State of the Art of Artificial Intelligence (AI) to Support Design Space Exploration (DSE) in Aerospace Engineering

DEFAINE Project Report D5.1 and D5.4

Author, company:

- Ingo Staack, Linköping University
- Siva Dasari, GKN Aerospace Sweden
- Petter Andersson, GKN Aerospace Sweden
- Petter Krus, Linköping University
- Max Baan, ParaPy
- Bastiaan Beijer, KE-chain
- Massimo Panarotto, Chalmers University
- Tobie van den Berg, GKN Fokker Aerostructures
- Robin Augustinus, GKN Fokker Elmo
- Kristian Amadori, Saab AB
- Gianfranco la Rocca, TU Delft

Version:
2.2

Date:
April 11, 2023

Status:
Final / Released

Confidentiality:
Public

Change log

Revision	Date	Prepared by	Checked by	Description
1.0	06/04/2021	Ingo Staack	WP5 Partners	First version
1.1	22/09/2022	Ingo Staack	Several Partners	Upgraded first version
2.0	17/02/2023	Siva Dasari	Ingo Staack, Petter Andersson	Reworked Table of Content Added Section 2 Removed several parts of Ver. 1.1 and relocated parts of Ver 1.1 to the Appendix A-C
2.1	31/03/2023	Ingo Staack	Several Partners	Export from Latex; Shared to the whole Consortium for review
2.2	11/04/2023	Ingo Staack	Several Partners	Final Version

Acknowledgements

This research is funded by the ITEA3 call 8 project DEFAINE of the European Union, Grant No. 13040.

Table of Contents

1. DEFAINE Project Introduction and Background

1.1. DEFAINE Project Introduction

The Design Exploration Framework based on AI for froNt-loaded Engineering (DEFAINE) project focuses on accelerating the entry into service of novel solutions in order to explore new product development approaches that can drastically reduce lead time. The early design phase of aircraft components and their performance estimation is a complex multidisciplinary problem, and it involves analysing the effects of aero performance, mechanical functions as well as manufacturability. For this, simulations play a vital role to help to have a better understanding of the functional behaviour and to predict possible failure modes in design concepts. However, high-fidelity simulations can take significant time to compute. Typically, a large system has millions, if not billions of possibilities to explore in the design space [16]. Thus, it is impractical to explore the design space by conducting simulations for all possible design concepts due to the time constraints. Hence, to minimize the number of simulations which are needed to investigate a certain design space, different approaches are used. This document focuses on reviewing different AI approaches to explore large design spaces while at the same time reduce the computationally expensive simulations for such design space exploration.

1.2. Aim and Scope

The focus is on AI methods for design space exploration primarily by means of surrogate modelling, to enable front loaded engineering in Aerospace engineering. This document describes the state-of-the-practice of industrial partners in DEFAINE and the state-of-the-art from literature and provide suggestions on applying AI methods and tools.

1.3. Background

1.3.1. Design Exploration Framework

Design space exploration (DSE) refers to the activity of exploration and investigating design alternatives prior to system implementation. This is used for rapid prototyping, optimization and system integration [16]. In rapid prototyping, DSE helps to generate several prototypes before the system implementation. By simulating these prototypes, engineers can increase the understanding of the impact of design decisions. In optimization, DSE can be used for optimization by eliminating the lower quality designs and selecting a set of design candidates for further analysis. The elimination is done by comparing one design to another using predefined metrics, for instance, design requirements. In system integration, DSE can be used to find legal assemblies and configurations that satisfy all global design constraints for the integration of multiple components into a working whole system.

The exploration of design space increases the engineer’s understanding of the design problem [36]. The exploration must be done carefully due to a large number of design alternatives. A large system may have millions, if not billions of design alternatives, and it may have infinite alternatives for some design problems [16]. In addition, a larger complex system also has a larger number of design constraints that must be satisfied by every valid design alternative or solution.

Furthermore, the analysis of these design alternatives includes higher computational costs. This is where surrogate modelling can play an important role in exploring many design alternatives without the need of time computational simulations for analysis.

1.3.2. Surrogate modelling

In surrogate modelling, the aim is to determine a continuous function $\hat{f}$ (model) of a set of design variables $\mathbf{x} = x_1, x_2, \dots, x_n$ from a limited amount of available data D (shown in Fig. 1.1). The available data D represents the exact evaluations of the function f, and in general cannot carry sufficient information to uniquely identify f. Thus, surrogate modelling deals with two problems which are constructing a model $\hat{f}$ from the available data D, and evaluating the error ε of the model [24]. The prediction of the simulation-based model output using surrogate modelling approach is formulated as follows:

$$f(x) = \hat{f}(x) + \varepsilon(x)$$

Where $\hat{f}(\mathbf{x})$ is the predicted output and $\varepsilon(\mathbf{x})$ is the error in the prediction. The construction of response surface model involves several steps (shown in Fig. 1.1) [27]:

(i) Design of experiments: it is the sampling plan in design variable space for setting of all possible combinations of the design variables ($\mathbf{x} = (x_1, x_2, \dots, x_d)^T$; T means transpose of vector)

(ii) Numerical simulations: Let f be the black-box function (simulations), evaluate f on design points $y_i = f(\mathbf{x}_i)$ where $x_i \in R^d$ and $y_i \in R$

(iii) Construction of surrogate model: Consider the data $D = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$, given the data, a continuous function $\hat{f}$ is determined to evaluate new design point $\hat{y} = f(\hat{x}_i)$

(iv) Model validation: Assessing the predictive performance of $\hat{f}$ from the available data D.

This description of surrogate modelling is also presented in [8].

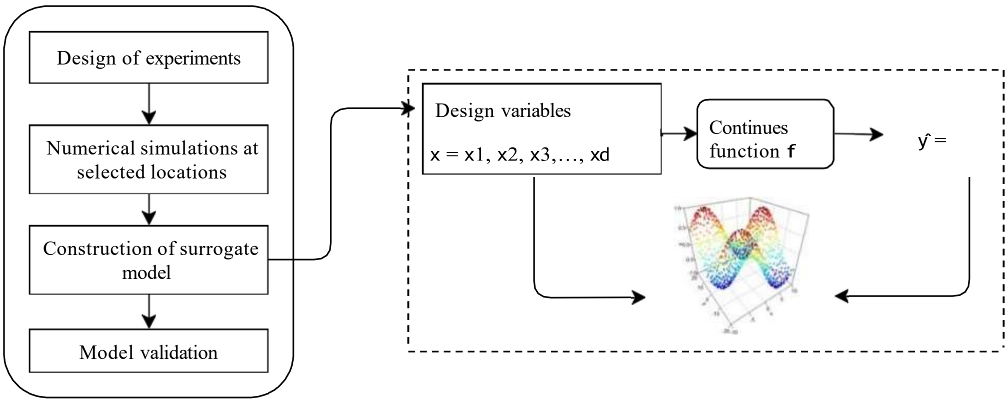


Figure 1.1: Surrogate Modelling.

1.3.3. Sensitivity Analysis

Sensitivity analysis is to study how the output behaviours relate to variations in the inputs [29]. This sensitivity analysis allows engineers to get a better understanding of design parameters. For instance, it helps for screening, parameters prioritization, and parameters interactions [27]. Furthermore, it could help to identify interesting regions in design space which can be explored further.

1.3.4. Front-loaded Engineering

The front-loading approach described in the project management literature often stress the need to increase the resources involved in early phases where the design is still open and it is still possible to exploring different options in contrast to making design changes later in the development change that is very costly and often prolong a projects time schedule [35]. Another approach to front loading of product development projects is the Knowledge Based Engineering methodology (MOKA), where the engineering knowledge is systematically captured pragmatically to enable re-use [42]. Here, methods and tools are developed and continuously improved by the experience from previous projects.

The development of a framework for Multidisciplinary Design Exploration including methods and tools for automation of engineering activities will enable a systematic continues improvement. The use cases in the DEFAINE project will contribute to the evaluation of the methods and tools as well as the integration into a common framework developed within the project The framework will enable the effective exploitation of the front-loaded product development approach in combination with Artificial Intelligence (AI). Front loading can significantly reduce the inefficiencies of the current engineering approach by enabling large-scale design exploration at the beginning or even before the actual start of a project. To this purpose, the DEFAINE framework will enable fast generation of distributed, re-configurable multidisciplinary engineering workflows, build on flexible sets of design and analysis automation solutions, largely based on Knowledge Based Engineering (KBE). The workflows are fed with varying sets of parameters and requirements, to automatically perform multiple design exploration studies using distributed and scalable computing infrastructures. With the help of AI techniques, the produced and stored design solutions and data are analysed to mine new knowledge, such as implicit rules and requirements. Using dedicated tools and modelling techniques, the mined knowledge is captured and fed back into the framework, to improve the search towards favourable solutions. At the actual start of the project, pre-optimized solutions for the given set of requirements, can be directly retrieved from the sets of pre-stored designs [43].

