LICENTIATE THESIS

Metamodel Based Multi-Objective Optimization

Kaveh Amouzgar

JÖNKÖPING UNIVERSITY
School of Engineering

Department of Product Development
SCHOOL OF ENGINEERING, JÖNKÖPING UNIVERSITY
Jönköping, Sweden 2015
Metamodel Based Multi-Objective Optimization
Kaveh Amouzgar

Simulation and Optimization Group
Department of Product Development
School of Engineering, Jönköping University
SE-551 11 Jönköping, Sweden
kaveh.amouzgar@ju.se

Research Series from the School of Engineering, Jönköping University
Department of Product Development
Dissertation Series No. 13, 2015
ISBN 978-91-87289-14-9

Published and Distributed by
School of Engineering, Jönköping University
Department of Product Development
SE-551 11 Jönköping, Sweden

Printed in Sweden by
Ineko AB
Källered, 2015
To my mother, Mahnaz
Abstract

As a result of the increase in accessibility of computational resources and the increase in the power of the computers during the last two decades, designers are able to create computer models to simulate the behavior of a complex products. To address global competitiveness, companies are forced to optimize their designs and products. Optimizing the design needs several runs of computationally expensive simulation models. Therefore, using metamodels as an efficient and sufficiently accurate approximate of the simulation model is necessary. Radial basis functions (RBF) is one of the several metamodeling methods that can be found in the literature.

The established approach is to add a bias to RBF in order to obtain a robust performance. The a posteriori bias is considered to be unknown at the beginning and it is defined by imposing extra orthogonality constraints. In this thesis, a new approach in constructing RBF with the bias to be set a priori by using the normal equation is proposed. The performance of the suggested approach is compared to the classic RBF with a posteriori bias. Another comprehensive comparison study by including several modeling criteria, such as problem dimension, sampling technique and size of samples is conducted. The studies demonstrate that the suggested approach with a priori bias is in general as good as the performance of RBF with a posteriori bias. Using the a priori RBF, it is clear that the global response is modeled with the bias and that the details are captured with radial basis functions.

Multi-objective optimization and the approaches used in solving such problems are briefly described in this thesis. One of the methods that proved to be efficient in solving multi-objective optimization problems (MOOP) is the strength Pareto evolutionary algorithm (SPEA2). Multi-objective optimization of a disc brake system of a heavy truck by using SPEA2 and RBF with a priori bias is performed. As a result, the possibility to reduce the weight of the system without extensive compromise in other objectives is found.

Multi-objective optimization of material model parameters of an adhesive layer with the aim of improving the results of a previous study is implemented. The result of the original study is improved and a clear insight into the nature of the problem is revealed.
First of all I would express my sincerest gratitude to my current and previous supervisors, Professor Peter Hansbo and Professor Niclas Strömberg for all the support, advice and provisions. I also would like to thank my co-supervisor Professor Kent Salomonsson for his support, discussions and guidance. The expert knowledge from each of their fields and pedagogical skills provided a superb and relaxed learning environment for which I am deeply grateful for. In addition, a thank you to Professor Amos H.C. Ng for his valuable comment on this thesis. I would like to thank my colleague at the department for the easy going atmosphere that each and everyone created.

Finally, and most importantly, my deepest gratitude goes to my family.
Supplements

The following supplements constitute the basis of this thesis.

Suplement I  
K. Amouzgar, A. Rashid, N. Strömberg *Multi-Objective Optimization of a Disc Brake System by using SPEA2 and RBFN*. In proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, IDETC/CIE 2013, August 4-7, Portland, USA.

*K. Amouzgar was the main author. A. Rashid wrote the disc brake simulation section and N. Strömberg contributed with advice regarding the work.*

Suplement II  
K. Amouzgar, N. Strömberg *An approach towards generating surrogate models by using RBFN with a priori bias*. In proceedings of the ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, IDETC/CIE 2013, August 17-20, Buffalo, USA.

*K. Amouzgar was the main author and N. Strömberg contributed with advice regarding the work.*

Suplement III  
K. Amouzgar, M. Cenanovic, K. Salomonsson *Multi-objective optimization of material model parameters of an adhesive layer by using SPEA2*. In the proceedings of the 11\textsuperscript{th} World Congress on Structural and Multidisciplinary Optimization, WCSMO-11 2015, June 7-12, Sydney, Australia.

*K. Amouzgar and M. Cenanovic were the main authors and K. Salomonsson contributed with advice regarding the work.*

Suplement IV  
K. Amouzgar, N. Strömberg *Radial basis functions with a priori bias in comparison with a posteriori bias under multiple modeling criteria*. To be submitted to the journal of Structural and Multidisciplinary Optimization.

