A New Tool for Automation of Mission Analysis Process for Aero-Engine Turbine Rotors

by

Ravza ASCI

THESIS PRESENTED TO ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
IN PARTIAL FULFILLEMENT FOR A MASTER'S DEGREE
WITH THESIS IN MECHANICAL ENGINEERING
M.A.Sc.

MONTREAL, SEPTEMBER 13, 2025

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

Mr. Hany Moustapha, Thesis Supervisor
Department of Mechanical Engineering at École de technologie supérieure

Mr. Patrick Germain, Thesis Co-supervisor
Department of Mechanical Engineering at École de technologie supérieure

Mr. Georges Kaddoum, President of the Board of Examiners
Department of Electrical Engineering at École de technologie supérieure

Mr. Francois Garnier, Member of the jury
Department of Mechanical Engineering at École de technologie supérieure

THIS THESIS WAS PRENSENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND PUBLIC

SEPTEMBER 10, 2025

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGMENT

I would like to express my deepest gratitude to Dr. Hany Moustapha, whose guidance and encouragement have been invaluable throughout my master's journey. I am deeply grateful for always hearing me out and providing direction. This project is part of an Industrial Research Chair (2011-2028) under the supervision of Dr. Moustapha and partially funded by MITACS.

I am also thankful to Najeh Najeh and Pascal Doran for their support and contributions during my work. I hope to carry your enthusiasm into my future work, just as you do now.

To Mohamed A.K.I. Ahmed, thank you for always being there for me. You consistently guided and supported me; mentally, technically, and academically. I am thankful for having a great friend like you.

My sincere appreciation goes to my beloved family: my mother Emine, my father Fahrettin and my sisters Selva and Şule individually. Even distance could not diminish your encouragement. I love you all so much.

Last but not least, I am grateful to my best friend and husband, Mustafa Uzun, for making this two-year adventure possible and turning all the good memories into reality. Thank you for bringing out the best in me and accompanying me through my “boldly go”. Your existence is the greatest gift for me.

UN NOUVEL OUTIL POUR L'AUTOMATISATION DU PROCESSUS D'ANALYSE DE MISSION POUR LES ROTORS DE TURBINE DE TURBOMOTEUR

Ravza ASCI

RÉSUMÉ

La durée de vie d'un rotor d'aube de turbine est un paramètre crucial dans la conception et l'exploitation d'un turboréacteur. La durée de vie du composant dépend de la mission qu'il doit accomplir, ce qui fait référence aux conditions environnementales auxquelles il sera exposé durant les différentes phases d'un vol, telles que le décollage, la montée et la croisière. Puisque ces conditions d'exploitation affectent la durée de vie du rotor de turbine, elles doivent être prises en compte lors de l'évaluation de la durée de vie. Cette procédure est appelée « analyse de mission » et nécessite une série d'analyses thermiques, structurelles et de durée de vie, exécutées en séquence.

Bien qu'essentielle, l'analyse de mission est un processus particulièrement intensif. Des dizaines de missions peuvent devoir être analysées pour une seule géométrie de pale de turbine. Pour chaque mission, une séquence d'analyses thermiques, structurelles et de durée de vie doit être réalisée et, pour chaque analyse, les fichiers doivent être nommés et stockés manuellement, et les bons fichiers doivent être fournis aux outils. Étant donné qu'une grande quantité de données est produite pour chaque mission et chaque analyse, la gestion manuelle de ces données et leur transfert entre les analyses peuvent entraîner une perte de productivité à mesure que le nombre de missions augmente. L'automatisation de l'ensemble du processus et l'intégration des différentes analyses offrent une solution efficace pour gagner du temps et éviter des répétitions inutiles.

Cette recherche vise à développer un nouvel outil d'automatisation pour le processus d'analyse de mission des rotors de turbine, dans le cadre d'une Chaire de recherche industrielle (2011-2028) chez l'École de technologie supérieure (ÉTS). L'outil d'automatisation est appelé AMAP (*Automated Mission Analysis Process*), et il a pour objectif de minimiser le travail manuel des ingénieurs, de rationaliser le processus, de fournir un flux de travail bien défini ainsi qu'une gestion des données structurée pour les équipes d'ingénierie. Un gain de temps de 20 % à 65 % a été observé lors de l'évaluation de l'outil comparé aux procédures manuelles. De plus, une réduction moyenne du temps d'environ 30 % est attendue pour un cas d'essai typique. Cela va améliorer la qualité et la productivité du processus de conception des turbines, en réduisant le temps de conception, les coûts, et en menant à des conceptions de turbines plus fiables.

Mots-clés: Automatisation de processus; Analyse de mission; Rotor de turbine; Turboréacteurs; Évaluation de durée de vie en fatigue à faible nombre de cycles (LCF); Calcul de durée de vie des moteurs aéronautiques

A NEW TOOL FOR AUTOMATION OF MISSION ANALYSIS PROCESS FOR AERO-ENGINE TURBINE ROTORS

Ravza ASCI

ABSTRACT

The life of a turbine blade rotor is a critical parameter in the design and operation of a gas turbine engine. The component's life depends on the mission it will perform, which refers to the environmental conditions it will be exposed to during the different phases of a flight, such as take-off, climbing, and cruise. Since these operating conditions will affect the turbine rotor's life, they must be considered during the life evaluation. This procedure is called "mission analysis" and it requires a series of thermal, structural and lifing analysis in a sequence.

Besides being essential, mission analysis is a highly intensive process. Dozens of missions may need to be analyzed for just one turbine blade geometry. For each mission, a sequence of thermal, structural and lifing analyses must be performed and for each analysis, files must be named and stored manually, and the right files must be provided to the tools. Since a large amount of data is produced for each mission and each analysis, handling them manually and transferring data between analyses may lead loss of productivity as number of missions increases. Automation of the overall process and integration of different analyses offers an efficient solution to save time and avoid unnecessary repetitions.

This research aims to develop a new automation tool for the mission analysis process of turbine rotors as a part of an Industrial Research Chair (IRC , 2011-2028) at École de technologie supérieure (ÉTS). The automation tool is called AMAP (Automated Mission Analysis Process), and it targets minimizing manual work to be done by the engineers, streamline the process, provide a well-defined workflow and well-structured data management to the engineering teams. A time gain of 20% to 65% was obtained in the evaluation of the tool compared to manual procedures. Moreover, an average time reduction of approximately 30% is expected for a typical test case. This will enhance the quality and productivity of the turbine design process by shortening design time, reducing costs, and eventually leading to more dependable turbine designs.

Keywords: Process Automation; Mission Analysis; Turbine Rotor; Gas Turbine Engines; Low Cycle Fatigue (LCF) Life Evaluation; Aero-Engine Lifing

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABREVIATIONS

2D/3D	Two/Three dimensional
AMAP	Automated Mission Analysis Process
AR	Action Research
BCG	Boundary Condition Generator
ÉTS	École de technologie supérieure
FEA	Finite Element Analysis
FEM	Finite Element Model
GUI	Graphical User Interface
HCF	High Cycle Fatigue
HPC	High-Performance Computing
HPT	High Pressure Turbine
IRC	Industrial Research Chair
LCF	Low Cycle Fatigue
LPT	Low Pressure Turbine
NATO	North Atlantic Treaty Organization
OOP	Object-Oriented Programming
PT	Power Turbine
S-N	Stress - Cycle Number

INTRODUCTION

Gas turbines are the engines primarily used for aircraft propulsion. Their design is highly iterative and multidisciplinary. Due to the complexity of gas turbine aero-engines, designing reliable engines is both time-consuming and costly. The aero-engine market is highly competitive, and companies focus on reducing time and cost required for engine design.

All aerospace companies rely on their in-house tools, knowledge, and practices to design superior aero engines and maintain a competitive advantage. A summary of the ongoing projects under the ÉTS's Industrial Research Chair (IRC) is presented in Figure 0.1. As shown, these projects have extensive scope from specific fuel consumption optimization to rotor design and finite element analyses. This thesis focuses on mission analysis automation, as part of the mentioned research projects, specifically targeting the low cycle fatigue life evaluation of turbine rotors (as indicated in Figure 0.1). This project aimed to develop a tool to reduce the time required for early stages of engine design by developing engineering tools that implement optimization, automation, and integration.

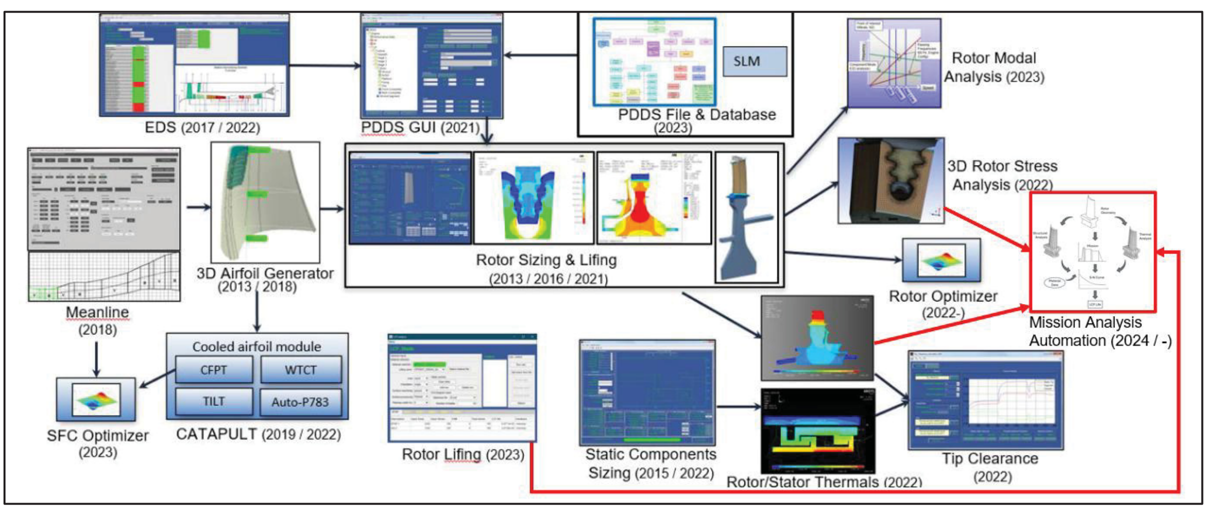

Figure 0.1 ÉTS IRC projects[1]

[1] Figure adapted from Ahmed (2025)

Turbine blades are one of the most critical components of aero-engines, as they are subjected to the high temperature gases exiting the combustion chamber. This makes accurate thermal, structural, and fatigue life assessments crucial to ensure the dependable design of turbine rotor geometries. Considering these and the impact of a turbine blade's service life on its design and overall reliability, the development of an automation tool to enable faster and more accurate life evaluations plays a critical role for aerospace companies to enhance their competitiveness in the market.

The overall life evaluation process of a turbine is referred to as transient mission analysis, since the blade's life depends on the mission profile it conducts. Although the main goal is to estimate the component's life, thermal and structural simulations must also be performed, as life prediction depends on the thermal and mechanical loads experienced throughout the mission. This process is illustrated in Figure 0.2. In addition to temperature and stress distributions across the turbine blade geometry, material data and mission definitions are also considered in the analysis. It is important to note that a turbine may perform different missions at different times, thus, analyses of a large number of mission scenarios for a single geometry are essential. As a result, the mission analysis procedure is highly iterative and manual.

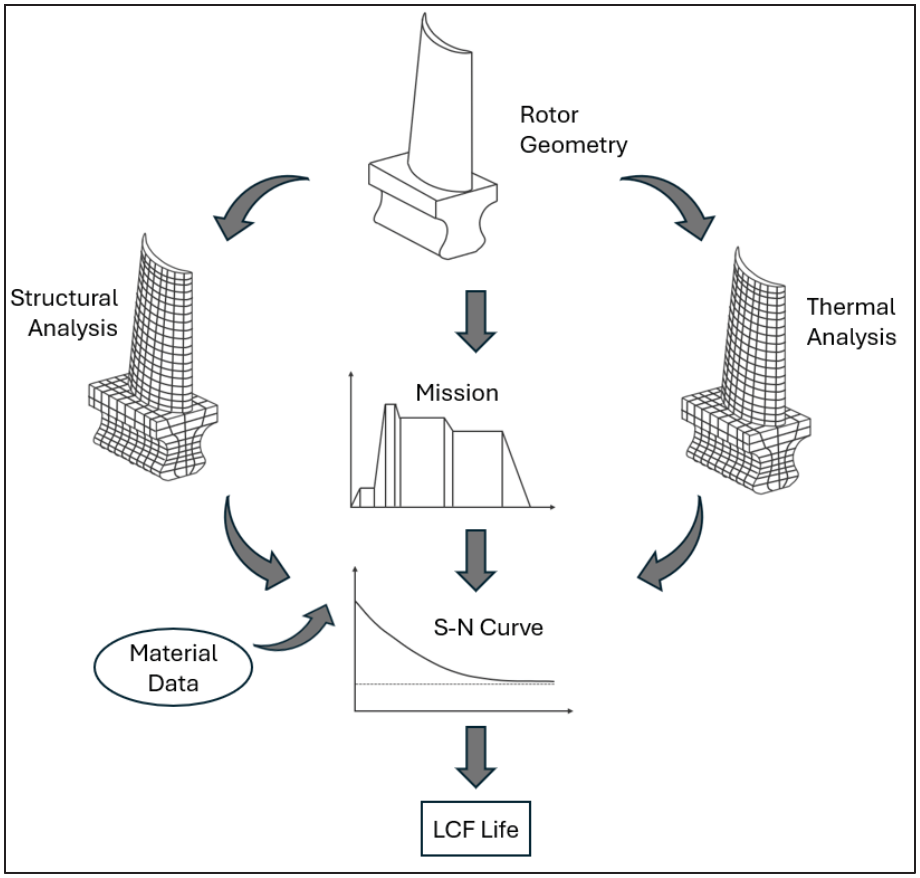

Figure 0.2 Mission analysis process[2]

To address this challenge, this thesis proposes an automation solution. The research is conducted in three main stages: (1) investigation of the automation potential in the current manual process, (2) proposal and implementation of an automation solution, and (3) testing and validation of the final artifact. In alignment with these objectives, an automation tool is developed and presented in this study. The tool prioritizes user experience, introduces process innovations, and integrates thermal, structural, and fatigue life evaluations. A reliable and efficient solution that automates most of the mission analysis process has been successfully developed. This tool was primarily developed for the Detailed Design Phase. However, due to the substantial time reduction it could be integrated at the Pre-Detailed Design Phase.

[2] Figure adapted from Moustapha et al. (2003)

CHAPTER 1

LITERATURE REVIEW

This chapter presents a comprehensive overview of the key concepts, principles, and challenges involved in the design, analysis, and operation of gas turbine engines. It begins by description of the structure and working principles of aero-engines, followed by a detailed discussion of the multidisciplinary nature of engine design processes. The mission cycle of an aircraft is then examined to highlight the operational conditions affecting turbine components. Subsequently, the principles and applications of finite element analysis are presented, additionally, fatigue phenomena and life assessment are explored. The chapter concludes with a discussion on mission analysis, including thermal, structural, and lifing evaluations, and the importance of automation in streamlining complex simulation workflows.

1.1 Overview of Gas Turbine Aero-Engines

Gas turbines are the engines used for aircraft propulsion. Gas turbine engines are generally referred to as aero-engines. They mainly consist of five primary components, each with a distinct function. Air is drawn in through the inlet and directed to the compressor with an adjusted orientation. The compressor slows down the air and increases its pressure, preparing it for combustion. In the next stage, the combustion chamber injects the fuel into the compressed air and burns the mixture. The high-temperature air is then delivered to the turbine, which expands and accelerates the gas. As a result, the turbine blades rotate, generates power to drive the compressor. After leaving the turbine, the air enters the nozzle, where it is further accelerated before being ejected from the engine.

Aero-engines are divided into two sections: hot and cold. The cold section consists of the components before combustion, namely the inlet and compressor. The hot section includes the remaining components, starting from the combustion chamber up to the nozzle.

The physics of the flow through the gas turbine engine is ideally modelled by Brayton cycle as presented in Figure 1.1 (Saravanamuttoo et al., 2017). Although gas turbine engines operate in an open cycle, they can be approximated as closed cycles under certain assumptions. Air at ambient conditions is drawn into the compressor, where its temperature and pressure increase isentropically. It then enters the combustion chamber, where combustion occurs at constant pressure, resulting in heat addition. The high-temperature gas moves to the turbine, where it produces power by expanding isentropically. Finally, the gas is released through the exhaust at ambient conditions, completing the cycle of the gas turbine engine with constant pressure heat rejection (Çengel & Boles, 2006).

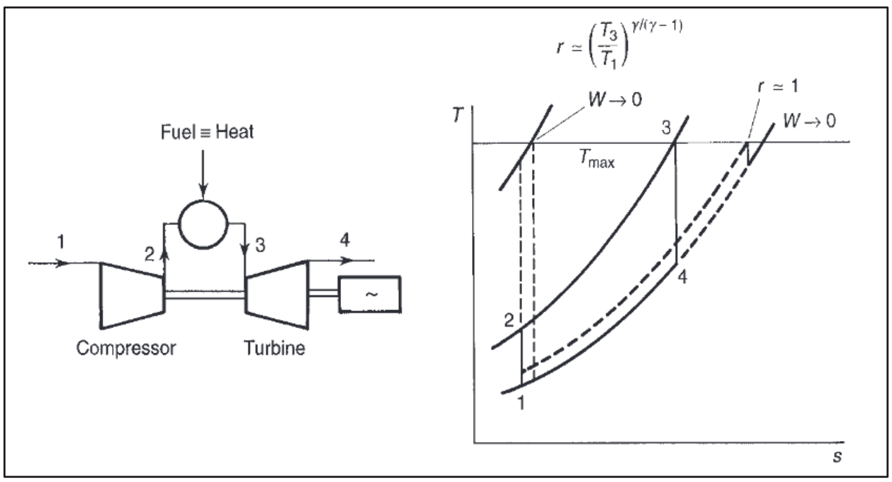

Figure 1.1 Simple Brayton cycle[3]

Gas turbine configuration may vary depending on power requirements and design objectives. In terms of components, multi-stage compressors and turbines may be used according to specific design needs. Multi-spool arrangement are used if high thermal efficiency is (Saravanamuttoo et al., 2017). Figure 1.2 illustrates a schematic of an engine with a three-stage turbine (Boyce, 2002). In this case, the first stage of the turbine is used to power the centrifugal compressor, second and third stages are used to drive the axial compressors and fan rotors, respectively.