2. State of the Practice on Design Space Exploration

Every industrial aerospace partner within the DEFAINE project was asked to provide state of the practice on DSE and Artificial Intelligence (AI), and their realisation and implementation plans based on the findings within DEFAINE. Also, the anticipated market access impact of AI-enabled front-loaded engineering has been evaluated by each company. The following are the questions addressing both DSE and AI that were answered by each partner.

1. What knowledge do you have of current state-of-the-art (SotA), advantages and disadvantages of different methods and their applicability to your company?
2. In which way to you use any of these current techniques within your current work?
3. In which way are you planning to use these techniques (possibly developed by or with assistance of another partner)?

In the following, the answers are described by the industrial DEFAINE partners:

2.1. GKN Aerospace Sweden

Design Space Exploration

GKN Aerospace Sweden has experience in design exploration studies for detailed design optimization tasks within a specific discipline as well as more multidisciplinary design explorations for structural engine components. The main disciplines/engineering areas involved are CFD-Aero performance, FEM-Solid mechanics and DFM-Producibility. Advancements in Parametric Design /CAD/KBE has enabled a Multidisciplinary integration platform.
GKN Aerospace Sweden use design exploration in early phases with the purpose to better understand the relation between a requirements space and solutions space. What happens if a requirement changes? What solutions are still valid? Following the set-based approach to have a wide range of solutions in the beginning and sort out unfeasible solutions as the project proceeds and the requirements mature. Also, the methods can be used for minor sensitivity studies of more local design concepts/non-conformance.

Artificial Intelligence

GKN Aerospace Sweden has some experience of using AI for design space exploration via surrogate modelling and sensitivity analysis. The applied methods are linear regression, multi-adaptive regression splines (MARS), random forests (RF) and support vector machines (SVM). Furthermore, deep learning algorithms to analyse production data have been successfully applied, as well as Python tensor flow modules have been used to learn from images to automatically identify non-conformance in manufacturing processes. However, not so much information on AI-methods specific advantages and disadvantages has been systematically analysed for their applicability within GKN Aerospace Sweden. With the help of the DEFAINE project, the aim is to clarify the use and application of AI to improve decision support methods and tools.

Market Access

As a component Tier 1 supplier GKN Aerospace Sweden mainly functions as an independent risk and revenue sharing partner in cooperation with the major engine manufacturers. One of GKN Aerospace Sweden's strengths is lightweight design, which is possible by the use of advance welding technologies and more recently the use of additive manufacturing. GKN Aerospace Sweden's aim is to enhance the efficiency within multidisciplinary design (MDO), and specifically the opportunities provided by SotA development within AI and knowledge-based engineering (KBE) with the help of the results of the DEFAINE project.

2.2. GKN Fokker Aerostructures

Design Space Exploration

DSE is seen as the means to change the engineering process from a single design point activity to a sensitivity/gradient-based design approach, based on a multidimensional cloud of points. This enables engineers to fully understand the behaviour and response of the product design space with respect to its requirements space. This is only possible with a high level of automation, hence the need for AI based engineering applications and data analysis.
At GKN Fokker Aerostructures, DSE has already been used in the company on a limited scale, intention is to broaden the application in use as well as detail, and as such becoming part of the standard toolset of the engineer.

Artificial Intelligence

Artificial Intelligence is being applied in GKN Fokker Aerostructures. It is not a specialism, so no fundamental development is taking place. The scope of knowledge is on the industrial application of these techniques. In engineering, the main forms in which AI is used are:

(1) Expert systems / KBE Systems
(2) Optimization algorithms / search algorithms, and
(3) Response Surface Modelling (based on machine learning).

Typically, commercial tools are being used: the ParaPy KBE system and the Optimus process integration and design optimization application. Some of the main challenges are non-technical like introducing these technologies in the business or training employees in the associated specialist knowledge. Also other applications of AI outside the scope of DEFAINE such as augmented reality and collaborative robots are taking place. At GKN Fokker Aerostructures, both AI and DSE techniques are actively being used in aircraft programs:

(1) Applications based on ParaPy are being used to automatically generate geometry, cost and weight models for moveables. This is enabling very rapid updates of the FEM analysis for example. Challenges are in application robustness, development time and deployment.
(2) The AI/RSM (response surface methodology) and DSE techniques of Optimus are used to Build top-down weight estimation models based on actual data.
(3) To determine test loads for statically undetermined loads applications.
(4) Sizing of product geometry for minimum weight satisfying strength constraints, in which use is made of RSM-based allowable models, and
(5) To build strength allowable models based on limited test data.

Mixed Integer programming optimization algorithm (for non-continuous design variables), and Radial Basis Function and Kriging for RSM building are most commonly used. Also, the combination of building a RSM with adaptive DOE, running an optimization based on this RSM and subsequently perform an additional (native tool-based) optimization around the found optimum with the RSM, are applied techniques examples. These techniques have successfully been applied in several trade studies and have driven design choices. In these projects, commercial packages such as Optimus and ParaPy as well as open-source packages such as the RCE workflow system and Scipy (Powell algorithm) for flap mechanism sizing are used.

In DEFAINE, GKN Fokker Aerostructures is planning to make the application of the named techniques more efficient:

- Improve the KBE system-based applications, incl. their (re)configuration and deployment in the DEFAINE framework.
- Increase the experience in applying RSM and search methods and where possible leverage the partners and supplier's knowledge to make these more effective.
- Decrease the time to set up (Optimus) workflows by automation.

With more effective engineering services, workflows and sizing methods large scale studies can be performed, resulting in many data points. The Optimus MIP algorithm is very effective and robust in the sizing optimization as used. However, runtimes are relatively long, so the quest for more effective optimization algorithms able to be run with non-continuous design variables continues. GKN Fokker Aerostructures is expecting machine learning techniques from the DEFAINE project to:

- Effectively search through (large) datasets to rapidly find design solutions, design patterns, for new design cases.
- Present the most important information to engineers.

Market Access

In the current market, there is an increased demand for wider trade studies. Improvements from DEFAINE can have an important impact to GKN Fokker Aerostructures capability to execute such studies. An AI-enabled front-loaded design development process will not only lead to a more robust and more optimal product design, but it also opens up the potential of introducing new financial models and design lead time offerings by GKN Fokker Aerostructures, seen as disruptive in the market today and leading to new market potential.

2.3. GKN Fokker Elmo

Design Space Exploration

DSE provides the means to gain more knowledge and exploit more optimization potential within the design space. GKN Fokker Elmo exploits DSE within automated electrical wiring interconnect system (EWIS) design. Current optimization and trade studies are mostly done on (sub-)component level, through optimization on local EWIS optimization modules. More extensive global DSEs over the entire EWIS design process are planned but not yet executed. At GKN Fokker Elmo, experience has been gained with commercial platforms such as Optimus as well as open-source alternatives such as RCE and Python DSE libraries.

Within DEFAINE, GKN Fokker Elmo plans to use DSE to fill design database with results that can be used for front loading. Specific DSE/DoE algorithms will be used to make a smart selection of design points to run, in addition to deciding which parameters should be run with a higher variation of design points based on correlations between design parameters and the relevant output parameters.

Artificial Intelligence

On the broader scale, AI and machine learning applications within GKN Fokker Elmo are still limited. Basic knowledge is available through trainings and application of some prototyping. No machine learning is being used on mature applications yet. For design optimization, GKN Fokker Elmo uses a variety of optimization algorithms, for example the Python package *networkx* which provides a wide range of algorithms to find the shortest or most optimal path between different points on a pre-defined graph of nodes and edges. Such kind of optimizations assist in automatic routing of signals within an aircraft. In addition, GKN Fokker Elmo has developed several algorithms specifically for design-related tasks. Examples are an algorithm to pick the most optimal set of wire harness parts to complete a wire harness assembly, as well as an algorithm to select the optimal arrangement of wire-contact combinations on an aircraft connector,

Among the other DEFAINE partners, GKN Fokker Elmo is expecting to use AI methods within the front-loading process. For example, AI could assist in finding the best match for a given set of input parameters compared to a large database of front-loaded design results. In addition, AI algorithms in the direction of data analysis could assist in creating realistic estimates where input data might be missing in early conceptual stages of the design.

