UNIVERSITY OF SOUTHAMPTON

On the characterization of measured geometry for the facilitation of uncertainty propagation and robust design

by

Jennifer A. Forrester

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the
Faculty of Engineering and Physical Sciences
School of Engineering

November 2019

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND PHYSICAL SCIENCES
SCHOOL OF ENGINEERING

Doctor of Philosophy

by Jennifer A. Forrester

Political, environmental, and economic factors are driving an increased need to understand variability in aerospace product performance. Although there is a plethora of geometric inspection data gathered over many years, and an array of methods for informing design decisions based on statistically defined uncertainty, uncertainty in performance due to geometric variability is seldom assessed beyond the modification of geometric design parameters in existing analyses. The practical use of these approaches is limited by the cost of simulation, the lack of accurate statistical definition of geometric variation, and the complexity involved in modifying geometric definitions for their implementation in existing analysis.

In this thesis, methods are developed to enable three sets of inspection measurement data from aero-engine components to further inform design. In the first case, turbine disc firtree slot flank geometry is characterized using existing parameters, fitted to point cloud measurement data using particle swarm optimization. In the second case, measurements of combustor features correspond directly to existing parameters. Implementation of changes to those parameters within the analysis mesh is achieved using a modified polar coordinate based mesh morphing algorithm with surrogate model based transformations. The final data sets are point cloud measurements along a blade firtree flank and have a form that differs from the design. Here, both defining the uncertainty and applying the geometric changes within the simulation require novel application of existing techniques. The complex shape of the variable geometry is modelled using a combined Kriging and principal component analysis based approach, and realized through radial basis function morphing of an existing mesh.

The techniques enable the facilitation of design optimization in an uncertainty framework. This is achieved by providing an automated approach to producing a data-based reduced set of uncertain geometric variables, and associated probability distributions, from which appropriate designs of experiment can be sampled. The computer experiments can be executed using the free form deformation based mesh morphing methods for integration of the uncertainty with existing complex workflows.

Contents

List of Figures

List of Tables

Declaration of Authorship

I, Jennifer Forrester, declare that this thesis and the work presented in it is my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. Either none of this work has been published before submission, or parts of this work have been published as:

    Forrester and Keane [81]

Signed: .....

Date: .....

Acknowledgements

For his guidance and support over many years, I extend sincere thanks to my supervisor, Prof. A. J. Keane, without whose unerring advocacy my PhD could not have been completed. Thank you as well to Dr. A. Sóbester for his critical eye, and to Dr S. Quinn for his endorsement.

I thank Rolls-Royce plc., including Dr. R. Bates, for their backing, and for provision of the data that has enabled me to ensure that my work is directly applicable to current engineering problems. Thank you for all the assistance received from members of the Rolls-Royce UTC at the University of Southampton who have helped whenever I have asked and, in particular, Dr I. Voutchkov and Dr. D. Toal who have been exceedingly patient with me throughout.

I will be forever grateful to my friends and family for their motivation and words of wisdom. And, for the crucial emotional support, encouragement, indispensable advice, proof reading, technical expertise, and provision of many delicious meals, thank you to my husband, Dr. A. I. J. Forrester.

Nomenclature

Acronyms

ALS	alternating least squares
ANOVA	analysis of variance
CAD	computer aided design
CAM	computer aided manufacturing
cdf	cumulative distribution function
CFD	computational fluid dynamics
CMM	coordinate measuring machine
CPU	central processing unit
CSM	computational structural mechanics
DoE	design of experiment
DSO	design search and optimization
FDN	fixed distance neighbour
FEA	finite element analysis
FFD	free-form deformation
GA	genetic algorithm
GUM	Guide to the expression of uncertainty in measurement
HCF	high cycle fatigue
HF	high fidelity
ICA	independent component analysis
II	inner intermediate
iid	independent, identically distributed
IP	inner primary
ISO	International Organization for Standardization
KL	Karhunen-Loève
KNN	K-nearest neighbour
KS	Kolmogorov-Smirnov
LCF	low cycle fatigue
LF	low fidelity
LHS	Latin hypercube sampling
LLE	local linear embedding

MC	Monte Carlo
MFO	multi-fidelity optimization
MLE	maximum likelihood estimate
MOO	multi-objective optimization
MPM	machining pattern model
MSE	mean squared error
NGV	nozzle guide vanes
NSGA	non-dominated sorting genetic algorithm
NURBS	non-uniform rational basis-splines
OA	orthogonal array
OI	outer intermediate
OLH	optimal Latin hypercube
OP	outer primary
OR	operations research
PC	principal component
PCA	principal component analysis
PCE	polynomial chaos expansion
pdf	probability density function
POD	proper orthogonal decomposition
PSO	particle swarm optimization
RANS	Reynolds-averaged Navier-Stokes
RBDO	reliability based design optimization
RBF	radial basis function
RDO	robust design optimization
RMSE	root mean squared error
RPD	robust parameter design
RSM	response surface models
RTDF	radial temperature distribution factor
SCFM	stochastic computational fluid mechanics
SFEM	stochastic finite element method
SNR	signal-to-noise ratio
SQ	sparse quadrature
SQP	sequential quadratic programming
SSE	sum of squares due to error
SVD	singular value decomposition
SWT	Smith-Watson-Topper
TDF	temperature distribution factor
UP	uncertainty propagation
UQ	uncertainty quantification
VEGA	vector evaluated genetic algorithm

General mathematics and statistics

α	stable distribution shape parameter
$\bar{x}$	sample mean
β	stable distribution shape parameter
$\boldsymbol{\mu}$	mean, or expectation, vector
χ^2	χ^2-test hypothesis test statistic
δ	stable distribution location parameter
Γ	gamma function
γ	stable distribution scale parameter
$\hat{P}$	Kernel density estimator
λ	eigenvalue
λ_N	normal skewness parameter
$\lambda_{(L)}$	Lagrangian multipliers
$\bar{\mathbf{x}}$	mean vector
$\mathbf{\Sigma}$	diagonal matrix of singular values
$\mathbf{A}$	symmetric matrix
$\mathbf{C}$	rank correlation matrix used for the Iman-Conover method
$\mathbf{D}$	diagonal matrix of eigenvalues
$\mathbf{E}$	matrix of eigenvectors
$\mathbf{e}$	eigenvector (loading vector)
$\mathbf{I}$	identity matrix
$\mathbf{J}$	left orthogonal matrix of singular vectors
$\mathbf{K}$	right orthogonal matrix of singular vectors
$\mathbf{L}$	lower triangular matrix for decomposition
$\mathbf{L}_{\mathrm{IC}}$	lower triangular matrix used for the Iman-Conover method
$\mathbf{P}$	point set matrix in the principal coordinate system (matrix of score vectors)
$\mathbf{p}$	point in principal component space (score vector)
$\mathbf{R}$	matrix of residuals
$\mathbf{U}$	upper triangular matrix for decomposition
$\mathbf{V}$	covariance matrix, often denoted $\mathbf{\Sigma}$
$\mathbf{X}$	point set matrix in the standard basis (Cartesian coordinate system)
$\mathbf{Y}$	vector of random variables
E	expectation
N	Gaussian (normal) distribution
P	probability
S	stable distribution
U	uniform distribution
$\mathscr{F}$	Fourier transform
μ	mean, or expectation
ϕ	characteristic function
ρ	correlation

ρ'	Spearman rank correlation
Σ	singular value
σ	standard deviation
σ^2	variance
a	positive constant
b	positive constant
c	real constant
d	number of variables (dimensions)
D^*	Kolmogorov-Smirnov two-sample test statistic
d_s	difference in statistical rank
E_i	expected counts for χ^2 hypothesis test
F	cumulative distribution function
f	generic function, or objective function
f_L	loss function for non-linear principal component analysis
h	Kernel function bandwidth
K	Bessel function
k	number of constants
K_s	Kernel smoothing function
L	statistical likelihood
N	number of eigenvectors
n	number of points
N_b	number of bins for χ^2 hypothesis test
n_r	number of random variables
n_{OA}	number of variables in an orthogonal array
O_i	observed counts for χ^2 hypothesis test
P	probability density function
r	linear correlation coefficient
t_{OA}	orthogonal array strength
X	random variable
Y	random variable
Z	random variable
z	z-test hypothesis test statistic

Response surfaces and optimization

$\boldsymbol{\theta}$	vector of Kriging constant parameters
$\boldsymbol{p}$	vector of Kriging constant parameters
$\boldsymbol{r}$	random vector
$\boldsymbol{V}$	vector of particle velocities
$\boldsymbol{w}$	vector of weights
δ	random change
δ_K	regressing Kriging validation metric

ϵ_{CV}	cross-validation error
ϵ_{RMS}	root mean squared error
λ	regularization parameter
$\boldsymbol{\Psi}$	matrix of basis functions: Gram matrix
$\mathbf{c}$	basis function centres in multiple dimensions
$\mathbf{p}$	vector of 'uncontrollable' parameters
$\mathbf{u}$	combined vector of $\mathbf{p}$ and $\mathbf{v}$
$\mathbf{v}$	vector of design variables which can contain uncertainty
$\mathbf{x}$	evaluation point in multiple dimensions, or design vector (optimal point)
$\mathbf{y}$	vector of response values
$\mathbf{z}$	vector of changes to $\mathbf{p}$ or $\mathbf{v}$
B	breaking strength
L	applied load
p	realization of an 'uncontrollable' parameter
P_f	probability of failure
v	realization of a design variable which can contain uncertainty
ν	Matérn kernel function constant parameter
Φ_{PSO}	constriction factor
ϕ_{PSO}	sum of particle swarm constants
$\boldsymbol{\psi}$	vector of correlations
ψ	basis function
τ^2	noise variance (nugget for regressing Kriging)
θ	Kriging, or kernel function, constant parameter
c	number of constraints
c_1	cognitive constant
c_2	social constant
d	number of dimensions
g	constraint function
K	cross-validation subsets
l	Matérn kernel function constant parameter
m	number of goals
N	Monte Carlo sample size
n	number of data, or sample, points
n_c	number of basis function centres
n_s	number of points left out in cross-validation
n_t	number of points in validation set
p	Kriging, or kernel function, constant parameter
r	Euclidean distance between $\mathbf{c}^{(i)}$ and $\mathbf{c}^{(i)}$
s^2	Kriging predictor error
V	particle velocity
w	weighting parameter

x	variable value in a single dimension
y	response value

Geometry modelling

(s, t, u)	parametric surface parameters
(x, y, z)	algebraic surface parameters
δ_m	sum of the weighted squared deviations for polynomial regression
ϵ_{CV}	cross-validation error
ϵ_{RMS}	root mean squared error
η	spline smoothing objective function
ϵ	vector of residuals
$\mathbf{a}$	Bézier curve control points
$\mathbf{d}$	vector of displacements
$\mathbf{u}$	vector of polynomial coefficients
$\mathbf{y}$	vector of response values
$\mathbf{Z}$	matrix of postulated polynomial terms
$\mathcal{B}$	Bézier curve
$\mathcal{N}$	non-uniform rational b-spline curve
$\mathcal{S}$	b-spline curve
ϕ	radial basis function
θ	Kriging parameter
$\tilde{\eta}$	alternative spline smoothing objective function
φ	Powell's spline smoothing objective function
b	b-spline basis functions
D	third derivative in spline
d	orthogonal residual
G	rational Bézier spline function
G_x, G_y	orthogonal spline functions
h	mean data point spacing
K	cross-validation subsets
k	degree of spline
l, m,and n	Bernstein polynomial degree in three coordinate directions
m	number of knots in spline
n	number of points
n_b	number of basis functions
n_k	number of knots minus one
n_s	number of points left out in cross-validation
n_t	number of points in validation set
O	spline order
P	spline control points
p	linear polynomial function

R	sum of the squares of the orthogonal residuals
r	radial distance from displaced boundary nodes
S	sum of the squares of the residuals (may be weighted)
s	point positions or displacements
S_{max}	spline smoothing factor
T	spline knots
t	distance along curve
u	polynomial coefficient
v	spline constant
W	weight function
w	control point weight
K_s	number of spline segments
m_p	polynomial degree
s_S	spline smoothing parameter
$\boldsymbol{b}$	vector of b-spline basis functions
$\boldsymbol{P}$	vector of spline control points
$\boldsymbol{T}$	spline knot vector
$\boldsymbol{v}$	vector of spline constants
$\boldsymbol{W}$	vector of weights

Firtree flank angle fitting

(x, y)	Cartesian coordinates
(x_1, y_1)	First tangent arc centre
(x_2, y_2)	Second tangent arc centre
α	Firtree flank angle
l	Firtree flank length
r_1	First tangent arc radius
r_2	Second tangent arc radius

Morphing of complex holes

(r, θ, z)	polar coordinates
(x, y, z)	Cartesian coordinates
δr	prescribed change in r
δx	'correction' in x
δy	'correction' in y
$\mathbf{p}$	control points
$\mathbf{q}$	control points
d	hole diameter
m	counter for number of control points
n	counter for number of control points
T	temperature

T_{av}	area averaged temperature
T_{in}	inlet temperature
Z	Kriging predictor for z-value model
h	heights defining combustor port cross-sectional geometry

Firtree flank contact surface fitting

(x, y, z)	Cartesian coordinates
α	polynomial regression coefficients
$\boldsymbol{\alpha}$	vector of polynomial regression coefficients
$\boldsymbol{\nu}$	matrix of polynomial terms
ν	scaling factor for regressing Kriging as defined by Picheny et al. [207]
τ^2	nugget for regressing Kriging as defined by Picheny et al. [207]
d	distance from reference flank position

Dovetail joint lifing analysis

ν	Poisson's ratio
ω	angular velocity
ϕ	hoop distance around disc
ρ	density
σ_a	stress amplitude
σ_{ar}	equivalent fully reversed stress amplitude
$\sigma_f^{'}$	fatigue strength coefficient
σ_{max}	peak stress
$\varepsilon_f^{'}$	fatigue ductility coefficient
b	fatigue strength exponent
c	fatigue ductility exponent
E	Young's modulus
N_f	number of cycles to failure
r	radial distance
U	displacement

Chapter 1

Introduction

1.1 An historical perspective

Engineers often build predictive models to estimate the likely performance of a component or system design. Such models vary in fidelity and complexity, from simple 'back-of-the-envelope' type calculations, using well established formulae, to detailed computer codes. All of these models, however, make certain assumptions and approximations about the real-life scenario that they model. It is the methods employed to enable the designer to take account of the effects of these assumptions that are the focus of this thesis. In particular, we are concerned with critical component design in aero-engines. Assumptions are typically introduced to models as:

- boundary conditions,
- geometry, and
- solid or fluid material properties.

The true variability, or *uncertainty*, in each of these input assumptions results in a combined uncertainty in the performance prediction. Common sources of variability are manufacturing processes and operating environment, as well as uncertainty through lack of understanding, or *model error*.

As computational processing power increased throughout the 1980s and beyond, efforts into developing more and more accurate and computationally expensive codes for performance prediction grew. They continue to this day. These physics-based codes have been refined such that, in many cases, the confidence in their predictions is considerable. There is now often very little uncertainty in their underlying calculations. These analysis codes remain, in general, the basis of most detailed engineering design. They tend to be based on computational structural mechanics (CSM), or computational fluid dynamics

(CFD), codes. These codes (usually using finite element analysis (FEA), or finite volume methods respectively) are now notably accurate for static, linear, and even many non-linear structural, or inviscid flow problems but do still contain considerable uncertainty and continue to be refined for complex models involving, for example: contact, fatigue, dynamic impact, viscous flow, and prediction of flow separation.

Traditionally, uncertainty has not been entirely ignored; designers have applied *safety factors* or *reserve factors* to introduce a *margin of safety* to their calculations in an attempt to ensure that designs do not fail to meet requirements. The choice of a suitable value for the required safety factor is generally based on experience and perceived risk. This is well established practice and many national and international design standards continue to use safety factors within their recommended calculations.

As Choi et al. [41] state, "as modern structures require more critical and complex designs, the need for accurate approaches to assess uncertainties ... has increased significantly". There are applications where factors of safety are still the most rational approach given that the consequences of poor performance or failure are of little significance and there is no need to expend additional effort to quantify the uncertainty in the prediction made by the mathematical model, or in cases where very large safety factors can be readily accepted. However, in applications where the risks are particularly high, for example in the nuclear industry, the safety factor based approach is no-longer acceptable. Indeed, designers of nuclear power plants were among the first to begin to quantify uncertainty in their models and assumptions. *Risk*, or *probability of failure*, in such applications is a probability of risk to human life, among other serious effects.

If we consider aerospace design specifically, engineers and manufacturers must strive to remain competitive in a global market. It is not sufficient to build an aircraft or engine that only 'does the job' and is considered 'safe'. These are assumed prerequisites. A global customer-driven, or *value-driven*, market requires that all products are cheaper to produce, more economic (cheaper to run), more reliable (cheaper to maintain), and have a longer usable life than those of their competitors. Companies such as Rolls-Royce plc. have entered into 'total care' packages with their customers to underwrite the quality of their engines. As a result, they are responsible for bearing the cost of maintenance throughout a pre-determined engine life. In addition, the demand for affordable air travel has increased steadily, reaching over 275 million passengers in 2017, despite a small fall in the years 2008 to 2010 [42][1]). Governments have also been pressurized into introducing tight requirements on noise and emissions, as well as efficiency, following the sharp rise in air travel and an increased public concern over environmental impact and fossil fuel

[1] On 15th March 2010 Harry Bush, CAA Director of Economic Regulation, said, "today's figures show the biggest fall in passenger numbers since the second world war, highlighting the enormous impact the recession has had on the aviation industry. Passenger numbers are now back to the level they were six years ago and, although they will certainly rebound, the pace of recovery is uncertain and it could be a number of years before they reach their peak level again". He was proven correct as numbers continued to rise again after 2010 at a similar rate to the previous decade.

depletion. All of these design drivers combine to result in multiple constraints on designs and, due to these, unless step changes in the design approach can be found, increases in design complexity to meet them. Increased complexity inevitably leads to more need to deal with the introduction of uncertainty.

In this thesis, we consider the design of aero-engines and their components where complexity and uncertainty are of particular concern. The turbofan engine, figure 1.1, is already one of the most complex pieces of machinery ever produced. Although its pioneer Frank Whittle said that simplicity was a hallmark of his jet engine, "not once in the history of the jet engine has it been truly simple, not in theory, not in practice, not in manufacture, not in application" [235]. The ever tightening requirements on noise, emissions, and efficiency have, in general, been addressed over recent years using a variety of approaches aimed primarily at improving the performance of the *nominal* design. Consider a component: its current nominal design is reached through initial *innovation* in the early *preliminary design* stage, followed by successive iteration in the *detailed design* phase until constraints on, for example, stress, deflection, and weight, are met with appropriate safety factors. The computer code that uses FEA to predict these performance parameters involves time consuming solutions of partial differential field equations, which may, depending on the fidelity, size, and linearity of the problem consume days of computing effort. To ensure that undesired effects are not introduced due to manufacturing error, tolerance bounds are specified for the component geometry and as many runs of the analysis code as can be afforded are executed at the extremities of the tolerance bands, which are adjusted if required. Often the choice of tolerances takes into account what is achievable with the current manufacturing process. A number of components are made to ensure that they conform with the designers' specification and if loading tests are possible they can be tested. The component is then tested as part of the whole engine, after which its design is not altered unless it exhibits signs of failure. There may be a number of engine tests and the manufacturing process must be productionized[2] and the component proved in tests in the production engine as well. The process is greatly simplified in this description but we now have a component nominal design with tolerance specifications, typically designed for worst case load cycles and with nominal material properties.

To improve upon the current design for a new, more stringent, set of requirements one, unsurprisingly, begins the iterative cycle again, perhaps searching for stronger materials, tighter tolerances, or using *model verification* to improve the mathematical model being used, based on previous tests. This process of innovation followed by *evolution* is described by Keane and Nair [130] as a precursor to *automation* - a prerequisite for *optimization*.

[2]Oxford spelling as used by the Oxford University Press, and defined in the Oxford English Dictionary [259] is used throughout this thesis.

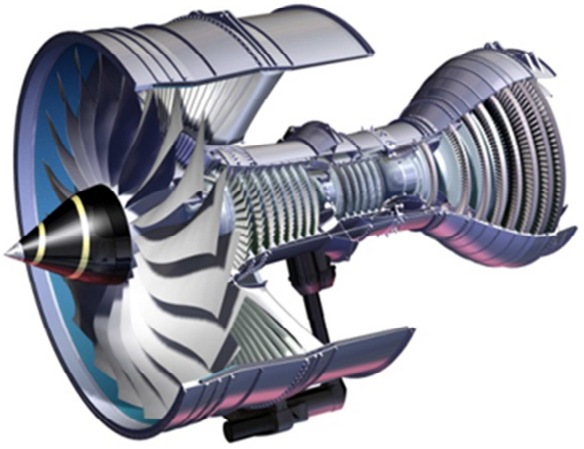


Figure 1.1: Rolls-Royce Trent series turbofan engine, courtesy of Rolls-Royce plc.

In the 1990s, aerospace companies, and many others, began to turn to formal optimization, or more accurately termed *design search and optimization* (DSO), as a tool for searching for superior product designs. The approaches enabled them to squeeze additional performance from current design concepts without the expense of new innovation. Many developments around this time in *response surface modelling* for optimization enabled these improvements to be made at an affordable computational cost. Keane and Nair [130] stated in 2005 that "a formidable range of optimization methods is available to designers - so much so that most current design-related optimization research focuses on how such methods should be deployed, rather than on developing new search methods per se". Design of consistently similar components could be automated using parametric computer aided design (CAD) packages, and DSO algorithms. At the time, however, the expertise lay with software specialists and researchers rather than with engineers in industry. Over a decade later, a number of commercial DSO software packages are now available that are integrated with, or with close links to, CAD software.

The analyses of product performance discussed thus far, with or without automation and DSO, are termed *deterministic*; the inputs are fixed at single, usually nominal, values and the predicted performance is a prediction at a single design point, figure 1.2. There may be multiple input and output values, such as those with changes in time or with field outputs, but these are still only analysed at a single design point.

Recently, for critical aerospace components, deterministic optimization has seen a rise in popularity, perhaps due in part to the availability of user friendly software as well as the drive from economic factors. In the longer term, new innovation is required to meet regulatory constraints, but in the coming decades further refinement of existing

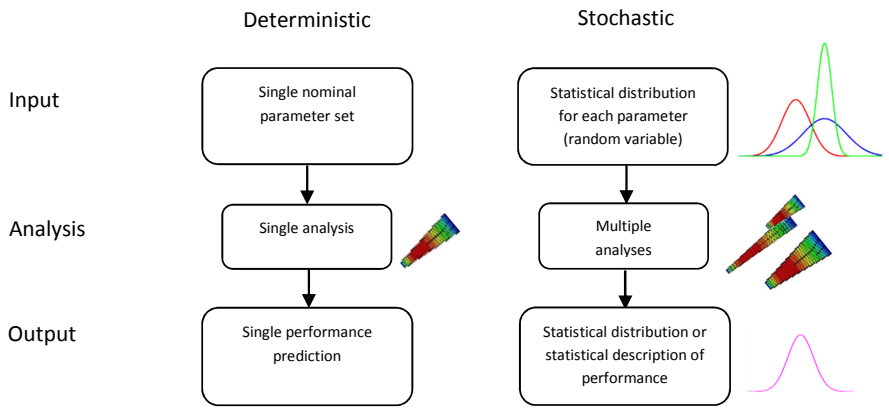


Figure 1.2: Comparison of deterministic and stochastic analysis

technology must continue. The use of deterministic optimization has been seen to result in large performance gains, but often the theoretically 'best' design lies close to constraint, or operational, boundaries. When tested, these nominally high performing components can still fail to 'meet the mark' (reach the desired, or designed, level of performance). If tolerances have been tightened, and models and manufacturing processes have been refined, the under-, or over-, performance of the design is now due to uncertainties (known and *irreducible*, or unknown). This raises two important questions for the designer:

1. given these uncertainties, how close should the design really be to the boundaries, and
2. is a high performance design that may be particularly sensitive to irreducible uncertainties really the 'best'?

Assuming that what is uncertain is known, in response to these questions the designer is now turning to *stochastic* analysis, figure 1.2, which has received much attention in the engineering literature over recent years. The input variables are considered to be random, or stochastic, in nature and *uncertainty propagation* (UP)[3] techniques are employed, often through multiple analysis solutions, to predict the statistical distribution of the performance. Choi et al. [41] are clear that a deterministic approach can only result in an over-designed or under-designed system. As the title of Choi's book suggests, the problem then becomes one of *reliability*. Reliability is concerned with variation close to constraints. Where in the past one would have calculated a factor of safety or reserve factor, one should investigate the probability of failure, or risk. The second question is

[3]UP as the calculation of uncertainty at design points of interest is sometimes referred to as *uncertainty quantification* (UQ), but will only be termed UP in this thesis to distinguish it from the quantification of the input distributions of uncertain variables.

one of *robustness*, where a key output of interest is the variation in performance, often described by the *variance* or *standard deviation*. This is a *probabilistic* description of uncertainty propagation in that it is concerned with the determination of *probability density functions* (pdfs) or their descriptors, *statistical moments*. Reliability is primarily concerned with the extremities, or 'tails' of the pdf, where robustness is concerned with the central region close to the mean or nominal. There are alternative non-probabilistic approaches that are not the focus of this thesis.

Perhaps the increase in the use of DSO has also been in part due to the increase in computer processing power. Search of the design space requires multiple runs of analysis code, and even *response surface models* (RSMs) require reasonably large number of runs, the number increasing exponentially with the number of *design variables*; the so-called 'curse of dimensionality' [78]. Uncertainty propagation methods face this same difficulty for searching the uncertain space; where deterministically only a single evaluation of a computational model was required for each assessment, now multiple runs are needed for accurate response prediction. The first step to creating a practicable approach is to reduce the number of variables by determining which are expected to be of most consequence to the performance; *sensitivity analysis* or *variable reduction*. Removing variables in error can result in overly optimistic prediction of performance variation. For this reason, when using these techniques for reliability prediction, one must be particularly cautious. However, when using uncertainty propagation for design improvement these approaches can provide useful guidance for the 'best' employment of limited computational effort. In some cases, a designer may have experience that guides them to investigate a particular set of variables for which they wish to determine the resulting performance variation.

A noteworthy example is that of geometric parameters, where tolerances are specified for measurable dimensions. If variation in these measurable dimensions can be shown to have either a marked, or negligible, effect on performance then inspection tolerances can be tightened or relaxed, respectively. The result can be reduction of costly scrappage or rework, or improved real performance, with no change to the fundamental design.

In wishing to approach a design problem in a stochastic manner, the engineer currently faces an abundance of possible methods. They can be categorized as two distinctly different classes: *intrusive*, and *non-intrusive*. The first, developed initially for reliability-based problems, requires modification of the underlying analysis code. The second considers the analysis code as a so-called 'black-box'. The focus of this work is on non-intrusive methodologies. Although some FEA solvers now incorporate stochastic analysis as part of their analysis suite, they are not yet widely available or applicable to many types of problem. Existing analysis can also use a combination of software packages and the codes themselves can be in constant flux through research and development. The non-intrusive procedures allow flexibility in choice of analysis code and can be simply converted for application to a multitude of different problems. For these reasons we use

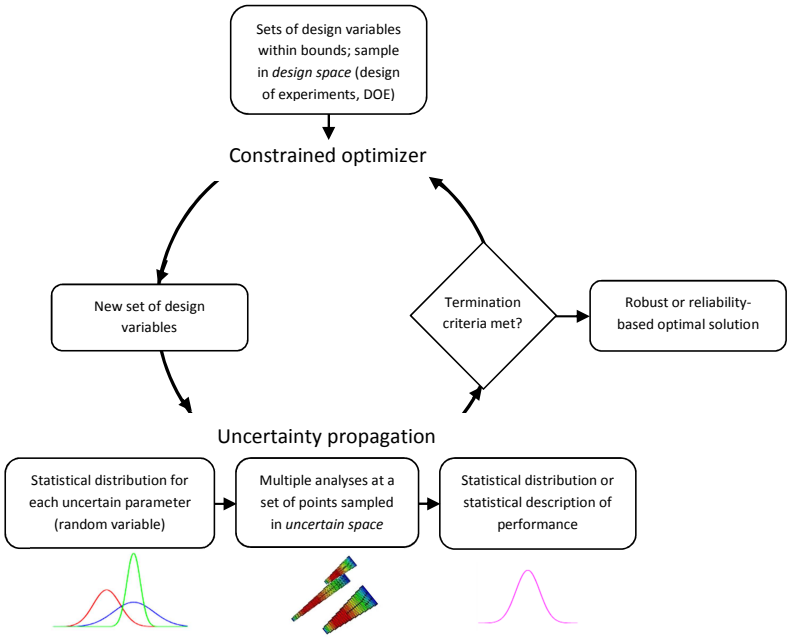


Figure 1.3: Stochastic analysis integrated into design search and optimization (DSO)

'black-box'-type approaches here for aerospace engineering design. They can be applied both to gain improvements in existing designs, yet can also keep up with new innovations that are currently in the preliminary design stage for air travel in twenty five years time or more.

The next step in design improvement is to enhance the DSO process with stochastic analysis, improving the designer's ability to answer our second question: is the deterministic optimum really the 'best' design? An automated stochastic analysis can be combined with deterministic DSO as either *robust design optimization* (RDO), or *reliability based design optimization* (RBDO), figure 1.3. The concept behind both is the notion that the nominally most highly performing design might also perform most variably and that the search should be a combination of a search for the most highly performing and most consistent design. In both RDO and RBDO, the measure of performance is generally either the most probable value, the *mode*, or the average value, the *mean*, of the output parameter. The measure of variability is the standard deviation for RDO, or probability of failure for RBDO. The optimization problem is now one of dual objectives and there are many possible *multi-objective* optimization (MOO) approaches, either requiring a weighting between the two objectives or resulting in an informative set of 'best' possible designs lying on, what is called, a *Pareto front* [75].

Despite the plethora of robust optimization methods in the literature, and more recently in some commercial software such as Isight^{TM} [46], most publicly accessible application of robust optimization remains in the academic literature and its routine use in engineering is far from common practice, particularly where the deterministic analyses are already extremely complex or the input uncertainties are poorly understood. This problem can be pronounced where the variability is in geometry.

1.2 The facilitation of practical stochastic analysis for geometric uncertainty

It is perhaps not surprising that UP and RDO are not routinely applied in the engineering design of complex systems, for a number of reasons.

1. Existing simulation workflows are usually:
    - expensive (time-consuming to run),
    - continually evolving, and
    - complex (often involving multiple different analysis types and codes, and with highly non-linear response functions).
2. Accurate and informative performance variability predictions require:
    - input variability to represent the 'truth' accurately (including correlations between uncertainties), and
    - large numbers of simulations to be run.
3. Designs are defined by multiple design variables (including geometric parameters) which:
    - may be large in number, and
    - are often not the same parameters that accurately represent the uncertainty.

These features point to three major 'stumbling blocks' for stochastic analysis:

1. cost,
2. lack of knowledge (or data),
3. appropriate means of CAD or mesh manipulation.

As discussed previously, the difficulty of dealing with high cost simulations and high numbers of variables is prohibitive and has been addressed by many researchers through,

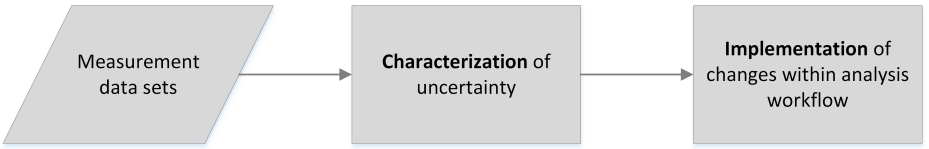


Figure 1.4: Necessary steps as a precursor to geometric uncertainty propagation and robust design optimization

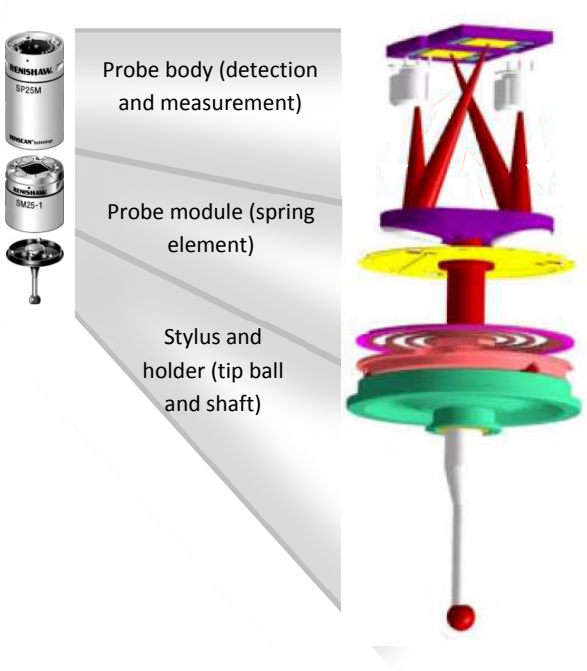


Figure 1.5: Renishaw SP25M measuring probe [231]

for example: *meta-models*, effective *sampling* methods, and *dimension reduction*. This must be considered, and some such techniques are used throughout this thesis, but it is the remaining two 'stumbling blocks' upon which this work is focused. Even small studies can be rendered almost impossible, or the results misleading, by lack of understanding of the real input uncertainty and an inability to implement modifications to existing code effectively. The necessary means of obtaining, or quantifying, uncertainty data varies from problem to problem. These **crucial steps** (figure 1.4), which enable accurate UP and RDO, are seldom addressed in the literature and independent variables with *normal Gaussian* distributions or simple intervals are often, not unreasonably in some cases, assumed for input variables, simplifying the algebra, e.g. Putko et al. [218].

For geometric uncertainty, there is a wealth of measurement data sometimes already stored by manufacturers; routine inspection processes are already making measurements using devices such as coordinate measuring machines (CMMs) with a physical probe (such as that of Renishaw, figure 1.5), resulting in sets of points in Cartesian coordinates. An example measurement data set from a firtree-shaped profile is shown in figure 1.6.

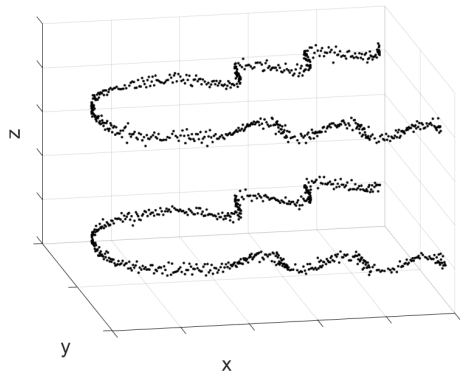


Figure 1.6: Typical coordinate measuring machine (CMM) data set for a single firtree-shaped slot. Probe traces front and rear profile (quantity of data limited by time and probe access restrictions)

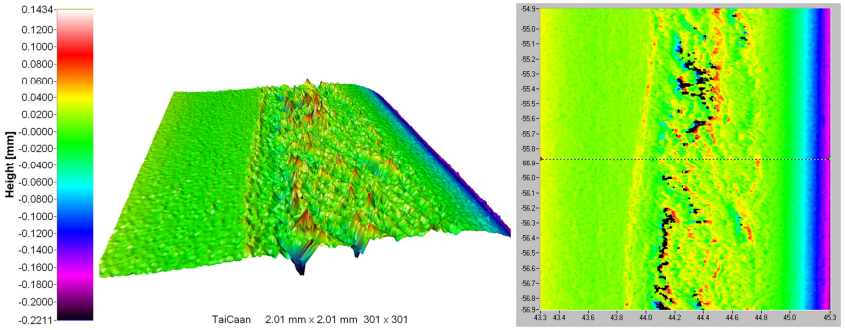


Figure 1.7: White light sensor image of a single blade firtree flank section post-service showing resultant profile shape alteration

Similar data sets of varying resolution can be obtained using alternative devices. For example, a white light scanning device[4] was used to gain an insight into the true surface profile of a small section of a single turbine blade pressure flank after service, figure 1.7. The resulting sets of points on the surface (*point clouds*) might cover only a linear measurement or a full surface. Although the financial cost of full three dimensional scans is high, recent developments mean that the ratio of scan accuracy to time and cost is increasing rapidly [204]. This work is undertaken in the understanding that **an automatic approach for incorporating this increasing quantity of available data into parametric design geometry for design search and analysis is likely to be beneficial, enabling robustness studies that use realistic inputs, and strengthening the feedback loop between design and production**.

There are, however, obstacles to this automation, apparent in cases of geometric uncertainty. Firstly, the parameter set used for automated design may need to be replaced

[4]Profile measured courtesy of Taicaan Technologies Europe, using a Xyris™ profiler. Image created using Taicaan Technologies BODDIES™ software suite.

with a new set of 'uncertain' parameters which represent the 'true' component shape, and can be sufficiently accurately approximated by statistical distributions with appropriate correlations, i.e. can be **characterized**. Secondly, for 'black-box' approaches, parameter changes within the uncertain space must be able to be **implemented** through modification of either the geometric definition or the discretized version of the geometry (mesh) in the existing analysis. It is noted that any implementation must be made to an analysis code that is of high enough fidelity to reflect the effects of changes in the order of magnitude of the uncertainty.

In this thesis, we therefore concern ourselves with the central question:

> **how can measurement data from inspection be used to characterize geometric uncertainty effectively, and implement changes within existing analyses, for the facilitation of automated uncertainty propagation and robust design?**

We begin in chapter 2 by introducing a collection of existing techniques which are either employed throughout this thesis or are considered central to the UP or RDO frameworks discussed throughout. First, those which are usually employed in the design search and optimization part of the problem, are described. Due to their versatility, these sampling schemes, response surface methods, and optimization routines, are central to this work, also being applied here to geometry fitting and manipulation. We follow this with details of more traditional geometry fitting routines, and means of evaluating fit accuracy. Given that statistical validity is a requirement, some statistical tools of note are also described, including the recently formulated *stable* distribution, which allows for greater spread and peakiness than the popular Gaussian normal. We then delve into *principal component analysis* (PCA) as a procedure for dimension reduction, which is critical if UP or RDO are to be practicable. The remaining methods explained in chapter 2 are *spatial deformation* and mesh manipulation tools, used widely for implementing free-form morphing of existing geometry in various forms.

The following chapter shows how this plethora of existing methods from chapter 2 can be applied in order to address the above central question, compared with the way in which this problem has previously been undertaken.

In the next three chapters, we address three different aero-engine component uncertainty problems. Each chapter demonstrates how a set of tools from chapter 2 can be applied to facilitate the use of existing data, from inspection, in a UP or RDO workflow. In each case, component performance in service is known to vary from the predicted value, and further investigation into the effects of known geometric uncertainties is needed. Chapter 4 addresses a problem where implementation of changes to design parameters is already

feasible within existing workflows, but characterizing the inputs for uncertainty analysis requires a novel approach. In chapter 5 there are measurements of existing parameters but the complex workflow precludes changes to parametric geometry models, requiring an automated approach to implementing geometric changes within an existing CFD mesh. Chapter 6 introduces a problem, with more generic application, where the shape of a surface depicted by measurement data sets cannot be represented by parameters corresponding to design parameters, or easily adjusted parameters in existing geometric models. Here, the uncertainty, manifesting as complex curves, is modelled using a novel approach that combines existing techniques to result in a reduced set of characterized parameters. Following this, a method of implementing changes to these new parameters within an existing workflow is demonstrated.

Throughout chapters 4, 5, and 6, tools have been selected depending on the attributes of the measurement data and the analysis to which the geometric changes are to be applied. Although the approaches themselves are not necessarily novel, their combination and application to the characterization and implementation problem is original. In chapter 7 the approaches taken are summarised with a description of how the methods are applied within the framework given in chapter 3. The contributions made throughout this work are reviewed, and areas for future development are highlighted. In particular, we point to the generalized approaches developed throughout chapter 6 that could be extended to, and tested on, larger measurement data sets, for further application as measuring technologies advance.

Chapter 2

Existing methods

Consider a concept design for a component that performs sufficiently well under 'working' conditions. Following its initial design, the concept has been fixed. It has then been refined through increased analysis model fidelity, and further still by design process automation and DSO. At this point, we observe that the performance of the part varies from the predicted nominal and parts can be unreliable compared with their supposedly less well performing predecessors. The increased performance afforded to us through DSO needs to be maintained, but we must now also consider uncertainty. If the uncertainties in input parameters to the analysis model can be identified and characterized, then we may quantify the expected uncertainty in performance at the optimal design point, by implementing changes through existing workflows, using UP (sometimes also referred to as UQ). It may then also be possible to understand the effects of the uncertainties throughout the design space, and extend the optimization procedure to RDO and RBDO. To facilitate this further-informed design decision making process, a number of different techniques and processes need be combined to good effect. This chapter introduces a set of useful tools, figure 2.1, with a review of the literature pertaining to those that are considered central to this thesis. The methods which have been directly applied throughout chapters 4, 5, and 6, are italicised in the table. The remaining methods discussed are either alternatives to those chosen, or are the methods that may be facilitated by the **characterization** or **implementation** of uncertainty on which this thesis is focused.

Response surface modelling and optimization tools			
Sampling plans (space-filling)	**Kriging; a radial basis function model**	**Uncertainty propagation (black-box and non-linear)**	**Design search and optimization**
Full-factorial Optimal Latin hypercube Orthogonal array based	*Interpolating* *Regressing* *Fixed noise parameter* *Universal*	Monte Carlo Using a meta-model Quadrature	Single objective Using a surrogate model Local searches *Global searches* Multi-objective optimization Robust design optimization Reliability-based design optimization

Geometry measurement and modelling tools		
Curve and surface fitting	**Evaluating the accuracy of fitting**	**Geometry and mesh manipulation**
Registration and segmentation Polynomial models and *least-squares regression* Orthogonal least-squares and similar Spline fitting Orthogonal distance based spline fitting	Goodness-of-fit and confidence *Validation by re-sampling* *'Proof-by-example'*	Surface mapping *Spatial deformation*

Statistical and mathematical methods	
Statistical methods	**Dimension reduction**
Popular distributions *Stable distributions* *Hypothesis tests* *Correlation*	*Principal component analysis*

Table 2.1: Tools for uncertainty studies in engineering design discussed and employed (italicised) in this thesis

2.1 Space-filling sampling plans

Sampling plans are integral to informed design selection through analysis, or testing. Whether selecting computer or physical experiments for design space modelling or uncertain space modelling, or taking sample measurements, the selection of sample points is critical to the cost of the process and the accuracy of the resulting parameter prediction. For many of the methods presented in this chapter, an analysis code is sampled at various input parameter values. Geometric measurement points can also be considered sample points and formally selected using an appropriate strategy, although this is often not the case. In both of these cases a sampling strategy for building a model of the whole space, where all regions are equally important, as accurately as possible, using the smallest possible number of points or to a number of points (time and cost) budget: a *design of experiment* (DoE), is usually required. These methods are termed *space-filling*, and some popular space-filling sample plans are given in detail in appendix A.1:

- the simplest, evenly spaced, *full factorial* design [72],
- pure random sampling,
- *Latin hypercube sampling* (LHS) [172] and *optimal Latin hypercube* (OLH) sampling [121, 181], and
- *orthogonal arrays* (OAs) [237, 103, 278].

Further problem-specific refinement of these sampling methods, and alternative non-space-filling designs for application to propagating uncertainty, are discussed as appropriate throughout this chapter.

2.2 Fitting design space models to sample points

Much like the design of sampling plans is integral to an informed design selection process, so too is the modelling of a design space predicated upon such a DoE. A suitable model can provide a fast-running (low-cost) alternative to evaluating a complex analysis workflow. In many engineering design cases similar to those this thesis seeks to facilitate, such models can ensure that automated design searches are practicable. These models are termed *surrogate models* (also referred to as *response surface models* (RSMs), meta-models, *emulators*, or simply *approximations*).

Over recent years there has been a significant rise in the popularity of *radial basis function* (RBF) based models, and in particular *Kriging*[1] [242, 79] for surrogate modelling in

[1]Kriging is named in honour of Danie Krige; the South African mining engineer who first developed the method in 1951.

engineering design search, which is a Gaussian process based model (*Gaussian process emulator*) that assumes that the points on which the functions are built are realizations of a random process. These models are central to much of the work in this thesis. Not only are they useful in the context of enabling costly optimization, including design search with uncertainty, but they are also applied in later chapters to model geometric surfaces and displacements in the context of deformation of geometry.

These kinds of models are flexible in that, provided there are enough sample points, a generic model can be constructed based on any number of input variables. Although this means that their modelling power extends to hyperspaces, they are not a panacea because the number of sample points needed increases exponentially with the number of variables (dimensions); the curse of dimensionality [78].

In this section, we introduce the method of constructing a radial basis function (RBF) model and describe the special case of the Kriging model in detail.

2.2.1 Interpolating Kriging

Model construction

Kriging is a specific case of a radial basis function (RBF) model used most prolifically to fit curves or multi-dimensional surfaces to data points in surrogate modelling for design space representation in *design optimization*. RBFs are discussed by Broomhead and Lowe [21] and reviewed in terms of surrogate modelling by Forrester et al. [78]. They are a weighted sum of *basis functions*. Using the notation of Forrester and Keane [79], the basis functions, ψ, are functions of the Euclidean distance between the evaluation point, $\mathbf{x}$, and n_c basis function centres, $\mathbf{c}^{(i)}$, $||\mathbf{x}-\mathbf{c}^{(i)}|| = r$. The estimated function is expressed as

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{n_c} w_i \psi(r). \tag{2.1}$$

The basis functions may be fixed, varying only with r, or parametric, varying with an additional parameter or parameters. Whatever form the basis functions take, w can be calculated via interpolation at the sample points. In matrix form, if the bases are considered to be at the data point locations; $n = n_c$ and $\mathbf{c}^{(i)} = \mathbf{x}^{(i)}$, the interpolation condition is

$$\mathbf{y} = \mathbf{\Psi}\boldsymbol{w}, \tag{2.2}$$

where $\mathbf{y}$ is a vector of sample response values; the measured data, $\mathbf{\Psi}$ is an $n \times n$ matrix of basis function values evaluated at the data points, also known as the *Gram matrix*, and $\boldsymbol{w}$ is the vector of weights.

In reality, a limited number of standard basis functions are used depending on the application. The parametric Gaussian basis function

$$\psi(r) = e^{-r^2/2\sigma^2}, \tag{2.3}$$

is often a popular choice because it has two benefits for response modelling. First, $\mathbf{\Psi}$ is then a positive definite (symmetric) matrix, greatly simplifying and guaranteeing a solution to (2.2) via *Cholesky factorization*[2]. Second, it is possible to estimate the error of the model at any point. For parametric basis functions it becomes important to find a solution for $\mathbf{w}$ efficiently and to estimate error. This is because one must also determine the value of the other parameters (e.g. σ for the Gaussian basis function) and $\mathbf{w}$ will need to be calculated multiple times during this process. Parameter selection is generally a minimization of the estimated model error metric, which is usually a cross-validation error or a sum of squared prediction errors (section 2.8).

One may notice the similarity between the above model formulation and a weighted spline-based curve fit. Cressie [44] p180, describes the "formal connection" between Kriging and splines but notes that "there is a large divergence in how they are applied and how their results are interpreted". Benefits of Kriging include the ease of prediction calculation and mean-squared error (MSE) prediction, discussed later in this section.

Kriging is an extension of the approach described thus far. It assumes that the response is a realization of a Gaussian process where its Gram matrix is a set of Gaussian basis functions which take the form of the correlation between two random variables, $\mathbf{Y}$:

$$\psi_{ij} = \text{cor}\left(Y(\mathbf{x}^{(i)}), Y(\mathbf{x}^{(j)})\right) = \exp\left(-\sum_{k=1}^{d} \theta_k |x_k^{(i)} - x_k^{(j)}|^{p_k}\right), \tag{2.4}$$

and covariance (known as a *covariance kernel* in the data-mining community):

$$\sigma^2\psi_{ij} = \text{cov}\left(Y(\mathbf{x}^{(i)}), Y(\mathbf{x}^{(j)})\right) = \sigma^2 \exp\left(-\sum_{k=1}^{d} \theta_k |x_k^{(i)} - x_k^{(j)}|^{p_k}\right), \tag{2.5}$$

where d is the number of dimensions and θ_k and p_k are parameters which can optimized, or tuned. The distance metric used here is non-Euclidean in that it is the absolute value of the distance between points in a given dimension. Jones [124] explains that the use of the Euclidean distance without the use of parameters such as θ_k and p_k to scale it, are "sensitive to the units of measurement". Normalizing all the data to a unit interval is standard practice to overcome this problem. However, this "treats all variables as equally important – something that is almost never true". Kriging mitigates this effect by the tuning of the values of θ_k and p_k.

[2]"Given a symmetric positive definite matrix $\mathbf{A}$, the Cholesky decomposition is an upper triangular matrix $\mathbf{U}$ with strictly positive diagonal entries such that $\mathbf{A} = \mathbf{U}^{\mathrm{T}}\mathbf{U}$", [307]

The value of constant $\theta_k \geq 0$ is related to how rapidly the function changes in the k^{th} dimension. It is usually given a value between 10^{-3} and 10^2 [78]. The smoothness of the model in the k^{th} dimension is determined by $0 < p_k \leq 2$ (with the inclusion of p this is sometimes referred to as the *power exponential kernel* [196]). It is often fixed at $p = 2.00$ (the *Gaussian kernel* [196, 207]), or slightly lower at $p = 1.99$ to improve numerical stability. This creates smooth functions as lower values of p exhibit roughness. The correlation, ψ, with changing values of p and θ are plotted by Forrester et al. [78].

Alternative covariance kernels include the *Matérn kernel* [207, 226]:

$$\sigma^2\psi(r) = \frac{2^{(1-\nu)}}{\Gamma(\nu)}\left(\frac{r\sqrt{2\nu}}{l}\right)^{\nu} K_\nu\left(\frac{r\sqrt{2\nu}}{l}\right), \tag{2.6}$$

where Γ is the gamma function [141], K_ν is a modified Bessel function of the second kind [141], l is a positive constant which is equivalent to $\frac{1}{\theta}$, and ν is another positive constant. ν is typically set to $\frac{3}{2}$. The Gaussian kernel is, in fact, a specific case of the Matérn kernel where $\nu \to \infty$. Picheny et al. [207] compare the use of the Gaussian and Matérn $\frac{3}{2}$ kernel concluding that the choice is of "limited influence" and that "other factors, such as ... the robustness of the covariance parameter estimation" dominate performance.

Model training

Let us consider an interpolating Kriging model. If it is assumed that each data point, y_i, is a realization of the random field Y (a field with Gaussian normal distribution with mean μ and variance σ^2), then $\mathbf{y}$ can be substituted for $\mathbf{Y}$. If μ, σ, $\boldsymbol{\theta}$, and $\boldsymbol{p}$ are found such that the statistical *likelihood*, L, is maximized then we are selecting the model for which the data points sampled are the most likely responses at the given $\mathbf{x}$'s.

We turn our attention now to the salient points relating to this maximization, discussed in depth by Jones [124]. To capture the correlation between the y_is, consider the multivariate normal distribution

$$\mathbf{y} \sim \mathrm{N}(\boldsymbol{\mu}, \mathbf{V}). \tag{2.7}$$

$\boldsymbol{\mu}$ is the $n \times 1$ expectation vector $\mathrm{E}(\mathbf{y})$ and $\mathbf{V}$ is the $n \times n$ covariance matrix $\mathrm{cov}(\mathbf{y})$. Such a distribution has the density function

$$P(\mathbf{y}) = \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{V}|^{\frac{1}{2}}}\exp\left\{-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu})^T\mathbf{V}^{-1}(\mathbf{y}-\boldsymbol{\mu})\right\}, \tag{2.8}$$

and it can be shown that the likelihood of an individual data set given this distribution is

$$L(\mathbf{y}|\boldsymbol{\mu}, \mathbf{V}) = \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{V}|^{\frac{1}{2}}}\exp\left\{-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu})^T\mathbf{V}^{-1}(\mathbf{y}-\boldsymbol{\mu})\right\}, \tag{2.9}$$

[122]. Given that μ and σ^2 are constant in this case, $\boldsymbol{\mu}$ can be expressed as

$$\boldsymbol{\mu} = \mathbf{1}\mu, \tag{2.10}$$

where $\mathbf{1}$ is and $n \times 1$ vector of ones, and the covariance $\text{cov}(y_i, y_j) = \sigma_{y_i}\sigma_{y_j}\text{cor}(y_i, y_j)$ in matrix form is

$$\mathbf{V} = \text{cov}(\mathbf{y}) = \sigma^2\boldsymbol{\Psi}, \tag{2.11}$$

[309]. Substituting these into equation 2.9 results in

$$L(\mathbf{y}|\mu, \sigma) = \frac{1}{(2\pi\sigma^2)^{\frac{n}{2}}|\boldsymbol{\Psi}|^{\frac{1}{2}}}\exp\left\{-\frac{1}{2\sigma^2}(\mathbf{y} - \mathbf{1}\mu)^T\boldsymbol{\Psi}^{-1}(\mathbf{y} - \mathbf{1}\mu)\right\}. \tag{2.12}$$

The problem can be simplified by taking the natural logarithm of the likelihood function such that derivatives can be taken with respect to μ and σ^2:

$$\ln(L) = -\frac{n}{2}\ln(2\pi) - \frac{n}{2}\ln(\sigma^2) - \frac{1}{2}\ln|\boldsymbol{\Psi}| - \frac{(\mathbf{y} - \mathbf{1}\mu)^T\boldsymbol{\Psi}^{-1}(\mathbf{y} - \mathbf{1}\mu)}{2\sigma^2}. \tag{2.13}$$

Setting the derivatives equal to zero and solving yields estimates for μ and σ^2 in terms of $\boldsymbol{\Psi}$:

$$\hat{\mu} = \frac{\mathbf{1}^T\boldsymbol{\Psi}^{-1}\mathbf{y}}{\mathbf{1}^T\boldsymbol{\Psi}^{-1}\mathbf{1}}, \tag{2.14}$$

and

$$\hat{\sigma}^2 = \frac{(\mathbf{y} - \mathbf{1}\hat{\mu})^T\boldsymbol{\Psi}^{-1}(\mathbf{y} - \mathbf{1}\hat{\mu})}{n}. \tag{2.15}$$

Substituting back into (2.13) results in the *concentrated ln-likelihood function*:

$$\ln(L) \approx -\frac{n}{2}\ln(\hat{\sigma}^2) - \frac{1}{2}\ln(|\boldsymbol{\Psi}|), \tag{2.16}$$

which one seeks to maximize using some preferred numerical optimization routine in order to specify $\boldsymbol{\theta}$ and $\boldsymbol{p}$, using (2.14) and (2.15) to estimate μ and σ^2. Typically, techniques such as combined global and local searches, section 2.3, are preferred because the concentrated ln-likelihood function is generally highly multi-modal [78].

To reiterate, in contrast to other RBF models, the Kriging model is trained, not by minimizing the estimated error, but by choosing parameters that maximize the model likelihood; the probability of achieving the y values in the sample or data set. Kriging "is the least assuming method..." considered "...in terms of the range of function forms it can emulate, and it is for this reason that it is so effective" [79]. In tests on multidisciplinary design optimization problems Simpson et al. [260] find Kriging models "yield global approximations that are slightly more accurate" than the other response surface methods tested.

Model evaluation; prediction

Once estimates have been obtained for $\boldsymbol{p}$ and $\boldsymbol{\theta}$, the function value at any $\mathbf{x}^*$ can be predicted. In order to do this, it is considered that the prediction should be consistent with our model of the correlation, with the parameters chosen thus far. As such, the new value, $\hat{y}$ is incorporated into the correlation matrix, $\boldsymbol{\Psi}$, to give the enlarged correlation matrix, $\boldsymbol{\Psi}^*$:

$$\boldsymbol{\Psi}^* = \begin{bmatrix} \boldsymbol{\Psi} & \boldsymbol{\psi} \\ \boldsymbol{\psi}^T & 1 \end{bmatrix} \tag{2.17}$$

where $\boldsymbol{\psi}$ is a vector of correlations between each of the existing points and the new point at $\mathbf{x}^*$. The ln-likelihood (2.13), removing the terms that are independent of $\hat{y}$, then becomes

$$\ln(L) = \frac{-\begin{bmatrix} \mathbf{y} - \mathbf{1}\hat{\mu} \\ \hat{y} - \hat{\mu} \end{bmatrix}^T \begin{bmatrix} \boldsymbol{\Psi} & \boldsymbol{\psi} \\ \boldsymbol{\psi}^T & 1 \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{y} - \mathbf{1}\hat{\mu} \\ \hat{y} - \hat{\mu} \end{bmatrix}}{2\sigma^2}. \tag{2.18}$$

By arranging as a quadratic function in $\hat{y}$ and setting equal to zero, an analytical solution for the $\hat{y}$ with maximum $\ln(L)$; the maximum likelihood estimate (MLE) of $\hat{y}$, can be found:

$$\hat{y}(\mathbf{x}^*) = \hat{\mu} + \boldsymbol{\psi}^T \boldsymbol{\Psi}^{-1}(\mathbf{y} - \mathbf{1}\hat{\mu}). \tag{2.19}$$

This is also known as the *Kriging mean*. Details of its derivation can be found in [78] and [108].

A demonstration of interpolating Kriging

Interpolating Kriging is used here to model a known curve:

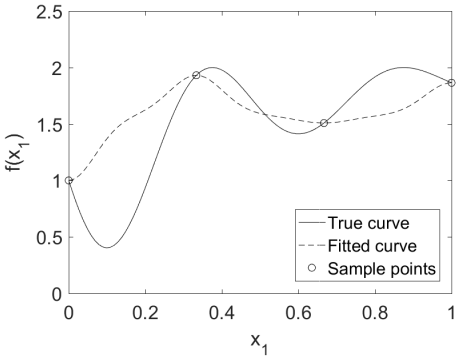
$$f(x_1) = -(\sin(4\pi) + 1)e^{-2x_1} + 2, \tag{2.20}$$

using an increasing number of sample points evenly distributed in x_1, figure 2.1. A generally accepted 'rule of thumb' suggests that $10 \times d$ (the number of dimensions) is a sufficient number of computer experiments for fitting a response surface model of this type [159]. This is reflected in this example, with 10 points providing a 'good' fit of the given curve.

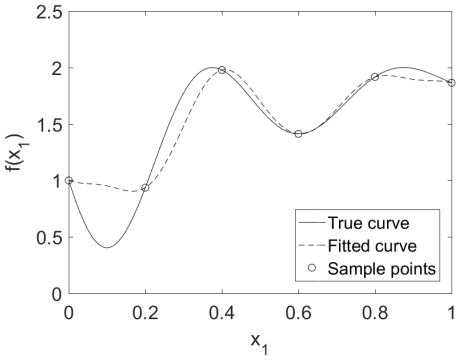
A two-dimensional function (a surface), based on the 'bird function' of Mishra [178]:

$$f(x_1, x_2) = \sin(-10x_1)e^{(1-\cos(-10x_2))^2} + \cos(-10x_1)e^{(1-\sin(-10x_2))^2} + (10x_2 - 10x_1)^2, \tag{2.21}$$

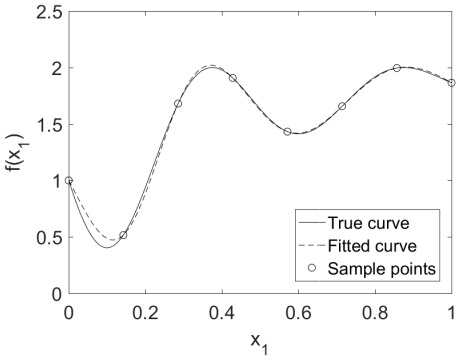
figure 2.2, is used to further demonstrate the use of a Kriging model to approximate a true response. The function is fitted using an increasing number of OLH sample points



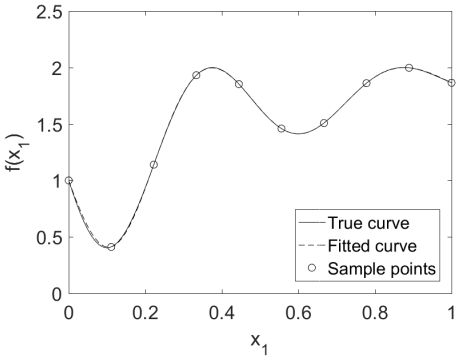
(a) 4 evenly distributed sample points



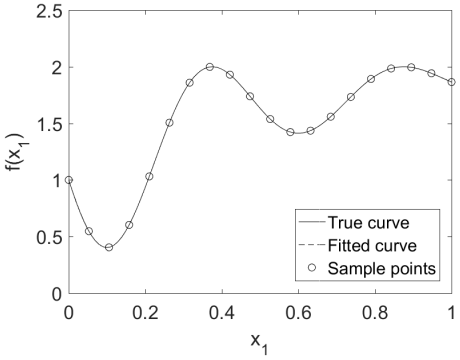
(b) 6 evenly distributed sample points



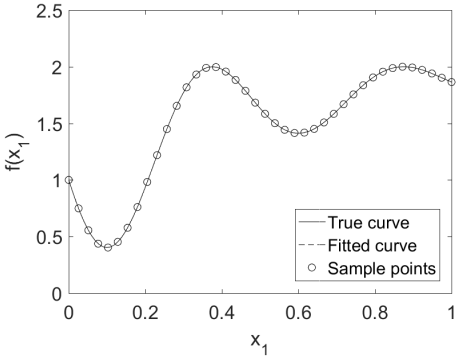
(c) 8 evenly distributed sample points



(d) 10 evenly distributed sample points



(e) 20 evenly distributed sample points



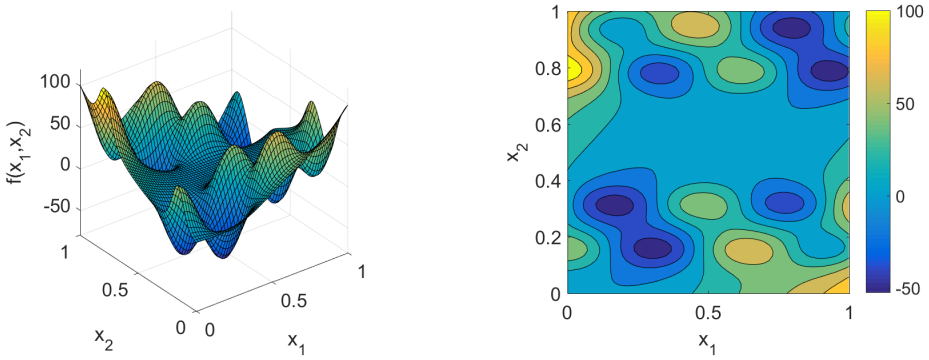
(f) 40 evenly distributed sample points

Figure 2.1: Interpolating Kriging fit in a single dimension

(section 2.1 and appendix A.1.3)[3], figure 2.3. Given the complexity of this particular function the $10d$-rule falls short of accurately representing all of the surface contours. It does, however, point to regions of peaks and troughs, useful for optimization, but possibly insufficient for surface reconstruction.

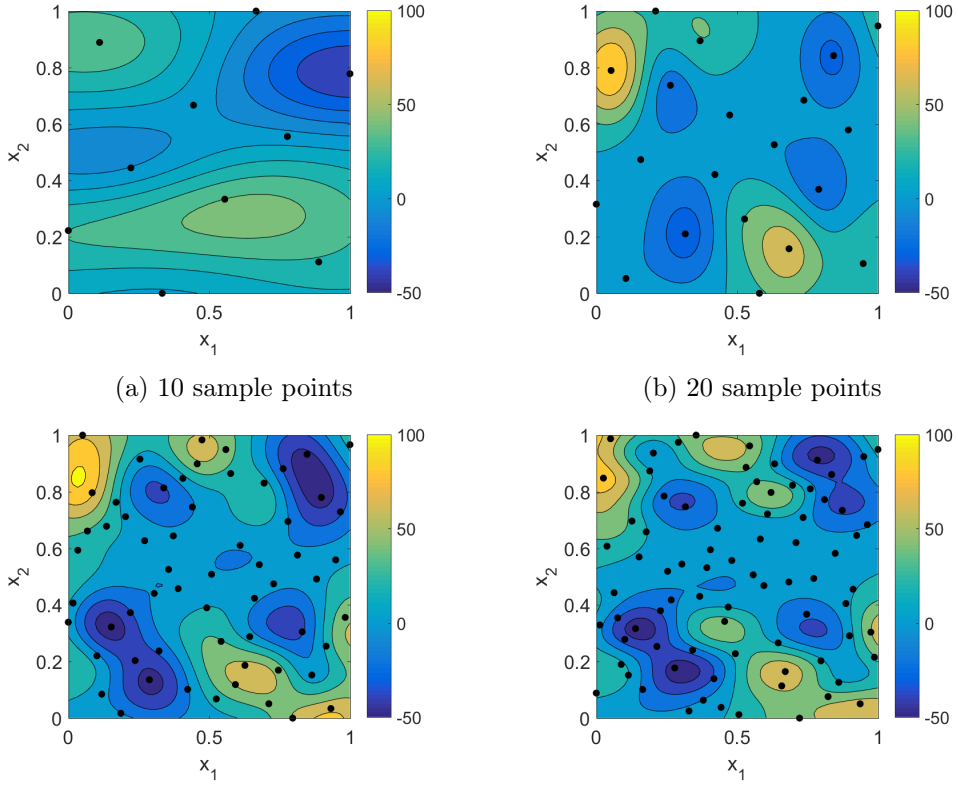
Kriging models have some limitations of note: they are known to lack accuracy near

[3]The OLH samples are optimized using the Morris and Mitchell [181] maximin criteria and a genetic algorithm (appendix A.4) with 10 individuals and 20 generations. As discussed in appendix A.1.3, a truly optimal solution is untenable given time constraints.



(a) Surface plot (b) Contour plot

Figure 2.2: Example surface function example for demonstration of Kriging



(a) 10 sample points (b) 20 sample points

(c) 60 sample points (d) 80 sample points

Figure 2.3: Interpolating Kriging fit in two dimensions using optimal Latin hypercube sampling

bounds, and they are unable to represent discontinuities or non-smooth functions without the tuning of p. It is also noted that variables should be normalized prior to constructing the Kriging model [198] to ensure that each dimension is considered of equal importance during hyperparameter tuning. This means that the same bounds can be applied for hyperparameters in each direction and that the resulting hyperparameters can be interpreted to provide some understanding of the relative effect of each variable on the output.

Evaluation of the accuracy of these models, and others, is discussed in section 2.8.

2.2.2 Regressing Kriging

So far our description of RBFs and Kriging has considered interpolation of data, yet there are many difficulties with interpolation, including over-fitting, where the fitted shape has a high-order due to the interpolation constraint. To fit a regressing model through the data, we assume that the error can be considered to be random (normally distributed). One can take an approach similar to that of polynomial fitting and, instead of increasing the model order, increase the number of bases, n_c, until $\delta_K^2/(n - n_c)$ no longer significantly reduces with an increase in bases [129]. δ_K^2 is a validation metric, usually the sum of the squared deviations of the model from the training points.

Alternatively, one can include the *regularization parameter* λ in the main diagonal of $\mathbf{\Psi}$ [104, 77], typically in the range 10^{-4} to 10^{-1} for normalized data. The set of equations then becomes

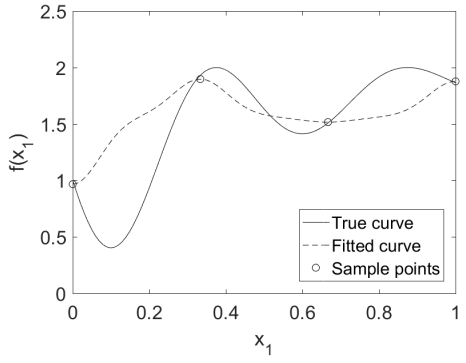
$$\mathbf{w} = (\mathbf{\Psi} + \lambda\mathbf{I})^{-1}\mathbf{y}, \tag{2.22}$$

where $\mathbf{I}$ is an $n \times n$ identity matrix. λ is then related to the variance of the measurement *noise* and

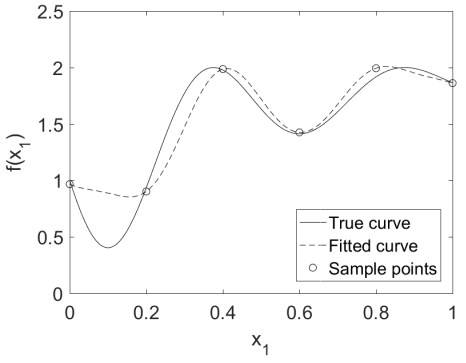
$$\tilde{\mathbf{\Psi}} = (\mathbf{\Psi} + \lambda\mathbf{I}) = \begin{bmatrix} 1+\lambda & \psi_{1,2} & \psi_{1,3} & \dots & \psi_{1,n} \\ \psi_{2,1} & 1+\lambda & \dots & \dots & \dots \\ \psi_{3,1} & \dots & 1+\lambda & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \psi_{n,1} & \dots & \dots & \dots & 1+\lambda \end{bmatrix}. \tag{2.23}$$

Estimates $\hat{\mu}$ (2.14), $\hat{\sigma}^2$ (2.15), and $\hat{y}(\mathbf{x}^*)$ (6.10), remain valid and λ can be determined along with the other Kriging parameters, $\boldsymbol{\theta}$ and $\boldsymbol{p}$, yielding the least-squares solution. One could also consider the relationship between λ and a known variance in the data to fix the value of λ for the fit. This type of approach is discussed in chapter 6.

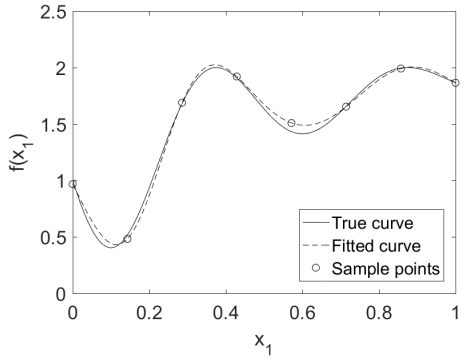
To illustrate regressing Kriging a small amount of random noise (with variance $\tau^2 = 1.5^2$ has been added to each sample point, i.e. in the f-direction (the response), not orthogonal to the curve as it could be in the case of geometric fitting. The functions 2.20 and 2.21 are used again, resulting in the fitted curves and surfaces in figures 2.4 and 2.6. The ln-likelihood is plotted with varying λ and θ for the single dimension problem, showing that there is a distinct minimum, figure 2.5.



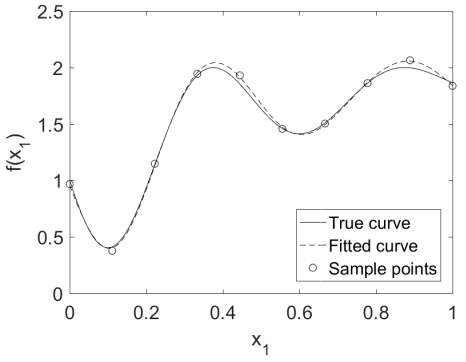
(a) 4 evenly distributed sample points



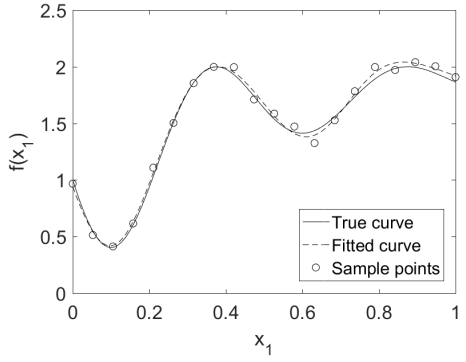
(b) 6 evenly distributed sample points



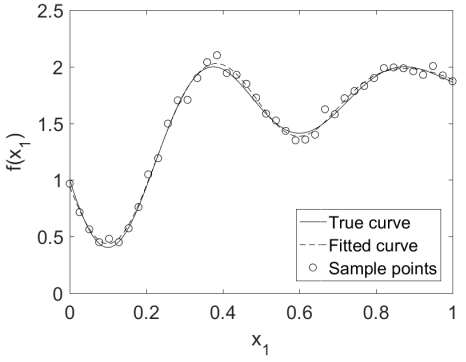
(c) 8 evenly distributed sample points



(d) 10 evenly distributed sample points

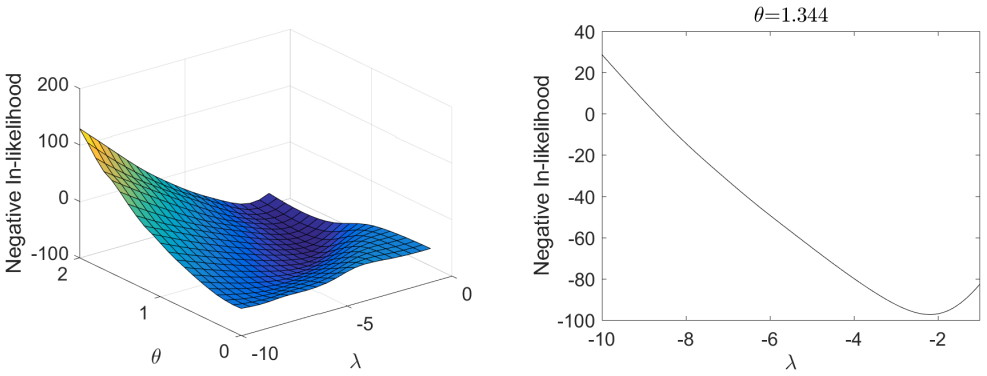


(e) 20 evenly distributed sample points



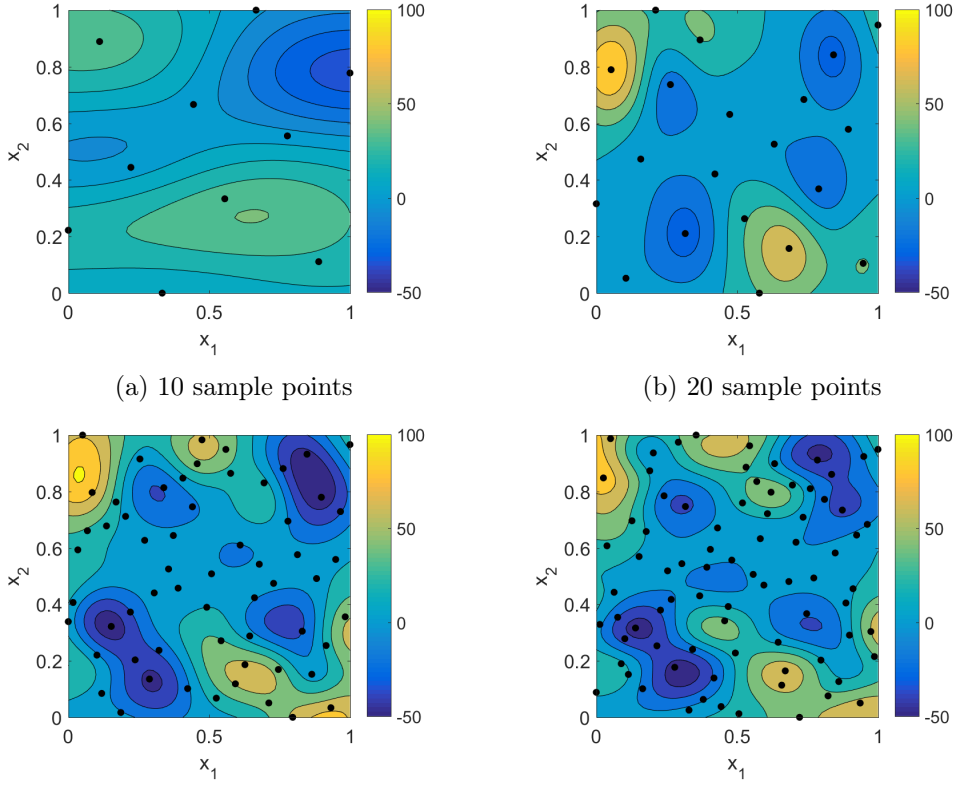
(f) 40 evenly distributed sample points

Figure 2.4: Regressing Kriging fit in a single dimension



(a) λ vs. θ vs. neg. ln-likelihood (b) λ vs. neg. ln-likelihood at chosen θ

Figure 2.5: Variation in negative ln-likelihood using regressing Kriging in a single dimension, 40-point sample



(a) 10 sample points (b) 20 sample points

(c) 60 sample points (d) 80 sample points

Figure 2.6: Regressing Kriging fit in two dimensions optimal Latin hypercube sampling

2.2.3 Regressing Kriging with a fixed noise parameter

If the level of noise is known, as may be the case with measurement data, then a relatively little known method, used for Kriging-based optimization, by Picheny et al. [207] could be applied. The approach introduces an additional variance parameter τ^2, which is the known noise variance. The method is described in detail in chapter 6 with application to uncertainty characterization and is presented in appendix A.2 by illustrating it on the test functions 2.20 and 2.21.

2.2.4 Universal Kriging

A Kriging model assumes that each data point is a realization of a random field with Gaussian normal distribution with mean μ. *Universal Kriging* [44] attempts to include additional known trends in the response by assuming that the random field has a Gaussian normal distribution with mean that varies with $\mathbf{x}$ according to a specified function: a quadratic, or cubic for example. It can be argued that this approach is limited in its applicability. It has proved useful, however, in chapter 6 of this thesis.

An extension of universal Kriging is *Blind Kriging* [125], where the mean function used is selected from a set of candidates based on a Bayesian forward selection technique.

2.3 Design search and optimization

Given either a physics based computational model (analysis workflow) or a surrogate model of component performance, for example, to determine the 'best' design parameters one might turn to optimization, or the more aptly termed process of design search and optimization (DSO).

2.3.1 Single objective optimization; local and global approaches with constraints

Optimization with continuous real variables has a clear definition, and if we ignore any variation (stochastic or otherwise) then the process is deterministic and set out as follows for a single objective or goal function.

Find

$$\mathbf{x} \in \mathbb{R}^d$$

which minimizes

$$f(\mathbf{x}) \tag{2.24}$$

subject to the constraints

$$\begin{aligned} g_j(\mathbf{x}) &= 0 \qquad j = 1, ..., c_e \\ g_j(\mathbf{x}) &\leq 0 \quad j = c_e + 1, ..., c \\ \mathbf{x}_{\mathrm{L}} &\leq \mathbf{x} \leq \mathbf{x}_{\mathrm{U}} \end{aligned}$$

where $\mathbf{x}$ is a vector of design variables (the *design vector*). There are d variables and c constraints, of which c_e are *equality* constraints, and $c - c_e$ are *inequality* constraints [225]. In engineering design f is the chosen performance measure and *objective function* of the optimization.

Searches for an optimal design point can either be

(a) *local*: given a starting point, $\mathbf{x}_0$, the search moves downhill towards the optimum (*descent methods*) until a specified *convergence* condition is met [91, 131], or

(b) *global*: moving about the entire space in such a way as to ensure that multiple minima are not ignored [211].

Reviews of local and global approaches are provided in appendices A.3 and A.4, and a complete guide to the topic was written by Rao [225] in 2009.

Here, we demonstrate the use of a local optimizer (implemented via MATLAB's `fmincon` function) applied to a dovetail joint design case study presented in appendix B. The dovetail joint analysis is constructed as a two-dimensional contact problem with geometry as shown in figure 2.7, and nominal design stresses as shown in figure 2.8.

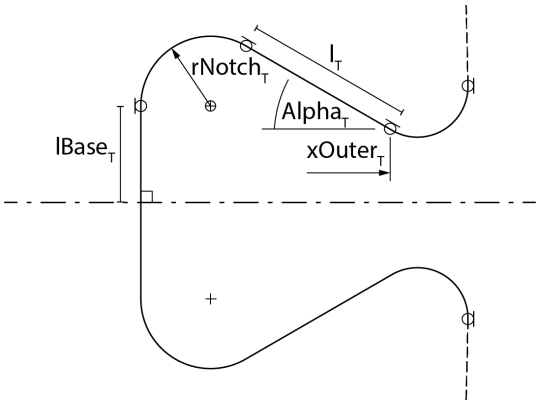


Figure 2.7: Dovetail joint slot parametric geometry for local DSO demonstration

The `fmincon` interior-point algorithm is a development of SQP where the iterations are a mixture of Newtonian-type linear steps and gradient-based steps using trust-region methods. Full details can be found in the research report by Byrd et al. [24].

In the dovetail contact problem $f(\mathbf{x})$ is the slot peak principal stress (away from edge of contact), calculated using a finite element model in ABAQUS software. $\mathbf{x}$ consists of

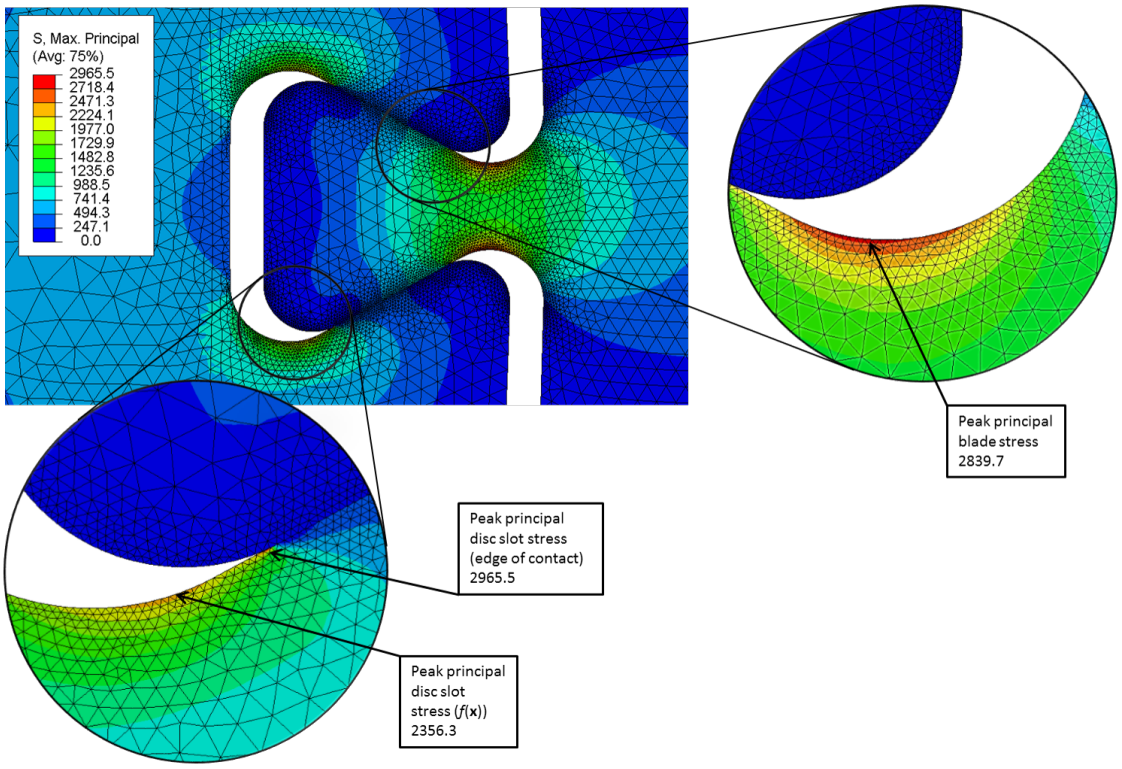


Figure 2.8: Peak principal stress (N/mm^2) contour map for nominal dovetail design

two design variables: AlphaT; the flank angle, and rNotchT; the slot notch radius. The design is constrained to be symmetric top and bottom and there are three constraint functions $g_j(\mathbf{x})$:

$$\begin{aligned} g_1(\text{rNotchT}, \text{AlphaT}) &= \min(\text{cT}, \text{cB}) \\ g_2(\text{rNotchT}, \text{AlphaT}) &= \text{wNeck} \\ g_3(\text{rNotchT}, \text{AlphaT}) &= \text{wPostT} + \text{wPostB} \end{aligned} \tag{2.25}$$

In order to plot a contour of the 'true' response, for illustrative purposes, a large (225-point) OLH sample is taken. x_1 is allowed to vary in the range [20,50] degrees and x_2 with the bounds [0.5,4] mm. The objective function is fitted to all the sample points using a Kriging model, figure 2.9. Each of the constraints is modelled in the same way using their values at the same set of sample points.

Using `fmincon` a constrained search, with workflow as illustrated in figure 2.10, and initial starting point at the nominal design, results in an optimum on the bound and constraint boundary (at AlphaT = 20°, and rNotchT = 3.2427 mm), figure 2.11. The search required 57 evaluations of the function.

The function is not very expensive in this case; each run takes only five minutes. Even so, 57 evaluations takes a significant amount of time ($4\frac{3}{4}$ hours). In section 2.3.2 we discuss expediting this process.

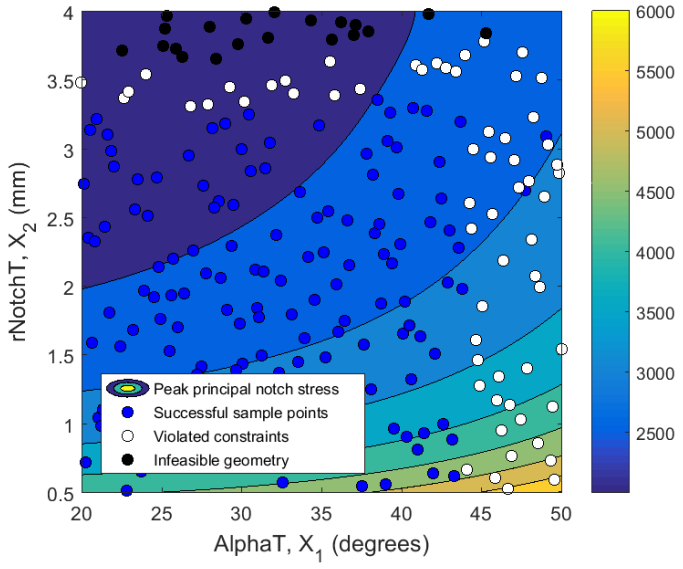


Figure 2.9: Contour plot of notch peak principal stress (N/mm^2) fitted using 225 OLH sample points

In this example case, the problem is fast-running and it is shown, by making a large number of calls to the function, that the function is not highly non-linear with multiple minima. A local search routine tends towards the only minima. In reality few engineering design problems with complex analysis codes can immediately be solved using local approaches. In particular, when there are large numbers of variables whose interactions with each other and the constraints are not obvious, one can easily find that a search gets stuck in an unexpected local minima at a constraint boundary. To ensure that the full design space is adequately searched, we turn to global optimization routines [211]. These include the common *genetic algorithms* (GAs) [92] and the more recently popularised *particle swarm optimization* (PSO) [62] routines which are described in appendix A.4.

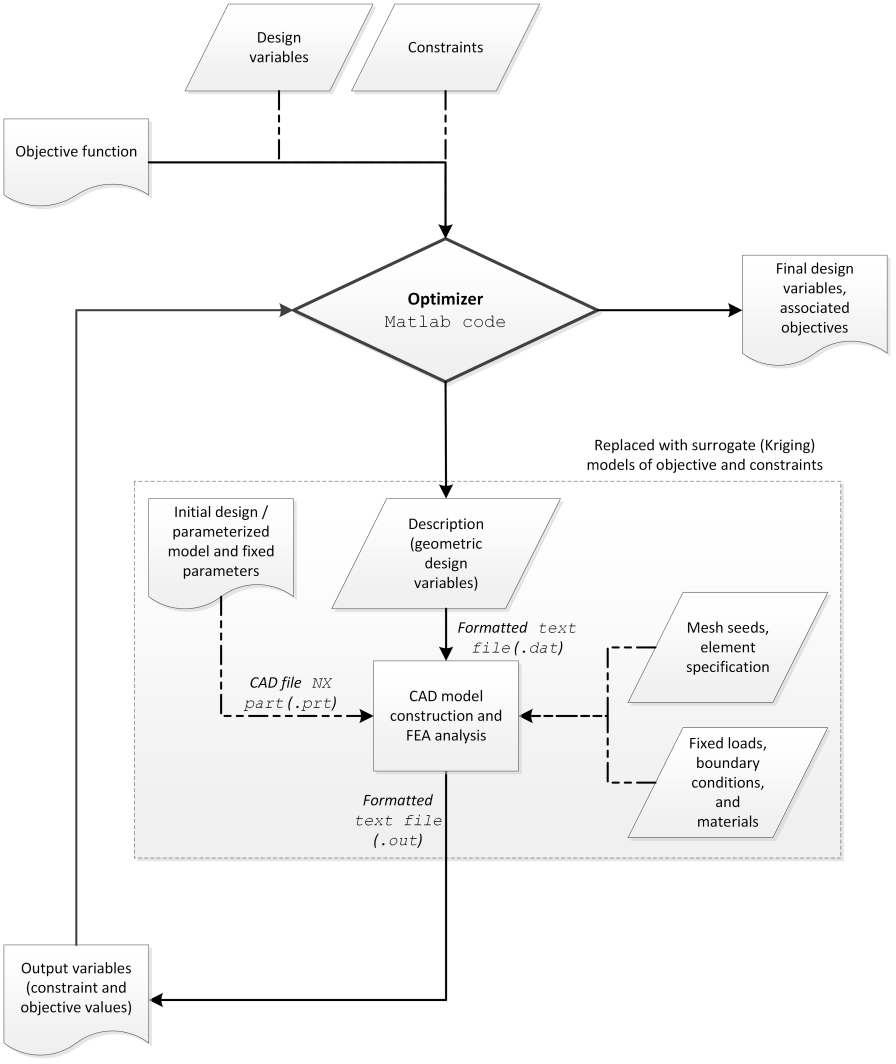


Figure 2.10: Automated dovetail DSO workflow

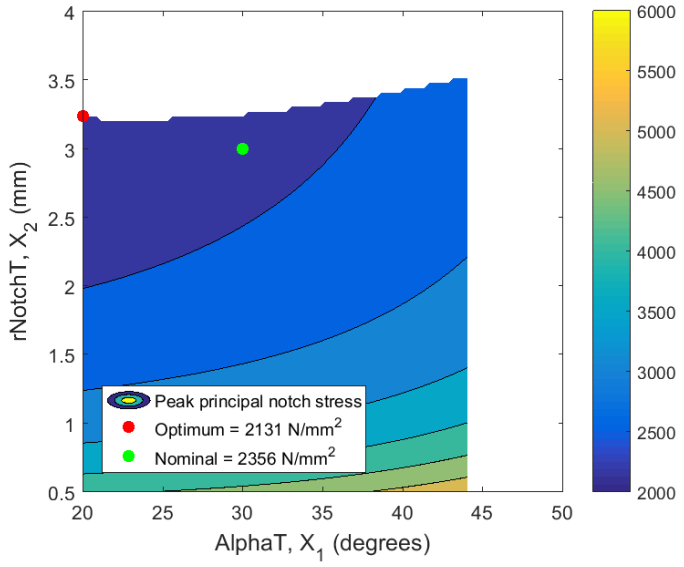


Figure 2.11: Contour plot of notch peak principal stress (N/mm^2) fitted using 225 OLH sample points with constraints and optimum point shown

2.3.2 Surrogate models for design optimization

In engineering design, the objective function is often calculated using some complex code, set of complex codes, or even using a physical experiment. In such cases, and even for relatively cheap code as demonstrated in section 2.3.1, multiple evaluations of the objective can be expensive in terms of time. To combat this problem, it can make sense to approximate the true function as a fast-running analytical model, such as those presented in section 2.2, prior to using any of the routines discussed above to search for optimal designs [78]. In terms of optimization such models not only increase the speed of the optimization such that a previously impracticable problem may be made possible, but they open up the opportunity to use some previously inaccessible methods due to the ability to directly calculate gradient information[4]. However, the fitting of such models does not come without its own associated cost (of function evaluation at sample points and of the optimization involved with fitting the model), and the accuracy of the model should be understood and taken into account. It is prudent to re-evaluate the true function at the chosen optimum and compare it to the predicted value, for example.

Modelling in this way can also be used for constrained optimization; it is possible to approximate the constraints themselves using surrogate models. One such approach is that of *classifiers* using *support vector machines* to approximate the hyperplane that separates classes [119, 97, 248]. If one is already using a model to approximate a continuous objective function then it might make some sense, however, to use a similar techniques to model the constraint function.

To illustrate the benefits of a meta-model approach, we build a model of the two-dimensional dovetail joint (section 2.3.1) objective function. Using only 20 OLH samples ($10 \times d$, where d is the number of dimensions), a Kriging-based surrogate model can be used to expedite the search, resulting in an almost identical optimal design point as that reached with 57 evaluations and a direct downhill search of the true function, figures 2.12 and 2.13. The area of the design space immediately surrounding the optimum point will now require further investigation, but this illustrates the utility supplied by emulators, particularly for visualizing a design space: using the surrogate, the model of the design space shows the designer that there is a tendency towards a single minima and, as such, a local downhill search of the space can be seen to be suitable. A direct downhill search, with no such response surface to observe, would not have provided this assurance.

Here, only 20 runs of the full simulation have been needed. The optimum value is shown to be at AlphaT = 20°, and rNotchT = 3.2430 mm, which is accurate to three decimal places, although the surrogate predicted principal stress at this point is slightly lower.

Surrogate approaches can also be usefully combined directly with the optimization process; an efficient iterative process that searches for an optimum whilst improving the

[4] Complex and often expensive calculation of the gradients of the original function is then not required but could, if available, be used to enhance the surrogate model.

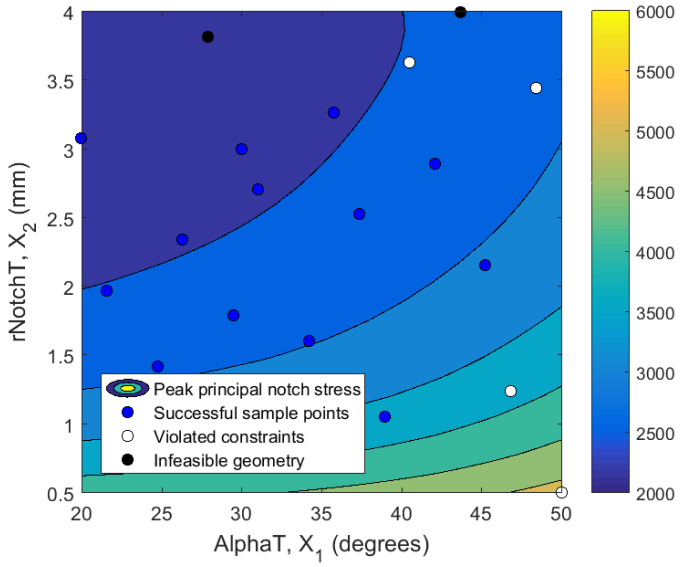


Figure 2.12: Kriging based contour plot of notch peak principal stress (N/mm^2) fitted using 20 OLH sample points

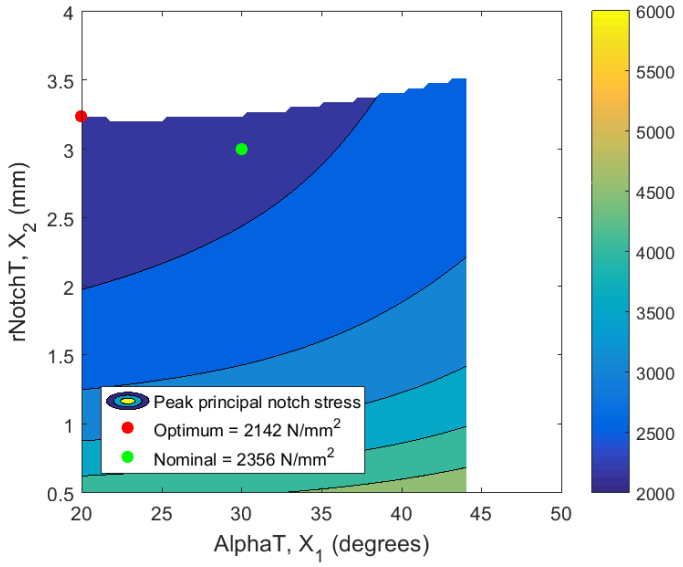


Figure 2.13: Kriging based contour plot of notch peak principal stress (N/mm^2) fitted using 20 OLH sample points with constraints and optimum point shown

model being searched in only the most promising regions can be utilized in a meta-modelling framework. Jones [124] explains how a probabilistic Gaussian process based approach can be refined by using the *probability of improvement* to select additional evaluation points. He also explains how *expected improvement* can be used as an alternative criterion. This approach calculates the expected magnitude of improvement at a given point.

This method uses *infill* criteria to determine the best location for model refinement

within an optimization routine. If the optimization is constrained then it is possible to further extend this probabilistic infill criteria to include the probability of the new point being feasible by simply multiplying it by either the probability of improvement or the expected improvement. In the latter case this is described as the *constrained expected improvement*.

2.3.3 Multi-objective optimization (MOO)

In reality, design problems with only a single objective are unlikely. Even where a single performance measure is paramount to design, it is probable that an additional cost measure will need to be considered [202]. Multiple objectives need not be conflicting, but when they are (for example: strength and weight are almost always conflicting), the problem becomes one of a multi-objective nature.

The simplest means of dealing with such a problem is to create a single aggregated objective function with pre-defined weightings assigned to each goal. Using this approach the optimization formulation (2.3.1) is augmented such that we aim to minimize

$$\sum_{i=1}^{m} w_m f_m(\mathbf{x}), \tag{2.26}$$

where $\mathbf{x} \in \mathbb{R}^d$ and $\sum_{i=1}^{m} w_m = 1$ and there are m goals. The main difficulty with this approach is the subjective nature with which weightings are likely to be chosen and the likelihood of high-performing radical designs being missed [202]. A variation on this theme that still cannot guarantee a unique solution and requires some subjective input from the designer is the reformulation of the problem defining all but one objective as constraints (the *ϵ-constraint* method [169, 177]). Other *weighted metric methods* minimize some weighted distance of the function from an *ideal solution* (e.g. using the *Manhattan norm* or *Euclidean norm* [177], using the Tchebycheff metric [177]). The ideal solution, therefore, needs to be found by optimizing each individual objective function prior to embarking on the multi-objective search. Another special case of the use of a weighting function is to minimize a *value function* or *utility function* which relates the objectives via *exchange constants* [4]. However, this utility function may be difficult to define, and must be *strongly decreasing*[5] for this approach to be successful [49]. These approaches are generally described as *a priori* methods. Some developments of this type of technique are described as *interactive* where choices of weightings, or other such factors, are made or updated during the optimization process [169].

It is widely acknowledged, therefore, that global searches without the need for assumed weightings or overly subjective or difficult to pose objective function formulations, are preferable for multi-objective optimization problems; these are the *posteriori* methods,

[5] "This means that the preference of the decision maker will increase if the value of an objective function decreases while all other objective values remain unchanged" [177].

where the decisions on trade-offs are made after an optimal set of possible solutions has been arrived at [169]. These searches result in a set of *non-dominated*[6] *Pareto optimal points*[7] and corresponding solutions; a *Pareto optimal set*. We therefore turn to a special group of global optimization techniques that are specifically designed to search for solutions that are globally Pareto optimal (the definitions of Pareto optimality apply to the entire feasible space [162]); *genetic multi-objective algorithms*.

Marler and Arora [162] provide an overview of genetic multi-objective algorithms in the context of engineering design, noting that in "much of the literature on multi-objective genetic algorithms, constraints are not addressed directly. It is assumed that a penalty approach is used to treat constraints". This overview describes "one of the first treatments of multi-objective genetic algorithms": the *vector evaluated genetic algorithm* (VEGA) introduced by Schaffer [246]. VEGA is founded on the idea that the minima of each of the individual objectives are vertices of the Pareto optimal set which makes the assumption that each objective has a unique minimum. The method creates small subsets of a *population*[8] (*sub-population*) within the given *generation*[9], by evaluating one single objective at a time. These sub-populations are then combined to make a new population (in the next generation). To avoid clustering of results around the individual objective minima Schaffer [246] used cross-breeding between sub-populations and selection preferences for non-dominated points [162]. The fundamental difficulties and usefulness of the VEGA approach have been well documented [75, 271]. Srinivas and Deb [271] suggested a new approach based on the ideas of Goldberg [92] for a *non-dominated sorting genetic algorithm* (NSGA), which with further refinement [50] is still one of the most popular today. This scheme uses an approach known as *ranking*, where all of the points are sequentially ranked and temporarily removed from the population according to whether they are non-dominated with respect to all other points. The *fitness* metric for determining which points are passed on to the next generation is based on the rank.

Further developments of this type of algorithm and alternative methods for applying genetic algorithms to multi-objective problems are ubiquitous. We will refrain from describing them all, but make a note of the type that use *tournament selection* introduced by Horn et al. [106] and available in the MATLAB toolbox [167]. In this scheme, pairs of points are 'pitched against each other' in terms of non-dominance with respect to the remaining points, and the 'best' are selected for carrying forward into the next generation. If points are tied, *fitness sharing* [271] is used to select the 'best'.

[6] "A vector of objective functions, $\mathbf{F}(\mathbf{x}^*) \in \mathbf{Z}$, is non-dominated iff there does not exist another vector, $\mathbf{F}(\mathbf{x}) \in \mathbf{Z}$, such that $\mathbf{F}(\mathbf{x}) \leq \mathbf{F}(\mathbf{x}^*)$ with at least one $\mathrm{F}_i(\mathbf{x}) < \mathrm{F}_i(\mathbf{x}^*)$. Otherwise, $\mathbf{F}(\mathbf{x}^*)$ is *dominated*." [162]. $\mathbf{Z}$ is the feasible criterion space. $\mathbf{F}$ is a vector of objective functions. $\mathbf{x}$ is the vector of design variables at a point in the design space.

