A CASE STUDY OF THE DECISION-MAKING BEHIND THE AUTOMATION OF A COMPOSITES-BASED DESIGN PROCESS

Khanolkar, Pranav Milind;
Vrolijk, Ademir;
Olechowski, Alison

University of Toronto

ABSTRACT

Automation and artificial intelligence (AI) are increasingly seen as appealing tools to perform design tasks traditionally accomplished by human designers. In today's digital economy, industries aim to adopt these tools to improve the efficiency of their complex design processes. But how does one decide what parts of their existing design process should be automated and which automation/AI tool to implement? With these questions in mind, we present a case study highlighting a company's decision-making process in converting its existing designer-dependent design process to one supported by automation. In this case study, we observed the company's decisions in selecting and rejecting certain automation and AI methods before finalizing a heuristics-based automation method that proved highly efficient compared to the company's traditional human-driven design program. In addition, we present three key discussion points observed in this case study: (1) the importance of implementing the designer's heuristics in the automation framework, (2) the importance of a uniform and modular design automation framework, and (3) the challenges of implementing AI methods.

Keywords: Design automation, Decision making, Case study, Design process

Contact:
Khanolkar, Pranav Milind
University of Toronto
Canada
pranavm.khanolkar@mail.utoronto.ca

Cite this article: Khanolkar, P. M., Vrolijk, A., Olechowski, A. (2023) 'A Case Study of the Decision-Making behind the Automation of a Composites-Based Design Process', in *Proceedings of the International Conference on Engineering Design (ICED23),* Bordeaux, France, 24-28 July 2023. DOI:10.1017/pds.2023.6

1 INTRODUCTION

The design process is, at once, the most important and most complex part of developing any (engineered) product or system. To navigate this process successfully, a designer must often gather and comprehend large amounts of information from multiple domains (Rechtin and Maier, 2010; De Weck *et al.*, 2011). They must then process this information—sometimes creatively (Hsiao and Chou, 2004) —to decide on the product's form and function (Cyert and March, 1963). Designers are under time pressures to make these decisions: delays might hurt a product's commercial success or draw unwanted oversight, further complicating the development process (Brainard and Szajnfarber, 2019; Collopy and Hollingsworth, 2011). But designers must also make these decisions carefully, as revising design mistakes is costly (Chua and Hossain, 2011; Love, 2002). Thus, the engineering process is about making good decisions efficiently, which is crucial for the success of the product (Buede and Miller, 2016).

Automation and artificial intelligence (AI) are increasingly seen as tools to accomplish these aims. Industry 4.0 proponents believe automation will steadily increase its role in the development process, making it crucial to creating products efficiently (Lasi *et al.*, 2014). AI is steadily gaining ground in the design process: gathering and parsing customer requirements (Hou *et al.*, 2019), generating design concepts (Fujita *et al.*, 2021) and evaluating complex options (Zhou *et al.*, 2009). Literature also illustrates the influence of AI/automation tools on a designer's approach to a design problem. With AI-based generative tools in the design process, designers can be directed to approach a design problem as an abstraction instead of an idea of a final design solution (Saadi and Yang, 2023). Industry-based case studies have highlighted the use of AI tools that enable the designers to focus on the problem-finding, scope and framing, whereas the AI covers the automated problem-solving to provide effective user-centred solutions (Verganti *et al.*, 2020). Considering the trajectories that these tools are on, scholars are now framing automation and AI as ways to relieve some of the burdens of the effort-and-time-intensive tasks traditionally performed solely by humans (Lee *et al.*, 2018; Marion and Fixson, 2020).

But how does a designer decide what parts of their existing design process should be automated? While the efficiency gains from automation and AI are clear, it is not (yet) a plug-and-play replacement. Design processes are often embedded across different arms of an organization, on multiple platforms, with frequent inputs from different individuals. Thus, considerable effort is required to implement a system that would meaningfully impact the organization: simple automated processes would not be robust to changing environments, and narrow implementations of AI likely offer too little savings to be worth the investment. The literature does not help practitioners navigate this question either: too often, descriptions of AI implementation omit how they selected the (part of) process to automate—leaving others to fend for themselves when trying to implement in their context. Clarifying how designers make these decisions will allow us, as scholars, to create guidance for organizations navigating this process.

To this end, we present a case study where one company transitioned from a human-driven design process to one supported by automation and AI. Through this case study, we aim to illustrate the decision-making process behind a company's transition from a time-intensive manual design process to an efficient automated one. Further, we highlight the insights and challenges observed in the transition. This case study is based on a medium-sized firm, referred to as "the company", which manufactures fibre-reinforced plastics (FRP) components for industrial applications. For this case study, we studied this company for 14 months, collecting research data through meeting notes, emails, and interviews. One of the authors assisted the company's team, which focused on transitioning the company's traditional design process to an automated one.

