Financial Analysis in Multidisciplinary Design Optimization

by

Rishab Mardia

Submitted to the
Department of Mechanical Engineering
in Partial Fulfillment of the Requirements for the Degree of

Master of Engineering in Advanced Manufacturing and Design

at the

Massachusetts Institute of Technology

February 2021

Signature of Author: ____
Department of Mechanical Engineering
February, 2021

Certified by: ____
Maria Yang
Professor of Mechanical Engineering
Thesis Supervisor

Accepted by: ____
Nicolas Hadjiconstantinou
Chairman, Department Committee on Graduate Theses

Financial Analysis in Multidisciplinary Design Optimization

by

Rishab Mardia

Submitted to the Department of Mechanical Engineering
on February, 2021 in Partial Fulfillment of the
Requirements for the Degree of
Master of Engineering in Advanced Manufacturing and Design

Abstract

MDO is moving beyond the small group of NASA and Aerospace companies and is increasingly being adopted by organizations around the world. With MDO, we can optimize across multiple disciplines and find the ideal design which maximizes benefit to the company and society. Given the complexity of working with multiple disciplines and stakeholders, it is important to have a single metric which teams and organizations can use to choose the best design. Since financial metrics play a dominant role in the decision-making process, we can use them to choose the best design for the company.

In the thesis, we created a framework for doing financial analysis in MDO. We applied the framework to the baseplate, a component used within the excavator pump, and optimized across three different disciplines of cost, natural frequency and temperature to find the baseplate design with the highest sales potential. We focused on sales as it is the most important financial metric for the product, but a similar framework can be used for maximizing profit, NPV, IRR or any other financial metric.

We used two approaches for finding the best design for the company. In the first approach, we found designs which minimized cost and temperature, while increasing the natural frequency. We then converted the cost and temperature data into sales and chose the design with most sales. In the second approach, we only set one objective of maximizing sales and chose the design with the highest sales. In both the approaches we were able to significantly increase sales. We would recommend approach 1 as we get higher sales with the method, and because of limitations within the optimization software OptiSLang in regards to implementing approach 2. Approach 2 might become the better option in the coming years as MDO software, including OptiSLang, is in the early stage and might significantly improve. Approach 2 also has the advantage of MDO teams only setting one objective, helping establish consistency and uniformity in MDO implementation.

We believe MDO has a lot of potential. Similar to CAD, it is an extremely powerful tool. Some of the challenges to successful implementation were: computational resources, high quality and reliable financial data and early stage MDO software. Organizations which implement MDO will create better products which maximize savings and financial benefit.

Thesis Supervisor: Maria Yang

Title: Professor of Mechanical Engineering

Table of Contents

Acknowledgements

The MIT-Danfoss MDO project has been one of my best experiences. The project was a true hallmark to industry-academia collabration. I was lucky to be part of this extremely talented, fast-moving team. We had amazing advisors in Maria Yang (Professor of Mechanical Engineering), Mark Shu (Innovation Director at Danfoss), and Jose Pacheco (Co-director of the MEngM program) who mentored us through the process. We also could not have done it without support from Danfoss engineers Bennett Torrance, Neil George and Bala Chakkaravarthy who gave us so much of their time and expertise in making the project succesful. We are grateful to Dr. Michael Schimmelpfennig (Ansys) for supporting us on the project and to Professor David Hardt for offering us computational resources for the project.

Kevin Marty (Danfoss PGP) was our manager and did an excellent job in leading the project. Teammates Antoine Yazbeck and David Mimery have become my most memorable friends at MIT and we have learnt so much from each other. It is great to have diverse team members from many different nationalities! The project started in the second half of Spring 2020, when Coronavirus (Covid-19) was spreading rapidly thoughout the country and the world. It was invaluable to have team members who would support each other during pandemic. We had many memorable zoom calls and some fun socially distanced meetings (within Cambridge guidelines).

Finally, I would not be here without my family. My parents did pay for my tution to MIT. I am grateful for Bharat and Manita Mardia for releasing me into the world. My grandparents Suresh and Sajjan Bafna for taking care of me when I was young. My “iron lady” grandmother Vimla Mardia for preparing me for life. My sister Swati Mardia for always overacheving. My nephew Rishi Bafna for being the light in the journey. Adit Bafna for being the perfectionist, and hope the thesis inspires Suvrat Bafna and Rahil Singhi to do great things in college and beyond.

Chapter 1: Introduction

Multidisciplinary Design Optimization (MDO) is a field of engineering in which we find the global optimum across multiple disciplines, ranging from structures to economics. We can use MDO to design and customize almost any product from an aircraft wing to a tennis racket. For instance, we can choose the range of the length of the tennis racket, desired limit on the weight, the strength requirement and can use MDO to create a tennis racket which is optimized for our game.

Initially developed by NASA, MDO has been around for 50 years (Dunbar, 2020). The test cases have largely been used within the Aerospace industry. With decreasing computing costs, MDO is now becoming increasingly viable for mainstream applications and many companies are looking to incorporate MDO into their design process. In the project, Danfoss partnered with MIT to test MDO on their industrial components and use the project as a benchmark for expanding MDO throughout the organization.

In MDO, we are trying to optimize many different disciplines and variables. It is easy to lose sight of the final objective. Individuals and users also have biases which makes them prioritize one discipline or variable over another. For instance, a thermal engineer might be singularly focused on minimizing temperature whereas the finance team might be more interested in cost. The allure of MDO is that it helps us choose the optimum design across multiple disciplines. However, it is important to have one key metric which teams and organizations can use to choose the best design. Almost all companies in the world are interested in maximizing their sales, revenues and profitability. The importance of financial metrics in decision-making allows us to use them in MDO for choosing the best design for the company.

In this thesis, we created a framework for doing financial analysis in MDO. We created our own internal cost simulation so that MDO would efficiently calculate the cost of each design point. Since sales is the most important financial metric for the product, we created a sales model in which MDO would identify the design with the most amount of sales potential. A similar method can be used for creating a model which identifies the design with maximum profit, NPV, IRR or any other financial metric.

Some of the challenges we faced while implementing MDO were computational limitations, early stage MDO software and availability of models of sufficient fidelity. We were able to overcome all of them, including the disruption caused by the pandemic, and successfully execute the project. We created a financial template which can be used as a reference for scaling MDO throughout the organization. We believe that MDO has the potential to become an important design tool and will add significant value and financial benefit to organizations using them in their design process.

Chapter 2: Background

History about Danfoss

Danfoss is a 70 year old automotive and energy company founded by Mads Clausen in 1933. From his parent’s farm in Nordborg, Denmark, the company has now expanded to a thirty thousand employee global company (Danfoss, 2020). Danfoss operates in four major segments: Power Solutions, Cooling Solutions, Vehicles, and Heating Solutions. Danfoss is a leading supplier for energy solutions around the world. Its products help in solving important problems from urbanization to global food supply.

For the project we worked on power solutions and focused on the excavator market. Danfoss makes pumps which are used by excavator companies such as Komatsu and Caterpillar. The component used for MDO is the baseplate of the Digital Power Controller (DPC) for the pump. The DPC baseplate protects the PCB and is mounted in the excavator. In Figure 1 is the company logo and in Figure 2 is an excavator with Danfoss products.

Figure 1: Danfoss Logo (Danfoss, 2020)

Figure 2: Excavator with Danfoss products (Danfoss, 2020)

Danfoss Innovation Accelerator

Our project was sponsored by Danfoss' Innovation Accelerator in Cambridge, Massachusetts. The office is a key innovation hub for Danfoss, and plays a crucial role in identifying and implementing external technologies. Located near MIT in the CIC building (Figure 3), the office is at the center of one of America's most innovative ecosystems. The team responsible for DPC baseplates is located in Danfoss' manufacturing facility in Ames, Iowa. The broader pump team is located in Scotland, UK.

MDO project was the winner of the company-wide competition for new moonshot project ideas. The winning team was led by Danfoss engineers Jens Paulik and Neil George. Danfoss is currently in the early stage of MDO implementation.

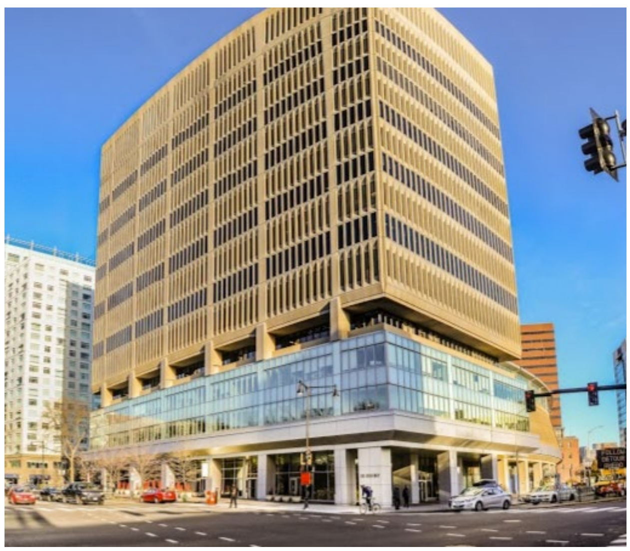


Figure 3: CIC building in Cambridge (CIC, 2020)

<u>Product used for MDO case study</u>

The baseplate covers the PCB and is mounted on the excavator (Figure 4 and Figure 5). The PCB controls the pump. The goal of the project is to reduce the maximum temperature and cost while improving the natural frequency. Heat on the PCB is an issue and we also need high frequency to avoid resonance with other components. Danfoss engineers have added fins to the product to increase heat transfer and reduce the maximum temperature. In the project, we added more fins, changed fin characteristics like fin thickness and height, and increased the baseplate thickness. The product was ideal as we were able to demonstrate the value of MDO within the project timeline. Other products were not suitable for MDO as there was no clear benefit in optimizing them across different disciplines or had significant engineering complexity such that we could not successfully model them with the available time and computational resources. Also, an additional plus was that we were able to work with the Danfoss plant in Ames, which was in a similar time zone.

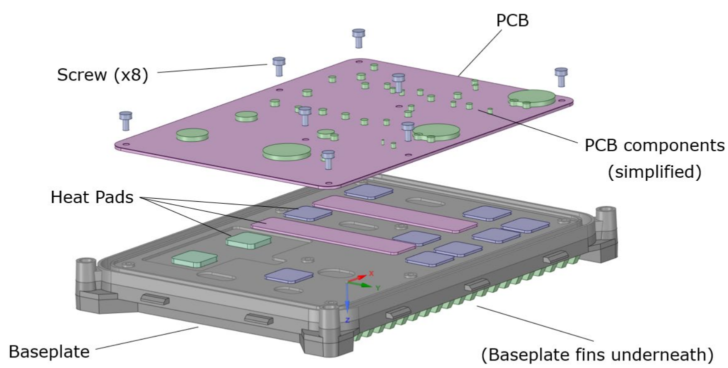


Figure 4: CAD of the baseplate with the PCB

Figure 5: Actual baseplate

Chapter 3: Research Methods

Research methods is an overview of the tools and processes that helped us successfully implement MDO. In the first half, we explore the framework for implementing MDO, process for choosing the MDO software and the computational resources needed for MDO. In the second half, we look at the incorporation of Agile into the project and the two different workflows used for baseplate optimization.

Division of the project

The project was a team effort with fellow MIT graduate students Antoine Yazbeck and David Mimery and our manager Kevin Marty from Danfoss. Our advisors Maria Yang (Professor of Mechanical Engineering), Mark Shu (Innovation Director at Danfoss) and Jose Pacheco (Co-director of the MEngM program) mentored us through the project. We worked with many stakeholders across MIT, Danfoss and other organizations (see Acknowledgements, pg. 5, for more details).

As mentioned in the previous page, the goal of the project was to demonstrate the value of MDO by optimizing the baseplate design. In order to find the optimum design we had to maximize the natural frequency and minimize the temperature and cost. To do this we split the project into the three disciplines based on our past experience and interests. Antoine created the Modal Model, David created the Thermal Model and I created the Financial Model. All of us worked together on background research and on integrating the different models and implementing MDO. We helped each other on our models and worked as one team. References to Antoine's and David's thesis can be found in the appendix.

Framework for implementing MDO

MDO can be used in several different fields ranging from structures to economics. The first step is to choose the variables, set the objectives and define the constraints. In the second step we choose the fidelity of the project. Choosing the correct amount of fidelity is essential to the success of the project and involves a balance between robustness and process physics. Low amount of fidelity might undermine the findings of the optimization as the simulation would not accurately model the system, and a high amount of fidelity would result in inefficient optimization runs. The final step is choosing the architecture of the optimization. For the literature review we have explored Multidisciplinary Feasible and Collaborative Optimization (Balesdent et al., 2010).

Figure 6: SpaceX Launch (SpaceX, 2020)

In the example, we are using MDO for launch vehicle design. We would first identify the different disciplines relevant for MDO. This could be aerodynamics, propulsion, structures, cost and trajectory optimization. Then we can set the objective, which for this case would be maximization of the payload mass, minimization of gross lift-off weight, and minimization of launch vehicle cost. The design variables would be masses, diameters, and propulsion variables like chamber pressure and mixture ratio. The coupling variable would be dry mass, specific impulse and length to the diameter. For the equality constraints we can specify the desired orbit and payload mass needed for the mission. In the inequality constraint we can define the maximum load factor and minimum nozzle exit pressure. The framework is summarized in Figure 7.

- Decomposition into different disciplines : Aerodynamics, Propulsion, Structure, Cost,…Trajectory Optimization
- Objective Function: Maximization of payload mass, minimization of gross lift-off weight, minimization of launch vehicle cost
- Design Variables : Masses, Diameters, Propulsion Variables (chamber pressure, mixture ratio)
- Coupling Variable : Dry Mass, Specific Impulse, Length to Dia.
- Equality Constraints : Specifications of the mission such as desired orbit and payload mass
- Inequality Constraints : Maximum Load Factor, Minimum Nozzle Exit Pressure

Figure 7: MDO Framework used for launch vehicle design

For the architecture we could do the Multidisciplinary Feasible (MDF) method. As shown in Figure 8, MDF involves analyses at the sub-system level. The sub-system would refer to the specific discipline or component. Some of the advantages are that the system is easy to implement, there are a limited number of variables, and solutions are available even when the optimization is stopped. Some of the disadvantages are that this architecture does not take advantage of coupling between disciplines. There is a high calculation cost and the framework can only be applied to simple solutions. There also needs to be clear communication and transparent management.

The other approach is the Collaborative Optimization (CO) method. This is a two-level optimization and over here there is greater freedom for sub-systems. In addition to global parameters, we have local variables, constraints and optimizers. Some of the advantages are that the system has high modularity and the sub-systems are easy to modify. We can have optimization methods adapted to each sub-system. The disadvantages are that the system is not robust and the computational efficiency decreases with the number of coupling variables.

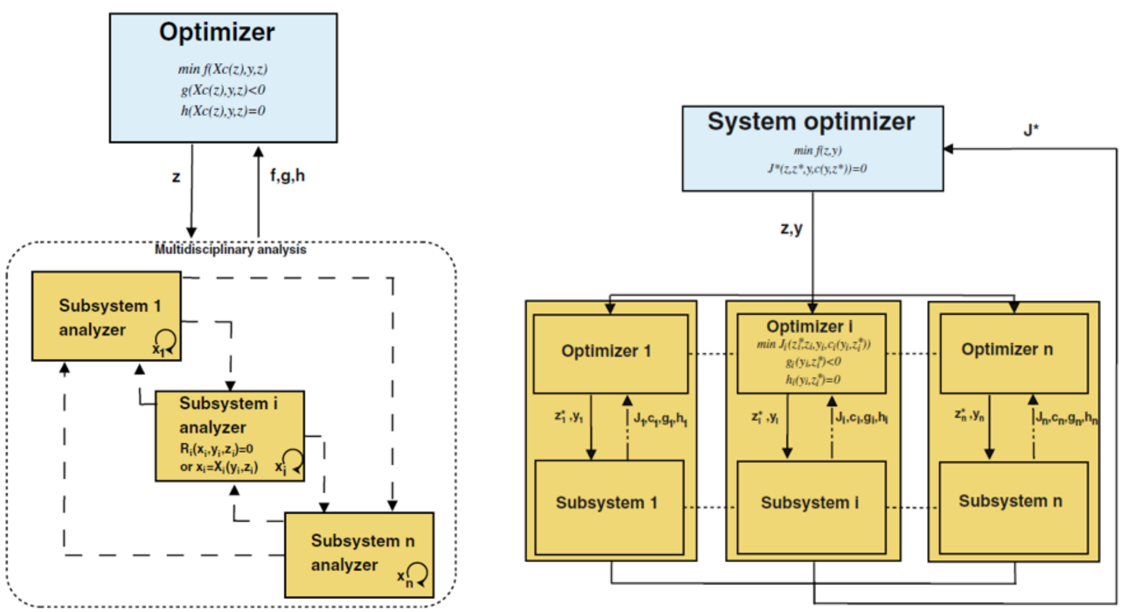


Figure 8: Multidisciplinary Feasible (Left) and Collaborative Optimization (Right) (Balesdent et al., 2010)

MDO Software

		Avg score			
Key weighting factors for Danfoss	**Criteria**	**OptiSlang**	**Heeds**	**3DS**	**MSC**
5	Price	1.0	3.0	3.0	1.0
7	Compatibility	3.0	2.5	2.5	3.0
8	Flexibility	3.0	1.0	2.0	3.0
6	Documentation	3.0	3.0	3.0	3.0
4	How widely adopted	3.0	1.5	3.0	2.0
8	Areas/Application specialities	3.0	1.5	2.0	3.0
5	Efficiency of computation	2.0	2.0	2.0	3.0
	WEIGHTED SCORE AVERAGE	2.65	2.01	2.43	2.67
	RANKING	2	4	3	1

Table 1: Matrix for evaluating different MDO software

We evaluated different MDO software used within the industry (Table 1). Ansys-owned OptiSLang was Danfoss' preferred software as it was already tested within the company. iSight and Simulia are part of Dassault Systemes'(3DS) product offering and are available on MIT's IS&T system. Heeds is owned by the industrial giant Siemens. MSC's Nastran was one of the first MDO software used within the industry. Credit goes to NASA's office of technology utilization for launching Nastran. Nastran was later acquired by MSC and is independently run as part of MSC's ecosystem (MSC, 2020). Having a strong early mover advantage, MSC has established itself as the leader in the industry. About 3000 companies in the aviation, automotive and higher education industry use Nastran (Enlyft, 2020). 1286 companies are part of the MSC One ecosystem which allows users to link Nastran with other simulation and optimization software. Given that Nastran is established in the industry, it is compatible with most commercial software. The costs are high, and the Nastran bundle can be in $20-45k range (Wong, 2010).