[7] "A point, $\mathbf{x}^* \in \mathbf{X}$, is Pareto optimal iff there does not exist another point, $\mathbf{x} \in \mathbf{X}$, such that $\mathbf{F}(\mathbf{x}) \leq \mathbf{F}(\mathbf{x}^*)$, and $\mathrm{F}_i(\mathbf{x}) < \mathbf{F}_i(\mathbf{x}^*)$ for at least one function." [162].

[8] "A population represents a group of potential solution points" [162].

[9] "A generation represents an algorithmic iteration" [162].

An example of the application of multi-objective optimization for RDO is given in section 2.5.

2.3.4 Multi-fidelity optimization (MFO)

At this point, the potential for *multi-fidelity* approaches when applied to uncertainty and robustness are worthy of note. A design optimization problem might be developed from a nominal analysis model by parameterizing existing geometry or material properties etc. However, the fidelity of such a model, although sufficient to reflect performance changes on a large scale, might not be sufficient to reflect the effect of changes in the uncertain space (e.g. use of isotropic properties rather than spatially varying ones, or fixed attachment rather than contact algorithms). As such, multi-fidelity-type algorithms may tend to play a part in effectively applying robust optimization. A short review of existing work in this area shows that mostly, although not exclusively, multi-fidelity (or *multi-level*) optimization tends to exploit the probabilistic nature of meta-model fits by building global models around less accurate cheap simulations and refining them in areas of probable improvement using higher fidelity expensive codes. Ng and Willcox [184] use a Monte Carlo (appendix A.5) based approach for multi-fidelity robust design, Forrester et al. [80] use *co-Kriging* for multi-fidelity optimization, and Huang et al. [110] "exploit the results of Kennedy and O'Hagan [132]" with a sequential Kriging methodology. Most approaches are based on a 'two-tier' problem formulation (two types of analysis; low fidelity (LF) and high fidelity (HF)) but there are also suggested frameworks for a larger number of levels, or even dynamically choosing model fidelity during the search process [152]. Multi-fidelity approaches have not been investigated fully in this thesis with respect to their use in robust design, and although potentially applicable to characterizing measurement data, their use has not been investigated for the data sets reviewed here.

In summary, there are a plethora of well-established methods for optimization and choosing the most appropriate, given their individual restrictions requires some thought. We can however, whilst being aware of the alternatives, restrict use of existing algorithms to those that are non-intrusive ('black-box') and suitable for the minimization of highly non-linear functions with non-linear constraints. Searches applied throughout this thesis are therefore typically global in the first instance with refinements sought, if necessary, using a local search starting at the globally obtained minimum. Surrogate models are also used in lieu of expensive simulations to enable optimization with practical expense and to gain some of the other advantages as discussed throughout.

2.4 Uncertainty propagation

Here we discuss means of estimating statistical properties of a 'black-box' (hidden) function, based on a sample of input variables with known statistical properties. The process

is referred to here as UP, although, as discussed previously, it is often termed uncertainty quantification (UQ), a description also used to describe the quantification of input uncertainties. The function (f) is assumed to be a complex computational model, or combination of multiple models, and it is also assumed that there is a possibility that the function is non-linear. These assumptions are necessarily true in the case of physical experiments, but in the case of mathematical computational models one might consider using a raft of approaches that propagate local uncertainty around a design point within the analysis code itself. These so-called intrusive methods rely on modification of the underlying computational modelling code. One might, for example, look to using intrusive methods for the stochastic finite element method (SFEM) [60], by incorporating additional functionality into FEA code. These approaches are not without their own problems with respect to prohibitive cost but, if embedded within commercial code effectively, could increase the practicality of implementing UP. However, many designs (including those used within this thesis) require multiple analysis codes in a workflow, or use codes that have been developed 'in-house', making universally applicable UP sitting within an optimization suite, rather than within the performance calculation, appealing. This thesis therefore, considers only non-intrusive methods.

Uncertainty propagation (UP) could, therefore, be termed 'sampling for direct statistical approximation', which is far from being a new concept. As discussed in chapter 1, the application of statistically rigorous assessment of the effects of uncertainty in engineering design is of increasingly significant interest due to a number of economic and political drivers.

Mathematical representation of uncertainty

Prior to propagating input uncertainty, it must be defined in some way. Usually this is via some mathematical representation that can be propagated to a resulting useful representation of performance uncertainty. This thesis is focused on the use of available data, to which probabilistic definitions can be estimated, or fitted. Probabilistic representation of inputs include:

- probability density functions (pdf's),
- statistical moments (often only mean and variance, but could include skewness, kurtosis etc. if the pdf is of a recognized form),
- *polynomial chaos expansions* (PCEs) (polynomial-type mathematical expansions that approximate pdf's in terms of other pdf's), and

- *p-boxes* constructed on a specified confidence interval [69, 8, 325] (from probability bounds analysis which combines standard interval analysis with classical probability theory to result in a region of probabilty bounded not by individual values but by cumulative distribution functions (cdf's)).

Agarwal et al. [1] asserts "when sufficient data is available, any form of variability can be mathematically represented by probability density functions. There is almost universal agreement on this issue". Other methods have been developed to allow quantitative performance predictions to be obtained if the input data are lacking in completeness, knowledge, or quantity, and in some cases is itself only qualitative (such data are usually drawn from expert opinion). Oberkampf et al. [189] describe different categories of uncertainty representation and pose a set of UP problems with varying levels of detail in the input uncertainties. A summary of the different solutions to these problems arrived at by different research communities with different uncertainty characterizations are described in Ferson et al. [70]. We refrain from extending this work to non-probabilistic methods, which include *fuzzy sets* and *possibility theory* [127].

Despite the statement made by Agarwal et al. [1], the quantification of uncertainty based on measurements is seldom addressed in the literature [13][10]. O'Hagan [190] proposes that the statistics community very much addresses the quantification of the uncertainty in the input vector but that the applied mathematics community does not. In fact, Oladyshkin and Nowak [193] make some 'in-roads' in this area by showing that polynomial chaos expansions based methods can be extended to allow for sparseness in data without the need for pdf inputs, rather only data or histograms. O'Hagan [190] points to *Bayesian inference* to define pdfs. Perhaps one can surmise that the gathering of data is, in general, a rather time-consuming task of less interest to researchers than the mathematics of UP. Or maybe it is the practicalities of data capture that have inhibited researchers in studying the use of measured data sets in UP. Equally, some quantities are difficult, or impossible, to measure. In addition it can be difficult to capture and maintain the effects of cross-correlations. Typically, independent Gaussian normal distributions or simple intervals are assumed despite the fact that many acknowledge that if the input distributions are wrong then the result may be inaccurate.

A note on polynomial chaos expansions for propagating uncertainty

Polynomial chaos expansions (PCEs) have gained recognition recently by the applied mathematics community for their use in propagating uncertainty to predict the distributions of outputs of computer simulations. O'Hagan [190] defines PCEs as "an expansion to construct a random variable with a desired distribution as a function of another given

[10] We see this issue begin to be addressed in the field of operations research (OR) with linear programming [12] and financial portfolio optimization for *stochastic programming* (linear or quadratic programming with uncertainty) [52], and more recently for biomedical engineering [161].

random variable" in his critique of them from a statistician's viewpoint. The expansion is a polynomial series with orthogonal polynomial basis[11]. The resulting truncated PCE (it must be truncated to some predefined order) is an approximation or surrogate of the output distribution.

Formulation of the PCE requires calculation of of the *mode strengths*; the weightings in the PCE formulation. The non-intrusive 'black-box' formulation in which we are interested, determines these weightings either via numerical integration or via some sort of regression model. PCEs, although popular for UP in real-world modelling, have not gained such a reputation in engineering design. They have limitations for non-linear problems, lack of versatility for different inputs once constructed, and inherent error in using truncated functions, as discussed by O'Hagan [190]. Given that the PCE model is made to approximate the output distribution, yet the computational cost is equivalent to building a model of the function itself, it appears to lack many of the advantages of emulators or even direct integral estimation applied to moment estimation, for robust design.

Mathematical methods employed for propagating uncertainty

Once mathematical definitions of the uncertain inputs are available, UP can be achieved using one of two families of methods:

- *Monte Carlo* (MC) (usually employing surrogate models, section 2.3.2), or
- *quadrature* (also known as *stochastic collocation* [98, 321])

based methods. These are presented in detail in appendix A.5 and A.6, respectively. Comparisons of different approaches by Liu et al. [157] and Stern et al. [275] on specific UP problems conclude that meta-models are the "most promising method" and are "more efficient" than the alternatives.

To demonstrate the UP process using MC, let us consider our dovetail design problem from section 2.3.1 and appendix B. The two design variables can be selected freely by the designer but the reality of manufacturing and service is that symmetry from top to bottom may not, in fact, be a reality. At the chosen design point (we shall assume that this is the nominal design), we wish to determine the effect of this uncertainty on the performance parameter (disc notch principal stress). We begin with four uncertain parameters (rNotchT, AlphaT, rNotchB, and AlphaB), which vary independently from

[11] Calculation of the polynomial basis set will result in the Legendre polynomials [216] if the input variable (known as the *germ*) is uniform on [-1,1], the Hermite polynomials [216] if the germ has a standard normal distribution, and the Laguerre polynomials [216] if the germ is an exponential random variable on $[0, \infty)$.

one another. They vary in the following ranges:

$$\begin{aligned} -3.5^\circ \leq \quad & \Delta\text{AlphaT} \quad \leq 3.5^\circ \\ -3.5^\circ \leq \quad & \Delta\text{AlphaB} \quad \leq 3.5^\circ \\ -0.35\,\text{mm} \leq \quad & \Delta\text{rNotchT} \quad \leq 0.35\,\text{mm} \\ -0.35\,\text{mm} \leq \quad & \Delta\text{rNotchB} \quad \leq 0.35\,\text{mm}, \end{aligned} \tag{2.27}$$

where Δ is the change from nominal setting of each parameter. In order to propagate these uncertainties using Monte Carlo sampling or related methods, we require a model of the distribution of each uncertain parameter and thousands of evaluations of the performance at randomly selected values of the parameters. This is possible, but time-consuming, even for this relatively fast-running simulation. Using quadrature based methods one can expedite the process but hundreds of evaluations are still likely to be required and the process has other limitations, appendix A.6. Therefore, the most practical solution is to create a four-dimensional response surface model over the uncertain range and evaluate this cheap surrogate in preference to the full simulation.

Instead of a direct random sample of thousands of points over the four-dimensional uncertain space, an OLH sample is created in the same way as for DSO in section 2.3.2. This time, however, the sample exists in the smaller uncertain region and a $10d$ sample is 40 points space-filling over $d = 4$ dimensions. A regressing Kriging model is fitted over the four-dimensional space. To visualize the resulting response, the model has been plotted as six sets of two-dimensional surfaces showing the variation across the two dimensions whilst maintaining fixed values of the other two dimensions at the nominal, figure 2.14. The black points show the uncertain variable values of the forty sample points in each pair of dimensions. Even with only four dimensions visualization is difficult but this gives us an indication of the non-linear effects resulting from the lack of symmetry in the design.

The Kriging model is assessed by evaluating the resulting noise filtered using the regression model. The standard deviation of this noise over 41 points (the sample and the nominal) is 0.1886 units of principal stress (MPa), and the range 0.9875 MPa, so the level of regression is extremely small. A leave-one-out cross-validation, as discussed later in section 2.8, is then performed. If the model is refitted using the same process, which involved a genetic algorithm (section A.4.1), followed by a local downhill search (section 2.3.1), then the percentage error is 0.72%. If the refit only repeats the downhill search then the error is predicted to be 0.70%. Re-calculating the likelihood only, also results in a predicted percentage error of 0.70%. This model can now be used to propagate the uncertainty. The full process is shown in figure 2.15, where the model is built first (1), followed by UP using the techniques described in this section (2).

Consider that the variation in the angles ΔAlphaT and ΔAlphaB are distributed normally with mean $\mu_A = 0$ and standard deviation $\sigma_A = 1.0$. The variation in notch

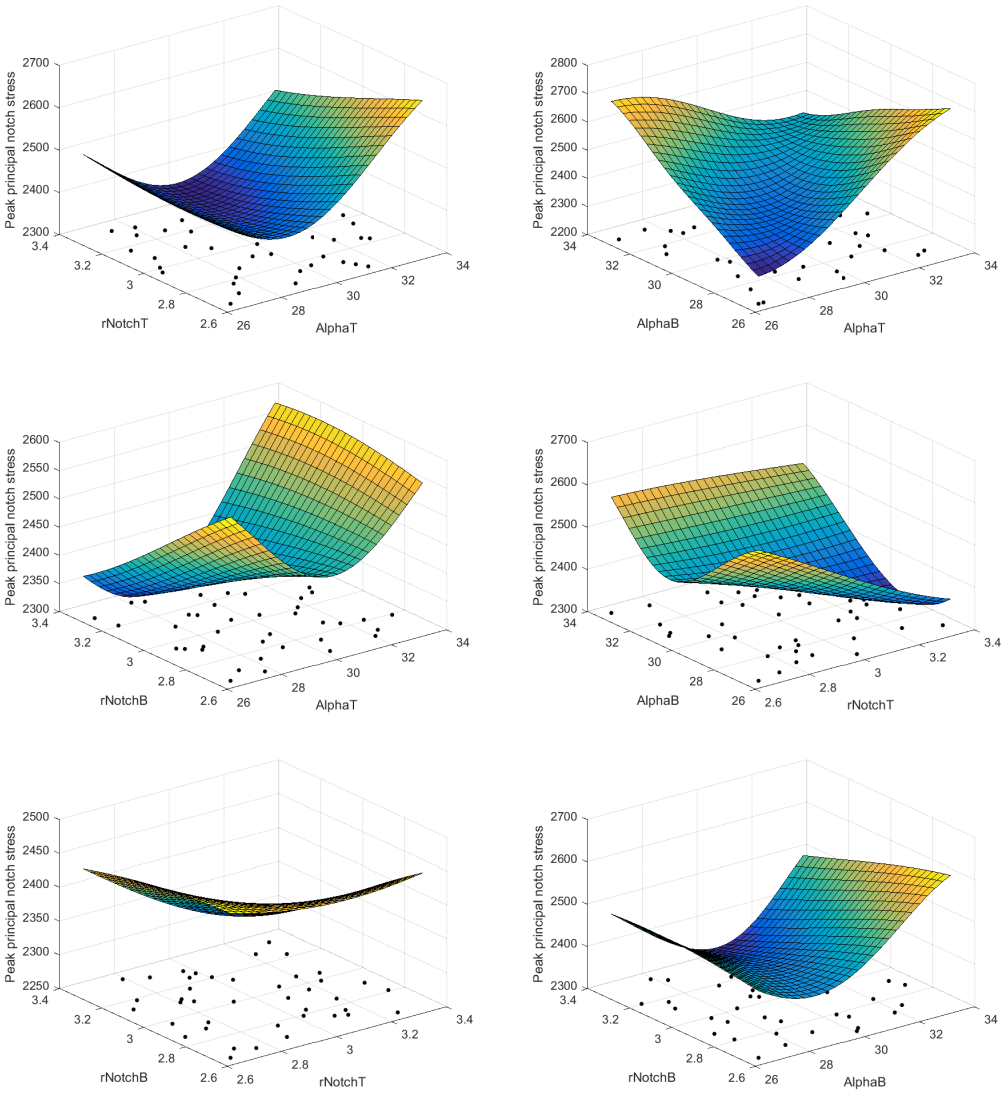


Figure 2.14: Response surfaces showing Kriging model values throughout the uncertain space for dovetail example uncertainty propagation problem

radii ΔrNotchT and ΔrNotchB are distributed normally with mean $\mu_r = 0$ and standard deviation $\sigma_r = 0.1$. The Kriging model bounds truncate any sampling to $3.5 \times \sigma$. The input distributions are shown in figure 2.16. Using a MC sample of 1000 results in the output performance distribution shown in figure 2.17. The results clearly show the detrimental effects that could be seen if uncertainty is ignored in design; in this 'toy' problem, not only is the spread seen in the peak stresses significant (in the region of 20% of the nominal peak stress), but the mode, mean, and median peak principal stresses all lie above the nominal when uncertainty is considered. In this case, although the performance distribution is bell-shaped, it is peaky with long tails.

We note here that, without exception, for the UP methods discussed here to provide results that are not misleading, they reply upon correctly and accurately quantified

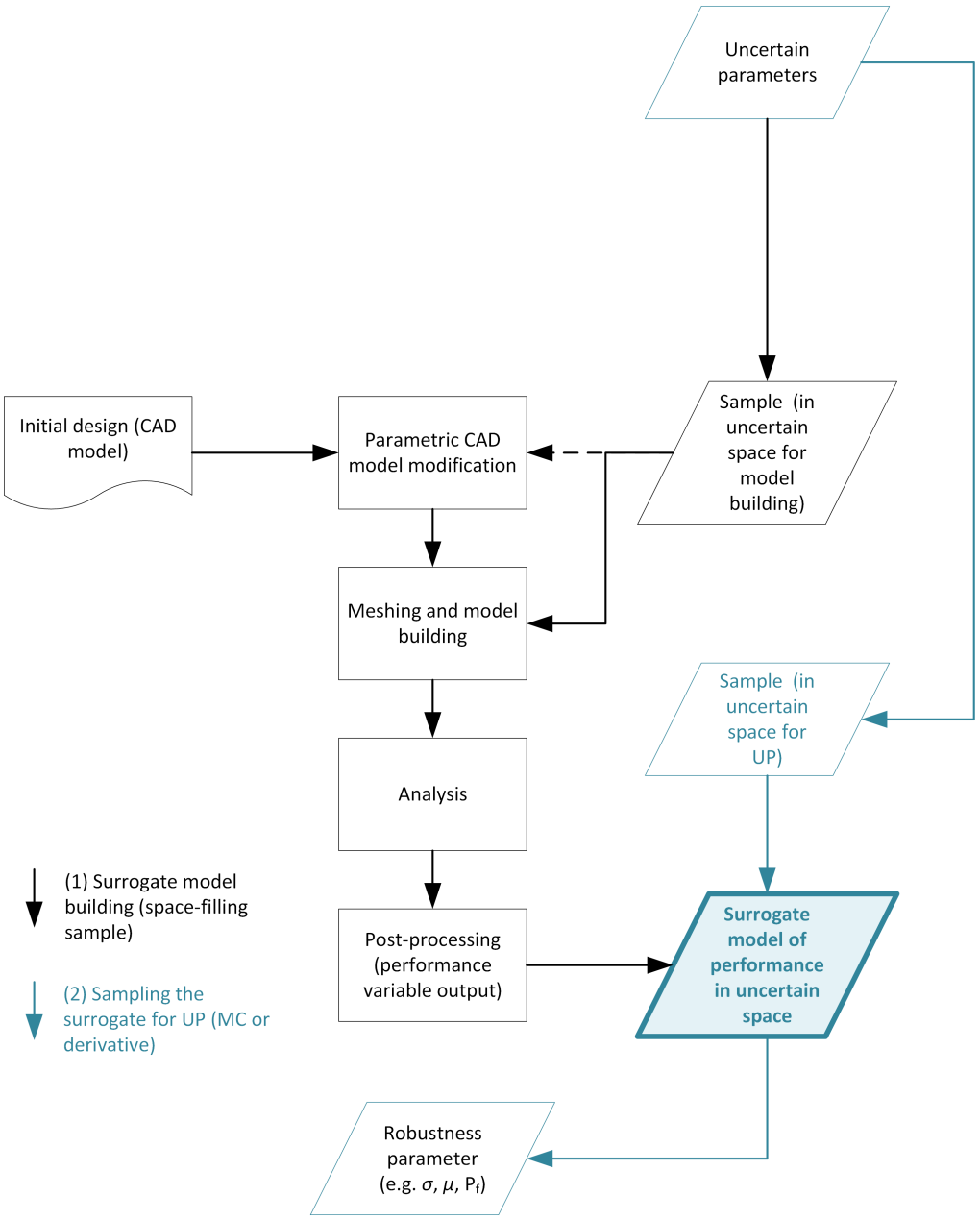


Figure 2.15: Uncertainty propagation using a surrogate model

input uncertainties, i.e. the correct distributions must be attributed to the inputs [189] and any correlations accounted for, section A.8.4.

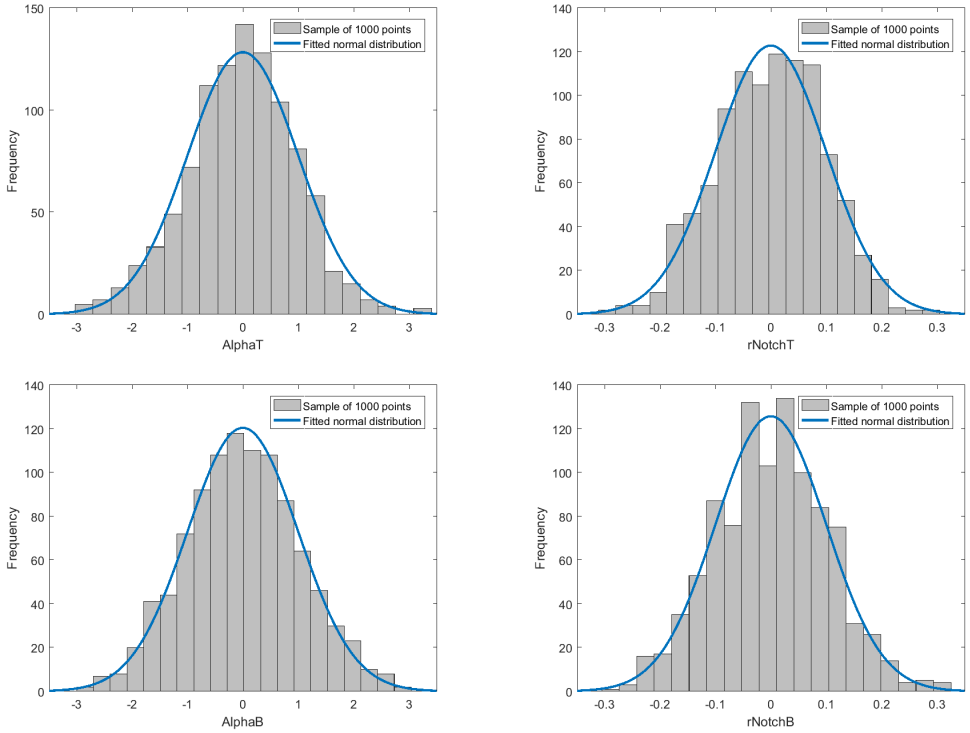


Figure 2.16: Dovetail example uncertainty propagation problem input parameter distribution functions and sample histograms

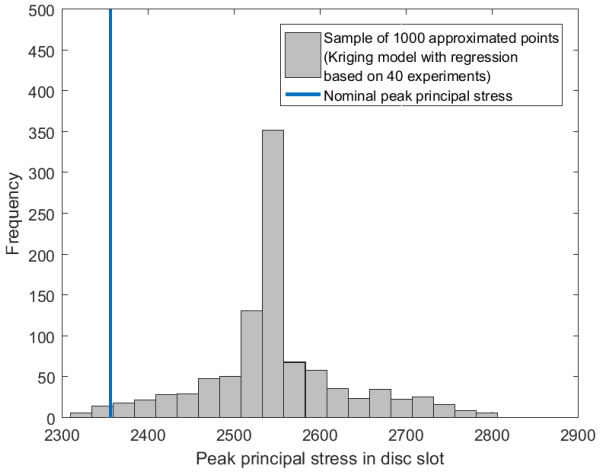


Figure 2.17: Dovetail example uncertainty propagation problem output distribution of performance showing nominal performance

2.5 Robust design optimization

It is widely acknowledged that the founder of the robust design approach is Genichi Taguchi, who pioneered the use of probabilistic approaches for quality engineering in Japan in the 1950s. He faced the challenge of improving product quality after World War II despite poor-quality raw materials and manufacturing equipment, and a lack of skilled

workers introducing an increased level of variability into the manufacturing process [206]. The fundamental principle of Taguchi's robust design is the minimization of variance in product performance due to *noise factors* (uncontrollable parameters) by appropriate selection of *control factors* (design variables) [33, 206]. Taguchi was most interested in 'loss' expressed as a *signal-to-noise ratio* (SNR) as the objective of minimization, i.e. a quantifiable value of the cost to society (other than cost caused by intrinsic product function), that can be reduced by parameter design [281]. Taguchi wasn't the first person ever to consider the reduction of variation to be important; in a panel discussion by Nair et al. [183] Lorenzen recalls hearing of agricultural studies in the early 1940s with the aim of producing "products whose yield was robust to different growing conditions" [234]. In fact, the Rothamsted Classical Experiments creating a soil sample archive began in 1843 [160]. Taguchi was, however, the first to formalize the approach and successfully implement it in product manufacture.

After its success in Japan, the methodology that Taguchi popularized took until the 1980s to be applied effectively in the USA and the wider world. His philosophy, often known as robust parameter design (RPD), uses experimental design, or design of experiments (DoE), by way of orthogonal array samples [103], for selecting which sets of control variables should be chosen for experiment given a limited number of possible product tests. The results of these experiments are scrutinized to firstly determine the set of control factors resulting in least variability in performance, then using a *scale factor* (one of the control factors) to shift the mean close to the desired value, and running a further confirmation experiment.

The Taguchi approach is far reaching; "developed to improve product quality and reliability in industrial engineering" [201]. In 2006 Park et al. [201] gave a detailed overview of robust design as developed from the Taguchi method and stated that "recently, this technology has been expanded to various design areas". When applied to design, the function is a model of a critical performance parameter, and is typically a computer model. For example, if we consider the effect of a single variable on performance, we can illustrate the concept as a choice of alternative designs (A and B) in figure 2.18 [206]. In this plot, assuming we wish to minimize the performance parameter, there is a choice to be made between the nominally best performing design and the least variation design. Of course, this trade-off may not be necessary, depending on the shape of the performance curve. As Park et al. [201] describe, an expansion of methodologies for robust design has ensued since the 1950s, in particular, due to a number of difficulties associated with the method when applied to design:

- a scale factor might not, in fact, exist,

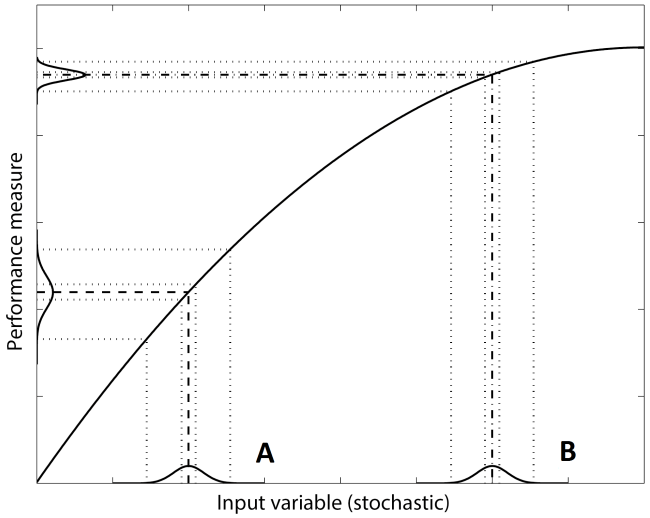


Figure 2.18: The essence of robust design as illustrated using a choice of two single stochastic (normally distributed random) design variables A and B [206]

- the SNR is only effective as an objective when factors affecting the mean and variance are separated, eliminating *confounding*[12] effects,
- the number of experiments required may be prohibitively costly, although this difficulty is by no means limited to Taguchi methods,
- the approach is only suitable for a single characteristic function,

and, possibly most prohibitively,

- the Taguchi method is not capable of including constraints, unless a limited number of candidate designs are proposed.

In the early 1990s the momentum shifted to the use of well established optimization routines for robust optimization to allow for ease of the use of continuous variables and incorporation of constraints. Beyer and Sendhoff [14] review the methods in RDO from a broader perspective, including the simultaneously developing approaches of the operations research (OR) community.

Robust design optimization, based on the definition by Park et al. [201], is formulated as the multi-objective problem:

find

$$\mathbf{v} \in \mathbb{R}^d$$

[12]Confounding is the misinterpretation of cause and effect (main effects or interactions) between variables due to the failure to account for an influential, confounding, factor.

to minimise a combination of

$$\mu_f(\mathbf{v}, \mathbf{p})$$

and

$$\sigma_f(\mathbf{v}, \mathbf{p}),$$

subject to the constraints

$$g_j\left(\mathbf{v} + \mathbf{z}_\mathrm{v}, \mathbf{p} + \mathbf{z}_\mathrm{p}\right) \leq 0$$

where $j = 1, ..., c$, and $\mathbf{v}_\mathrm{L} \leq \mathbf{v} \leq \mathbf{v}_\mathrm{U}$.

$\mathbf{p}$ are 'uncertain' or 'uncontrollable' parameters; they contain irreducible uncertainty (often described as 'noise') which may be distributed normally or otherwise, and are the same for all designs. $\mathbf{v}$ are design variables[13], which can also be uncertain. $\mathbf{z}$ is a vector of changes in either $\mathbf{v}$ or $\mathbf{p}$ due to uncertainty. There are d design variables and c constraints. μ_f and σ_f are the mean and standard deviation of the performance function f[14], respectively, and are given by the well-documented definitions (for example [200]):

$$\mu_f = \mathrm{E}[f(\mathbf{u})] = \int_{-\infty}^{+\infty} f(\mathbf{u}) P(\mathbf{u}) \mathrm{d}\mathbf{u}, \tag{2.28}$$

and

$${\sigma_f}^2 = \mathrm{E}[(f(\mathbf{u}) - \mu_f)^2] = \int_{-\infty}^{+\infty} [f(\mathbf{u}) - \mu_f]^2 P(\mathbf{u}) \mathrm{d}\mathbf{u}, \tag{2.29}$$

where P is the *joint probability density function* of the variables $\mathbf{u}$ which is a combined vector of design $\mathbf{v}$ and $\mathbf{p}$. One might equally formulate the problem to minimize some alternative measure of distribution spread, such as the size of a given confidence interval.

The evaluation of this mean and standard deviation throughout the design space (at all possible values of f, or at sufficient sample points for optimization) is not an insignificant task. There is no closed-form analytical solution of the minimization problem, in general, for problems of practical interest. In certain circumstances, "when interactions of the design variables and the design parameters are negligible and deviations of them are small" [201], the mean (μ_f) can be approximated as a function of the individual variable and parameter means, and the variance (${\sigma_f}^2$) as a function of the first and second derivatives of the model function with respect to the variables and parameters. The first derivatives, or *gradients*, are *sensitivities* of the function to the variables and the second derivatives are *second order sensitivities*. Figure 2.19 shows how a sensitive design, 'C', although deterministically 'better', is less robust than designs 'A' and 'B'; the same variation in v results in a potentially much higher value and bigger range in f (it is more

[13] It is typical to use $\mathbf{x}$, as is the case for the remainder of this chapter, but to avoid confusion with $\mathbf{x}$ from a measurement set in later chapters, we refer to the design variable vector as $\mathbf{v}$.

[14] f can sometimes confusingly be referred to as the 'objective function'. We shall refrain from using this description in this thesis, using the term objective function only to refer to the function which is the objective of an optimization, which varies depending on the construction of the optimization problem.

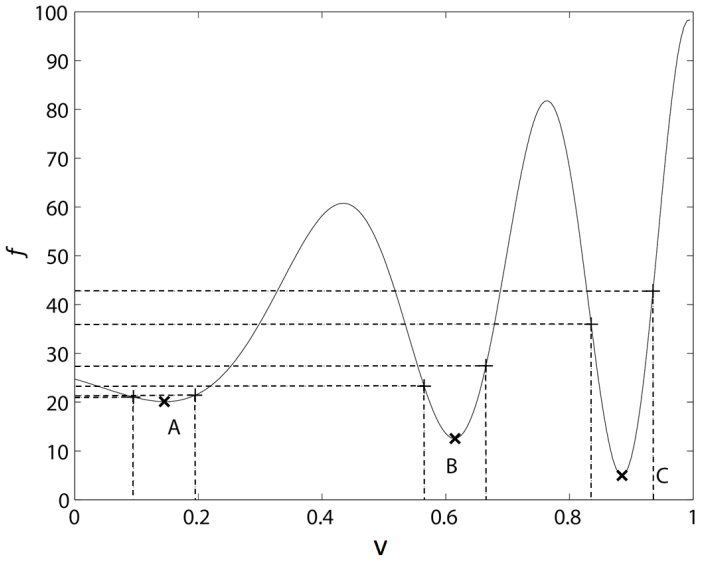


Figure 2.19: 'Robustness' as minimum sensitivity illustrated using a single design variable (v); the performance (f) of design 'A' varies less than that of designs 'B' and 'C' for the same change in v, i.e. it is less sensitive to uncertainty in v

sensitive). Park et al. [201] does note, however, that while "calculation of the second-order sensitivity is viable for mathematical or small scale engineering problems, but is difficult to apply to large-scale problems".

Han and Kwak [100] calculate second order sensitivities directly from the equations of motion for a lumped mass system to "minimize the maximum sensitivity" where the method "requires no statistical information on the uncertainties". This is perhaps unsurprising as the deviations are similar in magnitude and small relative to the design space, and are independent. Putko et al. [218] and Taylor III et al. [285] use sensitivity derivatives for robust design in CFD, using *automatic forward (direct)* and *reverse (adjoint)*[15] differentiation procedures.

The complexities involved with direct differentiation of the output of complex computer codes can be eliminated by the use of RSMs: as described by Chen et al. [34] and Chen et al. [33]. They propose the use of quadratic RSMs as a surrogate for the expensive computer model or physical experiment. The derivatives are relatively easily calculated from the mathematical functions. Park et al. [201] suggest that the use of RSMs can be risky because sensitive regions of the design may have been 'smoothed out' by the RSM. This is arguable, particularly in recent years where RSMs have become more and more sophisticated [79, 219]. In particular, with the common use of radial basis functions [22, 21], and Kriging models, first applied to engineering design by Sacks et al. [242] (section 2.2). The sensitivity method is not appropriate however, if the uncertainties

[15]This method is typically intrusive; requires modification of the underlying analysis code, but black-box reverse-mode application of automatic differentiation is also possible with increased CPU time and computer memory [285].

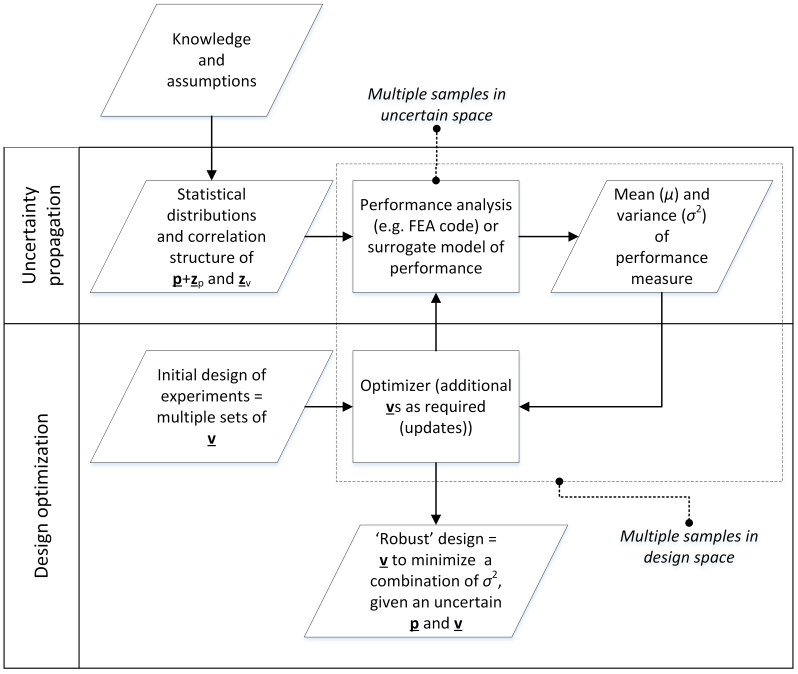


Figure 2.20: A typical robust design optimization (RDO) workflow

have asymmetric or differing distributions as only the mean and variance of the input parameters are used in the output estimations[16].

For problems where the response is non-linear in the region of uncertainty, where there are interactions or correlations between variables, where the uncertainty is large in magnitude, or where uncertainty is distributed differently between variables, or in an unusual or skewed manner, it may be preferable to propagate uncertainty locally in order to calculate means and variances. This process of RDO is illustrated in figure 2.20.

This approach requires much more knowledge in the local region of uncertainty where the mean and variance are to be calculated (the *uncertainty space*, or sometimes referred to as the *stochastic space*). There are multiple uncertain spaces, analogous to *design points* in deterministic optimization. Inherently, more information requires more cost. Where derivation of sensitivities can be expensive in terms of their calculation, propagated uncertainties can be expensive in terms of the multiple additional evaluations of the computer model. This multiple space approach within the overall design space, sometimes referred to as *optimization space*, has been adopted by Roos and Hoffmann [236] in a sequential manner by evaluating uncertainty in areas of 'best' deterministic designs and iterating around them.

[16]It is also noted that the distributions of uncertainty in variables could also be different depending upon the location in the design space. Such variation is difficult, if not impossible, to determine without a great deal of experimentation. In the case of geometric uncertainty, even accurate determination at a nominal design point (as is the focus of this thesis) will be an improvement on current practices. The approach discussed here would also be appropriate if further knowledge across the design space was available

Here we take the whole design space of the 'bird function' illustration from section 2.2.1 to demonstrate the RDO process as a multi-objective design problem. f is now constructed as a function of the design variable vector $\mathbf{x}$ (referred to in the general case above as $\mathbf{v}$), and two constant parameters $\mathbf{p}$ ($\mathrm{p}_1 = -10$, $\mathrm{p}_2 = -10$):

$$f(x_1, x_2) = \sin(\mathrm{p}_1 x_1) e^{(1-\cos(\mathrm{p}_2 x_2))^2} + \cos(\mathrm{p}_1 x_1) e^{(1-\sin(\mathrm{p}_2 x_2))^2} + (\mathrm{p}_1 x_1 - \mathrm{p}_2 x_2)^2 \quad (2.30)$$

where x_1 (or the variation in x_1, z_{x_1}) and $\mathbf{p}$ are uncertain:

$$\begin{aligned} z_{x_1} &\sim \mathrm{N}(0,\, 0.03) \\ \mathrm{p}_1 &\sim \mathrm{N}(-10,\, 1) \\ \mathrm{p}_2 &\sim \mathrm{N}(-10,\, 2). \end{aligned} \quad (2.31)$$

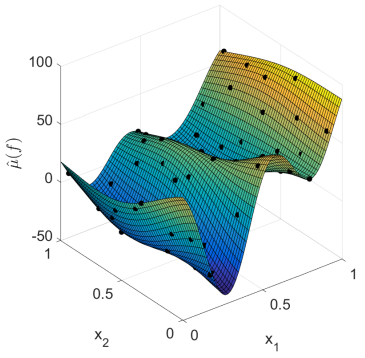
An OLH sample of the design space is made. At each of the $n = 60$ sample points the mean and standard deviation of f are evaluated by means of UP. The UP approach taken is Monte Carlo sampling; evaluation of a sample mean and standard deviation using 1000 randomly drawn uncertain values from the given distributions at each design point. Regressing Kriging models are fitted to give a prediction of the mean function $\hat{\mu}(f)$, and the standard deviation function $\hat{\sigma}(f)$, figures 2.21a and 2.21b respectively. The regressing models are used because Monte Carlo sampling results in values containing some noise.

Using a single objective global optimization (with the PSO algorithm, appendix A.4.2) results in two different optimal designs, figures 2.21c and 2.21d. A front of Pareto-optimal non-dominated points can be plotted from either the existing sample points, figure 2.22a, or using a larger sample of the Kriging predictor, 2.22b. Further investigation could be employed to obtain a more accurate optimal front with, for example, multi-objective expected improvement and a multi-objective optimizer such as NSGA-II to select a suitable Pareto-optimal design, depending upon the problem.

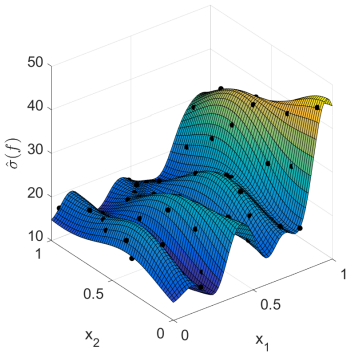
A second example uses different definitions of the uncertain variables, which are not all centred on the nominal value:

$$\begin{aligned} z_{x_1} &\sim \mathrm{N}(0,\, 0.03) \\ \mathrm{p}_1 &\sim \mathrm{S}(2,\, 0.75,\, 1,\, 0) \\ \mathrm{p}_2 &\sim \mathrm{U}[-3,\, 1]. \end{aligned} \quad (2.32)$$

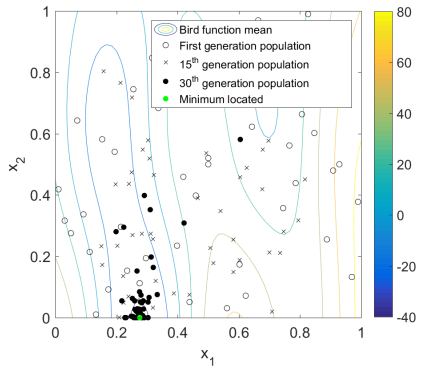
S refers to a stable distribution, section A.8.2. The single-objective optimum designs are substantially further from each other, figure 2.23, and the Pareto-front is discontinuous, figure 2.24. Such a design would warrant further investigation, but this example serves to illustrate how combinations of uncertainties in already complex problems can result in designs of sub-optimal variance in performance. Studies that include uncertainties can



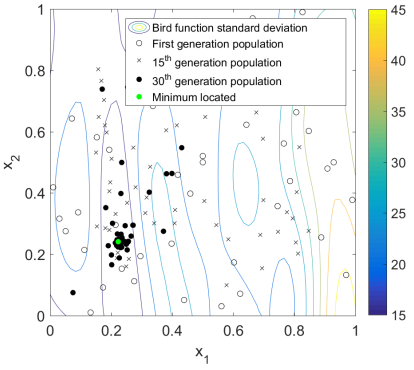
(a) Fitted mean function



(b) Fitted standard deviation function



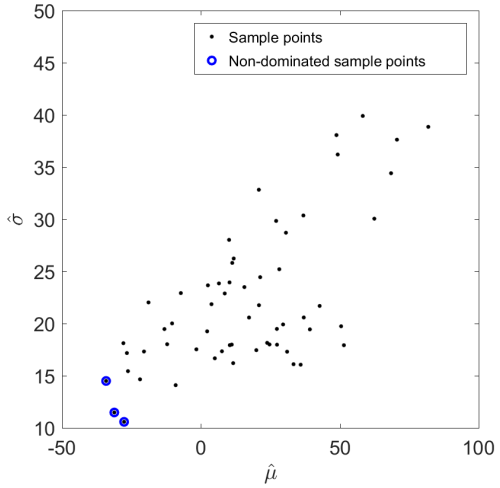
(c) Global optimum for minimum mean using particle swarm optimization (PSO)



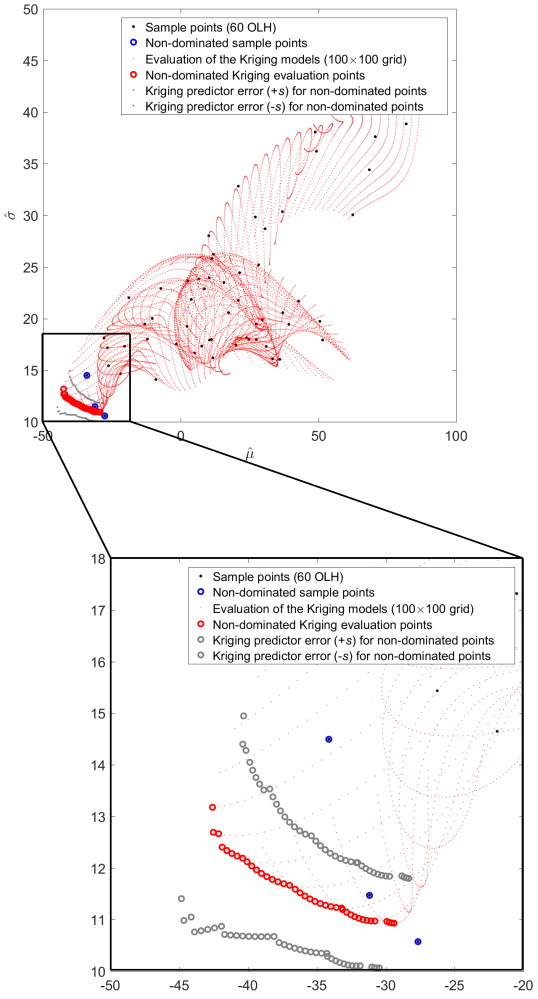
(d) Global optimum for minimum standard deviation using particle swarm optimization (PSO)

Figure 2.21: Example two-variable robust design optimization (RDO) using a function with one uncertain design variable and two additional uncertain parameters

further inform the design decision making process and improve robustness of performance under irreducible uncertainty.

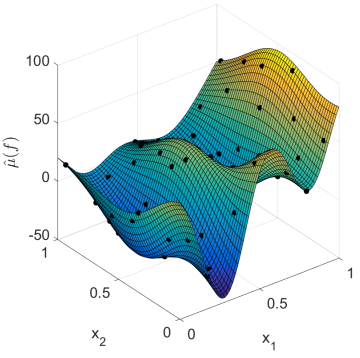


(a) Design space sample showing non-dominated points

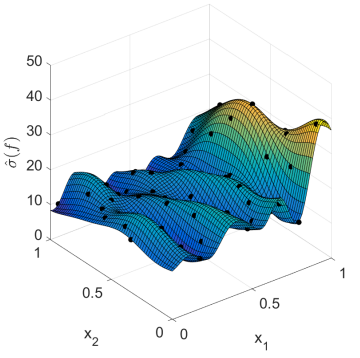


(b) Kriging predictor full-factorial sample showing non-dominated points

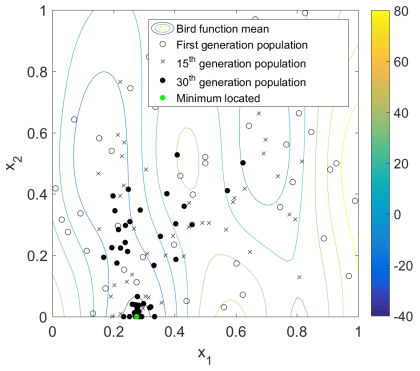
Figure 2.22: Example two-variable multi-objective robust design space search, showing Pareto-optimal robust design points



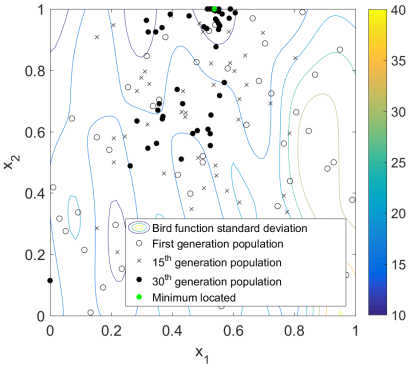
(a) Fitted mean function



(b) Fitted standard deviation function

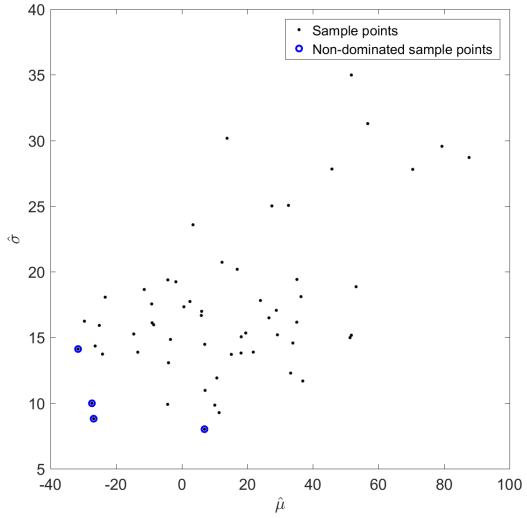


(c) Global optimum for minimum mean using particle swarm optimization (PSO)

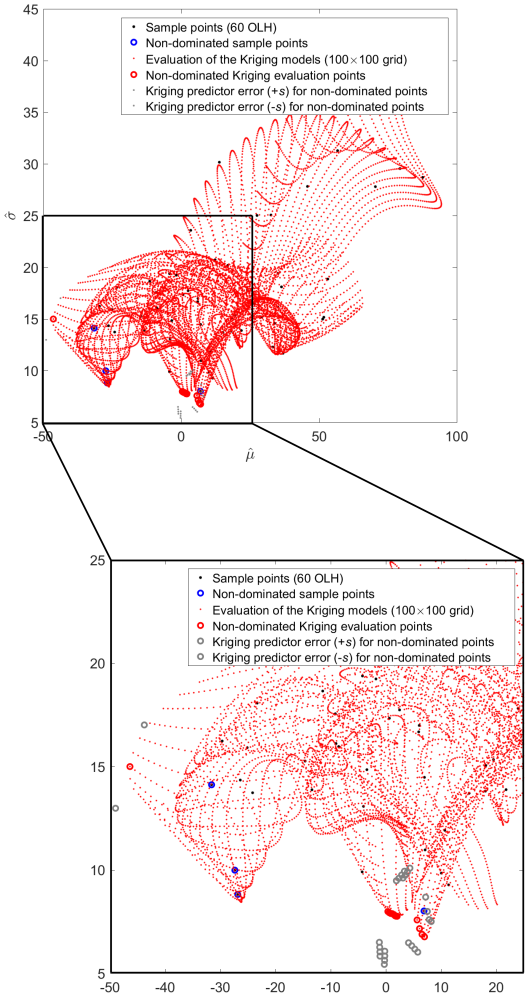


(d) Global optimum for minimum standard deviation using particle swarm optimization (PSO)

Figure 2.23: Second example two-variable robust design optimization (RDO) using a function with one uncertain design variable and two additional uncertain parameters



(a) Design space sample showing non-dominated points



(b) Kriging predictor full-factorial sample showing non-dominated points

Figure 2.24: Second example two-variable multi-objective robust design space search, showing Pareto-optimal robust design points

2.6 Reliability-based design optimization

Although this thesis was intended to develop methods for facilitating black-box robust design optimization, it is also relevant to the assessment of reliability. It is therefore pertinent to be clear here about the distinction between design for robustness and reliability-based design, or robust design optimization (RDO) and reliability-based design optimization (RBDO), and to briefly introduce the latter. The approaches used for both are similar, and can even be the same (e.g. Monte Carlo sampling), but the goal of each is fundamentally different.

Reliability, described as 'quality', like robust design, was popularized during the 1980s, specifically in January 1987, when the chief executive officer of the Motorola company in the USA launched "The Six Sigma Quality Program". It had been developed by Motorola employees Bill Smith and Mikel Harry, as a means of eliminating defects but was now to be used as the central quality control process throughout the entire business. The quality focus of the company was a reaction to observing the greatly improved quality of television sets manufactured after a Motorola factory had been taken over by the Japanese [217]. General Electrics's chief executive officer, Jack Welch then adopted 'Six Sigma' as a central business strategy in 1995, from where the approach became well known globally.

The central goal of 'Six Sigma' is a reduction in defective parts. 'Sigma' refers to the standard deviation of the measure of performance with levels of conformance; lower and upper specification limits, or tolerances. 'Six Sigma' is a "quality improvement process and philosophy" that strives towards designs with six standard deviations of performance from the mean falling within the conformance limits, i.e. 99.9999998% of parts conforming, or 0.002 defects per million [138]. This assessment of quality in terms of 'defects per million' is a measure of reliability and can also be expressed as a probability of failure, P_f.

Applied to engineering design, we consider the historically used factor of safety defined by Shigley and Mischke [255] as:

$$\text{Factor of safety} = \frac{\mathrm{B}}{\mathrm{L}}.$$

where B is the "actual breaking strength in application" and L is the applied "load".

Here, and in most texts, it refers to material strength where load to failure is the critical performance measure. In reality the uncertainties in the mathematical model, and in the service conditions (environment), lead to a statistical distribution of the performance measure and hence the factor of safety. As Shigley and Mischke [255] state, "a major danger in the use of ... the factor of safety is to believe that if" it "is greater than 1, the product having such a factor is safe". Figure 2.25 shows clearly that a factor of safety

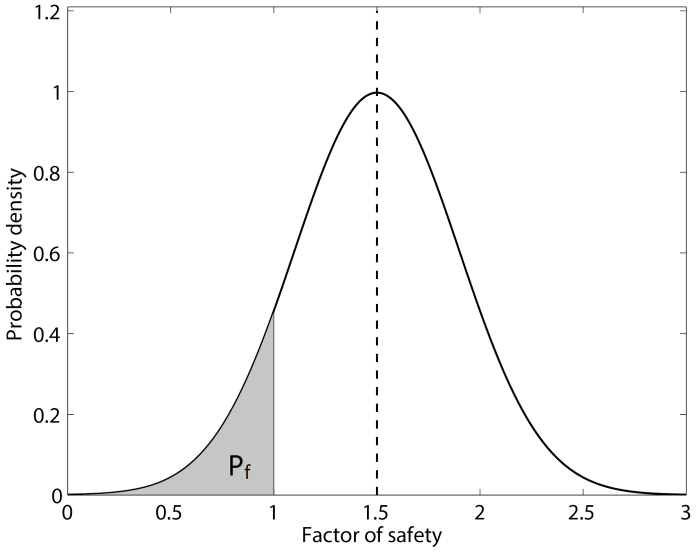


Figure 2.25: The distribution of the factor of safety and resulting probability of failure (P_f) in the presence of uncertainty

exceeding 1 can potentially lead to a large probability of failure (i.e. region below 1 under the curve), depending on the true statistical distribution.

It is the goal of design for reliability to minimize P_f, in contrast with robust design where the goal is to minimize the variance (σ^2) of the performance. In both cases, one might also wish to minimize or maximize the mean performance, in which case the optimization becomes one of multiple objectives, chapter 2.3.3.

Koch et al. [138] illustrated the use of "robust optimization" applied to design for 'Six Sigma' in their 2004 paper, and these methods are now readily available in some commercial software such as Isight^{TM} [46].

This probabilistic approach, where the value of interest is the probability of failure, is usually referred to as *reliability analysis* and many relevant and state of the art uncertainty propagation techniques originate from this subject area. They can be based on the definition of a *limit state function* [279]. Some of these techniques may also be of particular use if, for example, the most likely 'best' designs lie close to constraint boundaries. In this case one may need to estimate accurately both the expected level of variation and the probability of exceeding the critical constraint.

Reliability-based design can be more difficult because it requires an accurate propagation of uncertainty in the tails of the statistical distributions where they are often asymptotic. Sampling, section 2.1, modelling, section 2.2, and propagation, section 2.4, can all introduce inaccuracy 'in the tails'.

2.7 Fitting geometric curve and surface models to measured points

Section 2.2 presents a popular approach for fitting a mathematical model based on a set of data points: a response surface. Such a surface exists in multiple dimensions: *hyperspace*. In the case of geometry fitting from measurement, we are concerned only with true space, in up to three dimensions, and although the response surface methodologies are applicable, there are a number of other means of curve and surface fitting that have been utilized for many more decades.

A significant amount of literature pertaining to the fitting of geometric models can be found in publications discussing reverse engineering. In particular, during the 1990s, when use of computer aided design (CAD), computer aided manufacturing (CAM), and the use of coordinate measuring machines (CMM) had become widespread. In 1995 Chivate and Jablokow [40] reviewed surface definitions and fitting algorithms used in reverse engineering and rated the use of *algebraic*[17] surfaces and *parametric*[18] surfaces for a number of criteria and four types of fitting method. Their conclusion that parametric surfaces, mostly fitted using "the method of least squares" [40] or sometimes using *smoothing*, were preferable, carries through reverse engineering software today where spline-based parametric surfaces routinely form the basis of their coding, e.g. 3D RESHAPER®, and PolyWorks®. The flexibility of these definitions is emphasized by Chen and Liu [35] who state that they "offer one canonical form of the representation of both analytical and free-form shapes".

Even with advances in reverse engineering, the process remains somewhat dependent on either manual interaction, or some kind of cleaning process, to remove regions of inaccuracy or missing data points, or to specify the degree or order of the analytical definition, or the level of smoothing. The latest research focuses mainly on expediting the cleaning of polygonal meshes, often using optimization algorithms, prior to then fitting *non-uniform rational b-splines* (NURBS) if required [31, 86, 2]. Biomedical research now also draws on these approaches for building models of natural structures (bone joints, for example). They do not usually draw on information known about the expected form of the artefact, although a template-based approach was proposed by Buonamici et al. [23] in 2018, for applying a PSO algorithm for reverse engineering.

[17] An algebraic surface is one "that can be represented by an"..."equation $f(x, y, z) = 0$ in 3-space or by a *homogeneous* polynomial equation $f(x, y, z, w) = 0$ in 4-space. A homogeneous polynomial is one having all terms of the same degree" [40]. 85% of machined surfaces can be represented by a subset of these, *quadratic surfaces*, or *natural quadrics*, which can be defined using ten coefficients [40]. Similarly, for quadratic curves the z-terms are removed and six coefficients are required.

[18] A parametric curve is one that is defined in terms of a parameter along its length, denoted t in this text. A parametric surface is characterized by two such parameters, t and s, with three functions: $x = x(t, s)$, $y = y(t, s)$, and $z = z(t, s)$. *Spline-based surfaces* fall into this category and are defined by a set of control points where the surface remains within their *convex hull*.

For routine inspection measurement data, still largely CMM based due to their accuracy [228, 304], approaches for fitting are necessarily automated. There is conflict between the requirement to quickly determine whether multiple parts are within tolerance, and the longer-term goal of understanding the uncertainty in the manufacturing process in order to enhance the design or redetermine more appropriate tolerances. Therefore, with the inspection process, although the expected (nominal) shape or form is known, it is not always used to enhance the fitting process. Typically, if the form appears to fit reasonably then segments are considered individually and fitted using a least-squares type algorithm. Where shapes are seen to deviate from straight lines or planes and arcs or fillets, polynomials or splines are fitted in much the same way.

Despite advances in measurement technology and surface reproduction with CAD software, any statistical error in the measurements or fitting routines are generally not readily available. Such data may be embedded in measurement systems or CAD software, if it is recorded at all. In an attempt to ensure that measurement uncertainty be routinely incorporated into measurement processes the International Organization for Standardization (ISO) presented a framework by publishing its 'Guide to the expression of uncertainty in measurement' ('GUM') [116] in 1995, and a supplement in 2006 [117], although a number of works including that of Willink [316] highlight pitfalls in its approach. Most measuring system manufacturers quote a resolution pertaining to the spacing of possible measurements, rather than giving an expected statistical deviation. Researchers continue to work towards devising rigorous evaluations of CMM uncertainty [36, 74].

Meta-models (such as Kriging or other RBF-based models) can also be used to fit geometry, although this has seldom been presented in the literature. For surrogate models, a probabilistic measure of 'goodness of fit', maximum likelihood[19], is used to fit a model, but least squares[20] or other criteria, including the orthogonal least squares approach taken by Chen et al. [32] for signal processing, are also possible.

Radial basis functions have been experimented with for geometrical fitting but have not found their way into mainstream CAD software as far as the author is aware: Carr et al. [30] used RBFs for interpolation of surfaces for medical imaging of the skeletal trunk and later enhanced their algorithm using polyharmonic RBFs and greedy algorithms to reconstruct multiple three-dimensional objects, including a turbine blade with internal structure [29] (the advantages of Carr et al. [29]'s algorithm is its efficiency and its mesh repair (hole-filling) capability), Ohtake et al. [192] use least-squares based RBFs to enhance *adaptive partition of unity approximations* used for several overlapping data sets, and Liu et al. [158] suggest an *adapted K-means algorithm* to reduce RBF centres for

[19]"Maximum likelihood, also called the maximum likelihood method, is the procedure of finding the value of one or more parameters for a given statistic which makes the known likelihood distribution a maximum" [308].

[20]Least-squares estimators are, in fact, the maximum likelihood estimators in the case where errors are statistically independent, and normally distributed with constant variance [19].

improved efficiency of RBF based surface fitting. They have also recently been employed for surface modification, section 2.11. Kriging is afforded even less interest for geometric surface construction, although it is suggested by Liamaiem and ElMaraghy [151] for complete surfaces and used for enhancing existing surfaces by Grinstead et al. [95].

The latest developments in scanning and measuring hardware [89] technologies could soon mean that even more data are available, leading to increased scope for routinely integrating measurement with design processes, but potentially even more need to understand the uncertainties in the data, and to ensure that models are manageable in terms of variable numbers.

In this section, we discuss some important precursors, before referring to appendix A.7 for details of the commonly used curve and surface definitions (excluding response surface models, section 2.2) and the processes by which such definitions can be fitted.

2.7.1 Precursors to fitting

Point cloud data

Discrete measurement data can be collected as a set of points referred to as a point cloud, or a set of pixels with varying (intensity) referred to as an *image*. The points may be distributed randomly over a curve or surface, or regularly distributed in some kind of grid (akin to a full-factorial sample in DSO)[21]. There is no truly *continuous* measurement data; with extremely high resolution the number of points or pixels becomes larger, but they remain points or pixels each pertaining to an individual piece of measurement data. One might consider that increasing the resolution of measurements could reach a close-to-continuous measurement but, notwithstanding the still inherently discrete nature of the data, one is limited by computational expense, measurement time, and physical restrictions of the measuring equipment. Pixel-based images, although important, are not the focus of this thesis.

We note that although point cloud data for a single part might include multiple sets of points from different viewpoints, with the part supported in different positions, in multiple scanning processes, or "sometimes multiple sensor integrated approaches may be necessary to combine different characteristics of several scanning devices [254]" [111], this work is limited to single data sets due to the nature of the 'real' sets that are available to us.

[21] Although, point measurements may be manually distributed as selected by a measurement system user, examples of intelligent placing of point measurements in an automated or adaptive way have not been found in the literature; such sampling appears to remain with the expertise of designers using surrogates and DSO.

Shape or 'form' assessment

Through the assessment of form, it is possible to determine how the curves can be most simply defined, and which category of curve fitting algorithm is most appropriate. It may be possible to define the curves as:

- simple composite curves (straight lines and arcs joined using constraints) defined by end points, centre points, lengths and radii,
- more complex algebraic formulation curves (e.g. higher degree polynomials, ellipses etc.), or
- generalized algebraic curves such as radial basis functions, or
- complex parametric curves such as splines.

At the lower end of this scale, certain assumptions and constraints about the geometry are required, but the shape is defined by fewer variables in total. As the curve definitions become more general and more widely applicable, the greater the number of variables required to define them.

Implicit curves and surfaces, i.e. functions of the form $f(x_1, ..., x_n) = 0$, where $n = 2$ for curves and $n = 3$ for surfaces, which cannot be expressed explicitly as a function of one variable, introduce specific peculiarities in curve fitting, and often in the distribution of error in measurement. There is a non-unique value of one parameter at a given value of the other and measuring, for example, a circle or sphere is often difficult. It is often important to measure such shapes, e.g. holes or conics, and to inspect them for size, position, and correctness of form. In the late 1990s there was a flurry of papers dedicated to fitting circles and ellipses to CMM point measurements. Mostly, they make use of the recorded probe direction to allow the use of fixed regressor assumptions in the least-squares minimization, e.g. Watson [305] uses a convex local minimization and Chernov and Lesort [38] refine an algorithm for short arcs. There is a subset of research dedicated to *spherical regression* due to its importance for analysis of global geographical data. These methods are not explored further here.

If, as might be the case for analysis requirements only, a geometric definition for CAD reconstruction or manufacture is not needed, then an alternative discrete representation with a suitable quality and fidelity e.g. a *mesh*, might be more appropriate. This is where fitting, or characterization as discussed in chapter 1 may not be needed.

Interpolation and regression

Fitted geometric models may either be regressing or interpolating, where an interpolating model fits precisely through the given set of points, assuming implicitly that there is no

error or noise in the data points. In reality, this is seldom the case, but interpolating approaches can often be invoked easily. For example CAD packages, including NX™, can easily create spline curves of a chosen degree through given points and loft given curves to create spline surfaces with edge boundary constraints. Such curves and surfaces are defined by using parameters that include each of the point locations. The NX™ default spline interpolation function uses piece-wise polynomial splines and allows the user to specify the degree. MATLAB also has a cubic spline interpolation function for this type of spline as described by de Boor [48]. The interpolation condition results in a linear system which is *row diagonally dominant* and can be solved by *Gaussian elimination*. Curves and surfaces interpolated in this way can be particularly useful for visualization but can be limiting in their practical use because fictitious roughness can be generated in the surface (*over-fitting*) if the points are not actually noise-free, and at the extremities or at gaps in a data set the immediately adjacent points can have a disproportionate effect on the shape of the fitted curve. In addition, they inherently have the disadvantage associated with many complex model definitions that the number of variables (if used for further DSO or RBDO) is likely to be restrictive.

In contrast, regression assumes that the points contain some form of error and aims to fit the 'best' model of a chosen form given that there is some error. The resulting curve or surface will not intersect the points exactly unless the chosen curve form is able to pass exactly through them, which is extremely unlikely for large 'real-world' data sets with curves using fewer defining parameters than there are points.

The process of fitting either a regressing or interpolating model of geometry is akin to that of building a surrogate model as described in chapter 2.2. Indeed, regression methodology in curve fitting was a predecessor to such techniques. All of the routines follow the same basic approach: given a pre-defined curve or surface form, first determine a system by which 'goodness-of-fit' of a curve or surface can be quantified, and then use this measure as the objective function for an optimization of the curve or surface parameter values.

Form deviation is often indistinguishable from other noise and attempts have been made to filter form deviation [319, 229] using Gaussian process methods. We propose using a Kriging-based fit, based on the work of Picheny et al. [207] in chapter 6. An understanding of the true measurement error magnitude is required.

2.7.2 Registration and segmentation

Registration

It is usually important to determine the global position of the feature being measured, or its position relative to other features. The position may be required because it has an effect on component performance, or movement of the feature in space may position

it far from the nominal, therefore extending the required bounds for the fitting routine. There are papers dedicated to the localization problem alone, e.g. Rivest [233].

Segmentation

Where the measurements represent a component built of multiple curves or surfaces another difficulty, addressed mainly by the reverse engineering and engineering inspection communities, is that of determining the point at which one parameterized segment ends and the next starts. An early work by Hudson [113] categorizes curves made of two segments as being of one of three types:

1. with join point lying strictly between two successive x values and curves either side do not have equal slopes at the join,
2. with the join lying at a measured value of x with no restrictions on the slopes of the curves, or
3. with join point lying strictly between two successive x values and curves either side have equal slopes at the join.

If the join type is unknown then fitting is applied three times with all three assumptions. For curve types that are constrained to have equal slope at the join (type 3), an iterative solution is used where the fit (linear regression[22]) is conducted at successive join point locations. Such an approach is generally too expensive, even now, when considering a whole component, but may be of use for individual feature inspection and geometric parameter determination in uncertainty propagation and robust optimization[23].

When reverse engineering became more prolific in the 1990s, segmentation was "usually done manually" [35]. Mostly, b-spline curves and surfaces were fitted to manually selected regions and the curves and surfaces interrogated for points of maximum curvature [35]. Thompson et al. [286] use a feature-based approach with least median squares and Bézier curves. The method was still highly user interactive, although points with high expected error, based on distance and orientation, are automatically discarded depending on feature type. Huang and Tai [112] and Tai and Huang [282] use, what they describe as a *median method* to construct new points by averaging the positions of each set of three neighbouring points, from which a curvature can be calculated for segmentation. These methods are referred to as *edge-based* segmentation.

Attempts at automation of segmentation only really begin to enter the literature in the 2000s. Huang and Menq [111] first construct a triangular mesh with the data points

[22]Linear regression was the only practical approach with the limited computer processing power of the 1960s.

[23]However, no examples of this have been found in recent literature.

and then calculate curvature tensor values by Euler's theorem [59]. A threshold value of curvature is specified for the segmentation. Woo et al. [318] and Li et al. [149] also suggest mesh-based approaches. Other such methods include that of Rabbani et al. [220] who use *K-nearest neighbour* (KNN), and *fixed distance neighbour* (FDN), methods to calculate normal vectors from the raw data, requiring user input of threshold values of smoothness. An approach by Rahayem and Kjellander [221] recursively splits initially segmented surfaces where threshold error requirements aren't met. Mesh-based approaches are usually referred to as *region-based* as they split the surfaces into sets of elements, or regions according to the properties of the elements. Those mentioned here are also *top-down* algorithms as the mesh is fitted to the entire object before segmenting.

The automatic segmentation problem continues to be of interest; human interaction introduces inconsistency, the potential for human error, and makes statistical representation of fitting error difficult or impossible. Vančo and Brunnett [298] attempt to automatically segment surfaces using normal vectors from the original data set, in a *bottom-up* approach which locates the main surfaces, growing them to a point where the given threshold is met.

2.7.3 Common curve and surface fitting methods

Methods used for defining and fitting geometry (curves and surfaces) to data have developed in parallel, and prior to, the definitions and fitting routines for developing models of responses. Although similar in many ways, the potential simplicity of a limited number of dimensions but complexity of a potentially intractable number of points in geometric modelling, as opposed to high-dimensional spaces and sparse sample points in response surface modelling, has meant that they are usually considered from slightly different perspectives. In this thesis, we are interested in both; the geometric uncertainty problem requires the conversion of geometric definitions based on an intractable numbers of points, to design spaces of practical size.

There are a number of classical mathematical models and associated fitting methods for dealing with geometric fitting, some of which are described in appendix A.7:

- polynomial models and least-squares regression,
- orthogonal least-squares,
- spline fitting, and
- Bayes regression.

2.8 Evaluating the accuracy of fitting (curves, surfaces, and meta-models)

Given that a curve, surface, or meta-model has been constructed which is believed to be the best possible prediction of the 'true' geometry or function, one may wish to assess the 'correctness' of the fit in a quantitative manner. In order to compare the closeness of fit for one data set with that of another, a single numerical value (a kind of score) may be appropriate. Alternatively, one may wish to understand the statistical confidence that we have in the fit or a particular property of the curve. For some purposes, one may also be interested in accuracy in a particular region of the curve or surface.

Origins of fit inaccuracy

During the fitting process for measurement data we are aiming to represent 'real' uncertainty due to the manufacturing process, or in-service wear, accurately. The fitting process itself, however, can introduce further inaccuracy. We refer to this as *error*. It is a combination of our ability to select the correct analytical curve or surface construction, parameters, and fidelity (we shall describe this as *fitting error*), and our ability to deal appropriately with the *measurement error*, which may be described as *noise* because it is usually random if equipment has been correctly calibrated. Weckenmann et al. [306] and Dowling et al. [61] discuss the range of potential errors through measurement. Erkan et al. [63] reviews errors introduced through CMM tip radius correction software and shows them to be in the order of 1.0e^{-6}mm for Mitutoyo software.

Similarly, models in place of computational analysis chains can also contain noise, usually from numerical truncation error. The error in the surrogate model fitting process is akin to the error in the measurement fitting process.

In both cases systematic errors need be removed through *validation* of either the measurement system or computer code prior to use.

2.8.1 'Goodness-of-fit' and confidence

A universally popular metric for assessment of the error in a model or surface fit to a set of data points is the *mean squared error* (MSE), also known as the *residual mean square* or *standard error*, or its square root: the *root mean squared error* (RMSE). Forrester et al. [78] describe the process of determining this metric as 'model testing' and

$$\varepsilon_{\text{RMS}} = \sqrt{\frac{\sum_{i=0}^{n_t}(y^{(i)} - \hat{y}^{(i)})^2}{n_t}}, \tag{B.19}$$

where n_t is the number of test points (one, or a number of, sample or measurement evaluations, held back as a validation set).

Other residual-based measures are also used, such as the *sum of squares due to error* (SSE), and *R-square* (the "square of the correlation between the values and the predicted values" [165]). In general these measures are defined in terms of model building, where the regressor, x, has no error. In the case of curves fitted using orthogonal least-squares, one must take care to use similar measures using the orthogonal residual values. To compare the global accuracy of a model or a number of fitted curves the use of such measures is reasonable. However, is has three pitfalls.

1. If we wish to compare the accuracy of different approaches to fitting then these measures will obviously tend to prefer the least-squares based fits. Any such measures will be biased in this context.
2. Using the MSE metric alone, over-fitting will result in a low MSE.
3. It is seldom possible to hold back sample points where the size of the set is limited by expense or data availability. One might even argue that, for small sample sets, reducing the training set size still further for validation purposes is too detrimental to the accuracy that one aims to evaluate.

These test metrics also rely upon the fact that the *form* of the curve or surface is correct. If this is unknown then one can construct a statistical test of the assumption that the errors are normally distributed and ensure avoidance of over-fitting, such as the test of Ralston and Rabinowitz [224] for a polynomial fit. *Confidence intervals* can also be calculated using the MSE at a given confidence level for the curve or surface coefficients assuming that the deviations from the true curve can be considered to be normally distributed [165]. However, these approaches do not avoid the requirement for additional data sets for model testing. A common error assessment in these circumstances is *cross-validation*.

2.8.2 Cross-validation

If the data points used for fitting are some large set of randomly selected points from a population then there is much freedom over the choice of training dataset and testing dataset. One may choose multiple new random sets from the same population or randomly select a subset of the original data for testing. In the case of measurement, the data points are not randomly distributed over the curve or surface, they are generally relatively evenly spread. In addition, there is only one set. In fact, even a second physically measured dataset on the same part would not be a valid testing set as it would not result in points drawn from the same population. In the case of a surrogate model

the training set is often too small. Cross-validation [101] is a validation of a model by re-sampling the original data set.

Given that a model has already been built using all the sample points and the parameters for the model have been determined through surface fitting or model training, *K-fold cross-validation* can be used to estimate the likely global error in the model. The original sample is split into K sub-sets of roughly equal size (n_s) randomly selected without replacement. In turn, each subset is left out and the fitting, or training, process repeated. The same equation is then used to predict the left out points and a *loss function* is calculated at each of these points. Usually the *squared error* is used as the loss function. Averaging over all n_s left out and re-predicted points in all K sets results in a mean cross-validation error of

$$\varepsilon_{\mathrm{CV}} = \frac{1}{K}\sum_{j=1}^{K}\left[\frac{1}{n_t}\sum_{i=1}^{n_t}(y^{(i)} - \hat{y}_j^{(i)})^2\right]. \tag{2.33}$$

More specifically, *leave-one-out cross-validation*, where $K = n$ and $n_s = 1$, is a popular approach but is prohibitive if the training routine is expensive. For Kriging, where the tuning of hyperparameters can be expensive, a reasonable estimate of the full cross-validation error is a cross-validation error where the optimized parameters are fixed but the predictor (a function of the correlation matrix, $\mathbf{\Psi}$), is re-calculated.

Cross-validation gives an estimate of the *prediction error*, which although it is not a meaningful number in itself for measurement data, is a good comparator (it is not *unbiased* in a statistical sense but if the data set is large this causes little difficulty). Cross-validation error can also be used as the objective function for minimization in model training for other RBF-based models where likelihood is used for Kriging models.

A related, and popular approach for estimating error in a predictive model is *bootstrapping* [101, 216]. This is typically used when there is only one sample, which is drawn from an *independent, identically distributed* (iid) population, and involves taking repeated sub-samples drawn *with replacement*. Given the evenly spread nature of the measurement data, one "can usually get away with pretending that these are iid uniformly distributed over the measured range" [216], but must be particularly careful if the fitting routine requires sequential points, or if there is any "small-scale clumpiness" [216] in the points. Intuitively, a leave-one-out scheme (where points are drawn *without replacement*) is preferable for measurement data as it gives equal weighting to all remaining points, where bootstrapping effectively increases the weighting of any points drawn multiple times. Of course, the larger the data set, the smaller this effect.

2.8.3 Proof-by-example

An alternative, or concurrent method for assessing fits for data sets of known form, is one of 'proof by example'. If one assumes a 'perfect', or nominal, geometry with a representative number of points, perturbs said points by introducing Gaussian noise of a representative magnitude, and uses the same algorithm to fit this artificially generated data set as that of the measured data set, then one can conclude that, if the normal error assumption holds, then this is equivalent to the expected error. The effectiveness of a given fitting routine can be demonstrated 'by example'. A statistically robust prediction would again require multiple fits to multiple randomly perturbed simulated data sets. Even so, this may be preferable, especially if comparing different fitting routines for similar geometric features. This approach will also allow for errors caused by registration if representative rotations and translations can also be applied to the synthetic data points.

2.8.4 A special case: Kriging predictor error

None of the error estimates discussed thus far separate noise in the sample points from error due to the model prediction itself. They also result in an overall error value, and local variation in prediction error is not considered. A significant advantage of Kriging (section 2.2), and other similar approaches, is that an estimate of the mean squared error in the predictor, $\hat{y}(\mathbf{x}^*)$, can be evaluated as a function of $\boldsymbol{\psi}$ at any $\mathbf{x}^*$. This *Kriging predictor error* can be shown to be

$$s^2(\mathbf{x}^*) = \sigma^2 \left[1 - \boldsymbol{\psi}^T \boldsymbol{\Psi}^{-1} \boldsymbol{\psi} + \frac{(1 - \boldsymbol{\psi}^T \boldsymbol{\Psi}^{-1} \mathbf{1})^2}{\mathbf{1}^T \boldsymbol{\Psi}^{-1} \mathbf{1}} \right]. \tag{2.34}$$

In order to reach this relatively easily evaluated analytical solution, $\hat{y}(\mathbf{x}^*)$ is treated as a Gaussian random variable and the mean squared error of this predictor averaged over the Gausssian random process. This approach is described by Sacks et al. [242], and a detailed derivation of this result from first principles is given by Hoyle [108] based on the work of Cressie [44]. This prediction error estimate is fundamental to advanced methods to enhance models through adaptive sampling routines.

2.9 Statistical methods

There is little to be gained from describing the many well-documented statistical tools used in data analysis here. The fundamentals are covered in many large tomes. However, given that the subject of this thesis is the facilitation of a probabilistic understanding of performance in order to inform design choices in the face of measurable uncertainty, it is important that a short overview of some of the applied statistical theory is provided.

In appendix A.8 a brief overview of some pdfs that are typically found to be of use in engineering design. Such distributions include, the often assumed Gaussian normal and *uniform* distributions, along with the *truncated* normal likely to be experienced if components are rejected as out of tolerance, and the *log-normal* favoured for some material property variations. The general form for fitting unusually shaped distributions: the *Kernel density*, is also presented. A section is also dedicated to a more recently popularised stable distribution, which can be particularly useful for representing skewness, or where tails of a distribution are heavy. We also present the *hypothesis tests* which have been used in this work to assess the accuracy of distributions fitted to input variables, although plenty more alternative tests exist.

It is not just the distributions of variables themselves that must be accurate, but the correlation structures between them. Correlation and *analysis of variance* (ANOVA) are of great relevance to the engineer who wishes to use data to enhance the design process. They can enable reduction of variable numbers and ensure that correlation between measured variables is not ignored; assuming independence between variables is commonplace with the lack of any other information, but might lead to misleading predictions of performance variation. A description of ANOVA and measures of correlation are given in appendix A.8.

Once pdfs of, and relationships between, variables have been established the task of sampling the correlated variables can become difficult. In which case, the statistician turns to *copulas*, and popularly the *Gaussian copula*. These enable multi-variate distributions to be constructed such that correlations maintained during sampling, appendix A.8.

2.10 Dimension reduction

Undoubtedly the most popular approach to reducing dimensions which, in its simplest form is a linear representation of an original variable space, is principal component analysis (PCA). PCA is also known by various other names, including *classical scaling*. Wold [317] lists some of these by discipline:

- *singular value decomposition* (SVD) is sometimes used in place of PCA in "numerical analysis",
- Karhunen-Loève (KL) decomposition (or expansion) is referred to in the field of "electrical engineering",
- *Eigenvector analysis* or *characteristic vector analysis* is used in the "physical sciences",

and others include:

- the *Hotelling transformation* in image analysis, and
- *principal factor analysis* in chemistry.

This method results in a new variable set of *principal components* (PCs) that are uncorrelated, which is beneficial for statistical sampling, avoiding the need for copulas (appendix A.8). Details of this method, applied in chapter 6, and of its variants, are provided in appendix A.9.

It is also worth noting that PCA has other uses, not just dimension reduction. As listed by Wold [317], the uses of PCA are:

- simplification (reducing the size of a data matrix to a size that can be interpreted more easily by human observation),
- data reduction (the approximation of a large data set to a less complex model structure),
- modelling (of a physical or chemical system by isolating the 'main effects' from the noise, the data matrix is the *training data set*),
- prediction (of unknown data points in the modelled system),
- classification (finding *classes* (*clusters*, or groups) of similar *objects* or data points),
- outlier detection (by identifying sample points or objects that lie outside of known clusters), and
- *unmixing* (where the *end members*, either known or identified, in the new principal component or 'factor' space are used to define each object or data point as a 'mixture' (often used in the field of geology) [82, 137]).

2.11 Geometry and mesh manipulation

The methods described thus far can be used to enable the designer to:

1. understand the data (select appropriate variables that define the uncertainties and ensure that we a have suitable probabilistic definition of them): **characterize** the geometric uncertainties, and
2. plan an appropriate 'black-box' UP, RDO, or RBDO study (select sample plans and robustness parameters or objective functions, and optimization algorithms with, or without, surrogate models).

There is, however, a final step that is not trivial in the case of geometric uncertainties: **implementing** the changes to geometry within the analysis. Uncertain variables, such as boundary conditions or loads require only a trivial change to a numeric value, but geometric uncertainty is non-trivial to implement, unless the existing geometry has the same parametric definition (defining variables) as the uncertainty being studied.

Analyses typically contain a geometry created in a CAD tool, which is then meshed, i.e. discretized into a set of *elements* connected by *nodes* at their vertices. Whatever the type of mesh (solid, or shell), it is fully defined (in a geometric sense) by the nodal locations and their connectivity with each other. In order to implement the sampled changes within the workflow, a modification must either be made to the original geometric model, or the mesh, whilst maintaining associated loading and mesh quality. Modifying the original geometry means that a fully automated and robust meshing process be employed but modifying the mesh leaves the designer without a CAD model of any analysis points of interest. In this section, we review the methods available for making modifications to the mesh, in particular, although the manipulation techniques can also be applied to CAD models by manipulating the positions of the defining points or vertices whilst maintaining the mathematical curve or surface definition.

When a deformation is applied to an existing geometry of any definition to morph it into an alternative geometry of 'free-form' i.e. "an object whose surfaces are not of a more easily recognized class such as planar and/or natural quadric surfaces" [26], a number of mathematical methods can be employed. The deformation is usually facilitated by software applications through manual interactions. Such manipulations (like the physical moulding of clay) can be for many purposes: usually design or re-design, reconstruction of an existing entity, or parameterization of a new product topology based on movement of control points. With design and reconstruction in the sense of a single designer experimenting with ideas prior to analysis, or creating a visually appealing item, the algorithms used need only reduce computational effort such that the computer processing and graphics can keep up with the user. Given the processing power now available in modern computational hardware, the large number of points needed to define a complex free-form surface structure completely, and the processing required to move the points, no-longer tend to be prohibitive in this respect.

A 'true' body shape is actually a free-form object of infinitely small precision (defined by an infinitely large number of points). Therefore, in the uncertainty problem one is typically presented with a parametric design and multiple free-form representations of 'the truth' expressed as point clouds or similar, of varying precision and accuracy, section 2.7. The goal is ultimately to morph an existing parametric model and/or its associated appropriately refined analysis mesh such that mesh connectivity and quality is maintained. Deforming the geometric model and re-meshing maintains control of rule-bases that ensure mesh quality but is likely to introduce more computational noise. In

addition, if the deformation requires manual intervention then it is also infeasible for large numbers of model evaluations.

Existing algorithms for manipulating geometries and meshes with varying degrees of automation fall into one of three categories:

- spatial deformation; deforming an object by "warping its ambient space" [83], or
- *surface mapping*; creating a mapping directly from one geometry to another whilst minimizing either the "angular or area distortions" [150], or
- *volume-based* or *voxel-based* morphing, where a voxel is a 3D cube akin to a pixel in 2D space,

and all three are often referred to as *morphing*. They can be either *mesh-based* or *structural*. Although pixels and voxels might indeed be the result of measurement (X-ray and micro computed tomography (µCT) for example), such data processing is beyond the current scope of this thesis, which is focused on processing of point-cloud data.

Free-form deformation (FFD), a popular form of spatial deformation, is commonplace as a tool in engineering design where the performance parameters are those related to fluid interactions (aerodynamic or hydrodynamic surfaces) [188]. Its potential (and that of related techniques) for representing uncertainty for propagation and robustness studies is yet to be fully explored. Algorithms have been developed and implemented for software designed primarily for imaging and for computer graphics in the design of realistic animation (in the gaming and film industries). Some are applied to reverse engineering problems. However, in both cases the morphed geometry is an approximation, rather than a statistically accurate definition of the required shape, and the algorithms are only just becoming routinely into FEA and CFD meshing and analysis tools for engineering design. Examples are Siemens NX™, Altair® HYPERMORPH™, and Enginsoft Optimal Solutions Sculptor™. The Siemens NX™ morphing capability moves an existing mesh associated with geometry to match a new but 'similar' geometry. This approach has been used in the dovetail analysis in this thesis. The capability is also seen in HYPERMORPH™ as part of a larger suite of "simulation driven design" tools, and as a stand-alone tool Sculptor® based around what they call 'arbitrary shape deformation'. The algorithms used within these packages are the intellectual property of the respective companies and are not accessible. They could be surface-based or spatial methods, or combinations of both.

Reasonably recent developments, embedded into ANSYS fluent CFD software, use radial basis function models of nodal displacements to deform meshes: termed *RBF morph* by the first publisher of the method [15].

2.11.1 Surface mapping

There are a number of approaches to morphing surfaces using various parameterizations [73].

These approaches are usually:

- largely interactive (due to partitioning and/or feature pairing), and
- based on *planar* parameterization (which introduces a need to partition the two meshes into consistent patches, and can also result in undesirable 'seams'), or
- based on *spherical* parameterization for closed surfaces, which does not require splitting [253, 6].

The methods usually require that surfaces are "within the same *homotomy* class" (are *isotopic* to each other), i.e. "one can deform to another smoothly" [150]. If the target surface is not genus zero (a sphere or plane) then searching for the optimal mapping can result in a locally, rather than globally, optimal map. For these reasons there are numerous variations of these schemes documented including *consistent* parametrizations where the original mesh is not maintained [140], the direct *manifold* parameterization approach to mapping a desired mesh definition onto the surface of an alternative undesirable mesh [326], the globally optimal mapping of Li et al. [150], and *automated wrapping* where the mapping is to and from an intermediate 'auxiliary' surface [258]. A comprehensive overview of mesh parameterization methods, terminology, and applications is given by Sheffer et al. [252], although they state that given the "amount of active research", "no survey on the topic can, realistically, be truly complete". A review of surface deformation methods is also provided by Botsch and Sorkine [16]. Further refinements to these methods continue to be published, e.g. [109, 65].

One could envisage editing only the surface mesh and then re-meshing the resulting solid. This approach has been taken for bio-engineering problems where similar, but not identical structures, such as bones are modelled. Sigal et al. [258] use surface morphing techniques to reduce the cost associated with processing of each individual specimen scan. In an engineering design context one could use a morphed surface mesh as a seed for the uncertain component mesh. However, the process could remain expensive depending on component mesh complexity and level of automation. Neither would it necessarily be significantly less noise-free or reliable than re-meshing a morphed solid geometry from scratch. Surface-based approaches could yet provide some benefits but are considered beyond the scope of this thesis.

2.11.2 Spatial deformation

Rather than editing vertices and manipulating neighbouring surface nodes, spatial approaches manipulate a body by 'warping' the surrounding space: *free-form deformation* (FFD). They fall into one of four categories: 3-dimensional volume-based, 2-dimensional surface-based, 1-dimensional curve-based, and zero-dimensional point-based deformation schemes. The limiting factor in the application of these methods to the design and uncertainty problem, as touched on earlier, is that, in most cases (excepting point-based methods), controlling the precise displacement of points within the component mesh can be "arduous in the extreme" [83].

Where such approaches are used for automated design, via DSO for example, it is only the number of control points used that determines the computational effort, and not the thousands of points defining the actual shape. The designer is free to select and position such control points as they wish within the constraints of the algorithm used. However, this design approach is limited:

- irrespective of the structure of the initial geometry, the deformed model has a free-form structure (is defined by multiple points), and
- there is no direct conversion from known deformed feature points to control-point location required to match the shape. This would require an iterative process or optimization for which the selection of the most appropriate initial control point locations may also be non-trivial.

In addition to these *meshless morphing* techniques there are the mesh-centred approaches reviewed by Staten et al. [272] and Sieger et al. [257], some of which are referred to as *mesh warping*, *mesh moving*, or *mesh smoothing*. A promising mesh-based method, the RBF morph employs a RBF-based model of nodal displacements at a surface to predict nodal displacements for non-surface nodes in a method which combines free-form deformation (FFD) and mesh-based variational methods.

In appendix A.10 we discuss the fundamentals of FFD, which is used in this thesis, and review the radial basis function (RBF) morph developed by de Boer et al. [47] and now embedded in ANSYS Fluent as (rbf-morph)™, written by Biancolini [15].

Having presented some of the most fundamental and useful approaches for the practical application of UP and RDO in the face of measured geometric uncertainty, we continue to the next chapter where we examine the process of combining these methods. The UP and RDO literature pertaining to geometric uncertainty is reviewed, showing the approaches that are typically adopted, and areas where further research and development is needed.

Chapter 3

A process for modelling geometric uncertainty

The automated process of RDO, or RBDO, as built up from a deterministic optimization workflow by combining with the characterization of measured uncertainties and including surrogate modelling, is encapsulated in the flowcharts, figures 3.1 to 3.5. Figure 3.1 shows the deterministic optimization process, where performance is evaluated at various design points using an analysis workflow based on parametric CAD and discretizing (meshing) the geometry for analysis. The analysis is often costly and the process of optimization is, therefore, often enhanced using a surrogate model of the design space (figure 3.2). If one wishes to design for robustness or reliability then uncertainty propagation can be added at each sample point in the design space, requiring sampling in the uncertain space also. This process, with additional steps for characterizing measurements for UP, and also enhanced using surrogate models is shown in figure 3.3. This integration of surrogate models for both performance in the uncertain space and variability in the design space the process can potentially be made practicable in terms of cost. Figures 3.4 and 3.5 illustrate the same processes with surrogate models used either for RDO and RBDO, or UP, but not both, respectively.

The many tools that can be considered useful for enabling this process have been reviewed and described in detail as appropriate in chapter 2. In this chapter we review how other researchers have applied these, and other methods, and how they can be combined for studies of uncertainty in engineering design.

Given that DSO, UP, and RDO all aim to 'inform design decisions', in preference to 'giving answers', we note that, where surrogate modelling is used, the designer is provided with the capability to visualize the geometrical variation, the design or uncertain spaces (although usually only in two dimensions concurrently), and understand the trade-offs, making the proposed Kriging-based processes described in the following chapters particularly appealing.

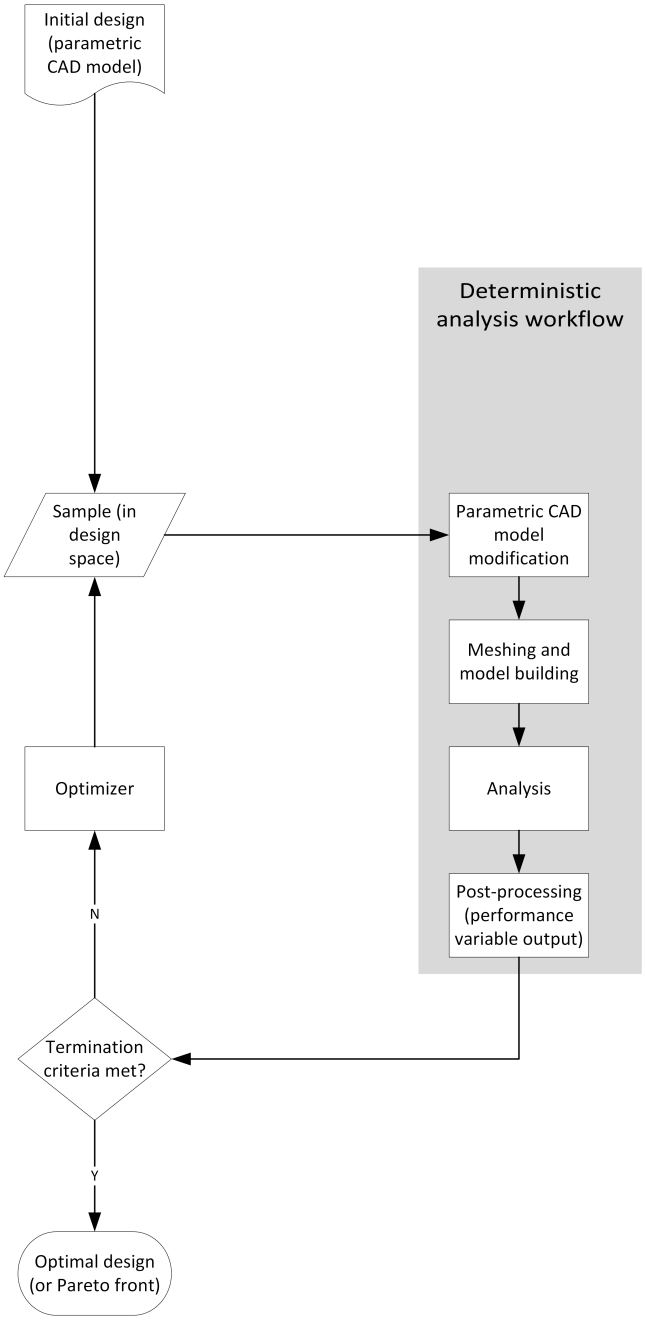


Figure 3.1: Deterministic optimization

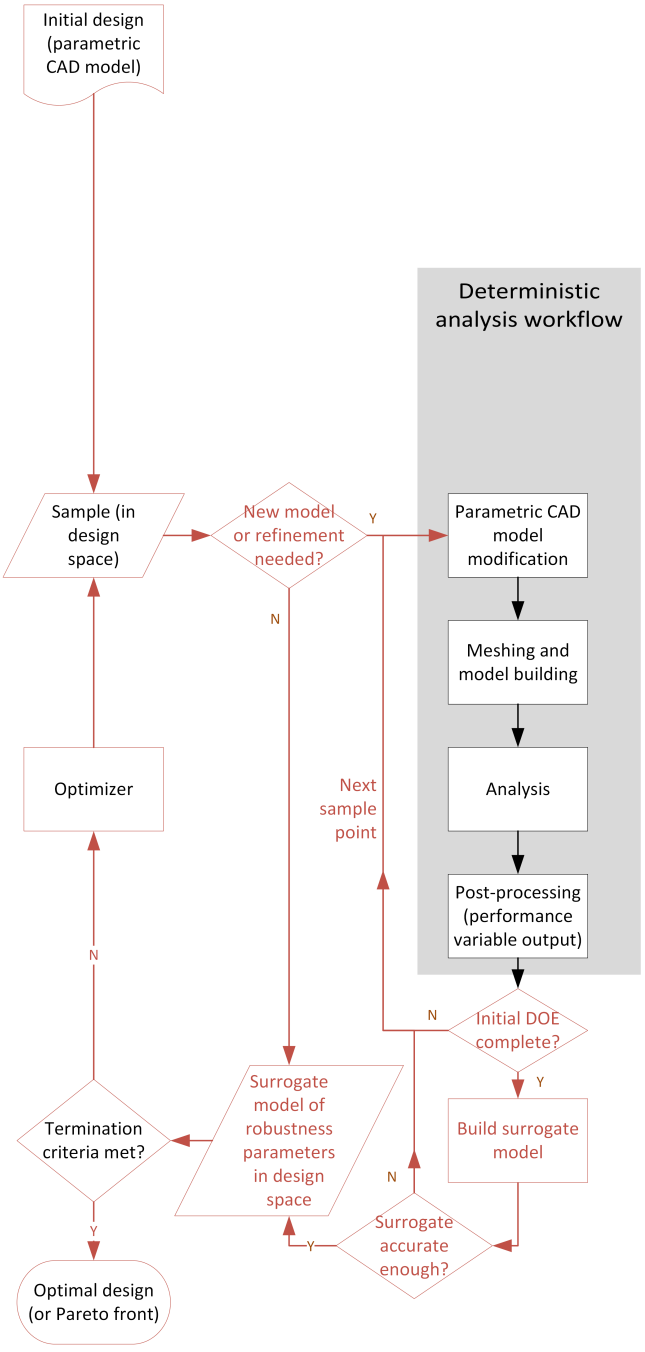


Figure 3.2: Deterministic optimization, enhanced using surrogate modelling

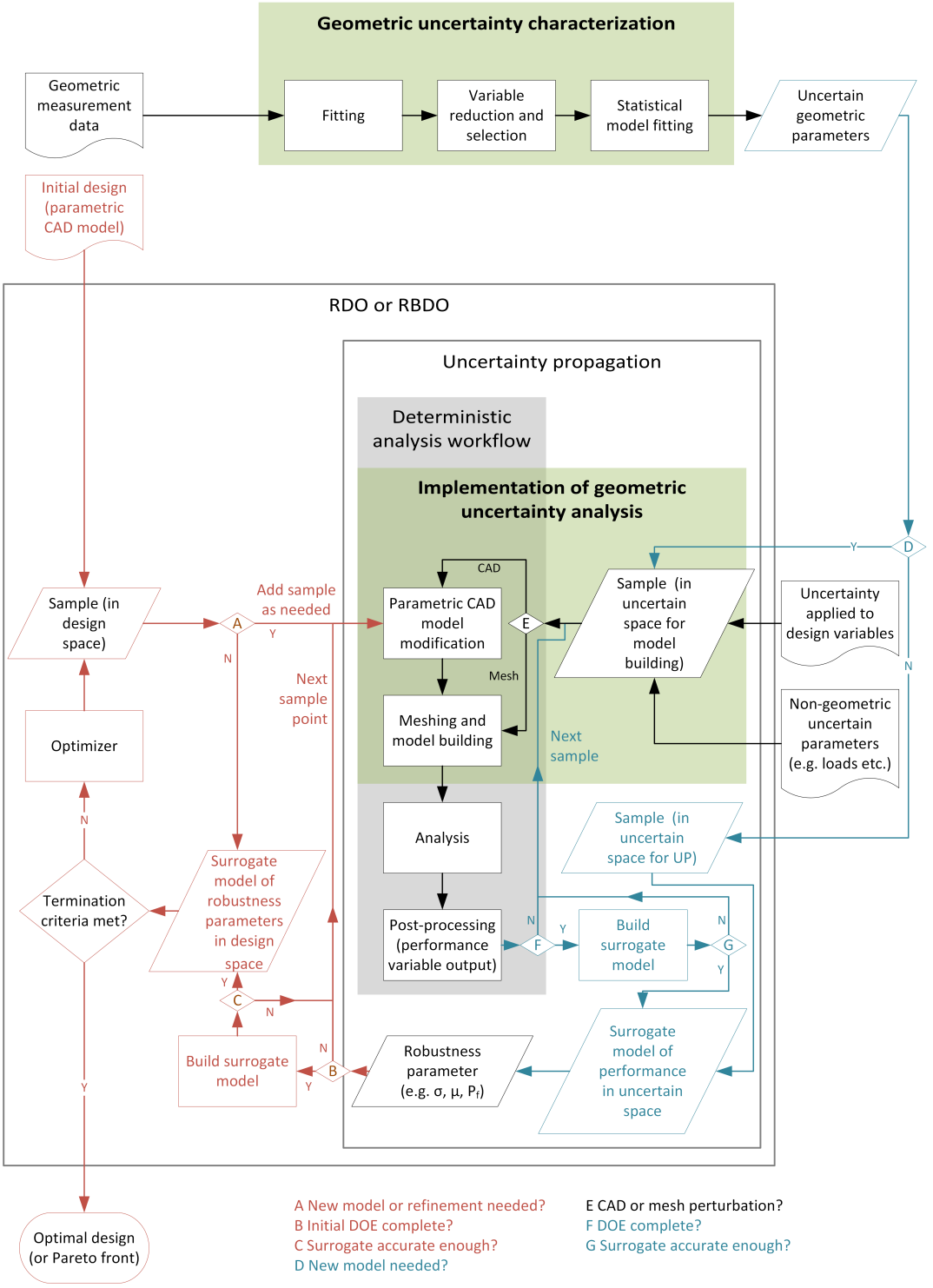


Figure 3.3: Surrogate-enhanced stochastic optimization with surrogate-enhanced uncertainty propagation

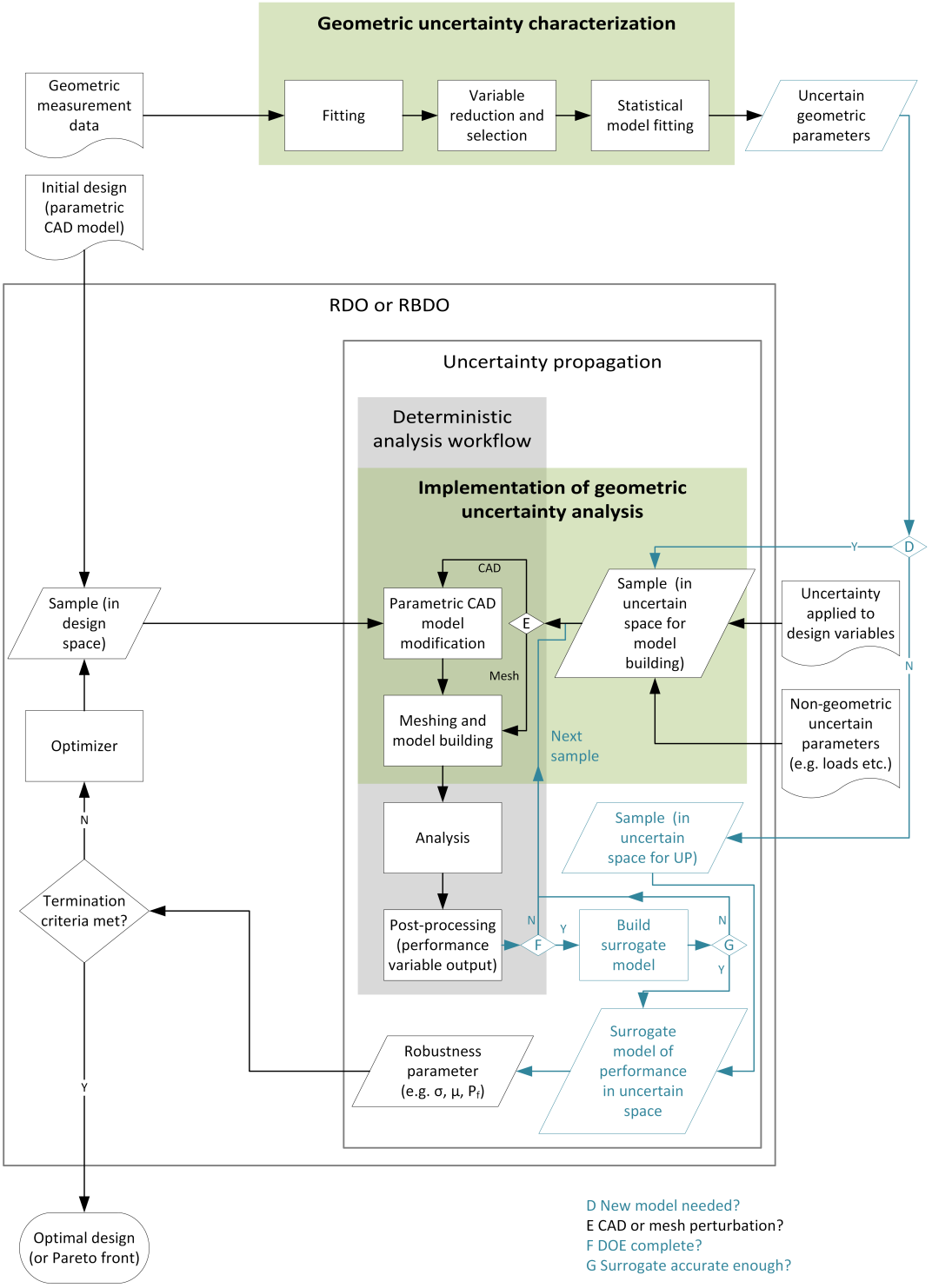


Figure 3.4: Stochastic optimization with surrogate-enhanced uncertainty propagation

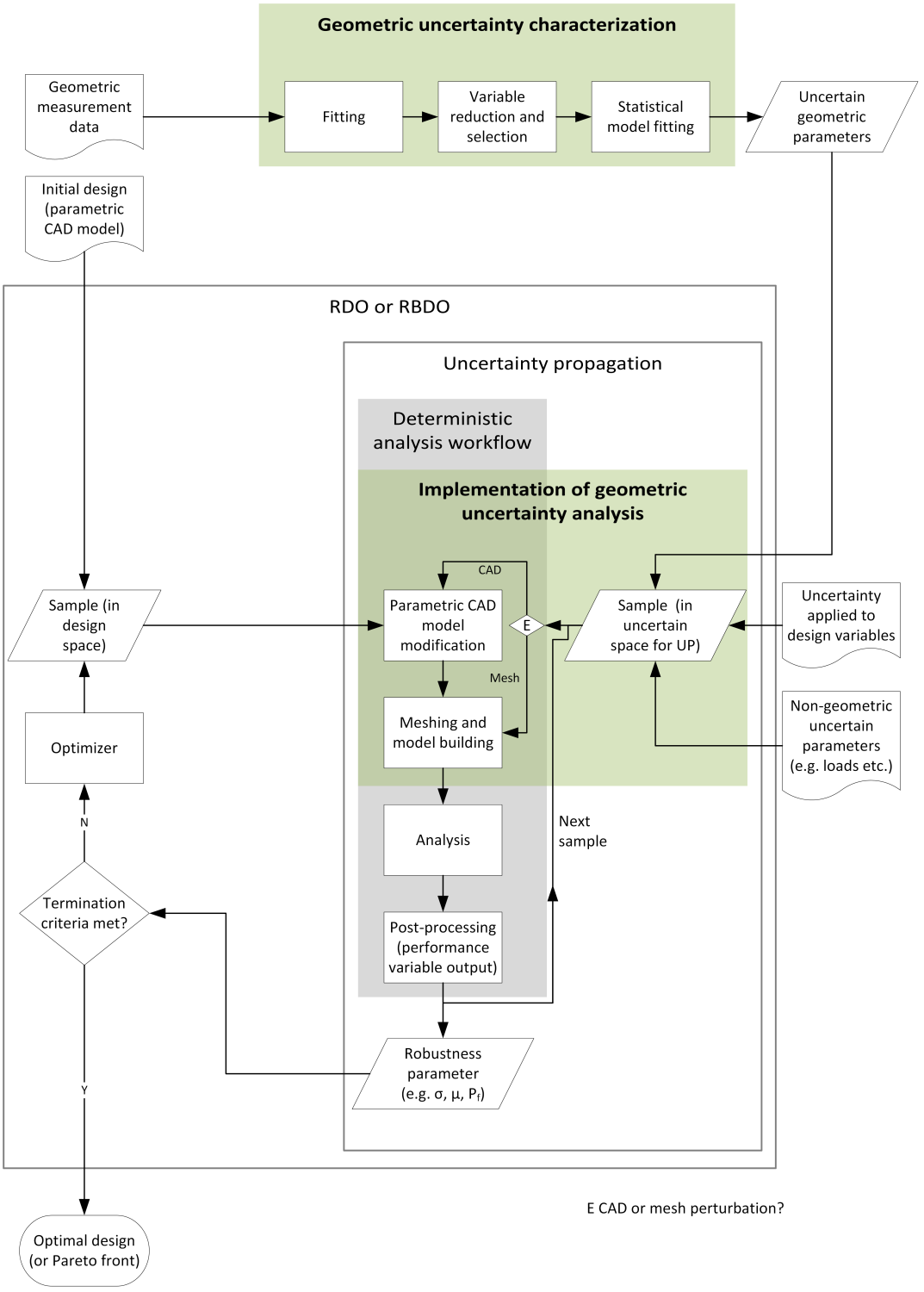


Figure 3.5: Stochastic optimization with uncertainty propagation using meausurements

Uncertain parameter definition

Chapters 2.5 and 2.6 consider RDO and RBDO in the context of known uncertain inputs. In reality, robust-optimal design choices are usually based on some educated guesses of statistical inputs, rather than implementing an 'uncertainty characterization' or modelling process (as shown in figures 3.5, 3.4, and 3.3), and assumed invariant across the design space. In recent examples Fang et al. [66] study the effects of uniformly distributed values of material properties on truck cab fatigue life, Campobasso et al. [27] assume that 13 design variables are "affected by non-correlated normally distributed uncertainty" in the design of wind turbine rotors, Renault et al. [230] turn to experiments to determine the extent of different component moduli variation and then select an algorithm "which implicitly considers Gaussian distribution" for brake squeal uncertainty propagation, and Dey et al. [56] assume that ply rotational uncertainties in the composite lay-up of a conical structure are uniformly distributed within a range. Such assumed inputs enable design searches, but can lead to scepticism in the process if predicted performance uncertainty appears to contradict anecdotal observations. If however, we have some real data pertaining to some of the uncertainties, one could hypothesize that it is possible to use these data to further inform and refine the robust design process. In the context of geometric uncertainty and measurement data, this thesis seeks to implement such uncertainty characterization, in an automated fashion, using 'real' data sets. This hypothesis is made notwithstanding the fact that measurements are made on physical manufactured components and as such are made at a single design point only and often for inspection at non-random points in the manufacturing process.