[3] Figure taken from Saravanamuttoo et al. (2017)

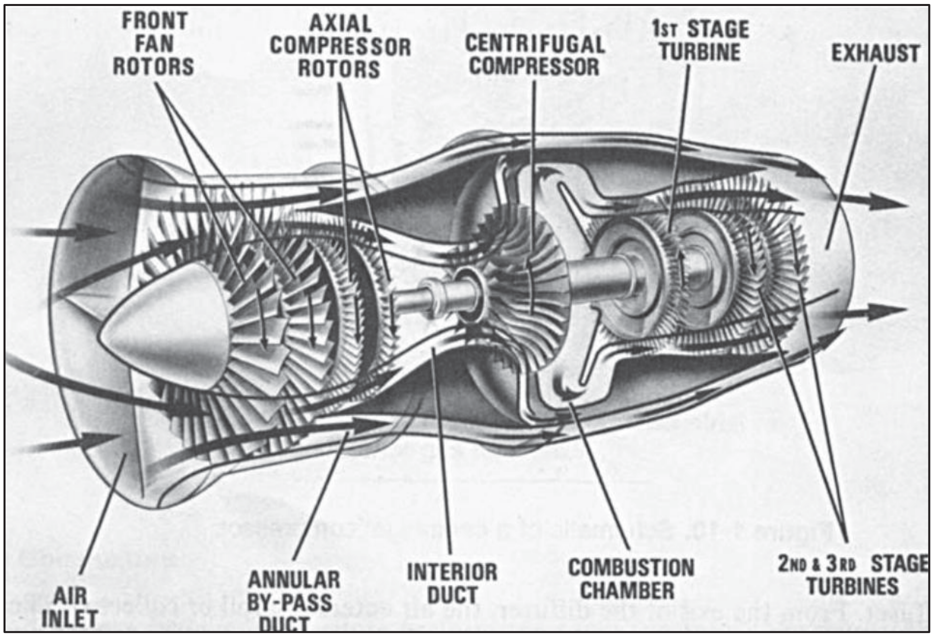

Figure 1.2 Schematic of a gas turbine engine[4]

There are different types of gas turbine engines. Today's engines are derived from turbojet engines (Çengel & Boles, 2006) and they may include additional components to a turbojet configuration, depending on the application. For example, for lower-speed aircraft, turboprops can be used, with an extra spool to drive the propeller. In helicopter applications, turboshafts are used, requiring a complex gearbox and a power turbine. To achieve high subsonic speeds, turbofans are employed, which use a fan at the front and have two separate (hot and cold) jets (Saravanamuttoo et al., 2017) as exemplified in Figure 1.2 (Boyce, 2002).

1.1.1 Aero-Engine Design

Aero-engines design starts from the large system level user requirements. Figure 1.3 illustrates that engine design progresses from the overall system level down to the component level, whereas manufacturing follows the opposite direction (NATO, 2006).

[4] Figure taken from Boyce (2002)

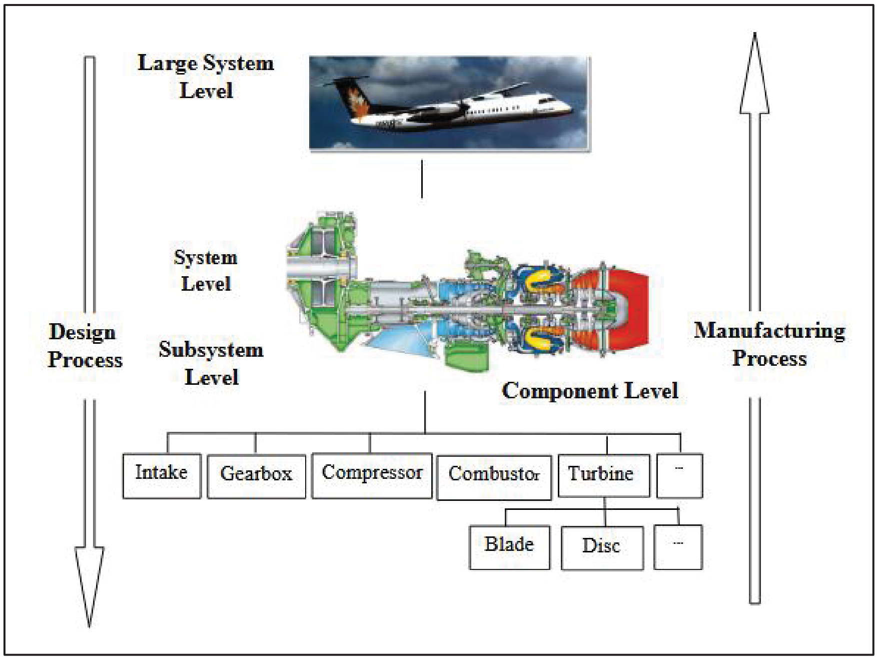

Figure 1.3 Schematic of an engine development program[5]

Engine level design is composed of various components such as intake, combustor or turbine as shown above. According to Peeters et al., this process must be modeled according to its intended purpose, with varying levels of fidelity for each discipline (2005). The preliminary design stage of those components is highly complex. It involves numerous challenges, including the integration of multiple engineering disciplines and the need for extensive design considerations. Figure 1.4 represents the general workflow of engine design, summarizing its complexity and multidisciplinary organization (Saravanamuttoo et al., 2017). The dashed line identifies the preliminary design phase, and it contains thermodynamics, aerodynamics and mechanics to be taken into consideration. Iterative and complex nature of an engine design process also can be seen in Figure 1.4 (Saravanamuttoo et al., 2017).

---

[5] Figure taken from NATO (2006)

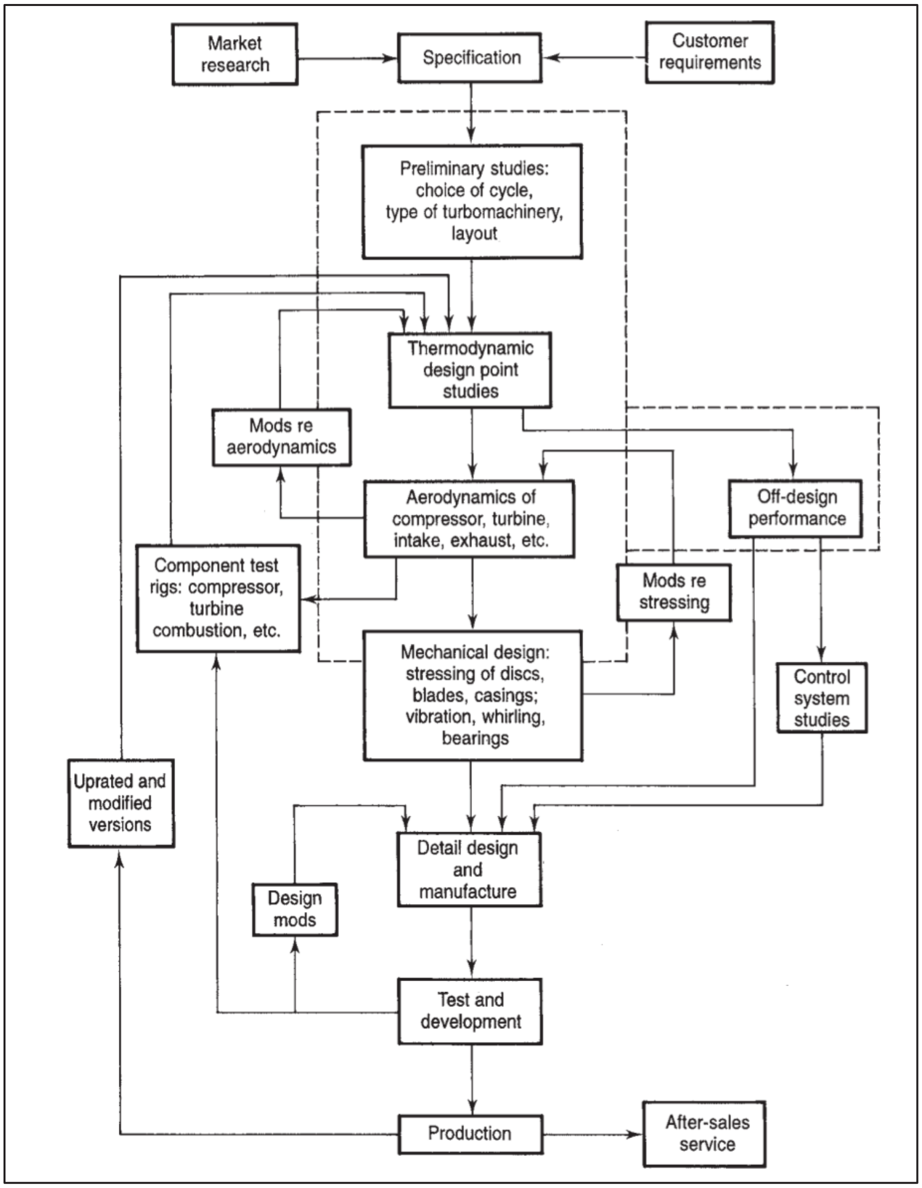

Figure 1.4 Gas turbine design process[6]

According to Panchenko et al., an efficient design must be optimized based on the performance parameters and limitations specific to each discipline, while maintaining a balance among the requirements of key areas such as aerodynamics, structural mechanics, material selection, weight considerations, manufacturing, and cost. They also highlight that effective

[6] Figure taken from Saravanamuttoo et al. (2017)

communication among disciplines is essential to ensure timely delivery of results throughout the design process (2002).

Another one of the key challenges in the development of high-performance gas turbine engines is the need to accelerate the design process. Shortening the overall design timeline has become a critical objective for engine manufacturers aiming to remain competitive. As noted by Peeters et al., the development cycle for new engine programs took approximately five years in 1990. By 2000, this period was reduced to three years, largely due to the integration of practical analysis tools into the design workflow (2005).

It is worth noting that human interaction, along with data and software usage and management, plays a significant role throughout the design process. Due to the highly iterative nature of this process, considerable amounts of data and software resources are required. These challenges can be addressed through process automation and tool integration (Paramassivam, 2020). According to Warde et al., as product design gets more and more complex, pressure of competition drives new solutions like framework integration, which enables more efficient assessment of product efficiency and generation of new models (2019).

Ouellet et al. state that, the design of aero-engines conventionally consists of three main phases: conceptual design, preliminary design and detailed design (2014). However, the traditional approach has a significant limitation: it leads to an unbalanced allocation of disciplines during the time-limited conceptual design phase, which restricts the designer's ability to enhance the quality (Panchenko et al., 2002). On the other hand, the intense competition in the gas turbine market forced the experts to use the time more efficiently, in order to reduce the design duration, leading to the new solutions for the design phases. As a result, the design phase is lately considered to have two stages: pre-detailed and detailed design (Moret et al., 2017).

The early stages of engine design, and the ability to manage them effectively, are crucial for engine manufacturers to maintain a competitive advantage. During the pre-detailed design phase, the primary objective is to rapidly generate feasible design solutions, which requires numerous iterations across various architectures and analyses. As Panchenko et al. noted, at this stage, many simplifications and estimations are made to enable the investigation of a wide range of design concepts (2002).

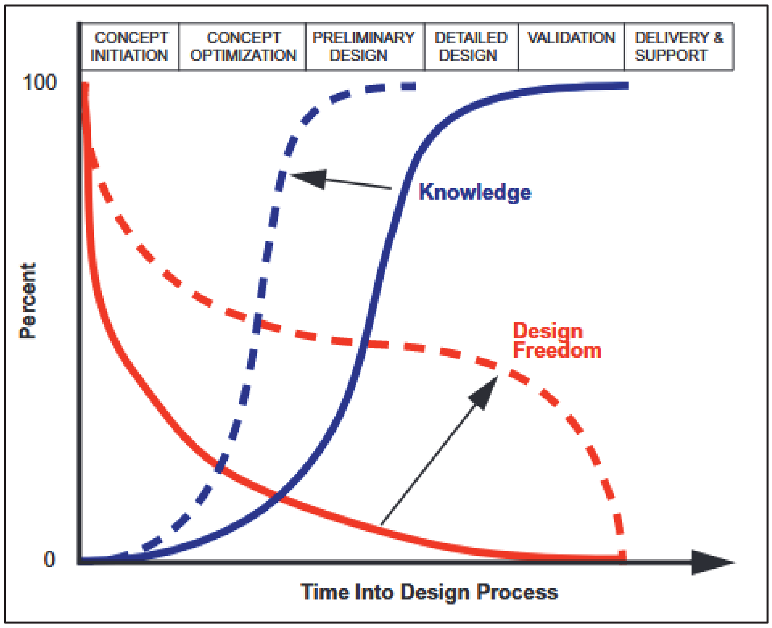

Figure 1.5 Knowledge and design freedom over time[7]

As illustrated in Figure 1.5, the level of knowledge increases throughout the design process, rising almost exponentially during the preliminary design phase, as indicated by the solid lines (NATO, 2006). In contrast, design freedom decreases over time. This highlights the critical importance of the preliminary design phase in engine development. Because design flexibility is significantly limited during the detailed design phase, correcting an error introduced in earlier stages becomes both challenging and costly. In some cases, it may cause a complete redesign, resulting in substantial losses of time, resources, and financial investment.

[7] Figure taken from NATO (2006)

Referring to the dotted lines, the preceding issue can be prevented through the application of multidisciplinary optimization techniques, which aim to increase knowledge acquisition during the early stages and preserve design freedom in the later phases. This can be achieved by evaluating a wide range of configurations, adopting iterative and multidisciplinary design approaches, and implementing automation. As demonstrated in Figure 1.6, these strategies also contribute to reducing overall development costs (NATO, 2006).

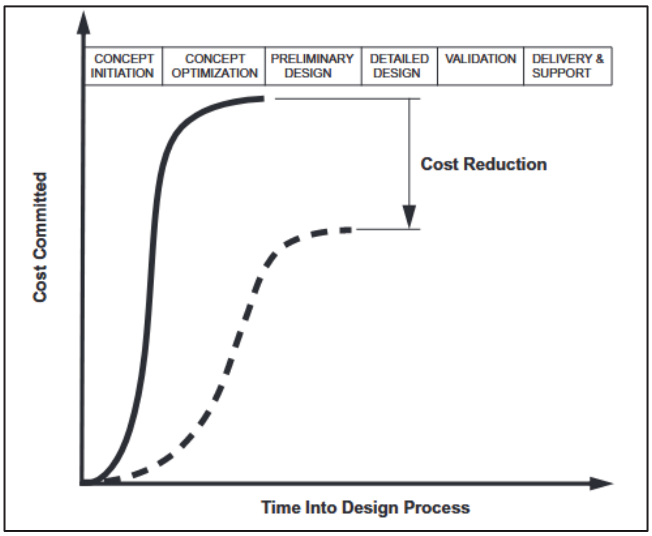

Figure 1.6 Variation of cost by time[8]

The need for conducting a large number of analyses and iterations over various design options in the pre-detailed phase can be addressed by involving numerous experts from different disciplines, thereby improving design accuracy. On the other hand, Panchenko et al. reported that the lead time between iterations, as well as the overall duration of the preliminary design phase, tends to increase due to the complexity of the analyses and the complex interactions among different disciplines. This issue is primarily attributed to the use of non-standard tools and file formats across different teams, as well as the reliance on manual processes (2002). These findings highlight the importance of streamlining workflows among engineers and

[8] Figure taken from NATO (2006)

standardizing tools as critical steps toward reducing design time. Moreover, automation not only reduces lead time in preliminary design phases but also contributes to cost reduction and enhances the reliability of new engine designs.

1.2 Turbine Rotors

Turbine is a critical component for an aero-engine. Figure 1.7 presents a photo of an experimental high pressure turbine (HPT) stage (Papadogiannis, 2023). It can be observed that stator and rotor are composed of a disc and a row of blades.

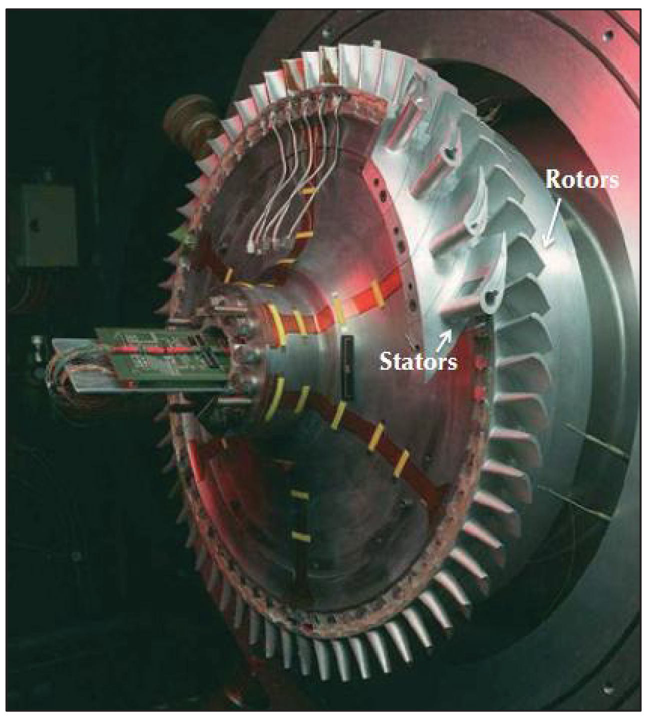

Figure 1.7 HPT of an experimental engine[9]

A turbine is responsible for expanding the hot and compressed gas, produce the mechanical energy and transfer it to the shaft. Figure 1.8 illustrates the variation of flow characteristics

[9] Figure taken from Papadogiannis (2023)

along the turbine (Boyce, 2002). It can be observed that total and static pressure and temperature values decrease throughout the turbine stages. This shows that the turbine blades absorb the energy contained in the hot gas.