At the second place was OptiSLang. We used OptiSLang for the project as the software is already established within Danfoss. OptiSLang was a German startup company, Dynardo, which was recently acquired by Ansys. Combining forces with Ansys has given OptiSLang several advantages. Ansys is widely used for simulations by 11,000 companies (Enlyft, 2020). Ansys offers customer and technical support in 40 countries around the world (Ansys, 2020). Ansys is also expensive starting at $30k (Lavi, 2020). We luckily did not need to pay for using Ansys and OptiSLang as the thesis is sponsored by Danfoss.

At the third and fourth position are 3DS' Simulia and iSight and Siemens' Heeds software. Owned by some of world's well-known industrial companies, the software are rated poorly in our study not because they are terrible but because they are not the right fit for Danfoss. Both the MDO software are aimed towards the mass market and would not be suitable for complex simulations which Danfoss' products would require. Both the choices, however, are excellent for companies using MDO on simple products. Companies already using Dassault CAD products like SolidWorks and CATIA can easily add on Simulia and iSight to their simulations. Simulia and iSight are compatible with standard software such as MATLAB and Excel (Dassault Systemes, 2020). Heeds is the most affordable MDO software at only 500$ per month (Siemens, 2020). It is focused on being user-friendly and would be a great choice for individuals and companies new to MDO.

The team received access to OptiSLang thanks to sponsorship from Danfoss. MDO software can be extremely expensive and this might be a limiting factor for many companies. We had a direct line with OptiSLang's support team in Germany and access to commercial products, which are much more comprehensive than the student edition of Ansys and OptiSLang. We really enjoyed working with OptiSLang throughout the project. However, as mentioned in the feedback section there is scope for improvement and we recommend Danfoss to try MSC's Nastran and compare it with OptiSLang.

Computational resources

One of the most important ingredients for success in MDO is access to powerful computational tools. We can really attest to this. Throughout the project we were hobbled by lack of computational resources. We were innovative about our limitation and able to carry forward the project thanks to access to a powerful PC in Professor Hardt's lab.

There are two ways of running the powerful optimizations needed for MDO. The first step would be to invest in hardware and the second step would be to connect the PC with high performance computing resources. The first approach would involve buying a $5,000 PC. It is recommended that the PC has intel Xeon Gold with 16 cores, 192 Gb memory, 2 TB storge and Windows 10 operating system (Ansys, Calculate the Simulation Speed-up and ROI of a New Workstation, 2020). Depending on the application, we might also need access to a graphic card. For instance, running Ansys Polyflow would require Nvidia's Tesla or Quadro series. The investment seems significant but is essential to running MDO. As shown in Figure 9, the hardware upgrade would pay for itself in less than 3 months for a typical design engineer. The savings would be due to the reduction of computation time, allowing engineers to do more work.

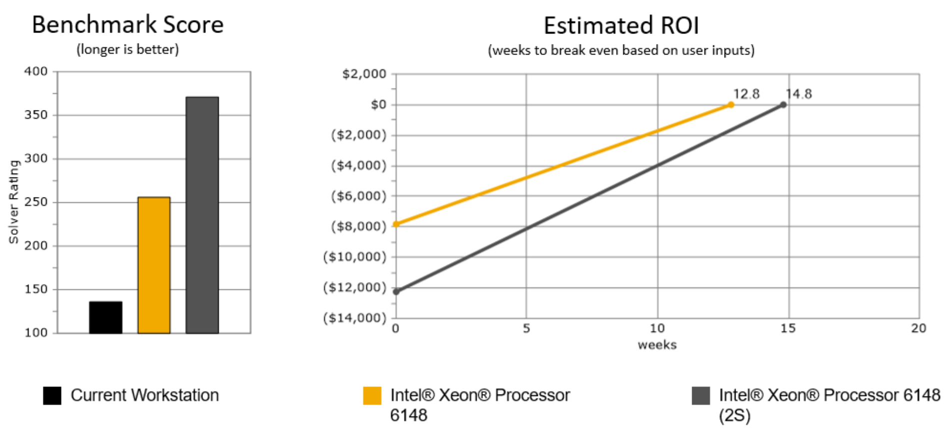


Figure 9: Plots showing the benefit of investing in hardware resources (Ansys, 2020)

The other approach would be to connect a simple laptop such as an 850$ HP G5 Zbook mobile workstation with HPC resources. They can range from Amazon's AWS to Ansys' cloud system. Renting cores might be an effective way for rapidly decreasing the computation time. For instance, if a "CFD simulation takes 16 days on a single core, then adding 32 cores might reduce the simulation time to only four hours! Doubling the number of cores from 32 to 64 might cut the time in half again – from four hours to two hours" (Ansys, Ansys Cloud, 2020). Adding the number of cores saves a lot of time, but the value per core as shown in Figure 10 decreases after a certain point. With parallel computing, we can run multiple jobs and thus ROI increases in a linear fashion with the number of cores. With HPC parametric, HPC customized for design exploration, we can simultaneously run multiple design points.

MDO is in the early stage of mainstream adoption and the resources supporting it are expanding. Computing has been one of the biggest limiting factors for MDO. The rapid lowering of computational costs from $160 billion per gigaflop in 1960 to $0.03 today (Figure 11) is helping MDO expand beyond the small enclave of NASA and Aerospace companies (Wikimedia Foundation, 2020). As mentioned in the MDO software section, all major design software companies are beginning to offer MDO in their portfolios. They are also offering innovative HPC solutions to support MDO. For OptiSLang, we would recommend using Ansys' HPC parametric solution as it would allow us to compute multiple design points in a short period of time.

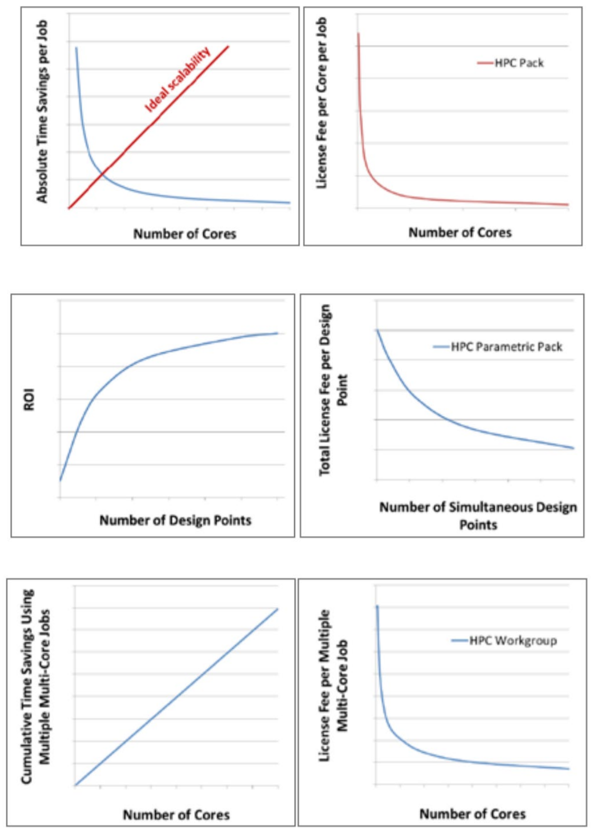


Figure 10: Improvement in performance with more number of cores and design points (Ansys, 2020)

Cost of gigaflop in respective years (adjusted):

- 1960 : $160 billion
- 2000 : $1510
- 2011 : $2.07
- 2017 : $0.03

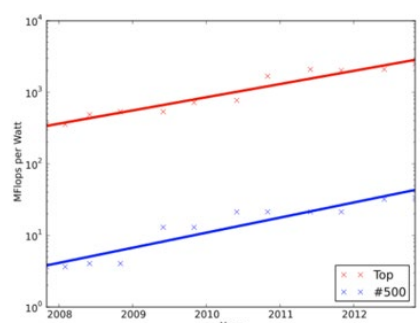


Figure 11: Reduction in cost and improvement in energy efficiency (Wikimedia Foundation, 2020)

Being in MIT, we had access to plethora of computational resources. We connected with MIT Lincoln Laboratory's Supercloud system. However, we soon realized that OptiSLang was not on the network and installing the software would take longer than the project timeline. We then contacted Ansys to access the company's HPC offering, Ansys Cloud, but installing SolidWorks into the server for Ansys Cloud was not viable for a small project. We were also not able to access OptiSLang on Danfoss' servers as access was restricted to employees within the company. We tried purchasing computation from third-party HPC suppliers like Nimbix at a very reasonable price of $1,500 for 10,000 core-hrs. However, Nimbix also did not have OptiSLang installed on its servers and installing it would take about 2 months.

Due to various roadblocks with access to HPC we decided to revert back to option 1. We used our laptops for limited runs where we tested the minimum viable optimization, and we ran the complete run on Professor Hardt's powerful lab PC. Due to Coronavirus, we could not physically log in to the computer and we used TeamViewer to remotely connect with the PC. Due to limitations on computing, as a team we had to be efficient in using the lab PC and we collectively agreed to do only the optimization runs which would add value to the project. We also adjusted the project scope and timeline to minimize the number of optimization runs. Through the project, we learnt that it is critical to have computational resources and to be smart about which optimizations to run. The project setup is shown below in Figure 12.

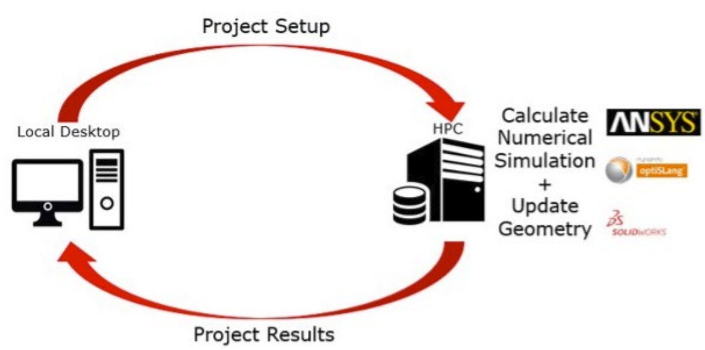


Figure 12: Project setup for MDO

Implementation of Agile into the project

We used Agile to structure our project. Agile has been widely used in software companies in Silicon Valley, and is now moving into other industries. More information on Agile can be found in the appendix. The project is split into iterations and each iteration is reviewed and tested. The advantage of Agile is that early feedback prevents failure in the end when it is too late and allows teams to pivot the project. Software is ideal for Agile as iterations are very quick and do not require significant capital investments. From our experience, we believe Agile can also be successful for hardware products. Simulating real-world conditions and getting feedback on the results from the simulations can be extremely valuable to the product development cycle. All of us have taken the course Product Design and Development at MIT under our advisor Professor Yang, where we used Agile for creating hardware products.

We split the project into three iterations (Figure 13). In each iteration we would add more parameters and complexity. We initially created a simple product on the Ansys workbench and conducted the optimization. In the first iteration, I created a completely theoretical cost model as I was still working with Danfoss on collecting cost information. In the second run we added more thickness and other parameters. I learnt from the first iteration that it is very important to have a simple interface with other simulations in the Ansys workbench, so that other team members can seamlessly connect their simulations with the cost model. In the second iteration, the cost model simulation would automatically link with other simulations, and the parameters in the cost model would automatically link with parameters in the global parameter set. In the third simulation, I finished collecting the relevant cost information from teams within Danfoss and also developed an understanding of their business. I updated the cost model with actual data, and successfully received results in the third iteration. I personally think implementation of MDO with Agile was effective and would recommend MDO teams to consider using Agile workflow.

	Iteration 1	**Iteration 2**	**Iteration 3**
Simulation Model	Low-fidelity model	High-fidelity model	High-fidelity model
CAD Model	Low-fidelity simulation	Medium-fidelity simulation	High-fidelity simulation
Design Variables	3 design variables	3 design variables	11 design variables

Figure 13: The three iterations used for the project

Workflow for optimization

Since the CAD file for the baseplate is in SolidWorks, there are two ways to connect the CAD with Ansys. The two different workflows are shown in Figure 14. In the first method, the file can be imported into SpaceClaim, Ansys' CAD editing software. In the second method, the SolildWorks file can be directly connected with Ansys. In an ideal world, parameters and information from SolidWorks should directly transfer into Ansys. However, as mentioned earlier, Ansys only recently acquired OptiSLang and there is no seamless interface and information transfer yet between the CAD, Ansys and OptiSLang. The parameters from SolidWorks would need to be redefined in SpaceClaim. Running SolidWorks directly also takes more time, and this makes a significant impact when we are running multiple design points. In the first half of the project, we were not able to use OptiSLang to run multiple design points on SolidWorks. However, thanks to additional support from the Ansys team in Germany we were able to fix the problem and run optimizations on the SolidWorks file.

The other approach, which avoids a lot of the issues mentioned in the first approach, is to directly download the CAD file to SpaceClaim and create new parameters in SpaceClaim. The downside with this approach is that we lose fidelity when we use SpaceClaim. SolidWorks is a far superior designing software, and creating complex shapes and modifications is extremely difficult in SpaceClaim. We created simplified geometry in SpaceClaim and the results from the simulation were surprisingly close to the earlier approach.

Our conclusion was that both methods are equally good but both have tradeoffs. With SolidWorks we get high fidelity results with more implementation and computation time, and with SpaceClaim we get lower fidelity results but faster implementation and computation time. Teams can also use a hybrid approach in which low fidelity, quick simulations are done in SpaceClaim and high fidelity, time consuming simulations are done in SolidWorks.

Figure 14: Workflow I and Workflow II

Chapter 4: Developing the Cost Model

For the project, we created our own cost model. Every time OptiSLang runs a design point, our cost simulation calculates the cost for that design point. We created the model based on data from aPriori, a cost simulation software used within Danfoss. The cost model only covers the cost of the baseplate and does not include the cost of the PCB. The cost consists of three different sub-components: the material cost, the process cost, and the supplier profit.

One of the objectives of our MDO project was to reduce cost for the baseplate. Cost is one of the key drivers for any organization and any business decision. The supplier quote for the baseplate is 16.4$. In MDO, we have to obtain the cost for different geometries. Getting actual supplier quotes for multiple different orientations we are testing in MDO would not be viable due to the project timeframe. In order to address the pricing for the baseplate, we partnered with Danfoss India to obtain cost information from aPriori. aPriori is a cost simulation software used throughout Danfoss to calculate cost of different CAD geometries. The assumptions the team made are shown below in Figure 15.

- **Production Location** : USA-North Central (IL-MI-MN-OH-WI)
- **Annual Volume** : 500 (Batch size considered 250 pcs)
- **Material** : Aluminum, Cast, ANSI AL380.0 ; Material Cost ~ 2.1 USD/Kg
- **Process Routing** : Melting / High Pressure Die Casting / Trim / 3 Axis Mill / Debur
- **Machined Surfaces :** Holes & Flat Profiles (Highlighted in yellow); Due to tolerance requirements from dwg

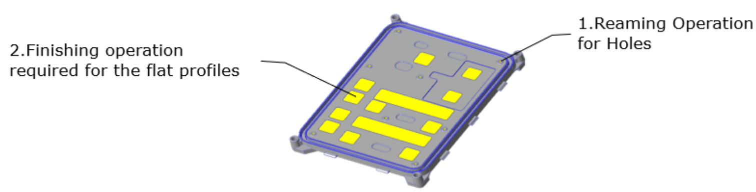


Figure 15: Assumptions used for calculating the baseplate cost in aPriori

Internal Cost Simulation & aPriori – Partnership with Danfoss India

We explored the parameters and parameter ranges shown in Figure 16 during the MDO process. In order to get the approximate cost, we looked at the comprehensive set of design points which covered the entire range. We used a base model as a reference point and changed the individual parameters. We had to balance between having enough points for a rigorous model and not too much to overwhelm the Danfoss India team with work. aPriori on average takes 15 minutes to calculate the cost for a design point. Increasing the tolerance and details can increase the time to 2 hours. Danfoss has already tried linking aPriori with Ansys and running simulations. We have tried a different approach by creating our own cost model. Even though directly using aPriori would be more accurate, creating our own model in Excel has several advantages as it is significantly faster. We save 15 minutes per iteration and thereby save a lot of computation time and cost, the key limitations in implementing MDO.

Parameter range	Design Points
Fin Thickness : 1 to 15	1 3 6 9 12 15
Fin Height : 10 to 18	10 12 14 16 18
Number of Fins : 2 to 40	2 6 10 14 18 22 26 30 34 38 40
Baseplate Height : 0.1 to 7	1 3 5 7
Baseplate Thickness : 0.1 to 12	2 4 6 8 10 12
Hor. Number of Fins : 0 to 7	1 3 5 7

Figure 16: aPriori data points used for creating the cost simulation. For fin thickness, we were interested in the range of 1 to 15 mm. We received aPriori cost estimates of the base model with a fin thickness of 1, 3, 6, 9, 12 and 15 mm.

Converting CAD files into cost information

We sent the step files to Danfoss India and received the cost information for all the design points. This is the estimated selling price of the supplier. We received the total cost and their breakdown in the three sub-components: material cost, process cost and supplier profit (Figure 17). Material cost is the cost for the raw material. Process cost includes the labor, direct overhead, setup, tooling, inventory and handling, packaging, indirect overhead and administrative expenses. The profit is the approximate margin for the supplier. In Figure 18, you can see the cost for different baseplate designs. Figure 19 is a sample step file sent to Danfoss India and Figure 20 is a snapshot of aPriori estimating the cost of a baseplate.

