Collaborative Multidisciplinary Design Optimization
A Framework Applied on Aircraft Systems and Industrial Robots

Edris Safavi
To Aida
As far as the laws of mathematics refer to reality, they are not certain, and as far as they are certain, they do not refer to reality.

Albert Einstein (1879-1955)
In a product development process, it is crucial to understand and evaluate multiple and synergic aspects of systems such as performance, cost, reliability and safety. In order to improve the foundations for decision-making, this thesis presents methods that are intended to increase the engineering knowledge in the early design phases.

In complex products, different systems from a multitude of engineering disciplines have to work tightly together. Collaborative design is defined as a process where a product is designed through the collective and joint efforts of domain experts. Thus, a Collaborative Multidisciplinary Design Optimization (CMDO) process is proposed in the conceptual design phase in order to increase the likelihood of more accurate decisions being taken early on.

To enable higher fidelity based CMDO, it is necessary to validate the tools and models utilized. This can be done with so-called low cost demonstrators. The physical demonstrators increase the engineer’s confidence regarding the final product by validating the models as well as revealing many unknowns and thus further increasing the engineering knowledge.

The performance of the presented methods is demonstrated with two industrial applications, aircraft conceptual system design and industrial robot design.
Thinking is easy, acting is difficult, and to put one's thoughts into action is the most difficult thing in the world.

Johann Wolfgang Von Goethe (1749-1832)
ACKNOWLEDGEMENTS

There are many people that I would like to truly thank for their support and valuable advice during the course of the work.

Firstly I would like to express my special thanks to my supervisor Professor Johan Ölvander, Head of Division, who gave me the opportunity to work in the research group. Your commitment, motivation, enthusiasm and patience as a supervisor, have always been appreciated.

Beside of my supervisor, my sincere thanks also go to the rest of my wonderful colleagues at the Division of Machine Design. Thank you for offering me a wonderful workplace with an enjoyable environment.

Special thanks go to my friend Dr. Mehdi Tarkian for stimulating discussions, sharing his experience, and contributing to this work. Your support and care helped me overcome delays and stay focused on the work. I greatly value this friendship and I deeply appreciate your belief in me.

I am grateful to Dr. Hampus Gavel, my industrial supervisor, for the valuable discussions that have led me to better understanding and enriched my ideas. I would also like to acknowledge the support given by the industry, especially Mr. Martin Jareland and Mr. Sören Steinkellner from SAAB Aeronautics and Professor Xiaolong Feng from ABB. I would also like to thank the Swedish Defence Materiel Administration, FMV, for supporting this work.

I would like to acknowledge Mr. Varun Gopinath, for contributing to this work, the late nights in the empty corridors, and for all the fun we have had over the last 2 years.

Finally, I would like to express my gratitude to my family, Majid, Farah, Armin and Kosar for always believing in me. Last but not the least I thank my wonderful wife Aida, for always supporting me unconditionally.

Linköping, April 2013
A man should choose a friend who is better than himself. There are plenty of acquaintances in the world; but very few real friends.

Chinese proverb
APPENDED PAPERS

The following three papers are appended and will be referred to by their Roman numerals. The papers are printed in their originally published state, except for changes in formatting and correction of minor errata.


The following paper is not included in the thesis but constitutes an important part of the background.

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAD</td>
<td>Computer Aided Design</td>
</tr>
<tr>
<td>CAE</td>
<td>Computer Aided Engineering</td>
</tr>
<tr>
<td>CAVE</td>
<td>Conceptual Aircraft Vehicle Engineering</td>
</tr>
<tr>
<td>CMDO</td>
<td>Collaborative Multidisciplinary Design Optimization</td>
</tr>
<tr>
<td>EHA</td>
<td>Electro-hydrostatic Actuator</td>
</tr>
<tr>
<td>EMA</td>
<td>Electromechanical Actuator</td>
</tr>
<tr>
<td>FEM</td>
<td>Finite Elements Modeling</td>
</tr>
<tr>
<td>GA</td>
<td>Genetic Algorithm</td>
</tr>
<tr>
<td>GIU</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>HLCT</td>
<td>High Level CAD template</td>
</tr>
<tr>
<td>KBE</td>
<td>Knowledge Based Engineering</td>
</tr>
<tr>
<td>KBS</td>
<td>Knowledge Based System</td>
</tr>
<tr>
<td>MDC</td>
<td>Master Definition Component</td>
</tr>
<tr>
<td>MDF</td>
<td>Master Datum File</td>
</tr>
<tr>
<td>MDO</td>
<td>Multidisciplinary Design Optimization</td>
</tr>
<tr>
<td>MDR</td>
<td>Master Definition Reference</td>
</tr>
<tr>
<td>MDS</td>
<td>Master Definition Structure</td>
</tr>
<tr>
<td>MEA</td>
<td>More Electric Aircraft</td>
</tr>
<tr>
<td>RAPID</td>
<td>Robust Aircraft Parametric Interactive Design</td>
</tr>
<tr>
<td>SHA</td>
<td>Servo Hydraulic Actuator</td>
</tr>
<tr>
<td>VBA</td>
<td>Visual Basic for Applications</td>
</tr>
</tbody>
</table>
You can’t expect to meet the challenges of today with yesterday’s tools and expect to be in business tomorrow.

Unknown Source
# TABLE OF CONTENTS

Table of Contents

<table>
<thead>
<tr>
<th>Part I</th>
<th>Introduction ....................................................................................................................... 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHAPTER I.</td>
<td>Background ........................................................................................................ 3</td>
</tr>
<tr>
<td>CHAPTER II.</td>
<td>Research method ................................................................................................ 5</td>
</tr>
<tr>
<td>CHAPTER III.</td>
<td>Research objective .............................................................................................. 7</td>
</tr>
<tr>
<td>CHAPTER IV.</td>
<td>Thesis outline ..................................................................................................... 9</td>
</tr>
<tr>
<td>Part II</td>
<td>Frame of references ......................................................................................................... 11</td>
</tr>
<tr>
<td>CHAPTER V.</td>
<td>Modeling and simulation .............................................................................................. 13</td>
</tr>
<tr>
<td>V.1</td>
<td>Geometric modeling .............................................................................................................. 14</td>
</tr>
<tr>
<td>V.2</td>
<td>System modeling ................................................................................................................... 16</td>
</tr>
<tr>
<td>V.2.1</td>
<td>Dynamic models .................................................................................................................. 17</td>
</tr>
<tr>
<td>CHAPTER VI.</td>
<td>Collaborative design .................................................................................................... 19</td>
</tr>
<tr>
<td>CHAPTER VII.</td>
<td>Multidisciplinary design optimization (MDO) ........................................................................ 21</td>
</tr>
<tr>
<td>VII.1</td>
<td>Surrogate modeling ........................................................................................................... 22</td>
</tr>
<tr>
<td>VII.2</td>
<td>Optimization methods ......................................................................................................... 22</td>
</tr>
<tr>
<td>VII.2.1</td>
<td>Simplex .......................................................................................................................... 23</td>
</tr>
<tr>
<td>VII.2.2</td>
<td>Complex .......................................................................................................................... 23</td>
</tr>
<tr>
<td>CHAPTER VIII.</td>
<td>Rapid Prototyping ........................................................................................................ 25</td>
</tr>
<tr>
<td>Part III</td>
<td>Contribution ..................................................................................................................... 27</td>
</tr>
<tr>
<td>CHAPTER IX.</td>
<td>Conceptual aircraft system modeling and optimization ........................................................................ 29</td>
</tr>
<tr>
<td>IX.1</td>
<td>Conceptual aircraft vehicle engineering (CAVE) ........................................................................ 30</td>
</tr>
<tr>
<td>IX.1.1</td>
<td>Modeling approach .............................................................................................................. 32</td>
</tr>
<tr>
<td>IX.1.2</td>
<td>Simulation and solution sequence ................................................................................... 33</td>
</tr>
<tr>
<td>IX.1.3</td>
<td>User interface ................................................................................................................... 33</td>
</tr>
<tr>
<td>IX.1.4</td>
<td>Framework evaluation ........................................................................................................ 34</td>
</tr>
</tbody>
</table>
Part I

Introduction

Part I of the thesis introduces design methods and challenges. The research method implemented in this work is also presented together with the aims of the thesis and the major research objectives.
Far and away the best prize that life has to offer is the chance to work hard at work worth doing.
Theodore Roosevelt (1858-1919)
The conceptual design phase is one of the earlier phases of modern engineering where one or many design concepts are selected and optimized with respect to set of initial requirements (Brandt et al. 1997). Hence, one of the main goals of conceptual design study is to explore many feasible solutions and select a few of them for further analysis in later design phases (Ulrich et al., 2000).

