AUTOMATIC DESIGN OF WIRING PATTERN FOR CAR SEAT HEATERS

CEYHUN SENGUL
HAMON ABDOLLAHIFAKHR

MASTER THESIS 2010
MECHANICAL ENGINEERING
 AUTOMATIC DESIGN OF WIRING PATTERN FOR CAR SEAT HEATERS

CEYHUN SENGUL
HAMON ABDOLLAHIFAKHR

This Master Thesis project has been performed at the School of Engineering of the Jönköping University within the discipline of mechanical engineering. This thesis is the last part of the Master’s program. The authors are responsible for all opinions, conclusions and results in thesis.

Supervisor: Fredrik Elgh
Examiner: Fredrik Elgh
Extent: 20 credits (D level)
Date: 2010/08/11

Number of archive:
Abstract

This project aims to develop design automation in product development. Design automation causes increase in producibility and decrease in product cost and manufacturing lead time.

The study at hand is proposed to provide a new method and to introduce procedure to the design of wiring pattern for a car seat heater for Kongsberg Automotive, KA. KA is a Norwegian company and a global provider of engineering, design, and manufacture for seat comfort, driver and motion control systems, fluid assemblies, and industrial driver interface products. The method that currently is used in the company to create a wiring pattern is neither sufficient enough nor automated.

In order to design the wiring pattern, at first procedure is handled by the designer. Secondly, car seat heater 2D layout is imported and then, the dimensions of the elements are defined as constraints. Then VBA codes are opened and the program is run. The result will be a wiring pattern in different 2D layouts. To make the design process easier, we have modeled five different layouts; wiring pattern of one element, two elements, three elements, five elements (with two back sides) and one element trapezoidal 2D layout.

The algorithm written in VBA (Visual basic for application) creates the pattern according to the dimensions of the elements which are used as inputs to define constrained parameters. The created macros are simple to use and easy to modify, independent from the programming knowledge. The user is only responsible with parameter input and running the program. The solution gives wiring pattern for a car seat heater.

Keywords

CATIA V5 Wiring Pattern
Design Automation
Kongsberg Automotive (KA)
ProceodoStudio
Producibility
Seat Heater
Visual Basic Applications
Acknowledgement

We are very happy to see that, we succeeded a project which will be useful for the application being developed in Jonkoping University in co-operation with the company Kongsberg Automotive.

We would like to take this opportunity to thank our tutor Mr. Fredrik Elgh for his support and guidance all through the project.

We are very grateful to our family, supporting and loving us from far distances, Turkey and Iran.

Special thanks to our friends and beloveds for their love and encourage.
# Table of Contents

1 Introduction ............................................................................................................................................. 1  
   1.1 Project description .......................................................................................................................... 1  
      1.1.1 Seat-heaters.............................................................................................................................. 1  
   1.2 Company introduction .................................................................................................................... 2  
   1.3 Objective of the project .................................................................................................................. 3  

2 Theoretical background .......................................................................................................................... 4  
   2.1 Design and process ............................................................................................................................ 4  
   2.2 Producibility .................................................................................................................................... 5  
   2.3 Design automation .......................................................................................................................... 5  

3 Approach .............................................................................................................................................. 6  
   3.1 Computational implementations ....................................................................................................... 6  
      3.1.1 CATIA V5 .................................................................................................................................. 6  
      3.1.2 Knowledge Advisor .................................................................................................................... 7  
      3.1.3 Macros ...................................................................................................................................... 7  
      3.1.4 VBA ......................................................................................................................................... 7  

4 Presently used method ............................................................................................................................ 8  
   4.1 Creating outline from the model.......................................................................................................... 8  
   4.2 Automated design and geometry definition ...................................................................................... 9  

5 Problem analysis .................................................................................................................................. 11  
   5.1 Application system developed .......................................................................................................... 11  
   5.2 Design process ................................................................................................................................ 13  
      5.2.1 Specific problem for KA .............................................................................................................. 13  
      5.2.2 Creating the layout ..................................................................................................................... 14  
      5.2.3 Design structure matrix .............................................................................................................. 14  
   5.3 Design variants ................................................................................................................................ 16  
   5.4 CATIA V5 template, KWA and VBA Macros .................................................................................... 19  

6 Generative Design ............................................................................................................................... 21  
   6.1 New idea creation ............................................................................................................................... 21  
   6.2 Horizontal filling ............................................................................................................................... 23  
   6.3 Vertical filling .................................................................................................................................. 23  
   6.4 Using symmetric lines ....................................................................................................................... 24  
   6.5 Function of angle .............................................................................................................................. 25  
   6.6 Step function: .................................................................................................................................. 26  
      6.6.1 Heaviside step function: ................................................................................................................ 27  
      6.6.2 Piecewise constant function......................................................................................................... 27
### Table of figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Wiring pattern in a carrier with sinusoidal loops</td>
<td>2</td>
</tr>
<tr>
<td>1.2</td>
<td>A cushion element</td>
<td>2</td>
</tr>
<tr>
<td>2.1</td>
<td>Model for design automation handling</td>
<td>6</td>
</tr>
<tr>
<td>2.2</td>
<td>3D model of a seat</td>
<td>8</td>
</tr>
<tr>
<td>4.2</td>
<td>2D outlines created from the 3D model</td>
<td>8</td>
</tr>
<tr>
<td>4.3</td>
<td>The new pattern is designed manually</td>
<td>9</td>
</tr>
<tr>
<td>4.4</td>
<td>Wire pattern with constraints</td>
<td>9</td>
</tr>
<tr>
<td>4.5</td>
<td>Sketch with sub elements</td>
<td>10</td>
</tr>
<tr>
<td>4.6</td>
<td>The final sketch that the algorithm will be applied</td>
<td>10</td>
</tr>
<tr>
<td>5.1</td>
<td>ProceedoStudio main structure that system using</td>
<td>11</td>
</tr>
<tr>
<td>5.2</td>
<td>Main screen of ProceedoStudio</td>
<td>12</td>
</tr>
<tr>
<td>5.3</td>
<td>ProceedoStudio execution control environment</td>
<td>12</td>
</tr>
<tr>
<td>5.4</td>
<td>Main structure of design process with general sections is structured</td>
<td>13</td>
</tr>
<tr>
<td>5.5</td>
<td>Creating layout</td>
<td>14</td>
</tr>
<tr>
<td>5.6</td>
<td>Schema of the system</td>
<td>15</td>
</tr>
<tr>
<td>5.7</td>
<td>Example of the car seat heater pattern</td>
<td>16</td>
</tr>
<tr>
<td>5.8</td>
<td>Example of car seat heater layout</td>
<td>16</td>
</tr>
<tr>
<td>5.9</td>
<td>Two common problems of car seat heater layout to fulfill</td>
<td>17</td>
</tr>
<tr>
<td>5.10</td>
<td>Samples of different car seat heater layout</td>
<td>18</td>
</tr>
<tr>
<td>5.11</td>
<td>One sample of car seat heater layout to show starting wiring pattern</td>
<td>18</td>
</tr>
<tr>
<td>5.12</td>
<td>Sample of symmetric wiring pattern for car seat heater</td>
<td>18</td>
</tr>
<tr>
<td>5.13</td>
<td>Schema of implementing design automation concepts for car seat heater</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>- From customer specification, and using tools to have automated design as an output</td>
<td></td>
</tr>
<tr>
<td>5.14</td>
<td>Examples of CATIA V5 template</td>
<td>20</td>
</tr>
<tr>
<td>5.15</td>
<td>VBA macros and geometrical input are given as examples</td>
<td>20</td>
</tr>
<tr>
<td>5.16</td>
<td>Inputs and the outputs desired from the system</td>
<td>21</td>
</tr>
<tr>
<td>6.1</td>
<td>Simplified samples of car seat heater layout</td>
<td>22</td>
</tr>
<tr>
<td>6.2</td>
<td>Different elements use in car seat heater layout</td>
<td>23</td>
</tr>
<tr>
<td>6.3</td>
<td>Example for horizontal filling</td>
<td>23</td>
</tr>
<tr>
<td>6.4</td>
<td>Example for vertical filling</td>
<td>24</td>
</tr>
<tr>
<td>6.5</td>
<td>Example of symmetric lines</td>
<td>24</td>
</tr>
<tr>
<td>6.6</td>
<td>Sinusoidal graph</td>
<td>25</td>
</tr>
<tr>
<td>6.7</td>
<td>Sinusoidal implementing on rectangular pattern</td>
<td>25</td>
</tr>
<tr>
<td>6.8</td>
<td>Different patterns of wiring pattern (using function of angle)</td>
<td>26</td>
</tr>
<tr>
<td>6.9</td>
<td>Step function graph</td>
<td>27</td>
</tr>
<tr>
<td>6.10</td>
<td>Heaviside step function graph</td>
<td>27</td>
</tr>
<tr>
<td>6.11</td>
<td>Piecewise constant function graph</td>
<td>28</td>
</tr>
<tr>
<td>6.12</td>
<td>Piecewise constant function on oscilloscope</td>
<td>28</td>
</tr>
<tr>
<td>6.13</td>
<td>Honeycomb wired pattern</td>
<td>29</td>
</tr>
<tr>
<td>6.14</td>
<td>Honeycomb pattern</td>
<td>29</td>
</tr>
<tr>
<td>6.15</td>
<td>Quadratic element wired pattern</td>
<td>30</td>
</tr>
<tr>
<td>6.16</td>
<td>Quadratic elements pattern</td>
<td>30</td>
</tr>
<tr>
<td>6.17</td>
<td>Peano’s Mapping</td>
<td>31</td>
</tr>
<tr>
<td>6.18</td>
<td>Geometry generation of the peano curve</td>
<td>31</td>
</tr>
<tr>
<td>6.19</td>
<td>Hilbert’s Mapping</td>
<td>31</td>
</tr>
<tr>
<td>6.20</td>
<td>Generating Hilbert’s space-filling curve</td>
<td>32</td>
</tr>
<tr>
<td>6.21</td>
<td>Different space filling-curves</td>
<td>32</td>
</tr>
<tr>
<td>6.22</td>
<td>Example of slice and create points’ method, in this figure number of nodes in each line is same (four nodes in each line)</td>
<td>33</td>
</tr>
<tr>
<td>6.23</td>
<td>Example of slice and create points’ method, in this figure number of nodes in each line is different</td>
<td>33</td>
</tr>
<tr>
<td>6.24</td>
<td>Example to offset lines</td>
<td>34</td>
</tr>
</tbody>
</table>
1 Introduction

