MASTER THESIS

Optimisation under uncertainty and constraints of a tube bundle heat exchanger

Supervisors:  
Prof. Hrvoje Jasak  
Dr. Henrik Rusche

Student:  
Tessa Uroić

Zagreb, 2014
I hereby declare that this thesis is entirely the result of my own work except where otherwise indicated. I have fully cited all used sources and I have only used the ones given in the list of references.

I would like to express my sincere gratitude to my supervisors, Prof. Hrvoje Jasak and Dr. Henrik Rusche, who have given me many valuable comments, supported me throughout my thesis and allowed me to explore and work in my own way.

I am also grateful to Prof. Zvonimir Guzović for his continuous support and mentorship during my studies.

Finally, I would like to thank my colleague and partner Vanja Škurić, for his immense patience, help and love.
# Contents

1. Introduction ........................................................................................................... 1

2. Probability ............................................................................................................. 2
   2.1. Definition of Probability ........................................................................... 2
       2.1.1. Frequentistic Definition ................................................................ 2
       2.1.2. Classical Definition ...................................................................... 2
       2.1.3. Bayesian Definition ...................................................................... 3
   2.2. Sample Space and Events ......................................................................... 4
   2.3. The Three Axioms of Probability Theory ................................................. 4
   2.4. Conditional Probability ............................................................................. 5
   2.5. Characteristics of the Data ........................................................................ 6
       2.5.1. Central Measures .......................................................................... 6
       2.5.2. Dispersion Measures .................................................................... 7
       2.5.3. Other Measures ............................................................................ 7
       2.5.4. Measures of Correlation ............................................................... 8
   2.6. Uncertainty Modeling ............................................................................. 10
       2.6.1. Random Variables ...................................................................... 10
   2.7. Methods of Reliability ............................................................................ 20
       2.7.1. Failure Events and Basic Random Variables ..................................... 20
       2.7.2. Linear Limit State Functions and Normal Distributed Variables ...... 21
       2.7.3. The Error Propagation Law ............................................................. 23
       2.7.4. Non-linear Limit State Functions .................................................... 25
       2.7.5. Sampling Methods ...................................................................... 27

3. Dakota Software Package and Optimization Capabilities .................................. 30
   3.1. Optimization ............................................................................................. 31
       3.1.1. Optimization Formulations ............................................................... 32
       3.1.2. Optimization Methods .................................................................. 34
   3.2. Optimization Under Uncertainty ............................................................. 36
       3.2.1. Uncertain Variables ....................................................................... 37
   3.3. Uncertainty Quantification ...................................................................... 38
       3.3.1. Sampling Methods ....................................................................... 39
       3.3.2. Reliability methods ....................................................................... 42

4. Multi-objective Optimization ............................................................................. 47
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Aggregating Functions</td>
<td>50</td>
</tr>
<tr>
<td>4.2</td>
<td>Multi-objective Optimization Algorithm MOGA and Pareto Front</td>
<td>50</td>
</tr>
<tr>
<td>5.</td>
<td>Tube Bank Heat Exchanger Problem</td>
<td>54</td>
</tr>
<tr>
<td>5.1</td>
<td>Problem Parameters</td>
<td>55</td>
</tr>
<tr>
<td>6.</td>
<td>Results and discussion</td>
<td>63</td>
</tr>
<tr>
<td>6.1</td>
<td>Multi-objective Genetic Algorithm Options</td>
<td>63</td>
</tr>
<tr>
<td>6.2</td>
<td>Uncertainty Quantification</td>
<td>73</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Uncertainty Analysis of Design 1 with Relaxed Constraints</td>
<td>74</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Uncertainty Analysis of Design 2 and Design 3 with Relaxed Constraints</td>
<td>82</td>
</tr>
<tr>
<td>6.2.3</td>
<td>Uncertainty Analysis of Designs with Strong Constraints</td>
<td>90</td>
</tr>
<tr>
<td>7.</td>
<td>Conclusion</td>
<td>95</td>
</tr>
</tbody>
</table>
List of figures

Figure 1  Venn diagrams illustrating the union of events (left) and the intersection of events (right) [4]...4  
Figure 2  Illustration of skewness................................................................................................................. 8  
Figure 3  Illustration of kurtosis................................................................................................................... 8  
Figure 4  Two examples of paired data sets [1]............................................................................................ 9  
Figure 5  Illustration of A) a cumulative distribution function and B) a probability density function for a 
continuous random variable [4].................................................................................................................. 12  
Figure 6  Illustration of A) a cumulative distribution function and B) a probability density function for a 
discrete random variable [4]....................................................................................................................... 12  
Figure 7  Illustration of probability density and cumulative distribution functions for different distribution 
types. Mean $\mu = 160$ and standard deviation $\sigma = 40$ are the same for all the distributions except the 
Exponential distribution for which the expected value and the variance are equal [4] .............................. 16  
Figure 8  Example of the density and distribution function of a normally distributed random variable 
deﬁned by the parameters $\mu = 160$ and $\sigma = 40$ [4].................................................................................. 17  
Figure 9  Illustration of the relationship between a Normal distributed random variable and a standard 
Normal distributed random variable [4]......................................................................................................... 18  
Figure 10 Double sided and symmetrical $1 - \alpha$ confidence interval on the mean value [4]...............19  
Figure 11 Illustration of the two-dimensional case of a linear limit state function and Normal distributed 
variables $x$ [4]........................................................................................................................................... 22  
Figure 12 Illustration of the error propagation law: The transformation of the density function $f_Y(y)$ 
according to the relation $y = g(x)$ and the linear approximation of the relation between the two random 
variables [4]................................................................................................................................................ 25  
Figure 13 Illustration of the linearization proposed by Hasofer and Lind [5] in standard normal space [4] 
.................................................................................................................................................................... 27  
Figure 14 An example of Latin Hypercube sampling with four bins in design parameters $x_1$ and $x_2$. The 
dots are the sample sites. [7]...................................................................................................................... 41  
Figure 15 Distribution of input sample points for Monte Carlo $\Delta$ and Latin Hypercube + sampling for 
number of samples equal to 5 and 10 [7]..................................................................................................... 42  
Figure 16 Graphical depiction of calculation of cumulative distribution function in the original uncertain 
variable space [7]........................................................................................................................................ 43  
Figure 17 Graphical depiction of integration for the calculation of cumulative distribution function in the 
transformed uncertain variable space [7]...................................................................................................... 44  
Figure 18 Example for the rank of 10 individuals and the corresponding probability to participate in the 
reproduction process, represented on a circular diagram [9]....................................................................... 49  
Figure 19 Schematic representation of the concept of Pareto optimal front [9]........................................ 51  
Figure 20 The domain is divided into three zones.................................................................................... 54  
Figure 21 The optimization workflow [1].................................................................................................. 56  
Figure 22 The loosely-coupled interface between Dakota and a simulation code [7]............................... 57  
Figure 23 The dimensions of the computational domain ........................................................................... 58
Figure 24 An example of computational mesh ................................................................. 59
Figure 25 Boundary conditions ...................................................................................... 59
Figure 26 Depiction of cyclicAMI boundary condition ................................................. 60
Figure 27 Temperature field of one of the optimal solutions ........................................ 61
Figure 28 Pressure field of one of the optimal solutions ................................................ 61
Figure 29 Domain for the strong constraints [1] ............................................................. 63
Figure 30 A comparison of different number of generations, each generation has 40 members........ 64
Figure 31 A comparison of different number of generations, each generation has 80 members........ 65
Figure 32 A comparison of two cases with approximately the same number of evaluations .... 66
Figure 33 A comparison of two cases with approximately the same number of evaluations .... 67
Figure 34 A comparison of different mutation rates for strong and relaxed constraints .......... 68
Figure 35 Pareto fronts for relaxed and strong constraints ............................................. 69
Figure 36 A comparison of different crossover rates ..................................................... 70
Figure 37 Custom design 1 – temperature and pressure field ....................................... 71
Figure 38 Custom design 2 – temperature and pressure field ....................................... 71
Figure 39 Custom design 3 – temperature and pressure field ....................................... 72
Figure 40 Pareto front and three points corresponding to three custom designs .......... 73
Figure 41 The black triangle denotes Design 1 point on the Pareto front ......................... 74
Figure 42 Temperature field of Design 1 ..................................................................... 74
Figure 43 Pressure field of Design 1 ............................................................................. 75
Figure 44 The normal distribution of $x_{137T}$ ................................................................ 75
Figure 45 The normal distribution of $y_{137T}$ ................................................................ 76
Figure 46 Cumulative distribution function for pressure drop (Design 1) ..................... 77
Figure 47 Relative density function for pressure drop (method: Latin Hypercube sampling) 79
Figure 48 Relative density function for pressure drop (method: Monte Carlo sampling) .... 79
Figure 49 Distribution of the uncertain variable, $T_{wall}$ ............................................... 80
Figure 50 Cumulative distribution function for temperature increase (Design 1) ............ 81
Figure 51 Density value for temperature increase ....................................................... 81
Figure 52 The green triangle denotes Design 2 point and the red triangle denotes Design 3 on the Pareto front ........................................................................ 82
Figure 53 Temperature field of Design 2 ..................................................................... 82
Figure 54 Pressure field of Design 2 ............................................................................. 83
Figure 55 Temperature field of Design 3 ..................................................................... 83
Figure 56 Pressure field of Design 3 ............................................................................. 83
Figure 57 Cumulative distribution function for pressure drop (Design 2) ..................... 84
Figure 58  Cumulative distribution function for pressure drop (Design 3).................................84
Figure 59  Relative density function for pressure drop (method: Latin Hypercube sampling, Design 2) .85
Figure 60  Relative density function for pressure drop (method: LHS, Design 3) ..........................86
Figure 61  Relative density function for pressure drop with finer mesh (10000 cells) for Design 3 .......86
Figure 62  Cumulative distribution function for temperature increase (Design 2).............................87
Figure 63  Cumulative distribution function for temperature increase (Design 3) ............................88
Figure 64  Density value for temperature increase (Design 2) ..........................................................88
Figure 65  Density value for temperature increase (Design 3) ..........................................................89
Figure 66  The positions of the three designs from [1] on the corresponding Pareto fronts.................90
Figure 67  Temperature and pressure field of Design 1 .................................................................91
Figure 68  Temperature and pressure field of Design 2 .................................................................91
Figure 69  Temperature and pressure field of Design 3 .................................................................92
Nomenclature

\[ E_i \quad \text{possible event} \]
\[ E[Y] \quad \text{expected value of a function Y} \]
\[ f_X(x) \quad \text{probability density function} \]
\[ g(x) \quad \text{the limit state function} \]
\[ g(X) \quad \text{uncertain response of an uncertain variable X} \]
\[ N_A \quad \text{number of experiments where A occurred} \]
\[ n_{exp} \quad \text{total number of experiments} \]
\[ p \quad \text{pressure field} \]
\[ P(A) \quad \text{probability of an event A} \]
\[ P_X(x) \quad \text{cumulative distribution function} \]
\[ r_{XY} \quad \text{sample correlation coefficient} \]
\[ s_{XY} \quad \text{sample covariance} \]
\[ s^2 \quad \text{sample variance} \]
\[ u_i, u_j \quad \text{velocity fields} \]
\[ V_X \quad \text{coefficient of variation} \]
\[ w_i \quad \text{weighting coefficient} \]
\[ z \quad \text{response level} \]

\[ \alpha \quad \text{sensitivity factor} \]
\[ \beta \quad \text{reliability index, coefficient of volumetric expansion} \]
\[ \Delta T \quad \text{temperature increase} \]
\[ \Delta p \quad \text{pressure drop} \]
\[ \varepsilon \quad \text{statistical error} \]
\[ \eta \quad \text{sample coefficient of skewness} \]
\[ \kappa \quad \text{sample coefficient of kurtosis} \]
\[ \lambda_i \quad i^{th} \text{ moment of a continuous random variable} \]
\[ \mu_X \quad \text{mean (or expected value) of a continuous random variable} \]
\[ \rho \quad \text{fluid density} \]
\[ \sigma^2 \quad \text{variance of a continuous random variable} \]
\[ \sigma_X \quad \text{standard deviation of a continuous random variable} \]
Abbreviations

CFD - Computation Fluid Dynamics
MOGA - Multi-objective Genetic Algorithm
SOGA - Single-objective Genetic Algorithm
NBI - Normal Boundary Intersection
GA - Genetic Algorithm
OF - Objective Function
EA - Evolutionary Algorithm
CPU - Central Processing Unit
DACE - Design and analysis of computer experiments
LP - Linear Programming
NLP - Nonlinear Programming
SQP - Sequential Quadratic Programming
DIRECT - Division of rectangles
JEGA - John Eddy's Evolutionary Algorithm
CAD - Computer Aided Design
OpenFOAM - Open Field Operation and Manipulation
OUU – Optimization under uncertainty
UQ – Uncertainty Quantification
MPP – Most probable point
FORM – First Order Reliability Method
SORM – Second Order Reliability Method
SBO – Surrogate-Based Optimization
RBDO – Reliability-based Design Optimization
LHS – Latin Hypercube Sampling
MC – Monte Carlo Sampling
CDF – Cumulative Density Function
Abstract

Optimization is a discipline of numerical mathematics which aims to improve the operation of a system or process as good as possible in some defined sense. Optimization algorithms work to minimize (or maximize) an objective function, typically calculated by the user simulation code, subject to constraints on design variables and responses.

Parameters and settings of tube bank heat exchanger multi-objective optimization are described. The problem consists of finding the best locations of the tubes to increase heat exchange (i.e. the temperature increase of the fluid) while at the same time limit the pressure loss. The two corresponding numerical parameters to optimize are the average temperature difference $\Delta T$ and pressure drop $\Delta p$ between inflow and outflow.

The set of coupled numerical tools to solve the multi-objective optimization consists of open source optimization software Dakota, open source CFD toolbox OpenFOAM and open source software for geometry creation Salome. Multi-objective Genetic Algorithm is used to obtain optimal designs.

Results of the optimization process are presented on corresponding Pareto fronts. Different parameters of the Multi-objective Genetic Algorithm are examined and discussed.

Uncertainty quantification or nondeterministic analysis is the process of characterizing input uncertainties, forward propagating these uncertainties through a computational model, and performing statistical or interval assessments on the resulting responses. Uncertainty quantification is related to sensitivity analysis in that the common goal is to gain an understanding of how variations in the parameters affect the response functions of the engineering design problem. For uncertainty quantification, some or all of the components of the parameter vector, are considered to be uncertain as specified by particular probability distributions.

Effect of uncertain tube coordinates and uncertain tubes’ temperature is examined and presented in graphs and histograms.

Three optimal points from [1] are chosen and their robustness evaluated by means of the uncertainty quantification methods.

Key words: optimization under uncertainty, constraints, multi-objective optimization, computational fluid dynamics
**Sažetak**

Optimizacija je disciplina numeričke matematike za unaprijeđivanje rada sustava ili procesa da bi rezultati bili najbolji mogući. Optimizacijski algoritmi za zadaću imaju minimalizaciju ili maksimizaciju funkcije cilja koja se proračunava simulacijom, uz odgovarajuća ograničenja varijabli.

U radu su opisani parametri i postavke optimizacije po višestrukim funkcijama cilja sa ograničenjima varijabli cijevnog izmjenjivača topline. Traže se optimalne pozicije cijevi kako bi se povećao prirast temperature fluida uz što manji pad tlaka u izmjenjivaču. Temperaturni prirast i pad tlaka su ujedno i funkcije cilja za ovaj problem.

Za optimizaciju su korištena tri računalna alata: open source alat Salome za konstrukciju geometrije, open source paket za računalnu dinamiku fluida OpenFOAM i open source software za optimizaciju Dakota.

Ispitivani su različiti parametri genetskog algoritma za optimizaciju po višestrukim funkcijama cilja te su rezultati prikazani preko Pareto linija.

Provedena je i kvantifikacija nesigurnosti varijabli – ispitivano je kako male promjene u varijablama, koje su zadane normalnom raspodjelom, utječu na promatrani sustav, odnosno funkcije cilja.

Utjecaj nesigurnih koordinata jedne cijevi i nesigurne temperature stijenki cijevi prikazan je kumulativnim distribucijama i histogramima.

**Ključne riječi:** optimizacija, nesigurnost varijabli, višestruke funkcije cilja
**Prošireni sažetak**

Optimizacija je disciplina numeričke matematike za unaprijeđivanje rada sustava ili procesa da bi rezultati bili najbolji mogući. Optimizacijski algoritmi za zadaću imaju minimizaciju ili maksimizaciju funkcije cilja koja se proračunava simulacijom, uz odgovarajuća ograničenja varijabli.