Information gained during the conceptual phase creates a product knowledge foundation. Henceforth, gathering more information in early design stages is beneficial for better decision-making. This is especially true of complex and unconventional products with limited prior information.

Ironically, in most product development, the more knowledge gained, the less freedom is left to actually apply the knowledge, see Figure 1. This is mainly due to the design process reaching more expensive and complex stages where more people and departments are involved. Hence, fundamental design decisions in the conceptual phase are desired as these are cheaper than in later phases. However, in order to do so successfully, accurate knowledge of the product is necessary.

![Figure 1: The Information and freedom paradox in a design and manufacturing process adapted from (Jenkinson et al., 1999) & (Mavris et al. 2000).](image-url)
Fast and efficient empiric models are traditionally used in the early phases. However, the knowledge generated is limited to past products and true innovation is thus limited. It is on the other hand possible to increase the level of knowledge by applying more detailed physics-based models, which can be validated continuously through rapid prototyping, see (Halleberg, 2012) & (Amadori, 2012). In order to sustain a holistic perspective it is important to implement Multidisciplinary Optimization (MDO) which is believed to be beneficial in the conceptual design phase (Tarkian, 2009), (Amadori, 2012), (Giesing et al., 1998), (Fonseca et al., 1998) & (Lundström, 2012). It has been pointed that MDO implemented on physics-based models requires an integrated design framework. To this end geometric models are a necessity to provide the required geometric input to the various physics-based models.

Creating a highly complex and multidisciplinary framework is done in a wider organizational context where the inputs of all involved domain experts have to be extracted and stored in the models. Well-specified collaborative methods are a necessity to enable such work flows. The following domains will therefore be reviewed in Part II:

1. Modeling and Simulation
2. Collaborative design
3. Multidisciplinary Optimization
4. Rapid prototyping

Gathering more information early in design phases, i.e. the conceptual phase, is a fundamental initiative of this thesis. The models currently used at conceptual level can provide limited information about the final products. Hence, these models have to be replaced with higher fidelity models by using new methods and techniques in modeling and simulation. Developing more detailed models is difficult for conceptual engineers who have only superficial knowledge of the final product. Here is where the collaborative design methods proposed in this thesis come into the picture.

A complex product is often multidisciplinary by nature due to the variety of domains involved in designing a product. A multidisciplinary optimization framework thus needs to be developed in order to search for a set of optimal design parameters integrated in a product.

Finally, the models employed in a multidisciplinary optimization framework need to be validated and the results evaluated. This is done by using a physical prototype. The methods used to create the physical prototype rapidly are presented in this thesis as rapid prototyping.
Scientists have always tried to find the perfect classification of science. This has been a continual dilemma since 415 B.C. when Plato began to define and classify science into geometry, math, and art (Zeyl, 2000). Even though the classifications are dynamic and ever changing, it is still crucial to classify the created scientific output. This is in order to increase the level of reproducibility of the knowledge gained.

The hypotheses suggested in this study are straightforward and are of an implicit nature. For example, by using more detailed models and validating them in conceptual design it is possible to obtain more information about the final product. This results in an increase in designer confidence regarding the product, which reduces product development time and consequently the design cost. The framework to verify the first hypothesis, concerning obtaining more information about the final product, is fully developed in this research. However, proving the other hypothesis regarding reducing product development cost and time is hard without evaluating the methods directly in industry. Even if the methods are evaluated an implemented in industry, it is not possible to obtain empirical data from different teams developing the exact same products with the same initial conditions but using different methods.

Reproducibility of the knowledge gained during the research is an important attribute in process of producing knowledge (Tarkian, 2012). To progress science, it is important to verify and use the collected knowledge in other research. Hence, stating the scientific methods used to gain the knowledge in research project is essential in order to facilitate the process of reproducing the knowledge.

Scientific method is defined by the Oxford English Dictionary (Oxford online dictionary 2013) as "a method or procedure that has characterized natural science since the 17th century, consisting in systematic observation, measurement, and experiment, and the formulation, testing, and modification of hypotheses." To better formulate the scientific method used in a research the term epistemology is used.

Epistemology is the branch of philosophy concerning the nature of knowledge, its assumptions and foundations, and its extent and validity (Paul E., 1967). The applied research methods in this study can be a mixture of various scientific procedures, which can be categorized in the contradictive tendencies of epistemology, e.g. empiricism vs. rationalism
and atomism vs. holism. Knowledge gained from empiricism is based on experience via observations and is called induction or empiricism. By contrast, deduction or rationalism is the knowledge gained without observations or based on the logic. To better understand the problem under evaluation, two philosophical concepts are defined as atomism and holism. A system is broken down into smaller parts or components in atomism or reductionism. Holism is defined as the opposite of atomism, where more value is put on the system as a whole (Gunnarson, 2009).

This work is a mixture of the above paradigms. Mathematical models such CAE and CAD models are constructed based on atomistic-rationalism. The models are connected together following holistic-rationalistic approach, e.g. in a CMDO process. The framework validations and verifications (simulation and physical test) are done following atomistic-rationalistic approach, see Figure 2.

![Figure 2: The relation between the epistemological tendencies, debates and produced knowledge in this work, adapted from (Gunnarson, 2009)](image.png)
CHAPTER III. RESEARCH OBJECTIVE

The primary objective of this research is to offer methods and tools in modeling and simulation to conceptual engineers. This will provide more information in the early stages of the design process.

The second objective is to propose a multidisciplinary design optimization framework (MDO) to integrate the models and search for a good set of parameters to optimize the product time efficiently.

The third objective is to identify how collaborative methods can help to enable a design framework such as CMOD.

The fourth and final objective is to provide the means to cost efficiently evaluate and validate the framework. The resulting framework is thus proved through two case studies; conceptual aircraft design and conceptual industrial robot design.
We can’t solve problems by using the same kind of thinking we used when we created them.

Albert Einstein (1879-1955)
This thesis is divided into four parts, each consisting of several chapters.

The first part is the introduction where the background to the work, the research objectives and the research method are described.

In the second part, the frame of reference is presented, considering the involved domains in this work. The need for new methods to be applied in conceptual analysis is also addressed in this section.

The third part of the thesis presents the contributions and elaborates on the proposed methods. The methods are then utilized in two different industrial applications (conceptual aircraft system design, and conceptual industrial robot design) for thorough evaluation and validation.

The fourth and the last part conclude the thesis with discussions, conclusions and suggestions for future work.
Recognizing the need is the primary condition for design.
Charles Eames (1907–1978)
Part II

Frame of References

Part II of the thesis consists of a review of existing and relevant literature, highlighting the gaps that the present work is intended to fill. Important theoretical concepts are briefly introduced as a base for the contributions that are presented.
The whole of science is nothing more than a refinement of everyday thinking.
Albert Einstein (1879-1955)
In this chapter, modeling and simulation as a main contribution of this work is explained. As illustrated in the first objective of this work, new models of higher fidelity need to be used at the conceptual level to provide more detailed information about the design.

As presented by Raynould (1996), “a model is a representation of a system that replicates part of its form, fit, function or a mix of the three, in order to predict how the system might perform or survive under various conditions”. Therefore, a model plays an important role in measuring the outcomes when physical experiments are impossible or impractical. Models can be classified into four main categories as verbal, mathematical, physical, and schematic where verbal models use textual or oral description and mathematical models use mathematical relations to describe the system. Physical models are typically scale models recalling the system in terms of appearance. Schematic models express the system in an abstract way to better understand it, see Figure 3.