1.1 Project description

In today’s world almost all of the large scaled companies are holding only the assembly lines in their own plants and most of the production and manufacturing facilities are provided by the SMEs (small and medium sized enterprise). In order to compute within each other, subcontractors have to show on-time response in quotations with high accuracy in delivery and cost estimations to be able to do such co-operations with the large-scaled companies.

Kongsberg Automotive, KA, is a Norwegian company producing miscellaneous solutions and engineering facilities for automotive industry. The research and development facilities of the company are held in the plant located in Mullsjö, Sweden.

KA has identified a need of support in quotation making process for car seat-heaters in order to reduce design lead-time and increase the accuracy in cost-estimation.

In order to meet the company’s need, in School of Engineering, Jönköping, an application program is developed which is called ProceedoStudio.

The project is involved with the estimation of wiring pattern, wire properties, manufacturing data and the manufacturing costs of a new product. The wiring pattern of the system is provided with an algorithm.

The objective with this project is either to explore new approaches or improve already existing algorithm for wiring pattern of the car seat heaters which is not sufficient enough.

1.1.1 Seat-heaters

The most common method used in the company is FOAM-AD design. In a seat-heater formed by FOAM-AD, a carrier material (a combination of foam and textile) is filled with the sinusoidal loop shaped wires between its layers. Heating wire is implemented to the carrier in a special pattern in order to provide optimum heating system in a car seat considering the requirements specified by the customer.
The complete element is then glued to the seat foam.

According to Gaspa.R.L (2005), the heating element is connected to TCU (Temperature Control Unit). This control unit receives information from the main computer in the car by a so-called Lin Data Bus. From a NTC-thermistor (Negative Temperature Coefficient) placed in the heating element, the control unit gets information about present temperature near the seat surface. This information is then used to regulate the heating element.

1.2 Company introduction

*Kongsberg Automotive* (KA) was founded on 24 March 1987 as a continuation of the activities previously handled by the *Automotive Parts Division of Kongsberg Våpenfabrikk*. 
KA is a global provider of engineering, design, and manufacture for seat comfort, driver and motion control systems, fluid assemblies, and industrial driver interface products (Kongsberg Automotive, 2010).

Headquarter of the company is in Kongsberg, Norway, and has almost 50 facilities in 20 countries. KA, with revenues of about MEUR 623 (2009) and approx. 9,000 employees, provides system solutions to vehicle makers around the world. The company’s main target group is involved in the automotive, commercial vehicle and industrial markets.

The product portfolio includes gearshift systems, cables for a wide variety of application, fuel lines, tubing and hoses, couplings, clutch actuators, stabilizing rods, seat-heaters, seat ventilation, lumbar supports, head restraints, arm rests, steering columns, pedals, electronics and displays (Kongsberg Automotive, 2010).

The plant in Mullsjö, Sweden, is driving the research and development applications for the seat comfort systems including the head restraints and seat-heaters. The prototype planning and testing facilities also take place in the plant.

1.3 Objective of the project

The whole project, which is being held in the School of Engineering in co-operation with the KA, includes the entire process required to create a quotation for a car seat-heater. This research attempts to develop a design automation method supporting automated systems for product design, process planning and cost estimation (Elgh and Cederfeldt 2005). Once the geometry is created by the company, the requirements for quotation process such as wiring pattern, wire type and length, material data and cost estimation etc. are met by the project itself.

The aim of this thesis is to create an algorithm for the wiring pattern of the car seat-heater. Combined with the ongoing research project in School of Engineering, this study will reduce the manual and time-consuming work held in the company. Moreover, the solution will result in shortening the quotation lead-time and prevent hand-made mistakes by means of an optimized pattern.

This work will be performed in CATIA V5 via using the Visual Basic macros to create a generative design. The algorithm will only be parametrically dependent to the shape and size of the seat. Since the solution is obtained by the algorithm itself, the user dependency is limited only with parameter
inputs which define the 2D outline of the heater. By this way, the necessity to the program knowledge is disabled.

2 Theoretical background

The development of the systems in automated design process is achieved by implementing the frame that combines knowledge processing with information handling. The design process is handled considering the customer requirements defined as product specifications. These dependencies enable the designers to generate new designs in order to facilitate producibility and decrease cost and production lead time etc.

2.1 Design and process

Design knowledge has been adapted to human’s society when they accepted the products with multifunctional acts in their life. By growing of modern societies, engineering design becomes more dominant in lives. Thus, design engineers make designs more applicable. The design engineer performed both engineering and low level management work and was in a position to gain familiarity with the organization and the key people. (Ertas.A and Jones.J.C 1996)

Design concept is the new systematic frame of the knowledge to solve the variety of requirements by creating the solution in assist of design, manufacturing and assembling. The design process begins with an identified need and concludes with satisfactory qualification and acceptance testing of the prototype (Ertas.A and Jones.J.C 1996). In order to achieve the computable products price in the market and to satisfy the customer specifications. Most companies are interested and adapted design engineers to order for their profitability and to be able to respond quickly with competitive prices and a short product delivery time (Elgh 2007).

In the competitive product environment, cost and manufacturing lead time with respect to customer requirements are the main basic constrains that have to be reduced in production process. It is required to find the new systematic way that has confirmed in the product development to achieve this goal.

The cost in different steps of processes such as design, manufacturing process is relevant. More efforts on design approach are to decrease the manufacturing cost and desirable final product by considering the resource limitation in market.
2.2 Producibility

Producibility is the relevant concept to ensure the companies’ success in reducing the constraints. Producibility awareness implies approach in system level design and detail design phase to enhance the actual manufacture and assembly of the product, the approach encompasses design for manufacture taken a step close to actual manufacturing and assembling of a product i.e. Design for manufacturing (DFM) (Elgh 2007). That is addressed by design automation. Design is made of two scientific part; Design process, and Designed object. They can be automated in progress.

Incorporation of design automation with the design cost, manufacturing cost, and requirements by supporting of effective concept is a helpful procedure to decrease the cost and lead time in whole of the process.

2.3 Design automation

There are many definitions of design automation in different areas in sciences like electrical engineering to concepts and so on in other knowledge. In mechanical engineering, in the case of product development, the definition could be more close in the order of Using the CAD systems and IT knowledge in automating and managing the design.

Design automation reduce the important feature cost, lead time, and manage the large amount of information by using tools, and organized concept simultaneously. Using CAD/CAM tools has lots of efficient to achieve the goals. Other advantages are succeeding in:

- Reduce design life cycle that makes products serviceable to other purposes.
- Choice of the design, manufacturing process of product is improved.
- Cut the number of testing down to increase the accuracy.
- Making the congenial of the design specification and customer requirements.
- Providing easy feedback to meet manufacturing ability limitation.
- Simplifying the modification on the product design from the feedback requirements.
- To understand the new methods by accomplish and use of computer’s programming and CAD software to improve producibility.

Utilizing the design automation to meet automated design variant in the case of using modeling and managing the of the knowledge are two important aspects for increasing life cycle requirements in
design automation, and the result to use as knowledge base to achieve the desirable requirements with the respect to customer specification.