Općenito, problem optimizacije se može definirati:

minimizirati: \[ f(x) \]

\[ x \in \mathbb{R}^n \]

ograničenja:

\[ g_L \leq g(x) \leq g_U \]
\[ h(x) = h_t \]
\[ a_L \leq A_i x \leq a_U \]
\[ A_e x = a_t \]

\[ x_L \leq x \leq x_U \quad (1) \]

U ovoj formulaciji, \( x = [x_1, x_2, \ldots, x_n] \) je vektor realnih varijabli ili parametara konstrukcije. Vektori \( x_L \) i \( x_U \) predstavljaju donja i gornja ograničenja parametara konstrukcije. Ograničenja definiraju dozvoljeno područje vrijednosti koje mogu poprimiti parametri. Cilj optimizacije je minimizacija funkcije \( f(x) \) tako da se zadovolje ograničenja. Ograničenja mogu biti linearna ili nelinearna, sa uvjetima nejednakosti ili jednakosti. Nelinearna ograničenja \( g(x) \) s nejednakosti, imaju definirane i gornje i donje vrijednosti. Nelinearna ograničenja \( h(x) \) s jednakosti imaju zadana ciljana vrijednost \( h_t \). Nelinearna ograničenja s nejednačću tvore sustav jednakboja \( A_i x \), također imaju donje i gornje zadane vrijednosti, \( a_L \) i \( a_U \). \( A_i \) je matrica koeficijenata za linearni sustav. Linearna ograničenja s jednakboću također tvore sustav jednakboja \( A_e x \), gdje je \( A_e \) matrica koeficijenata, a zadana ciljana vrijednost jednaka je \( a_t \).

Metode optimizacije mogu se podijeliti na gradijentne, nigradijentne i surogat metode.

Optimizacija sa nesigurnošću varijabli često je potrebna u inženjerskim problemima kada nesigurna vrijednost varijabli može utjecati na funkcije cilja i u konačnici, na rad sustava. Za kvantificiranje nesigurnosti koriste se metode uzorkovanja i gradijentne metode. Metode uzorkovanja generiraju skupove ulaznih varijabli koje
odgovaraju raspodjeli koju zadaje korisnik, procjenjuju vrijednost funkcije cilja te
smještaju konstrukciju u odgovarajući interval vrijednosti funkcije cilja. Za funkciju
cilja računaju se srednja vrijednost, standardna devijacija, intervali pouzdanosti te se
dobiva kumulativna raspodjela vjerojatnosti.

Gradijentne metode nazivaju se i metodama za procjenu pouzdanosti, a njihova
upotreba je uglavnom računalno manje zahtjevna od upotrebe metoda uzorkovanja. Ove
su metode bolje i u računanju statističkih podataka za događaje manje vjerojatnosti.

„Za zadani skup nesigurnih ulaznih varijabli, \(X\), i zadane realne vrijednosti
mogućih odgovora sustava, \(g\), koja je vjerojatnost da će funkcija cilja biti manja ili
jednaka nekoj vrijednosti, \(z\)?” To se pitanje matematički može zapisati: 
\[
P[g(X) \leq z] = F_g(z),
\]
dje \(F_g(z)\) kumulativna raspodjela vjerojatnosti nesigurnog odgovora
(vrijednosti funkcije cilja) \(g(X)\).

Optimizacija po višestrukim funkcijama cilja koristi se kod problema kod kojih
postoje dvije ili više funkcija cilja koje se simultano žele optimirati. Uobičajeno su te
funkcije cilja suprotno standardno, odnosno poboljšanje vrijednosti jedne funkcije znači
pogoršanje vrijednosti druge. Zato ne postoji jedinstveno optimalno rješenje, već skup
rješenja koja se mogu prikazati u prostoru funkcija cilja. Tada su optimalna rješenja
poredana u tzv. Pareto liniju (slika 1). Sva su Pareto rješenja ravnopravna i ni jedno nije
bolje od drugih sve dok ne postoji vanjski subjektivni kriterij za ocjenu.
Za dobivanje Pareto rješenja, koristi se evolucijski algoritam koji se temelji na Darwinovim postavkama teorije evolucije. „Geni“ (karakteristike) mogućih konstrukcijskih rješenja križaju se i mutiraju. Takvim postupcima dobivaju se nova rješenja, čije se performanse uspoređuju s performansama drugih konstrukcijskih rješenja. Početna rješenja određuju u inicijalnoj populaciji sa određenim brojem članova, od kojih se zatim dio križa ili mutira određenom brzinom te nastaje nova populacija. U novoj populaciji preživljavaju samo dominantni članovi, tj. oni čije su performanse bolje. Tako se, kroz određeni broj populacija, određuju najjači članovi – konstrukcije čije su se performanse pokazale najboljima.

U radu je optimiran cijevni izmjenjivač topline korištenjem open source softwarea za generiranje geometrije Salome, open source alata za računalnu dinamiku fluida OpenFOAM te open source softvera za optimizaciju Dakota. Na slici 2 prikazana je domena proračuna. Domena je podijeljena na tri dijela, prvi dio je ulaz, u drugom dijelu su smještene cijevi izmjenjivača, a treći dio je izlaz. Cijevi su smještene u zasebnom odjeljku kako bi se spriječila interakcija s rubnim uvjetima. Koordinate središta cijevi su ograničene da bi se spriječili sudari i preklapanja – minimalna udaljenost između središta je 21 mm, što odgovara promjeru cijevi (20 mm) i veličini jedne najmanje ćelije računalne mreže koja stane između stijenki cijevi. Parametri optimizacije su koordinate središta cijevi, a funkcije cilja su pad tlaka (minimizacija) i povećanje temperature (maksimizacija). Veća izmjena topline uzrokovat će i veći pad tlaka, pa su ta dva cilja suprotstavljena. Za optimizaciju se stoga koristi MOGA, optimizacija po višestrukim funkcijama cilja.

**Slika 2** Domena proračuna
Zrak ulazi u domenu s temperaturom $T_{\text{inlet}} = 293$ K i zagrijava se prolazeći između cijevi. Temperatura stijenki cijevi je konstantna, $T_{\text{wall}} = 353$ K. Reynoldsov broj jednak je 14 kako bi se osiguralo laminarno i stacionarno strujanje, a definiran je za dimenziju promjera cijevi i ulaznu brzinu $v_{\text{inlet}} = 0.01$ m/s. Rubni uvjeti gornje i donje stijenke postavljeni su kao cyclicAMI jer dozvoljavaju različit broj čelija. Tlak na izlazu jednak je atmosferskom tlaku.

Jednadžbe koje opisuju sustav rješavaju se u OpenFOAM alatu:

Jednadžba kontinuiteta

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u_i)}{\partial x_i} = 0$$  

(2)

Jednadžba količine gibanja

$$\frac{\partial (\rho u_i)}{\partial t} + \frac{\partial (\rho u_i u_j)}{\partial x_j} = \rho g_i - \frac{\partial}{\partial x_i} \left( p + \frac{2}{3} \mu \frac{\partial u_j}{\partial x_j} \right) + \frac{\partial}{\partial x_j} \left[ \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \rho \right]$$  

(3)

Energijska jednadžba

$$\frac{\partial (\rho e)}{\partial t} + \frac{\partial (\rho u_i u_i)}{\partial x_i} = \rho g_i u_i - \frac{\partial (\rho u_i)}{\partial x_i} \left( p + \frac{2}{3} \mu \frac{\partial u_j}{\partial x_j} u_i \right) + \frac{\partial}{\partial x_i} \left[ \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) u_i \right] + \frac{\partial}{\partial x_j} \left[ \lambda \left( \frac{\partial T}{\partial x_i} + \frac{\partial T}{\partial x_j} \right) \right] + \rho Q$$  

(4)

Evaluacija jednog skupa parametara (pozicija cijevi) uključuje sljedeće korake:

1. generiranje geometrije (cijevi) u softwareu Salome, koji koristi zadane pozicije cijevi iz programa za optimizaciju Dakota
2. generiranje odgovarajuće domene i računalne mreže u snappyHexMeshu i blockMeshu
3. CFD simulacija, tj. rješavanje odgovarajućih jednadžbi alatom OpenFOAM
4. izračun vrijednosti funkcija cilja (pad tlaka, povećanje temperature) koristeći swak4Foam
U prvom je dijelu rezultata prikazan utjecaj različitog broja generacija i veličine populacije na izgled Pareto linije.

Slika 3 Usporedba različitog broja generacija uz konstantu veličinu populacije (40 članova)

Veći broj generacija dovodi do bolje statističke konvergencije, što je očekivano jer je proveden veći broj evaluacija (broj evaluacija približno je jednak umnošku broja generacija i veličine populacije), kao što je pokazano na slici 3. 20 generacija dalo je manje padove tlaka za iste odgovarajuće promjene temperature, u usporedbi s 5 i 10 generacija. 15 generacija pokazalo je rezultate vrlo sličnima onima dobivenim za 20 generacija, pa se može zaključiti da je optimizacija s 15 generacija efikasnija jer je uz manje evaluacija došla do istih rezultata. Također, većim brojem generacija pokriven je širi raspon promjena temperature i tlaka, ali neka su područja rijetko pokrivena.

Na slici 4 prikazana je usporedba približno istog broja evaluacija, ali različiti brojevi generacija i veličina populacije. Veći broj generacija pokazuje bolje rezultate, bolja je pokrivenost širokog raspona temperaturu, ali postoje i rijetko pokrivena područja. U nekim dijelovima bolji su rezultati većeg broja generacija, a udrugima je obrnuto. Može se zaključiti da ni jedna Pareto linija nije u potpunosti statistički konvergirala.
Slika 4 Usporedba MOGA metode s približno istim brojem evaluacija

Također, uspoređivane su različite stope mutacije, slika 5. Stopa mutacije jednaka 1 znači da je svaki član nove generacije mutirao, a stopa mutacije 0,75 znači da je 75% nove generacije mutiralo. Pareto linija s većom stopom mutacije je šira i bolje konvergirala, što je očekivano jer mutacije dovode do većih promjena u konstrukciji (pozicije cijevi) i nude više opcija za optimalno rješenje.

Slika 5 Usporedba različitih stopa mutacije
Na sljedećoj je slici prikazana usporedba različitih stopa križanja. Ukupni broj križanja jednak je umnošku stope križanja i broja članova u populaciji. Veća stopa križanja dovodi do šire Pareto linije koja pokriva veći raspon padova tlaka i promjene temperature, ali je potreban i veći broj evaluacija za statističku konvergenciju, слиka 6.

Slika 6 Usporedba različitih stopa križanja


Polja temperature i tlaka za konstrukciju prikazana su na slikama 7 i 8.
Koordinate prve cijevi s lijeva su $x_1 = 0,06694869$ m i $y_1 = 0,03221098$ m. Odgovarajući pad tlaka iznosi $\Delta p = 0,00013923$ m$^2$/s$^2$, a povećanje temperature je $\Delta T = 28,0194$ K.

Uvodi se normalna raspodjela koordinata prve cijevi, srednja vrijednost jednaka je koordinatama originalne konstrukcije, a standardna devijacija iznosi 0.002 m. Raspodjele su prikazane na slikama 9 i 10. Funkcija cilja je pad tlaka.
Slika 9  Normalna raspodjela nesigurne koordinate $x_1$

Slika 10  Normalna raspodjela nesigurne koordinate $y_1$

Na slici 11 prikazana je kumulativna raspodjela vjerojatnosti za pad tlaka.
Slika 11  Kumulativna raspodjela vjerojatnosti za pad tlaka

Npr. prema metodi uzorkovanja Latin Hypercube, vjerojatnost da je tlak manji od 0,00014 m²/s², jednaka je 67%.

Latin Hypercube i Monte Carlo metode dale su slične rezultate, a obje su koristile 200 uzoraka. Funkcije nisu glatke, ali pokazuju dobar trend kumulativne funkcije. Mean Value je dala glatku krivulju, ali postoji velika razlika između ove metode i metoda uzorkovanja, osim u repovima i sredini distribucije, gdje su rezultati slični. Daljnja analiza je pokazala da je za slaganje sa metodama uzorkovanja potrebno povećati relativni korak.

Slika 12  Histogram za relativnu gustoću intervala pada tlaka
Na slici 12 prikazan je histogram koji pokazuje relativnu gustoću intervala funkcije cilja. Može se uočiti da je raspodjela približna normalnoj, ali blago nagnuta. Izračunata srednja vrijednost jednaka je \( \mu_{\Delta p} = 0,0001395 \, \text{m}^2/\text{s}^2 \), što je očekivano jer je pad tlaka za originalnu konstrukciju jednak 0,00013923 \, \text{m}^2/\text{s}^2. Standardna devijacija jednaka je \( \sigma = 0.0000017 \, \text{m}^2/\text{s}^2 \). Ova se konstrukcija može proglasiti relativno stabilnom zbog uzdignute normalne raspodjele funkcije cilja (histogram, slika 12) i relativno strme kumulativne raspodjele.

Ostale procjene i detaljni prikazi dani su u nastavku Diplomskog rada.
1. Introduction

In general, optimization theory is a body of mathematical results and numerical methods for finding and identifying the best candidate from a collection of alternatives without having to explicitly evaluate all possible alternatives [2]. To apply the mathematical results and numerical techniques of optimization theory to concrete engineering problems, it is necessary to define the boundaries of the engineering system to be optimized, to define the quantitative criterion on the basis of which candidates will be ranked to determine the best, to define a model that will express the manner in which the variables are related.

Regardless of the performance criterion selected, in the context of optimization the best solution will be the one with the minimum or maximum value of the performance index. In many practical solutions, it is desirable to achieve a solution that is the best with respect to a number of different criteria, but this is often not possible. The following sections will provide additional information about this problem and different approaches will be described.

Uncertainty quantification tries to determine how likely certain outcomes are if some aspects of the system are not exactly known, [3]. Many problems in natural sciences and engineering are associated with sources of uncertainty. For example, parametric variability: the dimensions of a work piece in a process of manufacture may not be exactly as designed and instructed, which would cause variability in its performance.

Uncertainty propagation is the quantification of the uncertainties in system output(s) propagated from the uncertain inputs. The targets of uncertainty propagation analysis can be to evaluate low order moments of the outputs, i.e. mean value and standard deviation, and evaluate the reliability of the outputs. The latter is especially useful in reliability engineering where outputs of a system are usually closely related to the performance of the system.

In the following chapters, methods for uncertainty quantification will be described and results of the uncertainty analysis presented.
2. Probability

Probability theory forms the basis of the assessment of probabilities of occurrence of uncertain events and thus constitutes a cornerstone in the risk and decision analysis. Only when a consistent basis has been established for the treatment of the uncertainties influencing the probability that events with possible adverse consequences may occur, it is possible to assess the risks associated with a given activity and thus to establish a rational basis for decision making. The level of uncertainty associated with a considered activity or phenomenon may be expressed by means of purely qualitative statements but may also be quantified in terms of numbers or percentages.

Basics of probability theory will be introduced in this chapter, [4].

2.1. Definition of Probability

2.1.1. Frequentistic Definition

The frequentistic definition of probability is the typical interpretation of probability by an experimentalist. In this interpretation the probability \( P(A) \) is simply the relative frequency of occurrence of the event \( A \) as observed in an experiment with \( n \) trials, i.e. the probability of an event \( A \) is defined as the number of times that the event \( A \) occurs divided by the number of experiments that are carried out:

\[
P(A) = \lim_{n_{\text{exp}} \to \infty} \frac{N_A}{n_{\text{exp}}}
\]  

(2.1)

\( N_A \) is number of experiments where \( A \) occurred, \( n_{\text{exp}} \) is the total number of experiments.

If a frequentist is asked what the probability of achieving a “head” when flipping a coin is, he would principally not know what to answer until performing a large number of experiments. In the mind of a frequentist, probability is a characteristic of nature.

2.1.2. Classical Definition

The classical probability definition originates from the days when probability calculus was founded by Pascal and Fermat. The inspiration for this theory can be found
in games: cards or dice. The classical definition of the probability of the event A can be formulated as:

\[
P(A) = \frac{n_A}{n_{tot}}
\]  

(2.2)

where \(n_A\) is the number of equally likely ways by which an experiment may lead to A, \(n_{tot}\) is the total number of equally likely ways in the experiment.

There is no real contradiction to the frequentistic definition, but the following differences may be observed:

- The experiment does not need to be carried out as the answer is known in advance. For example, according to the classical definition of probability, the probability of achieving a “head” when flipping a coin would be 0.5, as there is only one possible way to achieve a “head” and there are two likely outcomes of the experiment.
- The classical theory gives no solution unless all equally possible ways can be derived analytically.

2.1.3. Bayesian Definition

In the Bayesian interpretation the probability \(P(A)\) of the event A is formulated as a degree of belief that A will occur:

\[
P(A) = \text{degree of belief that A will occur}
\]  

(2.3)

The degree of belief is a reflection of the state of mind of individual person in terms of experience, expertise and preferences. The Bayesian interpretation of probability is subjective or, more precise, person-dependant. It includes frequentistic and classical interpretation in the sense that the subjectively assigned probabilities may be based on experience from previous experiments (frequentistic) as well as considerations of e.g. symmetry in coin-flipping problem (classical).
2.2. Sample Space and Events

The set of all possible outcomes of an experiment is called the sample space (denoted $\Omega$). An event is defined as a subset of a sample space and thus a set of sample points. If the subset is empty (i.e., contains no sample points) it is said to be impossible. An event is said to be certain if it contains all sample points in the sample space (i.e., the event is identical to the sample space).