*K. Amouzgar was the main author and N. Strömberg contributed with advice regarding the work.*
## Contents

Abstract i
Acknowledgements iii
Supplements v

### Introduction
1 Multi-Objective Optimization 3
   1.1 Background ........................................ 3
   1.2 Classifications .................................... 4
   1.3 Multi-objective optimization approaches .............. 5
   1.4 Evolutionary methods ................................ 6

2 Metamodels 9
   2.1 Background .......................................... 9
   2.2 Design of Experiments (DoE) .......................... 10
   2.3 Metamodeling methods .............................. 10
   2.4 Radial basis functions ............................. 11
      2.4.1 RBF with a posteriori bias ...................... 12
      2.4.2 RBF with a priori bias ......................... 13
   2.5 Metamodel validation .............................. 13

3 Summary of appended papers 17
   3.1 Supplement I ....................................... 17
   3.2 Supplement II ...................................... 17
   3.3 Supplement III .................................... 18
   3.4 Supplement IV .................................... 18

4 Future works 19
Contents

Bibliography 20
Appended Papers 27
Supplement I 29
Supplement II 41
Supplement III 53
Supplement IV 61
Introduction

Introduction and motivation

As a result of increasing global competitiveness, developing complex and successful products with fast lead times and lower price has become the aim of manufacturing companies. For a complex product such as a car or an aircraft, several groups from different disciplines must interact simultaneously during the development process. The disciplines must be able to work in parallel and the knowledge and skills of the different groups need to be used during the product development process. Their final goal is to find a design that is optimal with regards to all disciplines, rather than a design that is optimized to each discipline sequentially. The field that uses optimization methods to find an optimal solution of a design of a system that involve a number of disciplines or subsystems is called multidisciplinary design optimization (MDO).

The design of a complex product requires finding the response of the product under external loads. The response can be obtained by performing physical experiments or computer simulations. Optimizing the design needs several evaluations of the response in an iterative manner. Therefore, the integration of the optimization methods with computational models/simulations is inevitable and designers are attracted to simulation based designs in order to predict the behavior of their product before producing expensive physical prototypes. Despite the exponentially increasing computing power, the complexity of models, such as finite element analysis and computational fluid dynamics are expanding as well. Design and analysis of computer simulations during MDO was first introduced by Sacks et al. [38] in 1989.

The use of approximation based optimization methods have gained attention recently. In these methods the complex and computational expensive model is replaced with a simple analytical function. The analytical function approximates the actual model and is called metamodel, i.e., model of a model. The process of constructing the metamodel is called metamodeling.

The optimization process can be carried out using the metamodel instead of the complex model, which is referred as metamodel-based design optimization (MBDO). A comprehensive review on the use of the metamodeling in MDO can be found in Simpson et. al [44].

In addition to model approximation, metamodels can be beneficial in design space exploration, problem formulation and optimization support according to Wang and Shen [47].

In real world engineering design, more than one objective is most often targeted simultaneously during
Contents

the optimization process, which is called multi-objective optimization (MOO). For example in the aerospace industry, basically the main aim is to reduce the weight, cost and fuel consumption of an aircraft, while maximizing the performance and safety. The rapid growth of computer power changes the focus of researchers to global optimization approaches rather than local search methods.

Multi-objective optimization of an engineering problem requires several evaluations of each objective within the design space, i.e., the space of all design variables, leading to a large number of simulation runs. Depending on the complexity of the problem each run could entail hours of computation time to find a set of optimal solutions. Therefore, coupling computational tools with mathematical procedures or algorithms for design optimization is practically difficult and computationally expensive. Thus, the urge to employ metamodels in MOO is more substantial. Metamodel-based multi-objective optimization (MB-MOO), seems to be an effective approach in MOO of complex designs and products. Several publications target the idea of employing metamodels in MOO [3, 13, 24, 28, 30, 31, 32, 37].

Aim and delimitations

MB-MOO is a broad field that can be used in different applications. However, here the aim is to only study and perform MB-MOO on engineering applications.

For the metamodeling part the focus is only on the radial basis functions (RBF) as the metamodeling method used in multi-objective optimization problems (MOOP). A new approach towards generating the RBF is proposed. The bias in the proposed approach is considered to be known a priori while in the classic RBF the bias is unknown from the beginning and an orthogonality condition is imposed to find the bias. The strength Pareto evolutionary algorithm (SPEA2) is the method employed in this thesis for solving multi-objective optimization problems. MB-MOO of disk brake system of a heavy truck with three objectives and three design variables are performed. MOO of the material model parameters of an adhesive layer based on a previous study is conducted.

Outline

The thesis is structured by giving a short background on optimization in general and different classifications of optimization methods in the next chapter. Multi-objective optimization and the approaches for finding the solutions are also presented. In chapter 2, metamodels, design of experiments and metamodel validation are described. Also the proposed RBF approach is defined in detail. In chapter 3, the summary of the appended papers is given. Finally, ideas for future works are discuss in chapter 4.
1 Multi-Objective Optimization

1.1 Background

Optimization is the process of making things better. Life is full of optimization problems which all of us are solving multiple times each day in our life.

Haupt et al. [18] defines optimization as fine-tuning the input of a process, function or device to find the maximum or minimum output(s).