Market Access

GKN Fokker Elmo aims to use the project results to offer a unique new service to aircraft OEM's regarding the design of an electrical wiring interconnection system (EWIS) architecture. In addition, the existing EWIS design capability can be strengthened. By using results from the EWIS architecture optimization in the EWIS detail design, the EWIS design capability within GKN Fokker Elmo will improve, being able to provide cheaper and lighter wire harnesses to a large variation of projects within the aerospace market. With the help if AI-enabled front-loading, technologies can be exploited on a wide range of aircraft types, including commercial aircraft and business jets. In addition, due to recent market changes application is envisioned on for example urban air mobility (UAM) and short-range electrical vehicle platforms.

2.4. Saab AB

Design Space Exploration

Saab uses DSE at the department involved in the DEFAINE project within aircraft conceptual design studies. DSE is for Saab a necessity for future development to ensure that future products will be resilient to changes and provide relevant solutions to the customer. One example of DSE that has been developed and presented by Saab is technology assessment for future aircraft [44]. The in-house experience from that is that efficient visualization and exploration of the results is a must to gain insights from the amount of data. The basic prerequisite for being able to apply efficient DSE techniques is to develop efficient automated design processes. Knowledge about different (DSE) methods and design automations also contributed from collaborations with Linköping University. Weight estimations methods have been developed from statistics and in house data/calculation, and surrogate models have been created based on regression analysis. This is limited to available and accessible (IP restrictions!) data. DSE is seen as a way of gathering more data and from there apply modern metamodelling techniques for creation of models suitable for conceptual design.

DSEs are currently applied in early aircraft conceptual design, mainly based on the build-in capabilities offered by the PACElab tool. Some design of experiment (DOE) work has been performed for technology assessments and published [47]. Combination with uncertainty management has also been performed [45] [46]. This has been based on functionalities developed in MATLAB environment/ language. In industrial practice, DSE is already a standardized way of working. Typical usage of DSE has been for technology assessment where very large set of data need to be created. In one of Saab's published articles, sensitivities were used to reduce computational time to make DSE over a larger design space viable. Typical aircraft configuration explorations with regards to different technologies and different on-board systems will be one area of future, advanced DSE application. The goal is to set up all relevant problems that will be addressed within a holistic (complete and complex) DSE approach.

Artificial Intelligence

AI applications are not the primary topic for Saab in the DEFAINE project. Saab's aircraft pre-design and system design department currently makes use of AI for surrogate modelling, regression analyses, etc.. Further AI techniques have been employed in other parts of Saab, but not for aircraft conceptual design yet.
Within the pre-design phase, AI techniques for surrogate models, DSE exploration, and possibilities to provide decision support are of highest interest, e.g., as an attractive and more capable alternative to classical surrogate models. In stochastic models with discrete choices, the classical surrogate models currently used are not suitable or capturing all the relevant aspects. Consequently, investigating the possibilities to apply AI techniques to such problems is of high interest.

Market Access

The ability to perform efficient aircraft conceptual design (ACD) through front loading is a key to remain competitive and enable efficient development of new products. The DEFAINE project results will be implemented in current development methods and will leverage a baseline for some advanced design methods at Saab. Furthermore, the front-loaded engineering results obtained within DEFAINE are intended to complete current internal tool setups and especially the application of AI methods will enable continuous design improvements thought enlarged DSE and higher fidelity analyses.

3. State-of-the-Art on AI Methods in Aerospace Design Engineering

From Section 2 (the state-of-the-practices in industry), the following four identified main areas of interests of AI applications, based on requirements and expectations of AI from the industry partners are:

I. **AI for surrogate modelling:**
AI methods advantages and disadvantages and their applicability for surrogate modelling.

II. **AI-based decision support:**
AI methods and tools for design space exploration to improve decision support.

III. **Data analysis:**
Effectively search through (large) datasets in order to rapidly find design solutions, design patterns, for new design cases.

IV. **Interpretable surrogate models:**
Present the most important information to engineers.

Based on these needs, the focus is narrowed to present review on methods that have been used for surrogate modelling in aerospace applications for the purpose of design space exploration, their advantages and disadvantages, and which of the methods could provide important information on task of interest.
Yolanda et al., states that surrogate based optimization has proven very useful for exploratory design tasks as:

(1) it offers a global view of the characteristics of the design space, and
(2) it enables engineers to refine the design of experiments (DOE),
(3) to perform sensitivity analyses,
(4) characterize trade-offs between multiple objectives, and
(5) it helps to modify the design space if needed [24].

The authors have presented surrogate model-based optimization framework and illustrated step by step that includes setting up a problem, surrogate modelling, dimensionality reduction, sensitivity analysis and design space refinement. The case study of this study is that response surface based multi-objective optimization of a compact liquid-rocker radial turbine. More details on the purpose of use case, design parameters, design objectives, and constrains are provided in [24]. The response surface model in that work was built using Polynomial response, and their conclusion is that it helped to identify feasible design space for the studied design objectives. The response surface model-based optimization resulted in a 5% efficiency improvement compared to the baseline. This study and its detailed framework may serve as a valuable reference and guideline when setting-up own multi-objective optimization studies. Also, it includes results for AI-based data analysis and decision support (which are the second (II) and third (III) industrial areas of interests listed above). Also, guidance on dimensionality reduction and design space refinement are given in this work. However, as a delimitation, it make use of one response surface method only. Nevertheless, one may use a different response surface method depending on the complexity of the design problem at hand.

Cheng et al., have applied surrogate based optimization for a structural optimization for the non-circular vent hole on an aeroengine turbine disk [39]. They have presented a framework for the studied optimization task (use-case) in which the authors have proposed an improved support vector regression (ISVR) to extract some ignored valuable information from the existing database to build surrogate models. The proposed ISVR method is a new alternative to compared to surrogate-based optimization using support vector regression (SVR) or finite element method (FEM) based optimization. The results show that the proposed ISVR method is more suitable and valuable for engineering optimization, and it has two advantages: higher computational efficiency (92% reduction compared to time consuming finite element simulations) and better optimization effect compared to other methods. Similar to Yolanda et al., study, this study has also presented a detailed framework for surrogate based optimization using SVR, and it can guide on setting up an engineering design optimization problem, especially, extracting the valuable information (can address the fourth industrial areas of interest) from the sample database to build more accurate surrogate models.

Benaouali et al., have presented a fully automated framework for high-fidelity multidisciplinary optimization of aircraft wing [3]. The proposed framework integrates set of popular commercial software tools such as SIEMENS NX for geometric modelling, ICEM CFD for aerodynamic meshing, ANSYS FLUENT for flow solution, MSC.PATRAN for structural finite element modelling, and MSC.NASTRAN for structural sizing. In this framework, a surrogate-based optimization strategy is adopted to reduce the high cost of simulation models and allow the efficient solution of high-fidelity optimization problems. For the surrogate model generation, radial basis function (RBF) is selected, and all this implementation is done in Matlab. This paper provides a detailed framework for wing design optimization which could be used as guidance for setting up similar problems, however, not so much analysis of RBF on why this method is selected, for example advantages of RBF when it is applied on high-fidelity optimization problems.

Kudela et al., have presented a review on recent advances and applications of surrogate models for finite element methods computations [23]. The authors have presented literature (from the year 2013 to 2022) about methods for surrogate modelling, sensitivity analysis and uncertainty quantification, and surrogate based optimization for different applications which includes aerospace applications. Their literature shows that both Kriging and ANN are more widely used to build surrogate models, and that Kriging are popular methods to perform sensitivity analyses, too. This review study has very detailed information on all the methods that are applied to build surrogate models, and, hence, it provides list of alternative methods (can address the first and second industrial areas of interests), and the analysis on these methods to be able to choose more suitable method for a given task. The whole list of tables for references for methods are given in [23]. All these literature studies with different engineering applications show the potential of AI methods use to improve decision support methods and tools (can address the second industrial areas of interest).