Consider the events $E_1$ and $E_2$ shown in Figure 1. The subset of sample points belonging to the event $E_1$ or $E_2$ is denoted as the union of the events $E_1$ and $E_2$ written as $E_1 \cup E_2$.

![Venn diagrams](image)

Figure 1  Venn diagrams illustrating the union of events (left) and the intersection of events (right) [4]

The subset of sample points belonging to $E_1$ and $E_2$ is called the intersection of $E_1$ and $E_2$ and is written as $E_1 \cap E_2$. The intersection of these two events is illustrated in the right portion of Fig 1.

The two events are said to be mutually exclusive if they are disjoint (i.e., if they have no common sample points). In this case, the intersection of $E_1$ and $E_2$ is empty ($E_1 \cap E_2 = 0$).

2.3. The Three Axioms of Probability Theory

Probability theory is based on the following three axioms:

Axiom 1

Probability of the event $E$ lies between 0 and 1, when probability is equal to 0, the event will never happen, and for 1, the event $E$ is the only outcome.
$0 \leq P(E) \leq 1$ \hspace{1cm} (2.4)

for any given event $E \subset \Omega$ where $P$ is the probability measure, $\Omega$ is the sample space and $E$ is defined as a subset of a sample space.

Axiom 2

Probability of all events, i.e. probability of the entire sample space, is equal to 1.

$$P(\Omega) = 1$$ \hspace{1cm} (2.5)

where $\Omega$ is the sample space.

Axiom 3

Given that $E_1, E_2 \ldots$ is a sequence of mutually exclusive events (i.e. $E_1 \cap E_2 = 0$ etc.) then:

$$P \left( \bigcup_{i=1}^{\infty} E_i \right) = \sum_{i=1}^{\infty} P(E_i)$$ \hspace{1cm} (2.6)

These three axioms form the sole basis of the theory of probability.

2.4. Conditional Probability

Conditional probabilities are of special interest in risk and reliability analysis as they form the basis of the updating of probability estimates based on new information, knowledge and evidence.

The conditional probability of the event $E_1$ given that the event $E_2$ has occurred is written as:

$$P(E_1|E_2) = \frac{P(E_1 \cap E_2)}{P(E_2)}$$ \hspace{1cm} (2.7)

It is seen that the conditional probability is not defined if the conditioning event is the empty set ($P(E_2) = 0$).

The event $E_1$ is said to be probabilistically independent of the event $E_2$ if:

$$P(E_1|E_2) = P(E_1)$$ \hspace{1cm} (2.8)
implying that the occurrence of the event $E_2$ does not affect the probability of $E_1$. From Eq. (2.7) the probability of the event $E_1 \cap E_2$ may be given as:

$$P(E_1 \cap E_2) = P(E_1 | E_2) P(E_2)$$  \hspace{1cm} (2.9)

and it follows immediately that if the events $E_1$ and $E_2$ are independent, then:

$$P(E_1 \cap E_2) = P(E_1) P(E_2)$$  \hspace{1cm} (2.10)

### 2.5. Characteristics of the Data

In this section, the numerical characteristics of the observed data containing important information about the data and the nature of uncertainty associated with these. These are also referred to as sample characteristics. Descriptive statistics play an important role in engineering as a standardized basis for assessing and documenting data obtained for the purpose of understanding and representing uncertainties.

#### 2.5.1. Central Measures

One of the most useful numerical summaries is the sample mean. If the data set is collected in the vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ the sample mean $\bar{x}$ is simply given as:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$$  \hspace{1cm} (2.11)

The sample mean may be interpreted as a central value of the data set. if, on the basis of the data set, one should give only one value characterizing the data, the sample mean would normally be used.

Another central measure is the mode of the data set, i.e. the most frequently occurring value in the data set. When data samples are real values, the mode in general cannot be assessed numerically, but may be assessed from graphical representations of the data.

It is often convenient to work with an ordered data set which is readily established by rearranging the original data set $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ such that the data are arranged in increasing order as $x_1^0 \leq x_2^0 \leq \cdots \leq x_i^0 \leq \cdots \leq x_{n-1}^0 \leq x_n^0$. 
The median of the data set is defined as the middle value in the ordered list of data if \( n \) is odd. If \( n \) is even, the median is taken as the average value of the two middle values.

### 2.5.2. Dispersion Measures

The variability or the dispersion of the data set around the sample mean is also an important characteristic of the data set. The dispersion may be characterized by the sample variance \( s^2 \) given by:

\[
s^2 = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})^2
\]

(2.12)

The sample standard deviation \( s \) is defined as the square root of the sample variance. From Eq. (2.12) it is seen that the sample standard deviation is assessed in terms of the variability of the observations around the sample mean value \( \bar{x} \).

### 2.5.3. Other Measures

Whereas the sample mean, mode and median are central measures of a data set and the sample variance is a measure of the dispersion around the sample mean, it is also useful to have some characteristics indicating the degree of symmetry of the data set. To this end, the sample coefficient of skewness (Figure 2), which is a simple logical extension of the sample variance is suitable. The sample coefficient of skewness \( \eta \) is defined as:

\[
\eta = \frac{1}{n} \frac{\sum_{i=1}^{n} (x_i - \bar{x})^3}{s^3}
\]

(2.13)

This coefficient is positive if the mode of the data set is less than its mean value (skewed to the right) and negative if the mode is larger than the mean value (skewed to the left).
In a similar way, the sample coefficient of kurtosis $\kappa$ (Figure 3) is defined as:

$$\kappa = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})^4$$  \hspace{1cm} (2.14)

which is a measure of how closely the data are distributed around the mode (peakedness).

2.5.4. Measures of Correlation

Observations are often made of two characteristics simultaneously as shown in Figure 4 where pairs of data observed simultaneously are plotted jointly along the x-axis.
and the y-axis (two-dimensional scatter diagram). A measure of correlation between the observed data sets, the sample covariance $s_{XY}$ is useful, and defined as:

$$s_{XY} = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})$$

(2.15)

The sample covariance has the property that, if there is a tendency in the data set that the values of $x_i$ and $y_i$ are both higher than $\bar{x}$ and $\bar{y}$ at the same time, and the trend is linear, than most of the terms in the sum will be positive and the sample covariance will be positive. The opposite trend will result in a negative sample covariance. Such behavior is referred to as correlation.

![Figure 4 Two examples of paired data sets [1]](image)

In the scatter diagram to the left in Figure 4 there is only little correlation between the observed data pairs whereas the opposite is evident in the example to the right.

The sample covariance may be normalized with respect to the sample standard deviations of the individual data sets $s_X$ and $s_Y$ and the result is called the sample correlation coefficient $r_{XY}$ defined as:

$$r_{XY} = \frac{1}{n} \sum_{i=1}^{n} \frac{(x_i - \bar{x})(y_i - \bar{y})}{s_X s_Y}$$

(2.16)

The sample correlation coefficient is limited to the interval $-1 \leq r_{XY} \leq 1$ and the extreme values of the interval are only achieved in case the data pairs are perfectly
correlated, implying that the points on the scatter diagram lie on a straight line. For the example shown in Fig. 4 there is almost zero correlation at the left hand side and almost full positive correlation on the right hand side.

2.6. Uncertainty Modeling

For the purpose of discussing the phenomenon uncertainty in more detail, it is assumed that the universe is deterministic and that the knowledge about the universe is perfect. This implies that it is possible to achieve perfect knowledge about any state, quantity or characteristic, by means of e.g. a set of exact equation systems and known boundary conditions. It would be possible to gain knowledge about phenomena which cannot be directly observed or have not taken place yet. In principle, following this line of reasoning, the future as well as the past would be known or assessable with certainty.

Whether the universe is deterministic or not is a rather deep philosophical and physical question. Despite the obviously challenging aspects of this question its answer is, however, not a prerequisite for purposes of engineering decision making because even if the universe is deterministic, our knowledge about it is still in part incomplete and/or uncertain.

It has become standard to differentiate between uncertainties due to inherent natural variability, model uncertainties and statistical uncertainties. The first mentioned type of uncertainty is often denoted as aleatory uncertainty and the two latter types are referred to as epistemic uncertainties.

2.6.1. Random Variables

The performance of an engineered system, facility or installation may usually be modeled in mathematical and physical terms in conjunction with empirical relations.

For a given set of model parameters the performance of the considered system can be determined on the basis of this model. The basic random variables are defined as the parameter that represents the available knowledge as well as the associated uncertainty in the considered model.

The basic random variables must be able to represent all types of uncertainties that are included in the analysis. The uncertainties which must be considered are physical uncertainty, statistical uncertainty and the model uncertainty. The physical uncertainties are typically uncertainties associated with the environment, the geometry
of the structure and the material properties. The statistical uncertainties arise due to incomplete statistical information e.g. due to a small number of material tests. Finally, the model uncertainties must be considered to take into account the uncertainty associated with the idealized mathematical descriptions used to approximate the actual physical behavior of the structure.

Modern methods of reliability and risk analysis allow for a very general representation of these uncertainties ranging from non-stationary stochastic processes and fields to time invariant random variables. In most cases, it is sufficient to model the uncertain quantities by random variables with given cumulative distribution functions and distribution parameters estimated on basis of statistical and/or subjective information. Therefore, the following is concerned with a basic description of the characteristics of random variables.

**Cumulative Distribution and Probability Density Functions**

A random variable, which can take any real value, is called a continuous random variable. The probability that such random variable takes a specific value is zero, as there is an infinite number of values in real space. The probability that a continuous random variable, $X$, is less than or equal to a value, $x$, is given by the cumulative distribution function:

$$F_X(x) = P(X \leq x) \quad (2.17)$$

In this section, capital letters denote a random variable and lowercase letters denote an outcome or realization of a random variable. An example of a continuous cumulative distribution function is illustrated in Figure 5.A.

For continuous random variables the probability density function is given by:

$$f_X(x) = \frac{dF_X(x)}{dx} \quad (2.18)$$

An example of a continuous probability density function is illustrated in Figure 5.B.

The probability of an outcome in the interval $[x; x + dx]$, where $dx$ is small, is given by $P(X \in [x; x + dx]) = f_X(x)dx$. 

Random variables with a finite or infinite countable sample space are called discrete random variables. For discrete random variables the cumulative distribution function is given as:

\[ P_X(x) = \sum_{x_i < x} p_X(x_i) \]  \hspace{1cm} (2.19)

where \( p_X(x_i) \) is the probability density function given as:

\[ p_X(x_i) = P(X = x_i) \]  \hspace{1cm} (2.20)

A discrete cumulative distribution function and probability density function is illustrated in Figure 6.

![Figure 5 Illustration of A) a cumulative distribution function and B) a probability density function for a continuous random variable [4]](image)

![Figure 6 Illustration of A) a cumulative distribution function and B) a probability density function for a discrete random variable [4]](image)
Moments of Continuous Random Variable and the Expectation Operator

Probability distributions may be defined in terms of their parameters or moments. Often cumulative distribution functions and probability density functions are written as $F_X(x; p)$ and $f_X(x; p)$ respectively to indicate the parameters $p$ (or moments) defining the functions. Whether the cumulative distribution and density function are defined by their moments or by their parameters is a matter of convenience and it is generally possible to establish one from another.

The $i$th moment $\lambda_i$ of a continuous random variable is defined by:

$$\lambda_i = \int_{-\infty}^{\infty} x^i f_X(x) dx$$

(2.21)

The mean (or expected value) of a continuous random variable $X$ is defined as the first moment:

$$\mu_X = E[X] = \int_{-\infty}^{\infty} x f_X(x) dx$$

(2.22)

where $E[\ ]$ is the expectation operation. Its linearity property is useful since it can be used, for example, to find the following formula for variance of a random variable $X$ in terms of more easily calculated quantities:

$$\sigma_X^2 = E[(X - \mu_X)^2] = E[(X^2 - 2\mu_X \cdot X + \mu_X^2)]$$

$$= \mu_X^2 + E[X^2] - 2\mu_X \cdot E[X] = \mu_X^2 + E[X^2] - 2\mu_X^2$$

(2.23)

The variance $\sigma_X^2$, is described by the second central moment, i.e. for continuous random variables it is:

$$\sigma_X^2 = Var[X] = E[(X - \mu_X)^2] = \int_{-\infty}^{\infty} (x - \mu_X)^2 f_X(x) dx$$

(2.24)

where $Var[X]$ denotes the variance of $X$.

The ratio between the standard deviation $\sigma_X$ and the expected value $\mu_X$ of a random variable $X$ is denoted the coefficient of variation $V_X$ and is given by:
The coefficient of variation provides a useful descriptor for the variability of a random variable around its expected value.

**The Probability Distribution for Functions of Random Variables**

In some cases it is interesting to be able to derive the cumulative distribution function $F_Y(y)$ for a random variable $Y$ which is given as a function of another random variable $X$, i.e. $Y = g(X)$, with given cumulative distribution function $F_X(x)$. Under the condition that the function $g(X)$ is monotonically increasing and furthermore, represents a one-to-one mapping of $x$ into $y$, a realization of $Y$ is only smaller than $y_0$ if correspondingly the realization of $X$ is smaller than $x_0$ which in turn is given by $x_0 = g^{-1}(y_0)$. In this case the cumulative distribution function $F_Y(y)$ can be readily determined by:

$$F_Y(y) = P(Y \leq y) = P(X \leq g^{-1}(y)) \quad (2.26)$$

which is also written as:

$$F_Y(y) = P(Y \leq y) = P(X \leq g^{-1}(y)) \quad (2.27)$$

$$F_Y(y) = F_X(g^{-1}(y)) \quad (2.28)$$

The probability density function $f_Y(y)$ is simply given by:

$$f_Y(y) = \frac{dF_X(g^{-1}(y))}{dy} \quad (2.29)$$

which immediately leads to:

$$f_Y(y) = \frac{dg^{-1}(y)}{dy} f_X(g^{-1}(y)) \quad (2.30)$$

and:

$$f_Y(y) = \frac{dx}{dy} f_X(x) \quad (2.31)$$
It is noticed that the application of Equations (2.30) and (2.31) necessitates that \( g(x) \) is at least one time differentiable in regard to \( x \).

If the function \( g(x) \) instead of being monotonically increasing is monotonically decreasing, a realization of \( Y \) smaller than \( y_0 \) corresponds to a realization of \( X \) larger than \( x_0 \), in which case it is necessary to change the sign of the derivative \( \frac{dx}{dy} \) in Eq. (2.31). Generally, for monotonically increasing or decreasing one-to-one functions \( g(x) \) there is:

\[
f_Y(y) = \left| \frac{dx}{dy} \right| f_X(x)
\]

Consider the random vector \( Y = (Y_1, Y_2, ..., Y_n)^T \) with individual components given as one-to-one mapping monotonically increasing or decreasing functions \( g_i, i = 1,2, ..., n \) of the components of the random vector \( X = (X_1, X_2, ..., X_n)^T \) as:

\[
Y_i = g_i(X)
\]

then there is:

\[
f_Y(y) = \left| J \right| f_X(x)
\]

where \( \left| J \right| \) is the numerical value of the determinant of \( J \) given by:

\[
\left| J \right| = \begin{vmatrix}
\frac{dx_1}{dy_1} & \cdots & \frac{dx_1}{dy_n} \\
\vdots & \ddots & \vdots \\
\frac{dx_n}{dy_1} & \cdots & \frac{dx_n}{dy_n}
\end{vmatrix}
\]

Finally, the expected value \( E[Y] \) of a function \( g(x) \) of the random vector \( X = (X_1, X_2, ..., X_n)^T \) is given by:

\[
E[Y] = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} g(x) f_X(x) dx_1 ... dx_n
\]
The Central Limit Theorem and Derived Distributions

The central limit theorem states:

The probability distribution for the sum of a number of random variables approaches the Normal distribution as the number becomes large.

In principle, the theorem is valid as long as the number of independent contributions to the sum is “large”. This implies that the sum may not be dominated by one or just a few random variables and furthermore, that the dependency between the random variables in the sum is not too strong. There is no requirement on the type of distributions of the random variables entering the sum, but if the distributions are skewed, the number of variables which is required for the validity of the theorem increases.

Figure 7  Illustration of probability density and cumulative distribution functions for different distribution types. Mean \( \mu = 160 \) and standard deviation \( \sigma = 40 \) are the same for all the distributions except the Exponential distribution for which the expected value and the variance are equal [4]
The Normal Distribution

The significant practical importance of the central limit theorem lies in the fact that even though only weak information is available regarding the number of contributions and their joint probability density function, rather strong information is achieved for the distribution of sum of the contributions.