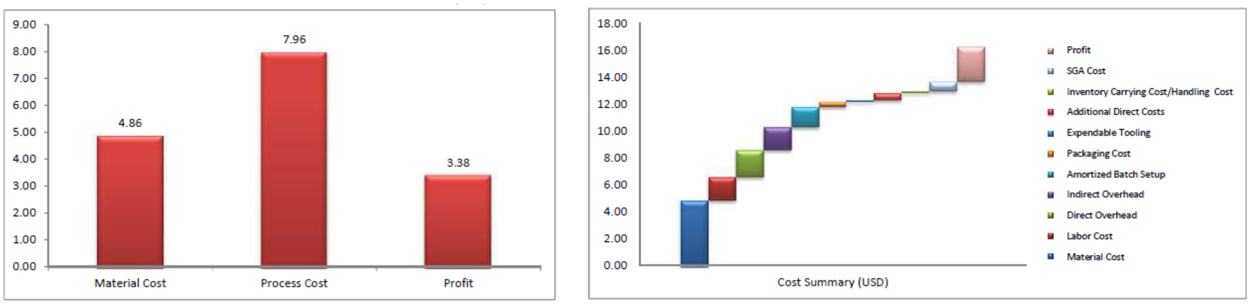


Figure 17: Breakdown of baseplate cost in aPriori

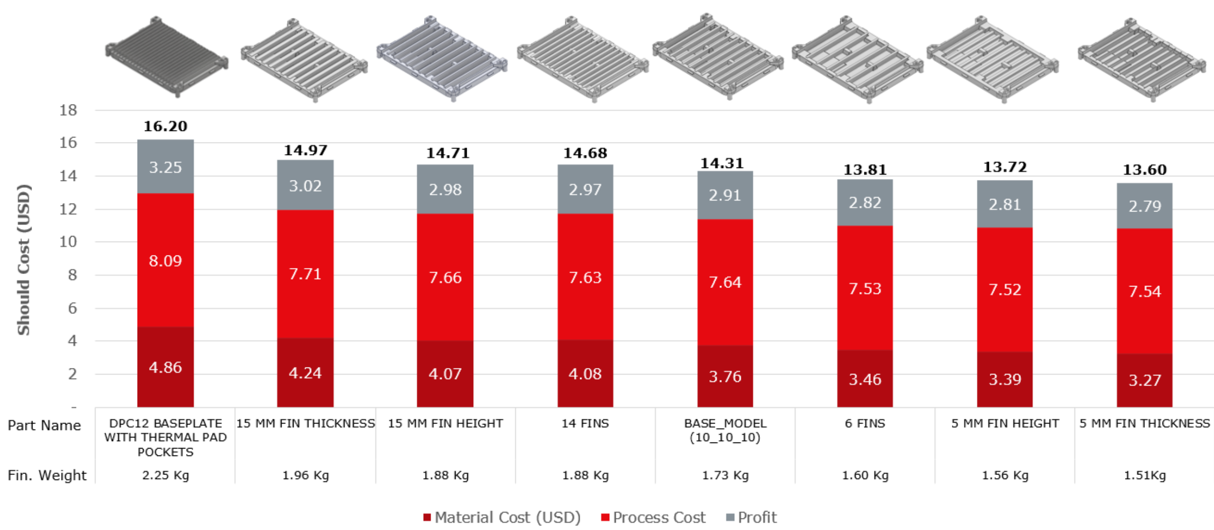


Figure 18: Cost for different baseplate designs. The cost for each design is broken down into cost for the raw material, process cost and supplier profit.

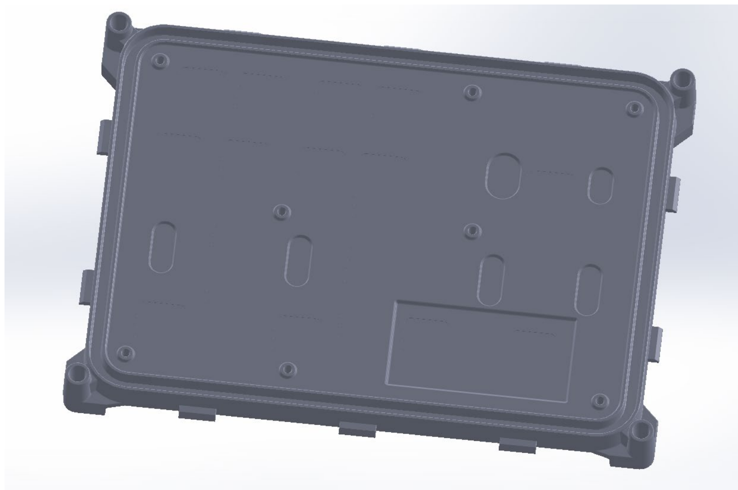


Figure 19: A step file sent to Danfoss India

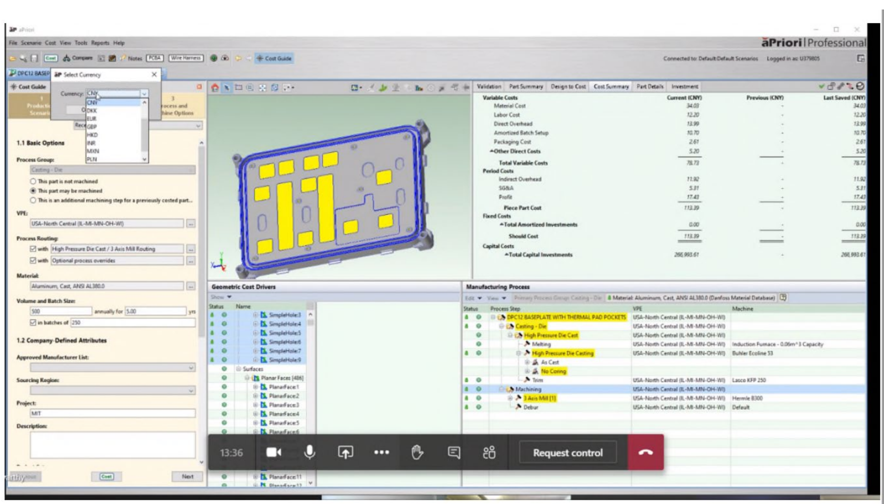


Figure 20: Snapshot of aPriori

Cost Model Concept

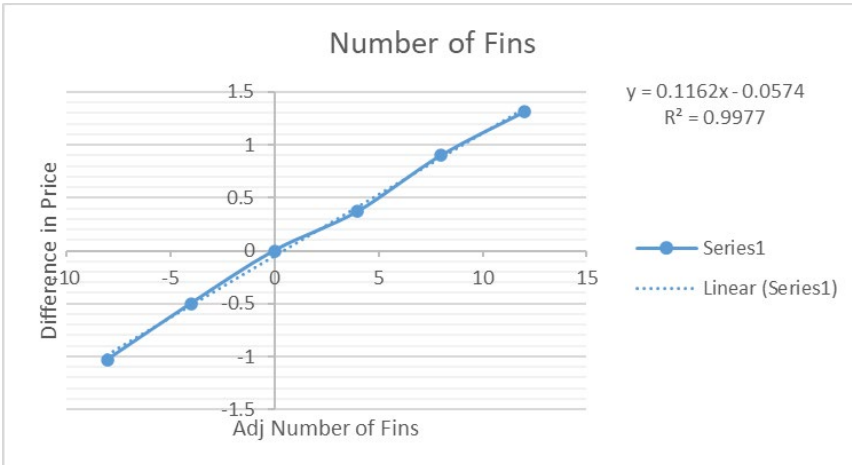


Figure 21: Plot showing the impact of number of fins on cost

Let us say we were trying to see the effect of the number of fins on the baseplate cost. We would find the cost of the baseplate with 2 fins, 4 fins, and continue in the increments of 4 to 40 fins. As shown in the plot (Figure 21), we can see the change in baseplate cost if we were to increase the number of fins of the base model of the baseplate, which has 10 fins. On the x-axis, since values are referenced to the base model, 5 shows the baseplate with 15 fins and -5 shows the base plate with 5 fins. As seen in the plot, the values in aPriori are linear, giving us confidence in our model. For almost all parameters we have very high coefficients of regression, and for only fin angle and horizontal fins we have slightly lower coefficients of regression (Table 2). The remaining plots can be seen in the appendix.

Dimension	Slope	Constant	R^2
Number of fins	0.12	-0.06	0.99
Fin Height	0.09	0.04	0.99
Fin Thickness	0.14	-0.04	0.99
Fin Angle	0.02	0.02	0.97
Baseplate Height	0.03	-0.01	1.00
Baseplate Thickness	0.35	-0.09	0.99
Hor. Fins	0.05	0.05	0.89
Hor. Fin Height	0.09	0.04	0.99
Hor. Fin Thickness	0.14	-0.04	0.99
Hor. Fin Angle	0.02	0.02	0.97

Table 2: Slope, constant and R^2 for different parameters

Usage of a ratio to work around the limited number of design points

Adding horizontal fins is not a focus for our project as they were not recommended by the design team at Danfoss. We still added them to the baseplate design to show the value of MDO and the potential benefits of adding horizontal fins in the future. Due to limitation on the number of design points we were receiving from Danfoss India as they were doing this out of courtesy, we made an important assumption for calculating the cost of horizontal fin parameters and baseplate with different materials. Using the data points for the number of horizontal fins, we calculated the slope of the number of horizontal fins. We then compared the slope for the number of horizontal fins (0.05) with the slope for the number of vertical fins (0.116), and found the ratio between the two to be 0.42. We used this ratio to calculate the cost of horizontal fin thickness and height, by multiplying the cost of the respective vertical parameters times the ratio. For instance, if increasing the vertical fin height by 1'' increased the cost by 1$, then the cost for increasing the horizontal fin height by 1'' would be $0.42.

We used a similar approach for baseplate material. The cost for the base model (Aluminum alloy A380) is $14.31. The cost for the base model with Aluminum alloy A360 is $14.65 and base model with A383 is $14.92. We used the ratio between the prices to calculate the approximate cost of the design with different materials. Let us say if the current design in OptiSLang with material A380 was $14.31, then the cost of the same design with material A383 would be $14.92.

Cost Model Simulation

To show how the cost model works, let us say we have a design in OptiSLang with all parameters equal to 10. We would then calculate the difference from the base model and use the slope from Table 2 (Cost Model Concept, pg. 25) to calculate the net price difference. Since this is a regression, even a difference of zero would have a negligible positive or negative value. As you can see in Table 3, for a fin angle of 84 degrees, the difference would be 4 and the relative price difference would be $0.12. We combine the price differences and find the cumulative price difference. Then we add the cumulative price difference to the price of the base model ($14.31), and we find the cost of the current design in OptiSLang. For the current orientation shown in the example below (Table 3), the net price difference is $1.028 and the price for the current model would be $15.34.

Dimension	Current Model	Base Model	Difference	Price Difference
Number of fins	15	10	5	0.524
Fin Height	14	10	4	0.390
Fin Thickness	11	10	1	0.095
Fin Angle	84	80	4	0.120
Baseplate Height	0	0	0	-0.010
Baseplate Thickness	3.3	3.3	0	-0.090
Hor. Fins	0	0	0	0.000
Hor. Fin Height	10	10	0	0.000
Hor. Fin Thickness	10	10	0	0.000
Hor. Fin Angle	10	80	-70	0.000

Table 3: Cost calculation for the current design point

Comparison of internal cost simulation with aPriori

We are adding the differences of individual parameters and adding them to find the cumulative price difference. However, the cost for changing multiple parameters might be different than changing the parameters individually and adding their difference. This can be due to economies of scale and different machining processes. In order to record the difference, we used a benchmark value. The parameters for the benchmark design point are shown below (Figure 22). The cost according to aPriori would be $16.91 and the cost according to the internal cost simulation would be $15.34.

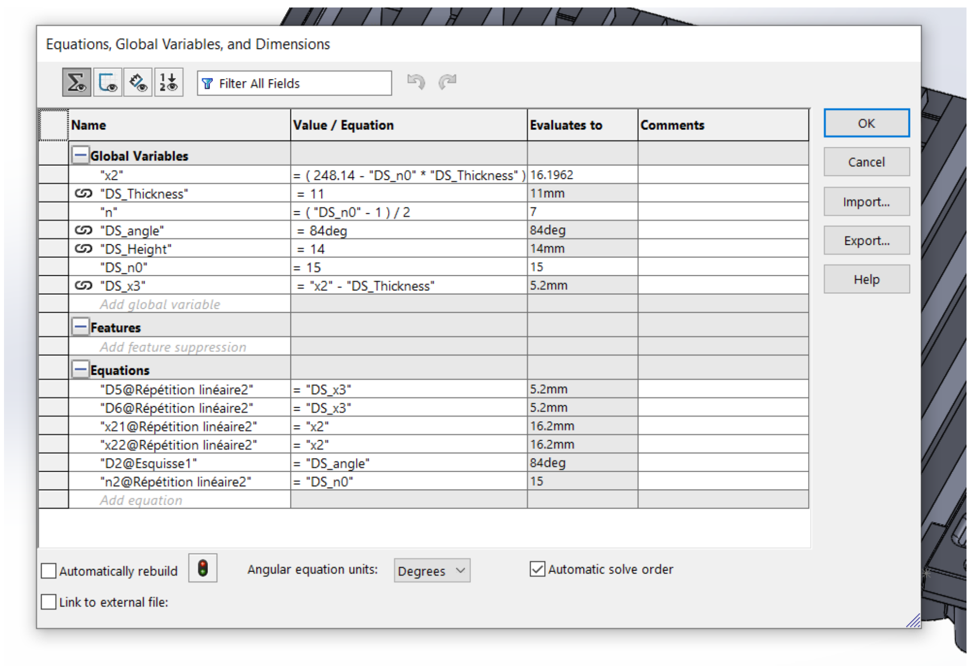


Figure 22: SolidWorks dialog box showing the parameters of the benchmark design

Comparison with supplier quote

The cost for manufacturing a single piece of the current baseplate in China is 114 Renminbi or $16.32. The quote is based on production volume of 20,000 pcs and is multiplied by a factor to avoid disclosing confidential Danfoss information. The irony is that the cost for manufacturing a part in China matches with the aPriori estimate for manufacturing the part in USA. It could be that aPriori significantly underestimates the cost of manufacturing or that Danfoss should look for alternate suppliers with lower costs. Obviously, this is beyond the scope of the thesis, and we are more concerned with finding a design with the lowest cost, irrespective of the supplier.

Supply chain of the baseplate

The baseplate is made in China by a supplier. The parts are then assembled with the PCB by an electronics contract manufacturer in Europe. The assembled baseplate is shipped to the Danfoss facility in Scotland, where it is combined with the pump. The pump is then shipped to the customers. Every dollar in savings in the baseplate translates to higher savings towards the end of the value chain. As mentioned in the sales model, it is important to take a holistic view as it might be beneficial to the company if we slightly increase the baseplate cost but sell to more number of customers. Since the baseplate is only one of the many components of the pump, and the profit from selling a pump far exceeds the cost of the baseplate.

Chapter 5: Connecting Cost Model with Ansys and OptiSLang

Linking cost model with Ansys

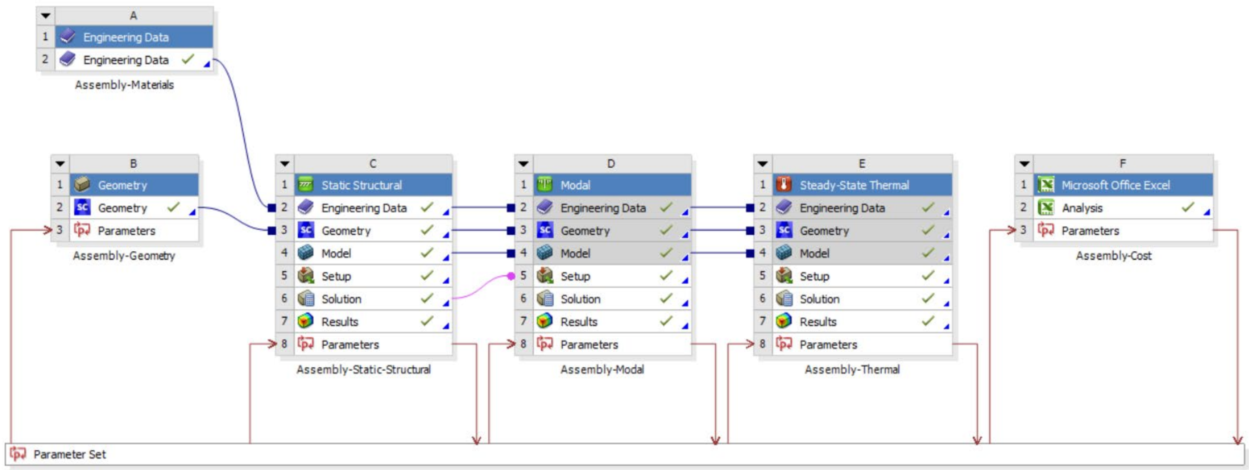


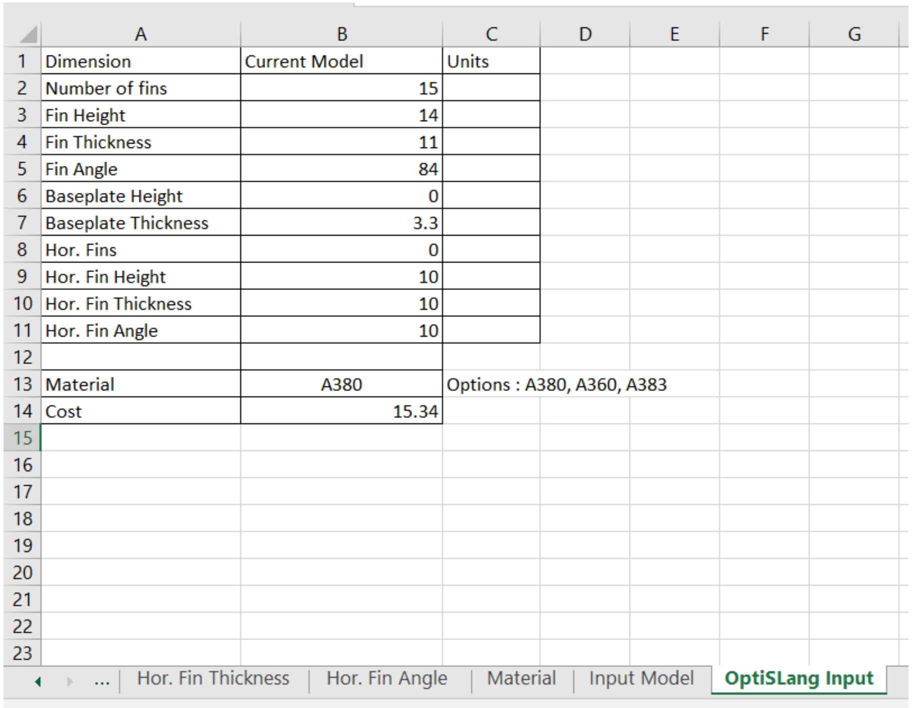
Figure 23: Ansys workflow

In the Ansys workflow (Figure 23), the CAD file is located in the Geometry box. Ansys simulates the external forces and conditions, and calculates the natural frequency and heat flow in the modal and thermal simulations. More information on the two simulations can be found in the thesis written by my teammates Antoine and David. The output information from the geometry, modal and thermal simulation goes into the parameter set and then goes into the cost model. As shown in the global parameter set (Figure 24), information about the geometry like the number of fins, fin thickness etc., the natural frequency, and maximum temperature are used as inputs for the cost model. As shown in the OptiSLang input tab (Figure 25), the information is transferred to the cost model. OptiSLang is very well integrated with Excel, and "define name" function can be used to create input and output relations (Figure 26). If a variable has a unit, then it is very important that the variable is defined in a consistent way along with the unit in order to align variables in the cost model with variables in the parameter set (Figure 27). Using the information in the OptiSLang input tab, the cost model calculates the cost for manufacturing the current baseplate design, and outputs the current price. This value is then sent to the global parameter set.