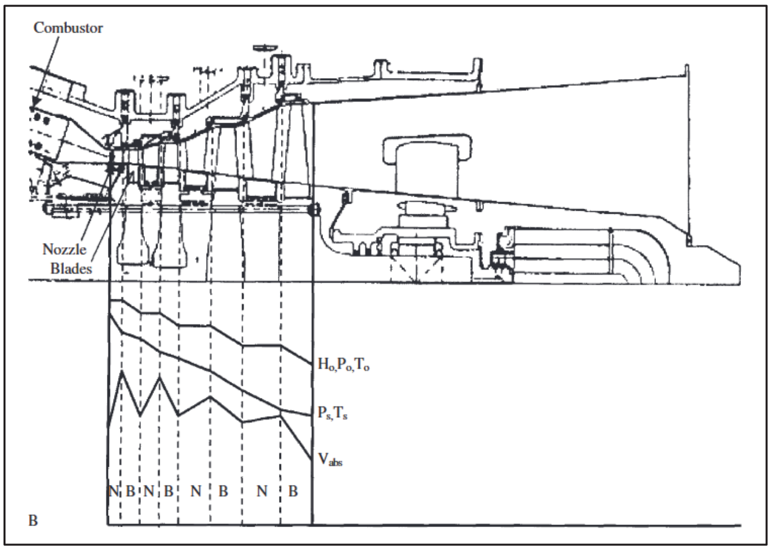

Figure 1.8 Schematic of a turbine flow characteristics[10]

A turbine stage consists of a stator and a rotor. They are commonly referred as vanes and blades as well. Figure 1.9 illustrates the six distinct parts of a rotor blade: shroud, airfoil, platform, fixing, disc and coverplate (Twahir, 2021).

[10] Figure taken from Boyce (2002)

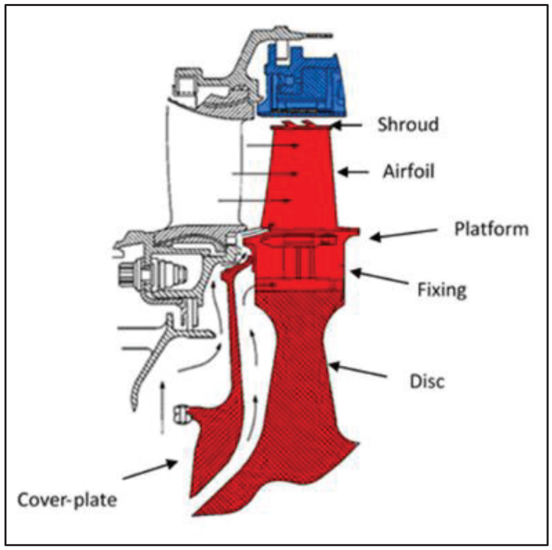

Figure 1.9 Turbine rotor components[11]

As previously mentioned, the turbine receives the hot gases exiting the combustion chamber and rotates at high angular velocities. This creates a highly hostile environment for the turbine rotors. As a result, the blades are subjected to significant thermal and mechanical stresses (Moret, 2018). The cyclic aerodynamic forces also can be added to this statement. Due to these extreme conditions, Liu et al. state that high-pressure turbine blades are among the most critical components in the engine. Moreover, the on-wing time of engines and their operational costs are directly influenced by the service life of turbine blades (2021). This highlights the importance of turbine rotors in terms of maintenance and overall cost of an aero-engine. Therefore, it can be concluded that decelerating turbine life consumption is a key factor in gas turbine design for reducing operational expenses.

1.3 Mission Profile of a Gas Turbine

The mission of a gas turbine is the combination of different phases in a flight. It is also referred as “flight cycle”. For an aircraft propulsion engine, the mission cycle would consist of take-

[11] Figure taken from Twahir (2021)

off, climbing, cruise, descent and landing phases (2003). Figure 1.10 represents the variation of consumed engine power over a typical mission of a commercial gas turbine. Moustapha et al. report that the engine operation begins at idle and gradually transitions to take-off thrust through controlled throttle input. During take-off, both engine power and turbine temperature increase. Once the aircraft reaches cruising altitude, the engine is throttled back and operates at a reduced power setting. During descent, thrust is further decreased, and reverse thrust may be applied upon landing, before the engine returns to idle. Typically, only one full throttle cycle occurs during a single flight (Moustapha et al., 2003).

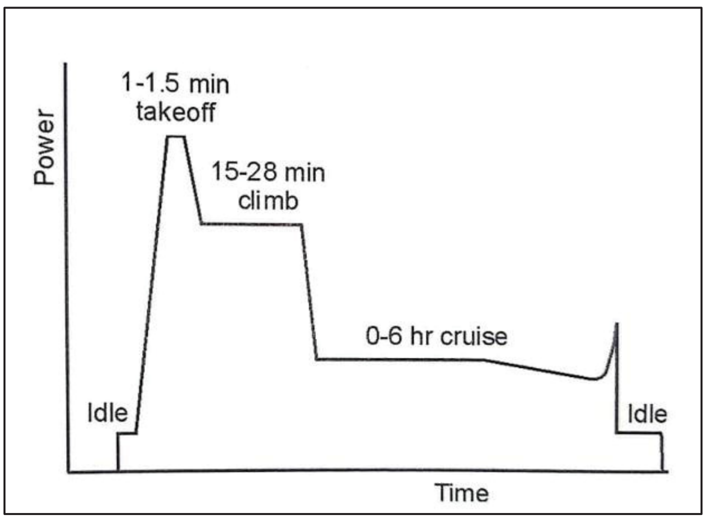

Figure 1.10 Typical commercial engine mission[12]

An example mission profile is illustrated in Figure 1.11, where variations in altitude and fuel flow are presented throughout the flight (Tinga et al., 2000). A noticeable increase in fuel flow is observed during the take-off phase, while significantly lower fuel consumption is evident during the cruise segment.

[12] Figure taken from Moustapha et al. (2003)

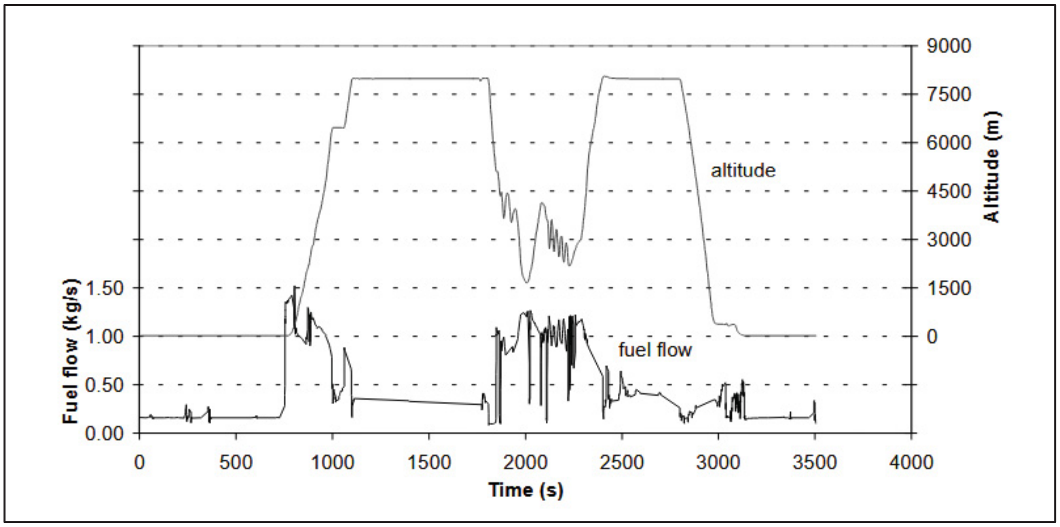

Figure 1.11 Variation of fuel flow and altitude during the selected mission[13]

Due to the transient nature of a flight cycle, the turbine is subjected to varying levels of temperature and stress throughout each mission. Figure 1.12 illustrates the variation in stress experienced at the tip, midspan, and hub sections of a turbine blade for the mission presented in Figure 1.11 (Tinga et al., 2000). A noticeable increase in stress occurs during the take-off phase, where the most severe operating conditions are encountered. During the cruise phase, stress levels are reduced, and they decrease even further during descent.

[13] Figure taken from Tinga et al. (2000)

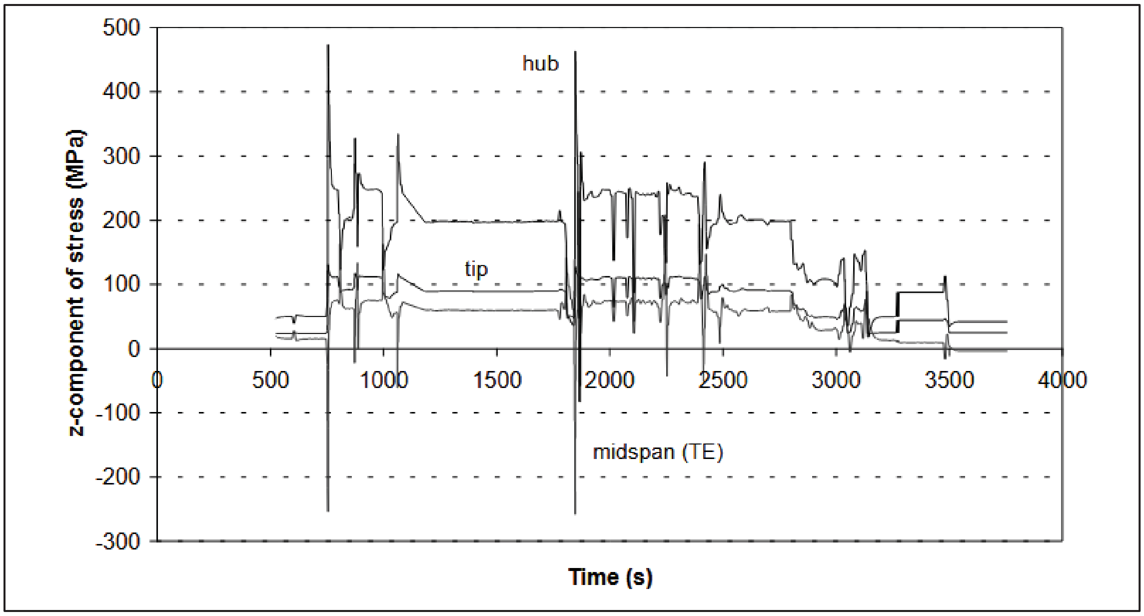

Figure 1.12 Stress variation during Mission cycle at three blade locations [14]

Liu et al. stated that the service life of turbine blades is highly dependent on in-service operating conditions, such as take-off thrust derating (2021). This highlights the importance of mission definition in turbine design, particularly considering the time-dependent thermal and structural stresses to which the blades are exposed. Moreover, the mission profile of an aero-engine may evolve as the design process progresses. Therefore, numerous analyses must be carried out to ensure the turbine's reliability and safety under its intended operating conditions.

1.4 Finite Element Analysis and Application to Blades

Finite element analysis (FEA) is a computational technique employed to solve mathematical representations of physical components. It is commonly applied in addressing structural and thermal problems, among various other engineering challenges (Kurowski, 2004). Chakrabarty et al. claimed that this numerical method is used to solve various types of engineering problems, including steady-state, transient, linear, and nonlinear cases (2016).

---

[14] Figure taken from Tinga et al. (2000)

In all fields of engineering, real-world problems are typically governed by high-order nonlinear equations that do not admit analytical solutions. As a result, alternative approaches must be developed in place of exact solutions. According to Chakrabarty, the finite element method is one such approach, particularly suitable for problems involving complex geometries and boundary conditions (2016).

In finite element method, the governing differential equations are first reformulated into a form suitable for numerical solution. This is achieved by approximating the variation of the variables using simple functions defined by the values of the dependent variables at the nodes of each element. The conversion from differential to algebraic equations is typically performed using either the variational principle or the weighted residual method. For each element, a set of algebraic equations is obtained, and when assembled, these form a global system that represents the entire domain. This procedure is summarized in Figure 1.13 (Chakrabarty et al., 2016). Once the set of algebraic equations is assembled, the system can be solved using numerical solvers. The results obtained from finite element analysis are expected to converge toward the exact solution in a stable manner, typically through a damped oscillatory behavior (Chakrabarty et al., 2016).

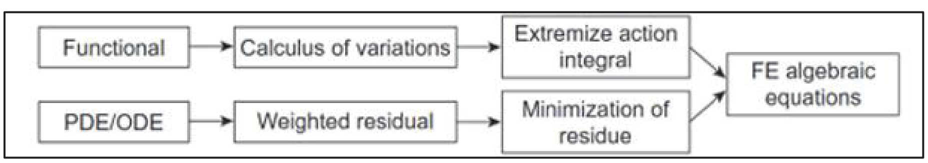

Figure 1.13 Finite element analysis problem formulation steps[15]

By simplifying highly complex equations into a set of algebraic expressions for a wide range of applications, the finite element method offers significant advantages. Logan et al. claimed that it enables the modeling of bodies with irregular geometries and heterogeneous materials,

[15] Figure taken from Chakrabarty et al. (2016)

allows for the application of general loading and diverse boundary conditions, and accommodates nonlinear material behavior. Additionally, it provides flexibility to refine the mesh in regions requiring higher accuracy, modify the finite element model at low cost, and incorporate dynamic effects into the analysis (Logan, 2017).

Finite element models are discretized representations of continuous physical components. According to Chakrabarty, this discretization process constitutes a fundamental aspect of finite element analysis (2016). This step is commonly referred to as “meshing” or “grid generation”. The element shape may vary depending on the nature of the problem. The mesh can be composed of two-dimensional or three-dimensional geometric elements, such as triangles, quadrilaterals, tetrahedra, or hexahedra. For each application, the appropriate element shape must be selected, and a suitable mesh resolution must be applied (Chakrabarty et al., 2016). Lee et al. emphasized that model construction requires careful planning, a thorough understanding of the system, and precise execution (2019).

A typical finite element analysis follows a series of steps, as illustrated in Figure 1.14 (Paramassivam, 2020). First, the key parameters must be identified to clearly define the problem. Using these parameters, the geometry of the model is constructed. In the following step, appropriate material properties are assigned to each part of the geometry. Next, an element type is selected based on the problem characteristics and the geometry’s features. Once this is completed, the mesh can be generated. After meshing, the model is prepared for the application of boundary conditions. Finally, the problem is solved using a numerical solver, and the results are either visualized or interpreted for further analysis (Paramassivam, 2020). Additionally, it is worth noting that those steps can be categorized as “pre-processing”, “processing”, and “post-processing”.

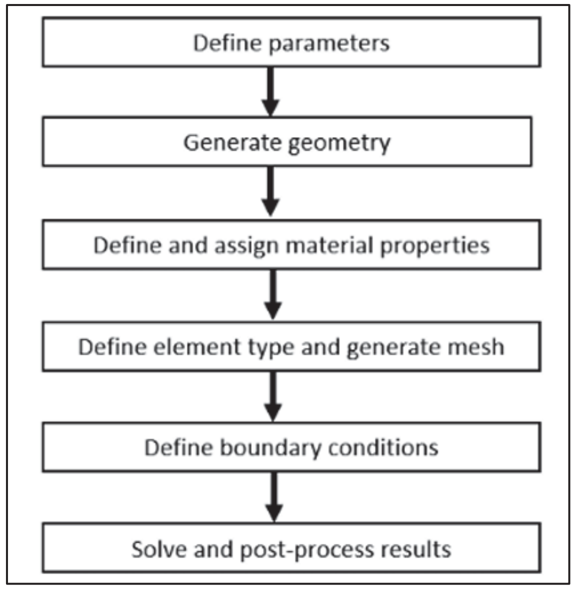

Figure 1.14 Finite element analysis steps[16]

Examples of finite element analyses and their application areas include damage detection, storage and transportation systems, heat release rate estimation, dispersion modeling, and the thermomechanical response of structures (Chakrabarty et al., 2016). Logan et al. further expand these areas to include any type of structural investigations, magnetic and electrical potential analyses, and even applications in biomechanical engineering (2017). Furthermore, Tinga et al. explain the use of the finite element method in fluid dynamics problems to calculate the variations in the heat and heat transfer coefficient along the hot gas path (2000).

1.4.1 FEA in Turbine Blade Design

Among various engineering applications, finite element analysis plays a critical role in the structural design of turbomachinery. Some representative examples are discussed in this section. Figure 1.15 illustrates the finite element model (FEM) of a blade (Singh et al., 2021). Three-dimensional elements are used to construct the model, offering several advantages. They

[16] Figure taken from Paramassivam (2020)

enable high-fidelity results by accurately capturing complex geometries and providing detailed distributions of variables such as stress, strain, and temperature. This level of detail is particularly important when analyzing components with complicated shapes. Only a single sector containing one blade is used in the analysis instead of the entire blade row, due to the cyclic symmetry of the blades.