Below, in section 2, we present the case study in which we describe the company's decisions, starting with the motivation and aim to implement an automation or AI program for designing FRP components. Next, we describe certain automation/AI-based programs and the corresponding decisions by the company to implement or not implement these programs. Finally, we present the automation programs that design and generate manufacturing specifications efficiently as per the company's aim. In section 3, we discuss the insights gained from the decisions made in this case study.

2 CASE STUDY

This case study focuses on converting a traditional designer-driven FRP-based flange design process to an automated one. The FRP flange design process involves specifying and manufacturing composite layers that are sequentially stacked and applied to the flange mould with a resin. We tracked the

https://doi.org/10.1017/pds.2023.6 Published online by Cambridge University Press

decisions that led to the outcomes of the endeavour to automate the design process. In this section, we describe the motivation to automate the traditional flange design process, followed by the development and selection of the two automation programs and an automated manufacturing generation program in the following sub-sections.

2.1 Background and motivation

The company designs and manufactures a wide range of corrosion-resistant FRP and dual laminate equipment, from piping systems to ducts, stacks, hoods, covers, cells, and other miscellaneous custom equipment. Their engineering team has used different design methods provided in the engineering codes and standards. The company has also developed internal design tools using Excel and Mathcad software to expedite the design process. However, the company observed the existing process of designing and generating the manufacturing specifications to be very time-consuming.

The company's main aim was to create a system that streamlines the majority of product designs and specifications generation processes towards improving the efficiency of the supply chain. The existing system for designing FRP-based components utilizes 80% of the total designing and manufacturing time, thereby creating a major bottleneck in the overall supply chain. The company traditionally used MS VBA-based Excel templates for designing the composite layers for their respective components, followed by transferring the final dimensional data of the composite layers to a SolidWorks program to generate files required for CNC manufacturing of the thin composite layers. This collaborative environment was judged to be too inefficient and required effort-intensive inputs from designers to achieve the final design solution. This could potentially lead to the company losing future orders due to the incapacity of the engineering department to streamline the bottleneck. As such, the engineering manager proposed an automated design framework that optimizes designs and generates manufacturing specifications as a faster alternative to the traditional design process. To achieve this aim, the company formed a project team comprising the engineering manager, the lead engineering designer, and the program developer. The engineering manager made the major decisions, including the project's requirements, expectations, and deliverables. The design-development team comprising the designer and the developer, worked towards developing a new design system to replace the existing one.

As the first step towards developing an automated design framework, the engineering manager and the designer decided to select FRP flanges as the first component for which to build the framework. For context, flanges are part of the piping systems in which they are used to connect two pipes or a pipe and any type of fitting or equipment. The designer stated that their FRP flange design process is highly complex, involving a designer's inputs in multiple stages, several interdependent and iterative calculations, and multiple constraint checks to get the final design. Moreover, the current FRP flange design process requires using an MS VBA-based Excel template for designing 11 different types of flanges. The traditional flange design process is depicted in Figure 1. This traditional flange design program starts with the designer entering flange design requirements into an Excel sheet, followed by designing the composite layer sequence in the Excel program. The Excel program automatically evaluates this layer sequence by generating the design's performance and final dimensions. Suppose the flange design does not satisfy the constraints set according to the design requirements. In that case, the designer modifies the composite-layer-sequence and/or modifies the dimensional parameter value, which significantly impacts the design's final dimensions and evaluation performance. This iterative process optimizes the design to achieve the best solution.

Once the design is finalized, the designer instructs the program to generate the manufacturing specifications and transfer the final composite layers with corresponding dimensions to the SolidWorks software to generate the manufacturing files. This flange design process required 20-30 minutes to complete, which is quite time-consuming per the company's designer. This process involved frequent designer interventions and iterations in the composite-layer-sequence generation and design optimization stages (as indicated by blue arrows in Figure 1). Once the design was finalized, the manufacturing file generation in SolidWorks further added to the inefficiency.

As such, the company selected FRP flanges to build and test the new design automation or AI framework, with an aim to improve the efficiency of the flange design process and serve as a blueprint that can be adapted to build similar design automation frameworks catered to other FRP components. According to the engineering manager, the automated framework for designing flanges should extract the flange design requirements from a document, then automatically generate and evaluate the composite layers with correct dimensions corresponding to the design requirements and standards.

https://doi.org/10.1017/pds.2023.6 Published online by Cambridge University Press

Once the evaluation is completed, the program should generate the manufacturing specifications files required for manufacturing the composite layers of the flanges.