Application of stochastic variations to existing analyses

Richardson et al. [232] implement multi-objective DSO, and multi-objective RDO (with the NSGA-II non-dominated sorting algorithm), using an intrusive stochastic finite element solver where the uncertainties are the loading and material properties. Jansen et al. [120] apply the same approach to the nodal positions in a truss structure. Such methods including the *spectral stochastic finite element method* (SFEM) [279, 273] are popular as they can be easily implemented for simple FE models with an appropriate solver. They internally implement polynomial chaos expansions (section 2.4), which are more tenable when using modification of the internal code. Again, distributions are assumed, but these are more complex models: the load is "expanded in the polynomial basis", and the Young's Modulus varies across the parts as a Gaussian or log-normal random field, [284, 44, 208]. Similar stochastic analysis is possible with *stochastic computational fluid mechanics* (SCFM) [153]. If the analysis chain is not capable of intrusive approaches, or their implementation overly complex, then changes cannot be implemented by making modifications to values of cells in the embedded matrix calculations, but must be fed into the inputs: the mesh definition and boundary conditions.

There are few examples of papers dedicated to methods of implementing uncertain geometric changes in the literature. There are descriptions of 'shape optimization' where shapes are modified in the design space. Typically, this implies defining shapes according to design intent (parameterization) or taking a free-form approach using predetermined lattice point locations (section 2.11). For example, Koshakji et al. [139] apply this approach to sailboat bulb and rudder hydrodynamic design, and Sieger et al. [256] three shape deformation approaches to this end in automibile aerodynamic design. However, many uncertainties, including geometric ones, are spatial variations, such that their definition has high dimensionality. We, therefore, require a characterization of the uncertainties that both

- reduces the space for practical UP and DSO, and
- can be implemented through non-intrusive means.

PCA, or KL expansion, is often applied to reduce the dimensionality of large random fields, most often for intrusive UP, e.g. Richardson et al. [232]. We are beginning to see the approach appearing for non-intrusive UP and RDO. For example, Schillings and Schulz [247] and Liu et al. [157] use non-intrusive approaches for a RDO study and a UP comparison, respectively, for an aerofoil geometry. Schillings and Schulz [247] apply assumed truncated normal distributions centred on nominal values for uncertain parameters such as Mach number and apply a random field based on KL expansions (section 2.10) for a reduced-space set of perturbations of surface points for geometric uncertainty applied to a relatively simple 2D model. They apply non-intrusive PCE using adaptive sparse grids (section 2.4) for the UP, and compare the use of a number of different robustness objective functions used in a single-objective robust optimization using SQP (section 2.3).

When PCA is applied for DSO, there is a subtle difference in that the variable set is reduced in size using data from performance assessment, rather than measurement. Diez et al. [58], however, use this stochastic view of geometric variation to apply the KL expansion in order to reduce the size of the design space in the DSO of a catamaran hull-form using FFD, and Garzon and Darmofal [88] and Roy et al. [239] also use PCA for dimension reduction in this way.

In all of the example cases the implementation of changes to the analyses to reflect the uncertainties in the resulting variable sets are trivial given their existing construction. Similar examples of dimension reduction and geometry manipulation in a non-intrusive UP or RDO framework applied to large CAD models, meshes, or analysis chains have not been found.

Selecting the tools

The appropriate set of tools for the characterization and implementation of uncertain geometry is therefore mainly dependent upon the available data, the existing design and analysis process, and the goal of the study.

In the case of geometric inspection data, although it is plentiful in its existence, it is often not integrated with design[1], and is only called for in cases of failure of components. Indeed, if the analysis is not set up to make use of such data, then turning to the investigation of parameters whose uncertainties can be easily implemented within the existing analysis is sensible in the first instance.

In this thesis, we begin from the standpoint that we have some geometric inspection data which we wish to integrate into existing design workflows, and that the goal is both UP, and RDO. The choice of tools and workflow will depend upon the factors listed.

1. Data
   (a) type:
      i. none,
      ii. parameters (design variables or other),
      iii. point clouds (2D or 3D), or
      iv. pixels or voxels (not addressed here),
   (b) quantity:
      i. small sample, or
      ii. large sample (>1000).
2. Existing analysis (usually for a nominal design or deterministic DSO DoE)
   (a) set-up:
      i. full parametric automated workflow (parametric in design variables with efficient re-meshing and/or morphing capability),
      ii. full parametric automated workflow (parametric in both design variables and uncertain parameters with efficient re-meshing and/or morphing capability),
      iii. workflow with automated connectivity between processes and automated meshing, but without a parametric CAD model, or
      iv. workflow where meshing process is not automated or is prohibitively expensive,
   (b) cost:

[1] Trainer et al. [293] recently highlighted the benefits and difficulties associated with CMM, CAD and computer aided manufacturing (CAM) data exchange within the supply chain.

i. very cheap (thousands of Monte Carlo samples (code runs) are practical),
ii. cheap (tens/hundreds of Monte Carlo samples are practical),
iii. expensive (Monte Carlo samples are impractical but hundreds of samples are possible), or
iv. very expensive (only tens/hundreds of samples are practicable),

(c) type:

i. finite element or fluid mechanics analysis software containing intrusive stochastic analysis code,
ii. FEA or CFD with no built-in stochastic analysis capability, or
iii. a workflow using multiple analysis software components with no intrusive uncertainty propagation capability.

It is assumed that analyses are of sufficient fidelity that the analysis is capable of determining changes in performance due to uncertainties of the scale measured.

The ideal scenario for UP (without RDO or RBDO) would be a scenario where thousands of components are produced, all of them measured, and all the 'true' geometries implemented in analysis, providing a direct Monte Carlo sample (section A.5) of the manufactured uncertainty space, and resulting in an unbiased estimate of the performance distribution and statistical moments. This scenario simplifies the process to UP only, as shown in figure 3.6.

In this case, the sample is random by necessity but the process by which the geometric changes are implemented might be through a number of alternative methods. A very cheap analysis code is required, or a large amount of time available, in order to run the thousands of analyses, and this ideal precludes any design optimization as there is no characterization of the geometry as a variation of anything but the original design. Figure 3.7 shows the options available for fitting the data (characterization) and implementing changes to the analysis CAD or mesh. The green processes are performed using methods in chapter 2. The grey processes require manual intervention.

If an automated process is required, then the use of parametric CAD is not possible if the meshing process is not automated, or if the workflow is built with software where mesh morphing is not integral. Re-meshing can also introduce additional noise. The alternative mesh morphing route does not provide new design geometries for future use, but avoids the potential noise and additional expense of re-meshing. It may also be the only feasible approach if the meshing process is not automated.

In truth, neither analyses nor measuring processes are often very cheap so this ideal case is not yet, to the author's knowledge, a reality. The reality is one of limited data, expensive analyses, and a desire to undertake full RDO. To do so, samples are required from multi-dimensional design or uncertain spaces. The process remains impracticable if

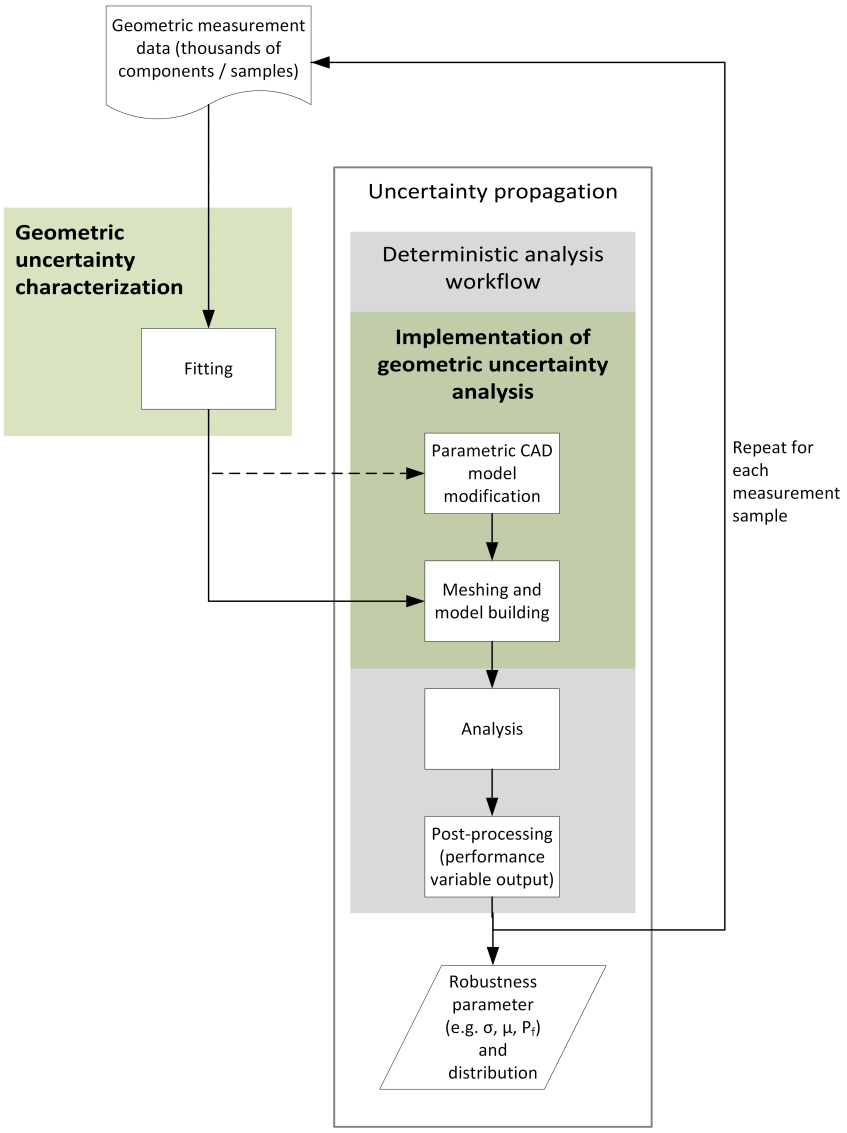


Figure 3.6: Ideal UP workflow (a direct Monte Carlo sample)

these hyperspaces are built from too many dimensions. Therefore, the characterization process must include the fitting of the geometry itself, along with careful parameter selection, and statistical model generation. If the data are simple parameters then this process only requires statistical distribution fitting, but correlations must be considered. There is little variation in the approaches taken for this, but independence is regularly assumed. If, however, point cloud data are available this process can be performed in a number of different ways, figure 3.8[2]. This figure adds the variable reduction and statistical modelling steps (in orange) that are not required for large data sets with inexpensive analyses. It illustrates the possible approaches in the circumstances in which

[2]It is assumed that the CMM data samples are made randomly, which is likely if the measurements have been made during routine inspection, given that inspection processes are often designed to ensure a distribution of samples randomly over time and manufacturing equipment. One could imagine a scenario of integrated inspection and design where the inspection programme was, at least in part, controlled by robust design requirements.

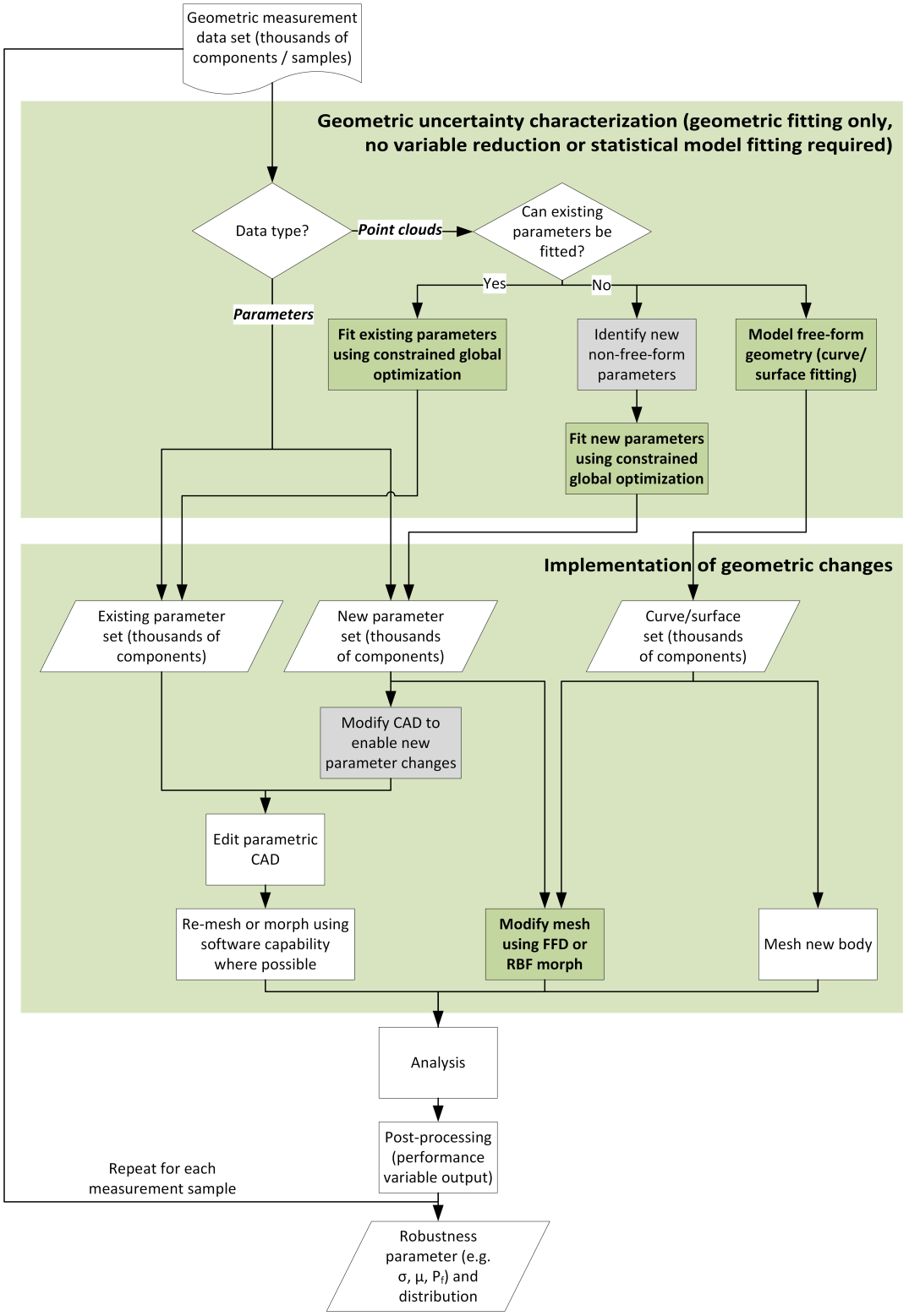


Figure 3.7: Fitting (characterization) and implementation of geometric changes for UP (where all parts are measured and thousands of analyses are possible)

we are most interested, showing methods that use existing CMM data as an input to uncertainty analysis and can be applied to any existing mesh-based analysis of suitable fidelity.

Although the "accurate characterization and efficient computation of geometric uncertainty remains an open problem" [126], there are some examples in the literature, from a variety of different research communities.

- O'Hare [191] successfully registered a U.S. patent for data reduction in coordinate measuring systems. The patent follows a similar workflow to that of figure 3.8 but maintains that segmentation is required prior to reduction and is non-specific regarding either algorithm except to suggest that methods can include a "Gaussian filter", "extracting uniformly spaced points along specified cross-sectional profiles", and "best-fit" and "least-squares" operations. The patent by O'Hare [191] refers to user selection of parameters and algorithms and points out that the benefit is to "help reduce the data burden associated with scanning technologies". Such a document is suggestive of one of the difficulties faced by designers: that prior processing within software associated with the CMM hardware can disguise, or even remove, uncertainties in which the designer is interested.

- Joskowicz et al. [126] approaches the question of uncertainty from the perspective of tolerances. They suggest the modelling of uncertainty in point locations as "approximate uncertainty zones" using a "linear parametric model". Given that the uncertainties are based on tolerance bands, and are a further approximation to these, this method appears to lack the statistical rigour of approaches using geometric fitting with algebraic parametrizations.

- Garzon and Darmofal [88] create a model of aerofoil geometry by direct subtraction of the measured points from equivalent points on the nominal shape. No reference was made to a 'registration' of the points via least-squares fitting or otherwise. They investigate the reduction of the variables by PCA (section 2.10) applied to 206 variables (103 points in two dimensions). The UP is applied using Monte Carlo sampling (with 2000 trials taking one hour) and a number of performance outputs assessed. They find that, although six parameters (modes) appear to represent over 99% of the geometric variance, 15 modes are needed to model 99% of the performance variance. A finding that the work in chapter 6 might go some way to explaining.

- Ray et al. [227] represented measured surface deviations in turbine rear frame struts using an interpolating RBF model in order to identify sensitive regions of the geometry before employing FFD in sensitive regions across 12 alternative design configurations.

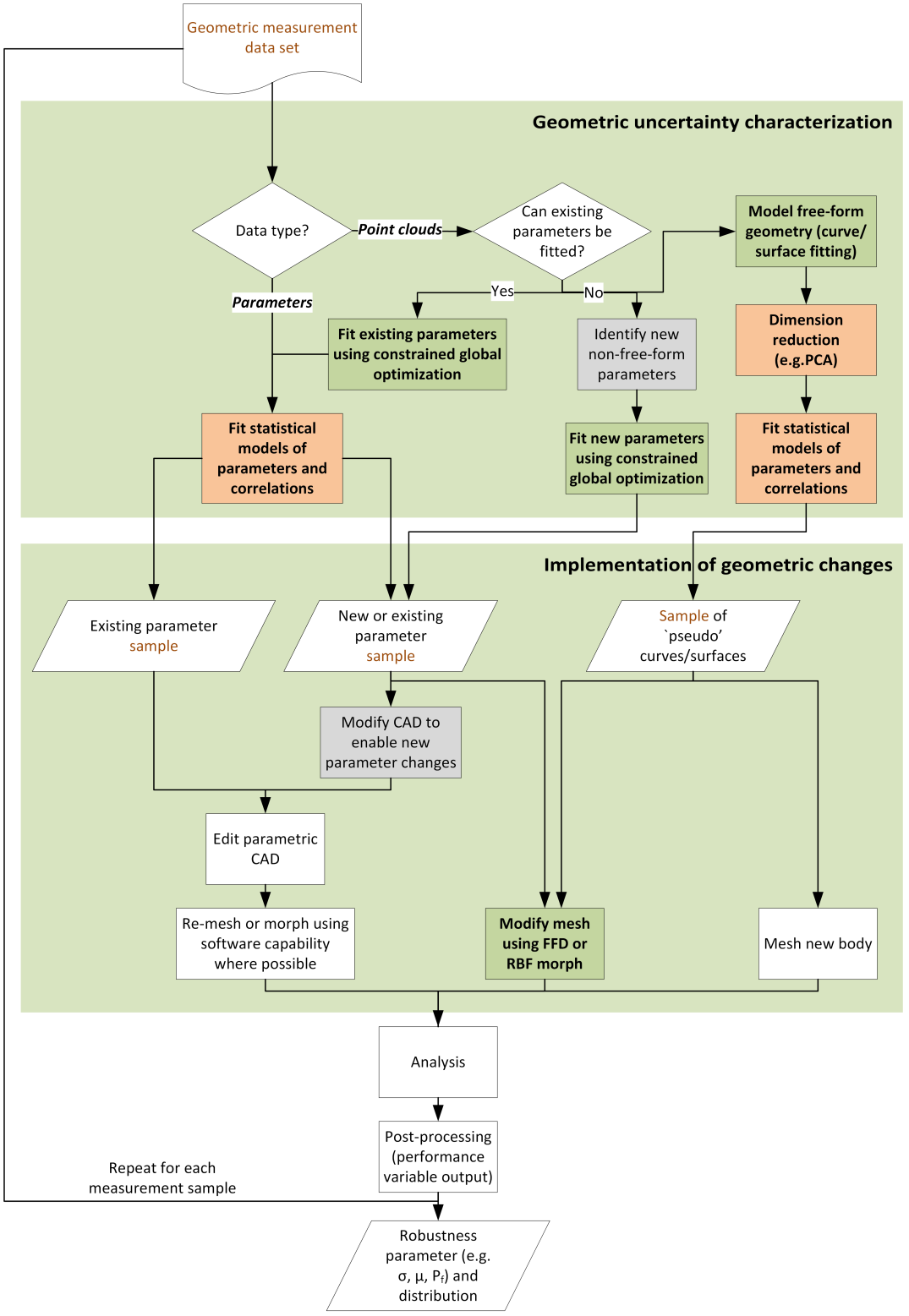


Figure 3.8: Characterization and implementation of geometric changes for UP (where a sample of parts is measured)

There are also researchers who are considering uncertainty from alternative perspectives, outside of engineering design, who are also applying modelling approaches in the face of uncertainty.

- Mangado et al. [161] provide a comprehensive review of uncertainty propagation approaches, including characterization of the inputs, in biomedical engineering, including a table to case studies within the literature and the types of characterization and models used. They point to the work of Baldwin et al. [9] who applies PCA and mesh-morphing based on a small sample of scanned knee joints. The work in this thesis could be applicable in this field, although measurement sample sizes may be more limited.

- Chilès et al. [39], for example, use co-Kriging where the covariance is identified from the structural data to model geological potential fields and quantify the uncertainty in the resulting models. Such uncertainty is analogous to the measurement error in our problem.

- Poniatowska [214] models free-form surfaces based on CMM data in order to identify and remove systematic machining errors (the machining pattern model, MPM). The fitting routine presented uses NURBS to fit the surfaces and works on the premise that the model "should not contain any effects of random phenomena..., while the model residuals should not include the systematic component". The iterative routine tests for normality of the residuals using the Kolmogorov-Smirnov (KS) statistical test, and uses *Moran's I statistic* [64] to ensure spatial randomness. This is essentially a constrained local downhill search. Poniatowska [214] assumes that the CMM measurement error is not significant.

Figure 3.9 splits 'characterization' and 'implementation' into generic procedures, and shows an overview of the methods that other researchers, including those discussed above, have applied to problems with a number of differing data types, analysis features, and overall goals.

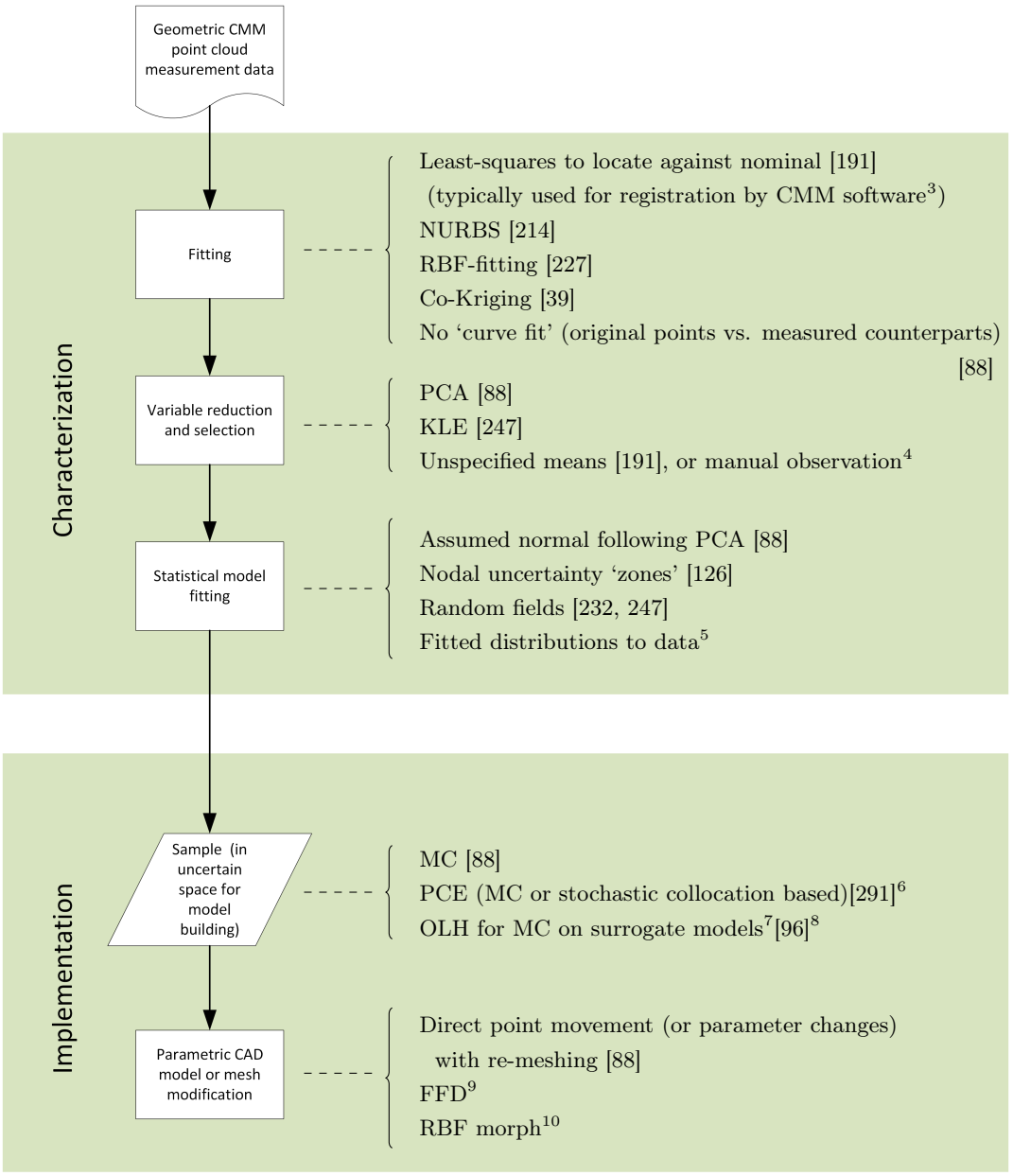


Figure 3.9: Examples of non-intrusive methods used to characterize measured geometry and to apply it to existing analysis

[3]Details of methods used within proprietary software packages, including Zeiss ALYPSO[324] and Mitutoyo COSMOS[179] for example, are not publicly available. However, a paper written to assess software accuracy by [301] stated that "all" of the packages tested in their work "were based on the Gaussian least-squares method". Individual companies and users can also implement their own proprietary processes using coding interfaces within some software.

[4]Rozza and Manzoni [240] refers to mappings on sub-domains which can be built "by hand" for "geometrical reduction".

[5]Ross and Cheah [238] assess the benefits of fitting distributions to data sets in order to improve UP estimates of product life-cycles. They conclude that use of data provides "opportunities to better characterize data and increase confidence in the results", when compared with the use of assumed "default" distributions. However, no examples of similar studies have been found applied to engineering design problems.

[6]Tootkaboni et al. [291] apply PCE to propagate uncertainties in design variables in a topology optimization.

[7]Rajabi [222] compare the results of PCE and surrogate based propagation in the prediction of behaviour of groundwater systems.

In the remaining chapters we present some inspection data sets representing 'real' variation in geometry and the means by which the methods in chapter 2 have been applied using the logic shown in figure 3.8 and an appropriate means of implementation as shown in figure 3.7.

[8]Gul et al. [96] use emulators to model local regions (the uncertain space to which we refer in this thesis) of the response in order to predict uncertainties in machining parameters due to variation in tool geometry and material properties. They refer to the use of these uncertain space models as "in-situ emulators".

[9]Examples in the literature for the use of FFD for measured uncertain geometry have not been found. However, the approach is often applied to RDO (sections 2.11 and 2.5).

[10]Examples in the literature for the use of RBF morphing for measured uncertain geometry have not been found but Petrone et al. [205] apply the ANSYS RBF morph to the robust design of a Formula 1 front wing in the face of uncertainty in the steering and tyre geometries.

Chapter 4

Uncertainty characterization using existing parameters

In this chapter, an approach has been developed for applying coordinate measuring machine point cloud data for firtree and dovetail slot flanks (figure 4.1) in an uncertainty framework. Data analysis and fitting have been applied to a true data set, resulting in a statistical description of the flank angle (an existing design parameter): the **characterization** step highlighted in figure 1.4 and shown in greater detail in figures throughout chapter 3. Given these characterized parameters, and an existing analysis code, UP and DSO, as described in sections 2.4 and 2.5, could then be applied to analysis chains to produce realistic predictions of uncertainty in performance (here, expected disc life).

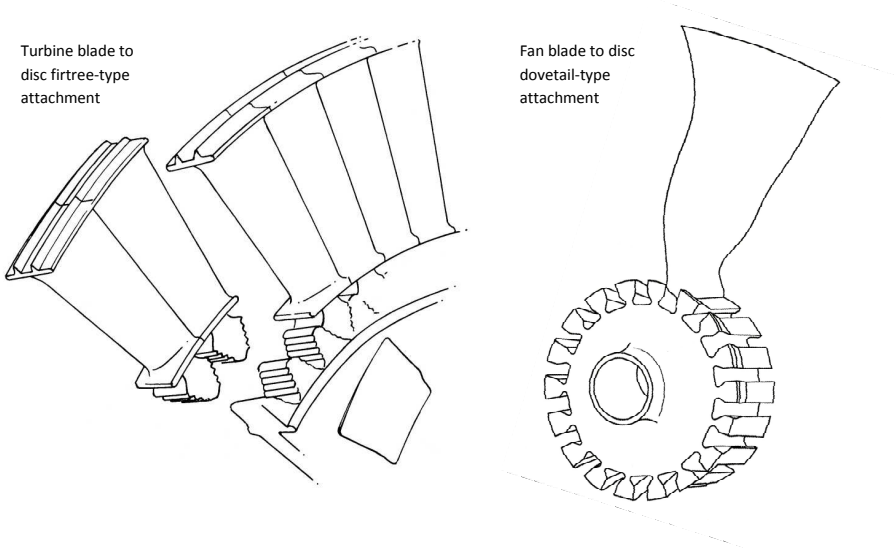


Figure 4.1: Images illustrating firtree and dovetail type attachments to turbine and fan discs, images from Dailey, G. M. (Rolls-Royce plc., Derby, GB) [45] and Ventura and Klinetob [299]

4.1 Problem description: firtree flank angle uncertainty

Whether in the fan or turbine, the attachment of each of the blades to the supporting disc is critical. Single blade loss is not desirable but can be contained, however disc failure could be catastrophic to an engine or aircraft as a whole. Discs are therefore inspected and replaced periodically during service. Increasing both their total life prior to replacement and the allowable period between inspections would be hugely beneficial. In addition, both blades and discs are expensive to produce; a reduction of scrappage and rework could result in significant savings. The geometric shape of the joint, typically a *firtree* or *dovetail* (figure 4.1), is critical to its performance. These designs rely on the contact between angled *flanks* to restrain the blades and transfer load from centrifugal forces[1]. The notches are radiused to minimize stress concentrations where failure initialization usually occurs. However, manufacturing processes and service conditions at elevated temperatures may result in uncertainty in the disc slot and blade joint shapes affecting component performance in terms of expected life.

A set of CMM data for broached and shot peened firtree-shaped slots in high pressure turbine discs was provided by Rolls-Royce plc. It is suggested by experienced designers, and demonstrated in a study by Deshpande et al. [55], that the shape of the firtree pressure flanks, and in particular their angles, is critical to disc performance in terms of

[1]The load on one 10cm long turbine blade attachment can exceed 18 tonnes; the weight of a fully laden double decker bus, in temperatures of up to 1600°C. The much larger fan blade, although moving more slowly than the turbine blade, still causes large centrifugal loads on the attachment but over a much larger area and at the greatly reduced temperatures of the ambient air.

expected life. Appendix B describes the means by which life is predicted, and the critical performance parameter, the peak principal notch stress.

68 slots have been measured over twelve discs with a touch-based Mitutoyo CMM with Renishaw SP25M probing system, figure 1.5, chapter 1. Each slot has four measurement sets over the cross-sectional shape (front and rear, left and right). Each data set is a point cloud given in the Cartesian coordinate frame with origin at the centre of the disc and each slot at 90 degrees to vertical. There are three pressure flanks per data set (a total of 408 flanks measured twice each (front and rear)). The CMM probe uses a 0.5 mm diameter ruby stylus and the scanning increment is set in the region of 20 to 30 μm. The resolution for this probing system is claimed to be less than 0.1 μm by Renishaw [231]. An illustration of the point cloud for a full slot is shown in figure 1.6, chapter 1. A single data set (of artificially generated variations to dummy geometry) is shown in figure 4.2.

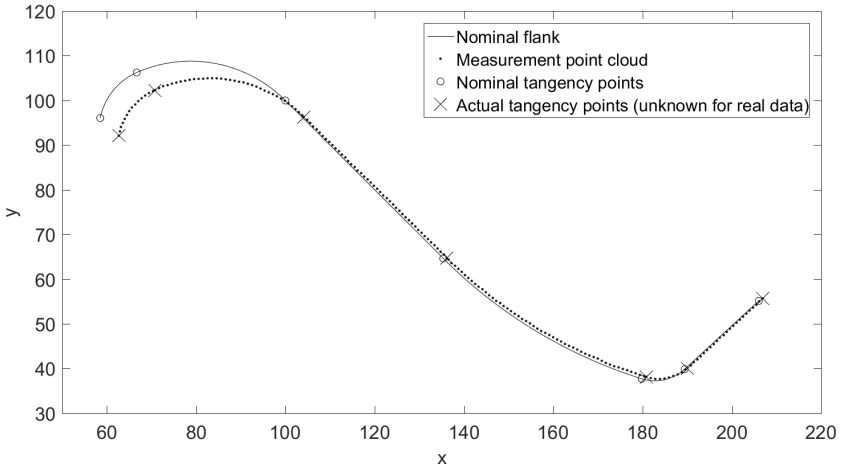


Figure 4.2: Example disc firtree slot measurement data set (artificially generated data and geometry) for a single pressure flank with nominal geometry shown for comparison

Point cloud fitting routines typically assume random error in the measurements and, to be of any use in assessing performance variance, the error must be sufficiently small compared with form variation. A number of authors, such as Weckenmann et al. [306] and Dowling et al. [61], discuss the range of potential errors through measurement. Erkan et al. [63] reviews errors introduced through CMM tip radius correction software which, although potentially significant, are shown to be in the order of 1.0×10^{-6}mm for Mitutoyo software using a similar probe radius and scanning increment. We can surmise from the literature that tight process and environmental controls and calibration, combined with highly developed hardware and software, make these assumptions reasonable. Here, we aim to define a fitting routine for pressure flank data that is:

- efficient in its implementation,

- sufficiently accurate to capture the underlying geometric uncertainty,
- fits appropriate parameters.

4.2 Characterization of uncertain parameters

A standard approach to this type of problem is to first select the type of analytical definitions for the curve segments, determine which points belong to which curve segment, and then apply a fitting process to the individual segments, figure 4.3.

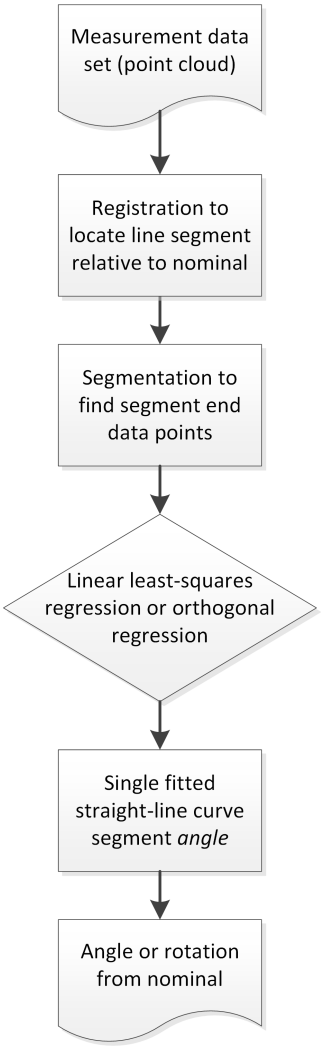


Figure 4.3: Workflow: a typical fitting process to determine the angle of a straight-line curve segment

By inspection, the data (figure 4.2) appears to represent a straight-line pressure flank with tangent arcs at either end, in line with the nominal design intent (a straight line 'flank' with a tangent arc at either end). The nominal design is also shown in figure 4.4. There are additional features present beyond the contact flanks and their tangent arcs.

These additional arcs and lines are included in the measurement data, however in this study we assess the effects of variation in contact flank, and the flank angle in particular.

Segmentation with common tangents is notoriously difficult and inaccurate, or even impossible (section 2.7.2). In addition, intuitively, if any points from the arcs are included through inaccurate segmentation then their positions could have a disproportionate effect on the line angle prediction and we would be relying on the segmentation to determine the line length. Therefore, in this case the arc-line-arc segment (A-B-C-D in figure 4.4), assuming a tangency constraint at both line endpoints, is fitted in a single process akin to orthogonal least squares regression (section A.7.2): the minimization of a root mean squared error (RMSE). In this case, the combined arc-line-arc segment is defined in two dimensions by the arc centres ((x_1, y_1) and (x_2, y_2)) and corresponding radii (r_1 and r_2), figure 4.4 (6 parameters). The parameter of interest: flank angle (α) can then be calculated from these values geometrically, along with additional parameters of interest such as flank length (l).

This six-variable optimization is performed using an evolutionary particle swarm based global-local search algorithm (described in section A.4.2, and by Toal et al. [289] for kriging hyperparameter tuning).

Unfortunately, the segmentation problem is then only deflected to the ends of the arcs, instead of the ends of the straight line. The questions, "how many of the data points, and which ones, should be included to minimize fitting error?" needs to be answered.

One wishes to include the maximum number of relevant points whilst ensuring that points from the next tangential geometric segment boundary are not included. We use a single data set to test the RMSE fitting approach using a nominal geometry with prescribed orthogonal random perturbation of the order of the observed variation. The number of data points included for fitting is plotted against the resulting angle discrepancy from nominal in figure 4.5. The 'true' flank angle variation, in this case is known to be zero. Figure 4.5 illustrates that the fitted flank angle is most accurate (there is no angle change introduced by the fitting process), not when only points expected to be within the region of the nominal flank itself are measured, but when at least 10 points within the tangent arcs are also included (circles, rather than crosses), i.e. direct regression of the segmented straight-line curve section introduces error (crosses). Intuitively, as shown in this test, extending too far (into additional geometric features) results in the accuracy of fit being further compromised ('plus' markers). Of course, if we can be sure exactly which points belong to which segment then the result is also correct (the star).

It is not possible to simply define x and y coordinate values of the extents based on the nominal design. This is because the true measurements can deviate considerably from the nominal position by an unknown magnitude or direction. Accurate registration (section 2.7.2) of the curve would first be required; again notoriously difficult when curve or surface segments are tangent.

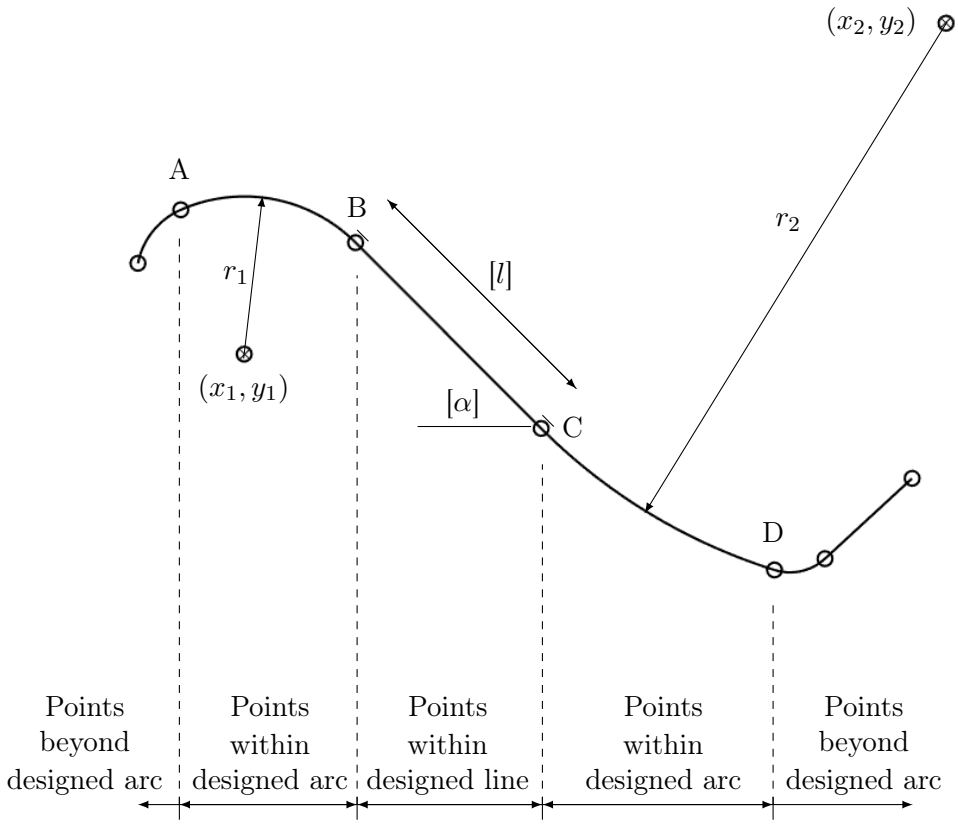


Figure 4.4: Firtree slot pressure flank cross-sectional definition, showing defining parameters, [derived parameters], and data point regions

In conclusion, the plot, figure 4.5, shows that, for curves joined at tangents, whatever the registration and segmentation process, error will be introduced during the RMS fitting process, unless additional segments either side of the segment of interest are included in the fitting process. In addition, further points beyond these segments that are not included in the fitted definition, must not be included.

Here, we therefore propose extending the global optimization to include the end point locations (to define the extents of the data set) segmentation and fitting can be combined, and the requirement for registration removed. By including arc segments either end of the flank line of interest during the fitting process, greater accuracy can be guaranteed.

The proposed method is to increase the number of variables in the optimization to eight. The extents of the arc-line-arc segment are defined as two distances along the curve from the first point (at 1 unit) to the last point at n units, bounded by the nominal extents of the straight line plus a tolerance. This approach increases the expense of the optimization but is designed to ensure that the variance in fitted parameters, particularly flank angle, is as small as possible. The global optimization routine should be able to

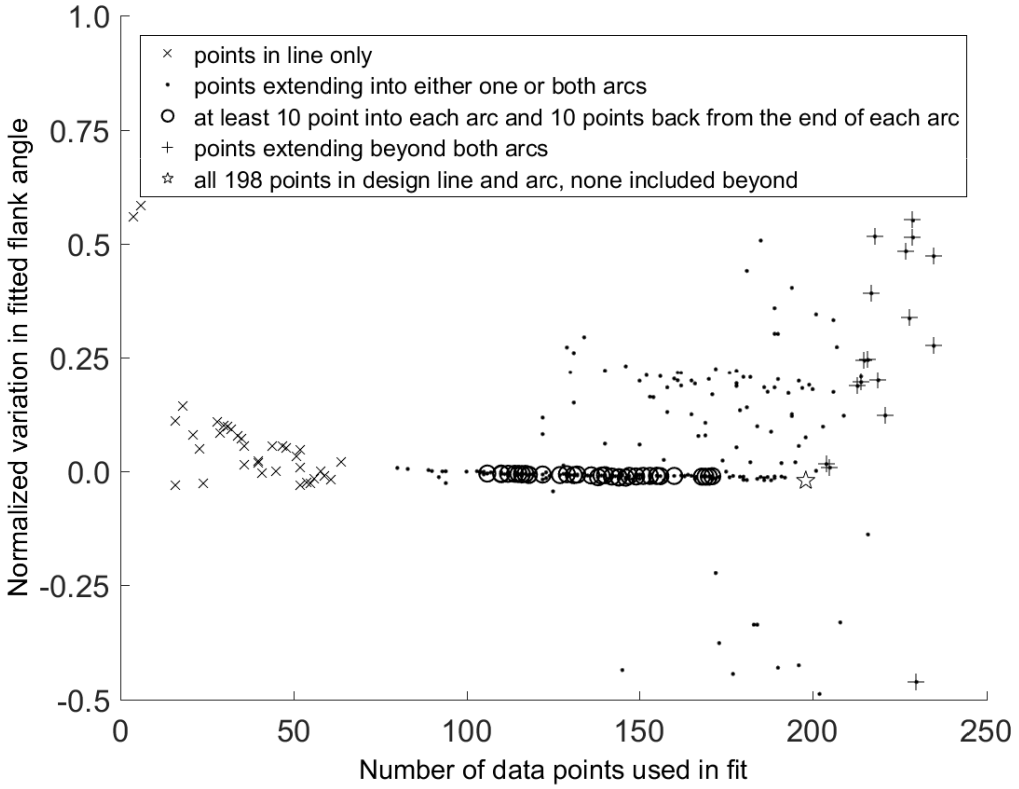


Figure 4.5: Number of data points used for fit vs. predicted angle for a single pressure flank, illustrating variation of fitted angle with point selection

ensure convergence in the optimization and deal more easily with the increase in variables than local routines. Figure 4.6 shows the proposed workflow.

The workflow performs:

- global evolutionary optimization using a particle swarm based optimizer, section A.4.2, and uses the
- orthogonal root mean squared error calculation (section A.7.2) as the objective function.

This approach is tested on an artificially generated data set created by displacing and perturbing points on an example nominal curve segment, figure 4.7.

Using an artificial point cloud based on the true geometry, an offset of 0.1 mm is introduced in both the x and y directions. In this case, the proposed fitting routine results in a predicted angle that differs from nominal by 0.0063 degrees. This process is repeated 20 times with an offset randomly drawn from a uniform distribution in the interval $[-0.13, 0.13]$ in both x and y. The mean predicted error over the 20 fits is 0.0024°. The convergence of the particle swarm optimization (PSO) algorithm for seven of the fits is

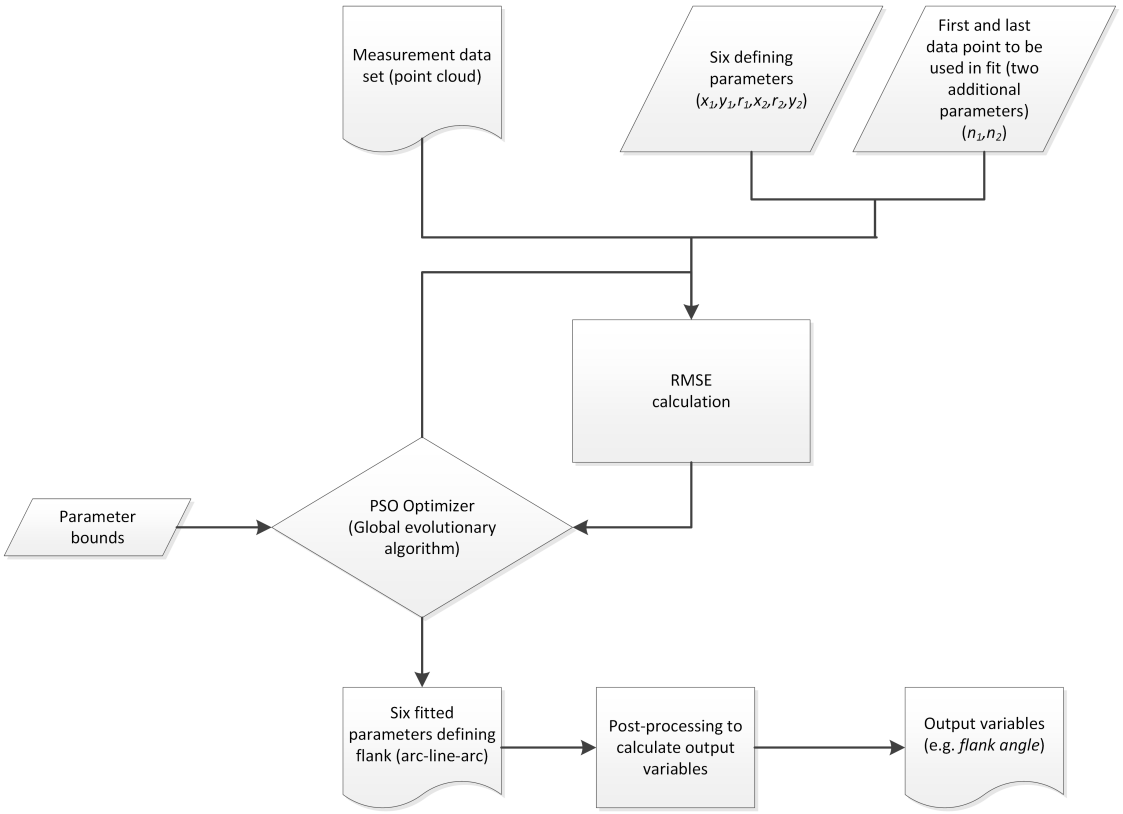


Figure 4.6: Workflow: the combined segmentation and fitting process for a firtree or dovetail contact flank

given, figure 4.8. Convergence is reached for all 20 data sets well within 3000 evaluations (60 generations of population size 50), therefore this number of generations is chosen for all further fits in order to minimize expense.

To compare the proposed eight-variable approach with the typical registration-segmentation

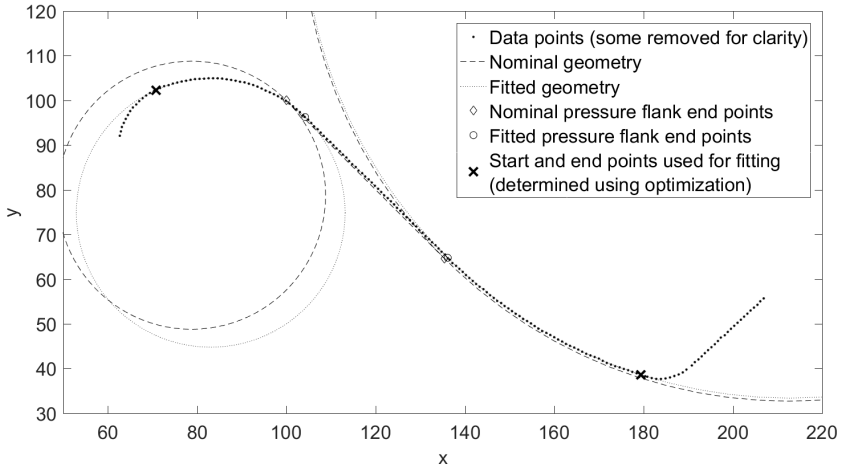


Figure 4.7: Scatter plot showing an artificially generated data set with normally distributed perturbations fitted using an eight-variable optimization procedure

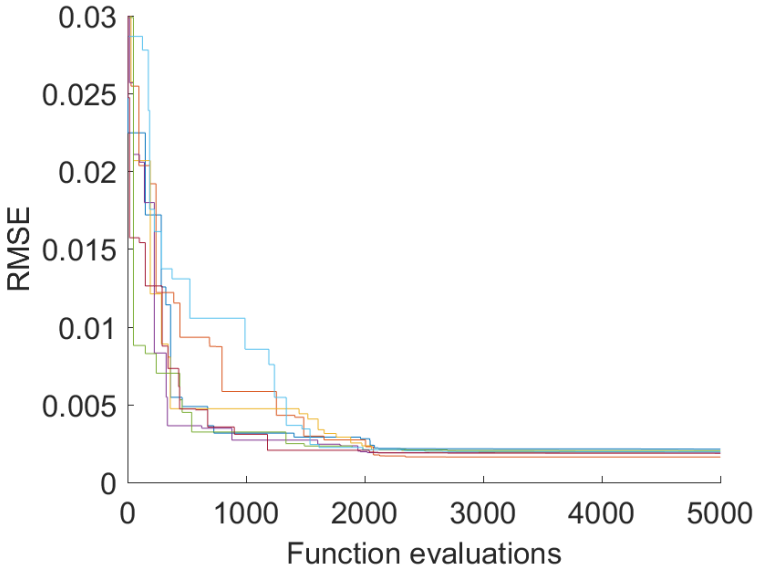


Figure 4.8: Convergence plots of the particle swarm optimization used to fit pressure flank angle using eight variables (convergence shown for seven fits out of 20 for clarity)

type approach (figure 4.3), 120 sets of points are generated with a random angular variation distributed $\mathrm{N}(0, 0.5^2)$ degrees and a random translation distributed $\mathrm{N}(0, 0.05^2)$ mm. Five alternative approaches are compared:

(i) orthogonal regression fitting of the line only based using region between designed line end points,

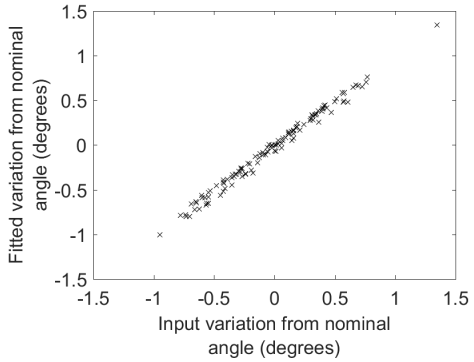
(ii) orthogonal regression fitting of the line and arcs using the designed arc end points (arcs only fitted to locate and segment the straight line segment),

(iii) orthogonal regression of the designed line only for registration (data re-centred on design line), followed by fitting as (ii),

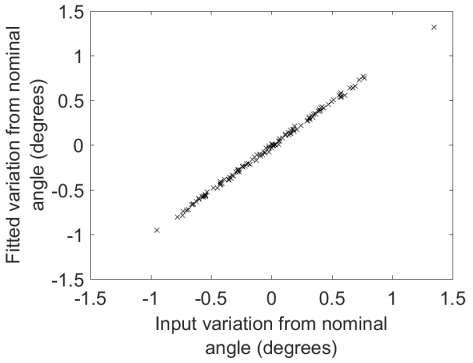
(iv) orthogonal regression of the designed line only for registration (data re-centred on design line), followed by orthogonal regression fitting using half of the points within the designed arcs, and

(v) the proposed combined eight-variable optimization.

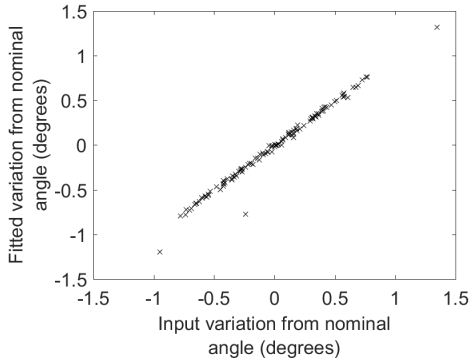
Figure 4.9 plots the actual (known) angle variation against the fitted angle using four typical approaches. Figure 4.10 shows the same plot for the proposed optimization approach. Figure 4.11 plots the variation in straight line flank angle output from each experiment, compared with the histogram in the known error of the fitted values. The same plot for the proposed method is given in figure 4.12.



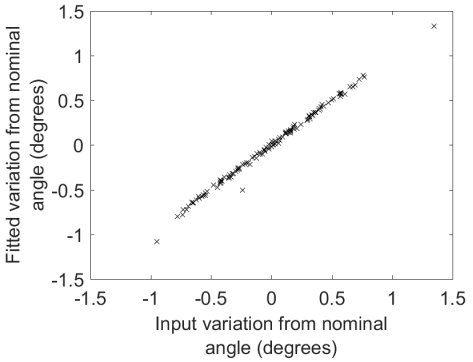
(a) Line only orthogonal regression (i)



(b) Arcs included for registration and segmentation, orthogonal regression (ii)



(c) Registration and fitting by orthogonal regression, full designed arcs (iii)



(d) Registration and fitting by orthogonal regression, half way into designed arcs (iv)

Figure 4.9: Input variation in angle vs. fitted variation in angle for 120 artificially generated data sets with randomly selected input flank angle (0.0 degrees = nominal)

The errors from fitting the known input distribution to the output distribution, are tabulated in table 4.1. There is closest agreement between the input angle and fitted angle from the combined approach. A statistical Kolmogorov-Smirnov (KS) test (appendix A.8) confirms the null hypothesis that 'the input and output angles are randomly drawn from the same distributions' cannot be rejected for any of the fits. However, the ks test statistic shows closest agreement from the proposed method (iv). The error is in the order of 3% of the measured angular variation for the proposed approach.

Table 4.1: Fitting errors and ks-test p-value and statistic values for straight line angle fits for point cloud measurements of a straight line with tangential arcs, using a selection of orthogonal distance based approaches

	Type of fit				
	(i)	(ii)	(iii)	(iv)	(v)
Maximum error	0.1384	0.0560	0.5374	0.2710	0.0517
Mean error	0.0206	0.0156	0.0177	0.0018	0.0007
Standard deviation of error	0.0451	0.0160	0.0560	0.0327	0.0178
ks-test p-value	0.998	0.999	0.999	0.999	0.999
ks test statistic	0.050	0.042	0.042	0.033	0.025

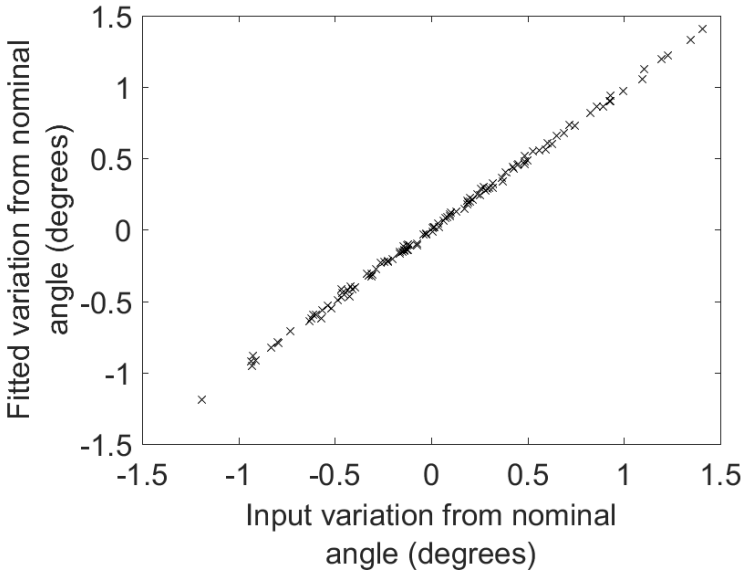
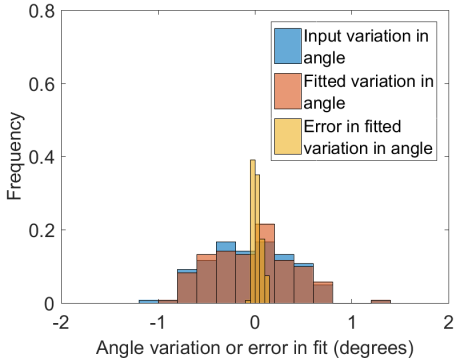
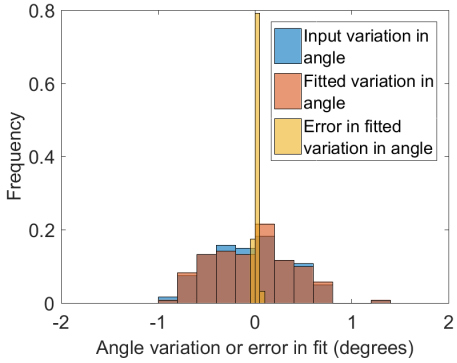


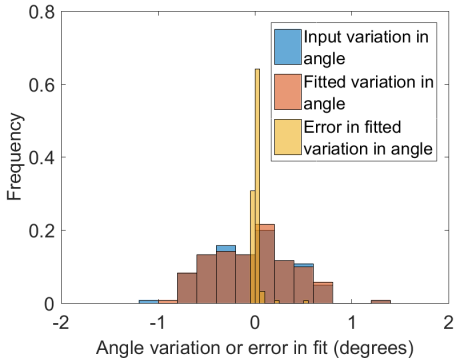
Figure 4.10: Input variation in angle vs. fitted variation in angle for 120 artificially generated data sets with randomly selected input flank angle (0.0 degrees = nominal), combined eight-variable optimization based fitting (v)



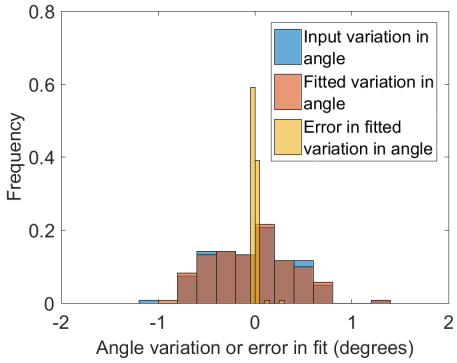
(a) Designed line extents only orthogonal regression (i)



(b) Arcs included for registration and segmentation, orthogonal regression (ii)



(c) Registration and fitting by orthogonal regression, full designed arcs (iii)



(d) Registration and fitting by orthogonal regression, half way into designed arcs (iv)

Figure 4.11: Histograms of flank angle showing input angle vs. error in fitting for that input set

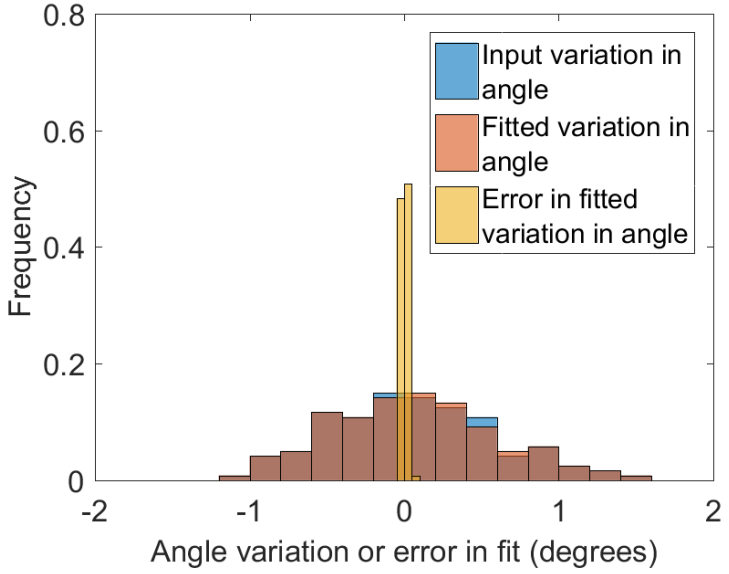


Figure 4.12: Histogram of flank angle showing input angle vs. error in fitting for that input set, combined eight-variable optimization based fitting (v)

This combined registration-segmentation-fitting and optimization based approach has also been applied to a data set supplied by Rolls Royce plc.[2]. Fitted parameters can then either be used directly to reconstruct and analyse individual manufactured disc slots, or the set as a whole can be statistically interrogated and distributions drawn for the parameters of interest. For these data, in order to examine the distribution of flank angles either the clockwise rotation from nominal or the measured angle from nominal slot centreline could be considered. Histograms of both for all measured flanks are given, figure 4.13. The sample statistical moments are also displayed for comparison, table 4.2[3].

Table 4.2: Sample statistical moments and range of fitted angles and rotations of pressure flanks for full data set (normalized for confidentiality)

	Rotation about firtree axis	Rotation about contact flank
Mean	-0.0150	0.0076
Standard deviation	0.1081	0.1089
Skewness	-0.3568	0.1543
Kurtosis	3.9657	4.0879
Range	0.8618	0.8352

A normal probability plot for each, figure 4.14, confirms that both sets of data closely follow a normal distribution close to the mean. However, we observe a deviation in the tails. There is a slightly higher chance of an increase in angle over nominal and a slightly lower chance of a decrease in angle than would be expected with a truly normal distribution (confirmed by a positive skewness, table 4.2). A greater chance of negative

[2]Details of the raw data are confidential but normalized results are discussed in this chapter.
[3]Skewness and kurtosis are corrected for bias [164].

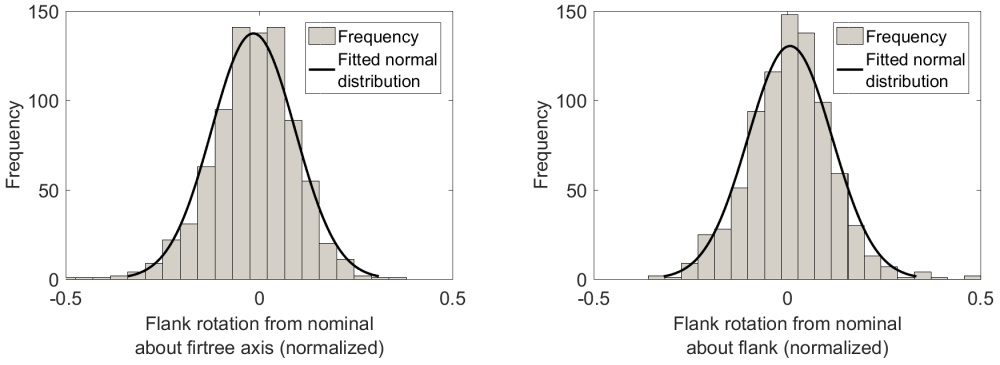


Figure 4.13: Histograms of fitted rotations from nominal for full data set (normalized for confidentiality)

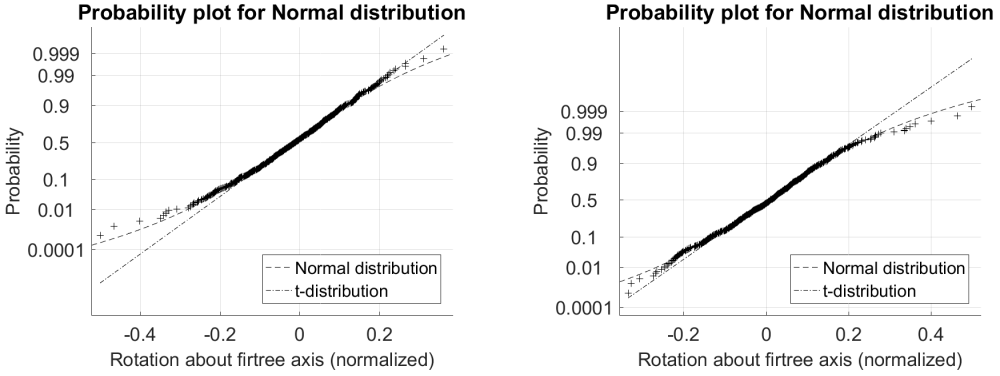


Figure 4.14: Probability plots of fitted rotation for full data set (normalized for confidentiality)

rotation and lower chance of positive rotation are also shown (confirmed by a negative skewness, table 4.2). Both distributions exhibit a peakiness, or tendency to have more outliers, greater than that of a normal distribution, i.e. greater than three. We note, therefore, that an input distribution for uncertainty propagation of a normal nature could be overly optimistic in the tails. A t-distribution is heavier in the tails, and is shown to be more appropriate in figure 4.14.

A statistical z-test is performed on the data with the null hypothesis that, assuming a normal distribution with standard deviation equal to the sample standard deviation, the data comes from a distribution with mean equal to the sample mean. For both rotation and angle, it is confirmed that the 95% confidence interval for the mean contains the sample mean (the hypothesis cannot be rejected at the 5% level). However, if we hypothesize that the data comes from a distribution with mean equal to the nominal then the hypothesis can be rejected at the 5% level; we observe a shift from the nominal design. A KS test is also performed to test the hypothesis that the sample has been drawn from a fitted t-distribution. This hypothesis cannot be rejected at the 5% level for either rotation or angle.

4.3 Implementation of analysis with fitted parameters

The distributions (fitted uncertainties) that have been ascertained could now be applied in a robust framework as demonstrated in chapter 2 through uncertainty propagation and robust design of a dovetail joint. Geometric changes in an existing analysis workflow can simply be made by implementing changes to the existing parameters of the design geometry prior to any analysis mesh being constructed. Alternatively, geometry manipulation can be made to a meshed geometry. Such approaches are investigated further in chapters 5 and 6, and applied using linked CAD and analysis software in sections 2.3, 2.4, and 2.5.

4.4 Contributions

It has been shown that direct regression of segmented curves bounded by tangential joins to surrounding segments, can yield inaccurate values for the measurand: the straight line angle in this example case. A method by which tangency constraints can be combined with a traditional least squares approach, and enhanced by simultaneous fitting of the adjacent curves, has been implemented. The method uses additional location variables alongside a particle swarm based global optimization to eliminate the need for an additional registration and segmentation process. With the use of an artificially generated set of curve segment data sets, it is demonstrated that error is reduced with the use of the proposed approach. The method has also been applied to a pressure flank angle fitting problem using 'real' data sets.

The approach could be taken for any parameterized composite curve or surface point cloud measurement set for individual features where the constraint between segments is known, and is of particular use if the segments are tangential. If, however, it is thought that the tangency constraint may not be met then further investigation would be needed, and perhaps a search for step changes in orthogonal error during the fitting process would highlight lack of tangency. We refrain from investigating such a situation here given that our true data sets do exhibit tangency.

This chapter is an exemplar of good practice for obtaining accurate distributions of design parameters given measured uncertainty from manufacture, and it has highlighted pitfalls associated with segmenting point clouds in this context.

Chapter 5

Implemention of geometric uncertainty on an existing mesh

When routine measurements are not complex, but are measurements of existing design parameters, the **implementation** step in figure 1.4 can be a simple change to geometry models. This step appears as "CAD model construction and FEA analysis" in the DSO workflow, figure 2.10, or "performance analysis", figure 2.20, and can be applied by changing the original design model in the CAD tool and then re-meshing for each sample analysis or iteration. In some cases, however, this is either unsuitable, or not possible, because:

- a fully parametric workflow for DSO does not yet exist and only UP, not RDO is considered appropriate,
- small uncertainties mean that computational noise[1] introduced by re-meshing could mask effects of interest, or
- re-meshing for each run is prohibitively expensive.

In many cases expense is a driver, however lengthy the meshing process, because the cost of UP or RDO is inevitably high. The case study presented here is, in fact, subject to all three constraints.

Here, we present a new approach for implementing geometric changes in hole diameters directly within a CFD mesh, rather than through the CAD model; "multiple samples in uncertain space", figure 2.20. As with many performance predictions, the existing mesh has been constructed in a largely manual way and its modification is undesirable because of the many loading conditions for which multiple analyses have been performed

[1]Computer simulations can contain noise, or non-systematic error, often considered Gaussian in distribution, and attributed to numerical rounding and discretization.

over months and years. Understanding the effects of uncertainty using the existing mesh is preferable in this case, with the benefit of "isotopological meshes and one to one correspondences" between analysis samples [161]. In addition, the meshing tool used does not include any proprietary morphing facility linked to the geometry definition.

Here, a FFD-based workflow is set up, with further developments to this in order to deal with some complexities in surface shape. Although not all available data are in the form of complex point clouds, all analysis meshes are constructed as such, and maintaining the surface integrity and quality of the meshes is assumed paramount. The proposed approach ensures that this is the case.

5.1 Problem description: complex meshed holes

The combustor performance parameters of interest in the analysis are the peak near wall exit temperatures, not because they directly affect the combustor components, but because the expected variation in this temperature is a critical input parameter for the design and analysis of downstream components: the nozzle guide vanes (NGVs) and the turbine. Without an accurate statistical understanding of uncertainties such as these, pessimistic worst-case scenarios can result in over-design and under-performance.

The input temperature profile (*traverse*) needed for turbine nozzle guide vane (NGV) and blade thermo-mechanical design is uncertain. It is usually based upon the output *radial temperature distribution factor* (RTDF) profile[2] and near-wall *temperature distribution factor* (TDF)[3] predicted by the combustor computational fluid dynamics (CFD) analysis (using Reynolds-averaged Navier-Stokes equations (RANS)), or by experiment. Limited numbers of experiment are sometimes possible, but there is often a significant difference observed between experiment and analysis. RTDF (averaged) temperatures match well when compared with the sector TDF, which shows large discrepancies (over-prediction of TDF range) close to the walls (figures 5.1, 5.2, and 5.3)[4]. It is this range that is critical for the NGVs and turbine blades, with current analyses potentially resulting in over-design or an inability to meet design requirements.

It has been suggested that geometric variation in port and cooling holes due to manufacturing uncertainty may be responsible for some of the near wall variations. In addition, a process for routinely investigating the effects of changes in measured hole diameters

[2]
$$RTDF = \frac{T - T_{av}}{T_{av} - T_{in}}, \tag{5.1}$$

where T is the temperature averaged over the circumference, T_{av} is the area averaged temperature (nominal = 1449.3 K), and T_{in} is the inlet temperature (751.0 K).

[3]
$$TDF = \frac{T - T_{av}}{T_{av} - T_{in}}, \tag{5.2}$$

where T is the temperature at points around the circumference.

[4]These figures are illustrative purposes only. The results that they represent are confidential.

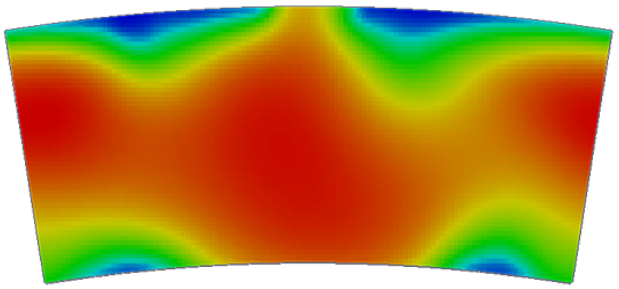


Figure 5.1: TDF contour map throughout analysis sector exit section, as provided by Rolls-Royce plc., red = maximum, blue = minimum temperature (scale removed for confidentiality)

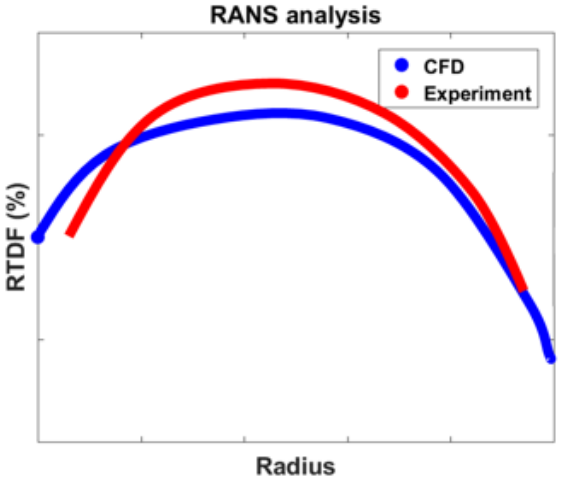


Figure 5.2: Representative sketch of RTDF (%) averaged over area, based on data provided by Rolls-Royce plc.

would be of use for concession analysis. The existing analysis workflow is given in figure 5.4, showing where geometric uncertainty can be included if mesh manipulation is used.

Inspection processes during manufacture provide data pertaining to the diameters of both the cooling holes and ports.

Cooling hole data

The measurement data for cooling hole diameters available for this study consist of a set of 1580 measured holes of 10 different nominal diameters, distributed through 12 different combustor liners. The holes are distributed around the liner in rows in a single sector of a liner (figure 5.5). Four holes are measured per row around the full liner at 90°, 180°, 270°, and 360°, with an additional two holes measured in the igniter sectors in the outer wall, bay 1.

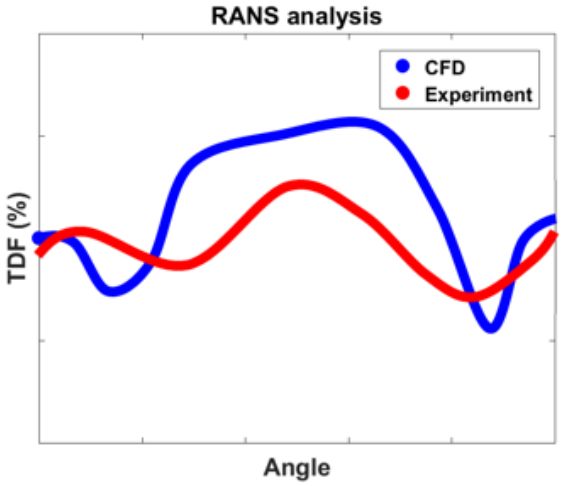


Figure 5.3: Representative sketch of TDF (%) at 5.92% of the radial extent, based on data provided by Rolls-Royce plc.

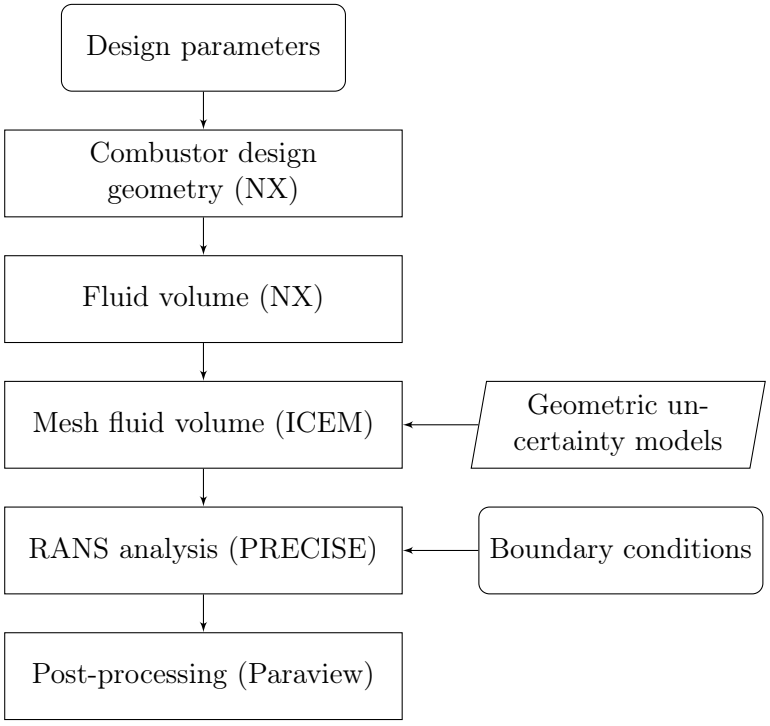


Figure 5.4: Flowchart showing the analysis chain for the performance assessment (flow characteristics including temperature) of the combustion system with the inclusion of geometric uncertainty at the mesh level

Measurements are taken using precision pins of discrete diameters. The holes are manufactured using laser drilling. It is understood that the laser calibration can drift, requiring periodic recalibration. The order that the holes are drilled is unknown but one would expect that the cutter works around the liner in rows. It is noted that some of the measured holes do not appear in the analysis sector, and that the analysis sector is half of one tile wide but that the tiles are offset from one another so do not align with the

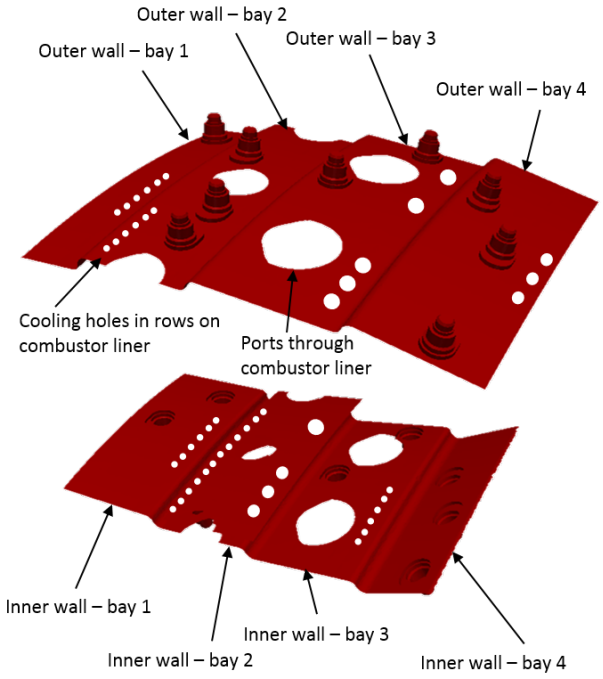


Figure 5.5: Sector of the combustor liners showing illustrative layout of cooling holes and ports in rows on liner bays

sector.

Port hole data

The ports are holes in the combustor walls and liners, sometimes referred to as dilution holes. Figure 5.6 shows an example of a port hole for the combustor of interest, but the surface shape varies depending on the combustor design. There are a limited number of measurements available for the port geometry, but this is an uncertainty that is expected to have a significant effect on temperature predictions. There are four types of tile: outer primary (OP), inner primary (IP), outer intermediate (OI), and inner intermediate (II). There are 40 measurements of port diameter for the OP, IP, and II ports and 44 for the OI tiles (10 or 11 measurements on each of 4 tiles).

Analysis of suitable DOEs determined by characterization of the diameter measurements can be used to provide useful insight into the expected near wall exit temperature variation due to cooling hole and port diameter uncertainty. It is hypothesised that these DOEs can be most readily facilitated through the implementation of FFD-based morphing of the CFD mesh.

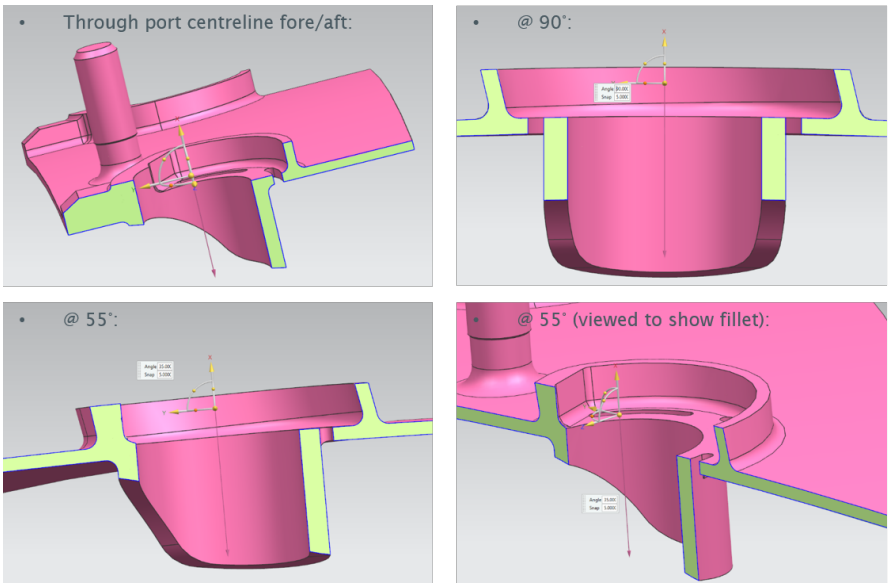


Figure 5.6: Port hole shape

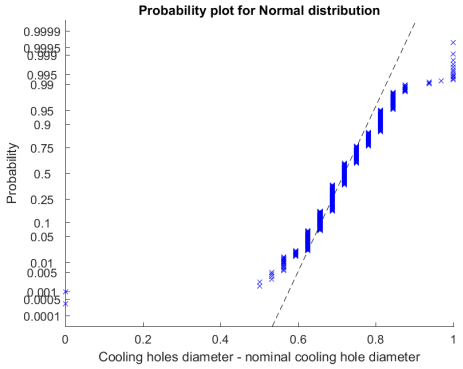
5.2 Hole diameter uncertainty characterization

There is no curve or surface fitting possible with the available data since measurement is by offering up a range of pins to gauge diameter. Therefore, it must be assumed that the holes remain circular and the characterization of the uncertainty is wholly a statistical review to determine the uncertain parameters and their applicable probability density functions, i.e. "Fit statistical models of parameters and correlations" in figure 3.8.

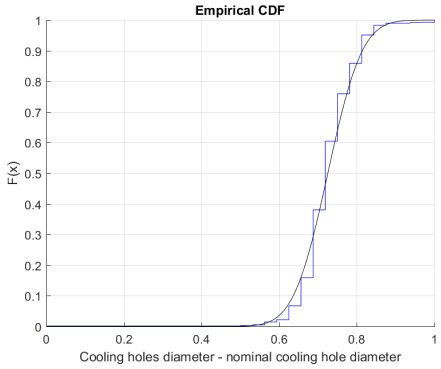
Cooling hole uncertainty characterization

A review has been made of the cooling hole measurements to ascertain whether the hole diameters fit any known statistical distributions and whether there are any correlations in hole diameters of the same row or tile, for example. Assessments of the data using the Jarque-Bera and Lilliefors tests conclude that the null hypothesis that the change in diameter (δd) for the cooling holes is normally distributed can be rejected at the 5% level. The sample data set is compared with a normal distribution in the plots shown (figure 5.7). Neither a discrete distribution, such as the Poisson, or an extreme value distribution were reasonable fits to the data either. Therefore, the diameter variation cannot be considered to be a single random parameter from which diameter changes are drawn for uncertainty assessment. There are more than 100 cooling holes in an analysis sector so it is essential that the holes be grouped in some way in order to facilitate uncertainty studies.

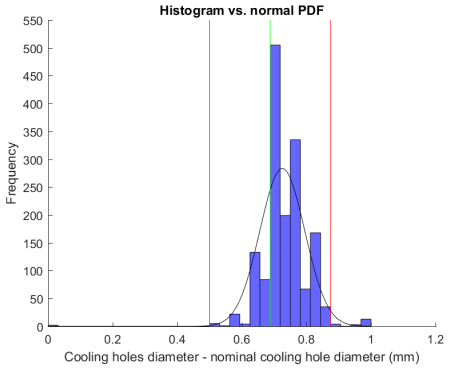
It is likely that any slippage in hole diameter cutting due to changes in the laser cutting system over time will affect holes of the same diameter in the same row if they are cut as



(a) Probability plot



(b) Cumulative distribution function



(c) Histogram, showing fitted normal distribution, nominal and tolerance bands

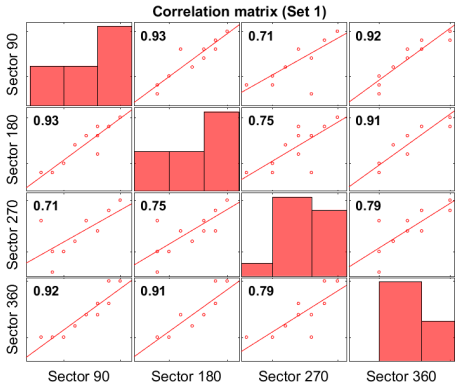
Figure 5.7: Cooling hole diameter data to normal distribution comparison (diameters normalized for confidentiality)

a batch. By plotting (for all 12 engines) the diameters of the holes in one sector against the diameters of nominally equal holes in the other sectors, the correlations between them can be assessed. There are 35 sets of holes which are nominally the same diameter and row. The plots are shown here for three example sets, and the corresponding correlation coefficients shown (figure 5.8). In general, the correlations are high; we see that, for example, if the diameter (d) is high in set 1 in the 90°sector, then it will also be high in all the other sectors of that engine.

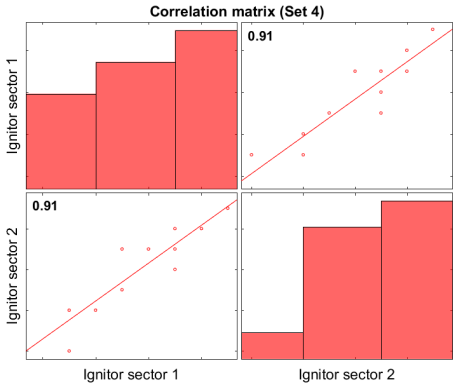
In fact, if we assess all of the holes in each of the rows against each other, then we can also conclude that, in general, the correlation is high, excepting a couple of outlying points. Correlations are shown here for row ACR (set 4) with two outlying points (found by observation) removed, figure 5.9. Each axis is a set of holes of the same diameter.

Other assessments were made to:

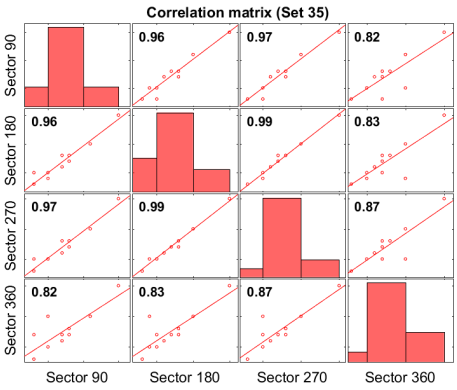
- check for patterns through the expected order of manufacture but, given only four measurements per row, only the most distinct of trends would be discernible: none were observed,
- look for distributions according to hole nominal diameter: none were observed, and



(a) Set 1 of 35



(b) Set 4 of 35



(c) Set 35 of 35

Figure 5.8: Correlation matrix plot for cooling holes, same nominal size and row (values on axes removed for confidentiality)

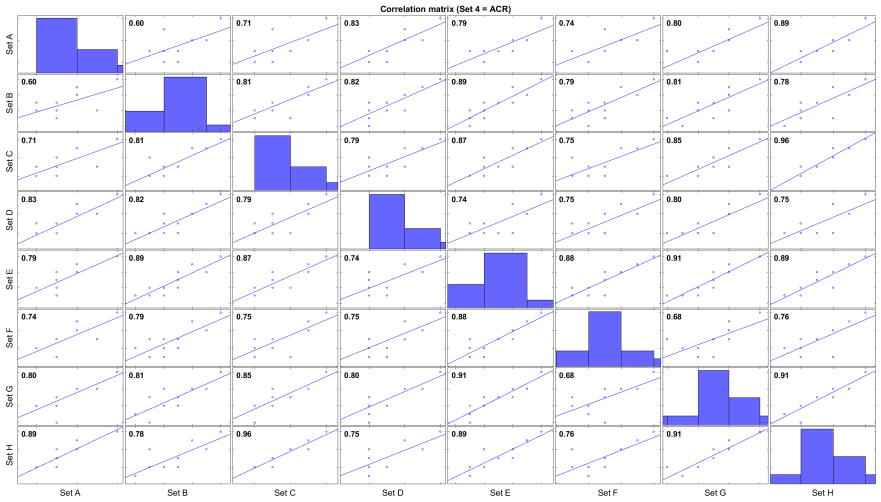


Figure 5.9: Correlation matrix plot for cooling holes in the same row (values on axes removed for confidentiality)

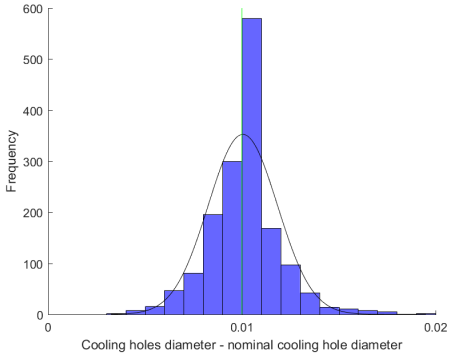


Figure 5.10: Throughout row cooling hole diameter variation histogram, normalized for confidentiality (μ and σ an order of magnitude lower than across hole variation)

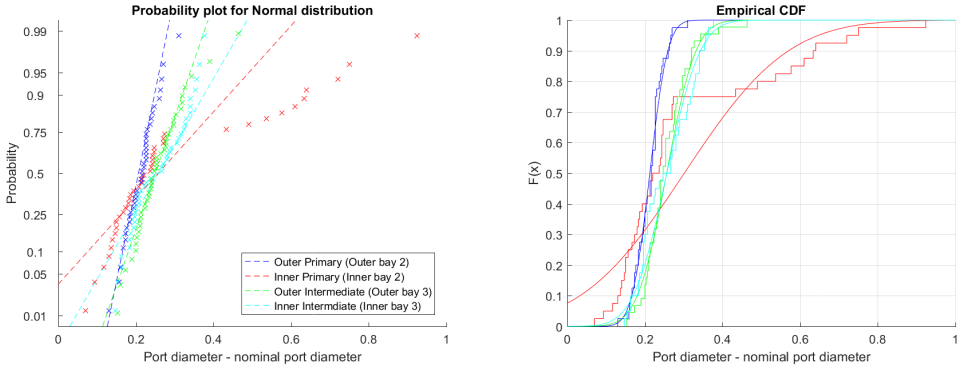
- search for correlations between holes on the same tile: none were observed.

Given the strong correlation between the hole diameter variability in individual rows, the uncertainty study requires that the sensitivity to variability in hole diameter throughout the rows (the distribution of the residuals, figure 5.10) be considered first, and separately to the across-row variation. To assess the effects of both uncertainties at the current nominal design, multiple runs of the CFD analysis are required where hole diameters are modified for each run.

Port hole uncertainty characterization

Considering each tile type individually, the distribution of their measured port diameters has been compared with a normal distribution (figure 5.11). OP, OI, and II ports are closest to a normal distribution and, depending on which statistical test is used it is possible to conclude that a hypothesis that the samples are drawn from a normal distribution cannot be rejected at the 5% level. It is not possible to reach the same conclusion, however, for the IP ports.

An assessment of correlations suggests that tiles on which the nominal diameters are largest (OI and II), there appears to be a significant correlation between the tiles measured, i.e. if a hole is large on one OI tile then the same hole is likely to be large on another OI tile. However, the quantity of data is very small. In this case, evaluation of output temperatures should be made using the actual measurement points. This will provide an initial assessment, and demonstrate the potential of the proposed methodology, but for statistically rigorous review additional measurement data would be required in order to include any correlations and to allow accurately fitted distributions.



(a) Normal probability plots (b) Cumulative distribution functions

Figure 5.11: Port diameter measurement data normal probability plots and cumulative distribution (cdf) plots for comparison with a normal distribution (data normalized for confidentiality)

5.3 Implementation of generic hole diameter change in a fluid volume mesh

In order to implement changes in cooling hole and port diameters, whilst maintaining the integrity and topology of the CFD mesh on a local scale, a mesh morphing based approach is proposed:

1. determine a grid (lattice) of control points within which mesh vertices (nodes) can move and outside of which they remain fixed, then
2. move the grid points within the lattice boundary, using a free-form deformation (FFD) algorithm to position the remaining nodes within the lattice.

The most commonly documented and readily available FFD algorithm is that of Sederberg [250, 188], which uses Bernstein polynomials to define the shape of the deformation within the bounds of a cuboid lattice with lattice points that are equally spaced along the edges, section 2.11.