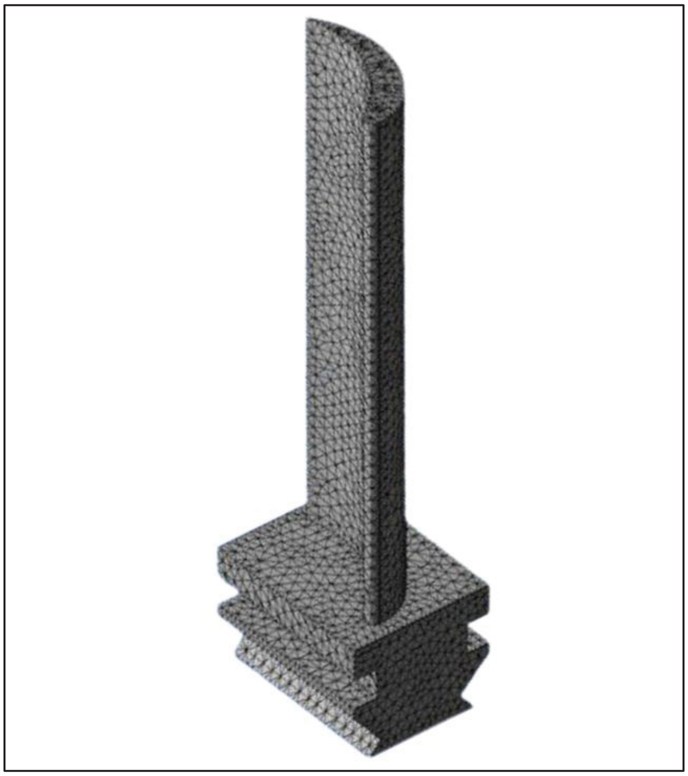

Figure 1.15 3D FEM of a turbine blade [17]

Three-dimensional finite element analyses often require significant computational resources, including both time and cost. Robinson et al. claimed that to support effective decision-making in the early stages of design, finite element analysis is frequently performed on simplified, dimensionally reduced models. Compared to detailed 3D models, 2D models allow for significantly faster analysis while maintaining acceptable global accuracy (2011). In line with

[17] Adapted from Singh et al. (2021)

this principle, two-dimensional modeling may be preferred when the geometry and loading conditions are suitable. For example, axisymmetric parts can be approximated using their cross-sectional profiles (Robinson et al., 2011). A representative example is shown in the Figure 1.16, where a two-dimensional model is used to analyze the hub region of a turbine disc (Zucca et al., 2002). Since the disc hub has a rotationally symmetric geometry and loading, the 2D axisymmetric model provides a sufficiently accurate solution. This approach allows for a substantial reduction in computational effort while still capturing the essential structural behavior of the component.

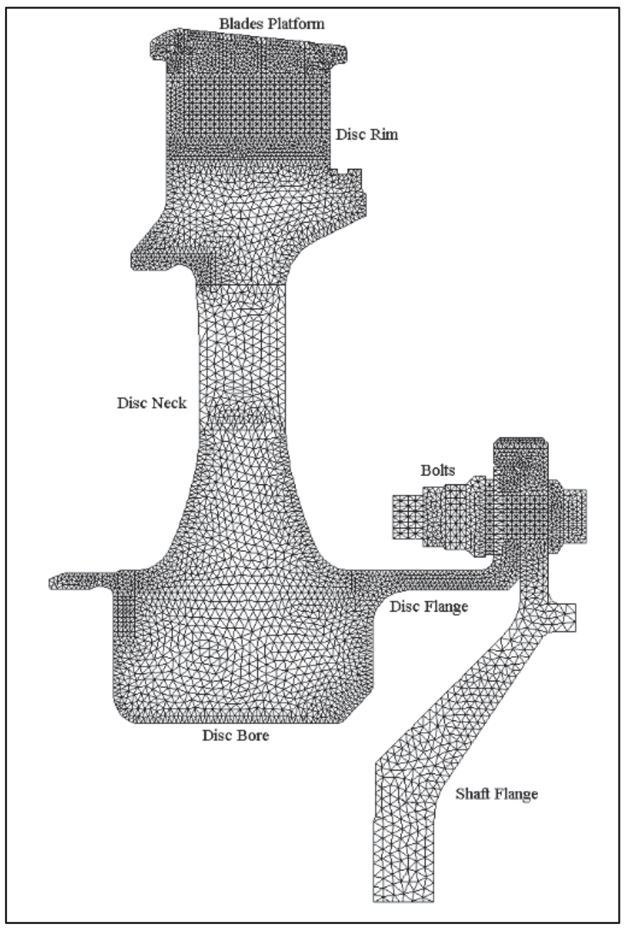

Figure 1.16 2D FEM of a turbine disc hub[18]

[18] Figure taken from Zucca et al. (2002)

Although the use of two-dimensional elements has become a preferred modeling approach in the aerospace industry, they may fail to capture local details with sufficient precision. This limitation will be noticed most in complex geometries, where 2D elements cannot accurately predict the resultant stress fields and hence decrease local accuracy (Robinson et al., 2011). This may lead to poor design decisions and result in a significant waste of time and resources for the company. Shim et al. claimed that the use of three-dimensional elements becomes essential to realistically approximate the key features of structural response in such regions. As a solution, combining lower-dimensional and higher-dimensional elements within a single finite element model can offer a better modeling approach (2002). According to Robinson et al., although this requires more computational effort compared to some 2D models commonly used in industry, they offer improved accuracy in representing the structural behavior of the system, which is an important factor for making reliable design decisions (2011).

According to Shim et al., accuracy and computational efficiency are key issues with finite element analysis, which cause analysts and engineers to look for alternatives that are not only accurate and precise but also economical in the estimation of structural component behavior (2002). To overcome this, various modeling approaches have been explored. Most analysis techniques are founded on lower-dimensional finite elements. However, accurately capturing model local stress concentrations requires merging the lower-dimensional and higher-dimensional elements in one finite element model (Shim et al., 2002). One effective way to achieve this is through the use of hybrid models, also known as mixed-dimensional modeling. This method combines 2D and 3D representations within a single framework, allowing for efficient yet accurate analysis of the engines' critical components. In practice, this is often achieved by applying dimensional reduction to the 3D geometry, where certain regions are simplified and represented using 2D elements. Dimensional reduction refers to the approximation of a specific region of a structure by representing it with lower-dimensional geometric elements (Robinson et al., 2011).

Hybrid modeling reduces the number of degrees of freedom by using two-dimensional elements in suitable regions, while employing three-dimensional elements in complex areas to accurately capture localized stress fields (Robinson et al., 2011). By reducing the degrees of freedom of the model, both computational cost and analysis time can be significantly decreased.

Mixed-dimensional modelling is particularly useful during the early design phases or in parametric studies, where multiple simulations and a high level of accuracy are required. McCune et al. claimed that in many finite element analysis cases, it is often beneficial to integrate lower-dimensional elements with higher-dimensional ones within a single model (2000). A representative example is depicted in the Figure 1.17, illustrating how a turbine rotor geometry can be modeled using a mixed-dimensional approach. In this method, the blade is modeled in three-dimensional mesh elements where two-dimensional mesh elements are assigned to the disc hub.

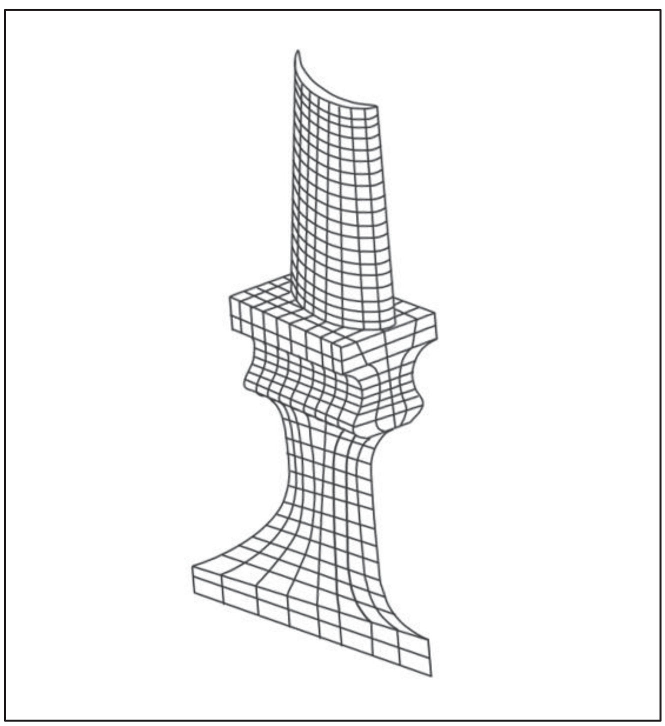

Figure 1.17 Schematic representation of hybrid modeling of a turbine blade

1.5 Life Estimation and Fatigue Analysis

Durability is considered the most critical attribute a gas turbine can possess. The service life of turbine components can be constrained by several damage mechanisms such as creep, erosion, corrosion, and fatigue (Mirhosseini et al., 2020 and H. Liu et al., 2021). Creep, for instance, refers to the slow, permanent deformation of a material under mechanical stress, often leading to microstructural damage such as grain boundary cracks and eventual fracture (Liu et al., 2021). Conversely, fatigue is defined as the accumulation of structural damage resulting from cyclic loading, which can initiate cracks and lead to eventual failure even at stress levels below the material's ultimate strength (Bathias & Pineau, 2013).

In aero-engine turbine blades, fatigue cracks typically arise from self-repeating and varying loads encountered during mission cycles. Different loading types significantly influence not only the fatigue behavior of materials but also the approach used to predict fatigue life (Bannantine et al., 1990). The definition of a mission cycle, involving multiple phases and varying operational conditions, plays a crucial role in determining the service life and fatigue damage of a turbine blade. Moustapha et al. highlighted that cumulative damage caused by throttle variations is the most critical factor affecting blade life (2003).

Turbine blades operate under extreme mechanical and thermal conditions, including high gas temperatures and extreme rotational speeds, which create significant cyclic stresses and local deformations (Bhatti et al., 2006). As a result, the life of turbine blades is often limited by low cycle fatigue (LCF). A comprehensive understanding of fatigue behavior is essential to improve design practices, enhance life prediction accuracy, and meet operational requirements concerning maintenance, cost, and customer satisfaction.

Stress-cycle (S-N) curves can be used to understand the fatigue and life relation. Figure 1.18 shows us the life limit of a material depending on the amplitude of the cyclic applied load (Sonsino, 2007). As the cyclic stress amplitude decreases, the life cycles to failure increases.

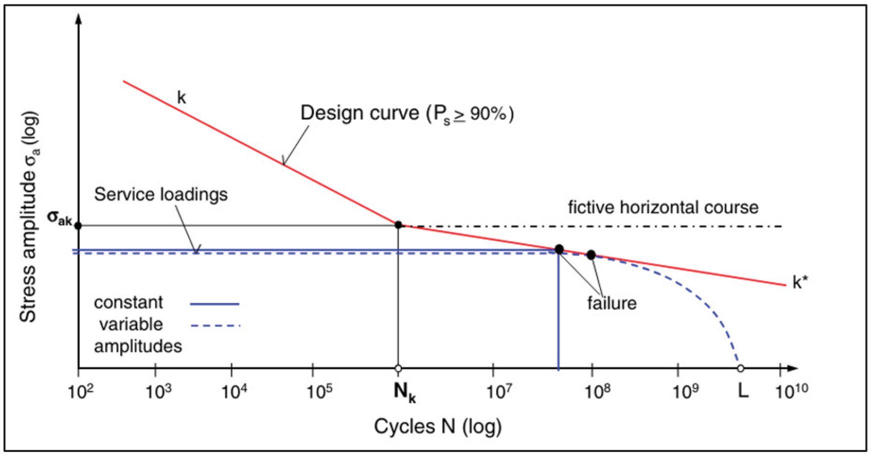

Figure 1.18 S-N curve[19]

Fatigue can generally be classified into two main categories. High cycle fatigue (HCF) typically occurs after more than 10^4 to 10^5 loading cycles, where stresses remain within the elastic range and plastic deformation is minimal (Farahmand et al., 1997). In turbine components, HCF is generally driven by vibrations. On the other hand, LCF arises when cyclic loading exceeds the elastic limit, causing macroscopic plastic deformation and failure within fewer than 10^4 to 10^5 cycles (Farahmand et al., 1997). LCF is common in turbines due to frequent startups, shutdowns, and variable centrifugal loads during operation, i.e. different mission phases, which lead to significant stress and strain fluctuations over time. The difference between LCF and HCF is demonstrated in Figure 1.19 (Yu et al., 2015). The figure presents LCF cycles with blue lines and HCF cycles with red lines.

[19] Figure taken from Sonsino (2007)

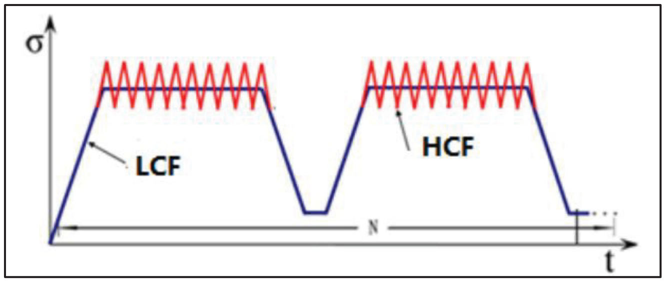

Figure 1.19 Cyclic loadings in stress-time plot[20]

Several studies have been conducted to understand and predict fatigue-induced failures. Mirhosseini et al. conducted failure investigations using techniques such as macroscopic examination, chemical analysis, fractography, and metallography, and developed damage accumulation models based on their findings (2020). The cumulative nature of fatigue damage, dependent on the number of loading cycles, has been discussed by Miner (1945).

The relationship between the life of a structure and the applied load is complex. Although many investigations exist, some authors have avoided establishing definitive laws due to the influence of rare factors such as permanent deformation. Enormous number of tests are needed to state an entirely exact formula relating fatigue and life (Palmgren, 1924). As Palmgren and Zaretsky demonstrated for ball bearings, life prediction can be approached empirically (Palmgren, 1924 and Zaretsky, 1997), while Liu et al. proposed data-driven models (2021).

Lifing models are typically categorized into total life models and crack growth models. Total life models, such as the Palmgren-Miner rule, predict the time to failure without considering the moment of actual crack formation. These are aligned with the safe life approach, which assumes that components are retired before any visible damage appears. In contrast, crack growth models follow the damage tolerance approach, which accepts the presence of initial material defects. These models aim to track crack propagation and remove components before

[20] Figure taken from Yu et al. (2015)

failure becomes critical. They rely on localized stress histories and are valid only at specific locations which crack initiation is expected (Tinga et al., 2000).

Miner states that, due to the absence of primary knowledge on the material behavior, fatigue analysis approaches are proceeding gradually. A solution to handle fatigue, would need precise tools such as loading condition data, S-N curves, means of stress cycles at different levels, an approach to associate different loading cycles (1945). Thus, it is very complex to determine the exact loadings that an engine is exposed during operation and to express the relation between life and fatigue. Additionally, stress concentration factors play a role in fatigue life reduction, as noted by Hu et al. (2013).

Finally, different failure mechanisms such as fatigue, creep, or oxidation may act simultaneously on a component, each requiring a dedicated lifing strategy (Tinga et al., 2000). Thus, selecting an appropriate lifing model depends on the dominant failure mechanism.

1.6 Mission Analysis of Aero-Engines

Mission analysis refers to the comprehensive evaluation of the thermal, structural, and fatigue-related conditions that a turbine blade is subjected to throughout its operational life. The purpose of this analysis is to evaluate the mechanical and thermal loads encountered under various operating scenarios such as take-off, climb, and cruise and to investigate their effects on the component. Finally, the results of mission analysis serve as the basis for evaluating the blade’s structural integrity and service life. This process is typically conducted through a combination of thermal, structural, and lifing analyses, and is often implemented using numerical methods within commercial software environments such as ANSYS, Abaqus etc. or in-house codes.

1.6.1 Thermal Analysis

Thermal analysis constitutes the first step of the mission analysis and is conducted to calculate the temperature distribution within the component. To accurately capture the thermal behavior over the course of the mission, a transient analysis must be performed, accounting for time-dependent variations in physical quantities. The governing equations for thermal analysis are derived from thermodynamics and heat transfer principles.

To apply thermal analysis equations for the geometry, finite element analysis is employed through numerical methods. The thermal model is a finite element model used to calculate the temperature distribution in the component (Tinga et al., 2000). Kurowski states that, although it may seem less intuitive compared to structural analysis, it generally requires less computational effort. In thermal analysis, the primary unknown is temperature, which is a scalar quantity. As a result, only one degree of freedom needs to be assigned to each node in a thermal finite element model, regardless of whether the model is two-dimensional or three-dimensional (2004).

To ensure that the thermal analysis problem is well-posed, not only material properties must be given, but also appropriate boundary conditions must be defined. A well-posed problem is one that yields a unique and stable solution under given physical conditions. In the context of thermal finite element analysis, this typically requires specifying either temperature (Dirichlet), heat flux (Neumann), or combination of those (Robin) boundary conditions (Cummings et al., 2015) on the surfaces of the model. Without proper boundary conditions, the thermal problem may become underdetermined.

Finally, it is crucial to evaluate the temperature distribution along the geometry, as temperature gradients induce stress, which must be accounted for in the structural analysis.

1.6.2 Structural Analysis

The structural model, also called mechanical model, is the finite element model that calculates the stress and strain distribution within the component depending on the temperature distribution and loading conditions (Tinga et al., 2000). The primary objective of this analysis is to evaluate whether the structural integrity of the turbine blade can be maintained throughout its mission and to determine its resistance to different loads. Structural analysis aims to obtain the stress and strain at each node, considering the defined boundary conditions. For the three-dimensional model, each node can have three to six degrees of freedom depending on type of element used to model the component. This makes the structural simulation more computationally expensive compared to the thermal analysis.

According to Tinga et al., stress in a rotating part is primarily a result of two reasons: centrifugal forces generated due to rotation and temperature gradients in the material (2000). Bhatti et al. also emphasize that a turbine disk should be evaluated under both centrifugal stresses resulting due to high-speed rotation and thermal stresses caused by non-uniform temperature distribution (2006). These combined loads must be included in the structural analysis to satisfy the mechanical requirements of the component throughout the mission profile.

The structural model is a finite element representation used to compute the distribution of stress and displacement within the component. It is essential to evaluate these quantities accurately along the geometry, as they contribute to fatigue, which will be considered in the subsequent life analysis. It should be noted that the structural analysis problem must also be well-posed, which requires the definition of appropriate material properties, boundary conditions, and applied loads.

1.6.3 Lifing Analysis

Fatigue life evaluation is crucial for ensuring the structural integrity and safety of turbine blades. It assists not only in improving blade designs but also in identifying repair strategies for damaged components (Yu et al., 2015). Once the stress and strain distributions over time are obtained from the structural analysis, lifing analysis can be conducted to determine the component's life in terms of cycle number.