The Normal probability distribution is thus applied very frequently in practical problems for the probabilistic modeling of uncertain phenomena which may be considered to originate from a cumulative effect of several independent uncertain contributions.

The Normal distribution has the property that the linear combination $S$ of $n$ Normal distributed random variable $X_i, i = 1,2, ..., n$:

$$S = a_0 + \sum_{i=1}^{n} a_i X_i$$

is also Normal distributed. The distribution is said to be closed in respect to summation.

Figure 8 shows the probability density and distribution function of a Normal distributed random variable. The area between the interval of $\mu - \sigma \leq x \leq \mu + \sigma$ for the density function corresponds to 68.3% of the total area of the density function which always has the value of 1.

![Figure 8](image.png)  

Figure 8  Example of the density and distribution function of a normally distributed random variable defined by the parameters $\mu = 160$ and $\sigma = 40$ [4]
One special type of the Normal distribution should be mentioned, namely the Standard Normal distribution. In general a standardized (sometimes referred to as reduced) random variable is a random variable which has been transformed such that it has an expected value \( E[X] = 0 \) and a variance \( V[X] = 1 \), i.e. the random variable \( Y \) defined by:

\[
Y = \frac{X - \mu_X}{\sigma_X} \tag{2.38}
\]

\( Y \) is a standardized random variable. If the random variable \( X \) follows the Normal distribution, the random variable \( Y \) is standard Normal distributed. In Figure 9 the process of standardization is illustrated.

![Illustration of the relationship between a Normal distributed random variable and a standard Normal distributed random variable](image)

**Figure 9** Illustration of the relationship between a Normal distributed random variable and a standard Normal distributed random variable [4]

It is common practice to denote the cumulative distribution function for the standard Normal distribution by \( \Phi(x) \) and the corresponding density function by \( \varphi(x) \).

**Confidence Intervals**

It is essential that statistical uncertainty is quantified and taken into account. A classical approach for the quantification and the communication of this uncertainty is by the use of confidence intervals. The \( 1 - \alpha \) confidence interval on an estimate defines an
interval within which the estimated parameter will occur with a predefined probability, with \( \alpha \) being the significance level (Figure 10).

![Standard normal distribution](image)

**Figure 10** Double sided and symmetrical \( 1 - \alpha \) confidence interval on the mean value [4]

If the case is considered where the standard deviation \( \sigma_X \) of an uncertain variable \( X \) is known with certainty and the mean value is unknown, then the double sided and symmetrical \( 1 - \alpha \) confidence interval on the mean value is given by:

\[
P \left[ -k_{\alpha/2} \leq \frac{\bar{X} - \mu_X}{\frac{1}{\sigma_X \sqrt{n}}} \leq k_{\alpha/2} \right] = 1 - \alpha
\]

where \( n \) is the number of samples planned for the estimation of the mean value.
From Eq. (2.39) it is seen that the confidence interval limits depend on $\alpha$, $n$ and $\sigma_X$. Typically $\alpha$ is chosen as 0.1, 0.05 and 0.01 in engineering applications. Narrow confidence intervals may be achieved by increasing the number of experiments, which on the other hand, may be expensive to achieve and in some cases not even possible for practical reasons.

### 2.7. Methods of Reliability

#### 2.7.1. Failure Events and Basic Random Variables

In reliability analysis of technical systems and components the main problem is to evaluate the probability of failure corresponding to a specified reference period. However, other non-failure states of the considered component or system may also be of interest, such as excessive damage, unavailability, etc.

In general, any state which may be associated with consequences in terms of costs, loss of lives and impact to the environment is of interest. In the following section no differentiation will be made between these different types of states. For simplicity, all these events will be referred to as being failure events, bearing in mind, however, that also non-failure states may be considered in the same manner.

It is convenient to describe failure events in terms of functional relations. If they are fulfilled, the considered event will occur. A failure event may be described by a functional relation, the limit state function $g(x)$, in the following way:

$$F = \{g(x) \leq 0\} \quad (2.40)$$

where the components of the vector $x$ are realizations of the basic random variables $X$ representing all the relevant uncertainties influencing the probability of failure. In Eq. (2.40) the failure event $F$ is simply defined as the set of realizations of the function $g(x)$, which is zero or negative.

Having defined the failure event, the probability of failure $P_F$ may be determined by the following integral:

$$P_F = \int_{g(x) \leq 0} f_X(x)dx \quad (2.41)$$
where $f_X(x)$ is the joint probability density function of the random variables $X$. This integral is, however, non-trivial to solve and numerical approximations are expedient. Various methods for the solution of the integral in Eq. (2.41) have been proposed including numerical integration techniques, Monte Carlo simulations and asymptotic Laplace expansions. Numerical integration techniques very rapidly become inefficient for increasing size of the vector $X$. In the following section the focus is directed toward the widely applied and quite efficient First Order Reliability Methods (FORM) [8], which furthermore are consistent with the solutions obtained by asymptotic Laplace integral expansions.

### 2.7.2. Linear Limit State Functions and Normal Distributed Variables

For illustrative purposes first the case where the limit state function $g(x)$ is linear function of the basic random variables $X$ is considered. Then the limit state function may be written as:

$$g(x) = a_0 + \sum_{i=1}^{n} a_i x_i$$

(2.42)

The safety margin is then defined as $M = g(x)$. Failure can be defined by:

$$M \leq 0$$

(2.43)

If the basic random variables are Normal distributed, the linear safety margin $M$ defined through:

$$M = a_0 + \sum_{i=1}^{n} a_i X_i$$

(2.44)

is also Normal distributed with mean value and variance:

$$\mu_M = a_0 + \sum_{i=1}^{n} a_i \mu_{X_i}$$

(2.45)
\[
\sigma_M^2 = \sum_{i=1}^{n} a_i^2 \sigma_{X_i}^2 + 2 \left( \sum_{i=1}^{n} \sum_{j=i+1}^{n} a_i a_j \sigma_{X_i} \sigma_{X_j} \rho_{ij} \right) \]

(2.46)

\(\rho_{ij}\) are the correlation coefficients between the variables \(X_i\) and \(X_j\).

Defining the failure event by Eq. (2.40), the probability of failure can be written as:

\[
P_F = P(g(x) \leq 0) = P(M \leq 0)
\]

(2.47)

which in this simple case reduces to the evaluation of the standard Normal distribution function:

\[
P_F = \Phi \left( \frac{0 - \mu_M}{\sigma_M} \right) = \Phi(-\beta)
\]

(2.48)

where the reliability index \(\beta\) is given as:

\[
\beta = \frac{\mu_M}{\sigma_M}
\]

(2.49)

In the case where the basic random variables are uncorrelated, the reliability index \(\beta\) as defined in Eq. (2.49) has a geometrical interpretation as illustrated in Figure 11 where a two dimensional case is considered.

**Figure 11** Illustration of the two-dimensional case of a linear limit state function and Normal distributed variables \(x\) [4]
In Figure 11 the limit state function $g(x)$ has been transformed into the limit state function $g(u)$ by standardization of the random variables as:

$$U_i = \frac{X_i - \mu_{X_i}}{\sigma_{X_i}}$$  \hspace{1cm} (2.50)

such that the random variables $U_i$ have mean values equal to zero and standard deviation values equal to one.

The reliability index $\beta$ has the simple geometrical interpretation as the smallest distance from the line (or generally the hyper-plane) forming the boundary between the safe domain and the failure domain, i.e. the domain defined by the failure event. It should be noted that this definition of the reliability index does not depend on the limit state function but rather the boundary between the safe domain and the failure domain. The point on the failure surface with the smallest distance to the origin is commonly called the design point or the most likely failure point.

It is seen that the evaluation of the probability of failure in this simple case reduces to a simple evaluation in terms of mean values and standard deviations of the basic random variables, i.e. the first and second order information.

2.7.3. The Error Propagation Law

The results given in Eq.(2.45) and (2.46) have been applied to study the statistical characteristics of errors $\varepsilon$ accumulating in accordance with some differentiable function $h(x)$, i.e.

$$\varepsilon = h(x)$$  \hspace{1cm} (2.51)

where $x = x_1, x_2, ..., x_n$ is a vector of realizations of the basic random variables $X$ representing measurement uncertainties with mean values $\mu_X = \mu_{X_1}, \mu_{X_2}, ..., \mu_{X_n}$ and covariances $C_{X_iX_j} = \rho_{ij}\sigma_{X_i}\sigma_{X_j}$ where $\sigma_{X_i}$ are standard deviations and $\rho_{ij}$ are the correlation coefficients. The idea is to approximate the function $h(x)$ by its Taylor expansion including only the linear terms, i.e.:

$$\varepsilon \cong h(x_0) + \sum_{i=1}^{n} (x_i - x_{i,0}) \frac{\partial h(x)}{\partial x_i} \bigg|_{x=x_0}$$  \hspace{1cm} (2.52)
where \( x_0 = x_{1,0}, x_{2,0}, \ldots, x_{n,0} \) is the point in which the linearization is performed, normally chosen as the mean value point.

\[
\frac{\partial h(x)}{\partial x_i} \bigg|_{x=x_0}, \quad i = 1, 2, \ldots, n
\]

are the first order partial derivatives of \( h(x) \) taken in \( x = x_0 \). From Equations (2.52), (2.45) and (2.46) it is seen that the expected value of the error \( E[\varepsilon] \) can be assessed by:

\[
E[\varepsilon] \cong h(\mu_X)
\]  

(2.53)

and its variance \( Var[\varepsilon] \) can be determined by:

\[
Var[\varepsilon] \cong \sum_{i=1}^{n} \left( \frac{\partial h(x)}{\partial x_i} \bigg|_{x=x_0} \right)^2 \sigma_{X_i}^2 + 2 \sum_{i=1}^{n} \sum_{j=i+1}^{n} \left( \frac{\partial h(x)}{\partial x_i} \bigg|_{x=x_0} \right) \left( \frac{\partial h(x)}{\partial x_j} \bigg|_{x=x_0} \right) \rho_{ij} \sigma_{X_i} \sigma_{X_j}
\]  

(2.54)

It is important to notice that the variance of the error as given by Eq. (2.54) depends on the linearization point, i.e. \( x_0 = x_{1,0}, x_{2,0}, \ldots, x_{n,0} \).

In Figure 12 the basic principle of the error propagation law is represented for the two-dimensional case in which a random variable \( Y \) depends on only one random variable \( X: y = g(x) \). The function \( y = g(x) \) is linearized in the mean value point and the relationship between \( Y \) and \( X \) is approximated with

\[
y = g(\mu_X) + \frac{\partial g(x)}{\partial x} \bigg|_{x=\mu_X} (X - \mu_X).
\]

The distribution of the derived random variable \( y \) represented on the vertical axis in Figure illustrates the approximated distribution resulting from the error propagation law. Depending on the form of the transformation function \( h(x) \), the distribution can differ from the true distribution for \( y \) significantly.
Figure 12  Illustration of the error propagation law: The transformation of the density function $f_Y(y)$ according to the relation $y = g(x)$ and the linear approximation of the relation between the two random variables [4]

### 2.7.4. Non-linear Limit State Functions

When the limit state function is non-linear in the basic random variables $X$, the situation is not as simple as outlined in the previous section. An obvious approach is, however, to consider the error propagation law to represent the failure domain in terms of a linearization of the boundary between the safe domain and the failure domain, i.e. the failure surface. Still, the question remains how to do this appropriately.

Hasofer and Lind [5] suggest to perform this linearization in the design point of the failure surface in normalized space. The situation is illustrated in the 2-dimesional space in Fig 13.

A principal sketch is given, illustrating that the failure surface is linearized in the design point $u^*$ by the line $g'(u) = 0$. The $\alpha$-vector is the outward directed normal vector to the failure surface with the shortest distance $\beta$ to the origin.
As the limit state function is in general non-linear, one does not know the design point in advance. This has to be found, e.g. by solving the following optimization problem:

$$
\beta = \min_{u \in \{g(u) = 0\}} \sqrt{\sum_{i=1}^{n} u_i^2} \tag{2.55}
$$

This problem may be solved in a number of different ways. Provided that the limit state function is differentiable, the following iteration scheme may be followed:

$$
\alpha_i = \frac{-\frac{\partial g}{\partial u_i}(\beta \alpha)}{\left[\sum_{i=1}^{n} \frac{\partial g}{\partial u_i}(\beta \alpha)^2\right]^{1/2}}, \quad i = 1,2, ..., n \tag{2.56}
$$

$$
g(\beta \alpha_1, \beta \alpha_2, ..., \beta \alpha_n) = 0 \tag{2.57}
$$

First, a design point is guessed $u^* = \beta \alpha$ and inserted into Eq.(2.56) whereby a new normal vector $\alpha$ to the failure surface is found. Then this $\alpha$-vector is inserted into Eq. (2.61), from which a new $\beta$-value is calculated.

The iteration scheme will converge in a few iterations and provides the design point $u^*$ as well as the reliability index $\beta$ and the outward normal to the failure surface in the design point $\alpha$. As already mentioned, the reliability index $\beta$ may be related to the probability of failure. The components of the $\alpha$-vector may be interpreted as sensitivity factors giving the relative importance of the individual random variables for the reliability index $\beta$. 
Second Order Reliability Methods (SORM) follow the same principles as FORM. However, as a logical extension of FORM, the failure surface is expanded to the second order in the design point. The result of a SORM analysis may be given as the FORM \( \beta \) multiplied with a correction factor evaluated on the basis of the second order partial derivatives of the failure surface in the design point. The SORM analysis becomes exact for failure surfaces given as a second order polynomial of the basic random variables. In general, however, the result of a SORM analysis can be shown to be asymptotically exact for any shape of the failure surface as \( \beta \) approaches infinity. The details of SORM analysis can be found in the literature, e.g. Madsen et al. [6].

### 2.7.5. Sampling Methods

Consider again the probability integral defined in Equation (2.41):

\[
P_F = \int_{g(x) \leq 0} f_X(x) dx
\]  

(2.58)
It has been seen that FORM methods may successfully be applied for the evaluation of this integral. The integral may also be estimated by so-called sampling techniques. In the literature, a large variety of simulation techniques can be found. Here it is just noted that sampling techniques have proven their value especially for the problems where the representation of the limit state function is associated with difficulties. Such cases are e.g. when the limit state function is not differentiable or when several design points contribute to the failure probability.

However, as all sampling techniques have their origin in the so-called crude Monte Carlo Method, the principles of which will be briefly outlined below.

The basis for sampling techniques is well illustrated by rewriting the probability integral in Eq. (2.58) by means of an indicator function:

\[
P_F = \int_{g(x) \leq 0} f_X(x)dx = \int I[g(x) \leq 0]f_X(x)dx
\]

(2.59)

where the integration domain is changed from the part of the sample space of the vector \( X = X_1, X_2, \ldots, X_n \) for which \( g(x) \leq 0 \), to the entire sample space of \( X \) and where \( I[g(x) \leq 0] \) is an indicator function equal to 1 if \( g(x) \leq 0 \) and otherwise equal to zero. Equation (2.59) is in this way seen to yield the expected value of the indicator function \( I[g(x) \leq 0] \). Therefore, if now \( N \) realizations of the vector \( X \), i.e. \( x_j, j = 1, 2, \ldots, N \) are sampled it follows from sample statistics that:

\[
P_F = \frac{1}{N} \sum_{j=1}^{N} I[g(x) \leq 0]
\]

(2.60)

is an unbiased estimator of the failure probability \( P_F \).

The principle of the crude Monte Carlo sampling technique rests directly on the application of Equation (2.60). A large number of realizations of the basic random variables \( X \), i.e. \( x_j, j = 1, 2, \ldots, N \) are generated and for each of the outcomes \( x_j \) it is checked whether or not the limit state function taken in \( x_j \) is positive. All the samples for which this is not the case are counted \( (n_F) \) and after \( N \) simulations the failure probability \( P_F \) may be estimated through:
which then may be considered a sample expected value of the probability of failure. In fact, for $N \to \infty$ the estimate of the failure probability becomes exact. However, simulations are often costly in computation time and the uncertainty of the estimate is thus of interest. It is easily realized that the coefficient of variation of the estimate is proportional to $1/\sqrt{n_F}$.

In this chapter, basics of probability theory were introduced: the three axioms of probability theory, conditional probabilities, useful characteristics of the data (central measures, dispersion measures, etc.), uncertainty modeling with random variables, as well as first order reliability methods. In the next section, optimization capabilities of Dakota software package will be presented.
3. Dakota Software Package and Optimization Capabilities

In the previous chapter, basics of probability theory were introduced. In this section, open source optimization software, Dakota, will be presented and its optimization capabilities overviewed, [7].