Hole through a flat plate

First, a generic hole through a flat plate, within a cuboid lattice is considered. The mesh volume for such a problem can be visualized as shown in figure 5.12. The diameter changes to be made require that:

- the positions of the points on the extremities of the cuboid containing this mesh region remain fixed,
- the hole remains cylindrical,

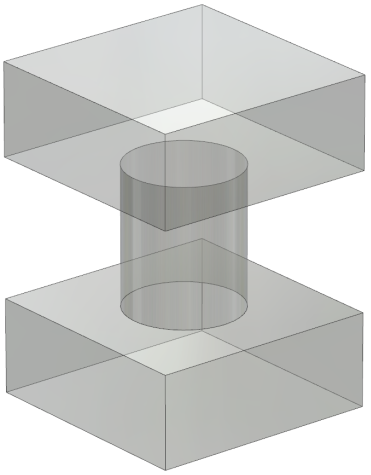


Figure 5.12: Illustration of region of mesh extents requiring morphing through a hole (square; cuboid control lattice)

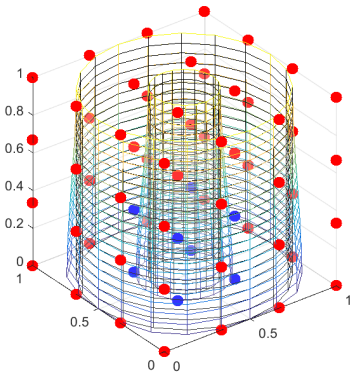
- elements maintain their quality (do not become too skewed for example), and
- there is a smooth transition in element size throughout the region and with the fixed elements surrounding it.

With the Sederberg algorithm all of the control points influence all points in the mesh so that it is not possible to meet all these constraints. Continuity and mesh size smoothness can be maintained by ensuring there are enough carefully positioned control points but the corner control points exhibit a 'pull' (influence) on the mesh that causes circularity of the hole to be lost. This is not a problem outside of the hole, but could result in sliver elements at the lattice boundary. The difficulty is illustrated by using the Sederberg algorithm to deform a cylinder, figure 5.13.

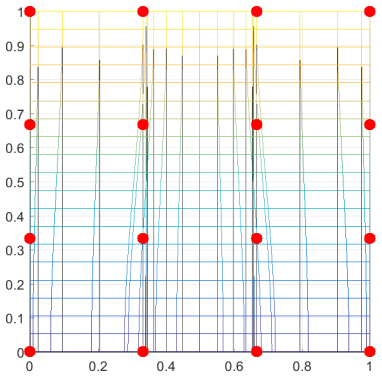
Given the above limitations of the FFD technique for maintaining shape integrity, a modification of the readily available FFD code is required. It might be possible to modify the FFD to utilize alternative curve formulations such as those described by Sóbester and Forrester [264]. However a generic circular hole, where circularity is maintained (or even where not), requires only modifications to nodal positions radially. As such, the problem could be greatly simplified by converting the vertex coordinates to a polar coordinate system. This then allows the deformation to be conducted in (r, z)-space, maintaining circularity and fixing θ.

The proposed approach for editing the diameter of a generalized circular hole through a flat surface is described in the flowchart of figure 5.14. In this polar case, the extents of the lattice form a dumbbell shape (figure 5.15).

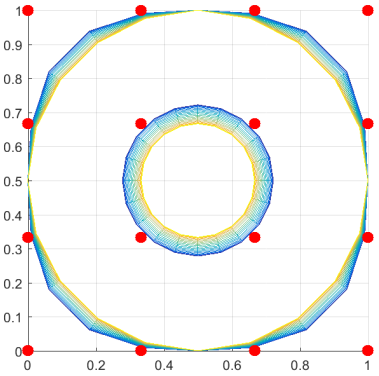
By following the workflow described in figure 5.14 through, the FFD process can be illustrated clearly in the two polar dimensions. Consider region 1 of figure 5.15b: if the points are transformed to polar coordinates (r, θ, z), θ is ignored as it will be kept



(a) Isometric view



(b) Side view



(c) Top view

Figure 5.13: Free-form deformation (FFD) of a cylinder using 16 control points

constant, and r and z are normalized such that their range is from 0 to 1, then the nodes will lie on a [0,1] grid of points. For this central region 1 (the hole), a 2×2 lattice is constructed around the points and the grid is morphed by moving the control points at $r = 1$ according to the proposed change in radius. This is shown using a regular grid of points in figure 5.16. The increase in cell size in the r-direction is spread evenly throughout r. The remaining regions are also normalized such that the nodal points are distributed over a 1×1 (r, z) square. Only the inner-most lattice points are moved, ensuring that nodal positions at the extremities are maintained (C^0 continuity), and there are four control points in the z-direction to ensure tangency at the vertical boundaries (C^1 continuity) when using the (figure 5.17).

In the r-direction there is a step change in element size at the region boundaries. To maintain continuity in this respect (C^2 continuity), additional grid points are required. In order to determine the positions to which any additional control points need be moved, we turn to the Bézier spline, section A.7.3. For a smooth connection between two successive Bézier curves defined by $n+1$ control points $\mathbf{p}$ and $m+1$ control points $\mathbf{q}$, the position

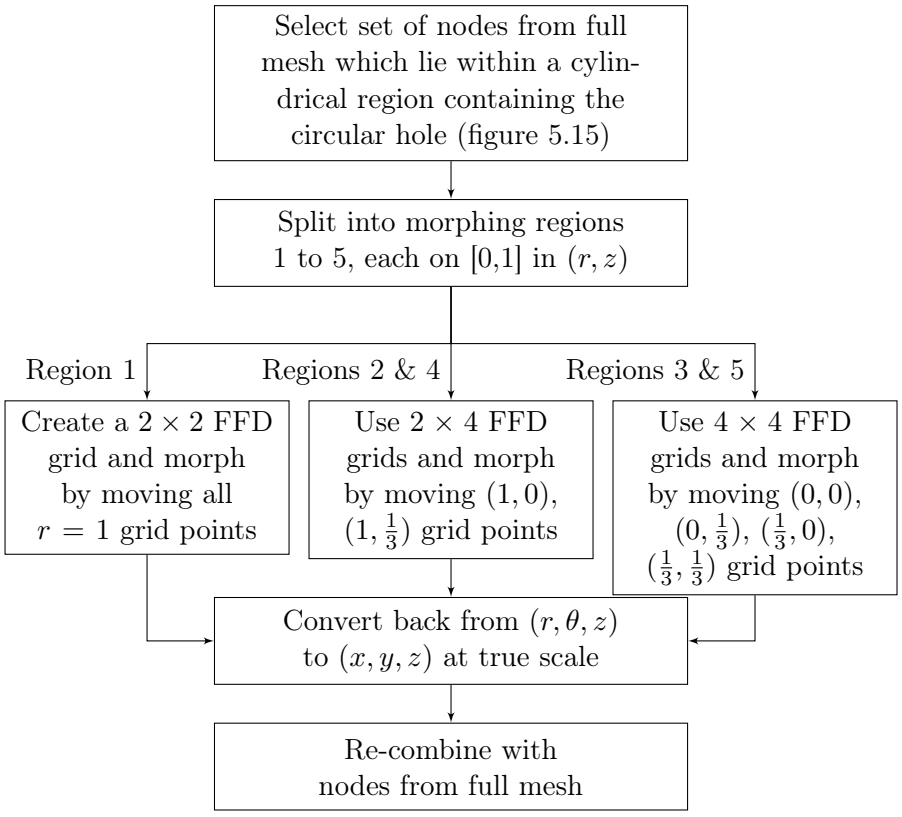
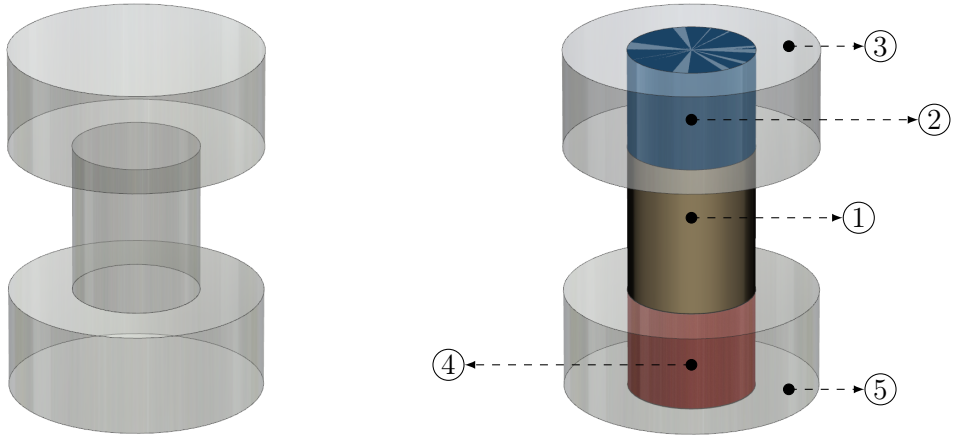


Figure 5.14: Flowchart showing the use of free-form deformation (FFD) to alter hole diameter in a computational fluid dynamics (CFD) mesh of a hole through a flat plate by adjusting nodal positions



(a) Full polar lattice mesh extents

(b) Polar lattice mesh regions

Figure 5.15: Illustrations of region of mesh extents requiring morphing through a hole (polar system; cylindrical control lattice)

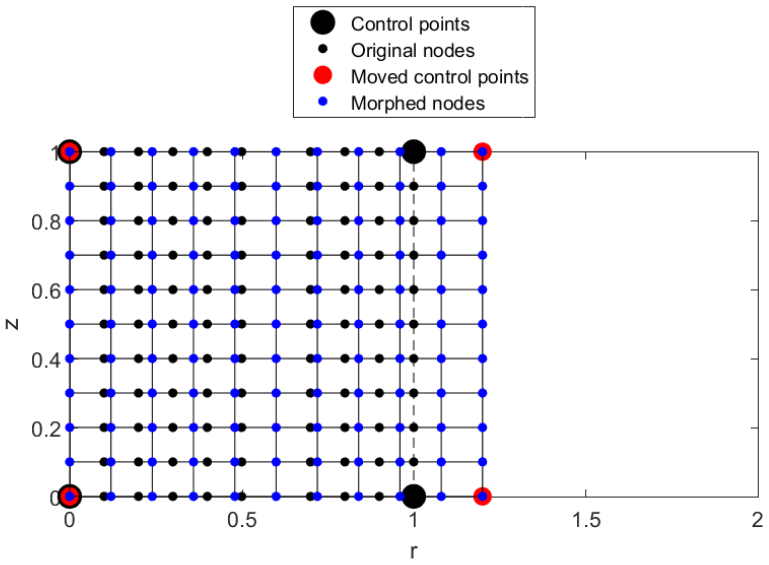


Figure 5.16: The free-form deformation of region 1 (the hole) using transformation to polar coordinates

of the point of intersection can be calculated [264] as:

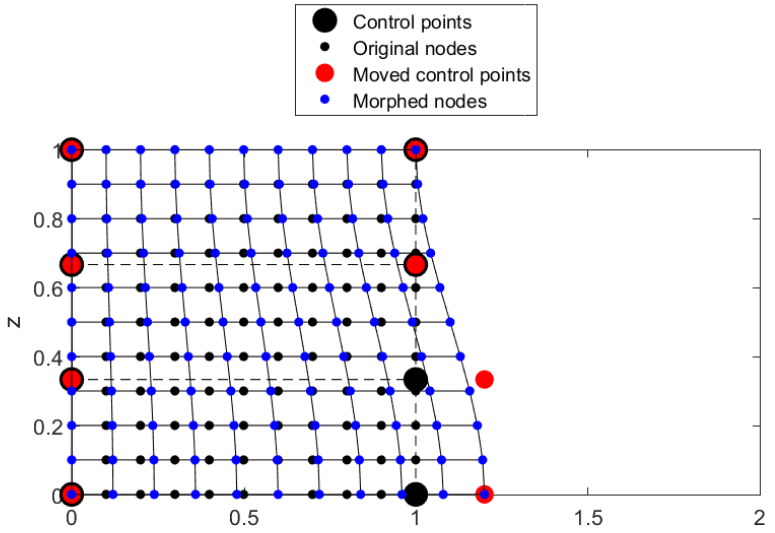
$$\mathbf{p}^{(n)} = \mathbf{q}^{(0)} = \frac{m}{m+n}\mathbf{q}^{(1)} + \frac{n}{m+n}\mathbf{p}^{(n-1)}. \tag{5.3}$$

In this case, the intersect is pre-defined as the modified radius of the cooling hole. Rearranging the above equation gives an equation for the original $\mathbf{q}^{(1)}$ r-coordinate:

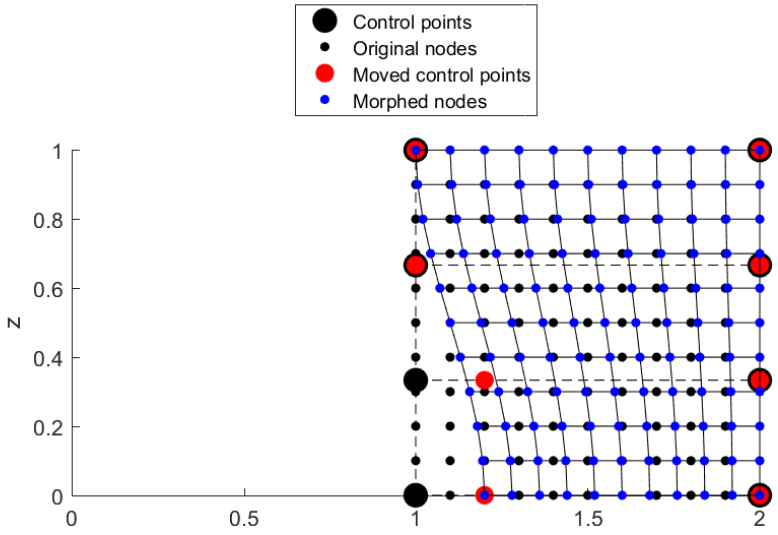
$$\begin{aligned} \mathbf{q}^{(1)} &= \frac{m+n}{m}\left(\mathbf{p}^{(n)} - \frac{n}{m+n}\mathbf{p}^{(n-1)}\right) \\ &= \frac{3}{2}. \end{aligned} \tag{5.4}$$

If, for example, using the normalized square regions, an increase in radius of 0.2 normalized units is applied then this results in $\mathbf{q}^{(1)} = 1.6$, where $n = 1$, $m = 3$, and $\mathbf{p} = \begin{bmatrix} 0 \\ 1.2 \end{bmatrix}$. Considering the external region to be a unit square, $\mathbf{q}^{(1)}$ can be calculated to be at $r = 1\frac{1}{3}$. This polar deformation is shown in figure 5.18.

Once the deformation has been made in the normalized polar coordinate system, the new point positions can be re-scaled and combined with the original θ values to convert back to Cartesian coordinates and then used to replace the nodal positions in the original mesh, figure 5.19.



(a) Regions 2 & 4



(b) Regions 3 & 5

Figure 5.17: The free-form deformation of regions 2, 3, 4, and 5 (outside the hole) using transformation to polar coordinates

Hole through a curved component (bottom part of a combustor port)

In reality, the surfaces through which holes are meshed are not often flat. The combustor casing cooling holes and ports are cut through curved surfaces (where z changes with r for the surface nodes). If the curvature is known, then a relatively simple transformation of the nodal points in the surrounding region can be applied prior to deformation, and in reverse to re-locate nodes within the original mesh. If the surface is complex, or curvature unknown, a generic approach is proposed, figure 5.20. The scheme removes the curvature in way of the holes by first modelling the generic surface using the existing surface nodes as the sample points for a Kriging [78] fit, although other surface fitting techniques could be used. The resulting *predictor* (Z) is subtracted from all of the nodal z-coordinates to

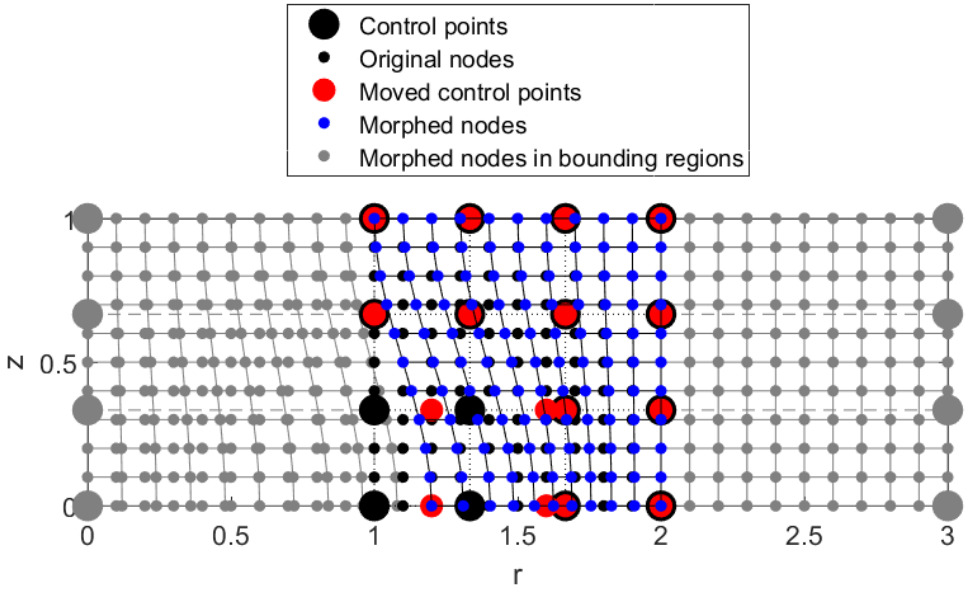


Figure 5.18: The free-form deformation of region 3 using transformation to polar coordinates, showing smoothing with an additional two control points

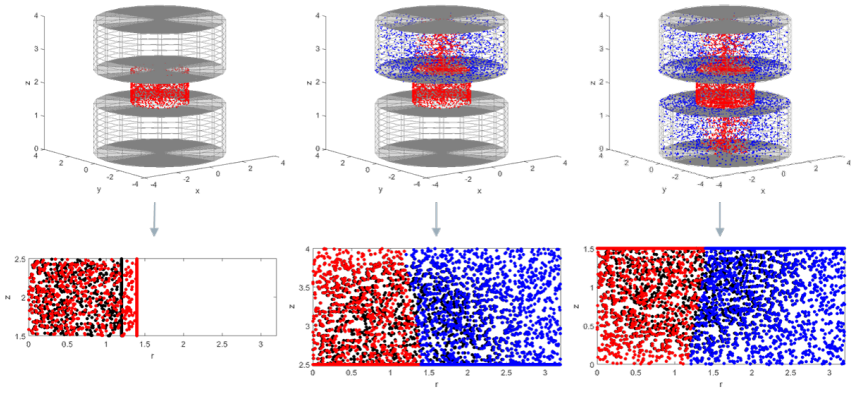


Figure 5.19: Illustration of the FFD of a mesh through a hole in a flat plate using mocked-up example node set (original nodes = black, morphed nodes = red and blue)

remove, and then later re-instate, the curvature. To perform this modification for top and bottom surfaces that are not matching, the morphed region must also be split at a hole mid-plane.

For example, consider the lower section of the port (where the curvature is not extreme) (figure 5.21a)[6] and, using the Kriging predictor fitted to the nodes on the lower surface (figure 5.22)[6], first perform a coordinate transformation on all the points (figure 5.21b)[6]:

$$z_{t_i} = z_i - Z(x_i, y_i). \tag{5.5}$$

The central hole section (1 lower) can be isolated from the rest of the mesh (the points, that when transformed, lie above the $z = 0$ plane). If the original positions of these points are taken (figure 5.23a)[6], and a transformation is performed to 'flatten' the region using the same predictor (figure 5.23b)[6]:

$$z_{f_i} = \frac{z_i}{Z_i}, \tag{5.6}$$

then FFD can be performed in the same way as previously demonstrated (figure 5.23c)[6], before 'un-flattening' and converting back to true scale. Regions 4 and 5 can be determined and morphed from their transformed positions (they do not need 'flattening'), before transforming and converting back to their original relative surface positions and scale (figure 5.24)[6].

When the morphed nodes of the lower part of an example port hole with artificially large increase in diameter are combined, the resulting morphed nodes in figure 5.25[6] are produced.

This methodology is 'neat' and generic but limited to solids with surfaces where a mid-plane can be defined through the entire hole, and where there is no extreme curvature (no large changes in z; also a limitation of standard Kriging).

[5] A limitation of the method is that this must be possible, i.e. a mid-plane must create a full circle with no gaps.

[6] Scales on all port geometry removed for confidentiality.

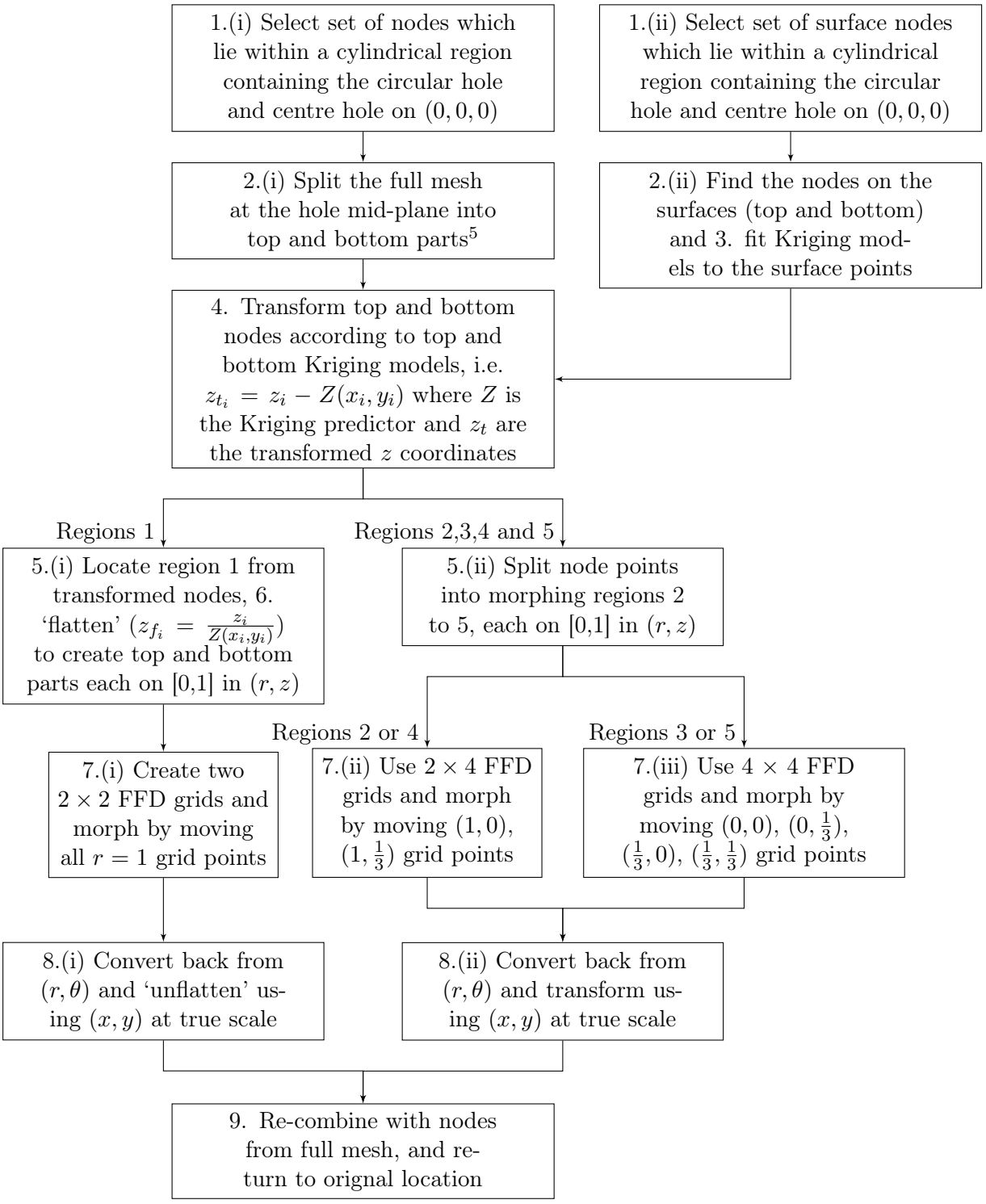
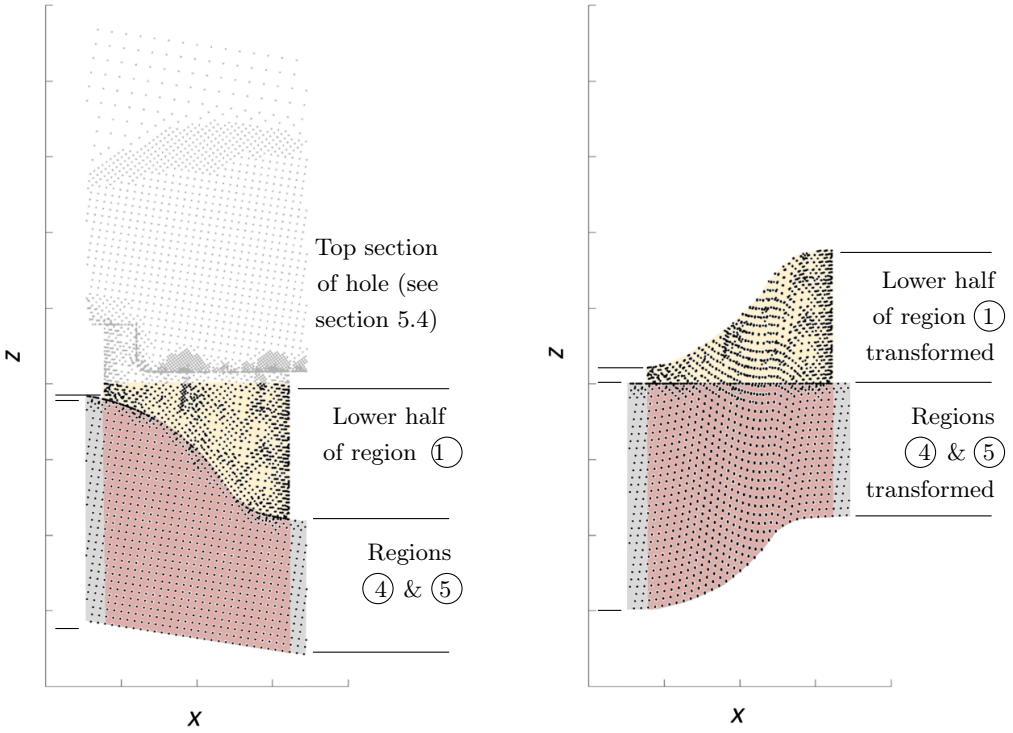


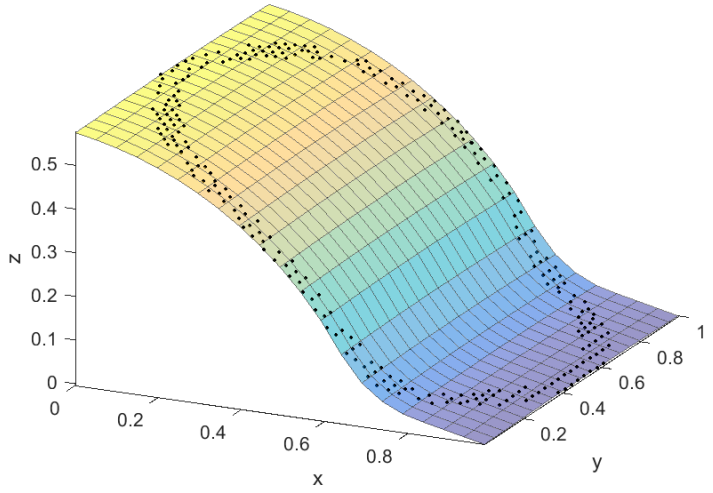
Figure 5.20: Flowchart showing the use of free-form deformation (FFD) to alter hole diameter in a computational fluid dynamics (CFD) mesh of a hole through a solid with non-specified curvature on it's surface by adjusting nodal positions



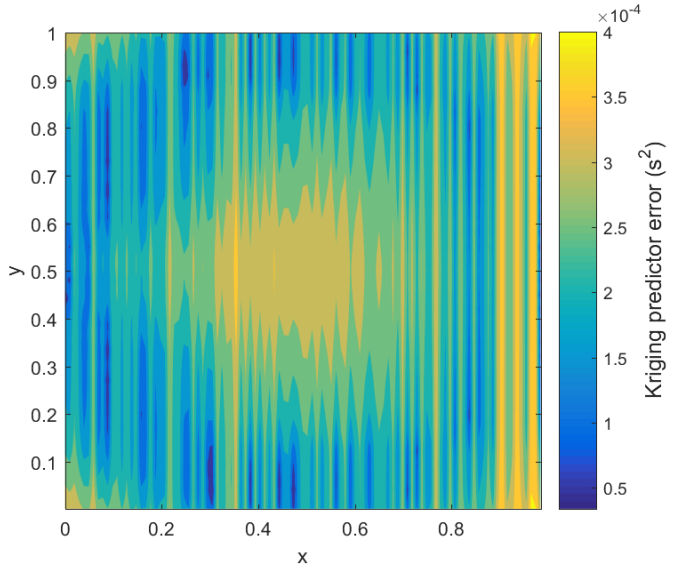
(a) Original example port mesh points (bottom section isolated)

(b) Example port bottom mesh points transformed using Kriging predictor)

Figure 5.21: Example port bottom section Kriging-based transformation (step 4 in figure 5.20, regions as figure 5.15b)

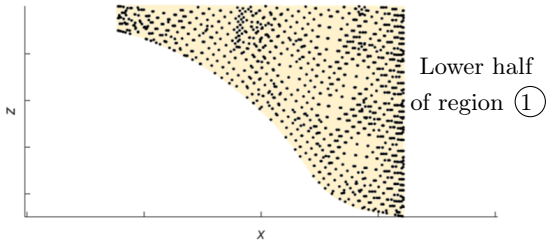


(a) Fitted surface (normalized for confidentiality)

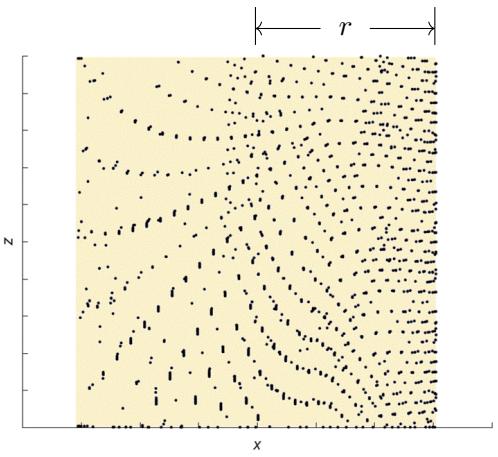


(b) Kriging predictor error (normalized for confidentiality)

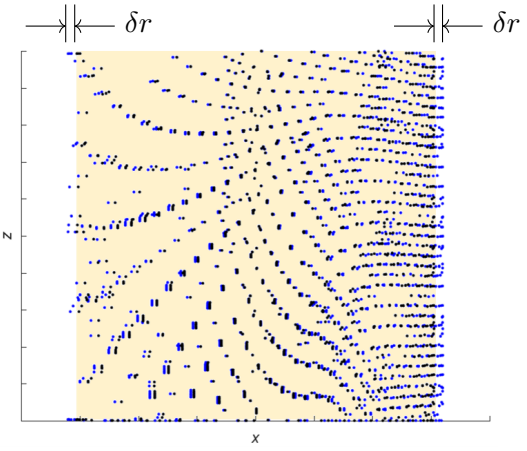
Figure 5.22: Kriging fit (for step 4 in figure 5.20) to the bottom solid surface for port deformation, showing Kriging predictor error (section 2.8)



(a) Original points

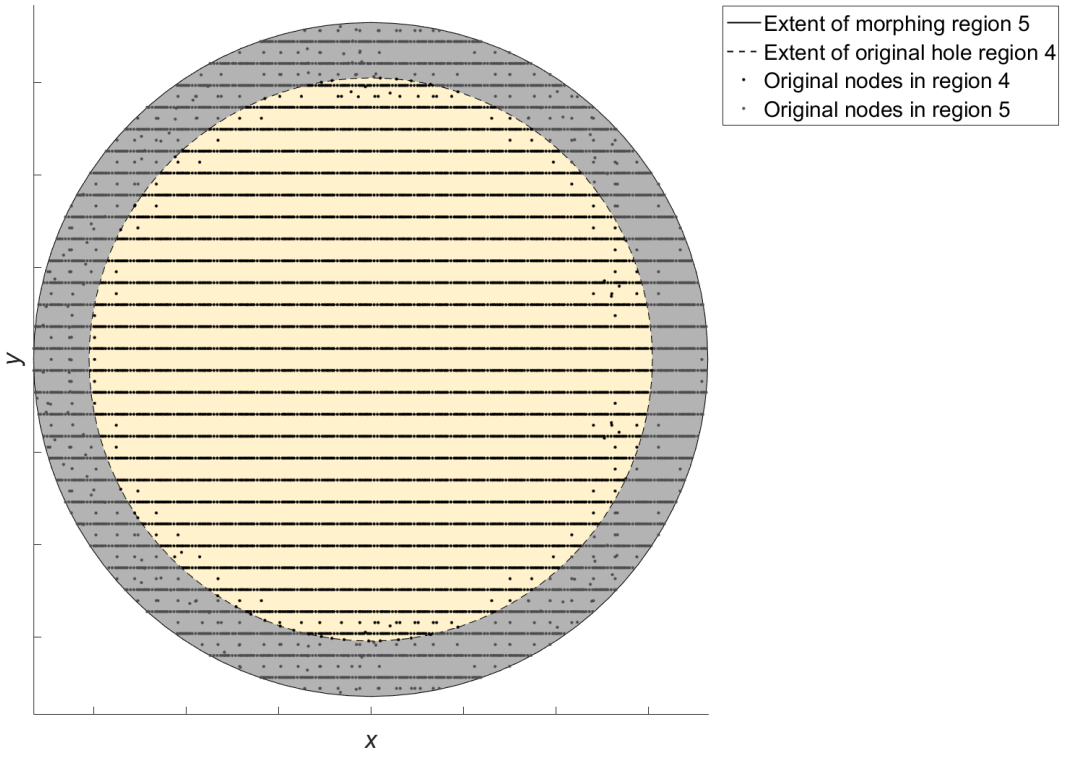


(b) Lower half of region 1, points 'flattened'

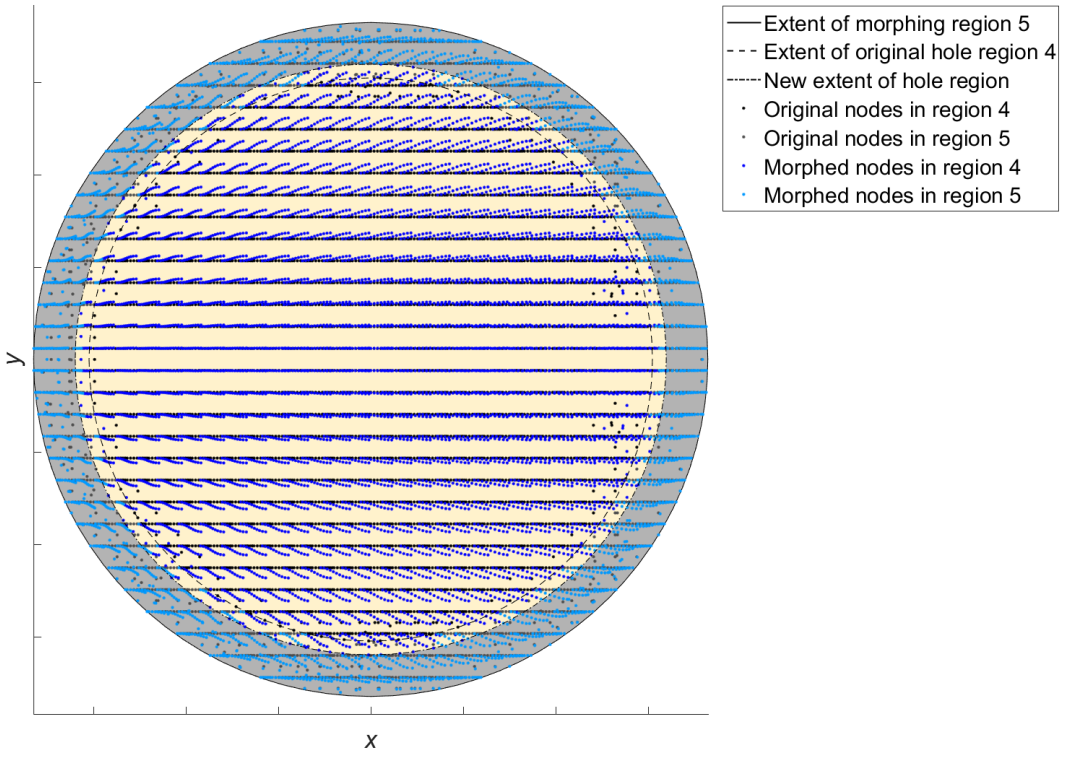


(c) Lower half of region 1, blue = deformed (morphed) points after 'flattening'

Figure 5.23: Example port region 1 (lower) morphed using free-form deformation in polar coordinates



(a) Original points (top view)



(b) Deformed points (top view)

Figure 5.24: Example port regions 4 and 5 (lower) free-form deformation (vertical (z) coordinates remain unchanged)

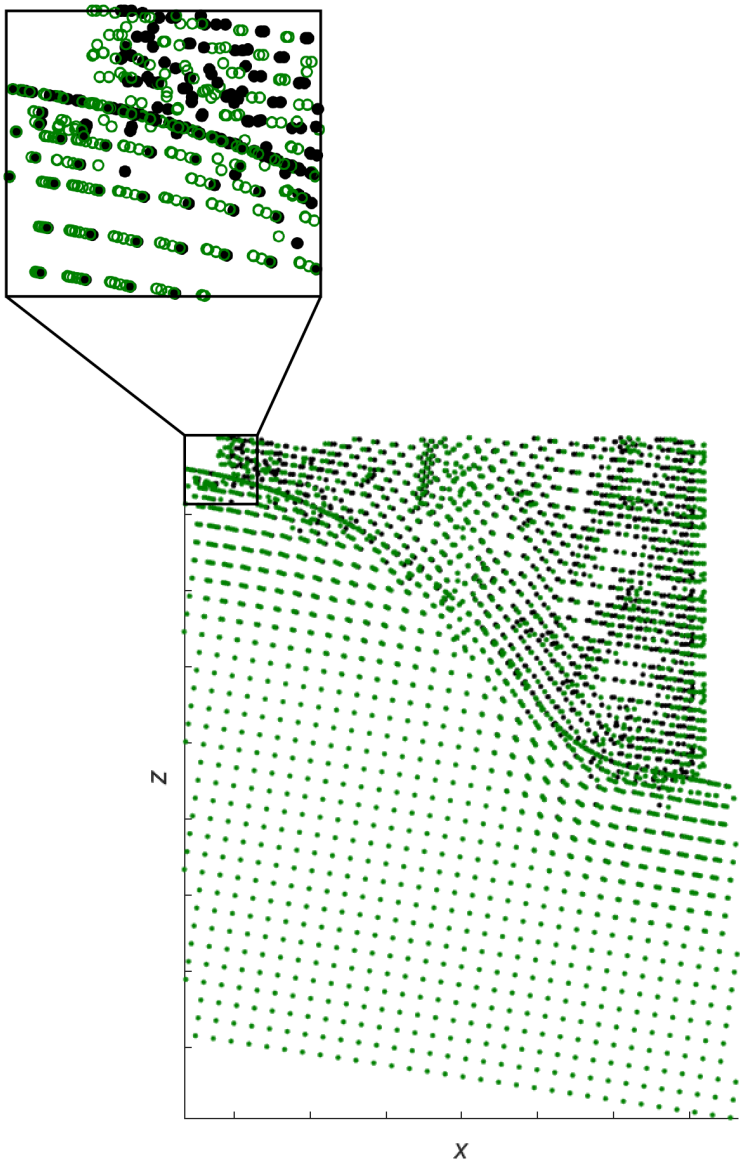


Figure 5.25: Example morphed port lower section (original points = black, new points = green)

5.4 Refinement of the generic hole diameter change implementation for application to complex port surface geometry

Unfortunately the port geometry for the case study includes an additional feature that renders the generic solution developed in section 5.3 inapplicable: there is a step in the complex surface. The step makes the Kriging surface impossible to construct because, on the vertical surfaces of the component, there can be multiple z values for each single (r, θ) or (x, y) position. However, because we wish to:

- remove and then re-instate z, and
- maintain a surface in the vertical region,

it is proposed that, for this known individual port geometry, the nodes in the region of the step can be treated in a geometry-specific way. Ports in a number of combustors exhibit this feature such that development of a problem-specific solution is of interest despite its limited range of application.

The workflow used for the stepped geometry is given in figure 5.26. The fundamental difference between the methodology for the non-stepped and stepped surfaces is that the non-stepped bottom section is morphed in its transformed state, whereas the stepped surface is morphed prior to applying corrective nodal displacements to the surfaces. One can imagine that further steps could be made to enhance this approach for components with steps with non-planar tops where both a transformation and a correction would be required.

The port-specific morph requires the user to input a number of fixed nominal values for each port, in addition to the radius change, in order to split the mesh into a slightly different set of regions, figure 5.27. These parameters are shown on a cross-section of a port in figure 5.28.

The first correction is applied to the morphed nodes that exist within the extents of the vertical surfaces in x, and above a pre-determined mid-point on the fillets. These are the morphed nodes shown in figure 5.30[6]. This example uses an artificially large increase in hole radius for demonstration purposes. The morphed nodes originally on the vertical surface, must be corrected by placing them back onto the surface. This correct x location (shown in red) is found by using a vertical Kriging predictor fitted to the original vertical surface nodes. The x-location of the nodes aligned with the step must also be adjusted. This correction is prescribed by calculating the distance to the vertical surface and reducing the displacement linearly to zero at $x = 0$. The correction is shown on one side of the port in figure 5.30[6].

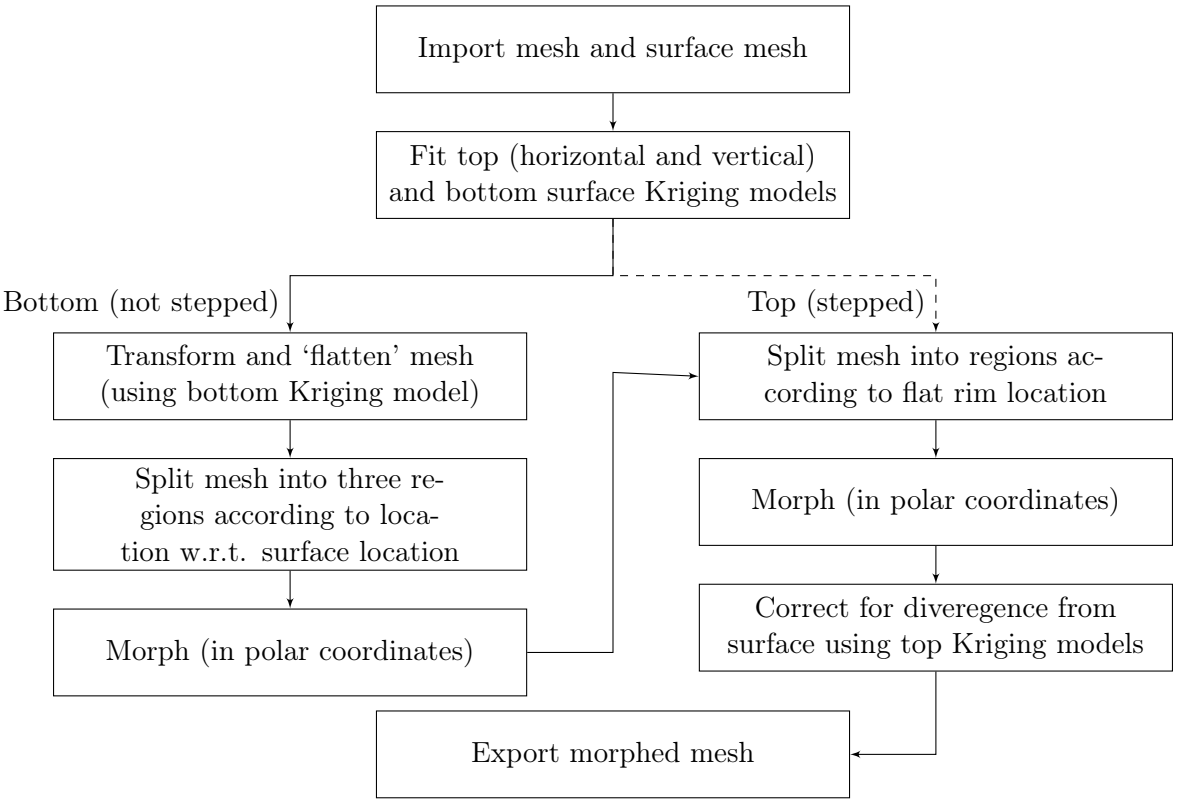


Figure 5.26: Flowchart showing the modification of port diameters in a CFD mesh using free-form deformation

Below the same pre-determined mid-point on the fillets, morphed points must also be moved back towards the surface, but moving them in x would create potentially large skew in elements in the region. Therefore, these points are corrected in x only by the value of δx at the top of the fillet, and then by δy at this new position. The resulting nodal positions are shown in figure 5.31[6] and 5.32[6]. All of the corrected nodes are shown in figure 5.33[6].

The final nodal positions for the morphed nodes in the top stepped section for the negative radius change example are shown in figure 5.34[6].

This algorithm has been applied to the CFD mesh for analysis runs where an individual port is morphed to a measured diameter value, and further runs where all measured ports are morphed. Each of the resulting meshes has been checked for quality using the ICEM mesh quality checker. For the magnitudes of changes needed, the new meshes meet the same mesh quality requirements as the original mesh.

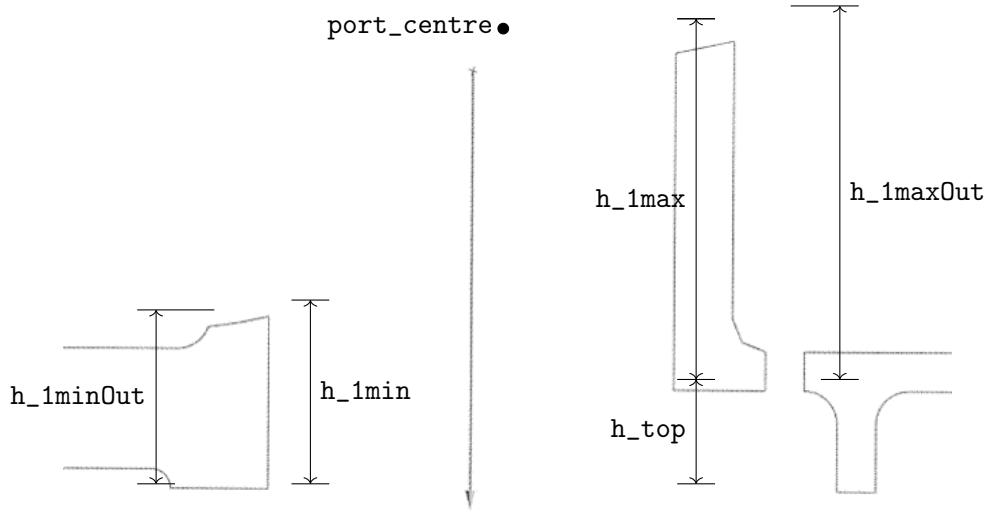


Figure 5.27: Cross-section through inner primary (IP) port showing geometrical extent parameters needed as inputs to morphing process

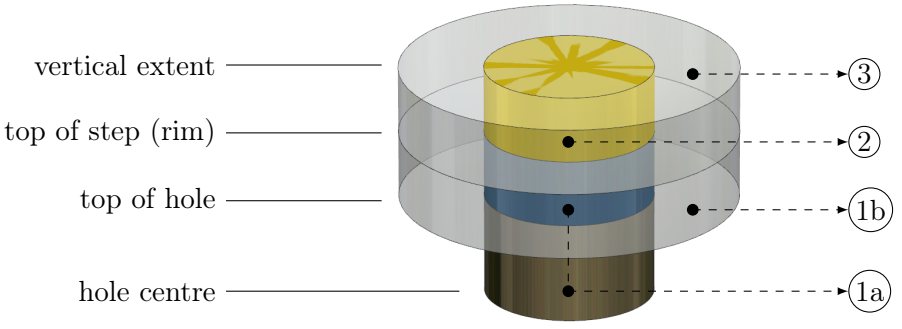


Figure 5.28: Illustration of regions of mesh extents at the top of a port hole (section with a step) requiring morphing (polar system; cylindrical control lattice)

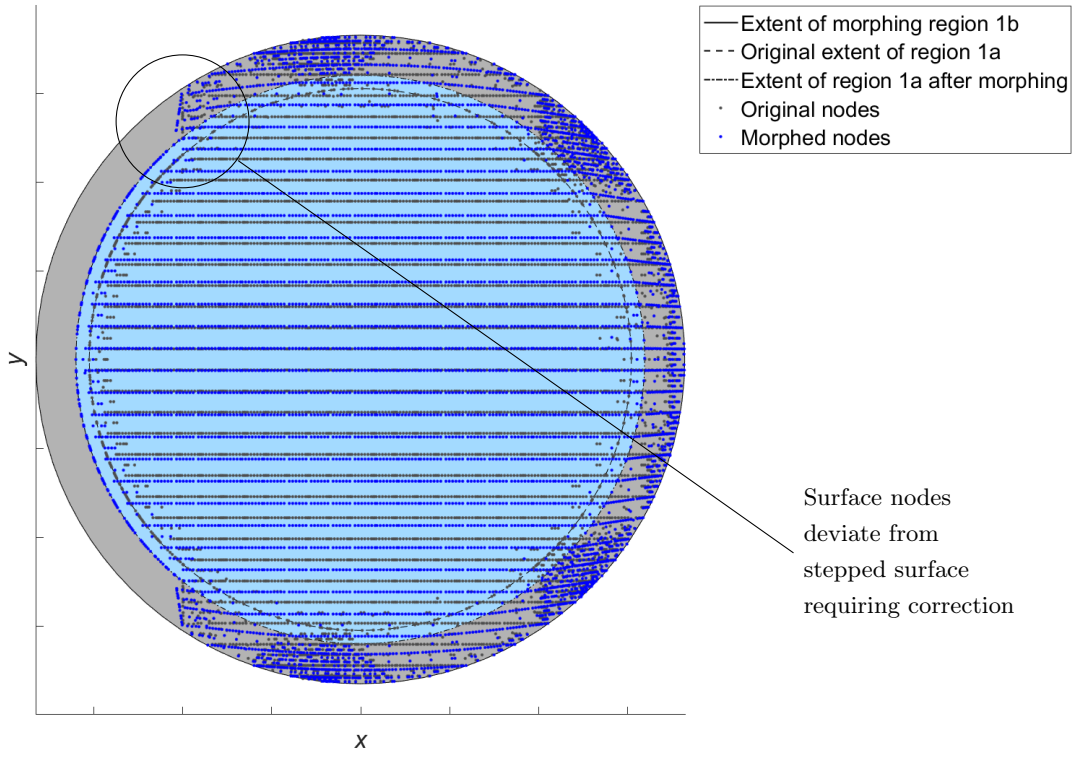


Figure 5.29: Regions 1a and 1b prior to application of corrections for step and fillet

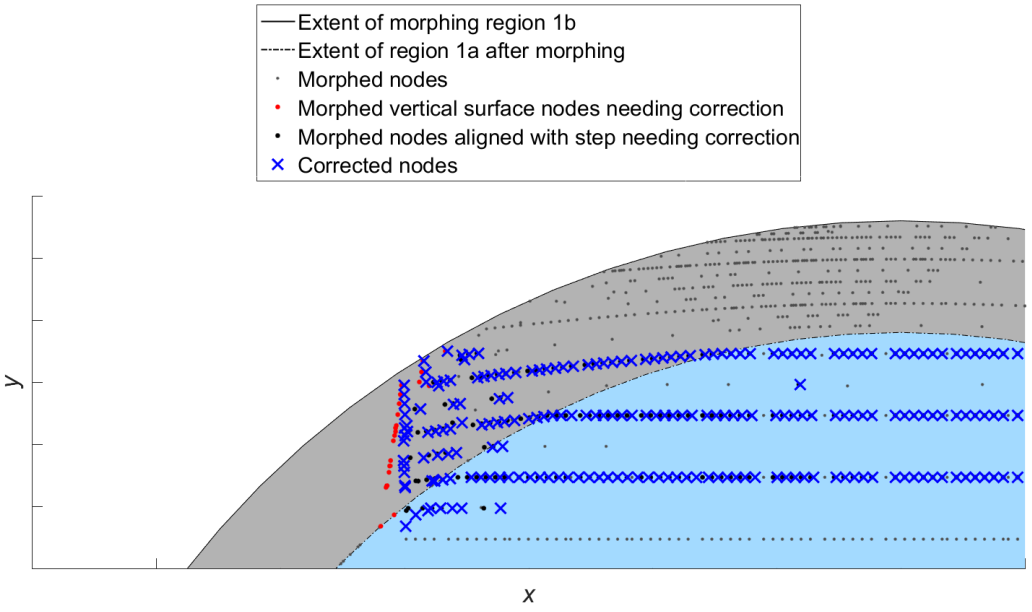


Figure 5.30: First correction made to morphed nodes to ensure port surface shape is maintained (adjustment by δx for points between the fillet mid-point and the rim)

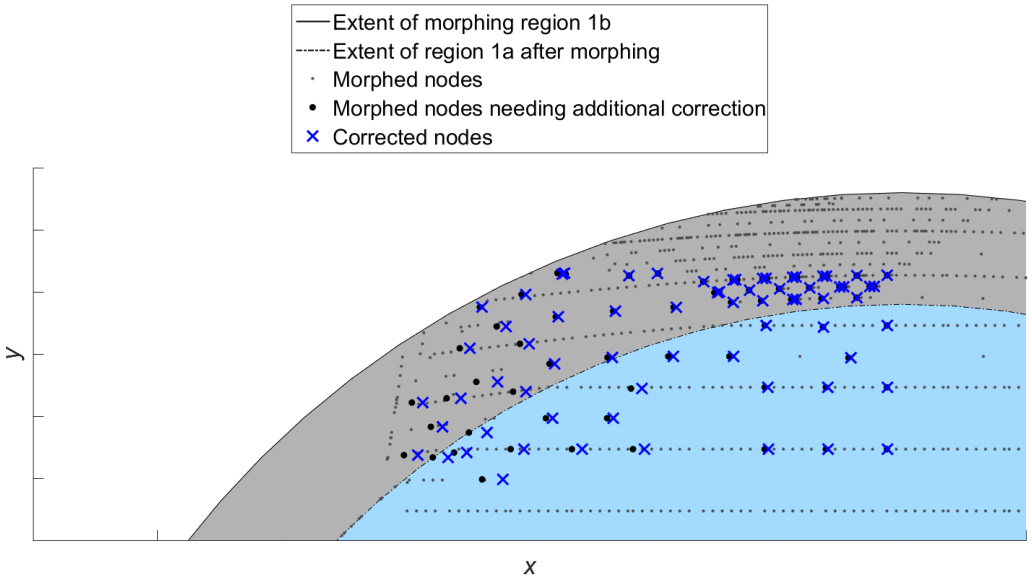


Figure 5.31: Second correction made to morphed nodes to ensure port surface shape is maintained (adjustment by δx and δy for points between the fillet midpoint and the top of the hole), $x - y$ view

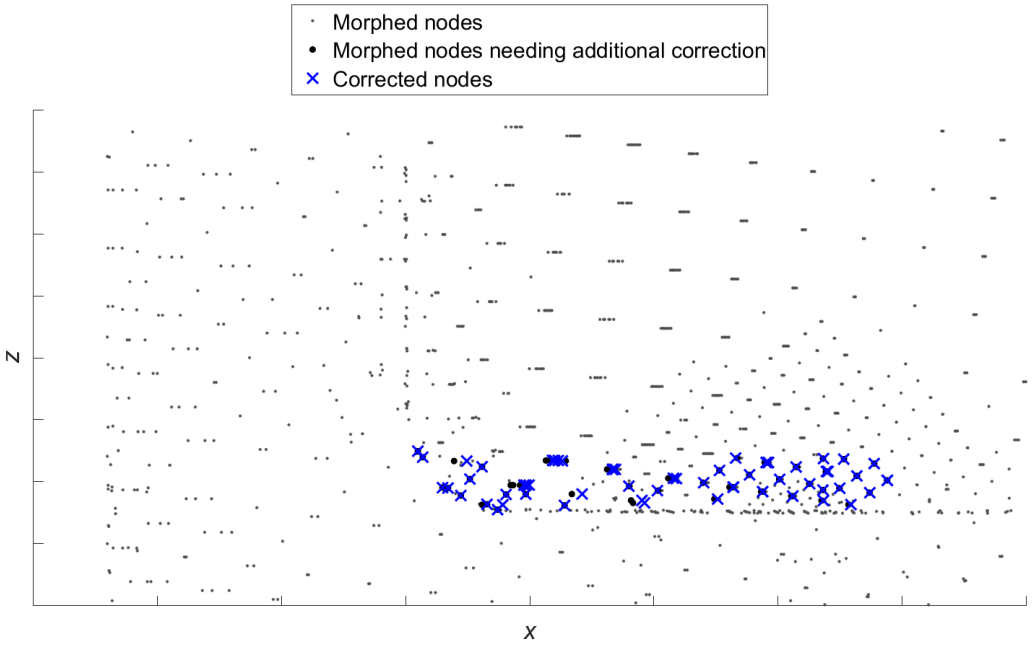


Figure 5.32: Second correction made to morphed nodes to ensure port surface shape is maintained (adjustment by δx and δy for points between the fillet midpoint and the top of the hole), $x - z$ view

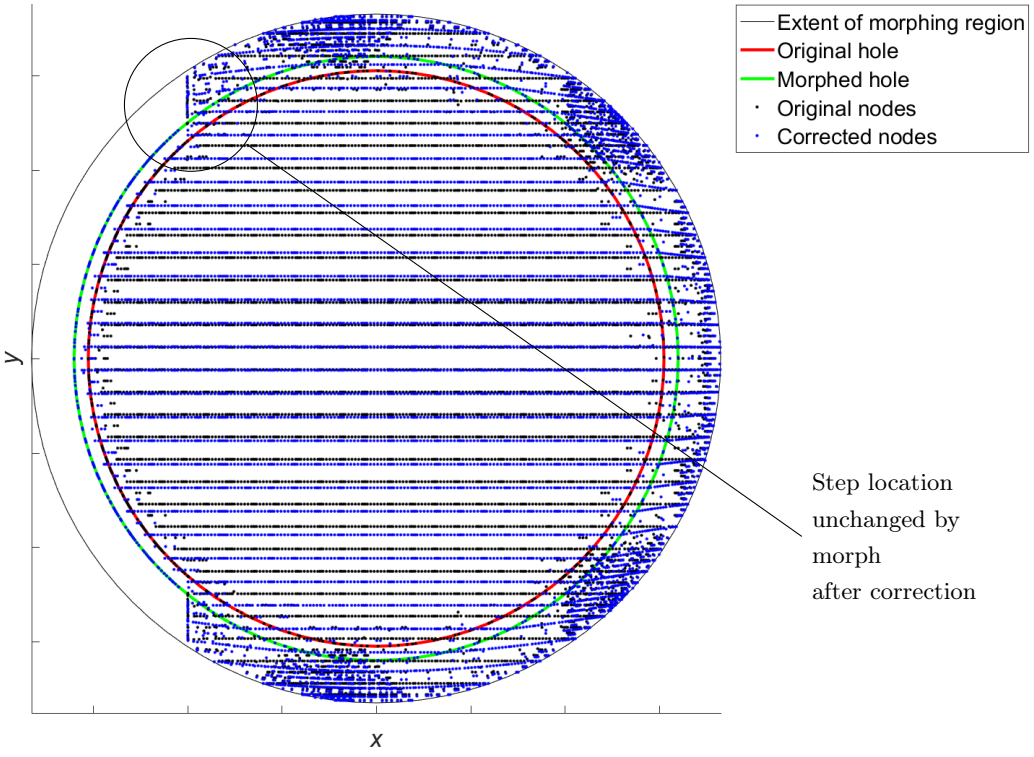


Figure 5.33: Corrected nodes in region 1, $x - y$ view

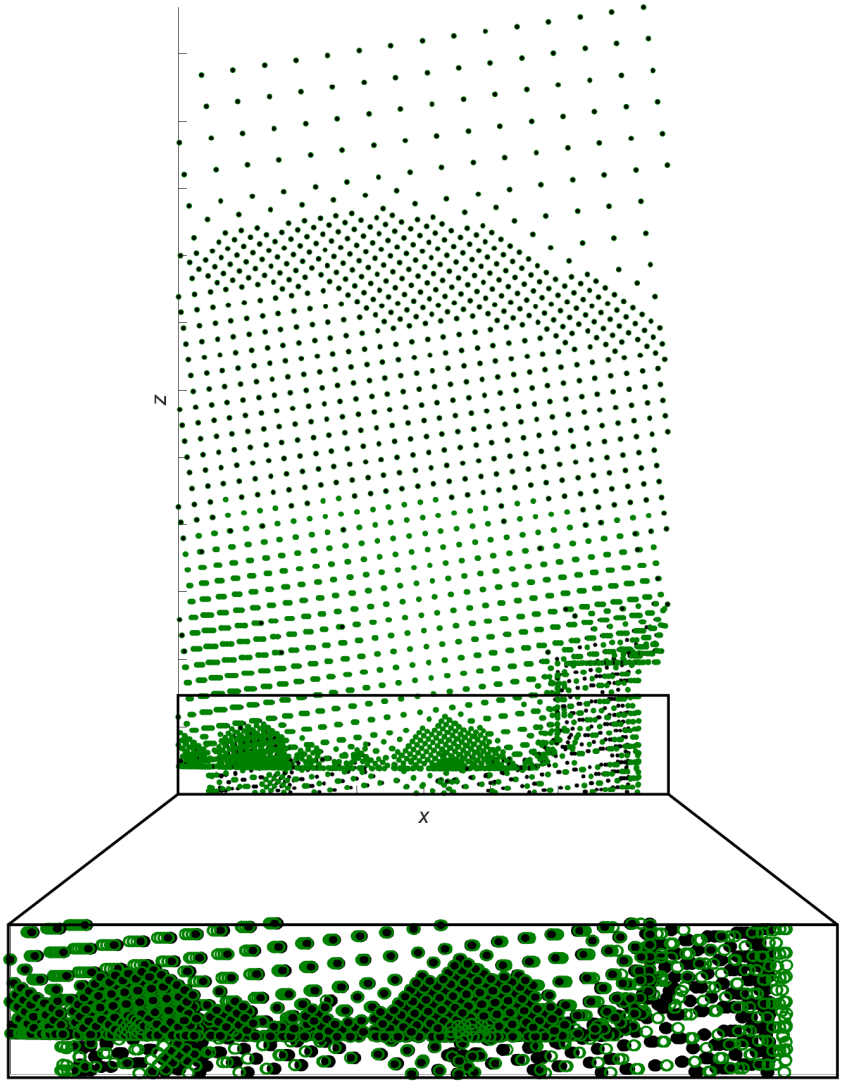


Figure 5.34: Morphed nodes in top section (large positive displacement example), $x - z$ view (original points = black, new points = green)

5.5 Contributions

A novel polar-coordinate based free-form deformation scheme for the application of measurement uncertainty to the diameters of meshed holes, and an associated Kriging-based transformation for holes in generic complex surfaces, has been presented. This approach allows for small magnitude geometric changes to be applied to large and complex pre-existing meshes without the requirement for significant manual intervention. The process can be applied to any mesh through curved surfaces. The top and bottom surfaces are not required to take the same form. The only limitation to the approach is that, in the region close to each hole, where the mesh is to be manipulated, the surfaces must be smooth and continuous. This requirement is a stipulation of the surface fitting techniques that are applied.

In the case of a typical combustor port, as provided by Rolls-Royce plc., the outer surface exhibits a step in the surface. In this case a problem-specific approach has been taken to transform the data points prior to deformation in order to maintain mesh quality and ensure that surface nodes remain attached to the surface.

It is shown that the polar-coordinate based morph is efficient for implementing changes within generic smooth surfaces. Given the array of components in existing products that contain circular holes, this approach could be widely applicable in engineering design. The RBF morph that has been discussed in section 2.11 and applied in chapter 6 could also have application for these kinds of problems. However, the difficulties associated with surface nodes being constrained on to existing geometric surfaces would still be encountered. As such, the RBF morph would most likely provide an alternative means of ensuring smooth continuous meshes, but not an additional way to define the 'along-surface' movements of surface nodes. A combined RBF morph and Kriging or RBF-based surface transformation based approach might warrant further investigation.

Chapter 6

Uncertainty characterization and implementation with re-fitted parameters

Having characterized existing parameters using point-cloud data from inspection measurements in chapter 4, and implemented changes to an existing analysis mesh based on measurements of existing parameters in chapter 5, we now look into a more complex and ubiquitous problem, where measurement data shows geometric changes that cannot be represented using existing parametrizations. In this chapter, we return to the firtree to investigate both the most appropriate means of accurately **characterizing** uncertainty, and the means by which new parameters can be sampled, and **implemented** through modifications to pre-existing FEA meshes, i.e. the application of both steps presented in figure 1.4.

A Kriging-based fitting routine, developed for fitting useful curves to CMM point data, is presented in this chapter. An approach for extracting a suitable set of uncertain parameters and their distributions from these curves; a 'pseudo-curve' sample for UP and RDO studies, is proposed and demonstrated. It is then shown that, through the use of the RBF morphing technique, meshed geometry can be manipulated using samples of these fitted parameters in order to implement changes in existing analysis workflows.

6.1 Problem description: firtree flank contact surface uncertainty

A second set of data pertaining to 'firtree' joints has been provided by Rolls-Royce plc. This set is generated using the same CMM and probing system described in chapter 1 but, unlike the data used in chapter 4, provides points on the blade root (example blade

Figure 6.1: Photograph of an example turbine blade with 'firtree'-type attachment to disc, courtesy of Rolls-Royce plc.

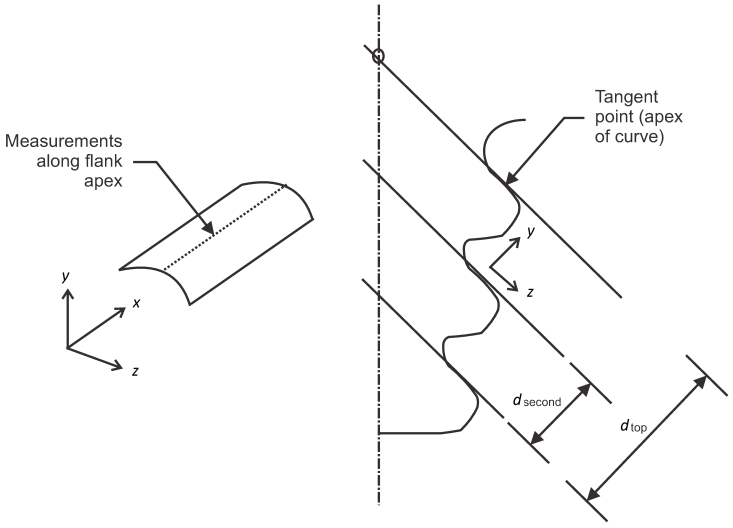


Figure 6.2: Schematic showing location of turbine flank measurements on turbine blade 'firtree' joints

shown in figure 6.1), rather than the disc slot, and measurements are made along the apex of 'barrelled' flank surfaces, figure 6.2.

The data points (27 in all) are given in two dimensions: fixed x, and y (x along the nominally straight-line blade flank and y perpendicular to the nominal flank at equal intervals along x). A typical normalized data set for a single blade (six flanks) is plotted

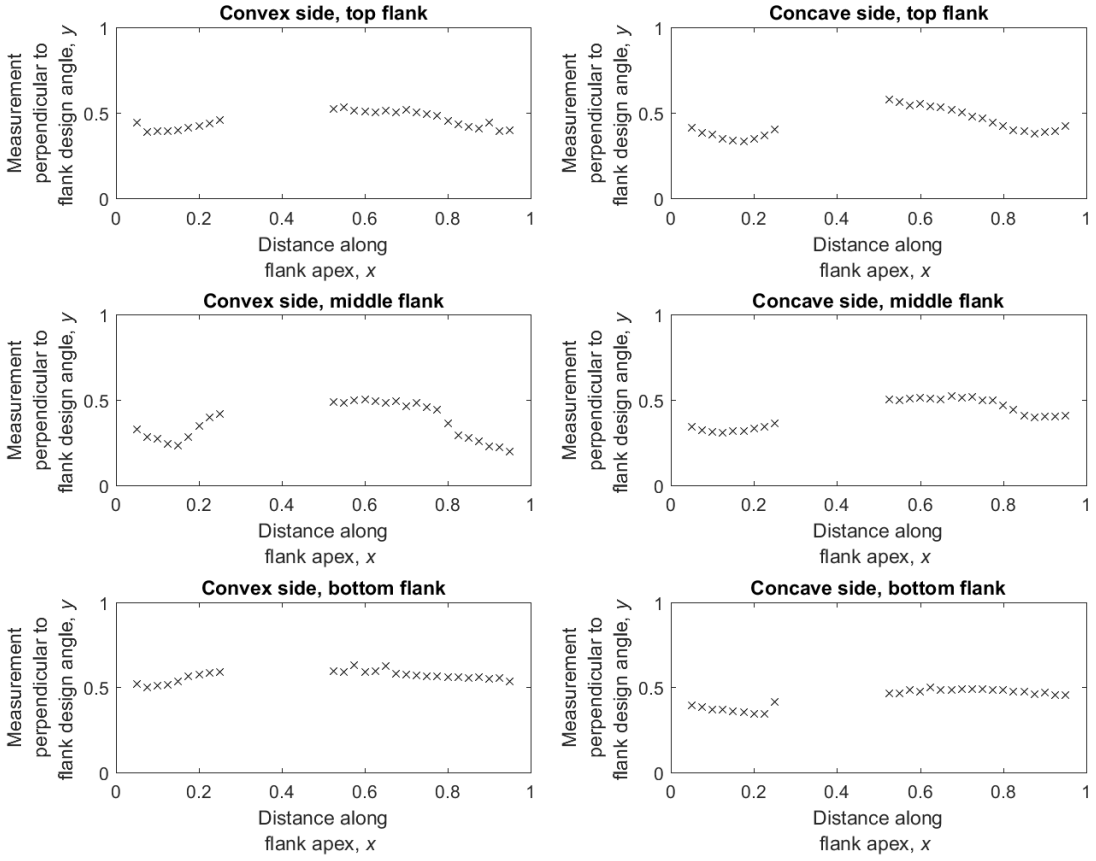


Figure 6.3: Example normalized data set for a single blade (six flanks) (normalized for protection of sensitive information, nominal flank line at y=0.5)

in figure 6.3. The gaps in the data are due to the clamping fixture used during use of the CMM.

By inspection, it is clear that the nominally straight line apex is not, in fact, straight at all. The most appropriate analytical representation needs to take account of:

(a) curve position (registration),

(b) measurement noise, and

(c) any gaps or non-uniformity in the measurement point locations.

In addition, the parameters defining the curve must:

(d) be small in number for practicable uncertainty studies, and

(e) ensure accuracy of fitted statistical parameter distributions.

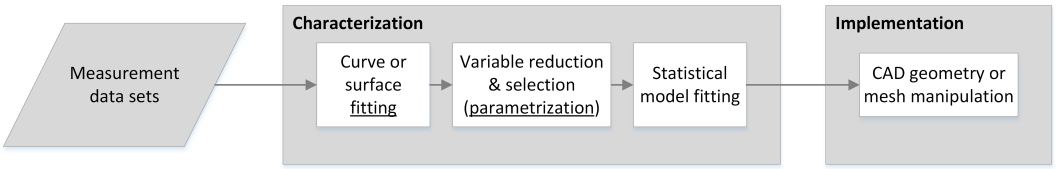


Figure 6.4: Extended steps as a precursor to geometric uncertainty propagation and robust design optimization

Approaches can either maintain or eliminate correlations between parameters, but if variables can be selected so as to be independent then subsequent uncertainty studies can be greatly simplified.

Where, in chapter 4, the characterization process is one of curve fitting alone, here this step is split into 'fitting' and 'variable reduction and selection' processes, figure 6.4. The fitting task ensures the accuracy of the initial measured data by addressing (a) to (c), and the reduction and selection process meets requirements (d) and (e).

Once suitable parameters with accurately determined statistical properties are known, their distributions can be sampled within an UP or RDO framework, facilitated through modifications to the CAD model or mesh.

6.2 Curve fitting

It is proposed that Kriging, or surrogate model, based curve fits can be utilized to both effectively model measured curves (with known measurement error magnitude), and define an appropriate set of variables for uncertainty analysis.

6.2.1 Representing flank 'pitch' or 'crown offset' (registration)

An offset in the data compared with the nominal is observed. This could be attributed to an error in jig positioning or in the manufacturing of the individual flank, or to the presence of a discrepancy in the position of the flank relative to a fixed datum on the firtree root, i.e. an uncertainty in the firtree branch spacing, or 'pitch'. One must determine whether this pitch information can be extracted from the shape uncertainties, such that it might be considered independently during analysis; a problem of registration of the data.

The measurements are initially made at a distance, d, from a reference flank position and design angle, figure 6.2. y is then calculated by subtracting the nominal distance for each 'crown'. Clearly this results in the inclusion of any overall movement in the flanks relative to each other in the measurements. This is visible in the data, figure 6.3, as a

difference in the average vertical position of the curves with respect to each other. The 'true' difference between the flank apex positions is referred to as pitch.

If one considers that the data includes both a pitch change between flanks, as well as an angle along the flank, and a waviness consisting of shape and roughness, then one could remove either the offset due to pitch, or both the offset and the angle, from the feature prior to embedding the data in the geometry.

It is not possible to determine with certainty how much of the data offset from nominal can be attributed to the surface variation, and how much can be attributed to distortion of the overall firtree shape, i.e. flank pitch, twist etc. Indeed, if one was to consider studying the effects of such changes, measurement of the related parameters, such as notch radii and other overall shape parameters would be required. Therefore, a pragmatic approach is taken; the angular variation is considered to be part of the local flank shape and is not filtered from the data but the overall shift, or pitch, is removed by locating the mean position of a linear regression line and data points on the nominal flank apex, figure 6.5.

6.2.2 Accounting for measurement error

Classical analytical curve fitting

Observed in the data sets, as in most CMM data sets, are two types of shape information: the large scale trends in the data, and the smaller scale surface roughness. It is assumed that the large-scale trends are correct; the accuracy of the measurement equipment is sufficient to represent these. However, measurement data are subject to noise due to the inherent inaccuracies of the machinery, environmental factors, and human factors, in addition to the true surface roughness of the part. In order to fit a curve to the data one must decide on an approach to fitting that is suitable given the intended use of the fitted curve and what is known about the accuracy of the measurements.

Initially, polynomial regression models are fitted. It is determined using the null hypothesis discussed in section A.7.1 that, for an example blade, a polynomial with degree greater than four would be likely to result in over-fitting (the convergence of the variance is plotted for six flanks, figure 6.6).

Polynomials of degree one (linear) through to degree four are fitted through the data sets using classical least squares (see section A.7.1). For comparison, an 'over-fitted' degree nine polynomial fit is also applied. We also apply a smoothing spline fit through the data, and an interpolating spline for comparison (see section A.7.3). The fits are shown for three example flanks, figures 6.7, 6.8, and 6.9.

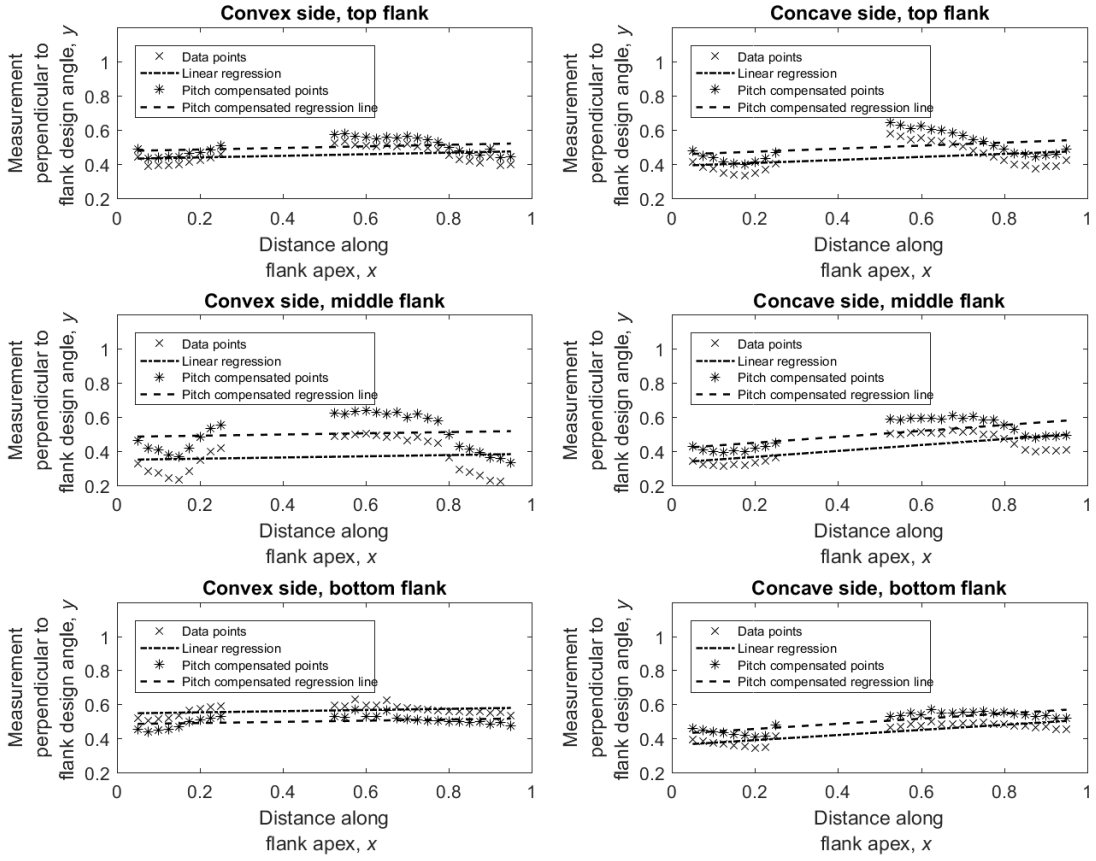


Figure 6.5: Example data set for a single blade (six flanks) (normalized for protection of sensitive information) showing linear regression line and offset to compensate for flank pitch

It is clear that interpolation of the points can produce anomalies when the curves are extended into regions without measurement (the extremities and the gap). This can be avoided to some extent with the regression fits. However, the spline smoothing factor is somewhat arbitrary, and the fixed polynomial curve form defines the resulting variance of the deviations of the points from it. In reality, a 'real' value for the variance of measurement data points is, at best, obtainable from tests of the measurement system, and at worst, expected to be consistent over all the flanks. Therefore, we turn to an approach used extensively in optimization for meta-modelling: Kriging (see section 2.2).

A regressing Kriging-based curve fit with known noise

Kriging is a specific form of radial basis function modelling that assumes that data points are drawn at random from a Gaussian normal distribution (section 2.2.1). Given that error in measurement manifests as noise in the data points, it is reasonable to first consider regressing Kriging, section 2.2.2, where this 'noise' is represented by a term λ.

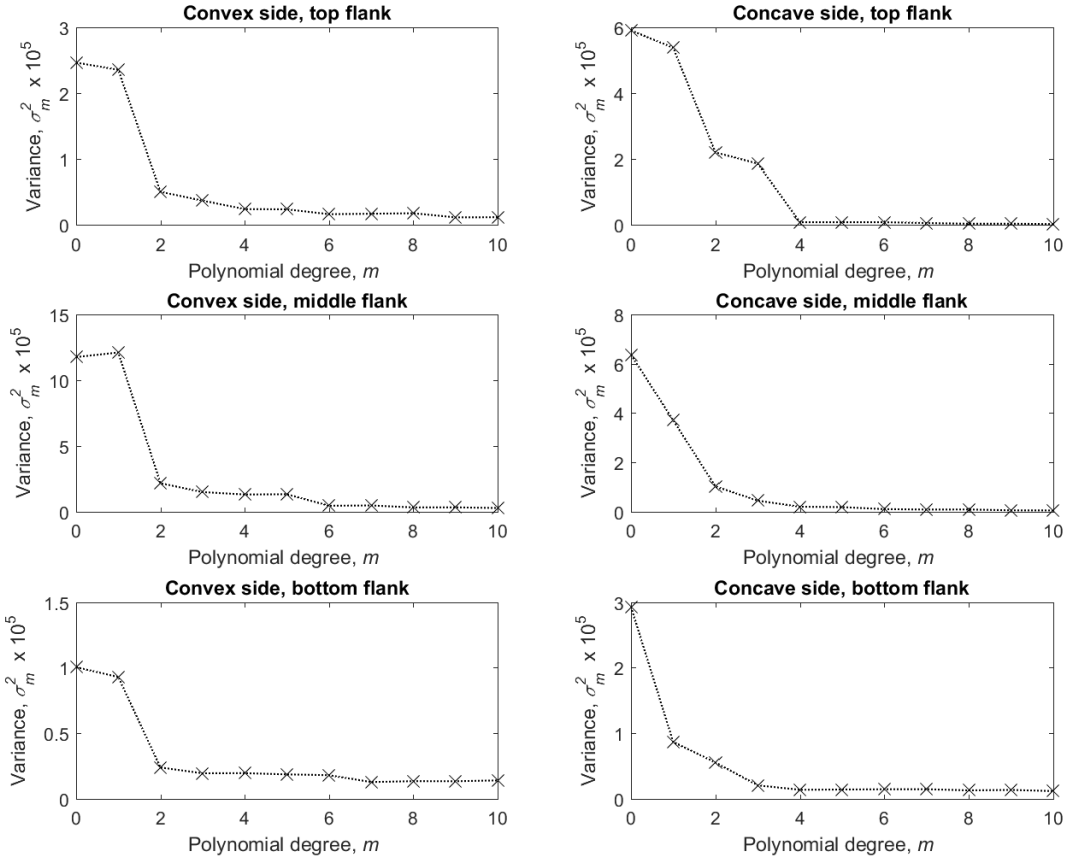


Figure 6.6: Decline of the combined data point variance, σ_m^2, from the fitted model with increasing polynomial degree (example data set for a single blade (six flanks))

This approach has been employed, for modelling digital image based measurements, by Wang et al. [303]. In later work Wang et al. [302] allow for known variations in error magnitude by applying a varying scaling factor to the regression term. However, λ has no direct relationship with the value of the noise magnitude (or its variance) and needs to be estimated along with other parameters of the Kriging model. Here, instead, we look to a refinement to the regressing Kriging algorithm where it has been modified to define the noise explicitly as a known, or estimated, variance. In the case of measurement points one may at least know the typical degree of uncertainty for a particular measurement type, or may even have a value of the expected magnitude of the error.

Therefore, we take the approach of Picheny et al. [207] and consider y_i's to be realizations of $\tilde{Y}_i := Y_i(\mathbf{x}^{(i)} + \varepsilon_i)$ where ε_i are normally distributed independent measurement errors, then one can consider the expressions thus far in terms of τ^2, the variance of the measurement error (noise) variable.

Picheny et al. [207] define the Kriging predictor in terms of a matrix of covariance kernels,

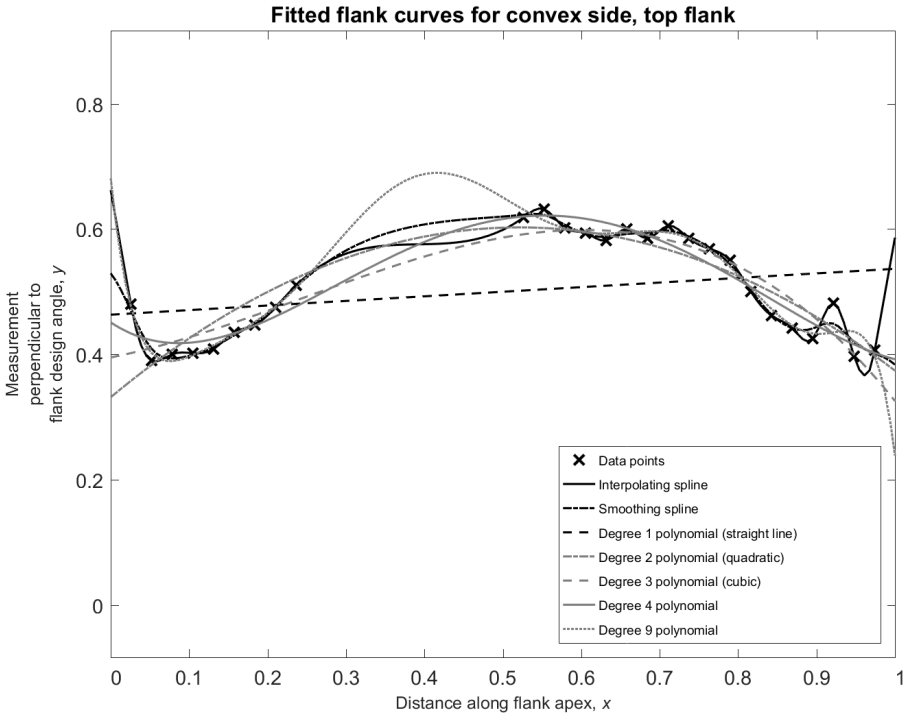


Figure 6.7: Data points and seven different flank apex curve fits for example convex side top flank (normalized)

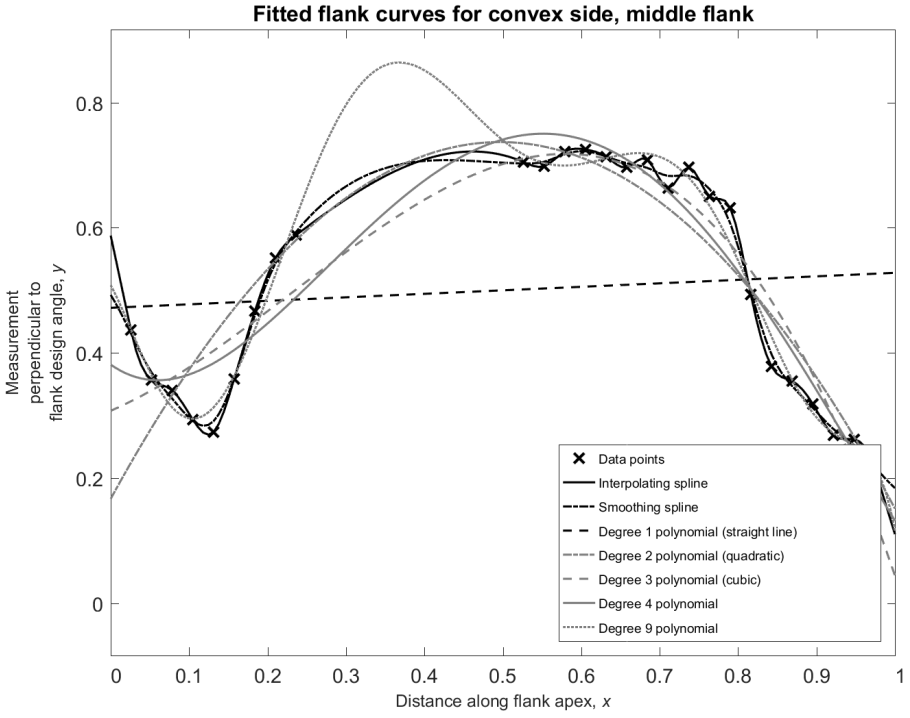


Figure 6.8: Data points and seven different flank apex curve fits for example convex side middle flank (normalized)

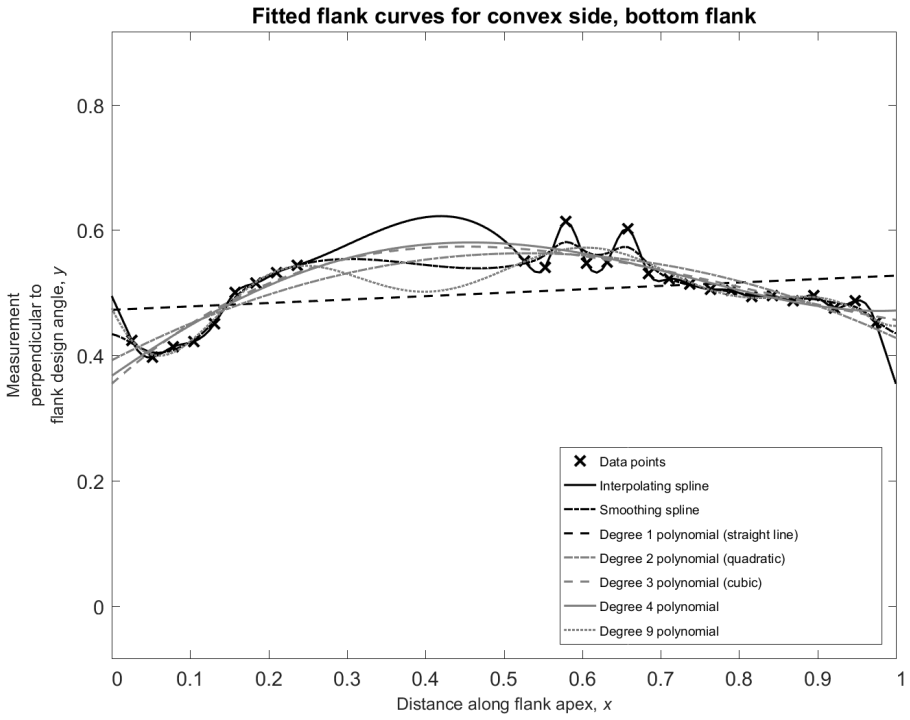


Figure 6.9: Data points and seven different flank apex curve fits for example convex side bottom flank (normalized)

$\boldsymbol{K}$, where $k_{ij} = \sigma^2\psi_{ij}$. Thus $\boldsymbol{K}$ is equal to $\mathbf{V}$ in section 2.2. According to Picheny et al. [207], the noise variance τ^2 can be added to the leading diagonal of this covariance matrix;

$$\tilde{\mathbf{V}} = (\mathbf{V} + \tau^2\mathbf{I}) = (\sigma^2\boldsymbol{\Psi} + \tau^2\mathbf{I}) \quad = \begin{bmatrix} \sigma^2+\tau^2 & \sigma^2\psi_{1,2} & \sigma^2\psi_{1,3} & ... & \sigma^2\psi_{1,n} \\ \sigma^2\psi_{2,1} & \sigma^2+\tau^2 & ... & ... & ... \\ \sigma^2\psi_{3,1} & ... & \sigma^2+\tau^2 & ... & ... \\ ... & ... & ... & ... & ... \\ \sigma^2\psi_{n,1} & ... & ... & ... & \sigma^2+\tau^2 \end{bmatrix}, \quad (6.1)$$

in much the same way as the 'noise parameter' λ can can be added to the leading diagonal of the correlation matrix, $\boldsymbol{\Psi}$. This parameter τ^2, however, has physical meaning, where λ does not. Where, for an interpolating model one could divide through the covariance matrix by σ^2 (equation 2.11), to result in a correlation matrix $\boldsymbol{\Psi}$ independent of σ^2, this is now not possible. Therefore, instead of substituting this expression for $\tilde{\mathbf{V}}$, one maintains $\tilde{\mathbf{V}}$ in the likelihood. A solution for $\hat{\mu}$ can then be expressed as

$$\hat{\mu} = \frac{\mathbf{1}^T\tilde{\mathbf{V}}^{-1}\mathbf{y}}{\mathbf{1}^T\tilde{\mathbf{V}}^{-1}\mathbf{1}}, \quad (6.2)$$

by taking derivatives with respect to μ, but an analytical expression for $\hat{\sigma}^2$ and $\hat{\tau}^2$ in terms of $\tilde{\mathbf{V}}$ cannot be found. The concentrated ln-likelihood function then becomes

$$\ln(L) \approx -\ln(|\tilde{\mathbf{V}}|) - (\mathbf{y} - \mathbf{1}\mu)^T \tilde{\mathbf{V}} (\mathbf{y} - \mathbf{1}\mu), \tag{6.3}$$

[207]; a function of σ^2 and τ^2 which are both parameters to be estimated using the preferred optimization routine along with $\boldsymbol{\theta}$ and $\mathbf{p}$. Following through the same logic, the predictor (also known as the Kriging mean) becomes

$$\hat{y}(\mathbf{x}^*) = \hat{\mu} + \mathbf{v}^T \tilde{\mathbf{V}}^{-1} (\mathbf{y} - \mathbf{1}\hat{\mu}), \tag{6.4}$$

and the error (also known as *Kriging variance*) can be evaluated as

$$s^2(\mathbf{x}^*) = \sigma^2 - \mathbf{v}^T \tilde{\mathbf{V}}^{-1} \mathbf{v} + \frac{(1 - \mathbf{v}^T \tilde{\mathbf{V}}^{-1} \mathbf{1})^2}{\mathbf{1}^T \tilde{\mathbf{V}}^{-1} \mathbf{1}}, \tag{6.5}$$

where $\mathbf{v}$ is a vector of correlations between the existing points and the new point at $\mathbf{x}^*$ multiplied by σ^2.

Therefore, in a problem where noise exists in the data points, one can consider including noise as a parameter, λ, in the correlation matrix, with the advantage that computational effort can be reduced with the ability to evaluate σ^2 analytically. If, however, one wishes to determine an estimate of its value, or input a known variance, then noise can be added to the covariance matrix as τ^2. Its value, along with the value of σ^2, can then be determined by numerical optimization given suitable bounds. The same approach can be taken if the data point variance τ^2 is known, and only σ^2 is then determined by numerical optimization.

This methodology could potentially be extended to problems where the noise varies depending on the data point, in which case λ and τ^2 become diagonal matrices; diag($[\lambda_1 ... \lambda_n]$) and diag($[\tau_1^2 ... \tau_n^2]$) [207], rather than constants: the *heterogeneous* case. This can lead to problems of over-fitting and an alternative based on bootstrapping [37] is discussed by Yin et al. [322].

However, one discovers that the process of maximizing the likelihood in order to determine τ^2 and σ^2 is sensitive to the range of values allowed for their calculation; plotting the negative ln-likelihood over a range of values of τ^2 and σ^2 for an individual problem shows that there is not a clear minimum, figure 6.10, where a similar exercise using λ yields clear results and a robust solution, figure 6.11. It is noted that one finds that the value of σ^2 is the same, to the computational precision of 5 significant figures, whether the likelihood is maximized by specifying λ, or τ^2 and σ^2. The explanation is thus:

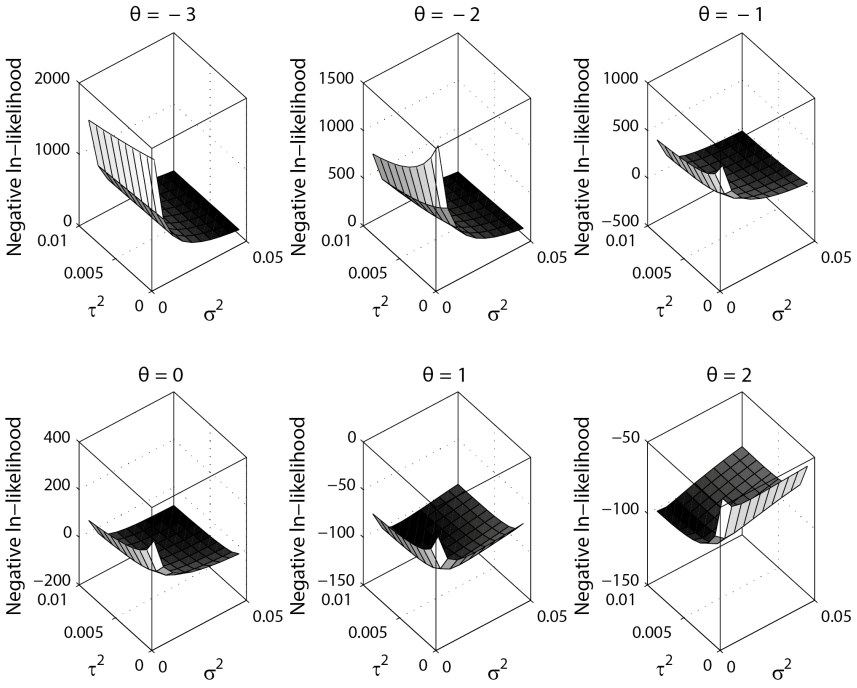


Figure 6.10: Negative ln-likelihood plotted for a range of σ^2 and τ^2 values for fixed θ

$\tilde{\mathbf{V}}$ can be divided through by $\sigma^2 + \tau^2$ such that

$$(\sigma^2+\tau^2)^{-1}\tilde{\mathbf{V}} = \begin{bmatrix} 1 & \frac{\sigma^2}{\sigma^2+\tau^2}\psi_{1,2} & \frac{\sigma^2}{\sigma^2+\tau^2}\psi_{1,3} & \dots & \frac{\sigma^2}{\sigma^2+\tau^2}\psi_{1,n} \\ \frac{\sigma^2}{\sigma^2+\tau^2}\psi_{2,1} & 1 & \dots & \dots & \dots \\ \frac{\sigma^2}{\sigma^2+\tau^2}\psi_{3,1} & \dots & 1 & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \frac{\sigma^2}{\sigma^2+\tau^2}\psi_{n,1} & \dots & \dots & \dots & 1 \end{bmatrix}, \tag{6.6}$$

and if τ^2 is significantly smaller that σ^2 then the term $\frac{\sigma^2}{\sigma^2+\tau^2}$ tends to 1 and

$$\tilde{\mathbf{V}} \approx (\sigma^2+\tau^2)\mathbf{\Psi}. \tag{6.7}$$

If this term is expressed as $(1-\nu)$ where $\nu = \frac{\tau^2}{\sigma^2+\tau^2}$, then τ^2 is known as the *nugget* and ν is the *scaling factor* [207]. Equation (2.15) then becomes an estimate for $\sigma^2+\tau^2$ where $\sigma^2+\tau^2 \approx \sigma^2$ if τ^2 is very small or the data are interpolated.

Given this knowledge, one can improve the robustness of the fitting routine with noisy data where τ is of interest by either

- considering carefully the bounds for σ^2 and τ^2, or scaling $\mathbf{y}$ such that bounds for σ^2 and τ^2 don't vary between data sets, or

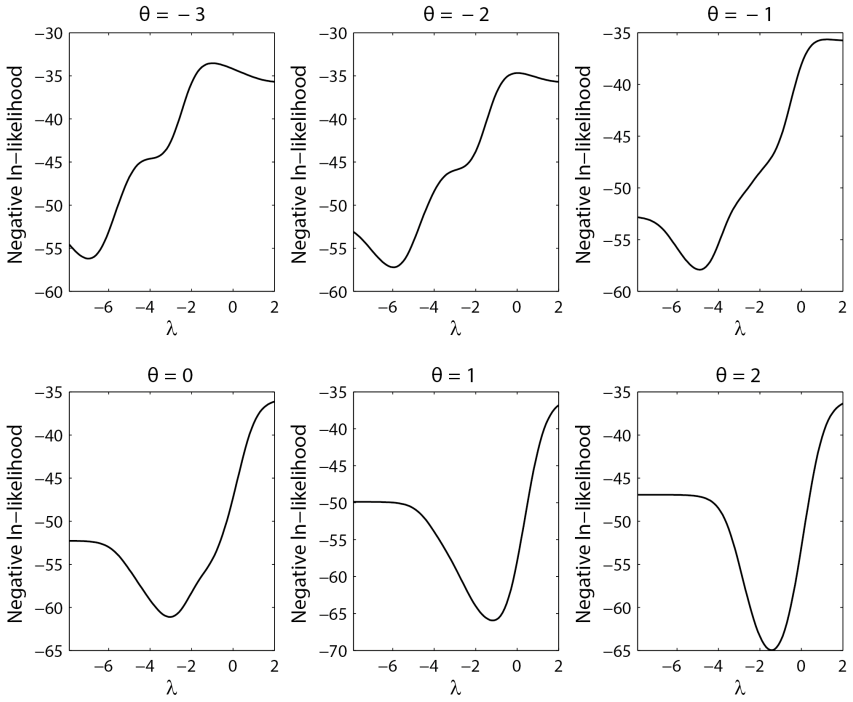


Figure 6.11: Negative ln-likelihood plotted for a range of λ values for fixed θ

- considering τ^2 to be very small, calculating σ^2 using the λ approach, and then fixing the value of σ^2, or performing a further local search around the σ^2 and θ values yielded.

The second approach will ensure stability of the process for different data sets. However, given that the data sets involved in this problem are similar in magnitude for both $\mathbf{y}$ and uncertainty in $\mathbf{y}$, it is considered sufficient to scale $\mathbf{y}$. One can test this approach by performing the global optimization routine multiple times for the same data set. The same model is reached for all trials. It is also clear that both approaches; with λ, or σ^2 and τ^2, should result in the same line if τ^2 is not fixed, as illustrated by the fit on a sample data set, figure 6.12.