Accurately calculating the remaining service life of turbine blades is critical, as it directly impacts engine on-wing time and overall operational costs (Liu et al., 2021). Effective lifing analysis supports maintenance planning and ensures safe engine performance. As Tinga et al. stated, extending inspection intervals and component life significantly reduces maintenance costs (2000). Achieving this requires dependable components, designed and validated through extensive and reliable lifing analyses conducted early in the design process.

In this study, the lifing analysis specifically focuses on evaluating LCF behavior, regarding to the fact that cyclic loads may induce localized plastic deformations on turbine blade. This may lead fatigue damage and potential fracture in highly stressed areas. Depending on availability, either commercial solvers or in-house developed codes can be used for life assessment, considering the material properties and the pre-calculated stress or strain results obtained from the structural analysis.

1.7 Automation in Gas Turbine Design

In mission analysis, which involves a large number of simulations across thermal, structural, and lifing analyses, automation is especially critical due to the scale and complexity of data. This study focuses on automating these procedures as much as possible to improve consistency and reduce design cycle and overall mission analysis time. Automation, a central component of the Industry 4.0 paradigm, plays a critical role in transforming traditional industrial workflows into faster, more flexible, and more accurate systems. Originated in Germany and often labeled as the fourth industrial revolution, Industry 4.0 emphasizes digitalization, real-time data, and machine learning, by placing automation at the core of modern design and manufacturing strategies (Oesterreich & Teuteberg, 2016).

Over the last three decades, the gas turbine industry has undergone significant technological advancements, particularly in terms of digital tools and analysis methods (Ouellet et al., 2014). As a result, automation has become essential for streamlining design processes and maintaining competitiveness. The shift from test-based design in the 1990s to analysis-driven approaches in the 2000s, as observed in turbine engine development cycles, demonstrates the growing importance of analytical tools in reducing development time (Peeters et al., 2005).

This evolution has made workflow automation essential, not only to accelerate processes, but also to eliminate repetitive, low-value tasks. However, as Tarkian notes, automation should be implemented carefully to ensure it supports, rather than supresses, engineers' creativity (2012). Engineers generally avoid inefficient work (Mendel, 2011), and automation addresses this by minimizing time-consuming tasks such as model setup, data entry, and file organization (Peoc'H, 2019 and Tarkian, 2012).

McManus et al. report that up to 70% of an engineer's time can be spent on non-value-added activities (2005), while Peoc'h states that 40% of engineering time may yield no useful knowledge (2019). Workflow automation aims to reduce this waste and increase time available for high-level design thinking. Notably, its purpose is not to replace engineers but to assist them, allowing greater focus on analyzing results and making informed decisions.

In the context of mission analysis, automation enables engineers to run hundreds of finite element simulations efficiently, improving the ability to process and interpret large data sets. This, in turn, allows for more iterations and enhances the reliability of final results. Automated analysis methods are well-suited for handling complex, data-intensive simulations and producing more realistic designs under budget constraints.

An integrated and automated framework is especially vital in multidisciplinary workflows. As highlighted by Panchenko et al., lack of tool compatibility, manual data transfer, and inconsistent file formats between teams contribute to errors and longer iteration times (2002). Standardizing tools and streamlining workflows are essential for effective collaboration, automation addresses both issues directly. In this study, the proposed automation system addresses these issues by creating a unified approach for pre-processing, simulation, and post-processing.

The growing reliance on early-stage modeling and simulation, particularly in mission analysis, further highlights the need for automation. Accurate, purpose-driven models support better design decisions and reduce risk, when they used early (Lee et al., 2019). In the context of mission analysis, this also directly contributes to safety evaluations, where early detection of critical conditions is essential. Moreover, as mission analysis complexity increases, manual handling becomes less feasible. Automating this process ensures that simulations can be completed rapidly and accurately, improving not just speed but also the quality of safety assessments.

NATO reports present that traditional design approaches, which rely heavily on manual modeling and test iterations, are increasingly being replaced by automated simulation modules. These systems enable early integration of analysis, simplify data handling, and facilitate more agile design loops (2006).

It should be noted that in industries such as gas turbine development, reducing the design cycle is a core business priority. Customers demand quicker delivery and lower costs, which in turn require fewer manual tasks and more streamlined processes. The automation of mission analysis responds to this pressure by reducing lead time, improving consistency, and enhancing safety evaluation.

Finally, while automation supports productivity, it does not diminish the importance of human input. Paramassivam states that engineering creativity and interpretation remain central.

Automation tools should be seen as a means of enhancing efficiency, not as replacements for engineering expertise (2020). Building on these findings, this study develops a mission analysis automation framework adapted for analysis tools of gas turbine applications, aiming to improve iteration speed and safety-critical decision-making.

1.8 Chapter Conclusion

In this chapter, a detailed literature review is conducted as part of the research study. Key aspects of aero-engine design, lifing and FEA analyses of turbines, and the role of mission analysis in the design process are explained.

CHAPTER 2

PROBLEM DEFINITION AND METHODOLOGY

This chapter defines the primary problem addressed in this thesis and presents the methodology followed throughout the study. First, the current limitations and challenges of the manual process are identified, highlighting the need for automation. Once the foundation for the automation proposal approach is established, the scope and objectives of the project are discussed. Finally, the general methodology is described, including the research approach, tool development strategy, integration process, and the coding approach used in the implementation of the automation solution.

2.1 Problem Statement

The mission analysis of a turbine blade consists of several sequential analysis steps. Even though it primarily focuses on the life of the turbine rotor, transient thermal and structural considerations must be taken into account to accurately evaluate its lifespan. This procedure is summarized in Figure 2.1. A single turbine geometry can undergo several mission profiles, making the mission analysis process highly repetitive, as the rotor life must be evaluated for each distinct mission definition. As highlighted in Figure 2.1, to fully analyze a mission profile, first, the rotor geometry is defined and material data is assigned, then transient thermal and structural analyses are conducted, finally, lifing evaluation is performed based on the outcomes of previous steps.

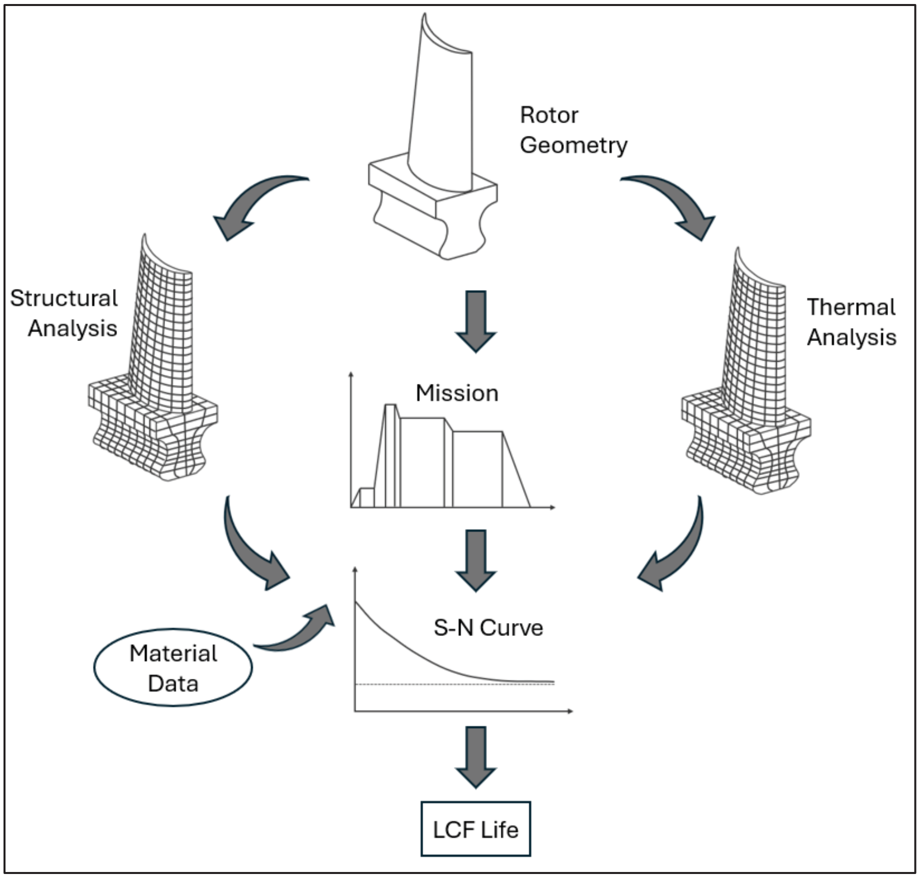

Figure 2.1 Components of mission analysis process[21]

Traditionally, such analyses are performed manually. Due to the repetitive nature of mission analysis and the involvement of human interactions, reliance on manual processes significantly increases the risk of human error. This may lead to longer analysis times for turbine rotor life and extended project timeline. In addition, it causes to spend valuable engineering time on non-value-added tasks. To address these challenges, this study focuses on exploring potential automation opportunities within the current mission analysis process.

[21] Figure adapted from Moustapha et al. (2003)

2.2 Scope and Objectives

The scope of this study includes life evaluations of turbine rotor geometries. It should be noted that only low cycle fatigue is considered in the lifing analysis. Within the scope of this thesis, iterative tasks will be identified, and the application of automation will focus exclusively on these repetitive processes. Once the iterative workflow is defined, upstream activities will be excluded from the scope, and all upstream data and information will be assumed to be available beforehand.

This study proposes an agile automation solution for the iterative mission analysis process. It aims to automate this procedure as much as possible. The primary objective of the automation is to reduce the time required for turbine blade life evaluations and reduce the risk of human error. Additionally, it supports the standardization and streamlining of the mission analysis process among engineers. The automation will not only shorten project timelines but also contribute to more accurate design decisions. These improvements would provide companies with a competitive advantage in the market and create opportunities to have more contracts.

2.3 Research Approach

The research will begin with a detailed analysis of the current process in Section 3.1, where the need for automation will be discussed. Potential improvements to the mission analysis process will be investigated, and automatable tasks will be identified. Once the need for automation is established, an automation approach will be proposed in Section 3.2.

The development of the automation tool will be carried out as part of the ÉTS Industrial Research Chair, which was established in 2011, as illustrated in Figure 2.2. The final artifact will serve as a key component of a detailed design system. The design of the application will follow iterative deployment, feedback collection, and improvement cycles, with user experience placed at the core of the project. Throughout the development process, engineering best practices will be applied to ensure quality and reliability. The implementation of the automation will be demonstrated in Section 3.3.

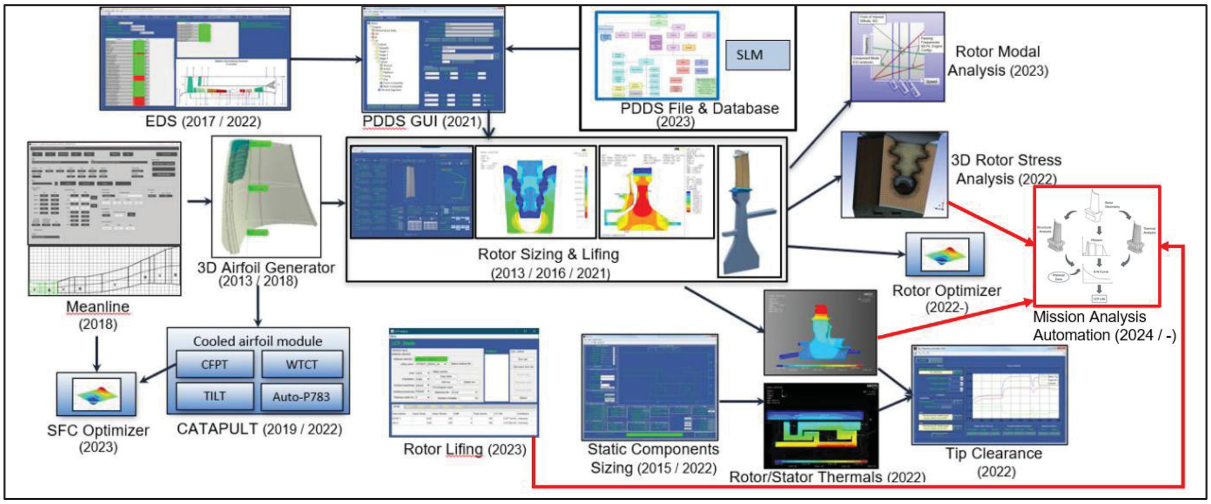

Figure 2.2 Mission analysis automation as a part of ÉTS IRC projects[22]

Once the automation tool is designed and implemented, it will be tested on two turbine blade rotor geometries: one from a low-pressure turbine and one from a power turbine. The results will be presented in CHAPTER 4, focusing on both the validation of the outcomes and the evaluation of the tool's efficiency.

2.4 Methodology of the Automation Tool Development

This study focuses on the development of a tool that automates the manual mission analysis process and integrates multiple analysis tools within a united workflow. In this approach, each analysis tool in the workflow will be considered as a black box. The side responsibilities of the automation software will include verifying input accuracy, preparing analysis run batches, delivering the correct inputs to the analysis tools, checking the outputs, and informing users of process information and errors within the workflow. The primary focus will be to automate the overall process as much as possible and streamline it for all users.

[22] Figure adapted from Ahmed (2025)

During the tool development, principles of Action Research (AR) methodology are planned to be followed, as described by Sein et al. (2011) and Twahir (2021). This approach is particularly adequate to address practical problems through iterative development, user feedback, and continuous improvement. Since mission analysis for turbine blade design is complex and specific to the field, the AR method provides a flexible way to develop a practical tool that fits mission analysis workflow.

In AR methodology, requirements are not necessarily fixed at the beginning but are progressively shaped as the project evolves (Sein et al., 2011). This aligns well with the development of the automation tool, where inputs from engineers will shape the architecture of the tool. Rather than aiming for a fully defined end product from the beginning, the tool will be developed through iterative prototypes, tested and reviewed by users, as also illustrated in the Figure 2.3 (Sein et al., 2011).

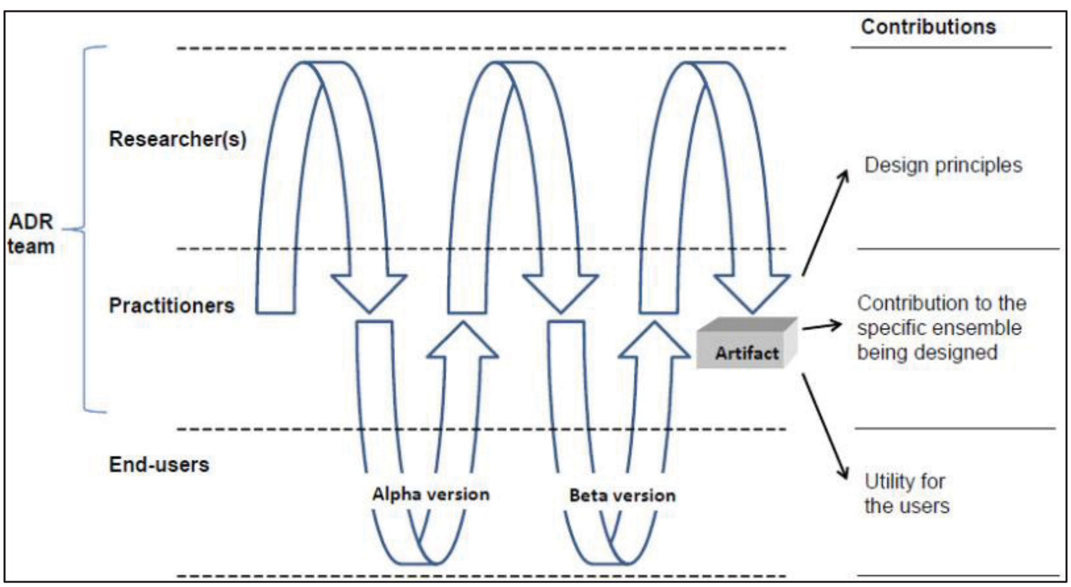

Figure 2.3 Action research methodology[23]

[23] Figure taken from Sein et al. (2011)

This approach also enables automation tool to improve the user collaboration and standardized data handling practices. The iterative process encourages strong user engagement throughout development, ensuring that the final product not only meets functional requirements but also aligns with the practical preferences of engineering teams.

In the context of the automation tool development, the artifact corresponds to the automation software itself, which integrates tools such as Boundary Condition Generator (BCG), FEA solver, and Lifing Tool. The Action Research methodology enables the tool to be developed by focusing on one analysis step at a time, allowing a beta version to be released for engineers for each step and taking their feedback into consideration for subsequent versions.

2.4.1 Integration

To reduce costs and development time in turbine rotor design, integrating automated tools into the mission analysis workflow has become increasingly important. As highlighted by NATO, the integration of engineering tools allows for more efficient coordination across disciplines, better information sharing, and smoother data management (2006).

Automation not only minimizes repetitive manual work but also improves collaboration between different engineering teams. This ensures that all processes are aligned and reduces the risk of delays or errors. Overall, an integrated and automated framework enables same quality, reduced costs, and shorter cycle times (NATO, 2006), aligning directly with the project objective of automating the mission analysis process for turbine rotors.

Developing an automated tool that connects software programs within an engineering framework involves addressing several challenges. First, integrating processes, workflows, and procedures is necessary to automate workflows that are traceable, reduce duplication, and align with product lifecycle management. Second, integrating tools is critical because different tools generate various types of data, which must be converted and exchanged in a reusable and traceable format across activities. Lastly, integrating disciplines is essential, as distinct disciplines and tools produce different outputs. Automation should establish connections between these outputs, enabling interaction and creating an automated feedback loop among engineering tools (Binder et al., 2019).

Automated mission analysis process will integrate the different software programs used for different stages of the framework. This will allow engineers to efficiently coordinate workflows, share data smoothly, and ensure that tools can work together without an interruption.

The automation tool will be designed to integrate Boundary Condition Generator (BCG), a FEA solver for thermal and structural analyses, and Lifing Tool, which are used at different stages of the mission analysis workflow. Each of those software tools require specific inputs with proper formats and gives various outputs. Even though these tools were used in the manual workflow, integrity was not maintained, the connections were weak, and the integration processes relied heavily on manual actions. As part of this integration, the automation tool will ensure that the files exchanged between these software tools are consistently formatted and compatible with each system's requirements. In addition, the tool will manage the proper storage and organization of these files throughout the process, reducing the risk of errors caused by manual file handling. This integrated approach will streamline data flow, improve reliability, and ultimately enhance the efficiency of the overall mission analysis process.