Dasari et al., have focused on tree models as they could provide comprehensibility (*if-then* rules) that could be used to interpret model behaviour [7]. The purpose of this study is to support design space exploration of turbine rear structure (use-case) using tree based surrogate models. They have investigated M5P, and random forests (RF), linear regression and SVR methods and concluded that tree methods, especially RF, performed well on non-linear, high-dimensional, and small size of datasets. The authors have demonstrated the use of RF to build surrogate models (1) to predict the output of design objectives for as many design samples as possible to reduce computational expensive simulations needed to explore the design space (2) to get insights about the importance of design parameters towards the design objectives, and when needed, reduce number of parameters to narrow down its space for further analysis, and (3) to extract *if-then* rules that could help to better understand the reasoning of prediction behaviour of the surrogate model [8]. The presented approach can address the interpretability aspect of surrogate models and hence can present the important information to engineers to support decision making (can address the fourth industrial area of interest).

4. Recommendations on AI methods and Software Tools

4.1. Methods for Surrogate Modelling

Table 4.1 gives a comprehensive overview and lists example publications describing state-of-the-art methods for surrogate model generation including the methods capabilities and limitations.

4.2. Software Tools and Programming Languages

4.2.1. Languages for AI/NN Applications

Basically, almost every programming language can be used for programming AI/NN. In practice, the widest spread of AI/NN programming today is within the languages Python and Java. Other alternatives are:

- Prolog
- LISP
- C++

A listing with more language-specific details can be found at [6].

However, for the application of AI/NN for product development from and (technical) engineer point of view, the following widely used programming languages may be more convenient to be used. These are mainly:

- MATLAB
- Java
- C#
- Visual Basic .NET
- R (language for statistical analysis)
- Python
- JavaScript
- C / C++
- Julia

Other special neural network (NN) languages are:

- SNNS – Stuttgarter Neuronal-Network Simulator
- EpsiloNN Neural Description Language, University of Ulm (C language; discontinued?)
- OpenNN (C++; cross-platform)

Table 4.1: Surrogate Modelling methods overview.

Ref	Methods	Capabilities	Limitations or notes to consider.
[23] [24] [12] [40] [15]	Polynomial response surfaces, and linear regression.	For problems that are not high-dimensional, display low modality, or where data are relatively inexpensive to compute, the use of polynomial surrogates may be an attractive (and correct) choice. Recommended when the sample size is low.	Polynomial surrogates remain generally not well suited for the nonlinear, multidimensional problems. Multivariate polynomial method is not sensitive to the change of the sample size. It can only applied to regression tasks.
[23] [38] [12] [40] [15] [36]	Kriging (universal Kriging, ordinary Kriging, and simple Kriging)	Widely used methods for building surrogate models due to its better performance. It can effectively represent highly nonlinear and multidimensional functions.	Important feature of Kriging is the selection of a suitable covariance function, hence, more knowledge is needed. This method is the can perform well when the sample size is high. The use of Kriging method is not trivial to construct surrogate models due to its global optimization process.
[23][3] [12] [40][8]	Radial basis functions	Recommended when the data has high-order non-linearity.	There is no firm conclusion in the literature that show whether RBF are better than the others.
[23] [39] [12]	Support vector regression	It can handle both continuous and categorical data. SVR able to approximate more complex landscapes because of its kernels.	
[23] [14]	ANN and MLP	Well known approach for constructing simple and fast approximations of complex computer codes.	Encoding is needed for categorical data. Requires a lot of data for very complex models. Large amount of trail-and-error associated with the use of this technique.
[23] [8]	Decision trees and random forests	Can handle all types of data, shown better for non-linear, high-dimensional and small size of samples, and can provide information (if-then rules) to able to explain model prediction reasoning.	
[23]	Adaboost	It is an ensemble method i.e., combination with weak learners, like decision trees. Used in both classification and regression.	
[40]	Bayesian networks	Can be applied for both continuous and categorical data. Recommended when the noise in data is high.	Determines class based on probability.
[15] [8]	MARS	Shown better for the data with linear relationship, but could also be used for non-linear data as it created splines in design space. The major advantages of using the MARS is to be accuracy and major reduction in computational cost associated with constructing the met model.	Only applied for regression tasks.
[23] [20] [21] [22]	SVD	It can used for solving problems in structural optimization, multiple regression (surrogate modelling/interpolation), and dimensionality reduction	Only applied for regression tasks.

4.2.2. Benchmarking Actors and Platforms

Beside many open-source research activities also many commercial actors are very active on the research/use of deep learning and its application and execution. Gartner published 2021 a benchmark among the 20 biggest actors of data science and machine learning (DSML) platforms, see the *"Magic Quadrant for Data Science and Machine Learning Platforms"* report[1], identifying strengths and cautions of the entire platforms. A graphical representation of this benchmark is shown in Figure 4.1 [13].

Figure 4.1: The Gartner Magic Quadrant for Data Science and Machine Learning Platforms in 2021. Source: [13]

4.3. Libraries for AI/NN Engineering

4.3.1. Python

Python offers numerous libraries for AI/NN engineering such as:

- TensorFlow (library; Python/C++; Apache-License 2.0)
- SMT Surrogate Modeling Toolbox by M. Bouhlel [4]
- Scikit learn (built on Numpy, SciPy and matplotlip)
- Keras
- PyTorch
- Pandas
- Anoconda (a free open-source python distribution platform)

[1] https://www.gartner.com/reprints/?id=1-25DG78TT&ct=210303&st=sb

To get started with AI and Python, a simple "click and run" Machine Learning Code Generator can be found at https://ml-generator.herokuapp.com/.

4.3.2. Julia

Also Julia supports a bench of AI/NN engineering toolboxes. Widely spread are:

- MLJ (Machine Learning in Julia) toolbox:
  provides a common interface and meta-algorithms for selecting, tuning, evaluating, composing, and comparing over 160 machine learning models written in Julia and other languages.
- NeuralPDE.jl: Scientific Machine Learning for Partial Differential Equations[2].

4.3.3. Java

Weka[3] is one of the open-source software tools that is implemented in Java. Initially, the tool has a collection of basic machine learning algorithms that can be used to build surrogate models or response surface models for the purpose of design space exploration. Later, a deep learning module[4] has been added to the Weka tool that contains neural networks, CNN, RNN, and LSTMs. Furthermore, Weka has algorithms for data preparation, classification, clustering and visualization. It can run on Windows, MacOS and Linux.
Weka is easy to use due to its graphical user interface. The developer has put up a number of tutorials and courses on how to use it, see https://www.cs.waikato.ac.nz/ml/weka/index.html.

4.3.4. R Programming

R is a programming language initially developed for statistical computing[5]. It has modules for data mining and machine learning algorithms. R software is an open-source environment and has command line interface. Third party GUIs such as RStudio and Jupyter (a notebook interface in Anaconda tool) are available for free use.

4.3.5. 4.1.5 MATLAB

MathWorks is one of the leaders in the 2021 Gartner Magic Quadrant for Data Science and Machine Learning Platforms, see Figure 4.1. Special libraries – in MATLAB nomenclature called toolboxes – are:

- Deep Learning Toolbox
- Image Processing Toolbox
- Neural Network Toolbox
- Reinforcement Learning Toolbox
- Statistics and Machine Learning Toolbox
- Computer Vision Toolbox
- Fuzzy Logic Toolbox
- Statistics Toolbox

[2] See https://neuralpde.sciml.ai/dev/.
[3] See https://www.cs.waikato.ac.nz/ml/weka/.
[4] See https://deeplearning.cms.waikato.ac.nz/#wekadeeplearning4j-deep-learning-using-weka.
[5] See https://www.r-project.org.

5. Summary

In engineering disciplines, the utilization of surrogate modelling has been increased as it provides great opportunity to explore design space without needing more computationally expensive simulations. However, the selection of suitable methods for surrogate modelling for a given problem is not straight forward due to the complexity of the problems and the capabilities and limitations of different methods. Hence, this document focused on describing both state-of-the-practice of industrial partners within the DEFAINE consortium, and the state-of-the-art to present latest developments on methods to build surrogate models for the purpose of design space exploration, sensitivity analysis, and design optimization.