A general optimization problem is formulated as:

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad g(x) \leq 0 \\
& \quad h(x) = 0 \\
& \quad x_{\text{lower}} \leq x \leq x_{\text{higher}}
\end{align*}
\]

The main aim in an optimization problem is to find the values of the design variables \(x\) that minimize the objective function \(f\). The formulation in eq. 1.1, can also be used for maximizing the \(f\) by minimizing \(-f\). The design variables are usually bounded within a range of lower and higher limit. The design variables can be continuous or discrete. The continuous variables can take any value, while the discrete ones can be chosen from certain discrete values, within the design range. Most of the optimization problems are constrained by inequality or equality constraints, shown by vectors \(g\) and \(h\). The design variables that satisfy the constraints are called feasible, and the others are infeasible design points. A linear programming (LP) problem is an optimization problem with linear objective and constraints. If either any of the objective or constraints is non-linear, it is called non-linear programming (NLP). The objective and constraint functions can be in the form of an analytical function. Also, they can be an experiment or a model that is defined by the governing equations, e.g., a finite element model. It can be a model of a model or metamodel of a complex model. The solution of an optimization problem is called the optimum solution.
1 Multi-Objective Optimization

1.2 Classifications

Since engineering design optimization problems can be formulated and defined in several different ways, a single optimization solver, method or algorithm suitable for all problems does not exist. Numerous classifications, categorizations or grouping of different optimization algorithms are proposed in the literature. Deb [9] classifies the optimization algorithms into five main groups:

1. Single-variable optimization algorithms. The algorithms in this group are categorized for solving optimization problems with a single variable. Deb classifies this group into two general methods; direct and gradient methods. The later uses the information from the first or second order derivative of the function to direct the search towards the optimum, while the first do not employ the derivatives.

2. Multi-variable optimization algorithms. This group is also divided into two direct and gradient based methods. It is obvious that the algorithms in this group are used for solving problems with more than one variable.

3. Constrained optimization algorithms. Constraints were not involved in the first two groups. In this group, which is mostly used in engineering problems, constraints are handled by algorithm in different ways. The problem can be a single or multi variable constraint optimization problem.

4. Specialized optimization algorithms. Some algorithms are designed to be ideal for a specific type of optimization problem, e.g., integer programming and geometric programming. They are also called traditional optimization algorithms, because they have been in use since the sixties.

5. Non-traditional optimization algorithms. This group of algorithms have mostly been developed during recent years and are relatively new. Evolutionary algorithms (EA), simulated annealing and swarm algorithms are some popular examples.

Local and global optimization algorithms is another classification reported in the literature. The first group of algorithms only tries to find a local optimum close to the initial solution, while the global algorithms have a better chance of finding the global or near global optimum solution. Traditional as well as most of the gradient based algorithms are parts of the local optimization algorithms. The non-traditional algorithms have better global properties. Multimodal optimization algorithms are aimed at finding several optima, in problems with multiple global solutions [10].

The global optimization algorithms can also be classified into deterministic and stochastic (or heuristic) algorithms [9, 48]. In deterministic algorithms the objective and constraint functions will follow the same sequence of state during the optimization run. The stochastic algorithms are inspired by nature. They are suitable for discrete optimization problems and large design spaces, unlike the deterministic algorithms. On the other hand, finding an optimal solution is not guaranteed and they usually require more objective function evaluations.
1.3 Multi-objective optimization approaches

When only one objective function is involved in the problem, it is called single-objective optimization, as defined in (1.1). However, in most of engineering problems more than one objective function has to be minimized, leading to multi-objective optimization. A multi-objective optimization problem (MOOP) can be formulated as

\[
\begin{align*}
\min_x & \quad f_1(x), \ldots, f_m(x) \\
\text{subject to} & \quad g(x) \leq 0 \\
& \quad h(x) = 0 \\
& \quad x_{\text{lower}} \leq x \leq x_{\text{higher}}
\end{align*}
\]  

(1.2)

In real-world applications, most of the optimization problems involve more than one objective. The objectives in engineering problems are often conflicting, e.g., to maximize performance, minimize cost, maximize reliability, etc. In this case, one extreme solution would not satisfy both objective functions and the optimal solution of one objective will not necessarily be the best solution for the other objective(s). Therefore different solutions will produce trade-offs between different objectives and a set of solutions is required to represent the optimal solutions of all objectives. The variable bounds of an optimization problem restrict each decision variable to a lower and upper limit which institutes a space called a decision variable space. In multi-objective optimization, values of the objective functions create an m-dimensional space called objective space. Each decision variable in the variable space corresponds to a point in objective space. A solution within the feasible design space is non-dominated or Pareto optimal if none of the objective functions can be improved by a feasible solution without worsening at least one other objective function. The set of all feasible solutions that are non-dominated by any other solution is called the Pareto optimal set or non-dominated set. If the non-dominated set is within the entire feasible search space, it is called globally Pareto optimal set. The values of objective functions related to each solution of a Pareto optimal set in objective space is called a Pareto-front. For two, three or more conflicting objectives, the Pareto-front is respectively a curve, a surface or a hyper-surface.

Deb [8] classifies the approaches towards solving MOOP in two groups.

- Ideal multi-objective optimization, where a set of solutions in form of a trade-off curve is obtained and the desired solution is selected according to some higher level information of problem.

- Preference based multi-objective optimization, which by using the higher level information a preference vector transforms the multi-objective problem to a single-objective optimization. The optimal solution is obtained by solving the single-objective problem.