A model represents a system and thus a simulation is the execution of the model over a specific time. Simulation is about performing a physical or a virtual experiment on the model. Therefore, a behavior prediction of the product can be achieved by performing simulation on the model in various conditions. By altering the fidelity level of the simulation model, the precision of the results can be modified. Simulation models are widely used in industries because of proven benefits in comparison to physical tests, of which some are listed below:

- Less costly
- Possibility to work on an entirely new system
- Easier to change and modify
- Better access to particular parameters, which are hard to measure in physical models
- Time-efficient evaluation of certain technologies

Simulation models therefore play an essential role in the design process.
Collaborative Multidisciplinary Design Optimization

Figure 3: A 3 axis industrial robot prototype (left), and a schematic layout of the industrial robot wiring system (right)

V.1 GEOMETRIC MODELING

As discussed in previous chapters, lower fidelity models provide a lower level of information about the final product. Geometric models have great potential to be used in conceptual design to provide more information about the systems and their components.

Geometry modeling can be utilized to obtain better estimations of the various design properties. CAD tools are widely used to generate geometries, but due to limited information regarding the product in the early design stages, simplifications are applied during geometric modeling. This will make the geometry less accurate and remodeling needs to be done repeatedly in order to define accurate models.

Considering the time and cost spent to generate accurate CAD models, CAD tools are traditionally used later in the design process when more information about the final product has been gathered, (Tarkian, 2009), (Ledermann et al., 2005), see Figure 4. This issue can be solved by using parametric models (Berard et al., 2008). The term parametric modeling denotes the use of parameters to control the CAD models. Some features such as their dimension and shape can be controlled using parameters. The advantage of using parametric modeling in conceptual design thus fits the requirements of conceptual design, which is to rapidly generate and evaluate concepts.
There are many methods available for integrating the CAD tools early in the design process. In this thesis, the presented method to create geometry is the result of research projects conducted by Tarkian (2012) and Amadori (2012).

In general, geometric modifications made on a CAD model will either alter the shape of the elements (morphology) or alter the number of elements (topology). Topological parameterization is accomplished by defining templates and context manuals, see Figure 5. A top-down approach is proposed by Tarkian (2012) and Amadori (2012) as a practical way to accomplish the design automation.

Top-down design is defined by Tarkian (2012) as; “Critical information is placed on a hierarchal top level and branches down to all lower component levels in the product. Thereby the holistic representation of the product is in focus and as a result the complexity is managed and the possibility to revise the product structure and parametrically modify the morphology of the geometry is improved”. In this approach, the geometry of a product is divided into different assemblies and each assembly is defined by importing the geometries of high-level templates stored in template libraries, see Figure 6. In this work the geometry is divided in 3 main assemblies:
- **MDF** (Master datum file or Product skeleton): Contains all placement and rotational arrays that lead all the components to find their position automatically.
- **MDS** (Master definition structure or Mechanical element assembly): contains the actual geometry of the complete product.
- **MDC** (Master definition component): enables easy replacement of the components in the assembly.

The geometric templates are stored outside the geometric model and initiated parametrically using scripts.

![Diagram of MDF, MDC, and MDS template libraries](image)

*Figure 6: The relationships between the assemblies and template libraries, adapted from (Tarkian, 2009)*

**V.2 System Modeling**

A system is composed of interrelated components working together towards common objectives and purposes. The behavior of one system therefore depends on multiple levels of various sub-systems and intricate component relationships thus often result in greater system complexity.

A system typically consists of the following element levels:

- **Part level**
- **Component level**
- **Sub-System level**
- **System level**

These levels are explained in this thesis mainly by means of examples from the aircraft domain. A part is thus considered to be a small element in the system used to build the components, e.g. a resistor or inductor in an electric motor. The electric motor is defined as a component of the flight control actuation sub-system (FCAS). The FCAS works together with other aircraft sub-systems, e.g. the environmental control system (ECS) or power generation system.

A common practice to simulate systems is to utilize dynamic modeling, simulating the behavior of a system over time. The information gained is used to better evaluate various system properties.
V.2.1 Dynamic models

Dynamic models are usually presented by differential equations that present the behavior of a system over time. A variety of commercial tools exist for this purpose that are specialized to different domains, e.g. Adams for mechanical systems (MSC software, 2013) or Dymola for modeling complex and multi-domain systems (Dassault System, 2012) and MATLAB Simulink for multi-domain simulation and model-based design (Mathworks, 2012).

In this study Dymola is used to create the dynamic models and run the simulation. Dymola is based on the Modelica language. Modelica is an object-oriented language used to model complex physical systems (Modelica Association, 1999) & (Fritzson, 2004). The object-oriented language implemented in Modelica enables model integration and model evaluation through the following features:

- **Connectors**: used to manage the complexity of integrating parts, components and sub-systems. As mentioned earlier, a complex system is divided into smaller parts such as sub-systems, components and parts. Parts are modeled as separate modules and connected together using connectors in the same way as in physical systems to build the components. This means that the numerical flow between the components is resolved by the tool instead of the designer. This is a very essential feature in conceptual design when rapid evaluation of different concepts is required.

- **Hierarchical modeling**: this feature enables parallel work on the complex system by separating the system into a few components developed by domain experts.

- **Classes and instances**: It is possible to define models as classes and then reuse the models as instances in other classes, efficiently reducing the development and maintenance time of the modeling (Tarkian et al., 2008) and (Fritzson, 2004).
Good design begins with honesty, asks tough questions, comes from collaboration and from trusting your intuition.

Freeman Thomas
Collaborative design is described by Wang et al. (2002), as; “a design process where a product is designed through the collective and joint efforts of many designers”. Collaborative design is also referred as concurrent design or interdisciplinary design (Wang et al., 2002). A complex product can be considered as a single complex system with many subsystems working together. The working principles behind each subsystem involves multitude of engineering disciplines. A straightforward method to control and manage the complexity of conceptual system engineering is through Collaborative Design (Kvan, 2000) and (Peng, 1994).

Complex products generally follow a model driven approach in order to include all related design activities in a collaborative and efficient manner. In this work however, the phrase collaborative design is used to describe a more specific kind of collaboration, which is between conceptual engineers and domain experts.

Manufacturing companies are generally structured into several engineering departments, with domain experts who have specific knowledge about their area of expertise. Domain experts develop high fidelity engineered subsystems. Conceptual engineers however, are required to define the requirements and overall architecture of a future product.

Domain experts are better suited to develop new subsystems as they tend to have intuitive understanding of the nature of their systems and can better estimate parameters, which are used to predict system and component properties.

Collaboration between engineers with domain experts can reduce the cycle time in the development of complex systems (La Rocca et al., 2007). This approach brings forth many advantages, some of which are mentioned below:

- Domain experts can develop models, which are simple enough to be used in a conceptual study and still reflect the performance characteristics of an actual system.
- Verification and validation of models are important tasks. Engineers proficient in their profession are able to conduct verification and validation tasks more effectively whereas conceptual engineers may not be equipped with the resources to verify the models themselves (Steinkellner, 2011).
Domain experts have intuitive understanding of the nature of their domains and can better estimate parameters, which are used to predict system and component properties. They can also estimate technology trends which can be incorporated into the models (La Rocca, 2011).

Nowadays with a more computerized design process, collaborative design can become even more effective. This can be done by developing frameworks which allow simultaneous work on complex systems, reduce manual and sequential operations and ultimately speed up the design process. As an example Johansson et al. (2003), shows the advantages of utilizing the web service technology by using an internet-based standard (e.g. XML provided by World Wide Web Consortium (2012)), to exchange design data between the engineers. The results point to the fact that using web service technology facilitates the integration of computational tools and models regardless of computer platform and operation system and enable parallel computing to reduce simulation and optimization time.
CHAPTER VII.

MULTIDISCIPLINARY DESIGN OPTIMIZATION (MDO)

As outlined earlier, complex products usually require multiple models from multiple disciplines. Hence, it is necessary to integrate the models of a product. Furthermore, to reach global optima, a holistic perspective is necessary due to conflicting optimal solutions for the individual sub-systems. Separately optimizing sub-systems leads to sub-optimal systems. MDO is defined by Giesing et al. (1998) as “a methodology for the design of complex engineering systems and sub-systems that coherently exploits the synergism of mutually interacting phenomena”. MDO is widely recognized as a promising method to couple sub-systems and optimize the product holistically.