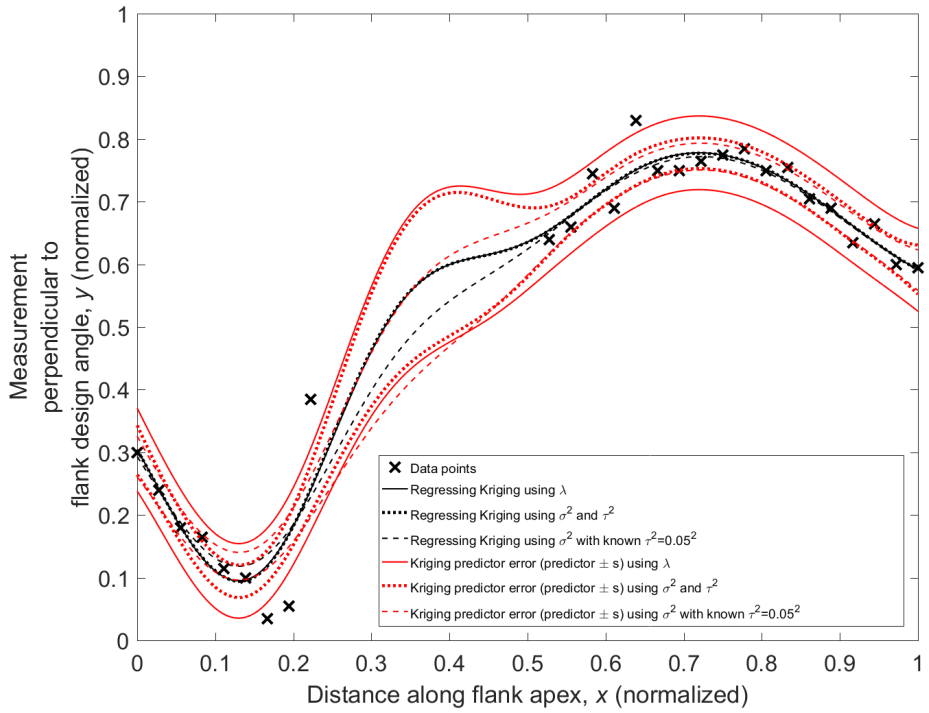


Figure 6.12: Kriging fit for example noisy one-dimensional data set, optimizing ln-likelihood by varying different hyperparameters

6.2.3 Universal Kriging for dealing with gaps in measurement data

Given that a section of the data set is 'missing' (caused by the fixture holding the component in place during measurement), and that data exists from 0.05 to 0.95 in x when normalized but the ends of the flank extend from 0.00 to 1.00, the curve fit must appropriately fill the gaps. The classical approaches and ordinary or regressing Kriging all 'fall short' in this respect as they either result in curve behaviour in these regions that is overly dependent upon local trends at each end of the missing patch, or tend to return to the mean value. To address this difficulty one can consider using universal Kriging [44, 79]. Universal Kriging uses a mean term, $\hat{\mu}$, that is a function of $\mathbf{x}$. This term allows an overall known trend in the data to be included. For typical surrogate modelling problems this is unpopular because one does "not usually have apriori knowledge of the trends in the data and specifying them may introduce inaccuracies" [79]. However, if an appropriate function is chosen, this mean term can 'guide' the model through regions of missing data.

If the mean term is considered to be an m-order polynomial regression with coefficients α_i for $i = 1, ...m$ then

$$\hat{\mu}(\mathbf{x}) = \sum_{i=0}^{m} \alpha_i \nu_i(\mathbf{x}). \tag{6.8}$$

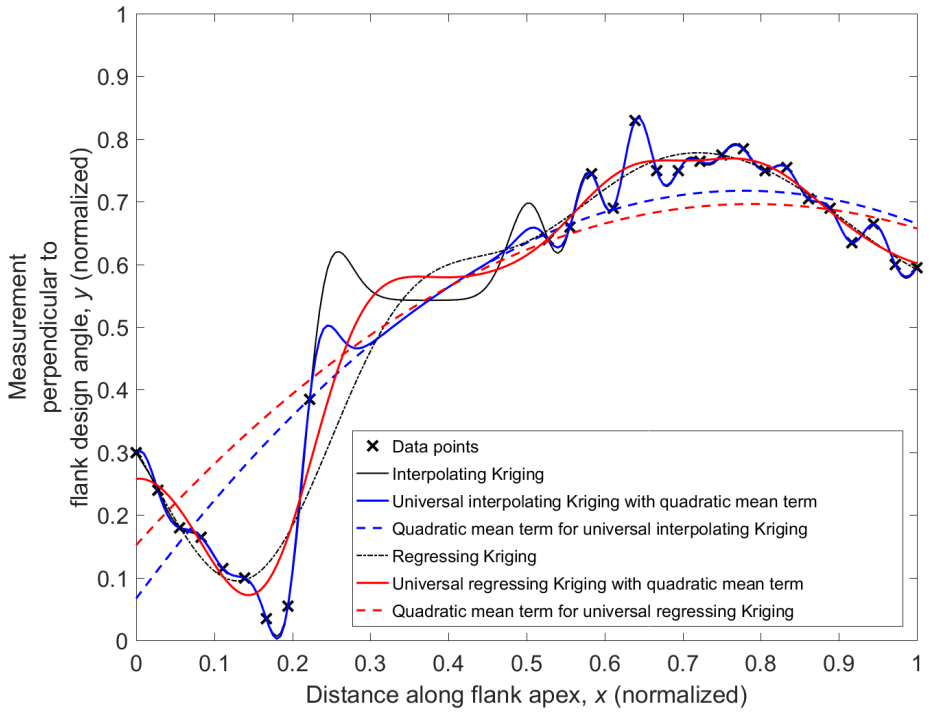


Figure 6.13: Universal Kriging models and associated mean functions for example noisy one-dimensional data set

It makes sense for the order of the polynomial to remain low to avoid over-fitting as previously discussed. A quadratic polynomial is used here as a 'guide', or 'trend' for the universal Kriging model. Following from equation 2.14, the non-biased linear generalized-least-squares estimator for $\hat{\boldsymbol{\alpha}}$ becomes

$$\hat{\boldsymbol{\alpha}} = \frac{\boldsymbol{\nu}^T \boldsymbol{\Psi}^{-1} \mathbf{y}}{\boldsymbol{\nu}^T \boldsymbol{\Psi}^{-1} \boldsymbol{\nu}}, \tag{6.9}$$

where $\boldsymbol{\nu}$ is a matrix of polynomial terms calculated at each $\mathbf{x}$. The model variance and predictor can then be calculated by replacing μ with $\boldsymbol{\mu} = \boldsymbol{\alpha}\boldsymbol{\nu}$ in equations (2.15) and (6.10). The resulting models and associated quadratic mean functions for an example normalized data set are plotted, figure 6.13.

Similarly to ordinary Kriging, the noise value can be specified by setting the value of τ^2. Figure 6.14 shows the resulting model as τ is varied from near-interpolation at $\tau = 0.05$ to $\tau = 0.1$.

Given that the data in this problem contains some noise, that the level of that noise is unknown, but that a value of noise is informative, it is appropriate to use regressing Kriging with σ^2 and τ^2 as parameters, rather than λ. Given also that there is a large gap in the data, it is appropriate to use universal Kriging, rather than ordinary Kriging. The resulting fits for three example flank data sets are shown, figures 6.15, 6.16, and 6.17. This is a novel approach in its application to geometry modelling.

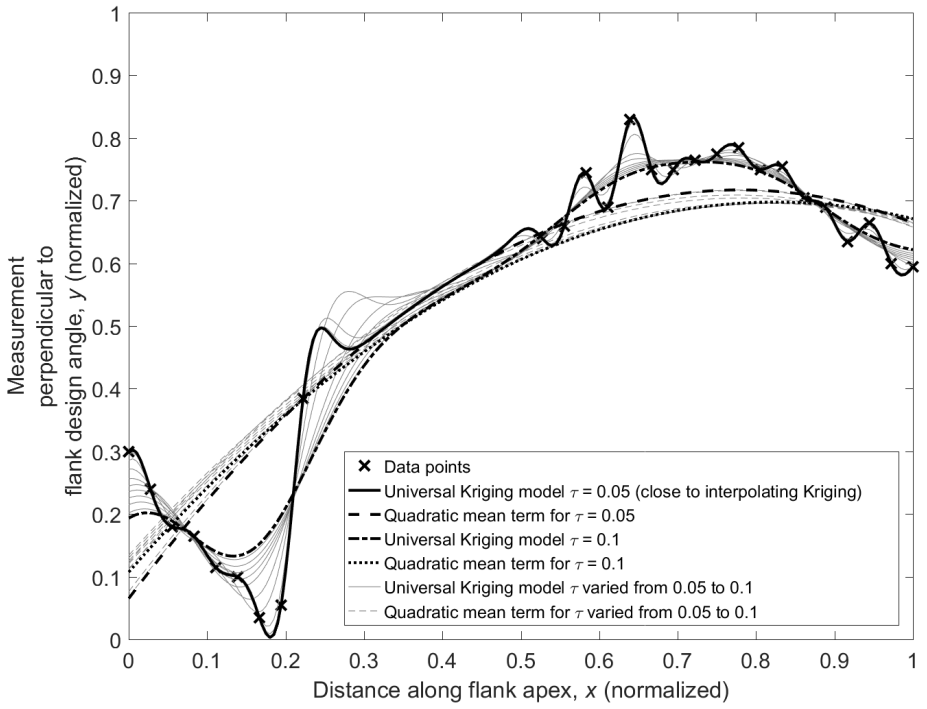


Figure 6.14: Universal Kriging models with specified noise values with associated mean functions for example noisy one-dimensional data set

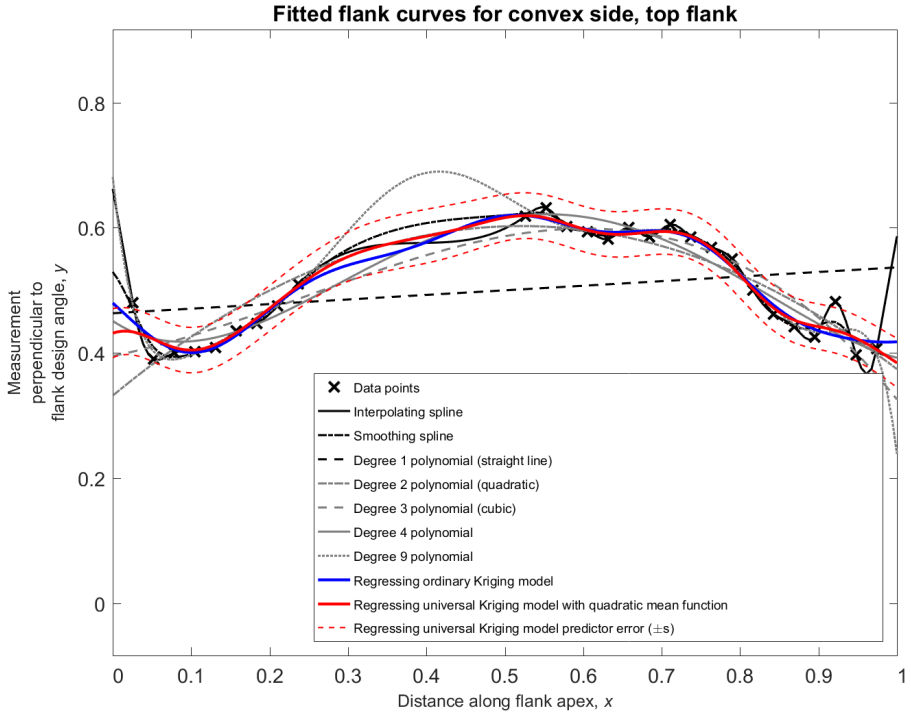


Figure 6.15: Data points and nine different flank apex curve fits for example convex side top flank (normalized)

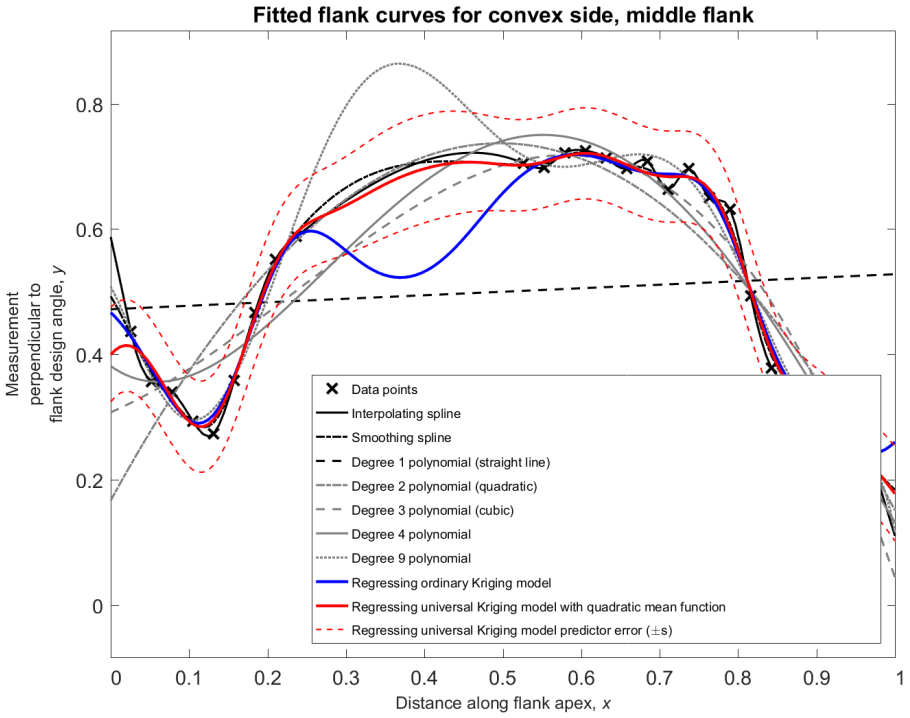


Figure 6.16: Data points and nine different flank apex curve fits for example convex side middle flank (normalized)

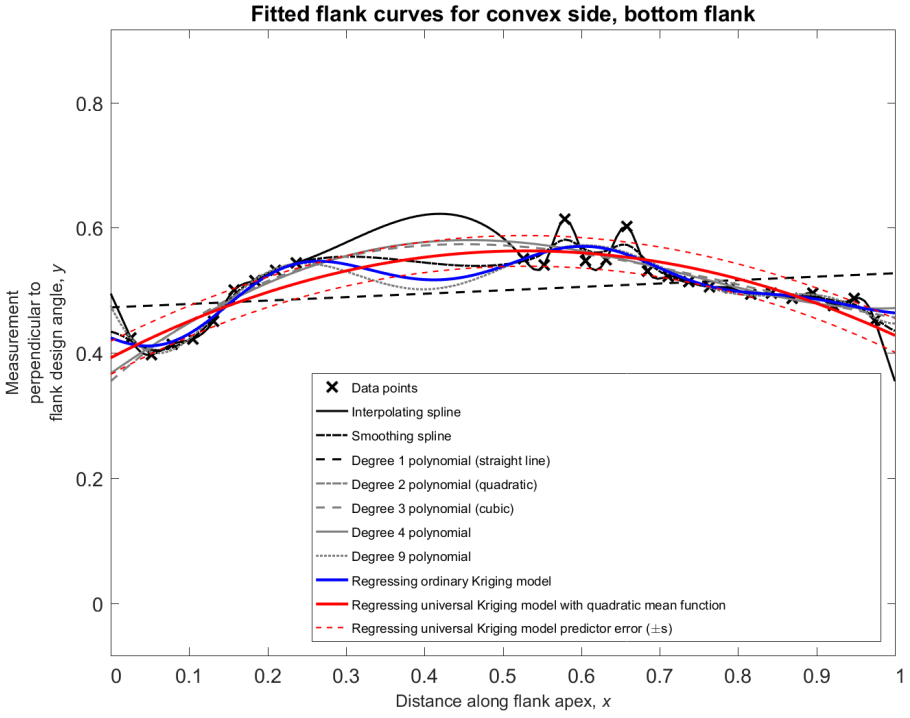


Figure 6.17: Data points and nine different flank apex curve fits for example convex side bottom flank (normalized)

There is no perfect solution to this fitting process; a gap over a large proportion of the surface will always require some level of 'guesswork'. Here, although the universal Kriging approach still results in a slightly uncharacteristic dip in some surfaces, this is considered the most appropriate routine. Only further measurements could refine this. Assuming that we have an accurate depiction of the curve defining the deviation from the nominally straight barrelled flank surface for a number of flanks, a set of appropriately defined uncertain variables is now required.

6.3 Variable reduction and selection for statistical model fitting

Observation of multiple sets of the fitted curves, such as those in figure 6.18, shows distinct trends for each of the flanks. Given that one can observe these trends 'by eye', one can hypothesize that it should be possible to represent these curves and their variation (uncertainty) mathematically.

Each of the existing fitted curves are defined by the Kriging predictor (or Kriging mean):

$$\hat{y}(\mathbf{x}^*) = \hat{\mu} + \boldsymbol{\psi}^T \boldsymbol{\Psi}^{-1}(\mathbf{y} - \mathbf{1}\hat{\mu}), \tag{6.10}$$

which is a function, in a single dimension, of θ and $\mathbf{y}$ (a vector of 27 measurement point locations in y; the response vector), a regression parameter (either τ or λ), and σ^2. Two processes are proposed for parameterizing the curves to enable them to be sampled using many fewer variables, and then for 'pseudo-curves' to be created for implementation in geometric models. The first process is described by the flowchart in figure 6.19 and section 6.3.1. The second possible process is described by figure 6.20 and section 6.3.2. Both approaches propose interpolating refits of the Kriging model such that the regression parameter (τ or λ) is removed, and then the size of the response vector $\mathbf{y}$ is reduced, whilst also attempting to ensure that it is informative for uncertainty modelling (statistical models can be fitted to the resulting variables).

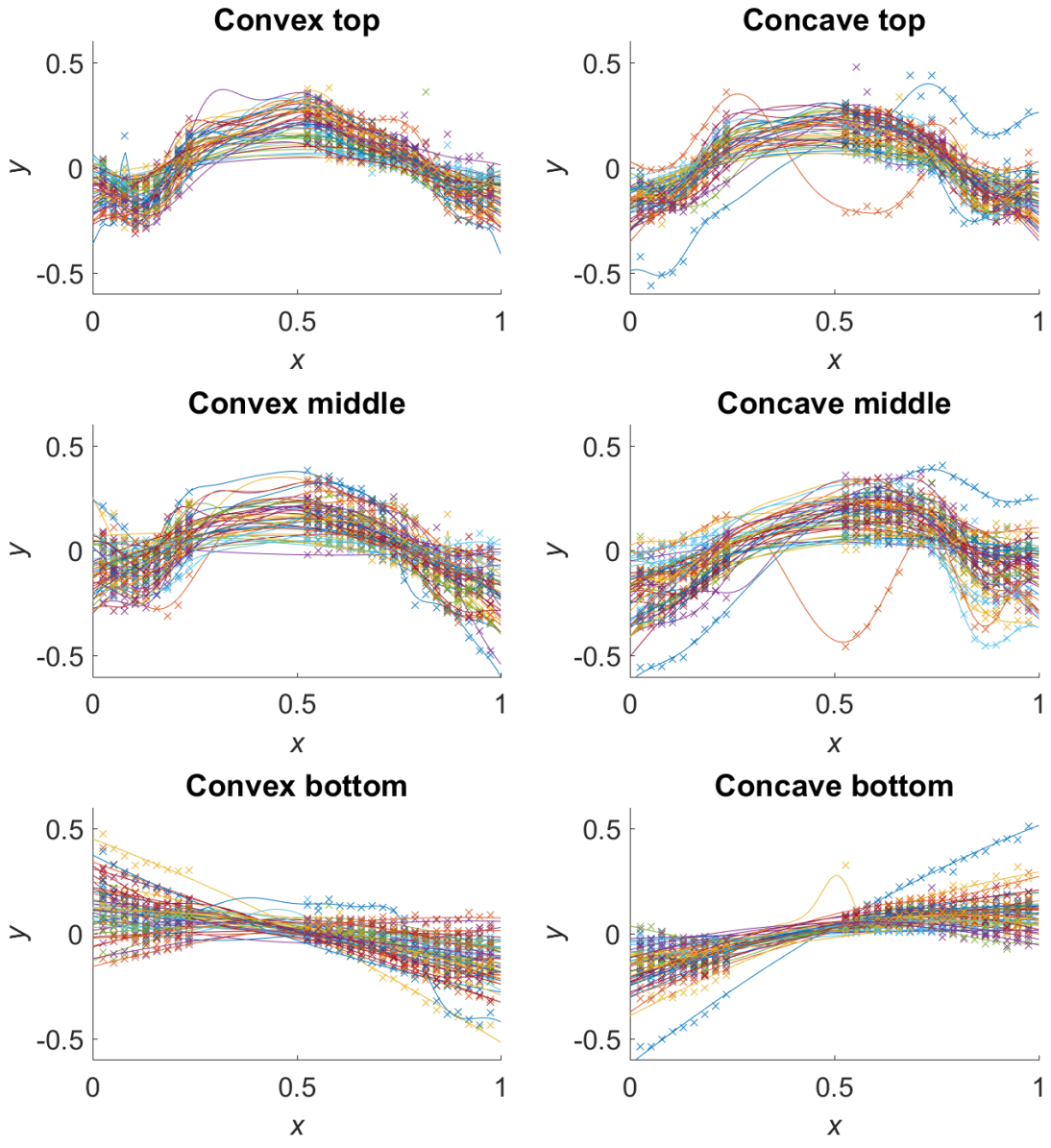


Figure 6.18: Measured curves (all 46 blades), fitted using universal regressing Kriging

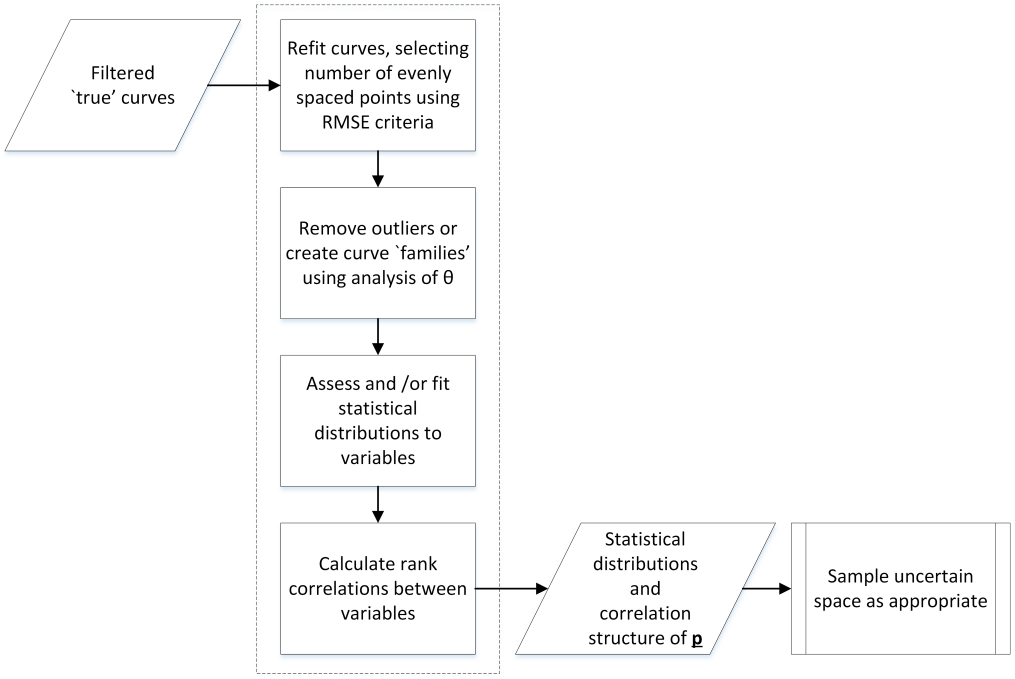


Figure 6.19: Proposed workflow for parametrization of uncertain curves using equally spaced correlated point variables

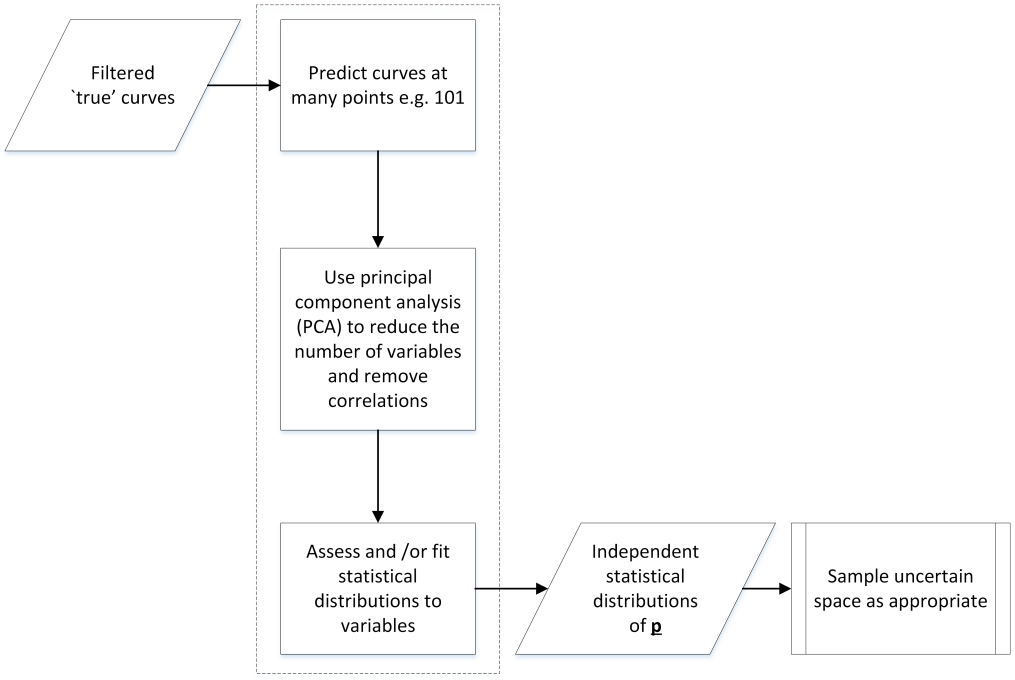


Figure 6.20: Proposed workflow for parameterization of uncertain curves using principal component analysis (PCA) to determine uncorrelated point variables

6.3.1 Parameterization of uncertain curves using correlated points

Here, the measured curves are refitted using interpolating Kriging by evaluating the Kriging predictor at increasing numbers of evenly distributed points. The convergence of the curve fit is evaluated over the full set of 46 blades and 6 flanks. An example of the convergence of a single blade is shown in figure 6.21.

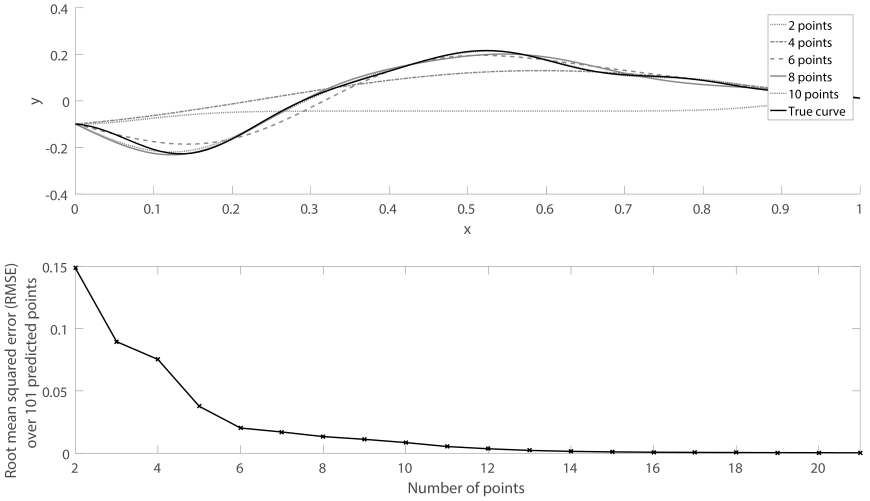


Figure 6.21: Convergence of curve refit using increasing numbers of evenly distributed points for a single blade firtree, convex side top flank

A convergence criterion can be set as, for example, RMSE<0.05. To specify the number of points required, we select the number of points for which 99.87% of curves reach the convergence criteria, i.e. curves requiring greater than three standard deviations above the mean number of points are discounted. This results in a requirement for 8 and 7, 9 and 8, and 4 and 4 points for the top, middle, and bottom flanks respectively. The refits based on this number of points are plotted for the convex side top flank, figure 6.22.

Plotting histograms of the eight sets of y-values at the points, figure 6.23, reveals an impression of their distributions, some of which appear Gaussian. Chi-square significance tests have been conducted to test the hypothesis that the samples of y have been drawn from a normal distribution with mean and standard deviation estimated by the sample values. The hypotheses cannot be rejected at the 5% level for any of the 8 point locations. A histogram of the fitted θs can also be plotted, figure 6.24. It is clear that there are a number of 'outlying' curves, where the θ value indicates curves with a different underlying shape (values changing along x at a significantly different rate, or smoothness). These are shown in figure 6.22 and removed for sampling. However, where there are significant numbers this could suggest more than one distinct 'family' of curves to be sampled.

Given a set of x locations, with corresponding distributions of y, it is now possible to sample y values, in order to create 'pseudo' curve sets, as required. It is clear, however

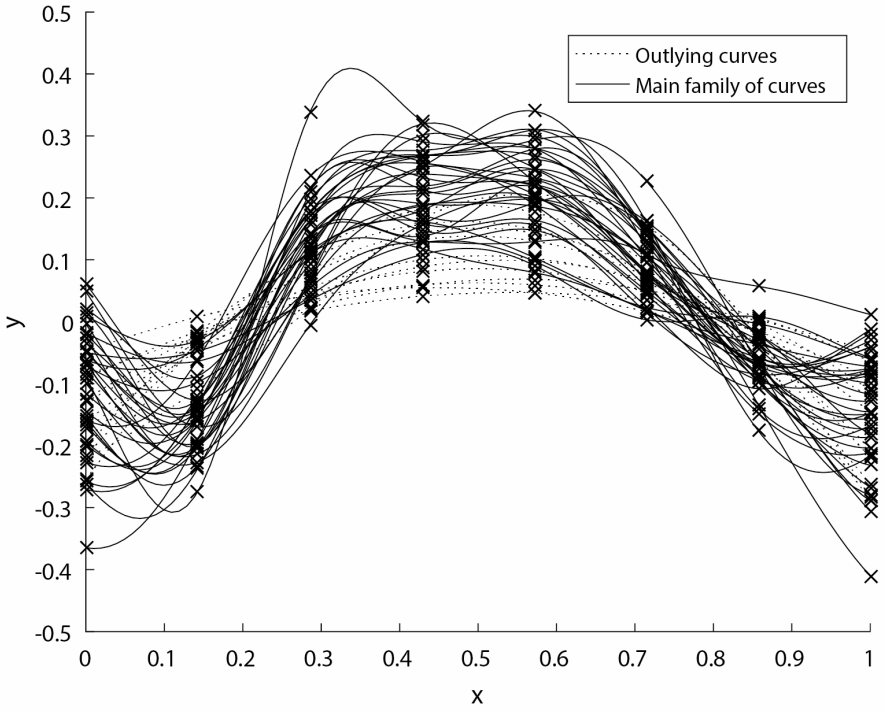


Figure 6.22: Measured firtree flank (convex top) curves refitted using 8 evenly distributed points (46 blades)

that there is not only correlation between the x-values as defined by $\mathbf{\Psi}$, but also in the values of y. The Iman-Conover approach to sampling with correlation is used, section A.8.4. To demonstrate the approach, we show a set of 30 'pseudo-curves' sampled randomly from the y distributions, figure 6.25. The correlation structure is shown using the scatter plot, 6.26. This set is used for demonstration purposes, but one could also sample using some kind of space-filling algorithm over the range of ys, thereby enabling models of the uncertain space to be made.

To demonstrate that the process works for alternative numbers of evenly spaced points, the process is repeated with a ninth point. The 'pseudo-curves' are plotted in figure 6.27 and correlation structure shown in figure 6.28.

This approach has been presented by the author of this thesis in a conference paper [81].

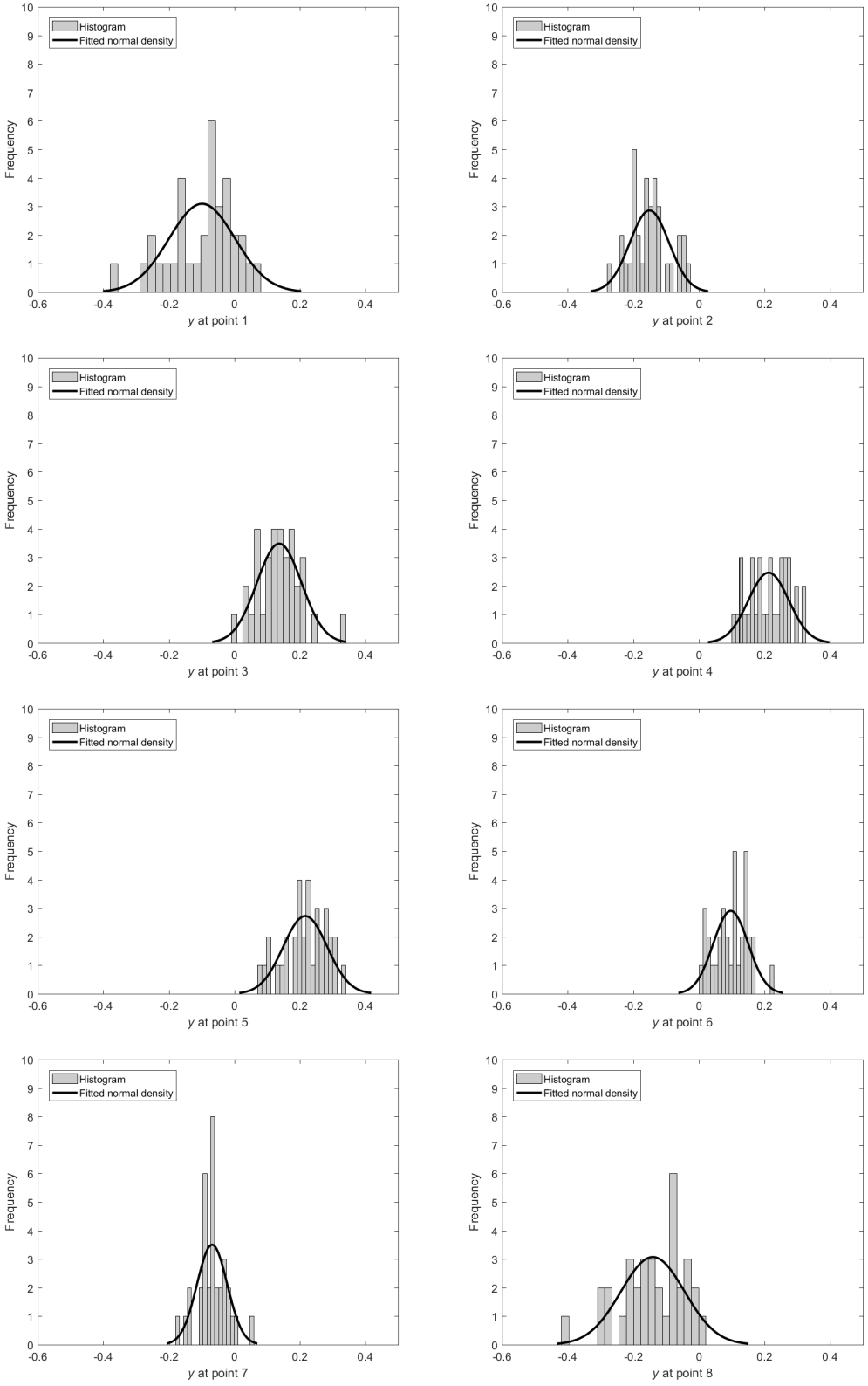


Figure 6.23: Histograms of y-values drawn from refitted firtree flank (convex top) curves with evenly distributed x-values (34 blades (outlying sets removed)).

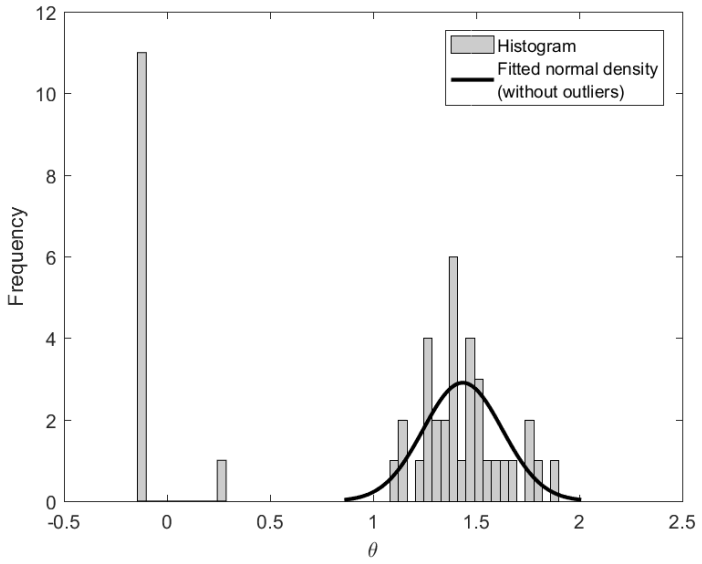


Figure 6.24: Histogram of θ values drawn from refitted firtree flank (convex top) curves with evenly distributed x-values (46 blades, 12 outliers)

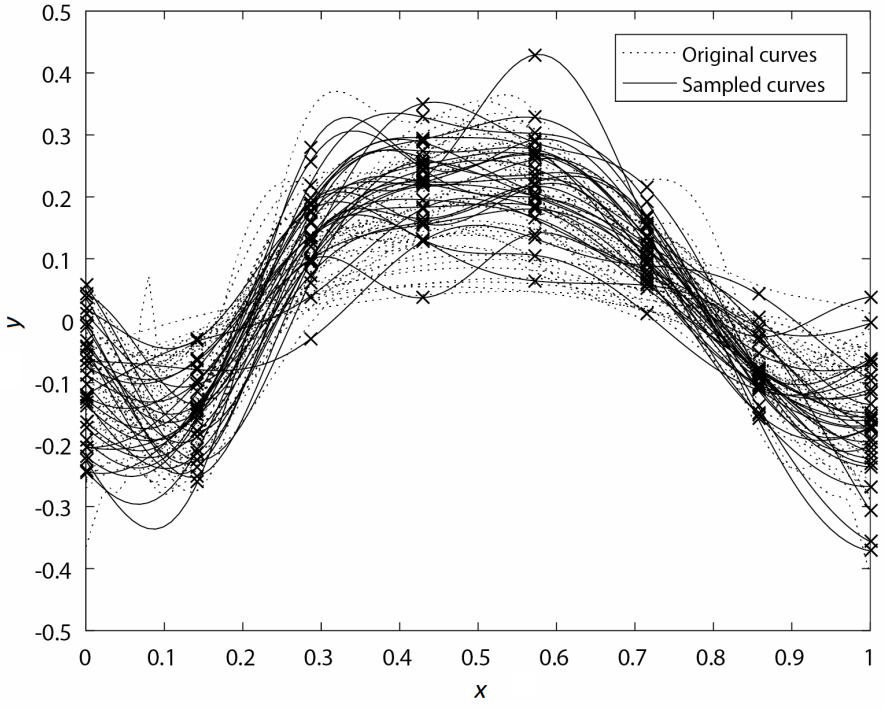


Figure 6.25: Random sample of 30 'pseudo-curves' from the determined extracted y-distributions using the Iman-Conover correlated sampling method (8 variables)

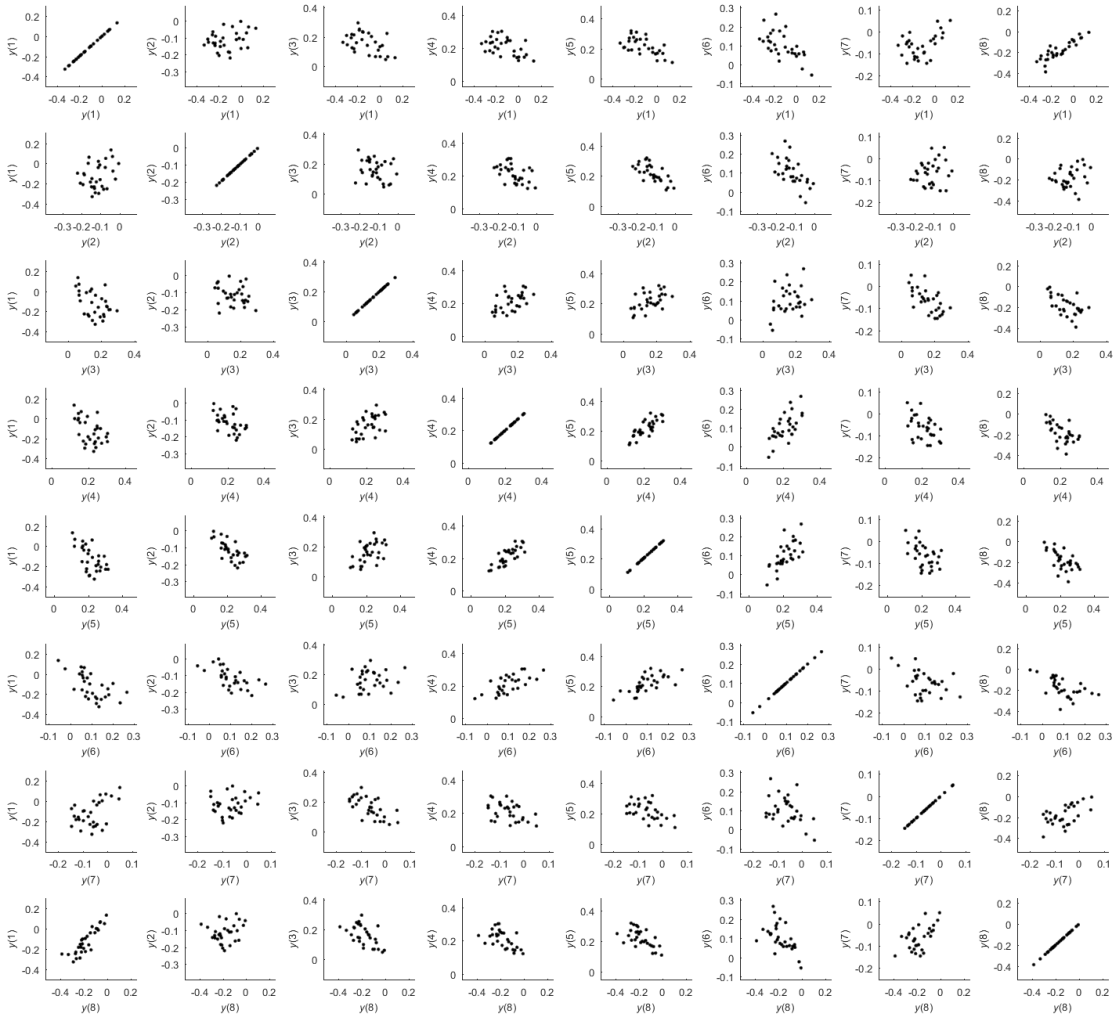


Figure 6.26: Scatter plot of 30-point Monte Carlo sample of curves using 8 correlated variables and the Iman-Conover correlated sampling method

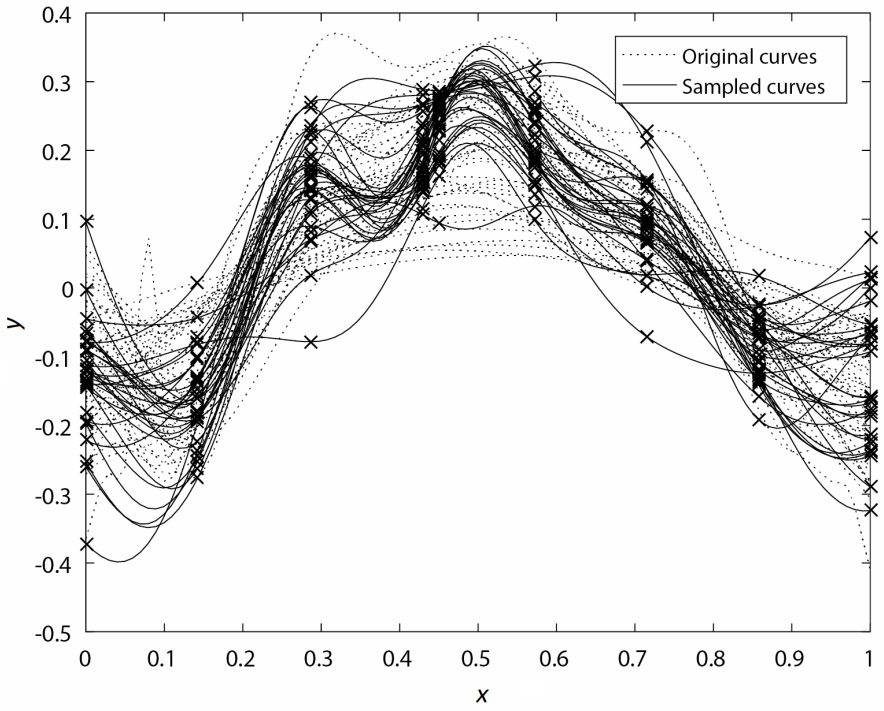


Figure 6.27: Random sample of 30 'pseudo-curves' from the determined extracted y-distributions using the Iman-Conover correlated sampling method (9 variables)

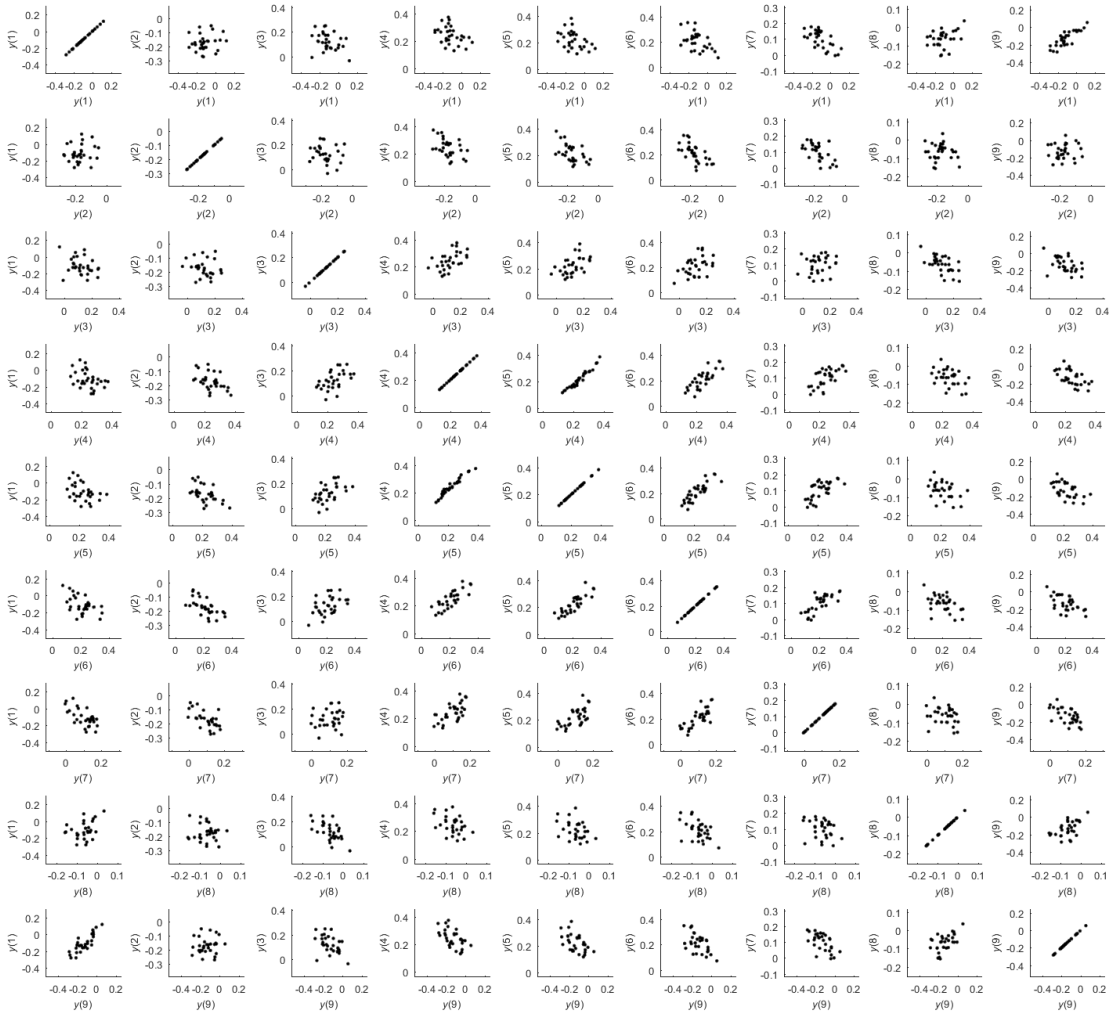


Figure 6.28: Scatter plot of 30-point Monte Carlo sample of curves using 9 correlated variables and the Iman-Conover correlated sampling method

6.3.2 Parameterization of uncertain curves using principal components

Section 6.3.1 results in reasonably well fitted parameters defining the uncertainty in the geometry, i.e. normal distributions can be fitted to the parameters with confidence at the 5% level once outliers are removed. However, the need to fit correlations between the variables, as well as fitting distributions, introduces an additional step in the process where inaccuracy can be introduced. A potential alternative is the use of principal components (PCs) instead of evenly distributed points locations. PCA is shown on known curves in appendix A.9.6, and then tested on the firtree flank measurement data sets here.

Considering the two-dimensional data sets from section 6.1, we have measurements representing curves along 6 flanks on each of 46 turbine blade firtree roots. If each flank has a different 'type' of curve, then there are 46 sets of measurement points defining each flank curve. There are 27 measurement points along each flank, i.e. 27 variables. In section 6.3.1 27 points were replaced by an equally spaced set of 8 alternative points and the hypothesis that the new points (variables) could have been drawn from a normal distribution could not be rejected. Sets of 'pseudo-curves' could then be drawn from these distributions using a correlated Monte Carlo sample [114]; an enabling methodology for uncertainty or robust design studies of the component, figure 6.19.

Now, we consider that PCA could be used beneficially in this case. In particular, determining a set of uncorrelated variables could be preferable and PCA might be considered a more general approach. We begin by taking the most straight-forward path: use PCA to reduce the existing 27 measured variables to a smaller number of principal components. For the data pertaining to the convex side top flank e.g. figure 6.18, we find that 78.5% of the variance can be explained by one PC. However, it takes 8 PCs to represent 97.5% of the variance, and 27 to represent it all. Histograms of each of these variables are plotted and normal and stable distributions [186, 187], described in appendices A.8 and A.9, are fitted, figure 6.29. It is not possible, at the 5% level, to reject the hypothesis that the sample has been drawn from a normal distribution for any of the PCs.

With this approach, if the distributions fitted were deemed reasonable and the number of PCs selected sufficiently represented the variance, a sample could be taken from these PCs, converted to original variables, and then the resulting 'pseudo-curves' fitted from these points, using a regressing Kriging model or alternative, to take account of the gap in the data. However, this approach means that statistical distributions are being fitted to data that is known to contain measurement error. It is proposed that a preferable approach is to remove measurement noise and any gaps prior to variable reduction.

If we begin by using our universal regressing Kriging models to predict the y-locations at a large number of defining points along each measured curve, then we start with a large set of y-locations that define each of the curves as a set of discrete point locations. Here,

we predict y-locations at 101 points. The process can now follow the process shown in figure 6.20.

Linear PCA is performed on the data sets. For the convex side top flank, the 'importance' of each PC (% of the variance that is explained by each component) is plotted in figure 6.30. The first 20 principal components explain 100% of the variance and the first 8 explain 98.9%[1].

Each of the principal components can be plotted against each of the others and outliers can be observed where points lie outside of the typical 'trend' line. For example, when the first component is plotted against the eighth component in figure 6.31 the first data set (component) is highlighted as an outlier. There is, however, a popular mathematical approach to identifying the outliers: using *Hotelling's* T^2 *statistic*. This statistic is a "measure of the multivariate distance of each observation from the center of the data set", [168]. However, no curves are identified as 'extreme' (there are no outliers) when applying Hotelling's T^2 statistic in this case.

Using linear PCA we can, therefore, represent all of the curves using principal components. The convergence of the RMSE of the resulting curve representations are shown in figure 6.32 for an example curve. It is observed that the RMSE is an order of magnitude smaller than the RMSE when evenly distributed points are used, figure 6.21. This is true for all data sets. A single curve is reconstructed using increasing numbers of PCs in figure 6.33, and a selection of convex top flank curve refits using 8 PCs are shown in figure 6.34.

In order to sample the uncertain space for UP using the PCs, reasonable fits to statistical distributions are needed. Using the convex side flank, histograms of the first 8 principal components are plotted with fitted normal distributions and stable distributions in figure 6.35. χ^2 goodness of fit tests are applied in all cases. The null hypothesis that the PCs are drawn randomly from a normal distribution with sample mean and sample variance cannot be rejected at the 5% level for PCs 1-4, and 7. The remaining three PCs cannot be fitted well using normal distributions, but, by observation, the stable distributions offer a clear improvement. Unfortunately, the χ^2 test statistic cannot be computed reliably for the stable distribution in these cases. These results might suggest that, in this case, normal distributions would be sensible where the normality hypothesis cannot be rejected at a reasonable level, but that a stable distribution be used, if it results in a visible improvement, where the hypothesis can be rejected. However, when the normality hypothesis cannot be rejected, the stable distribution tends to be almost identical to the normal distribution. When the hypothesis veers close to rejection, a stable distribution appears to represent the data better. This trend is observed for all flanks, some requiring fewer PCs to represent the same proportion of the variance. Given these observations,

[1]When the number of variables (or dimensions, N) is greater than the number of observations (n), all the variation in the observations can be fully explained using $(n-1)$ principal components i.e. 45 components in this case [168].

we propose that stable distributions could, in fact, be more appropriate in general, than their normal counterparts.

We also observe that, for the first principal component in particular, the true distribution may be truncated. With very little data it is difficult to confirm this. However, it might be pertinent to consider truncating the samples for UP in order to ensure that performance variation predictions are not overly pessimistic.

Given these findings, it is now possible to select a Monte Carlo sample of 'pseudo-curves' using the eight independent principal components, distributed according to a stable distribution with parameters as fitted to the sample PCs. An example of such a sample is plotted in figure 6.36. We observe two undesirable properties of the curves that have been introduced by using the PCA approach:

1. the peak at $x \approx 0.1$ that is visible in only one of the measured curves becomes a feature of note, in that it influences sample curves, and
2. there is at least one curve that appears to be somewhat unusual in shape compared with the others.

We next turn to the approach used without PCA, i.e. the interrogation of the θ parameter.

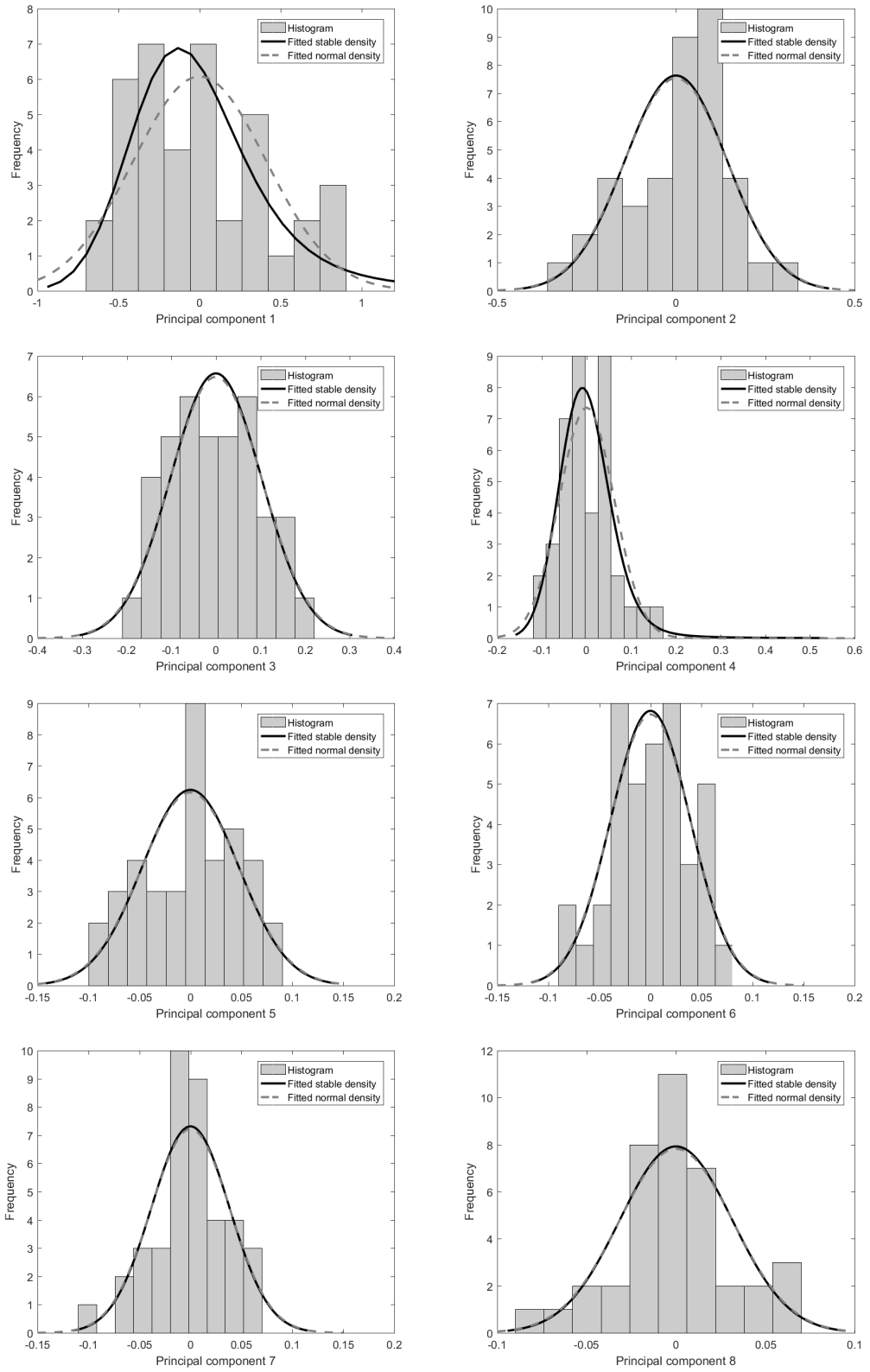


Figure 6.29: Histogram of all principal components with fitted distributions (PCA on 27 data points measured on convex side top flank of turbine blade firtree root)

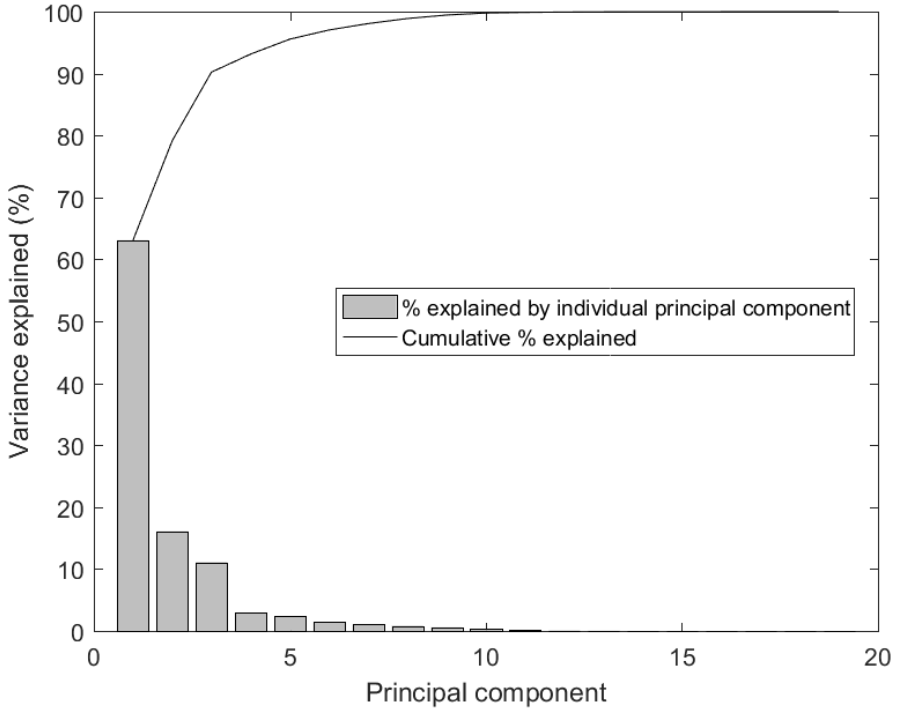


Figure 6.30: Percentage variance explained by principal components for convex top flank curves

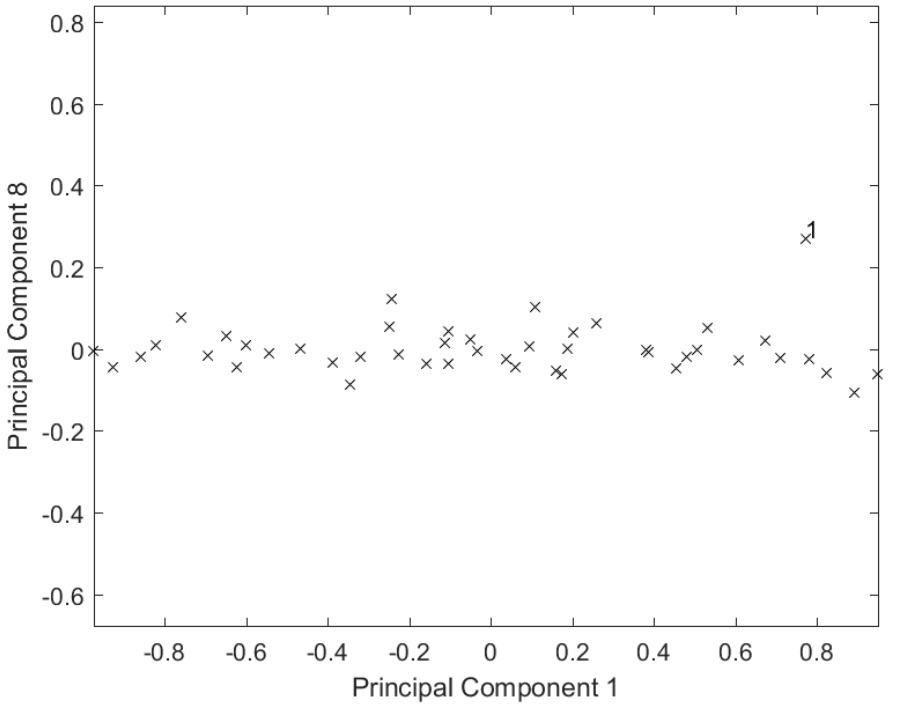


Figure 6.31: First principal component vs. eighth principal component for convex top flank curves, highlighting a potential outlier

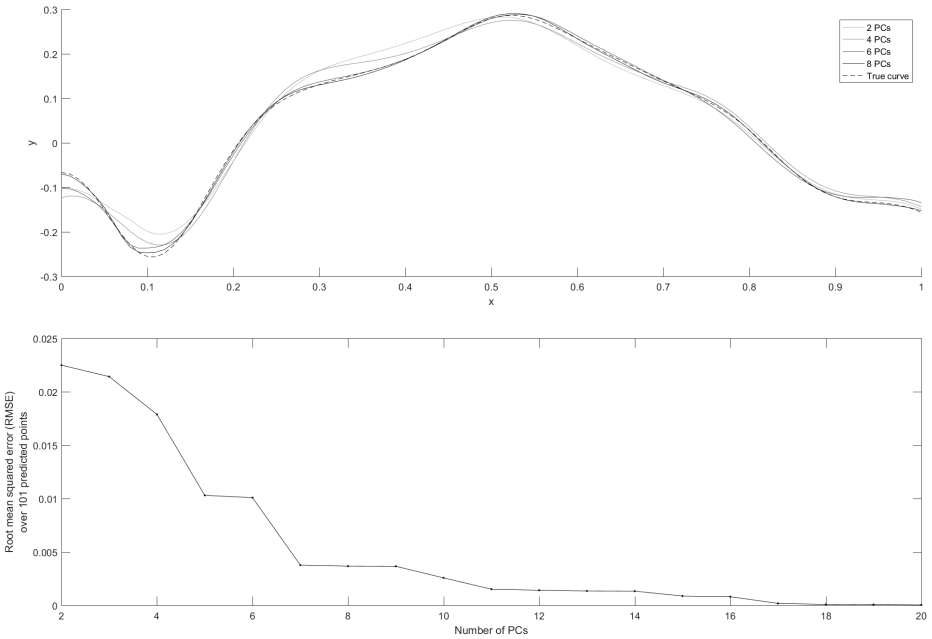


Figure 6.32: Convergence of root mean squared error (RMSE) for curve fits using an increasing number of principal components (PCs)

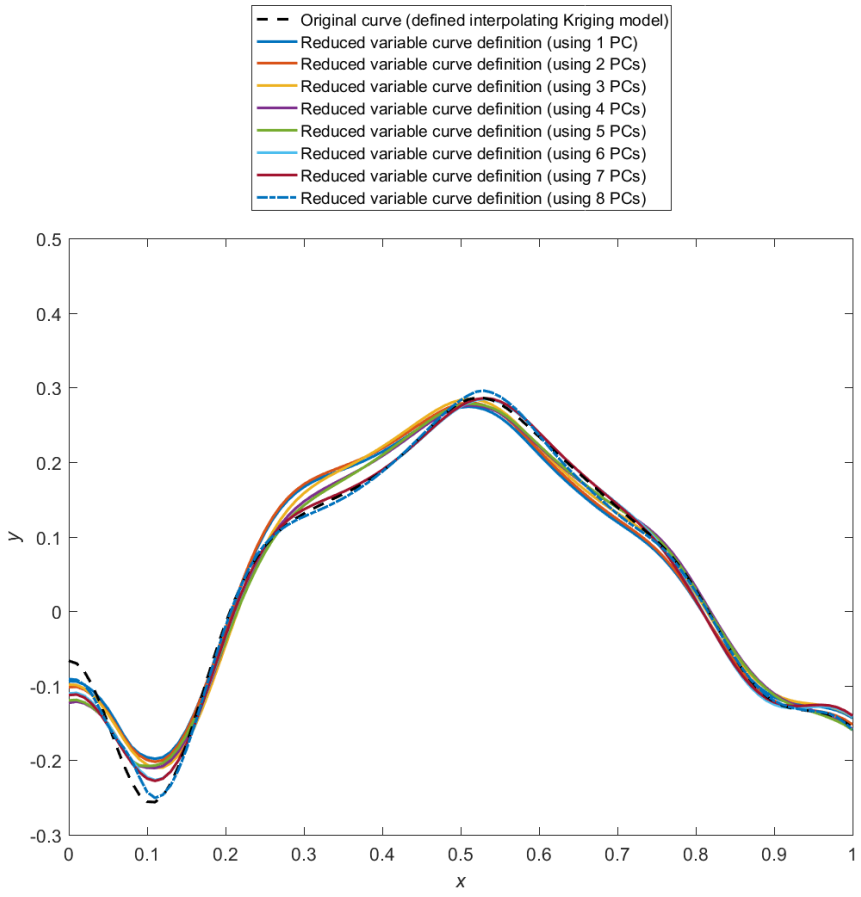


Figure 6.33: Measured firtree flank (convex top) example curve refitted using increasing number of principal components

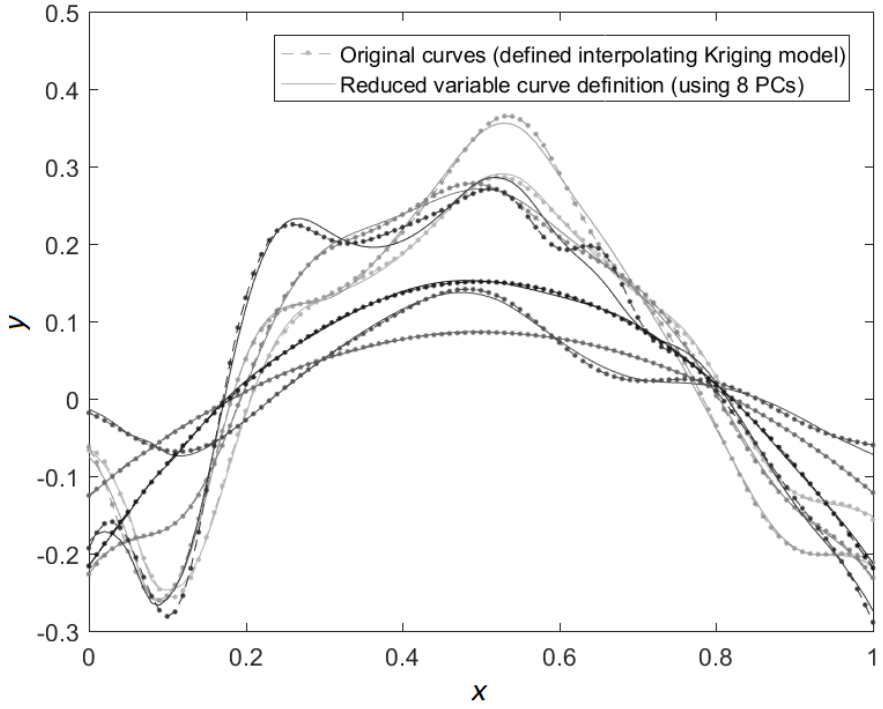


Figure 6.34: Measured firtree flank (convex top) curves refitted using 8 principal components (selection of 7 blades)

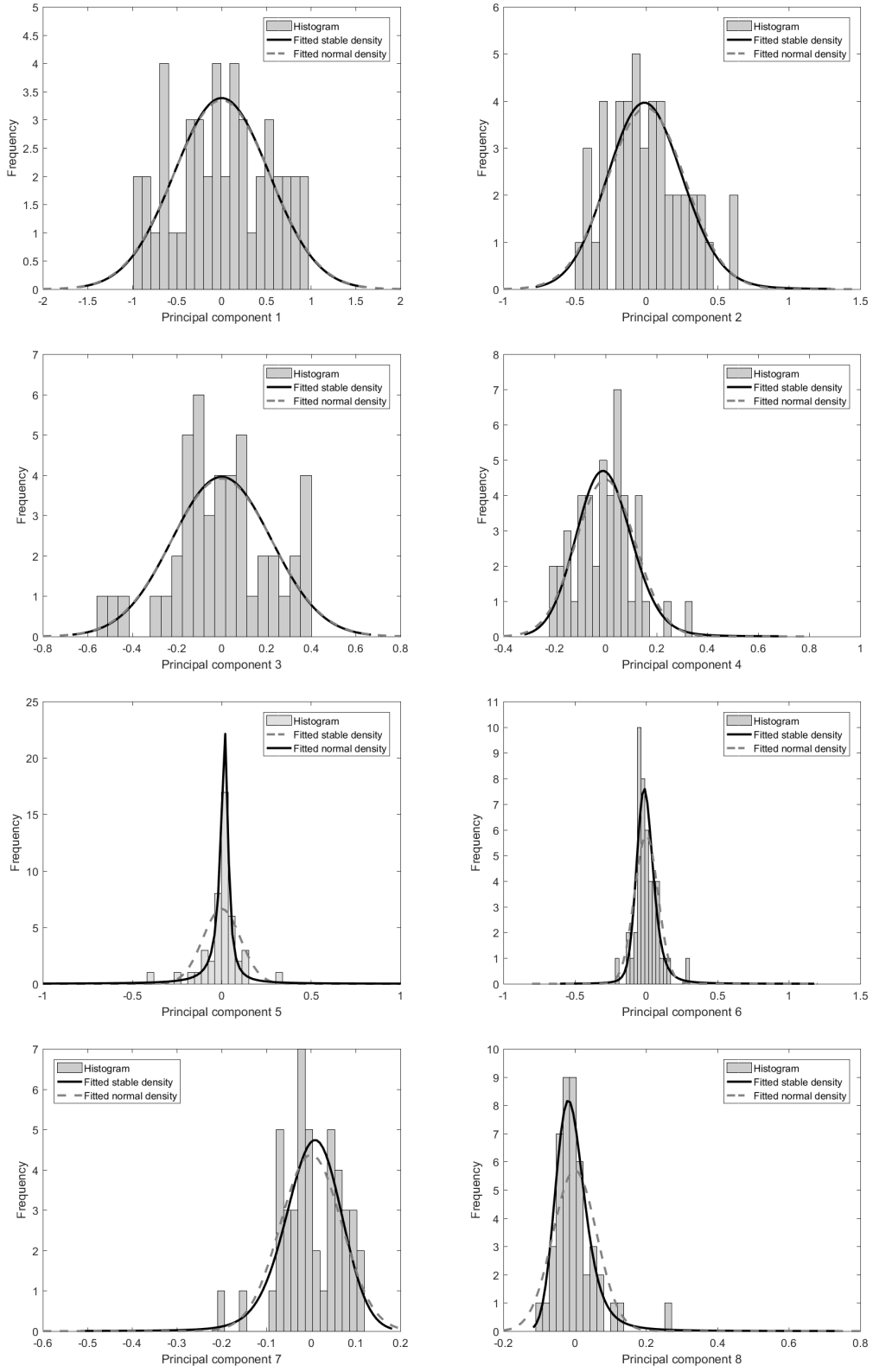


Figure 6.35: Histogram of first eight principal components with fitted distributions (PCA on 101 interpolating Kriging model points on fitted convex side top flank of turbine blade firtree root)

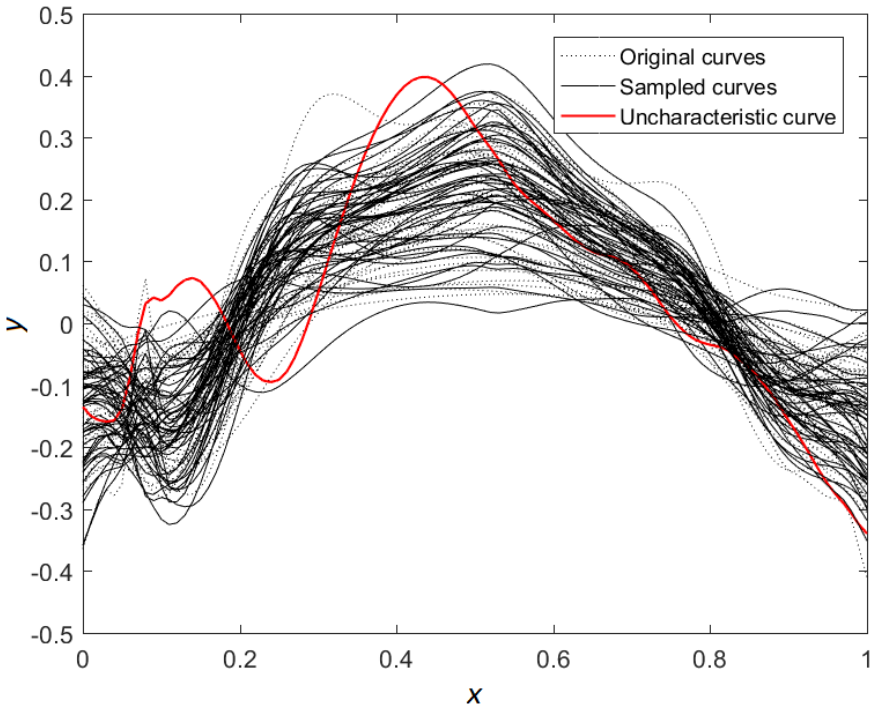


Figure 6.36: Random sample of 50 'pseudo-curves' from eight fitted principal component stable distributions, representing firtree flank convex top curves

6.3.2.1 Enhancing the parametrization of uncertain curves using θ

When a curve is fitted through a small number of sparsely distributed points using a Kriging model, the resulting curve is defined by the point locations and the additional Kriging hyperparameters determined by maximizing the ln-likelihood. For an interpolating model this is just one parameter θ. For a set of sparsely spaced points, changing the θ value modifies the curve as shown in figure 6.37, where an interpolating Kriging model is constructed through a set of sparse points and a set of Kriging model curves with varying theta, and therefore likelihood, are also plotted.

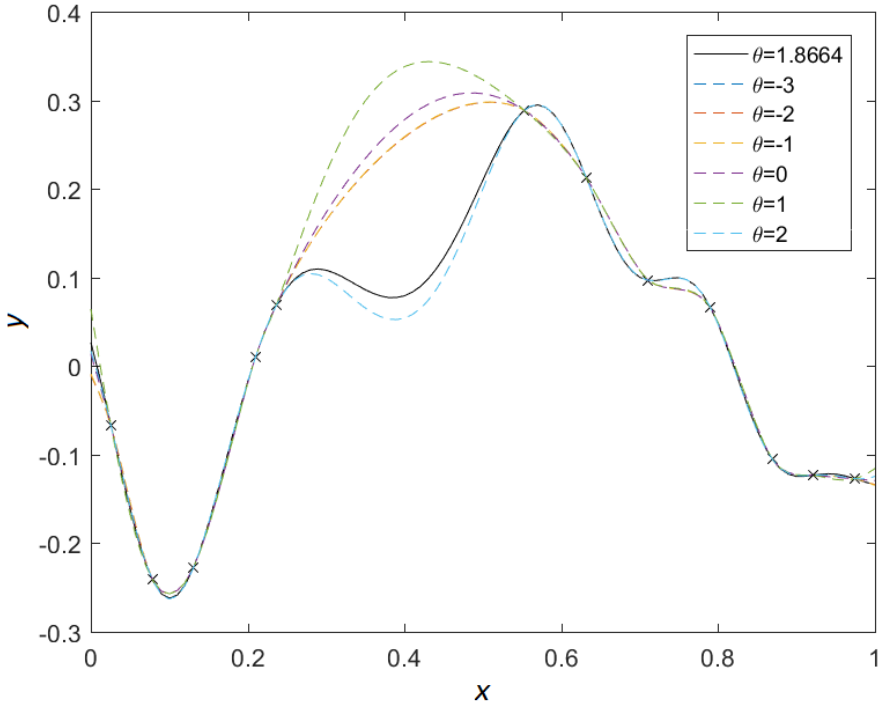
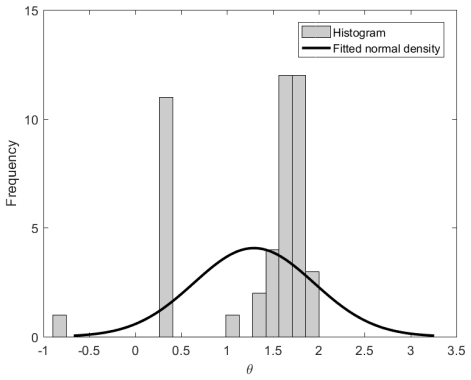


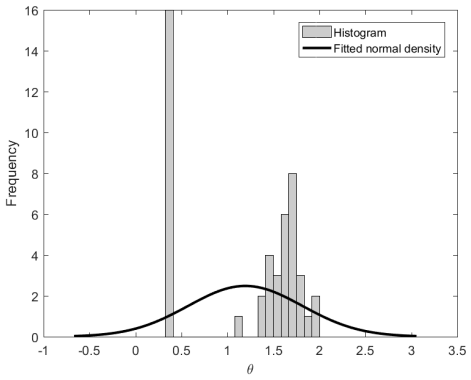
Figure 6.37: Interpolating Kriging model through 12 sparse points, showing maximum likelihood curve with $\theta = 1.8664$, and alternative curves with varying values of θ

The real data sets about, after regression, are, however, almost fully defined by the 101 predicted points, i.e. as the number of points defining the curve increases, the range of feasible θ values decreases exponentially with the number of known points. In other words, θ becomes a property of the curve that depends entirely on the the point locations. As such, it should not be included as an additional variable in the PCA. It does, however, provide us with a significant insight into the data. θ describes the 'waviness' of the curve.

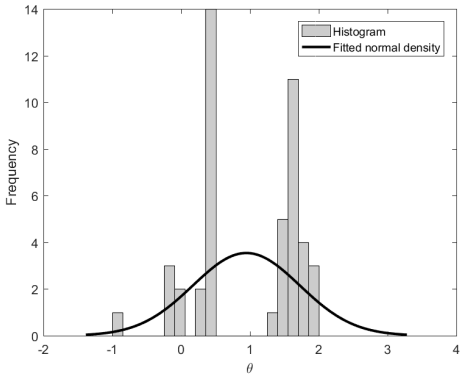
If, prior to PCA, the set of curves defined by 101 points are modelled as a set of interpolating Kriging models then a θ value is then known for each fitted curve. A histogram of the θ values for each of the 6 sets of 46 curves are plotted, figure 6.38.



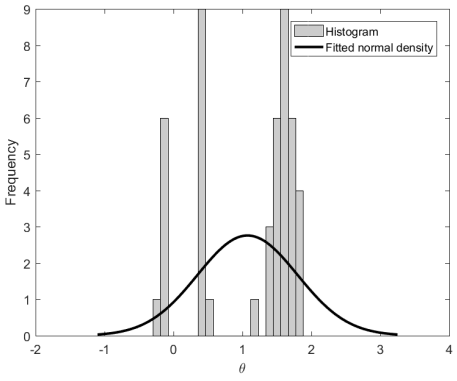
(a) Convex side, top flank



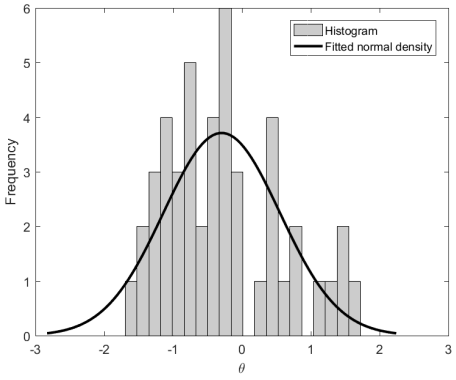
(b) Concave side, top flank



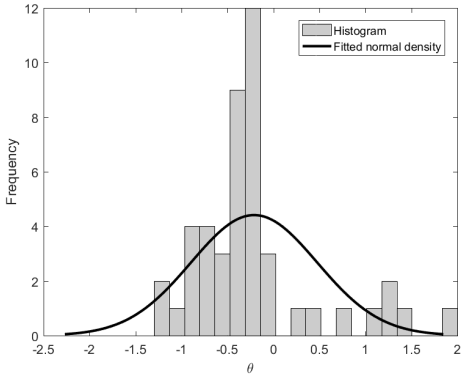
(c) Convex side, middle flank



(d) Concave side, middle flank



(e) Convex side, bottom flank



(f) Concave side, bottom flank

Figure 6.38: Histograms of θ for each flank with fitted normal distributions

It is clear from these plots that there are groupings of curves in each flank set. We can also observe some potential outliers. In particular, the convex side top flank, to which our previous demonstration has related contains an outlying curve (which, not surprisingly, turns out to be the curve with the extra peak at $x \approx 0.1$. It also has two distinct groups. By repeating the PCA process separately for the curves in the two groups we can determine a reduced set of variables for each group.

34 of the measured curves belong to the largest set. This set has a RMSE of 0.01 after 8

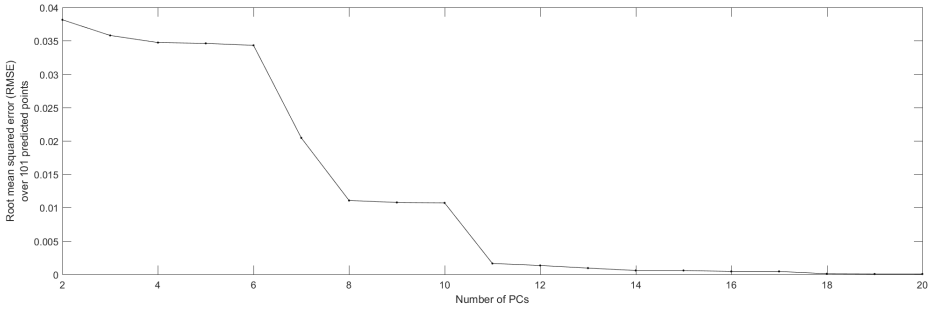


Figure 6.39: Convergence of root mean squared error (RMSE) for grouped curve fits using an increasing number of principal components (PCs), first group

principal components, figure 6.39, and at this point, 99.1% of the variance is explained, figure 6.40. The distributions of these PCs, given in figure 6.41, can be fitted using normal, or stable, distributions; the hypothesis that the samples have been drawn from a normal distribution with sample mean and variance cannot be rejected at the 5% level.

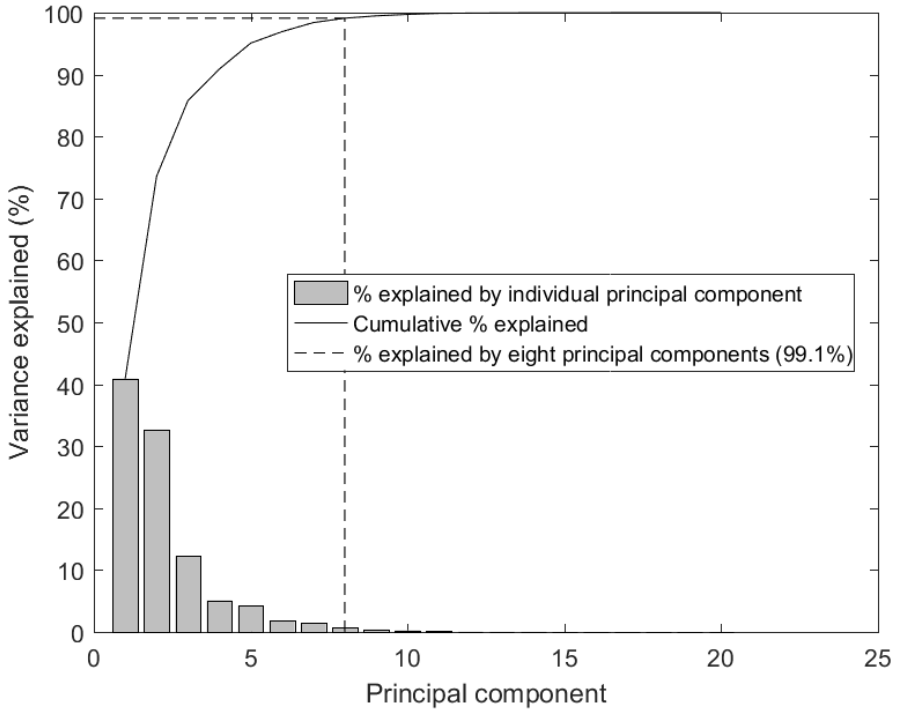


Figure 6.40: Percentage variance explained by principal components for convex top flank curves, first group

The remaining set contains 11 flanks. The variance in this set can be fully explained by only two PCs, figure 6.43, and the RMSE is insignificant, even using only one PC, figure 6.42. The distributions of these PCs, figure 6.44, however, appear to fit more closely to a uniform distribution than a normal one. A χ^2 goodness of fit test for normality rejects the null hypothesis at the 5% level. It is difficult to draw any conclusions from only 11 data sets. However, in this case, a uniform distribution would be preferable for drawing

samples. If a transformation to normal is used and goodness of fit test used on this small sample, the hypothesis as such cannot be rejected at the 5% level in either case.

A MC sample of 50 curves is drawn from each set, using the fitted distributions. The resulting samples are plotted, figures 6.45 and 6.46.

The samples are clearly very much less noisy than the MC set in figure 6.36. The outlying curve is no-longer affecting the sampling of characteristic curves, and multiple families of curves can now be accounted for. The two families determined through this approach are visible by observation in the full sample shown in figure 6.18.

Groups such as these would require additional analyses, i.e. having two groups means that, for surrogate based UP, response surface models of the uncertain space will need to be built for the effects of both curve, or surface, families. In this case, a model over 8 variables for the first family and a model over only 2 variables for the second. When a MC sample is then made of the surrogate for UP, the model to be used for each sample will be selected based on the probability of each family occurring: $P_1 = \frac{34}{45}$ and $P_1 = \frac{11}{45}$.

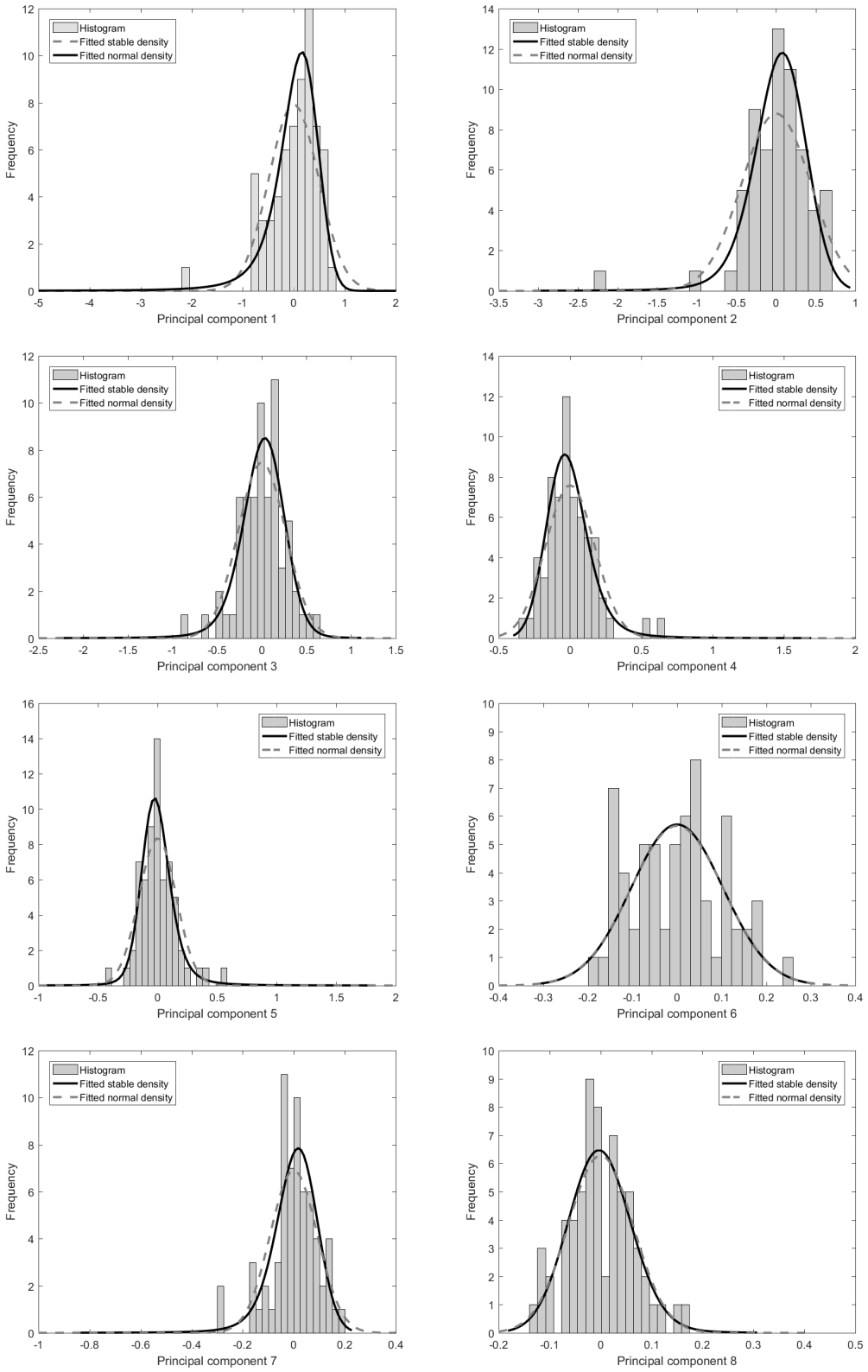


Figure 6.41: Histogram of first eight principal components with fitted distributions (PCA on 101 interpolating Kriging points on fitted convex side top flank of turbine blade firtree root), first group

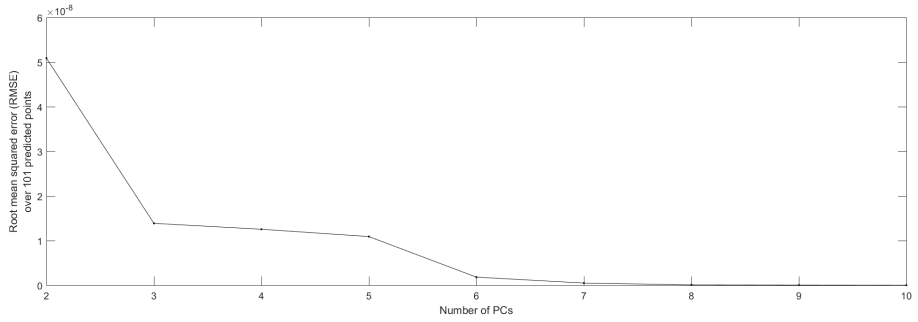


Figure 6.42: Convergence of root mean squared error (RMSE) for grouped curve fits using an increasing number of principal components (PCs), second group

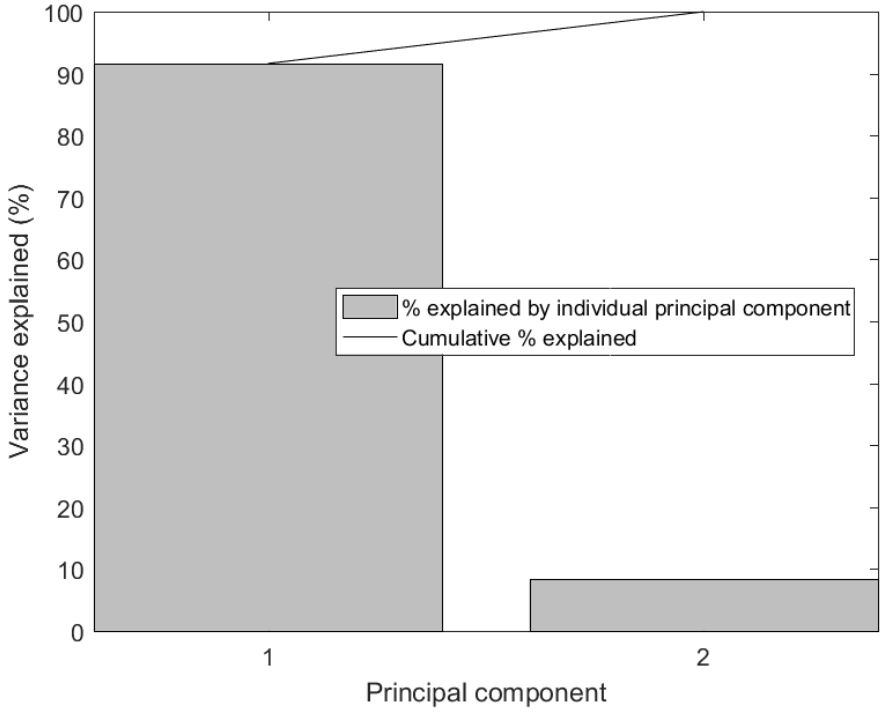


Figure 6.43: Percentage variance explained by principal components for convex top flank curves, second group

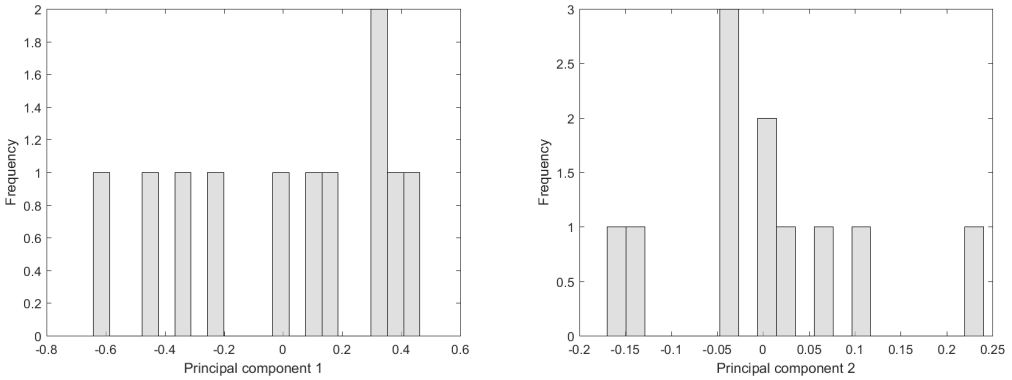


Figure 6.44: Histogram of first two principal components (PCA on 101 interpolating Krig points on fitted convex side top flank of turbine blade firtree root), second group

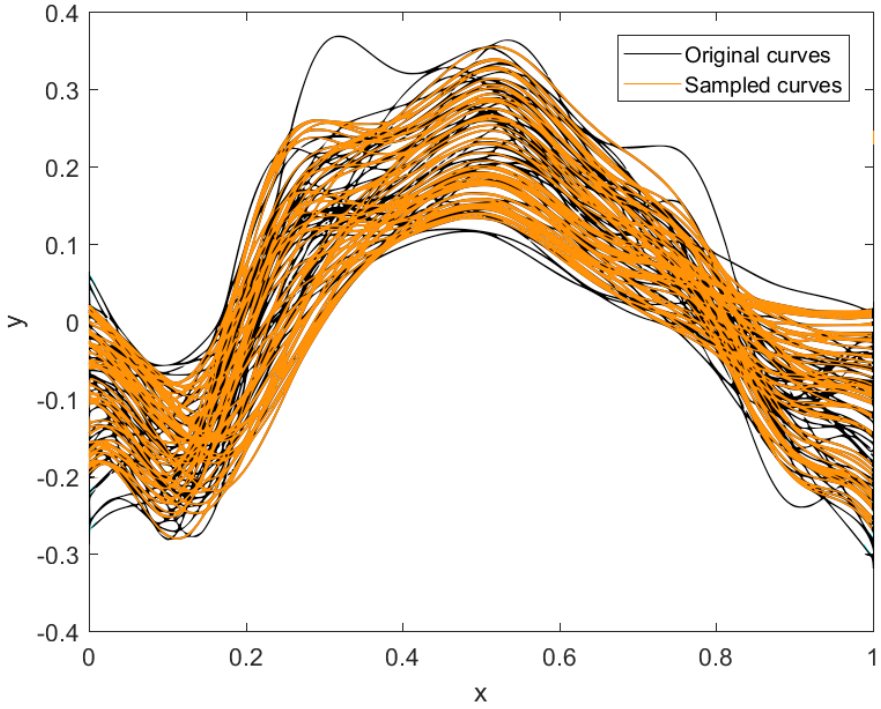


Figure 6.45: Random sample of 50 'pseudo-curves' from eight fitted principal component stable distributions, representing firtree flank convex top curves, first group

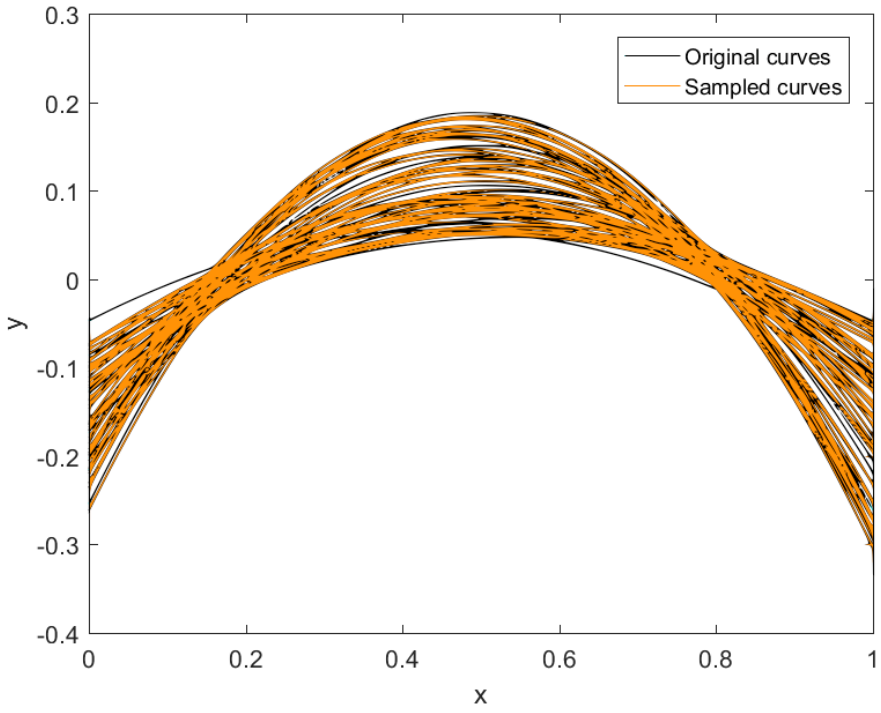


Figure 6.46: Random sample of 50 'pseudo-curves' from two fitted principal component uniform distributions, representing firtree flank convex top curves, second group

6.4 Summary of uncertainty characterization of measured curves

A series of procedures to characterize measured curves, for the subsequent study of the effects of uncertainty, has been developed. Through application to sets of turbine blade contact flank measurement data, in section 6.2 it is demonstrated that, by use of universal regressing Kriging with a parameter τ, we can account for measurement error and gaps in point cloud inspection data. Given a set of these noise-free and gap-free curves, two alternative approaches have then been considered for reducing the number of variables that represent the curves, section 6.3: using either correlated evenly spaced point locations, or a reduced set of independent principal components. In the first case, the Kriging parameter θ is used to identify outlying curves and families of curves. In section 6.3.2.1, this method is then combined with the PCA based approach to provide a more effective means of reducing the number of variables whilst ensuring that statistical models of their variation are as accurate as possible. It has been shown that assessment of RMSE convergence can provide a means of selecting the number of PCs which is more meaningful that the somewhat arbitrary choice of % variance explained at which to make the 'cut-off'. The recommended characterization workflow (a combination of figures 6.19 and 6.20) is given in figure 6.47. This workflow includes the conversion of the reduced parameter set sample to a set of fully-defined 'pseudo-curves' defining the uncertain geometry to be analysed. It is the method by which these curves are included in an existing analysis that is the subject of the following section (6.5).