Deb further assumes that there are two goals in multi-objective optimization:

1. To find a set of non-dominated solutions with the least distance to the Pareto optimal set.
1 Multi-Objective Optimization

2. To have maximum diversity in the non-dominated set of solutions.

Extensive studies have been conducted in multi-objective optimization algorithms. However, most of the research avoids the complexity in the true multi-objective optimization problem by transforming the problem into single-objective optimization by use of some user defined parameters. The straightforward approach for solving a MOOP is to convert the objectives to one objective function by means of different techniques. Two popular methods are proposed for this transformation [16, 49]. The weighted sum method in which the objectives are converted to one objective by multiplying the sum of objectives to a weight vector. The other technique is considering all objectives except one as constraints and limiting them by a user defined value (\(\epsilon\)-constraint). Deb very well presents some of the most important similar methods, called classical methods by him, in one chapter of his book [8].

1.4 Evolutionary methods

The characteristic of evolutionary methods which use a population of solutions that evolve in each generation is well suited for multi-objective optimization problems. Since one of the main goals of MOOP solvers is to find a set of non-dominated solutions with the minimum distance to the Pareto-front, evolutionary algorithms can generate a set of non-dominated solutions in each generation. Low requirement of prior knowledge from the problem, less vulnerability to shape and continuity of the Pareto-front, ease of implementation, robustness and the ability to be carried out in parallel are some of the advantages of evolutionary algorithms listed in [21].

The first goal in evolutionary multi-objective optimization is achieved by a proper fitness assignment strategy and a careful reproduction operator. Diversity in the Pareto-set can be obtained by designing a suitable selection operator. Preserving the elitism during generations must be carefully considered in evolutionary algorithms. Elite preserving operators, as Deb calls them in [8], are introduced to directly carry over the elite solutions to the next generation. Coello [5] presents the basic concepts and approaches of multi-objective optimization evolutionary algorithms. The book further explores some hybrid methods and introduces the test functions and their analysis. Various applications of multi-objective evolutionary algorithms (MOEA) are also discussed in the book. Debs book [8] is another comprehensive source of different MOEAs. The book divides the evolutionary algorithms into non-elitist and elitist algorithms.

All researchers agree upon that the invention of the first MOEA should be attributed to David Schaffer with his Vector Evaluation Genetic Algorithm (VEGA) in the mid-1980s, aimed at solving optimization problems in machine learning [40].

The most representative, discussed and compared evolutionary algorithms are Non-dominated Sorting GA (NSGA-II) [12], Strength Pareto Evolutionary Algorithm (SPEA, SPEA2) [52, 51], Pareto Archived Evolution Strategy (PAES) [26, 27], and Pareto Enveloped Based Selection Algorithm (PESA, PESA II) [6, 7].

Extensive comparison studies and numerical simulation on various test problems shows a better overall behavior of NSGA-II and SPEA2 compared to other algorithms. In cases where more than two objectives
1.4 Evolutionary methods

are present, SPEA2 seems to indicate advantages over NSGA-II [51]. However, the new version of the well known NSGA-II, has recently been developed by Deb et. al. [11, 20] named NSGA-III. They have improved the performance of the older version in many objective optimization problems. They tested the algorithm on problems with 3 to 15 objectives and reported satisfactory results.

In the present thesis, the SPEA2 algorithm is coded in MATLAB and used in the supplements. Details of the algorithm can be found in the original paper [51].

The nature of evolutionary algorithms in solving MOOP, which requires several computational expensive function evaluations, provokes the need for employing metamodels. The next chapter is devoted to metamodels, metamodeling and defining the new proposed approach in constructing RBF.
2 Metamodels

2.1 Background

The increase in accessibility of computational resources and the hike in the power of the computers, enables designers to create computer models, e.g., finite element models, to simulate the behavior of a complex product. For example, in automotive industry, crash simulations and noise and vibration simulations are now a default procedure in all companies. Also, the desire to change the models to create better results and designs introduced the use of optimization methods by use of the simulations. However, the simulation models have long running times and for most of them the gradient information of the models are unknown. Therefore, the optimization process by using these computer simulations are too slow. Moreover, a single optimization run may not end to a satisfactory result. In this case, the designer might want to gain more insight into the problem by changing the design variables or find the relation between the design variables and objective or simply change the problem formulation. On the other hand, the result may be satisfactory in the first run, but the designer wants to explore better solutions by running the optimization process again. In both cases, more optimization runs, leading to more time consuming simulations are required. This difficulty is more significant in MOOP. As mentioned in chapter 1, the characteristic of EAs in solving MOOP requires numerous function evaluation and simulation runs. Thus, there is a need in creating a fast and simplified model that represents the expensive complex model with sufficient accuracy within the entire design space. The simplified model is called a surrogate model. A model that is a surrogate of a detailed simulation model is called a metamodel. It is obvious that a metamodel should be computationally affordable or at least less expensive than the actual model.