2.4.2 Coding Approach

To ensure an agile and efficient coding practice for the automation tool, various programming alternatives were investigated. After evaluation, Python was selected as the most suitable language due to its flexibility, extensive libraries, and engineers' experience with it, making it ideal for developing the required automation framework.

Among the programming approaches available in Python, Object-Oriented Programming (OOP) was identified as the best fit for this project. OOP provides clear code organization, reusability, and scalability, all of which align well with the project requirements. By applying OOP principles, a modular and adaptable framework can be created, which will be reusable across different mission cases within the automation process.

One of the key requirements of an intelligent integration tool is traceability, the ability to track and regulate data flow throughout the system (Coito et al., 2019). OOP supports this need by structuring the system into well-defined objects that can be used individually and in combination. Moreover, OOP enables modular design, allowing the integration of new modules or updating the existing ones. OOP also allows the system to be decomposed into smaller, more manageable subsystems, significantly reducing code complexity and making the framework more adaptable for future modifications which are needed for intelligent automation framework (Coito et al., 2019).

Another critical advantage of OOP is the inheritance concept, which enables the creation of a general "superclass" that defines shared attributes and methods (Twahir, 2021). This eliminates unnecessary coding across modules and minimizes rework, as common functionalities can be inherited by subclasses. Additionally, OOP facilitates centralization of data, improving the management and accessibility of shared information across the system (Twahir, 2021).

On the other hand, Python offers specific libraries, which enables users to conduct FEA operations directly through Python commands. This capability has high potential for streamlining analysis runs, automating post-processing, and handling model validation. Therefore, it provides a valuable opportunity to enhance the effectiveness of the project.

Finally, Python offers several features that support best practices in software development. One notable example is the "pytest" library, which provides simple framework for both unit and integration testing. Additionally, Python includes powerful libraries for logging, file writing and reading, and data visualization, which is making Python well-suited for developing and maintaining engineering tools.

Overall, Python provides a comprehensive environment aligned with coding best practices. Considering these capabilities, the project aims to deliver a scalable, maintainable, and agile automation solution.

2.5 Chapter Conclusion

This chapter outlined the problem addressed in the thesis, along with the scope and main objectives of the study. It also introduced the overall research strategy, and the methodology followed for the development of the automation tool. Rather than going into detailed technical steps, this chapter provided a detailed overview of the adopted research approach. The following chapter will present a deeper analysis of the current manual workflow, and the design and implementation of the proposed automation solution will be discussed.

CHAPTER 3

ANALYSIS AND DESIGN OF THE AUTOMATION TOOL

This chapter is dedicated to the analysis and design of an automation tool developed in the context of this thesis study. The current manual mission analysis process will first be examined, and an automation solution will be proposed. Then, the tool's architecture and implementation will be discussed in terms of development requirements, overall software architecture, individual automation modules, and complete workflow integration.

3.1 Detailed Analysis of the Manual Process

The existing manual process involves both a defined workflow and an associated dataflow. Workflow is the illustration of the sequence of steps being performed in a process and it focuses on procedure itself. Dataflow refers to how data is manipulated or transferred through a system. In this section, the present manual workflow and dataflow of the mission analysis process will be presented in alignment with the project context and objectives.

3.1.1 Manual Workflow

The mission analysis process is not an isolated procedure for evaluating the structural endurance of an aero-engine turbine blade. It relies on inputs from various engineering teams, including performance, design, turbine aerodynamics, and secondary air systems. This dependency arises from the need for problem specifications before the analysis can begin. For instance, geometry definitions, mission profiles, and aerodynamic loads are essential inputs that must be provided in advance. Once the turbine rotating structures team collects this information, they can initiate the analysis for each mission. This workflow is illustrated in Figure 3.1, where the grey box highlights the repetitive tasks for each mission. Yellow boxes present the manual tasks, where green boxes stand for automated ones. Thus, it can be observed that the overall workflow usually depends on manual actions. Since each engineer conducts analyses differently and often uses their own macros for specific tasks, automation is needed not only to save time but also to streamline and standardize the process across engineers.

The repetitive tasks in the workflow can be classified into four categories: Boundary Condition Generator (BCG), thermal, structural, and lifing analyses. As these categories are identified, each analysis step can be clearly defined, and its potential for automation can be evaluated individually. The following sections focus on BCG, thermal, structural and lifing analyses in of the scope of the thesis study.

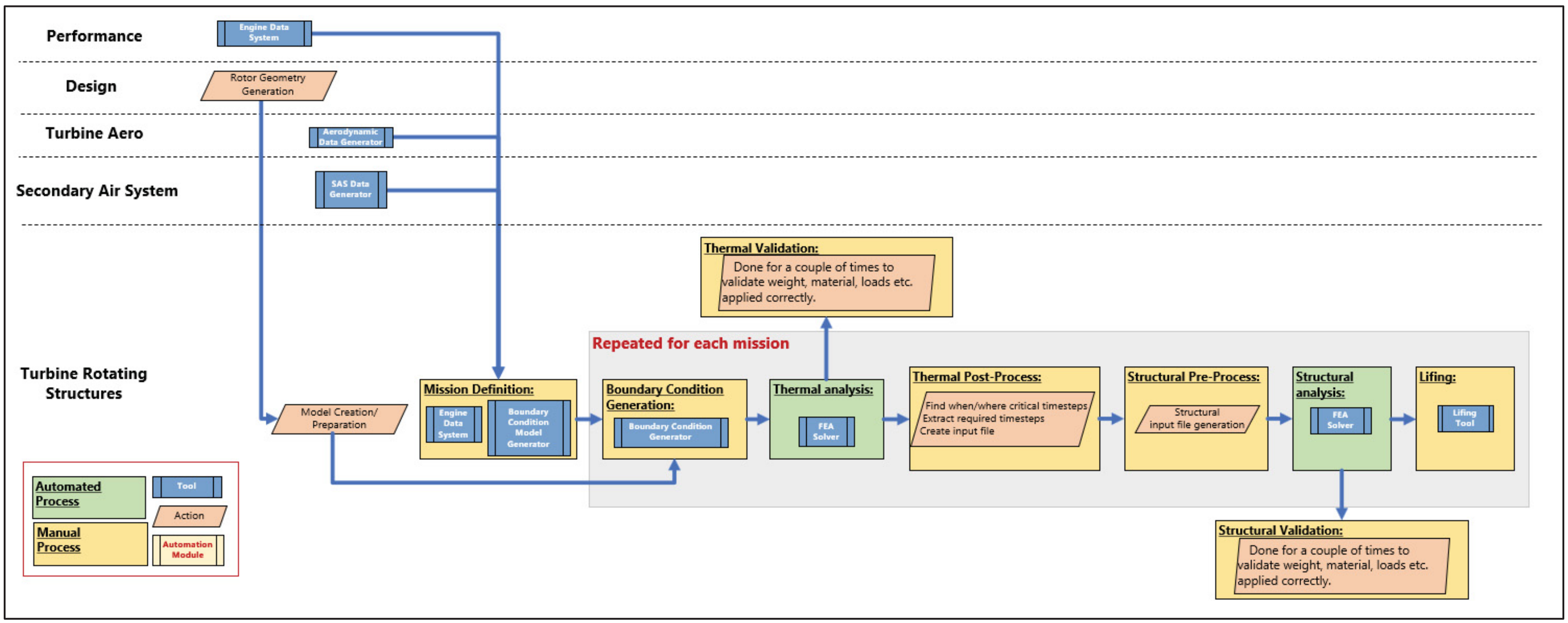

Figure 3.1 Present manual workflow of mission analysis

3.1.1.1 BCG Analysis

An in-house code is used to generate of boundary conditions. It will be referred to as Boundary Condition Generator (BCG). It uses inputs provided by other engineering teams, primarily the geometry and mission definition of the rotor blade. Based on this information, BCG processes that information to provide boundary conditions for the thermal analysis and delivers output files containing this data for downstream use. BCG's graphical user interface (GUI) is used for this process, which depends on manual actions and enhancing risk of human error.

3.1.1.2 Thermal Analysis

BCG analysis can be considered as the pre-processing step for thermal analysis. Once the boundary condition files are received from the upstream process, the thermal case becomes ready to run. A FEA solver is preferred for conducting this analysis.

The analysis execution itself is represented in the green box in Figure 3.1. This indicates that once the solution is initiated in the tool, the calculations are organized and performed automatically. After completion of the analysis, the case proceeds to the post-processing stage. The primary objective of post-processing is to extract significant information from the thermal result file and filter the time steps required for structural analysis. This filtering step is essential for the downstream process, as structural analysis is more computationally expensive than thermal analysis, and not all of the time steps need to be solved. This portion of the workflow is highly manual, as shown in the yellow box in Figure 3.1.

The visualization of results is also manual and depends heavily on the engineer conducting the process. Temperature distributions across the entire geometry or within specific regions can be examined. For time filtering, the first step is to select all critical nodes and assign names to them, as exemplified in Figure 3.2 for a generic turbine geometry. This process is carried out through the GUI of the FEA solver, where the manual nature of the task may result in some mis-clicks, leading to numerous repetitions and significant time loss. Once nodes are defined, the temperature histories of these nodes and the temperature differences between specific node pairs must be exported. These data are then plotted, and important time steps are identified. In addition to thermal results, mission definitions and shaft speed variations must also be taken into account during this selection. Once the critical time steps are determined, the case is ready for structural analysis.

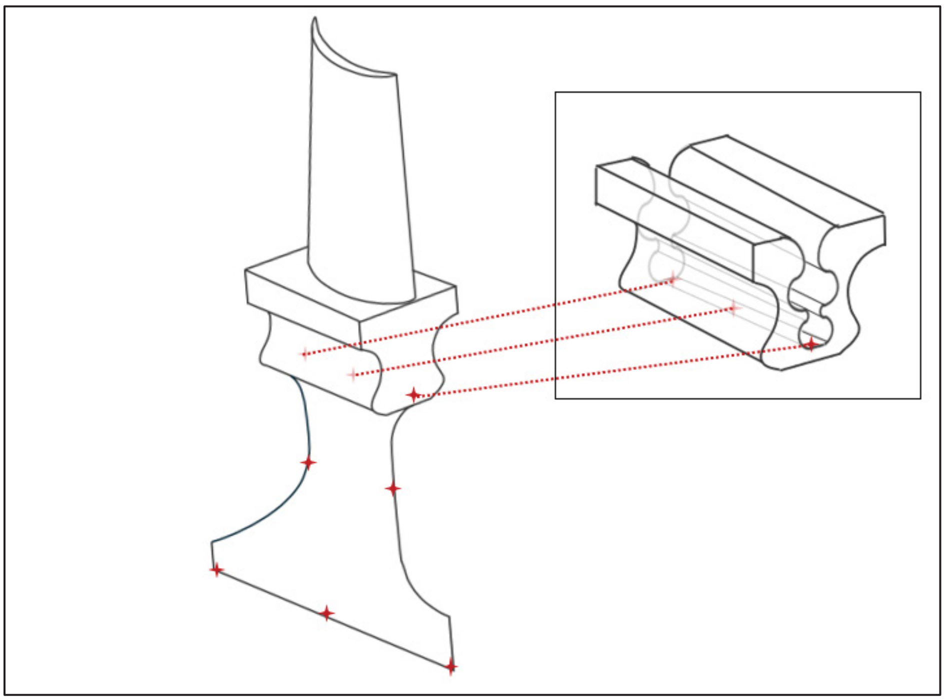

Figure 3.2 Node selection illustration for thermal post-processing

Thermal validation constitutes the final step of the thermal analysis. Although it is a manual process, it is not obligatory for every mission; performing validation for a few representative cases is generally sufficient, allowing the remaining analyses to proceed without it. The primary objective of thermal validation is to ensure that the case has been correctly configured; specifically, the weights, material definitions, and load conditions have been accurately assigned.

3.1.1.3 Structural Analysis

As the temperature distribution is evaluated throughout the turbine rotor body, the structural analysis can be performed. The thermal result file serves as an input for the structural analysis. Once it is generated by the upstream process, the structural analysis can be initiated manually. The first stage involves pre-processing, which includes generating an input file that defines the analysis specifications and time steps for the simulation. Although this file generation is performed manually, it enables the subsequent analysis to run automatically. This stage is represented by the yellow (pre-processing) and green (processing) boxes in Figure 3.1.

Similar to the thermal analysis, a FEA software is used for conducting structural analyses. The pre-generated input file, along with the structural database, is given into the tool. Followingly, the calculations proceed automatically. Once the structural result file is produced, the structural phase of the workflow is complete. While users may visualize these results for examination, detailed post-processing is primarily carried out during the lifing stage of the workflow.

Structural validation is very similar to thermal validation. It is also not required for every mission and primarily aims to ensure that the case is accurately configured in terms of weight definition, material properties, and applied loads.

3.1.1.4 Lifing Analysis

This research specifically focuses on LCF life evaluation. Once the structural results are obtained for the turbine rotor over the mission profile, a life assessment can be carried out to determine the component's endurance.

Lifing can be considered a comprehensive structural post-processing step, as its primary objective is to utilize structural data to predict the component's life. Therefore, in the manual workflow, the lifing process begins with the identification and naming of critical surfaces and components to be analyzed. Although a real turbine geometry can not be presented due to confidentiality, an example of surface and component selection in the lifing process is illustrated for a generic turbine geometry in Figure 3.3, highlighting critical parts of the geometry in red color. As in thermal post-processing, element selection is done via the FEA solver's interface, and the manual clicking involved may cause wasting time due to inefficiency of the GUI.

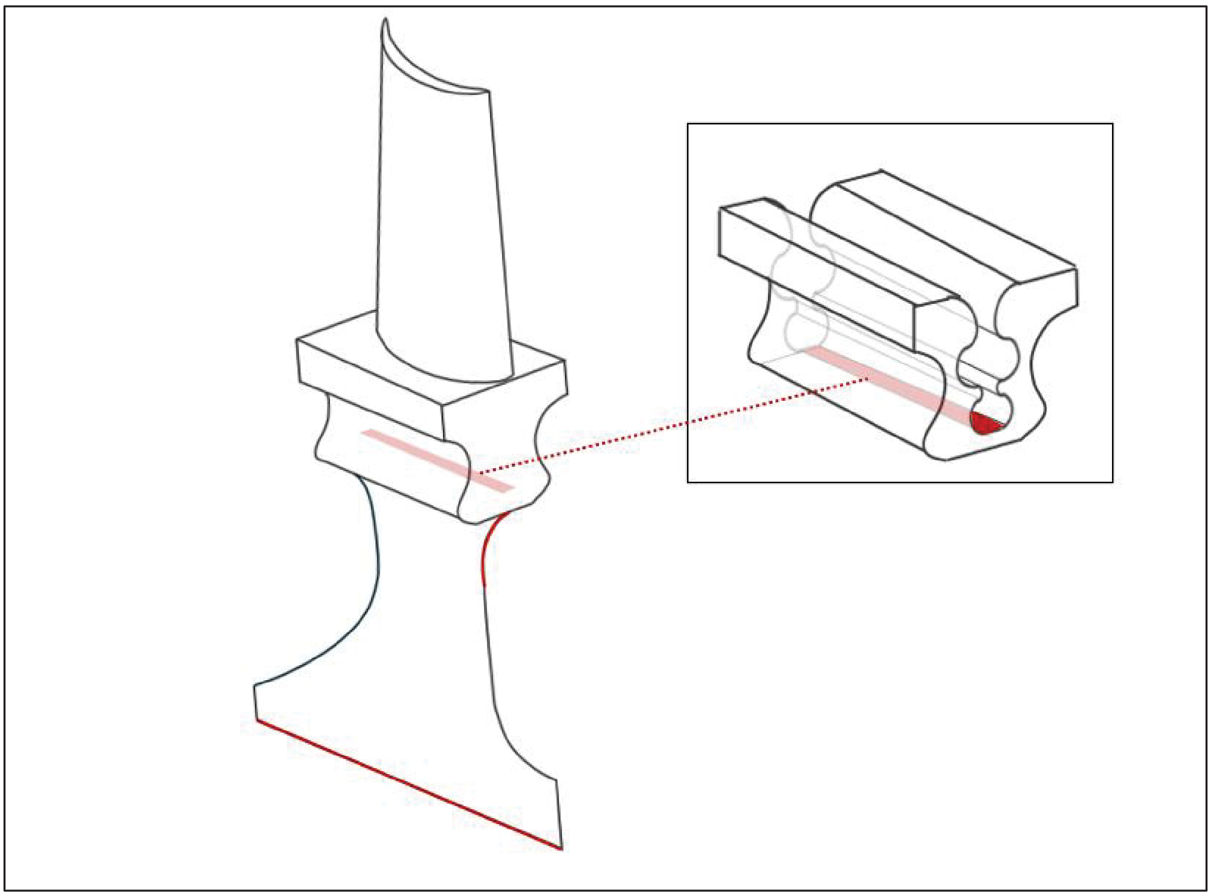

Figure 3.3 Component selection for lifing

Once the named selections are defined in the database, in-house macros and tools are employed to perform the LCF life assessment of the turbine blade. Finally, an in-house code, Lifing Tool is used to complete the component's life evaluation.

3.1.2 Present Dataflow

Data management is a key aspect of mission analysis process, as enormous amount of data is both used and generated throughout the workflow. Therefore, it is essential to manage these files properly. The main dataflow is illustrated in Figure 3.4. It should be noted that the diagram highlights only the main files; in practice, many additional files may need to be created and processed within the automated workflow.

The files presented in Figure 3.4 are specific to a given mission. The files highlighted in yellow represent the main user inputs, each containing specific information for a specific stage of the analysis. For instance, BCG receives its inputs through files that include the mission definition, thermal model database and geometric and aerodynamic specifications. These inputs are processed by BCG to generate boundary conditions required for the thermal analysis.