From the state-of-the-practice (Section 2), it is clear from the DEFAINE industrial partners that there is need for AI methods for surrogate modelling for design space exploration tasks. Hence, a review is conducted to present latest developments in order to address the identified areas-of-interests by industrial partners. In brief, (1) frameworks from studies [24] [39] [3] could be used to setup design optimization problem with the use of surrogate modelling (2) approaches or methods from studies [7] [8] could be used to build more compressible surrogate models to provide more understanding of design space (3) methods with capabilities and limitations from Table 4.1 could be used to choose suitable methods for surrogate modelling, and (4) tools from Section 4.2.1 could provide an understanding on available tools for implementing surrogate models. Furthermore, more detailed descriptions about methods such as SVD and NN, and example use cases are given in the Appendix.

Bibliography

[1] Alexa Aguilar, Patrick Butler, Jennifer Collins, Markus Guerster, Bjarni Kristinsson, Patrick McKeen, Kerri Cahoy, and Edward F. Crawley. Tradespace exploration of the next generation communication satellites. *MIT Open Access Articles*, 2019.

[2] Mark Azzam, Jan-Christoph Haag, and Peter Jeschke. Application concept of artificial neural networks for turbomachinery design. *Computer Assisted Methods in Engineering and Science*, 16(2):143–160, 2017. URL: https://cames.ippt.pan.pl/index.php/ cames/article/view/158.

[3] Abdelkader Benaouali and Stanisław Kachel. Multidisciplinary design optimization of aircraft wing using commercial software integration. *Aerospace Science and Technology*, 92:766–776, 2019.

[4] Mohamed Bouhlel, John Hwang, Nathalie Bartoli, Rémi Lafage, Joseph Morlier and Joaquim R. R. A. Martins. A Python surrogate modeling framework with derivatives. *Advances in Engineering Software*, p.102662, 2019. doi:https: //doi.org/10.1016/ j.advengsoft.2019.03.005.

[5] J. H. Bussemaker, T. de Smedt, G. la Rocca, P. D. Ciampa, and Björn Nagel. System architecture optimization: An open-source multidisciplinary aircraft jet engine architecting problem. In *AIAA AVIATION 2021 FORUM*, August 2021. AIAA Aviation 2021 Forum. doi:10.2514/6.2021-3870.

[6] Monomita Chakraborty. What are the Best Programming Languages for Artificial Intelligence, 2021. URL: https://www.analyticsinsight.net/what-are-the-best-programming-languages-for-artificial-intelligence/. [Online; accessed 2021-12-19].

[7] Siva Krishna Dasari. *Tree Models for Design Space Exploration in Aerospace Engineering*. PhD thesis, Blekinge Tekniska Högskola, 2019.

[8] Siva Krishna Dasari, Abbas Cheddad, and Petter Andersson. Random forest surrogate models to support design space exploration in aerospace use case. In IFIP International Conference on Artificial Intelligence Applications and Innovations, pages 532–544. Springer, 2019.

[9] Data Flair. Introduction to Learning Rules in Neural Networks. https://data-flair.training/blogs/learning-rules-in-neural-network/, 2022. [Online; accessed 05-February-2022].

[10] X. Feng, S. Sander-Tavallaey, and J. Ölvander. Cycle-Based Robot Drive Train Optimization Utilizing SVD Analysis. 33rd Design Automation Conference, Las Vegas, Nevada, U.S., 2007.

[11] Matthew E. Fizgerald and Adam M. Ross. Artificial Intelligence Analytics with Multi-attribute Tradespace Exploration and Set-based Design. In Proceedings of the 16th Conference on Systems Engineering Research CSER, 2019.

[12] Alexander IJ Forrester and Andy J Keane. Recent advances in surrogate-based optimization. Progress in aerospace sciences, 45(1-3):50–79, 2009.

[13] Gartner. Magic quadrant for data science and machine learning platforms. 2022, https://www.gartner.com/reprints/?id=1-25DG78TT&ct=210303&st=sb
[Online; accessed 07-February-2022].

[14] James Hammond, Nick Pepper, Francesco Montomoli, and Vittorio Michelassi. Machine learning methods in CFD for turbomachinery: A review. International Journal of Turbomachinery, Propulsion and Power, 7(2):16, 2022.

[15] Ruichen Jin, Wei Chen, and Timothy W Simpson. Comparative studies of metamodelling techniques under multiple modelling criteria. Structural and multidisciplinary optimization, 23(1):1–13, 2001.

[16] Eunsuk Kang, Ethan Jackson, and Wolfram Schulte. An approach for effective design space exploration. In Monterey Workshop, pages 33–54. Springer, 2010.

[17] Yongtae Kim, Youngsoo Kim, Charles Yang, Kundo Park, Grace X. Gu, and Seunghwa Ryu. Deep learning framework for material design space exploration using active transfer learning and data augmentation. NPJ Computational Materials, 7(1):140, 2021.
URL: https://doi.org/10.1038/s41524-021-00609-2.

[18] L. Knöös Franzen, R. Chaitanya Munjulury, and P. Krus. Ontology-assisted Aircraft Concept Generation. ICAS 2022 Conference, Stockholm, Sweden, 2022.

[19] Petter Krus. Computational Tools for Aircraft System. In Proceedings of the 26th International Congress of the Aeronautical Sciences, ICAS, 2008.
URL: http://www. icas.org/ICAS_ARCHIVE/ICAS2008/PAPERS/618.PDF.

[20] Petter Krus. Design Space Configuration for Minimizing Design Information Entropy. Smart Innovation, Systems and Technologies, 34(0):51–60, 2015.
URL: https://doi. org/10.1007/978-81-322-2232-3_5.

[21] Petter Krus. Models Based on Singular Value Decomposition for Aircraft Design. In Proceedings of the Aerospace Technology Congress, Swedish Society of Aeronautics and Astronautics, Stockholm, Sweden, 2016.

[22] Petter Krus. Design Space Configuration Trough Analytical Parametrization. Proceedings of ICoRD, v.1, p. 15–24. Research into Design for Communities, Springer Singapore, 2017.

[23] Jakub Kudela and Radomil Matousek. Recent advances and applications of surrogate models for finite element method computations: a review. Soft Computing, pages 1–25, 2022.

[24] Yolanda Mack, Tushar Goel, Wei Shyy, and Raphael Haftka. Surrogate model-based optimization framework: a case study in aerospace design. In Evolutionary computation in dynamic and uncertain environments, pages 323–342. Springer, 2007.

[25] J. Mandel. Use of the Singular Value Decomposition in Regression Analysis. American Statistics, 36(1):15–24, 1982.

[26] Berkin Ozisikyilmaz, Gokhan Memik, and Alok Choudhary. Efficient system design space exploration using machine learning techniques. 45th Annual Design Automation Conference, DAC ’08, p. 966–969, New York, U.S., 2008., doi:10.1145/1391469.1391712.

[27] Nestor V Queipo, Raphael T Haftka, Wei Shyy, Tushar Goel, Rajkumar Vaidyanathan, and P Kevin Tucker. Surrogate-based analysis and optimization. Progress in aerospace sciences, 41(1):1–28, 2005.

[28] Adam M. Ross and Daniel E. Hastings. The tradespace exploration paradigm. In INCOSE, 2005.

[29] Andrea Saltelli, Stefano Tarantola, Francesca Campolongo, and Marco Ratto. Sensitivity analysis in practice: a guide to assessing scientific models. Chichester, England, 2004.

[30] Scaled Agile Inc. Set-Based Design. https://www.scaledagileframework.com/ set-based-design/, 2021. [Online; accessed 23-Feruary-2022].

[31] SEARI. Multi-Attribute Tradespace Exploration (MATE). http://seari.mit.edu/ mate.php, 2022. [Online; accessed 05-February-2022].

[32] Satyabrata Sen and Neena Imam. Machine learning based design space exploration for hybrid main-memory design. In International Symposium on Memory Systems (MEMSYS), 9 2019. URL: https://www.osti.gov/biblio/1606980, doi:10.1145/3357526.3357544.

[33] Eric Specking, Gregory Parnell, Edward Pohl, and Randy Buchanan. Early design space exploration with model-based system engineering and set-based design. Systems, 6(4), 2018. URL: https://www.mdpi.com/2079-8954/6/4/45, doi: 10.3390/systems6040045.

[34] Eric Spero, Michael Avera, Pierre Vladez, and Simon Goerger. Tradespace exploration for the engineering of resilient systems. techreport, US Army Research Laboratory, ARL, ARL-TR-7288, May 2015. URL: https://apps.dtic.mil/sti/pdfs/ ADA622928.pdf.

[35] S. Thomke, and T. Fujimoto. The effect of front loading problem-solving on product development performance. Journal of Product Innovation Management, New York, 2000.