Dakota (Design Analysis Kit for Optimization and Terascale Applications) is a multilevel parallel object-oriented open source framework for design optimization, parameter estimation, uncertainty quantification and sensitivity analysis [7]. Dakota delivers a variety of iterative methods and strategies, and the ability to flexibly interface them to simulation code. Recent versions go beyond optimization to include other iterative analysis methods such as uncertainty quantification with nondeterministic propagation methods, parameter estimation with nonlinear least squares solution methods, and sensitivity/variance analysis with general-purpose design of experiments and parameter study capabilities.

The principal classes of Dakota algorithms, with brief descriptions, are summarized:

- **Parameter studies**: Parameter studies employ deterministic designs to explore the effect of parametric changes within simulation models, yielding one form of sensitivity analysis. They can help assess simulation characteristics such as smoothness, multi-modality, robustness, and nonlinearity, which affect the choice of algorithms and controls in follow-on optimization and uncertainty quantification studies.

- **Design of experiments**: Design and analysis of computer experiments (DACE) techniques are often used to explore the parameter space of an engineering design problem, for example to perform global sensitivity analysis. DACE methods can help reach conclusions similar to parameter studies, but the primary goal of these methods is to generate good coverage of the input parameter space.

- **Uncertainty quantification**: Uncertainty quantification methods (also referred to as nondeterministic analysis methods) compute probabilistic information about response functions based on simulations performed according to
specified input parameter probability distributions. Put another way, these methods perform a forward uncertainty propagation in which probability information for input parameters is mapped to probability information for output response functions.

- **Optimization**: Optimization solvers seek to minimize cost or maximize system performance, as predicted by the simulation model, subject to constraints on input variables or secondary simulation responses. Categories of algorithms include gradient-based, derivative-free and global optimization. Dakota also includes capabilities for multi-objective trade-off optimization and automatic scaling of problem formulations. Advanced Dakota approaches include hybrid (multi-method), multi-start local and Pareto-set optimization.

- **Calibration**: Calibration algorithms seek to maximize agreement between simulation outputs and experimental-data (or desired outputs). They are used to solve inverse problems (often referred to as parameter estimation or least-squares problems).

There are also more sophisticated strategies such as hybrid optimization, surrogate-based optimization, optimization under uncertainty and mixed aleatory/epistemic uncertainty quantification.

### 3.1. Optimization

Optimization is a discipline of numerical mathematics which aims to improve the operation of a system or process as good as possible in some defined sense. [3]

Optimization algorithms work to minimize (or maximize) an objective function, typically calculated by the user simulation code, subject to constraints on design variables and responses.
3.1.1. **Optimization Formulations**

This section provides a basic introduction to the mathematical formulation of optimization problems, [7].

A general optimization problem is formulated as follows:

\[
\begin{align*}
\text{minimize:} & \quad f(x) \\
\text{x} & \in \mathbb{R}^n \\
\text{subject to:} & \quad g_L \leq g(x) \leq g_U \\
& \quad h(x) = h_t \\
& \quad a_L \leq A_i x \leq a_U \\
& \quad A_e x = a_t \\
& \quad x_L \leq x \leq x_U
\end{align*}
\]  

(3.1)

where vector and matrix terms are marked in bold typeface. In this formulation, \( x = [x_1, x_2, ..., x_n] \), is an n-dimensional vector of real-valued design variables or design parameters. The n-dimensional vectors, \( x_L \) and \( x_U \), are the lower and upper bounds, respectively, on design parameters. These bounds define the allowable values for the elements of \( x \), and set of all allowable values is termed the design space or the parameter space. A design point or a sample point is a particular set of values within the parameter space.

The optimization goal is to minimize the objective function, \( f(x) \), while satisfying the constraints. Constraints can be categorized as either linear or nonlinear and as either inequality or equality. The nonlinear inequality constraints, \( g(x) \), are “2-sided” in that they have both lower and upper bounds, \( g_L \) and \( g_U \), respectively. The nonlinear equality constraints, \( h(x) \), have target values specified by \( h_t \). The linear inequality constraints create a linear system \( A_i x \) where \( A_i \) is the coefficient matrix for the linear system. These constraints are also 2-sided as they have lower and upper bounds, \( a_L \) and \( a_U \), respectively. The linear equality constraints create a linear system \( A_e x \) where \( A_e \) is the coefficient matrix for the linear system and \( a_t \) are the target values. The constraints partition the parameter space into feasible and infeasible regions. A design point is said to be feasible if and only if it satisfies all of the constraints. Correspondingly, a design point is said to be infeasible if it violates one or more of the constraints.
Many different methods exist to solve the optimization problem given by Equation (3.1), all of which iterate on $x$ in some manner. That is, an initial value for each parameter in $x$ is chosen, the response quantities, $f(x), g(x), h(x)$, are computed, often by running a simulation, and some algorithm is applied to generate a new $x$ that will either reduce the objective function, reduce the amount of infeasibility, or both. To facilitate a general presentation of these methods, three criteria will be used in the following discussion to differentiate them: optimization problem type, search goal, and search method.

The optimization problem type can be characterized both by types of constraints present in the problem and by the linearity or nonlinearity of the objective and constraint functions. For constraint categorization, a hierarchy of complexity exists for optimization algorithms, ranging from simple bound constraints, through linear constraints, to full nonlinear constraints. By the nature of this increasing complexity, optimization problem categorizations are inclusive of all constraint types up to a particular level of complexity. That is, an unconstrained problem has no constraints, a bound-constrained problem has only lower and upper bounds on the design parameters, a linearly-constrained problem has both linear and bound constraints, and a nonlinearly-constrained problem may contain the full range of nonlinear, linear, and bound constraints. If all of the linear and nonlinear constraints are equality constraints, then this is referred to as an equality-constrained problem, and if all of the linear and nonlinear constraints are inequality constraints, then this is referred to as an inequality-constrained problem. Further categorizations can be made based on the linearity of the objective and constraint functions. A problem where the objective function and all constraints are linear is called a linear programming (LP) problem. These types of problems commonly arise in scheduling, logistics, and resource allocation applications. Likewise, a problem where at least some of the objective and constraint functions are nonlinear is called a nonlinear programming (NLP) problem. These NLP problems predominate in engineering applications.

The search goal refers to the ultimate objective of the optimization algorithm, i.e., either global or local optimization. In global optimization, the goal is to find the design point that gives the lowest feasible objective function value over the entire parameter space. In contrast, in local optimization, the goal is to find a design point that is lowest
relative to a “nearby” region of parameter space. In almost all cases, global optimization will be computationally more expensive than local optimization. Thus, the user must choose an optimization algorithm with an appropriate search scope that best fits the problem goals and the computational budget.

The search method refers to the approach taken in the optimization algorithm to locate a new design point that has a lower objective function or is more feasible than the current design point. The search method can be classified as either gradient-based or nongradient-based. In a gradient based algorithm, gradients of the response functions are computed to find the direction of improvement. Gradient-based optimization is the search method that underlies many efficient local optimization methods. However, a drawback to this approach is that gradients can be computationally expensive, inaccurate, or even nonexistent. In such situations, nongradient-based search methods may be useful. There are numerous approaches to nongradient-based optimization. Some of the more well known of these include pattern search methods (nongradient-based local techniques) and genetic algorithms (nongradient-based global techniques).

Because of the computational cost of running simulation models, surrogate-based optimization (SBO) methods are often used to reduce the number of actual simulation runs. In SBO, a surrogate or approximate model is constructed based on a limited number of simulation runs. The optimization is then performed on the surrogate model.

In this chapter an overview of optimization approaches was given which underscores that no single optimization method or algorithm works best for all types of optimization problems.

3.1.2. Optimization Methods

In selecting an optimization method, important considerations include the type of variables in the problem (continuous, discrete, mixed), whether a global search is needed or a local search is sufficient, and the required constraint support (unconstrained, bound constrained, or generally constrained). Less obvious, but equally important, considerations include the efficiency of convergence to an optimum (i.e., convergence rate) and the robustness of the method in the presence of challenging design space features (e.g., nonsmoothness)
Gradient-based Methods

Gradient-based optimization methods are highly efficient, with the best convergence rates of all of the optimization methods. If analytic gradient and Hessian information can be provided by an application code, a full Newton method will provide quadratic convergence rates near the solution. More commonly, only gradient information is available and a quasi-Newton method is chosen in which the Hessian information is approximated from an accumulation of gradient data.

Thus, for gradient-based optimization, a critical factor is the gradient accuracy. Analytic gradients are ideal, but are often unavailable. For many engineering applications, a finite difference method will be used by the optimization algorithm to estimate gradient values. Dakota allows the user to select the step size for these calculations, as well as choose between forward-difference and central-difference algorithms. The finite difference step size should be selected as small as possible, to allow for local accuracy and convergence, but not so small that the steps are “in the noise”. This requires an assessment of the local smoothness of the response functions using, for example, a parameter study method. Central differencing, in general, will produce more reliable gradients than forward differencing, but at roughly twice the expense.

Nongradient-based Methods

Nongradient-based methods exhibit much slower convergence rates for finding an optimum, and as a result, tend to be much more computationally demanding than gradient-based methods. Nongradient local optimization methods, such as pattern search algorithms, often require from several hundred to a thousand or more function evaluations, depending on the number of variables, and nongradient global optimization methods such as genetic algorithms may require from thousands to tens-of-thousands of function evaluations. Clearly, for nongradient optimization studies, the computational cost of the function evaluation must be relatively small in order to obtain an optimal solution in a reasonable amount of time. In addition, nonlinear constraint support in nongradient methods is an open area of research and, while supported by many nongradient methods, is not as refined as constraint support in gradient-based methods. However, nongradient methods can be more robust and more inherently parallel than
gradient-based approaches. They can be applied in situations where gradient calculations are too expensive and unreliable. In addition, some nongradient-based methods can be used for global optimization which gradient-based techniques, by themselves, cannot. For these reasons, nongradient-based methods deserve consideration when the problem may be nonsmooth, multimodal or poorly behaved.

**Surrogate-based Methods**

Surrogate models approximate an original, high fidelity “truth” model, typically at reduced computational cost. In the context of minimization (optimization or calibration), surrogate models can speed convergence by reducing function evaluation cost or smoothing noisy response functions. There are three categories of surrogate-based minimization:

- Trust region-managed surrogate-based local minimization, with data fit surrogate, multifidelity models, or reduced-order models.
- Surrogate-based global minimization, where a single surrogate is built (and optionally iteratively updated) over the whole design space.
- Efficient global minimization: nongradient-based constrained and unconstrained optimization and nonlinear least squares based on Gaussian process models, guided by an expected improvement function.

**3.2. Optimization Under Uncertainty**

Optimization under uncertainty (OUU) approaches incorporate an uncertainty quantification method within the optimization process, [7]. This is often needed in engineering design problems when the effect of input parameter uncertainties on the response function of interest must be included. A typical engineering example of optimization under uncertainty would minimize the probability of failure of a structure for a set of applied loads, where there is uncertainty in the loads and/or material properties of the structural components.

In optimization under uncertainty, a nondeterministic method is used to evaluate the effect of uncertain variable distributions on response function of interest. Statistics on these response functions are then included in the objective and constraint functions of an optimization process. Different uncertainty quantification methods can have very
different features from an optimizing perspective, leading to the tailoring of optimization under uncertainty approaches to particular underlying uncertainty quantification methods.

If the uncertainty quantification method is sampling based, then three approaches are currently supported: nested optimization under uncertainty, surrogate based optimization under uncertainty and trust-region surrogate-based optimization under uncertainty.

Another class of optimization under uncertainty algorithms is called reliability-based design optimization (RBDO). Reliability-based design optimization methods are used to perform design optimization accounting for reliability metrics.

3.2.1. Uncertain Variables

Deterministic variables (i.e., those with a single known value) do not capture the behavior of the input variables in all situations. In many cases, the exact value of a model parameter is not precisely known. An example of such an input variable is the thickness of a heat treatment coating on a structural steel I-beam used in building construction. Due to variabilities and tolerances in the coating process, the thickness of the layer is known to follow a normal distribution with a certain mean and standard deviation as determined from experimental data. The inclusion of the uncertainty in the coating thickness is essential to accurately represent the resulting uncertainty in the response of the building.

Aleatory uncertain variables

Aleatory uncertainties are irreducible variabilities inherent in nature. They are characterized by having a sufficiently rich set of data as to allow modeling using probability distributions, and probabilistic methods are commonly used for propagating input aleatory uncertainties described by probability distribution specifications.

Epistemic uncertain variables

Epistemic uncertainties are reducible uncertainties resulting from a lack of knowledge. For epistemic uncertainties, data is generally sparse, making the use of probability theory questionable and leading to nonprobabilistic methods based on interval specifications.
3.3. Uncertainty Quantification

At a high level, uncertainty quantification or nondeterministic analysis is the process of characterizing input uncertainties, forward propagating these uncertainties through a computational model, and performing statistical or interval assessments on the resulting responses. This process determines the effect of uncertainties and assumptions on model outputs or results.

Uncertainty quantification is related to sensitivity analysis in that the common goal is to gain an understanding of how variations in the parameters affect the response functions of the engineering design problem. For uncertainty quantification, some or all of the components of the parameter vector, are considered to be uncertain as specified by particular probability distributions (e.g., normal, exponential, extreme value), or other uncertainty structures. By assigning specific distributional structure to the inputs, distributional structure for the outputs (i.e., response statistics) can be inferred. This migrates from an analysis that is more qualitative in nature, in the case of sensitivity analysis, to an analysis that is more rigorously quantitative.

Uncertainty quantification methods are often distinguished by their ability to propagate aleatory or epistemic input uncertainty characterizations, where aleatory uncertainties are irreducible variabilities inherent in nature and epistemic uncertainties are reducible uncertainties resulting from a lack of knowledge. Since sufficient data is generally available for aleatory uncertainties, probabilistic methods are commonly used for computing response distribution statistics based on input probability distribution specifications. Conversely, for epistemic uncertainties, any use of probability distributions is based on subjective knowledge rather than objective data, and nonprobabilistic methods based on interval specifications may alternatively be explored.

Variables and responses for uncertainty quantification

All the uncertainty quantification methods perform a forward uncertainty propagation in which probability or interval information for input parameters is mapped to probability or interval information for output response functions. The $m$ functions in the response data set are interpreted as $m$ general response functions.

Within the variables specification, uncertain variable descriptions are employed to define the parameter probability distributions. The continuous aleatory distribution
types include: normal (Gaussian), lognormal, uniform, loguniform, triangular, exponential, beta, gamma, gumbel, frechet, weibull and histogram bin. The discrete aleatory distribution types include: poisson, binomial, negative binomial, geometric, hypergeometric and histogram point. The epistemic distribution type is interval for continuous variables. For epistemic discrete variables, there are three types: integer range, integer set and real set. When gradient information is used in uncertainty assessment, derivative components are normally computed with respect to the active continuous variables.

3.3.1. Sampling Methods

Sampling method generates sets of samples according to the probability distributions of the uncertain variables and maps them into corresponding sets of response functions. Means, standard deviations, coefficients of variation (COVs) and 95% confidence intervals are computed for the response functions.

Currently, traditional Monte Carlo (MC) and Latin Hypercube sampling (LHS) are supported by Dakota, [7]. In Monte Carlo sampling, the samples are selected randomly according to the user-specified probability distributions (Figure 15). Latin Hypercube sampling is a stratified sampling technique for which the range of each uncertain variable is divided into \( N_S \) segments of equal probability, where \( N_S \) is the number of samples requested. The relative lengths of the segments are determined by the nature of the specified probability distribution (e.g., uniform has segments of equal width, normal has small segments near the mean and larger segments in the tails). For each of the uncertain variables, a sample is selected randomly from each of these equal probability segments. These \( N_S \) values for each of the individual parameters are then combined in a shuffling operation to create a set of \( N_S \) parameter vectors with a specified correlation structure. A feature of the resulting sample set is that every row and column in the Hypercube has exactly one sample. Since the total number of samples is exactly equal to the number of partitions used for each uncertain variable, an arbitrary number of desired samples is easily accommodated (as compared to less flexible approaches in which the total number of samples is a product or exponential function of the number of intervals for each variable, i.e. many classical designs of experiment methods).
Advantages of sampling-based methods include their relatively simple implementation and their independence from the scientific disciplines involved in the analysis. The main drawback of these techniques is the large number of function evaluations needed to generate converged statistics, which can render such analysis computationally very expensive, if not intractable, for real-world engineering applications. LHS techniques, in general, require fewer samples than traditional Monte Carlo for the same accuracy in statistics, but they can still be prohibitively expensive. Note that under certain separability conditions associated with the function to be sampled, Latin Hypercube sampling provides a more accurate estimate of the mean value than does random sampling. That is, given an equal number of samples, the LHS estimate of the mean will have less variance than the mean value obtained through random sampling.

