtaining supersonic speeds while expanding and thereby cooling down and decreasing in pressure. For this reason, converging-diverging nozzles can also be called *supersonic* nozzles.

In fact, the flow in an (ideal) nozzle can be completely characterized in terms of its throat to exit area ratio A_t/A_e and its inlet stagnation pressure to ambient pressure ratio P_t/P_∞ [25]. Using the area-Mach relation from (5.5) and the mass flow rate relation from (5.7), the exit Mach number M_e can be expressed in terms of the area ratio A_t/A_e. Note that there are two solutions for M_e, one subsonic and one supersonic. The isentropic pressure relation from (5.3) can then be used to relate M_e to the inlet stagnation pressure to ambient pressure ratio $^{P_t}/_{P_\infty}$. A nozzle will typically exhibit a range of states during operation: the over expanded, fully expanded and under expanded states.

5.2.3 Heat Transfer and Friction

In a word, the presence of heat addition and/or friction tends to drive a channel flow towards Mach 1. Fanno flow (1D adiabatic constant area flow with friction) and Rayleigh flow (1D frictionless constant area flow with heat addition) can be examined to confirm this property [25]. In a nozzle flow, typically heat is extracted from the flow through the nozzle walls leading to a decrease in stagnation temperature, whereas the presence of friction at the wall leads to a decrease in stagnation pressure. In an off-design case where a shock in the nozzle occurs, the loss in stagnation pressure can be very large.

The governing equation of motion for quasi-1D area-averaged flow in differential form [25] is

$$\left(\frac{1-M^2}{2(1+\frac{\gamma-1}{2})M^2}\right)\frac{dM^2}{M^2} = \frac{-dA}{A} + \frac{\gamma M^2}{2}\left(\frac{4C_f dx}{D}\right) + \left(\frac{1+\gamma M^2}{2}\right)\frac{dT_t}{T_t}, \tag{5.8}$$

where M is the Mach number, γ is the ratio of specific heats, A is the cross-sectional area of the nozzle, C_f is the friction coefficient of the nozzle wall, D is the diameter of the nozzle, T_t is the stagnation temperature at the wall of the nozzle, and x is the coordinate along the length of the nozzle. (5.8) shows that when friction (C_f) and heat addition $dT_t > 0$ is present, the tendency is to lower the Mach number in supersonic flows and increase the Mach number in subsonic flows.

5.3 Computational Models

An automated suite of coupled multidisciplinary analysis tools for the static aero-thermal-structural analysis of supersonic nozzles called MULTI-F was developed with Victorien Menier and Phil Avery. MULTI-F is written in Python and calls the open source codes SU2 [45] and AERO-S [50] for fluid and thermal/structural analyses. It features a hierarchy of model fidelity levels, ranging in both the physical fidelity (Quasi-1D/Euler/RANS for fluids and linear/nonlinear for structures) and mesh discretization. Users can easily run coupled aero-thermal-structural analyses for user-specified nozzle geometries, material combinations, and environmental parameters by managing a configuration file and associated input file specifying variable values. Figure 5.3 shows the workflow and capabilities of MULTI-F.

First an aerodynamic analysis is performed from which the inner wall temperature and thrust force are extracted. Next a thermal analysis is performed which estimates temperatures in the structural components of the nozzle. Then a structural analysis is performed which takes into account thermal strains due to the temperature increase. The multidisciplinary analysis has one-way coupling, *i.e.* displacements from the structural analysis do not feed back into a new aerodynamic analysis, since convergence of the fully coupled problem would increase the analysis expense.

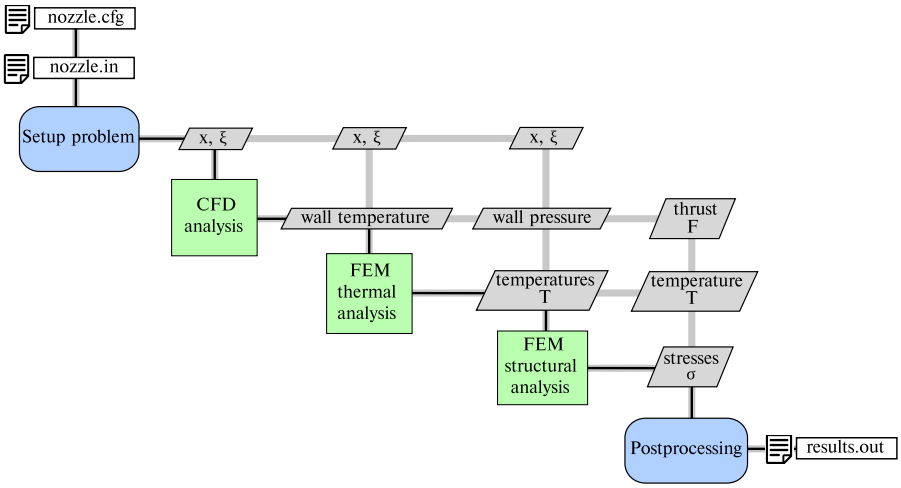


Figure 5.3: Workflow and capabilities of the multifidelity supersonic nozzle aero-thermal-structural analysis suite MULTI-F. The user provides a configuration file specifying the problem and a corresponding input file specifying design variable values. MULTI-F then performs the required analyses and outputs results to a file.

5.3.1 Aerodynamic Models

MULTI-F features three physical fidelities of aerodynamic models; two are discussed below (referred to hereafter as the low- and high-fidelity models) and used in RBDO of the nozzle system.

Low-Fidelity Model

The low-fidelity aerodynamic model solves the quasi-1D area-averaged Navier Stokes equation previously shown in (5.8) using `ode45`, a 4th-order accurate Runge-Kutta method for ODE integration. Since the governing flow equations are area-averaged, the low-fidelity model assumes all flow properties are homogenous in each cross-sectional slice of the nozzle. The integration requires the definition of the friction coefficient C_f and stagnation temperature T_t at the nozzle wall. An initial guess is made for both quantities and a fully coupled aerothermal analysis is carried out with Gauss-Seidel updates until convergence. A simplified thermal analysis is used where each wall layer is modeled as a group of equivalent thermal resistances; as a result only conduction in the radial direction is taken into account. Convergence of the fully coupled aerothermal analysis typically occurs in 6 to 7 iterations; the entire analysis for the low-fidelity aerodynamic model takes approximately 1 second.

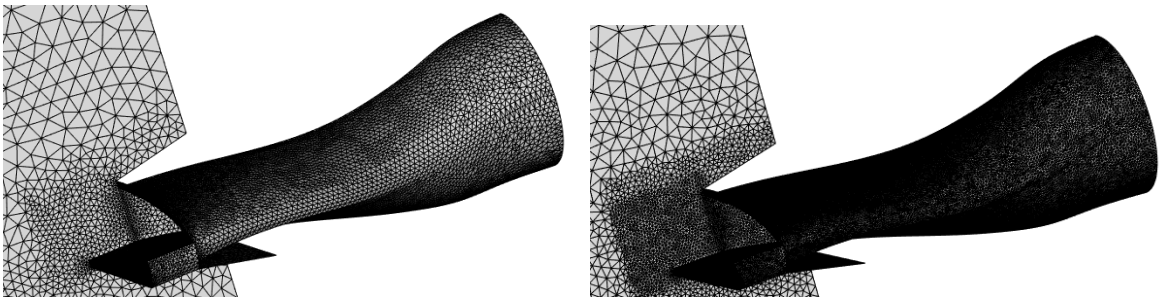
High-Fidelity Model

The high-fidelity aerodynamic models solve the steady, compressible Euler in 3D using SU2, an open-source software suite for multiphysics simulations [45]. The finite volume method is employed

in SU2 with median-dual, vertex-based control volumes [45]. The method is implemented using a standard edge-based data structure on a dual grid. Convective fluxes are evaluated at the midpoint of each edge; here the fluxes are discretized using the second-order accurate Jameson-Schmidt-Turkel (JST) scheme [74]. Time integration is performed using an Euler implicit method and the discretized linear system is solved using the Generalized Minimum Residual (GMRES) method [112]. In addition, SU2's agglomeration multigrid method is employed to accelerate convergence.

Since even small changes in the nozzle shape can have a dramatic impact on the flow physics, particularly near the throat of the nozzle, the aerodynamic model must be robust in addition to accurate and reasonably fast.[1] A set of three high-quality increasingly refined meshes, one for each fidelity, were constructed for a baseline nozzle shape. Hereafter they are referred to as coarse, medium, and fine. During optimization each mesh is deformed to fit the nozzle geometry by analytically projecting the baseline mesh onto the new geometry and then running the SU2 mesh deformation module to ensure a good quality volume mesh.

The baseline meshes are shown in Figure 5.4 and were initially generated using Gmsh with different levels of refinement in the nozzle interior, exit, plume, mid-field and far-field regions [57]. A postprocessing step using the Feflo.a-AMG Inria library ensures the mesh satisfies size requirements [85].



(a) Coarse Euler mesh (b) Fine Euler mesh

Figure 5.4: Coarse and fine meshes for high-fidelity aerodynamic model (Euler). *Courtesy: Victorien Menier*

5.3.2 Thermal and Structural Models

The nozzle thermal and structural analyses are calculated using AERO-S, an open-source finite element method (FEM) analysis software [50][2]. The thermal analysis solves a Poisson boundary value problem representing steady state heat transfer through the nozzle walls from the hot exhaust gases.

[1] Credit for mesh generation and extensive tuning of flow solver parameters to ensure robust aerodynamic high-fidelity models is due to Victorien Menier.

[2] Credit for thermal and structural FEM mesh generation and construction of robust thermal and structural models is due to Philip Avery.

The structural analysis solves an elastostatic boundary value problem representing the equilibrium of internal and external forces in the nozzle structure. Both analyses are variable fidelity. We consider three increasingly refined meshes, hereafter referred to as coarse, medium, and fine (see Figure 5.5). In addition, the structural model has the option to perform a nonlinear analysis which does not make the simplifying assumption of small displacements which the standard linear analysis makes.

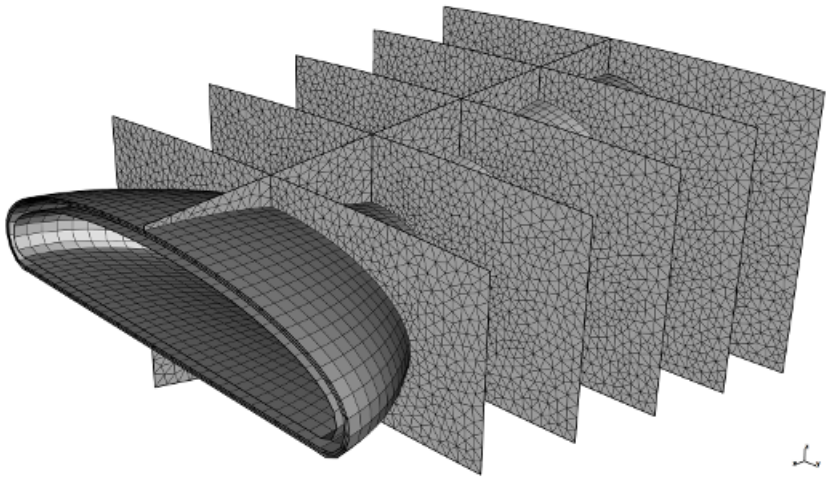


Figure 5.5: FEM mesh for structural model. *Courtesy: Philip Avery*

Thermal Analysis

The thermal model models the thermal layer, air gap, and load layers using conventional 8-node hexahedral finite elements and applies a static linear analysis. The geometries and mesh are generated using the OpenCascade library and the transfinite meshing algorithm of Gmsh [57]. The wall temperature from the aerodynamic analysis is interpolated onto the inner wall of the thermal model and serves as a fixed temperature boundary condition. A convection boundary condition is applied to the exterior of the outer load layer and is parameterized with a generalized heat transfer coefficient. The thermal model considers conduction only; convection in the air gap is neglected, as well as heat transfer due to radiation.

Structural Analysis

The structural model models the thermal layer and load layers as a multi-layer elastic shell and applies either a static linear or nonlinear analysis. The shell elements are modeled using standard Kirchhoff thin-plate theory; both the mid-surface in-plane strains and curvatures are assumed constant. Layered composite finite elements with three nodes and six degrees of freedom per node are used [69] and constructed by superposing a membrane triangle with drilling degrees of freedom [9]

and an assumed natural deviatoric strain (ANDES) bending triangle [86]. Lastly, to account for large displacements and associated geometric nonlinearity in the high-fidelity model, the co-rotational formulation is used [94]. This projection-based technique filters out the potentially large rigid body component of each element's motion, leaving only a relatively small elastic deformation to which the theory of linear elasticity can be applied.

The geometries and mesh are obtained in the same manner as for the thermal analysis. The wall pressure from the aerodynamic analysis is interpolated onto the inside of the inner load layer and furnishes a pressure boundary condition. Temperatures in the thermal layer and load layers are extracted from the thermal analysis and contribute to thermal strains which is the dominant source of stress. The stringers and baffles do not consider thermal strains as their temperature does not vary greatly. Baffles have a fixed and zero displacement at their exterior edges. The inlet of the nozzle is unconstrained in the model as this would lead to excessive strains due to thermal expansion; the problem of the attachment of the nozzle inlet to the turbine exit of the engine is viewed as an important but complicating factor left for a more detailed design study.

5.3.3 Validation and Robustness

Although SU2 and AERO-S have been independently validated and verified there is little to no openly available data for the validation and verification of the full coupled aero-thermal-structural nozzle model presented above. However, several studies were performed which allowed for the qualitative assessment of performance as well as quantitative assessment of robustness. The most useful study involved random parameter sweeps in the combined domain of both the nozzle problem's geometric deterministic variables and random parameters. A point and direction was randomly picked from the augmented reliability space (including the deterministic variables x and random parameters y) and a series of equally-spaced points were evaluated along the direction until the geometry of the nozzle violated the set of linear constraints characterizing reasonable nozzle shapes. In doing so, each set of sweep data could be examined for numerical noise or discontinuities indicative of lack of convergence, software bugs, or possibly discontinuous physical phenomena.

5.4 Parameterization

In this section the parameterization of the problem and the model is discussed. First the choice of the critical operating condition for the nozzle is discussed, which leads to the definition of mission parameters which inform some of the uncertainties present in the problem. Next, the parameterization of the nozzle geometry is discussed in preparation for shape optimization of the nozzle system. Finally, the identification and parameterization of the uncertainties in the nozzle problem is discussed in preparation for reliability analysis.

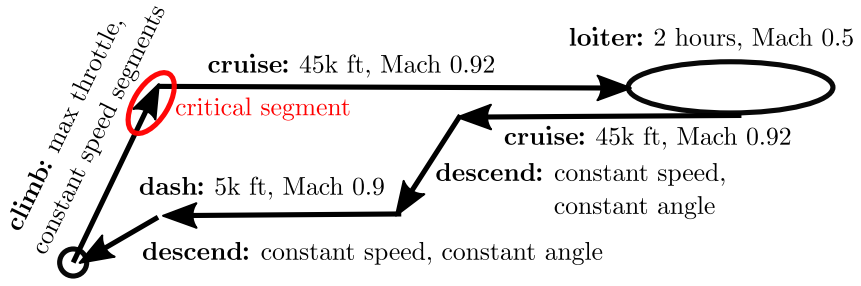


Figure 5.6: Schematic of typical reconnaissance mission with critical top-of-climb segment circled.

Table 5.2: Mission parameters.

Parameter	Units	Nominal Value
Altitude	ft (km)	40,000 (12.192)
Mach		0.511
Required thrust	N	21,500

5.4.1 Operating Condition

In order to characterize the critical operating condition for the nozzle, a typical reconnaissance mission was analyzed for a small high-subsonic unmanned military aircraft. The mission includes climbing as fast as possible to a cruise altitude of 43,000 ft, cruising at Mach 0.92 for a specified distance to an observation point, say 500 km, loitering at an altitude of 43,000 ft and Mach 0.5 for 2 hours and then returning to the takeoff point. On the return, the aircraft descends to 10,000 ft and cruises at Mach 0.9 in a high-speed dash segment lasting several kilometers before landing. Figure 5.6 illustrates the described mission profile.

The analysis of the mission in Figure 5.6 unsurprisingly shows that the climb segment is the most critical for nozzle performance since maximum thrust is required at all altitudes leading to the highest temperatures and pressures at the inlet of the nozzle. In particular, the state of climb right before beginning the cruise segment is the most critical in terms of stresses and temperatures experienced by the nozzle. Table 5.2 summarizes the mission parameters used in the nozzle analysis.

5.4.2 Geometry

The nozzle inlet (circular) and outlet (elliptical), and aircraft outer mould line are considered fixed. However each structural component of the nozzle is fully parameterized for the shape optimization of the nozzle. The shape of the inner wall of the nozzle which corresponds to the inside of the thermal layer directly in contact with the fluid is parameterized by elliptical cross-sections in the $\hat{y}\hat{z}$ plane which are centered along a nonlinear nozzle axis (the *centerline*) connecting the center of the

nozzle inlet to the center of the nozzle outlet. The centerline and the *major* and *minor* axes of the elliptical cross sections are parameterized with 3rd degree basis splines (B-splines). Manipulating the location of the B-spline control points gives effective control of a flexible inner wall shape. Here we consider a fixed knots vector for the B-splines. Figure 5.7 shows the B-splines parameterizing the inner wall shape. More information on B-splines can be found in the appendix in section D.1.

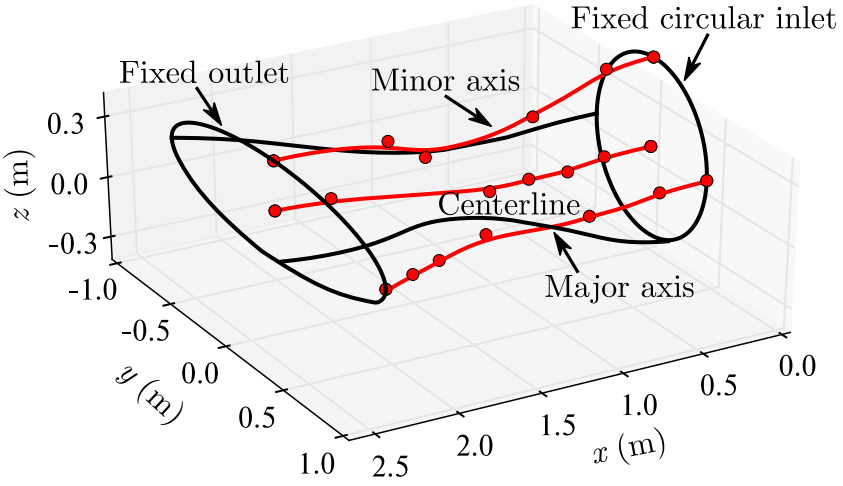


Figure 5.7: Schematic showing the parameterization of the inner nozzle wall using three 3rd degree B-splines. The B-splines and their control points are shown. The coordinates of the B-splines are used as design variables to manipulate the shape of the nozzle. Since the control points also form a convex hull for the B-spline, a simple set of linear constraints can formulated in terms of the control points in order to ensure control points are well spaced and steep changes in geometry are avoided.

The thermal layer and load layers are parameterized with piecewise bilinear functions defined using a polar coordinate system in $\hat{x}\hat{\theta}$ where $\hat{\theta}$ sweeps through the $\hat{y}\hat{z}$ plane. The thickness of the load layer t is defined in the wall normal direction $\hat{n}$ for a given (x, θ) pair. Each wall's normal $\hat{n}$ is defined relative to the underlying layer; for the thermal layer the wall normal is perpendicular to the inner wall shape. θ coordinates are considered fixed, however the x location of the piecewise bilinear breaks and the thickness t at each break are variable allowing the thickness of each layer to change.

The air gap and the baffles are parameterized with a uniform thickness that is allowed to vary. The axial location of each baffle, with the exception of the first and last is also allowed to vary. Finally the thickness of the stringers are parameterized with piecewise linear functions defined along the axial direction $\hat{x}$. Both the break locations x of the piecewise linear functions and the thickness of the stringers at each break location are allowed to vary. Two stringers are considered on the top and the bottom of the nozzle.

5.4.3 Uncertainties

Forty sources of aleatory uncertainty were identified in the nozzle system for consideration during RBDO. These sources include material properties (34), heat transfer uncertainties (2), and inlet (2) and atmospheric (2) conditions. Each source of uncertainty was parameterized using a probability distribution whose parameters were characterized from experimental data, simulations, or expert judgment. The parameterization for each uncertainty is listed in Table 5.3, with the exception of the material properties which are listed in detail in the appendix in section D.2.