Figure 2.1- Model for design automation handling

3 Approach

As it is mentioned above the scope of the whole project is to generate variant designs for seat heaters according to the customer specifications. This thesis project is concerned with the creation of wiring pattern of seat heaters.

3.1 Computational implementations

Writing an algorithm for the wiring pattern shows up with the need of computer tools usage. Selected tools are those that the company is using in design process and will be utilized to create the algorithm.

KA is using the CATIA V5 as the base tool for designing and managing the product.

3.1.1 CATIA V5

CATIA V5 is one of the most common CAD software packages provided by Dassault Systems which is leading solution for product design and innovation. The abilities of this software nowadays are growing and supported company trying to add more features and making it more powerful in the
area of engineering. Some of the abilities can be mention as design virtual, manufacture to simulation, analysis, assembly etc.

### 3.1.2 Knowledge Advisor

CATIA V5 Knowledge Advisor (KWA) is one of the useful parts which allow design engineers to capture their knowledge and reuse it as best practices. Some features of the knowledge advisor from Dassault website are:

- Captures and highlights engineering knowledge as embedded design specifications
- Provides easy definition and understanding of know-how
- Leverages knowledge capital to automate design tasks
- Leverages know-how to guide and assist you through the different design stages
- Shares and ensures compliance with design rules and constraints

### 3.1.3 Macros

Macro automates creation of semi product and makes user work more effective (CATIA macros, 2010). The macros in this system allow managing and recording the design in CATIA V5 system codes. And allow customers to run and customize their design. Some other gains from macro:

- Modifying every step of design
- Automating and managing of the work
- Creating the parameter
- Specifying desire customer’s requirement

### 3.1.4 VBA

VBA (Visual Basic for Application) is a powerful interface programming language VB6 linked with integrated develop environment that has been used for creation the objects and methods in CATIA V5 and it has graphical user interface. VBA is interacting with CATIA V5 and accessing to its features. In addition, it is one of the most useful tools in design automation with combination of the CATIA V5.
4 Presently used method

4.1 Creating outline from the model

The method that has been used for wiring pattern design within KA is implemented manually by hand. The customer company expressed its interest for a seat heater with a new car seat model. The 3D model of the new design is sent to KA automotive representing the shape, dimensions and parts needed to be heated.

![3D model of a seat](image1)

**Figure 4.1- 3D model of a seat**

Once the 3D file is received from the customer, a 2D outline is created in KA to see the dimensions of the parts to be filled with heating elements.

![2D outlines created from the 3D model](image2)

**Figure 4.2- 2D outlines created from the 3D model**

Depending on the customer specifications, the constraints for design are defined and the new pattern design is obtained manually by hand. The wire type, length and the pattern is selected in the way that the requirements of the customer are clearly met.
This process is followed by a quotation and sent to the customer, including cost estimation and production lead-time etc.

### 4.2 Automated design and geometry definition

When carried manually by hand, wiring pattern design is time consuming and not accurate enough for the quotation process. To avoid such disadvantages, an automated design is created at the moment but as it is mentioned above, it is not sophisticated enough.

The design objective is to distribute the heat all across the heating element equally when it is filled with wires. The center-line of the pattern is defined according to this concept. The center-line should be created with as possible as less wire usage such that the wire cost is reduced. Apparently the number of the curves used in center-line should be also minimized in order to simplify producibility.

The design concept also includes the constraints such as minimum distance between the wire and the edges and the minimum distance between the wires. Figure 4.4 illustrates such a sketch below:

![Wire pattern with constraints](image)
Following the implementation of the automated design to one element, new sub elements are defined with the ditches (bridges between the elements), and the new sketch is created in CATIA V5 as shown in figure 4.5 below:

![Figure 4.5- Sketch with sub elements](image)

In addition to the sketch obtained with the implementation of sub-elements, constrains are applied to the sketch to define the start and end points, as well as the transition points through the ditches.

![Figure 4.6- The final sketch that the algorithm will be applied](image)

The result will end in a center-line, passing through those points created by the algorithm.
5  Problem analysis

5.1  Application system developed

ProceedoStudio is an application program developed to test and analyze the concepts, in the school of engineering cooperating with KA. The program is used for estimating the wiring pattern, wire properties, manufacturing data and the manufacturing costs of a new concept. The program uses and solves the dependencies between the variables at one runtime such that the solution is a possible new concept. The main structure that the system uses is described in figure 5.1:

![Figure 5.1 – ProceedoStudio main structure that system using](image)

The application stores the database and different projects relevant to the database. The developed structure for KA is used to create a knowledge database for different kind of possible projects for a seat heater. Once the constraints and the input variables are defined by the user, Excel and MathCAD sheets are executed to calculate the necessary parameters for heat seat design. The necessary objects are fired sequentially as a result of interface engine that is used by the application. Consequently wire type, length, pattern and other parameters are defined by the application in order to manage design automation for the seat heater. The main screen of ProceedoStudio is illustrated in figure 5.2:
The output data gained by ProceedoStudio is realized as an input data for knowledgeware workbench in CATIA V5. Knowledge ware uses rule and parametric dependencies to manipulate the input data and create different design variants to get optimum solutions. CATIA V5 environment uses macros to automate the design process by recording and running the process to prevent repetition for a given task. The aim of this thesis work is to create an algorithm in Vba (Visual Basic) environment that uses the input data and constraints to create the optimum solution for the wiring pattern combined with the information gained by ProceedoStudio. The execution control environment can be seen below in the figure 5.3:
5.2 Design process

In general the design process is concerned with the manipulation of customer inputs with developed application systems in order to get an output creating an automation combined with different solutions for a given problem. The design process includes the database, the interface engine used to manipulate the inputs and turning them into parameters, knowledge base systems working with the parameters gained and eventually an output which is supposed to be a possible solution. The main structure of a process with general sections is described in figure 5.4:

![Figure 5.4 - main structure of design process with general sections is structured](image)

5.2.1 Specific problem for KA

In our case process is concerned with the seat heater design. The company, KA, shows its need for such a design process depending on the facts described below:

- 75 new variants constructed annually
- 12 months of lead time for design process is spend
- 4-6 months construction time for each new design is needed
- Product is defined approximately with 160 different variables.

As it is mentioned before, to shorten the design lead time and to achieve more accurate estimations, ProceedoStudio (design automation application program) is being implemented within the Jönköping University.
5.2.2 Creating the layout

Once the 3D file is received from the client, the seat model is converted to 2D model keeping the original dimensions and shape. By this way, the structure of the seat part, which is going to be settled with wires to obtain heating, is clear and executable. The schematic of the process is shown below in figure 5.5:

![Creating layout](image)

Figure 5.5 - Creating layout

5.2.3 Design structure matrix

Considering the different variables and the complexity of the system, Design Structure Matrix is created and so the problem is minimized to smaller loops in order to evaluate and to solve the problem easily.

The Design Structure Matrix (DSM) is a simple tool to perform both the analysis and the management of complex systems. It enables the user to model, visualize, and analyze the dependencies among the entities of any system and derive suggestions for the improvement or synthesis of a system.

For this problem, the electrical and geometrical sequence is defined separately as a result of DSM. The schema of the system is shown here and the centre wire pattern for a squab is highlighted to emphasize the topic of the project.
The variables are evaluated and the necessary parameters are gained by the interface engines used in ProceedoStudio such as MathCAD.

In brief MathCAD is computer software primarily intended for the verification, validation, documentation and re-use of engineering calculations. Such properties enable the program is very useful in the design process.

The MathCAD worksheets used for calculations are defined below in titles:

1. Associated Design Parameters
2. Current Consumption
3. Cushion Resistance
5. Design Input Parameters
6. Harness Power Consumption
7. Harness Resistance
8. Heating Wire Amplitude Factor
9. Heating Wire Length
10. Manufacturing Plant
11. Squab Power Consumption
12. Squab Resistance
13. Total Power
14. Total Resistance
15. Wire Selection
5.3 Design variants

The customer requirements in our case are seat heater wiring patterns for different variants, which are prepared by KA. The company works with so many varieties of car seat layouts from several cars, as their own customers.

The process of layout achievement is explained above in details. These layouts have different shapes and the ideal solution aims to fulfill all the area of the layout. The company is doing the wiring manually by finding the concept, and tries to implement it to the layout then if the design doesn’t meet the requirements then they implement another concept. Figure 5.7 illustrates an example of the car seat pattern which already was prepared by the company.

![Figure 5.7 - example of the car seat heater pattern](image)

Respect to these constrains, we are going to fulfill the pattern. And the goal is to cover the some areas that they have difficulties to fulfill in automation.

![Figure 5.8 – example of car seat heater layout](image)
The current solution is suitable for the rectangular shape. Almost all the areas that are casual, like that has covered by curve, have problem to fulfill with wires.