Outline of All Parameters

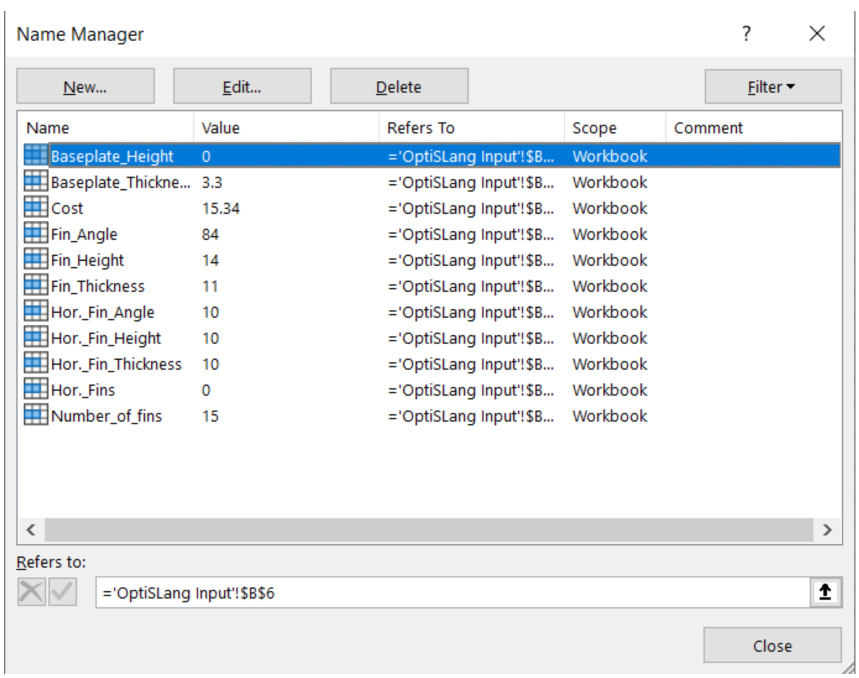
	A	B	C	D
1	ID	Parameter Name	Value	Unit
2	Input Parameters			
3	Geometry (B1)			
4	P11	fin_spacing	12	
5	P12	fin_thickness	6	
6	P13	fin_height	14	
7	Microsoft Office Excel (F1)			
8	P14	Fin_Thickness	6	
9	P15	Input_Volume	0.00069504	m^3
10	P16	Maximum_Temperature	48.577	C
11	P17	Natural_Frequency	574.57	Hz
12	P18	Number_of_fins	14	
*	New input parameter	New name	New expression	
14	Output Parameters			
15	Modal (D1)			
16	P3	Total Deformation Maximum	0.74496	m
17	P6	Total Deformation Reported Frequency	574.57	Hz
18	Steady-State Thermal (E1)			
19	P4	Temperature Maximum	48.577	C
20	P20	Geometry Volume	0.00069504	m^3
21	Microsoft Office Excel (F1)			
22	P19	Total_impact_due_to_the_new_design	3128.3	
*	New output parameter		New expression	
24	Charts			

Figure 24: The Global Parameter Set can be used to access and connect parameters of all models



	A	B	C
1	Dimension	Current Model	Units
2	Number of fins	15	
3	Fin Height	14	
4	Fin Thickness	11	
5	Fin Angle	84	
6	Baseplate Height	0	
7	Baseplate Thickness	3.3	
8	Hor. Fins	0	
9	Hor. Fin Height	10	
10	Hor. Fin Thickness	10	
11	Hor. Fin Angle	10	
12			
13	Material	A380	Options : A380, A360, A383
14	Cost	15.34	

Figure 25: OptiSLang input tab in the cost model



Name	Value	Refers To	Scope	Comment
Baseplate_Height	0	='OptiSLang Input'!$B...	Workbook	
Baseplate_Thickne...	3.3	='OptiSLang Input'!$B...	Workbook	
Cost	15.34	='OptiSLang Input'!$B...	Workbook	
Fin_Angle	84	='OptiSLang Input'!$B...	Workbook	
Fin_Height	14	='OptiSLang Input'!$B...	Workbook	
Fin_Thickness	11	='OptiSLang Input'!$B...	Workbook	
Hor._Fin_Angle	10	='OptiSLang Input'!$B...	Workbook	
Hor._Fin_Height	10	='OptiSLang Input'!$B...	Workbook	
Hor._Fin_Thickness	10	='OptiSLang Input'!$B...	Workbook	
Hor._Fins	0	='OptiSLang Input'!$B...	Workbook	
Number_of_fins	15	='OptiSLang Input'!$B...	Workbook	

Figure 26: Variables defined for the cost model as seen in the name manager

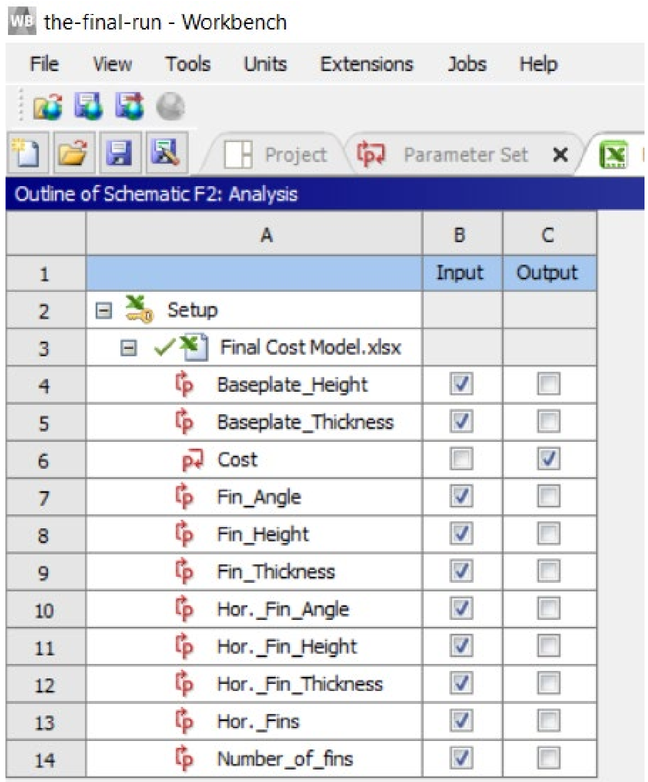


Figure 27: Edit configuration for the cost model toolbox can be used to define the relation between variables in the cost model and variables in the global parameter set.

Linking OptiSLang with Ansys

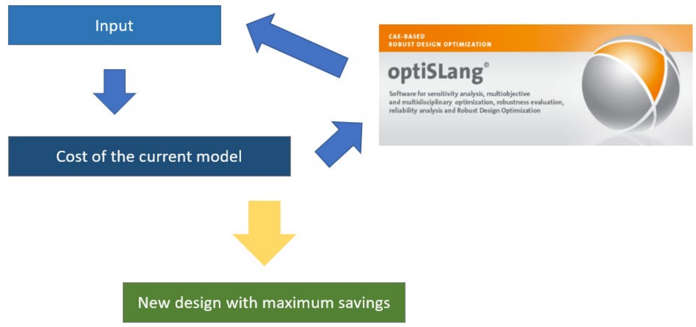


Figure 28: Feedback loop for the project

OptiSLang can be connected with Ansys in two different ways. OptiSLang can be used as a plug-in and can be connected with the parameter set (Figure 29). The second way would be to download the Ansys file into a standalone OptiSLang application (Figure 30). Our initial approach was to use OptiSLang as a plug-in as it is a convenient and faster option. However, we later switched to using the OptiSLang application as it offers more flexibility and is more robust. Our recommendation would be to use the OptiSLang plug-in for simple tasks and the OptiSLang application for more complex tasks.

As shown in the feedback loop (Figure 28), information about the current model goes into the input of the cost model. The cost model calculates the price of the current baseplate design and sends it to OptiSLang. Based on the objectives and constraints, OptiSLang evaluates the current design points and choses the next one. OptiSLang continues iterating until it finds the best design point.

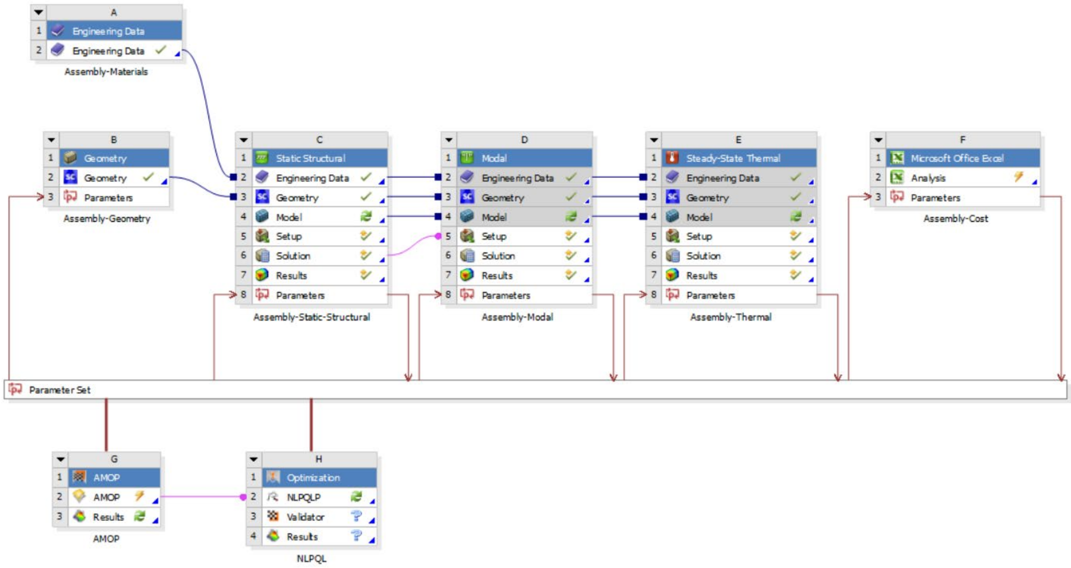


Figure 29: OptiSLang plug-in

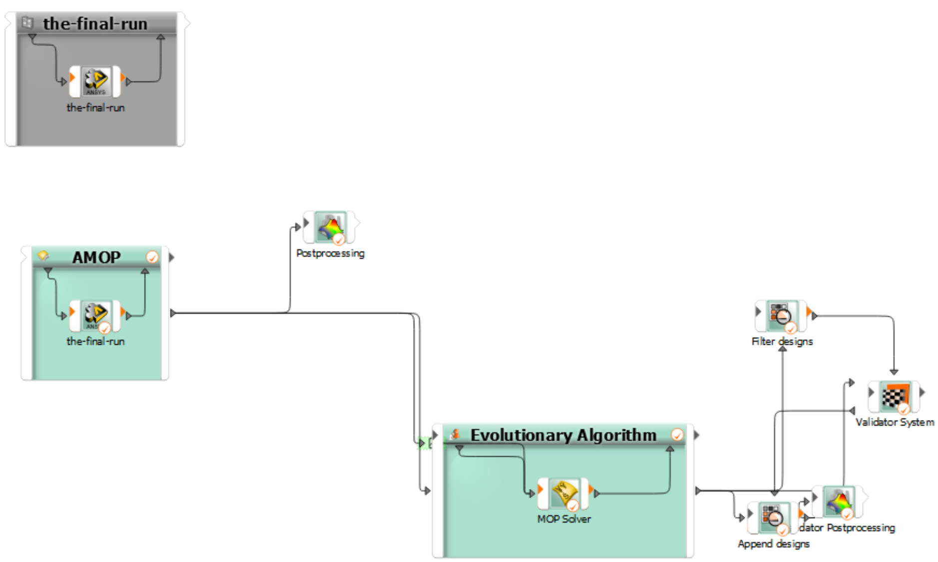


Figure 30: OptiSLang application

Choosing the algorithm

Different algorithms are used by OptiSLang to find the optimum design point (Figure 31). OptiSLang uses green color to recommend the optimization software. Users also have the flexibility of using algorithms in yellow, even though they are not preferred by OptiSLang for the current task. Algorithms in red, however, cannot be used for the task. For our optimization, we used AMOP for the sensitivity analysis and Evolutionary Algorithm for optimization. More information about this can be found in the appendix.

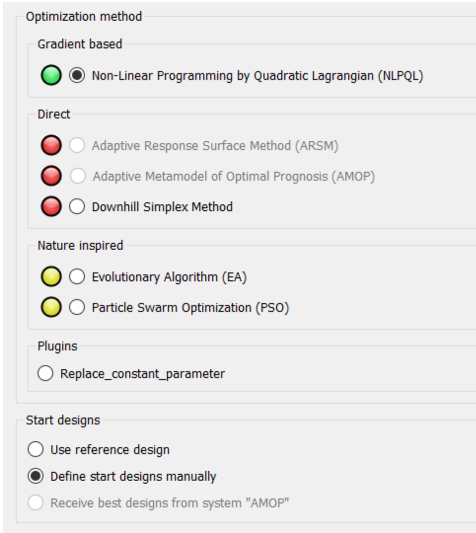


Figure 31: OptiSLang recommending the optimization method

Setting up OptiSLang!

The first step would be to complete all the simulations in the Ansys workbench. Next, import Ansys workbench into OptiSLang. Create a sensitivity analysis. Choose the parameters you are interested in and define the relevant ranges (Figure 32). The range can be continuous, discrete, constant, and can also involve functions and dependencies. Choose the input and output responses you are interested in (Figure 33). Set target COP value and maximum number of design points. OptiSLang will stop when it reaches the desired COP or when it reaches the maximum number of design points. Once the sensitivity analysis is complete, the data is then sent to optimization. With results from the sensitivity analysis, we can see the impact of the inputs on the output response. As shown in the plot (Figure 34), we can see how the cost varies with different fin spacing and thickness. In the COP Matrix (Figure 35), we can see the quantitative impact of different inputs on the output parameters. The highest contributing factor to cost is the number of fins (indirectly indexed through fin spacing) at 51%, followed by fin thickness at 34%. Here we can also see that Ansys' correlation coefficient, COP, is very high at 97%. Indicating that we can successfully proceed to optimization.

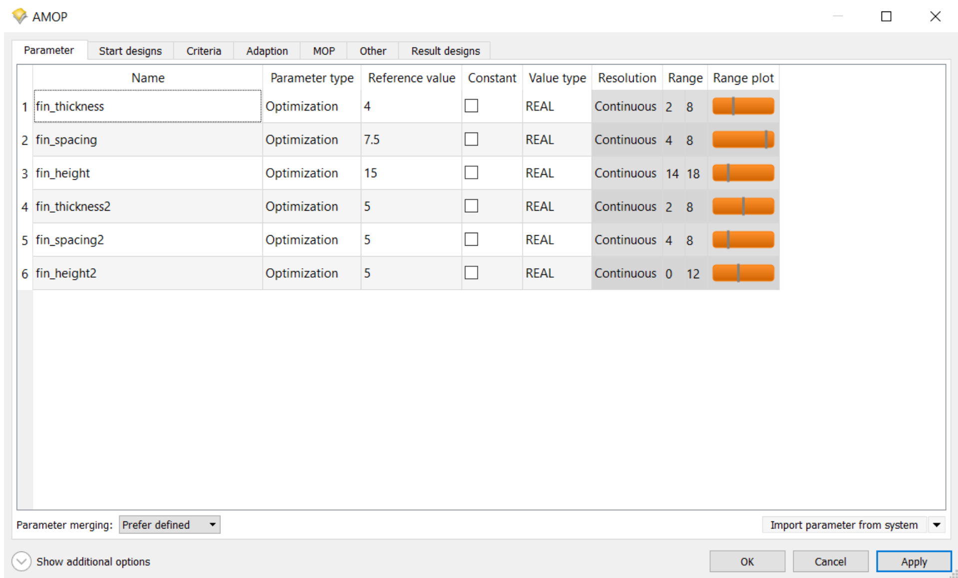


Figure 32: Selecting the variables for optimization

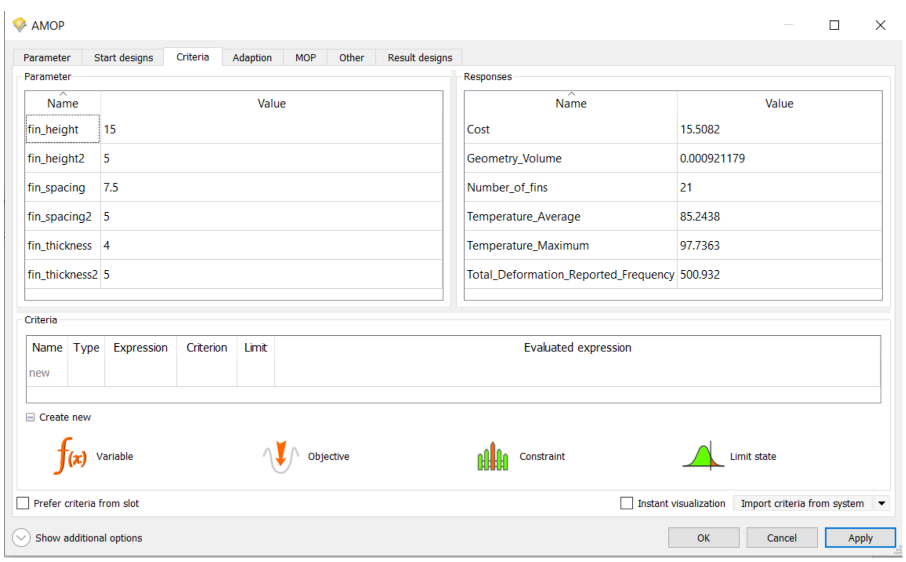


Figure 33: Choosing the parameters and responses for sensitivity analysis

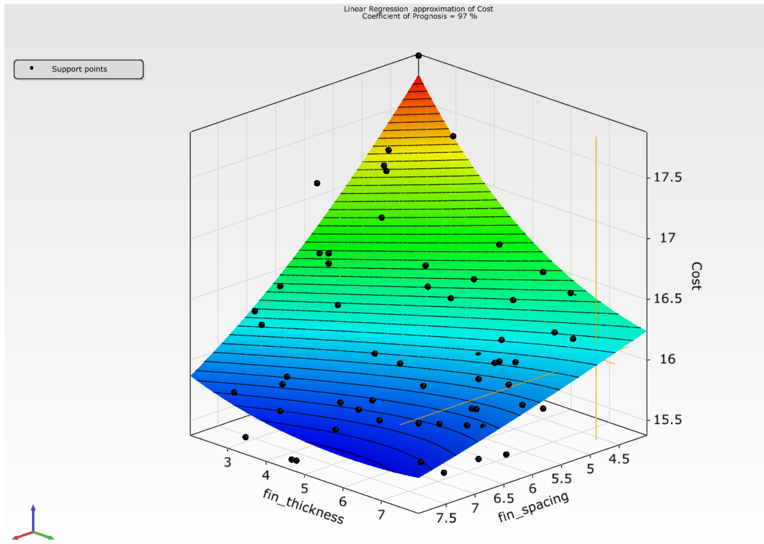


Figure 34: A sensitivity analysis showing the impact of fin thickness and fin spacing on cost