The 34 material properties correspond to the ceramic matrix composite (thermal layer), graphite bismaleimide (load layer, stringers, and baffles), and titanium honeycomb (load layer and baffles) materials. They include constitutive and thermal properties as well as strength-related properties such as maximum strain or yield strength and maximum service temperature. When appropriate, the lognormal distribution was fit to test data available from manufacturer datasheets. Otherwise, a uniform distribution was chosen to parameterize the uncertainty in the material properties with a typical coefficient of variation for the given property.

The two heat transfer uncertainties considered are the thermal conductivity of the air in the air gap and the generalized heat transfer coefficient defining the rate of rejection of heat from the exterior nozzle wall to the ambient aircraft structure. The thermal conductivity of air is a function of temperature, however this relationship is not modeled due to simplicity and thus this uncertainty could be considered as epistemic. The uniform distribution was chosen due to its maximum entropy property. The generalized heat transfer coefficient was characterized via Monte Carlo on a simple 1D thermal resistance model and a lognormal distribution was found to be a suitable fit.

The inlet stagnation pressure and temperature of the nozzle are area-averaged and therefore homogenous across the inlet area. They were characterized by propagating turbofan engine parameter uncertainties and atmospheric uncertainties through a simple F100-PW-220 engine model using Monte Carlo and fitting a lognormal distribution to the nozzle inlet conditions.

Finally the nominal and mean values of the atmospheric pressure and temperature were obtained from the 1976 standard atmosphere model. A typical coefficient of variation for day-to-day pressure and temperature fluctuations was then assumed for a lognormal distribution characterizing these uncertainties.

5.5 Optimization Statements

Minimum mass optimization is considered for the nozzle problem subject to constraints on performance criteria such as thrust and failure criteria such as maximum temperatures and stresses. In this section the quantities of interest in the nozzle problem are identified as objective and constraints, and standard practice deterministic design optimization and reliable design optimization problem formulations are presented. A reduced reliable design optimization problem formulation is

Table 5.3: Heat transfer, inlet and atmospheric aleatory uncertainty parameterizations. Material uncertainties are described in detail in the appendix in section D.2.

Parameter	Units	Nominal Value	Distribution
Thermal conductivity	$\frac{\text{W}}{\text{m-K}}$	0.0425	$\mathcal{U}(0.0320, 0.0530)$
Heat transfer coefficient to environment	$\frac{\text{W}}{\text{m}^2\text{-K}}$	12.62	$\ln\mathcal{N}(2.5090, 0.2285)$
Inlet stagnation pressure	Pa	97,585	$\ln\mathcal{N}(11.5010, 0.0579^2)$
Inlet stagnation temperature	K	955.0	$\ln\mathcal{N}(6.8615, 0.0119^2)$
Atmospheric pressure	Pa	18,754	$\ln\mathcal{N}(9.8386, 0.0323^2)$
Atmospheric temperature	K	216.7	$\ln\mathcal{N}(5.3781, 0.0282^2)$

also presented based on the dimension reduction undertaken in the section 5.5.3. It is the solution of the reduced problem that is presented and compared in the results section 5.7.

5.5.1 Quantities of Interest

The quantities of interest in the nozzle design optimization problem include the mass of the nozzle $M(x, y)$ as a surrogate for cost, as well as the thrust $F(x, y)$ and component temperatures T_i and structural failure criteria S_i for the ith nozzle component (refer to Figure 5.1 for component numbering). Each quantity is a function of both the deterministic design variables x related to geometry and the random parameters y.

Expected Mass Objective

Since the geometry of the nozzle governed by design variables x is considered to be deterministic, the expected mass objective is easily calculated. The mass is formulated as a sum over N finite elements, each with deterministic volume V_i and random density ρ_i:

$$\begin{aligned} E[M(x,y)] &= E\left[\sum_{i=1}^{N} \rho_i V_i\right] \\ &= \sum_{i=1}^{N} E[\rho_i] V_i \\ &= M(x, E[y]). \end{aligned} \tag{5.9}$$

Thrust Constraint

The nozzle thrust is a primary performance requirement for the nozzle. Per Table 5.2 it must remain above 21,500 N. The thrust F is calculated via a integration over the exit area of the nozzle. Failure due to insufficient thrust occurs when

$$\frac{F(x,y)}{21,500} < 1. \tag{5.10}$$

Temperature Constraints

Temperature constraints are not implemented for the thermal layer, or for the middle or outer load layers. This follows from the following considerations:

1. A steady-state problem is being analyzed. As a result heat cannot accumulate anywhere.
2. The fluid temperature is highest at the inlet. As a result no measure can be taken in the model to decrease the maximum temperature in the thermal layer short of decreasing the inlet temperature. The inlet temperature distribution and material properties of the thermal layer are designed prior to optimization to ensure failure due to excessively high temperatures does not occur in the thermal layer.
3. If the inner load layer is cool enough, then the middle and outer load layers will also be cool enough as the temperature decreases towards the exterior nozzle walls.

Thus thermal failure for the ith node in the inner load layer occurs when

$$T(x,y)_i \equiv \frac{T_{2,i}\,(x,y)}{T_{max,2}(y)} \geq 1, \tag{5.11}$$

where $T(x,y)_i$ is the temperature ratio for the ith node in the inner load layer, $T_{2,i}$ is the temperature at the ith node in the inner load layer (component 2), and $T_{max,2}$ is the maximum allowable service temperature for the inner load layer's graphite-bismaleimide material.

Structural Failure Criteria Constraints

The structural failure of each nozzle component is analyzed and constrained. The notation $S_j(x,y)_i$ is used to denote the structural failure criterion for the jth component (see Figure 5.1 for component numbering) at the ith finite element node. Since each component uses different material properties, different failure criteria must be considered for each. For clarity the dependence of nodal stresses and strains on the component number j has been dropped in the following descriptions.

The structural failure constraint for the thermal layer's ceramic matrix composite material uses a failure criterion based on maximum principal strain. In other words failure is assumed to occur when at any FEM node, $\epsilon_I \geq \epsilon_f$ where ϵ_I is the maximum (tensile) principle strain and ϵ_f is the maximum allowable principal strain. The principle strains are denoted by $\epsilon_I \geq \epsilon_{II} \geq \epsilon_{III}$. Failure under compression is assumed to not occur. Structural failure at the ith node in the FEM thermal layer mesh occurs when

$$S_1(x,y)_i \equiv \frac{\epsilon_{I,i}(x,y)}{\epsilon_f(y)} \geq 1. \tag{5.12}$$

The structural failure constraints for the load layer's graphite-bismaleimide composite material use a failure criterion based on maximum in-plane strains ϵ_1 and ϵ_2 in the local in-plane material

coordinate system (denoted by subscripts 1 and 2). In other words failure is assumed to occur when the in-plane strain in directions 1 or 2 exceeds the maximum allowable in-plane strain $\epsilon_{f,1}$ or $\epsilon_{f,2}$. Note that $\epsilon_{f,1}$ and $\epsilon_{f,2}$ differ for tensile and compressive strains. In addition, failure by in-plane shear is assumed to occur when the in-plane shear strain $\gamma_{12} \geq \gamma_f$. Failure due to out-of-plane strains is neglected for consistency with the shell-element approximation. As a result, failure in the inner load layer, outer load layer, stringers, and faceskins of the baffles occurs for the ith FEM node and jth component when

$$S_j(x,y)_i \equiv \max_{\{1,2,12\}}\{S_j^1(x,y)_i, S_j^2(x,y)_i, S_j^{12}(x,y)_i\} \geq 1, \tag{5.13}$$

where $S_j^1(x,y)_i$, $S_j^2(x,y)_i$, and $S_j^{12}(x,y)_i$ correspond to failure due to tensile or compressive strain in the two local in-plane material directions and failure due to in-plane shear strain:

$$S_j^1(x,y)_i \equiv \begin{cases} \dfrac{\epsilon_{1,i}(x,y)}{\epsilon_{f,1,tension}(y)} & \text{when } \epsilon_{1,i} \geq 0, \quad (5.14a) \\ \dfrac{\epsilon_{1,i}(x,y)}{\epsilon_{f,1,compression}(y)} & \text{when } \epsilon_{1,i} < 0, \quad (5.14b) \end{cases}$$

$$S_j^2(x,y)_i \equiv \begin{cases} \dfrac{\epsilon_{2,i}(x,y)}{\epsilon_{f,2,tension}(y)} & \text{when } \epsilon_{2,i} \geq 0, \quad (5.15a) \\ \dfrac{\epsilon_{2,i}(x,y)}{\epsilon_{f,2,compression}(y)} & \text{when } \epsilon_{2,i} < 0, \quad (5.15b) \end{cases}$$

$$S_j^{12}(x,y)_i \equiv \frac{\gamma_{12,i}(x,y)}{\gamma_f(y)}. \tag{5.16}$$

The structural failure constraints for the titanium honeycomb material uses the von Mises failure criterion. In other words failure is assumed to occur when $\sigma_{vm} \geq \sigma_y$ where σ_{vm} is the von Mises stress and σ_y is the yield strength. Structural failure at the ith node in the FEM mesh occurs when

$$S_3(x,y)_i = \frac{\sigma_{vm,i}(x,y)}{\sigma_y(y)} \geq 1. \tag{5.17}$$

For the nozzle baffles, a symmetric composite panel material with fixed layer thickness ratios is assumed. It consists of a top and bottom graphite-bismaleimide layer and a middle titanium honeycomb layer. The maximum in-plane strain failure criterion is used for the graphite-bismaleimide layer and the von Mises stress failure criterion is used for the titanium honeycomb layer as mentioned above.

Local Constraint Approximation via One Global Constraint

Constraints 5.11 through 5.17 are defined at every node in the finite element mesh which results in a large number of local nonlinear constraints. A standard approach taken to reduce the number of constraints is to replace each batch of local constraints with a single global constraint [15]. The global constraint is chosen to approximate the behavior of the most critical local constraints. Kresselmeier-Steinhauser functions [15] and modified p-norm functions [70] are common choices. In this dissertation the modified p-norm function is chosen to form a global constraint approximation for each batch of local temperature or structural failure criteria constraints. Here a batch refers to all the local temperature constraints or failure criteria constraints for a single component (*i.e.* thermal layer, inner load layer, etc.) in the nozzle model. The modified p-norm function $g(q)$ is given by

$$g(q) = \left(\frac{1}{N} \sum_{i}^{N} q_i^p \right)^{\frac{1}{p}}, \tag{5.18}$$

where p is a constant usually chosen to be around 8 and q is a vector with each local constraint value in a batch of N local constraints. For $\leq$ constraints, the modified p-norm function under approximates the constraints (in contrast to the Kresselmeier-Steinhauser function which over approximates the constraints). As p increases, $g(q)$ approaches the value $\max_i\{q_i\}$, however numerical difficulties ensue with too large a value of p.

Linear Constraints

A set of linear constraints $Ax \leq b$ is used to enforce a well-posed geometry during the course of the optimization. It includes constraints on the control point coordinates of the B-splines parameterizing the inner wall shape, as well as constraints on the break locations and thicknesses of the piecewise bilinear and linear parameterizations of the wall layers, stringers, and baffles. In addition, the set of linear constraints also inherently includes the bounds on the design variables x. The set of geometric constraints includes minimum separation distances between B-spline control points or break locations as well as limits on the slopes between break locations and control points. The former ensures the parameterization remains well-defined whereas the latter effectively ensures sharp changes in nozzle geometry do not occur.

5.5.2 Reliability-Based Design Problem

The chance constraints for the nozzle reliability-based design optimization (RBDO) problem can be specified independently, each with a prescribed probability of failure $\bar{p}_f$:

$$\begin{aligned}
P\left[\frac{F(x,y)}{21,500} < 1\right] &\leq \bar{p}_f, && \text{(thrust)} \\
P\left[T(x,y) \geq 1\right] &\leq \bar{p}_f, && \text{(load layer temperature ratio)} \\
P\left[S_1(x,y) \geq 1\right] &\leq \bar{p}_f, && \text{(thermal layer failure crit.)} \\
P\left[S_2(x,y) \geq 1\right] &\leq \bar{p}_f, && \text{(load layer inside failure crit.)} \\
P\left[S_3(x,y) \geq 1\right] &\leq \bar{p}_f, && \text{(load layer middle failure crit.)} \\
P\left[S_4(x,y) \geq 1\right] &\leq \bar{p}_f, && \text{(load layer outside failure crit.)} \\
P\left[S_5(x,y) \geq 1\right] &\leq \bar{p}_f, && \text{(stringers failure crit.)} \\
P\left[S_6(x,y) \geq 1\right] &\leq \bar{p}_f, && \text{(baffle 1 failure crit.)} \\
P\left[S_7(x,y) \geq 1\right] &\leq \bar{p}_f, && \text{(baffle 2 failure crit.)} \\
P\left[S_8(x,y) \geq 1\right] &\leq \bar{p}_f, && \text{(baffle 3 failure crit.)} \\
P\left[S_9(x,y) \geq 1\right] &\leq \bar{p}_f, && \text{(baffle 4 failure crit.)} \\
P\left[S_{10}(x,y) \geq 1\right] &\leq \bar{p}_f. && \text{(baffle 5 failure crit.)}
\end{aligned} \tag{5.19}$$

However it is more useful to consider the nozzle as a series system and specify a single system failure probability constraint. In this case, recall the system failure probability $p_f(x)$ is given by

$$\begin{aligned}
p_f(x) = P[\{(x,y) : F(x,y) < 21,500\} \cup \{(x,y) : T(x,y) \geq 1\} \\
\cup \{(x,y) : S_1(x,y) \geq 1\} \cup \ldots \cup \{(x,y) : S_{10}(x,y) \geq 1\}],
\end{aligned} \tag{5.20}$$

where the probability is calculated over the support of the random parameters y for a given design x. Thus the series system RBDO problem is formulated as

$$\begin{aligned}
\underset{x \in \mathbb{R}^{96}}{\text{minimize}} \quad & E[M(x,y)] \\
\text{s.t.} \quad & p_f(x) \leq \bar{p}_f \\
& Ax \leq b,
\end{aligned} \tag{5.21}$$

where $p_f(x)$ is defined in (5.20). In this thesis we consider a prescribed failure probability $\bar{p}_f = 10^{-6}$ which approximates stringent military reliability requirements (*i.e.* one failure per million climb segments).

The chance-constrained problem can alternately be formulated with quantile constraints whose

benefits were enumerated in sections 2.1.5 and 4.1.3:

$$\begin{aligned}
\underset{x\in\mathbb{R}^{96}}{\text{minimize}} \quad & E[M(x,y)] & \\
\text{s.t.} \quad & z_F(x;\bar{p}_f)/21500 & \geq 1 \\
& z_T(x;\bar{p}_f) & \leq 1 \\
& \vdots & \vdots \\
& z_{S_1}(x;\bar{p}_f) & \leq 1 \\
& \vdots & \vdots \\
& z_{S_{10}}(x;\bar{p}_f) & \leq 1 \\
& Ax & \leq b.
\end{aligned} \tag{5.22}$$

The problem formulations in equations 5.21 and 5.22 require the calculation of 13 coupled quantities of interest, reliability analyses for 12 of the 13 over a 40-dimensional random space, and optimization in a deterministic space with 96 design variables. In the next section Sobol' indices are used to motivate a reduction in the size of the design problem.

5.5.3 Dimension Reduction

A reduction of the full RBDO problem is motivated by the facts that 1) some quantities of interest are consistently very far from their failure region for large variations in design and 2) some random variables have little effect on the "important" quantities of interest. Here "important" refers to a quantity of interest that has a non-zero chance of failure for any geometrically possible nozzle design. Once the solution of the reduced RBDO problem has been obtained, the stochastic response of neglected quantities of interest along with the inclusion of neglected random variables may be estimated to ensure the dimension reduction was appropriate.

In the nozzle problem, three quantities were determined to have high impact on the nozzle performance in terms of failure: thrust F, temperature ratio in the inner load layer T, and failure criteria in the thermal layer $S \equiv S_1$. An inadequately shaped nozzle can easily lead to an insufficient amount of thrust being generated due to constriction of the mass flow rate. The inner load layer can easily become too hot if the thermal layer or the air gap is too thin. Finally, the strains in the thermal layer can easily become too large due to thermal strains which depend on the temperature and layer thickness. Finally, the mass is also considered as it is included in the objective function. Thus, the reduced nozzle problem only considers four quantities of interest: mass, thrust, inner load layer temperature ratio, and the thermal layer structural failure criterion. The reduction decreases the number of deterministic design variables x from 96 to 54.

With the four critical quantities of interest identified through means of engineering experience

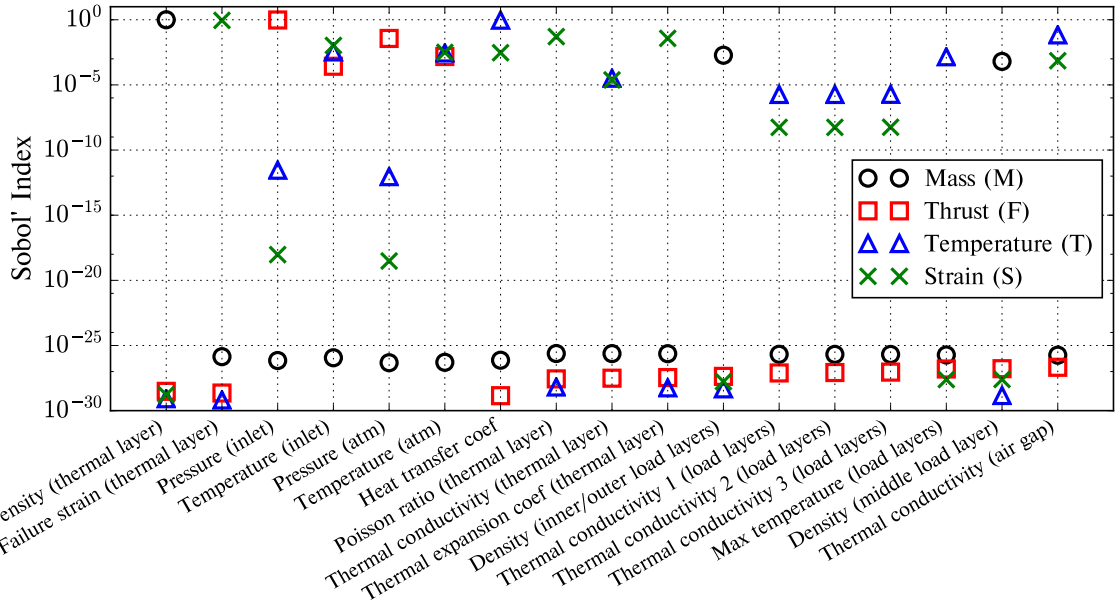


Figure 5.8: Total order Sobol' indices for the 17 random variables which contribute most to the variance of the four quantities of interest considered in the nozzle problem. Eliminating variables with indices less than 10^{-5} reduces the problem's random dimension from 40 to 14. First order Sobol' indices (not shown) are nearly the same implying interaction effects can be neglected. The Sobol' indices are calculated using a sparse grid level 1 polynomial chaos expansion with three anisotropic adaptive refinement iterations, resulting in a total of 3,405 function evaluations for the 3D Euler nozzle model and FEM thermal and structural models.

and expert judgement, the number of random parameters can then be reduced via inspection of Sobol' indices (see Appendix E). The results of the Sobol' index calculation is shown in Figure 5.8 for the 17 random parameters with the largest maximum total-order Sobol' index (*i.e.* value greater than 10^{-6}). Clearly, the variability in densities only affect mass, whereas thermal conductivities, and inlet and atmospheric pressures and temperatures and heat transfer coefficients have a large impact on the remaining quantities.