Variety of the layouts:

The company has so many layouts to work with them here we choose some random variants to implement the algorithm.
Figure 5.10 –Samples of different car seat heater layout

Figure 5.11 – One sample of car seat heater layout to show starting wiring pattern

The output would be design of wire in the selected car seat patterns with respects to constrains.

Figure 5.12 – Sample of symmetric wiring pattern for car seat heater
According to the diagram of design automation, the customer specification is used as input, (in our case is the car seat layout) and output is the design variant of wiring which suits to the input. What we supposed to do is using the design automation as a tool to reach the automatic design variant of wiring pattern in which we meet the automated design variant advantages in result.

Figure 5.13 – schema of implementing design automation concepts for car seat heater, from customer specification, and using tools to have automated design as an output

5.4 CATIA V5 template, KWA and VBA Macros

CATIA V5 Knowledge Advisor (KWA) allows designers and design engineers to capture their knowledge and reuse it as best practices. Users can embed knowledge through formulas, rules, reactions and checks which can be leveraged at any time. User can make better decisions, exploit intents and reach optimal results through the exploration of design alternatives with regards to rules. It is possible to convert implicit practices into explicit knowledge, thus automating design and reducing risk and cost of repetitive tasks. As an integrated product, CATIA V5 Knowledge Advisor can be used in conjunction with all other products as an illustration of a pervasive knowledge.
For the creation of design automation system, Visual Basic scripts are used. This way manipulating and executing the tasks become easier, faster and prevents us from repetitions. The inputs are used to create outputs by defining the relations and the rules between them. The VBA Scripts are created by recording the macros. This enables the manipulation of parameters. Below in figure 5.14 you can see examples of CATIA V5 template:

![Figure 5.14 – examples of CATIA V5 template](image)

The CATIA V5 template is then used to define structured geometrical input parametrically and Visual Basic macros are created to define the wiring pattern and the critical points, connections such as bridges and curves. Figure 5.15 represents screenshots for VBA macros and geometry:

![Figure 5.15 - VBA macros and geometrical input are given as examples](image)
This project uses the CATIA V5 template used in ProceedoStudio, and structured geometrical input, and defines an automated design for the squad centre wire pattern which is obtained by an algorithm created in Visual Basic.

Before starting programming in Visual Basic, there are some constraints that have to be considered as inputs. As a result, the objective is to define the centre line for an optimum heating across the squad. Inputs and the outputs derived from the system are shown in the figure 5.16:

![Figure 5.16 - inputs and the outputs desired from the system](image)

6 Generative Design

6.1 New idea creation

Some of the possible design variants are represented above dimensionless. Even though there are more than hundreds of different variants, here we are interested on most frequently used ones. Moreover these variants are applicable to any other variants with some modifications. The next task is to decide the method to fulfill the areas seen in layouts. In order to make the design process more clear and modular, we simplify the variants and draw them as they are consist of only
rectangular elements. Initially we put only the star-end points and the transition points as constraints. These points are not in their exact location at the moment but only simulated to emphasize that they exist. The simplified layouts can be seen in figure 6.1 below:

Figure 6.1 - simplified samples of car seat heater layout

There are five different variants simulated above and if it is analyzed carefully it is obvious that the variants are consisting of similar elements which enable us to use same algorithm for the same elements and then combine them for different variants. The elements are named above and these names will be used also in programming from now on.
Even though number (1) and (3) seem to be the same elements, they are actually different from each other since the locations of the start-end points and the transition points are different. This difference is important because of the pattern used in wiring will pass through these points. As the aim of the wiring is to fill the area with as much as wire possible and provide uniform heating, using feasible algorithm to supply an efficient result. However, number (4) and (5) are symmetric as. Therefore, this will help to use same or a little bit different algorithms for those elements and make programming process easier and logical.

### 6.2 Horizontal filling

One possible idea that can be used in filling the elements is using horizontal lines all over the elements. Initially nodes are created according to the constraints inside the element, and then these nodes are connected to each other with horizontal lines. This solution is quite efficient for simple elements such as rectangular shapes but when curves and transition elements are included within the elements, the complexity of the element can create difficulties in implementing horizontal wiring. One example for this kind of filling (in one element) is shown in figure 6.3:

![Figure 6.3 – Example for horizontal filling](image)

### 6.3 Vertical filling

Another idea in filling the inner area is vertical filling which resembles to horizontal filling considering the logic used in connecting the nodes. The only difference is nodes are defined
vertically and consequently the result is vertical lines. This sort of filling can be applied to the rectangular area as horizontal filling does, but the efficiency of the method is not predictable when using this method all over the layout. The presence of the transition points and thin and long elements can restrict the usage of vertical lines due to the number of windings. Another important concern for this concept is complicated shapes cannot be filled with only vertical lines.

![Figure 6.4 – Example for vertical filling](image1)

6.4 Using symmetric lines

The layout is divided into two parts with a vertical symmetry axis, and then the rest is filled with horizontally or vertically according to the structure of the elements. As it is seen in the figure, the center line algorithm is same for both parts. Area is fulfilled uniformly and the appearance seems quite attractive. Considering the advantages of the method, applying symmetric lines for wiring could be logical but apparently there are too many windings within the element which is the result of symmetric line usage.

![Figure 6.5 – Example of symmetric lines](image2)
Considering the manufacturing aspects, too many windings make the machine slow down, so the production is interrupted. As a result, an increase in production lead time is seen which is not desirable.

Since we have to consider the limitations for production, this method has to be evaluated carefully with all its advantages and disadvantages.

6.5 Function of angle

The sine function $\sin x$ is one of the basic functions encountered in trigonometry. The function has sinusoidal shape with possibility to control amplitude of the graph. The general form of the function is shown below in figure 6.6:

$$F(x) = a \sin (bx + c) + d$$

![Figure 6.6 – sinusoidal graph](image)

The concept is to implement the sinusoidal shape in car seat layout as the wiring pattern by using the sine function. The amplitude of the graph is controlled with respect to constrains. Using the sine function for pattern creation of car seat heater, could be the one of the effective concepts that makes design easier.

![Figure 6.7 – sinusoidal implementing on rectangular pattern](image)

For the beginning of the line the amplitude of the graph is half of the whole amplitude of the graph.


\[ B = \frac{a}{2} \quad f(x) = B\sin(2\pi) \]

![Figure 6.8 – Different Patterns of wiring pattern (using function of angle).](image)

For the first sinus shape the equation can be written as:

\[ F(x) = B\sin(2\pi) \]

\[ F(x) = \sum_{a/2}^{a} \sin(2\pi) \]

The result is achieved by number of iterations. As conclusion, concept is chosen to show possibility of designing wiring pattern by formulating the shape of design. This concept could work efficiently in simple shapes but for complex shape it needs to be studied more in advance. Implementation of the concept to car seat layout requires high level of research.

### 6.6 Step function:

Function of the real numbers is step function. A function \( f: \mathbb{R} \rightarrow \mathbb{R} \) is called a step function if it can be written as a finite linear combination of semi-open intervals \([a, b) \subseteq \mathbb{R}\). Therefore, a step function \( f \) can be written as it is defined in Mathworld, 2010:

\[ f(x) = a_1 f_1(x) + \cdots + a_n f_n(x). \]

Where \( a_i \in \mathbb{R} \), \( f_i(x) = 1 \) if \( x \in [a_i, b_i) \) and 0 otherwise, for \( i = 1, \ldots, n \).

A function \( f \) defined on an interval \([a, b]\) so that \([a, b]\) can be partitioned into a finite number of subintervals on each of which \( f \) is a constant, Also known as a simple function and more general a real function with finite range.
Heaviside step function:

The Heaviside step function is a mathematical function denoted \( H(x) \), or sometimes \( \theta(x) \) or \( \Pi(x) \) (Abramowitz and Stegun 1972, p. 1020) and also called the unit step function. The term Heaviside step function can represent either a piecewise constant function or a generalized function, (Mathworld, 2010).

### 6.6.2 Piecewise constant function

A function is said to be piecewise constant if it is locally constant in connected regions separated by a possibly infinite number of lower-dimensional boundaries (wolfram, math world). When defined as a piecewise constant function, the Heaviside step function is given by

\[
f(x) = \begin{cases} 
1 & \text{if } x \in [a_i, b_i) \\
0 & \text{if } x \notin [a_i, b_i) 
\end{cases}
\]
When defined as a generalized function, it can be defined as a function \( \theta(x) \) such that

\[
\int \theta(x) \phi'(x) \, dx = -\phi(0)
\]

for \( \phi'(x) \) the derivative of a sufficiently smooth function \( \phi(x) \) that decays sufficiently quickly (Kanwal 1998). <http://mathworld.wolfram.com/PiecewiseConstantFunction.html>.