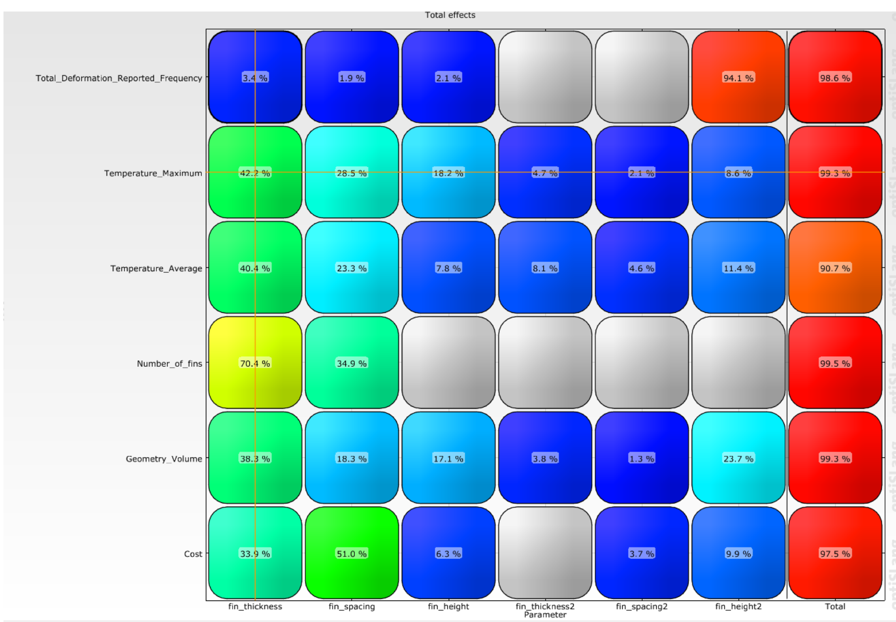


Figure 35: COP Matrix shows the impact of input parameters on the output. Here we see that OptiSLang is able to predict cost with 98% certainty. In this optimization, fin thickness and the fin spacing contribute to 34% and 51% of the cost respectively.

Doing an Optimization – Single Objective and Multi-Objective

Once the sensitivity analysis is complete, it is very easy to add different optimizations. To put in perspective, sensitivity analysis took us couple of days and adding an optimization took only 10 minutes. Even though OptiSLang gives us the option of performing both sensitivity analysis and optimization together, we would recommend completing the sensitivity analysis and then adding different optimizations to it. In the optimization task bar (Figure 36), we can choose the objectives and constraints for the project. We used two approaches for the project. In the first approach we did a multi-objective run. We minimized the cost and temperature while setting the natural frequency as a constraint. The output is a pareto front with best designs shown in red dots (Figure 37). The black dots are all the designs evaluated by OptiSLang. In the second approach, we did a single objective run. Here, we directly tried maximizing the sales. The designs evaluated by OptiSLang are shown in light green and the best design is shown in red (Figure 38). By clicking on the point, we can see the characteristics of the respective design point. We can also output all the relevant information into a spreadsheet.

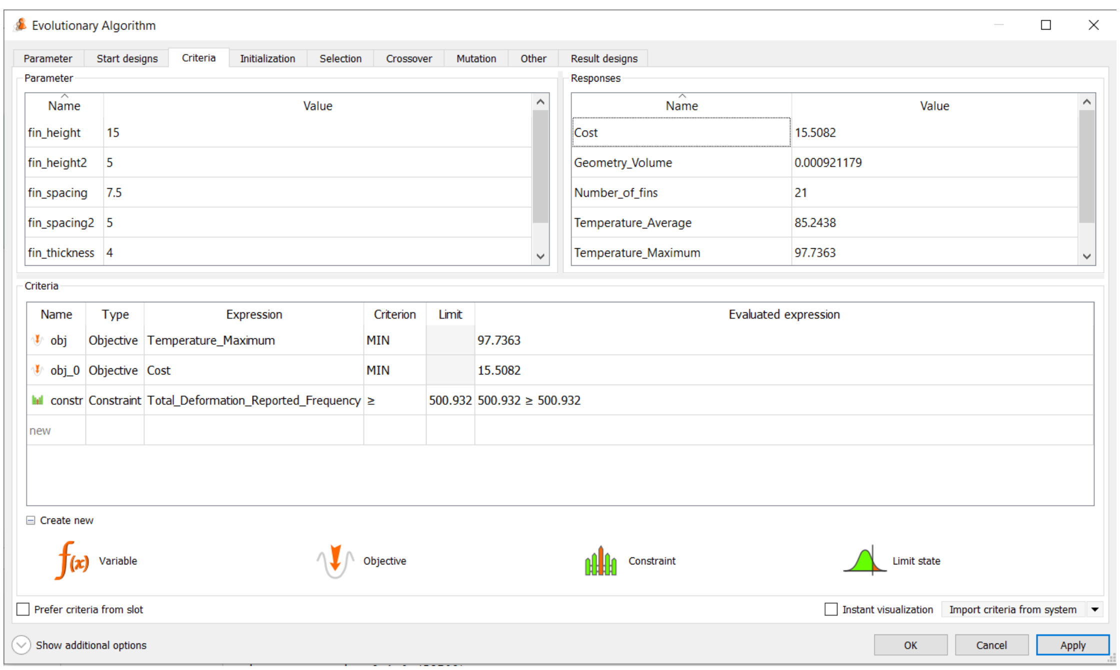


Figure 36: Choosing the parameters and responses for optimization

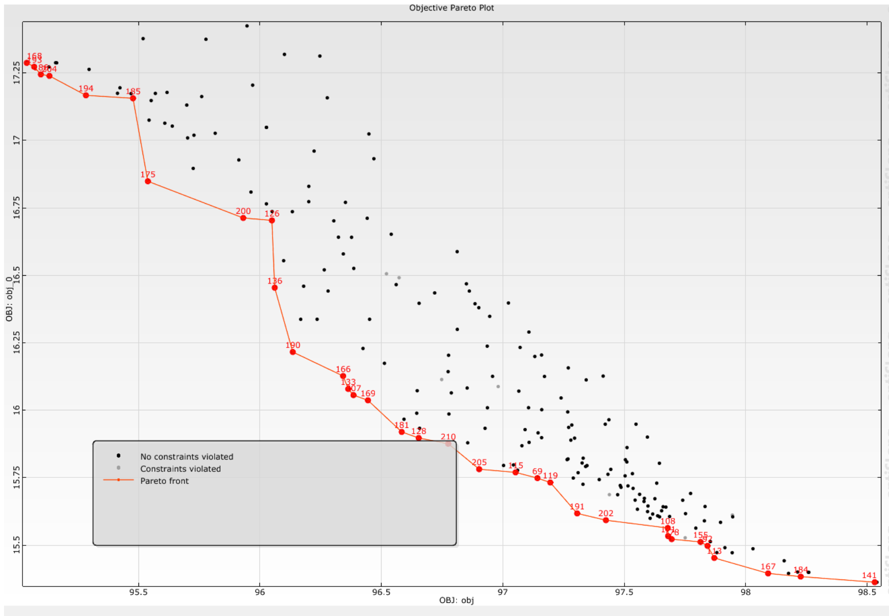


Figure 37: Output from the multi-objective run. A pareto front of cost versus temperature.

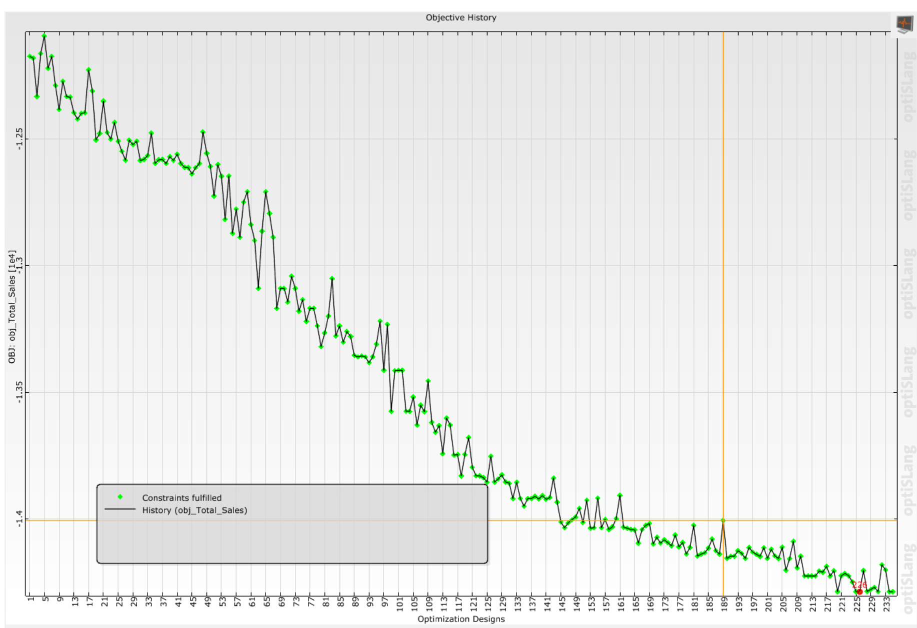


Figure 38: Output from a single objective run

Chapter 6: Sales Model

The need for a single holistic metric

Our natural instinct was to try to minimize the cost. As shown in Table 4, we were able to find a design point with a cost of only 15.4$. However, reducing the cost made our temperature worse. As shown in the minimum temperature design point in Table 4, when we tried minimizing the temperature the cost became worse. We soon realized that we have a dilemma. How do we choose a design which satisfies both the objectives? For instance, Antoine might choose a design with the lowest cost and David might choose the design with the lowest temperature. Notwithstanding the fact that we only have a small team. Danfoss, however is a large, global, company operating in many different countries around the world. Every business unit and team will have a different way of looking at the problem. Also, over here it is a bit simple as we only have two objectives. It is possible that in other real-life scenarios we will have even more objectives, adding complexity to the problem. This shows the need for a universal holistic metric which we can use to choose the best design for the company.

Parameters	Minimal Temp. Design	Minimal Cost Design
Max. Temperature	95.0 C	98.5 C
1st Natural Frequency	518 Hz	631 Hz
Baseplate Cost	17.3 $	15.4 $

Table 4: Design points with minimal temperature and cost

Single holistic value: sales

For our case, that single holistic metric is sales. Sales is just the number of pieces we sell. The formula for sales (Equation 1) is straightforward. Sales is equal to the current sales plus the new sales due to lower cost and new sales due to lower temperature. Reducing the cost of the baseplate will make the product more competitive and will help Danfoss get more customers in the excavator market. If we are able to drastically reduce the price of the baseplate, not only will we sell the baseplate to more number of customers in the excavator market, but we can also sell the baseplate for other applications like the forklift market. Similarly, by lowering the temperature of the baseplate, we can store the baseplate in more locations within the excavator. This flexibility might be very attractive to some OEMs and will lead to more sales. As you can see in Figure 39, we converted the cost and temperature data into sales.

Sales = current sales+ new sales (C) + new sales (T)

New sales (C) = additional sales due to lower cost

New sales (T) = additional sales due to lower temp.

Equation 1: Sales formula

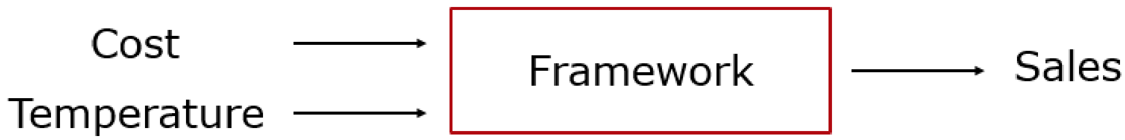


Figure 39: The conversion process

<u>Sales model example - fictional</u>

To show how the conversion works, let us do an example! The sales numbers which we used here are fictional, due to the limited time and scope of the project. When Danfoss makes the decision to go from Design A to Design B, all Danfoss needs to do is update the table with the latest market information. The formula for sales is shown in Equation 1. Let us say the current sales for this example is 10,000 pcs and the design improves by 1.75$ and 2.5 C. Looking at Table 5, we know that 1.75$ falls between 1$ and 2$ and thus the new sales due to lower cost will be 7,500 pcs. Similarly, 2.5 C falls between 2C and 3C and therefore there will be 10,000 pcs in new sales due to lower temperature. The total sales for this design point will be 10,000 plus 7,500 plus 10,000, and this will be equal to 27,500 pcs.

Cost Imp.		New Sales
0 $	0.5 $	+1000 pcs
0.5 $	1 $	+5000 pcs
1 $	2 $	+7500 pcs
2 $	3 $	+10000 pcs

Temp. Imp.		New Sales
0 C	0.5 C	+1000 pcs
0.5 C	1 C	+5000 pcs
1 C	2 C	+7500 pcs
2 C	3 C	+10000 pcs

Table 5: New sales for cost and temperature improvement

Sales Model 2 – adding negative values

Cost Range ($)		Sales (pcs)
0	0.5	1000
0.5	1	5000
1	2	7500
2	3	10000

Positive

Cost Range ($)		Sales (pcs)
-0	-0.5	-1000
-0.5	-1	-5000
-1	-2	-7500
-2	-3	-10000

Negative

Table 6: Sales model with improvement and worsening of cost

To the sales model, we have added negative values. In the previous scenario, we only looked at the increase in sales in pcs if the design improved by a certain amount. This time we are also looking at the tradeoff between temperature and cost and are also quantifying the loss in sales. As shown in Table 6, if the design point improves by 1.5$ then we would get 7,500 pcs more in sales and if design becomes 1.5$ worse in cost then we would lose 7,500 pcs in sales. We applied the same concept with temperature. With the added complexity, we realistically model that improving on temperature would lead to more sales, but the subsequent increase in cost would also make the product unattractive to few of the present and future customers.

Sales model – Approach 1 and Approach 2

We looked at two different approaches to the sales model. The first method was to set the objectives to minimize cost and temperature with natural frequency as a constraint. Then run the optimization and convert the cost and temperature data from the pareto front into sales. Pareto front consists of the best design points chosen by OptiSLang. In the second approach, we directly linked the cost and thermal model in Ansys to the sales model, and we only set one objective in OptiSLang of maximizing sales.

Approach 1

Same way as we did in the example (Sales model example – fictional, pg. 42), we converted all the cost and temperature information of all design points into sales (Figure 40). On the chart's left-hand axis, we have sales and on the right-hand axis we have cost. On the x-axis we have the 33 best design points chosen by OptiSLang. We are choosing the best amongst the best design points. In the red line is sales and in the blue line is the cost. Looking at the red line, we can see that there is a maximum sales of 15,000 pcs. However, in this unique situation design points 1 to 7 all have the same maximum sales of 15,000 pcs. To choose the best between them, we look at cost. Design point 7 is the one with the lowest cost. As you can see, design point 7 has the maximum sales with the lowest cost. Thus, design point 7 is the most profitable design point for the company.

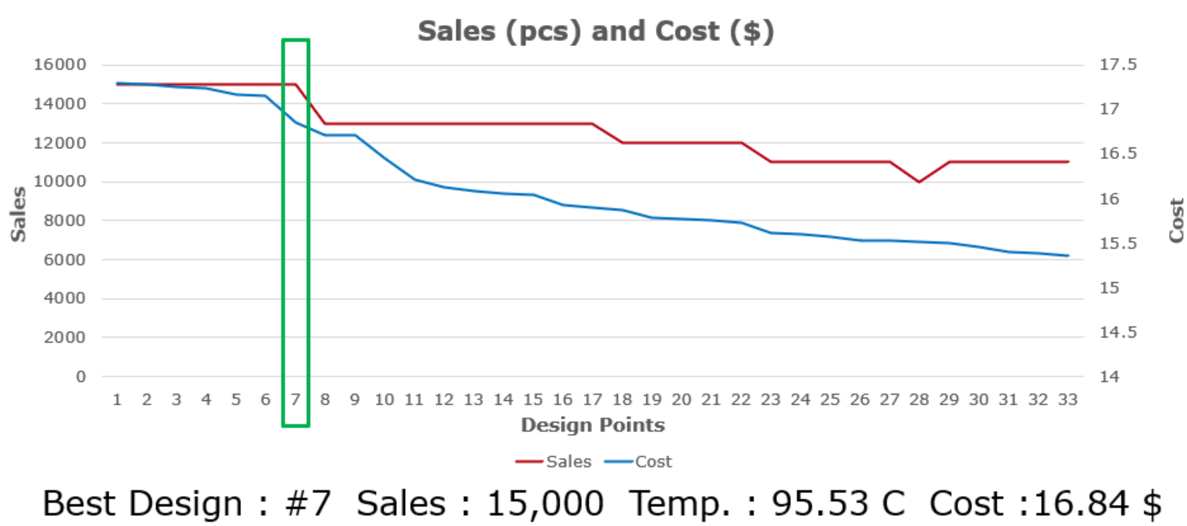


Figure 40: Chart showing sales and cost of the best design points

We were able to use OptiSLang to increase sales from 10,000 to 15,000 pcs. Before we get to too excited, it is important to note that the sales numbers used over here are fictional. The reason we focused on sales is that the profit from selling a pump system is far greater than the baseplate cost. This might not be the case for other MDO applications. In that case we can create a profit framework and that would be very similar to the sales framework we just showed. Looking at Table 7, we can see the characteristics for the existing design and the best design chosen by OptiSLang. The best design has a slightly higher cost but has a much better natural frequency and maximum temperature, and this leads to more sales.