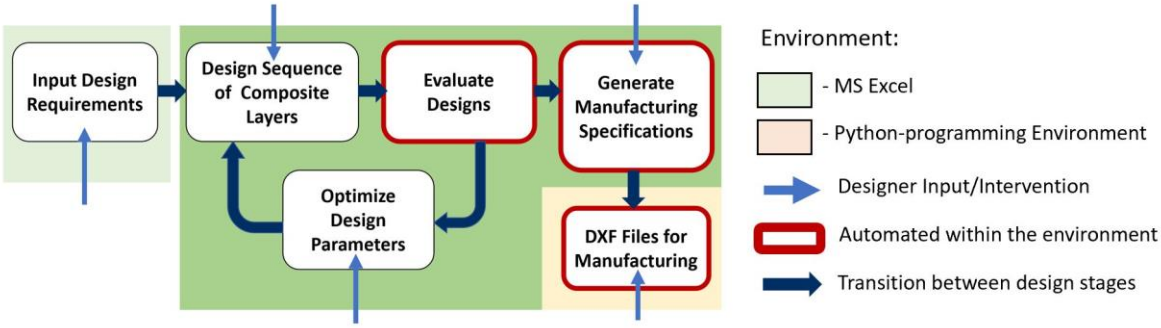

Figure 1. The company's traditional flange design program

As the first step in this conversion process, the engineering manager decided to transfer the complete flange design program and the calculations from the Excel template to a Python-based environment. The Python-based environment provided a wide range of functions that can be highly beneficial for developing the design automation and manufacturing-specifications-generation framework, so it was deemed suitable for this application by the engineering manager. The design-development team created a separate Excel template where the designer can provide design inputs per the flange's requirements. The purpose of this Excel template is to make it easier for the Python program to extract all the input values and then use them in the calculations. Once this flange design process was replicated in the Python environment, the engineering manager decided to focus on automating the generation of the composite-layer-sequence process, then optimizing the dimensional parameters and layer modification to achieve the best set of composite layers corresponding to the flange requirements. As such, the design-development team worked towards developing an initial random optimization algorithm that, ultimately, was not a worthwhile replacement for their existing method.

2.2 The failure of a random optimization method and challenges of AI implementation

Per the engineering manager, the complete flange-design automation program should consist of two key aspects: (1) automatic generation of an appropriate sequence of composite layers and (2) optimizing this sequence and their dimensions. The project team reviewed their past design methods and successful designs as a starting point for developing a flange-design algorithm. The rise of AI in different engineering fields inspired the project team to seek AI-based methods to automate their FRP flange design process. Reviewing these aspects, the project team discussed whether an artificial intelligence-based machine learning method is applicable for automating and optimizing the FRP flange design process. The project team realized that most AI techniques, including supervised machine learning and deep learning, are data-driven, requiring large amounts of labelled data for the AI-based models to train, learn, and then make accurate predictions or automate design generation. In addition, the complexity of the input-to-output variable, which should include the design requirements, sequence of composite layers, dimensional parameter value, and categorical input regarding the type of flange, proved extremely challenging for the development and application of AI to the flange design process. Due to the lack of labelled data and the complexity of the input-output variables involved in the flange design process, the project team did not go ahead with AI-based machine learning methods.

In response to the failure of implementing data-driven AI methods in the design framework, the project team decided to develop a system that would automatically evaluate and compare multiple flanges designs. To evaluate multiple flange design alternatives and to achieve the best one among them, the engineering manager and the designer developed a points-based system catered explicitly to evaluating a flange design. This points-based system was an algorithm to score a flange design based on four evaluation criteria for assessing flange designs: (1) appropriate sequence of composite layers, (2) interference/clearance of the design due to spot-facing, (3) design safety factors, due to stress applied, and (4) thickness contribution to the hub of the flange. First, the points-based system individually allotted points corresponding to each of these four criteria for a flange design. These points were either positive or negative depending on the evaluation results of the flange design on these four criteria. Then, these individual points obtained for each of the four criteria were summed to obtain the final score for a

https://doi.org/10.1017/pds.2023.6 Published online by Cambridge University Press

flange design. The individual points obtained for each of the four criteria were then summed to obtain the final score of the flange design, and the design with the highest score was finalized.

The development of the points-based system presented an opportunity to apply AI-based reinforcement learning (RL). A RL method includes the development of an AI-based agent which takes actions within an environment for which it gets rewards or penalties, usually depicted as positive or negative scores. With this thought, the RL agent learns to take actions that lead to maximum rewards and minimum penalties. In addition, the benefit of the RL approach is that it does not necessarily require labelled data. In the Python-based flange design program, the actions would have been the generation and modification of the composite layer sequence and the modification of the dimensional parameter. The points-based system that evaluates the four criteria would serve as the rewards and penalties. Although this method of developing and applying an RL agent seemed promising, its implementation in the flange design program was completely dependent on the reliability of the company's points-based system. As such, the project team focused on developing an algorithm that they expected would quickly generate and evaluate a large sample of flange designs using the points-based system without designer intervention. In addition, the results from this algorithm would also prove helpful in assessing whether the points-based evaluation system was reliable or not.