A metamodel is created by a mathematical approximation function of a detailed and usually computationally expensive simulation or experiment. In the actual model, a vector of design variables, as input to the model, result in a vector of output which contains the responses. A set of data points are created with one of the methods described in the next section, and by running the actual simulation the responses are obtained. The analytical approximation function is constructed from the collected set of inputs and their responses, which is called fitting the metamodel. Now, the metamodel can be used in optimization problems, where an efficient and accurate surrogate needs to be run several times instead of the actual simulation or experiment. In this case, thousands of function evaluations can be performed in the same time as it would take to run
2 Metamodels

only one simulation.

There are several advantages in using metamodels in place of the actual simulations. Using advanced global optimization algorithms which require more iterations and function evaluations is one of the advantages. Also, since the metamodel provides a view over the entire design space, detecting errors in the simulation model is easier. On the other hand, the main draw back of using metamodels is the new source of error that is introduced. Minimizing the error in approximating the actual simulation is very crucial in metamodeling. The error can be reduced by increasing the number of data set and their responses during the process of fitting the metamodel, however, this will increase the time of metamodeling. Also, the accuracy of metamodels could be improved by updating the metamodel during the optimization process. Selecting the proper metamodeling method could be another measure to improve the performance of metamodels.

2.2 Design of Experiments (DoE)

The set of initial data points can be created by different methods and strategies such that the metamodel represents the response adequately well over the whole design space. The process of placing the design points within the design space is called design of experiments (DoE).

Several widely used methods, known as “classic” experimental designs, such as factorial or fractional factorial [34], central composite design (CCD) [4, 34], face-centered design, Box-Behnken [34], Koshal’s DoE, D-optimal designs are discussed, studied and compared extensively in literature. The computer or simulation models are deterministic, which means repeated runs of a specific DoE will result in the same response. In this case, the space-filling sampling methods which spread the sample points all over the design space are recommended [22]. One the most well known space-filling strategies in creating DoEs is the Latin hypercube sampling (LHS) method [38, 42]. The LHS technique creates samples that are relatively uniform in each single dimension, while subsequent dimensions are randomly paired to fit an \( m \)-dimensional cube. LHS can be regarded as a constrained Monte Carlo sampling scheme developed by McKay, Conover, and Beckman [33] specifically for computer experiments.

Hammersley sequence sampling (HSS) is another popular space-filling DoE strategy. HSS produces more uniform samples over the \( m \)-dimensional space than LHS [23, 42]. It uses a low discrepancy sequence (Hammersley sequence) to uniformly place \( n \) points in an \( m \)-dimensional hypercube, described in details in [23].

Figure 2.1 shows an illustration of uniformity of a set of 50 sample points over a unit square, using a random distribution, LHS and HSS sampling technique.

2.3 Metamodeling methods

Many metamodeling methods have been developed for metamodel-based design optimization problems. The most popular and studied metamodels are: response surface methodology (RSM) or polynomial regression [2], Kriging [38], radial basis functions (RBF) [17], support vector regression (SVR) [46] and neural
Extensive surveys and reviews of different metamodeling methods and their applications by Simpson et al. [42, 45], Wang and Shan [47] and Forrester and Kean [15] can be found in literature.

In addition, several studies have been carried out on comparison of accuracy and effectiveness of different metamodeling methods [1, 14, 22, 25, 35, 43, 50]. Despite all the research, there is no unanimous agreement of dominance of one metamodeling method over the others.

It can be argued that a real engineering application may be totally different from the benchmarks, although in all comparison studies benchmarks and test problems are employed to compare the results of different techniques. Moreover, a number of parameters influence the choice of an accurate method such as non-linearity, number of variables, associated sampling technique, internal parameter setting of each method and number of objectives in optimization problems [41].

Radial basis functions is one of the methods that demonstrated promising results in most of the comparative studies. In the next section a brief definition and formulation of RBF is presented.

2.4 Radial basis functions

Radial basis functions for interpolating multivariate data were first used by Hardy [17]. He proposed RBFs as an approximation function by solving multi-quadratic equations of topography based on coordinate scattered data with interpolation.

The concept is based on the linear combination of some radial basis functions with weights that are symmetric and centered at each sampling point. The basis functions are expressed in terms of the Euclidean distance of the sampling points $x$ from a center point $c_i$, which typically is taken to be the design variable $\hat{x}_i$ at the $i_{th}$ sampling point, in the form of

$$ r = \|x - c_i\|. \quad (2.1) $$

The general form of a radial basis function without bias is
2 Metamodels

\[ f(x) = \sum_{i=1}^{n} \lambda_i \phi(r), \quad (2.2)\]

where \( n \) is the number of sampling points, \( \phi = \phi(r) \) is a radial basis function, \( \lambda_i \) is the weight for the \( i_{th} \) basis function, and \( f(x) \) is the approximation function. Some of the most commonly used radial basis functions are

- **Linear**: \( \phi(r) = r \)
- **Cubic**: \( \phi(r) = r^3 \)
- **Gaussian**: \( \phi(r) = e^{-\gamma r^2}, 0 \leq \gamma \leq 1 \) \quad (2.3)
- **Quadratic**: \( \phi(r) = \sqrt{r^2 + \gamma^2}, 0 \leq \gamma \leq 1 \)

where \( \gamma \) is a positive shape parameter.