5.5.4 Reduced Reliability-Based Design Problem

Following the motivation presented in section 5.5.3, a reduced RBDO formulation can be made for the nozzle problem. We consider 54 deterministic design variables and 14 random parameters. The quantities of interest include the mass of the nozzle M, the thrust of the nozzle F, the temperature ratio in the inner load layer T, and the strain failure criterion in the thermal layer S. The reduced

chance constrained design problem is

$$\begin{aligned} \underset{x\in\mathbb{R}^{54}}{\text{minimize}} \quad & E[M(x,y)] \\ \text{s.t.} \quad & p_f(x) \leq \bar{p}_f \\ & Ax \leq b, \end{aligned} \tag{5.23}$$

where the system failure probability $p_f(x)$ is given by

$$p_f(x) = P[\{(x,y) : F(x,y) < 21,500\} \cup \{(x,y) : T(x,y) \geq 1\} \cup \{(x,y) : S(x,y) \geq 1\}]. \tag{5.24}$$

The reduced quantile constrained design problem is

$$\begin{aligned} \underset{x\in\mathbb{R}^{54}}{\text{minimize}} \quad & E[M(x,y)] & \\ \text{s.t.} \quad & \frac{z_F(x;\bar{p}_f)}{21,500} & \geq 1 \\ & z_T(x;\bar{p}_f) & \leq 1 \\ & z_S(x;\bar{p}_f) & \leq 1 \\ & Ax & \leq b. \end{aligned} \tag{5.25}$$

5.5.5 Standard Practice Deterministic Design Problem

A standard practice deterministic design of the nozzle is carried out for comparative purposes with the reliability-based nozzle designs. Safety factors γ (see section 4.2.1) and conservative material properties $\tilde{y}$ *i.e.* basis values (see section 4.2.3) are used. B basis values are calculated assuming the distribution of the random parameters y are exactly known as this is the case considered in the reliability-based design optimization procedure in this thesis. In other words, variability in the distribution parameters for y are assumed to be negligible which corresponds to an ideal situation where characterization of the random parameters of the problem such as material properties, etc. is based upon an infinite number of physical samples. In such a situation, the B basis value is simply the 10th percentile of the distribution for y_i.

The reduced problem described in section 5.5.4 with 14 random parameters and 4 quantities of interest is considered here. The random parameters corresponding to *loads* on the nozzle system are taken to be fixed at their means. These include the inlet stagnation pressure and temperature P_{in} and T_{in} and the atmospheric pressure and temperature P_∞ and T_∞. The effect of the variability of the loads on the nozzle system is accounted for by a safety factor γ in the nozzle responses for thrust F, inner load layer temperature ratio T, and thermal layer structural failure criterion S. The random parameters corresponding to *properties* of the nozzle system are replaced by deterministic B basis values. These include the remaining 10 parameters such as densities, failure strains, thermal

Table 5.4: Fixed random parameters for the standard practice nozzle design. Load variables are fixed at their mean whereas property variables are replaced with a B basis value. The B basis value assumes the exact distribution of the random variable is known; variables leading to a critical nozzle quantity of interest response at lower values are replaced with their 10th percentile whereas variables leading to critical response at high values are replaced with their 90th percentile. Material property key: *CMC*: ceramic matrix composite, *Gr-BMI*: graphite-bismaleimide, *Ti-honeycomb*: titanium honeycomb.

Name	Variable	Value	Units	Type
CMC density	ρ_{CMC}	2,449.463	$\frac{\text{kg}}{\text{m}^3}$	B basis
CMC failure strain	ε_{CMC}	5.776×10^{-2}	–	B basis
Inlet stagnation pressure	P_{in}	98,980.308	Pa	mean
Inlet stagnation temperature	T_{in}	954.866	K	mean
Atmospheric pressure	P_∞	17,753.237	Pa	mean
Atmospheric temperature	T_∞	216.697	K	mean
Heat transfer coefficient	h_∞	6.662	$\frac{\text{W}}{\text{m}^2\text{-K}}$	B basis
CMC Poisson ratio	ν_{CMC}	0.41	–	B basis
CMC thermal conductivity	k_{CMC}	1.442	$\frac{\text{W}}{\text{m-K}}$	B basis
CMC thermal expansion coef.	α_{CMC}	2.496×10^{-7}	–	B basis
Gr-BMI density	ρ_{GR}	1,572.	$\frac{\text{kg}}{\text{m}^3}$	B basis
Gr-BMI max allowable temperature	T_{GR}	501.	K	B basis
Ti-honeycomb density	ρ_{TC}	181.01	$\frac{\text{kg}}{\text{m}^3}$	B basis
Air thermal conductivity	k_{air}	0.0509	$\frac{\text{W}}{\text{m-K}}$	B basis

conductivities, etc., as well as the heat transfer coefficient to the ambient environment. Table 5.4 gives the fixed values $\tilde{y}$ used in the deterministic design problem.

The standard practice deterministic design problem is

$$\begin{aligned} \underset{x\in\mathbb{R}^{54}}{\text{minimize}} \quad & M(x,\tilde{y})] & \\ \text{s.t.} \quad & \frac{F(x,\tilde{y})}{21,500\gamma} & \geq 1 \\ & \gamma T(x,\tilde{y}) & \leq 1 \\ & \gamma S(x,\tilde{y}) & \leq 1 \\ & Ax & \leq b, \end{aligned} \tag{5.26}$$

where $\tilde{y}$ are the fixed B basis values for the random parameters y and γ is a safety factor. In this dissertation the safety factor is varied between 1.2 and 1.5.

5.6 RBDO Approach

The multifidelity sequential approach discussed in section 4.5 and described in Algorithm 6 is used to solve the reduced RBDO problem with quantile constraints from (5.25). The Adaptive Kriging Inverse Reliability Analysis (AKIRA) method discussed in chapter 3 is used to provide estimates of and sensitivities for the high-fidelity quantile constraints. Together the methods are an efficient strategy for tackling large-scale RBDO problems where nonlinear phenomena, many uncertainties, and small failure probabilities are present and function evaluations are expensive.

As discussed in section 5.3, a variety of model fidelities are available for analyzing the nozzle. In the multifidelity sequential approach (section 4.5), the high-fidelity quantile constraints are approximated using the anchored decomposition of a lower-fidelity model. Anchored decomposition (section 4.5.2) motivates approximation of stochastic constraints by fixing the random parameters y at a fixed instance y_c, *i.e.* the anchor point. In the nozzle application the anchor point is chosen as the mean of the random parameters since the nozzle system is coupled; other choices would lead to a larger number of function evaluations (section 4.5.6). In addition the low-fidelity approximation is corrected using a first-order additive correction to satisfy global convergence requirements (section 4.4.3).

A series of deterministic optimizations using the corrected and decomposed low-fidelity model are then managed within a trust region to provide a solution to the high-fidelity RBDO problem (section 4.4). SNOPT, an SQP-based optimization algorithm is used to solve each subproblem [60]. Gradients are estimated for the coupled nozzle system via finite difference. Both step sizes and convergence tolerances are estimated from the parameter sweeps used to assess model noise.

To ensure convergence, reliability analyses must be periodically performed using the high-fidelity model. These reliability analyses are performed using AKIRA which solves the system-level inverse reliability analysis problem for a prescribed probability of failure $\bar{p}_f = 10^{-6}$ (section 3.4). AKIRA models the constraints using independent kriging response surfaces and adaptively adds points to the kriging models to decrease uncertainty in the estimated quantiles yielding the prescribed probability of failure.

When a single-fidelity model is used, the described RBDO approach is still valid but the same model fidelity is used for both the subproblem optimizations and the reliability analyses.

5.7 Results

Table 5.5 summarizes and compares the solutions of the standard practice deterministic design problem from (5.26) and the reduced reliability-based design optimization problem from (5.25) for various model fidelities. Probabilities of failure have been estimated for deterministic designs via a post-processing step using a kriging-based forward reliability analysis. A detailed discussion and interpretation of the results is given in the remainder of this chapter, however several trends are

immediately clear:

1. *The standard practice deterministic design using safety factors and basis values does not achieve the desired reliability.* In some cases, *e.g.* the low-fidelity model, the reliability is lower for the deterministic design, however in other cases, *e.g.* the high-fidelity model, the reliability is much greater than required. Basis values and safety factors are unable to directly control system reliability.

2. *The computational cost of the RBDO problem is an order of magnitude greater than the deterministic problem.* The additional cost is due to the need to perform multiple deterministic optimizations interspersed with reliability analyses to solve the RBDO problem.

3. *The number of high-fidelity model evaluations exceeds the number of low-fidelity model evaluations.* In contrast to most multifidelity methods, the low-fidelity model does not appear to substantially decrease the number of high-fidelity model evaluations. The reason is that reliability analyses must be performed using the high-fidelity model to ensure convergence to an optimum for the high-fidelity model, and such reliability analyses constitute the majority of the cost of solving the RBDO problem. In addition the subproblem optimizations for the nozzle problem using the low-fidelity model are usually easily solved within the trust region bounds.

4. *The critical constraints in the nozzle problem are the thrust and temperature in the inner load layer.* The structural failure criteria in the thermal layer is not an active constraint due to the high performance of the ceramic matrix composite.

5.7.1 Low-Fidelity Model Optima

The optimization results for the low-fidelity model are interesting in their own right due to a polarization of the design variables over a small transition in system reliability. In addition, examination of the low-fidelity model gives intuition as to the behavior of the reliable nozzle design problem. Figure 5.9 qualitatively compares the optimal standard practice deterministic design nozzle geometry with the optimal reliable design nozzle geometry corresponding to $\bar{p}_f = 10^{-6}$. Figure 5.10 compares the optimal inner wall geometry for the standard practice design and two reliable designs, one specifying $\bar{p}_f = 10^{-5}$ and the other specifying $\bar{p}_f = 10^{-6}$. The standard practice design uses a safety factor $\gamma = 1.5$ and B basis values.

As shown in Table 5.5, the standard practice deterministic low-fidelity design is actually fairly reliable with an estimated system failure probability of 2.58×10^{-5}. However this reliability is still lower than the desired 10^{-6}, and the insufficient reliability of the standard practice design allows it to achieve a lower mass, 330 kg instead of 1,033 kg for the reliable design. The cause of the large

Table 5.5: Comparison of standard practice deterministic and reliability-based design optimization (RBDO) results for the nozzle problem. The deterministic optimization results use conservative material properties (B basis values) and safety factors to account for uncertainty. The RBDO results directly account for uncertainty and use the multifidelity sequential method presented in chapter 4 with the AKIRA method presented in chapter 3 for reliability analyses. In the table, "Fidelity" refers to the highest fidelity model used in the optimization; for example "Low" refers to a single-fidelity optimization using the low-fidelity nozzle models, whereas "High" refers to multifidelity optimizations using the high-fidelity model. The total number of function evaluations N_{eval} and gradient evaluations N_{grad} are shown. In multifidelity optimizations the number corresponding to the low-fidelity model is shown on the left, and the number corresponding to the high-fidelity model is shown on the right separated by "/". The optimization objective is expected mass, and $\hat{p}_f$ is the nozzle system failure probability estimated using AKIRA or a kriging-based forward reliability analysis. Finally, the active constraints at the optimum are listed: F refers to thrust, T refers to the inner load layer temperature ratio, and S refers to the thermal layer structural failure criterion.

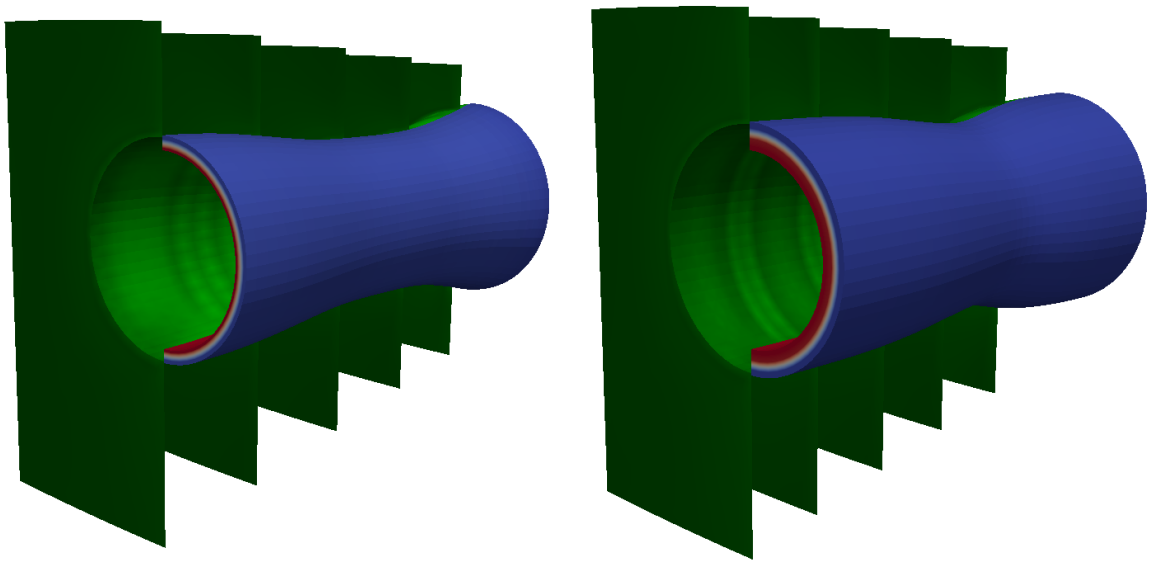
Optimization	Fidelity	N_{eval}	N_{grad}	Objective	$\hat{p}_f$	Critical Constraints
Det.	Low	51	49	329.16	2.58×10^{-5}	F, T
RBDO	Low	386	26	341.73	10^{-5}	F, T
RBDO*	Low	916	70	1,033.11	1.19×10^{-6}	F, T
Det. ($\gamma = 1.5$)	High	8	7	384.17	$< 2.31 \times 10^{-8}$	F
Det. ($\gamma = 1.2$)	High	5	4	380.18	$< 2.31 \times 10^{-8}$	F
RBDO	High	8 / 317	6 / 7	297.51	10^{-6}	F, T

*problem is slightly infeasible for $\bar{p}_f = 10^{-6}$

increase in mass for the reliable design is illustrated well by Figure 5.11 which compares the shapes of the nozzle exit for the standard practice design and the reliable design with $\bar{p}_f = 10^{-6}$. First, the throat area of the reliable design is larger, which allows a larger mass flow rate and therefore increased thrust, thereby ensuring reliable thrust generation. Second, the thermal layer and the air gap are very thick in order to decrease the temperature in the load layer. As a result, the mass of the nozzle is greatly increased.

Figure 5.12 compares the optimal values of the 54 design variables in the low-fidelity optimizations. It clearly shows that the deterministic design chooses the thinnest possible walls and baffles in order to minimize mass and manages the thickness of the air gap to ensure a feasible temperature constraint. On the contrary, the reliable design with $\bar{p}_f = 10^{-6}$ (red circles) chooses wall layers with the maximum possible thickness and a maximum air gap thickness. In fact, the temperature constraint is the driving constraint for the low-fidelity RBDO problem and maximizing wall layer thickness and the air gap is the only way to decrease the temperature in the inner load layer enough. In fact, the air gap is the most effective means to decrease the temperature in the inner load layer with small increases in mass since the air does not contribute to the weight of the nozzle. For this reason, the reliable design with $\bar{p}_f = 10^{-5}$ (green circles) chooses to increase the air gap to meet the temperature constraint while retaining wall layer thicknesses at their lower bounds.

Comparing the reliable designs for $\bar{p}_f = 10^{-5}$ and 10^{-6}, it becomes clear why the two designs,



(a) Standard practice design.

(b) Reliable design.

Figure 5.9: Comparison of optimal nozzle geometry for the low-fidelity model. The inlet of the nozzle is shown. The left half of the nozzle shows the structural model (darker = less stress, lighter = more stress), and the right half shows the thermal model (blue = cooler, red = hotter). The color scales for both nozzles are the same, which enables a qualitative comparison between the optimal geometries.

although only an order of magnitude apart in reliability, are so different in terms of expected mass (342 vs. 1,033 kg). The 10^{-5} design features minimum gauge wall layers and a modest air gap, whereas the 10^{-6} design features maximum gauge wall layers and a maximum thickness air gap. Between the system failure probabilities of 10^{-5} and 10^{-6} there is a fast transition in the nozzle design from thin to thick layers, *i.e.* a phase transition where the input uncertainties, in particular temperature and thermal material properties, dominate any geometry change that might be used to decrease the load layer temperature and reduce failure.

Figure 5.13 shows the convergence of the low-fidelity nozzle RBDO with $\bar{p}_f = 10^{-6}$. In fact, the RBDO problem is slightly infeasible for the prescribed probability of failure, however the nonlinear infeasibilities are decreased as much as possible. A total of 916 low-fidelity model evaluations and 8 trust region iterations are required before the algorithm terminates at a slightly infeasible point.

5.7.2 High-Fidelity Model Optima

The results for the high-fidelity model show the potential for excessive conservatism with well-intentioned deterministic design procedures. Table 5.5 shows the deterministic design using a safety factor $\gamma = 1.5$ and B basis values has an expected mass of 384 kg with a system failure probability at least two orders of magnitude smaller than the desired reliability. On the other hand, the reliable

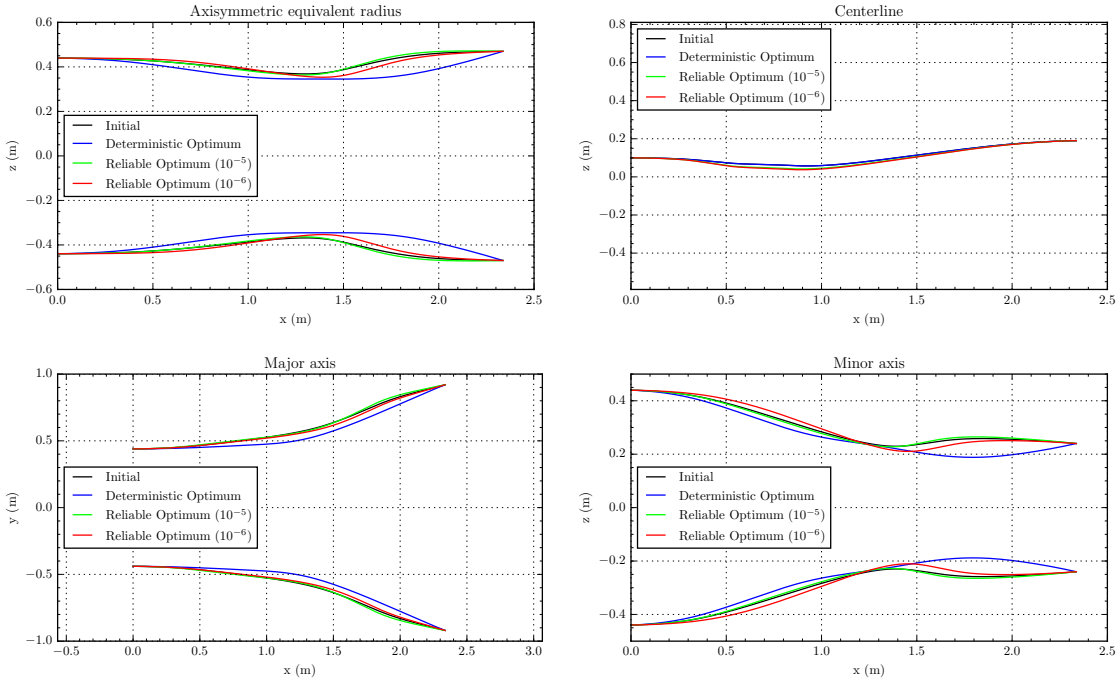


Figure 5.10: Comparison of optimal inner wall geometry for the low-fidelity model. Note the increased throat radius for the reliable designs as compared to the standard practice deterministic design.

design features 22.6% reduction in mass while still meeting the requested reliability $\bar{p}_f = 10^{-6}$. Using a smaller safety factor, $\gamma = 1.2$, which may be more appropriate for military aircraft, does little to ameliorate the overdesign of the deterministic optimization. Figure 5.14 qualitatively compares the optimal standard practice deterministic design ($\gamma = 1.2$) with the optimal reliable design nozzle geometry and Figure 5.15 compares the optimal shapes of the inner nozzle walls. The obvious larger diameter of the standard practice design is reflected by its increased mass compared to the reliable design.

Figure 5.12 compares the optimal design variable values for the two deterministic designs and the reliable design. In nearly all cases, minimum gauge thickness wall layers are chosen to minimize mass. Most of the variability between the designs is found in the shape of the inner wall. As shown in Figures 5.14 and 5.15, the standard practice designs feature a very large throat in order to boost mass flow rate and therefore thrust. In the deterministic optimizations the thrust constraint is the driving constraint and therefore has a dominant role in sizing the nozzle geometry. This effect was not seen in the low-fidelity model optimizations because the Euler aerodynamic model predicts lower thrust than the quasi-1D area averaged aerodynamic model.