After this point-based system was implemented in the flange design program, the design development team developed an algorithm that generates a random sequence of composite layers within certain constraints set by the company based on past designs. Next, the algorithm randomly generates the dimensional parameter, which influences the dimensions of these layers required for subsequent evaluation of the flange design. Once the evaluation is complete, the design is allotted a score based on the four criteria through the points system. To test this algorithm on a large sample of designs, the project team decided to use this algorithm to generate 1,000 random sequences of composite layers followed by 100 randomly generated dimensional parameters. Once the 100,000 iterations were complete and the scores obtained for each flange design represented by each iteration, the algorithm retrieved the top five designs with the highest score to be finally reviewed and selected by the company.

One of the major limitations of this algorithm was the time taken to process 100,000 iterations of flanges designs. Although automated, the project team observed that this flange design algorithm would take approximately one second to process one flange design and 100,000 seconds to run the entire algorithm, which was deemed too inefficient to the team. Although this method was fully automated, it takes significant time, i.e., approximately 28 hours, to get the best flange design over the time taken using the traditional method, i.e., 20-30 minutes. In addition to the time limitation, the designer observed that the top-scored designs were not reliably optimal, which may have reflected certain flaws in the points-based system. On close inspection, the project team observed that the top-scored designs satisfied the constraints by a larger margin than expected, indicating the use of more material than required, resulting in a potential overdesign and wasted resources. Besides that, the project team detected that the scoring system of certain evaluation criteria wrongly outweighs the others, such that the better designs got a lower score. As the point-based system was based on trial-and-error and the project team's intuition of how the different evaluation criteria should be scored, it did not completely reflect the designer's heuristics, which further led to the uncertainty of the algorithm to obtain the best design. Since the points-based system proved unreliable in evaluating the designs, the project team decided to remove this point-based design evaluation method and thus, the AI-based RL approach was not explored further.

Due to the inefficiency of the random-optimization method and the challenges in implementing AI-based methods, the project team decided to focus on developing an automation algorithm that effectively reflects the designer's heuristics. However, before starting another attempt at automation, the project team decided to shift towards generating the manufacturing specifications of the flange designs, with the aim to eliminate the use of the traditional SolidWorks software-based program.

2.3 Manufacturing specifications generation-automation program

As the random optimization method did not show promising results, the project team decided to shift focus toward implementing manufacturing-specifications-generation in the Python-based flange design program. This shift from flange design automation to manufacturing-specifications-generation automation was based on the decision to replace the final time-consuming and expensive software of the traditional process, i.e., SolidWorks program, and have a complete Python-based program that processes flanges designs and generates manufacturing specifications. As a reminder, the

https://doi.org/10.1017/pds.2023.6 Published online by Cambridge University Press

manufacturing specifications contain three aspects: (1) a manufacturing specifications sheet, (2) a Bill of Materials (BOM) and (3) manufacturing files that are sent to the manufacturing team.
First, the engineering manager decided that the manufacturing specification sheet and the BOM sheet generated by the Python program should be the same as the one generated by the previous Excel-based program. The main reason for this decision was to provide consistency and convenience for the end users (manufacturing team), to refer to the files easily, and get the corresponding values quickly (as the location of the values within the file is known through repeated interaction with such files). Hence, appropriate templates in MS Excel were created for manufacturing specifications and BOM sheets. Then, the Python program was modified to transfer the data of the final flange design to appropriate locations within these templates, then convert these filled templates to PDF format.
Next, the project team moved on to generate manufacturing files. These files are stored in the Drawing Exchange Format (DXF), which contains two-dimensional shapes of the composite layers along with their dimensions, which are then sent for CNC manufacturing. The previous SolidWorks-based DXF generation program would take approximately 3-4 minutes to generate a batch of five flanges and required frequent designer intervention to run this program for each flange. To reduce the time and effort for generating the DXF files, the project team decided to implement a DXF generation capability within the Python programming environment developed for flange design. As the first step of this implementation, the flange design-Python program was modified to include the function that automatically retrieves the type of flange-based composite layers and their corresponding dimensional values. This step is followed by generating the DXF files corresponding to each composite layer. Once DXF generation was implemented in the Python program, it was tested on multiple flange designs. As it successfully generated DXF files without designer intervention, the design-development team decided to create and implement this aspect into a separate Python program to focus solely on creating DXF files from existing Excel-based flange design templates. Such a new DXF-generation program could further eliminate the effort and time-intensive SolidWorks program for DXF-generation for existing flange designs. Once this separate program was created, the project team found it considerably faster to generate DXF files of the composite layers for flange designs.
This separate Python program could process multiple Excel-based flange design templates in a designer-specified folder and automatically generate DXF files for all the templates within the specified folder. Compared to the traditionally used SolidWorks-based program, the new Python-based DXF-generation program could generate DXF files for more than ten flange templates in less than a minute. As an add-on to this program, the design development team decided to include a function that detected the BOM information from the Excel program and then exported this information in a PDF file. The new Python-based DXF-generation program is depicted in Figure 2.