### 6.6.3 Rectangle function:

The rectangular function (also known as the rectangle function, unit pulse, or the normalized boxcar function) is defined in Wikipedia, 2010 as:

\[
\text{rect}(t) = \mathbb{1}(t) = \begin{cases} 
0 & \text{if } |t| > \frac{1}{2} \\
\frac{1}{2} & \text{if } |t| = \frac{1}{2} \\
1 & \text{if } |t| < \frac{1}{2},
\end{cases}
\]
It is a simple step function. Alternate definitions of the function define rect to be 0, 1, or undefined. In some resources Rectangular function is considered in terms of the Heaviside step function<http://mathworld.wolfram.com/RectangleFunction.html>.

The step function is being used in different areas of science to evaluate and predict the results. For example in electrical engineering field to study on signals, and in Design automation we are more interested in graph result of step function and use it as a concept in design environment. Considered in this specific subject (car seat heater), for simple shapes it is possible to use step functions and have rectangular shape. In the beginning the graph looks like a series of small steps. Further study is needed for implementing the step function in complex layout shapes.

6.7 Honey comb:

In this concept, implementing the honeycomb pattern to the car seat heater layout geometry makes wiring in hexagons.

![Figure 6.13- Honeycomb wired pattern](image1)

In order to satisfy the constraints, we can implement the honeycomb geometry with specific distance from the edges and also give the dimensions to pentagons; in other words first we can define honeycomb geometry by dimensions and then locate it in the layout.

![Figure 6.14 – Honeycomb pattern](image2)
6.7.1 Quadratic elements

The idea of honey comb returns to another concept which is involved in meshing the layout with quadratic elements with respect to min. wire distance and min. edge distance. This way the nodes are defined and connecting the nodes with the lines will result in a pattern which is more efficient in use compared to the honey comb.

![Figure 6.15 – Quadratic element wired pattern](image)

In this case the rectangular pattern is more effective. It is possible to control edge distances also.

![Figure 6.16 – Quadratic elements pattern](image)

This concept that has presented to create a pattern for car seat heat wiring is difficult to meet the requirements with respect to all specifications. Implementing the honeycomb or rectangular concept needs further studies in the future.

6.8 Space filling-curve:

In mathematical analysis, a space-filling curve is a curve whose range contains the entire 2-dimensional unit square (or more generally an N-dimensional hypercube). This also called Peano curves. It is a fractal curve which can be written as a Linden Mayer system.
A Linden Mayer system invented by Hilbert (1891) which one of its limitation is that a plane-filling curve, fills a square. Traversing the polyhedron vertices of an $n$-dimensional hypercube in Gray code order produces a generator for the $n$-dimensional Hilbert curve. The Hilbert curve can be simply encoded with initial string "L", string rewriting rules "L" -> "+RF-LFL-FR+", "R" -> "-LF+RFR+FL-", and angle $90^\circ$ (Peitgen and Saupe 1988, p. 278).
A Linden Mayer system, also known as an L-system, is a string rewriting system that can be used to generate fractals with dimension between 1 and 2. Several example fractals generated using Linden Mayer systems:<http://mathworld.wolfram.com/LindenmayerSystem.html>:

![Space-Filling Curves](image)

**Figure 6.20 – Generating Hilbert ‘space-filling curve’**

It needs more study to implement space filling curve in designing of the pattern.

### 6.9 Slice and create points

This concept is especially important for curved shapes and also applicable to all other possible shapes and it is involved in dividing the layout with equal distance lines and implements the nodes on them in two different ways:
In the first method we define specific number of nodes on the each line and in the second one we define nodes as much as possible with respect to constrains, minimum edge distance and minimum wire distance.

Figure 6.22 – example of slice and create points’ method, in this figure number of nodes in each line is same (four nodes in each line)

Connecting the nodes with the lines will result in a pattern following the original shape with respect the constraints.

6.10 Offset lines

The ideal solution in creating pattern for different shapes might be offset lines.

By this way the pattern follows the original shape. The application is achieved by offsetting the exact shape of the layout with respect to constraints (min. edge and wire distance). Afterwards the nodes are created on those lines such that allowing us to create the pattern as a continuous line.

An example is shown in figure 6.24:
Possible obstacles for this concept show up in programming phase. Considering the start and end points as constraints and the necessity of using a continuous line for the pattern, the application of the concept becomes harder.

However, this concept when combined with slicing and creating nodes on the lines can create a good solution for the problem which will be investigated in the following phase.

7 Critical parts

The methods which can be used when developing the algorithm are described briefly above. As it is mentioned before there are some constraints (start and end points, minimum edge distance, minimum wire distance, maximum and uniform fulfill, minimum winding etc.) that we have to consider in algorithm creation process. Before we continue with the evaluation, we have to define the critical parts in the structure. Identification of these parts will guide us to decide the method which will be used in algorithm creation. The critical parts involved in the seat heater structure are curves, transition points and the usage of the return wire:

7.1 Curves

At the beginning of the section the elements in a layout are represented and handled as rectangular elements. However mostly the elements are not rectangular which is an important point and has to be taken into consideration when programming. The nodes have to be defined according to the curve and the wiring path should be created that as much as filling is supplied. Otherwise it is possible to have a gap between the wiring and the edges that have curved shape. Since the aim is to have uniform and as much as possible wiring, the curves have to be handled carefully to have a good result in wiring.
7.2 Transition points

Another critical point in a layout that has to be considered is the location of transition points. Most of the layouts are consisting of more than one element, and consequently these elements contain transition points as well. The wiring pattern has to follow the transition points with a specific angle. The locations of the points limit and guide us in wiring type selection and the node creation. As a result the path of the wiring is defined such that the lines created between the nodes cross through the points once, and as these points behave like start and end points, all the wiring process is affected by the transition points.

7.3 Return wire

A return wire is the continuous line from top to the bottom (end point), crossing through the transition points while filling the area as well. It is desirable to include a return wire in order to reduce the complexity of wiring. Once the nearest nodes to the edges are created, return wire is defined with connecting these points in order to reach the end point. The rest of the filling becomes simpler by this way.

However, usage of a return wire is dependent to the locations of the transition points, so it should be clarified which of those points are going to be used and which will be ignored. By this way unnecessary wiring is prevented and some ditches are left wireless as a good development.
7.4 Solutions to the critical parts

The presence of a curve in an element is introduced as a critical point and defined above. The filling of a curved element should be done by curves.

In order to create curves, after the layout is defined in CATIA V5, we record a macro of the curve and check the used commands of curve creation that VBA uses. Afterwards these commands can be used for creating curves between two points. Furthermore, when a curve is subjected in an element, the filling lines can have the shape and the path of a curve with an increasing factor of a radius. This way the nearest wiring will have the same radius with layout and the last one will be horizontal. The algorithm will be created containing this solution.

It is mentioned that the transition points are very important since they are the connection points between two different elements. The most important point that has to be clarified in this part is to define which of the points will be used and which ones will be ignored in wiring.

Figure 7.1-Connecting transition points

Return wire usage is a critical part but not considered as a problem to be solved. Usage of a return wire, the reason that makes it desirable to use, and the benefits are explained above. The implementation of the wire will be investigated in programming phase.

8 Evaluation

In previous sections the possible wiring methods, usage and implementation in an element are described. Evaluation of those ideas should be handled carefully in order to avoid redundant work on the process.
The usage of Step function to create rectangular mesh, Fourier series and Honeycomb model is not quite mature considering the CATIA V5 and VBA programming aspects. Since they need further studies to be applicable for this project, the concepts are not studied in programming.

Symmetric line usage could be a good solution in the point of algorithm creation but when production process is considered, this concept fails to be a good solution due to the number of winding. The presence of the windings should be avoided or at least minimized due to the manufacturing difficulties as it is already an input criterion for pattern creation.

In this case, two possible methods are left; horizontal and vertical pattern creation.

Vertical pattern creation is not applicable to the side elements which are narrow and long in shape. Applying this method will fulfill the area but as a result there will be a lot number of winding as it is illustrated to the right.

![Figure 9.1-Example of vertical wiring on side element](image)

Consequently vertical lines could be used for some elements but not in whole layout due to the presence of side elements.

Remaining methods that have to be evaluated are horizontal wiring, slice and create points and offset lines. These methods seem to be sufficient solution for the problem since they are applicable to all elements.

For further evaluation, the solutions are compared to each other according to the critical functions and afterwards decision making process is defined as matrices.

### 8.1 Critical functions for proposed solutions:

For a sufficient comparison, critical functions for proposed solutions are defined and evaluated as below. Solutions are graded according to the ability to meet requirements.
**Maturity:** The proposed solution for the problem should not be a very new idea such as a new created theory. The use of such concept can be very complex and tough for implementation. Not only creation process but also practicing method should be effective and easy from the user point of view.

**Being compatible with constraints:** Investigating that if the suggested solution satisfies the constraints such as: minimum wire distance, minimum edge distance, minimum radius in winding, etc.