The air gap also plays a pivotal role for managing the temperature in the load layer — both

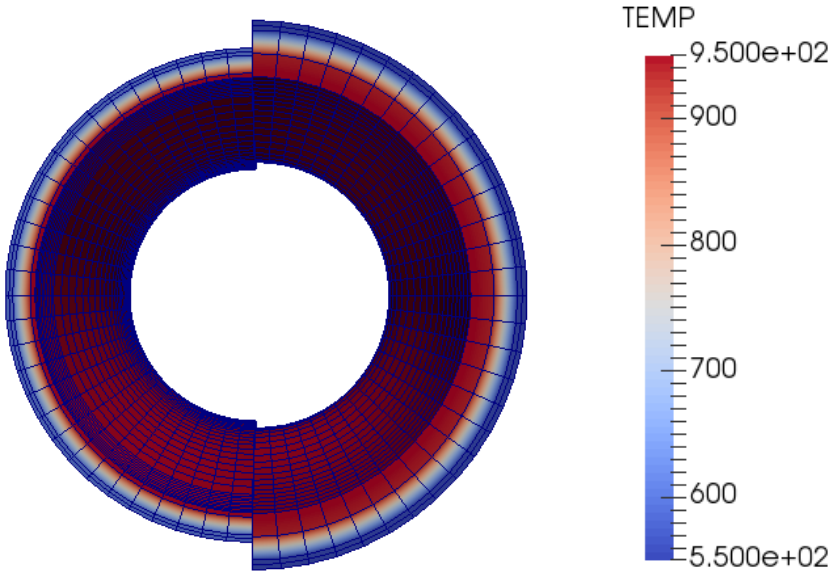


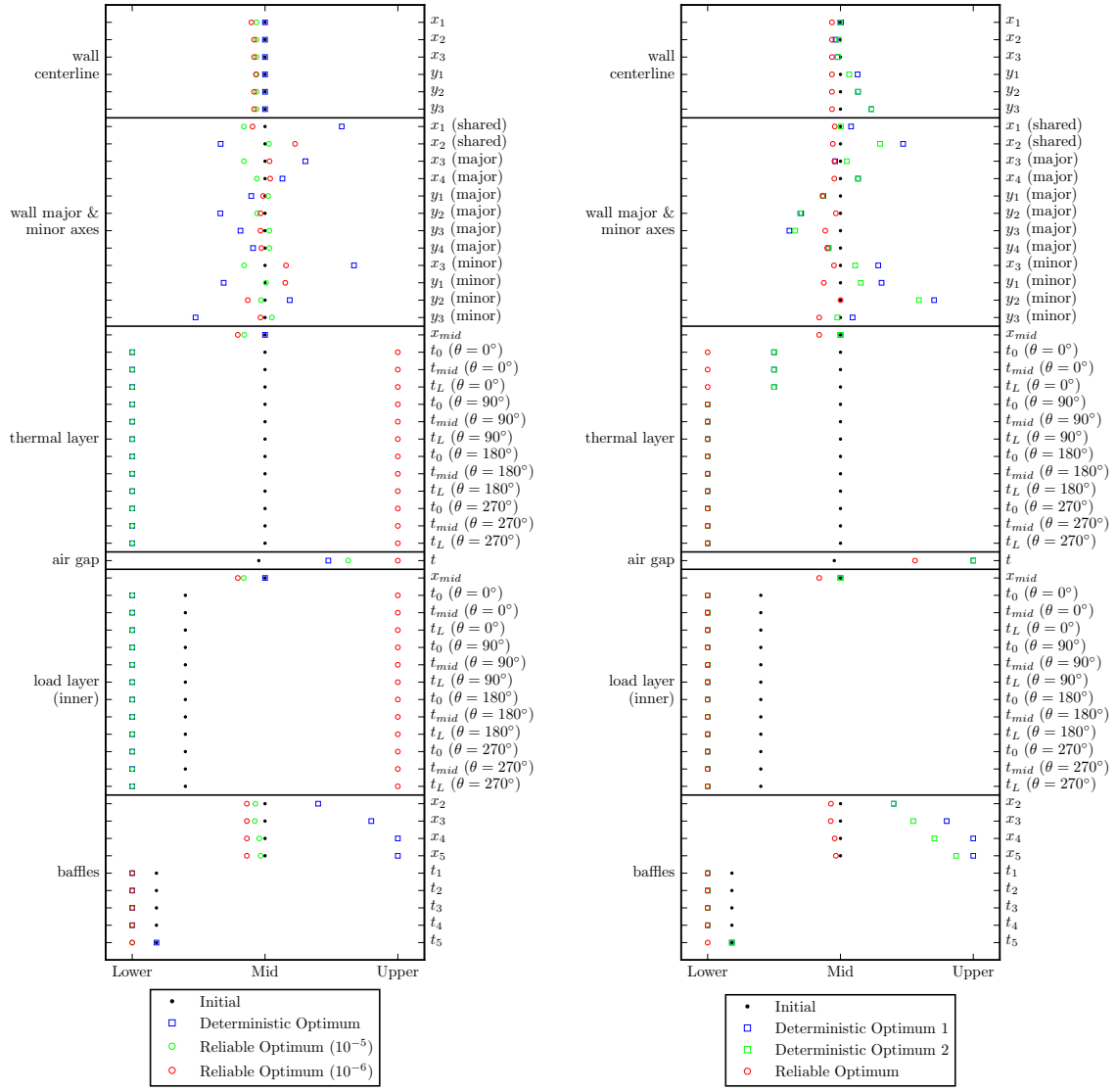
Figure 5.11: Comparison of temperature distribution in nozzle for the optimal standard practice design (left half) and the optimal reliable design ($\bar{p}_f = 10^{-6}$; right half) as seen from the exit of the nozzle looking towards the inlet. The temperature scales are the same allowing qualitative comparison. The thermal FEM mesh is shown in order to distinguish between the wall layers. Note the increased throat radius of the reliable nozzle, as well as increased thermal layer and air gap thickness, which leads to decreased temperatures in the load layer, as shown by its darker blue color.

standard practice designs choose the maximum thickness air gap in order to decrease temperatures. They also increase the thickness of the thermal layer near the inlet, where the temperatures are highest, to additionally decrease the load layer temperature. Figure 5.16 compares the nozzle geometry and temperature distributions at the inlet of the nozzle. The higher temperatures in the reliable nozzle indicate that the standard practice nozzle designs are more conservative than required.

The RBDO with the high-fidelity nozzle model uses the multifidelity sequential approach outlined in chapter 4. Only three iterations are required for convergence, requiring 8 low-fidelity model evaluations and 317 high-fidelity model evaluations. Despite the use of a multifidelity approach, the larger number of high-fidelity model evaluations is due to the need to perform reliability analyses with the high-fidelity nozzle model. All 317 high-fidelity model evaluations were required by AKIRA during four reliability analyses.

5.8 Discussion

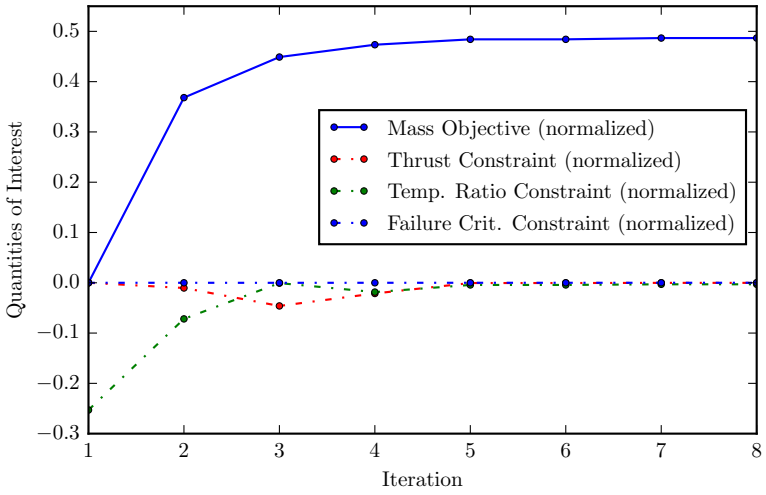
The optimization results for the nozzle system presented in section 5.7 show that reliability-based design optimization can be successfully applied to complex large-scale engineering systems. When efficient reliability analysis and optimization algorithms are used, such as those proposed in this



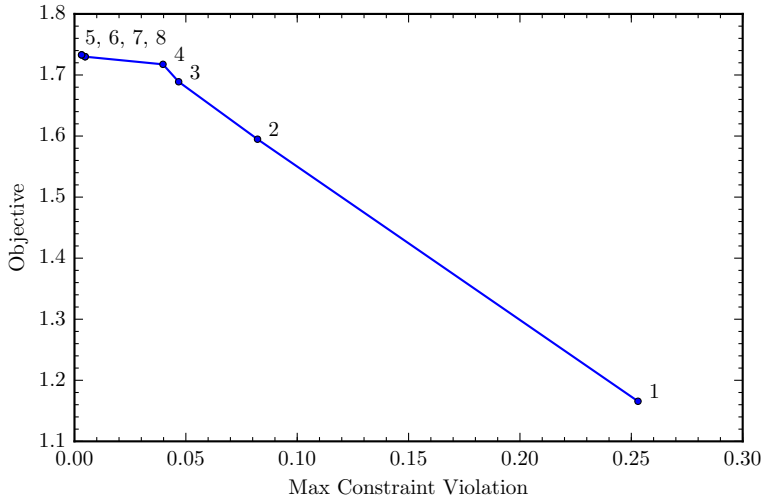
(a) Low-fidelity model.

(b) High-fidelity model.

Figure 5.12: Comparison of optimal design variables for the low-fidelity model and the high-fidelity model. For the low-fidelity model a standard practice deterministic optimum is compared with two reliable designs, one with prescribed failure probability $\bar{p}_f = 10^{-5}$ and one with $\bar{p}_f = 10^{-6}$. Note the quick transition in optimal reliable nozzle geometry between a probability of failure 10^{-5} and 10^{-6}, where the less reliable design ($\bar{p}_f = 10^{-5}$) features thin wall layers and a moderate air gap, and the more reliable design ($\bar{p}_f = 10^{-6}$) features thick wall layers and a maximum air gap. For the high-fidelity model two standard practice deterministic designs (design 1 with safety factor $\gamma = 1.5$ and design 2 with $\gamma = 1.2$) are compared with a reliable design with $\bar{p}_f = 10^{-6}$. Nearly all designs choose minimum gauge thickness wall layers; however the wider nozzle geometry for the standard practice designs leads to increased mass.

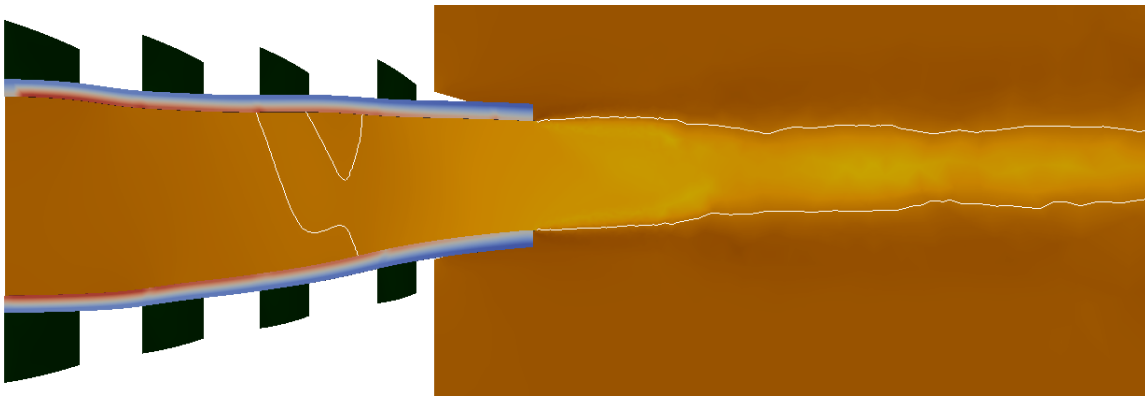


(a) Convergence of objective and constraints

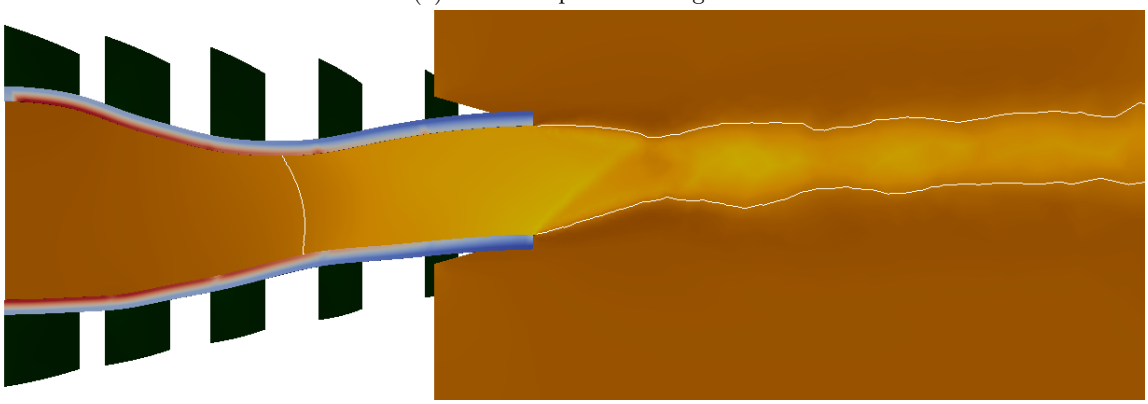


(b) Objective vs. constraint violation

Figure 5.13: Convergence of the low-fidelity nozzle RBDO using the multifidelity sequential RBDO approach (chapter 4) with AKIRA (chapter 3) for system probability of failure $\bar{p}_f = 10^{-6}$. The mean of the 14 random parameters is used as the anchor. 54 geometric deterministic design variables are considered. The low-fidelity model consists of a quasi-1D area-averaged aerodynamic analysis and coarse mesh FEM thermal and structural analyses. For a specified probability of failure of 10^{-6} the RBDO problem is actually not feasible as shown by the incremental improvements towards decreasing constraint infeasibility.



(a) Standard practice design.



(b) Reliable design.

Figure 5.14: Qualitative comparison of optimal nozzle geometry for the high-fidelity model and its effect on temperature and Mach number. The coupled aerothermostructural analysis is performed for the optimal nozzle geometry at the most probable point (MPP) of failure for the 14 random variables, *i.e.* for the combination of random variables that is most likely to yield system failure. The nozzles are shown split down their centerline with flow from left to right. Temperatures in the nozzle wall are shown (blue = cooler, red = hotter), as well as the Mach number (darker = smaller, lighter = larger). The color scales for both nozzles are the same, which enables a qualitative comparison between the optimal geometries. The sonic lines which mark where the Mach number is 1 in the fluid flow are shown as white contours. Since the inlet flow is subsonic, the flow after the sonic lines is supersonic. Note that a shock begins to form at the center of the nozzle behind the throat for the standard practice design in the off-design conditions at the MPP.

dissertation, the cost of solving such problems can become manageable. For the nozzle problem, the expense of the RBDO problem solution was only an order of magnitude more than that required for a deterministic optimization. At the same time, the solution of the RBDO problem yields a design that meets specified reliability requirements, which is generally difficult to reliably achieve with standard practice margin-based techniques such as safety factors and basis values. The optimal nozzles shown in section 5.7 were designed for a target probability of failure of 10^{-6}. In comparison,

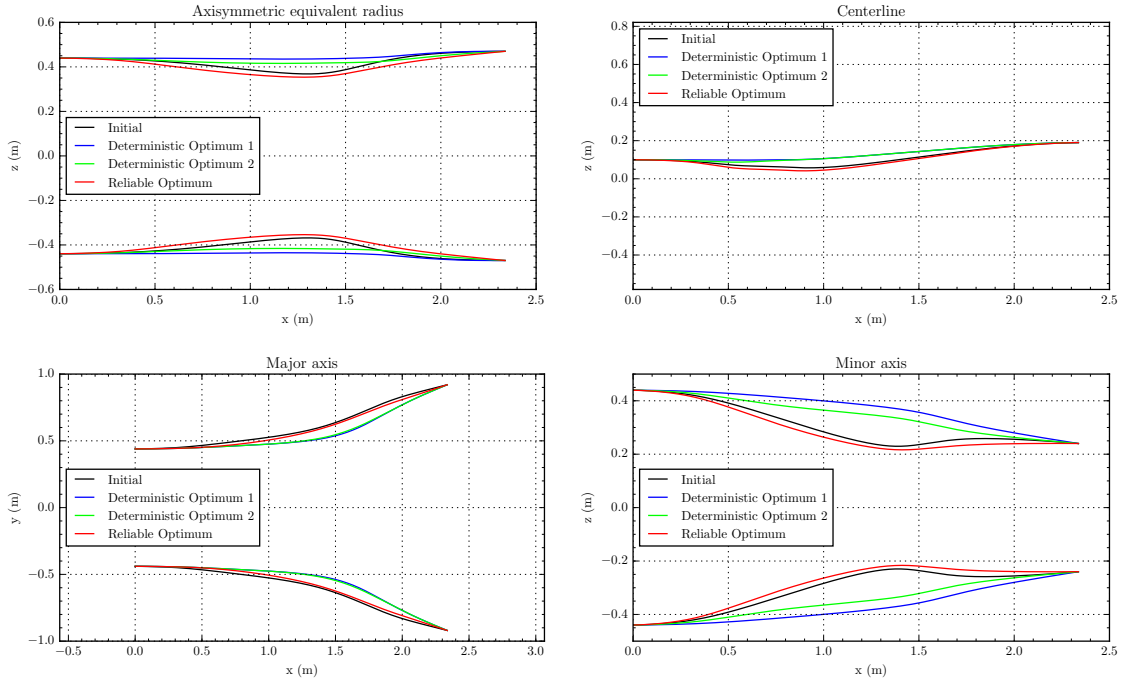


Figure 5.15: Comparison of optimal inner wall geometry for the high-fidelity model.

a safety factor and basis value approach led to under conservative designs for the low-fidelity model and over conservative designs for the high-fidelity model since these methods do not directly control reliability, although they do build in margin.

Why do the low-fidelity and high-fidelity models lead to different designs? While fundamentally both models are derived from the same physics, they differ in their approximation of the physics. The low-fidelity model features an area-averaged flow which is effectively 1D whereas the high-fidelity model features a fully 3D flow, although without viscous effects. The low-fidelity thermal and structural models are also coarser than the high-fidelity model. These differences lead to different predictions of the quantities of interest which ultimately affect the optimal design. For example, the thrust for the high-fidelity nozzle is lower than for an equivalently shaped low-fidelity nozzle. Since the high-fidelity model features a fully 3D flow, flow velocities in the nozzle have both an axial and radial component. The radial component does not contribute to the forward thrust force, and thus the estimated thrust is lower for the high-fidelity model. This is why the high-fidelity nozzle is more thrust critical than the low-fidelity nozzle, leading to its overdesign using standard practice methods.