Figure 14 demonstrates Latin Hypercube sampling on a two-variable parameter space. Here, the range of both parameters, $x_1$ and $x_2$, is [0, 1]. Also, for this example both $x_1$ and $x_2$ have uniform statistical distributions. For Latin Hypercube sampling, the range of each parameter is divided into $p$ “bins” of equal probability. For parameters with uniform distributions, this corresponds to partitions of equal size. For $n$ design parameters, the partitioning yields a total of $p^n$ bins in the parameter space. Next, $p$ samples are randomly selected in the parameter space, with the following restrictions:

1. each sample is randomly placed inside a bin
2. for all one-dimensional projections of the $p$ samples and bins, there will be one and only sample in each bin.
In a two-dimensional example such as that shown in Figure 14, these LHS rules guarantee that only one bin can be selected in each row and column. For $p = 4$, there are four partitions in both $x_1$ and $x_2$. This gives a total of $4^2 = 16$ bins, of which four will be chosen according to the criteria described above. Note that there is more than one possible arrangement of bins that meet the LHS criteria. The dots in Figure 14 represent the four sample sites in this example, where each sample is randomly located in its bin. There is no restriction on the number of bins in the range of each parameter, however, all parameters must have the same number of bins.

The actual algorithm for generating Latin Hypercube samples is more complex than indicated by the description above.
3.3.2. Reliability methods

Reliability methods provide an alternative approach to uncertainty quantification which can be less computationally demanding than sampling techniques. Reliability methods for uncertainty quantification are based on probabilistic approaches that compute approximate response function distribution statistics based on specific uncertain variable distributions. These response functions include response mean, response standard deviation and cumulative or complementary cumulative distribution functions (CDF/CCDF). These methods are often more efficient at computing statistics in the tails of the response distributions (events with low probability) than sampling based approaches since the number of samples required to resolve a low probability can be prohibitive.

The methods all answer the fundamental question: “Given a set of uncertain input variables, \( X \), and a scalar response function, \( g \), what is the probability that the response function is below or above a certain level, \( z \)?” The former can be written as \( P[g(X) \leq z] = F_g(z) \) where \( F_g(z) \) is the cumulative distribution function (CDF) of the uncertain response \( g(X) \) over a set of response levels. The latter can be written as \( P[g(X) > z] \) and defines the complementary cumulative distribution function (CCDF).

This probability calculation involves a multi-dimensional integral over an irregularly shaped domain of interest, \( D \), where \( g(X) < z \) as displayed in Figure 16 for...
The case of two variables. The reliability methods all involve the transformation of the user specified uncertain variables, $X$, with probability density function $p(x_1, x_2)$, which can be non-normal and correlated, to a space of independent Gaussian random variables, $u$, possessing a mean value of zero and unit variance (i.e. standard normal variables). The region of interest, $D$, is also mapped to the transformed space to yield, $D_u$, where $g(U) < z$ as shown in Figure 17.

![Graphical depiction of calculation of cumulative distribution function in the original uncertain variable space](image)

**Figure 16** Graphical depiction of calculation of cumulative distribution function in the original uncertain variable space [7]

The Nataf transformation, which is identical to Rosenblatt transformation in the case of independent random variables, is used in Dakota to accomplish this mapping. This transformation is performed to make the probability calculation more tractable. In the transformed space, probability contours are circular in nature as shown in Figure 17 unlike in the original uncertain variable space, Figure 16. Also, the multi-dimensional integrals can be approximated by simple functions of a single parameter, $\beta$, called the reliability index. $\beta$ is also the minimum Euclidean distance from the origin in the transformed space to the response surface. This point is also known as the most probable point (MPP) of failure. Note, however, the methodology is equally applicable for generic functions, not simply those corresponding to failure criteria; this
nomenclature is due to the origin of these methods within the disciplines of structural safety and reliability. Note that there are local and global reliability methods. The majority of the methods available are local, meaning that a local optimization formulation is used to locate one MPP. In contrast, global methods can find multiple MPPs if they exist.

![Graphical depiction of integration for the calculation of cumulative distribution function in the transformed uncertain variable space][1]

**Figure 17**  Graphical depiction of integration for the calculation of cumulative distribution function in the transformed uncertain variable space [7]

**Local reliability methods**

Local reliability methods include the Mean Value method and the family of most probable point (MPP) search methods, [8]. Each of these methods is gradient-based, employing local approximations and/or local optimization methods.

**Mean Value**

The Mean Value method (MV) is the simplest, least-expensive reliability method because it estimates the response means, response standard deviations, and all CDF/CCDF response-probability-reliability levels from a single evaluation of response functions and their gradients at the uncertain variable means. This approximation can have acceptable accuracy when the response functions are nearly linear and their distributions are approximately Gaussian, but can have poor accuracy in other situations. The expressions for approximate response mean $\mu_g$, approximate response variance $\sigma_g^2$, response target to approximate probability/reliability level mapping $f(\beta)$.
$(\bar{z} \rightarrow \bar{p}, \bar{\beta})$, and probability/reliability target to approximate response level mapping $(\bar{p}, \bar{\beta} \rightarrow z)$ are

$$\mu_g = g(\mu_x)$$  \hspace{1cm} (3.2)

$$\sigma_g^2 = \sum_i \sum_j \text{Cov}(i,j) \frac{dg}{dx_i}(\mu_x) \frac{dg}{dx_j}(\mu_x)$$  \hspace{1cm} (3.3)

$$\bar{z} \rightarrow \beta: \ \beta_{CDF} = \frac{\mu_g - \bar{z}}{\sigma_g}, \ \beta_{CCDF} = \frac{\bar{z} - \mu_g}{\sigma_g}$$  \hspace{1cm} (3.4)

$$\bar{\beta} \rightarrow z: \ z = \mu_g - \sigma_g \cdot \beta_{CDF}, \ z = \mu_g + \sigma_g \cdot \beta_{CCDF}$$  \hspace{1cm} (3.5)

respectively, where $x$ are the uncertain values in the space of the original uncertain values ("x-space"), $g(x)$ is the limit state function (the response function for which probability-response level pairs are needed), and $\beta_{CDF}$ and $\beta_{CCDF}$ are the CDF and CCDF reliability indices, respectively.

With the introduction of second-order limit state information, MV calculates a second-order mean as

$$\mu_g = g(\mu_x) + \frac{1}{2} \sum_i \sum_j \text{Cov}(i,j) \frac{d^2g}{dx_i dx_j}(\mu_x)$$  \hspace{1cm} (3.6)

This is commonly combined with a first order variance (Equation (3.3)), since second-order variance involves higher order distribution moments (skewness, kurtosis) which are often unavailable.

The first-order CDF probability $p(g \leq z)$, first order CCDF probability $p(g > z)$, $\beta_{CDF}$ and $\beta_{CCDF}$ are related to one another through:

$$p(g \leq z) = \Phi(-\beta_{CDF})$$  \hspace{1cm} (3.7)

$$p(g > z) = \Phi(-\beta_{CCDF})$$  \hspace{1cm} (3.8)

$$\beta_{CDF} = -\Phi^{-1}(p(g \leq z))$$  \hspace{1cm} (3.9)
\begin{align}
\beta_{CCDF} &= -\Phi^{-1}(p(g > z)) \quad (3.10) \\
\beta_{CDF} &= -\beta_{CCDF} \quad (3.11) \\
p(g \leq z) &= 1 - p(g > z) \quad (3.12)
\end{align}

where $\Phi(\cdot)$ is the standard normal cumulative distribution function. A common convention in the literature is to define $g$ in such a way that the CDF probability for response level $z$ of zero (i.e., $p(g \leq 0)$) is the response metric of interest. Dakota is not restricted to this convention and is designed to support CDF or CCDF mappings for general response, probability and reliability level sequences.

With the Mean Value method, it is possible to obtain importance factors indicating the relative importance of input variables. The importance factors can be viewed as an extension of linear sensitivity analysis combining deterministic gradient information with input uncertainty information, i.e. input variable standard deviations. The accuracy of the importance factors is contingent of the validity of the linear approximation used to approximate the true response functions. The importance factors are determined as:

\[ \text{ImpFactor}_i = \left( \frac{\sigma_{x_i} \frac{dg}{dx_i} (\mu_{x_i})}{\sigma_g} \right)^2 \quad (3.13) \]

In this chapter, optimization capabilities of Dakota optimization open source software were overviewed. Methods for optimization under uncertainty were described. In the following chapter, parameters and challenges of Multi-objective optimization will be presented.
4. Multi-objective Optimization

In the last chapter, optimization capabilities of Dakota optimization open source software were overviewed. In this chapter, Multi-objective optimization will be described, [9].

Multi-objective optimization implies that there are two or more objective functions that we wish to optimize simultaneously. Often these are conflicting objectives, such as cost and performance. The answer to a multi-objective problem is usually not a single point. Rather, it is a set of points called the Pareto front. Each point on the Pareto front satisfies the Pareto optimality criterion, which is stated as follows: a feasible vector is Pareto optimal if there exists no other feasible vector which would improve some objective without causing a simultaneous worsening in at least one other objective. Thus, if a feasible point exists that can be improved on one or more objectives simultaneously, it is not Pareto optimal: it is said to be “dominated” and the points along the Pareto front are said to be “non-dominated”.

Mathematically speaking, a multi-objective problem consists of optimizing (i.e. minimizing or maximizing) several objectives simultaneously, with a number of inequality or equality constraints. The problem can be formally written as follows:

\[
\text{Find } x = (x_i) \forall i = 1, 2, ..., N_{\text{param}} \\
\text{such as } f_i(x) \text{ is a minimum (resp. maximum), } \forall i = 1, 2, ..., N_{\text{obj}}
\]

subject to:

\[
g_j(x) = 0, \quad \forall j = 1, 2, ..., M
\]

\[
h_k(x) \leq 0, \quad \forall k = 1, 2, ..., K
\]

where \(x\) is a vector containing the \(N_{\text{param}}\) design parameters, \((f_i)_{i=1}^{N_{\text{obj}}}\) the objective functions and \(N_{\text{obj}}\) the number of objectives. In this study, only inequality constraints are considered and are prescribed as bounded domains. In other words, upper and lower limits are imposed on all parameters:

\[
x_i \in [x_{i,\text{min}}; x_{i,\text{max}}] \quad i = 1 ... N_{\text{param}}
\]
The objective function \((f_i)_{i=1\ldots N_{\text{obj}}}\) returns a vector containing the set of \(N_{\text{obj}}\) values associated with the elementary objectives to be optimized simultaneously. A common practice to solve such a problem is to use a trade-off between the objectives by linearly combining them using some fixed weights prescribed by the user. The resulting single objective function is optimized. The first limitation of this kind of approach is that the choice of the weights associated with each objective obviously influences the solution of the optimization problem. A bad choice can lead to completely sub-optimal results in comparison with the solution obtained by considering the interrelated objectives in an independent manner. Moreover, this method does not allow access to all the set of optimal solutions.

The Evolutionary Algorithms are semi-stochastic methods, based on an analogy with Darwin’s law of natural selection. Each configuration \(x\) is considered as an individual. The parameters \(x_i\) represent its genes. The main principle is to consider a population of \(N\) individuals, i.e., a set of individuals covering the search domain and to let it evolve along generations (or iterations) so that the best individuals survive and have offsprings, i.e., are taken into account and allow to find better and better configurations.

The characteristics of the EA are based on the approach proposed by Fonseca and Fleming [10]. The genes (sometimes called characters) of the individuals (also called strings or chromosomes) are the \(N_{\text{param}}\) design parameters, encoded using a floating-point representation. The initial population is a set of quasi-random configurations in the domain defined by the limits imposed on the parameters. The creation of a new generation from the previous one is performed by applying genetic operators to the individuals of the present generation. The fitness-based selection is applied. At each generation, the individuals are classified as a function of their corresponding objective values, leading to a rank within the population and finally to a fitness. The probability for an individual to participate in the reproduction process is determined by a probability based on its fitness value, linearly calculated from its rank in the classification. For example, for individual number \(i\) in a group of \(N\) individuals:

\[
\text{Fitness}(i) = \frac{N - \text{rank}(i) + 1}{\sum_j(N - \text{rank}(j) - 1)} 
\] (4.5)
with index $j$ varying from 1 to $N$.

Figure 18 depicts a simple example showing for a group of 10 individuals, the rank values and the corresponding probability to participate in the reproduction process, directly based on its fitness. Using this technique, individuals with equal rank have an equal probability to reproduce. Individuals associated with a higher fitness value have a better chance to survive and to take part in the reproduction process. In this way, better and better generations are generated step by step. EA’s operate on the entire population. Thus, they offer a good potential to explore the whole search space and to avoid local optima. Their good robustness is mainly due to the fact that there is no evaluation of the derivatives of the objective function. Moreover, the process can iterate further even if some evaluations fail.

![Pie chart showing probabilities](image)

**Figure 18** Example for the rank of 10 individuals and the corresponding probability to participate in the reproduction process, represented on a circular diagram [9]

The main drawback associated to EAs in general remains their cost in terms of computing (CPU) time. On the other hand, due to the fact that the evaluations are performed independently, they are easily parallelizable [9].
4.1. Aggregating Functions

The most straightforward approach in handling multiple objectives is the use of an arithmetical combination of all the objectives. Thus, the resulting single function can be studied with any of the single objective algorithms either evolutionary or classical. Aggregating approaches are the oldest mathematical programming methods found in literature.

Applied to Evolutionary algorithm, the aggregate function approach does not require any change to the basic search mechanism. Therefore, it is efficient, simple and of easy implementation. It can be successfully used on simple multi-objective optimizations that present continuous convex Pareto fronts. An example of this approach is a linear sum of weights of the form:

$$\min \left( \sum_{i=1}^{m} w_i f_i(X) \right)$$  \hspace{1cm} (4.6)

where the weight $w_i$ represents the relative importance of the $m$-th objective function. The weighting coefficients are usually assumed to sum at 1:

$$\sum_{i=1}^{m} w_i = 1$$  \hspace{1cm} (4.7)

Aggregate function may be linear or nonlinear. Both types have been used with evolutionary algorithms but, generally speaking, aggregating methods have some limitations in generating complex Pareto fronts.

4.2. Multi-objective Optimization Algorithm MOGA and Pareto Front

In this section, the concept of Pareto frontier is explained, [11].

Much different from single-objective problem, it is difficult to minimize or maximize all objective function simultaneously when objective functions are in trade-off relationship. The Pareto optimal solution is introduced. For instance, in a two-objective optimization problem, each design objective corresponds to a coordinate axis and the goal is to minimize one of the objective and maximize the other. However, both goals cannot be simultaneously achieved for there is a conflicting relationship between
them. Being optimal for one objective implies being suboptimal for the other. From the
definition above, the Pareto optimal solutions are compromise solutions in the feasible
region.

If a large number of design options need to be evaluated and their performance
plotted in the objective function space, the outer boundary of this collection of points
would define the borderline limit beyond which the design could not be further
improved. This is a Pareto line that separates the feasible and infeasible regions, Figure
19. In multi-objective optimization, the Pareto front is defined as the border between the
region of feasible points, for which all constraints are satisfied, and the region of
infeasible points.

![Figure 19 Schematic representation of the concept of Pareto optimal front](image)

If this border is shown as a function of the objectives, it’s a continuous line, or
even a surface in case of more than two objectives, that contains possible optimal
combinations of the objectives.

For example, in the heat exchanger problem, decreasing the value of pressure drop
while keeping the value of temperature increase constant would move the design point
into infeasible region. Increasing the value of pressure drop would move the design
point away from the optimal Pareto line. All points on the Pareto front are considered
equally good until a subjective preference is given.
Multi-objective genetic algorithm (MOGA) mimics the natural selection process in which a superior creature evolves whilst inferior creatures fade out from their population as generations go by [12]. There are many advantages of MOGA such as the capability of exploring a large design space and no gradients are needed. But, the most important one is that MOGA can compute multiple independent objective functions simultaneously in one optimization run without converting multiple objective functions into a single objective by weighted linear combination.