**Applicability of the solution to the different shapes:** Means that the solution is supposed to be applicable to different variants and shapes, not only specific shapes. In other meaning the pattern should follow the outer counter of the elements.

**Ability to fill the inner area of the element:** The solution should fulfill the area as much as possible in order to achieve maximum wiring.

**The homogeneity of the filling:** The wiring would not be concentrated in some areas. The solution should supply a homogeneous filling for uniform and optimum heating with respect to constraints.

**Manufacturing aspects:** Even if the result gained from the application satisfies the functions above, the manufacturing aspects can limit the usage of the solution. The wiring pattern should be able to be manufactured with the least cost possible, which implies a fast an automated manufacturing. For instance the pattern should have as less as winding and irregular movement as possible.

### 8.2 Decision making process

This process is carried out with the help of decision making matrices. One of them is called Pugh matrix, which is created by Stuart Pugh, used for comparing the new concepts to the existing one. And the other is called "Weighted Sum of Attributes" decision making matrix, which helps to make a decision between the concepts by weighing the criteria.

The Pugh method represents a matrix with the critical functions located as rows of matrix and the columns representing the new concepts. The first column is the reference column representing the currently used method which is called datum. New concepts are listed in the other columns. The comparison between the datum and the concepts are made by ‘+’, ‘−’, and ‘S’ symbols. If the new
concept seems to be better than the datum we put a ‘+’, if seems to be worse we put the sign ‘−’, and if it is quite same, the ‘S’ sign is used. The last row is used to sum the ‘+’, ‘−’, and S signs up. The concept with most ‘+’ signs represents the most effective idea.

In defense of this matrix approach to engineering design Pugh states, "The matrix does not make the decisions: it is simply a procedure for controlled convergence onto the best possible concept and is not composed for absolutes in the mathematical sense; the decisions remain with the user" (Pugh 1990).

The "Weighted Sum of Attributes" method weights the criteria and the concepts are summed according to their related values in order to make a ranking among design concepts. The critical functions are listed in rows with a weight factor and the concepts are in columns including one more ‘weighted value’ column for each concept. The method helps to identify redundant design concepts and remove them from further consideration.

The "Weighted Sum of Attributes" decision matrix is an important decision tool, but its limitations need to be well understood by the decision maker.

The matrix is illustrated in figure 9.2:

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Weights</th>
<th>Concept Variants</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Concept 1</td>
</tr>
<tr>
<td>Value</td>
<td>WV</td>
<td>Value</td>
</tr>
<tr>
<td>Sum:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Weighted Sum:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 9.2-Sample of Decision matrix**

**8.2.1 Implementation of matrices**

In this chapter, we will describe the implementation of decision making tools to our problem. The Pugh method and the Weighted Sum of Attributes matrices are applied to our problem to make the decision process more clear and understandable.
The figure on the right shows us the process being held at the moment. This concept is used as Datum (reference) in the “Pugh matrix”. As it is seen the concept is consist of horizontal and vertical lines. The proposed solutions, described above in “idea creation” chapter are compared to this concept with respect to the critical functions. The matrix is illustrated in following chapter.

8.2.1.1 The “Pugh Matrix”:

The applied form of our problem into Pugh Matrix can be seen below in figure 9.1

<table>
<thead>
<tr>
<th>FUNCTION</th>
<th>CONCEPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>compatible with constraints</td>
<td>horizontal filling</td>
</tr>
<tr>
<td>applicable to different shapes</td>
<td>S</td>
</tr>
<tr>
<td>area fulfill</td>
<td>_</td>
</tr>
<tr>
<td>homogeneity</td>
<td>S</td>
</tr>
<tr>
<td>maturity</td>
<td>S</td>
</tr>
<tr>
<td>manufacturing ability</td>
<td>S</td>
</tr>
<tr>
<td>Total plus (+)</td>
<td>0</td>
</tr>
<tr>
<td>Negatives (-)</td>
<td>1</td>
</tr>
<tr>
<td>(S)ame</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 8.1-Pugh Matrix
If we analyze the matrix and define the results we can see that;

Horizontal and vertical filling have (naturally) nearly the same properties with the concept. Symmetric line usage is not efficient due to the lack of manufacturing ability which is an important aspect.

Step functions method, honey comb, and space filling curves don’t have satisfactory advantage over the reference concept and not enough mature to be considered.

Slicing and creating point’s concept and the offset lines method can be called as the most advantageous ones when compared to datum.

8.2.1.2 **Weighted Sum of Attributes Matrix**

As it is described above, this matrix is used to define a ranking between the concepts with respect to critical functions.

Here in the matrix the functions are identified with a weight factor (WF), depending on the influence they have on the solution. Weight factors have the values between’ 1-10 ‘and defined in WF column.

The concepts have two values defined; first one is weight (W) which is used to determine the sufficiency of the concept with respect to criteria, and it is described as:

1- Low 3-Medium 5-High

The other is the weighted value (W\*WV), which is the result of weight factor multiplied with weight (W*WV).
Eventually the results show us symmetry lines, step functions method, honey comb method, and space filling curves have low maturities to use them in this research. It needs more studies to make them applicable. Although offset lines concept seems to be the best solution for the problem as a design perspective, but not to be sure it is feasible in programming.

Since there is no big difference between the rests of the concepts, a combination of all the methods can also be used to create the solution. Different algorithms will be created, implemented in CATIA V5 and the solution will be examined. This process will be repeated till the optimum wiring is achieved.

### 9 Programming and CATIA V5 implementation of the concepts

#### 9.1 Trial process in programming

The results gained from the decision making matrices lead us to concentrate our studies more on creating a pattern within the layout such that the outer contour is followed and uniform area fulfill...
is satisfied. In order to achieve our goals we used three different methods for the creation of macros and programming in VBA. These methods are listed below:

9.1.1 Scaling the layout

The first attempt in programming was made to the concept scaling. This concept refers to a linear transformation that shrinks the structure of the elements by a scale factor with respect to constraints. First scaling is dependent to the Min Edge Distance and the rest are dependent to the Min Wire Distance.

Scaling of the outer shape gives us a homogenous and uniform filling as seen in the figure. Before starting to create an algorithm, the very first consideration was to make a continuous pattern. The illustrated pattern is shown in the figure 9.1.

When starting to create algorithm we faced some objections during the process. These were:

- Creation of the points for vertical lines.
- Transition points position
- Applicability to all elements.

To obtain a continuous line we needed to include vertical lines to connect the scaled geometries. To do so we had to fix points on the lines. Those points could not be determined since there was no constrain or reference for those points. As it is seen from the figure transition points’ positions are not stable and chosen randomly. Which means this selection can interrupt or not be aligned with other elements’ transition points. Another objection showed up when it is considered to be applied to different elements. In other words, the concept can be applied to layouts consist of only one element or to the top element (sub element_2) because of the number of the transition points and their locations. Having more than two transition points will interrupt the pattern and the developed method has to be modified to keep continuity.

Figure 9.1- Example of concept scaling.
9.1.2 Offsetting lines (curves) separately

The difficulties faced in the first concept, made us to switch another concept but keeping the main logic. In this concept the idea was offsetting the lines or curves involved in the geometry one by one with respect the constraints. The redundant lines would be trimmed and so the rest gives us the pattern.

![Procedure of Offsetting lines one by one](image)

**Figure 9.2- Procedure of Offsetting lines one by one**

By this way the outer counter of the geometry is still kept but the wiring method and the pattern is changed. The result is shown in the figure.

![Pattern shows up after trimming redundant lines](image)

**Figure 9.3-Pattern shows up after trimming redundant lines**

The result manually gained by implementing the concept in CATIA V5 was sufficient for creating a pattern which fits the outer counter of the element one on one. However, when creating the algorithm the objections showed up and center line pattern became unable to program in VBA.

The most challenging objection was related to trimming lines. It is required to define which lines were going to be used in pattern creation and which were redundant. There was no reference for the purpose so that the lines separation could not be achieved.

In order to define a distinction between the lines could be defining points on the curves. By this way the redundant lines would be eliminated and the rest lines and curves would keep giving us the center line pattern. Following attempt is concentrated on point creation on the curves and lines.
9.1.3 Points creation on the lines and curves

Even if the first two attempts in modeling and programming were supposed to give sufficient area fulfill in result, the objections in programming made those concepts inapplicable. The experienced studies lead us that the solution for the problem could be creating points on the edges (independent from the shape) with respect to constraints, and then connecting those points by projecting (offsetting) the edges. This method refers to a combination of tested methods in order to achieve the main goal of the project.

The method is illustrated in figure 9.4 with different cases. The constraints Min Edge Distance and Min Wire Distance are labeled. The positions of points which will be used for wiring are also dependent to constraints. This operation is followed by creating the inner points needed for connecting the curves of the pattern and the return wire as well. The upper screenshot indicates the definition of the points and the screenshots underneath are used to illustrate the different possible shapes for an element.