The benefits of dimension reduction for the nozzle problem should also not be overlooked. The full nozzle RBDO problem featured 96 design variables, 40 random parameters, and 13 quantities of interest. Using AKIRA to build 13 kriging models in a 40 dimensional space, while possible, would

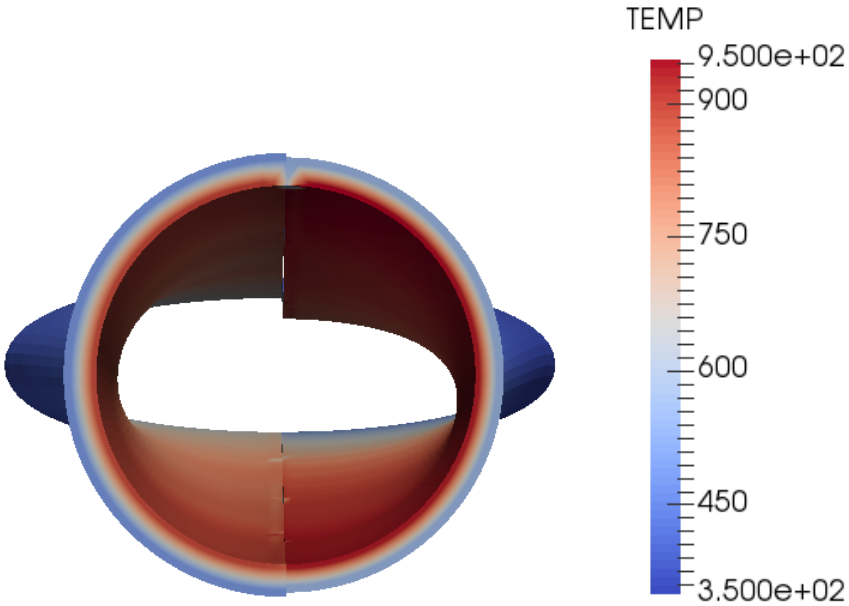
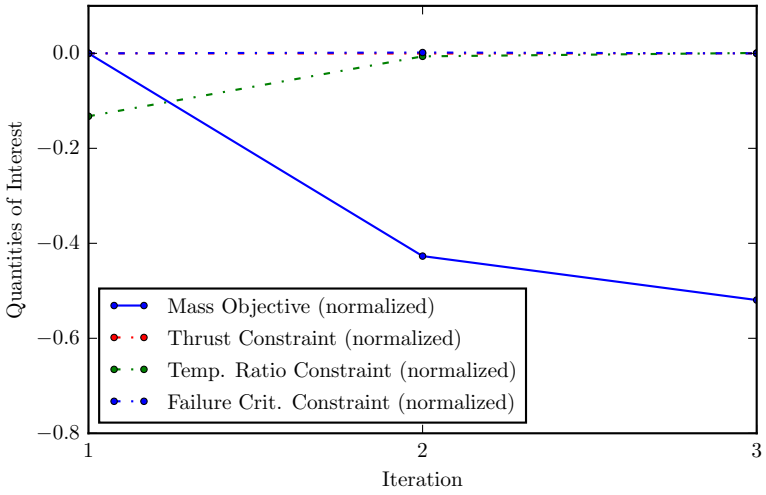


Figure 5.16: Comparison of temperature distribution in nozzle for the optimal standard practice design (left half) and the optimal reliable design ($\bar{p}_f = 10^{-6}$; right half) as seen from the inlet of the nozzle looking towards the exit. The temperature scales are the same allowing qualitative comparison. Note the decreased throat area of the reliable nozzle, as well as the increased thickness of the nozzle wall for the standard practice design. The standard practice design is overbuilt, and actually experiences lower temperatures at the most probable point of failure (MPP) as well as increased thrust, as compared to the reliable design.

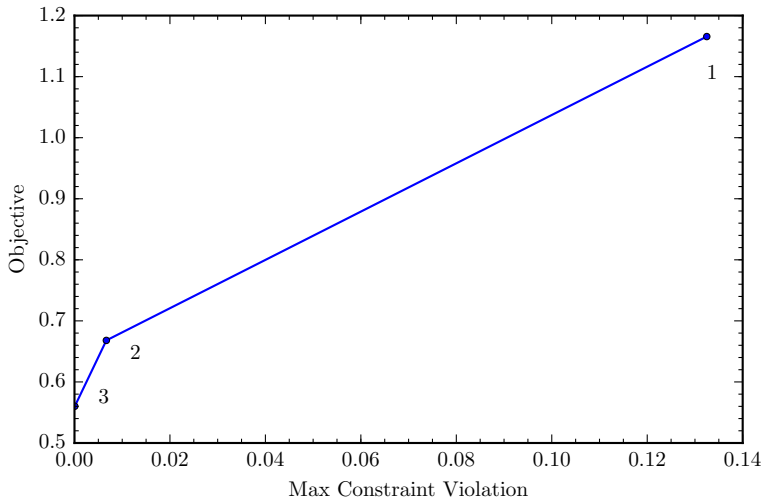
be challenging and certainly quite expensive. In addition, the optimization itself would be larger with more constraints and design variables, which would lead to increased expense when calculating finite difference gradients. Calculating Sobol' sensitivity indices to motivate dimension reduction by fixing random parameters that contributed a negligible variance to quantities of interest was a key step in making the solution of the nozzle RBDO problem possible. In doing so, the number of random variables dropped from 40 to 14 making the problem's solution more manageable.

AKIRA and the multifidelity sequential RBDO algorithm performed well on this larger more complex problem. Convergence of the optimization was relatively fast, requiring 8 iterations for the low-fidelity optimization and only 3 for the high-fidelity optimization. Even though all optimizations started with an initially infeasible design, and the low-fidelity RBDO problem was in fact infeasible, infeasible iterates did not pose any difficulties for the trust region method managing the RBDO algorithm.

An interesting feature of the multifidelity sequential RBDO algorithm for the nozzle problem is the large ratio of high-fidelity model evaluations compared to low-fidelity model evaluations. Typically one would expect the low-fidelity model to be used more frequently in order to minimize



(a) Convergence of objective and constraints



(b) Objective vs. constraint violation

Figure 5.17: Convergence of the high-fidelity nozzle RBDO using the multifidelity sequential RBDO approach (chapter 4) with AKIRA (chapter 3) for system probability of failure $\bar{p}_f = 10^{-6}$. The mean of the 14 random parameters is used as the anchor. 54 geometric deterministic design variables are considered. The high-fidelity model consists of a 3D Euler aerodynamic analysis and FEM thermal and structural analyses. Convergence is achieved in just three iterations.

the use of the more expensive higher-fidelity model. Although the low-fidelity model is used in this manner in the sequential RBDO method, a comparison of the function evaluation counts is

obfuscated since a reliability analysis must be performed each trust region iteration using the high-fidelity model in order to ensure convergence. No such reliability analysis need be performed for the low-fidelity model due to the nature of the anchored decomposition approximation. For the nozzle problem, the deterministic subproblems turn out to be relatively easy to solve (typically 2–5 outer SQP iterations are required for convergence), likely as a result of the many active bound constraints in the optimization. Thus, the majority of the RBDO cost is in the reliability analysis using AKIRA, which requires on the order of 100 high-fidelity function evaluations — a large number compared to the 5 function evaluations required during the solution of the subproblem using the low-fidelity model. For other problems, the behavior is different; for example in the cantilever beam problem from section 4.6.2, the majority of the RBDO cost lies in the solution of the deterministic subproblems.

The optimal nozzle designs, both for the low- and high-fidelity models, are qualitatively consistent with physical intuition, in general requiring larger throat areas to increase thrust as well as an increased air gap thickness to lower temperatures in the load layer. In addition minimum thickness wall layers are desired wherever possible in order to decrease mass. The phase transition in the optimal reliable low-fidelity nozzles between probabilities of failure 10^{-5} and 10^{-6} is particularly interesting. Such behavior highlights the importance of properly characterizing the input random variable distributions, which have a direct impact on the uncertainty in quantities of interest, as well as the importance of specifying appropriate reliability requirements. An improperly specified requirement, for example $p_f \leq 10^{-6}$ instead of 10^{-5} in the nozzle problem, has a large impact on the mass objective.

Indeed, the behavior observed for the optimal nozzle designs suggests a recipe for nozzle design:

1. Begin with minimum gauge wall layers, minimum air gap thickness, and a baseline nozzle shape.
2. Optimize the shape of the nozzle inner wall for minimum mass with respect to a reliability constraint on thrust.
3. Increase the air gap until the reliability constraint on temperature is met.
4. If the air gap reaches its maximum thickness, increase the thickness of the wall layers until the reliability constraint on temperature is met.

Of course, the recipe is dependent on the design and material properties but it effectively captures most of the "logic" of the reliability-based design optimizations presented in section 5.7.

Chapter 6

Conclusions

This dissertation has focused on exploring and solving the problem of designing reliable engineering systems. The question is often not so much how do we design a reliable system — engineers excel at adding margin to a design — but how do we design a system that meets a specific set of reliability requirements while maintaining or improving system performance. An overdesigned system may be extremely safe but also may have decreased performance and an underdesigned system may be hazardous to human life and livelihood. Design is about an appropriate balance between competing constraints and requirements, and reliability is a key aspect of that balance.

However, as was shown in chapters 1 and 2, estimating the reliability of a complex engineering system, which is necessary for its use in design, is expensive, even without performing a design optimization. For this reason, reliability-based design optimization has tended to focus on academic test problems and single-discipline applications, although there are exceptions [8, 5, 87]. This dissertation has made a modest contribution to the field of reliability-based design optimization by proposing a new Adaptive Kriging Inverse Reliability Analysis (AKIRA) algorithm for use in system RBDO, and a new method for multifidelity sequential RBDO using anchored decomposition; both are appropriate for complex systems. Both methods have been tested and demonstrated on academic test problems as well as a near industry-strength coupled supersonic nozzle design problem and have shown improvements over other commonly accepted methods used in RBDO.

In chapter 2 the concept of reliability was introduced and a survey of reliability analysis methods was made. Methods were separated into random sampling methods, most probable point (MPP) methods, response surface methods, and hybrid methods. Sampling methods, while accurate, can be prohibitively expensive for complex engineering systems due to the need for many limit state evaluations for the accurate estimate of a small probability of failure. However, they can be effectively used with response surface methods where limit states are approximated with a surrogate model and then cheaply sampled to obtain failure probability estimates. The kriging response surface is particularly promising due to the presence of an empirical error estimate, the kriging variance,

which can be used to guide response surface refinement. For this reason, the kriging response surface method was selected for use with AKIRA. However, the primary weakness of response surface methods is that the response surface may not accurately represent the limit state and can therefore lead to an incorrect failure probability estimate. As a result, hybrid methods were introduced which rely on the exact limit state function in regions where the response surface is deemed to be inaccurate. Classical MPP methods were also introduced for reliability analysis; they have a strong mathematical background and are an excellent choice for reliability analysis but are limited to linear (or quadratic) limit state surfaces, typically with a single MPP. Their accuracy degrades with limit state nonlinearities, especially in higher dimensions.

Chapter 2 also introduced the concept of the forward reliability analysis (reliability index approach) and the inverse reliability analysis (performance measure approach), the latter being more robust for application in RBDO. For this reason, AKIRA uses the inverse reliability analysis. The concept of multiple limit states was also addressed. In most engineering systems, reliability is defined with respect to several limit states and the failure probability of the system is often a key quantity of interest. For this reason, AKIRA also solves a system reliability analysis. Finally, design sensitivities for the probability of failure and quantile function were presented, as well as a derivation for design sensitivities when the technique of directional simulation is used for probability of failure integration.

Chapter 3 proposed the Adaptive Kriging Inverse Reliability Analysis algorithm for use in RBDO. The method builds independent kriging models of each limit state and adaptively refines them in regions contributing the most to the system failure probability estimate. Although a variety of methods could be used, the technique of directional simulation was chosen for failure probability integration since as a byproduct it yields a set of samples exactly on the limit state surface, which can be later used for limit state and design sensitivity approximation. Existing kriging-based reliability analysis methods were also surveyed, nearly all of which have been used to solve the forward reliability analysis problem. They differ primarily in their method of adaptive refinement for the kriging models. A new heuristic for model refinement inspired by directional simulation was also proposed for use with AKIRA and showed enhanced efficiency over EGRA, a commonly accepted kriging-based reliability analysis algorithm. The calculation of design sensitivities using AKIRA was also presented, including an adaptive cokriging method, which uses the refined limit state models from AKIRA to assist in the approximation of limit state sensitivities. Finally AKIRA was demonstrated on several academic test problems as well as a coupled supersonic nozzle problem in 14D. Comparisons with other reliability analysis methods show AKIRA has either increased accuracy or efficiency when solving the inverse reliability analysis problem. In addition, AKIRA solves the system inverse reliability analysis, which is uncommon among reliability analysis algorithms.

In chapter 4, the field of reliability-based design optimization was introduced. The formulation of the common minimum expectation RBDO problem was presented with both chance and quantile

constraints. Traditional methods for accounting for uncertainties were also discussed, including safety factors, partial safety factors, and basis values. Such methods do build margin into a design, but are unable to directly control the reliability of a system, unless the safety factors, basis values etc. are tailored for a specific design. In such situations, a reliability analysis or RBDO might as well be used. RBDO methods were also surveyed and classified as double-loop, sequential, and unilevel methods. Classical double-loop methods are usually the most expensive as they require a reliability analysis for each function evaluation in an outer optimization loop. On the other hand sequential methods separate the optimization and reliability analysis and solve a series of reliability analyses and deterministic optimizations until convergence. As a result, they tend to be more efficient. Finally unilevel methods decompose the reliability analysis into a set of equivalent conditions and embed this directly in the optimization formulation; they have been shown to be very efficient.

Chapter 4 also reviewed trust region methods for optimization which form an integral part of the proposed multifidelity sequential RBDO method. By relying on a corrected low-fidelity model during the solution of a series of optimization subproblems, an optimum of the high-fidelity model can be obtained. Trust region methods lend themselves well to multifidelity optimization as a cheaper lower-fidelity model can be used for the majority of the optimization. Finally a new method for sequential RBDO was proposed based upon the use of the anchored decomposition. The truncated anchored decomposition approximation effectively separates the deterministic design variables and the random parameters in the problem at an anchor point, thereby decoupling the reliability analysis and optimization. When embedded in a trust region method, the method has provable convergence. The algorithm was demonstrated on two academic test problems and shown to have comparable performance with other sequential RBDO methods, as well as being more efficient than double-loop methods. In addition, when the anchor point is chosen to be the MPP, the SORA method is recovered. However, other choices of anchors such as the random parameter means may be beneficial for coupled limit states in system reliability problems.

Finally chapter 5 introduced a coupled multidisciplinary supersonic nozzle design problem. The problem considered aerodynamics, heat transfer, and structures and designed the geometry of the complex nozzle system for minimum mass subject to a constraint on the system probability of failure, $p_f \leq 10^{-6}$. The physics of supersonic nozzles, the hierarchy of multifidelity models, and the parameterization of the geometry and 40 random variables were discussed. A full optimization problem with 13 quantities of interest, 96 design variables, and 40 random parameters was reduced with the aid of Sobol' sensitivity indices to a RBDO problem with 4 quantities of interest, 54 design variables, and 14 random parameters, thereby demonstrating an effective use of dimension reduction. The solution of the nozzle RBDO problem for various model fidelities was compared against designs using a standard practice deterministic approach for optimization consisting of basis values and safety factors to build in margin. The results for the nozzle RBDO problems demonstrate

the effectiveness of AKIRA and the multifidelity sequential RBDO algorithms on a realistic, high-dimensional problem, while also motivating the importance of RBDO as compared to standard deterministic design practices.

6.1 Contributions

Chapter 2 introduced the widely acknowledged benefits of performing the inverse reliability analysis over the forward reliability analysis during RBDO, however little work had been done on solving the inverse reliability analysis for the system reliability problem. The reason is likely because there is no guaranteed unique solution of the inverse problem when multiple limit states are present. As a result, many RBDO algorithms do not solve the system design problem, rather they solve a design problem with individual reliability constraints for each limit state. However the solution of the system design problem can lead to a design with better performance for equivalent system reliability. As a result, solving the system design problem is desired in many cases. The Adaptive Kriging Inverse Reliability Analysis (AKIRA) algorithm proposed in this dissertation addresses the gap described above: an *inverse* reliability analysis is performed for the *system*.

AKIRA mitigates the challenges above by using a constrained least squares problem that ensures a well-behaved solution to the inverse problem, even if it is not unique. The result can then be effectively used in quantile-constrained RBDO. AKIRA also takes advantage of several other techniques. As a kriging-based response surface approach to reliability analysis, it is efficient and demonstrates comparable performance to other kriging-based reliability analysis methods. A new heuristic for adaptive refinement of the kriging model for use in reliability analysis was also proposed and shown to give either more accurate or more efficient (less function evaluations) estimates of failure probabilities compared to EGRA, a well-accepted kriging-based reliability analysis method. Finally an adaptive cokriging method for design sensitivity estimation was also proposed. The new method enables efficient estimation of design sensitivities when limit state gradients are expensive to obtain, and when the most probable point is not sufficiently representative of the limit state contributions to the failure probability.

Chapter 2 also introduced a method for failure probability integration called directional simulation. Directional simulation appears to have not received as much notice as other integration methods such as importance sampling or subset simulation, likely due to the difficulty of robustly and cheaply performing the required linesearch for each simulated direction in the random space. The use of directional simulation in this dissertation for reliability analysis is not new, however the use of its results, namely a set of samples that lie exactly on the limit state surface(s) appears to be novel. These samples are used as an approximation to the limit state surfaces and aid in the calculation of design sensitivities during AKIRA. A new method for approximating the gradients of failure probability as a Riemann sum using the directional simulation samples was also proposed.

Finally the technique of directional simulation inspired the new adaptive kriging refinement heuristic used in AKIRA.

Chapter 4 presents a new multifidelity sequential method for RBDO based on anchored decomposition. The derivation of the sequential method is new and generalizes the idea of a limit state approximation based on a single point. Many sequential RBDO methods consider the approximation of the limit state at a single point, namely the MPP. Doing so requires a separate reliability analysis for each limit state to find each limit state's MPP. However, motivated by anchored decomposition, a shared point of approximation can be used instead such as the mean, thereby avoiding the need for multiple reliability analyses when there are multiple limit states. When the anchor (the point about which the approximation is based) is chosen to be the MPP in the proposed method, the existing Sequential Optimization and Reliability Assessment (SORA) method is recovered.

The proposed sequential RBDO method is further enhanced by using a trust region method to ensure convergence. Another enhancement is the introduction of lower-fidelity models to decrease overall RBDO cost, which leads to a multifidelity method. As a result, a high-fidelity reliability analysis need only be performed once every outer iteration. The proposed method was demonstrated on two academic test problems and shown to be more efficient than double-loop RBDO and comparable to existing sequential RBDO methods. Finally, the method was tested, in combination with AKIRA, on a coupled supersonic nozzle problem.

Finally, chapter 5 demonstrates the proposed algorithms in this dissertation on a coupled multidisciplinary supersonic nozzle design problem. The problem furnishes a fertile testbed for AKIRA and the multifidelity sequential RBDO method and shows that high-fidelity, complex RBDO problems can be solved in a reasonable amount of time with appropriate algorithm choice. In addition, the supersonic nozzle RBDO results are compared with standard practice deterministic designs and are shown to more effectively address reliability requirements, for only an order of magnitude increase in solution expense. Finally, the results obtained from the supersonic nozzle RBDO are informative for nozzle designers considering reliability and illustrative of the multidisciplinary behavior of the coupled nozzle system.

6.2 Future Work

The work presented in this dissertation is a stepping stone towards the application of reliability-based design optimization for complex engineering systems such as those found in industry. Here are a few thoughts regarding opportunities for improvement as well as future directions of research.

6.2.1 Philosophical Considerations

The question of what the reliability target for a given design should be must be answered prior to using RBDO. In this dissertation the question has largely been glossed over and reasonable characteristic failure probabilities have been prescribed in optimization formulations. Nevertheless, the determination of an appropriate reliability target for design warrants further serious investigation if the application of RBDO is to be useful [38]. Closely related is the identification and characterization of the uncertainties in a given design problem that give rise to uncertainty in a design's performance or safety. How do we account for the so-called "unknown unknowns" or even just the uncertainty in the estimated parameters of a probability distribution? Initial work in this direction co-authored by the author [36] has proposed a precision margin approach for RBDO which builds just enough margin into a design to meet reliability targets with a user-specified degree of confidence while accounting for parameter uncertainties. Distribution parameters aside, the choice of an uncertainty's chosen parameterization requires more investigation as well. Traditional methods use statistical tests such as the Anderson-Darling test [30] to determine goodness-of-fit, however even innocent changes to an uncertainty's parameterization as a lognormal or normal random variable (both of which might be statistically "good" fits) can have a significant impact on the optimal solution of a RBDO problem [38, 83]. As a result properly characterizing the uncertainties is paramount to the successful application of RBDO and requires further investigation.

The uncertainties present in a design problem also take a variety of forms — of course they can be separated, at times ambiguously, into aleatory and epistemic uncertainties [76], but in a real design problem they also often occur at different stages in design. It is well and good to account for uncertainties in a design's operating environment, material properties, and mathematical models, but what about uncertainties in the design process itself and its various stages? Acar takes an initial step in this direction by investigating the impact of flight tests on the design and reliability of aircraft, in addition to accounting for standard uncertainties in the engineering design such as material properties [1]. A more holistic view and use of RBDO is likely to yield better performing and more reliable designs in practice.

6.2.2 Directional Simulation

This dissertation has relied heavily on the technique of directional simulation for probability of failure integration, most notably during the AKIRA algorithm. Although directional simulation has improved convergence compared to Monte Carlo, a convenient interpretation in the standard normal space, and generates a set of samples exactly on the limit state surface, its use in this dissertation could benefit from several improvements.