### 2.4.1 RBF with a posteriori bias

Despite the good performance of the classic form of RBF in (2.2), expressed by \( \text{RBF}_{\text{pos}} \), in highly non-linear problems, it has been shown to perform badly for linear problems [29]. Therefore, in order to improve the performance of the classic RBF in linear responses the standard approach is to augment it with a linear polynomial:

\[ f(x) = \sum_{i=1}^{n} \lambda_i \phi(\|x - c_i\|) + \sum_{j=1}^{m} \beta_j b_j(x), \quad (2.4)\]

where \( b_j = b_j(x) \) represents the polynomial basis functions and \( \beta_j \) are the unknown regression coefficients of the polynomial biases. Replacing \( x \) and \( f(x) \) in (2.4) with the \( n \) vectors of design variables and corresponding response values leads to the following matrix format:

\[ \hat{f} = A\lambda + B\beta, \quad (2.5)\]

where \( A_{i,j} = \phi(\|\hat{x}_i - c_j\|) \), \( \lambda = [\lambda_1, \lambda_2...\lambda_n]^T, B_{i,j} = b_j(\hat{x}_i), \beta = [\beta_1, \beta_2...\beta_m]^T, \) and \( \hat{f} = [\hat{f}_1, \hat{f}_2...\hat{f}_n]^T \) are the sampling values. Since the number of unknown parameters is higher than the number of equations in (2.5), the equation is undetermined and can not be solved. This is overcome by adding the following orthogonality condition

\[ \sum_{i=1}^{n} \lambda_i b_j(c_i) = 0 \quad \text{for} \quad j = 1, 2...m. \quad (2.6)\]

Combining equations (2.5) and (2.6) will lead to the matrix form of
2.5 Metamodel validation

\[
\begin{bmatrix}
A & B \\
B' & 0
\end{bmatrix}
\begin{bmatrix}
\lambda \\
\beta
\end{bmatrix}
= \begin{bmatrix}
\hat{f} \\
0
\end{bmatrix}.
\] (2.7)

The unknown coefficients \(\lambda\) and \(\beta\) of the RBF will be obtained by solving (2.7).

2.4.2 RBF with a priori bias

In the present thesis, an approach based on a priori known bias, shown by \(RBF_{pri}\), is used to solve the RBF.

A parabolic or quadratic response surface is fitted to the sampling points as

\[
f(x) = \beta_0 + \sum_{i=1}^{n} \beta_i x_i + \sum_{i=1}^{n} \beta_{ii} x_i^2,
\]

\[
f(x) = \beta_0 + \sum_{i=1}^{n} \beta_i x_i + \sum_{i=1}^{n} \beta_{ii} x_i^2 + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \beta_{ij} x_i x_j.
\] (2.8)

If \(n\) vectors of DoEs and their corresponding responses are applied to any of the equations in (2.8), \(n\) equations will be obtained which can be expressed in the matrix form as

\[
\hat{f} = B\beta,
\] (2.9)

where the matrix of unknown regression coefficients \(\beta\) is determined by

\[
\hat{\beta} = (B^T B)^{-1} (B^T \hat{f}).
\] (2.10)

Then by knowing the regression coefficients of the bias a priori \(\beta = \hat{\beta}\), the unknown parameters in (2.5) which are the weights of RBF can be calculated by

\[
\lambda = A^{-1} (\hat{f} - B\hat{\beta}).
\] (2.11)

The proposed method eliminates any need of imposing the extra orthogonality condition in (2.6), because there is no undetermined equation.

2.5 Metamodel validation

The accuracy of a metamodel is influenced by several factors such as the type of metamodel and the quality and quantity of the DoE. Several different performance measures must be used to evaluate the accuracy of a metamodel. One of the methods measuring the accuracy of metamodels is by evaluating the residuals or the errors. The residuals are the differences between the actual function values and the predicted values by the metamodel at some sample points. The smaller the residuals or errors the better the model describes the actual function.
2 Metamodels

The accuracy evaluation of the resulting surrogate model is done by using two different error measures: 1) the standard statistical analysis, and 2) the predicted residual sum of squares (PRESS).

The most common standard statistical parameters used for validating the metamodel are root mean square error (RMSE), maximum absolute error (MAE) and the coefficient of multiple determination or \( R^2 \) value. These parameters are to some extent dependent to each other and they are defined as follows:

\[
\begin{align*}
\text{RMSE} &= \sqrt{\frac{\sum_{i=1}^{n} (\hat{f}_i - f_i)^2}{n}}, \\
\text{MAE} &= \max |\hat{f}_i - f_i|, \quad i = 1, \ldots, n, \\
R^2 &= 1 - \frac{SS_{\text{error}}}{SS_{\text{total}}},
\end{align*}
\]

(2.12)

where \( n \) is the number of test points selected to evaluate the model, \( SS_{\text{error}} \) is the sum of square errors, and \( SS_{\text{total}} \) is the total sum of squares calculated as

\[
\begin{align*}
SS_{\text{error}} &= \sum_{i=1}^{n} (\hat{f}_i - f_i)^2, \\
SS_{\text{total}} &= \sum_{i=1}^{n} (\hat{f}_i - \bar{f})^2,
\end{align*}
\]