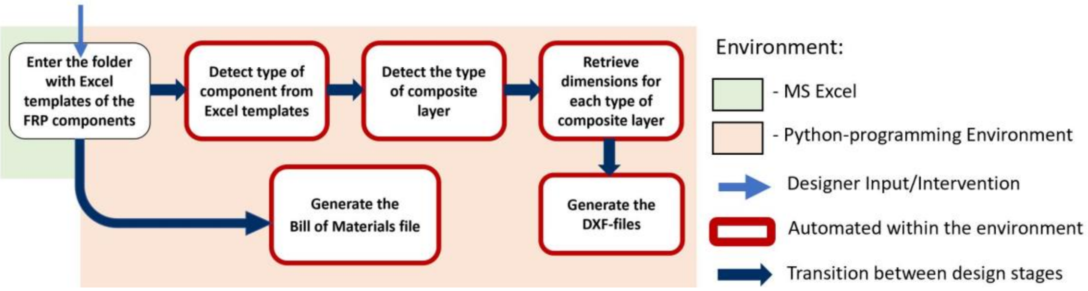

Figure 2. The new automated DXF-generation process

As this new Python-based DXF-generation program proved highly effective and efficient for the project team, they decided to extend it to generate other shapes of composite layers corresponding to other FRP components. In their view, this inclusion would further eliminate the use of SolidWorks for generating DXF files for other components besides flanges. In such a manner, the developer modified the DXF-generation program to detect the type of component and the type/types of composite layers corresponding to that component and then generate the DXF files, further followed by the generation of the BOM file for that component. In such a manner, the design development team implemented a total of nine shapes for 14 different components, including flanges, into the DXF generation program, which on testing, proved to be quicker than the previously used SolidWorks-based program.
As this manufacturing-specification-generation aspect of the Python-based flange-design program and the separate DXF-generation program proved effective, the project team made another attempt at automating flange design.

https://doi.org/10.1017/pds.2023.6 Published online by Cambridge University Press

2.4 Heuristics-based design automation method

Since the random optimization method combined with the point-based evaluation system proved ineffective, the project team discussed a new approach to automating the flange design generation and evaluation program. Based on the lack of data encoded within the program to generate the best designs, the project team sought to develop an algorithm that generates and evaluates designs completely based on their designer's heuristics.

The heuristics to be encoded in the flange design automation program were obtained from probing the designer's three years of experience designing approximately 300 FRP flanges using the traditional method. These heuristics included the designer's instincts to design flanges with the correct composite layer sequence and dimensions, which are expected to satisfy the evaluation criteria and constraints set by the design's requirements. After having multiple discussions with the project manager regarding the flange design requirements and final design expectations, the designer ideated a heuristics-based algorithm to be encoded in the automation program. This algorithm comprised a series of design-generation steps, evaluations, constraint-checks, and design modifications that the designer manually performed toward the final design. For developing a heuristics-based algorithm, the project team focused on automating two aspects, (1) generation of composite layer sequence and (2) optimization of the dimensional parameter to satisfy the flange design constraints. First, the program focused on generating a sequence of composite layers for a flange. The designer's heuristics in this sequence generation process acted as the constraints to get the correct sequence of composite layers corresponding to the flange's design requirements. Once a proper sequence of composite layers was generated, the algorithm evaluated the flange design using a default dimensional parameter set according to the designer's heuristics. As explained before, this dimensional parameter played an important role in computing the dimensions of the composite layers, which in turn influences the other three evaluation criteria, i.e., interference/clearance of the design due to spot-facing, design safety factors, and thickness contribution of the layers to the hub of the flange. Once the design with the default dimensional parameter is evaluated on these three criteria, the algorithm compares these evaluations with the design constraints corresponding to the three criteria. These constraints are automatically set according to designer's heuristics. Second, based on the three evaluation results and the corresponding constraints, the algorithm manipulates the dimensional parameter, i.e., increasing or decreasing the parameter's value, such that the three constraints are satisfied, and the program outputs the final design. Once the heuristics-based automation algorithm provided the final flange design, the designer reviewed it and generated the manufacturing specification files.