Parameters	Existing Design	Best Design #7
Max. Temperature	97.7 C	95.5 C
1st Natural Frequency	501 Hz	630 Hz
Baseplate Cost	15.5 $	16.8 $
Sales	10,000 pcs	15,000 pcs

Table 7: Characteristics of the existing design and the best design according to Approach 1

Approach 2

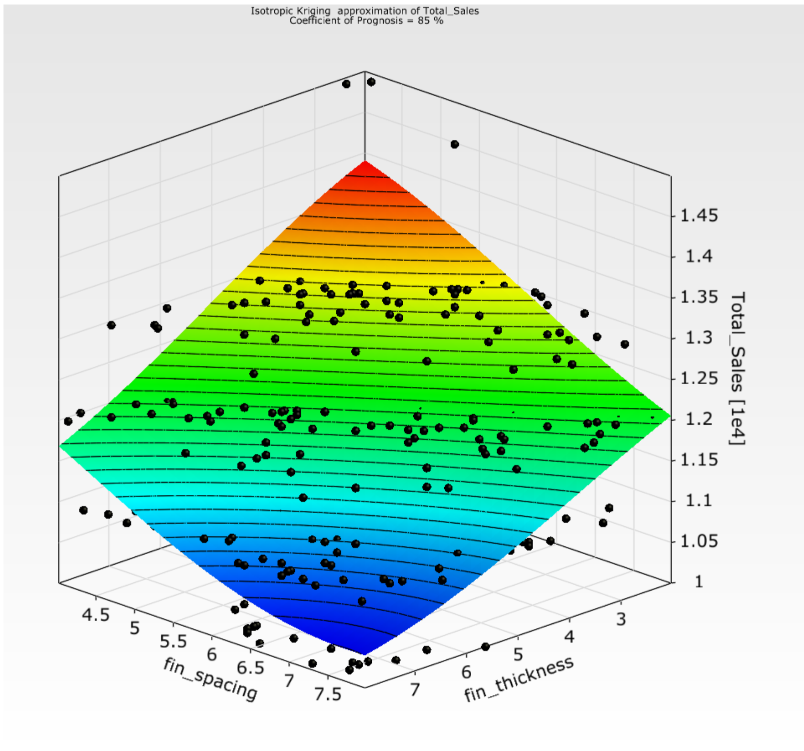


Figure 41: Sensitivity analysis for the sales model. The plot shows the impact of fin spacing (an indirect index for the number of fins) and fin thickness on sales.

The other approach which can be used is to create a sales model in Ansys. Sales would be defined in the same way as shown in Equation 2. The same calculation would instead be done in the sales model Excel spreadsheet in Ansys. For every iteration, cost (output from the cost model) and temperature (output from the thermal model) would be inputs for the sales model. The sales model would calculate the sales for each design point and would send the results to OptiSLang. In the optimization, there will be a need for only one objective and that would be to maximize sales. To make sure the design points are viable for the company, we can set a constraint for cost, temperature and natural frequency. The sensitivity analysis and optimization results are shown in Figure 41, Figure 42 and Figure 43. The maximum sales of 14,288 pcs closely matches the 15,000 pcs in sales from approach 1.

Sales = Current sales + New sales due to lower cost + New sales due to lower temperature

Equation 2 : Sales Formula

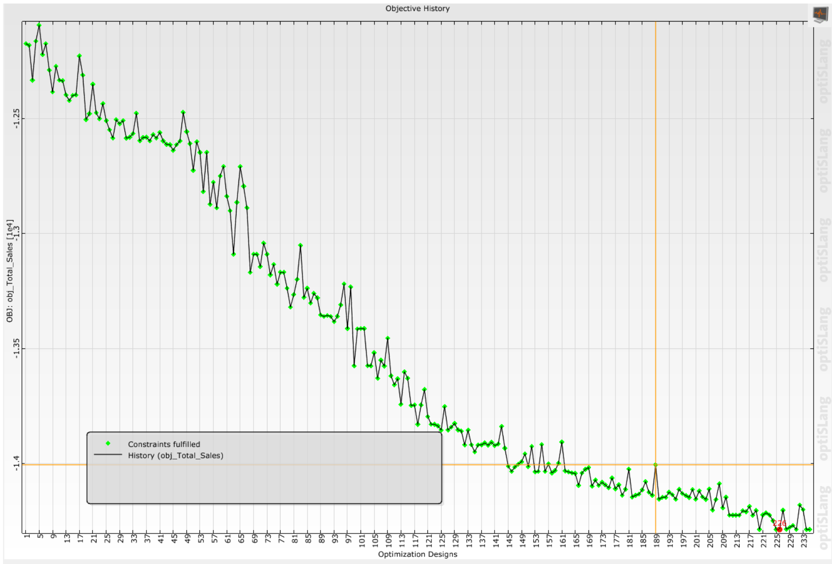


Figure 42: Output from sales model optimization. Sales is on the y-axis and the design points are on the x-axis. The red dot, design point 226, has the highest sales for the optimization. The characteristics of the design point are shown below in Figure 43.

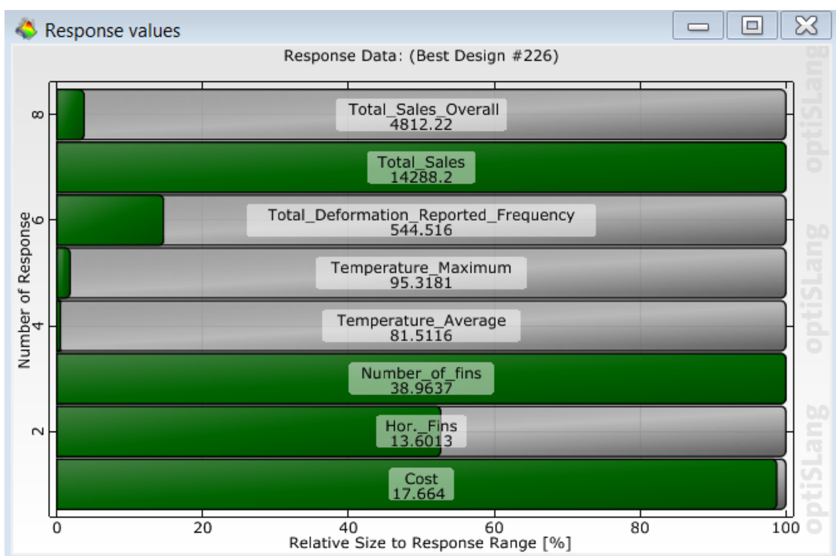


Figure 43: The design point with most amount of sales (14,288 pcs) for the optimization

Chapter 7: Results

Cost Model

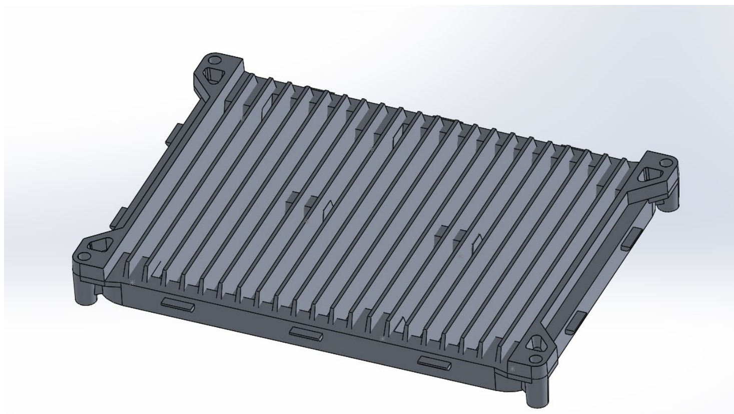


Figure 44: The current baseline design in SolidWorks

Parameters	Value	Unit
Temperature - Maximum	97.7	°C
Frequency - Mode 1	500.9	Hz
Cost – Baseplate	15.5	$

Design Parameters

Number of fins : 21

Fin thickness: 4 mm

Fin spacing: 7.5 mm

Fin height: 18 mm

Fin Angle : 87

No horizontal fins

No adjustment to edges and baseplate thickness

Figure 45: Characteristics of the baseline design

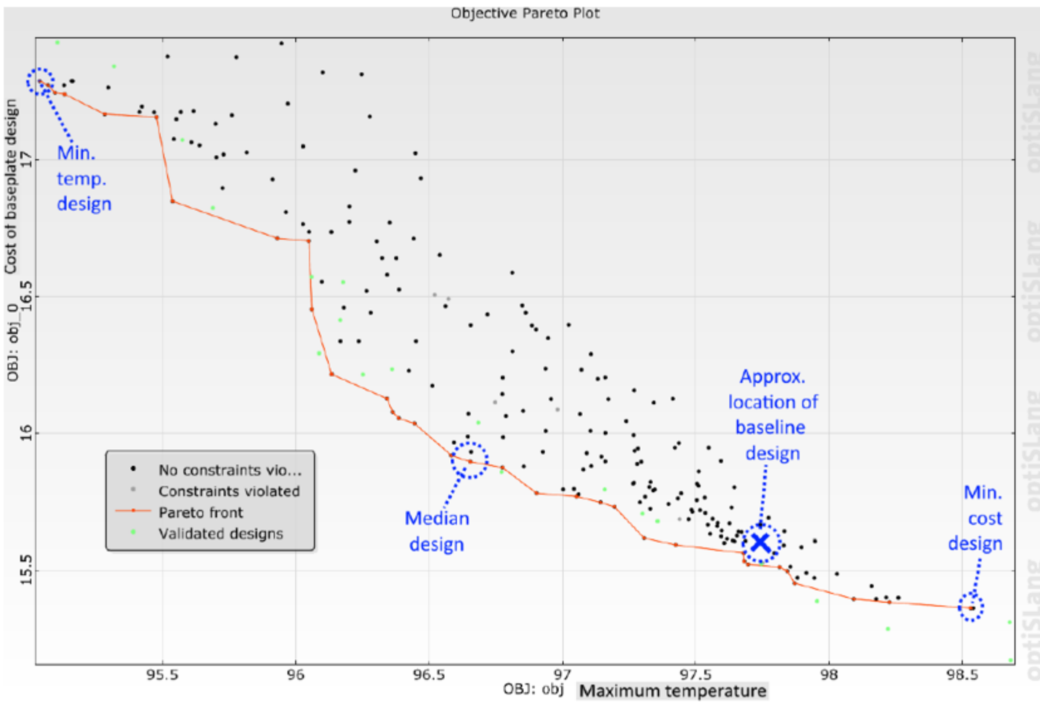


Figure 46: Pareto front of cost versus temperature

The cost simulation calculated the cost of the existing baseline design (Figure 44) to be 15.5$. The other characteristics of the baseline design are shown in Figure 45. The current design has 21 vertical fins and no horizontal fins. The MDO team collectively decided on a set of objectives which included minimizing cost and maximum temperature while setting a lower bound for natural frequency. As shown in the pareto plot (Figure 46) in the results, there is a tradeoff between temperature and cost. Given how close the baseline design is to the pareto front, we know that the baseline design is almost optimized. With the new data from OptiSLang we can choose which area we can optimize on. So, if we were to reduce the cost of the baseplate design we would move towards the right and if we were to reduce the temperature then we would move towards the left. By reducing the number of vertical fins we can lower the cost to $15.36. If maximum temperature is a concern for Danfoss, then we can add more vertical fins and decrease the maximum temperature to 95C.

Approach 1

Sales Model 1 shows the results of the sales model with only positive values and Sales Model 2 shows the results with the positive and negative values. In both the models, we were also able to increase the sales by 50%! Compared to the original design, the optimum design has a significantly better maximum temperature and natural frequency for a slightly higher cost (Table 8). Compared to recommendation for Sales Model 1, the recommendation for Sales Model 2 is more balanced with a slightly lower cost increase and lower performance improvement in maximum temperature and natural frequency. The best designs from Sales Model 1 and Sales Model 2 are also compared on the pareto front (Figure 47). The sensitivity analysis and optimization results are shown in the appendix.

Parameter	Original Design	Sales Model 1	Sales Model 2
Sales	10,000	15,000	15,000
Temperature	97.7	95.5	96.7
Frequency	501	630	611
Cost	15.5	16.84	15.9

Table 8: Results from Approach 1. The units are in pcs, C, Hz, and $ respectively.

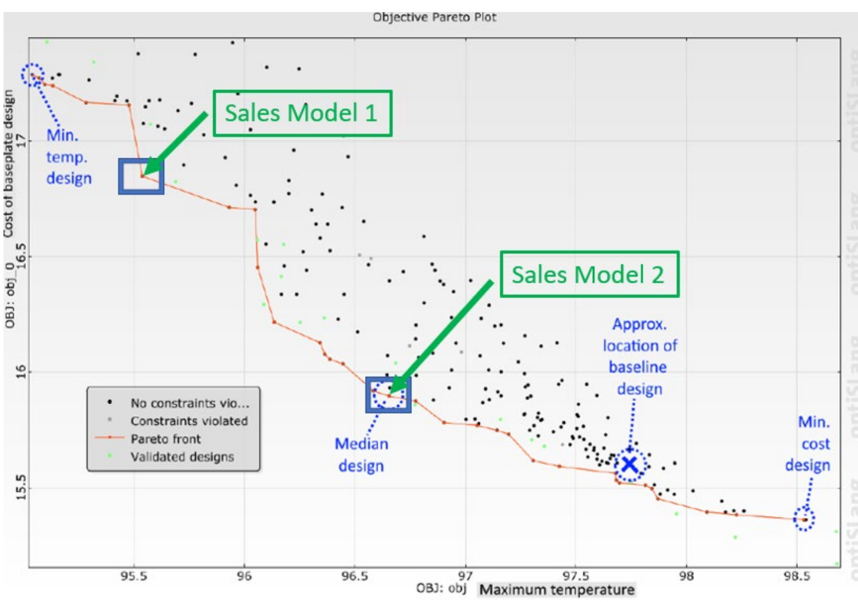


Figure 47: Best design from Sales Model 1 and Sales Model 2 on the pareto front

Approach 2

In the previous approach, we set multiple objectives in OptiSLang and converted the best design points into sales. In the second approach, we have combined the cost model with the sales model and have created a new sales variable in Ansys and OptiSLang. This time we only set one objective which is to maximize sales. Cost, maximum temperature and natural frequency are used as constraints in order to make sure the design points are viable. During the sensitivity analysis, we see the relation between variables and sales. We get a low COP, a correlation factor used within Ansys, as OptiSLang defines COP based on the relationship between the final output (sales) and initial inputs (the number of fins, fin thickness). Sales Model 1 has a COP of 85% and Sales Model 2 has a COP of 66%. Given the limitation of OptiSLang in which we cannot define COP based on the relation between intermediate variables (cost and temperature) and final output (sales), we get a low COP. Given that high COP is needed for optimization, we would recommend using approach 1 until OptiSLang develops additional functionalities. For Sales Model 1, we can maximize sales to 14,288 pcs and for Sales Model 2 we can maximize sales to 11,901 pcs (Table 9). The values are lower for approach 2, indicating that approach 1 is also more effective in maximizing revenues.

Parameter	Original Design	Sales Model 1	Sales Model 2
Sales	10,000	14,288	11,901
Temperature	97.7	95.3	97.3
Frequency	501	544	615
Cost	15.5	17.7	15.7

Table 9: Results from Approach 2. The units are in pcs, C, Hz, and $ respectively.

Comparison between Approach 1 and Approach 2

Approach 1 has a higher sales of 15,000 pcs for both Sales Model 1 and Sales Model 2 (Table 10). Approach 2 has a sales of 14,288 pcs with Sales Model 1 and a significantly lower sales of 11,901 pcs with Sales Model 2. One of the disadvantages we noticed with approach 2 was that OptiSLang calculates the COP based on the correlation between the initial input and the final output. COP is a correlation factor used within Ansys and a high COP would indicate strong correlation. OptiSLang recommends a high COP for conducting an optimization. In approach 2, the initial inputs (number of fins, fin thickness) calculate the cost and temperature, and cost and temperature calculate the final output (sales). The indirect relationship between the initial input and final output causes a lower COP, even though we have very effective cost and thermal models.

Currently, OptiSLang is only limited to initial inputs for optimization (based on our current knowledge). However, if OptiSLang were to include dependent inputs (cost and temperature) for optimization then we will have an equally high COP for approach 2. Another challenge with approach 2 might be getting detailed marketing information for optimization. Strong correlation, such as 0.1 $ in cost improvement with 10 pcs in new sales would be hard to quantify.

Due to the limitations with approach 2, the near-term approach should be to use approach 1 and convert cost and temperature information into sales. However, as Danfoss and Ansys improve their MDO capabilities, approach 2 will become an increasingly viable option. Approach 2 will be easy to scale throughout the entire company, as Danfoss would be able to direct every employee using MDO to use sales maximization as the only objective. The uniformity and consistency from approach 2 for choosing the best design, will help Danfoss in implementing an effective company-wide procedure for MDO.

	Approach 1	Approach 2
Sales Model 1	15,000	14,288
Sales Model 2	15,000	11,901

Table 10: Summary of Sales from different methods. Current sales is 10,000 pcs.