The optimization procedure of MOGA includes the following steps [13]:

1. Generation of initial population: The initial population is simply a collection of designs with design variable values taken as random numbers between the design variable bounds.
2. Clone testing: Each design is tested to ensure it is unique to encourage diversity amongst the initial population.
3. Evaluation of design fitness: A set of objective function values must be computed for each design. The form of the functions is not important. Constraints are computed for each design and used to penalize infeasible designs. A value is stored for each design that indicates how infeasible it is.
4. Reproduction: Reproduction is accomplished in three stages. First, a mating pool is created and all individuals are represented at least once in the mating pool. Individuals with a lower infeasibility value are represented more times in the mating pool, making it more likely that they will be given the opportunity to reproduce. All feasible designs are given an equal chance of reproducing. Individuals are randomly chosen from the mating pool. The number of individuals selected is determined by the size of the population, the number of children produced per crossover and the user input crossover rate.
5. Crossover: two individuals selected for reproduction are chosen as a mating pair. Each mating pair can produce any even number of offspring (value input by user). The offspring will be referred to as children. The children are the result of a convex combination of the parents.
6. Mutation: Children from the crossover stage are randomly selected for mutation. The number of mutations is determined by the number of children, the number of design variables and the user input mutation rate.
7. Insertion of children into population: Once the children have been created and mutated, the process of inserting them into the population begins. This is where Pareto dominance of the designs is considered. If there are infeasible members in the population, the one with the largest infeasibility value is selected for comparison and possible replacement. If the child that it is being compared to is feasible or has a smaller infeasibility value, then the child replaces it. Otherwise, the child is not allowed the entrance into the population. If all members of the population are feasible and the child is feasible, then the population is searched for a member that is dominated by the child. The function that performs the search returns the design that is most dominated by the child. This is done by keeping track of the current worst design as it searches and comparing each successive design to the current worst. If it is found that the child does not dominate any members of the population, then a check is run to see if any members of the population dominate the child. If the child is dominated by a member of the population, then it is not permitted entrance into the population. If the child is not dominated by any member of the population, then the population size is increased and the child is granted entrance. Because of the way new designs are admitted into the population, a sub-optimal design has a chance of remaining within the population and reproducing for a significant number of generations. This can cause the population size to become unnecessarily large which significantly affects the efficiency of the algorithm. Therefore, an additional capability is developed and all dominated designs can be removed from the population. This is termed an introduction of epidemic, as it eliminates all the unfit members of a population. The frequency at which epidemics occur is set by the user.

8. Pareto set quality metrics: Upon completion of the optimization, two quality metrics are computed for the resulting Pareto set. These quality metrics were developed by Wu and Azarm [14] distinct point metric (the design space is divided into a grid, a distinct choice is one that exists alone in a grid location); and cluster metric (gives an indication of how uniformly spaced the final population is, the target value is 1, which corresponds to a Pareto set in which every member is distinct).
5. **Tube Bank Heat Exchanger Problem**

In the previous chapter, Multi-objective optimization algorithm was described. In this chapter, tube bank heat exchanger optimization will be presented, as well as the parameters and settings of the problem.

The problem chosen to be examined is introduced in [9]. A two-dimensional model of a cross-flow tube bank heat exchanger is considered. One possible configuration can be seen in Figure 20. Air enters the domain at $T_{\text{inlet}} = 293$ K and is warmed up by passing between the tubes in which warm fluid flows in the corresponding practical application. The tubes are supposed to have a constant outer wall temperature, $T_{\text{wall}} = 353$ K. The outlet is at atmospheric pressure.

The optimization problem consists of finding the best locations of the tubes to increase heat exchange (i.e. the temperature increase of the fluid) while at the same time limit the pressure loss. The two corresponding numerical parameters to optimize are the average temperature difference $\Delta T$ and pressure drop $\Delta P$ between inflow and outflow.

The domain in Figure 22 is simulated. The Reynolds number is equal to 14, defined using the tube diameter $D = 20$ mm and the uniform velocity at the inlet, $v_{\text{inlet}} = 0.01$ m/s, to ensure laminar and steady flow. The length of the domain has been chosen to prevent any influence of the inflow or outflow boundary conditions on the inter-blade flow. The stability analysis in Barkley and Henderson [15] proved that three dimensional effects first appear at a Reynolds number around 188 in the flow around a cylinder. Furthermore, the flow around a single cylinder is steady up to a Reynolds number of 46 [16].

![Figure 20: The domain is divided into three zones](image-url)
In the present case, four cylinders are simulated and will interact with each other, but due to the low Reynolds number, a steady two-dimensional flow can still be assumed.

5.1. Problem Parameters

The numerical simulation of a physical problem can be performed using various geometries and/or boundary conditions. Here, for the different simulations, the boundary and inlet conditions are the same, only the computational geometries differ.

The outer dimensions of the computational domain, as well as the number of the tubes in the domain, are fixed and only the positions of the tubes inside the computational region are varied. The positions of all four tubes are always changed simultaneously, but their locations are kept within a predefined range to avoid crossing the boundaries. Furthermore, the positions are constrained to prevent overlapping and direct contacts between cylinders. Since all other properties and boundary conditions are constant, these position parameters are the only input parameters of the optimizing algorithm. After defining the computational geometry in Salome and snappyHexMesh, and obtaining a corresponding mesh, the numerical simulation can be performed. Here, OpenFOAM 2.3.0 solves the governing equations of the fluid flow as well as the energy equation. The two dimensional fields of pressure and temperature are obtained in this way and provide the two objective parameters, the average temperature and pressure differences: $\Delta T, \Delta p$ between inflow and outflow. The set of coupled numerical tools to solve the multi-objective optimization is shown in Figure 21.
A key Dakota advantage is access to a broad range of iterative capabilities through a single, relatively simple interface between Dakota and the simulator. Figure 22 depicts a loosely-coupled relationship between Dakota and the simulation code. Such coupling is often referred to as black-box, as Dakota has no (or little) awareness of the internal details of the computational model. Dakota and the simulation code exchange data by reading and writing short data files. Dakota is executed with commands that the user supplies in a text input file which specify the type of analysis to be performed, along with the file names associated with the simulation code. During operation, Dakota automatically executes the simulation code by creating a separate process external to Dakota.
Figure 22  The loosely-coupled interface between Dakota and a simulation code [7]

The solid lines in Figure denote file input/output operations inherent to Dakota or the simulation code. The dotted lines indicate passing or conversion of information. As Dakota runs, it writes out a parameters file containing the current variable values. Dakota then starts the user's simulation code, and when the simulation completes, reads the response data from a results file. This process is repeated until all of the simulation code runs required by the iterative study are complete.

The evaluation of an individual set of parameters (tube positions) requires the following steps:
1. generating the computational geometry (cylinders) in Salome using the position variables provided by Dakota
2. generating the appropriate domain and mesh by snappyHexMesh and blockMesh
3. the CFD simulation, i.e. the solution of the governing equations for the flow variable and the energy on the mesh generated in previous step in OpenFOAM
4. obtaining the values of objective functions (pressure drop, temperature increase) from functions included in swak4Foam
Step 1: Computational Geometry

The geometrical constraints are prescribed in terms of lower and upper limits on the parameters. The positions of the centers of the four cylinders (tubes) of the heat exchanger are given with their two dimensional coordinates, $x_i$ and $y_i$, where $i = 1, 2, 3, 4$. Dakota gives the coordinates to Salome which creates the cylinders. The CAD file from Salome is delivered to snappyHexMesh where the domain is created. The domain is separated by constraints in three non-overlapping zones in x-direction. The first part is the inlet zone and the last one is the outlet zone. The tubes are not allowed to be positioned in the first and last zone to prevent any interaction with the boundary conditions, and tubes’ center coordinates have upper and lower bounds which ensure that there are no intersections with the cyclic boundaries. The middle zone contains the tubes. Overlapping and direct contact are not allowed. This is achieved through constraints embedded in Dakota – collisions and overlapping are detected by checking the minimal distance constraint. If one of the constraints is broken, after the CFD simulation infeasible values of objective functions are assigned to these cases and they are rejected. The radius of the tubes is 10 mm, and minimal distance was set to 21 mm, thus the minimal allowed distance between tubes’ walls was 1 mm. The minimal distance constraint was introduced to ensure that at least one mesh cell can fit between the tubes. The response functions were pressure drop, temperature increase, and six additional distance constraint functions.

![Figure 23 The dimensions of the computational domain](image)

Step 2: Mesh Generation

The initial mesh is produced by blockMesh and then snappyHexMesh includes the cylinders delivered from Salome and refines the mesh around them. An example of the
The typical number of computational cells in a mesh is around 7000.

**Step 3: CFD Simulation**

The solver used in the simulation is buoyantBoussinesqSimpleFoam. The boundary conditions are shown in Figure 25.

On the left a velocity inlet is defined where $U = (0.01 \ 0 \ 0)$ m/s. The air temperature at the inlet is constant $T_{inlet} = 293$ K. Reynolds number is 14, which ensures steady state and laminar flow. Top and bottom boundary conditions are cyclicAMI which, as shown in Figure 26, allow different number and size of the cells on the cyclic boundaries. Wall boundary conditions with a constant temperature $T_{wall} = 353$ K are prescribed on all four tube surfaces. On the right, a pressure outlet condition relaxing to atmospheric pressure is imposed.
The governing equations are solved:

Continuity equation:

\[
\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u_i)}{\partial x_i} = 0 \tag{5.1}
\]

Momentum equation:

\[
\frac{\partial (\rho u_i)}{\partial t} + \frac{\partial (\rho u_i u_j)}{\partial x_j} = \rho g_i - \frac{\partial}{\partial x_i} \left(p + \frac{2}{3} \mu \frac{\partial u_j}{\partial x_j} + \mu \left(\frac{\partial u_i}{\partial x_i} + \frac{\partial u_j}{\partial x_j}\right)\right) \tag{5.2}
\]

Energy equation:

\[
\frac{\partial (\rho e)}{\partial t} + \frac{\partial (\rho e u_j)}{\partial x_j} = \rho g_i u_i - \frac{\partial (\rho u_i)}{\partial x_i} \left(\frac{2}{3} \mu \frac{\partial u_j}{\partial x_j} u_i\right) + \frac{\partial}{\partial x_i} \left[\mu \left(\frac{\partial u_i}{\partial x_i} + \frac{\partial u_j}{\partial x_j}\right) u_i\right] + \\
\frac{\partial}{\partial x_j} \left[\lambda \left(\frac{\partial T}{\partial x_i} + \frac{\partial T}{\partial x_j} + \frac{\partial T}{\partial x_k}\right)\right] + \rho Q \tag{5.3}
\]

The equations are reduced when the flow is isothermal (constant viscosity) and incompressible (constant density).

In flows accompanied by heat transfer, fluid properties are usually functions of temperature. The variations may be small and yet be the cause of the fluid motion. If the density variation is not large, it may be treated as constant in the unsteady and convection terms, and as a variable only in the gravitational term. This is called the Boussinesq approximation. Usual assumption is that the density varies linearly with temperature. If the effect of the body force is included on the mean density in the pressure term, as described in [17], the remaining term can be expressed as:
\[(\rho - \rho_0)g_i = -\rho_0 g_i \beta (T - T_0) \] (5.4)

where \(\beta\) is the coefficient of volumetric expansion.

**Step 4: Post-processing**

After convergence, the temperature difference between the inlet (uniform constant value) and the averaged value along the outlet is computed. The pressure difference between the inlet and outlet averaged pressure values is also computed (swak4Foam). These two differences are the two objectives of the optimization. The resulting temperature and pressure fields of one of the optimum solutions are presented in Figures 27 and 28, respectively.

![Figure 27 Temperature field of one of the optimal solutions](image1)

![Figure 28 Pressure field of one of the optimal solutions](image2)
In this chapter, parameters and settings of the tube bank heat exchanger problem were presented. In the next chapter, results of the optimization and uncertainty quantification will be presented and discussed.
6. Results and discussion

Results of the simulations will be given and discussed in this chapter. First, a comparison between different options of Multi-objective Genetic Algorithm is given. In the second section, uncertainty quantification results are provided for three optimal points obtained from the Pareto front. In the third section, sensitivity analysis for three optimal designs with strong constraints identified by [1] is described. Strong constraints implies less design options are possible. The tubes are constrained to four vertical corridors, each tube is put in its own corridor so collisions are impossible. Illustration is given in Fig 29.

![Figure 29 Domain for the strong constraints [1]](image)

6.1. Multi-objective Genetic Algorithm Options

Multi-objective Genetic Algorithm (MOGA) was used in the optimization of the tube bank heat exchanger. The number of members in the population and number of generations (iterations) were varied, as well as the crossover and mutation rate. Identical seed was defined in every case, and thus the initial population didn’t affect the final outcome of the optimization process. Crossover type was defined as shuffle random. In each new generation, 25% of the parents from the previous generation remained, and 75% of the new population were the new individuals (children). Mutation scale was set to 0.1, which means the genes of the new generation were changed 10%.

For details on physical boundary conditions, OpenFOAM and Salome settings see [1].

First, different number of generations (iterations) were tested. For more generations, better statistical convergence was expected. A comparison of 5, 10, 15 and 20 generations is shown in Fig 30. Each generation had 40 members.
A larger number of generations does show better statistical convergence as it corresponds to a higher number of evaluations \((\text{number of evaluations} \approx \text{number of generations} \times \text{number of members})\). For example, 20 generations resulted in designs with smaller pressure drops corresponding to the same temperature increase. 15 and 20 generations gave very similar results, so 15 generations is better as there was less evaluations. Also, a wider range of temperature increase and pressure drops was covered with a larger number of generations, but the additional area is sparsely covered.

Fig. 31 depicts Pareto fronts obtained for 5, 10, 15, and 50 generations with 80 members. More uniform allocation of points was expected for 50 generations, as well as better coverage of the sparse region.
Figure 31 A comparison of different number of generations, each generation has 80 members

These Pareto fronts show better coverage of the sparse region, but most of the points are located in the center of the plot, especially for the largest number of generations. Again, a larger number of generations results in designs with smaller pressure drops corresponding to the same temperature increase and shows better statistical convergence but is far more computationally expensive. The optimization process with 15 generations has $\approx 15 \times 80 = 1200$ evaluations, the one with 50 generations $\approx 50 \times 80 = 4000$ evaluations, and both Pareto fronts are similar.

In Fig. 32, a comparison of approximately the same number of evaluations but for different number of generations and members is given.
Figure 32 A comparison of two cases with approximately the same number of evaluations

It can be seen that larger number of generations shows better results as it is spread through a wider range of objective functions’ values, but also has some sparse regions. In some areas a larger number of generations gives better designs, and in other a larger number of population members gives designs with same temperature increase but smaller pressure drops.
Figure 33 A comparison of two cases with approximately the same number of evaluations

Figure 33 also depicts two Pareto fronts for two cases with approximately the same number of evaluations. Both show similar results and again the one with a larger number of generations covers a wider range of objective functions but is not statistically converged as there are better designs offered in some areas by the other Pareto front with larger population.
Figure 34 A comparison of different mutation rates for strong and relaxed constraints

Two cases with strong and two cases with relaxed constraints with different mutation rates are shown in Fig. 34. Mutation rate equal to 1 means that every member of the new generation has mutated, and mutation rate equal to 0.75 means that 75% of the new generation has mutated. Pareto front with relaxed constraints with larger mutation rate is wider and better statistically converged compared to the other Pareto with relaxed constraints and smaller mutation rate. A wider range was expected as mutations lead to more pronounced changes in design (positions of the tubes) and offer more solutions. In contrast, Pareto front with strong constraints and smaller mutation rate spreads further than Pareto with strong constraints and higher mutation rate. In this case, higher mutation rate didn’t lead to more widely spread solutions because strong constraints restrict the number of possible designs, i.e. the tubes cannot move freely through the entire domain. Fig 35 also illustrates this conclusion.
Fig. 35 could also explain sparse areas in some of the plots. It can be seen that two types of constraints gave almost identical solutions up to temperature increase of approximately 40.5 K. Strong constraints don't allow dense patterns of the tubes' positions, i.e. all four tubes cannot be positioned closely one to another. This leads to the conclusion that points missing from sparse areas are the cases rejected because collisions and overlaps of the tubes occurred. Also, the initial population (seed) was set up exactly the same for every case [1], to ensure that same settings in different runs would give the same results. This initial population had tubes positioned wide apart, so collisions were unlikely to happen in the next generation. Average number of collisions in one optimization run was 6, and there were 4 cases that did not converge. The cases that did not converge are considered to be transient. The designs with positions of the tubes that caused these transient solutions may also be missing from sparse areas. It is obvious that constraining the tubes to four corridors was a fine idea because wide range of solutions was obtained and there were no needless simulations conducted, as there were no rejected cases.
The total number of crossovers is equal to \( \text{crossover rate} \times \text{number of members in a population} \). A larger crossover rate will lead to a wider range of solutions as new generations change more but a larger number of evaluations is necessary for statistical convergence, as it is shown in Fig. 36.

The results have shown that higher number of evaluations will lead to better statistical convergence as more design options are examined. For approximately the same number of evaluations, higher number of generations will give a Pareto front that spreads through a wider range of objective functions' values but it won’t fully converge and there will be some sparse regions on the front. Points missing from sparse regions are probably rejected because collisions and overlapping of the tubes occurred. Higher mutation and crossover rate will produce more diversity in design options but full convergence cannot be guaranteed.

Significance of optimization is illustrated in the next example. Three logical and obvious designs of the tube bank heat exchanger were assumed and corresponding pressure drop and temperature increase calculated. The designs and corresponding temperature and pressure fields are presented in Figures 37, 38 and 39.

![Figure 36 A comparison of different crossover rates](image)
Figure 37  Custom design 1 – temperature and pressure field

Figure 38  Custom design 2 – temperature and pressure field
The tubes are lined up horizontally one behind the other in the first case, so very low pressure drop is expected. In the second case, they are lined up diagonally and higher pressure drop is expected, but also temperature increase. In the third case, the pattern of the tubes’ positions is criss-cross, and higher pressure drop than design 1 is expected, as well as higher temperature increase. The positions of the corresponding points in objective function space for each of the three custom designs are shown in Figure 40.