Further investigation of methods for solving the linesearch in directional simulation is required. Fortunately in this dissertation, an exhaustive linesearch could be performed on a kriging surrogate model that was cheap to evaluate, but even then efficiency improvements to the linesearch could have

large benefits for the speed of the directional simulation procedure. Reliably finding multiple roots during the linesearch is key for directional simulation to be competitive with methods such as Monte Carlo when limit states are highly nonlinear. In this dissertation a pattern search was employed for the line search with 200 points sampled along each direction. For applications with limit state functions that are expensive to evaluate or to increase efficiency, more advanced linesearch techniques can be used. For example, efficient global optimization [75] or other methods could enhance both the efficiency and accuracy of directional simulation by finding multiple root locations with a small number of limit state evaluations.

Further investigation for enhancing the convergence rate of directional simulation is also required. Ditlevsen and Madsen suggest that the convergence of the directional simulation estimate can be enhanced by adjusting the mean of the sampling distribution, however no general rule for doing so currently exists [38]. In addition, directional importance sampling has been previously studied [20, 37, 88] and can be used to increase convergence rate. Further research in this area would be welcome. Finally, deterministic point sets could be used for integration and could reduce the variance of the directional simulation estimate [91, 92]. Methods for efficiently generating deterministic point sets and/or incorporating them in a hybrid adaptive importance sampling framework could be beneficial.

6.2.3 AKIRA

The Adaptive Kriging Inverse Reliability Analysis (AKIRA) algorithm presented in this dissertation has shown that system inverse reliability analysis problems can be solved for use in RBDO problems. However there are a variety of opportunities for improvement of the algorithm. First and foremost is speed — AKIRA requires additional computation time compared to other reliability analysis methods due to the need to perform multiple probability of failure integrations on a kriging response surface during the solution of the inverse reliability analysis problem. Improved failure probability integration strategies can help alleviate some of the burden; the investigation of subset simulation is particularly interesting due to its robust and efficient method for approximating small probabilities of failure [13]. However, intelligent reuse of samples in areas where the kriging model has not substantially changed between iterations may also lead to greatly increased performance.

AKIRA is also currently limited to solving the system reliability analysis problem. Its extension for solving component reliability analysis problems is fairly straightforward and will be investigated in the future. AKIRA has also chosen to model individual limit states with independent kriging models. In fact, limit states may be correlated and approximating their correlation can lead to an improved kriging model for system reliability analysis. For example the Efficient Kriging Surrogate modeling Approach (EKSA) recently proposed a method for accounting for these correlations through the SVD of the sample data matrix [72]. Similar ideas can be used within AKIRA.

Finally the investigation of different refinement strategies for the kriging models in AKIRA,

beyond what was presented in this dissertation, is an important direction for future work. The proposed heuristic inspired by directional simulation can be further improved by the use of a clustering mechanism for simulated directions. Instead of selecting the sampled direction with the greatest contribution to the failure probability estimate, select a direction that corresponds to a cluster of directions which contribute the most to the probability of failure estimate. Such an improvement can additionally help avoid excessive refinement in regions where the limit state model is fairly accurate but also contributes a lot to the probability of failure. The further investigation of Stepwise Uncertainty Reduction (SUR) strategies for adaptive refinement is also necessary, as these methods promise an optimal refinement strategy although with increased expense [16, 26, 28].

6.2.4 Reliability-Based Design Optimization

The proposed multifidelity sequential method for RBDO has shown comparable or enhanced performance to existing RBDO methods. However there are several interesting areas for further investigation that could yield additional benefits. The first is reformulating the RBDO problem in the reliable design space [42] where the random space and deterministic design space are combined. Doing so would require modifications to AKIRA which would then build kriging models over the augmented reliability space instead of the random space. Such an augmented kriging model would yield more easily obtained design sensitivities and also potentially make solving the RBDO problem less expensive. As a result, a separate reliability analysis, which otherwise would require building a kriging model from scratch for each new design iterate, would not be needed each iteration — instead the existing kriging model could be refined each iteration. The challenge of working in the augmented reliability space however are its increased dimension due to the combination of the deterministic and random spaces.

The proposed sequential method for RBDO could also be further extended by considering multiple anchor points, either for when multiple MPPs are present in the random space, or simply for increasing the number of information sources used to guide the solution of the deterministic subproblem. The selection of multiple anchor points for RBDO and their weights is currently an open question. The anchor points could also be allowed to vary during the solution of the deterministic subproblem as was the case in [14] where the MPP was replaced with an approximation dependent on the design variables. Allowing the anchor point(s) to vary can possibly enhance convergence by providing a better estimate of limit state behavior during the solution of the deterministic subproblems. Finally, the bifidelity sequential RBDO algorithm could be extended to incorporated multiple models with different fidelity, as well as consider different trust region implementations. Of particular interest are improved corrections for the lower-fidelity models. Multipoint kriging corrections for example could further enhance convergence of the RBDO algorithm by incorporating the design history of the optimization.

6.2.5 Application to Realistic Design Problems

Finally, the benefits of RBDO and its applicability for design today require further illustration through the demonstration of more RBDO methods on additional realistic design problems, thereby further motivating its use in industrial applications. This dissertation showcases a coupled supersonic nozzle application, however additional details could be added to make a more convincing case for RBDO. A fully coupled nozzle analysis would be ideal; this dissertation only considers a one-way coupled analysis, although this appears to be common practice in industry. In addition, the inclusion of geometric uncertainties and additional characterization of uncertainties such the inlet temperature distribution as a function of nozzle radius etc. are important improvements for an even more realistic RBDO problem. In addition, a multipoint optimization problem considering different design points in the flight envelope is a practical next step for realistic applications. The use of coupled adjoint gradients would greatly enhance the capabilities of the optimization; however their investigation and use is left for other dissertations.

The use and role of dimension reduction in the RBDO process has not been extensively investigated in this dissertation, however it is also an important aspect of applying RBDO to realistic design problems. Further investigation of methods such as active subspaces [32], proper orthogonal decomposition, etc. and their inclusion in the RBDO problem is a fertile area for future research. If indeed a wide variety of physical problems have low-dimensional structure [105] as appears to be the case in many instances [32], the opportunities for accelerating RBDO with dimension reduction techniques are considerable.

Appendix A

Applications

A.1 Short Column

Variations of the short column problem are commonly used in the RBDO literature to demonstrate reliability analysis and RBDO methods [77, 111, 3, 11, 42, 83]. The problem concerns a short column with rectangular cross-section subject to two bending moments M_1 and M_2 and an axial load P. In the version considered here [11], the loading and the column's yield strength Y are considered to be independent normal random variables. In addition the breadth b and height h of the column's cross-section are considered to be independent normal random variables; thus $y = [M_1, M_2, P, Y, b, h]^\top$. The deterministic design variables are the means of the geometric parameters; thus $x = [\mu_b, \mu_h]^\top$ with initial values of 500 mm each. Table A.1 describes the variables in the problem.

The normalized limit state ($g(x, y) \leq 0$ corresponds to failure) which governs yielding in the column is

$$g(x, y) = 1 - \frac{4M_1}{bh^2Y} - \frac{4M_2}{b^2hY} - \left(\frac{P}{bhY}\right)^2. \tag{A.1}$$

Table A.1: Variables in the short column problem.

Name	Variable	Distribution	Mean	CoV
Moment 1 (kN-m)	M_1	Normal	250	0.3
Moment 2 (kN-m)	M_2	Normal	125	0.3
Axial load (kN)	P	Normal	2500	0.2
Yield strength (MPa)	Y	Normal	40	0.1
Breadth (mm)	b	Normal	x_1	0.1
Height (mm)	h	Normal	x_2	0.1

Table A.2: Variables in the steel column problem.

Name	Variable	Distribution	Mean	Std. Dev.
Yield stress (MPa)	F_s	Lognormal	400	35
Dead weight load (kN)	P_1	Normal	500	50
Variable loads (kN)	P_2, P_3	Gumbel	600	90
Flange breadth (mm)	B	Lognormal	200	3
Flange thickness (mm)	D	Lognormal	17.5	2
Profile height (mm)	H	Lognormal	100	5
Initial deflection (mm)	F_0	Normal	30	10
Elastic modulus (GPa)	E	Weibull	21	4.2

A.2 Steel Column

The steel column problem considers the yielding of a T-shaped steel column with geometric uncertainties when subjected to several uncertain loads [77]. There are nine random variables $y = [F_s, P_1, P_2, P_3, B, D, H, F_0, E]^\top$ described in table A.2. The length of the column is fixed at 7.5 m. This problem can pose difficulties for reliability analysis methods as the limit state is poorly behaved when D is small and P_1, P_2, and P_3 are large.

The limit state ($g(y) \leq 0$ corresponds to failure) which governs yielding in the column is given by

$$g(y) = 1 - F_s - P\left(\frac{1}{2BD} + \frac{F_0}{BDH}\frac{E_b}{E_b - P}\right), \tag{A.2}$$

where P and the Euler buckling load E_b are given by

$$P = P_1 + P_2 + P_3, \tag{A.3a}$$

$$E_b = \frac{\pi^2 EBDH^2}{2L^2}. \tag{A.3b}$$

A.3 Cantilever Beam

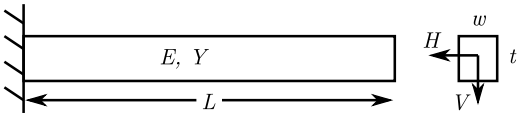


Figure A.1: Schematic for the proposed cantilever beam problem subject to a lateral and vertical load [132, 126]. Elastic modulus E, yield strength Y, and horizontal and vertical loading H and V are uncertain. $L = 100$ inches and w and t are deterministic design variables.

The cantilever beam problem is illustrated in Figure A.1 [132, 126]. Its four random variables $y = [Y, E, H, V]^\top$ are summarized in Table A.3. The width w and thickness t of the beam's rectangular

Table A.3: Random parameters in the cantilever beam problem.

Name	Variable	Distribution	Mean	CoV
Yield strength	Y	Normal	40,000	0.05
Elastic modulus	E	Normal	2.9×10^7	0.05
Horizontal load	H	Normal	500	0.2
Vertical load	V	Normal	1,000	0.1

cross-section are considered as deterministic design variables. Two limit states are considered: one related to stress and one related to displacement. The stress S in the beam is given by

$$S(x, y) = \frac{600}{wt^2}V + \frac{600}{w^2t}H, \tag{A.4}$$

whereas the displacement of the beam D is given by

$$D(x, y) = \frac{4L^3}{Ewt}\sqrt{\left(\frac{V}{t^2}\right)^2 + \left(\frac{H}{w^2}\right)^2}. \tag{A.5}$$

Failure occurs when the stress S exceeds the yield stress Y or the displacement D exceeds $D_0 = 2.2535$ inches. The limit states are normalized such that a value less than zero corresponds to failure. The normalized limit states $g_s(x, y)$ and $g_D(x, y)$ are given by

$$g_S(x, y) = 1 - \frac{S(x, y)}{Y}, \tag{A.6}$$

$$g_D(x, y) = 1 - \frac{D(x, y)}{D_0}. \tag{A.7}$$

A.4 Convex Limit State

This convex limit state in the standard normal space was introduced by Borri and Speranzini [21]:

$$g(u) = 0.1(u_1 - u_2)^2 - \frac{u_1 + u_2}{\sqrt{2}} + 2.5. \tag{A.8}$$

A.5 Highly Nonlinear Limit State

The following limit state was introduced by Grandhi and Wang [65]:

$$g(y) = 2.5 + 0.00463(y_1 + y_2 - 20)^2 - 0.2357(y_1 - y_2), \tag{A.9}$$

where $y_i \sim \mathcal{N}(10, 3^2)$, $i = 1 \ldots 2$ and are uncorrelated.

Appendix B

Kernels

B.1 Squared Exponential

A common choice for the covariance kernel in engineering applications due to its suitability for representing smooth functions is the squared exponential [108] or radial basis function covariance function

$$k(x,y) = \sigma_f^2 \exp\left(-\frac{1}{2}(x-y)^\top M(x-y)\right), \tag{B.1}$$

where M is a symmetric matrix and the hyperparameters are defined as $\theta = (M, \sigma_f^2)$. When $M = \ell^{-2}I$, the radial basis function is isotropic, that is it makes an assumption of isotropy on the characteristic length scale ℓ variation in all dimensions of the input space. When $M = \text{diag}\left([\ell_1 \dots \ell_n]^\top\right)^{-2}$, a different characteristic length scale ℓ_i is associated with each dimension and the radial basis function covariance matrix is anisotropic.

B.2 Matérn

The Matérn kernel is given by [108]:

$$k(r) = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}r}{\ell}\right)^\nu K_\nu\left(\frac{\sqrt{2\nu}r}{\ell}\right), \tag{B.2}$$

where $\nu, \ell > 0$ are hyperparameters, r is the distance between the points whose covariance is being estimated, and K_ν is a modified Bessel function. ν is a smoothness parameter; when $\nu = 1.5$ an assumption of a once differentiable function is made. In this case

$$k_{\nu=1.5}(r) = \left(1 + \frac{\sqrt{3}r}{\ell}\right)e^{-\frac{\sqrt{3}r}{\ell}}. \tag{B.3}$$

B.3 Multivariate Matérn

The bivariate Matérn kernel is constructed from univariate Matérn kernels $k(r \mid \nu, \ell)$ [62, 29] as

$$k_{11}(r) = \sigma_1^2 k(r \mid \nu_1, \ell_1) \tag{B.4a}$$

$$k_{22}(r) = \sigma_2^2 k(r \mid \nu_2, \ell_2) \tag{B.4b}$$

$$k_{12}(r) = \rho_{12}\sigma_1\sigma_2 k(r \mid \nu_{12}, \ell_{12}), \tag{B.4c}$$

where $|\rho_{12}|$ must satisfy the conditions stated in [62] of the form $|\rho_{12}| \leq f(\nu_1, \nu_2, \nu_{12}, \ell_1, \ell_2, \ell_{12})$. There are a total of 9 hyperparameters in the bivariate Matérn model, not including nugget effects. The full bivariate Matérn model is

$$k(r) = \begin{bmatrix} k_{11}(r) & k_{12}(r) \\ k_{12}(r) & k_{22}(r) \end{bmatrix}. \tag{B.5}$$

Appendix C

Optimization Theory

This appendix accompanies chapter 4 and reviews general concepts from optimization theory. The necessary conditions for optimality are briefly reviewed as well as a practical discussion on termination criteria for algorithms. Then basic concepts for common gradient-based optimization algorithms are introduced including augmented Lagrangian methods and SQP methods. Algorithms based on the SQP method have been used in both the solution of AKIRA's inverse reliability analysis problem as well as the reliability-based design optimization problems presented in chapters 4 and 5. For a broad overview of optimization theory including detailed derivations and algorithm spefications, the reader is referred to any general text on optimization theory; see for example Nocedal and Wright [93] or Conn, Gould, and Toint [31].

C.1 Optimality Conditions

Here the necessary conditions for optimality are stated for reference as they are used to assess termination of optimization algorithms used in later chapters. We consider optimality conditions for the inequality constrained optimization problem from (4.1) restated below:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{s.t.} \quad & g(x) \geq 0. \end{aligned} \tag{4.1}$$

The first order optimality conditions, also known as the *Karush-Kuhn-Tucker* (KKT) conditions [93], are

$$\nabla_x f(x^*) = \nabla_x g(x^*)^\top \lambda, \tag{C.1a}$$

$$g_i(x^*) \geq 0 \text{ for all } i = 1 \dots m, \tag{C.1b}$$

$$\lambda_i \geq 0 \text{ for all } i = 1 \dots m, \tag{C.1c}$$

$$g_i(x^*)\lambda_i = 0 \text{ for all } i = 1 \dots m, \tag{C.1d}$$

where $\nabla_x g(x)$ is the $m \times n$ Jacobian matrix of the constraints and λ is a vector of Lagrange multipliers. The first condition in (C.1a) is known as *stationarity* and requires that the gradient of the Lagrangian is zero at an optimum. (C.1b) states that the solution must be feasible at an optimum and (C.1c) requires non-negativity of the Lagrange multipliers. Finally, (C.1d) is the *complementary slackness* condition. A point that satisfies the conditions in (C.1) is known as a *first-order KKT point*.

Note that the first condition in (C.1a) shows the objective gradient can be written as a linear combination of constraint gradients at an optimum. An equivalent condition is thus that the objective gradient be perpendicular to the null-space of the constraint gradients. This is succinctly written using the *reduced gradient* $Z^\top \nabla_x f(x)$ where Z is a matrix whose columns form a basis for the null-space of the active constraint gradients $\nabla_x g(x)$. Thus the stationarity condition is equivalent to requiring $Z(x^*)^\top \nabla_x f(x^*) = 0$ at an optimum.

Note that the KKT conditions given in (C.1) are valid only when first-order perturbations are sufficient for assessing optimality [31]. For application to nonlinear optimization they require a *constraint qualification*; the two commonly used are the linear independence constraint qualification (LICQ) and the weaker Mangasarian-Fromovitz constraint qualification (MFCQ) [93]. The LICQ specifies that at a feasible point the gradients of the active inequality constraints should be linearly independent which means the Jacobian matrix has full row rank. The MFCQ specifies that at a feasible point a direction exists where all active constraint gradients are positive.

Unless the optimization problem is convex, second-order optimality conditions must be included for nonlinear optimization problems to ensure optimality. Such a condition uses the Hessian of the Lagrangian defined as

$$H(x, \lambda) = \nabla^2 f(x) - \sum_{i=1}^{m} \lambda_i \nabla^2 g_i(x). \tag{C.2}$$

The second-order optimality condition is

$$p^\top H(x^*, \lambda^*) p \geq 0 \quad \text{for all } p \text{ such that } \nabla_x f(x^*)^\top p = 0, \tag{C.3}$$

which is equivalent to requiring the reduced Hessian of the Lagrangian $Z(x^*)^\top H(x^*, \lambda^*) Z(x^*)$ to be positive semidefinite [61].

C.2 Termination Conditions

Practical termination conditions for an optimization should be robust and account for a variety of situations including the precision τ_f in the user-provided objective and constraint functions. The termination criteria stated here primarily come from [61].

A tolerance on the constraint violation should be specified to ensure feasibility:

$$\|g(x)\| \leq \tau_c. \tag{C.4}$$

Likewise a tolerance on the the optimality should be specified:

$$\left\|\nabla_x f(x) - \nabla_x g(x)^\top \lambda\right\| \leq \sqrt{\tau_f}, \tag{C.5}$$

where the norm of the reduced gradient for active constraints $g(x)$ should be less than the square root of the function precision τ_f. In the case that no constraints are active, *i.e.* the optimum lies in the interior of the feasible region, the optimality condition may be expressed as

$$\frac{\|\nabla_x f\|}{1 + |f|} \leq \sqrt[3]{\tau_f}, \tag{C.6}$$

where Gill et al recommend the relaxed cubed root instead [61]. The normalization of the objective gradient avoids difficulties when the objective is poorly scaled. It is also beneficial to include a check on the size of the objective gradient if the initial point of the optimization is close to the optimum:

$$\|\nabla_x f\| \leq \tau_f. \tag{C.7}$$

C.2.1 Soft Convergence

So-called soft convergence termination criteria measure for diminishing returns in the optimization progress. A standard cause of such issues are noisy or incorrect gradients or function evaluations.

A measure of the progress for the objective function is helpful.

$$\frac{f^{k-1} - f^k}{1 + |f^k|} < \tau_f, \tag{C.8}$$

as well as a measure of the progress in the design variables such as

$$\frac{\|x_{k-1} - x_k\|}{1 + \|x_k\|} < \sqrt{\tau_f}. \tag{C.9}$$

In certain cases requiring the soft convergence criteria to be met for several consecutive iterations (2 or 3) prior to termination can help ensure soft convergence has been achieved [48].

C.3 Augmented Lagrangian Methods

An augmented Lagrangian algorithm incorporates a problem's constraints into the objective function to yield an unconstrained optimization problem while avoiding the ill-conditioning and lack of differentiability issues that occur with penalty and barrier methods [61, 31]. In doing so, existing efficient solvers for unconstrained optimization can be used to solve the constrained problem.