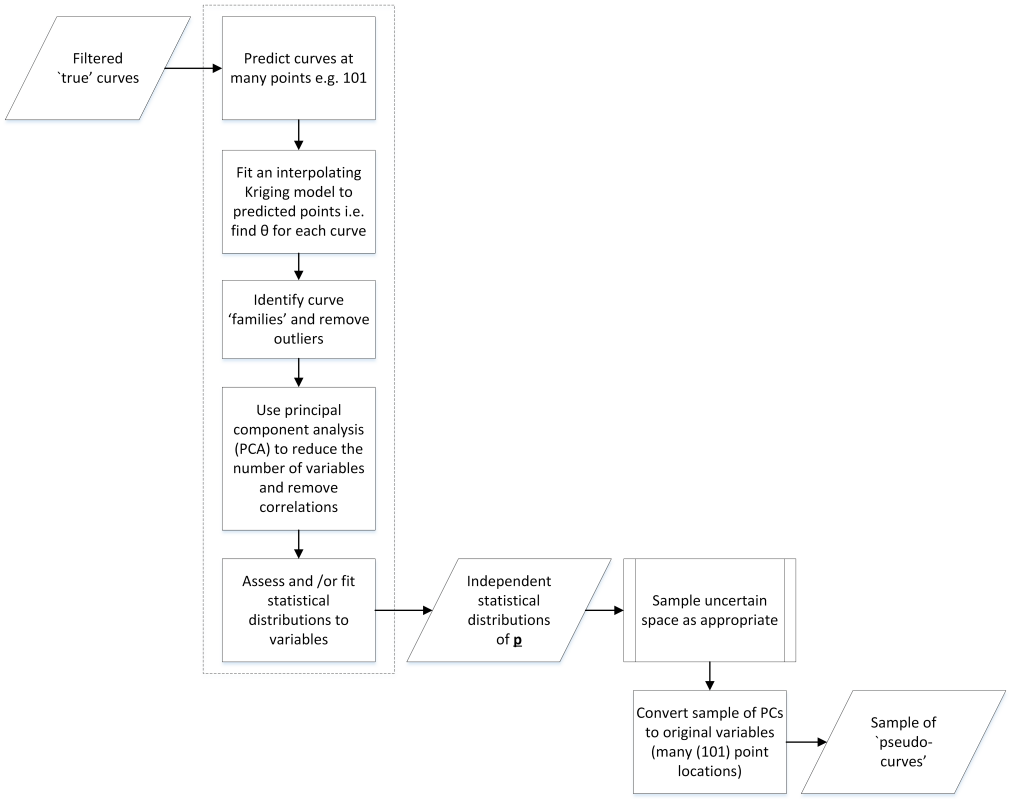


Figure 6.47: Proposed workflow for a 'θ-enhanced' parameterization of uncertain curves using principal component analysis (PCA) to determine uncorrelated point variables

6.5 Implementation of geometric changes

Given that we now have a means of generating 'pseudo-curves' from CMM measurements of curves, we next need an approach by which to alter existing analyses to reflect sampled uncertain geometry. In this case, each sample is represented as an interpolating Kriging model (through 101 points) of the distance of the sampled curve from the nominal geometry. The nominal geometry is a straight line (along the apex of a partial section of a cylindrical surface as shown in figure 6.2). It could be possible to modify the CAD model of the geometry and then re-mesh it for analysis, but this route could be impractical if meshing and applying boundary conditions is an intricate and manual process. Re-meshing could also introduce unwanted noise and expense, as discussed in chapter 3. It is proposed that RBF morphing could be used to enable existing meshes to follow uncertain geometries without excessive distortion, removing the need for re-meshing and providing a more robust solution to the integration of geometric uncertainty in existing analysis workflows.

Initially, a simple rectangular two-dimensional mesh is considered. This mesh represents the original mesh with no geometric uncertainty. From the uncertainty characterization process we have a sampled 'pseudo-curve' defined by 101 point locations. A Kriging

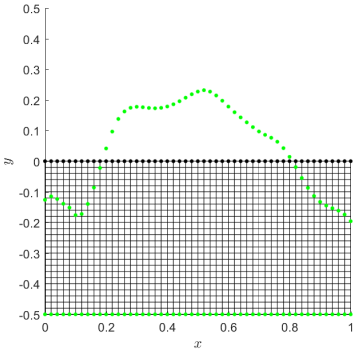
model can be fitted to these points cheaply as there are so many points and the the range of potential θ values is known. This is a model of the orthogonal distance from the existing straight-line surface. The value of this distance can be predicted at each of the existing boundary node positions using the Kriging model. By also specifying the displacement of the base nodes to be zero, it is then possible to morph the mesh. The RBF morph is performed by fitting an interpolating RBF model to the combined set of nodal displacements, $\mathbf{d}$, with the $x - y$ coordinates in real space equivalent to $\mathbf{x}$ in a two-dimensional response surface problem. The nodal displacements for the interior nodes are then predicted using the fitted RBF model. The results of this process are illustrated in figure 6.48 for RBF models that use three different basis functions. A Gaussian RBF based morph results in elements that appear more highly distorted than the other models with displacements extending beyond the surface locations, figure 6.48. This finding concurs with that of de Boer et al. [47], who determined that the Gaussian RBF resulted in large mesh distortions for their examples. It is maybe not surprising that the model is essentially fitting a surface that rapidly changes from 'flat', at the zero displacement boundary points to extremely 'wavy' at the remaining points. Gaussian models can perform badly when waviness is significantly variable throughout a dimension.

The process of implementing this morph on a two-dimensional mesh is described by the flowchart, figure 6.49. The grey boxes show how RBF morphing is typically used: making modifications to existing meshes using new geometric models for design optimization.

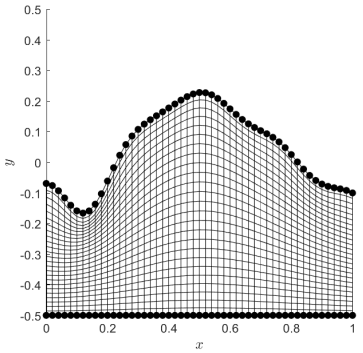
The measurements that are available, however, pertain to the ridge on a solid mesh. In order to morph a solid mesh we propose a two-stage RBF morph. If the measurement data was a point-cloud representing a surface, and the 'pseudo-curve' was, in fact, a 'pseudo-surface', then the morph is simply an extension of the process demonstrated thus far, but with $\mathbf{x}$ consisting of the three Cartesian coordinates of each nodal point, and $\mathbf{d}$ the displacement of the surface nodes. In this case, however, a prediction of the displacements of the surface nodes that aren't measured (beyond the location of the curve alone) is needed. The approach used here is to first use the RBF morphing technique to morph only the surface nodes. In this case $\mathbf{x}$ are the two-dimensional $x - y$ coordinates of nominal curve combined with the coordinates of the edges with zero deformation. The fitted RBF function is then used to predict the locations of the surface nodes in between. To demonstrate this, a constant displacement is given to a straight-line curve on a flat surface, figure 6.50. The same three RBF basis functions as used for figure 6.48 and discussed in section 2.11 are applied.

Applied to the sampled 'pseudo-curve', this results in the surfaces shown in figure 6.51. If the flat surface is replaced with the original 'barrelled' flank surface then the morphed surface is transformed to that shown in figure 6.52.

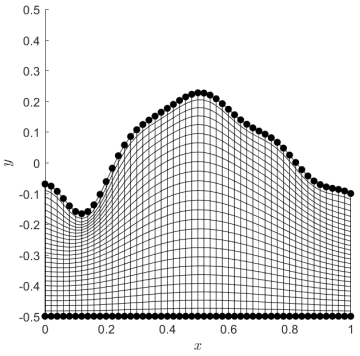
These surface deformations, combined with any constrained (zero deformation) nodes, are the response for our second RBF model, and $\mathbf{x}$ is now composed of the three-dimensions



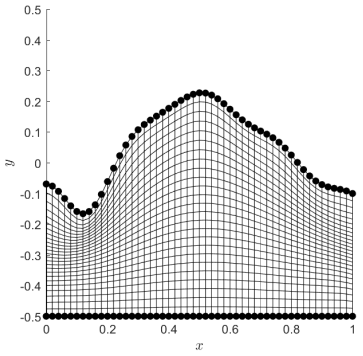
(a) Original mesh and Kriging predictor point locations, including zero deformation points



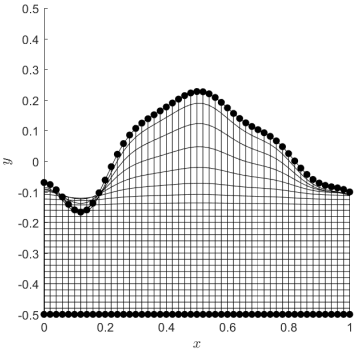
(b) Cubic RBF



(c) RBF using compact support basis with C^2 continuity



(d) Thin plate spline RBF



(e) Gaussian RBF

Figure 6.48: RBF morph of randomly sampled curve from a convex side top flank, first group set, using Kriging predictor at nodal coordinates of boundary in x

$x-y-z$ of each of the nominal nodal positions of the solid mesh. If the original mesh is the regular mesh in figure 6.53, then the resulting deformed mesh using the cubic spline basis is shown in figure 6.54.

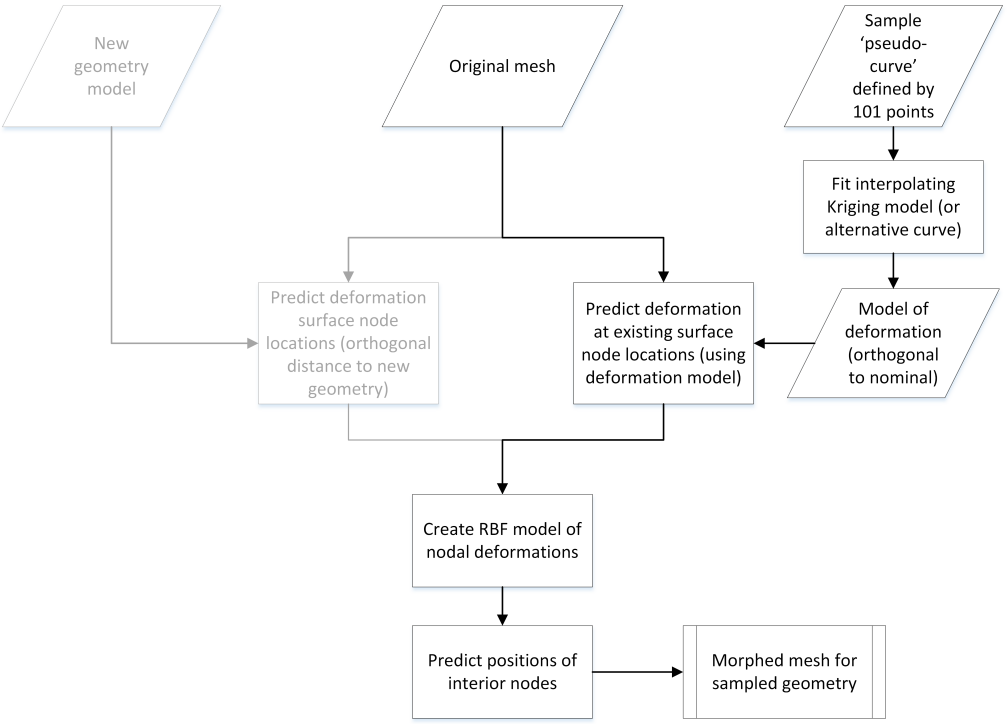
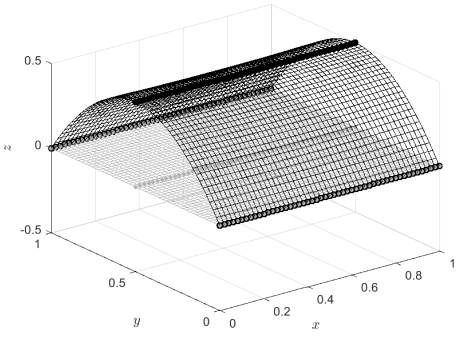
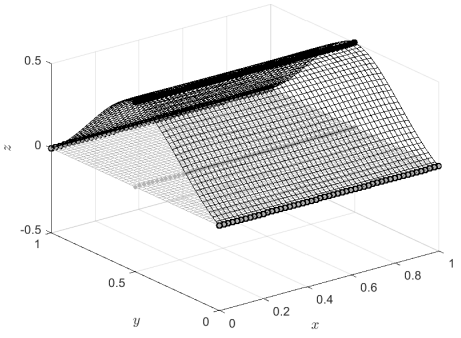


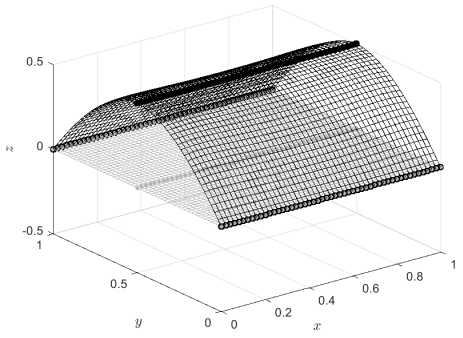
Figure 6.49: Workflow for the implementation of uncertain changes in an existing 2D analysis mesh using RBF morphing



(a) Cubic RBF

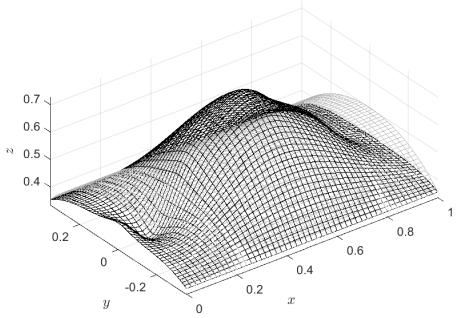


(b) RBF using compact support basis with C^2 continuity

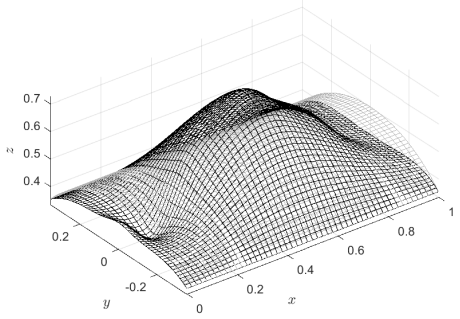


(c) Thin plate spline RBF

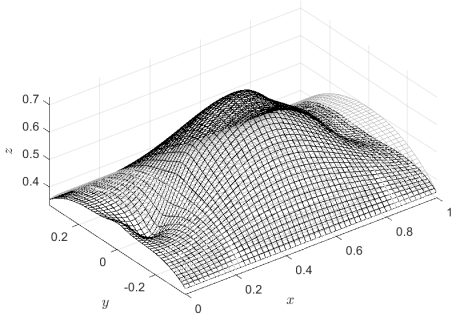
Figure 6.50: RBF morph of flat plate with straight-line deformation at the centre



(a) Cubic RBF

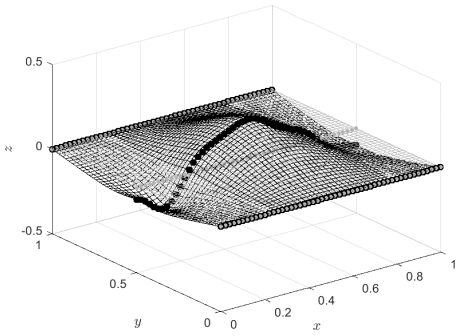


(b) RBF using compact support basis with C^2 continuity

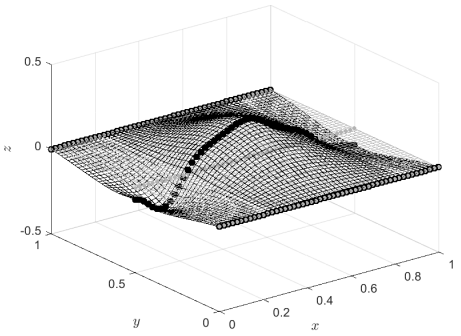


(c) Thin plate spline RBF

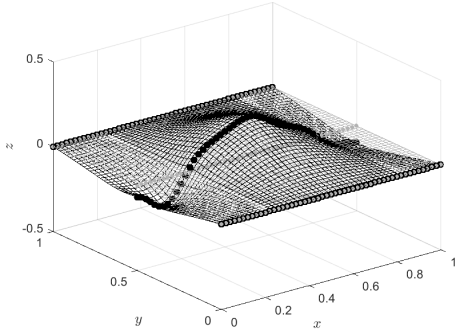
Figure 6.51: RBF morph of flat plate with 'pseudo-curve' deformation at the centre



(a) Cubic RBF



(b) RBF using compact support basis with C^2 continuity



(c) Thin plate spline RBF

Figure 6.52: RBF morph of 'barrelled' flank surface with 'pseudo-curve' deformation along the apex

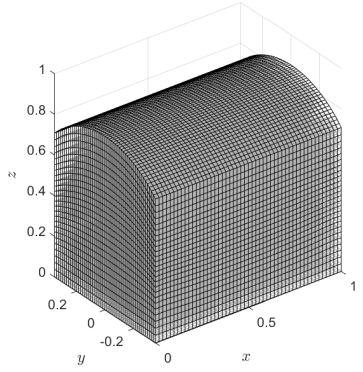
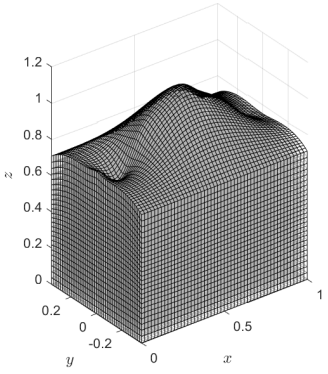
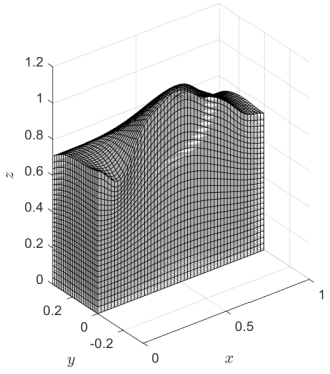


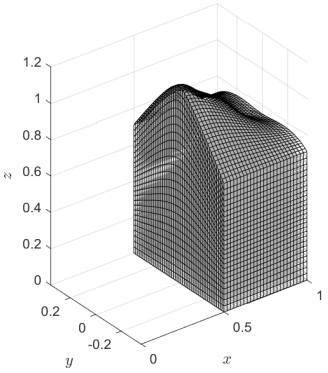
Figure 6.53: Solid mesh for demonstration of the application of measured, or sampled, uncertainty to the 'barrelled' surface of a solid component



(a) Morph of full solid mesh



(b) Section at $y = 0$



(c) Section at $x = 0.5$

Figure 6.54: RBF morph of 'barrelled' flank surface with 'pseudo-curve' deformation along the apex in a sold mesh, using a thin plate spline basis

6.6 Summary of the implementation of uncertainty in geometry as sampled 'pseudo-curves', using mesh morphing

The full implementation process, as demonstrated in section 6.5, for a solid mesh is shown in figure 6.55.

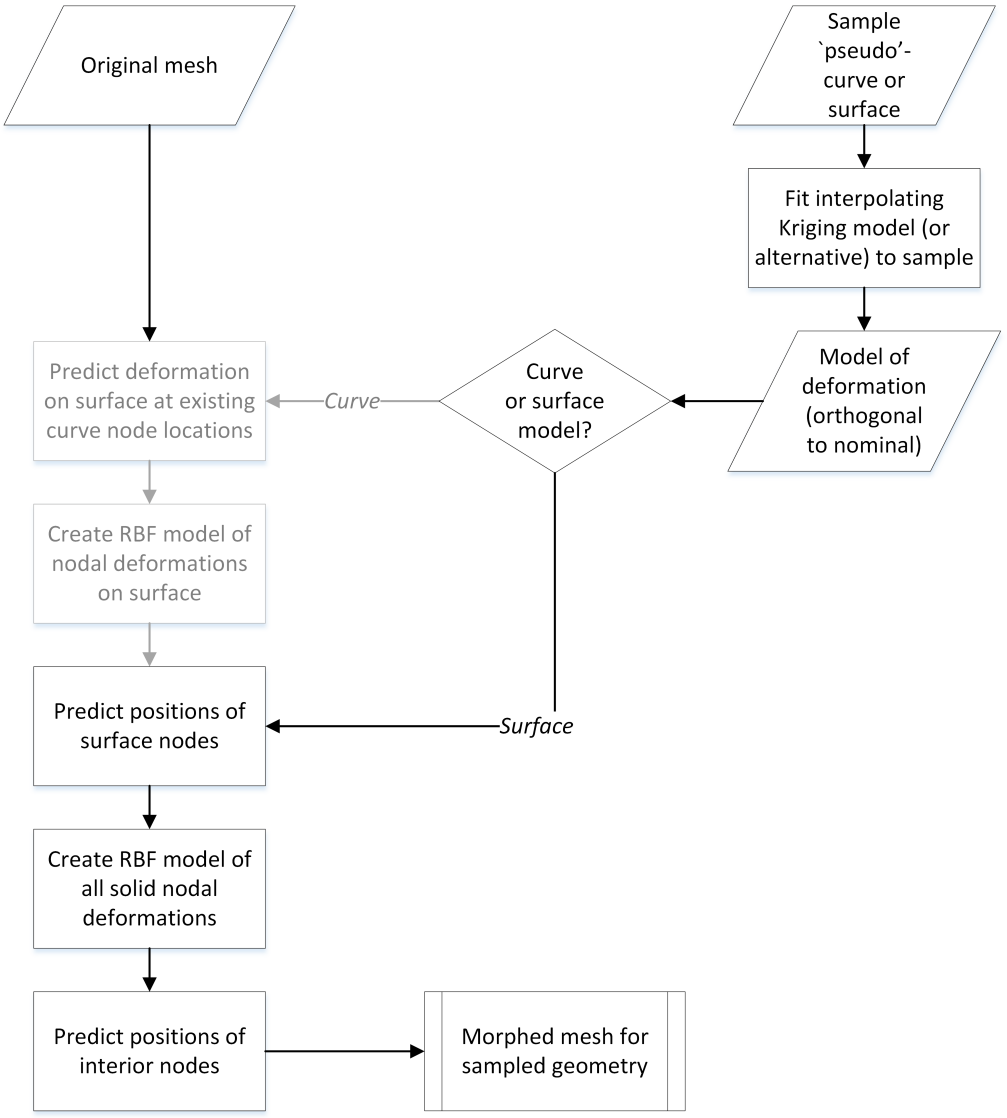


Figure 6.55: Workflow for the implementation of uncertain changes in an existing analysis solid mesh using RBF morphing

This approach enables a precise representation of sampled geometric uncertainty in an existing mesh. The modifications are made to nodal positions only and associated boundary conditions need not be modified. Given a point cloud of surface measurements, or a 'pseudo-surface', instead of a 'pseudo-curve' the workflow can be simplified through removal of the added processes shown in grey.

6.7 Summary and contributions

Throughout this chapter a method has been developed by which real measurement data, in this case a set of point clouds representing a nominally straight-line turbine blade firtree flank apex, can be converted into a significantly reduced-size set of variables which can be sampled in order to create 'pseudo-curves' or 'pseudo-surfaces'. In turn, we have shown how these sampled curves or surfaces can be analysed through modification of existing meshes. The full generic process is provided in detail in figure 6.56. This flowchart is an expanded version of figure 6.4 for geometry where point cloud inspection data are available. The fitting part of the process is simply described as a "regression fit" in this workflow because the 'best' curve fitting approach may vary depending on the specific data. The universal Kriging approach used for the supplied data set was, however, shown to perform well in this case. The use of Picheny's regression parameter τ is also beneficial [207].

Although the workflow has been demonstrated on 'real' data sets, the data has all been normalized for confidentiality and, for this reason, the approach has been applied to a fabricated mesh section, rather than a mesh of the true design.

A number of well-developed methods have been combined to successfully address a problem to which they have not previously been applied. Here we highlight a number of contributions, where the approach used is novel in this particular application.

- The use of fixed-variance Kriging (Kriging reformulated using the work of Picheny et al. [207]) to fit point cloud measurement data, i.e. using the 'nugget term' to filter noise of a known magnitude (measurement error).
- The use of Universal Kriging to approach the 'gappy data' problem with a visible trend (dealing with gaps, occlusions, and sparse points).
- The development of a combined Kriging and PCA based approach to reformulating the parameterization of curves and surfaces where both
  - (i) the number of parameters needs to be minimized,
  - (ii) the statistical distributions defining parameters need to be meaningful (to be accurately represented using a probability density function), and
  - (iii) variables must either be independent, or correlations identified and maintained.
- The use of RBF morphing in an uncertain geometry context.

The methodology for re-parameterizing curves and surfaces defined by large sets of point locations is particularly novel in the use of the θ parameter to identify curve 'families'

and outliers, thereby improving the ability of the PCA approach to fit distributions to the defining variables. In addition, for both the artificial functions considered, and the 'real' data set, it appears that the recently developed stable probability distribution could offer a more appropriate distribution in some cases when sampling uncertain variables. It should, at the least, be included for consideration. Typically, engineers have used the assumption of normality of variables for uncertainty modelling, which could be overly pessimistic. We also note the use of RMSE convergence in the selection of PCs having some novelty in this application. Our findings, that convergence occurs beyond what might be considered a suitable percentage of variance to represent, might go some way to explaining the findings of Garzon and Darmofal [88] as discussed in chapter 3. Further research on additional data sets would be required to draw conclusions from these findings, but including RMSE convergence assessment in the PCA procedure when applied to geometry certainly appears to be beneficial.

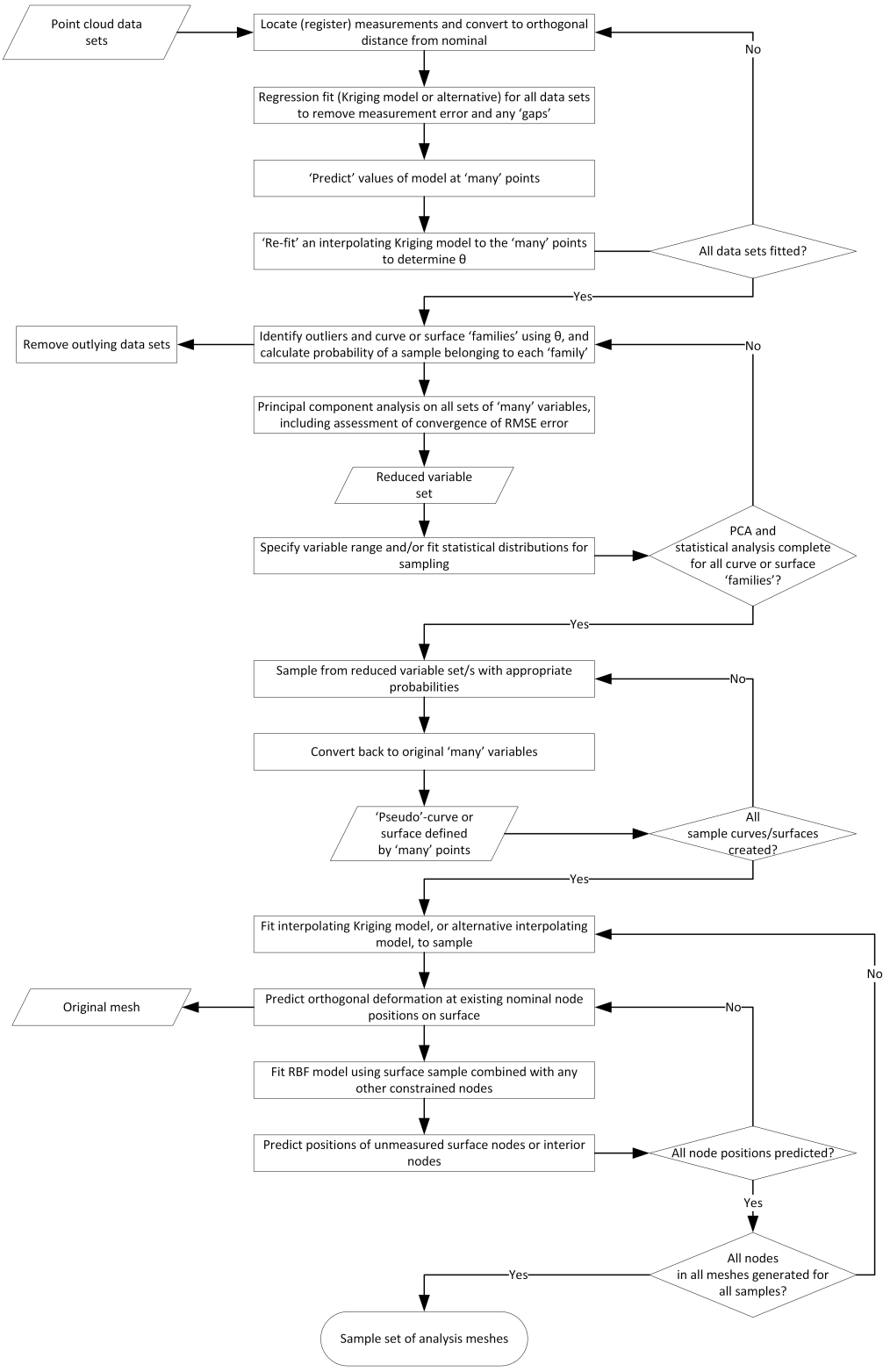


Figure 6.56: Workflow for the characterization and implementation of measured uncertainty in an existing analysis mesh using Kriging, PCA, and RBF morphing

Chapter 7

A summary of the proposed data-driven geometric uncertainty integration methods for design, and contributions made

7.1 Process and methods overview

This thesis has used the three case studies (chapters 4, 5, and 6), where data have been available, to demonstrate the accurate representation of geometric uncertainty for application to existing analyses. Doing so through an automated process is difficult due to the differing nature of various sets of measurement data. Here, we discuss how the approaches developed through the case studies broadly follow the available methods detailed in chapter 3. The flowchart (figure 7.1) is based upon the uncertainty characterization workflow from figure 3.8, which shows how point cloud measurement data sets can be converted to usable parameter sets for sampling. In addition, the flowchart includes the implementation step of the process, which utilizes a sample of this parameter set. It is assumed that both the analysis, and the measurement process, are expensive such that very large samples of either are not possible. Therefore, requisite steps of dimension reduction (to enable sampling of expensive analyses) and statistical distribution fitting (to enable UP when many fewer measurement data sets exist than the thousands needed for direct MC) are included.

The chart, therefore, provides an outline process for integrating inspection data into existing analyses for the facilitation of uncertainty and robustness studies. The steps

recommended through each of the case studies are highlighted. These are largely automated processes, which are preferable to the alternative manual processes shown in grey boxes.

In chapter 4 (process shown in blue in figure 7.1) we provide an exemplar of reliable fitting using existing design parameters to characterize uncertainty. We apply constrained **global optimization**, using a **particle swarm based algorithm** (section A.4.2), to find the parameters of interest. Statistical models are then fitted using the **normal distribution** (section A.8.1) and tested using the **z-test** (section A.8.3). Implementation of analyses using sampled parameter values then becomes a simple modification of numeric values in existing code, under certain circumstances.

In chapter 5 (process shown in red in figure 7.1), the opposite is true: characterization of uncertainties is relatively simple as no curve fitting is required, but implementation is difficult because existing code is not set up for automated meshing, and the analysis is expensive and complex. An adaptation of the **free-form deformation** method (section A.10.0.1) has been adopted in this case.

The data from chapter 6 are representative of typical free-form geometry. It is not obvious, in this case, how to parameterize the measured geometry to best effect. The recommended approach developed by the author (shown in green in figure 7.1) uses **Kriging models** (section 2.2), **PCA** (section 2.10), and statistical modelling with **normal distributions** (section A.8.1) assessed using the χ^2 **goodness-of-fit test** (section A.8.3), or **stable distributions** (section A.8.2), to do so effectively, prior to continuing to apply meta-modelling methodologies (**RBF**s, section 2.2) as an **RBF morph** (section A.10.0.2).

The developed detailed approaches, along with the overall flowchart (figure 7.1, provide a clear process for automating the integration of point-cloud inspection data into existing analyses in order to inform design decisions in an uncertainty framework.

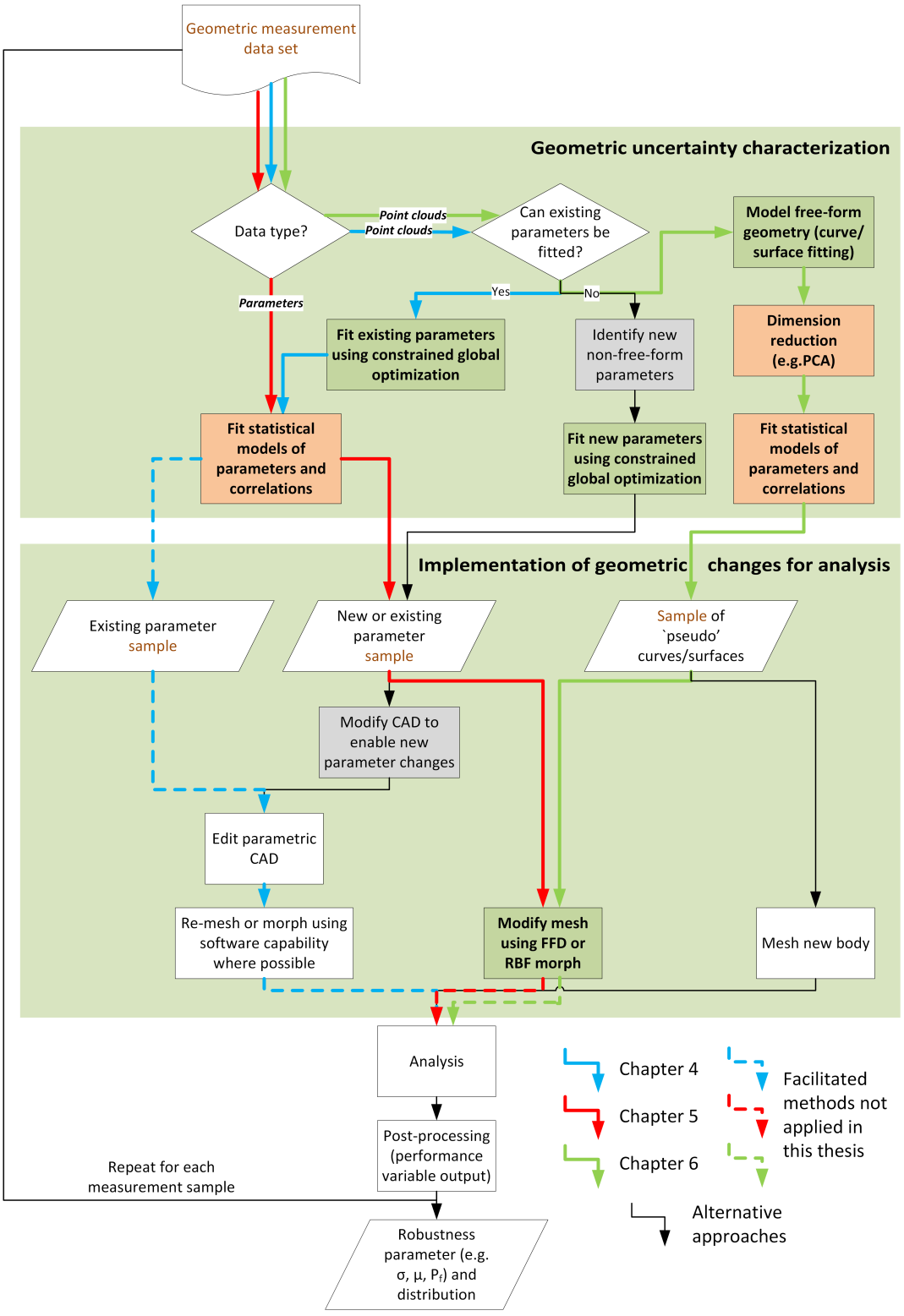


Figure 7.1: Characterization and implementation of geometric changes in existing analysis workflows based on point-cloud measurement data

7.2 Conclusions and contributions

There are relatively few attempts to integrate measurement data with the process of robust design available in the literature. More commonly, uncertain geometry is applied to relatively fast-running problems without complex meshes. In order to enable integration of these design tools with existing analyses, facilitating technologies are required. In this thesis, through application to three different data sets, such automating technologies have been explored.

1. After identifying that existing design parameters can be fitted to point cloud data in the case of turbine blade firtree flank cross-sectional measurements, it has then been possible, through appropriately constrained global optimization, to determine the statistical distribution of flank angle.

2. Given a set of measurements for existing design parameters for a highly complex analysis workflow where a time-consuming and manual meshing process had been undertaken for the nominal design, an automated approach to modifying meshes through holes, based on free-form deformation in a polar coordinate reference frame, has been presented.

3. Given multiple sets of point cloud measurement data, which deviate from the nominal in a free-form manner, a sophisticated combination of Kriging-based curve fitting and dimension reduction by means of principal component analysis has been applied for uncertainty characterization. An automated process based on RBF morphing has then been adopted to demonstrate the implementation of geometric changes sampled from fitted statistical distributions of the reduced variable set.

We then presented an overall approach to integrating inspection data based on these case studies in section 7.1. Throughout the thesis, the methods applied have been well-established, but their application in this context has previously been limited. When approaching each of the real problems, the 'standard' approaches have 'fallen short' of having the ability to represent the data as required, leading to use of alternative, usually surrogate and DSO based methods, in novel ways, as reviewed below.

The first study (chapter 4) presented a 'simple' problem of fitting known parameters to point cloud data. Commonly, fitting of straight lines, or other basic features, to point clouds involves the identification of the points that exist within the feature by segmentation, followed by a least-squares, or similar, fit. In cases where it is difficult to locate the boundaries of the feature, it is not clear how best to proceed. Given that uncertainty studies require accurate representation of relatively small changes, and that errors at end point locations can have a disproportionate effect on the angle of the fitted line, standard processes are unacceptable. Indeed, even removing points close to the edge

of the straight-line feature for 'safety' is shown to have a significantly detrimental effect on the accuracy of the fitted line angle. Segmenting the data well beyond the curve of interest and applying a constrained global optimization procedure has been shown to be necessary in this case. Such global methods applied to full data sets with little segmentation, are likely to be beneficial in the integration of many measurements of uncertainty.

To integrate uncertainty as changes to existing parameters in an already complex, expensive, and largely manual analysis, in chapter 5 we turned to free-form deformation. As a means of controlling free-form shape changes in an intuitive manner, and for controlling such changes using a set of control point locations this approach has many advantages. However, given that every point within a control lattice is influenced by the movement of any control point, ensuring that the shape matches the measurements can be difficult, if not impossible. The data supplied in this case took the form of diameters of holes through non-flat surfaces. In this case, ensuring circularity of the holes in the meshes, and maintaining surface nodes on surfaces, proved impractical with traditional free-form deformation. A polar-coordinate based approach was developed for maintaining hole circularity whilst Kriging-based models of the surfaces were used to first transform complex surface shapes to flat-plate locations before morphing the nodal positions. A problem-specific alternative, again using Kriging models, although any surface-fitting model could be applied, has been used for the special case of holes in 'stepped' surfaces.

A major point of the novelty in the work appears through the approach developed to deal with 'free-form-type' point-cloud data throughout chapter 6. Initially, we apply universal Kriging to deal with 'gappy' measurement data with a visible trend, and the use of fixed-variance Kriging (reformulated using the work of Picheny et al. [207]) using the nugget term to filter noise of a known magnitude (measurement error). Neither of these methods are typically applied to measurement data, rather for response surface modelling.

Following on from the 'error filtering' step, an innovative Kriging-based approach to reformulating the parameterization of curves and surfaces is presented. For uncertainty studies of the type discussed, there is a requirement for:

i the number of parameters to be minimized,

ii the statistical distributions of defining parameters to be meaningful (to be accurately represented with a probability density function), and

iii the variables to either be independent, or correlations identified and maintained.

By first parameterizing the measured curves using a large number of points through an interpolating Kriging model it is possible to use the hyperparameter θ to categorize the curves and detect outlying data sets. This is particularly original, not least because

Kriging is seldom used to represent geometry. Given this grouping of the data, principal component analysis has then been used to reduce the large defining set of point locations to a smaller set for each group effectively. For the data available, well-fitting statistical distributions for each of the variables resulted. In this case, stable distributions were also highlighted as a potentially better distribution function to use in these cases, in preference to the typically ubiquitous normal distribution function: a density function that is state-of-the-art, and somewhat unique in its use for stochastic analysis in engineering design.

Given the then reduced set of variables defining the uncertainty in a nominally straight line component feature, RBF morphing using a thin-plate-spline based RBF function has been demonstrated on a representative solid mesh. Although the RBF morphing technique is not new, its particular relevance in geometric uncertainty studies has been highlighted. Given sampled models of the uncertainty of the type developed through the presented process, RBF morphing is the 'tool of choice' for implementation. No workable and practicable alternatives have been identified.

7.3 Future work

A framework for integrating measured point-cloud uncertainty with design analysis has been developed. It has been tested on a number of cases, but further work would be required to ensure that the approaches are applicable in many more cases.

In particular, the process developed throughout chapter 6 should be applicable, and could potentially be even more beneficial, if applied to measurements over surfaces or full components. Currently, a large quantity of inspection data are collected at limited point locations or along specified curves due to the cost and time associated with gathering larger data sets, but this cost is reducing with developments in measurement and imaging technology. Therefore, further work to implement and test our approach on such measurement sets would be useful going forward.

There are also other areas where the author recommends further research to add to existing findings and approaches. RBF morphing, for example, is usually used to morph meshes where parametric changes can first be made to designs through a linked geometry model, and applied to DSO. In chapter 6 it is used to make free-form-type changes to a mesh. It could, however, have uses for cases such as that of chapter 5. The difficulty here is that the modified surface node positions (or movements) must be known in order to apply the RBF morph to the interior nodes. Further investigation is required, but it might be possible to enhance or simplify the current procedure using this method.

Another such area is where an original combination of steps has been used to parameterize curves using large sets of Kriging model point locations. This approach incorporates principal component analysis. The most elementary form of PCA - linear PCA - has

been used, and although it had resulted in variables with well-fitting distributions for our case study, there may be circumstances under which some more sophisticated versions of PCA or non-linear PCA could prove better suited. It would be prudent to assess these using further data sets.

Finally, we use Kriging (and its θ hyperparameter) to represent the data for variable reduction, but then use a non-Gaussian radial basis function for interior nodal point morphing in the implementation of the sample of the variables. In this case, an additional fitting process is required because there are additional zero deformation points included for creating the RBF model. However, in the case where the measurement data fully represents a component all surface nodes locations would have been included in an interpolating Kriging model. Investigation as to the appropriateness of this model in the prediction of interior node positions would be an interesting study, perhaps leading to a reduced cost overhead in such cases. Here, we point to the additional complexity in these cases of determining how best to define the points: orthogonal from nominal, or using some alternative spherical coordinate frame, for example. We also note that the θ parameter used for identifying curve families would become two θ values for a surface and that further development and testing would be needed here.

In conclusion, a number of techniques have been applied throughout this thesis such that geometric uncertainty studies can more easily and effectively be informed by 'real' data from inspection using existing analyses. Routine inspection often results in data akin to that of our case studies meaning that this work is useful and applicable to many robustness problems. However, further work could enhance these approaches to increase their applicability to a more extensive range of problems.

Appendix A

Methods: the detail

A.1 Space-filling sampling plans

A.1.1 Full-factorial sampling

Intrinsically, the most simple space-filling approach is a full factorial design [72], where all 'strata' (or 'bins') of an equally spaced grid are sampled at their centre, figure A.1. There are a number of major flaws in the full-factorial approach when applied to model

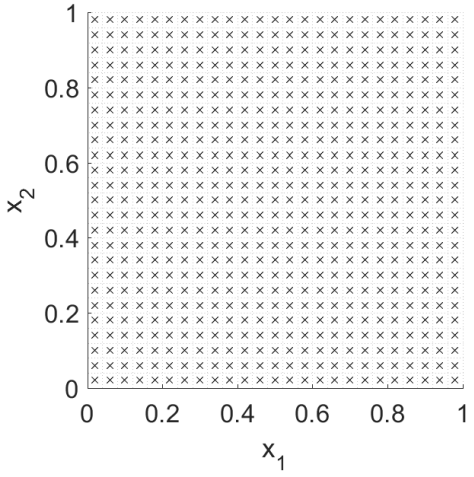


Figure A.1: Full factorial sampling in two dimensions, 25 × 25 samples

building:

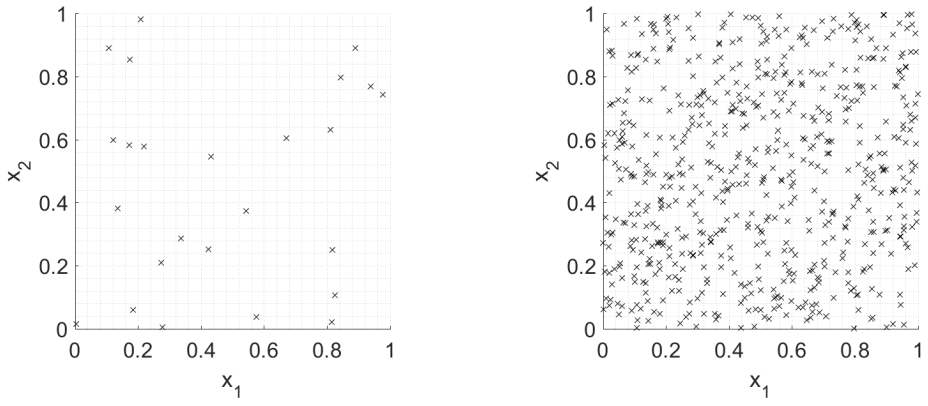
- the resulting designs of experiments must be of a specific size, i.e. the number of samples must be a product of the number of levels in each dimension,
- sets of points overlap when projected onto any single axis which is arguably an unnecessary duplication,

- depending on the problem, such an approach can lead to considerable unnecessary expense in sampling the extremities where few points lie in reality, and
- there is a possibility of aliasing (missed function features) if the true function is cyclic or contains cyclic features.

The method was first introduced as an alternative to the investigation of individual factors concurrently. It is seldom used now in experimental design but is representative of many surface measurements from probes due to sampling on a time, rather than a spatial, scale [245].

A.1.2 Pure random sampling

Instinctively, one might consider drawing pseudo-random [216] values from a uniform distribution in each dimension as a representative sample throughout a hyperspace. However, as we observe from an example of such a sample (figure A.2a), a number of points with values of one of more vary similar variable values can be created or some regions overlooked completely. To cover the full space, we add more and more points, tending in number back to a full factorial or more (figure A.2b). However, this sample continues to exhibit the same undesirable behaviour, in particular the wasted cost of almost identical points.



(a) 25 sample points (b) 625 sample points

Figure A.2: Pure uniform random sample in two dimensions

A.1.3 Optimal Latin hypercube sampling

Latin hypercube sampling (LHS) was introduced by McKay et al. [172] for statistical approximation using computer experiments as an alternative to full factorial or pure random sampling. It is a development of *stratified sampling* where, for uniformly distributed variables (or in a design case, equally important regions across the design space),

projections of the selected points onto axes are made as uniform as possible "by splitting the range of that variable into a relatively large number of equal sized bins and generating equal sized random sub-samples within these bins" [78]. Each variable is split into *strata* and they are sized for equal calculable *marginal probability* ([64] and [277]), or proportion if considering discrete subsets of a population. Random stratified sampling then draws randomly from the 'bins', whereas Latin hypercube sampling, based on the *Latin square*[1] (figure A.3), draws randomly only once from each strata e.g. figure A.4.

A	B	C	D
B	C	D	**A**
C	D	**A**	B
D	**A**	B	C

Figure A.3: Latin square, 4 samples

The LHS points can either be central to the strata; a *lattice sample* [195], or more likely taken assuming equal probability across each strata, distributing points randomly within 'bins'. This scheme removes the 'overlap', reducing the expense of sampling the entire

[1] A Latin square is an N × N matrix with each line and column a permutation of 1, 2, ..., N such that each number only appears once in each line and column.

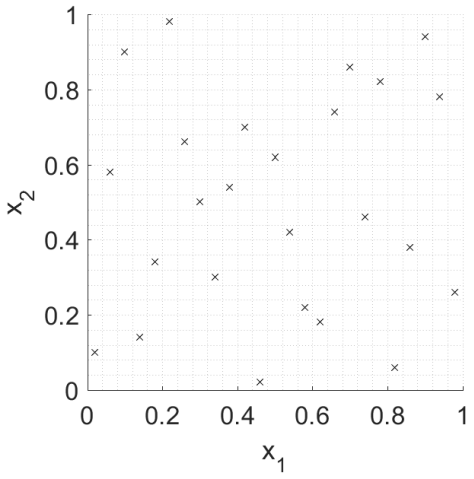


Figure A.4: Random Latin hypercube sampling in two dimensions, 25 samples, points with equal probability across strata

space, but the random nature means that this type of sample may not always exhibit the 'best' space-filling properties.

To improve upon the LHS, a number of researchers have developed optimal Latin hypercube (OLH) designs based on various optimality criteria. Popular criteria include the *maximin* criterion of Johnson and Moore [121] and Morris and Mitchell [181]'s scalar-valued rank to order alternative plans by their space-filling ability. The Morris and Mitchell [181] approach requires some form of global optimization routine and as such they choose simulated annealing (section 2.3). Forrester et al. [78] use an evolutionary algorithm by Box [18] to the same end. Figure A.5 provides a two-dimensional example of an OLH sample constructed using the Morris and Mitchell [181] criteria, compared with an equivalent random LHS.

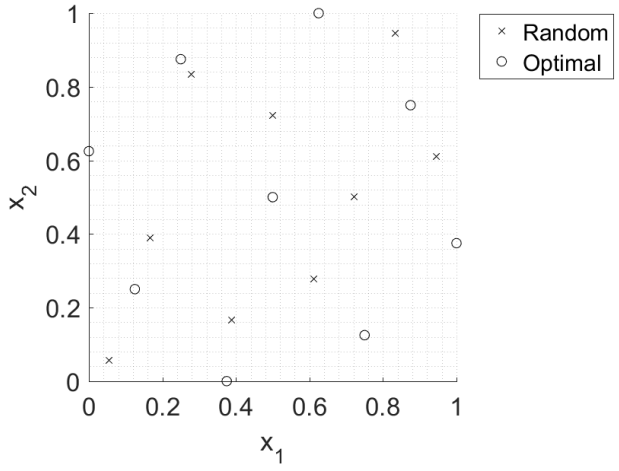


Figure A.5: Optimal (using Morris and Mitchel [181] criteria) Latin hypercube sample in two dimensions (25 samples, points with equal probability across strata), compared with a similar random Latin hypercube sample

These optimal[2] designs are extremely popular for building response surfaces (models of the design space also referred to as surrogate models or meta-models) but they do have one small pitfall: if the resource available to evaluate the points is unexpectedly changed then the sequence of points whose evaluation has completed will not necessarily be space-filling if cut short, or easily extended if additional resource becomes available. To combat this one could use an alternative plan based on, for example, 'Sobol' sequences such as LPτ [265]. This is not so well regarded, but the space is filled sequentially so the process may be interrupted at any point. Or, one must determine an easily reachable initial sample number prior to using other appropriate criteria to update the sample in regions of interest [78].

[2]Note that, as the number of points and dimensions increases, the number of possible permutations of a Latin hypercube increases at an incredible rate. For example, McKay and Rogoyski [171] calculate that there are more than 7.58×10^{24} permutations for a Latin square with just 10 points and 1.5×10^{86} for 15 points. As such, a truly 'optimal' Latin hypercube sample is untenable at a workable computational cost when large numbers of points are needed, rather they are 'optimized' at practical expense.

A.1.4 Orthogonal array based sampling

Orthogonal arrays (OAs), introduced by C. R. Rao in the 1940s, popularized by Genichi Taguchi, and explained by Ross [237] for quality engineering and in detail by Hedayat et al. [103] and Stufken [278], are also samples that derive from a stratified approach. They are a type of *fractional factorial* sample; a partial set of a full-factorial experiment selected in order to understand the main effects and low-order interactions, reducing the number of samples by ignoring high-order interactions between variables [182]. The OA is a matrix consisting of rows of experiments with entries from a set of symbols, or *levels*. If each $n_{\mathrm{OA}} \times t_{\mathrm{OA}}$ sub-matrix, where n_{OA} is the number of variables (*factors*), contains all possible $1 \times t_{\mathrm{OA}}$ row vectors at the same frequency, then the OA is said to be of *strength* t_{OA}. A Latin hypercube can be considered a special case of an orthogonal array where the number of experiments, or rows, is equal to the number of levels and the strength is one. Certain properties of some OAs are a distinct advantage for experimental design, in particular: uniformity properties over the entire space, and *orthogonality*, or lack of correlation, between design vectors [19]. It should be made clear here that "an orthogonal array is not an array whose columns or rows are orthogonal" in the sense that their *inner product* is zero, Hedayat et al. [103][3].

Figure A.6 illustrates two different OA designs. Construction of OA designs is not via a simple repeated algorithm, but using tables of designs that have been identified. A library of such OAs, produced by Sloane [261], accompanies the book by Hedayat et al. [103]. Figure A.7a shows an orthogonal design based on the OA in figure A.6a with random perturbations within the three levels. To combine the desirable properties of the Latin hypercube with those of the OA, Tang [283] introduced the OA-based Latin hypercube for design of computer experiments. The OA is transformed into a LHS whilst preserving its combinatorial structure, figure A.7b; each of the nine samples lies within one of the OA level combinations and none of the samples lie within the same nine rows or columns (strata for the hypercube).

Although OA-based samples are described for use in modelling, Sándor and András [244] apply them for the estimation of multivariate normal probabilities by employing a transformation based on a truncated normal density to a sample made using a uniformly distributed probability density function. Sándor and András [244] use the methodology used by Vijverberg [300] which is often applied to determining probabilities of rare events; *importance sampling*. Leary et al. [148] extend the OA approach to OA-based Latin hypercube samples.

[3] It is, in fact, the least squares estimators for the design vectors in a linear regression model of the design space that are orthogonal, or uncorrelated, and only if the strength of the OA is two or greater, [278, 103, 19]. According to [103], the name was first coined for these specific combinatorial structures of arrays by K. A. Bush in his thesis of 1950.

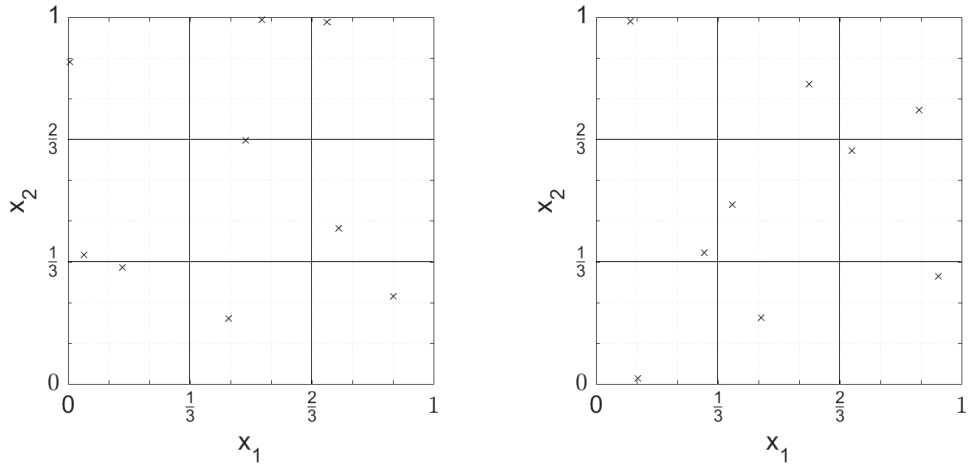
0	0
0	1
0	2
1	0
1	1
1	2
2	0
2	1
2	2

(a) Sample-size 9, with 2 factors, 3 levels, and of strength 2 (OA(9,2,3,2))

0	0	0
0	1	1
1	0	1
1	1	0

(b) Sample-size 4, with 3 factors, 2 levels, and of strength 2 (OA(4,3,2,2))

Figure A.6: Examples of orthogonal arrays



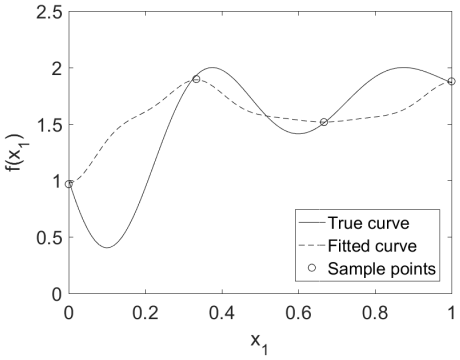
(a) Orthogonal design (orthogonal array based random sample)

(b) Orthogonal array based Latin hypercube sample

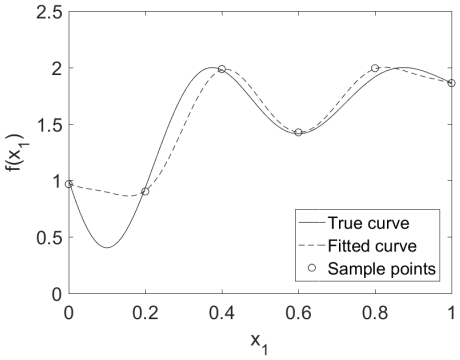
Figure A.7: Orthogonal array based samples in two dimensions (9 points)

A.2 Kriging with known noise

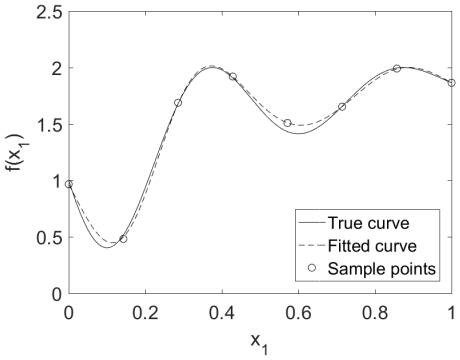
In the regressing Kriging approach by Picheny et al. [207] there are two variance parameters: the noise variance τ^2, and the Gaussian process variance σ^2. In this example, τ is fixed as 0.05 for the 1-dimensional example and 1.5 for the 2-dimensional example. σ^2 is found by optimization, although it is sometimes appropriate and feasible to search for both, or for τ using a fixed σ^2. Figures A.8 and A.10 show, as expected, a very similar fit to the regressing Kriging models in section 2.2.2. Again, a minimum does exist, figure A.9.



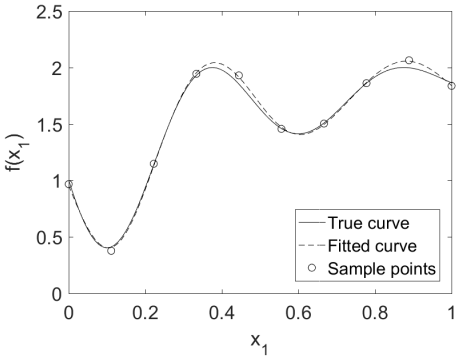
(a) 4 evenly distributed sample points



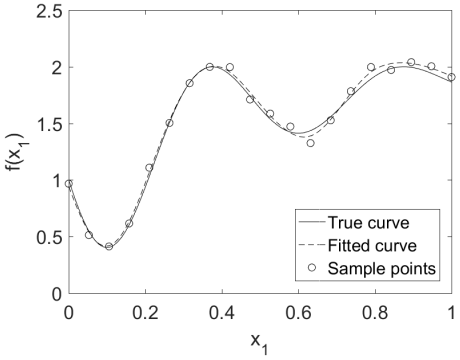
(b) 6 evenly distributed sample points



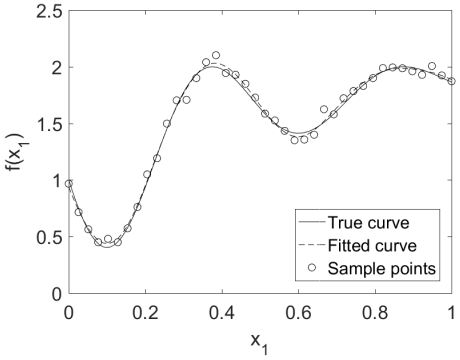
(c) 8 evenly distributed sample points



(d) 10 evenly distributed sample points

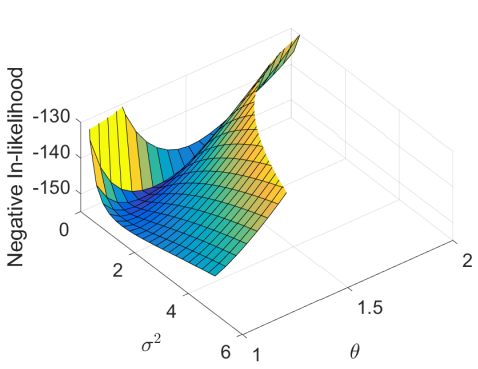


(e) 20 evenly distributed sample points

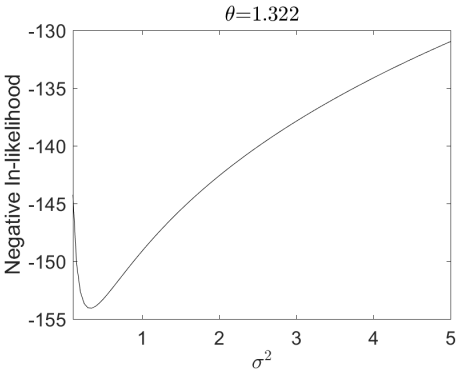


(f) 40 evenly distributed sample points

Figure A.8: Regressing Kriging fit with a fixed noise parameter in a single dimension

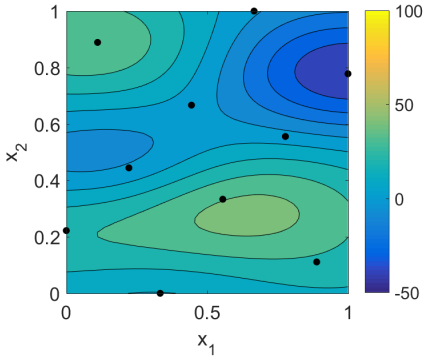


(a) σ^2 vs. θ vs. neg. ln-likelihood

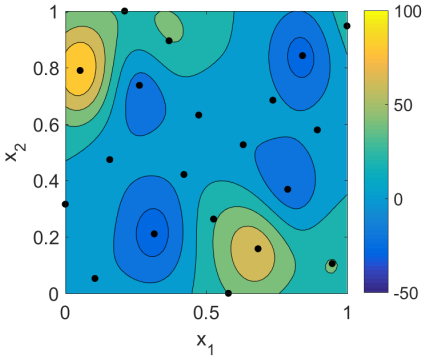


(b) σ^2 vs. neg. ln-likelihood at chosen θ

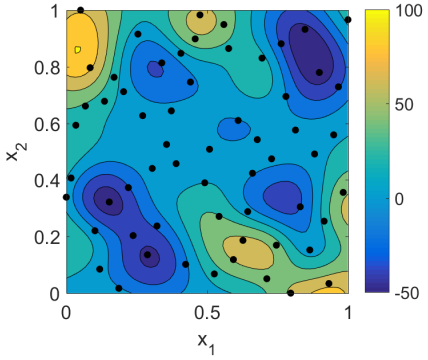
Figure A.9: Variation in negative ln-likelihood using regressing Kriging with a fixed noise parameter in a single dimension, 40-point sample



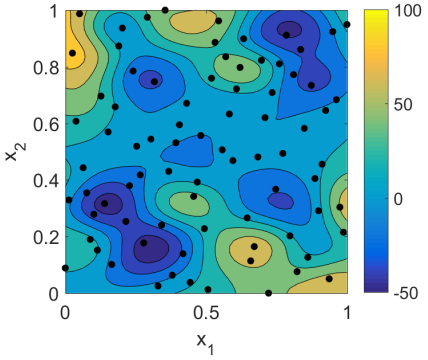
(a) 10 optimal Latin hypercube sample points



(b) 20 optimal Latin hypercube sample points



(c) 60 optimal Latin hypercube sample points



(d) 80 optimal Latin hypercube sample points

Figure A.10: Regressing Kriging fit with a fixed noise parameter in two dimensions

A.3 Local searches for an optimal design

Considering the definition of optimization given in section 2.3, if f is a linear function of independent variables with linear constraints (g) then long-established and efficient *linear programming* procedures based on *active set methods* such as *simplex methods* can be employed [216]. In many commercial codes enabling such approaches these routines have been superseded by *interior-point methods* first published by Karmarkar [128] who suggested that they converged much faster and could allow polynomial constraints [216]. These approaches have also been developed for quadratic objective functions and this is termed *quadratic programming* [91].

If there are no constraint functions (g) then a large number of *unconstrained* methods can be applied [91]. These include *Newton's method* and other second derivative methods, first derivative based *quasi-Newton methods*, and methods where the derivatives are approximated using *finite differences*.

If the constraints are non-linear then a number of approaches can be taken as described by Keane and Nair [130]. Possibly the most obvious, but not necessarily always feasible approach, is to transform the problem such that constraints are eliminated entirely if, for example, there is a direct known relationship between the constraint and one of the variables. If it is not possible to remove the constraints then equality constraints can be included through the use of *Lagrangian multipliers*, which alter the objective function to include a subtracting of the scaled equality constraint values at $\mathbf{x}$, i.e.

find

$$\mathbf{x} \in \mathbb{R}^n$$

which minimizes

$$f(\mathbf{x}) - \sum_{j=1}^{r_e} \lambda_{(L)j} g_j(\mathbf{x}). \tag{A.1}$$

The multipliers ($\lambda_{(L)j}$) become additional variables in the minimization problem but reduce it to the solution of a set of coupled non-linear equations. This approach can be extended to include inequality constraints by the addition of *slack variables* to the objective function. However, this methodology is limited by the requirement to include only constraints that are active at the optimum point and it is not always known which constraints these are. As an alternative Keane and Nair [130] introduce Zoutendijk's *feasible directions method* [327], which can be used for only inequality constraints but focuses on drawing the optimizer away from constraint boundaries. Starting at an active constraint boundary, it determines the direction in which the search will move by maximizing a linear function itself subject to linear constraints, by linear programming. If another constraint is hit, then the process is repeated, until an optimum is found and a local unconstrained search is carried out. This method can become particularly inefficient if the optimum lies in a relatively 'flat' region.

A popular and intuitive approach to dealing with non-linear constraints, is to apply a *penalty function* to any points where constraints are violated. This results in a number of difficulties, particularly for gradient-based searches, due to the effective discontinuity of the new objective function. A number of variations on this basic idea, including combined Lagrangian and penalty function based methods, are described by Keane and Nair [130]. Penalty function based approaches are none-the-less popular and, with some improvements can address some of the difficulties. For example, penalties may be applied separately for each constraint, the penalty may be made to vary depending on the size of the violation and could even be made to extend within the feasible (*interior*) space [78].

The most established method for applying non-linear constraints in local searches is *sequential quadratic programming* (SQP). SQP uses a quadratic approximation to the the 'Lagrangian' (the modified objective function after applying Lagrangian multipliers). The minimization of the 'Lagrangian' is iterative and uses linear constraints, resulting in an efficient process. It also allows for the active constraints to change at each step.

A.4 Global searches for an optimal design

Universally, these methods search the potential *design space* using some kind of iterative approach. The key difference is that the chosen locations at which evaluations of the function are obtained are chosen not necessarily moving closer to the optimum at each step in the process. They all include some random element of test design point selection and were popularized by the introduction of *evolutionary strategies* such as genetic algorithms (GAs) [92], introduced by Holland [105], where the point selection seeks to mimic natural selection. Many are described as *heuristic methods* due to the inexact, but globally improved, nature of the result. Developed in a similar period, there is also *simulated annealing* [194], which is analogous to the search for the minimal potential energy condition played out by cooling crystal structures. More recently, particle swarm based methods [289], introduced by Eberhart and Kennedy [62], that attempt to mimic the behaviour of flocking birds, have been proven to work well on some engineering problems.

In the case of constraints applied to global searches, one could deal with the constraints as discussed above, by modification of the objective function with a penalty term. Furthermore, the difficulties experienced with penalty functions by sequential gradient-based descent methods due to discontinuities are somewhat less critical if the search is moving in a non gradient based manner. There are other 'repair'-based techniques given a short overview by Keane and Nair [130], which result in 'flat' regions where constraints are violated, but these "are not simple to implement" or "widely popular" [130].

A.4.1 Genetic algorithms

The development of the genetic algorithm is usually attributed to John Holland, whose work of the previous decade was published in a book on the subject in 1975 Holland [105]. Goldberg [92] provides both the most commonly referenced tome on the subject of genetic algorithms and the basis of the genetic algorithm function (`ga`) in MATLAB. The popularity of genetic algorithm based searches for improvement has been growing ever since and continues to do so. Rather than using supplementary information about gradients, as most local searches do, genetic algorithms attempt to replicate the natural process of Darwinian evolution using only the fitness (objective function values) of *parents* (points) to move towards an optimal individual (design). Historically, these algorithms typically began with a random population represented as a set of binary strings (*chromosomes*) but more recently it has become more usual to use *real coded* approaches, where the chromosomes are defined by the double-precision floating-point format, if the variables are either continuous or of mixed format [28, 167]. The MATLAB (`ga`) function even uses, by default, a real-coded approach for integer optimization problems [51].

Given an initial population and corresponding fitness values, further populations (*generations*) are then created from this initial set by probabilistic processes of *reproduction*, *crossover*, and *mutation*. Reproduction and crossover reward fitness i.e. a chromosome with a higher 'fitness' value has a greater probability of 'mating' and producing 'offspring' for the subsequent generation. Mutation is a secondary 'insurance policy' which randomly introduces changes to chromosome values with a given probability of occurrence. The probability of mutation is very small but works to reduce the chance of missing something important: in nature, a characteristic that could increase fitness, in engineering, an undiscovered region of the design space with the potential for high-performance.

The `ga` function in MATLAB allows the probabilistic definitions of the functions defining the above processes to be user-defined, but for the engineering problems encountered during this thesis, the default values and functions for continuous variables suffice.

- A population size of 50 (selected using a uniform distribution over the variable space) for problems with five or fewer variables.
- A maximum of 75 generations unless an alternative convergence criteria is specified.
- A function to select the parents for the next generation (*selection function*) which "lays out a line in which each parent corresponds to a section of the line of length proportional to its scaled value. The algorithm moves along the line in steps of equal size. At each step, the algorithm allocates a parent from the section it lands on. The first step is a uniform random number less than the step size." [167]. This is equivalent to 'roulette-wheel' selection [155, 92] where the probability of

selection of the i^{th} individual is:

$$\mathrm{P}_i = \frac{f(\mathbf{x}_i)}{\sum_{i=1}^{n} f(\mathbf{x}_i)}, \tag{A.2}$$

differing only in so much as the starting point is at a random point at the beginning of the line, rather than a random point anywhere in a full circle. It is ensured that a minimum of 0.05 × 'the population size' individuals survive to the following generation.

- A 'scattered' crossover function is used; the parent parameter values are switched at random. Practically this is applied using a random binary vector for each 'mating' where values of 1 result in a 'switch' (the value from the second parent) and values of 0 maintain the value from the first parent. Other than 'elite' children (surviving parents), 80% of the next generation are produced by crossover.

- A Gaussian mutation function is used, in the unconstrained case, which adds a random value $\delta x_{i,j}$ for all parents i and dimensions j at each generation k where δx_i $\mathrm{N}(0, \sigma_k)$. σ_1 is the range of the initial population for an unbounded problem and reduces linearly to zero at the final generation. In a constrained problem the mutations are directional and the direction and step size are randomly generated to ensure bounds and constraints are satisfied.

- Prior to selection, a 'scaling function' scales the raw fitness scores based on the *rank* of each individual, removing the effect of 'scale' in the fitness function.

- The optimization process stops prior to completing the specified number of generations if the improvement over the most recent set of improving generations is less than a pre-set tolerance .

When non-linear constraints are introduced the MATLAB `ga` function employs the penalty-based "globally convergent augmented Lagrangian algorithm" developed by Conn et al. [43].

A.4.2 Particle swarm optimization

The particle swarm, first developed by Eberhart and Kennedy [62], was inspired by the behaviour of flocks of birds. To select a feeding site they use both new knowledge (observing other birds) and their own memory of the best spots to select the best location. This summary is based on the description given by Toal et al. [289]. An initial set (population) of *particles* (the birds) is selected using, for example, a perturbed Latin hypercube [172] sample within the given bounds. Each particle is given a random initial velocity and the objective function evaluated at the current location for all particles. The current global best position ($\mathbf{x}_{gb}$) and individual best position ($\mathbf{x}_{pb}$) are then calculated.

The velocities of each particle are then updated for the i^{th} iteration according to the following formula.

$$\boldsymbol{V}_{i+1} = w\boldsymbol{V}_i + c_1\boldsymbol{r}_1(\mathbf{x}_{pb} - \mathbf{x}_i) + c_2\boldsymbol{r}_2(\mathbf{x}_{gb} - \mathbf{x}_i), \tag{A.3}$$

where w is the inertial weight with $w\boldsymbol{V}_i$ giving the current particle velocity. c_1 and c_2 are constants defining 'cognitive' and 'social' behaviour of the swarm and $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ are random vectors in the range [0,1]. In terms of the search algorithm, $c_1\boldsymbol{r}_1(\mathbf{x}_pb-\mathbf{x}_i)$ pushes a point towards it's own previous best location, and $c_2\boldsymbol{r}_2(\mathbf{x}_gb-\mathbf{x}_i)$ determines how much influence the current global best position has on the velocity of each point. The original algorithm used $c_1 = c_2 = 2.0$. Given this new velocity, the new particle positions are simply calculated as:

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \boldsymbol{V}_{i+1}, \tag{A.4}$$

the objective at each point evaluated, and the process repeated for a given number of generations, or until a specified stopping criterion is reached.

Refinements of the process that remove the need to specify a reducing inertial weight have been developed. This involves the use of a *constriction factor*, Φ_{PSO}, alongside a choice of maximum velocity. The velocity is then calculated as:

$$\boldsymbol{V}_{i+1} = \Phi_{PSO}[\boldsymbol{V}_i + c_1\boldsymbol{r}_1(\mathbf{x}_{pb} - \mathbf{x}_i) + c_2\boldsymbol{r}_2(\mathbf{x}_{gb} - \mathbf{x}_i)], \tag{A.5}$$

where

$$\Phi_{PSO} = \frac{2}{|2 - \phi_{PSO} - \sqrt{\phi_{PSO}^2 - 4\phi_{PSO}}|}, \tag{A.6}$$

and

$$V_{i+1} \leq V_{\max}, \tag{A.7}$$

with $\phi_{PSO} = c_1 + c_2$ and $\phi_{PSO} > 4$.

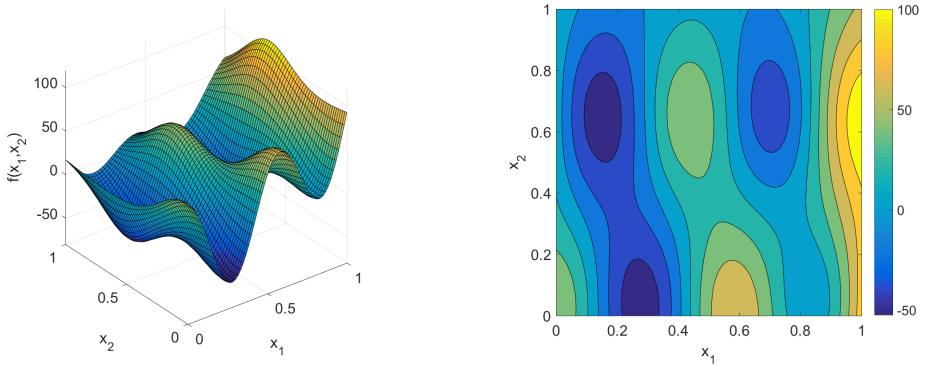
Further developments to the approach are the *hybridized* PSO algorithms, where local searches are combined with the global routine described above. Here, the algorithm of Toal et al. [289] is applied. During each iteration, after the objective function has been evaluated at each point, a single particle is selected for refinement using sequential quadratic programming (SQP), which can be derived from Newton's method, [185]. The location of this selected particle is then refined by using a partially converged run of a local optimization routine e.g. `fmincon` [163]. In addition to this local refinement of a selected individual particle on each iteration, a proportion of the particles are re-initialized (replaced with a new set) prior to the next iteration. The positions of the new set are selected to fill unexplored regions using a maximin distance criterion [181, 78]. This hybridization provides further global coverage to minimize the chances of missing a true global minimum.

At the final generation, a fully converged local search is performed at the selected 'best' particle to find the minimum point.

We present an example function, based upon a small section of the 'bird function' by Mishra [178]):

$$f(x_1, x_2) = \sin(-10x_1)e^{(1-\cos(-10x_2))^2} + \cos(-10x_1)e^{(1-\sin(-10x_2))^2} + (10x_1 - 10x_2)^2, \tag{A.8}$$

which exhibits multiple local minima across a two-dimensional space, figure A.11.

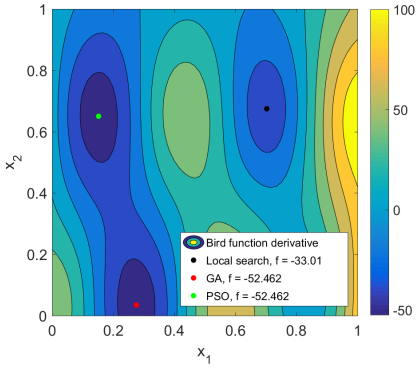


(a) Surface plot

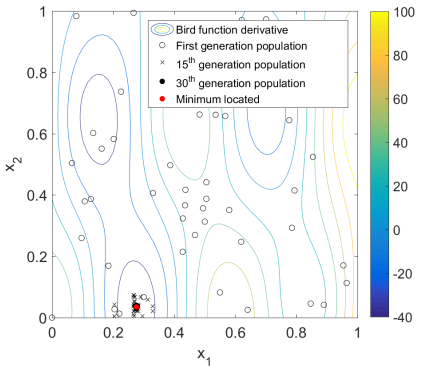
(b) Contour plot

Figure A.11: Example surface function example for demonstration of global searches

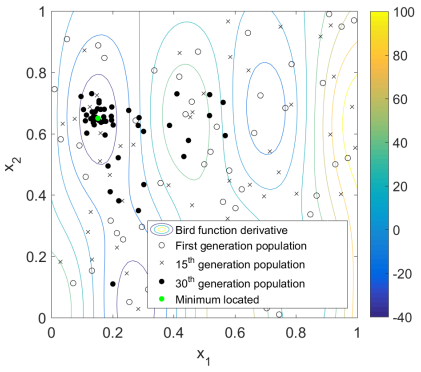
Figure A.12 shows the minima located using MATLAB's `fmincon` search using a random starting location, the genetic algorithm (`ga` with default options) based search result, and a particle swarm based search with an initial LHS, a population of size 50, 60 generations, and a maximum velocity of 0.075, i.e. each particle (point) is given a random velocity in the range $[-V_{\max}, V_{\max}]$, and $f(\mathbf{X})$ evaluated at each point.



(a) Local vs. global search



(b) Genetic algorithm based search



(c) Particle swarm optimization

Figure A.12: Global search examples

A.5 Monte Carlo (MC sampling)

The 'father' of sampling, the Monte Carlo method [266], was not the first ever foray into statistical sampling but its 'invention' was certainly pivotal in many fields. Its use was made practicable by the development of the first computers, and was the 'brain child' of Stanislaw Ulam, working with John von Neumann on a model of a thermonuclear reaction [175]. Independently, the same method was in fact developed by Enrico Fermi, but it was Ulam, along with Nicholas Metropolis who first published the method [176], and coined the name Monte Carlo. The basic premise of Monte Carlo is that a *simple random sample* of a system can be used to approximate the behaviour of the true system, and that as the size of the random sample increases the accuracy of the prediction converges towards the truth. In the case of a stochastic process with random input, we wish to predict the characteristics of the output pdf, based on a given input distribution: uncertainty propagation (UP). The advantage of MC is that it can be used to predict the full distribution and all its characteristics, and provides *unbiased* estimates of the mean and variance [307]:

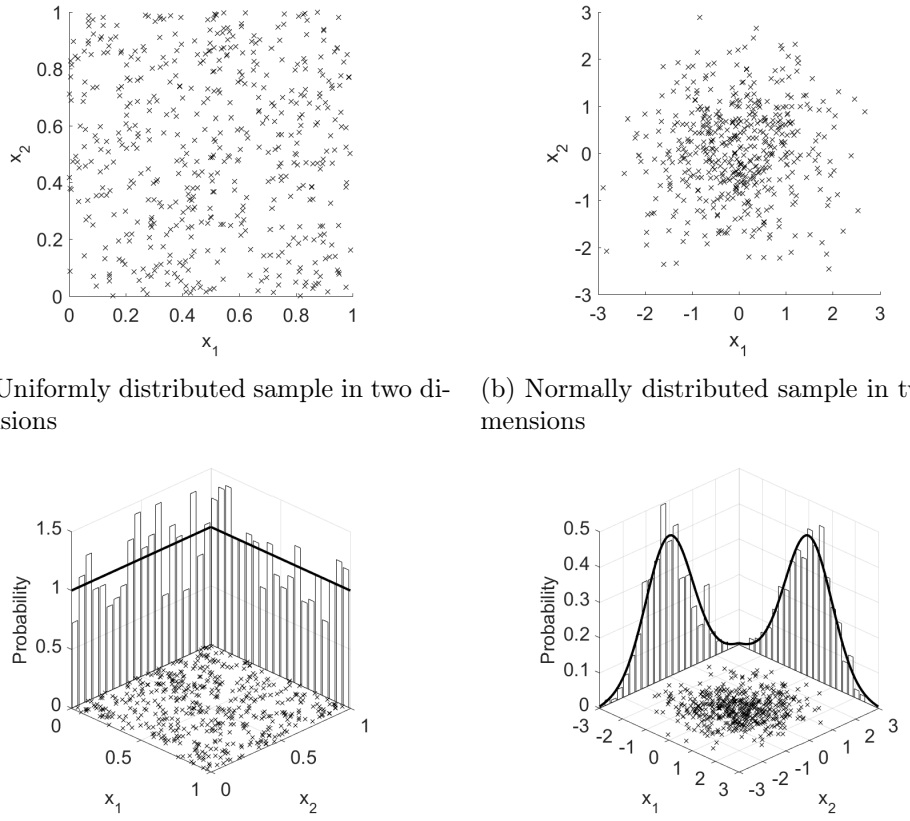
$$\hat{\mu}_{f_N} = \bar{f} = \frac{1}{N}\sum_{i=1}^{N} f_i \tag{A.9}$$

and

$$\hat{\sigma}^2_{f(N-1)} = \frac{1}{N-1}\sum_{i=1}^{N}(f_i - \bar{f})^2 \tag{A.10}$$

where N is the sample size and f is the performance function.

Figure A.13 shows the use of simple random sampling in two dimensions to illustrate Monte Carlo sampling of variables with uniform or normal probability distribution functions.



(a) Uniformly distributed sample in two dimensions

(b) Normally distributed sample in two dimensions

(c) Uniformly distributed sample in two dimensions, showing probability distribution and sample in each dimension

(d) Normally distributed sample in two dimensions, showing probability distribution and sample in each dimension

Figure A.13: Monte Carlo random sampling in two dimensions, 500 samples

The difficulty with simple random Monte Carlo sampling is one of expense. Although, since its conception, computing power has reached levels not imagined in 1949, the complexity of the computational models has also increased and continues to do so. By *central limit theorem*[4], it can be shown that the estimated probable error [266], using Monte Carlo reduces by a factor of $1/\sqrt{N}$, independent of the number of variables (dimensions) [216]. Despite the dimensional independence of the method, convergence of

[4] *Central limit theorem* was so-called by Georg Pòlya in 1920 [213]. The theorem takes many guises, now covers a variety of different results [25], and is described in many statistical texts including Sobol' [266] and Larsen and Marx [146]. Its foundations lie with Laplace, Chebyshev, Markov, and later developments with Poisson, Cauchy, and Levy, to name only a few [71].

typical engineering problems can require thousands of evaluations (samples) rendering its use impossible for complex problems. For this reason active research continues to this day seeking to improve dramatically on the cost of this established and reliable method.

McKay et al. [172] uses samples from uniformly distributed input variables and proves mathematically that, for an equally sized sample, "Latin hypercube sampling is better than random sampling for estimating the mean and the population distribution function". Where the input variables are distributed normally, or with any exponentially diminishing pdf, bounds that artificially truncate the sample are needed in order to define the equally probable strata. For normally distributed variables, however, this problem can be overcome by constructing a random LHS by perturbing a simple random sample drawn from the normal distribution, a technique introduced by Stein [274].

There are also a number of more recent developments of the basic Monte Carlo approach, developed mainly in financial modelling for investment banking, to reduce the numbers of sample points required. These are referred to as *quasi Monte Carlo* and differ from simple random Monte Carlo by sampling according to a *low-discrepancy*[5] sequence: *Halton* or *Sobol*, for example. A popular derivative of this kind of Monte Carlo for uncertainty is *Markov chain Monte Carlo* [3] where the sampling sequence is a *random walk* where each new sample is not independent of the previous ones. In this way, a sample that more closely resembles the desired distribution can be obtained with fewer samples.

However, none of these methods reduces the number of samples by the orders of magnitude that would be required to make them significantly useful in engineering design cases given the long-running workflows and, without exception, any other related methods are subject to the 'curse of dimensionality'; the number of samples required increases to the power of the number of dimensions, described particularly succinctly by Forrester et al. [78].

A.6 Quadrature

Instead of estimating the full distribution of the performance, it is often considered reasonable for RDO to estimate only the mean and variance of the performance function. Given that the mean and variance are integrals:

$$\mu_f = \int_{-\infty}^{+\infty} f(x_1...x_n)P(x_1...x_n)\mathrm{d}x_1...\mathrm{d}x_n \tag{A.11}$$

and

$$\sigma_f^2 = \int_{-\infty}^{+\infty} (f(x_1...x_n) - \mu_f)^2 P(x_1...x_n)\mathrm{d}x_1...\mathrm{d}x_n, \tag{A.12}$$

[5]Discrepancy is a measure of how far from *equidistribution* a sample is, i.e. low-discrepancy samples avoid regions of missing samples that can be introduced by an entirely random sample.

where P is the joint pdf of the variables x_i, it is also possible to use long-established *numerical integration* techniques for their estimation[6]: quadrature.

Full quadrature

The function is split into intervals and evaluated at the *abscissas* (the 'sample' points). Quadrature methods can be described as *open* or *closed* depending on how the extremities of the function are considered, *simple* or Gaussian with equally spaced or weighted abscissas, *non-adaptive* or *adaptive* where integrals are evaluated over the full range, or split. Quadrature may be applied after *variable transformation* and in multiple dimensions. Without exception though, all *full quadrature* schemes require large numbers of function evaluations. In addition, as the number of dimensions and required tolerance increases, the number of evaluations increases in an exponential manner.

Quadrature rules date back to "Newton, if not farther" [216]. The simplest historical rules include the 'trapezoidal rule', 'Simpson's rule', and 'Bode's rule' and 'extended' or 'composite' rules based on these are detailed by Press et al. [216]. All integral formulae are some form of weighted sum of evaluations at the sample points. For linear functions, even the simplest rule is exact, but for non-linear functions, or those of an asymptotic nature (we are dealing with both), the error in the integral estimation can be high, even with a large number of points. For cases of *improper integrals*, where the limits are infinite (such as the normal Gaussian pdf) it is typical to use extended rules such as the 'extended mid-point rule', or more commonly use transformations or expansions to split the integral into a set of simpler integrals e.g. *Fourier transforms* or *Taylor series expansions*. The polynomial chaos expansion (PCE) is one such expansion used specifically to expand a pdf.

Gaussian quadrature has non-uniform distribution of abscissas and it is possible to choose the weights to result in an exact solution for the integral of a polynomial function. Most popular algorithms for determining abscissas are are now based on the Gaussian approach: calculation of abscissas and weights to provide a precise answer for a specific type of function. Examples include the Gauss-Legendre or Chebyshev algorithms for problems with finite bounds, and Gauss-Laguerre or Gauss-Hermite for infinite bounds [135].

Most implementations of quadrature now include adaptive schemes. Standard schemes increase the number of sample points to the next available *level* (usually a large increase in evaluations) and compare the predictions from the two sets to give an error estimate. If the error is sufficiently small then the last set of points has been deemed superfluous to requirement but the cost of its evaluation has already been expended. Adaptive schemes,

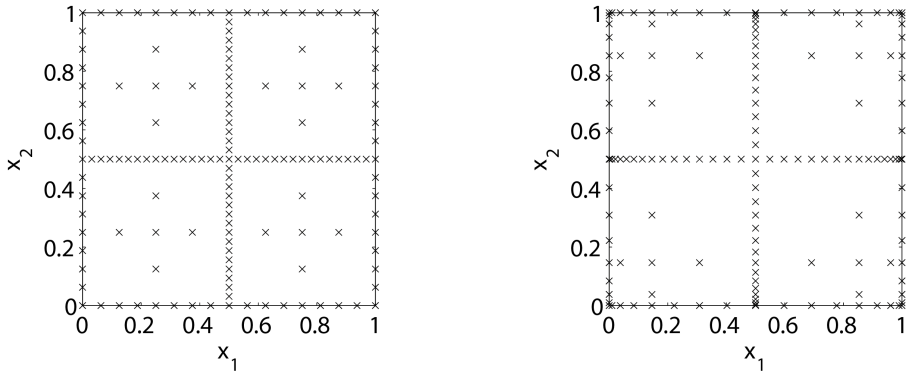
[6]These integrals generally do not have exact analytical forms so numerical methods are usually required.

however, evaluate this error from the first two sets and if the tolerance is not met, split the integral into two. They then recursively subdivide the resulting functions until each of the new functions have a sufficiently small error and the results can be summed. The MATLAB quadrature tools apply a variation on adaptive Lobatto quadrature [87, 251].

Quadrature schemes increase in complexity when extended to multiple dimensions and freely available code often only extends to two or three possible dimensions. In addition, the numbers of function evaluations are still significantly large and as such are restrictive in their use for engineering uncertainty calculations.

Sparse grids for quadrature

None-the-less, recently Xiong et al. [320] proposed the use of *sparse grid numerical integration* as a novel approach to statistical moment estimation for uncertainty propagation in engineering design. The methodology uses a significantly reduced (sparse) set of abscissas (*sparse quadrature* (SQ)). As with full quadrature, the accuracy of the integral is dependent upon the level of *polynomial exactness* chosen, and as one would expect, the cost (number of points) increases with exactness. Xiong et al. [320] have proven the potential of the technique for UP with analytical test problems with up to thirteen dimensions by comparing with a meta-model based approach and direct Monte Carlo sampling. Padulo et al. [197] also successfully apply a *reduced quadrature* based approach to moment calculation and extend this to RDO via an airfoil design problem. A toolbox for implementing the selection of sparse grids and their adaptive counterparts is provided by Klimke [136] and example grids are shown, figure A.14



(a) Clenshaw-Curtis sparse grid (b) Chebyshev sparse grid

Figure A.14: Example sampling plans (grids) for integration by quadrature (level 5 polynomial accuracy; 145 points)

There are still difficulties, which may not be insurmountable, with SQ:

- it relies on interpolating through a set of fixed points meaning that any 'noise' in the data, from whatever source, renders the results meaningless,

- there is no probabilistic measure of error in the fit, such that it is difficult to use the method independently and have an understanding of how well it has performed in its prediction,
- a mathematical description of the multivariate distribution of the combined uncertain variables is required,
- most schemes developed thus far are based on polynomial exactness, which may not be appropriate depending on the performance function, and
- if, for any reason, there are failed or missing sample points, the calculation fails.

Although many fewer sample points are required than for MC or full quadrature, SQ can still be prohibitively expensive for most practical engineering problems.

Once RDO has been employed using an approximation of the mean and central moments to determine an optimum design point, it will be pertinent to use a propagation technique to determine the full output pdf at that optimum. Only direct sampling, or RSM building followed by direct sampling of the model, enable estimation of the full output pdf. If the resulting point is close to a constraint boundary one may also wish to estimate the probability of failure, or even propose a further RBDO search locally.

A.7 Classical curve and surface fitting methods

A.7.1 Polynomial models and least-squares regression

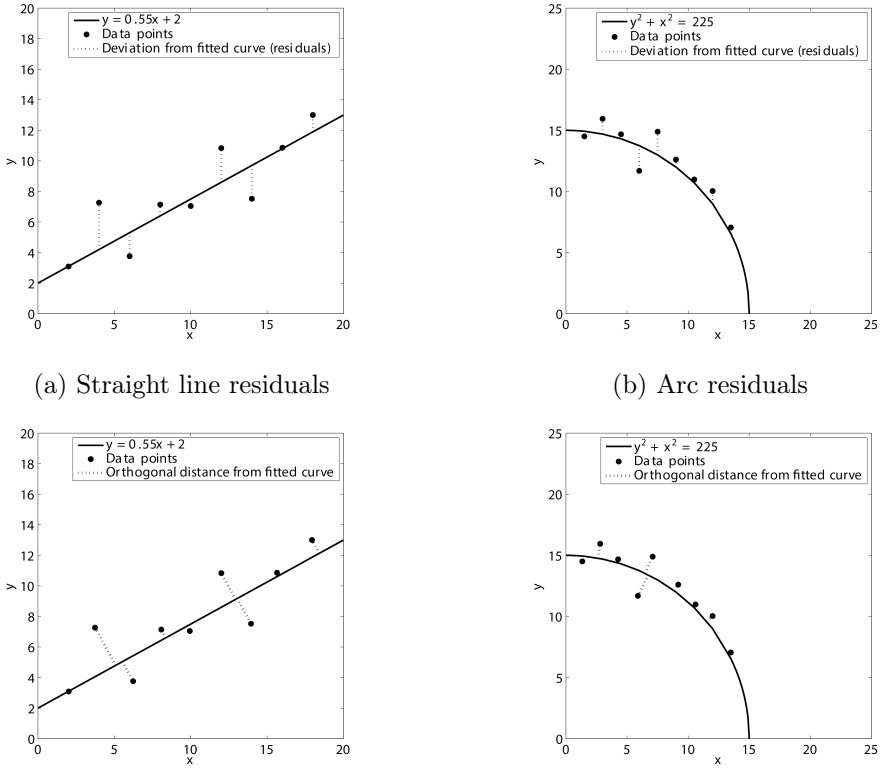
It may be assumed that the data points contain only random error (equal, uncorrelated, normally distributed error), or noise. Our goal is then to find the 'best' possible prediction of the 'true' curve assuming that our analytical form (polynomial of a given degree) is 'correct'. This is regression, and "the earliest form of regression analysis for linear problems was studied independently by Gauss[7] and Legendre[8] in the 17$^{\text{th}}$ century. Their method is a technique which minimizes the sum of the squares of the *residuals* (differences between fitted function and given data) to find a best fit" [41], figure A.15. If x is the variable which is controlled in an experiment, often termed the *independent* variable, or *regressor* and y is the measured variable, *dependent* variable, *regressand*, or *response*, then the least square method minimizes S, where

$$S = \sum_{i=1}^{n} (y_i - \hat{y}(x_i))^2, \tag{A.13}$$

[7]Carl Friedrich Gauss is usually credited for the inception of least squares [64] in 1795 although he didn't publish it until 1809 [267].

[8]Adrien-Marie Legendre first published the method of least squares in 1806 [267].

and $\hat{y}(x_i)$ is the estimated response at each of the n control points, x_i. $\hat{y}$ is a curve parametrized by $\mathbf{u} = u_{j=0}...u_{m_p}$ estimated parameters and the best estimate is 'trained' on the data points $y_i(x_i)$ for $i = 1, ..., n$.



(a) Straight line residuals

(b) Arc residuals

(c) Straight line orthogonal distances

(d) Arc orthogonal distances

Figure A.15: Illustration of residuals and orthogonal distances from a straight line and an arc

The method that Legendre named the method of *least squares* referred to fitting a polynomial model ($\hat{y}$) with $m_p + 1$ polynomial coefficients:

$$\hat{y}(x) = \sum_{j=0}^{m_p} u_j x^j, \tag{A.14}$$

where m_p is the polynomial degree.

Prior to fitting, one must determine the most appropriate degree of the polynomial model. This is critical in any application. A hypothesis test can be conducted as described by Ralston and Rabinowitz [224] to ensure that over-fitting (where additional parameters become superfluous to the prediction of the response and instead begin to describe the noise) does not occur. The polynomial has chosen degree (m_p), and if there is no error in the data then an additional term and it's coefficient $u_{m_p+1}^{m_p+1}$, will be zero. However, the errors are in fact considered normally distributed with variance σ^2 and points with unequal importance or unequal measurement accuracy are weighted with variance σ^2/w_i.

Ralston and Rabinowitz [224] apply the *null hypothesis* H_0: $u_{m_p+1}^{m_p+1} = 0$ and given that the curve of maximum-likelihood [216] will lie where the sum of the squared deviations from the points is minimized, they state an expected value for the variance:

$$\sigma_{m_p}^2 = \frac{\delta_{m_p}^2}{n - m_p - 1}, \tag{A.15}$$

where

$$\delta_{m_p}^2 = \sum_{i=1}^{n} w_i (y_i - \sum_{j=0}^{m_p} u_j^{(m_p)} x_i^j)^2 . \tag{A.16}$$

A null hypothesis that can not be rejected will result in a variance that is independent of polynomial degree and further increases in degree will result in negligible change in $\sigma_{m_p}^2$.

Having defined the polynomial degree, we turn to minimizing the sum of the squared deviations of the data points from the model values (S, which is unweighted). *Linear least-squares* can be used when the model is "linear in the parameters" [19], and the polynomial model can be written in matrix form as:

$$\mathbf{y} = \mathbf{Zu} + \boldsymbol{\epsilon}, \tag{A.17}$$

where $\mathbf{y}$ is a vector $y_1, y_2, ...y_n$ of sample response values with a vector of corresponding errors $\boldsymbol{\epsilon}$. $\mathbf{u}$ is the vector of the unknown coefficients, $\mathbf{Z}$ is an n by $(m_p + 1)$ matrix of postulated terms (in the case of a polynomial these are $z_{1n} = 1, z_{2n} = x, z_{3n} = x^2, ...z_{m_p n} = x^{m_p}$ for y_n). Similarly, S can be expressed in matrix form:

$$S(\mathbf{u}) = (\mathbf{y} - \mathbf{Zu})^T (\mathbf{y} - \mathbf{Z}\boldsymbol{\epsilon}). \tag{A.18}$$

By formulating equations for the partial derivatives of the sum of the squared deviations with respect to each of the polynomial coefficients, the equations are found to be linear and the unique solution is a minimum [145]; the 'best fit'. These are the *normal equations*:

$$\mathbf{Z}^T \mathbf{Z}\hat{\mathbf{u}} = \mathbf{Z}^T \mathbf{y}, \tag{A.19}$$

and there is a closed form solution. Linear least squares is often also termed *ordinary least squares* and numerous texts describe the method and approaches to solving this *linear system*, e.g. Lancaster and Šalkauskas [145], Larsen and Marx [146], Box and Draper [19], Myers and Montgomery [182], and Press et al. [216]. In particular, we direct the interested reader to Press et al. [216], for *LU decomposition*; a splitting of the matrix $(\mathbf{Z}^T\mathbf{Z})$ into lower and upper triangular matrices $\mathbf{L}$ and $\mathbf{U}$, by *Cholesky decomposition*; i.e.

an upper triangular matrix and its transpose, or *Gauss-Jordan elimination*; incorporating an identity matrix and using Gaussian elimination.

These methods still exhibit difficulties for complex problems, such as numerical instability, round-off error during matrix inversion, or *singular* matrices in certain circumstances. In addition, if the Z's have linear relationships for example, construction of the normal equations will result in a set of equations without a unique solution. *QR decomposition*, or *factorization*, can be utilized to overcome round-off errors and is a very numerically stable algorithm [166], used in MATLAB with pivoting for linear least-squares fitting. When the matrix is near-singular or rank-deficient (there are some independent linear relationships between columns of $\mathbf{Z}$) singular value decomposition (SVD) (section A.9.2) is typically used [216, 310].

Where there is more than one *explanatory variable* (regressor), it is termed *multiple regression* and we refrain from its discussion here, except to point out that *principal components regression* is employed to determine the regressor with most effect on the response, and is not involved with determining the principal components within the model of the response, the realm of principal component analysis (PCA, section A.9.1).

Where the set of equations derived by this approach is non-linear, one must take an iterative approach, or use a general unconstrained minimization routine [163] to converge to a solution.

There are a number of variations of least squares, linear or non-linear, including *weighted least squares*, also termed *moving least squares*, which is primarily used for multivariate regression to increase the level of interpolation through specific points, the extreme case being *interpolating moving least squares* [145]. There is also *iteratively weighted least squares* which revises weights at each iteration [64]. In general, procedures that attempt to reduce the sensitivity of regeression to uncertainty in the assumptions are referred to as *robust regression*. In particular "least squares estimation is known to be sensitive to *outliers*" [64], and the study of problem areas in the fit is sometimes referred to as *interior analysis*. A further noteworthy inclusion is *piecewise regression*, or *segmented regression*, problems for both continuous and discontinuous function predictions [170, 212, 288, 290].

A.7.2 Orthogonal least-squares and similar fitting routines

Crucially, traditional forms of least squares fitting make one over-riding assumption: that the error only occurs in the regressand, y, and not in the regressor, x. Willink [316] discusses the use of the non-orthogonal residual in the ISO standard for the expression of uncertainty in measurement (the 'GUM') [116, 117]. In the geometric point cloud problem, in two or three dimensions, one must consider that error occurs in all dimensions. Intuitively, the alternative similar method uses the *orthogonal distance*, or *orthogonal*

residual, d, in place of the residual discussed thus far, figure A.15. The minimization then becomes the minimization of R, where

$$R = \sum_{i=1}^{n} {d_i}^2(u), \tag{A.20}$$

and d is a function of $\mathbf{u}$. The goal is to find

$$\min_{\mathbf{u}} R. \tag{A.21}$$

This assumes that the variances in the x, y, and z coordinate measurements are equal. This is sometimes termed *normal least squares*. The minimization is generally non-linear with no closed-form solution so an iterative procedure must be employed.