[36] G. Gary Wang and Songqing Shan. Review of metamodeling techniques in support of engineering design optimization. Journal of Mechanical design, 129(4):370–380, 2007.

[37] Wikipedia. Künstliche neurale Netze, The Free Encyclopedia. https://de.wikipedia.org/ wiki/K%C3%BCnstliches_neuronales_Netz, 2022. [Online; accessed 13-January-2022].

[38] Chen-Zhou Xu, Zhong-Hua Han, Ke-Shi Zhang, and W Song. Surrogate-based optimization method applied to multidisciplinary design optimization architectures. In 31st congress of the International Council of the Aeronautical Sciences (ICAS 2018), 2018.

[39] Cheng Yan, Zeyong Yin, Xiuli Shen, Dong Mi, Fushui Guo, and Dan Long. Surrogate-based optimization with improved support vector regression for non-circular vent hole on aero-engine turbine disk. Aerospace Science and Technology, 96:105332, 2020.

[40] Dong Zhao and Deyi Xue. A comparative study of metamodeling methods considering sample quality merits. Structural and Multidisciplinary Optimization, 42(6):923–938, 2010.

[41] Navid Zobeiry and Anoush Poursartip. Theory-guided machine learning for process simulation of advanced composites, 2021. arXiv:2103.16010.

[42] Melody Stokes and the MOKA Consortium. Managing Engineering Knowledge: MOKA: Methodology for Knowledge-bBased Engineering Applications. ASME Press, ISBN: 9780791801659, 2001.

[43] DEFAINE project Application, ITEA4 Project, Grant Nr. 19009 DEFAINE, 2021, https://itea4.org/project/defaine.html [Online; accessed 17-Maj-2022]

[44] Christopher Jouannet, Kristian Amadori and Erik Bäckström. Selection of future technologies during aircraft conceptual design. 55th AIAA Aerospace Sciences Meeting, AIAA-2017-0233, 9 - 13 January 2017, https://doi.org/10.2514/6.2017-0233

[45] Bianchi Davi, Amadori Kristian, Bäckström Erik, Jouannet Christopher and Secco Ney. An Uncertainty-based Framework for Technology Portfolio Selection for Future Aircraft Program. AIAA Scitech 2021 Forum, AIAA-2021-1479, 2021 https://doi.org/10.2514/6.2021-1479

[46] Christopher Jouannet, Kristian Amadori, Erik Bäckström and Davi Bianchi. Uncertainty management in technologies prioritization for future aircraft program. AIAA Scitech 2020 Forum, AIAA-2020-0019, 2020 https://doi.org/10.2514/6.2020-0019

[47] Kristian Amadori, Erik Bäckström and Christopher Jouannet. Future technologies prioritization for aircraft conceptual design. AIAA Aerospace Sciences Meeting, AIAA-2018-1746, 2018, https://doi.org/10.2514/6.2018-1746

APPENDIX: Examples and In-depth Descriptions

A Singular Value Decomposition (SVD)

Singular value decomposition (SVD) introduced already in 1982 by Mandel [25] is an early machine learning (ML) algorithm most frequently used for *dimensionality reduction*. SVD applies *unsupervised learning* and is characterized but its low computational costs, enabling a just-in-time use of the algorithm. SVD can be interpreted as a kind of *multiple regression analysis* can be used to create models that often show good agreement around the data sets, used to establish the model. It is therefore useful for surrogate modelling as well as for important parameter identification and the creation of new artificial parameters of high sensitivity for dimensionality reduction.

A.1 SVD and Related Methods for Dimensionality Reduction

When the primary functional characteristics are highly correlated, it is an advantage to use *principal component analysis* (PCA). In this way the coordinate system is rotated in such a way that two new parameters, the principal components, become uncorrelated. First, the statistical properties become sounder, and secondly and perhaps more important, the explicit constraint can be set on the parameters that now provide a better fit around the statistical data set, such that the design space gets a shape more consistent with the space spanned by the data set. This is especially useful when used for system optimisation.
Rotating the coordinate system means that the explicit limitation of the variables can have a much tighter fit towards the measurement data. Using PCA, it is possible also to include the distribution of data in the design space, into the model.

A.2 Use and Application of SVD

Using SVD, introduced by Mandel [25] already in 1982, it is possible to do a PCA and create a model that has a few synthetic parameters as inputs and all the attribute of the design as outputs. The output of the model can include both design parameters and functional characteristics. It is then possible to quickly estimate a design from given requirements, by solving the resulting system of equations. Interestingly, it can also be used to estimate performance and other characteristics from limited data. In Feng [10], SVD was used to reduce the number of variables for optimization of an industrial robot.

Another very useful SVD application is for modelling of components and subsystems. Krus showed in [20] and [21] that SVD was used for models based on statistical data of components and systems, such as aircraft and aero engines. In this way models with high accuracy can be produced that can relate e.g., engine dimensions (such as diameter, length, etc.), to attributes such as weight, bypass ratio, trust and specific fuel consumption. It is also possible to include the year of introduction as one variable and in this way also have a mechanism for technology evolution over time, although looking at times beyond the present will take us outside the dataset that the model is based on (extrapolation).

In a design situation the SVD model can be used in the role of a surrogate model. Instead of making a parametric design of a higher fidelity, which is optimized for each situation, it is possible to optimize for a few situations and then build a SVD model based on these. In this way, a meta model with high accuracy can be obtained. Ones an optimal solution has been reached, it can be recalculated and be added to the set of data points the SVD model is based on.

Finally, SVD analysis can be used to evaluate a given parametrization by studying the correlation with the ideal SVD parameter set. This is useful since it sometimes is an advantage to have a parametrization that have a clearer interpretation than the synthetic SVD parameter set can provide. This was shown in [22]. Interestingly, it is also possible to derive the number of driving requirement in a design by studying several instances of a particular kind of product.

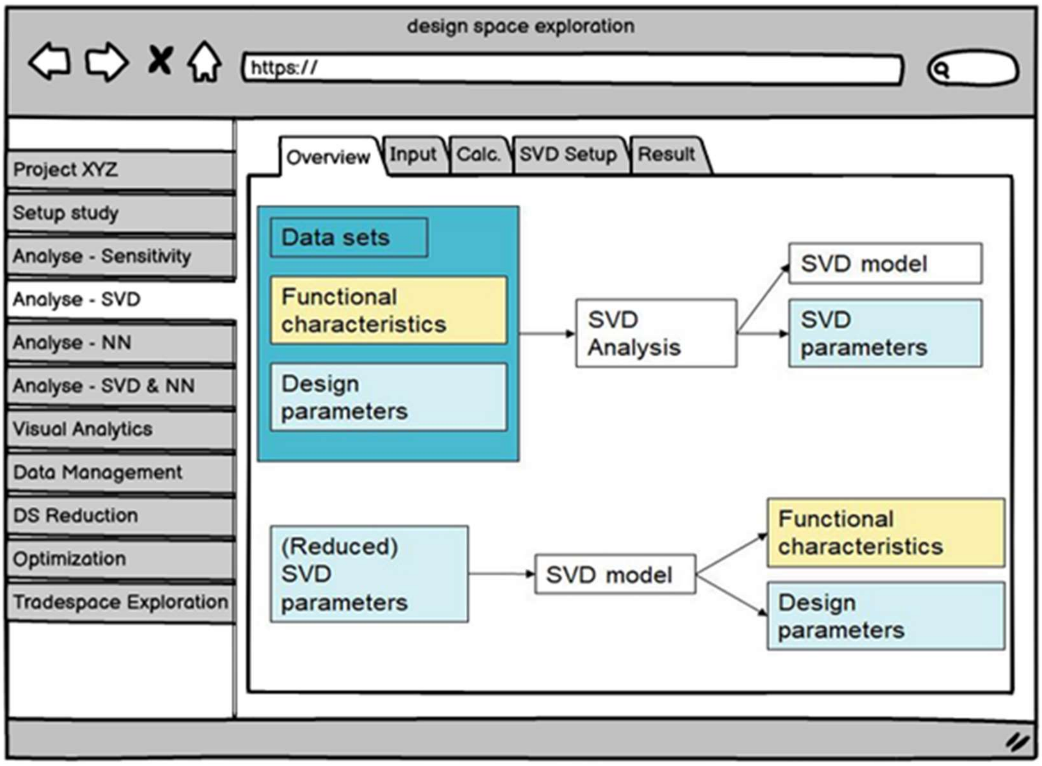


Figure A.1: SVD for parameter reduction and surrogate modelling.