Figure 3 summarizes the traditional and fully automated flange design process with respective environments and the aspects of the process that require human input and/or are automated. As observed from Figure 3, every aspect of the traditional flange design process has been automated. In this new automated process, the designer can enter the flange design inputs into an Excel template, from which the Python program can extract the values for further processing. Next, we observe that the new automation program leveraged the heuristics-based algorithm to automate the composite-layer-sequence generation and the parameter optimization. Instead, the traditional design process required frequent designer interventions and inputs in these sub-processes. Once a design is finalized, the program generates the manufacturing specifications sheet, BOM sheet and DXF files completely within the Python environment (refer to section 2.3), which proves more efficient than the traditional process.

The project team successfully tested this new automation program in designing 11 different flanges. The automated-flange design program significantly reduced design time which previously resulted from iterations in the traditional design process. The total time taken for designing and evaluating a FRP-flange is highly dependent on the type of flange. Traditionally, it would take a maximum of 30 minutes to design and evaluate an FRP flange, whereas the automation program can design a flange within 4 minutes. without requiring the designer's constant attention. A stage-wise time comparison between the traditional flange design program and the heuristics-based automation program was obtained by designing ten different types of flanges and recording the maximum time observed for each design stage, as displayed in Table 1. From Table 1, we can observe that the heuristics-based program is more time efficient at each step than the traditional flange design program. Overall, the automation program has proved to be 7.5 times faster than the traditional program, which, according to the manager, could be significant in the effort to remove the bottleneck from the supply chain.

https://doi.org/10.1017/pds.2023.6 Published online by Cambridge University Press

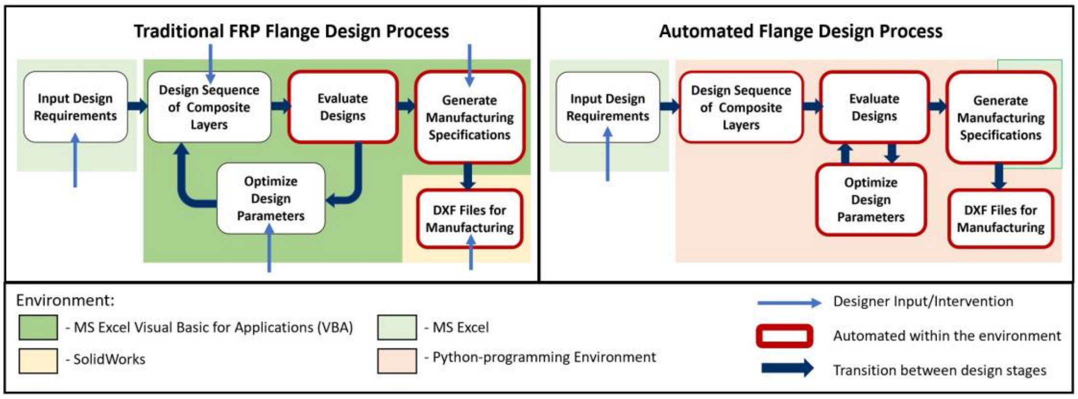

Figure 3. Side-by-side comparison of traditional and automated FRP flange design process

The company's journey, which includes the decisions made and actions taken towards converting the flange-design and manufacturing-specifications-generation processes from the traditional workflow to an automated one, is summarized in Figure 4.

Table 1. Time comparison between the traditional and automated flange design program

Stage of the flange design process	**Max. time: traditional design process (seconds)**	**Max. time: heuristic-based automated design process (seconds)**
Input design requirements	150	120
Generate composite-layer-sequence	300	< 1
Evaluate design	30	< 1
Optimize design parameters	1200	100
Generate manufacturing specifications	30	15
Generate DXF	90	< 1
Total time	**1800**	**240**

3 DISCUSSION

This case study tracked how one company transitioned from a time-intensive manual design process to an efficient automated heuristics-based design program and manufacturing specifications generation program. As described in the previous section and illustrated in Figure 4, this case study examined the decisions and actions that led to the development of these programs. The complete shift in the environment from MS VBA-based Excel and SolidWorks to a single Python-programming-based program provided the company with a more time-efficient and flexible environment for designing flanges and generating manufacturing files for several other components. We discuss the insights derived from the decisions and subsequent results below.

3.1 Importance of the designer's heuristics

Designers' heuristics, which are gained from several years of experience, could play a central role in automating a design process in case of a lack of historical data and incompatibility of AI-based methods. The traditional FRP flange design process, which involved several inter-dependent design calculations and multiple constraint checks, required major interventions and iterations from the company's designer. As described in section 2.2, this first attempt at automating the flange design process seemed too time-consuming and unreliable and led to the development of an automation framework that mainly reflected the designer's heuristics built from their experience designing flanges. In addition, the lack of previous historical data directed the company to develop an automation algorithm that reflects how the designer would ideally design the flange. The development of this algorithm required several iterations to most accurately represent their heuristics. Once the algorithm was complete, it was tested on multiple flanges and underwent various modifications to achieve a generalized automation program for designing 11 different types of flange designs. As such, the designer's heuristics from years of experience designing several FRP flanges were crucial to be encoded in this program. Hence, the lack of historical data to implement an AI algorithm and the unreliability and inefficiency of the first automation algorithm forced

https://doi.org/10.1017/pds.2023.6 Published online by Cambridge University Press

the team onto a different path, that is capturing an embedding of the designer's heuristics into the new and effective flange design automation program.