The most common augmented Lagrangian function L_A is

$$L_A(x, \lambda, \rho) \equiv f(x) - \lambda^\top g(x) + \frac{\rho}{2} g(x)^\top g(x), \tag{C.10}$$

where $g(x)$ is the vector of the active nonlinear constraints. λ is the vector of Lagrange multipliers associated with each active constraint and ρ is a positive penalty parameter. For a sufficiently large but finite value of ρ, a local unconstrained minimum x^* is guaranteed to exist for $L_A(x, \lambda^*, \rho)$ where λ^* are the Lagrange multipliers at the optimum [61]. Since λ^* and the required sufficiently large value of ρ are *a priori* unknown, any augmented Lagrangian method must develop estimates of both. Algorithm 7 outlines a simple augmented Lagrangian algorithm based on [61].

Algorithm 7: Basic augmented Lagrangian optimization algorithm

Data: initial guess x^0, initial penalty parameter ρ^0, initial multiplier estimate λ^0, initial set of active constraints g^0, $\alpha \in (0, 1]$, $\beta \in (1, 10+]$

Result: x^*, $f(x^*)$

for $k \leftarrow 0$ **to** k_{max} **do**
  if x^k *optimal* **then**
    return x^k, $f(x^k)$;
  $x^{k+1} \leftarrow$ Minimize the augmented Lagrangian given (x, λ^k, ρ^k);
  $g^{k+1} \leftarrow$ Update the active set of inequality constraints;
  $\lambda^{k+1} \leftarrow$ Update the Lagrange multipliers;
  if $\left\|g^{k+1}(x)\right\| \geq \alpha \left\|g^k(x)\right\|$ **then** `// sufficient decrease condition`
    $\rho^{k+1} \leftarrow \beta \rho^k$;
  else
    $\rho^{k+1} \leftarrow \rho^k$;
return x^{k+1},$f(x^{k+1})$;

The Lagrange multipliers λ^k may be estimated using a variety of different methods; first-order and second-order methods are covered in [61] however note that second-order convergence will require a second-order estimate [61, 59]. The choice of α and β can affect convergence if the initial penalty parameter ρ is non-optimal as they govern the rate at which ρ is adjusted. $\alpha = 0.9$ and $\beta = 10$ seem to be typical values.

The solution of the subproblem minimizing the augmented Lagrangian L_A can be obtained using any standard nonlinear unconstrained minimization algorithm. In particular, if the Hessian of the objective and constraints are available, efficient Newton-type methods can be used leading to quadratic convergence close to the optimum.

C.4 Sequential Quadratic Programming

In sequential quadratic programming (SQP) methods a series of subproblems consisting of a quadratic approximation to the Lagrangian subject to linearized constraints are solved. SQP methods may also be viewed as solving the KKT system via application of Newton's method [93]. They are generally considered to be one of the most efficient optimization methods for nonlinearly constrained optimization problems [93]. For the inequality-constrained optimization problem specified in (4.1) with objective $f(x)$ and constraints $g(x)$ each subproblem is typically written as [61, 93]:

$$\begin{aligned} \underset{p}{\text{minimize}} \quad & \nabla_x f(x^k)^\top p + \frac{1}{2} p^\top H(x^k, \lambda^k) p \\ \text{s.t.} \quad & \nabla_x g(x^k)^\top p + g(x^k) \geq 0, \end{aligned} \tag{C.11}$$

where x^k is the current iterate, λ^k are the associated Lagrange multipliers, and $H(x^k, \lambda^k)$ is the Hessian of the Lagrangian which is often replaced with a quasi-Newton approximation of the Hessian such as a BFGS update. A variety of methods exist for solving the quadratic subproblem in (C.11); for example the null-space approach is popular in many implementations. Additional details can be found in [93] or [61]. Algorithm 8 gives the basic form of an SQP algorithm. A practical implementation additionally requires a strategy for dealing with non-positive definite $H(x^k, \lambda^k)$ as well as a choice of merit function to ensure progress towards an optimal point. These details may be reviewed in any standard optimization text, see for example [61, 93, 31].

Algorithm 8: Basic sequential quadratic programming algorithm

Data: initial guess x^0, initial multiplier estimate λ^0, initial set of active constraints g^0
Result: x^*, $f(x^*)$

```
for k ← 0 to k_max do
    Evaluate objective and constraint values, gradients, and Hessian at x^k;
    if x^k optimal then
        return x^k, f(x^k);
    p^k, λ^{k+1} ← Solve the quadratic programming subproblem;
    x^{k+1} ← x^k + p^k;
    g^{k+1} ← Update the active set of inequality constraints;
return x^{k+1}, f(x^{k+1});
```

Appendix D

Nozzle Details

This appendix describes various details related to the implementation and parameterization of the nozzle model used in the nozzle design problem from chapter 5.

D.1 Basis-Splines

The inner wall of the nozzle is parameterized using a 3rd degree basis spline (B-spline). Although determination of a B-spline's shape requires the numerical solution of a recursive algorithm, B-splines have several nice properties including 1) they are a generalization of all splines, 2) in general, overfitting is avoided, and 3) any segment lies in the convex hull of neighboring control points [102]. A B-spline $s(t)$ is defined as

$$s(t) = \sum_{i}^{N} c_i N_i^n(t), \tag{D.1}$$

where there are N control points c_i and N basis functions N_i of order n. The basis functions depend on the additional definition of a knots vector and can be determined using de Boor's algorithm [102]. In this work, the knots vector is assumed to be fixed and equally spaced. The nozzle's shape is changed by altering the coefficients c_i of the B-spline.

D.2 Material Properties

Material properties are specified using the local coordinate system in figure D.1. An incomplete set of material properties was gathered from a variety of experimental data sources. Missing macroscopic properties are either determined through fundamental physical analysis of the material's structure or are approximated by values for similar materials. Table D.1 provides the properties for the ceramic matrix composite (CMC) material, table D.2 for the graphite/bismaleimide (GR-BMI) material,

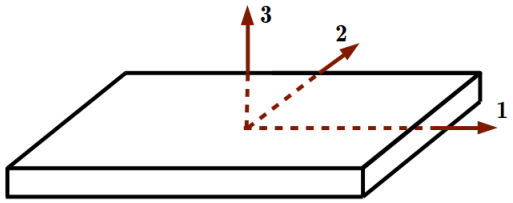


Figure D.1: Local coordinate system for specification of material properties in shell materials.

Table D.1: Isotropic material properties for the thermal layer's ceramic matrix composite.

Property	**symbol**	**Units**	**Value**	**Distribution**
Density	ρ	$\frac{\text{kg}}{\text{m}^3}$	2410	$\ln\mathcal{N}(7.7803, 0.0182^2)$
Elastic modulus	E	GPa	67.1	$\ln\mathcal{N}(4.2047, 0.0551^2)$
Poisson ratio	ν		0.33	$\mathcal{U}(0.23, 0.43)$
Thermal conductivity	k	$\frac{\text{W}}{\text{m-K}}$	1.41	$\mathcal{U}(1.37, 1.45)$
Thermal expansion coefficient	α	$\text{K}^{-1} \times 10^{-6}$	0.24	$\mathcal{U}(0.228, 0.252)$
Max service temperature	T_{max}	K	973	$\mathcal{U}(963, 983)$
Failure strain*	ϵ_f	%	0.07	$\ln\mathcal{N}(-2.6694, 0.1421^2)$

table D.3 for the titanium honeycomb (TI-HC) material, and table D.4 for the air gap. Lastly, D.5 gives the distribution for the assumed yield stress of the composite panel material used for the baffles.

D.3 Design Variable Bounds

Table D.6 records the nominal values and bounds of all necessary geometric parameters.

Table D.2: Laminate material properties for the load layer's graphite/bismaleimide composite layers.

Property	Symbol	Units	Value	Distribution
Density	ρ	$\frac{\text{kg}}{\text{m}^3}$	1568	$\mathcal{U}(1563, 1573)$
Elastic moduli	$E_1 = E_2$	GPa	60	$\mathcal{U}(57, 63)$
In-plane shear modulus	G_{12}	GPa	23.31	$\mathcal{U}(22.6, 24.0)$
Poisson ratios	$\nu_{12} = \nu_{21}$		0.344	$\mathcal{U}(0.334, 0.354)$
Mutual inf. coef. (first kind)	$\mu_{1,12} = \mu_{2,12}$		0.0	$\mathcal{U}(-0.1, 0.1)$
Mutual inf. coef. (second kind)	$\mu_{12,1} = \mu_{12,2}$		0.0	$\mathcal{U}(-0.1, 0.1)$
Thermal conductivity	$k_1 = k_2$	$\frac{\text{W}}{\text{m-K}}$	3.377	$\mathcal{U}(3.208, 3.546)$
Thermal conductivity	k_3	$\frac{\text{W}}{\text{m-K}}$	3.414	$\mathcal{U}(3.243, 3.585)$
Thermal expansion coef	$\alpha_1 = \alpha_2$	$\text{K}^{-1} \times 10^{-6}$	1.200	$\mathcal{U}(1.16, 1.24)$
Thermal expansion coef	α_{12}	$\text{K}^{-1} \times 10^{-6}$	0.0	$\mathcal{U}(-0.04, 0.04)$
Max service temperature	T_{max}	K	505	$\mathcal{U}(500, 510)$
Failure strain (tension)*	$\epsilon_{f,1} = \epsilon_{f,2}$	%	0.75	$\mathcal{U}(0.675, 0.825)$
Failure strain (compression)*	$\epsilon_{f,1} = \epsilon_{f,2}$	%	-0.52	$\mathcal{U}(-0.572, -0.494)$
Failure strain (shear)*	γ_f	%	0.17	$\mathcal{U}(0.153, 0.187)$

Table D.3: Isotropic material properties for the load layer's titanium honeycomb core.

Property	Symbol	Units	Value	Distribution
Density	ρ	$\frac{\text{kg}}{\text{m}^3}$	179.57	$\mathcal{U}(177.77, 181.37)$
Elastic modulus	E	GPa	1.90	$\ln\mathcal{N}(0.6441, 0.0779^2)$
Poisson ratio	ν		0.178	$\mathcal{U}(0.160, 0.196)$
Thermal conductivity	k	$\frac{\text{W}}{\text{m-K}}$	0.708	$\mathcal{U}(0.680, 0.736)$
Thermal expansion coefficient	α	$\text{K}^{-1} \times 10^{-6}$	2.97	$\mathcal{U}(2.88, 3.06)$
Max service temperature	T_{max}	K	755	$\mathcal{U}(745, 765)$
Yield stress	σ_Y	MPa	12.9	$\ln\mathcal{N}(2.5500, 0.1205^2)$

Table D.4: Properties of air in air gap between thermal and load layers.

Property	Symbol	Units	Value	Distribution
Density	ρ	$\frac{\text{kg}}{\text{m}^3}$	0	N/A
Thermal conductivity	k	$\frac{\text{W}}{\text{m-K}}$	0.0425	$\mathcal{U}(0.0320, 0.0530)$

Table D.5: Properties of panel structure used in baffles.

Property	Symbol	Units	Value	Distribution
Yield strength	σ_Y	MPa	324	$\ln\mathcal{N}(5.7736, 0.1196^2)$

Table D.6: Bounds for nozzle design variables

Parameter	**Units**	**Value**	**Lower Bound**	**Upper Bound**
Wall B-spline coefs	m	varies	min(-10%,0)	+10%
All layer thickness locations (non-dim)		0,0.5,1	0	1
Thermal layer thickness	m	0.03	0.01	0.05
Air gap thickness	m	0.005	0.003	0.01
Inside load layer thickness	m	0.002	0.0005	0.006
Middle load layer thickness	m	0.013	0.0064	0.0159
Outside load layer thickness	m	0.002	0.0005	0.006
Number of stringers		4	2	12+
Stringer thickness locations (non-dim)		0,0.5,1	0	1
Stringer thickness	m	0.01	0.002	0.01
Stringer height	m	N/A	N/A	N/A
Number of baffles		6	2	6+
Baffle location (non-dim)		0,0.2,0.4,0.6,0.8	0	1
Baffle thickness	m	0.01	0.0074	0.0359
Baffle height	m	N/A	N/A	N/A

Appendix E

Sobol' Indices

Dimension reduction in a problem can be motivated by the outcome of a global sensitivity analysis, for example by fixing input variables that do not contribute to variation in a function's output. Sobol' sensitivity indices provide a method for variance-based global sensitivity analysis of a function and have a close relationship with Analysis of Variance (ANOVA) techniques [12]. In a variance-based sensitivity analysis the effect of a function's input variables $x = [x_1 \dots x_n]^\top$ on the variation of the function's output $f(x)$ is analyzed. The effects of input variables x_i are characterized by several types. Main (or first order) effects characterize the impact of varying x_i by itself on variation in $f(x)$. Higher order effects characterize the impact of varying groups of x_i on variation in $f(x)$. Total order effects characterize the total impact of varying x_i either by itself or in combination with other x_j on variation in $f(x)$. As a result, total effects for each variable x_i are the quantity of interest when motivating dimension reduction. When a variable's total effect is zero, it effectively does not contribute to variation of $f(x)$ and it can be fixed at a nominal value without loss in accuracy.

Sobol' indices can be motivated by considering the decomposition of the function $f(x)$ for x defined on the unit cube in $\mathbb{R}^n$, *i.e.* $x_i \in [0, 1]$ for all elements x_i of x. $f(x)$ can be decomposed as [12]:

$$f(x) = f(x_1, \dots, x_n) = f_0 + \sum_{i=1}^{n} f_i(x_i) + \sum_{1 \le i \le j}^{n} f_{ij}(x_i, x_j) + \dots + f_{12\dots n}(x_1, \dots, x_n). \tag{4.37}$$

The decomposition in equation 4.37 is the same as the well-studied ANOVA decomposition from design of experiments literature and is also referred to as a Hoeffding decomposition [114]. It is valid if f_0 is constant and the integral of each term over its variables is zero [12]:

$$\int_0^1 f_{i_1,\dots i_s}(x_{i_1}, \dots, x_{i_s}) \mathrm{d}x_k = 0 \text{ for } k = i_1, \dots, i_s. \tag{E.1}$$

Now, let us consider the total variance D of the function $f(x)$, defined as

$$D = \int f(x)^2 \mathrm{d}x - f_0^2. \tag{E.2}$$

The relationship between the total variance D and the ANOVA decomposition from equation 4.37 is made clear by squaring both sides of equation 4.37 and integrating[1] over the unit cube [123]. The total variance D has the same decomposition

$$D = \sum_{i=1}^{n} D_i + \sum_{1 \leq i \leq j}^{n} D_{ij} + \ldots + D_{12\ldots n}, \tag{E.3}$$

where $D_{i_1 \ldots i_s}$ is given by

$$D_{i_1 \ldots i_s} = \int f_{i_1,\ldots,i_s}^2 \mathrm{d}x_{i_1} \ldots \mathrm{d}x_{i_s}. \tag{E.4}$$

The Sobol indices $S_{i_1 \ldots i_s}$ where s is called the order of the index are defined as

$$S_{i_1 \ldots i_s} \equiv \frac{D_{i_1 \ldots i_s}}{D}. \tag{E.5}$$

From the definition in equation E.5 it is clear that the Sobol' indices represent the fraction of the total variance contributed by the term $D_{i_1 \ldots i_s}$. Thus variables or combinations of variables with larger indices $S_{i_1 \ldots i_s}$ are more important. Examination of the Sobol' indices shows that 1) they are nonnegative since they are the ratio of two variances and 2) they sum to 1.

Several methods for the calculation of Sobol' indices can be used. Clearly, simple Monte Carlo can be used to estimate the integrals defining $D_{i_1 \ldots i_s}$. Archer et al proposed a more efficient Monte Carlo-based method for estimation of first order, higher order, and total order Sobol' indices using a resampling approach [12]. They also discuss a bootstrap approach for estimating confidence intervals on the obtained indices. Sobol' also discusses similar Monte Carlo-based methods for estimation of Sobol' indices [123]. Saltelli et al provide a general review of best practices for estimating first and total order indices using sampling methods [114]. Sudret introduced a method for calculation of Sobol' indices as an analytic postprocessing step once a polynomial chaos expansion (see section 2.4.1) of the function $f(x)$ has been constructed [125]. The method, whose cost is equivalent to that of constructing the polynomial chaos expansion, has been shown to be cheaper than Monte Carlo sampling methods.

[1] Doing so requires the assumption that $f(x)$ is square integrable.

Bibliography

[1] E. Acar, *Reliability-based design of representative wing and tail system together with structural tests*, Journal of Aircraft, 48 (2011), pp. 2130–2144.

[2] E. Acar, A. Kale, R. T. Haftka, and W. J. Stroud, *Structural safety measures for airplanes*, Journal of Aircraft, 43 (2006), pp. 30–38.

[3] H. Agarwal, *Reliability Based Design Optimization: Formulations and Methodologies*, PhD thesis, University of Notre Dame, Notre Dame, Indiana, 2004.

[4] H. Agarwal and J. E. Renaud, *New decoupled framework for reliability-based design optimization*, AIAA Journal, 44 (2006), pp. 1524–1531.

[5] J. Ahn and J. H. Kwon, *An efficient strategy for reliability-based multidisciplinary design optimization using BLISS*, Structural and Multidisciplinary Optimization, 31 (2006), pp. 363–372.

[6] N. M. Alexandrov, J. E. Dennis, R. M. Lewis, and V. Torczon, *A trust-region framework for managing the use of approximation models in optimization*, Structural Optimization, 15 (1998), pp. 16–23.

[7] N. M. Alexandrov, R. M. Lewis, C. R. Gumbert, L. L. Green, and P. A. Newman, *Approximation and model management in aerodynamic optimization with variable-fidelity models*, Journal of Aircraft, 38 (2001).

[8] M. Allen and K. Maute, *Reliability-based design optimization of aeroelastic structures*, Structural and Multidisciplinary Optimization, 27 (2004), pp. 228–242.

[9] K. Alvin, H. M. De La Fuente, B. Haugen, and C. A. Felippa, *Membrane triangles with corner drilling freedoms — I. the EFF element*, Finite Elements in Analysis and Design, 12 (1992), pp. 163–187.

[10] G. L. Ang, A. H. S. Ang, and W. H. Tang, *Optimal importance-sampling density estimator*, Journal of Engineering Mechanics, 118 (1992), pp. 1146–1163.

[11] Y. Aoues and A. Chateauneuf, *Benchmark study of numerical methods for reliability-based design optimization*, Structural and Multidisciplinary Optimization, 41 (2010), pp. 277–294.

[12] G. E. B. Archer, A. Saltelli, and I. M. Sobol, *Sensitivity measures, ANOVA-like techniques and the use of bootstrap*, Journal of Statistical Computation and Simulation, 58 (1997), pp. 99–120.

[13] S.-K. Au and J. L. Beck, *Estimation of small failure probabilities in high dimensions by subset simulation*, Probabilistic Engineering Mechanics, 16 (2001), pp. 263–277.

[14] M. A. Ba-abbad, E. Nikolaidis, and R. K. Kapania, *New approach for system reliability-based design optimization*, AIAA Journal, 44 (2006), pp. 1087–1096.

[15] J. Barthelemy and R. T. Haftka, *Approximation concepts for optimum structural design–a review*, Structural Optimization, 5 (1993), pp. 129–144.

[16] J. Bect, D. Ginsbourger, L. Li, V. Picheny, and E. Vazquez, *Sequential design of computer experiments for the estimation of a probability of failure*, Statistics and Computing, 22 (2012), pp. 773–793.

[17] B. Bichon and J. McFarland, *Inverse reliability analysis with EGRA*, in Proceedings of the 52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Denver, Colorado, 2011.

[18] B. J. Bichon, *Efficient Surrogate Modeling for Reliability Analysis and Design*, PhD thesis, Vanderbilt University, Nashville, TN, 2010.

[19] B. J. Bichon, M. S. Eldred, L. P. Swiler, S. Mahadevan, and J. M. McFarland, *Efficient global reliability analysis for nonlinear implicit performance functions*, AIAA Journal, 46 (2008), pp. 2459–2468.

[20] P. Bjerager, *Probability integration by directional simulation*, Journal of Engineering Mechanics, 114 (1988), pp. 1285–1302.

[21] A. Borri and E. Speranzini, *Structural reliability analysis using a standard deterministic finite element code*, Structural Safety, 19 (1997), pp. 361–382.

[22] D. Bryson and M. P. Rumpfkeil, *Comparison of unifed and sequential-approximate approaches to multifdelity optimization*, in Proceedings of the 58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Grapevine, TX, 2017.