B Artificial Neural Networks (ANN)

B.1 General Basics

Artificial Neural Networks (ANNs) are usually sequential. No parallelisation (like in a human brain) is usually implemented which distinct them by the currently so widely used so-called transformer networks for language modelling. The ANN size and shape (number of layers, number of nodes in each layer) must be (like the setup of an optimisation algorithm) adapted based on the problem at hand (number of, type and characteristic of the available datasets); it may be set by rules of thumb and user experience but may also be optimized by iterating with different setups (like e.g., CFD mesh independence analysis). The more is not always the better: over-training and too large network setups might lead to lower performance!

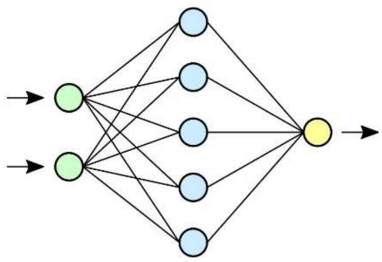


Figure B.1: A simplified one-layer ANN.

B.2 Use and Main Applications of ANN

ANN may be primarily used when no or low explicit knowledge is available. Also, they are widely used within control theory (Fuzzy systems).

Application Examples of ANN

NN may be applied to a wide range of engineering, data science, physics, and other applications. Typical NN named example applications are [9]:

- Handwriting Recognition
- Traveling Salesman Problem
- Image Compression
- Stock Exchange Prediction
- Some Just-in-time (JIT) compiler

Transparency & Understanding

A black box model (BBM) with (non-)transparency of the different layers:

- model level (simulation capacity)
- component level (separation possibility)
- algorithmic level (algorithm transparency)
-

But in case total understanding is required (as it is in the typical DEFAINE applications within conceptual design to gain understanding of the analysed data / surrogate models), explainable AI is of large interest (e.g., to give architecture recommendation, suggest changes, etc.).

B.3 ANN Learning Algorithms

- Supervised Learning
  - single layer: Delta-rule (also known as Perceptron-rule)
  - multi-layer: Backpropagation (aka a generalisation of the Delta-rule)
- Unsupervised Learning
- Reinforced Learning: kind of (inter-)active/dynamic learning (executable model/system must exist and being available!)

The activation Functions may be of linear or non-linear type.

Layer Structure

The layer of the network depends on and the training algorithm, e.g., can the delta-rule (modification in sympatric weight of a node is equal to the multiplication of error and the input) only train a single-layer NN (for a brief introduction to NN training rules, see https://data-flair.training/blogs/learning-rules-in-neural-network/).
The layer structure is often non-priori known and may be a result of *"trial and error"* (e.g., by application of evolutionary algorithm and (error) back-propagation →feed-forward networks) or rule-of-thumb and expertise.

Deep Learning

Deep learning (also: hierarchical learning) terminology is used for artificial neural network (ANN), mimicking a high abstraction capability through the use of multiple hidden layers.

Recurrent Networks

Includes feedback loops, with time delay acts as a dynamic memory (e.g., compare with agent-based simulation of behaviour/brain).

ANN Layer Features

The network layer structure and properties may be adapted to the task at hand. This selection has to be performed by the users. Widely used and adopted layer properties are:

- (alternating) Convolutional layers
- Pooling Layers
- Fully Connected Layers
- Back-propagation (algorithm)

Learning Rules

Depending on the type of ML and the type of problem at hand, different learning rules may be applied. The most common learning rules for NN are [9]:

- **Hebbian learning**
  identifies how to modify the weights of nodes of a network
- **Perceptron learning**
  starts its learning by assigning a random value to each weight.
- **Delta learning**
  the modification in sympatric weight of a node is equal to the multiplication of error and the input.
- **Correlation learning**
  the correlation rule is the supervised learning.
- **Outstar learning**
  may be applied in case nodes or neurons in a network are arranged in a layer. [9]

> **Fact Box .1** **Neural Network Setup and Training**
> While the NN theory is simple, it´s application and use is not due to the sheer infinite possibility of different shapes, pre-allocation and tuning (aka training) setups (compare with optimization algorithm setups). Zobeiry [41] shows for composite structure surrogate modelling a two-stage learning approach.

B.4 Problems and Disadvantages of Neural Networks

- The **training** of NN are multidimensional non-linear optimization problems with all invoked challenges: finding the global and not local optimum very costly and requires multiple runs with varying start values.
- **Availability of training data sets:** amount and configuration. Delicate problem to not train the wrong thing: e.g., brightness, see also discrimination.
- **Heuristic approaches:** overgeneralisation & overfitting so that the NN learns 1:1 the pattern of the training data set. This requires careful selection of the network architecture!
- **Encoding of the data:** *"...The coding of the training data must be adapted to the problem and, if possible, chosen to be redundancy-free. The form in which the data to be learned is presented to the network has a great influence on the learning speed and on whether the problem can be learned by a network at all. Good examples of this are voice data, music data or texts. Simply feeding in numbers, for example from a speech .wav file, rarely produces a successful result. The more precisely the problem is posed solely by pre-processing and coding, the more successfully an ANN can process it."* [37]
- **Start value problem:** the start values of the network weightings play an important role (like the algorithm setup and start values of an optimisation)!

C Examples and Implementation Strategies

This appendix shows different examples and implementation strategies using NNs for product engineering. Also, the necessary background explanations in form of nomenclature definitions are given as well as the involved work topics are listed.

C.1 Design Analysis

The objective of system design analysis is to obtain information about the nature of the design solution, and how it can be changed to fulfil the requirements. An overview of design analysis activities is given in Figure C.1. Here, different matrix methods are useful since they can be used to display and visualize the mapping of relations between system parameters and functional characteristics. The application of a collection of these methods to aircraft system design was presented in [19].

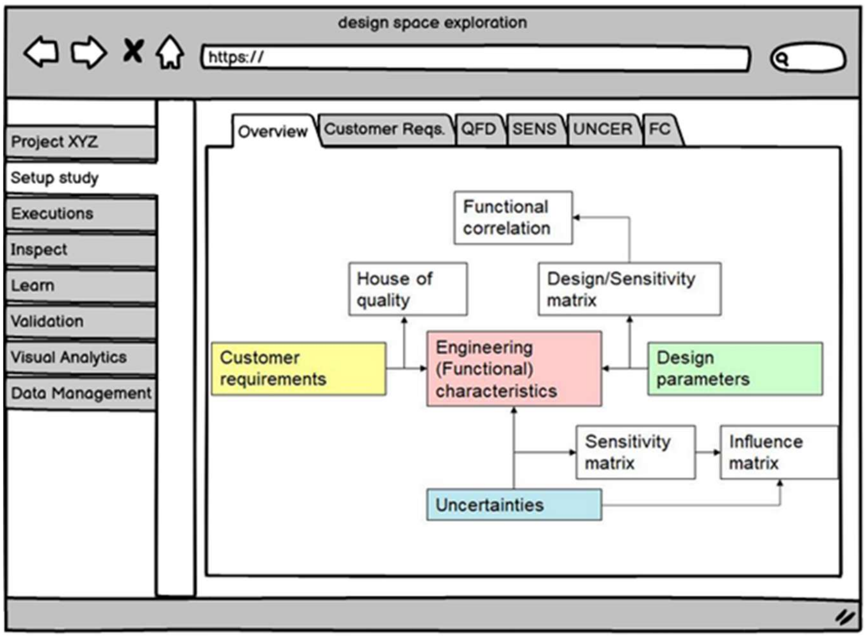


Figure C.1: Design analysis tools and relations.