**Figure 39 Custom design 3 – temperature and pressure field**
The first custom design is one of the optimal designs as it lies on the Pareto front, but the other two are not optimal as they are located away from the Pareto front, in the feasible region. Benefits of multi-objective optimization are pointed out, because, even for very simple designs, it is difficult to find numerous optimal solutions experientially. Pareto front offers numerous optimal solutions and decision maker can give additional subjective preference and choose the best option corresponding to the specific problem or system.

6.2. Uncertainty Quantification

Uncertainty quantification was done for three optimal points obtained from the optimization process. The three points were chosen from three different regions of the Pareto front: left, middle and right. Methods implemented in Dakota were utilized for the analysis: sampling techniques (Latin Hypercube and Monte Carlo) and reliability methods (Mean Value). The uncertainty was introduced through input variables. In the first case, a distribution was given for one of the tubes' coordinates \((x_1, y_1)\). In the second case, a distribution for the temperature of the tubes \((T_{\text{wall}})\) was given.
6.2.1. Uncertainty Analysis of Design 1 with Relaxed Constraints

The position of the point on the Pareto front which corresponds to Design 1 is shown in Fig 41.

![Figure 41 The black triangle denotes Design 1 point on the Pareto front](image)

Temperature and pressure fields for Design 1 are shown in Figures 42 and 43, respectively.

![Figure 42 Temperature field of Design 1](image)
The coordinates of the first tube on the left are $x_1 = 0.06694869 \text{ m}$ and $y_1 = 0.03221098 \text{ m}$. The corresponding temperature increase is $\Delta T = 28.0194 \text{ K}$, and pressure drop is $\Delta p / \rho = 0.00013923 \text{ m}^2/\text{s}^2$, on the plots denoted just as pressure drop $\Delta p$. The value of the pressure drop is taken directly from OpenFOAM and that is why the dimension is not Pa. To obtain the pressure drop in Pa, it has to be multiplied by density ($\rho = 1.225 \text{ kg/m}^3$).

**Uncertain variables** = $x_1, y_1$; **Response function** = pressure drop

The pressure drop becomes the only response function and the effect of the uncertain coordinates of the first tube is analyzed.

The normal distributions of the uncertain coordinates are shown in Figures 44 and 45.
The mean value of $x_1$ is $\mu_{x_1} = 0.06694869$ m, and the mean value of $y_1$ is $0.03221098$ m. Note that they are the same as the original design values, chosen such as to represent the optimal design. Standard deviation is identical for both variables, $\sigma = 0.002$ m. The standard deviation is very small, so the value of the density function in Figures 44 and 45 exceeds 1 as explained in Chapter 1.

In Figure 46, a resulting cumulative distribution function is given for three methods. The cumulative distribution function describes the probability that the pressure drop will have a value less than or equal to the response level which was prescribed in Dakota. The response levels were chosen close to the expected value (pressure drop of the original design) of the objective function, as there were very small variations in the tubes’ positions. (standard deviation is 10% of tube’s diameter). To evaluate the design's stability, steepnes of the cumulative distribution function must be observed. If it is very steep (similar to a step function), the design is stable.
For example, according to the distribution given by Latin Hypercube sampling, there is a 67% chance that the pressure drop will be less or equal to $0.00014 \times 1.225 = 0.000172$ Pa.

Latin Hypercube and Monte Carlo sampling gave similar results, both drawn 200 samples. The functions aren't smooth because they are derived from samples and uniform coverage of the whole objective function space cannot be ensured. Random sampling (Monte Carlo) gave worse results than Latin Hypercube, as expected, because random samples don’t have to be representative.

Mean Value gave a smooth curve, but there is a large difference from the sampling methods. This may be caused by the size of the relative gradient step size. Different step sizes are illustrated in Fig. and it can be seen that larger step sizes give curves closer to sampling methods’ curves (1e-1 case comes very close). Small gradient step sizes may cause poor results if the method doesn’t do evaluations far enough from the mean value of the uncertain variables. Mean Value method does only 5 evaluations, and Latin Hypercube and Monte Carlo use 200 samples (200 evaluations), so changing and examining gradient step size isn’t very time consuming if simulations run relatively fast.

Mean Value method returns importance factors which are a measure of the sensitivity of the response function to the uncertain input variables. Their physical
meaning is described in Chapter 2.7.4. The importance factor calculated for $x_1$ is 0.987 and for $y_1$ 0.013, which means that shifting the first tube in x-direction affects the pressure drop significantly. It can be seen in Fig. 42. that first two tubes are close to each other (the second tube is in the wake of the first one), so dragging them closer (or further apart) in x-direction affects the pressure drop, i.e. the stability of the flow. This represents important information if the design needs to be modified to meet certain construction demands, changing tube’s position in y-direction won’t change the output of the system significantly.

The sampling techniques calculate the mean value and standard deviations of the evaluations’ results, as well as skewness and kurtosis of the normal distribution. Figures 47 and 48 show the histograms which were plotted from the results given by the methods. The histograms show the density of pressure drop intervals, i.e. it can be seen that for varying the first tube’s coordinates, the pressure drop will most probably be close to 0.000139 m$^2$/s$^2$, as the density of that interval is the highest. The result is expected, as the value of the pressure drop for the original design was $\Delta p =$ 0.00013923 m$^2$/s$^2$.

Histogram derived from Monte Carlo method is given to illustrate the difference between sampling methods. Density of some bins is different for two methods, Latin Hypercube is closer to normal distribution. This was expected, as Monte Carlo takes random samples, and Latin Hypercube spreads the samples evenly, both considering the prescribed distribution of uncertain variables.
The histograms depict the distribution of the pressure drop well and are convenient to use to validate the calculated values. The mean ($\mu_{\Delta p} = 0.0001395 \text{ m}^2/\text{s}^2$) and standard deviation ($\sigma = 0.0000017 \text{ m}^2/\text{s}^2$) were given by sampling methods. Values for skewness (1.33) and kurtosis (6.04) were also calculated.

Probability density function could be drawn on the histograms and it would be positively skewed.
Stability of the design can also be evaluated from the histograms. If the histogram is very dense in some area, i.e. the function drawn on the histogram is very raised (leptokurtic), the design is stable. If the function is platykurtic, the design is unstable.

**Uncertain variable** = $T_{\text{wall}}$; **Response function** = temperature increase

The normal distribution of the uncertain input variable, tubes’ wall temperature, is shown in Figure 49. Mean value of the distribution is $\mu_{\text{temp,wall}} = 353$ K and standard deviation is $\sigma = 2$ K.

![Figure 49 Distribution of the uncertain variable, $T_{\text{wall}}$](image)

The cumulative distribution function given by Latin Hypercube sampling (200 samples) and Mean Value reliability method is depicted in Figure 50.
Figure 50  Cumulative distribution function for temperature increase (Design 1)

Two cumulative probability curves for two different methods match perfectly, as there is only one uncertain variable and positions of the tubes are constant (no changes in the appearance of the flow), so there is less numerical noise for the Mean Value method which employs numerical gradients. The slope of the curve is mild, i.e. temperature increase is sensitive to the variances of the tubes’ temperature, as expected.

Figure 51  Density value for temperature increase
The calculated mean value is $\mu_{\Delta temp} = 28.0198$ K and standard deviation is $\sigma = 0.9275$ K. The distribution of the output variable corresponds to the distribution of the input variable, but is slightly negatively skewed (Fig. 51).

6.2.2. Uncertainty Analysis of Design 2 and Design 3 with Relaxed Constraints

The position of the points on the Pareto front which correspond to Design 2 (green) and Design 3 (red) are shown in Fig 52.

![Figure 52](image.png)

Figure 52 The green triangle denotes Design 2 point and the red triangle denotes Design 3 on the Pareto front

Temperature and pressure fields for Design 2 are shown in Figures 53 and 54, respectively, and for Design 3 in Figures 55 and 56. The original design is similar for both cases, only the fourth tube in x-direction changes it’s position significantly.

![Figure 53](image.png)

Figure 53 Temperature field of Design 2
The coordinates of the first tube of Design 2 are $x_1 = 0.05440711$ m and $y_1 = 0.05365305$ m. The corresponding temperature increase is $\Delta T = 34.2402$ K, and pressure drop is $\Delta p = 0.00019562$ m$^2$/s$^2$. For Design 3, the corresponding values are: $x_1 = 0.05497646$ m, $y_1 = 0.04896847$ m, $\Delta T = 38.3614$ K, and $\Delta p/\rho = 0.00024937$ m$^2$/s$^2$. 
Uncertain variables = $x_1, y_1$; Response function = pressure drop

Uncertain coordinates are normal distributed with mean values equal to coordinates $x_1$ and $y_1$ of the original Designs. Standard deviation is identical for both Designs and all variables, $\sigma = 0.002 \text{ m}$.

In Figures 57 and 58, cumulative distribution functions for pressure drop are given for three methods.

**Figure 57** Cumulative distribution function for pressure drop (Design 2)

**Figure 58** Cumulative distribution function for pressure drop (Design 3)
Different number of samples (200 and 1000) in Latin Hypercube sampling gave almost identical results. Monte Carlo shows worse results, similar to Design 1.

Mean Value gave a smooth curve, and there is a much smaller difference to the sampling methods compared to Design 1 which would suggest there was less numerical noise in these cases.

The cumulative distribution functions are less steep than for Design 1, so Designs 2 and 3 are less stable.

The importance factors calculated for $x_1$ are 0.03 (Design 2) and 0.01 (Design 3) and for $y_1$ 0.97 (Design 2) and 0.99 (Design 3), which means that shifting the first tube in y-direction affects the pressure drop significantly.

Figures 59 and 60 show the histograms which represent density values corresponding to pressure drop intervals.

![Figure 59 Relative density function for pressure drop (method: Latin Hypercube sampling, Design 2)](image-url)
Latin Hypercube gave better results for Design 3 than Design 2, which can be seen when comparing the histograms. Further analysis has shown that original mesh density isn’t fine enough, because the changes that are considered here are two orders of magnitude smaller (Figure 61). Coarser mesh has given good optimization results.
compared to literature [9], but finer meshes should be used for uncertainty quantification in this particular case.

Increasing the number of response levels can be considered only if sample size is increased as well. Higher number of response levels with the same sample size gives more bins (intervals) and less samples will be distributed into each bin.

The means: 0.000195 m²/s² (Design 2), 0.000249 m²/s² (Design 3) and standard deviations: 0.0000043 m²/s² (Design 2), 0.00000388 m²/s² (Design 3) were given by sampling methods.

**Uncertain variable =** $T_{wall}$; **Response function = temperature increase**

The normal distribution of the uncertain input variable, tubes’ wall temperature, is the same as for Design 1, shown in Figure 41.

The cumulative distribution functions given by Latin Hypercube sampling (200 samples) and Mean Value reliability method are depicted in Figures 62 and 63.

![Cumulative distribution function for temperature increase (Design 2)](image)

**Figure 62 Cumulative distribution function for temperature increase (Design 2)**
Two cumulative probability curves for two different methods match perfectly, as for Design 1. The slope of the curve is mild, i.e. temperature increase is sensitive to the variances of the tubes’ temperature, as expected.

Figure 64  Density value for temperature increase (Design 2)
The calculated mean values are 34.2407 K, 38.3619 K and standard deviations are 1.13345 K, 1.26987 K, for Design 2 and Design 3, respectively. The distributions are slightly negatively skewed, Fig. 64 and 65.
6.2.3. Uncertainty Analysis of Designs with Strong Constraints

The stability of three points given by [1] is assessed by using the uncertainty quantification methods. The three design points were chosen from different areas of the corresponding Pareto fronts. The positions of the points are shown in Fig. 66.

Figure 66  The positions of the three designs from [1] on the corresponding Pareto fronts

Temperature and pressure fields for Designs are shown in Figures 67, 68 and 69 respectively.
Figure 67  Temperature and pressure field of Design 1

Figure 68  Temperature and pressure field of Design 2
Figure 69  Temperature and pressure field of Design 3

Table 1  Results of the uncertainty quantification for designs with strong constraints

<table>
<thead>
<tr>
<th></th>
<th>DESIGN 1</th>
<th>DESIGN 2</th>
<th>DESIGN 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original coordinates</td>
<td>$x_1 = 0.070291$ m</td>
<td>$x_1 = 0.054192$ m</td>
<td>$x_1 = 0.0575448$ m</td>
</tr>
<tr>
<td>(mean values)</td>
<td>$y_1 = 0.034583$</td>
<td>$y_1 = 0.050656$ m</td>
<td>$y_1 = 0.0504314$ m</td>
</tr>
<tr>
<td>Original pressure drop</td>
<td>$\Delta p/\rho = 0.000134484$ m$^2$s$^{-2}$</td>
<td>$\Delta p/\rho = 0.00019526$ m$^2$s$^{-2}$</td>
<td>$\Delta p/\rho = 0.00024404$ m$^2$s$^{-2}$</td>
</tr>
<tr>
<td>Original temperature</td>
<td>$\Delta T = 27.2558$ K</td>
<td>$\Delta T = 34.4228$ K</td>
<td>$\Delta T = 38.0548$ K</td>
</tr>
<tr>
<td>increase</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standard deviation of</td>
<td>$\sigma = 0.002$ m</td>
<td>$\sigma = 0.002$ m</td>
<td>$\sigma = 0.002$ m</td>
</tr>
<tr>
<td>the coordinates</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M. value and st. dev.</td>
<td>$\mu_{\text{temp,wall}} = 353$ K</td>
<td>$\mu_{\text{temp,wall}} = 353$ K</td>
<td>$\mu_{\text{temp,wall}} = 353$ K</td>
</tr>
<tr>
<td>for tubes’ wall</td>
<td>$\sigma = 2$ K</td>
<td>$\sigma = 2$ K</td>
<td>$\sigma = 2$ K</td>
</tr>
<tr>
<td>temperature</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The results of the uncertainty quantification of the three designs with strong constraints are presented in Table 1. Cumulative distribution function is shown for pressure drop (blue is Latin Hypercube sampling and red is Mean Value reliability method) and density value histograms are shown for pressure drop and temperature increase.

Cumulative distribution function for Design 1 is different for Mean Value and Latin Hypercube. Too small gradient step size may be the reason of this disagreement. Density value histograms give expected results, as the design is more sensitive to temperature increase.
Cumulative distribution functions for both Design 2 and 3 are very similar for both methods. Density histograms for pressure drops of Designs 2 and 3 didn’t come out as expected as there is a high dissipation. As it has been shown before, finer computational mesh may resolve this problem. Density histograms for temperature increase show sensitivity to tubes’ wall temperature changes. Better distribution of these histograms could be achieved if wider range of response levels was prescribed, as the tails of the distribution seem to expand further. Design 1 shows less sensitivity to temperature changes because it’s initial temperature increase is the smallest (tubes aligned horizontally).
7. Conclusion

Optimization is a discipline of numerical mathematics which aims to improve the operation of a system or process as good as possible in some defined sense. Optimization algorithms work to minimize (or maximize) an objective function, typically calculated by the user simulation code, subject to constraints on design variables and responses.

Parameters and settings of tube bank heat exchanger multi-objective optimization were described. The problem consists of finding the best locations of the tubes to increase heat exchange (i.e. the temperature increase of the fluid) while at the same time limit the pressure loss. The two corresponding numerical parameters to optimize are the average temperature difference $\Delta T$ and pressure drop $\Delta P$ between inflow and outflow.

The set of coupled numerical tools to solve the multi-objective optimization consists of open source optimization software Dakota, open source CFD toolbox OpenFOAM and open source software for geometry creation Salome. Multi-objective Genetic Algorithm is used to obtain optimal designs.

Results of the optimization process are presented on corresponding Pareto fronts. Different parameters of the Multi-objective Genetic Algorithm were examined and discussed.

It was shown that higher number of evaluations gives more solutions and better statistical convergence of the Pareto front. Higher mutation rate and crossover rate also give more possible solutions, as members of the population change more. Designs with strong and relaxed constraints were also examined and they give similar results.

Uncertainty quantification or nondeterministic analysis is the process of characterizing input uncertainties, forward propagating these uncertainties through a computational model, and performing statistical or interval assessments on the resulting responses. Uncertainty quantification is related to sensitivity analysis in that the common goal is to gain an understanding of how variations in the parameters affect the response functions of the engineering design problem. For uncertainty quantification, some or all of the components of the parameter vector, are considered to be uncertain as specified by particular probability distributions.

Effect of uncertain tube coordinates and uncertain tubes’ temperature was examined and presented in graphs and histograms.
Depending on the original positions of the tubes, designs are more or less stable, i.e. objective functions distributions correspond to uncertain variables' distributions, however, mesh density effect on the solution should be further examined.

Single-objective optimization wasn't conducted [1], and further development of the method should be included in future work.
References