Chapter 8: Next steps

Profit framework and the addition of other financial information

We focused on sales data as the profit from selling a pump is significantly greater than the baseplate cost. The baseplate is only a small component of the pump and the value lies in increasing the number of pumps being sold. We could also create a framework similar to the sales framework, in which we focus on profitability of the component. Similar to methods used for maximizing sales, we could convert the best designs from the multi-objective pareto front (approach 1) or use a single objective function to maximize profits (approach 2). We could also link the data with NPV, IRR and other advanced financial metrics and directly find the best design for the company. OptiSLang is very well integrated with Excel, allowing engineers, designers, and accountants and other key cross-functional stakeholders to easily access the data and help them in choosing the best design for the company. While scaling MDO, we would recommend Danfoss to create a financial template for OptiSLang which could be used throughout the entire company or a customized template for each business unit.

Using MDO to improve the value chain

aPriori not only gives us the cost of each design point but also gives us the breakdown of the expenses. On the high level, we get the process cost, material cost and supplier margin. We also get the detailed information of what contributes to each expense. For our project, we focused on the final cost of the baseplate. However, we can also link the sub-component costs with OptiSLang. For instance, instead of optimizing on the baseplate cost we could optimize the design for the process cost. This would be important if there was a labor strike or a steep projected increase in wages. Also, in case there was an increase in the commodity prices we could choose a design with low material usage. Having all the data linked with OptiSLang allows us to make immediate changes to our design, and would help the company react quickly to the changing business environment. We could also include units in different dimensions such as manufacturing time and create a design with quick lead time.

We live in a global world with rapidly shifting supply chains. Danfoss can use MDO to choose a design best optimized for a certain geography or location. Engineers could rapidly adjust the design for cost-sensitive regions and adjust the design for quality conscious customers. The design could be adjusted depending on if it were being manufactured in low-cost regions in Asia or high-cost regions in North America. Well implemented MDO would give significant design freedom, and would contribute to flexibility in manufacturing, supply chain, finance and other critical operations of the company. MDO linked with financial information can also be used to make strategic decisions. Using aPriori we can find the cost of manufacturing different designs in different countries. We can then use OptiSLang to recommend us not only the best design but also the best location and plant for manufacturing the component.

Cross-functional data driven organizations

Companies trying to implement MDO, should not view the process as simply adding a software tool to the design toolkit, but view this as a milestone moment in which they change towards interdisciplinary, data-driven companies. Large industrial companies currently work in silos. Organizations are neatly split into separate divisions. Marketing and sales give information to R&D, which in turn give to product development and then the information finally goes to supply chain and manufacturing. Finance double checks at different points and gives direction to the process. Many parts of the process are qualitative and the recorded data is lost in different communication channels.

In order for MDO to be successful and reach the potential described in the previous page, MDO users need access to high quality, reliable data. We need to know that if we improve the design by 1 deg. C we will get approximately 1000 new customers. And if that changes, then we need the sales team to immediately inform the MDO team. We need manufacturing and procurement teams to continuously double check on estimates from aPriori to make sure they are realistic. It is a chicken and egg dilemma. For MDO to be successful we need access to data, and for an organizational push towards data collection we need MDO to be effective.

There are secular trends underway like the steep reduction in computing costs (discussed in the background section) which would likely make MDO an essential tool for design. Companies which invest today in data collection will develop a competitive edge in using MDO. As hinted in the name, teams implementing MDO would need to be interdisciplinary. When merged with an interdisciplinary, data-driven organization, MDO can be an effective tool for creating products which maximize value for the company.

Feedback on the project

Implementing MDO was an ideal project for the pandemic as we could work remotely. If MDO is the direction towards which the industry is heading, then I wouldn't be surprised if industrial companies start resembling their technology counterparts. The best decisions we did was to split the project into parts we are interested in. I was always interested in business, and this was an amazing oppurtunity for me to explore the intersection of finance and engineering. The other good decision was to have regular weekly meetings during the Spring semester and daily updates during the summer.

We did have a few hiccups on the way. We had issues finding computational resources for OptiSLang. HPC resources within MIT are set up for standard products licensed to MIT. Given that OptiSLang is a new niche addition to the Ansys product lineup, we could not directly connect with HPC resources. We did find Nimbix but the lead time for installing OptiSLang onto their servers was not viable. We are thankful for Professor Hardt for giving us access to his lab's high performance PC. Teams looking to use MDO should invest in a $5k PC or invest time in installing OptiSLang onto their servers. The other key limitation during the project was OptiSLang. OptiSLang was only recently aquired by Ansys and is still in the process of being integrated with the company. We had to deal with many bugs and crashes. It might have been smarter if we had shown the value of MDO by using Simulia and iSight by SolidWorks as it is already present on the MIT license server. MDO is being integrated into mainstream with CAD and CAE applications and it will take couple of years before design exploration tools become user-friendly.

On the whole this has clearly been an awesome project! After having worked in manufacturing and design, it was great to be downstream and focus on innovation and make strategic recommendations which might have a large influence on the company and the industry.

Chapter 9: Conclusion

We were successfully able to use MDO to increase sales for the baseplate in hypothetical scenarios. In approach 1, we were able to increase sales from 10,000 to 15,000 pcs for both Sales Model 1 and Sales Model 2. In approach 2, the increase was lower at 14,288 pcs for Sales Model 1 and 11,901 pcs for Sales Model 2. Due to the higher sales number and limitations within OptiSLang we would recommend approach 1 for the near term. However, MDO software is increasing in sophistication and approach 2 might become the better option due to uniformity and consistency in implementation.

With the framework, we have shown how MDO can be used to create products which maximize value to the company. MDO can have a direct material impact on the bottom line of the company. Designs can be made which minimize cost or maximize sales. MDO could also potentially be used to maximize other financial metrics like NPV or IRR. We have created a template which Danfoss can use to implement MDO across the entire company. Using hypothetical scenarios, we were able to increase sales of the DPC baseplate by 50% while satisfying the natural frequency, maximum temperature and cost constraints. We were also able to find an optimized design which would minimize the cost and temperature while satisfying the natural frequency constraint. We explored many different ways of using MDO for choosing the best design for the company.

We used cost information from aPriori, a simulation software used within Danfoss, to build our own internal cost model. We were successfully able to integrate the cost model with OptiSLang, and evaluate the cost for each design point. We were also able to expand the cost model to include sales. Similarly, other financial metrics like profit, IRR or NPV can also be added to the model. We used AMOP algorithm for sensitivity analysis and Evolutionary algorithm for optimization. Though simplified for the scope of the project, the cost model and optimization met the criteria set by OptiSLang for an accurate and reliable run.

Teams at Danfoss can create similar simulations in Ansys for their respective products and can update the template with relevant market information. Once implemented, MDO can be an important tool for improving the company's design process and can help realize savings in the entire value chain. Limitations to successful implementation would be access to computational resources, high quality and reliable financial data, and early stage MDO software. Given the commitment shown by Danfoss throughout the entire project and the allocation of key people and resources, we are confident that MDO has the potential to add significant value to Danfoss. The project has been a very successful industry-academia collaboration between Danfoss and MIT, and we hope that the thesis can act as a framework for using MDO to create products which maximize financial benefit to the company and society.

Bibliography

Ansys. (2020). Ansys Cloud. https://www.ansys.com/products/platform/ansys-cloud.

Ansys. (2020). Ansys OptiSLang : Process Integration and Robust Design Optimization. https://www.ansys.com/products/platform/ansys-optislang.

Ansys. (2020). Calculate the Simulation Speed-up and ROI of a New Workstation. https://www.ansys.com/blog/simulation-speed-up-roi-workstation.

Balesdent, M., Bérend, N., Dépincé, P., & Chriette, A. (2010, March 11). *A survey of multidisciplinary design optimization methods in launch vehicle design*. https://www.researchgate.net/publication/226805357_A_survey_of_multidisciplinary_design_optimization_methods_in_launch_vehicle_design.

CIC. (2020). CIC. https://cic.com/cambridge.

Danfoss. (2020). Danfoss – Engineering Tomorrow, today. https://www.danfoss.com/en-us/.

Dassault Systemes. (2020). ISIGHT AUTOMATE DESIGN EXPLORATION AND OPTIMIZATION. https://www.3ds.com/fileadmin/PRODUCTS-SERVICES/SIMULIA/RESOURCES/simulia-isight-brochure.pdf.

Dunbar, B. (2020). *NASA Technology*. NASA. https://www.nasa.gov/centers/dryden/news/X-Press/stories/2008/10_08_technology.html.

Enlyft. (2020). *Top 5 products in the Computer-aided Design & Engineering market*. https://enlyft.com/tech/computer-aided-design-engineering.

Giarratana, C. (2017, September 6). *CATIA Pricing: Design in the Context of the Experience*. https://3dsman.com/catia-pricing/.

Green500. Green500 | TOP500 Supercomputer Sites. (2020). https://web.archive.org/web/20160620115925/http://www.top500.org/green500/.

Lavi, S. (2020, August 29). *ANSYS Workbench Pricing*. https://www.itqlick.com/ansys-workbench-platform/pricing.

MSC Software Corporation. (2020). MSC Nastran - Multidisciplinary Structural Analysis. https://www.mscsoftware.com/product/msc-nastran.

MSC Software Corporation. (2020). MSCOne It's time to explore a smarter way to access simulation software. https://www.mscsoftware.com/product/mscone.

Siemens. (2020). HEEDS Discover better designs, faster. https://www.plm.automation.siemens.com/media/global/en/heeds-brochure-61372_tcm27-66012.pdf.

SpaceX. (2020). SpaceX. https://www.spacex.com/.

Wikimedia Foundation. (2020, November 4). *FLOPS*. Wikipedia. https://en.wikipedia.org/wiki/FLOPS.

Wong, K. (2010, September 16). *MSC Software Introduces Six MD Nastran Desktop Bundles*. Digital Engineering. https://www.digitalengineering247.com/article/msc-software-introduces-six-md-nastran-desktop-bundles/.

Appendix

Thesis by Teammates

More information about the Danfoss-MIT collaboration can be found in “Case Study of Multidisciplinary Design Optimization Implementation Process Management” by Antoine Yazbeck and “Multidisciplinary Design Optimization of Part Geometry in CAD” by David Mimery.

Terms specific to OptiSLang

The definition of EA, AMOP, and COP can be found in “Methods for multi-disciplinary optimization and robustness analysis” guide by Dynardo (OptiSLang) GmbH.

Links on Agile

https://hbr.org/2016/05/embracing-agile

https://www.forbes.com/sites/stevedenning/2016/08/13/what-is-agile/?sh=50eef3cb26e3

https://www.atlassian.com/agile

Cost Model

Characteristics of the design point with the lowest cost according to the cost model.

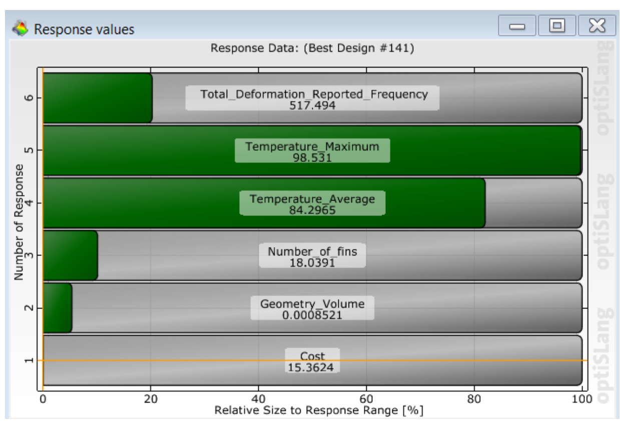


Figure 48: The design with the lowest cost

Approach 1

Characteristics of design points with the highest sales according to Sales Model 1 and Sales Model 2.

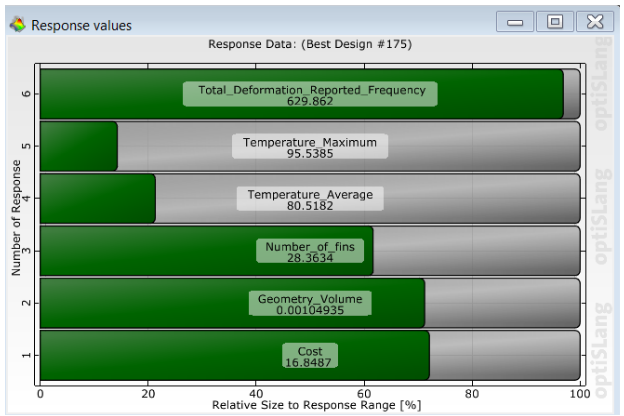


Figure 49: Best design for Sales Model 1

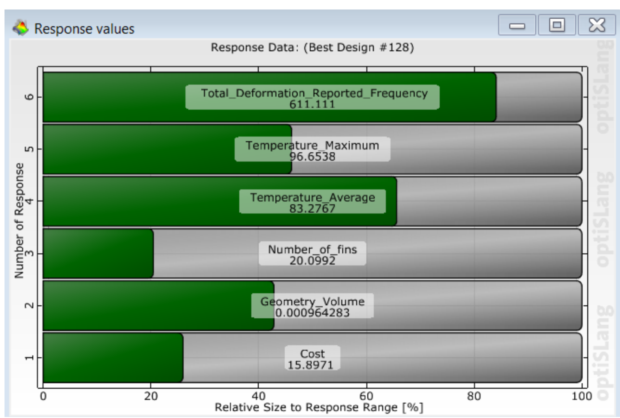


Figure 50: Best Design for Sales Model 2

Approach 2

To do a brief recap here. Sensitivity analysis (Figure 51 and Figure 52) is the first step in which we see the impact of the inputs (number of fins, fin thickness) on the output (sales). In the next step, we optimize across design points to find the baseplate design with the highest sales (Figure 53 and Figure 54). The design points with the highest sales are shown in red dots in the optimization plots and their characteristics are shown in Figure 55 and Figure 56.