Fact Box .2 **System Architecture/MBSE on the Example of Jet Engine Architecting (open data)**

This work lifts the importance of the different nature of design variables which can be of type:

- categorical,
- integer, or
- continuous

For more information see **"System Architecture Optimization: An Open-Source Multidisciplinary Aircraft Jet Engine Architecting Problem"** [5]. Also, the related open-source optimization framework data (Python code) is available at https://github.com/jbussemaker/OpenTurbofanArchitecting ▪

Fact Box .3 **Common (System) DSE using Machine Learning**

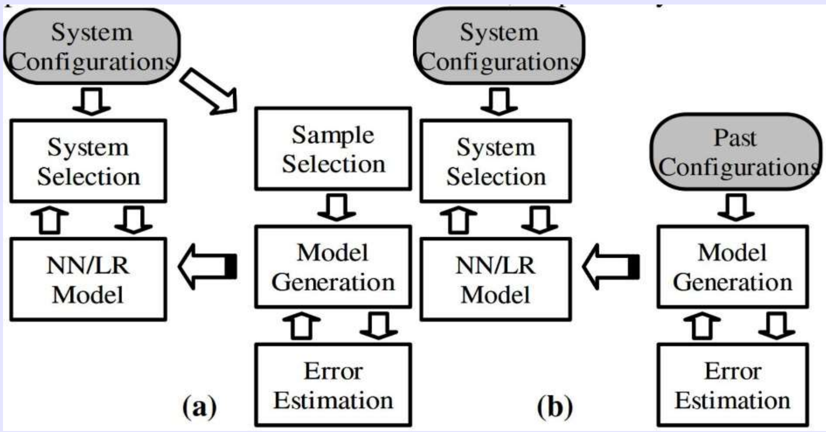


Figure FB3: Overview of DSE using predictive modelling: (a) sampled DSE and (b) chronological predictive models. Source: [26].

For more information see "**Efficient system design space exploration using machine learning techniques**" [26].

Fact Box .4 **DSE by ML for Computer Memory Design**

In the work by Sen & Imam [32], published 2019, the successful application of ML for DSE to find the optimal PC memory design/architecture. The used framework is shown in Figure FB4. Figure 5.

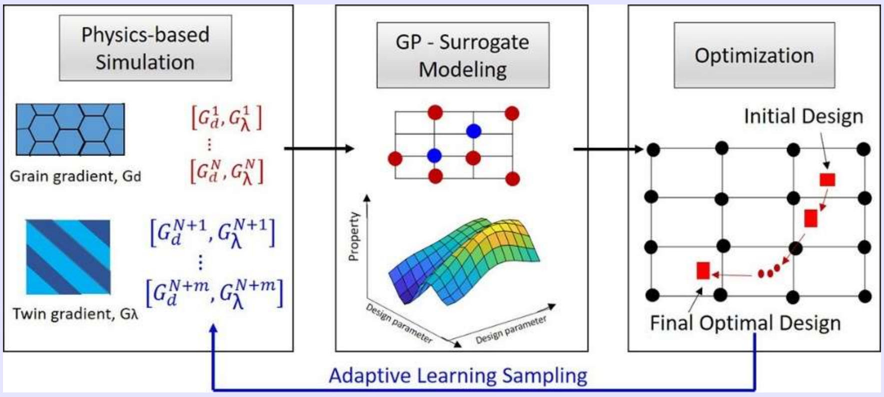
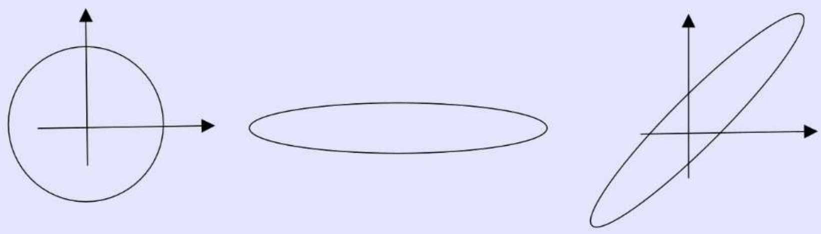


Figure BF4: Setup of the ML-based DSE for memory architecture in [32].

For more information see **"Machine Learning Based Design Space Exploration for Hybrid Main-Memory Design"** [32].

Fact Box .5 **Principal Component Analysis using SVD**

SVD is frequently used within DSE to reduce the size and the dimensionality of a design space. This is done be the change of the design space dimensions, thus the entire (design or other) parameter that span the (design) space into a new space based on artificial design parameters composed out of the original ones (see Figure FB5). The new artificial parameters, which are by a coupling (aka weighting) matrix coupled to the original (design) parameters are composed/sorted in a descending order of influence, so that with a known error margin the number of (the new artificial) SVD parameters can be often reduced by ca. 50% within a 5% error margin.



$\mathbf{S}$ $\qquad$ $\mathbf{S}\times\mathbf{W}$ $\qquad$ $\mathbf{X}=\mathbf{S}\times\mathbf{W}\times\mathbf{V}^{T}$

Figure FB5: Visualization of the design or trade-space change and reduction by the application of SVD.

For more information see **"Models Based on Singular Value Decomposition for Aircraft Design"** [21].

C.2 Tradespace

Definition C.2 **Tradespace / Trade Space**

… is the space spanned by the completely enumerated design variables, which means given a set of design variables, the tradespace is the space of possible design options. The expansion of this tradespace is a "creative recombination of current resources or systems to create a new system," which would involve generation of either new design variables or reconfigurations of existing combinations of variables. [28]

Interesting reading of a (multi-attribute) set-based design engineering involving AI is presented by Fitzgerald and Ross (MIT) in [11]. Typical task in set-based design (SBD) is the detection (alternative: a priori definition) of so-called sets, which are clusters in the tradespace (e.g., of similar design), see Figure C.2X. An extensive and well-documented tradespace analysis of communication satellites design can be found in [1]. Tradespace exploration is realized by many tools, a listing can be found in [34, tab.1]

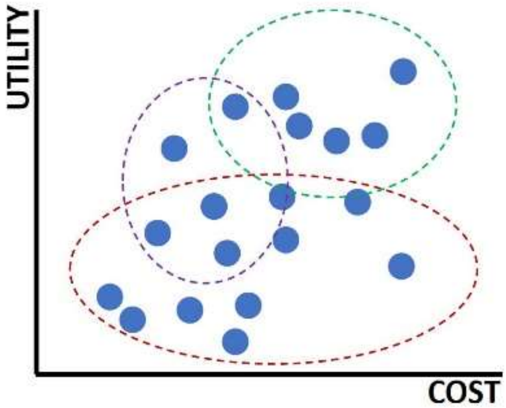


Figure C.2: Identification of clusters, so called sets within a tradespace. Source: [11]

C.3 Set-based Design

Definition C.3 **Set-based Design (SBD)**

*"Set-based design (SBD) is a practice that **keeps requirements and design options flexible for as long as possible during the development process.** Instead of choosing a single point solution upfront, SBD identifies and simultaneously explores multiple options, eliminating poorer choices over time."* [30]

Set-based design (SBD) comes with some drawbacks (mainly the additional simulation/calculation effort required) but is especially advantageous for applications with *"... a high degree of innovation, variability, or immovable deadlines"* [30]. For multi-objective or multi-domain problems, SBD enables flexibility (preserving a broad set of design options), cost reduction and prototyping speed through early and frequent validation of design alternatives [30]. An application example on SBD for unmanned aerial system (UAS) design is shown in [33], see Figure C.3.

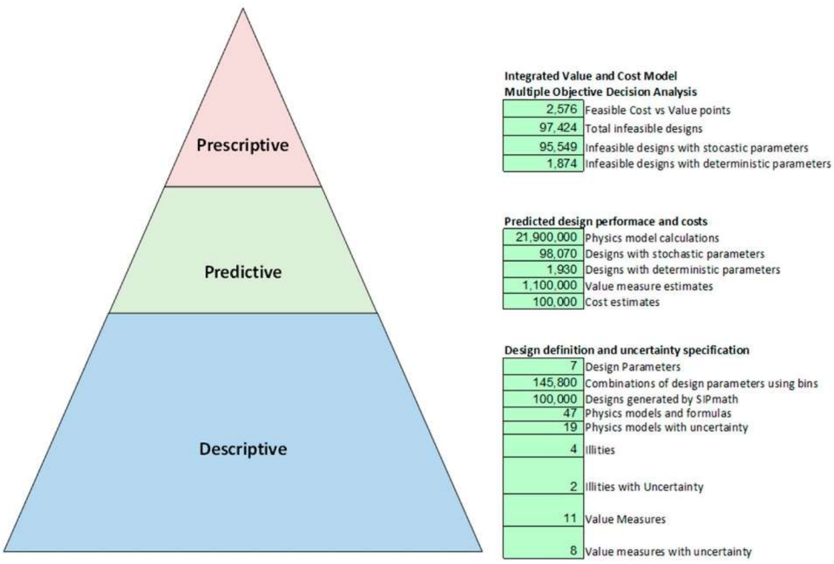


Figure C.3: Set-based design trade-off hierarchy in the UAV design study presented in [33].