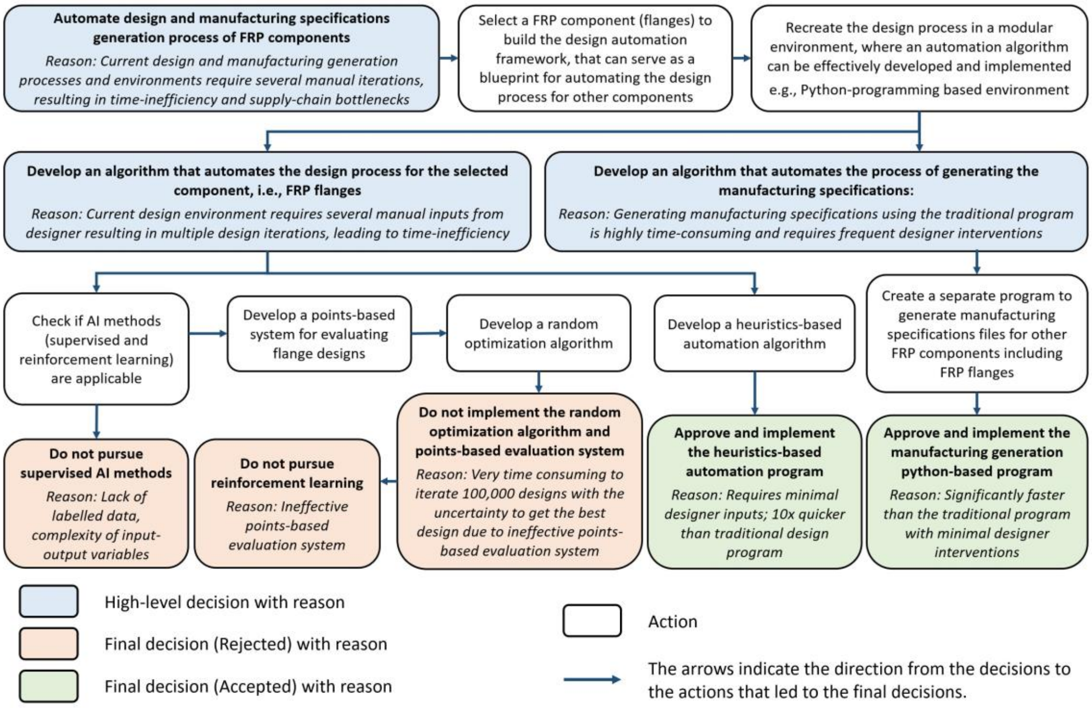

Figure 4. Overview of the conversion process of the traditional design system to an automated one

3.2 Importance of single modular automation framework

A modular framework for automating a component's design process could prove beneficial in accommodating future design modifications and error detection in the respective design modules and potentially serve as a blueprint for automating the design processes for other components. As described earlier, the complete shift from VBA-based Excel and SolidWorks to a Python-programming-based environment proved highly beneficial to the company in terms of efficiency. The new process in a single environment provided quick results and reduced the computational footprint by reducing the number of software platforms needed to get the final results. Besides improved efficiency, another important aspect of the flange design automation and the DXF generation frameworks was that these frameworks were developed in Python programming in a modular fashion. Such a modular framework further enables the company to modify specific aspects of the program to refer to and accommodate future changes in design calculations or even detect the cause of errors during the design or evaluation stage of the flange design process. Furthermore, a modular flange design automation framework could further serve as a blueprint for developing similar frameworks for other FRP components. In addition, the DXF generation program developed in a modular way enables the company to add more composite shapes for potential new components through significantly less effort compared to the previous SolidWorks-based DXF generation program. Thus, developing the design automation and DXF generation programs in a single environment and modularly proved highly efficient, flexible and potentially adaptive for the company.

3.3 Challenges for implementing AI

One of the key decisions in developing a modern automated framework for a design process is whether an AI-based method is suitable or not. The growth of digitization and AI algorithms has led to the emergence of several AI-based methods in different engineering fields, including design. So, it was natural for the company to seek AI-based methods to automate the design process corresponding to their FRP components. However, due to the lack of available data, which included latent information about the past flange designs, the variables involved, and the expected performances on the design evaluations, supervised AI methods were infeasible. In cases of lack of labelled data for implementing AI, there lies a trade-off of investing time in generating and labelling large amounts of data versus exploring other

https://doi.org/10.1017/pds.2023.6 Published online by Cambridge University Press

automation options. Suppose the data generation and labelling process is deemed to be extremely time-consuming, as in this case of designing flanges (20-30 minutes/flange). In that case, the feasible option is to explore other automation methods. In addition, as the points-based system proved unreliable and ineffective in evaluating the designs, the AI-based RL approach was also not explored further as an automation method. Thus, the lack of data, infeasibility of data generation and labelling, and an appropriate points-based design evaluation system were the major factors for the company's decision not to pursue AI-based methods, even though there was potential for such methods in this process.