[23] C. Bucher and T. Most, *A comparison of approximate response functions in structural reliability analysis*, Probabilistic Engineering Mechanics, 23 (2008), pp. 154–163.

[24] C. G. Bucher and U. Bourgund, *A fast and efficient response surface approach for structural reliability problems*, Structural Safety, 7 (1990), pp. 57–66.

[25] B. Cantwell, *AA 210 Course Notes*, Stanford, CA, 2014.

[26] C. Chevalier, J. Bect, D. Ginsbourger, E. Vazquez, V. Picheny, and Y. Richet, *Fast parallel kriging-based stepwise uncertainty reduction with application to the identification of an excursion set*, Technometrics, 56 (2014), pp. 455–465.

[27] C. Chevalier, D. Ginsbourger, and X. Emery, *Corrected kriging update formulae for batch-sequential data assimilation*, in Proceedings of the Mathematics of Planet Earth, Berlin, Heidelberg, 2014.

[28] C. Chevalier, V. Picheny, and D. Ginsbourger, *KrigInv: an efficient and user-friendly implementation of batch-sequential inversion strategies based on kriging*, Computational Statistics & Data Analysis, 71 (2014), pp. 1021–1034.

[29] J.-p. Chiles and P. Delfiner, *Geostatistics: Modeling Spatial Uncertainty*, Wiley, Hoboken, NJ, 2012.

[30] CMH-17, *Statistical methods*, in Composite Materials Handbook, CMH-17, 2012.

[31] A. R. Conn, N. I. M. Gould, and P. L. Toint, *Trust-Region Methods*, SIAM, Philadelphia, PA, 2000.

[32] P. G. Constantine, *Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies*, Society for Industrial and Applied Mathematics, Philadelphia, 2015.

[33] N. A. C. Cressie, *Statistics for Spatial Data*, Wiley, New York, 1993.

[34] R. De Meersman, *Least squares with non-linear equality constraints, application to closing of balances*, Journal of Computational and Applied Mathematics, 5 (1979), pp. 277–281.

[35] I. Deák, *Three digit accurate multiple normal probabilities*, Numerische Mathematik, 35 (1980), pp. 369–380.

[36] Z. Del Rosario, R. Fenrich, and G. Iaccarino, *Cutting the double loop: Theory and algorithms for reliability-based design optimization with statistical uncertainty*, International Journal for Numerical Methods in Engineering, (2019).

[37] O. Ditlevsen, P. Bjerager, R. Olesen, and A. M. Hasofer, *Directional simulation in gaussian processes*, Probabilistic Engineering Mechanics, 3 (1988), pp. 207–217.

[38] O. Ditlevsen and H. O. Madsen, *Structural Reliability Methods*, Wiley, Chichester, New York, 2005.

[39] O. Ditlevsen, R. E. Melchers, and H. Gluver, *General multi-dimensional probability integration by directional simulation*, Computers & Structures, 36 (1990), pp. 355–368.

[40] X. Du and W. Chen, *Sequential optimization and reliability assessment method for efficient probabilistic design*, Journal of Mechanical Design, 126 (2004), p. 225.

[41] X. Du, A. Sudjianto, and W. Chen, *An integrated framework for optimization under uncertainty using inverse reliability strategy*, Journal of Mechanical Design, 126 (2004), p. 562.

[42] V. Dubourg, B. Sudret, and J.-M. Bourinet, *Reliability-based design optimization using kriging surrogates and subset simulation*, Structural and Multidisciplinary Optimization, 44 (2011), pp. 673–690.

[43] V. Dubourg, B. Sudret, and F. Deheeger, *Metamodel-based importance sampling for structural reliability analysis*, Probabilistic Engineering Mechanics, 33 (2013), pp. 47–57.

[44] B. Echard, N. Gayton, and M. Lemaire, *AK-MCS: an active learning reliability method combining kriging and Monte Carlo simulation*, Structural Safety, 33 (2011), pp. 145–154.

[45] T. D. Economon, F. Palacios, S. R. Copeland, T. W. Lukaczyk, and J. J. Alonso, *SU2: an open-source suite for multiphysics simulation and design*, AIAA Journal, 54 (2016), pp. 828–846.

[46] M. Eldred, C. Webster, and P. Constantine, *Evaluation of non-intrusive approaches for wiener-askey generalized polynomial chaos*, in Proceedings of the 49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Schaumburg, IL, 2008.

[47] M. S. Eldred, H. Agarwal, V. M. Perez, S. F. Wojtkiewicz Jr, and J. E. Renaud, *Investigation of reliability method formulations in DAKOTA/UQ*, Structure and Infrastructure Engineering, 3 (2007), pp. 199–213.

[48] M. S. Eldred and D. M. Dunlavy, *Formulations for surrogate-based optimization with data fit, multifidelity, and reduced-order models*, in Proceedings of the 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Portsmouth, Virginia, 2006.

[49] M. S. Eldred, A. A. Giunta, S. S. Collis, N. A. Alexandrov, R. M. Lewis, and Others, *Second-order corrections for surrogate-based optimization with model hierarchies*, in Proceedings of the 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Albany, NY, 2004.

[50] C. Farhat, *AERO-S: a general-purpose finite element structural analyzer*, 2018.

[51] R. W. Fenrich and J. J. Alonso, *Reliable multidisciplinary design of a supersonic nozzle using multifidelity surrogates*, in Proceedings of the 18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Denver, Colorado, 2017.

[52] R. Fletcher, N. I. Gould, S. Leyffer, P. L. Toint, and A. Wächter, *Global convergence of a trust-region SQP-filter algorithm for general nonlinear programming*, SIAM Journal On Optimization, 13 (2002), pp. 635–659.

[53] R. Fletcher, S. Leyffer, and P. L. Toint, *A brief history of filter methods*, Preprint ANL/MCS-P1372-0906, Argonne National Laboratory, Mathematics and Computer Science Division, (2006), p. 36.

[54] E. Fox, *Issues in utilizing response surface methodologies for accurate probabilistic design*, in Proceedings of the AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials, Salt Lake City, UT, 1996.

[55] E. Fox and S. Reh, *On the accuracy of various probabilistic methods*, in Proceedings of the 41st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Atlanta, GA, 2000.

[56] A. Freudenthal, *Critical appraisal of safety criteria and their basic concepts*, IABSE: Congress 8th, (1968).

[57] C. Geuzaine and J. Remacle, *Gmsh: A 3-d finite element mesh generator with built-in pre- and post-processing facilities*, International Journal for Numerical Methods in Engineering, 79 (2009), pp. 1309–1331.

[58] R. G. Ghanem and P. D. Spanos, *Stochastic Finite Elements: A Spectral Approach*, Springer, New York, NY, 1991.

[59] P. E. Gill and W. Murray, *The computation of lagrange-multiplier estimates for constrained minimization*, Mathematical Programming, 17 (1979), pp. 32–60.

[60] P. E. Gill, W. Murray, and M. A. Saunders, *SNOPT: an SQP algorithm for large-scale constrained optimization*, SIAM Review, 47 (2005), pp. 99–131.

[61] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*, Academic Press Inc., London, 1981.

[62] T. Gneiting, W. Kleiber, and M. Schlather, *Matérn cross-covariance functions for multivariate random fields*, Journal of the American Statistical Association, 105 (2010), pp. 1167–1177.

[63] H. M. Gomes and A. M. Awruch, *Comparison of response surface and neural network with other methods for structural reliability analysis*, Structural Safety, 26 (2004), pp. 49–67.

[64] N. I. M. Gould, C. Sainvitu, and P. L. Toint, *A filter-trust-region method for unconstrained optimization*, SIAM Journal On Optimization, 16 (2005), pp. 341–357.

[65] R. V. Grandhi and L. Wang, *Higher-order failure probability calculation using nonlinear approximations*, Computer Methods in Applied Mechanics and Engineering, 168 (1999), pp. 185–206.

[66] F. Grooteman, *An adaptive directional importance sampling method for structural reliability*, Probabilistic Engineering Mechanics, 26 (2011), pp. 134–141.

[67] P. Hao, B. Wang, G. Li, Z. Meng, and L. Wang, *Hybrid framework for reliability-based design optimization of imperfect stiffened shells*, AIAA Journal, 53 (2015), pp. 2878–2889.

[68] S. Haykin, *Neural Networks and Learning Machines*, Pearson, McMaster University, Canada, 2009.

[69] F. Hemez, *The 3-node composite shell and isoparametric timoshenko beam elements*, Tech. Rep. CU-CAS-94-16, Center for Aerospace Structures, University of Colorado, Boulder, CO, 1994.

[70] E. Holmberg, B. Torstenfelt, and A. Klarbring, *Stress constrained topology optimization*, Structural and Multidisciplinary Optimization, 48 (2013), pp. 33–47.

[71] Z. Hu and S. Mahadevan, *Global sensitivity analysis-enhanced surrogate (GSAS) modeling for reliability analysis*, Structural and Multidisciplinary Optimization, 53 (2016), pp. 501–521.

[72] Z. Hu, S. Nannapaneni, and S. Mahadevan, *Efficient kriging surrogate modeling approach for system reliability analysis*, Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 31 (2017), pp. 143–160.

[73] C. Iorga, A. Desrochers, and C. Smeesters, *Engineering design from a safety perspective*, in Proceedings of the Canadian Engineering Education Association (CEEA12) Conference, Winnipeg, MB, 2012.

[74] A. Jameson, W. Schmidt, and E. Turkel, *Numerical solution of the Euler equations by finite volume methods using Runge Kutta time stepping schemes*, in Proceedings of the AIAA 14th Fluid and Plasma Dynamics Conference, Palo Alto, California, 1981.

[75] D. R. Jones, M. Schonlau, and W. J. Welch, *Efficient global optimization of expensive black-box functions*, Journal of Global Optimization, 13 (1998), pp. 455–492.

[76] A. D. Kiureghian and O. Ditlevsen, *Aleatory or epistemic? Does it matter?*, Structural Safety, 31 (2009), pp. 105–112.

[77] N. Kuschel and R. Rackwitz, *Two basic problems in reliability-based structural optimization*, Mathematical Methods of Operations Research, 46 (1997), pp. 309–333.

[78] S. L. Landau, *The New International Webster's Family Dictionary of the English Language*, Trident Press International, 1999.

[79] R. Lebrun and A. Dutfoy, *An innovating analysis of the Nataf transformation from the copula viewpoint*, Probabilistic Engineering Mechanics, 24 (2009), pp. 312–320.

[80] H. Li, *An Inverse Reliability Method and Its Applications in Engineering Design*, PhD thesis, The University of British Columbia, Vancouver, Canada, 1999.

[81] J. Li, J. Li, and D. Xiu, *An efficient surrogate-based method for computing rare failure probability*, Journal of Computational Physics, 230 (2011), pp. 8683–8697.

[82] J. Li and D. Xiu, *Evaluation of failure probability via surrogate models*, Journal of Computational Physics, 229 (2010), pp. 8966–8980.

[83] F. S. Lobato, M. S. Gonçalves, B. Jahn, A. A. Cavalini, and V. Steffen, *Reliability-based optimization using differential evolution and inverse reliability analysis for engineering system design*, Journal of Optimization Theory and Applications, 174 (2017), pp. 894–926.

[84] J. L. Loeppky, J. Sacks, and W. J. Welch, *Choosing the sample size of a computer experiment: A practical guide*, Technometrics, 51 (2009), pp. 366–376.

[85] A. Loseille and V. Menier, *Serial and parallel mesh modification through a unique cavity-based primitive*, in Proceedings of the 22nd International Meshing Roundtable, J. Sarrate and M. Staten, eds., Springer International Publishing, Cham, 2014, pp. 541–558.

[86] C. Militello and C. A. Felippa, *The first ANDES elements: 9-dof plate bending triangles*, Computer Methods in Applied Mechanics and Engineering, 93 (1991), pp. 217–246.

[87] S. Missoum, C. Dribusch, and P. Beran, *Reliability-based design optimization of nonlinear aeroelasticity problems*, Journal of Aircraft, 47 (2010), pp. 992–998.

[88] M. R. Moarefzadeh and R. E. Melchers, *Directional importance sampling for ill-proportioned spaces*, Structural Safety, 21 (1999), pp. 1–22.

[89] J. C. Musto, *The safety factor: Case studies in engineering judgment*, International Journal of Mechanical Engineering Education, 38 (2010), pp. 286–296.

[90] R. H. Myers, D. C. Montgomery, and C. M. Anderson-cook, *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*, John Wiley & Sons, Incorporated, 2016.

[91] J. Nie and B. R. Ellingwood, *Directional methods for structural reliability analysis*, Structural Safety, 22 (2000), pp. 233–249.

[92] ——, *A new directional simulation method for system reliability. Part i: Application of deterministic point sets*, Probabilistic Engineering Mechanics, 19 (2004), pp. 425–436.

[93] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer, New York, 1999.

[94] B. Nour-omid and C. C. Rankin, *Finite rotation analysis and consistent linearization using projectors*, Computer Methods in Applied Mechanics and Engineering, 93 (1991), pp. 353–384.

[95] J. Oakley, *Estimating percentiles of uncertain computer code outputs*, Journal of the Royal Statistical Society: Series C (applied Statistics), 53 (2004), pp. 83–93.

[96] A. B. Owen, *Monte Carlo Theory, Methods and Examples*, Unpublished, Stanford, CA, 2013.

[97] D. Padmanabhan, *Reliability-based Optimization for Multidisciplinary System Design*, PhD thesis, University of Notre Dame, Notre Dame, IN, 2003.

[98] R. M. Paiva, C. Crawford, and A. Suleman, *Robust and reliability-based design optimization framework for wing design*, AIAA Journal, 52 (2014), pp. 711–724.

[99] I. Papaioannou, *Non-intrusive Finite Element Reliability Analysis Methods*, PhD thesis, Technische Universitt Mnchen, Munich, Germany, 2012.

[100] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, and V. Dubourg, *Scikit-learn: Machine learning in Python*, Journal of Machine Learning Research, 12 (2011), pp. 2825–2830.

[101] B. Peherstorfer, K. Willcox, and M. Gunzburger, *Survey of multifidelity methods in uncertainty propagation, inference, and optimization*, SIAM Review, (2018).

[102] H. Prautzsch, W. Boehm, and M. Paluszny, *Bezier and B-spline Techniques*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2002.

[103] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*, Cambridge University Press, New York, NY, 1992.

[104] X. Qu and R. Haftka, *Reliability-based design optimization using probabilistic sufficiency factor*, Structural and Multidisciplinary Optimization, 27 (2004).

[105] H. Rabitz, Ömer. F. Aliş, J. Shorter, and K. Shim, *Efficient input-output model representations*, Computer Physics Communications, 117 (1999), pp. 11–20.

[106] R. Rackwitz, *Reliability analysis — a review and some perspectives*, Structural Safety, 23 (2001), pp. 365–295.

[107] I. Raju, J. Stadler, J. Kramer-white, and R. Piascik, *White paper on factors of safety*, Tech. Rep. NASA/TM-2009-215723/REV1 NESC-PB-04-05, NASA, Hampton, VA, 2012.

[108] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, Cambridge, MA, 2006.

[109] J. F. Rodriguez, J. E. Renaud, B. A. Wujek, and R. V. Tappeta, *Trust region model management in multidisciplinary design optimization*, Journal of Computational and Applied Mathematics, 124 (2000), pp. 139–154.

[110] M. Rosenblatt, *Remarks on a multivariate transformation*, The Annals of Mathematical Statistics, 23 (1952), pp. 470–472.

[111] J. Royset, A. D. Kiureghian, and E. Polak, *Reliability-based optimal structural design by the decoupling approach*, Reliability Engineering and System Safety, 73 (2001), pp. 213–221.

[112] Y. Saad and M. H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM Journal On Scientific and Statistical Computing, 7 (1986), pp. 856–869.

[113] J. Saleh and K. Marais, *Highlights from the early (and pre-) history of reliability engineering*, Reliability Engineering & System Safety, 91 (2006), pp. 249–256.

[114] A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto, and S. Tarantola, *Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index*, Computer Physics Communications, 181 (2010), pp. 259–270.

[115] G. I. Schuëller and R. Stix, *A critical appraisal of methods to determine failure probabilities*, Structural Safety, 4 (1987), pp. 293–309.

[116] W. Shakespeare, *Sonnet XVIII.*

[117] S. Shan and G. G. Wang, *Reliable design space and complete single-loop reliability-based design optimization*, Reliability Engineering & System Safety, 93 (2008), pp. 1218–1230.

[118] F. R. Shanley, *Historical note on the 1.5 factor of safety for aircraft structures*, Journal of the Aerospace Sciences, 29 (1962), pp. 243–244.

[119] M. A. Shayanfar, M. A. Barkhordari, M. Barkhori, and M. Barkhori, *An adaptive directional importance sampling method for structural reliability analysis*, Structural Safety, 70 (2018), pp. 14–20.

[120] P. B. Shelley, *Ode to the west wind*.

[121] M. Shinozuka, *Basic analysis of structural safety*, Journal of Structural Engineering, 109 (1983), pp. 721–740.

[122] I. Sobol', *Theorems and examples on high dimensional model representation*, Reliability Engineering and System Safety, 79 (2003), pp. 187–193.

[123] I. M. Sobol', *Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates*, Mathematics and Computers in Simulation, 55 (2001), pp. 271–280.

[124] S. Song, Z. Lu, and H. Qiao, *Subset simulation for structural reliability sensitivity analysis*, Reliability Engineering & System Safety, 94 (2009), pp. 658–665.

[125] B. Sudret, *Global sensitivity analysis using polynomial chaos expansions*, Reliability Engineering & System Safety, 93 (2008), pp. 964–979.

[126] R. Sues, M. Aminpour, and Y. Shin, *Reliability-based multi-disciplinary optimization for aerospace systems*, in Proceedings of the 42nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Seattle, WA, 2001.

[127] A. Takano, *Inferential statistical method for structural design*, AIAA Journal, 55 (2017), pp. 1026–1030.

[128] J. Tu, K. K. Choi, and Y. H. Park, *A new study on reliability-based design optimization*, ASME Journal of Mechanical Design, 121 (1999), pp. 557–564.

[129] S. Uryas'ev, *Derivatives of probability functions and integrals over sets given by inequalities*, Journal of Computational and Applied Mathematics, 56 (1994), pp. 197–23.

[130] S. Uryasev, *Derivatives of probability functions and some applications*, Annals of Operations Research, 56 (1995), pp. 287–311.

[131] X. Wang, *On the approximation error in high dimensional model representation*, in Proceedings of the 40th Conference On Winter Simulation, Miami, FL, 2008.

[132] Y.-T. Wu, Y. Shin, R. Sues, and M. Cesare, *Safety-factor based approach for probability-based design optimization*, in Proceedings of the 42nd AIAA/ASME/ASC/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Seattle, Washington, 2001.

[133] D. Xiu and J. S. Hesthaven, *High-order collocation methods for differential equations with random inputs*, SIAM Journal On Scientific Computing, 27 (2005), pp. 1118–1139.

[134] D. Xiu and G. E. Karniadakis, *Modeling uncertainty in flow simulations via generalized polynomial chaos*, Journal of Computational Physics, 187 (2003), pp. 137–167.

[135] B. Youn, K. Choi, and L. Du, *Adaptive probability analysis using an enhanced hybrid mean value method*, Structural and Multidisciplinary Optimization, 29 (2005), pp. 134–148.

[136] B. D. Youn, K. K. Choi, and L. Du, *Enriched performance measure approach for reliability-based design optimization.*, AIAA Journal, 43 (2005), pp. 874–884.

[137] A. Zanette, J. Zhang, and M. J. Kochenderfer, *Robust super-level set estimation using gaussian processes*, in Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, Dublin, Ireland, 2018.

[138] Z. Zhao, Z. Wang, and X. Duan, *A comparison of several proposed reliability analysis methods combining kriging and Monte Carlo simulation*, in Proceedings of the 2016 IEEE Chinese Guidance, Navigation and Control Conference, Nanjing, China, 2016.

[139] J. J. Zipay, C. T. Modlin, and C. E. Larsen, *The ultimate factor of safety for aircraft and spacecraft — its history, applications and misconceptions*, in Proceedings of the 57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, San Diego, CA, 2016.

[140] T. Zou and S. Mahadevan, *A direct decoupling approach for efficient reliability-based design optimization*, Structural and Multidisciplinary Optimization, 31 (2006), pp. 190–200.