Connecting various models and running a stable, time-efficient optimization to find the optimal compromises is difficult, especially in conceptual design when uncertainties are considerable. This is particularly important for products where various complex systems are tightly coupled.

An MDO process can be presented in three steps, see Figure 7, (Vandenbrande et al., 2006):

Step1: Selecting proper design parameters and feeding to the system model
Step2: Simulation of all critical aspects of multiple disciplines and performing an multidisciplinary design analysis.
Step3: Finding the best concept by controlling the selection of design points.

Figure 7: A general MDO framework (adapted from Vandenbrande, 2006)
Errors in MDO processes can occur due to either modeling insufficiency or an unexpected run time error. These create an undesirable system behavior, which can drive the optimization routine to inferior solutions. One way to exclude the error-prone tools from the MDO process is to replace the original models with metamodels or surrogate models.

### VII.1 Surrogate Modeling

The real model can be a physical model or a computationally demanding model which may not be time-efficient. However, Surrogate models, or metamodels, are approximate models which are numerically efficient and can mimic the behavior of the system in a given design space (Myers et al., 2009).

A surrogate model is created by first generating samples in the design space and performing experiments or simulation of the system. The surrogate model is then fitted to the samples using different methods, e.g. Anisotropic Kriging (Martins et al., 2005).

The accuracy of the surrogate model is highly dependent on an efficient sampling and surrogate modeling method. The numbers of samples or design of experiments (DOEs) and their placement over the design space therefore have great impact on accuracy. Uniform Latin Hypercube Sampling (LHS) is used in this work to fit surrogate models (Mckay et al., 1979). This method has been used for similar purposes in other researchers works (Tarkian, 2012) and (Persson, 2012). For more information regarding relevant sampling methods, see (Wang et al., 2002), (Persson, 2012) and (Myers et al., 2009).

Anisotropic Kriging is a modified version of Kriging which calculates the new desired point as a function of distance to the known point (Martins et al., 2005). The function is resolved by analyzing the model output values varying in design space and in different directions (Pebesma et al., 1998).

### VII.2 Optimization Methods

An optimization routine is used to search for the best solution in a given design space. In fact, optimization algorithms are used to automate the iterative and time-consuming process toward finding optimal designs. Technically, numerical optimization algorithms are classified into two main groups; gradient and non-gradient. Gradient-based methods are normally used when the gradient of the function is easily accessible and calculable. The basic requirement of a gradient-based method is thus the existence of continuous first order derivatives of the objective function.

Non-gradient methods are common in engineering problems when the problems are non-differentiable, discrete, non-smooth or non-linear. However, they are more computationally expensive than gradient-based methods since they use objective function evaluation to find the optima instead of Hessian and gradient information of the objective function. In this work the Simplex and Complex algorithms are used for design optimization, see Figure 8.
Figure 8: Classification of optimization algorithms. The utilized algorithms in this work are shaded dark gray.

VII.2.1 Simplex

The Nelder-Mead simplex-reflection method, presented in 1965, is an iterative direct search method of optimization classified as a non-gradient-based algorithm (Nelder et al., 1956). As illustrated in Figure 9, Simplex works by performing function evaluation at the vertices, replacing the point with the worst function value with a point with a better value. The number of points is $n+1$ where $n$ is the number of optimization variables. The new point is obtained by reflecting the worst point ($X_3$) along the line joining the worst point with the centroid of the remaining points ($x' 2$) toward the candidate point ($x' 1$ to $x' 5$). If a better point is not found, the Simplex shrinks to the smaller triangle (indicated by the gray triangle) by retaining the best point ($X_1$) and moving the other points toward this value. This iterative process continues until the desired convergence in function value is achieved (Nocedal et al., 1999).

Figure 9: One step in the Simplex method showing current simplex ($X_1$, $X_2$, $X_3$), the candidate points ($x' 1$ to $x' 5$) and the shrunken Simplex (gray triangle), adapted from (Nocedal et al., 1999).

VII.2.2 Complex

The Complex method refers to the geometric shape with $k \geq n+1$ points in an n-dimensional space where the k-points are known as the vertices of the complex. The Complex algorithm is a constraint Simplex method and is a non-derivative method, which uses the random points to
cover the design space (Box, 1956). The optimization process is begun by evaluating the randomly generated starting points and the worst point is replaced by the new point achieved by reflecting the worst point through the centroid of remaining points by a factor of α. Evaluation of the new point dictates the direction of movement of optimization for the next iteration.

The algorithm used in this work was derived from the Complex algorithm where a randomizing and forgetting factors were introduced to make the algorithm more stochastic, thus yielding the Complex-RF method (Krus et al., 2003). Compared to the Nelder-Mead Simplex method the Complex uses more points. This makes them more capable of handling constraints and not as prone to get trapped in local minima.
Conceptual design of a complex product commences with various challenges from concept evaluation to model validation. A good explanation of this challenge can be illustrated by “the known-unknown matrix” or Rumsfeld matrix (Loch et al., 2006) as seen in Figure 10.

As presented in previous chapters, uncertainties are rather high in the conceptual phase due to the lack of knowledge about the final product’s characteristics and performance. One way to obtain more information about the product is by employing more detailed models such as dynamic models and CAD models in an MDO framework. However, the models used in MDO frameworks have to be validated in order to provide reliable results. Manufacturing and testing of a physical demonstrator or prototype of the concept under evaluation can be considered as one solution. The test results can be used later to further validate the design tool.

The matrix described in Figure 10 shows the user information sources and user awareness. While the state of the human mind is mostly focused on the upper left section of the matrix (things that are known to be known), innovative engineering problems mostly lie in the lower right section (things that are unknown to be unknown), which makes the problem problematic and difficult to solve. Many unknowns (in terms of either “user awareness” or “information
source”) are discovered that cause a large deal of time-consuming redesigning and modification activities. This is more vital in conceptual design of unconventional product where less information about the product exists. Therefore, employing the methods to help the designer to rapidly find the shortcomings in the design and validate the models is very important. This process should be rapid in order not to halt the design process. Rapid prototyping has been proven to be a promising method in the designer’s hand early in the design process, not just to get a better feeling of the final product but also to obtain more validated information about the product and models.

Rapid prototyping is a group of techniques used to quickly fabricate a scale model of a physical part or assembly. The concept of prototyping is usually presented in later stages of design. However, Hallberg (2012) proves the advantages of using a physical prototype (viz. low cost demonstrator) early in the design process.

Prototypes are classified into four main categories based on their functionality (Ullman, 1992):

**Proof of concept**: emphasis on developing function of product with respect to requirements and less attention to exact material geometry and manufacturing process.

**Proof of product**: focus on component and assembly when functionality is as important as material, geometry and manufacturing process.

**Proof of process**: using exact material and manufacturing process.

**Proof of production**: used to verify the entire production process.

In this work, physical prototypes are used mainly for proof of concept and validation and evaluation of the virtual models.
In part III, the proposed theories are utilized with the aim of achieving a holistic conceptual design framework that has evolved for use in two separate engineering applications. In the first application example, a tool for conceptual aircraft vehicular system design (CAVE) has been proposed and used in an MDO process to search for an optimal set of parameters to optimize the aircraft systems.

In the second example, a study on a modular industrial robot is performed. In this example, the benefits of using a physical prototype early in the conceptual design phase are discussed. In both examples, the MDO process is created in order to rapidly change and evaluate new concepts by establishing the complex dependencies between geometric tools (CAD) and design analysis tools (CAE) early in the conceptual phase.
Anyone who has never made a mistake has never tried anything new.

Albert Einstein (1879-1955)
An aircraft can be viewed as an integrated set of systems – complex, multidisciplinary products which are optimized to maintain safe, comfortable and stable flight. The conceptual design of aircraft vehicle system begins with the definition of requirements and proceeds to a solution at a high abstraction level, see (Wang et Al., 2002).