4 SUMMARY

In summary, this case study highlighted the decision-making process behind the conversion of an effort-and-time-intensive design process to an efficient automated one. We described the decisions that led to the development, selection and rejection of certain automation and AI methods to obtain the final automation design and DXF generation programs. The insights gained in this case study which include the challenges encountered by the company in implementing AI and the subsequent development of single-environment, modular, and time-efficient heuristics-based design automation programs, could potentially serve as a reference for companies planning to automate their design processes.

ACKNOWLEDGMENTS

This project was funded by the company and the MITACS Accelerate program. The authors wish to thank Dr. Shahin Shadlou and William Debertin for their expertise.

REFERENCES

Brainard, S.M. and Szajnfarber, Z. (2019), "How government oversight adds time to contractor engineering work", Systems Engineering, Wiley Online Library, Vol. 22 No. 1, pp. 54–65.
Buede, D.M. and Miller, W.D. (2016), "The engineering design of systems: models and methods", John Wiley & Sons.
Chua, D.K.H. and Hossain, M.A. (2011), "Predicting change propagation and impact on design schedule due to external changes", IEEE Transactions on Engineering Management, IEEE, Vol. 59 No. 3, pp. 483–493.
Collopy, P.D. and Hollingsworth, P.M. (2011), "Value-driven design", Journal of Aircraft, Vol. 48 No. 3, pp. 749–759.
Cyert, R.M. and March, J.G. (1963), A Behavioral Theory of the Firm, Vol. 2, Englewood Cliffs, NJ.
Fujita, K., Minowa, K., Nomaguchi, Y., Yamasaki, S. and Yaji, K. (2021), "Design concept generation with variational deep embedding over comprehensive optimization", Proceedings of the ASME Design Engineering Technical Conference, Vol. 3B-2021, available at:https://doi.org/10.1115/DETC2021-69544.
Hou, T., Yannou, B., Leroy, Y. and Poirson, E. (2019), "Mining customer product reviews for product development: A summarization process", Expert Systems with Applications, available at:https://doi.org/10.1016/j.eswa.2019.04.069.
Hsiao, S.W. and Chou, J.R. (2004), "A creativity-based design process for innovative product design", International Journal of Industrial Ergonomics, available at:https://doi.org/10.1016/j.ergon.2004.05.005.
Lasi, H., Fettke, P., Kemper, H.G., Feld, T. and Hoffmann, M. (2014), "Industry 4.0", Business and Information Systems Engineering, Vol. 6, pp. 239–242.
Lee, J., Davari, H., Singh, J. and Pandhare, V. (2018), "Industrial Artificial Intelligence for industry 4.0-based manufacturing systems", Manufacturing Letters, Vol. 18, pp. 20–23.
Love, P.E.D. (2002), "Influence of project type and procurement method on rework costs in building construction projects", Journal of Construction Engineering and Management, American Society of Civil Engineers, Vol. 128 No. 1, pp. 18–29.
Marion, T.J. and Fixson, S.K. (2020), "The Transformation of the Innovation Process: How Digital Tools are Changing Work, Collaboration, and Organizations in New Product Development", Journal of Product Innovation Management, Vol. 38 No. 1, available at:https://doi.org/10.1111/jpim.12547.
Rechtin, E. and Maier, M.W. (2010), The Art of Systems Architecting, CRC press.
Saadi, J.I. and Yang, M.C. (2023), "Generative Design: Reframing the Role of the Designer in Early-Stage Design Process", Journal of Mechanical Design, American Society of Mechanical Engineers, Vol. 145 No. 4, p. 41411.
Verganti, R., Vendraminelli, L. and Iansiti, M. (2020), "Innovation and Design in the Age of Artificial Intelligence", Journal of Product Innovation Management, available at:https://doi.org/10.1111/jpim.12523.
De Weck, O.L., Roos, D. and Magee, C.L. (2011), Engineering Systems: Meeting Human Needs in a Complex Technological World, Mit Press.
Zhou, C.C., Yin, G.F. and Hu, X.B. (2009), "Multi-objective optimization of material selection for sustainable products: Artificial neural networks and genetic algorithm approach", Materials and Design, available at:https://doi.org/10.1016/j.matdes.2008.06.006.