Alternative orthogonal distance based fitting routines include *least absolute deviation regression*, and *least median of squares*, which is median-based and can be preferable where the error is asymmetrically distributed or contains outliers [40]. A popular method in the literature, for inspection, is the *minimum-zone method*, attributed to Le and Lee [147], where the distance between the two parallel curves which enclose all the data points is a minimum. Such a method, as with least-squares, is sensitive to outlying erroneous data points. It also suffers from a tendency to underestimate the 'true' deviation range from the nominal, especially where the number of measured points is small. It is well suited where inspection is the primary motive but it falls short as a means of representing the true curve.

A.7.3 Spline fitting

Splines are a series of piece-wise, or concatenated, curves connected at their end points, often with tangency or smoothness constraints at intersections. They range from piece-wise polynomial formulations to extensions of *Bézier splines* and curves, developed to give a parametric definition of curves for automobile shape design whilst enabling intuitive manipulation by designers. The popular *rational Bézier spline* is parametrically defined as

$$G(t) = \frac{\sum_{i=0}^{K_s} W(i)P(i)b_i(t)}{\sum_{i=0}^{K_s} W(i)b_i(t)}, \tag{A.22}$$

where G is the curve as a function of t, the distance along its length from P_0, and the number of control points is $K_s + 1$. $\boldsymbol{P}$ are control points, $\boldsymbol{W}$ are weights attributed to the control points. The spline is *rational* because t and W can be varied between 0 and any value; the parametric definition is rationalized by the denominator term. $\boldsymbol{b}$ is a vector of basis functions, which are *Bernstein polynomials* in the case of a Bézier spline.

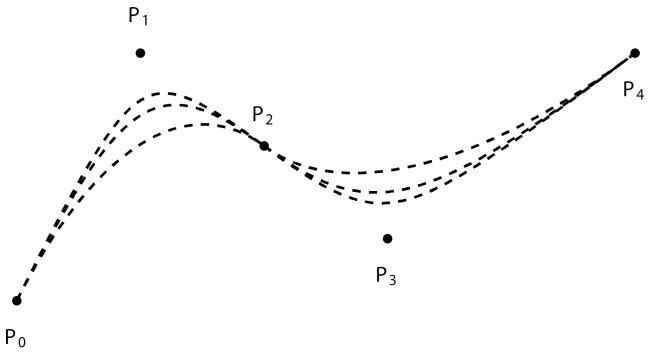


Figure A.16: Second order rational Bézier spline, varying weights of control points P_1 and P_3

Figure A.16 shows a second order rational Bézier spline with increasing weights, $\boldsymbol{W}$. Its order (O) is two because each curve segment is controlled by three ($O + 1$) points.

In their most generic form, *b-splines* and their analogous surfaces are termed NURBS, (*non-uniform rational basis-splines*), and NURBS surfaces respectively. They overcome a couple of key limitations of the Bézier splines: the degree of the curve being fixed by the number of control points, the lack of local control of the curve, fixed continuity and tangency between segments. This additional flexibility is obviously of benefit when local geometric perturbations need to be represented. To achieve this, the b-spline is split using *knots* and the basis function is a function of a *knot vector*, $\boldsymbol{T}$. This means that local control of weights and degree of continuity is possible and that curve segments do not have to end at control points. Further detail on the construction of NURBS and specific splines is available in many texts, e.g. de Boor [48], Salomon [243], and Farin [67].

Classical spline fitting (or *approximation*) approaches are documented by Dierckx [57]. The routines are either based on:

- weighted or unweighted least-squares, or
- a smoothing criterion in combination with, or constrained by, a least squares function.

Most fitting methods assume cubic splines and search for the number of knots and their positions, and the *b-spline coefficients* [57]. The parameter vector can be large, giving rise to an expensive optimization problem [85].

The least-squares problem is formulated as described in section A.7.1, and the alternative constrained smoothing problem can be formulated as

minimise:

$$\eta = \int_{x_1}^{x_n} (\hat{y}^{(k)}(x))^2 \, dx \tag{A.23}$$

subject to

$$S := \sum_{i=1}^{n} (w_i(y_i - \hat{y}(x_i)))^2 \leq S_{max}, \tag{A.24}$$

where S_{max} is a specified number referred to as the *smoothing factor*, which could have its origins in the variance of the measurement error if using inspection measurements. k is the chosen degree of the spline. "η can be seen as a measure for non-smoothness" and S "as a measure of closeness of fit" [57]. One might also consider the criterion of Powell, to determine a cubic spline $\hat{y}(x)$ that minimizes

$$\varphi = S + \sum_{j=1}^{m} (w_j D'''_j)^2, \tag{A.25}$$

where D'''_j is the third derivative (discontinuity jump) in the spline at the j^{th} knot. Powell's method, described by Dierckx [57], is a two-stage process where the minimization first determines the knots and then the remaining parameters. A similar approach, which combines (A.23) and (A.25) minimizes:

$$\tilde{\eta} = \sum_{j=1}^{m} (s^{(k)}(T_j+) - s^{(k)}(T_j-))^2, \tag{A.26}$$

subject to

$$S := \sum_{i=1}^{n} (w_i(y_i - \hat{y}_i(x_i)))^2 \leq S_{max}, \tag{A.27}$$

where T are the knots and k is the specified degree of the spline.

Further variations on this exist. For example, the MATLAB smoothing spline minimizes

$$s_S \sum_{i=1}^{n} (w_i(y_i - \hat{y}(x_i)))^2 + (1 - s_S) \int_{x_1}^{x_n} v(x_i) |\hat{y}''(x_i)|^2 dx, \tag{A.28}$$

where $\boldsymbol{v}$ is a vector of constants, $\hat{y}''(x_i)$ is the second derivative of $\hat{y}$, and s_S is the *smoothing parameter* which determines the balance of smoothing to fit, i.e. $s_S = 0$ results in a least-squares linear fit, and $s_S = 1$ results in a *natural* ('not-a-knot' end conditions) cubic interpolating spline. s_S can be specified, a maximum *error measure* (first term) value supplied, or a value of s_S determined by $s_S = 1/(1 + \frac{h^2}{6})$, where h is the average data spacing (the expected s_S for uniformly spaced data).

A.7.4 Orthogonal distance based spline fitting

The procedures described above are posed algebraically, in a similar manner to polynomial regression; y is a dependent variable, expressed in terms of x and a vector of parameters defining the function. If however, the spline is defined in terms of t, along its length, a minimization of the squared residuals necessarily normal to the curve, i.e. the l_2 *norm* or the squared Euclidian distance, is required. Given that the spline curve can be expressed as two individual spline functions $x = G_x(t)$ and $y = G_y(t)$ over a given range of t, Dierckx [57] expresses the orthogonal least squares minimization criterion as

$$S = \sum_{i=1}^{n} ({w_i}^2((x_i - \hat{G}_x(t_i))^2 + (y_i - \hat{G}_y(t_i))^2))^2. \tag{A.29}$$

The smoothing criterion can also be extended in an orthogonal sense: minimize

$$\tilde{\eta} = \sum_{j=1}^{m} ((G_x^{(k)}(T_j+) - \hat{G}_x^{(k)}(T_j-))^2 + (G_y^{(k)}(T_j+) - \hat{G}_y^{(k)}(T_j-))^2), \tag{A.30}$$

subject to

$$S \leq S_{max}. \tag{A.31}$$

The precise construction of the minimization problem varies, as do the parameters that are held fixed and the optimization algorithms used. They have in common however, as Chivate and Jablokow [40] point out, that curves are usually fitted using the method of least squares, achieved almost universally by fitting gridded data, or converting scattered data to a gridded form. Chivate and Jablokow [40] emphasise that a benefit of a parametric representation is the ability to robustly calculate intersecting curves, although the surfaces often need to be extended first. In addition, NURBS benefit from the ability to define curves and surfaces with discontinuities, and can also represent closed sectional profiles, e.g. airfoil sections [133]. Chivate and Jablokow [40] recommended parametric surface fitting over algebraic fitting for gridded data points and introduce a two stage process of fitting algebraic surfaces to generate a grid, followed by a parametric surface fit.

The orthogonal least squares problem as expressed (A.29 or A.30) is a single objective minimization with multiple bounded variables, the number of which depends upon the chosen parameterization of the curve. In a review of the fitting routines employed in CMM software for fitting curves to measured data, Dowling et al. [61] state that "the implementation of optimization routines and approximation algorithms, ..., is not standardized ..., and much variability exists". We find that the most commonly published algorithms are *Gauss-Newton* based local searches. Forbes [76] summarizes this procedure in a measurement context, although it is also well documented elsewhere, [163, 91, 185]. Atieg and Watson [7] compare three Gauss-Newton based methods and their efficiency in

terms of number of iterations for circle and sphere fitting. Nocedal and Wright [185] use a *Levenberg-Marquardt trust region* based local approach. Divergence and slow convergence are often encountered with these methods, especially where poor starting values are chosen [76]. If we wish to fit multiple (possibly hundreds of) similar data sets, where the true geometry may potentially be far from the nominal (the obvious starting point), a global search may be more appropriate to avoid convergence problems even if speed is sacrificed.

Given that splines are polynomial-based curve representations, the fitting routines "usually" utilized within co-ordinate metrology are Gauss-Newton type linear least squares solutions to orthogonal distance regression [249]. However, the more flexible b-splines include knots, and their inclusion as variables renders the optimization problem highly non-linear and the determination of a global optimum is more difficult [297]. According to Van Loock et al. [297], numerous methods have been proposed to solve this problem, many requiring a good initial guess of the knot locations and often being unable to guarantee global optimality.

Approaches to this problem differ in both chosen optimization routine and problem construction. Many propose a splitting of the optimization into stages (solving for one set of variables, and then another). For example, Demeulenaere et al. [54] propose a construction of the problem such that it is convex, by providing a large fixed number of candidate knot locations so a solution is more easily obtained. Van Loock et al. [297] enhance this structure to: improve numerical stability, expand the ability to introduce contraints, and reduce the number of active knots still further. An example of the application of b-spline fitting to point measurement data was given by Khameneifar and Feng [133] for the inspection of turbine blade sectional measurement data. Their optimization considers the measurement uncertainty as a constraint and is a least squares minimization of the Euclidean distances with constraints on continuity.

Surface-type, rather than curve-type problems introduce further difficulties in terms of expense due to the number of points and additional coefficients, as well as risking failure with unevenly distributed points and surfaces of non-zero *genus* (with 'holes', or *handles*). Gálvez et al. [85] (2012) gives a succinct summary of typical approaches and compares his GA-based approach with six alternative methods.

There is a large array of global optimization algorithms, as described in section 2.3. Most examples of their use for curve and surface geometry fitting from measurements fit NURBS surfaces. Gálvez and Iglesias [84] demonstrate the use of a particle swarm algorithm and Gálvez et al. [85] refer to a number of examples of papers demonstrating the use of global optimization.

A.7.5 Bayes regression

This review omits the Bayesian approach to regression, which is currently in its infancy in its application to point cloud measurement. Willink [316] and Lira et al. [156] are recent examples for straight line fitting to uncertain measurement data. In all the approaches discussed so far, "parameters are assumed to be unknown but fixed, and are estimated with some *confidence*" [164]. With the process utilizing Bayes' theorem, "the uncertainty about the unknown parameters is quantified using probability so that the unknown parameters are regarded as random variables" [164].

A.8 Statistical methods for uncertainty representation

A.8.1 Popular distributions

In this thesis we are mainly concerned with continuous data. It is usual, in the absence of any other information, to assume the Gaussian normal distribution:

$$P(x_1) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left[-\frac{(x_1-\mu)^2}{2\sigma^2}\right], \tag{A.32}$$

for a single random variable x_1, figure A.17a. This is a function describing the *marginal* (unconditional) probability of drawing each value of x_1 randomly. When there are multiple random variables with normal distributions their *joint* distribution function is given by:

$$P(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{d}{2}}|\mathbf{V}|^{\frac{1}{2}}}\exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\mathbf{V}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right], \tag{A.33}$$

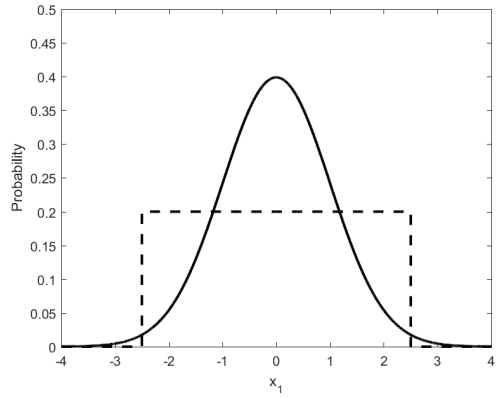
where $\mathbf{x}$ is a vector of d variables $x_1, ..., x_d$, $\boldsymbol{\mu}$ is a vector of means and $\mathbf{V}$ is the covariance matrix, which must be *symmetric*, *positive definite* for a distribution function to exist. In these cases, the variables may, or may not, be correlated. Figure A.17b shows an illustration of the bi-variate case for two uncorrelated variables.

It is common practise in uncertainty studies to also either consider input variables to be distributed uniformly such as the example in figure A.17a. The probability distributions may also require truncating, which may be necessary where inspection processes are used to identify components that exceed tolerance limits and remove them from service, figure A.17c.

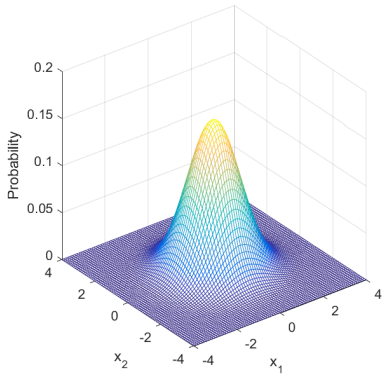
The log-normal distribution, figure A.17d:

$$P(x_1) = \frac{1}{2\pi x_1}\exp\left[-\frac{(\ln x_1)^2}{2}\right], \tag{A.34}$$

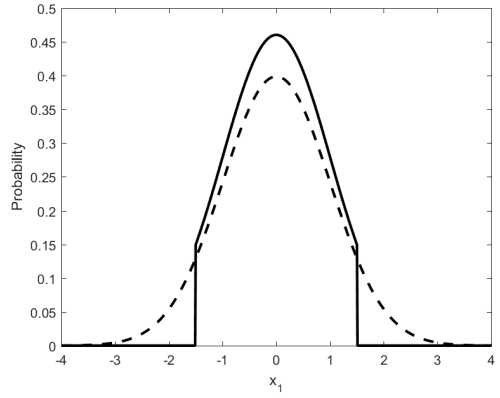
is also a popular distribution in engineering design, particularly for representing material property variations.



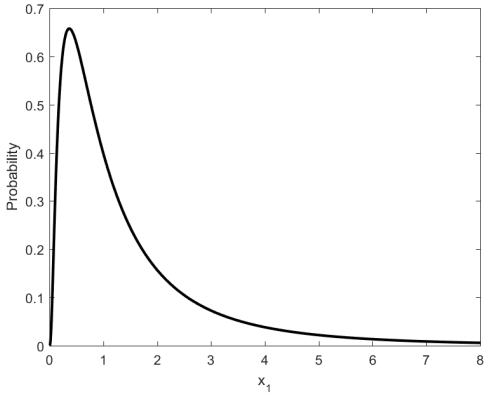
(a) Uniform distribution [-2.5,2.5] and normal Gaussian distribution $\mu = 0$ and $\sigma = 1$



(b) Bi-variate normal distribution for two independent marginal distributions $\mu = 0$ and $\sigma = 1$



(c) Normal distribution $\mu = 0$ and $\sigma = 1$, truncated at [-1.5,1.5]



(d) Log-normal distribution $\mu = 0$ and $\sigma = 1$

Figure A.17: Popular distributions in engineering design

When a 'standard' distribution function cannot be fitted with reasonable confidence, section A.8.3, a Kernel density estimator can be used to fit a sample. The Kernel function is given by

$$\hat{P}(x) = \frac{1}{nh} \sum_{i=1}^{n} K_s \left(\frac{x - x_i}{h} \right), \tag{A.35}$$

where $x_1, x_2, ..., x_n$ are a set of n random samples, $K_s()$ is a smoothing function, and h is the *bandwidth*. Usually, a normal kernel smoothing function is used, along with a theoretical optimal value for bandwidth (the MATLAB fitting function uses the optimal as defined by Bowman and Azzalini [17]). An example of such a fitted distribution is shown in figure A.18.

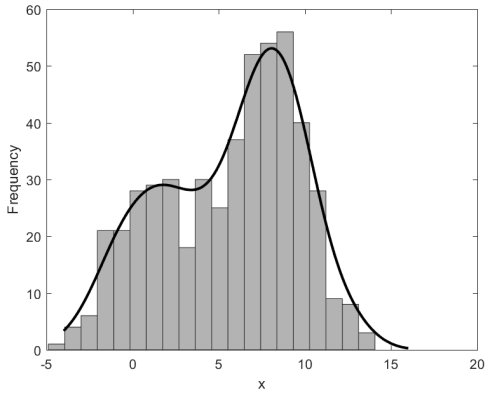


Figure A.18: Kernel function fitted to an example sample

A.8.2 Stable distributions

The stable distribution, applied in chapter 6, can be useful when data sets exhibit *heavy tails* or *skewness*.

A random variable (X) is said to be stable (previously *quasi-stable*) if, for the sum of two independent copies of it, the following holds true:

$$a_1 X_1 + a_2 X_2 \stackrel{c}{=} bX + c \tag{A.36}$$

where a_1, a_2, and b are positive constants, and c is a constant where $c \in \mathbb{R}$. $\stackrel{c}{=}$ means that the shape of the distribution of each is equal, but that it may be scaled or shifted. It follows that, for n_r iid stable random variables $(X_1, X_2, ..., X_{n_r})$

$$X_1 + X_2 + ... + X_{n_r} \stackrel{c}{=} b_{n_r} X + c_{n_r}. \tag{A.37}$$

Similarly, if there exists a positive constant b and real constant c where

$$X \stackrel{c}{=} bY + c, \tag{A.38}$$

then two random variables X and Y are of the same type.

It is possible to show that Gaussian, Cauchy, and Lévy distributions meet this criteria using the closed-form expressions of density. Other than the reflection of the Lévy distribution, there are no known other stable distributions with closed forms for their densities [187]. Therefore, to define the stable distribution, the characteristic function ϕ (the Fourier transform of the probability density function $P(x)$) is used. Given a random

variable X, its characteristic function is defined as a *complex* function of t:

$$\begin{aligned}\phi_x(t) &= \mathscr{F}_x[\mathrm{P}(x)](t) \\ &= \mathrm{E}[e^{itx}] \\ &= \int_{-\infty}^{\infty} e^{itx}\mathrm{P}(x)dx,\end{aligned} \tag{A.39}$$

where $\mathscr{F}_x$ is the Fourier transform with parameters $(1, 1)$, and E is the expectation. The process of determining the probability density function from the characteristic function is therefore the inverse Fourier transform

$$\begin{aligned}\mathrm{P}(x) &= \mathscr{F}_t^{-1}[\phi_x(t)](x) \\ &= \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{-itx}\phi_x(t)dt.\end{aligned} \tag{A.40}$$

Given the above, if a random variable, X, is stable, then it can be parameterized by a characteristic function in a number of different ways. Nolan [187] describes the most popular of these, the first of which, S_0 is the "simplest form for the characteristic function that is continuous in all parameters". This is the form used by the MATLAB `fitdist` function utilised in this work. The second form that Nolan [187] describes (S_1) has "nice algebraic properties", but the mode is unbounded and the parameters don't provide intuitive properties that are interesting in most applications. The $\mathrm{S}_0(\alpha, \beta, \gamma, \delta)$ parameterization is described as follows:

X is stable if

$$X \stackrel{c}{=} \begin{cases} \gamma Z + \delta & \alpha \neq 1 \\ \gamma Z + (\delta + \beta\frac{2}{\pi}\gamma\log\gamma) & \alpha = 1 \end{cases} \tag{A.41}$$

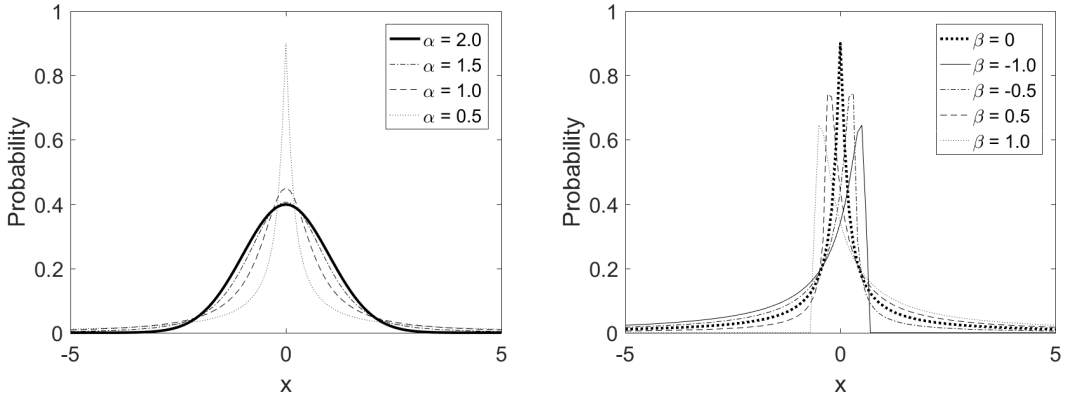
where Z has characteristic function

$$\begin{aligned}\phi_Z(t;\alpha,\beta) &= \mathrm{E}[e^{itz}] \\ &= \begin{cases} \exp\{-|t|^\alpha[1 - i\beta\,\tan\frac{\pi\alpha}{2}\,(\text{sign }t)]\} & \alpha \neq 1 \\ \exp\{-|t|[1 + i\beta\,\frac{2}{\pi}(\text{sign }t)\,\log|t|]\} & \alpha = 1. \end{cases}\end{aligned} \tag{A.42}$$

This leads to the characteristic function for X given by Weisstein [314] and MATLAB [168]:

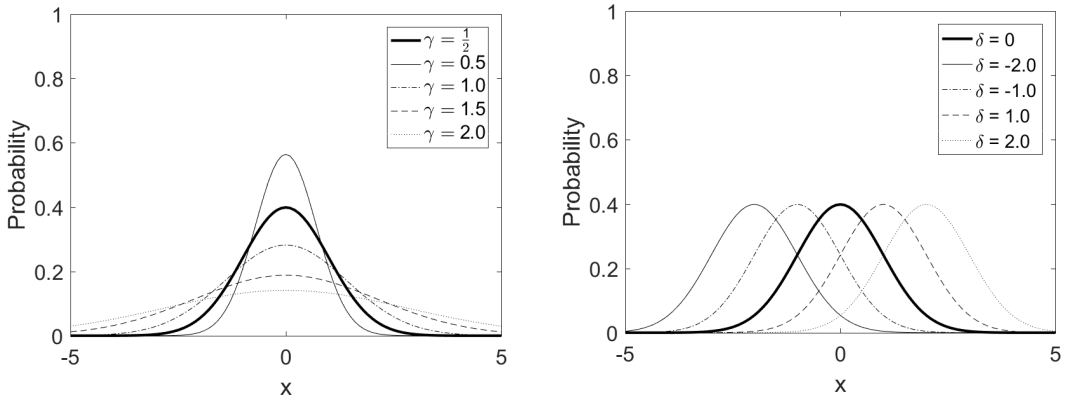
$$\begin{aligned}\phi_X(t;\alpha,\beta,\gamma,\delta) &= \mathrm{E}[e^{itx}] \\ &= \begin{cases} \exp\{i\delta t - \gamma^\alpha|t|^\alpha[1 + i\beta\,\tan\frac{\pi\alpha}{2}\,(\text{sign }t)\,(|\gamma t|^{(1-\alpha)} - 1)]\} & \alpha \neq 1 \\ \exp\{i\delta t - \gamma|t|[1 + i\beta\,\frac{2}{\pi}\,(\text{sign }t)\,(\gamma|t|)]\} & \alpha = 1, \end{cases}\end{aligned} \tag{A.43}$$

and α is a shape parameter defining the tails ($0 < \alpha \leq 2$), β is a second shape parameter describing skewness ($-1 \leq \beta \leq 1$), γ scales the distribution ($0 < \gamma < \infty$), and δ locates



(a) Varying α ($\beta = 0$, $\gamma = \frac{\sigma}{\sqrt{2}}$, $\delta = 0$)

(b) Varying β ($\alpha = 0.5$, $\gamma = \frac{\sigma}{\sqrt{2}}$, $\delta = 0$)



(c) Varying γ ($\alpha = 2$, $\beta = 0$, $\delta = 0$)

(d) Varying δ ($\alpha = 2$, $\beta = 0$,$\gamma = \frac{\sigma}{\sqrt{2}}$)

Figure A.19: The stable distribution with varying parameter values

it ($-\infty < \delta < \infty$).

Examples of the resulting distributions are shown (figures A.19a, A.19b, A.19c, and A.19d). It is noted that the the Gaussian normal distribution is a stable distribution S_0 with $\alpha = 2$, $\beta = 0$, $\gamma = \frac{\sigma}{\sqrt{2}}$, and $\delta = \mu$.

A.8.3 Hypothesis tests

Often referred to as 'goodness-of-fit' tests, there are a plethora of statistical tests for determining the probability that a sample has been drawn from a particular distribution. In particular, the z-test and χ^2-test have been applied during this work to determine whether a sample of a particular variable has a high probability of having been drawn from a normal distribution with either sample mean and sample standard deviation, or sample standard deviation and a mean equal to a nominal design value. Each test uses a test *statistic* to assess a null hypothesis. A *p-value* is calculated: it is "the probability that that a test statistic, at least as significant as the one observed, would be obtained assuming that the null hypothesis were true" [315].

The z-test statistic is

$$z = \frac{\bar{x} - \mu}{\frac{\sigma}{\sqrt{n}}}, \tag{A.44}$$

where $\bar{x}$ is the mean of the n-sized sample, and μ and σ are the mean and standard deviations of the normal distribution being tested. If the null hypothesis is correct then z has a *standard* normal distribution. The χ^2-test statistic is

$$\chi^2 = \sum_{i=1}^{N_b} \left[\frac{(O_i - E_i)^2}{E_i} \right], \tag{A.45}$$

where data are grouped into N_b bins, and O_i are the observed counts and E_i the expected counts is the distribution of the sample is the same as the hypothetical one. It can be used to test any distribution and the test statistic has a χ^2 distribution if there are enough counts.

The Kolmogorov-Smirnov two-sample test statistic:

$$D^* = \max_x(|\hat{F}_1(x) - \hat{F}_2(x)|), \tag{A.46}$$

the absolute difference between the cdfs (F) of the two data sets, is used to test the hypothesis that two samples have been drawn from the same distribution.

In hypothesis tests is it common to select a level of *significance* at which to test the hypothesis. This is the selected p-value below which it is possible to reject the null hypothesis. It sometimes referred to as the α-value, and often designated at 5%, or 0.05.

A.8.4 Correlation and the analysis of variance

The analysis of variance (ANOVA) can be applied to data sets to understand which groups within a population exhibit variation individually, or whether the variance is spread throughout the groups. This can be critical to reducing the number of variables for uncertainty studies and robust design. It does, as with most statistical tools, work most effectively when data sets are large, and works under the assumption that all the variation, ϵ, is normally distributed $\mathrm{N}(0, \sigma^2)$. The method tests the null hypothesis that the groups have equal means, where each group sample $y_{ij} = \mu_j + \epsilon_{ij}$ ($i = 1...n_j$ is the sample from the j^{th} group). By splitting the total sum of squares into a sum of the sum of squares between the groups (SSR) and within the groups (SSE), it is possible to test whether the ratio of the two is significantly high using the *F-statistic*.

As we have already mentioned, ignoring correlation between input variables by assumption of independence can result in misleading propagation of uncertainties to performance prediction. Correlated variables have a covariance which is non-zero and the *statistical*

correlation is

$$\mathrm{cor}(X_1, X_2) = \frac{\mathrm{cov}(X_1, X_2)}{\sigma_{X_1}\sigma_{X_2}}. \tag{A.47}$$

Correlation is, however, often difficult to ascertain. Indeed, relationships between variables can be complex. The simplest means of evaluating correlation is to assume that the relationships are linear. In which case a linear correlation coefficient can be calculated between sets of two sampled variables (x_1 and x_2) as

$$r^2 = \frac{(\sum x_1 x_2 - n\bar{x}_1\bar{x}_2)^2}{(\sum x_1^2 - n\bar{x}_1^2)(\sum x_2^2 - n\bar{x}_2^2)}; \tag{A.48}$$

a function of the sum of the squared residuals using linear regression.

An alternative to the linear coefficient (or *Pearson's correlation*) is the *rank correlation.* There are two commonly used rank correlation measures: the *Kendall rank correlation*, and *Spearman rank correlation.* They are non-parametric measures of the strength of the inter-dependency between the two variables of interest. The Spearman rank correlation coefficient is the most popular, and is defined as

$$\rho' = 1 - 6\sum \frac{{d_s}^2}{n(n^2-1)}, \tag{A.49}$$

where d_s is the difference in *statistical rank* between the corresponding variables, i.e. their numerical position in the sample when ordered in ascending value. n is the number of sample points.

A.8.5 Sampling with correlation

Multi-variate distributions can be constructed with correlation in some cases, where all variables exhibit the same distribution type, or exactly the same distribution, but this is approach is very limited. Instead, copulas can be applied to convert rank correlation to linear correlation, thereby enabling sampling in multiple dimensions, and for any marginal distributions.

Copulas make use of the fact that "applying the normal cumulative distribution function (cdf) ... to a standard normal random variable results in a random variable that is uniform on the interval [0,1]" and "applying the inverse cdf of any distribution, F, to a" uniform random variable on [0,1] "results in a random variable whose distribution is exactly F" [168]. Given these facts, by sampling a multivariate standard normal distribution with prescribed linear correlation (in the leading diagonal of the covariance matrix) it is possible to sample any set of distributions F with the same correlation. For example a bivariate standard normal distribution is sampled with a sample size $n = 1000$, and linear correlation between x_1 and x_2 of $\rho = 0.8$, as illustrated in figure A.20. This

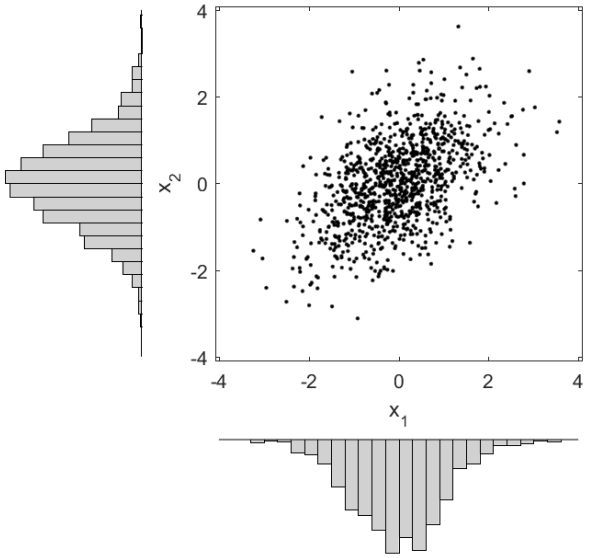


Figure A.20: Bivariate standard normal random sample

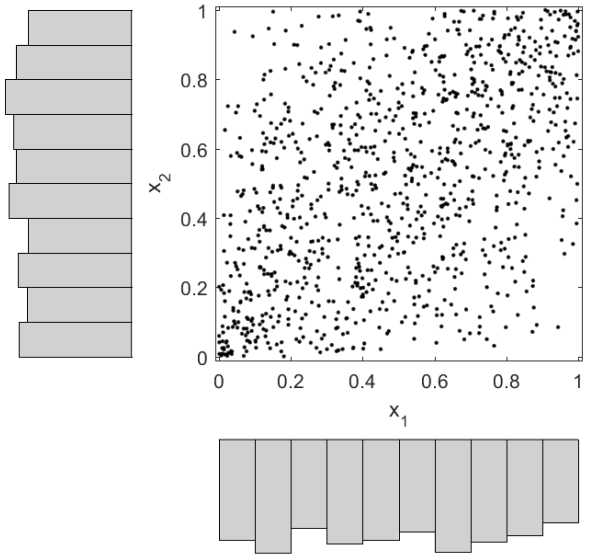


Figure A.21: Bivariate standard normal random sample converted to uniform bivariate sample via normal Gaussian cumulative distribution function

sample can be transformed via the normal cdf, to a bivariate normal sample on [0,1] as shown in figure A.21.

The difficulty here is that the transformation to F is non-linear, so the linear correlation is not, in fact, maintained. If a rank correlation is used, however, the correlation is preserved. The linear correlation coefficient ρ is still needed to parameterize the underlying

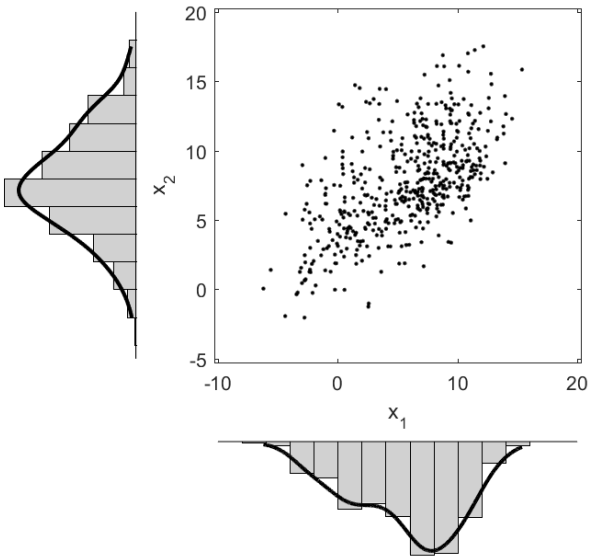


Figure A.22: Bivariate marginal kernel function distributions fitted to random correlated sample $n = 500$

multivariate normal distribution. It is possible to calculate this directly from the rank correlation, the formula being dependent upon the underlying distribution type.

Here, we demonstrate the use of a Gaussian copula to fit two correlated variables. Consider that we have a sample of $n = 500$, looking like the sample in figure A.22. The marginal distributions can be fitted using Kernel density functions as illustrated. The sample can be converted to a sample of the uniform distribution on [0,1], figure A.23, using the cdf of the kernel function. A copula, of chosen form, can then be fitted to this uniform sample, resulting in a value of ρ in the case of the Gaussian copula, and additional parameters if alternative copulas, such as the t-copula are used. A random sample can then be generated from this copula, figure A.24, and converted back to the original marginal densities using the kernel function cdf, figure A.25. The process can be applied in more than two dimensions.

The Iman-Conover method for inducing rank correlation among samples [114] is also widely used for Monte Carlo (MC) sampling. This is equivalent to the copula approach where, as above, a Gaussian copula is used. The procedure can be expressed using matrix algebra, in terms used by Iman and Conover [114], as follows:

The target rank correlation $\mathbf{C}^*$ is the rank correlation observed in a data set. $\mathbf{C}$ is the rank correlation of our MC sample set. $\mathbf{X}_i\mathbf{L}_{\mathrm{IC}}$ is a vector with desired correlation matrix $\mathbf{C}$ and $\mathbf{X}\mathbf{L}'_{\mathrm{IC}} = \mathbf{X}^*$ (a matrix with the same distribution in its rows as $\mathbf{X}_i\mathbf{L}'_{\mathrm{IC}}$), where $\mathbf{L}_{\mathrm{IC}}$ is a lower triangular matrix that results in $\mathbf{L}_{\mathrm{IC}}\mathbf{L}'_{\mathrm{IC}} = \mathbf{C}$, and $\mathbf{X}$ is an $n \times d$ matrix of samples with rows $\mathbf{X}_i$ $(i = 1, ...n)$, if $\mathbf{C}$ is equal to $\mathbf{C}^*$.

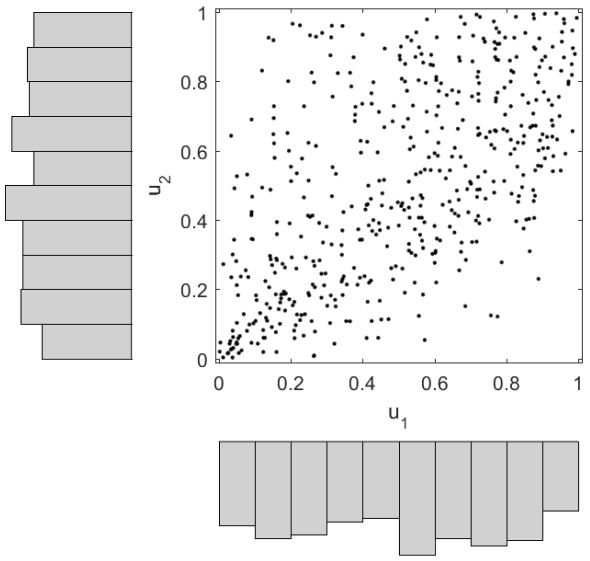


Figure A.23: Bivariate kernel function sample $n = 500$ converted to uniform correlated sample

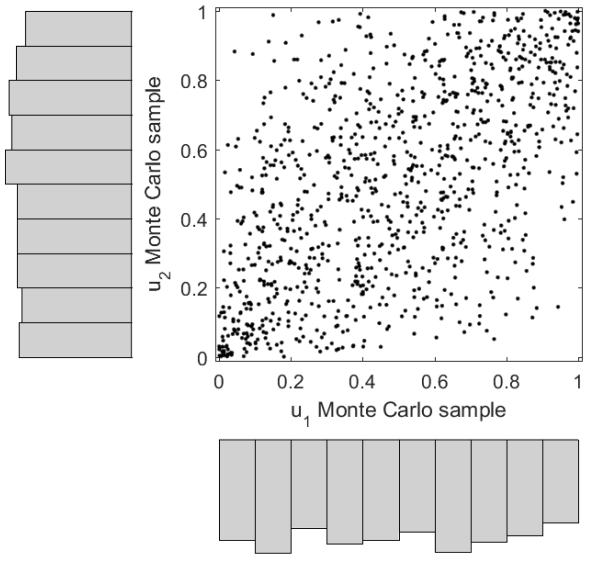


Figure A.24: New sample $n = 1000$ of uniform variables using fitted Gaussian copula

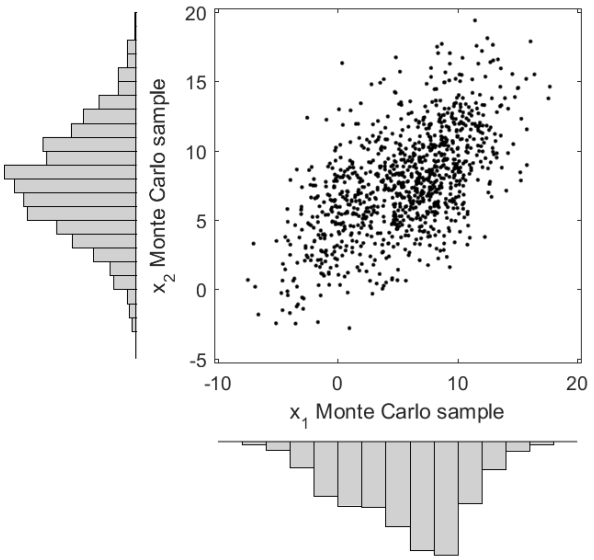


Figure A.25: New uniform sample $n = 1000$ converted back to original marginal densities

$\mathbf{L}_{\mathrm{IC}}$ is calculated by Cholesky factorization. Re-ordering $\mathbf{X}$ as the corresponding column of $\mathbf{X}^*$ results in rank correlation of the input sample set $\mathbf{X}$ that is approximately equal to $\mathbf{C}$.

This method has largely been superseded by the use of copulas in many fields, but its use is still popular due to the simple matrix implementation as described here.

A.9 Principal component analysis

A.9.1 The fundamentals

"Principal components analysis constructs independent new variables which are linear combinations of the original variables" [168]. This summary is constructed using multiple sources: Smith [262], Khan Academy [134], and MATLAB [168].

Let us consider a set of two variables x_1 and x_2. If these two variables are correlated in some way then it stands to reason that the data could be represented by a single variable only. In principal component analysis (PCA) the relationship between the variables is assumed to be a linear one and the original variables are replaced by two alternative *orthogonal* variables; the principal components. In the case where the correlation is deemed to be high enough, one principal component may be considered sufficient to represent the data. To illustrate this, a sample, size $n = 20$, of the population is taken, resulting

in two corresponding vectors $\mathbf{x}_1$ and $\mathbf{x}_2$:

$\mathbf{x}_1$	8.3075	8.7668	8.1482	8.9724	9.0638	8.9385	9.3133	9.6685
$\mathbf{x}_2$	27.2343	27.4585	28.4434	29.2260	29.5978	30.3069	30.8454	31.2393

$\mathbf{x}_1$	10.5157	10.5539	9.9300	11.0070	10.7451	10.7874	11.1429	11.1590
$\mathbf{x}_2$	31.9588	32.3425	33.2777	33.4706	34.0862	34.7381	34.9111	36.3877

$\mathbf{x}_1$	11.3752	11.8979	12.0818	12.2834
$\mathbf{x}_2$	36.7650	37.1490	38.1741	38.1577

Plotted in two dimensions, this sample is displayed in figure A.26a. The following steps are then taken to identify the principal components, figure A.26b.

1. Mean-centre the data and scale as appropriate (if the two variables are different scales or units it is typical to scale, or weight, the variables according to their inverse standard deviations):

$$\mathbf{x}_i = \frac{\mathbf{x}_{i(\mathrm{u})}}{\sigma_i} - \frac{\sum \frac{\mathbf{x}_{i(\mathrm{u})}}{\sigma_i}}{n} \quad i = 1, ..., d \tag{A.50}$$

   where $\mathbf{x}_{i(\mathrm{u})}$ is the unweighted sample vector, d is the number of variables, and $\sigma_{i(\mathrm{u})}$ is the standard deviation of the i^{th} unweighted variable ($d = 2$).

2. Calculate the sample covariance matrix ($\mathbf{V}$) of the weighted and centred data:

$$\begin{aligned} \mathbf{V}_{ij} &= \mathrm{cov}(\mathbf{x}_i, \mathbf{x}_j) \\ &= \langle (\mathbf{x}_i - \mu_i)(\mathbf{x}_j - \mu_j) \rangle \end{aligned} \tag{A.51}$$

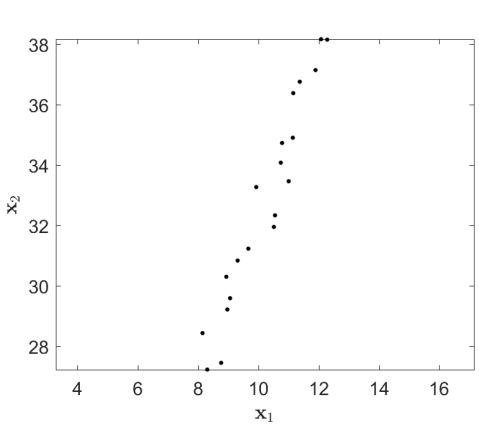
   where $i = 1, ..., d$, $j = 1, ..., d$, and $\langle\rangle$ denotes the mean.

3. Find the two unit eigenvectors of the covariance matrix, and their corresponding eigenvalues, by solving for all eigenvalues λ_i and all eigenvectors $\mathbf{e}_i$ using *eigen decomposition* [312]; a form of *matrix decomposition* of a square matrix also referred to as *matrix diagonalization*, i.e. solve

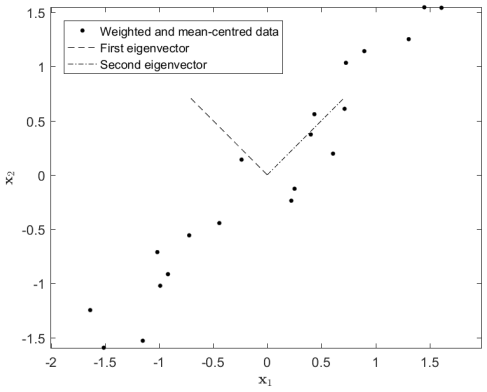
$$\mathbf{VE} = \mathbf{ED}, \tag{A.52}$$

   by factorization [313, 276, 53]:

$$\mathbf{V} = \mathbf{EDE}^{-1}, \tag{A.53}$$



(a) Two-variable sample size $n = 20$



(b) Two-variable sample size $n = 20$, weighted and mean-centred with principal components identified

Figure A.26: Finding the principal components of a two-variable sample

where $\mathbf{D}$ is the diagonal matrix of eigenvalues and $\mathbf{E}$ is the eigenvector matrix. Consider that

$$\mathbf{V}\mathbf{e}_i = \lambda_i \mathbf{e}_i \quad i = 1, ..., d \tag{A.54}$$

and therefore

$$(\mathbf{V} - \lambda_i \mathbf{I})\mathbf{e}_i = \mathbf{0}, \tag{A.55}$$

and, in turn

$$\det(\mathbf{V} - \lambda_i \mathbf{I}) = 0. \tag{A.56}$$

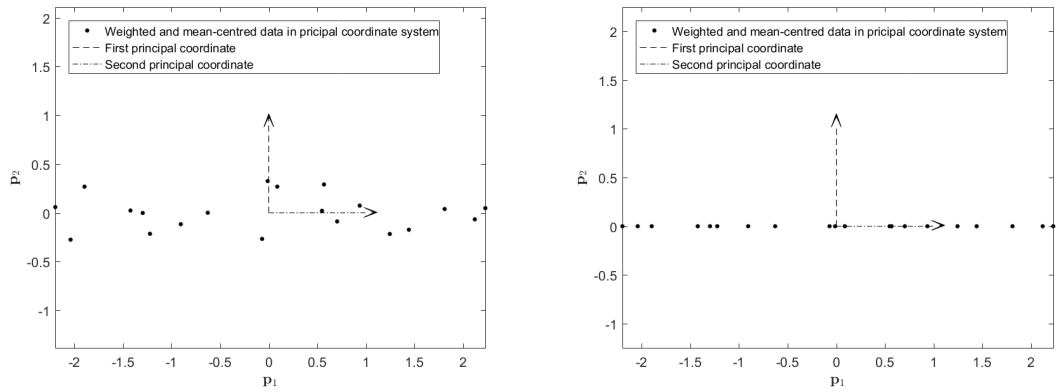
The matrix of eigenvectors ($\mathbf{E} = [\mathbf{e}_1\ \mathbf{e}_2\ ...\ \mathbf{e}_d]$) is also sometimes referred to as the *feature vector* or the matrix of *principal component coefficients*, and *eigenvectors* can also be termed *characteristic vectors*, *proper vectors*, or *latent vectors*[9]. These vectors are the principal component directions, and their eigenvalues (λ_i) give their size, or importance, relative to each other.

The vector with the largest corresponding eigenvalue is the *principal component*. These steps are also applicable to data sets $d > 2$. In this case ordering the eigenvectors according to eigenvalue size allows the most significant components to be identified. The values of each data point in *principal component space* can be calculated using:

$$\mathbf{P}^{\mathrm{T}} = \mathbf{E}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}} \tag{A.57}$$

This is a coordinate transformation from points set in the *standard basis* (Cartesian coordinate system) to the new principal coordinate system. This equation is true for coordinate transformations *iff* $\mathbf{E}$ is square with linearly independent columns (which is true for all $\mathbb{R}$), and is *orthonormal*, i.e. $\mathbf{E}^{-1} = \mathbf{E}^{\mathrm{T}}$ and $\det(\mathbf{E}) = 1$. These conditions

[9] $\mathbf{e}_i$ $(i = 1, ..., d)$ are also described as *loading vectors* [317]



(a) Sample plotted using both identified principal components

(b) Sample plotted using only the most 'important' principal component

Figure A.27: Two-variable sample plotted using principal components

are always true for a matrix of unit eigenvectors. To return to the original *standard* coordinate space:

$$\mathbf{X} = \mathbf{P}\mathbf{E}^{\mathrm{T}} \tag{A.58}$$

$\mathbf{P}$ (the original points transformed into the principal component axes)[10] are plotted (figure A.27a).

In this case, one might suggest that the majority of the data could be represented using only one variable (the principal component, $\mathbf{p}_1$), plotted with the second principal component values set to zero in figure A.27b. In fact, the components can be ranked in order of relative 'importance' by ordering them according the their eigenvalues (this 'importance' is really a value of the "variance that is 'explained' by the corresponding principal component" [168]). It is common practice to determine the percentage of the observed variability that is explained by each component, which is the percentage of the summed variances (eigenvalues). In this two-variable example, the first principal component explains $> 98.3\%$ of the variance. By removing the components that have an insignificant influence on the variance the number of dimensions can be reduced.

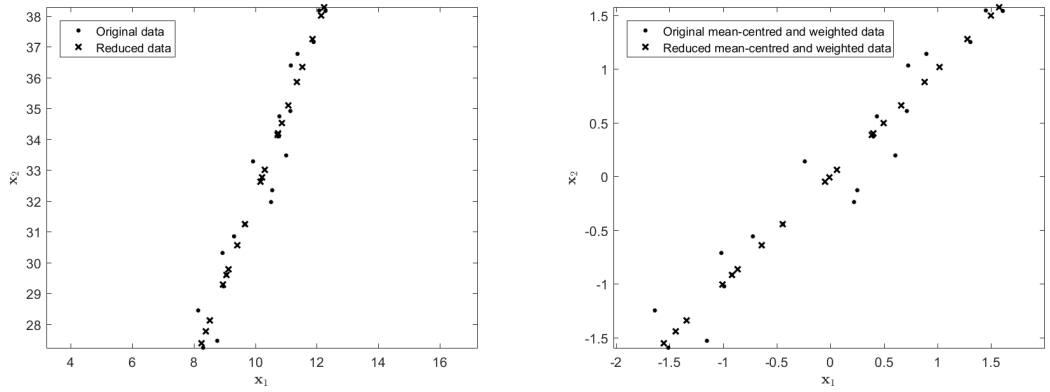
Once the significant components have been identified, the 'new' data points in the reduced space $\mathbf{P}_{\mathrm{r}}$, with some component values set to zero, can be transformed back into the standard coordinate space (figure A.28a):

$$\mathbf{X}_{\mathrm{r}} = \mathbf{P}_{\mathrm{r}}\mathbf{E}^{\mathrm{T}} \tag{A.59}$$

When mean-centring and weighting is removed, the transformed data looks like figure A.28b. This is a form of linear or least squares regression, where the formulation of the least squares model is:

$$\mathbf{X}_{\mathrm{r}} = \mathbf{1}\bar{\mathbf{x}} + \mathbf{P}_{\mathrm{r}}\mathbf{E}^{\mathrm{T}} + \mathbf{R} \tag{A.60}$$

[10] $\mathbf{P}$ is a matrix of *score vectors*, $\mathbf{p}_i$ $(i = 1, ..., d)$ [317]



(a) Weighted and mean-centred sample (original and reduced dimensions)

(b) Unweighted sample (original and reduced dimensions)

Figure A.28: Two-variable sample defined by a single principal component compared with the original data set, plotted in the standard basis

where $\mathbf{R}$ is a matrix of residuals (deviations, in each dimension, of the original points from their new (regressed) locations). $\bar{\mathbf{x}}$ is the mean vector, which was previously removed by mean-centring.

A.9.2 The role of singular value decomposition (SVD)

As is shown in section A.9.1, PCA relies on the fundamental relation in linear algebra:

$$\mathbf{A} = \mathbf{EDE}^{-1} \tag{A.61}$$

where $\mathbf{A}$ is any symmetric matrix with N independent eigenvectors. This is a decomposition of the matrix $\mathbf{A}$ into two component matrices (a diagonal matrix of eigenvalues ($\mathbf{D}$ with diagonal values λ_i) and a matrix of eigenvectors ($\mathbf{E} = [\mathbf{e}_1\mathbf{e}_2...\mathbf{e}_i]$); eigen decomposition.

Singular value decomposition (SVD) is a splitting of any $(j \times k)$ matrix into three component matrices:

$$\mathbf{A} = \mathbf{J\Sigma K}^{\mathrm{T}}. \tag{A.62}$$

$\mathbf{J}$ $(j \times j)$ and $\mathbf{K}^{\mathrm{T}}$ $(k \times k)$ are orthogonal matrices of *singular vectors* (*left* and *right*), and $\mathbf{\Sigma}$ is a diagonal matrix of *singular values*, Σ_i. Although these two decompositions appear to be similar, they are not identical unless $\mathbf{A}$ is symmetric or complex Hermitian (a case not discussed here). The reason is that $\mathbf{E}$ is clearly orthogonal to $\mathbf{E}^{-1}$ but $\mathbf{J}$ and $\mathbf{K}^{\mathrm{T}}$ are not orthogonal to each other [216].

To see how SVD can be applied to PCA, we multiply this second relation by the transpose of itself, and considering that the matrix $\mathbf{A}$ is, in our case, $\mathbf{X}$:

$$\mathbf{X}^{\mathrm{T}}\mathbf{X} = (\mathbf{K}\mathbf{\Sigma}^{\mathrm{T}}\mathbf{J}^{\mathrm{T}})(\mathbf{J}\mathbf{\Sigma}\mathbf{K}^{\mathrm{T}}) = \mathbf{K}(\mathbf{\Sigma}^{\mathrm{T}}\mathbf{\Sigma})\mathbf{K}^{\mathrm{T}}. \tag{A.63}$$

As such, the matrix $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ (a positive semi-definite symmetric matrix) has orthogonal eigenvectors $\mathbf{K}$ ($\mathbf{K}^{\mathrm{T}} = \mathbf{K}^{-1}$) and positive eigenvalues Σ_i^2 in $\mathbf{\Sigma}^{\mathrm{T}}\mathbf{\Sigma}$. Similarly, $\mathbf{X}\mathbf{X}^{\mathrm{T}}$ has the same eigenvalues as $\mathbf{X}^{\mathrm{T}}\mathbf{X}$, and has orthogonal eigenvectors $\mathbf{J}$.

In particular, if the matrix $\mathbf{X}$ is *standardized* (mean-centred and each variable set divided by its sample standard deviation as in our approach to PCA in section A.9.1), then the *cross-product* matrix $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ becomes the covariance matrix (not normalized[11]) [180]. $\mathbf{\Sigma}^{\mathrm{T}}\mathbf{\Sigma} = \mathbf{\Sigma}^2$ is equal to the eigenvalue matrix ($\mathbf{D}$) in the eigen decomposition of $\mathbf{V}$ ($\lambda_i = \Sigma_i^2$ and $\mathbf{E} = \mathbf{K}$).

The *most important* (with greatest variance) singular vectors $\mathbf{K}$ (components) are those with the biggest singular value (Σ). These correspond (as we have seen above) to the eigenvectors $\mathbf{K}$ and biggest eigenvalue λ.

Typically, iterative algorithms are used by mathematical software for the factorization needed for PCA (through eigen-decompostion or SVD): usually variations on *QR factorization* [141, 180, 143]. The eigen decomposition functions in MATLAB use either Cholesky factorization [141] or the QZ (generalized Schur decomposition) algorithm [93], depending on the structure of the matrix in question. The choice to use SVD is really one of expense when dealing with large data sets; it can be expensive to compute the covariance matrix or $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ but there are techniques such as the use of Householder reflections [141, 143] that are embedded in SVD code to expedite the process. There is also argument over the stability of some algorithms over others, although which is likely to be most robust will be problem size and structure specific.

A.9.3 Variants and alternatives

There are a number of different names for the mathematical methodology described above, including *proper orthogonal decomposition* (POD) and *Karhunen-Loève decomposition*. The original concept can be traced back to Pearson FRS [203] [210, 268] and the basis for the method shown in section A.9.1 can be attributed to Hotelling [107], as Jolliffe [123] describes. Interest in PCA has recently increased due to its possible application to dimensionality reduction[12]; a "hot topic" due to the "explosion" [268] of data

[11] Normalizing $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ using $(k-1)$ will yield precisely the same matrix $\mathbf{V}$ in section A.9.1 (usually considered to be an unbiased and *efficient* estimator, but by some measures shown to be biased and *inefficient* [263]).

[12] The general idea if dimensionality reduction is to assume that a high-dimensional data set has an *intrinsic* dimensionality [296] much smaller than its original (measured) dimensionality, and to find this intrinsic set with loss of the smallest possible amount of information.

that has been able to be gathered and stored computationally over recent decades. This includes, but is not limited to: scans and images, laboratory measurements, 'cloud' data about internet usage, mobile phone data, meteorological data, etc., across a number of disciplines and industries, and making practical use of this data difficult given its 'size'. It is not, therefore, surprising that many similar techniques have evolved. Here we review those that are most applicable to our own geometric measurement problem, which is not dissimilar to an image processing problem.

Mathematically, the dimension reduction problem can be considered in one of two possible ways: *feature reduction* (the selection of a subset of the most relevant variables from a large original set), and dimension reduction (finding a reduced size set of alternative variable to best represent the original).

The subject of dimension reduction is extensive, and developing at a significant rate. In the world of computer science it is part of the subject known as *data mining*, which encompasses all of the numerical methods designed to interrogate data sets and enable statistically grounded conclusions to be drawn about them. Data mining is really the data analytics that forms the basis of decision-making, and when the decision-making processes itself is automated the methods come under the terms *artificial intelligence* or *machine learning*. Here, we concern ourself with the data analytics only, although one could consider the process of drawing new 'pseudo' curves or surfaces based on this to be machine learning of a sort, particularly if new data was routinely added to the measurement (training) set.

In 2009 van der Maaten et al. [296] produced a document to accompany a dimension reduction toolbox. Within this document he provides a taxonomy of dimensionality reduction techniques and describes and tests a number of *non-linear dimensionality reduction* techniques that aim to "address the limitations of traditional techniques" (PCA (*classical scaling*)). The non-linear methods tested are:

- *kernel PCA*,
- *Isomap*,
- *maximum variance unfolding*,
- *diffusion maps*,
- *local linear embedding* (LLE),
- *Laplacian eigenmaps*,
- *Hessian LLE*,
- *local tangent space analysis*,
- *Sammon mapping*,

- *multilayer autoencoders*,
- *locally linear coordination*, and
- *manifold charting*.

van der Maaten et al. [296] also list some other approaches that were not included in his review, with reasons for their absence:

- *self-organizing maps* and *generative topographic mapping* (GTM), which combine dimension reduction with clustering,
- *independent component analysis* (ICA), designed for "blind" source separation, and
- *supervised* approaches including *discriminant analysis* and *neighbourhood components analysis*.

Further still, it should be noted that there are also many extensions of the non-linear approaches that van der Maaten et al. [296] compares, as well as *latent* techniques that are data-type specific.

Factor analysis is "closely related to PCA" [317] and was also described by Hotelling [107] in his early work. One should be careful when researching the literature as "often the two methods are confused and the two names are incorrectly used interchangeably" [317].

Correspondence analysis "is a special double-scaled variant of PCA that is much favoured in French-speaking countries and Canada and in some scientific fields" [317].

A.9.4 Non-linear PCA

An extension to standard (linear) PCA is that of *non-linear* or *categorical PCA*, which address two, perhaps obvious, limitations of the original approach:

1. it is unable to discover or deal with non-linear relationships between variables, and
2. it does not have a built-in mechanism by which non-quantitative (catagorical) variables can be considered.

Linting et al. [154] provide a detailed description of the method applied to data collected about childcare environments for six-month old children. It may not be immediately obvious but the approach to dealing with both of these limitations is the same; in order to effectively remove the non-linear relationships and convert non-numeric categories to numeric measurement levels non-linear PCA uses a non-linear transform. During

this process, the discrete categories and other variables are transformed into *category quantifications*. The process is referred to as *optimal quantification* (also *optimal scaling* or *optimal scoring*). These 'new' variables then exhibit variance and a linear relationship with each other, as is true of their quantitative continuous numeric counterparts when linear PCA is applicable.

The PCA approach has, thus far, been formulated as an eigenvalue problem solved using either an eigenvalue decomposition of the correlation matrix (section A.9.1), or a singular value decomposition (SVD) of the standardized data matrix (section A.9.2). In order to compare linear PCA with non-linear PCA we express it here as a least-squares regression problem where the goal is to find the loading vectors ($\mathbf{e}_j$, $j = 1, ...N$), such that the variance of each derived variable (component) is maximized or the sum of the squared Euclidean distances is minimised. Without decomposition this process is an iterative optimization procedure. The first step is to find only the first principal axis ($\mathbf{e}_1$), constrained to be a unit vector, that maximises the variance (where $\boldsymbol{\mu} = \mathbf{0}$; the data are mean-centred):

$$\begin{aligned} \mathrm{var}(\mathbf{X}\mathbf{e}_1) = ||\mathbf{X}\mathbf{e}_1||^2 &= \frac{\mathbf{e}_1^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{e}_1}{n-1} \\ &= \mathbf{e}_1^{\mathrm{T}}\mathbf{V}\mathbf{e}_1 \end{aligned} \tag{A.64}$$

or, that minimises the *construction error* (sum of the squared Euclidean distances) from the points (X) to the first principal component ($\mathbf{X}\mathbf{e}_1\mathbf{e}_1^{\mathrm{T}}$):

$$||\mathbf{X} - \mathbf{X}\mathbf{e}_1\mathbf{e}_1^{\mathrm{T}}||^2 = \mathrm{tr}((\mathbf{X} - \mathbf{X}\mathbf{e}_1\mathbf{e}_1^{\mathrm{T}})(\mathbf{X} - \mathbf{X}\mathbf{e}_1\mathbf{e}_1^{\mathrm{T}})^{\mathrm{T}}), \tag{A.65}$$

where tr is the *trace* (sum of the diagonal elements). Multiplying through, substituting $\mathbf{e}_1^{\mathrm{T}}\mathrm{e}_1 = 1$ and $\mathrm{V} = \frac{\mathbf{X}^{\mathrm{T}}\mathbf{X}}{n-1}$, and observing that $\mathrm{tr}(\mathbf{AB}) = \mathrm{tr}(\mathbf{BA})$, it is possible to see demonstrate that the two objectives are equivalent:

$$\begin{aligned} ||\mathbf{X} - \mathbf{X}\mathbf{e}_1\mathbf{e}_1^{\mathrm{T}}||^2 &= \mathrm{tr}((\mathbf{X} - \mathbf{X}\mathbf{e}_1\mathbf{e}_1^{\mathrm{T}})(\mathbf{X}^{\mathrm{T}} - \mathbf{e}_1\mathbf{e}_1^{\mathrm{T}}\mathbf{X}^{\mathrm{T}})) \\ &= \mathrm{tr}(\mathbf{X}\mathbf{X}^{\mathrm{T}}) - 2\mathrm{tr}(\mathbf{X}\mathbf{e}_1\mathbf{e}_1^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}) + \mathrm{tr}(\mathbf{X}\mathbf{e}_1(\mathbf{e}_1^{\mathrm{T}}\mathbf{e}_1)\mathbf{e}_1^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}) \\ &= c_1 - \mathrm{tr}(\mathbf{X}\mathbf{e}_1\mathbf{e}_1^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}) \\ &= c_1 - \mathrm{tr}(\mathbf{e}_1^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{e}_1) \\ &= c_1 - c_2(\mathbf{e}_1^{\mathrm{T}}\mathbf{V}\mathbf{e}_1), \end{aligned} \tag{A.66}$$

where c_1 and c_2 are constants.

To find the remaining (k^{th}) components (subject to each component being uncorrelated with the remaining components), the $k-1$ components that have already been found must be subtracted from $\mathbf{X}$ giving:

$$\hat{\mathbf{X}}_k = \mathbf{X} - \sum_{i=1}^{k-1}(\mathbf{X}\mathbf{e}_i\mathbf{e}_i^{\mathrm{T}}). \tag{A.67}$$

$\mathbf{e}_k$ can then be found either by maximising

$$||\hat{\mathbf{X}}_k \mathbf{e}_k||^2$$

or by minimising

$$||\hat{\mathbf{X}}_k - \hat{\mathbf{X}}_k \mathbf{e}_k \mathbf{e}_k^{\mathrm{T}}||^2,$$

and continuing iteratively by decreasing from $k = N$ to $k = 1$, all of the independent principal components can be found.

In non-linear PCA $\mathbf{X}$ is replaced with a matrix $\mathbf{Q}$, which contains a set of transformed variables ($\mathbf{q}_j = \phi_j(\mathbf{x}_j)$), for $j = 1, 2, ..., N$). The goal is still to minimise the squares of the orthogonal distances between the original data and the principal components. The equation for the total of all these distances is termed the *loss function* (f_L). f_L can be expressed in terms of component scores ($\mathbf{P}$), recalling that $\mathbf{P} = (\mathbf{E}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}})^{\mathrm{T}} = \mathbf{XE}$ [13]:

$$f_L(\mathbf{Q}, \mathbf{P}, \mathbf{E}) = \frac{1}{n} \sum_{j=1}^{N} \mathrm{tr}((\mathbf{q}_j - \mathbf{P}\mathbf{e}_j)^{\mathrm{T}}(\mathbf{q}_j - \mathbf{P}\mathbf{e}_j)). \tag{A.68}$$

Gifi [90] shows that the minimization of f_L is equivalent to the minimization of $_{L2}$ [154], where

$$f_{L2}(\mathbf{Q}, \mathbf{P}, \mathbf{E}) = \frac{1}{n} \sum_{j=1}^{N} \mathrm{tr}((\mathbf{q}_j \mathbf{e}_j^{\mathrm{T}} - \mathbf{P})^{\mathrm{T}}(\mathbf{q}_j \mathbf{e}_j^{\mathrm{T}} - \mathbf{P})). \tag{A.69}$$

In these loss functions $\mathbf{Q}$ is subject to the same constraints as $\mathbf{X}$, i.e. the variables are standardized, and the scores ($\mathbf{P}$) have zero mean, standard deviation of one, and are uncorrelated. The minimization is computed using *alternating least squares* (ALS), where $\mathbf{Q}$, $\mathbf{P}$, and $\mathbf{E}$ are updated alternately whilst holding the other two fixed.

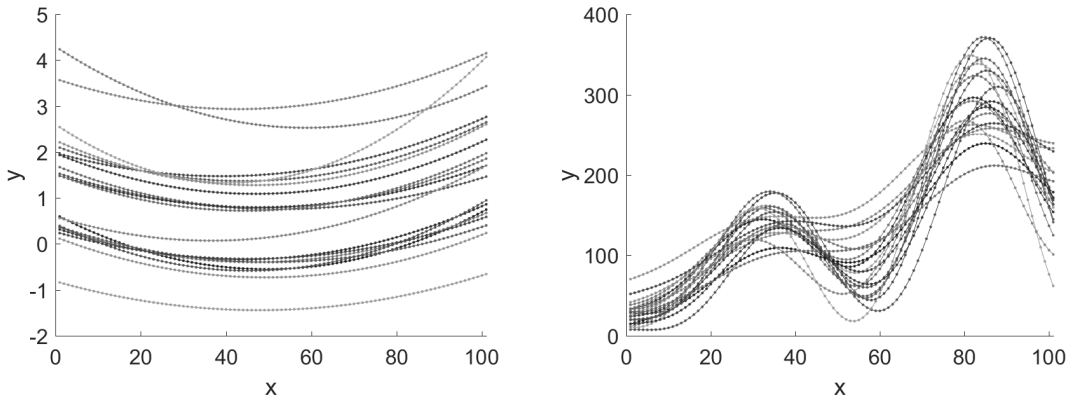
A.9.5 Updating a PCA estimate

If additional data are available, or likely to be made available regularly then there are a number of so-called *incremental*, *stream*, or *online* methods of PCA that can be used to efficiently update previous estimates without re-calculating the covariance matrix and eigenvalue decomposition [269].

A.9.6 Demonstration of PCA using curves defined by known random variables

In order to investigate the properties of the principal components and their distributions that are derived from curves that are created through natural variability in a set

[13] The j^{th} column of $\mathbf{P}$ is equal to $\mathbf{X}\mathbf{e}_j$



(a) Quadratic curves defined using 3 random variables

(b) Sine-based curves defined using 4 random variables

Figure A.29: Pre-defined curves with controlling random variables (example set of 20 randomly selected)

of unknown 'controlling variables', sets of curve families are constructed using known normally distributed random variables. The sets used are:

1. quadratic curves

$$y = (\tfrac{x-c_1}{c_2})^2 + c_3 \quad \begin{pmatrix} c_1 \sim \mathrm{N}(45,5) \\ c_2 \sim \mathrm{N}(51,5) \\ c_3 \sim \mathrm{N}(0.25,0.75) \end{pmatrix}, \tag{A.70}$$

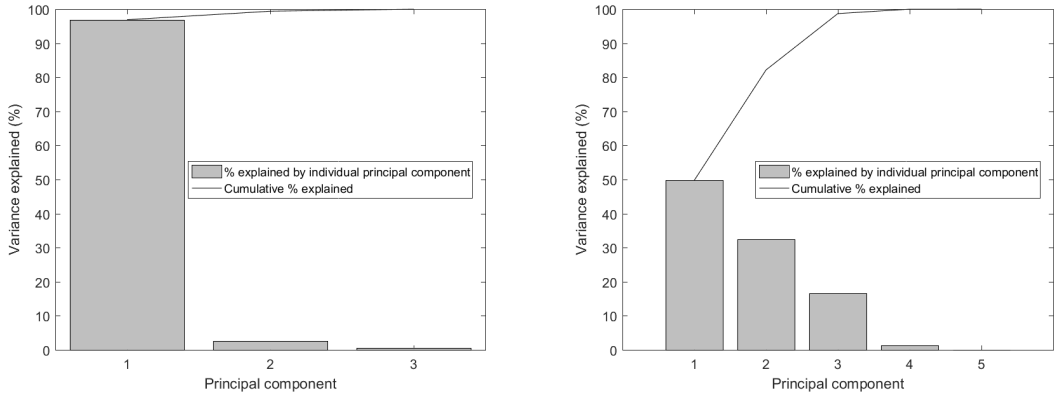
and

2. a 'lumpy' function

$$y = c_3 - c_1(x+c_4)^2 \sin(c(x+c_4)) \quad \begin{pmatrix} c_1 \sim \mathrm{N}(1,0.5) \\ c_2 \sim \mathrm{N}(2,0.1) \\ c_3 \sim \mathrm{N}(10,10) \\ c_4 \sim \mathrm{N}(10,1.5) \end{pmatrix}, \tag{A.71}$$

where $c_1, c_2, ..., c_k$ are the k 'controlling' constants; the normal random variables that control the shapes of the curves of the same family. c is a constant which is fixed at $c = 0.12$ and is a 'frequency' in the sine wave used to construct the 'lumpy' curve. Example sets of 20 randomly drawn curves from each of these curve families are shown in figure A.30.

The curves are defined by 101 points along them, i.e. x (in Cartesian coordinate space) $= 1, 2, ..., 101$. These values remain fixed. The variables for PCA are their corresponding y-values, which make up $\mathbf{X} = [\mathbf{x}_1\ \mathbf{x}_2\ ...\ \mathbf{x}_{101}]$ ($\mathbf{x} = [y_1\ y_2\ ...\ y_n]^{\mathrm{T}}$). n is the sample size (number of measurement data sets). It is important to note that although the controlling constants are normally distributed, the y-values are not multi-variate normal. Therefore,



(a) Quadratic-type (equation A.70) (b) Sine-type (equation A.71)

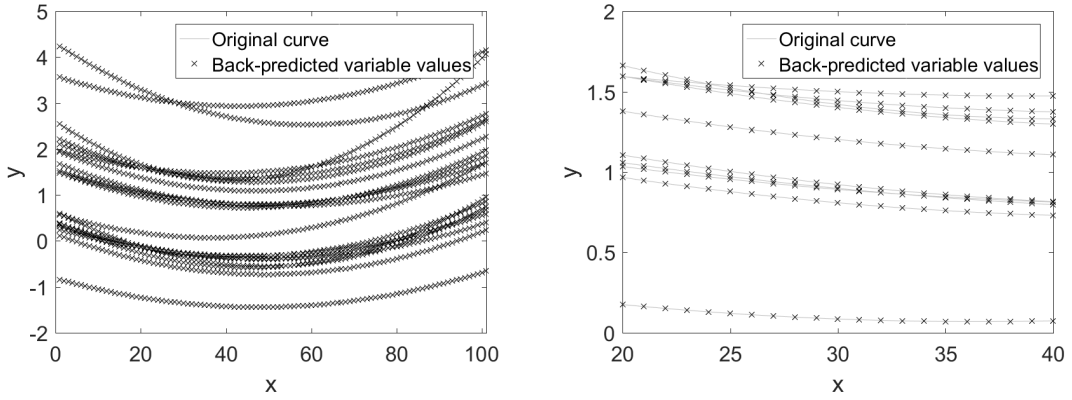
Figure A.30: Explained variance of the first principal components found for 10,000 randomly drawn curves

any inference as to the distributions of the underlying PCs is not possible [123]. In the unlikely case that they are, then it might be possible to use the "sampling properties of eigenvectors and eigenvalues of *Wishart random variables*" to sample the resulting PCs [123].

If PCA is conducted on 10,000 randomly drawn curves of the quadratic type that has been defined in equation A.70, then it is found that 96.9% of the variance is explained by just one component, but that 100% is explained using three components (figure A.30a). This can be shown by using the principal components to 'back-predict' the original variable values for the same set of 20 curves in figure A.31. The curves are identical. This is perhaps not surprising given that the curves are pre-defined using three random variables. By plotting histograms of the three variables (c_1, c_2, and c_3) (figure A.32) next to the histograms of the corresponding principal components (figure A.33) it is clear that these are not all also normally distributed (a Gaussian normal cannot be justified using a Chi-squared goodness of fit test at the 5% level; the null hypothesis that the sample is drawn from a normal distribution with sample mean and standard deviation can be rejected). In order to sample from them for uncertainty propagation it must be possible to define a distribution to sample from.

We attempt to fit the distributions of the principal components using stable distributions; "a class of probability distributions" that, although ... first characterized by Paul Lévy ... in the 1920's" [187], were not able to be widely and reliably computed until very recently. These distributions have justifiable application when either:

- the physical system gives solid theoretical grounds to expect non-Gaussian stable distributions (examples include the gravitational field of stars, and Brownian motion hitting times ([68, 294] referred to by Nolan [187])),



(a) Original scale plot (b) Zoomed in for clarity

Figure A.31: 'Back-predicted' quadratic-type curves, equation A.70

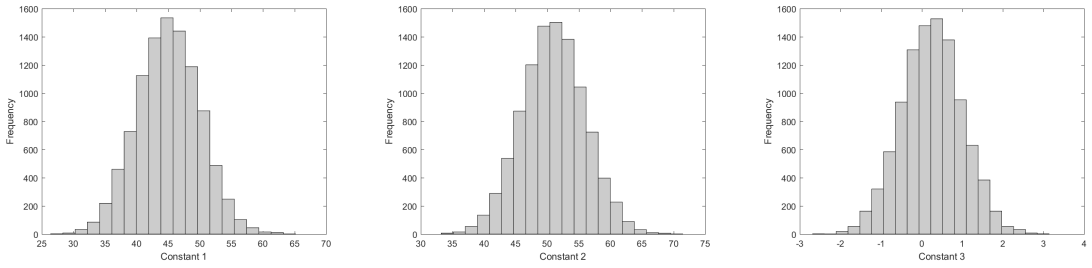


Figure A.32: Histograms of input random variables (c_1, c_2, and c_3)

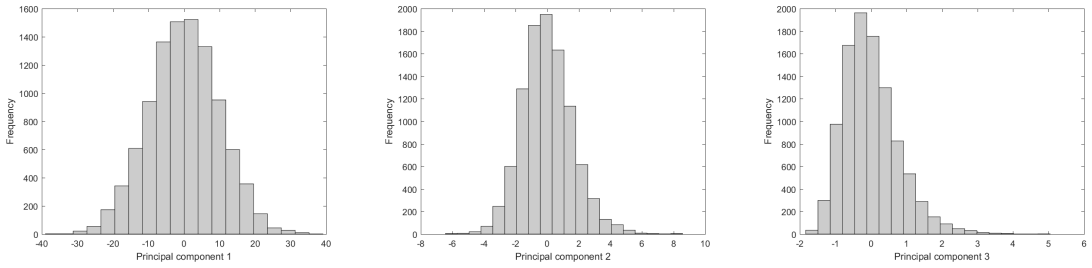


Figure A.33: Histograms of output principal components

- quantities that are expected to be derived from sums of many smaller terms (e.g. prices of stock), or
- the data sets exhibit heavy tails or skewness where "empirical evidence combined with central limit theorem" [187] is used as justification.

A detailed definition of the stable distribution is given in section A.8.2. The three principal component distributions (figure A.34) appear to be best fitted with a stable distribution, when observing the output of the 10,000 point sample. However, a Chi-squared goodness of fit test results in a rejection of the null hypothesis in the case of the third principal component. The p-values for each of the tests are given in table A.1.

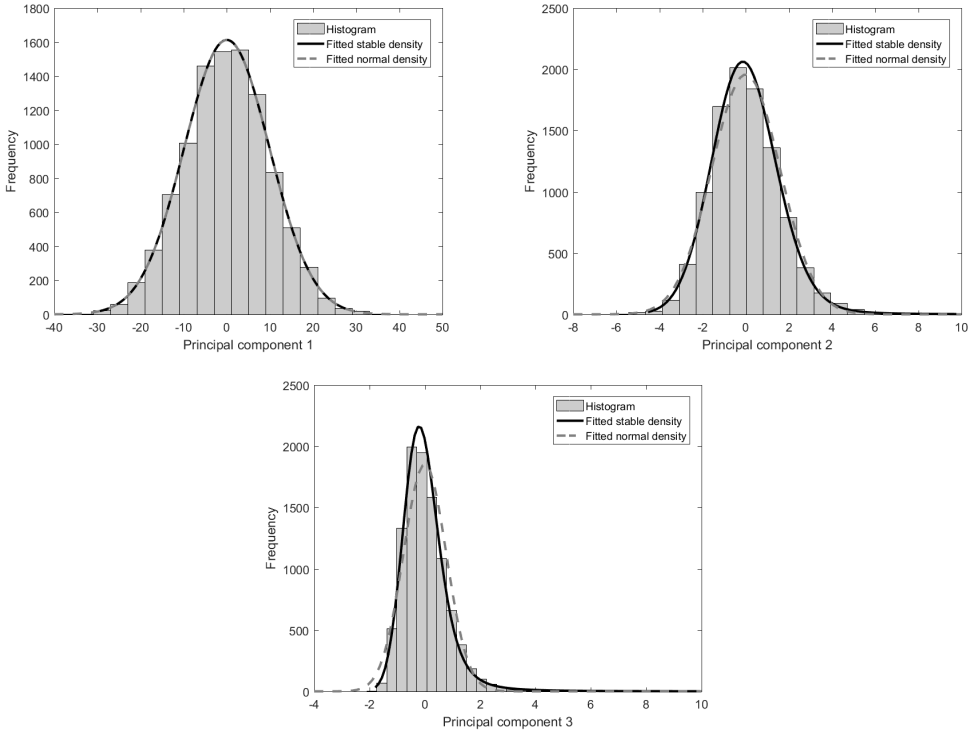


Figure A.34: Histograms of output principal components with fitted distributions (quadratic function of equation A.70)

	PC 1	PC 2	PC 3
Normal distribution χ^2 test	0.3922	0.0851	0.0000
Stable distribution χ^2 test	0.2523	0.2746	0.0099

Table A.1: p-values for χ^2 goodness-of-fit tests on principal component (PC) values when reconstructing a pre-defined curve with three normally distributed input variables (sample size $n = 10,000$)

Although the stable density for PC 1 is extremely close to the Gaussian distribution, the goodness-of-fit test returns a significantly worse p-value. Where the distribution is not so close to normal (PCs 2 and 3), the p-values show, as expected, that the stable distribution is a 'better' fit, i.e. the probability that the sample distribution is observed, given that the fitted distribution is correct, is higher for the stable distribution. Unfortunately, although this is true for the third PC, the p-value is not high enough to rule out rejecting the null hypothesis (H_0: PC 3 is distributed with a stable distribution with parameters determined by the MATLAB `fitdist` function).

Repeating the same experiment with sine-type curve results in the variance percentages explained in figure A.30b and the distributions of the first four principal components as shown in figure A.35 (the 5^{th} PC explains less than 0.05% of the variance so is not included. In this case, there is little or no skewness but the 3^{rd} PC appears to be

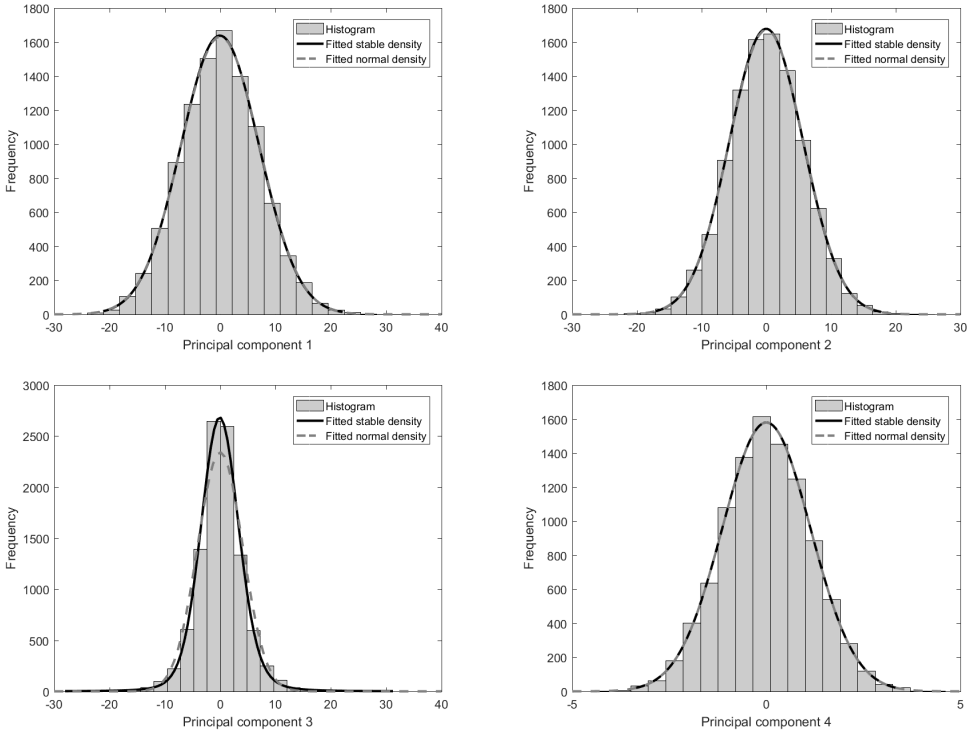


Figure A.35: Histograms of output principal components with fitted distributions ('lumpy function' of equation A.71)

described more accurately by the stable distribution because it is able to represent the more peaky mean region. The p-values for the chi-squared tests are given in table A.2.

	PC 1	PC 2	PC 3	PC 4
Normal distribution χ^2 test	0.0450	0.1500	0.0000	0.8020
Stable distribution χ^2 test	0.1798	0.2509	0.0000	0.5779

Table A.2: p-values for χ^2 goodness-of-fit tests on principal component (PC) values when reconstructing a pre-defined curve with three normally distributed input variables (sample size $n = 10,000$)

Clearly, these tests cannot lead to the conclusion that the stable distribution is always a more suitable fit for principal components than the normal distribution. It is known however, that standard goodness-of-fit tests are not always appropriate for stable distributions. Beaulieu et al. [10] use Monte Carlo based extensions to other methods in order to ensure that the tail behaviour is better captured. In the work of Nolan [186] a description of the difficulties associated with assessing stable distribution goodness-of-fit is given, but it is stated that "it is not surprising that one can fit a data set better with the 4 parameter stable model than with the 2 parameter normal model". In this thesis we stop short of delving into the intricacies of such tests. Instead, we can clearly observe from the above that *this set of curves, controlled by a set of normally distributed random variables, can be represented by a set of independent principal components that can*

reasonably be represented using stable distributions. If large samples such as these were available then a fitted Kernel function may be more appropriate but, as is often the case, one must make assumptions about the distributions where data sets are small.

An additional example curve has been considered, constructed to represent a measured surface with a moving defect. The function is based on a skewed normal approximation [5] with skewness and height and width scaling parameters distributed normally and defect location uniformly distributed:

$$y(z) = \frac{c_2}{4} \times \begin{cases} 0 & z < \frac{-3}{\lambda_N} \\ \frac{1}{8\sqrt{2\pi}} \mathrm{e}^{\frac{-z^2}{2}} (9\lambda_N z + 3\lambda_N^2 z^2 + \frac{1}{3}\lambda_N^3 z^3 + 9) & \frac{-3}{\lambda_N} \leq z < \frac{-1}{\lambda_N} \\ \frac{1}{4\sqrt{2\pi}} \mathrm{e}^{\frac{-z^2}{2}} (3\lambda_N z - \frac{1}{3}\lambda_N^3 z^3 + 4) & \frac{-1}{\lambda_N} \leq z < \frac{1}{\lambda_N} \\ \frac{1}{8\sqrt{2\pi}} \mathrm{e}^{\frac{-z^2}{2}} (9\lambda_N z - 3\lambda_N^2 z^2 + \frac{1}{3}\lambda_N^3 z^3 + 7) & \frac{1}{\lambda_N} \leq z < \frac{3}{\lambda_N} \\ \sqrt{\frac{2}{\pi}} \mathrm{e}^{\frac{-z^2}{2}} & \frac{3}{\lambda_N} \leq z \end{cases} \tag{A.72}$$

where

$$z = \frac{x - c_1}{c_3} \tag{A.73}$$

and

$$\begin{aligned} \lambda_N &\sim \mathrm{N}(4, 1) \\ c_1 &\sim \mathrm{U}[5, 95] \\ c_2 &\sim \mathrm{N}(1, 0.25) \\ c_3 &\sim \mathrm{N}(1, 0.25). \end{aligned} \tag{A.74}$$

Ten example curves sampled at 100 points representing measurement data points are shown in figure A.36. Perhaps unsurprisingly, linear PCA is not particularly successful

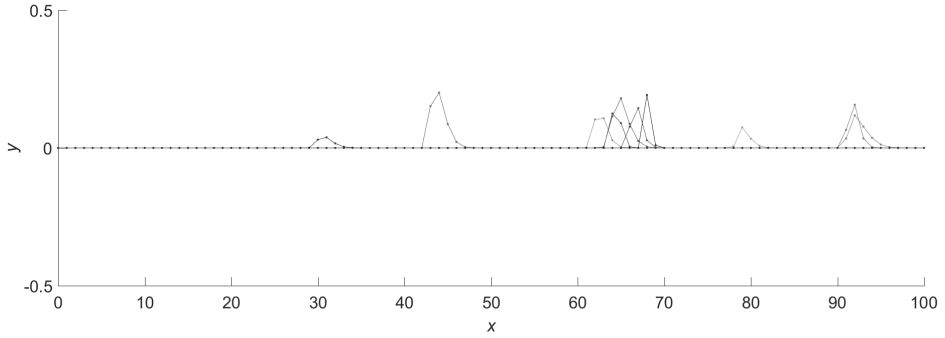
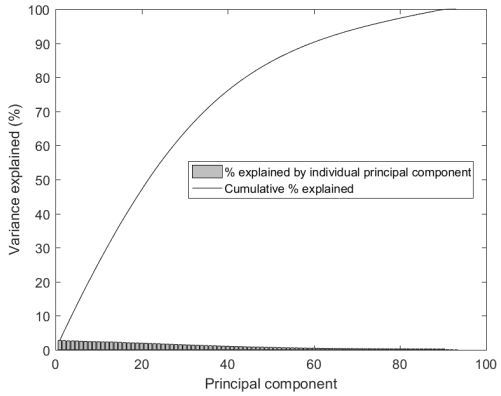
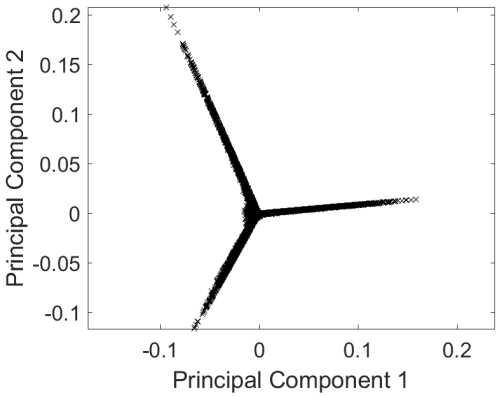


Figure A.36: Ten sets of artificially constructed data sets for a defect feature evaluated at 100 points

in determining a reduced set of points to represent these data sets adequately. Even with 10000 sets 73 principal components are required to represent 95% of the variance, figure A.37a. Figure A.37b gives some indication of the non-linearity caused by the shift in the location of the prescribed defect. Fitted curves using different numbers of principal



(a) Explained variance of the principal components



(b) First two principal components

Figure A.37: Linear PCA of data points from 10,000 randomly drawn 'defect' curves

components are shown in figure A.38. 93 components are required to represent 100% of the variance.

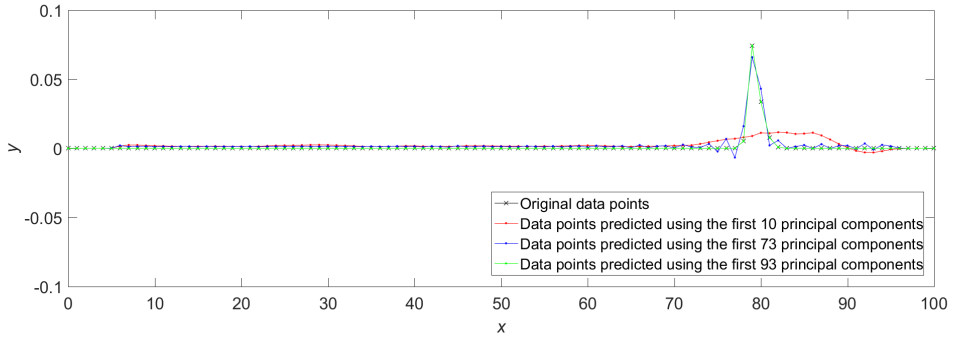


Figure A.38: Re-parameterization of an artificially constructed 'defect' data set using linear PCA

In reality, we are able to observe that the feature of interest has a varying location in x. Therefore, if a location parameter can be extracted prior to PCA, the results might be improved. If this was a real measurement set then the location of each 'bump' would be unknown but it is relatively easy to locate a peak in the data. If the data sets are all adjusted such that they are centred at the peak of the bump then linear PCA of the remaining parameters results in significantly improved variable reduction, figures A.39, A.40, and A.41. 99% of the variance can be represented by just four variables and 100% by eight variables.

We observe that it can be significantly advantageous to 'shift' data and add an extra variable prior to using PCA to reduce the number of defining variables significantly.

Fitted distributions of the resulting four principal components are shown figure A.42. We observe that an input uniform distribution for the location variables results in a

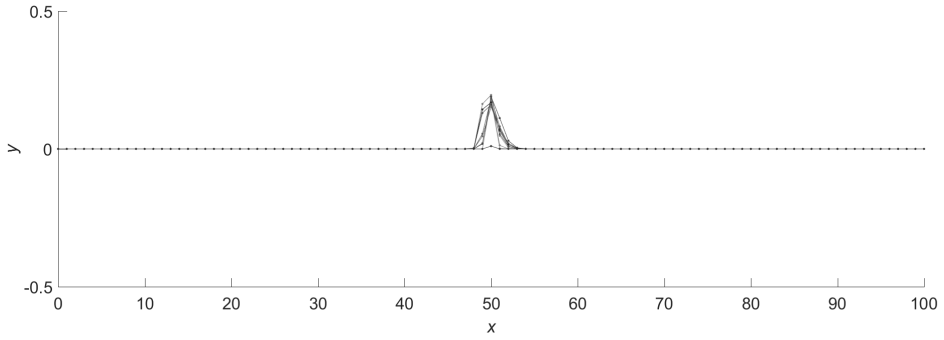


Figure A.39: 'Centred' artificially constructed 'defect' data set

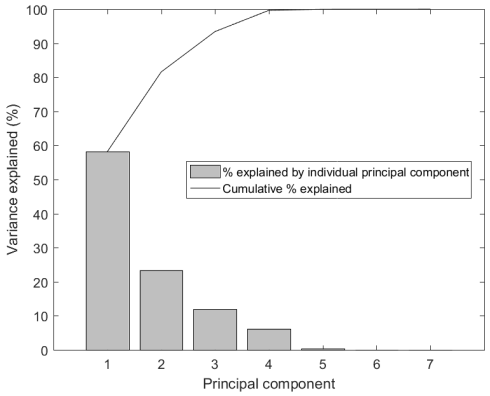


Figure A.40: Explained variance of the principal components from 'centred' artificially constructed 'defect' data set

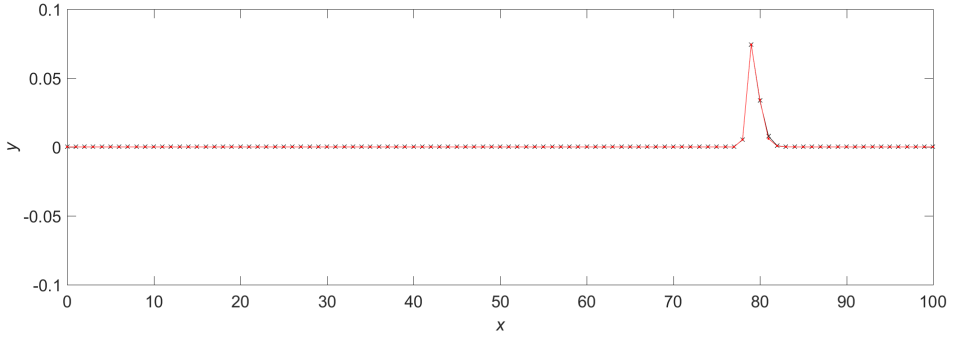


Figure A.41: Re-parameterization of an artificially constructed 'defect' data set using linear PCA with 4 new variables and a location parameter

uniform distribution for one of the principal components too. Assessment of the these distributions again suggests that stable distributions could, in general, be more appropriate than normal distributions; the χ^2 goodness-of-fit test is unreliable for the stable distribution [186, 187] but the null hypothesis that the samples can be drawn from a normal distribution can be rejected for the third principal component.

If the input variables, including the 'bump' position parameter are normally distributed then the findings are the same but when one of the other parameters is given a uniform

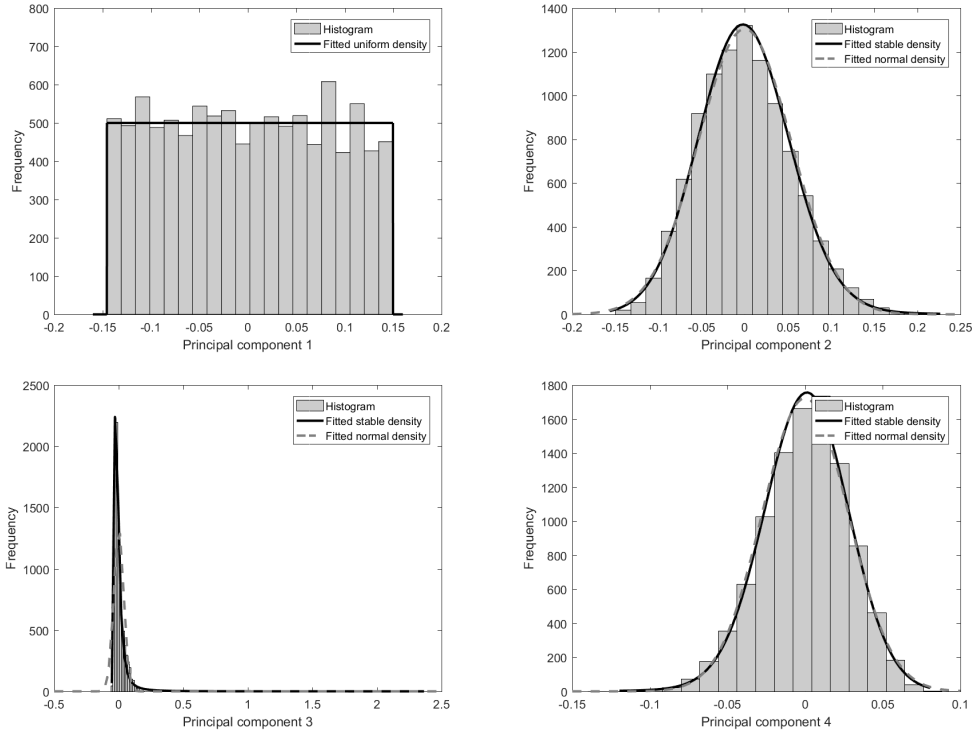


Figure A.42: Histograms of output principal components with fitted distributions ('bump' function)

distribution:

$$\begin{aligned} \lambda_N &\sim \mathrm{N}(4,1) \\ c_1 &\sim \mathrm{N}(50,15) \\ c_2 &\sim \mathrm{U}[0.5,1.5] \\ c_3 &\sim \mathrm{N}(1,0.25), \end{aligned} \tag{A.75}$$

the distribution of the first principal component (the most 'important') becomes difficult to fit, figure A.43.

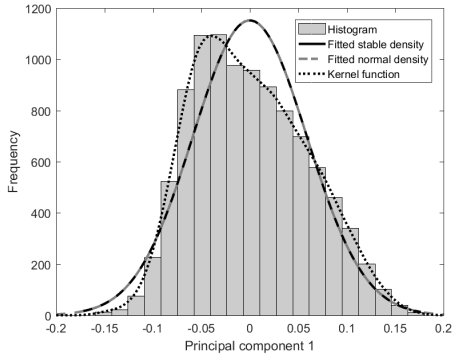


Figure A.43: Histogram of first principal component with fitted distributions ('bump' function with uniformly distributed height parameter c_2)

We note here that PCs are, by definition, independent so could be used, not just to reduce the number of variables, but also to provide alternative independent variables for sampling.

A.10 Geometry manipulation using spatial methods

A.10.0.1 Free-form deformation (FFD); volume-based spatial deformation

The popular algorithm for free-form deformation proposed by Sederberg and Parry [250] still remains useful today. The formulation is reasonably simple to implement compared with the surface techniques. There is no optimization required and the resulting meshes are intrinsically smooth within the control volume.

Sederberg and Parry [250] use a regular cuboid lattice of control points containing the solid mesh to be morphed. The control points are each the control points of defining Bézier spline curves, section A.7.3, the weightings of the respective points (b_i) being defined as Berstein polynomials (Sederberg uses *trivariate tensor product Bernstein polynomials*).

To understand the limitations of this approach, let us first consider a two-dimensional Bézier curve ($\mathcal{B}$) with three *control points* ($\mathbf{a}^{(0)}$,$\mathbf{a}^{(1)}$, and $\mathbf{a}^{(2)}$) [264, 311]:

$$\mathcal{B}(u) = \sum_{i=0}^{2} \mathbf{a}^{(i)} b_{i,2}(u), \tag{A.76}$$

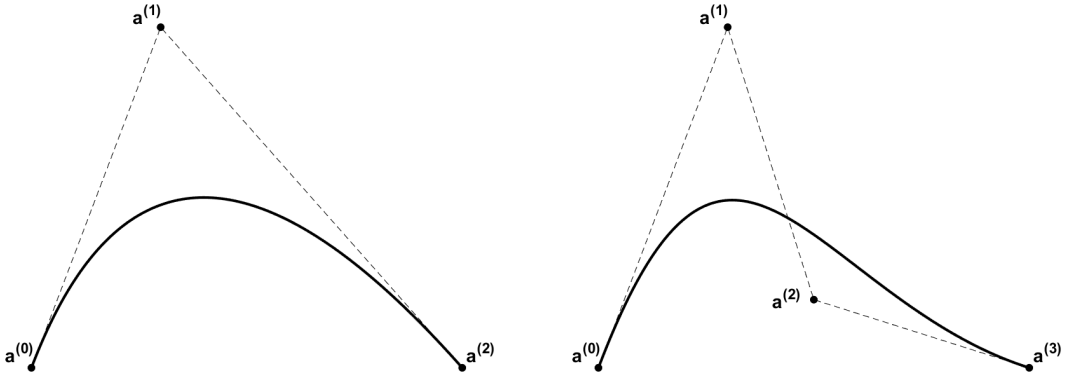
where u is a parameter along the curve in the range [0,1], and b is a quadratic Bernstein basis polynomial:

$$b_{i,2}(u) = \begin{pmatrix} 2 \\ i \end{pmatrix} u^i (1-u)^{2-i}, \quad i = 0, 1, 2, \tag{A.77}$$

where $\begin{pmatrix} 2 \\ i \end{pmatrix}$ is the *binomial coefficient*[14]. By plotting $b(u)$, figure A.45, it can be seen that it is only at the extreme control point positions ($\mathbf{a}^{(0)}$; $(u = 0)$ and $\mathbf{a}^{(2)}$; $(u = 1)$) that the other control points yield no influence on the curve location. This is because a curve of this definition starts and ends at the first and last control point.