At the present time, most of the models used at the conceptual level are empirical and statistical based equations which predict optimal design properties such as the power or weight of the system without considering system interaction. The models are also unable to provide information regarding the size or performance of the systems or their components, e.g. volume, hydraulic pressure and voltage of actuation system. On the other hand, the models are fast, simple and easy to develop.

Geometry and performance are interlinked facets of aircraft design, which determines the optimal solution to the vehicle-system architecture. This information may significantly increase the confidence of the conceptual engineers to select one or many suitable architectures. In addition, establishing the systems layout (Figure 11) requires a pragmatic approach during the conceptual phase, as the product cost is associated with the product life cycle. Therefore, the conceptual models need to be of higher fidelity to provide more detailed information about the system and facilitate the process of decision-making. On the other hand, the detailed models that are typically used in later phases of the design are complex, slow and not straightforward to deal with. Nevertheless, detailed models are much more efficient at providing specific information about the system such as dynamic performance, system interaction and sizing properties. They are thus not appropriate for conceptual design when effortless, rapid design is an essential requirement. Furthermore, creating and dealing with higher fidelity models are hard for conceptual engineers.

A research project has been initiated at Linköping University in collaboration with SAAB AB to fill the gap in conceptual design by developing a collaborative framework and employing the models with higher fidelity than current models. The result of this research is the CAVE (Conceptual Aircraft Vehicle Engineering) tool. CAVE has been developed with the aim of more evaluating aircraft systems using dynamic models, which current models are unable to.
IX.1 CONCEPTUAL AIRCRAFT VEHICLE ENGINEERING (CAVE)

CAVE is made up of aggregations of dynamic models developed in Dymola (Dassault System, 2012) and Excel (Microsoft, 2012) which is the graphical user interface (GUI). Dymola is based on the Modelica language for dynamic simulation of complex systems and so the standard library of Modelica is used along with the proprietary hydraulic library from Modelon AB (Modelon, 2012). In fact, CAVE consists of a set of Dymola models, which represent different technologies that must be evaluated at the conceptual design phase. These models can control and run through a graphical user interface.

The system architecture is defined as a set of systems, e.g. an environmental control system and a flight control actuation system, see Figure 11. The functionality of any system is associated with one or more technologies, e.g. bootstrap or reverse-bootstrap under an environmental control system (ECS). The technologies consist of different components. They can be developed in a collaborative manner with the participation of domain experts. Hence, the systems are modeled in a bottom-up activity from system components to system architectures. However, the user defines and simulates the system architecture and related technologies in a top-down approach once they have been modeled. The actual conceptual analysis is thus a top-down activity.

Establishing the system layout of an aircraft requires a structured approach and it is therefore important to also have a structured approach to conceptual design. To structure the conceptual phase of aircraft systems, the tasks that have to be completed are:

1. **Identification of Requirements**: These requirements are input to the tool on different levels, viz. aircraft level, system level and component level (see Figure 12).
   - **Aircraft level**: The basic requirements, e.g. mission profile (e.g. altitude, speed, engine thrust, outside temperature, etc. all as a function of mission time) are defined at aircraft level. The mission profile should also contain varying input to the systems as a function of time, for example a high cooling demand for the radar during an attack phase. The architecture of the aircraft system, including number of systems and technologies and the connections, has to be defined at aircraft level.
   - **System level**: The topology of the system is defined on the system level to determine the main sub-systems and technologies and their connections. For example, in order to
simulate the ECS in an aircraft there is a requirement to connect the ECS to consumers (avionics), heat sinks (fuel tanks or the atmosphere) and energy sources (the main engine, batteries, electrical motors, etc.). It is also possible to build a hybrid system where two different technologies of a system work together; Electro Mechanical Actuators (EMA) and Electro Hydrostatic Actuators (EHA), for example, work together to drive the flight control surfaces of an aircraft.

- **Component Level**: The properties of each technology defined by the working components and their connections. Parameters that determine the basic performance of each component, such as the efficiency, weight and component-specific performance characteristics are defined at component level. Equations describing the relation between component size and performance characteristics based on either statistics or laws of physics also need to be defined at component level.

2. **Hierarchical decomposition of subsystems**: The set of sub-systems that are aggregated to form an aircraft system can be differentiated according to the sequence of simulation.

- **Independent systems** - require that the user inputs the load on the system manually to begin the simulation. These systems are however not dependent on any other systems. As an example of this study, the FCAS requires the deflection and torque on the flight control surface as an input from user; the system is then simulated in order to calculate for example the consumed power and the cooling demand, which act as input to the dependent systems.

- **Dependent systems** - might rely on user input, but depend on the output from other systems in order to begin the simulation. The ECS can be taken as an example where the cooling loads are taken from the simulation of independent and dependent systems.

3. **Definition of Interfaces** - In this work, the models are designed so that they can be simulated individually; it was therefore found necessary to use power consumed by each system as the interface between systems. However, the interfaces between the components are defined by the characteristics of the system. For instance, actuators consume power during flight and a ratio of the consumed power is given as load (W) to the cooling system. Correspondingly, if the concept under evaluation involves a servo-hydraulic actuator, then the interfaces between the components of these systems are hydraulic fluid pressure, mass flow rate, etc.
CAVE is considered to be a collaborative facilitating tool to be used in conceptual design to simulate an aircraft system dynamically. The modeling strategy in CAVE thus has to be generic and parametric to facilitate collaborative design. A modeling strategy for CAVE has been identified as:

- The main entity to be analyzed in the main system, e.g. consumed power, is defined as interface between interacting subsystems. Although the flow between the systems is in terms of, for example power (Watt), the flow inside individual system is defined by the characteristics of the system, i.e. temperature and mass flow rate of air in the bootstrap technology.

- Inverse Models – Inverse models can be interpreted such that the meaning of the input and output functions are exchanged. For example, in a flight actuator system, the torque and deflection of the actuator are given as input and the system characteristics are extracted as output. Models developed using the Modelica language is acausal in nature. This means there is no distinction between the input and the output of the system.

- The models developed for the conceptual phase can be further improved through inheritance using object-oriented features of Modelica. The models can thus evolve into a higher degree of complexity and be reused in the detailed design stages.

Figure 12: Structure of CAVE

Figure 13: Solution sequence in CAVE to calculate the power consumption (watt)
IX.1.2 Simulation and solution sequence

The order or sequence of simulation is important because the system models are developed so that they can be simulated and analyzed individually. This means the various models corresponding to system architecture do not have to be connected together in a simulation model, but instead defined in the architecture definition sheet.

The whole system simulation procedure can be described as a top-down approach where (see Figure 13):

1. The independent systems such as FCAS, avionics or the landing gear systems are simulated first in order to calculate the power consumed over the flight mission.

2. A ratio of the consumed power by the independent systems is given as heat load to the cooling system.

3. The sum of the power consumed by all systems (dependent and independent) is given as input to the power generation system.

IX.1.3 User interface

The conceptual engineers’ task is to analyze the abstract requirements of the product and bring an engineering perspective or focus to the design task and so they are not required to be domain experts. The interface allows a focus on model parameter evaluation rather than model development so as to fit the initial requirements.

On the other hand, a collaborative tool should be able to bring together all the actors into a single workspace, which in turn increases the effectiveness of the collaboration. The tool should also be able to manage the many complexities that arise from collaboration. Wang et al. (2002) detailed some situations that can arise, along with a list of tools that provide solutions to the problems. In this project, the collaborative tool chosen was Microsoft Excel because:

- Excel is a widely used tool and most engineers are familiar with it.
- Excel enables collaboration by allowing simultaneous editing of documents when saved in a networked resource.
- Excel can communicate with other engineering software tools (like Dymola) using Windows COM (Component Object Model) objects which are well documented.
- Specification of requirements can be easily represented using tools like DSM (Design Structure Matrix (Sigmazone, 2011)) in Excel.

The interface should allow easy modification of the parameters in the dynamic model and also simulate the model. To evaluate the system architecture, it is necessary to complete the following tasks in CAVE’s GUI, see Figure 14:
1. Set the overall system architecture including the number of actuators, choice of cooling technology etc. in the architecture definition sheet.