(2.13)

where \( \hat{f}_i \) is the observed (actual) function value at the \( i \)th test point, \( f_i \) is the predicted function value at the \( i \)th design point, and \( \bar{f} \) is the mean of observed function values at the test points. The closer to 0 the RMSE and MAE are, the better the metamodel is, while the value of \( R^2 \) shall be closer to 1 for a better fit. Furthermore, when the performance of metamodeling methods is investigated across several test functions, the normalized values of the two errors \( NRMSE \) and \( NMAE \) are usually used. The normalized values are calculated by using the actual function values

\[
\begin{align*}
NRMSE &= \sqrt{\frac{\sum_{i=1}^{n} (\hat{f}_i - f_i)^2}{\sum_{i=1}^{n} (\bar{f}_i)^2}}, \\
NMAE &= \frac{\max |\hat{f}_i - f_i|}{\sqrt{\frac{1}{n}\sum_{i=1}^{n} (\hat{f}_i - \bar{f})^2}}.
\end{align*}
\]

(2.14)

The global accuracy of a metamodel is measured by RMSE, while MAE indicates the local performance. Some consideration shall be kept in mind when using the statistical error measures. A high value of \( R^2 \) might be misleading because it might be the result of over-fitting. Also the response can be insensitive to the design variables, in this case even for an accurate metamodel \( R^2 \) will be close to zero. These measures are not meaningful when used for validating the interpolating metamodeling methods, e.g., RBF, at the training data points (design points), i.e., the data points used for fitting the metamodel. In this case, off-training
2.5 Metamodel validation

(off-design) test points must be created.

For comparing the performance of a metamodeling method to other techniques, the relative difference measure in \(NRMSE\) and \(NMAE\) is introduced. For instance the comparison of \(RBF_{pri}\) to other methods can be accomplished by using the relative difference in \(NRMSE\) \((D_{NRMSE}^{i})\) of each method, given by

\[
D_{NRMSE}^{i} = \frac{NRMSE_{i} - NRMSE_{RBF_{pri}}}{NRMSE_{RBF_{pri}}} \times 100\% ,
\]

and the relative difference in \(NMAE\) \((D_{NMAE}^{i})\) is defined by

\[
D_{NMAE}^{i} = \frac{NMAE_{i} - NMAE_{RBF_{pri}}}{NMAE_{RBF_{pri}}} \times 100\% ,
\]

where \(i\) represents the different metamodeling methods; \(NRMSE\) and \(NMAE\) values of the \(i\)th method are referred by \(NRMSE_{i}\) and \(NMAE_{i}\); and \(NRMSE_{RBF_{pri}}\) and \(NMAE_{RBF_{pri}}\) are the corresponding \(NRMSE\) and \(NMAE\) values of the \(RBF_{pri}\) approach. The positive values of (2.15) and (2.16) indicate the degree of superiority of \(RBF_{pri}\) over other methods in percentage, and vice versa.

The predicted residual sum of squares or prediction error sum of squares (PRESS) is another method of measuring the accuracy of a metamodel often used in regression analysis [36].

The PRESS error is calculated by fitting the surrogate model to \(n - 1\) design points each time leaving out one design point which is used for testing the surrogate model. The PRESS statistic defined as sum of squares of \(n\) PRESS residuals is calculated by

\[
PRESS = \sum_{i=1}^{n} (e_{i})^{2} = \sum_{i=1}^{n} [\hat{f}_{i} - f_{i}]^{2} .
\]

The computation time for problems with large numbers of design point will be expensive, therefore a version of PRESS called \(k\)-fold cross validation or \(k\)-fold PRESS is used. The design points are divided into \(k\) subsets each containing \(p\) points. Here, instead of leaving out one design point at each iteration, a subset of points is withheld, and the surrogate model is trained with the other \(p - 1\) subsets of design points. If \(p = 1\), the original PRESS or leave one out error is obtained. The two most common PRESS errors are root mean square PRESS denoted as \(RMSE_{PRESS}\) and \(R^{2}_{prediction}\) calculated by

\[
RMSE_{PRESS} = \sqrt{\frac{PRESS}{n}},
\]

\[
R^{2}_{prediction} = 1 - \frac{PRESS}{SST} .
\]

Since the model recalculates the small residuals at each point, when they are not included in the surrogate model, PRESS statistics are good measures of predictability of the surrogate model. Lower \(RMSE_{PRESS}\) usually ensures the insensitivity of the model to any single point. Also, the measure is useful to overcome over fitting, created when all training points are used as centers in RBF or full quadratic terms are included.
2 Metamodels

in RSM. Cross validation errors are suitable metrics for comparison of interpolating metamodels with non-interpolating ones.
3 Summary of appended papers

3.1 Supplement I

A multi-objective optimization study by using $RBF_{pri}$ as the metamodel is performed on a disc brake system for a heavy truck. The design variables are the applied load of braking, brake pad material and the thickness of back plate. Three conflicting objective functions are considered. These are: 1) minimizing the maximum temperature of the disc brake, 2) maximizing the brake energy of the system and 3) minimizing the mass of the back plate of the brake pad. The first two objectives are obtained by a thermo-mechanical finite element analysis of the disc brake. Two metamodels are constructed for the finite element analysis, and MOO is applied on the metamodels. Resulting Pareto-fronts in objective and decision variable space are obtained. The results provide good insights to the problem. For instance, it is shown that the Pareto optimal solutions are resulted from using a brake pad with the lowest possible stiffness. A linear relation of applied braking load and brake energy was also revealed. The paper demonstrated that SPEA2 with $RBF_{pri}$ are powerful tools for performing MOO of multi-physics systems.