The thermal analysis step uses this boundary condition files together with the thermal database, and material properties to define temperature distribution across the rotor blade. The thermal results are then post-processed, where specific time steps are filtered and a set of times is selected. This list of time steps determines the time values at which the structural analysis will be executed.

To perform structural analysis, additional files such as the structural database and loading input are required. The structural analysis results are used as input for the life assessment, which also uses a component list to carry out LCF life assessments. Finally, the results are post-processed to generate reports and visualizations, providing an efficient and interpretable output for the user.

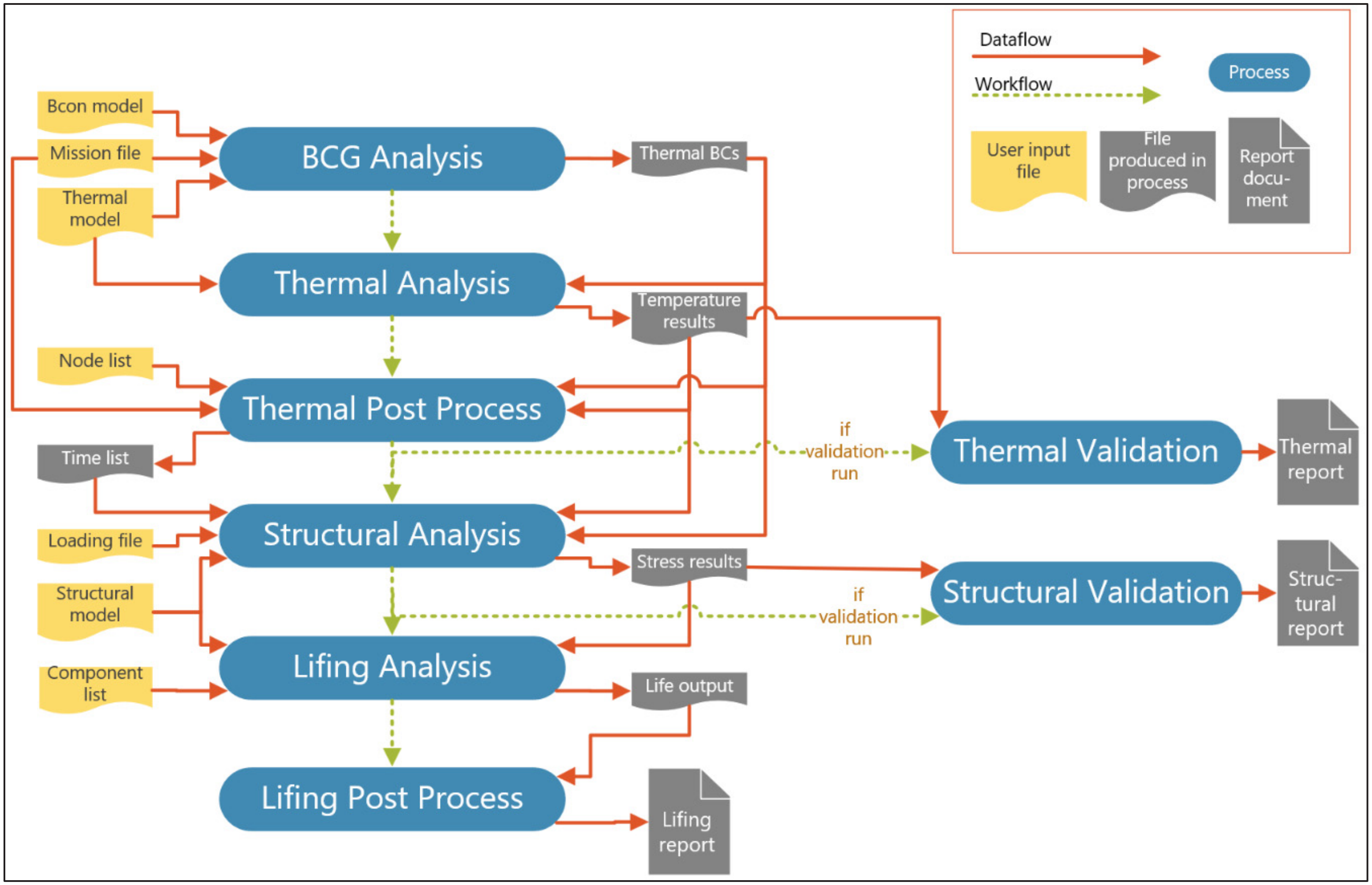

Figure 3.4 Dataflow of mission analysis process

The main outputs of the mission analysis process include thermal analysis results, structural analysis results, and the component life assessment results. Additionally, summary reports will be generated to provide users with clear and brief overviews of the analyses and outcomes.

As previously highlighted, the mission analysis process produces a huge amount of data. Considering that numerous files are generated for a single mission, and that multiple missions may need to be analyzed for a single geometry, the total number of files can increase rapidly. Therefore, proper file handling becomes critical to maintaining folder organization and avoiding errors. To address this challenge, a standardized folder structure, has been adopted by engineers. According to this convention, the result files related to each mission must be stored under a mission-named subfolder within specific analysis directories, ensuring that all inputs and outputs are systematically organized.

All data management and folder structuring tasks are currently performed manually. These tasks are repetitive, time consuming, and do not directly contribute to the engineering value of the analysis. As such, they place an unnecessary work on engineers and are well-suited to be subjected to automation.

3.2 Proposition of Automation

The automation of mission analysis process will require a software development. The research presented in this thesis will aim to design and implement an automation tool built for engineers who run mission analyses for turbine rotor geometries, aligning with industry standards and user expectations. A new environment will be developed from scratch to conduct integration between different analysis tools in the workflow and manage files in the dataflow. The tool will serve to streamline repetitive tasks, enhance traceability, and reduce manual effort in turbine rotor life assessments. The proposed name for this new tool is AMAP, which stands for "Automated Mission Analysis Process".

AMAP is designed to be an integrated tool that automates the mission analysis process as much as possible. Additionally, since maintaining user control over the process is essential, it will allow engineers to intervene when necessary.

The primary assignment of the AMAP tool is to ease engineers' work as much as possible. It will be capable of running on Windows and Linux. Additionally, it will not only automate analysis run but also manage data automatically to follow required folder structure as mentioned in Section 3.1.2.

Among other principles, AMAP will follow the existing workflow and dataflow without modification. This approach not only strengthens its foundation by relying on a verified and trusted process but also enhances the user experience by preserving the workflow users are already familiar with.

3.2.1 Future Automated Workflow

AMAP will provide an automated workflow, as illustrated in Figure 3.5, which is derived from the current manual workflow shown in Figure 3.1. The fundamental approach is to replicate and automate existing manual practices.

In Figure 3.5, the green boxes represent the tasks targeted for automation. AMAP aims to automate each of these tasks individually. The preferred tools for each type of analysis will remain the same. Although the repetitive nature of the tasks will be preserved, the tool itself will manage the entire process. Automation is expected to significantly reduce human error, required effort, and overall processing time.

While GUIs of the tools being used in the manual workflow, batch modes of the tools will run in automated workflow. AMAP will not only automate the execution of analyses but also ensure continuous integration among them. Similar to the manual approach, thermal and structural validations will be excluded from each mission cycle, as they are not required for every single mission analysis.

The only significant change in analysis steps between the manual and automated approaches is the proposed removal of the structural pre-processing phase. The tasks of this phase are integrated into the thermal post-processing and structural analysis phases as a result of the efficiency introduced by the automated workflow.

In conclusion, AMAP demonstrates that automation is particularly well-suited for mission analysis due to its repetitive nature and offers several advantages. It not only shortens the time needed for traditionally manual tasks but also improves the overall consistency and efficiency of complex mission analysis operations.

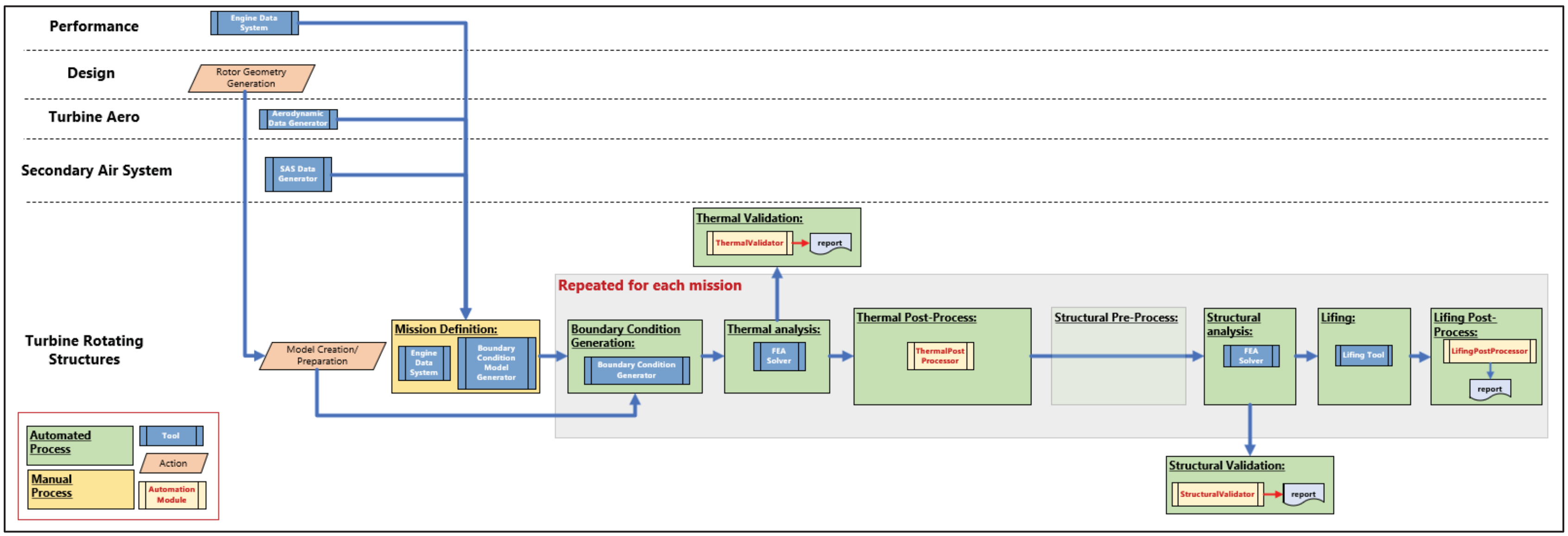

Figure 3.5 Future automated workflow of the mission analysis

3.3 Automation Architecture and Implementation

In this section, an architecture for the automation will be presented, and its integration will be discussed. First, the requirements for tool development will be addressed in terms of high-level, user experience, integration, and functional needs for the development of AMAP. Then, a general architecture proposal will be presented, explaining the modular structure of the software. This is followed by a description of the automation of individual modules, highlighting the specific objectives of each. Finally, the complete workflow integration will be reviewed to provide a better understanding of AMAP and how it operates as a complete software tool.

3.3.1 Tool Development Requirements

The tool development requirements define the set of needs and expectations from the automation tool to be designed. The final artifact must satisfy these requirements, which guide the development process by defining what the tool must do, how it should work, and under which constraints it should operate. The requirements may identify general needs, user experience expectations, and the specific functionalities that tool must provide.

The requirements for AMAP are classified in four categories: high-level, user experience, integration and functional requirements. These are presented in Table 3.1, which provides an overview of the critical requirements for the development of AMAP.

Table 3.1 Tool development requirements

Category	Number	Requirement
High-Level Requirements	1.	The tool shall be able to run on both Windows and Linux operating systems to support usage on local computer and HPC environments.
	2.	The tool shall manage the file naming of all outputs according to mission names.
	3.	The tool shall organize files and folders in line with in-house practices.
	4.	The tool shall iterate over any number of missions provided under the same input set.
	5.	The tool shall allow user interaction for decision-making steps.
	6.	The tool shall follow the established engineering best practices.
	7.	The tool shall adopt a modular design approach and provide standalone automation modules.
User Experience Requirements	1.	The automated workflow shall follow the manual process currently used by engineers to maintain familiarity.
	2.	The tool shall offer a clear and user-friendly command-line interface with descriptive input parameters and meaningful error messages.
	3.	The tool shall be usable by engineers with little or no programming experience.

Category	Number	Requirement
Integration Requirements	1.	The tool shall integrate Boundary Condition Generator tool executions into the overall framework.
	2.	The tool shall integrate the FEA solver executions into the overall framework.
	3.	The tool shall integrate in-house Lifing Tool executions into the overall framework.
	4.	The tool shall combine standalone automation modules into a single integrated workflow.
Functional Requirements	1.	The tool shall check input file existence and validate all input parameters before execution.
	2.	The tool shall verify the existence of output files, detect analysis errors and report them.
	3.	The tool shall generate detailed logs to track process activities.
	4.	On Windows, the tool shall run the integrated tools in batch mode.
	5.	On Linux, the tool shall submit the analysis executions to the HPC environment.
	6.	The tool shall apply a time-filtering algorithm to the thermal analysis results.
	7.	The tool shall provide user an option for the user to override or skip the time-filtering step.

Category	Number	Requirement
	8.	The tool shall generate Excel reports to summarize and compare thermal simulation results across missions.
	9.	The tool shall generate tables to summarize and compare lifing simulation results across missions and components.

3.3.2 General Architecture

As highlighted previously, one of the primal specifications of AMAP is its adoption of a modular approach for automation. A simplified workflow of the analysis procedure is presented in Figure 3.6. It is worth noting that although the automation of validation phases is excluded from the tool, AMAP still provides a framework which enables their implementation. In accordance with the modular structure and Python's OOP capabilities, the iterative process is divided into four models, each corresponding to an automation module: BCG, Thermal, Structural and Lifing Model as depicted in grey boxes in Figure 3.6. AMAP will provide the user with the ability to use the tool in standalone mode for individual module or in integrated mode to perform the complete analysis run.

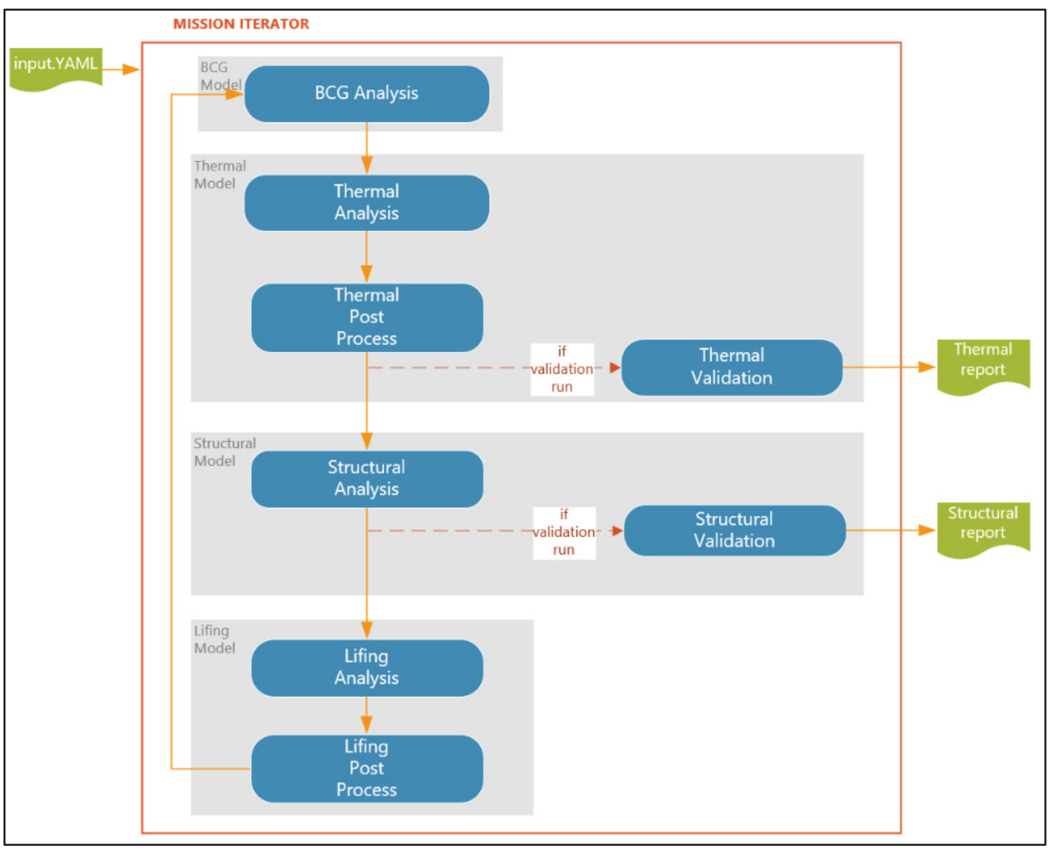

Figure 3.6 AMAP simplified workflow with model definitions

Each model includes subphases based on its operational requirements, which are indicated by the blue boxes in Figure 3.6. The integration of different models is managed by a higher-level script, referred to as the “Mission Iterator” in the Figure 3.6. AMAP will follow the dataflow presented in Figure 3.4. File names and analysis instructions will be transferred to models via text files. The outputs from each process will serve as inputs for the subsequent steps. This will ensure consistency and traceability and enable the full integration across the workflow.

3.3.3 Automation of Individual Modules

After gathering the end-user requirements, as outlined in Table 3.1 and general architecture of AMAP is established in the previous section, the implementation of the automation modules is initiated.

Automation of BCG is the first phase of the AMAP project. The primary objective of this module is to gather inputs, run the BCG tool in batch mode, and generate files which contain thermal boundary conditions for the downstream process. It is worth noting that these steps must be completed for each mission provided by the user.

The Thermal Model is the second phase of the AMAP project. As a standalone module, it assumes that the boundary condition files are provided either by the upstream process or directly by the user. Even though these files are generated in BCG Model, there are additional input files that must be provided by the user.

The thermal module consists of two parts: the automation of the thermal analysis and the automation of the thermal post-processing. The principal objective of the former is to obtain the temperature distribution across the turbine rotor geometry. The latter aims to filter time steps for structural analysis, identify critical time steps for the next phase, and generate informative reports to interpret the thermal results.