2. Process a time-dependent flight profile including altitude (m), speed (M), range (km) and etc. provided by the user to simulate an actual flight mission.

3. Define every system technology in separate sheets. The parameters associated with systems are defined explicitly in the corresponding system definition sheet.

4. Process the results of the simulation to calculate the power consumed by the system/aircraft and also predict the mass and volume of the system/aircraft.

Since the conceptual engineers are supposed to deal with only CAVE’s user interface, simulation of the systems is considered to run in the background. The GUI thus has to communicate with the simulation tool to run the simulation and extract the results. The GUI is therefore programmed to create Dymola script files, which change the necessary parameters in the simulation models, run the simulation and create and store the results. The results are stored as text files, which are then read back into Excel for further processing, e.g. calculate the overall dimensions and mass of the systems.

**IX.1.4 Framework evaluation**

During the conceptual phase many flight system architectures are to be evaluated. Even in a small aircraft, there may be a large number of possible configurations that satisfy the overall requirements. In this section, potential aircraft system architecture with the following systems will be evaluated using CAVE (See Figure 11). The aircraft systems modeled are classified as

---

**Figure 14: Multiple views of CAVE’s user interface**
independent and dependent systems. In this study the only independent system is FCAS with three different associated technologies: Electro-mechanical actuator (EMA), Electro-hydrostatic actuator (EHA) and Servo-hydraulic actuator (SHA). These technologies can be used on a conventional aircraft or an MEA (More Electric Aircraft). Airplanes classified as MEA generally use a mixture of these technologies. CAVE can be used to evaluate various types of technologies.

The dependent systems receive input from the independent system. The depended system in this study consists of an environmental control system and an electric power generation system. The three main technologies in the cooling system are Bootstrap System, Reverse Bootstrap System and Vapour Cycle System. The ECS of an aircraft is in charge of providing the proper working environment for the crew and passengers as well as other systems, such as the avionics system, the FCAS system, etc. In principle, in most conventional aircraft bleed air is used to pressurize the cabin and cool the other systems. Bootstrap (BS) and reverse bootstrap (RBS) systems are two examples of such bleed-air systems, see Figure 15.

On the other hand, bleedless technologies e.g. vapour cycle system (which acts like home air conditioners), have recently proposed for use together with bleed-air systems in unconventional aircraft, e.g. MEAs. Vapour cycle systems are created based on evaporation theory. In this system, a circulating liquid refrigerant is used to absorb heat from heat sources. The main components of this system are an electrically driven compressor, evaporator, and condenser and an expansion valve.

To energize the above-mentioned vehicle systems, an electric power generation system has been designed. In this model the power required to run all the systems is calculated using an inverse modeling approach. The power required is thus input to the system and the power extracted from the engines is calculated as output.

**IX.1.4.1 Design study**

A design study is made to more thoroughly evaluate CAVE. This study can be used to prove the concept of multidisciplinary design and collaborative design in conceptual level, which are always on demand in aircraft industries and a difficult task to implement.
The entire architecture of the aircraft vehicle system in this study consists of:

- Seven Flight Control Actuation Systems (FCAS):
  - Two EMAs to actuate the flap
  - Two EMAs to actuate the aileron
  - Two EHAs to actuate the elevator
  - One EHA to actuate the rudder
- A hybrid cooling system with a bootstrap system and a vapour cycle system with each is assumed to provide 50% of the cooling
- One "variable speed constant frequency" power generation system.

To run the system, a sample flight profile including speed, altitude, flap deflection, etc. has to be input to CAVE. This information can be used to predict the actuator deflection and also hinge moment on the aircraft flight control surfaces. The other inputs to the system are the design parameters associated with the chosen technology, e.g. motor voltage, hydraulic pressure, and the efficiencies of various components, which affect the performance of the systems.

On the one hand, the preliminary results show that by using the methods proposed in CAVE, the flexibility of the conceptual engineers to derive the empirical data can increase considerably. This can be achieved by evaluating and validating the models based on empirical data, e.g. the component data sheets. On the other hand, the quantity and quality of the information gathered during the conceptual design phase can be changed by using dynamic performance information about the systems. For example, Figure 16 shows the performance of the cooling system with respect to the cooling that has to be generated. It can clearly be seen that none of the reserved cooling technologies are able to provide the required cooling power. It is also clear that the bootstrap system can reach the necessary capacity only after a certain time. However, this can be improved by changing the design parameters, e.g. increase the ram inlet area or the heat exchangers’ parameters. Correspondingly, the maximum power generated by the vapor cycle system is too far from the required power. Hence, using a vapor cycle system seems to be infeasible even with optimized design parameters. It must be noted that the models and values used in this study illustrate a practical scenario and do not reflect real world performance, even though efforts have been made to target practical results.

![Figure 16: Cooling produced by CAVE’s bootstrap and vapor cycle systems](image)
IX.2 CAVE IN AN MDO PROCESS

It is always crucial to find the optimal values of the design parameters to have a better evaluation of the behavior of a system and make the best decision. Therefore, it is very important to evaluate the capability of CAVE by using it in an MDO process to search for a good set of design parameters with respect to the constraints applied to each system involved. Hence, the concept of collaborative multidisciplinary design optimization (CMDO) can be evaluated further. In this framework CAVE is used together with geometric and aerodynamic models which are created by domain experts separately. Hence, robust interfaces have to be constructed to provide an automatic interaction between these models.

The automated design and evaluation framework was implemented using modeFRONTIER (modeFrontier 2012), which allows various design tools to be integrated to create the metamodels and run the optimization. The framework consists of a geometric model (RAPID) (Staack et al., 2012), a simple standard aerodynamic model created in TORNADO as aerodynamic simulator software (TORNADO, 2013) and a dynamic model (CAVE), see Figure 17. The automated geometric model provides the analysis tool with geometric input. The aerodynamic model requires the size of the aircraft to calculate the aerodynamic forces as well as drag and lift coefficient ($C_d$, $C_l$). The dynamic simulation model of EMA needs information from the aerodynamic model, e.g. forces, to predict the mass and overall dimensions as well as the estimated power consumption of the actuator over a predefined flight profile.

![Figure 17: Multidisciplinary Design and Optimization of FCAS design](image)

In this work modeFRONTIER was also used to create surrogate models of all simulation models. As discussed in part II, surrogate models are used to further accelerate the optimization process, -which is essential in conceptual design phase.

The Anisotropic Kriging method (Martins et al., 2005) was used to create the surrogate model with 300 Uniform Latin Hypercube (ULH) samples. This high number of samples has been
chosen to increase the accuracy of the surrogate modeling. To evaluate the model, 50 random samples are generated and used to calculate the error between the original and the surrogate model. The error is calculated using Normalized Root Mean Square Error (NRMSE). A small amount of calculated error (0.05%) for the geometric and the aerodynamic model, as well as 1.1% for the dynamic model, show satisfactory results from the surrogate modeling.

**IX.2.1 Optimization formulation**

In the problem formulation the objectives function consists of minimizing the consumed power ($P$) by the system, minimizing the weight of the actuator ($W$), reducing the weight of the flap ($W_f$) and the position of the actuator with respect to the fuselage ($A_p$) due to less force being required to rotate the flap.

The explained objectives are combined to create the objective function of the optimization ($Z$). Two constraints are defined to ensure that the size of the actuator is always smaller than the corresponding size of its position on the wing. Hence, in the two constraints, geometry ($g(x)$, $h(x)$) and the volume ($A_v$) and width of the actuator ($A_w$) should be designed smaller than or equal to the volume of the actuator housing ($A_{hv}$) and the width of the actuator housing ($A_{hw}$) in the aircraft wing, see Eq. (1). The other geometric constraints such as actuator length and height are negligible due to having more space in the wing in the stated directions.