[14]The binomial coefficient is sometimes referred to, in probability theory, as "n choose k", and calculated explicitly using the formula:

$${}_nC_k = \begin{pmatrix} n \\ k \end{pmatrix} = \frac{n!}{(n-k)!k!}.$$



(a) Bézier curve defined using 3 control points (b) Bézier curve defined using 4 control points

Figure A.44: Example Bézier curves

Increasing the number of control points, and therefore the degree of the Bernstein polynomials (m) defining control point weightings, results in the generalized expressions:

$$\mathcal{B}(u) = \sum_{i=0}^{m} \mathbf{a}^{(i)} b_{i,m}(u), \tag{A.78}$$

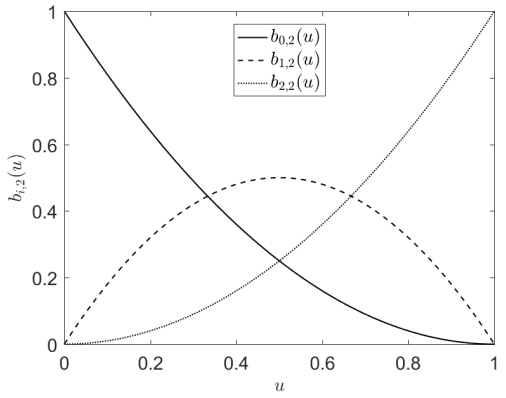
and

$$b_{i,m}(u) = \begin{pmatrix} m \\ i \end{pmatrix} u^i (1-u)^{m-i}, \quad i = 0, 1, ..., m, \tag{A.79}$$

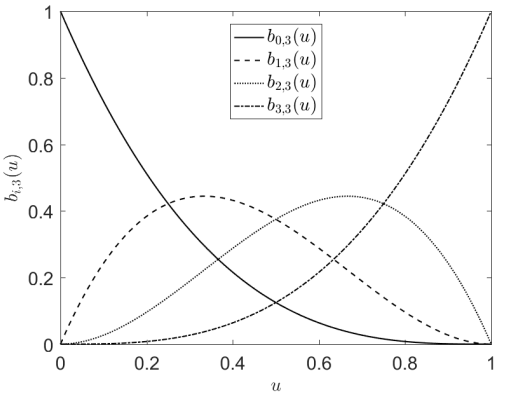
defining the curve. Examples of both curves (figure A.44) and their respective Bernstein polynomials (figure A.45) also show how the **vectors** $\overrightarrow{\mathbf{a}^{(0)}\mathbf{a}^{(1)}}$ and $\overrightarrow{\mathbf{a}^{(m)}\mathbf{a}^{(m-1)}}$ **define the direction of the curve at it's ends**. As such, if a nominally straight line is distorted using control point movement, the only means of maintaining the same gradient at the start and end points (with this type of curve definition) is to ensure that these vectors are also maintained, i.e. there must be a minimum of 5 control points (figure A.46). Adding more control points increases the degree of the curve and all points on the curve are influenced by every control point; there is only global, not local control.

Extension of this concept to three dimensions results in a formula for a *vector valued* displaced point ($\mathbf{X}(s,t,u)$) as the *tensor product* of the Bernstein polynomials in each dimension[15]:

[15] $\mathbf{X}(s,t,u)$ is a vector of the Cartesian co-ordinates of the displaced point, where $0 \leq s \leq 1$, $0 \leq t \leq 1$, and $0 \leq u \leq 1$.

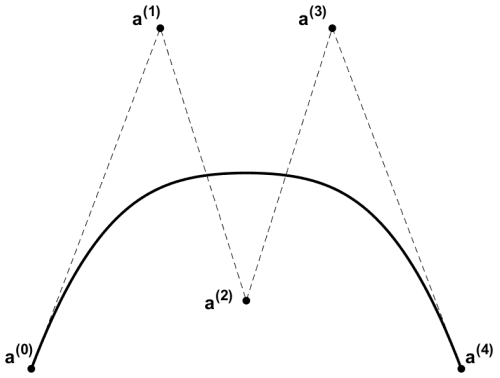


(a) Bernstein quadratic polynomials used in Bézier curve with 3 control points

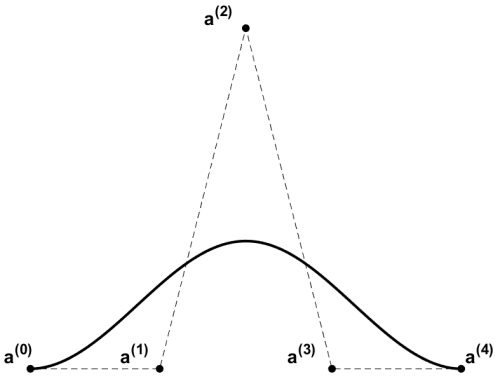


(b) Bernstein cubic polynomials used in Bézier curve with 4 control points

Figure A.45: Example Bernstein polynomials



(a) Bernstein degree 4 polynomials used in Bézier curve with 5 control points



(b) Bernstein degree 4 polynomials used in Bézier curve with 5 control points

Figure A.46: Example Bézier curves using degree 4 Bernstein polynomials

$$\mathbf{X}(s,t,u) = \sum_{i=1}^{l} \binom{l}{i} (1-s)^{l-i} s^i \left[\sum_{j=1}^{m} \binom{m}{j} (1-t)^{m-j} s^j \left[\sum_{k=1}^{n} \binom{n}{k} (1-u)^{n-k} u^k \mathbf{A}^{(i,j,k)} \right] \right], \quad \text{(A.80)}$$

where $\mathbf{A}$ is the lattice of control points $\mathbf{a}$, and l, m, and n are the polynomial degree (number of points minus 1) in each co-ordinate direction. In three dimensions the deformation procedure also follows the same 'rules' in each of the dimensions that have been demonstrated here in a single dimension, i.e. if the bounding box control points are within a mesh (a local mesh region within a body) then in order to maintain tangency with the outside mesh the original vectors between the outermost control points must be maintained and a smaller region within the outer bounding box cannot be controlled in

an isolated manner.

This three-dimensional volume-based deformation scheme is therefore limited to global shape changes with no specific local constraints. The original scheme has been developed by others by using alternative curve definitions. For example Griessmair and Purgathofer [94] use basis-splines (b-splines), which are *piecewise*. B-splines follow a similar formulation to the Bézier curve and an extension to a trivariate surface is equally feasible. A b-spline can be defined as:

$$\mathcal{S}_m(t) = \sum_{i=0}^{n} \mathbf{a}^{(i)} b_{i,m}(t), \tag{A.81}$$

noting that m is the degree of the spline, which can be chosen, rather than being determined by the number of control points. $t \in [k^{(0)}, k^{(n_k)}]$ is a combined value of the summed u's below a given point on the curve. $u \in [0, 1]$ for each curve *segment*. Segment end properties (at *joints*) are determined by the knot vector $\mathbf{K} = [k^{(0)}\ k^{(1)}\ ...\ k^{(n_k)}]$; a non-decreasing integer sequence of length $n_k + 1$, where:

$$n_k = m + n + 1. \tag{A.82}$$

End points of a b-spline have $m + 1$ repeated knot values. The first and last m knots are *external knots*. The remaining knots are referred to as *internal knots* and are not repeated (the curve is *uniform*). A Bézier curve is a b-spline with no internal knots. A *periodic* b-spline has no external knots. The b-spline basis functions are defined as:

$$\begin{aligned} b_{i,0}(u) &= \begin{cases} 1, & \text{if } k^{(i)} \leq t < k^{(i+1)} \text{ and } k^{(i)} < k^{(i+1)} \\ 0, & \text{otherwise} \end{cases} \\ b_{i,j}(u) &= \frac{t - k^{(i)}}{k^{(i+j)} - k^{(i)}} b_{i,j-1}(t) + \frac{k^{(i+j+1)} - t}{k^{(i+j+1)} - k^{(i+1)}} b_{i+1,j-1}(t), \end{aligned} \tag{A.83}$$

where $j = 1, 2, ..., m$ and b is therefore evaluated recursively.

To increase the flexibility of free-form deformation some researchers have turned to the more generalized form of the spline; the non-uniform rational b-spline (NURBS)[144]. The NURBS allows for repeated knot values within the curve (non-uniform). It also allows for the control points to have weightings (w_i) (rational). It is also possible to define a rational Bézier curve by including control point weightings but in general a Bézier curve is not rational. The NURBS is defined as:

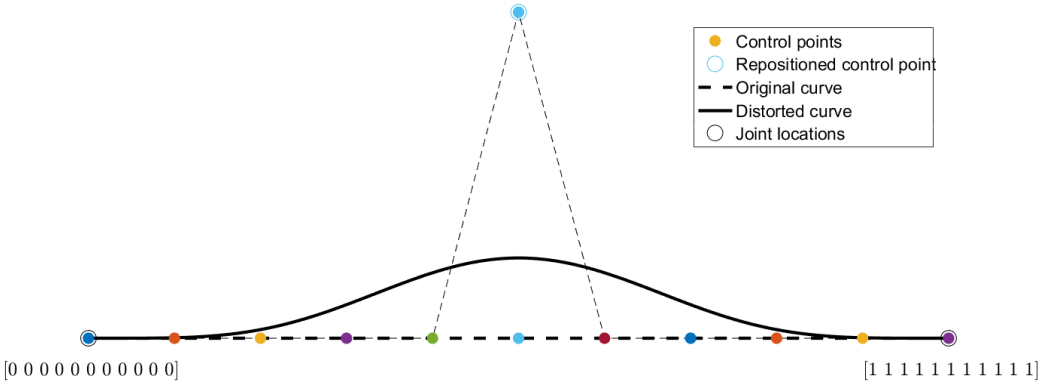
$$\mathcal{N}_m(u) = \frac{\sum_{i=0}^{n} w_i \mathbf{a}^{(i)} b_{i,m}(t)}{\sum_{i=0}^{n} w_i b_{i,m}(t)}, \tag{A.84}$$

and the basis functions are defined as in equation A.83.

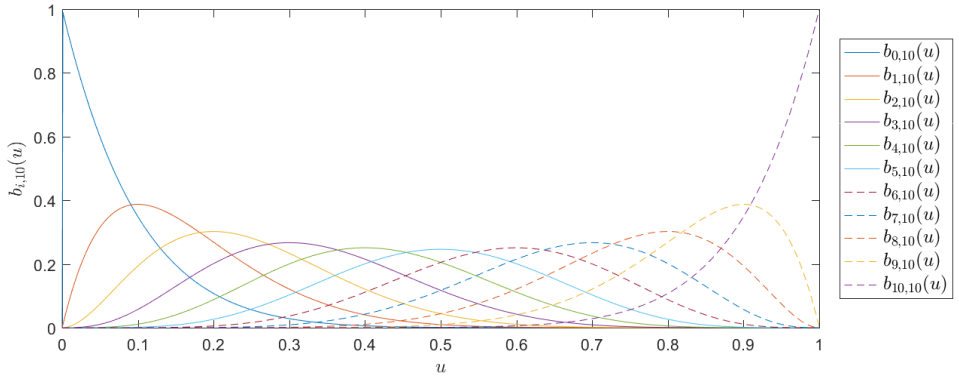
Figures A.47 to A.49 illustrate the piecewise nature of b-spline and NURBS based distortions on a simple curve. Eleven control points result in a degree 10 Bézier curve and all control points and corresponding basis functions influence the position of the curve

at all points except the ends. The knot vector in this case is

$$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]. \tag{A.85}$$



(a) Bézier curve with 11 control points, one control point moved



(b) Basis functions for a Bézier curve with 11 control points

Figure A.47: Example modified ('morphed') Bézier curve

Using the same control points and a cubic b-spline where each segment is controlled by $m + 1 = 4$ overlapping basis functions requires a knot vector:

$$[0\ 0\ 0\ 0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 8\ 8\ 8] \tag{A.86}$$

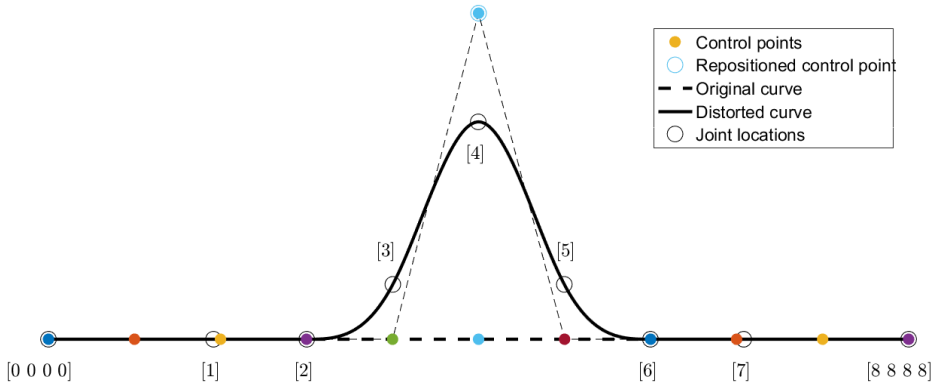
where the internal knots are equally spaced. Local control is achieved.

A NURBS is used to allow greater flexibility, using repeated knot values to allow local distortion to be specified using fewer segments. Here, a knot vector:

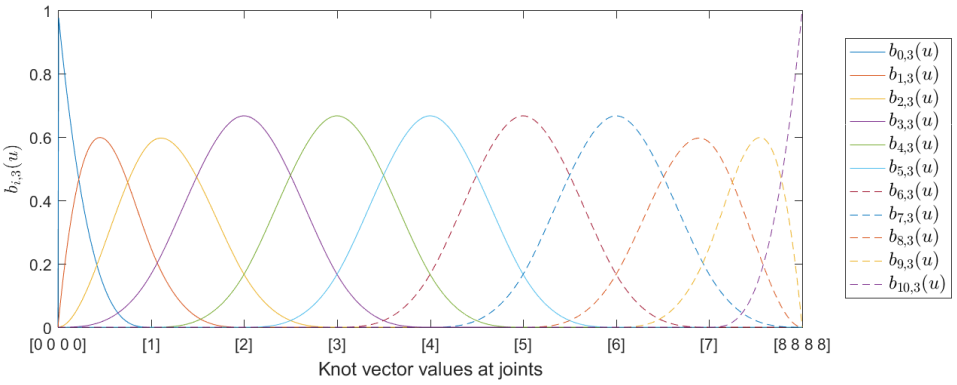
$$[0\ 0\ 0\ 0\ 1\ 1\ 1\ 2\ 3\ 3\ 3\ 4\ 4\ 4\ 4] \tag{A.87}$$

is used achieving local control with only 4 segments.

We can observe in these simple examples that the array of control points needed is not obvious, that the control points do not coincide with the curve. An *interpolating spline*



(a) B-spline curve with 11 control points, one control point moved

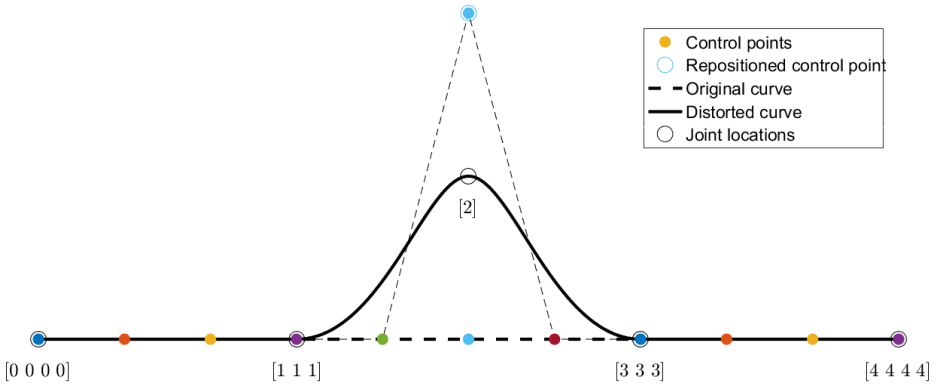


(b) Basis functions for a b-spline curve with 11 control points

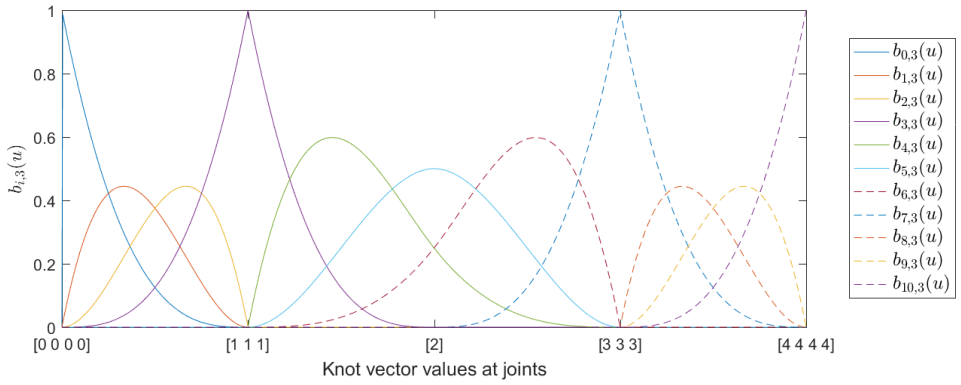
Figure A.48: Example modified ('morphed') b-spline curve

can be defined by calculating control point locations from prescribed joint positions but for morphing of solid meshes this would become infeasible in the trivariate case.

Beyond these approaches there are other extended forms discussed by Gain [83], which use non-cuboid lattices or subdivided regions, all of which result in a "proliferation in control points" and increased "versatility to the detriment of efficiency and ease-of-use".



(a) NURBS curve with 11 control points, one control point moved



(b) Basis functions for a NURBS curve with 11 control points

Figure A.49: Example modified ('morphed') NURBS curve

A.10.0.2 Radial basis function (RBF) morph; a new volume-based spatial deformation

Recently, an approach first introduced in the literature by de Boer et al. [47], has been applied in some CFD meshing and analysis software: notably, ANSYS Fluent with an implementation by Biancolini [15]. Mathematically the method is similar to the FFD algorithm of Sederberg and Parry [250] in that point positions or displacements are determined by a sum of basis functions, which can be weighted[16]:

$$s(\mathbf{x}) = \sum_{i=1}^{n_b} w_i \psi(r) + p(\mathbf{x}). \tag{A.88}$$

However the 'morphed' point positions are not controlled by a set of 'control points and knots' (there is no $\mathbf{a}$). Instead, the 'morphed' nodal positions throughout the full mesh are evaluated as a function of their radial distance from the known positions of the displaced boundary nodes: $(r = ||\mathbf{x} - \mathbf{x}_{\mathrm{b}}^{(i)}||)$,

[16] This equation is the same as that given in section 2.2 but with the addition of a linear polynomial p.

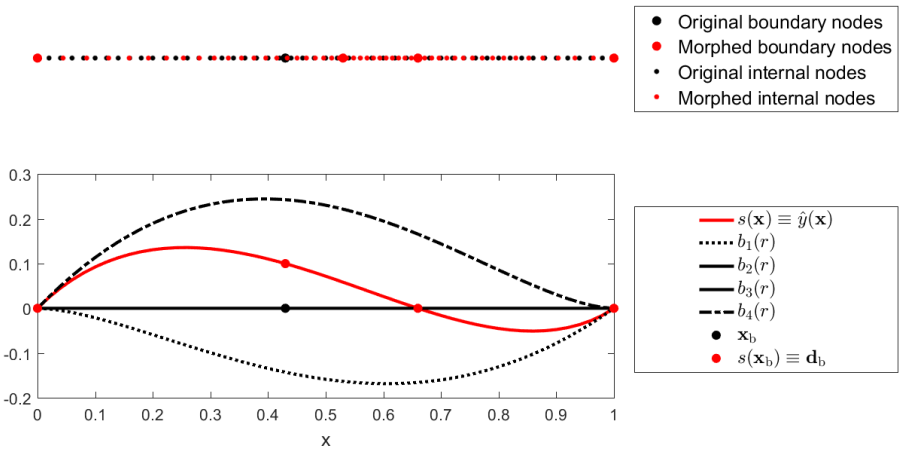


Figure A.50: RBF based morphing in a single dimension x using a thin plate spline basis, showing displacements and basis function values as a function of original x-position

where $\mathbf{x}_\mathrm{b}^{(i)}$ are the basis function centres ($\mathbf{c}^{(i)}$ in section 2.2). The basis functions can be of any chosen form. Any smooth basis function will create a smooth displacement function but the quality of the remaining mesh will be dependent upon this choice and the displacements and directions.

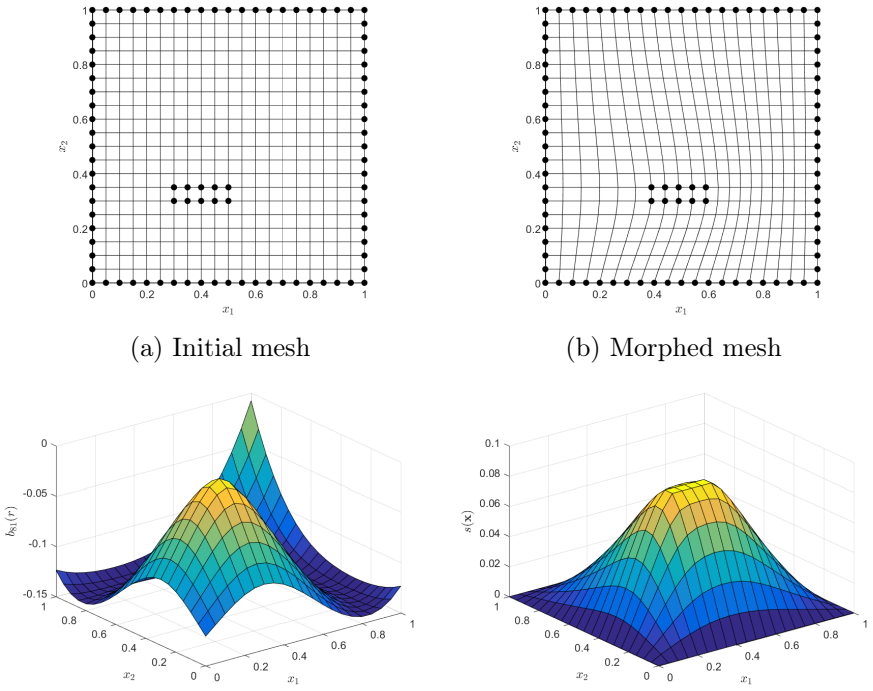
To demonstrate the theory a set of nodes ($\mathbf{x}$) in a single dimension are considered. A small subset of these nodes have known displacements in x ($s(\mathbf{x}_\mathrm{b}) = \mathbf{d}_\mathrm{b}$). These are equivalent to the *boundary*, or surface nodes. The remaining 'morphed' nodal positions must be determined creating an interpolating RBF model of nodal displacements based on the sample of boundary points. In this example a *thin plate spline* basis function:

$$\psi(r) = r^2 \ln r \tag{A.89}$$

has been used. The resulting displaced nodal positions, modelled nodal displacements, and the corresponding weighted basis functions $w_i\psi(r)$ (denoted $b_i(r)$) are plotted (figure A.50).

Extending the same principal to a set of points in two dimensions (using a thin plate spline basis and the same approach as de Boer et al. [47] is illustrated in figure A.51. The central set of boundary points are displaced in the x_1 direction only. The displacement at any point (s) is also shown, along with an example weighted basis function $b_{81}(r)$ (the basis function associated with 81^{st} boundary point at an original position of $(0.3, 0.35)$ in the example).

de Boer et al. [47] states that "the displacement can be interpolated separately for each spatial direction". This is demonstrated in figure A.52 in two dimensions. RBFs and



(a) Initial mesh

(b) Morphed mesh

(c) Weighted basis function $b_8 1(r)$

(d) Displacement s

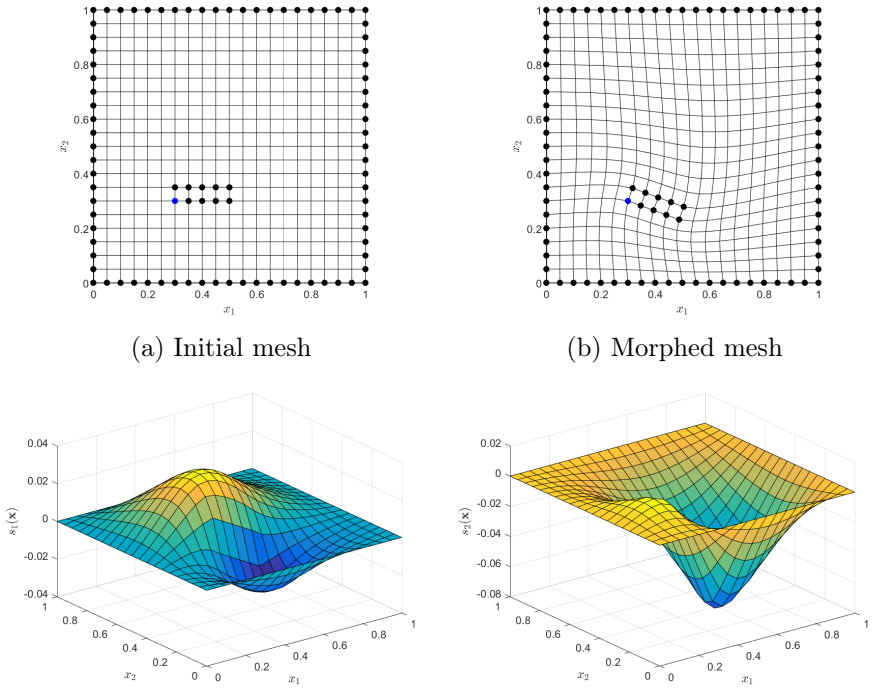
Figure A.51: Example 2D RBF based morph with displacement in the x_1-direction using a thin plate spline RBF

displacements are simply modelled in the x_1-direction to give s_1 and in the x_2-direction to give s_2.

The work of de Boer et al. [47] compares a number of different basis functions with respect to final mesh quality and efficiency for a given rotation and displacement combination for this problem and an airfoil-flap mesh. In this work it is concluded that, out of the fourteen basis functions tested, the compact polynomial basis function:

$$\psi(r) = (1-r)^4(4r+1), \tag{A.90}$$

and the thin plate spline were the preferred bases. Distortion of the mesh may not result in infeasible meshes where displacements are small, such as in sensitivity studies for uncertain geometry, but may result in significant noise depending on the problem. It is therefore important to maintain the best possible mesh quality.



(a) Initial mesh

(b) Morphed mesh

(c) Displacement s_1

(d) Displacement s_2

Figure A.52: Example 2D RBF based morph with displacement in the x_1 and x_1 directions (rotation about blue node) using thin plate spline RBFs

Four radial basis functions are demonstrated (including the two recommended by de Boer et al. [47]) in figure A.53:

$$\text{Cubic} : \psi(r) = r^3 \tag{A.91}$$

$$\text{Compact support} : \psi(r) = (1-r)^4(4r+1) \tag{A.92}$$

$$\text{Thin plate spline} : \psi(r) = r^2 \ln r \tag{A.93}$$

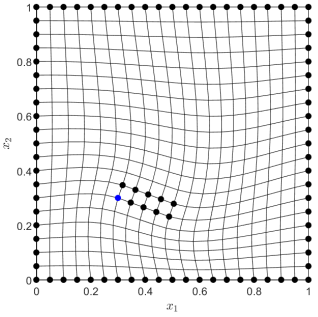
$$\text{Gaussian} : \psi(r) = \mathrm{e}^{-r^2/2\sigma^2} \tag{A.94}$$

$$\tag{A.95}$$

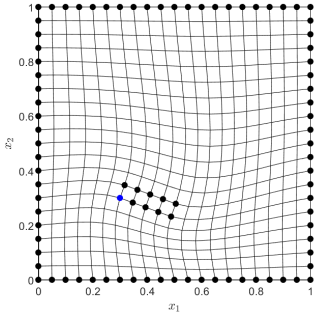
In addition, the Kriging basis (in 2 dimensions) is also considered:

$$\text{Kriging} : \psi(\mathbf{x}) = \exp\left(-\sum_{k=1}^{2} \theta_k |x_k^{(i)} - x_k^{(j)}|^{1.99}\right). \tag{A.96}$$

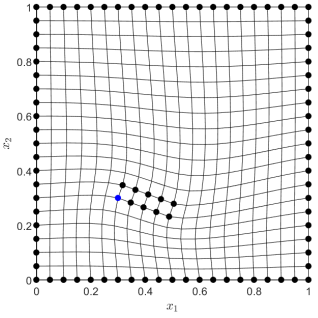
Where the radial basis functions are functions of the Euclidian distance between points and the sample points (basis function centres), the Kriging basis is a function of a non-Euclidian distance i.e. the distance is scaled according to θ_k in each of the dimensions. Tuning the Kriging model, as described in section 2.2, involves finding the values of θ where the likelihood is maximized. In this case, however, the result is almost precisely equivalent to the Gaussian RBF morph ((A.53d) in figure A.53).



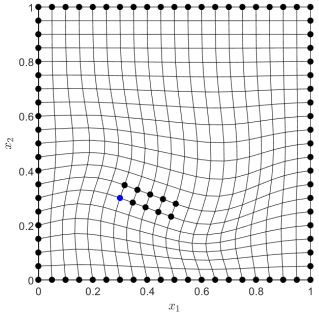
(a) Cubic morph



(b) Compact support basis function morph (function with C^2 continuity)



(c) Thin plate spline morph



(d) Gaussian RBF morph or Kriging morph

Figure A.53: Example 2D RBF and Kriging based morphs with displacement in the x_1 and x_1 directions

Appendix B

Lifing analysis of an aero-engine dovetail joint

The dovetail joint is a critical part of the fan blade assembly in an aero-engine, and in the turbine for some smaller engines. In particular, designers are concerned with the expected life of the components. As discussed throughout chapter 1, replacing parts at longer intervals is desirable form an economic perspective. This appendix presents the construction of a two-dimensional finite element analysis of a dovetail joint, used in chapter 2 to demonstrate some approaches on a realistic, but reasonably fast-running analysis. We consider the life of the disc, rather than the blade, to be the performance parameter in which the designer is most interested in this case.

A two-dimensional plane stress elastic contact analysis of a sector of a rotating disc containing a single blade with dovetail joint is constructed using NX™ for geometry manipulation and ABAQUS® for analysis. The meshed sector is shown in figure B.1.

Geometry

The disc sector geometry is shown in figure B.2. The split regions are regions of differing thickness, chosen to ensure that the loading due to mass distribution throughout the disc are correct, along with the mass of the blade itself.

The nominal dovetail joint geometry is shown in figure 2.7. The nominal design is symmetric about the sector centreline. The blade is positioned such that an equal gap exists between the blade and disc dovetails. On application of the load, the first point of contact is located by the finite element code.

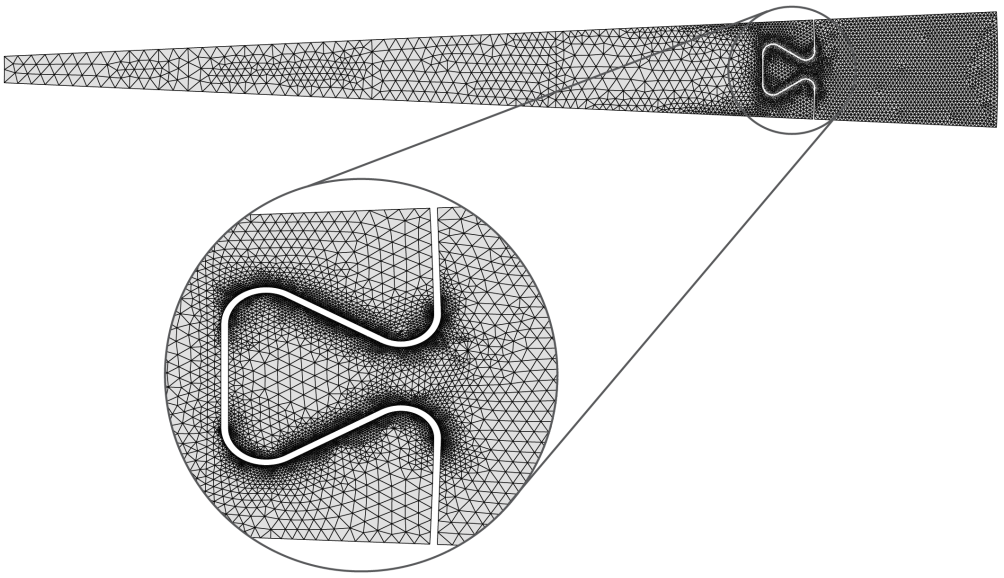


Figure B.1: Finite element analysis model of a disc sector with dovetail joint

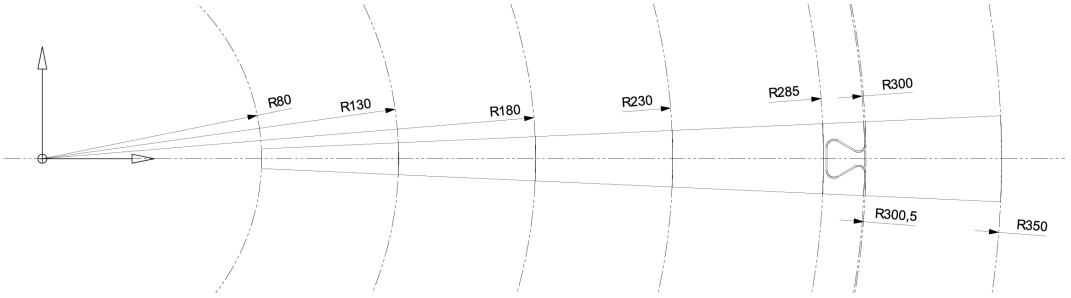


Figure B.2: Nominal geometry of disc sector (mm)

Loads and boundary conditions

The analysis applies centrifugal loading on the blade and disc due to their mass and angular velocity $\omega = 1250$ rad/s. The sector edges on the disc are constrained such that all points on these edges are free to move in the r-direction (radially) but restrained in ϕ-direction (direction of rotation, or *hoop* direction) ($U_\phi = 0$, where U is displacement) in a cylindrical coordinate system about the rotational axis of the disc. To avoid rigid body motion, automatic stabilization is specified in ABAQUS®.

Table B.1: Elastic properties of disc and blade at 600°C

Property	Value
Density, ρ	8080 kg/m^3
Poisson's ratio, ν	0.395
Young's modulus, E	174 GPa

Material properties

Both the disc and blade have elastic properties based on those of a typical precipitation hardened Nickel alloy, INCONEL® alloy 706[1] [270], table B.1. The running temperature is assumed to be 600°C for the purposes of elastic property selection.

Disc sectional geometry is usually designed to maintain close to constant stress throughout by varying disc thickness, an analytical formula for which can be derived [102]. In our two-dimensional simplification, the thickness, and therefore mass distribution, is varied in sections, increasing towards the centre of the disc. The thickness of the blade is selected to result in a total blade mass of 350g for the nominal design. Any further change in blade mass due to changes in joint geometry is considered a genuine effect of geometric changes.

Given that a sector of a rotating disc is being models, the hoop and *radial* stresses have been compared with those predicted by analytical theory and formulae [323, 102], prior to inclusion of the joint itself.

Contact interaction properties

A 'direct' equation solver is used with unsymmetric matrix storage allowed, and a full Newton solution technique specified such that all equations are solved fully at each iterative step. There is a prescribed contact interaction between the disc and blade dovetail surfaces and 'finite-sliding' is considered such that separation, sliding, and rotation, of finite amplitude are possible. The formulation is 'penalty' in both normal and tangential directions with a constant coefficient of friction of 0.4 applied. An initial increment size of 0.001 is specified with a minimum of $0.1\text{e}-12$ and maximum of 1.0. The maximum allowed number of iterations is given as 100.

It is noted that the coefficient of friction is unknown and potentially variable spatially, and over time, in such a situation [241]. There is little consistency in the previous dovetail and firtree joint studies pertaining to the choice of friction coefficient; Meguid et al. [174], Papanikos et al. [199], Hammouda and Fayed [99], and Vale et al. [295] use 0.3 to 0.5, 0.0 to 1.5, 0.25, and 0.0 or 0.25, respectively. Here we consider that the friction coefficient, akin to the applied loads, will affect the stress magnitude but not the

[1] A nickel-iron-chromium alloy used both for compressor and turbine discs [142].

proportion of variation in stress in the linear analysis which is being performed here. A value of 0.4 has been chosen arbitrarily based on the work of Rajasekaran and Nowell [223] who suggest that the friction coefficient varies between 0.1 and 0.65.

Mesh properties

The finite element mesh consists of quadratic triangular elements (ABAQUS® CPS6M). The mesh is generated in NX® with mesh seeds of 0.3mm on the notch edges, increasing to 1.0mm on the sector edges in the region of the joint, and increasing to 4.0mm at the disc centre, figure B.1. The 0.3mm seed size was determined by assessing the convergence of the principal stresses in the notch region with decreasing seed size. The nominal model has a total of 15159 elements with 31559 nodes, and takes 3 minutes and 11 seconds of central processing unit (CPU) time to complete with seventeen contact iterations and no cutbacks. The convergence of peak principal stresses in the notch is plotted, for both blade and disc notch mesh seed sizes, in figure B.3.

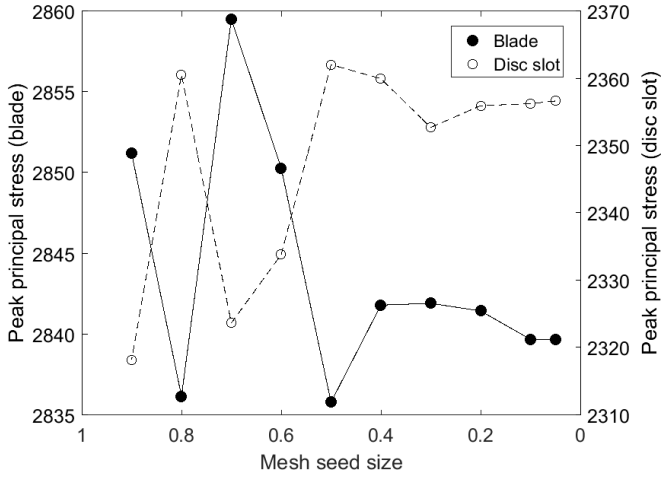


Figure B.3: Plot of notch peak principal stress (N/mm^2) convergence with increasing mesh fidelity (seed size (mm)) for dovetail analysis

Stress as a performance measure in lieu of life

The fatigue life prediction of joints such as these is the subject of much ongoing research, however it is known that mechanisms for fatigue failure, and therefore life, vary depending upon temperature, and are a combination of *low cycle fatigue* (LCF), *high cycle fatigue* (HCF) [280, 20], and *creep* [280]. The radial, low frequency, load cycles result in plastic strain and notch fatigue from which the LCF-life is predicted, and high frequency vibration in the hoop direction results in fretting fatigue [173, 223] in the contact region from which HCF-life is predicted. Here, therefore, LCF-life is considered the dominant

case, and the analysis has been simplified, using the *peak principal stress at the notch* as an indicative measure of the disc dovetail post low cycle fatigue (LCF) life.

Considering the *total life* approach [292], the number of cycles to failure N_f is given by Basquin's equation [115]:

$$\sigma_{ar} = \sigma_f^{'}(2N_f)^b, \tag{A.1}$$

where σ_{ar} is the *equivalent fully reversed stress amplitude*. $\sigma_f^{'}$ and b are material constants: the *fatigue strength coefficient* and *fatigue strength exponent* respectively. Including the effects of mean stress in this equation requires models such as the Smith-Watson-Topper (SWT) correction [115]:

$$\sigma_{ar} = \sqrt{\sigma_{max}\sigma_a}, \tag{A.2}$$

where σ_{max} is the peak stress and σ_a is the stress amplitude. However, it is not clear which stress is most appropriate. In an *equivalent stress (or strain) approach* extensions of the Tresca, or von Mises, yield criteria [11] are used. The alternative *critical plane approaches* [287] assumes that cracks initiate on planes of maximum shear stress. In reality, the mechanism can even "shift from one type to another" [287]. The SWT approach was developed for circumstances where the prevailing mechanism is tensile cracking. These *stress-life* approaches are considered applicable where applied strains are elastic. A *strain-life* approach is required, however, if strains occur within the elastic-plastic range, and the equations can be modified accordingly to account for strain, and using the SWT parameter:

$$\sigma_{max}\varepsilon_a = \frac{(\sigma_f^{'})^2}{E}(2N_f)^{2b} + \varepsilon_f^{'}\sigma_f^{'}(2N_f)^{b+c}, \tag{A.3}$$

where

$$\varepsilon_a = \frac{\Delta\varepsilon}{2},$$

the *strain amplitude*, and the additional material constants, $\varepsilon_f^{'}$, c, and E; the *fatigue ductility coefficient*, *fatigue ductility exponent*, and *modulus of elasticity*, are required.

Issler and Roos [118] and Posavljak and Maksimović [215] have applied SWT to the 'design for life' of nickel-based alloy dovetail joints (INCONEL 718 disc and INCONEL 939 blade, and steel blade and disc respectively). Issler and Roos [118] predict the number of cycles to crack initiation under to an applied oscillating radial load. They compare experimental results with life prediction using FEA and the SWT parameter. Their use of maximum principal stresses and strains proved to compare favourably with the experiments. In contrast, the SWT-based life predictions using von Mises stresses and maximum shear stresses over predicted the component life by a factor of around sixty. In Posavljak and Maksimović [215]'s purely analytical study, the principal stresses were applied using Morrow's mean correction to improve a design for extended fatigue life.

The dovetail joint might also be considered to exhibit the effects observed in *notch fatigue*. The work of Neuber [292] and Peterson [209] in the calculation of *fatigue stress concentration factors* provides equations for stress concentration based on the ratio of local stresses and strains at the notch (perpendicular to crack growth) to applied external stresses and strains, along with adjustments for notch geometry. The geometric adjustments are used to account for the effects of geometry using a material constant "determined from long life fatigue data for sharply notched specimens" [292].

In summary, the life N_f, is proportional to σ_{max} or $\sigma_{max}\varepsilon_a$, depending on whether the loading results in elastic or plastic behaviour. The dovetail disc slot is a notched component but an accurate finite element model allows stresses and strains to be evaluated locally without the application of a stress concentration factor to a notch-free model. It is the peak principal stress, perpendicular to the expected crack growth direction which is applicable in the notched cases, and this approach is further endorsed by the work of Issler and Roos [118] and Posavljak and Maksimović [215]. Inclusion of additional concentration factors due to geometry would only be deemed necessary for "sharp" notches, and the dovetail design parameters are selected to ensure this is not the case.

The plane stress model

The use of a simplified two-dimensional model limits the analysis to an either *plane stress* (thin disc), or *plane strain* (deep disc), approach. In reality, the true joint will exhibit stresses somewhere between the two results. However, given that the joint region of interest is 'thin', the plane stress assumption has been used for this study.

B.1 Automation of the analysis workflow

The analysis is automated through a NX parametric CAD model, used for constructing the joint geometry, and meshing. A python script applies loads, boundary conditions, and contact properties, to the ABAQUS input deck prior to analysis, and the output is interrogated using MATLAB for implementation of DSO and UP workflows. The process is illustrated in figure B.4.

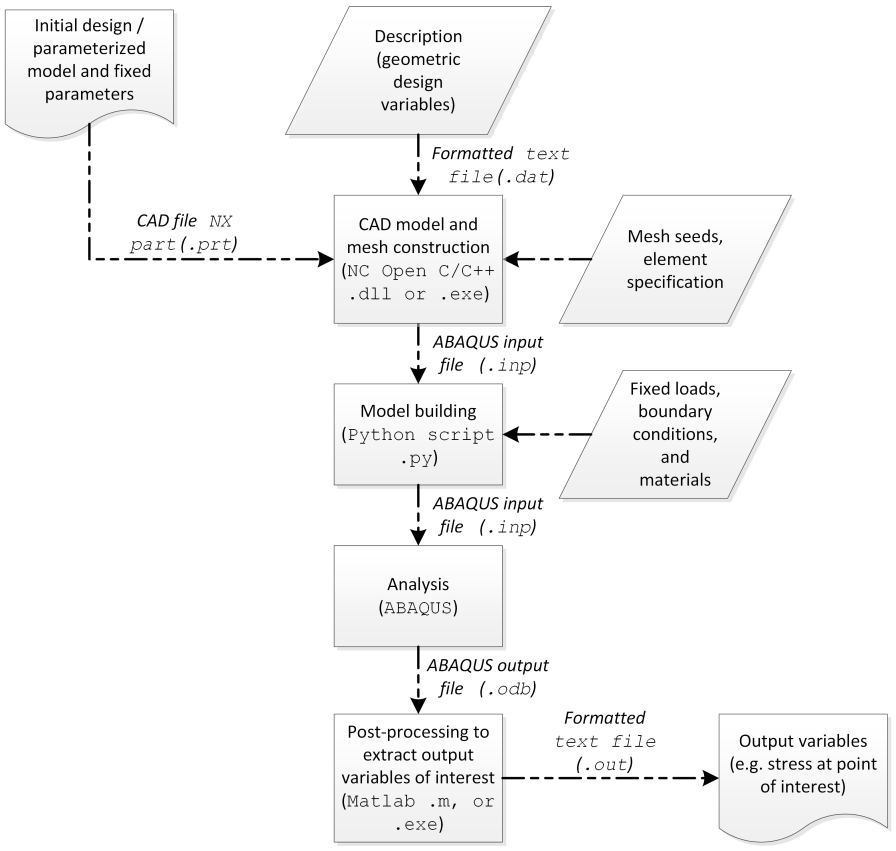


Figure B.4: Automated dovetail analysis workflow (showing code and file transfer)

Bibliography

[1] Agarwal, H., Renaud, J. E., Preston, E. L., and Padmanabhan, D. (2004). Uncertainty quantification using evidence theory in multidisciplinary design optimization. *Reliability Engineering and System Safety*, 85:281–294.

[2] Al-Ahmari, A. and Aalam, J. (2014). Optimizing parameters of freeform surface reconstruction using CMM. *Measurement*, 64:17–28.

[3] Andrieu, C., de Freitas, N., Doucet, A., and Jordan, M. I. (2003). An introduction to MCMC for machine learning. *Machine Learning*, (50):5–43.

[4] Ashby, M. F. (2000). Multi-objective optimization in material design and selection. *Acta Materialia*, 48:359–369.

[5] Ashour, S. K. and Abdel-hameed, M. A. (2010). Approximate skew normal distribution. *Journal of Advanced Research*, (1):341–350.

[6] Athanasiadis, T., Fudos, I., Nikou, C., and Stamati, V. (2012). Feature-based 3D morphing based on geometrically constrained spherical parameterization. *Computer Aided Geometric Design*, 29:2–17.

[7] Atieg, A. and Watson, G. A. (2003). A class of methods for fitting a curve or surface to data by minimizing the sum of squares of orthogonal distances. *Journal of Computational and Applied Mathematics*, 158:277–296.

[8] Aughenbaugh, J. M. and Paredis, C. J. J. (2006). The value of using imprecise probabilities in engineering design. *Journal of Mechanical Design*, 128:969–979.

[9] Baldwin, M. A., Langenderfer, J. E., Rullkoetter, P. J., and Laz, P. J. (2010). Development of subject-specific and statistical shape models of the knee using an efficient segmentation and mesh-morphing approach. *Computer Methods and Programs in Biomedicine*, 97:232–240.

[10] Beaulieu, M., Dufour, J., and Khalaf, L. (2014). Exact confidence sets and goodness-of-fit methods for stable distributions. *Journal of Econometrics*, (181):3–14.

[11] Benham, P., Crawford, R. J., and Armstrong, C. G. (1996). *Mechanics of Engineering Materials*. Pearson Education Ltd., 2nd edition.

[12] Bertsimas, D. and Brown, D. B. (2009). Constructing uncertainty sets for robust linear optimization. *Operations Research*.

[13] Bertsimas, D., Brown, D. B., and Caramanis, C. (2011). Theory and applications of robust optimization. *Society for Industrial and Applied Mathematics Review*, 53(3):464–501.

[14] Beyer, H.-G. and Sendhoff, B. (2007). Robust optimization - a comprehensive survey. *Computer methods in applied mechanics and engineering*, 196:3190–3218.

[15] Biancolini, M. E. (2009). RBF morph: a fast meshless morpher for Fluent. In *EnginSoft - Conference Abstracts, EnginSoft International Conference 2009, CAE Technologies for Industry*.

[16] Botsch, M. and Sorkine, O. (2008). On linear variational surface deformation methods. *IEEE Transactions on Visualization and Computer Graphics*, 14:213–230.

[17] Bowman, A. W. and Azzalini, A. (1997). *Applied Smoothing Techniques for Data Analysis*. Oxford University Press Inc., New York.

[18] Box, G. E. P. (1957). A method for increasing industrial productivity. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 6(2):81–101.

[19] Box, G. E. P. and Draper, N. R. (1987). *Empirical Model-Building and Response Surfaces*. John Wiley & Sons, Inc.

[20] Boyd-Lee, A. B., Harrison, G. F., and Henderson, M. B. (2001). Evaluation of standard life assessment procedures and life extension methodologies for fracture-critical components. *International Journal of Fatigue*, pages S11–S19.

[21] Broomhead, D. S. and Lowe, D. (1988). Multivariate functional interpolation and adaptive networks. *Complex Systems*, 2:321–355.

[22] Buhmann, M. D. (2004). *Radial Basis Functions: Theory and Implementations*. Number 12 in Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press.

[23] Buonamici, F., Carfagni, M., Furferi, R., Governi, L., Lapini, A., and Volpe, Y. (2018). Reverse engineering of mechanical parts: a template-based approach. *Journal of Computational Design and Engineering*, 5:145–159.

[24] Byrd, R. H., Gilbert, J. C., and Nocedal, J. (1996). A trust region method based on interior point techniques for nonlinear programming (Research report RR-2896), INRIA.

[25] Cam, L. L. (1986). The central limit theorem around 1935. *Statistical Science*, 1(1):78–91.

[26] Campbell, R. J. and Flynn, P. J. (2001). A survey of free-form object representation and recognition techniques. *Computer Vision and Image Understanding*, (81):166–210.

[27] Campobasso, M. S., Minisci, E., and Caboni, M. (2016). Aerodynamic design optimization of wind turbine rotors under geometric uncertainty. *Wind Energy*, (19):51–65.

[28] Carr, J. (2014). An introduction to genetic algorithms, Whitman College 345 Boyer Ave. Walla Walla, WA 99362.

[29] Carr, J. C., Beatson, R. K., Cherrie, J. B., Mitchell, T. J., Fright, W. R., McCallum, B. C., and Evans, T. R. (2001). Reconstruction and representation of 3D objects with radial basis functions. In *Proceedings of the 28th annual conference on computer graphics and interactive techniques*, SIGGRAPH, pages 67–76.

[30] Carr, J. C., Fright, W. R., and Beatson, R. K. (1997). Surface interpolation with radial basis functions for medical imaging. *IEEE Transactions on Medical Imaging*, 16(1):96–107.

[31] Cernescu, A., Bortun, C., and Faur, N. (2012). Reverse engineering and fem analysis for mechanical strength evaluation of complete dentures: A case study. In Telea, A. C., editor, *Reverse Engineering Ũ Recent Advances and Applications*, pages 263–276. InTech.

[32] Chen, S., Cowan, C. F. N., and Grant, P. M. (1991). Orthogonal least squares learning algorithm for radial basis function networks. *IEEE Transactions on Neural Networks*, 2(2):302–309.

[33] Chen, W., Allen, J. K., Tsui, K.-L., and Mistree, F. (1996). A procedure for robust design: Minimizing variations caused by noise factors and control factors. *ASME Journal of Mechanical Design*.

[34] Chen, W., Tsui, K. L., Allen, J. K., and Mistree, F. (1995). Integration of the response surface methodology with the compromise decision support problem in developing a general robust design procedure. In *Proceedings of the 1995 Design Engineering Technical Conferences, Advances in Design Automation*, pages 485–492.

[35] Chen, Y. H. and Liu, C. Y. (1997). Robust segmentation of CMM data based on NURBS. *The International Journal of Advanced Manufacturing Technology*, 13:530–534.

[36] Cheng, Y., Wang, Z., Chen, X., Li, Y., Li, H., Li, H., and Wang, H. (2018). Evaluation and optimization of task-oriented measurement uncertainty for coordinate measuring machines based on geometrical product specifications. *International Journal of Applied Sciences*, 9(1).

[37] Chernick, M. R. (2008). *Bootstrap Methods: A Guide for Practitioners and Researchers*. Second edition.

[38] Chernov, N. and Lesort, C. (2005). Least squares fitting of circles. *Journal of Mathematical Imaging and Vision*, 23:239–252.

[39] Chilès, J. P., Aug, C., Guillen, A., and Lees, T. (2004). Modelling the geometry of geological units and its uncertainty in 3D from structural data: The potential-field method. *Orebody Modelling and Strategic Mine Planning*.

[40] Chivate, P. N. and Jablokow, A. G. (1995). Review of surface representations and fitting for reverse engineering. *Computer Integrated Manufacturing Systems*, 8(3):193–204.

[41] Choi, S.-K., Grandhi, R. V., and Canfield, R. A. (2007). *Reliability-based Structural Design*. Springer-Verlag London Limited.

[42] Civil Aviation Authority (2018). *Aviation Trends Quarter 4 2012*. `www.caa.co.uk/airportstatistics`. Accessed 24$^{\text{th}}$ April 2019.

[43] Conn, A. R., Gould, N. I. M., and Toint, P. L. (1991). A globally convergent augmented lagrangian algorithm for optimization with general constraints and simple bounds. *SIAM Journal on Numerical Analysis*, 28(2):545–572.

[44] Cressie, N. A. C. (1993). *Statistics for Spatial Data*. John Wiley & Sons, Inc., revised edition.

[45] Dailey, G. M. (Rolls-Royce plc., Derby, GB) (2003). *Blade platform cooling, Patent no. 6506020*. London, GB. `www.freepatentsonline.com/6506020.html`.

[46] Dassault Systèms Simulia Corp. (2012). *Isight 5.7 Manual*. Engineous Software Inc.

[47] de Boer, A., van der Schoot, M. S., and Bijl, H. (2007). Mesh deformation based on radial basis function interpolation. *Computers and Structures*, 85:784–795.

[48] de Boor, C. (2001). *A Practical Guide to Splines*. Number 27 in Applied Mathematical Sciences. Springer-Verlag New York, Inc., revised edition.

[49] Deb, K. (2011). *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, Inc.

[50] Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197.

[51] Deep, K., Singh, K. P., Kansal, M. L., and Mohan, C. (2009). A real coded genetic algorithm for solving integer and mixed integer optimization problems. *Applied Mathematics and Computation*, (212):505–518.

[52] Delage, E. and Ye, Y. (2010). Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research*.

[53] Dellaert, F. (2008). Singular value and eigenvalue decompositions.

[54] Demeulenaere, B., Pipeleers, G., De Caigny, J., Swevers, J., De Schutter, J., and Vandenberghe, L. (2009). Optimal splines for rigid motion systems: A convex programming framework. *Journal of Mechanical Design*, 131.

[55] Deshpande, A. J., Keane, A. J., Sóbester, A., and Toal, D. J. J. (2012). Geometric parameterisation of firtree joints in gas turbine discs considering manufacturing variability. In *Proceedings of 3rd Aircraft Structural Design Conference*, Delft, Netherlands.

[56] Dey, S., Mukhopadhyay, T., Khodaparast, H. H., Kerfriden, P., and Adhikari, S. (2015). Rotational and ply-level uncertainty in response of composite shallow conical shells. *Composite Structures*, (131):594–605.

[57] Dierckx, P. (1995). *Curve and Surface Fitting with Splines*. Monographs on numerical analysis. Oxford University Press.

[58] Diez, M., Campana, E. F., and Stern, F. (2015). Design-space dimensionality reduction in shape optimization by Karhunen-Loève expansion. *Computer Methods in Applied Mechanics and Engineering*, (283):1525–1544.

[59] do. Carmo, M. P. (1976). *Differential geometry of curves and surfaces*. Englewood Cliffs, N.J. : Prentice-Hall.

[60] Doltsinis, I. and Kang, Z. (2004). Robust design of structures using optimization methods. *Computer methods in applied mechanics and engineering*, 193:2221–2237.

[61] Dowling, M. M., Griffin, P. M., Tsui, K.-L., and Zhou, C. (1997). Statistical issues in geometric feature inspection using coordinate measuring machines. *Technometrics*, 39(1):3–17.

[62] Eberhart, R. and Kennedy, J. (1995). A new optimizer using particle swarm theory. In *Sixth International Symposium on Micro Machine and Human Science*, pages 39–43, Nagoya, Japan. IEEE Press.

[63] Erkan, T., Mayer, R., and Woźniak, A. (2010). Surface probing simulator for the evaluation of CMM probe radius correction software. *International Journal of Advanced Manufacturing Technology*.

[64] Everitt, B. S. (2002). *The Cambridge Dictionary of Statistics*. Cambridge University Press, 2nd edition.

[65] Ezuz, D., Solomon, J., and Ben-Chen, M. (2018). Reversible harmonic maps between discrete surfaces. *ACM Transactions on Graphics*.

[66] Fang, J., Gao, Y., Sun, G., Xu, C., and Li, Q. (2015). Multiobjective robust design optimization of fatigue life for a truck cab. *Reliability Engineering and System Safety*, (135):1–8.

[67] Farin, G. E. (1999). *NURBS From Projective Geometry to Practical Use*. A K Peters, Ltd., 2nd edition.

[68] Feller, W. (1971). *An Introduction to Probability Theory and its Applications*. John Wiley & Sons, Inc., New York, 2nd edition.

[69] Ferson, S. and Donald, S. (1998). Probability bound analysis. In Mosleh, A. and Bari, R. A., editors, *Probability Safety Assessment and Management (PSAM 4): Proceedings of the 4th International Conference on Probabilistic Safety Assessment and Management*, volume 2, pages 1203–1208, New York City, USA. Springer-Verlag.

[70] Ferson, S., Joslyn, A., Helton, J. C., Oberkampf, W. L., and Sentz, K. (2004). Summary from the epistemic uncertainty workshop: consensus amid diversity. *Reliability Engineering and System Safety*, 85:355–369.

[71] Fischer, H. (2011). *A History of the Central Limit Theorem From Classical to Modern Probability Theory*. Springer.

[72] Fisher, R. A. (1971 (first published 1935)). *The Design of Experiments*. Macmillan, 9th edition.

[73] Floater, M. S. and Hormann, K. (2005). Surface parameterization: a tutorial and survey. *Advances in Multiresolution for Geometric Modelling*.

[74] Flynn, J. M., Muelaner, J. E., Dhokia, V., and Newman, S. T. (2016). Improving error models of machine tools with metrology data. *Procedia CIRP*, 52:204–209.

[75] Fonseca, C. M. and Fleming, P. J. (1995). An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary Computing*, 3:1–16.

[76] Forbes, A. B. (1989). Least-squares best-fit geometric elements. Technical Report NPL Report DITC 140/89, National Physical Laboratory.

[77] Forrester, A., Keane, A., and Bressloff, N. (2006). Design and analysis of 'noisy' computer experiments. *AIAA Journal*, 44(10):2331–2339.

[78] Forrester, A., Sóbester, A., and Keane, A. (2008). *Engineering Design via Surrogate Modelling - A Practical Guide*. John Wiley & Sons, Inc.

[79] Forrester, A. I. J. and Keane, A. J. (2009). Recent advances in surrogate-based optimization. *Progress in Aerospace Sciences*, 45:50–79.

[80] Forrester, A. I. J., Sóbester, A., and Keane, A. J. (2007). Multi-fidelity optimization via surrogate modelling. *Proceedings of the Royal Society A*, 463:3251–3269.

[81] Forrester, J. and Keane, A. (2017). Characterization of geometric uncertainty in gas turbine engine components using CMM data. In *World Congress of Structural and Multidisciplinary Optimization*, pages 361–374. Springer.

[82] Full, W., Ehrlich, R., and Klovan, J. (1981). Extended q model - objective definition of external end members in the analysis of mixtures. *Mathematical Geology*, (13):331–334.

[83] Gain, J. (2008). A survey of spatial deformation from a user-centered perspective. *ACM Transactions on Graphics*, 27:Article 107.

[84] Gálvez, A. and Iglesias, A. (2012). Particle swarm optimization for non-uniform rational b-spline surface reconstruction from clouds of 3D data points. *Information Sciences*, 192:174–192.

[85] Gálvez, A., Iglesias, A., and Puig-Pey, J. (2012). Iterative two-step genetic-algorithm-based method for efficient polynomial b-spline surface reconstruction. *Information Sciences*, 182:56–76.

[86] Gameros, A., Chiffre, L., Siller, H. R., Hiller, J., and Genta, G. (2015). A reverse engineering methodology for nickel alloy turbine blades with internal features. *CIRP Journal of Manufacturing Science and Technology*, 9:116–124.

[87] Gander, W. and Gautschi, W. (1998). Adaptive quadrature - revisited. Technical report, ETH Zürich.

[88] Garzon, V. E. and Darmofal, D. L. (2003). Impact of geometric variability on axial compressor performance. *Journal of Turbomachinery*, 125:692–703.

[89] Geng, Z. and Bidanda, B. (2017). Review of reverse engineering systems — current state of the art. *Virtual and Physical Prototyping*, 12:161–172.

[90] Gifi, A. (1990). *Nonlinear multivariate analysis*. John Wiley & Sons, Ltd.

[91] Gill, P. E., Murray, W., and Wright, M. H. (1981). *Practical optimization*. Academic Press Ltd.

[92] Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley.

[93] Golub, G. H. and Van Loan, C. F. (1996). *Matrix Computations*. The Johns Hopkins University Press, 3rd edition.

[94] Griessmair, J. and Purgathofer, W. (1989). Deformation of solids with trivariate b-splines. *EUROGRAPHICS '89*.

[95] Grinstead, B., Koschan, A., and Abidi, M. A. (2006). Geometry refinement of 3D surfaces using Kriging. In *IEEE Proceedings of the Third International Symposium on 3D Data Processing, Visualization, and Transmission*.

[96] Gul, E., Joseph, V. R., Yan, H., and Melkote, S. N. (2018). Uncertainty quantification of machining simulations using an in situ emulator. *Journal of Quality Technology*, 50(3):253–261.

[97] Gunn, S. R. (1998). Support vector machines for classification and regression. Technical report, University of Southampton. http://users.ecs.soton.ac.uk/srg/publications/.

[98] Gunzburger, M., Webster, C. G., and Zhang, G. (2017). Sparse collocation methods for stochastic interpolation and quadrature. In *Handbook of Uncertainty Quantification*.

[99] Hammouda, M. M. I. and Fayed, A. S. (2005). Numerical approach to the mechanics of disc-blade dovetail joints in aero-engine compressors. In *11th International Conference of Fracture, Turin, Italy*.

[100] Han, J. S. and Kwak, B. M. (2001). Robust optimal design of a vibratory microgyroscope considering fabrication errors. *Journal of Micromechanics and Microengineering*, 11:662–671.

[101] Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning: Data Mining, Inference, and Prediction*. Springer New York, second edition.

[102] Hearn, E. J. (1997). *Mechanics of Materials 2*. Butterworth-Heinemann, third edition.

[103] Hedayat, A. S., Sloane, N. J. A., and Stufken, J. (1999). *Orthogonal arrays: theory and applications*. Springer-Verlag New York, Inc.

[104] Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1).

[105] Holland, J. H. (1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. University of Michigan Press.

[106] Horn, J., Nafpliotis, N., and Goldberg, D. E. (1994). A niched Pareto genetic algorithm for multiobjective optimization. In *Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, volume 1, pages 82–87.

[107] Hotelling, H. (1933). *Analysis of a Complex of Statistical Variables into Principal Components*. Warwick & York Inc.

[108] Hoyle, N. (2006). *Automated Multi-Stage Parameterization of Internal Fluid Flow Applications*. PhD thesis, University of Southampton, Faculty of Engineering, Science and Mathematics.

[109] Hu, X., X.Fu, and Liu, L. (2018). Advanced hierarchical spherical parameterizations. *IEEE Transactions on Visualization and Computer Graphics*.

[110] Huang, D., Allen, T. T., Notz, W. I., and Miller, R. A. (2006). Sequential Kriging optimization using multiple-fidelity evaluations. *Structural and Multidisciplinary Optimization*, 32:369–382.

[111] Huang, J. and Menq, C.-H. (2001). Automatic data segmentation for geometric feature extraction from unorganized 3-D coordinate points. *IEEE Transactions on Robotics and Automation*, 17(3):268–279.

[112] Huang, M.-C. and Tai, C.-C. (2000). The pre-processing of data points for curve fitting in reverse engineering. *The International Journal of Advanced Manufacturing Technology*, 16:635–642.

[113] Hudson, D. J. (1966). Fitting segmented curves whose join points have to be estimated. *Journal of the American Statistical Association*, 61(316):1097–1129.

[114] Iman, R. L. and Conover, W. J. (1982). A distribution-free approach to inducing rank correlation among input variables. *Communications in Statistics B*, (11):311–334.

[115] Ince, A. and Glinka, G. (2011). A modification of Morrow and Smith-Watson-Topper mean stress correction models. *Fatigue and Fracture of Engineering Materials and Structures*, 34:854–867.

[116] International Organization for Standardization (ISO) (2008a). *Evaluation of measurement data - Guide to the expression of uncertainty in measurement*. Number 100. Joint Committee for Guides in Metrology, JCGM.

[117] International Organization for Standardization (ISO) (2008b). *Evaluation of measurement data - Supplement 1 to the "Guide to the expression of uncertainty in measurement" - Propagation of distributions using a Monte Carlo method*. Number 101. Joint Committee for Guides in Metrology, JCGM.

[118] Issler, S. and Roos, E. (2003). Numerical and experimental investigations into life assessment of blade-disc connections of gas turbines. *Nuclear Engineering and Design*, 226:155–164.

[119] James, G., Witten, D., and Tibshirani, T. H. H. (2015). *An Introduction to Statistical Learning*. Springer Texts in Statistics. Springer.

[120] Jansen, M., Lombaert, G., and Schevenels, M. (2015). Robust topology optimization of structures with imperfect geometry based on geometric nonlinear analysis. *Computer Methods in Applied Mechanics and Engineering*, (285):452–467.

[121] Johnson, M. E. and Moore, L. M. (1990). Minimax and maximin distance designs. *Journal of Statistical Planning and Inference*, 26(2):131–148. Available on researchgate.

[122] Johnson, R. A. and Wichern, D. W. (2014). *Applied Multivariate Statistical Analysis: Pearson New International Edition*. Pearson Education Ltd., 6th edition.

[123] Jolliffe, I. T. (2002). *Principal Component Analysis*. Springer-Verlag New York, Inc., 2nd edition.

[124] Jones, D. R. (2001). A taxonomy of global optimization methods based on response surfaces. *Journal of Global Optimization*, 21:345–383.

[125] Joseph, V. R., Hung, Y., and Sudjianto, A. (2008). Blind kriging: A new method for developing metamodels. *ASME Journal of Mechanical Design*, 130.

[126] Joskowicz, L., Ostrovsky-Berman, Y., and Myers, Y. (2010). Efficient representation and computation of geometric uncertainty: The linear parametric model. *Precision Engineering*, (34):2–6.

[127] Kacprzyk, J. and Orlovski, S. A., editors (1987). *Optimization Models Using Fuzzy Sets and Possibility Theory*. Springer Science+Business Media.

[128] Karmarkar, N. (1984). A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4):373–395.

[129] Keane, A. J. (2004). Design search and optimisation using radial basis functions with regression capabilities. In Parmee, I. C., editor, *Proceedings of the conference on adaptive computing in design and manufacture*, volume VI, pages 39–49, Berlin. Springer-Verlag London Limited.

[130] Keane, A. J. and Nair, P. (2005). *Computational Approaches for Aerospace Design - The Pursuit of Excellence*. John Wiley & Sons, Ltd.

[131] Kelley, C. T. (1999). *Iterative Methods for Optimization*. Frontiers in Applied Mathematics. Society fo Industrial and Applied Mathematics.

[132] Kennedy, M. C. and O'Hagan, A. (2001). Bayesian calibration of computer models. *Journal of the Royal Statistical Society Series B (Statistical Methodology)*, 63(3):425–464.

[133] Khameneifar, F. and Feng, H.-Y. (2013). Airfoil profile reconstruction under the uncertainty of inspection data points. *International Journal of Advanced Manufacturing Technology*.

[134] Khan Academy (2018). Linear algebra - alternate coordinate systems (bases) - change of basis.

[135] Khuri, A. I. (2003). *Advanced Calculus with Applications in Statistics*. John Wiley & Sons, Inc., 2nd, revised and expanded edition.

[136] Klimke, A. (2008). *Sparse Grid Interpolation Toolbox User's Guide*. Universität Stuttgart, Institut für Angewandte Analysis und Numerische Simulation. Version 5.1.

[137] Klovan, J. and Miesch, A. (1976). Extended cabfac and Qmodel computer programs for Q-mode factor analysis of compositional data. *Computers & Geosciences*, 1(3):161 – 178.

[138] Koch, P. N., Yang, R.-J., and Gu, L. (2004). Design for six sigma through robust optimization. *Structural and Multidisciplinary Optimization*, 26:235–248.

[139] Koshakji, A., A. Quarteroni, A., and Rozza, G. (2013). Free form deformation techniques applied to 3D shape optimization problems. *Communications in Applied and Industrial Mathematics*, 4(452). MATHICSE report 44.2013.

[140] Kraevoy, V. and Sheffer, A. (2004). Cross-parameterization and compatible remeshing of 3d models. In *In: Proceedings of SIGGRAPH*.

[141] Kreyszig, E. (1993). *Advanced Engineering Mathematics*. John John Wiley & Sons, Inc., 7^{th} edition.

[142] Kurzke, J. (2007). Gasturb details 5. `www.gasturb.de/Free/Manuals/manuals.html`.

[143] Lambers, J. (2011). Advanced topics in numerical linear algebra.

[144] Lamousin, H. J. and Waggenspack, W. N. (1994). NURBS-based free-form deformations. *IEEE Computer Graphics and Applications*, pages 59–65.

[145] Lancaster, P. and Šalkauskas, K. (1990). *Curve and Surface Fitting: An Introduction*. Academic Press Ltd.

[146] Larsen, R. J. and Marx, M. L. (2012). *An Introduction to Mathematical Statistics and Its Applications*. Pearson Education, Inc., fifth edition.

[147] Le, V.-B. and Lee, D. T. (1991). Out-of-roundness problem revisited. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3):217–223.

[148] Leary, S., Bhaskar, A., and Keane, A. (2003). Optimal orthogonal-array-based latin hypercubes. *Journal of Applied Statistics*, 30(5):585–598.

[149] Li, M., Fu, P., and Sun, S. (2009). 3D laser scanning data segmentation based on region dilation strategy. Number 9 in International Conference on Hybrid Intelligent Systems, pages 313–316. IEEE Computer Society, IEEE Press.

[150] Li, X., Bao, Y., Guo, X., Jin, M., Gu, X., and Qin, H. (2008). Globally optimal surface mapping for surfaces with arbitrary topology. *IEEE Transactions on Visualization and Computer Graphics*, 14(4):805–819.

[151] Liamaiem, A. and ElMaraghy, H. A. (1999). Curve and surface modeling with uncertainties using dual Kriging. *Journal of Mechanical Design*, 121:249–255.

[152] Lim, D., Ong, Y., Jin, Y., and Sendhoff, B. (2008). Evolutionary optimization with dynamic fidelity computational models. *Advanced Intelligent Computing Theories and Applications With Aspects of Artificial Intelligence (Series: Lecture Notes in Computer Science)*, 5227:235–242.

[153] Lin, G., Wan, X., Su, C., and Karniadakis, G. E. (2007). Stochastic computational fluid mechanics. *CiSI*, pages 21–29.

[154] Linting, M., Meulman, J. J., Groenen, P. J. F., and van der Krooij, A. J. (2007). Nonlinear principal components analysis: Introduction and application. *Psychological Methods*, 12(3):336–358.

[155] Lipowski, A. and Lipowska, D. (2012). Roulette-wheel selection via stochastic acceptance. *Physica A*, (391):2193–2196.

[156] Lira, I., Elster, C., and Wöger, W. (2007). Probabilistic and least-squares inference of the parameters of a straight-line model. *Metrologia*, 44:379–384.

[157] Liu, D., Litvinenko, A., Schillings, C., and V.Schulz (2016). Quantification of airfoil geometry induced aerodynamic uncertainties - comparison of approaches. *SIAM/ASA Journal of Uncertainty Quantification*, 5(1):334–352.

[158] Liu, H., Wang, X., Wu, X., and Qiang, W. (2006). Surface reconstruction based on radial basis functions network. In *Proceedings of the Third international conference on Advances in Neural Networks*.

[159] Loeppky, J. L., Sacks, J., and Welch, W. (2008). Choosing the sample size of a computer experiment: A practical guide. Technical Report Number 170, National Institute of Statistical Sciences.

[160] Macdonald, A. J. (2018). *Rothampsted Research: Guide to the Classical and other Long-term experiments, Datasets, and Sample Archive*.

[161] Mangado, N., Piella, G., Noailly, J., Pons-Prats, J., and Ballester, M. A. G. (2016). Analysis of uncertainty and variability in finite element computational models for biomedical engineering: Characterization and propagation. *Frontiers in Bioengineering and Biotechnology*, 4(85).

[162] Marler, R. T. and Arora, J. S. (2004). Survey of multi-objective optimization methods in engineering. *Structural and Multidisciplinary Optimization*, 26:369–395.

[163] MATLAB (2010a). *Mathworks® MATLAB® V5.0 (R2010a) Optimization Toolbox User's Guide*. The MathWorks, Inc.

[164] MATLAB (2010b). *Mathworks® MATLAB® V7.3 (R2010a) Statistics Toolbox User's Guide*. The MathWorks, Inc.

[165] MATLAB (2013). *Mathworks® MATLAB® V5.0 (R2013b) Curve Fitting Toolbox User's Guide*. The MathWorks, Inc.

[166] MATLAB (2015). *Mathworks® MATLAB® (R2015b) Curve Fitting Toolbox Documentation*. The MathWorks, Inc.

[167] MATLAB (2016). *Mathworks® MATLAB® V3.4 (R2016a) Global Optimization Toolbox User's Guide*. The MathWorks, Inc.

[168] MATLAB (2018). *Mathworks® MATLAB® R2018a Statistics and Machine Learning Toolbox Documentation*. The MathWorks, Inc.

[169] Mavrotas, G. (2009). Effective implementation of the ϵ-constraint method in multi-objective mathematical programming problems. *Applied Mathematics and Computation*, 213:455–465.

[170] McGee, V. E. and Carleton, W. T. (1970). Piecewise regression. *Journal of the American Statistical Association*, 65(331):1109–1124.

[171] McKay, B. D. and Rogoyski, E. (1995). Latin squares of order 10. *The Electronic Journal of Combinatorics*, 2(N3).

[172] McKay, M. D., Beckman, R. J., and Conover, W. J. (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245.

[173] McVeigh, P. A., Harish, G., Farris, T. N., and Szolwinski, M. P. (1999). Modeling interfacial conditions in nominally flat contact for application to fretting fatigue of turbine engine components. *International Journal of Fatigue*, 21(Supplement 1):S157–S165.

[174] Meguid, S. A., Kanth, P. S., and Czekanski, A. (2000). Finite element analysis of fir-tree region in turbine discs. *Finite Elements in Analysis and Design*, 35:305–317.

[175] Metropolis, N. (1987). The beginning of the Monte Carlo method. *Los Alamos Science*, pages 125–130.

[176] Metropolis, N. and Ulam, S. (1949). The Monte Carlo method. *Journal of the American Statistical Association*, 44:335–341.

[177] Miettinen, K. (1998). *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers.

[178] Mishra, S. (2007). Some new test functions for global optimization and performance of repulsive particle swarm method. *Munich Personal RePEc Archive*.

[179] Mitutoyo Intelligent Computer Aided Technology (2012). MCOSMOS software, Coordinate measuring machine software for users from entry-level to expert, bulletin no. 2020, `https://ecatalog.mitutoyo.com/cmimages/003/315/2020_MCOSMOS.pdf`.

[180] Moler, C. (2004). *Numerical Computing with MATLAB*. Web edition.

[181] Morris, M. D. and Mitchell, T. J. (1995). Exploratory designs for computational experiments. *Journal of Statistical Planning and Inference*, 43:381–402.

[182] Myers, R. H. and Montgomery, D. C. (1995). *Response Surface Methodology*. John Wiley & Sons, Inc.

[183] Nair, V. N., Abraham, B., MacKay, J., Box, G., Kacker, R. N., Lorenzen, T. J., Lucas, J. M., Myers, R. H., Vining, G., Nelder, J. A., Phadke, M. S., Sacks, J., Welch, W. J., Shoemaker, A. C., Tsui, K. L., Taguchi, S., and Wu, C. F. J. (1992). Taguchi's parameter design: A panel discussion. *Technometrics*, 34:127–161.

[184] Ng, L. W. T. and Willcox, K. E. (2014). Multifidelity approaches for optimization under uncertainty. *International Journal for Numerical Methods in Engineering*, 100:746–772.

[185] Nocedal, J. and Wright, S. J. (2006). *Numerical Optimization*. Springer, 2nd edition.

[186] Nolan, J. P. (1999). Fitting data and assessing goodness-of-fit with stable distributions. *Mimeograph, Mathematics Department, American University, Washington, DC*.

[187] Nolan, J. P. (2018). *Stable Distributions - Models for Heavy Tailed Data*. Birkhauser, Boston. In progress, Chapter 1 online at `http://fs2.american.edu/jpnolan/www/stable/stable.html`.

[188] Nurdin, A., Bressloff, N. W., and Keane, A. J. (2012). Shape optimisation using CAD linked free-form deformation. *The Aeronautical Journal*, 116(1183):915–939.

[189] Oberkampf, W. L., Helton, J. C., Joslyn, C. A., Wojtkiewicz, S. F., and Ferson, S. (2004). Challenge problems: uncertainty in system response given uncertain parameters. *Reliability Engineering and System Safety*, 85:11–19.

[190] O'Hagan, A. (2013). Polynomial chaos: A tutorial and critique from a statistician's perspective. *Submitted to SIAM/ASA Journal of Uncertainty Quantification*.

[191] O'Hare, J. (2016). Coordinate measuring system data reduction.

[192] Ohtake, Y., Belyaev, A., and Seidel, H. (2006). Sparse surface reconstruction with adaptive partition of unity and radial basis functions. *Graphical Models*, 68:15–24.

[193] Oladyshkin, S. and Nowak, W. (2012). Data-driven uncertainty quantification using the arbitrary polynomial chaos expansion. *Reliability Engineering and System Safety*, 106:179–190.

[194] Osman, I. H. and Kelly, J. P. (1996). *Meta-heuristics: Theory and Applications*. Kluwer Academic Publishers.

[195] Owen, A. B. (1992). Orthogonal arrays for computer experiments, integration and visualization. *Statistica Sinica*, 2:439–452.

[196] Paciorek, C. J. (2003). Nonstationary Gaussian processes for regression and spatial modelling.

[197] Padulo, M., Campobasso, M. S., and Guenov, M. D. (2011). Novel uncertainty propagation method for robust aerodynamic design. *AIAA Journal*, 49(3):530–543.

[198] Papalambros, P. Y. and Wilde, D. J. (2017). *Principles of Optimal Design: Modeling and Computation*. Cambridge University Press, third edition.

[199] Papanikos, P., Meguid, S. A., and Stjepanovic, Z. (1998). Three-dimensional nonlinear finite element analysis of dovetail joints in aeroengine discs. *Finite Elements in Analysis and Design*, 29:173–186.

[200] Papoulis, A. and Pillai, S. U. (2002). *Probability, Random Variables and Stochastic Processes*. McGraw-Hill Companies, Inc.

[201] Park, G.-J., Lee, T., Lee, K. H., and Hwang, H.-H. (2006). Robust design: An overview. *AIAA Journal*, 44(1):181–191.

[202] Parr, J. (2013). *Improvement Criteria for Constraint Handling and Multiobjective Optimization*. PhD thesis, Faculty of Engineering and the Environment Computational Engineering and Design Group.

[203] Pearson FRS, K. (1901). On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.

[204] Peggs, G. N., Maropoulos, P. G., Hughes, E. B., Forbes, A. B., Robson, S., Ziebart, M., and Muralikrishnan, B. (2009). Recent developments in large-scale dimensional metrology. *Proceedings of the Institute of Mechanical Engineers*, 223 Part B:571–595.

[205] Petrone, G., Hill, D. C., and Biancolini, M. E. (2013). Optimization under uncertainty using adjoint solver and RBF morph. In *10th International Conference on Evolutionary and Deterministic Methods for Design, Optimization and Control with Applications to Industrial and Societal Problems - EUROGEN 2013*.

[206] Phadke, M. S. (1989). *Quality engineering using robust design*. P T R Prentice-Hall, Inc.

[207] Picheny, V., Wagner, T., and Ginsbourger, D. (2013). A benchmark of kriging-based infill criteria for noisy optimization. *Journal of Structural and Multidisciplinary Optimization*.

[208] Pichot, G. (2016). Algorithms for gaussian random field generation. Technical Report RT-0484, INRIA.

[209] Pilkey, W. D. and Pilkey, D. F. (2008). *Peterson's Stress Concentration Factors*. John Wiley & Sons, Inc., third edition.

[210] Pinnau, R. (2008). Model reduction via proper orthogonal decomposition. In Schilders, W., van der Vorst, H., and Rommes, J., editors, *Model Order Reduction: Theory, Research Aspects and Applications*. Springer, Berlin, Heidelberg.

[211] Pintér, J. D. (2013). Handbook of global optimization. volume 2, chapter 15, pages 515–569. Springer Science & Business Media.

[212] Poirier, D. J. (1973). Piecewise regression using cubic spline. *Journal of the American Statistical Association*, 68(343):515–524.

[213] Pólya, G. (1920). Über den zentralen grenzwertsatz der wahrscheinlichkeitsrechnung und das momentenproblem. *Mathematische Zeitschrift*, 8:171–181.

[214] Poniatowska, M. (2015). Free-form surface machining error compensation applying 3D CAD machining pattern model. *Computer-Aided Design*, (62):227–235.

[215] Posavljak, S. and Maksimović, S. (2008). Redesign of dovetail joints based on estimated low cycle fatigue life. *Scientific Technicl Review*, LVIII(3-4):38–44.

[216] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (2007). *Numerical Recipes: The Art of Scientific Computing*, volume 1. Cambridge University Press, 3rd edition.

[217] Process Quality Associates Inc. (2006). *The Evolution of Six Sigma*. `www.pqa.net/ProdServices/sixsigma/W06002009.html`. Accessed 23rd May 2013.

[218] Putko, M. M., Newman, P. A., III, A. C. T., and Green, L. L. (2001). Approach for uncertainty propagation and robust design in CFD using sensitivity derivatives. In *15th AIAA Computational Fluid Dynamics Conference*, Anaheim, CA. AIAA.

[219] Queipo, N. V., Haftka, R. T., Shyy, W., Goel, T., Vaidyanathan, R., and Tucker, P. K. (2005). Surrogate-based analysis and optimization. *Progress in Aerospace Sciences*, 41:1–28.

[220] Rabbani, T., van den Heuvel, F. A., and Vosselman, G. (2006). Segmentation of point clouds using smoothness constraint. ISPRS Commission V Symposium 'Image Engineering and Vision Metrology', pages 248–253, Dresden. IAPRS.

[221] Rahayem, M. R. and Kjellander, J. A. P. (2011). Quadric segmentation and fitting of data captured by laser profile scanner mounted on an industrial robot. *International Journal of Advanced Manufacturing Technology*, 52:155–169.

[222] Rajabi, M. M. (2019). Review and comparison of two meta-model-based uncertainty propagation analysis methods in groundwater applications: polynomial chaos expansion and Gaussian process emulation. *Stochastic Environmental Research and Risk Assessment*, (33):607–631.

[223] Rajasekaran, R. and Nowell, D. (2006). Fretting fatigue in dovetail blade roots: Experiments and analysis. *Tribology International*, 39:1277–1285.

[224] Ralston, A. and Rabinowitz, P. (1978). *A first course in numerical analysis*. McGraw-Hill Companies, Inc., 2nd edition.

[225] Rao, S. S. (2009). *Engineering optimization: theory and practice*. John Wiley & Sons, Inc., 4th edition.

[226] Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. Massachusetts Institute of Technology.

[227] Ray, S. R., Avancha, R., Shankaran, S., and Dailey, L. (2017). Modeling surface variation and assessment of its impact on aerodynamic performance in turbomachinery applications. In *Proceedings of ASME Turbo Expo 2017: Turbomachinery Technical Conference and Exposition*.

[228] Razak, T. A., Baharudin, B., Shafee, K., and Shamsuddin, K. (2019). Application of a portable coordinate measuring machine onto automotive door panel for quality inspection activity. In *Advanced Engineering for Processes and Technologies*. Springer International Publishing.

[229] Ren, M. J., Cheung, C. F., and Kong, L. B. (2011). A robust surface fitting and reconstruction algorithm for form characterization of ultra-precision freeform surfaces. *Measurement*, 44:2068–2077.

[230] Renault, A., Massa, F., Lallemand, B., and Tison, T. (2016). Experimental investigations for uncertainty quantification in brake squeal analysis. *Journal of Sound and Vibration*, (367):37–55.

[231] Renishaw plc (2013). SP25M Technical Paper. `http://www.renishaw.com/en/sp25m-6688`.

[232] Richardson, J. N., Coelho, R. F., and Adriaenssens, S. (2015). Robust topology optimization of truss structures with random loading and material properties: A multiobjective perspective. *Computers and Structures*.

[233] Rivest, L.-P. (1998). Linear models for estimating the motion of rigid bodies with applications to geometric quality assurance. *Journal of the American Statistical Association*, 93(442):632–642.

[234] Robinson, T. J., Borror, C., and Myers, R. H. (2004). Robust parameter design: A review. *Quality and Reliability Engineering International*, 20:81–101.

[235] Rolls-Royce plc. (2005). *The Jet Engine*. Rolls-Royce plc.

[236] Roos, D. and Hoffmann, R. (2008). Successive robust design optimization of an electronic connector. In *Weimar Optimization and Stochastic Days*, number 5, Luthergasse 1d, D-99423 Weimar. DYNARDO - Dynamic Software and Engineering GmbH.

[237] Ross, P. J. (1988). *Taguchi Techniques for Quality Engineering.* McGraw-Hill Companies, Inc.

[238] Ross, S. A. and Cheah, L. (2018). Uncertainty quantification in life cycle assessments. *Journal of Industrial Ecology*, 23(2):335–346.

[239] Roy, P., Daviller, G., Jouhaud, J., and Cuenot, B. (2017). Uncertainty quantification-driven robust design assessment of a swirler's geometry. In *Initiative en combustion avancée.*

[240] Rozza, G. and Manzoni, A. (2010). Model order reduction by geometrical parametrization for shape optimization in computational fluid dynamics. In *5th European Conference on Computational Fluid Dynamics, ECCOMAS CFD.*

[241] Ruiz, C. and Nowell, D. (2000). Designing against fretting fatigue in aeroengines. In Fuentes, M., Elices, M., Martin-Meizoso, A., and Martinez-Esnaola, J. M., editors, *Fracture Mechanics: Applications and Challenges, Invited Papers presented at the 13th European Conference on Fracture (ESIS Publication)*, pages 73–95. Elsevier.

[242] Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989). Design and analysis of computer experiments. *Statistical Science*, 4(4):409–435.

[243] Salomon, D. (2006). *Curves and Surfaces for Computer Graphics.* Springer.

[244] Sándor, Z. and András, P. (2004). Alternative sampling methods for estimating multivariate normal probabilities. *Journal of Econometrics*, 120:207–234.

[245] Saunders, M. (2003). CMM inspection fundamentals `http://resources.renishaw.com/en/download/cmm-inspection-fundamentals-6855`.

[246] Schaffer, J. D. (1985). Multiple objective optimization with vector evaluated genetic algorithms. In *The First International Conference on Genetic Algorithms and their Applications*, pages 93–100, Pittsburgh. Hillsdale: Lawrence Erlbaum Associates.

[247] Schillings, C. and Schulz, V. (2015). On the influence of robustness measures on shape optimization with stochastic uncertainties. *Optimization and Engineering*, 16:347–386.

[248] Schölkopf, B. and Smola, A. J. (2002). *Learning with Kernels; Support Vector Machines, Regularization, Optimization, and Beyond.* Massachusetts Institute of Technology.

[249] Scott, P. J. and Forbes, A. B. (2012). Mathematics for modern precision engineering. *Philosophical Transactions of the Royal Society A*, 370:4066–4088.

[250] Sederberg, T. W. and Parry, S. R. (1986). Free-form deformation of solid geometric models. *Proceedings of SIGGRAPH '86*, 20.

[251] Shampine, L. F. (2008). Vector adaptive quadrature in MATLAB®. *Journal of Computational and Applied Mathematics*, 221:131–140.

[252] Sheffer, A., Praun, E., and Rose, K. (2006). Mesh parameterization methods and their applications. *Foundations and Trends in Computer Graphics and Vision*, 2(2):105–171.

[253] Shen, L. and Makedon, F. (2006). Spherical mapping for processing of 3D closed surfaces. *Image and Vision Computing*, 24:743–761.

[254] Shen, T.-S., Huang, J., and Menq, C.-H. (2000). Multiple-sensor integration for rapid and high-precision coordinate metrology. *IEEE/ASME Transactions on Mechatronics*, 5(2):110 –121.

[255] Shigley, J. E. and Mischke, C. R., editors (1996). *Standard Handbook of Machine Design*. McGraw-Hill Companies, Inc., second edition.

[256] Sieger, D., Menzel, S., and Botsch, M. (2012a). A comprehensive comparison of shape deformation methods in evolutionary design optimization. In *EngOpt 2012 - International Conference on Engineering Optimization*.

[257] Sieger, D., Menzel, S., and Botsch, M. (2012b). High quality mesh morphing using triharmonic radial basis functions. In *21st International Meshing Roundtable*, pages 1–15. Springer-Verlag.

[258] Sigal, I. A., Hardisty, M. R., and Whyne, C. M. (2008). Mesh-morphing algorithms for specimen-specific finite element modeling. *Journal of Biomechanics*, 41:1381–1389.

[259] Simpson, J. and Weiner, E., editors (1989). *The Oxford English Dictionary*. Oxford University Press.

[260] Simpson, T. W., Mauery, T. M., Korte, J. J., and Mistree, F. (2001). Kriging models for global approximation in simulation-based multidisciplinary design optimization. *AIAA Journal*, 39(12):2233–2241.

[261] Sloane, N. J. A. (2013). *A Library of Orthogonal Arrays*. `www.neilsloane.com/oadir/index.html`. Accessed 3rd July 2013.

[262] Smith, L. I. (2002). A tutorial on principal components analysis. Technical report, Department of Computer Science, University of Otago, NZ.

[263] Smith, S. T. (2005). Covariance, subspace, and intrinsic Cramér-Rao bounds. *IEEE Transactions on Signal Processing*, 53(5):1610–1630.

[264] Sóbester, A. and Forrester, A. I. J. (2015). *Aircraft Aerodynamic Design: Geometry and Optimization*. Aerospace Series. John Wiley & Sons, Ltd.

[265] Sobol', I. M. (1979). On the systematic search of a hypercube. *SIAM Journal on Numerical Analysis*, 16(5):790–793.

[266] Sobol', I. M. (1994). *A Primer for the Monte Carlo Method*. CRC Press LLC.

[267] Sorenson, H. W. (1970). Least-squares estimation: from Gauss to Kalman. *IEEE Spectrum Magazine*, pages 63–68.

[268] Sorzano, C., Vargas, J., and Pascual Montano, A. (2014a). A survey of dimensionality reduction techniques.

[269] Sorzano, C. O. S., Vargas, J., and Pascual-Montano, A. (2014b). A survey of dimensionality reduction techniques.

[270] Special Metals Corporation (2004). Inconel® alloy 706 publication number smc-091.

[271] Srinivas, N. and Deb, K. (1995). Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation*, 2:221–248.

[272] Staten, M. L., Owen, S. J., Shontz, S. M., Salinger, A. G., and Coffey, T. S. (2012). A comparison of mesh morphing methods for 3D shape optimization. In Quadros, W. R., editor, *Proceedings of the 20th International Meshing Roundtable*, pages 293–311. Springer Berlin Heidelberg.

[273] Stefanou, G. (2009). The stochastic finite element method: Past, present and future. *Computer Methods in Applied Mechanics and Engineering*, (198):1031–1051.

[274] Stein, M. (1987). Large sample properties of simulations using Latin hypercube sampling. *Technometrics*, 29(2):143–151.

[275] Stern, F., Volpa, S., Gaul, N., Choi, K. K., Diez, M., Broglia, R., Durante, D., Campana, E. F., and Lemma, U. (2017). Development and assessment of uncertainty quantification methods for ship hydrodynamics. In *55th AIAA Aerospace Sciences Meeting*.

[276] Strang, G. (2016). Diagonalizing a matrix `https://www.youtube.com/watch?v=U8R54zOTVLw`.

[277] Stuart, A. and Ord, J. K. (1994). *Kendall's Advanced Theory of Statistics: Distribution Theory*, volume 1. Halsted Press, an imprint of John Wiley & Sons, Inc., 6$^{\text{th}}$ edition.

[278] Stufken, J. (2011). A short overview of orthogonal arrays `www.newton.ac.uk/programmes/DAE/seminars/2011090517151.pdf?` Isaac Newton Institute for Mathematical Sciences Design and Analysis of Experiments Seminar, Cambridge, U.K.

[279] Sudret, B. and Der Kiureghian, A. (2000). Structural engineering, mechanics and materials: Stochastic finite element methods and reliability.

[280] Sujata, M., Venkataswamy, M. A., Parameswara, M. A., Suresh Kumar, M., and Bhaumik, S. K. (2006). Failure of life extended rotor discs of aeroengines. In Manjunatha, C. M., Ranganath, V. R., Paretkar, R. K., and Peshwe, D. R., editors, *XIV NASAS: Fatigue, Fracture and Ageing Structures*, pages 185–190, Nagpur. Visvesvaraya National Institute of Technology.

[281] Taguchi, G. (1986). *Introduction to Quality Engineering: Designing Quality into Products and Processes*. Nordica International Ltd.

[282] Tai, C.-C. and Huang, M.-C. (2000). The processing of data points basing on design intent in reverse engineering. *International Journal of Machine Tools and Manufacture*, 40:1913–1927.

[283] Tang, B. (1993). Orthogonal array-based Latin hypercubes. *Journal of the American Statistical Association*, 88:1392–1397.

[284] Taylor, J. (2009). Random fields: stats352-spatial statistics `http://statweb.stanford.edu/~jtaylo/courses/stats352/notes/random_fields.pdf`.

[285] Taylor III, A. C., Green, L. L., Newman, P. A., and Putko, M. M. (2003). Some advanced concepts in discrete aerodynamic sensitivity analysis. *AIAA Journal*, 41(7):1224–1229.

[286] Thompson, W. B., Owen, J. C., de St. Germain, H. J., Stark, S. R., and Henderson, T. C. (1999). Feature-based reverse engineering of mechanical parts. *IEEE Transactions on Robotics and Automation*, 15(1):57–66.

[287] Tipton, S. M. and Nelson, D. V. (1997). Advances in multiaxial fatigue life prediction for components with stress concentrations. *International Journal of Fatigue*, 19(19):503–515.

[288] Tishler, A. and Zang, I. (1981). A new maximum likelihood algorithm for piecewise regression. *American Statistical Association*, 76(376):980–987.

[289] Toal, D. J. J., Bressloff, N. W., Keane, A. J., and Holden, C. M. E. (2011). The development of a hybridized particle swarm for Kriging hyperparameter tuning. *Engineering Optimization*, 43(6):675–699.

[290] Toms, J. D. and Lesperance, M. L. (2003). Piecewise regression: a tool for identifying ecological thresholds. *Ecology*, 84(8):2034–2041.

[291] Tootkaboni, M., Asadpoure, A., and Guest, J. K. (2012). Topology optimization of continuum structures under uncertainty – a polynomial chaos approach. *Computational Methods in Applied Mechanics and Engineering*, 263:263–275.

[292] Topper, T. H., Wetzel, R. M., and Morrow, J. (1967). Neuber's rule applied to fatigue of notched specimens. Aeronautical structures laboratory report NAEC-ASL-1114, U. S. Naval Air Engineering Center, Philadelphia, Pennsylvania.

[293] Trainer, A., Hedberg, T., Feeney, A. B., Fischer, K., and Rosche, P. (2016). Gaps analysis of integrating product design, manufacturing, and quality data in the supply chain using model-based definition. In *Proceedings of the ASME 2016 International Manufacturing Science and Engineering Conference.*

[294] Uchaikin, V. V. and Zolotarev, V. M. (1999). *Chance and Stability.* Utrecht: VSP Press.

[295] Vale, T., Villar, G., and Menezes, J. (2012). Methodology for structural integrity analysis of gas turbine blades. *Journal of Aerospace Technology Management*, 4(1):51–59.

[296] van der Maaten, L. J. P., Postma, E. O., and van den Herik, H. J. (2009). Dimensionality reduction: A comparative review, Technical report, Tilburg University.

[297] Van Loock, W., Pipeleers, G., De Schutter, J., and Swevers, J. (2011). A convex optimization approach to curve fitting with b-splines. In *Preprints of the 18th IFAC World Congress*, pages 2290–2295, Milano, Italy. International Federation of Automatic Control.

[298] Vančo, M. and Brunnett, G. (2004). Direct segmentation of algebraic models for reverse engineering. *Computing*, 72:207–220.

[299] Ventura, P. D. and Klinetob, C. B. (2011). *Fan Blade Dovetail with Compliant Layer, Patent Application no. 20110033302.* USA. `www.faqs.org/patents/app/20110033302`.

[300] Vijverberg, W. P. M. (1997). Monte Carlo evaluation of multivariate normal probabilities. *Journal of Econometrics*, 76:281–307.

[301] Wäldele, F., Bittner, B., Busch, K., Drieschner, R., and Elligsen, R. (1993). Testing of coordinate measuring machine software. *Precision Engineering*, 15:121–123.

[302] Wang, D., DiazDelaO, F. A., Wang, W., Lin, X., Patterson, E. A., and Mottershead, J. E. (2016). Uncertainty quantification in DIC with Kriging regression. *Optics and Lasers in Engineering*, (78):182–195.

[303] Wang, D., DiazDelaO, F. A., Wang, W., and Mottershead, J. E. (2015a). Full-field digital image correlation with Kriging regression. *Optics and Lasers in Engineering*, (67):105–115.

[304] Wang, J., Leach, R., and Jiang, X. (2015b). Review of the mathematical foundations of data fusion techniques in surface metrology. *Surface Topography: Metrology and Properties*, 3.

[305] Watson, G. A. (1999). Least squares fitting of circles and ellipses to measured data. *BIT Numerical Mathematics*, 39(1):176–191.

[306] Weckenmann, A., Estler, T., Peggs, G., and McMurty, D. (2004). Probing systems in dimensional metrology. *CIRP Annals - Manufacturing Technology*, 53(2):657ÂŰ684.

[307] Weisstein, E. W. (2013a). *MathWorld - A Wolfram Web Resourse*. `www.mathworld.wolfram.com`. Accessed 26th June 2013.

[308] Weisstein, E. W. (2013b). *"Maximum Likelihood" from MathWorld - A Wolfram Web Resourse*. `www.mathworld.wolfram.com/MaximumLikelihood.html`. Accessed 16th January 2014.

[309] Weisstein, E. W. (2015a). *"Covariance" from MathWorld - A Wolfram Web Resourse*. `www.mathworld.wolfram.com/Covariance.html`. Accessed 8th October 2015.

[310] Weisstein, E. W. (2015b). *"Singular Value Decomposition" from MathWorld - A Wolfram Web Resourse*. `www.mathworld.wolfram.com/SingularValueDecomposition.html`. Accessed 24th September 2015.

[311] Weisstein, E. W. (2018a). *"Bernstein Polynomial" from MathWorld - A Wolfram Web Resourse*. `www.mathworld.wolfram.com/BernsteinPolynomial.html`. Accessed 20th February 2018.

[312] Weisstein, E. W. (2018b). *"Eigen Decomposition" from MathWorld - A Wolfram Web Resourse*. `www.http://mathworld.wolfram.com/EigenDecomposition.html`. Accessed 18th May 2018.

[313] Weisstein, E. W. (2018c). *"Factorization" from MathWorld - A Wolfram Web Resourse*. `www.mathworld.wolfram.com/MaximumLikelihood.html`. Accessed 7th June 2018.

[314] Weisstein, E. W. (2018d). *"Stable distribution" from MathWorld - A Wolfram Web Resourse*. `www.mathworld.wolfram.com/StableDistribution.html`. Accessed 28th June 2018.

[315] Weisstein, E. W. (2019). *"Hypothesis Testing" from MathWorld - A Wolfram Web Resourse*.

[316] Willink, R. (2008). Estimation and uncertainty in fitting straight lines to data: different techniques. *Metrologia*, 45:290–298.

[317] Wold, S. (1987). Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, (2):37–52.

[318] Woo, H., Kang, E., Wang, S., and Lee, K. H. (2002). A new segmentation method for point cloud data. *International Journal of Machine Tools and Manufacture*, 42:167–178.

[319] Xia, H., Ding, Y., and Wang, J. (2008). Gaussian process method for form error assessment using coordinate measurements. *IIE Transactions*, 40:931–946.

[320] Xiong, F., Greene, S., Chen, W., Xiong, Y., and Yang, S. (2010). A new sparse grid based method for uncertainty propagation. *Structural and Multidisciplinary Optimization*, 41:335–349.

[321] Xiu, D. (2015). Stochastic collocation methods: a survey. In *Handbook of Uncertainty Quantification*. Springer, Cham.

[322] Yin, J., Ng, S. H., and Ng, K. M. (2011). Kriging metamodel with modified nugget-effect: The heteroscedastic variance case. *Computers and Industrial Engineering*, pages 760–777.

[323] Young, W. C. and Budynas, R. G. (2002). *Roark's Formulas for Stress and Strain*. McGraw-Hill Companies, Inc., seventh edition.

[324] Zeiss, C. (Accessed 2019). Carl Zeiss Industrial Metrology, ZEISS metrology software overview `https://www.zeiss.co.uk/metrology/brochures.html?catalog=calypso`.

[325] Zhang, H., Mullen, R. L., and Muhanna, R. L. (2010). Finite element structural analysis using imprecise probabilities based on p-box representation. In *4th International Workshop on Reliable Engineering Computing*.

[326] Zhang, L., Liu, L., Ji, Z., and Wang, G. (2006). Manifold parameterization. *In: Advances in Computer Graphics*, 4035.

[327] Zoutendijk, G. (1960). *Methods of feasible directions*. Elsevier Publishing Company.