3.2 Supplement II

A new approach in creating RBF with the bias known a prior ($RBF_{pri}$) is presented. A polynomial bias is commonly added to the RBF to improve the performance of the metamodel in linear problems. The bias is considered to be unknown a priori in literature, and the coefficients of the RBF are obtained by imposing an orthogonality constraint. Six test functions and an engineering design problem in approximating the material properties of SGI (spherical graphite iron) are used for comparison of $RBF_{pri}$ with the classic RBF with a posteriori known bias ($RBF_{pos}$). The best of the four different radial basis functions (linear, cubic, Gaussian and quadratic) used to create the RBF, is chosen for each test problem by performing a separate comparison study. The results show robust performance of the cubic radial basis functions.

Several performance metrics including RMSE, MAE and their normalized values, and the relative difference of NRMSE and NMAE are used. They justify the good performance of $RBF_{pri}$, with a slight superiority over the classic $RBF_{pos}$.
3 Summary of appended papers

3.3 Supplement III

In order to apply multi-objective optimization on an engineering problem, this paper aims at improving the result obtained in a previous study by Salomonsson and Andersson [39], by performing multi-objective optimization of material model parameters of an adhesive layer. The errors between the simulation data and experimental data, where an adhesive layer was loaded in monotonically increasing peel and shear, are set as the two objectives. Nine model parameters (decision variables) are used from the original study. The objectives are obtained by two finite element simulations. In the original study, the two objectives were converted to one objective function and the preference based multi-objective optimization approach, discussed in section 1.3 is used to find an optimal solution. However, in this study a set of non-dominated solutions is created by using an ideal multi-objective optimization approach, that is the SPEA2. The conflicting nature of the two objectives is revealed from the obtained non-dominated solutions. A slight improvement in the selected optimal solution with regards to both objectives, compared to the original results, is retrieved by evaluating the Pareto-front.

3.4 Supplement IV

A similar comparative study as in supplement II, but more comprehensive, is performed in this paper. Multiple modeling criteria, including the type of radial basis functions, the dimensionality of the problem, sampling techniques and size of samples are considered in the study. The study is based on nine mathematical test functions with 2 to 16 variables. They are grouped into low (less or equal to 4 variables) and high dimensional functions. Four different radial basis functions, linear, cubic, Gaussian and quadratic, are used to create the metamodels. The sampling techniques employed are random (RND), Latin hypercube sampling (LHS) and Hammersley sequence sampling (HSS). The size of samples are grouped into low, medium and high. The overall result of this study also confirms the good performance of $RBF_{pri}$ in most of the test functions. The relative difference performance measure values reveal the slight superiority of $RBF_{pri}$ over $RBF_{pos}$ for most of the test functions. The study proposes the cubic radial basis function as a robust and reliable basis function, specially when the nature of the model is unknown. The HSS technique created metamodels with better accuracy. Also, by increasing the sample size the accuracy of both approaches improves. The $RBF_{pri}$ performs better than $RBF_{pos}$ with medium to high sample sizes. The out-performance of $RBF_{pri}$, is significant in the local error measure (NMAE) compared to the global accuracy (NRMSE) which is minor.
4 Future works

During this work, interesting ideas and possibilities for future works have been identified.

An investigation and thorough review of metamodeling and MOO strategies and their integration for high dimensional MDO problems with computational expensive black box functions, can be beneficial. Also a review on previous studies and implementations of MB-MOOP on products with several disciplines involved during the product developed processes, is an interesting idea for future work.

Implementing the MB-MOOP on more engineering test problems is another idea for future work. Furthermore, using MDO engineering problems for performing MB-MOOP is useful in gaining insights in details, challenges and obstacles of such problems. For instance, MOO study of an integrated design and manufacturing criteria of an industrial product can be conducted. An integrated product realization approach can be established. The approach can create a framework which combines virtual manufacturing process simulations and optimization methods to determine optimal product geometry based on manufacturing.

An extension of supplement III, is to use the MB-MOOP to find the Pareto-front and compare the results with the ones obtained by using the actual simulation. Also the MOO of the adhesive layer can be further studied by performing sensitivity analysis in order to better understand the nature of the problem and provide a broad insight to the decision makers. Further studies on the parameter setting of the optimization algorithm for a faster convergence is useful. Hopefully, a framework can be developed for such problems including curve fitting and parameter identification.

Supplement IV can be extended by implementing it on engineering test problems and considering other criteria in the comparison study of the proposed $RBF_{pri}$ approach such as, noisy behavior, non-linearity of problems and computational time. Also, it is interesting to perform a comparison study on the performance of $RBF_{pri}$ with other metamodeling methods.
Bibliography


Bibliography


Bibliography