The behavior of the actuator can be controlled by the design parameters, such as number of poles in the electric motor ($N$), current ($I$) [Amps], voltage [$V$], and gear ratio of the gear box ($G_r$). These also affect the total power and mass of the actuator. Actuator width [mm], actuator position [mm], and stroke length ($S$) [mm] are other design parameters given as input to the geometric model. The optimization problem can thus be formulated as illustrated in Eq. (1), where $\lambda_i$ and $\mu$ are constants that normalize the objective and penalty functions respectively, see (Krus et al., 2003).

\[
\begin{align*}
\text{Min}(Z) &= \lambda_1 P(x) + \lambda_2 W(x) + \lambda_3 W_f(x) - \lambda_4 A_p(x) + \sum \mu P \\
\text{subject to :} \\
g(x) : A_v - A_{hv} &\leq 0 \\
h(x) : A_w - A_{hw} &\leq 0 \\
x &= [N_r, V, I, G_r, A_p, A_w, S] \\
x_{\text{low}} &\leq x \leq x_{\text{up}}
\end{align*}
\] (1)

The constraints are added to the objective function using a penalty function according to Eq. (2):

\[
\begin{align*}
P_1 &= \max (0; g(x))^2 \\
P_2 &= \max (0; h(x))^2
\end{align*}
\] (2)

**IX.2.2 Result**

The Simplex algorithm is used to optimize the objective function with respect to the constraints. The continuous nature of the problem ensures high optimization speed using Simplex, which is important during the conceptual design phase. On average, a Simplex method needs around 156 iterations to converge to the optima. Optimization time for Simplex is around 10 minutes on an 8-core 3.3 GHz computer.
The convergence in objective function and two of the design parameters, gear ratio and motor current, are shown in Figure 18. Although the results show that the convergence occurs in iteration 156, there is a slight change in objective function from iteration 80 to 156 that can be eliminated by increasing the termination criteria, which will shorten the optimization time. The optimal point is thus:

\[ x = [2, 36, 126, 8, 983, 435, 600] \]

This study presents the potential of using CAVE in an MDO process. The results of the optimization also show a satisfactory optimization speed in a conceptual study even by using more detailed models.

Figure 18: Convergence in objective function (right) and convergence in gear ratio (top left) and current (bottom left)
A mind that is stretched by a new experience can never go back to its old dimensions.

Oliver Wendell Holmes, Jr. (1841–1935)
In order to create a multidisciplinary design framework, formally articulated and documented knowledge, in other words explicit knowledge is required (McInerney 2002). Another issue is design uncertainties in view of unconventional design. As described earlier, the test and evaluation of new concepts using physical prototypes can decrease uncertainty in design. Test how the physical prototype performs compared to the virtual model and thereby formulate new knowledge in order to improve the virtual model. The process of realizing a physical prototype should be fast in order not to halt the design process and simultaneously increase the explicit knowledge of the concept. This is especially true when the product is unconventional or complex, encompassing multiple disciplines, e.g. a modular industrial robot. Using a multidisciplinary design framework is therefore more vital for analyzing unconventional concepts.

Figure 19: The multidisciplinary design framework, including a physical prototype

As a second application in this thesis, an industrial robot is analyzed within an automated design framework with the help of a physical prototype. As shown in Figure 19, in the design
Collaborative Multidisciplinary Design Optimization

framework proposed by Tarkian (2012), the process begins with geometric modeling. The mass properties of the robot are extracted from a geometric model and used in a dynamic model to simulate the robot’s movements virtually and calculate the torque required for each of the robot’s joints. This helps the designer select the appropriate drivetrain for each joint. At the end, a downscaled physical prototype is created to evaluate and validate the virtual models and increase the explicit knowledge of the concept. However, the process of designing and manufacturing a downscaled prototype will not fully represent the real process of the full-scale product, but it can help evaluate and validate the design process and increase the explicit knowledge of the final product.

In this study, the different models in the framework are described in detail. The objective is to show that the design framework for modular industrial robots can be applied to design and set up concepts rapidly.

X.1 Dynamic Model

The dynamic model used in this work consists of three parts, a trajectory planner, models of the drive train components and a rigid body model. The trajectory planner computes the trajectory in joint spaces. The trajectory is utilized in the dynamic model to calculate the torque and force. These are the required driving force and torque in the rigid body model to create the motion. For more information concerning this project and the results obtained from dynamic model, see (Tarkian, 2009).

The drive train of the full-scale model of an industrial robot consists of complex components such as Harmonic drive, precise AC motor, and sensors used in complex closed loop control systems. The drivetrain components mentioned are replaced with less complex components in order to facilitate and accelerate the realization process at lower cost and with less technical effort, which is not the intention of this study. For example, AC motors with complex control and feedback systems are replaced by stepper motors with forward control system. The harmonic drive is also replaced with a planetary gear to increase the output torque of the motor at less cost. The holistic schematic of the drive train system illustrated in Figure 20 thus consists of power supply, motor driver, motor, and gearbox. The dynamic behavior of the drive train components is modeled using Dymola, where the Dymola standard library is used to create the dynamic model, see Figure 20. These models are stored in the drive train model of the dynamic model.

![Figure 20 Dymola model of the drive train (top) and actual drive train component (bottom) of the physical prototype](image-url)
X.2 GEOMETRIC MODEL

As explained in part II, the top-down constructed geometry model and the modifiable topology of the robot assembly enables any component to be changed easily. This is done by adding new links and drive train templates in the template library, see Figure 21. The result is new MDS and MDC templates in the template library. This is used to assemble new types of modular robots from the user interface, see Figure 22.

![Figure 21: The type of drive train was modified during the design phase of the concept realization robot which resulted in new templates in both the MDS and MDC libraries](image)

By specifying the drivetrain parameters in the dynamic model, see Figure 20, the dynamic properties of the drivetrain are further simulated, allowing suitable geometries for the links and selection of drivetrain to be processed faster. When the final design has been set, a physical prototype can be built.

X.3 PHYSICAL PROTOTYPE

The physical prototype is manufactured by a 3D prototyping machine (Dimension Elite from Cimquest Co. (2012)). 3D printers are suitable for this purpose since complex geometries are produced time-efficiently. The dynamic and geometric model can be evaluated using various test cycles performed on the prototype and evaluate the assembly of physical prototype respectively. The dynamic model can be further modified in order to calibrate the reference variables of the robot controller. If the physical prototype does not fulfill the defined requirements, a new concept process can be initiated. This iterative process continues until the prototype’s performance has been satisfactorily evaluated or the concept scrapped. In order to
evaluate the predictability of the framework, a trajectory performed by the prototype is compared to one simulated by the dynamic model, see Figure 23.

![Diagram](image1.png)

**Figure 23:** The process of concept realization begins with geometry definition, evaluated in a dynamic model. A controller is then automatically defined and finally a prototype manufactured.

Two separate test motions have been defined. The first test consists of 15 line segments of which 6 are on a vertical board surface (to write LIU). The second test consists of 6 line segments of which 4 are on a vertical board surface (to draw a rectangle), see Figure 24. By being able to conduct these trajectories it can be concluded that the set-up of the trajectory planner is correct and that the procedure from the dynamic model to the prototype robot controller is also properly integrated. It can also be assumed that the assembled geometries in the CAD model have kinematic validity, although if the prototype did not have the correct kinematic structure, then the performed trajectories would turn out to be very different to the computer simulations.

![Image](image2.png)

**Figure 24:** Comparison between the trajectories performed by the dynamic model and the physical prototype

A closer look at the speed of one of the axes of robot during the performed cycle shows the similarity between the physical prototype and the virtual dynamic model created in Dymola, as shown in Figure 25.
X.4 MULTIDISCIPLINARY OPTIMIZATION OF A MODULAR INDUSTRIAL ROBOT

In the second example, a new concept of seven DOF modular industrial robots is taken into consideration to further validate the framework. This project was conducted in 2010 by a group of students as a course project under the supervision of the author. The results of this project published as a scientific paper, see (Nezhadali et al. 2011).

To more industrialize the prototype, the drivetrain components used in first example are replaced with more precise components, i.e. brushless DC motors and precision planetary gears. The control system is also replaced with a more industrialized control system from National Instrument using Labview as control software (National Instrument, 2012). This helps to evaluate the concept and enables better comparison to a full-scale robot. The multidisciplinary framework is also used to design the second concept. The design parameters involved in the second concept are optimized using genetic algorithms (GA). The same methods as explained earlier have been applied in this study as well, i.e. a top-down approach when building the geometric model and the same type of dynamic models.