![Illustration of the different possible shapes for an element](image)

**Figure 9.4- Illustration of the different possible shapes for an element**

The distance between the points has to be defined according to the constraints. The distance from the edges and the points distances are able to be defined separately, however this is distance could be defined as length of the line (or curve). In other words, constraints cannot be defined as vertical or horizontal distances.
The attempts made in trial process shows the importance of point creation in the element. To be able to create points, a reference should be determined. After the method offsetting, “slice and create points” method seemed to be second best solution but we are more focused on creating the points considering the axis system as reference. In the following chapter this method will be described in details. As a result either due to the experience gained from trial process or other limitations such as maturity and limited time this method didn’t take place in trial process and so in programming. We skipped directly using horizontal and vertical lines as filling method and the axis system as a reference.

9.1.4 Using horizontal and vertical lines

As separate methods, either horizontal or vertical lines can be used for fulfilling the inner areas in the elements, but considering the shapes of the elements combined with constraints we have, the necessity of using both methods would be a solution for the problem. The reference point(starting point) for the concept is the axis system located in the mid-bottom of the main element which makes it possible and efficient to create inner points needed for pattern creation. The procedure and solution we got is explained briefly for all concepts one by one.

In this effort we got solution to solve the problem by creating lines. We conduct procedures first to create points and then connect created points by horizontal and vertical lines. The result for layout is the area fulfill with horizontal and vertical lines. In figure 9.6, the results are shown:

Figure 9.5- CATIA V5 considers the length of the line as distance
The procedure follows CATIA V5 template that have to be set by designer and continuing by implementing of VBA codes to reach the proper solution for each elements layout.

In next sections we are going to show the wiring pattern solution for one, two, three, and five element 2D layout. And more talk about CATIA V5 template and programming phase in each of the solution.

9.2 Programming and implementation in CATIA

9.2.1 Car seat heater wiring pattern in one element 2D layout:

In this solution we are discussing about wiring pattern in one element 2D layout. The procedure is begins by importing of geometry layout by designer. It is shown below in figure 9.7:
The next step is to set and arrange the CATIA V5 template in proper structure and give suitable name in each order geometrical sets in which specific names already has been define in codes. In figure 9.8 below, CATIA V5 template for one element is shown:

**Figure 9.8-CATIA V5 template for one element structure**

When creating the base sketch, the geometry of the sketch is constrained and parameters are defined to make it simply modifiable in the macro.

**Figure 9.9-CATIA V5 template and constrains for one element structure**

Here the parameter Main_El_Width_2 is used to control the upper edge width of the element.
In Main Element (order geometrical set) as a sub element of Elements, we create a sketch of lines in which this sketch is projected from the base sketch that imported by designer and then in CATIA V5 we join projected lines in order to be adjacent. In Main Element sub tree we set one order geometrical set to Connection Point _Main in order to set Start and End points.

![Figure 9.10-Projecting main element from base sketch](image)

The last element in template tree is, to add one order geometrical set as sub tree of Elements and change its name to Wiring Pattern. After fix every parameter to proper position and name then run the program. The solution to wiring pattern for one element in 2D layout is shown in figure 9.11:

![Figure 9.11- results of wiring pattern in one element 2D layout](image)

By Implementing of VBA codes, first the program will create start and end point, and calculate the number of points on the right side and left side by respect to Edge Dist and Min Wire Dist. After creating points, left side circle horizontal and vertical lines will be creating. The result will be wiring pattern in car seat heater 2D layout.
9.2.2 Car seat heater wiring pattern in two elements 2D layout:

In this section we are going to show the solution for creating wiring pattern in two element 2D layout. Almost procedure is the same with one element but still there are some differences. In this case two bridges exist and in the solution must satisfy passing through them.
Like one element after importing the geometry, the CATIA V5 template has to be arranged in proper way like in figure 9.15 below:

**Figure 9.15-CATIA V5 template for two elements 2D layout**

In this case the template Elements tree includes Main Element, Bridges and sub Element. The Bridge element includes Bridge_1 and Bridge_2. The Sub Element includes sub element geometry sketch and sub element connection points.

One important issue is measuring of these two or more elements. First we set measuring in total geometry and after testing our concept we found it is not suitable solution to control geometry size, and then we decide to test another concept that is measuring of each element one by one in each element. Measuring of two in this way is more feasible. Figure 9.16 shows the measuring of each element:
Figure 9.16-Constraining two elements structure

Each element has its own height and width we named each of them like: Main_El_Width, Main_El_Width_2, for main element and for sub element Sub_El_1_Width and Sub_El_Width_1, for bridges we call them Bridge Width and Bridge Height.

We make sketch of each element by projecting of line from original geometry and we use JOIN command in CATIA V5 to make them adjacent.

Figure 9.17- Projecting two elements structure from base sketch
After making every ordered geometrical set in proper name and position in CATIA V5 template, we open the VBA codes for two elements and run it. The program first creates points with respect to constrains and then creates lines and circles. The result will be wiring pattern for two element car seat heater 2D layout. Below in figure 9.18, the solution is shown:

Figure 9.18—Pictures show point creation and last result of wiring pattern in two elements 2D layout

One important issue in this effort is about bridges as is shown in figure 9.19. We tried to create Left Transition Point of first element with respect to the right vertical edge bridge one and Left Transition point of second element with respect to first vertical edge bridge one, so it will keep an angle between bridge vertical edge and connected line.

Figure 9.19—keep an angle between bridge vertical edge and connected line on the left side
The procedure to create connection line between two transition points in bridge _2 is same. Right transition point of main element creates with respect to left vertical edge bridge_2 and right transition point of sub element creates with respect to right vertical edge bridge_2.

![Diagram](image)

**Figure 9.20- keep an angle between bridge vertical edge and connected line on the right side**

### 9.2.3 Car seat heater wiring pattern in three elements 2D layout

In this section we are going to solve wiring pattern for three element 2D layout. This problem is carried by one main element two sub element and four bridges. As previous solutions, after importing geometry, the CATIA V5 template must be arranged in proper name and position.

![Diagram](image)

**Figure 9.21-Three elements structure 2D layout and CATIA V5 template**
Measuring of this case follows as like as two elements, each element has measured by its own height and width. Figure 9.23 shows measuring of three elements car seat heater:

After setting each geometrical set in proper name and position we implement our VBA codes and the result will be wiring pattern in three element car seat heater 2D layout.
Figure 9.24- wiring pattern on three elements 2D layout

As creating wiring pattern, the start, end and transition points of elements are created in connection points as order geometrical set in each element.

Figure 9.25-Defining Connection Points in CATIA V5 template
9.2.4 Car seat heater wiring pattern in five elements 2D layout

In this section we are going to design wiring pattern for five elements in car seat heater in 2D layout. This follows the same methods as for 3 elements. The differences are two back sides are added. Side elements are symmetric that makes procedure easier. The procedure follows by preparing CATIA V5 template for designing.

![Figure 9.26-five elements 2D layout and CATIA V5 template](image)

Template tree includes Elements and Wiring pattern. In Elements we define order geometrical sets suit with main geometry and in wiring pattern points and curves will be create automatically by program. In Elements sub tree includes Main Element, Bridge_1, Sub Element_1, Bridge_2, Sub Element_2, Bridge_3, Side Element_1, Bridge_4 and Side Element_2. The procedure continues by creating sketch of each element and bridges.
One of important issue is constraining the geometry. We define elements independent from each other. This method allows designer to control and change dimension of each element separately.
After setting template in proper name and position, we run the VBA codes for five elements and the result is wiring pattern in car seat heater 2D layout.

In CATIA V5 template connection points as start, end points and transition points belongs to each geometrical set of the its element.
9.2.5 Car seat heater wiring pattern in one element 2D trapezoidal layout

In this section we are going to show wiring pattern for 2D trapezoidal layout. In this case the procedure is same with wiring pattern in one element 2D layout. Difference between these two elements is only on geometrical shape.
We solve this problem by defining math relation to create points in the same slope with two vertical sided.

Figure 9.32-creating point along vertical sided

To solve this problem we use triangular similarity. Equation 9.33 introduces the triangular similarity.

By considering Relations below:

\[
\frac{AB}{EC} = \frac{AD}{DE}
\]

Figure 9.33-triangle similarity that is used to wiring pattern in one element trapezoidal 2D layout

By considering Relations below:

\[
AB = \text{Height of the element}
\]
BC = Width_2 – Width_1 (Difference between top horizontal sided and bottom horizontal sided),

AB – AD = Min Wire Dist then DE is a rate of angle, it means to create point in Min Wire Dist height the value DE should minus from BC+ Edge Dist from back sides.