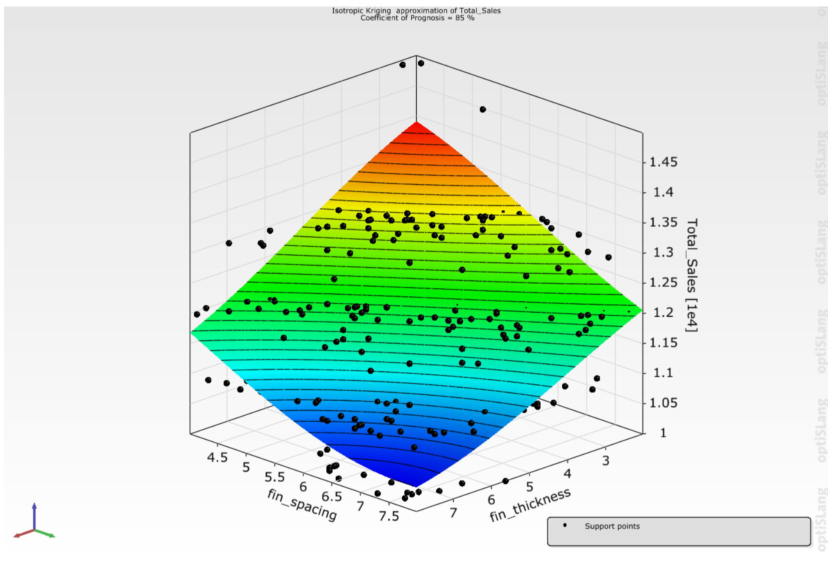


Figure 51: Sensitivity analysis for Sales Model 1

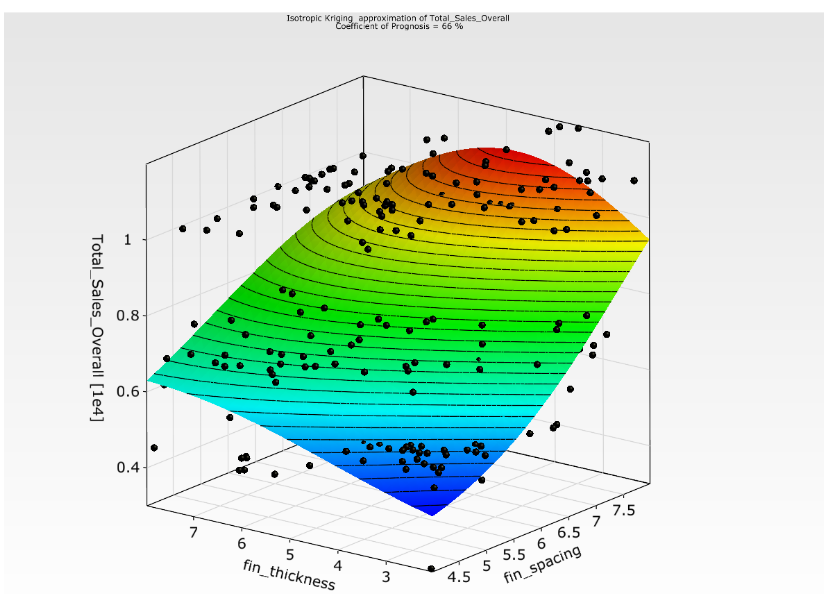


Figure 52: Sensitivity analysis for Sales Model 2

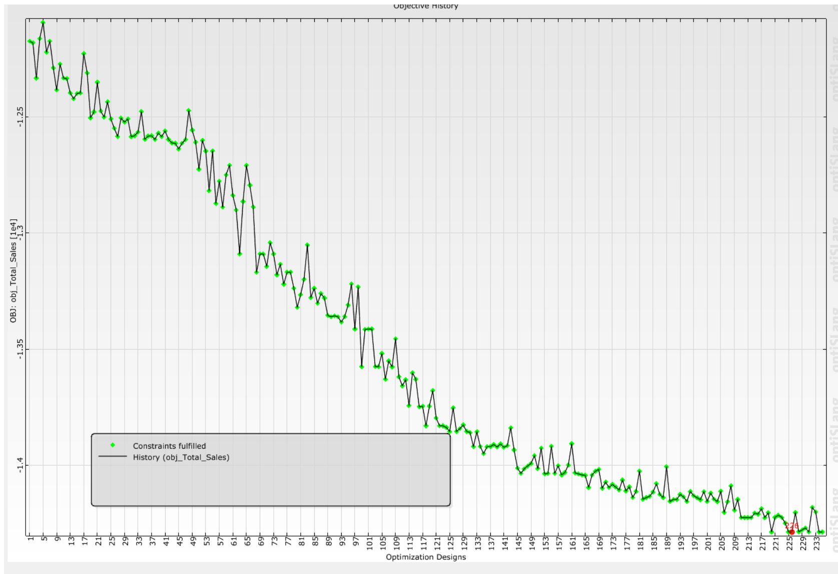


Figure 53: Optimization for Sales Model 1

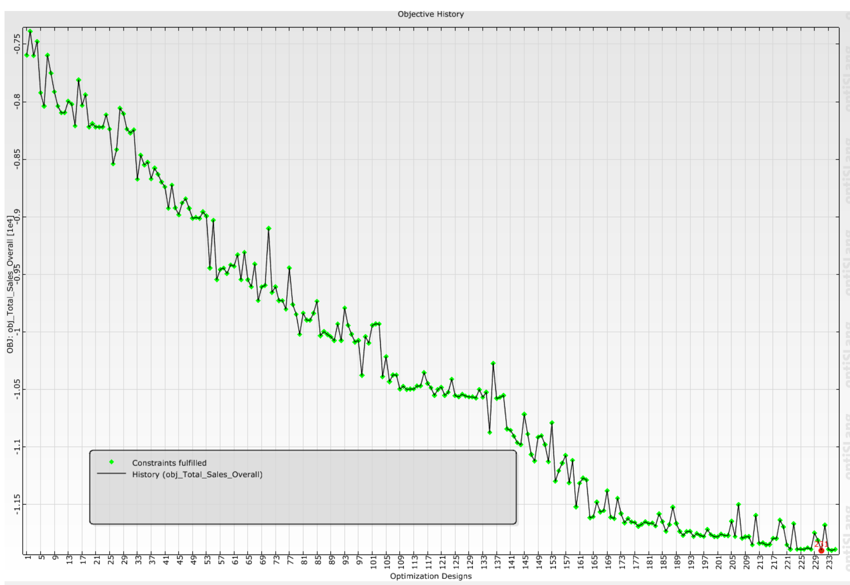


Figure 54: Optimization for Sales Model 2

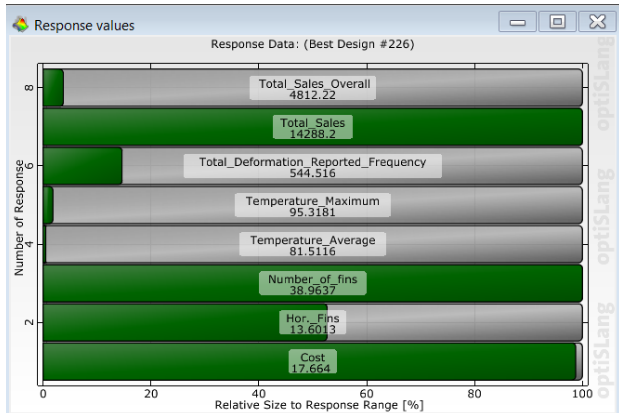


Figure 55: Best Design for Sales Model 1

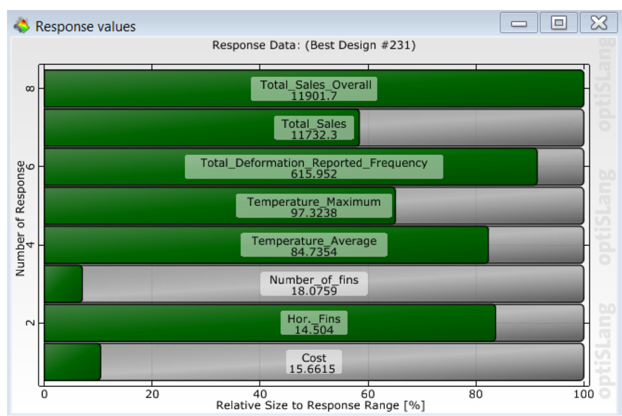


Figure 56: Best Design for Sales Model 2

Impact of variables on baseplate price

Remaining plots from Cost Model Concept section in pg. 25.

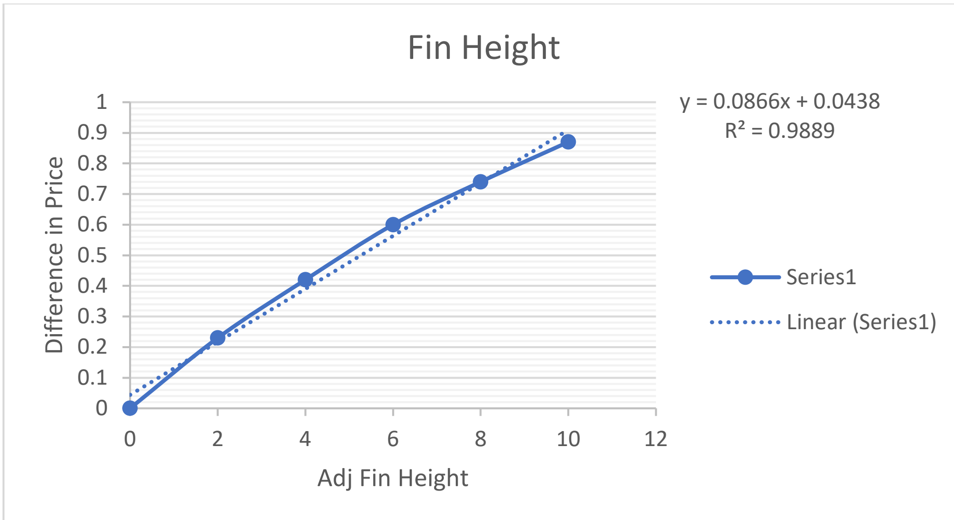


Figure 57: Impact of Fin Height on Baseplate Price

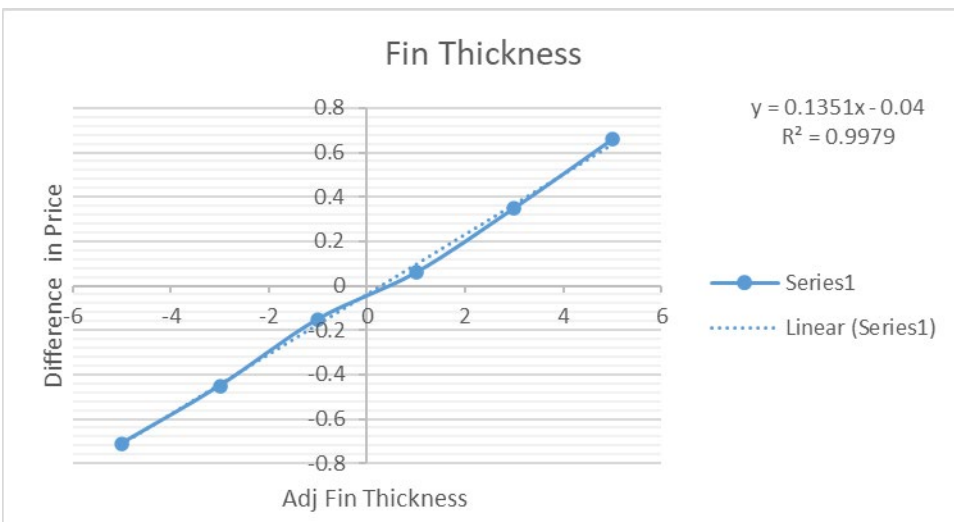


Figure 58: Impact of Fin Thickness on Baseplate Price

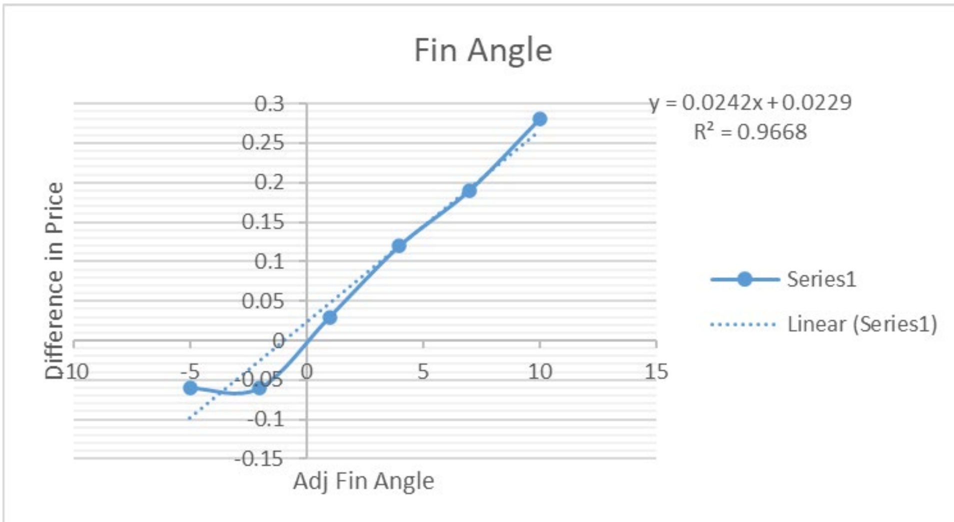


Figure 59: Impact of Fin Angle on Baseplate Price

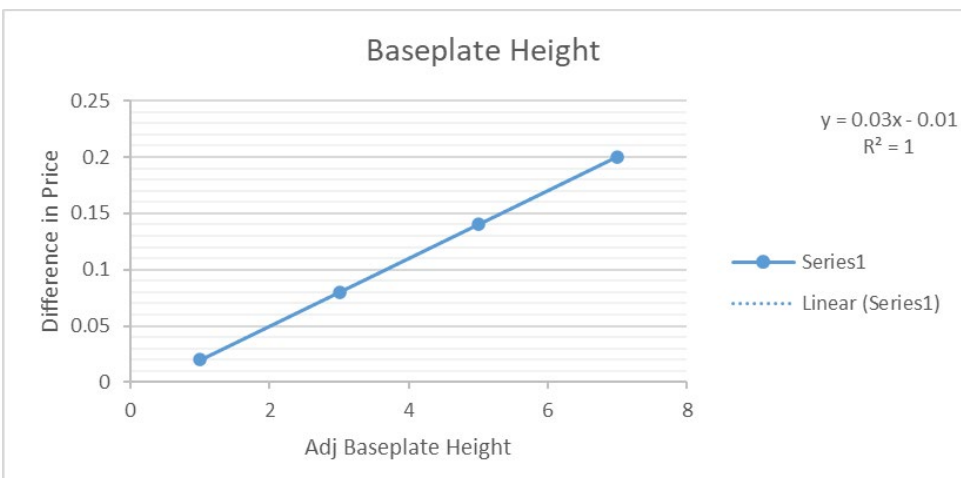


Figure 60: Impact of Baseplate Height on Baseplate Price

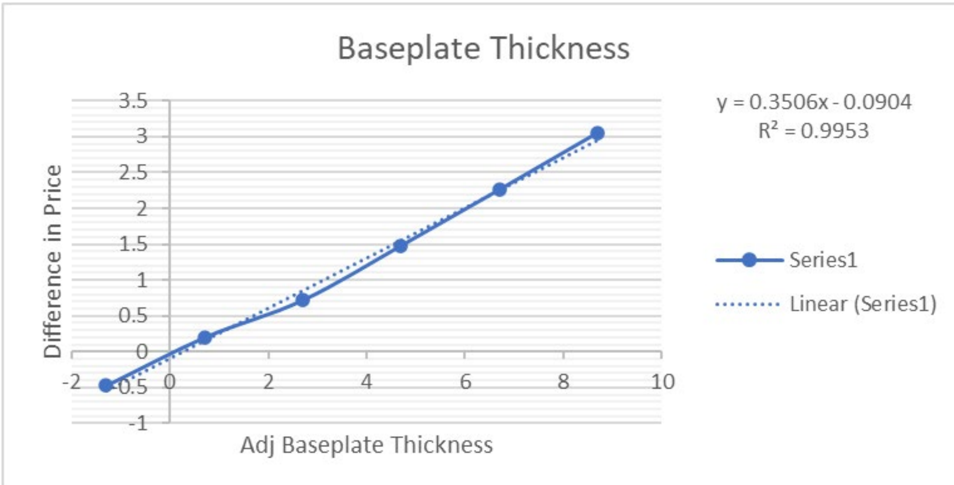


Figure 61: Impact of Baseplate Thickness on Baseplate Price

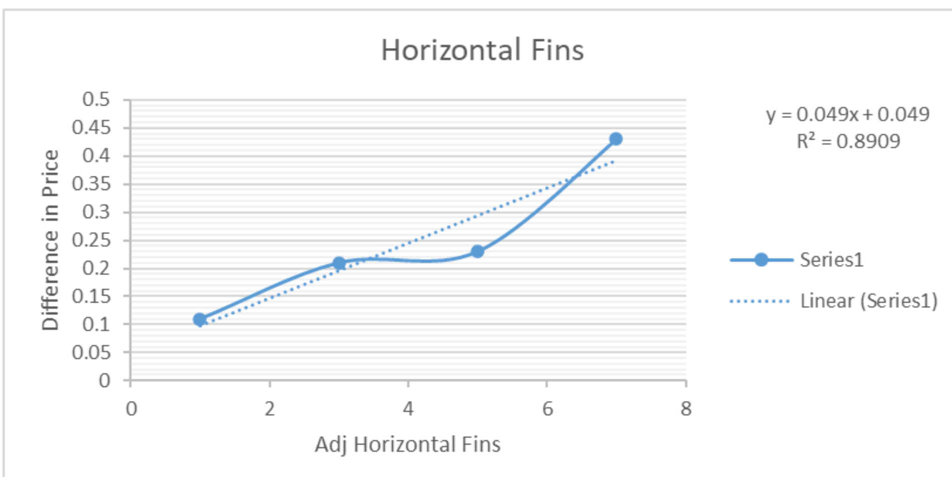


Figure 62: Impact of Horizontal Fins on Baseplate Price

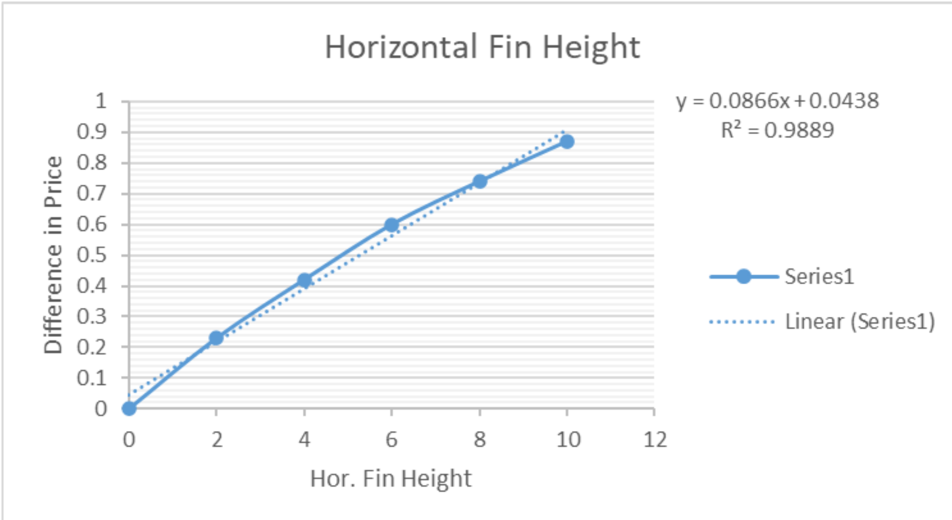


Figure 63: Impact of Horizontal Fin Height on Baseplate Price

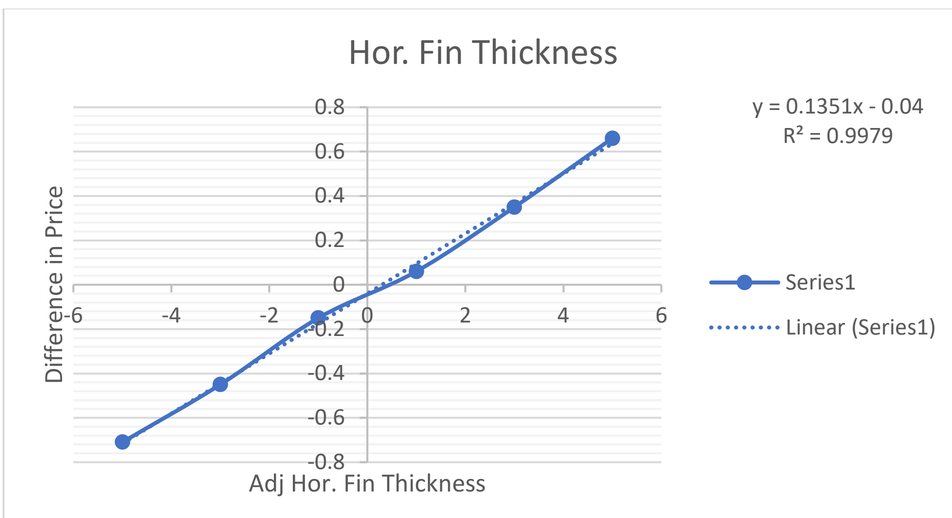


Figure 64: Impact of Horizontal Fin Thickness on Baseplate Price

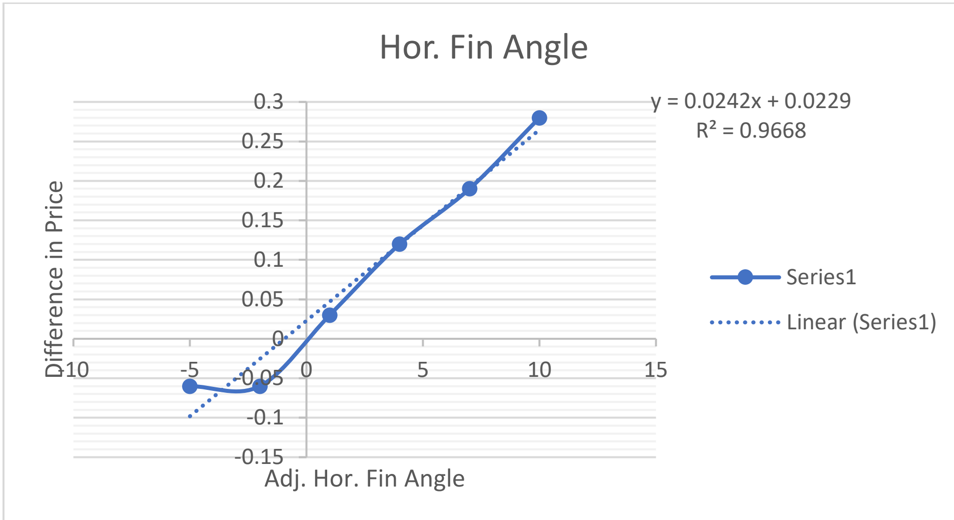


Figure 65: Impact of Horizontal Fin Angle on Baseplate Price