All the models are integrated in an MDO framework and are communicating with each other using Microsoft Excel as the user interface. The design is then optimized to find the minimum weight of the structure by selecting the lightest actuators from a library of actuators with respect to the required torque and actuator length constraints. The optimization can thus be formulated as illustrated in Eq. (1):

\[
\min Z = M \left( x_i \right) + \sum \mu P \\
\text{subject to :} \\
g1: L_i \left( x_i \right) - L_a \left( x_i \right) \leq 40 \text{ mm} \\
g2: T_r \left( x_i \right) - T_a \left( x_i \right) < 0 \\
x_{\text{low}} \leq x_i \leq x_{\text{up}}
\]

Again, the constraints are added to the objective function using a penalty function with a large constant value \( \mu \) according to Eq. (2):

\[
P_1 = \max \left( 0; g1(x) \right)^2 \\
P_2 = \max \left( 0; g2(x) \right)^2
\]
The results of the optimization were satisfactory and somehow predictable since the bigger actuator was selected for the second joint where a large amount of torque is required. The results of the optimization prove the narrow-down approach used to design the robot by selecting smaller motors for smaller links. The optimization process took almost 12 hours on a 3.2 GHz processors computer with 2 GB of ram. The long optimization time confirms the advantages of using metamodel in MDO process.

The optimized design is manufactured using 3D printer and assembled to create the robot, see Figure 26. Maximum 10% error is calculated by comparing the weight of links and actuators as an important factor to design industrial robots to CAD models.

![Figure 26 Prototype (left) vs. CAD model (right)](image)

The initial results from the test of the physical prototype further confirm the advantages of the proposed automated design framework. This can also be used to further validate the virtual models in the framework. The comparison between the results of the joint angle (position) in the virtual model and the physical prototype for one actuator is shown in Figure 27 to further validate the results.

![Figure 27: Trajectory tracking performance of an actuator](image)
Part IV concludes the thesis. As a final discussion, the topics presented in the contribution part are further discussed. The answers to the research questions presented in part I are given as a conclusion of the thesis. Directions for future research are also described.
Never is a waste of time if you use the experience wisely.
August Rodin (1840-1917)
This thesis proposes methods to gain more design knowledge early in the conceptual design phase. One way to gain more knowledge is by utilizing a model-based approach such as bringing more detailed models into the conceptual analysis, e.g. CAD models or dynamic models. However, creating higher fidelity models and also managing these models are crucial for conceptual engineers who lack deeper knowledge regarding the entire system and the underlying components. On the other hand, using a model-based approach involves many facets of the design process like requirement analysis, requirement specification, complexity management, model evolution management, model verification, model validation, etc.

Highly flexible collaborative design frameworks are thus key enablers to employ more detail models in conceptual design and manage the mentioned complexities. The proposed collaborative design framework should facilitate the process of dealing with high fidelity models created for conceptual engineers by domain experts. The concept of collaborative design is not a novel topic in product development. However, it has recently come to be considered a paradigm shift in product design methodologies and systems. The development of collaborative design frameworks is thus a current R&D trend (Li, 2007).

All complex multidisciplinary products, e.g. aircraft, require various disciplines to be concurrently evaluated so as to achieve an optimal solution. Multi-Disciplinary Optimization (MDO) allows the engineer to explore the design space and map interdisciplinary relations that exist in a system to be used. However, as mentioned before, dealing with different disciplines in conceptual phase requires a collaborative approach. Hence, a collaborative multidisciplinary optimization (CMDO) has been introduced to optimize the aircraft systems as a complex multidisciplinary product.

In this work, the CAVE framework enables collaborative design in the conceptual design. The proposed framework has been used within an optimization process in order to optimize the design parameters. Since variation in design is high in conceptual design, a high number of iterations is required in order to obtain optimized designs. Optimization time thus increases significantly. Surrogate modeling, as proposed in this work, has proven to be a profound method to reduce the time of optimization. Another benefit of replacing actual model with
Collaborative Multidisciplinary Design Optimization

surrogate models is to reduce the number of occupied software’s licenses of engineering tools integrated in optimization routine while it is running.

Nowadays, complex products are considered to be sets of integrated systems or systems of systems. They therefore have to be developed holistically. This is illustrated by the significant uncertainty, especially in the conceptual design phase. The uncertainties in the conceptual design phase are rather high. By using higher fidelity models, more information about the product and its systems is revealed.

The detailed models used in conceptual design have to be validated and verified. In this thesis, a method to produce downscaled prototypes from the concept under evaluation is presented. This prototype can be used to test the concept under different conditions and collect the data about the design and modify the design early in the conceptual phase where changes in design are less expensive than other phases. In the thesis the two modular robot examples have illustrated this issue.

It is demonstrated that by producing downscaled prototypes of the concept under evaluation, explicit knowledge about the product can increase, thereby boosting confidence in the concept. A variety of methods are also presented for how to decrease the time needed to generate the prototypes, thereby achieving rapid concept realization.

**XI.1 FUTURE WORK**

The future work of this study can be directed in three areas:

1. **Application in industrial projects**

Although the tool and methods proposed in this thesis have been developed in close collaboration with industry, further evaluation and validation is required in industrial development processes. So, for future work, the methods and tools can be further evaluated by testing in different industrial applications.

2. **Collaborative design**

The concept of collaborative design has been under discussion for many years but there are few references regarding the use of collaborative design in conceptual design phases and MDO settings. Collaborative design seems to be essential in conceptual design phases since a conceptual engineer needs to work with different models in different fields. However, there are few tools in the market that support collaborative design, but they are not well-designed for the conceptual phase. Future work can thus be directed in three disciplines,

- Methodology in terms of how collaborative design can be useful in conceptual design
- Implementation, how and where collaborative design can be implemented
- Generic tools for collaborative design, especially for conceptual phases and in cooperation with multi-disciplinary optimization.
A short review of the appended papers is provided here to explain the contribution of each paper.

**PAPER [I]**
**A COLLABORATIVE TOOL FOR CONCEPTUAL AIRCRAFT SYSTEMS DESIGN**

In this paper, the necessity to have more detailed models, e.g. dynamic models, in conceptual design as a main part of design is explained. The collaborative design method is presented as an effective method to integrate the detailed models into the design process effortlessly. The mentioned phrases are more specific to aircraft conceptual system design. In this paper, CAVE is presented as a collaborative tool in aircraft conceptual system design in order to further support collaborative design in the conceptual phase using more detailed models. A design study is defined to evaluate CAVE in the conceptual design of aircraft system architecture. In this study, 3 aircraft systems, i.e. environmental control system, flight control actuation system and power generation system, are presented and evaluated using a predefined flight profile.

**PAPER [II]**
**MULTIDISCIPLINARY OPTIMIZATION OF AIRCRAFT VEHICLE SYSTEM FOR CONCEPTUAL ANALYSIS**

In Paper II, the optimization is defined as an important tool to optimize the new concepts early in the design process. Hence, CAVE is used in an MDO process to search for a good set of design parameters to optimize the design. In fact, the intention of this paper is to prove the advantages of bringing optimization into early design phases and show the capability of CAVE to work in an MDO process. The MDO process consists of CAVE, a geometric model and an aerodynamic model, all in the same optimization loop. A case study is also performed to evaluate the framework and validate the models.

**PAPER [III]**
**RAPID CONCEPT REALIZATION FOR CONCEPTUAL DESIGN OF MODULAR INDUSTRIAL ROBOTS**

In this paper, testing the concept by means of a prototype is presented as one way to obtain more information early in the design process. This prototype is used to evaluate and validate the models created in the design process. Manufacturing of the prototype should be fast in order not to halt the design process. Therefore, a proof of concept prototype called rapid prototype is manufactured in order to evaluate the design framework and validate the models using preliminary tests.
It is the mark of an educated mind to be able to entertain a thought without accepting it.

Aristotle (384 BC – 322 BC)


Box M.J., “A new method of constrained optimization and comparison with other methods”, Computer Journal, 8, pp. 42-52. 1956


References