DE is a rate value, will when new point will create. In every vertical position of AB-AD of previous horizontal position the new points will create in along to the slope line and in the same direction be in the same. Measuring of the trapezoidal element layout, we consider top and bottom lines and height of the geometry. Figure 9.34 shows measuring trapezoidal geometry.

![Figure 9.34- measuring trapezoidal element 2D layout](image)

After setting CATIA V5 template in proper name and position, then run the program. The result is shown in figure 9.35:

![Figure 9.35-wiring pattern in one element 2D layout](image)
10 Results

10.1 Managing the solution

The created macros are collected in a main form and the designer uses this form to fulfill the necessary input parameters for pattern creation. The process is managed in several steps:

First the designer receives the 3D model from the seat manufacturer, creates the 2D layout of the seat.

Afterwards, the designer selects the 2D CATIA layout created in this project that fits the shape of the model received from the producer. Opens the file, then base sketch and constrains, in order to see which constrains refer to which dimension.

![Figure 10.1- Selecting the appropriate CATIA model](image)

The next step will be running the macro. Within the macro there is a form with the name Wiring Pattern. The designer opens the form and runs. When the form is run, a tool box appears with the definitions of the parameters that needed to be fulfilled. Designer chooses the layout type on the tool box, and according to the selection some of the input boxes are activated and the rest are deactivated.
Figure 10.2- General view of the tool box

Designer inputs the parameters determined on the original 2D layout (in mm), clicks the button “Design”, thus the pattern is created on the layout as a result.

Figure 10.3- Pattern created according to the input parameters
10.2 Evaluation of the final solution

Final solution that is proposed in this project is sufficient for the designer in terms of pattern design and preparing a fast and reasonable quotation.

The program created is:

- simple, user friendly and easy to use
- independent from program knowledge (user does not have know about programming or modeling)
- reduces design and quotation lead time
- allows the user to work with different types of layouts
- does not follow the outer contour one by one but still the created pattern is efficient and tolerable in terms of wire length

11 Conclusions

Even though this project ended in another point than considered, the result is quite sufficient for the user to create an automated wiring for a defined seat heater layout. The reached point shows that the created design is applicable for different layouts, depending on the height and width of the elements separately.

All the elements in a structure are defined so that they are independent from each other and can be modified in terms of constraints. This means that the user is free to change the height and width values of the shapes separately in a structure, thus the layout is modified and is re-dimensioned. Once the program is ran, wiring pattern is created, according to the nodes in the new layout.

As the scope of the project is to find the best solution for wiring that follows the outer contour of the layout and fulfills the inner area uniformly, the created ideas are tried to be implemented in this direction. Different attempts made in trial process were challenged by the objections and didn’t give a sufficient result as it is defined in programming phase.

The created design for the pattern is defined with horizontal and vertical lines, and the windings as curves. Although the project also proposes a solution for trapezoidal shapes, it is not given as a standard solution in full structure examples considering the other possible edge-shapes like curves.
The structures used as examples intentionally contain curves and slope lines within the shape stating the gaps still exist. Even though the wiring is uniformly distributed within the shape, the figure 11.1 shows the existing gaps due to the lack in following the outer contour in wiring.

![Figure 11.1-Presence of gaps in wiring](image)

In brief, the result is a sufficient solution for automating the design process of wiring in a heat seat layout. Simple to use, modifiable and makes the design process faster for the user. Thus the quotation process becomes faster and easier for the designer as the length of the wiring is achieved in a short while. Combined with the ongoing project ProceedoStudio in engineering school, the type of the wire is also defined and the necessary requirements are fulfilled for whole quotation process.

### 11.1 Further studies

In generative design phase, new ideas generated and evaluated in decision making matrices. The results gained from the matrices are tried in programming and implementation phase. But either due to the capability of programs or maturity of the concepts challenged the programming phase.

For further improvements in pattern creation, points creation on the curves and lines is highly recommended as a concept which can make the pattern be possible created independently from shape of the element. Nodes in the element will follow the outer contour and are fulfill will be improved.

The proposed mathematical concepts (especially the space filling curves), are not enough sufficient at the moment due to the lack of maturity. However, this method would give an optimum solution to the problem if considered as a further study.
12 References

Abramowitz, M. and Stegun, I.R (1972) “Handbook of Mathematical Functions”, UNITED STATES DEPARTMENT OF COMMERCE Luther H. Hodges, Secretary NATIONAL BUREAU OF STANDARDS , A. V. Astin, Director


Gaspa, R.L (2005)” Automated design of the outline of seat heater”, Diploma Work, Jönköping university,

Peiten and Saupe (1988);”Chaos and Fractals” Springer-Verlag, New York Inc.


Dassault systems, Documentation for CATIA V5 R 18

http://en.catiamacros.net/

http://en.wikipedia.org/wiki/Rectangular_function

http://mathworld.wolfram.com/


Appendix

Appendix. A: Trapezoidal Layout

Appendix. B: Pseudo Code
Appendix. A

Trapezoidal Element:

The concept of creating wiring pattern in trapezoidal layout comes up with locating the reference on the corner on the left side and creating start and end point is by designer. Also there is one possibility to change the horizontal distance of start and end point by respect to the reference on the corner in figure (1) reference and the position of start and end point is shown.

![Figure-A.1](image)

Designer defines order geometrical sets and creates start and end points in CATIA tree as a model shown in figure (A.2).

![Figure-A.2](image)
The procedures of wiring pattern follow by opening the wiring pattern toolbox and choose the trapezoidal layout type from drop down list, put the proper values then run the program by clicking on design button. The procedure is shown on figure(A.3).
Appendix B

Pseudo code of program:

This pseudo code explains how the macro runs the program. We follow explanation in general to show how procedure comes up to solution.

First, import 2D layout in CATIA that has been created from 3D model of seat. Layout is prepared in order to design of pattern.

Second, to set order geometrical sets to “Elements” then set sub geometrical set as “Main Element” and projection of main element and connection points,”Bridges” and projection of bridges, “Sub Element1” and projection of sub element and connection points. There is one more order geometrical set that has to be fixing in layout as a “Wiring Pattern“

Third, run the wiring pattern toolbox; in drop down list choose a layout type. Insert values of dimensions and constrains in the toolbox. And run the program.

In Figure(B.1) shows the CATIA tree, 2D layout is imported as “Base Sketch”,Then creating the elemens geometrical sets “Elements” and result comes in sub order geometrical set of “WiringPattern”.

![Figure-B.1](image-url)
Flow chart of Design Automation of wiring pattern for car seat heater is shown in Figure (B.2)

Figure-B.2
Flow chart in codes is shown in Figure (B.3)

```vba
Dim part1 As Part
Dim bodies1 As Bodies
Dim body1 As Body
Dim orderedGeometricalSets1 As OrderedGeometricalSets
Dim orderedGeometricalSet1 As OrderedGeometricalSet
Dim orderedGeometricalSets2 As OrderedGeometricalSets
Dim orderedGeometricalSet2 As OrderedGeometricalSet
Dim orderedGeometricalSets3 As OrderedGeometricalSets
Dim orderedGeometricalSet3 As OrderedGeometricalSet
Dim orderedGeometricalSets4 As OrderedGeometricalSets
Dim orderedGeometricalSet4 As OrderedGeometricalSet

Dim hybridShapeFactory1 As HybridShapeFactory
Dim hybridShapePointCoord1 As HybridShapePointCoord
Set partDocument1 = CATIA.ActiveDocument
Set part1 = partDocument1.Part
Set bodies1 = part1.Bodies
Set body1 = bodies1.Item("Layout")

Set orderedGeometricalSets1 = body1.OrderedGeometricalSets
Set orderedGeometricalSet1 = orderedGeometricalSets1.Item("Elements")

Set orderedGeometricalSets2 = orderedGeometricalSet1.OrderedGeometricalSets
Set orderedGeometricalSet2 = orderedGeometricalSets2.Item("MainElement")

Set orderedGeometricalSets3 = orderedGeometricalSet2.OrderedGeometricalSets
Set orderedGeometricalSet3 = orderedGeometricalSets3.Item("ConnectionPoints_Main")

Set orderedGeometricalSets4 = orderedGeometricalSet1.OrderedGeometricalSets
Set orderedGeometricalSet4 = orderedGeometricalSets4.Item("Sub_Element_1")

Set orderedGeometricalSets5 = orderedGeometricalSet4.OrderedGeometricalSets
Set orderedGeometricalSet5 = orderedGeometricalSets5.Item("ConnectionPoints_Sub_1")

Set orderedGeometricalSets6 = body1.OrderedGeometricalSets
Set orderedGeometricalSet6 = orderedGeometricalSets6.Item("WiringPattern")
```

**Import 2D Layout**

**Set Geometrical Sets**

**Result: Creating Wiring Pattern**

Figure-B.3