Since structural analysis is computationally more expensive and requires more resources, it is preferable to run it for fewer time steps. In the conventional process, nodal temperature distributions were examined entirely manually to filter the time steps during thermal post-processing. As a contribution of this thesis, a fully automated algorithm aligned with manual method is proposed and implemented into the AMAP code.

Development of the Structural Model for the automation of structural analysis constitutes the third phase of the AMAP project. The main purpose of the structural module to run FEA solver in batch mode to obtain the stress and strain distributions over the turbine rotor geometry and to generate screenshot images verifying the applied mechanical and thermal loading conditions.

The final phase of AMAP is the automation of the lifing analysis. The primary objective of the Lifing Model is to extract the stress variations of specified components from structural analysis results, execute the Lifing Tool for them, and post-process the results. After completion, this module provides tables summarizing the life values of the selected components for each mission.

3.3.4 Complete Workflow Integration

As highlighted previously, even though the model-by-model approach was adopted during the development of the tool, an integration between the models also must be implemented. This will allow users to run the tool in the integrated mode, in addition to the standalone modes described in previous section.

The integrated mode is capable of execution of the entire mission analysis process, as previously depicted in Figure 3.6. An upper-layer script has been implemented within the tool to provide this capability. Each module of AMAP can be examined in three different stages:

pre-processing, analysis run and post-processing. This structure and the overall tool integration is illustrated in Figure 3.7. When running AMAP, the user must specify the module name along with an optional integrated flag. The code checks the module name and starts execution from the selected point. As one module is completed, the tool checks if integrated mode is enabled and continues with the next module if applicable.

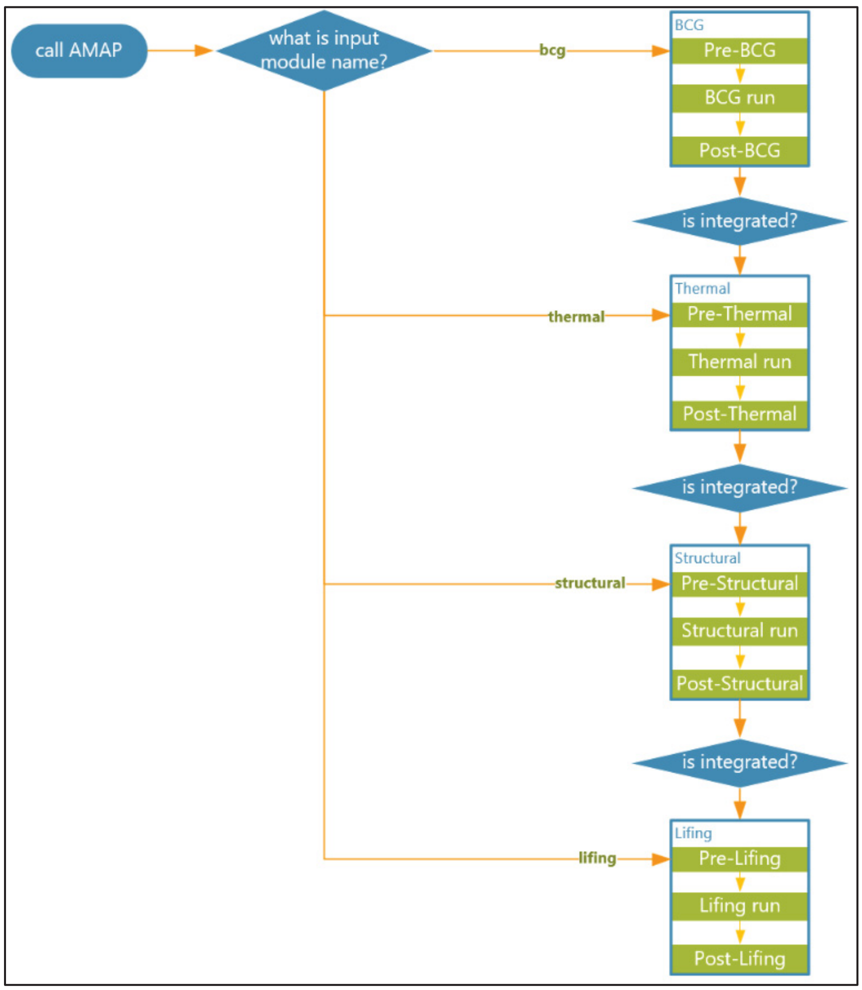

Figure 3.7 Module integration of AMAP

Another important feature of AMAP is that it is fully operational both on Windows and Linux platforms. This flexibility addresses the varying computational needs of users. When the test case involves a large number of missions, running on Windows becomes time-consuming. In such cases, it is more efficient to submit the analyses to a high-performance computing (HPC) cluster. On the other hand, if the user has less missions with smaller flight cycles, running the tool locally on a laptop is a practical option.

If the user runs AMAP locally on a Windows operating system, the workflow will follow the process presented in Figure 3.8. As AMAP is initialized by the user, it starts from BCG phase and continues to the lifing analysis. Iterations are completed within each module for all missions, meaning that the next module starts only after the jobs for all missions in the current module are finished. Red arrows represent those iterations in Figure 3.8.

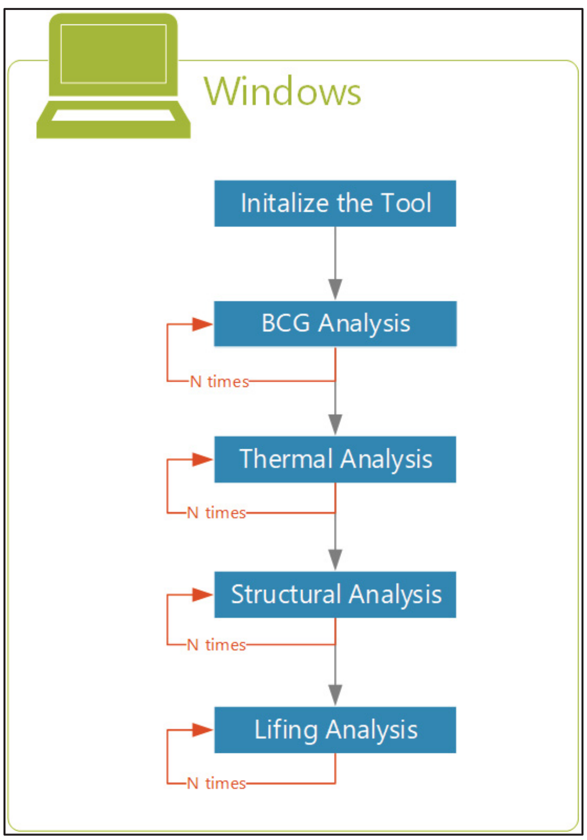

Figure 3.8 AMAP workflow in Windows

In Linux, the workflow remains the same as the Windows approach except that the tool submits the analysis jobs to the HPC to run the analyses on the computing nodes, rather than running on the local machine. The workflow in HPC is illustrated in Figure 3.9. The key difference is that the analysis jobs which submitted to HPC are only dependent on the previous step within the same mission. This means that the missions run independently from each other once they are submitted to HPC. Red arrows in Figure 3.9 represent the job dependencies.

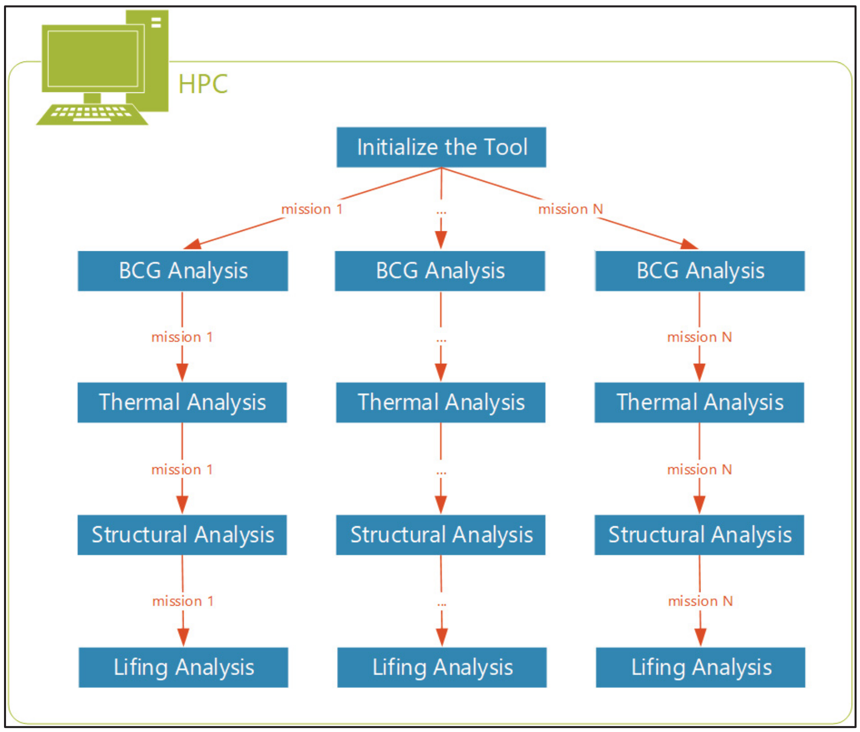

Figure 3.9 AMAP workflow in HPC

With the integration of different modules and the implementation of HPC support, the development of AMAP is completed, and the tool is ready for testing.

3.4 Chapter Conclusion

This chapter covered the detailed analysis and design of the automation tool AMAP. Firstly, manual mission analysis process was examined in detail, and an automation solution was proposed. Then, the tool development was discussed in terms of its requirements, architecture and implementation both for standalone modules and their integration. This provided an overview of the AMAP tool, which is now ready for testing and validation.

CHAPTER 4

TEST CASES AND RESULTS

The automation tool AMAP, as introduced in the previous section, is ready to be tested and validated. To assess its performance and reliability, two distinct turbine rotor geometries have been selected as case studies: low pressure and power turbine blades. In this chapter, these two test cases will be introduced, and the results obtained from AMAP executions will be discussed. Subsequently, the reduction in time spent through automation will be presented. Finally, user experience will be evaluated based on the collected feedback.

4.1 AMAP Execution for Test Case 1: LPT Geometry

The low pressure turbine (LPT) blade of a turboprop engine is selected for the first test case. The rotor has a one-blade configuration. This test case includes only one mission to be analyzed. To evaluate the iteration capability of AMAP, the same mission is duplicated, simulating a scenario in which two missions are processed by AMAP. When AMAP is executed for this test case, the iteration is successfully completed, and the results for the two identical missions are obtained. To avoid repetition, the results for only one of the missions will be presented.

The first phase of AMAP, the automated BCG analysis, generates a set of files previously referred to as boundary condition files. Once AMAP is run and these files are generated through the automated workflow, they are compared with the baseline files. Although the content cannot be presented here due to proprietary information, the results were identical to the baselines and have been verified.

In the second phase of AMAP, the automated thermal analysis generated the thermal simulation result file, a filtered list of time steps, and Excel summary reports. The main thermal result file is manually validated by engineers, and the temperature results are found to be identical to those generated manually. In this section, some of the analysis results will be discussed.

As a first step, AMAP provides the temperature history of selected nodes to the user. The generated graph is shown in Figure 4.1. For this test case, eight different nodes were selected. The temperature variation over time for these nodes was extracted from the thermal result file and is presented in the following graph. The results indicate that Node #1 exhibits significantly higher temperature values compared to the others, indicating a location exposed to higher thermal loads. On the other hand, nodes #3 through #8 show similar thermal trends, with Node #2 reaching the highest temperature among them. The temperature profiles remain mostly stable after initial transients, with distinct changes observed at specific time intervals, corresponding to operational condition variations depending on the mission definition.

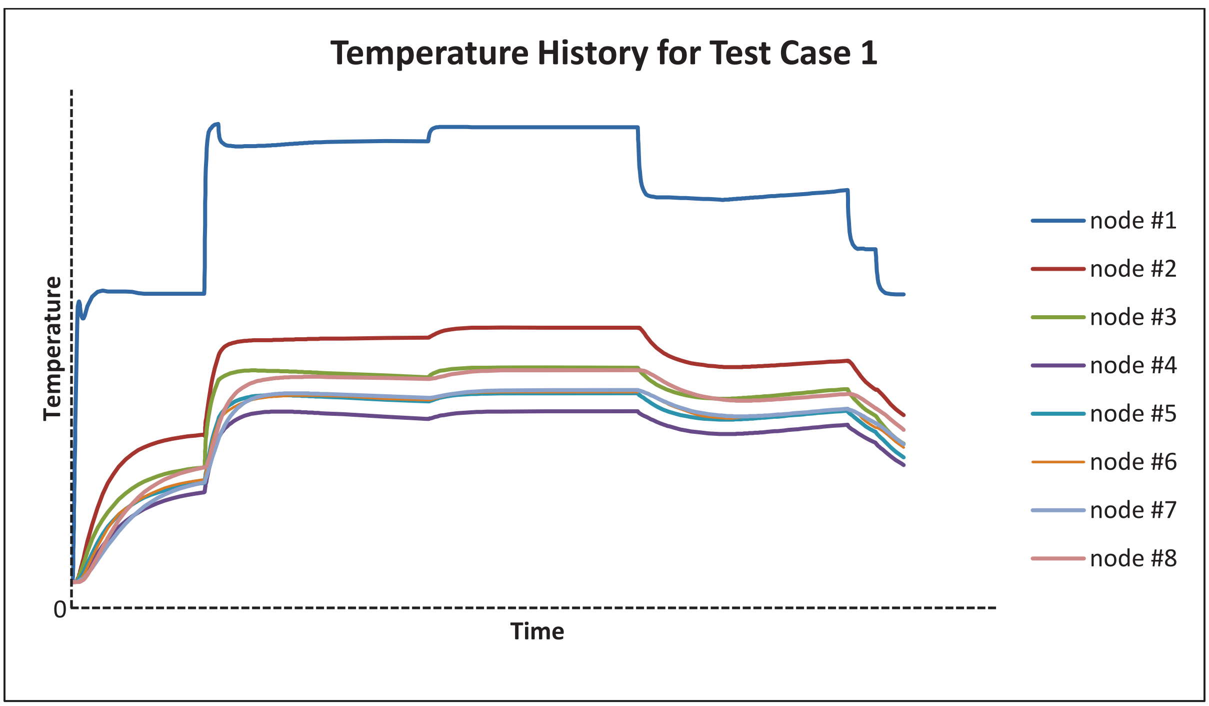

Figure 4.1 Temperature history of selected nodes graph for Test Case 1

In addition to individual temperature profiles, the temperature differences between selected node pairs are shown in Figure 4.2. These results help checking the thermal gradients present in the model. Among the compared pairs, Delta #1 exhibits the highest difference throughout most of the time, indicating a stronger gradient between those two nodes. Delta #2 and Delta #3 remains relatively low and stable, identifying more uniform thermal behavior.

In the transient thermal analysis conducted for this test case, a total of 353 time steps were computed and simulated. During the post-processing phase, these were reduced to the 136 most critical time steps to accelerate the process of the upcoming structural analysis. The selected time steps from the mission profile are illustrated in Figure 4.3. Red crosses indicate the extracted time steps, while the blue curve represents the complete mission profile consisting of all 353 time steps. The filtering methodology and results were reviewed and validated by the engineering team.

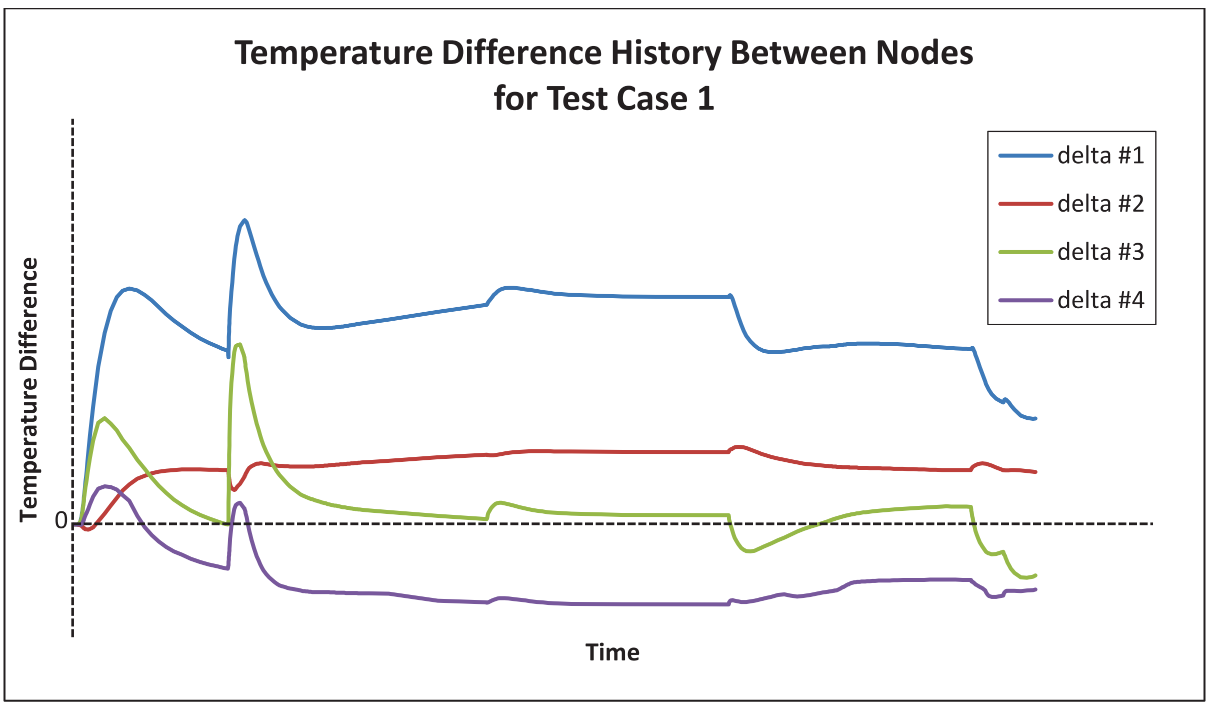

Figure 4.2 Temperature difference history of selected node pairs graph for Test Case 1

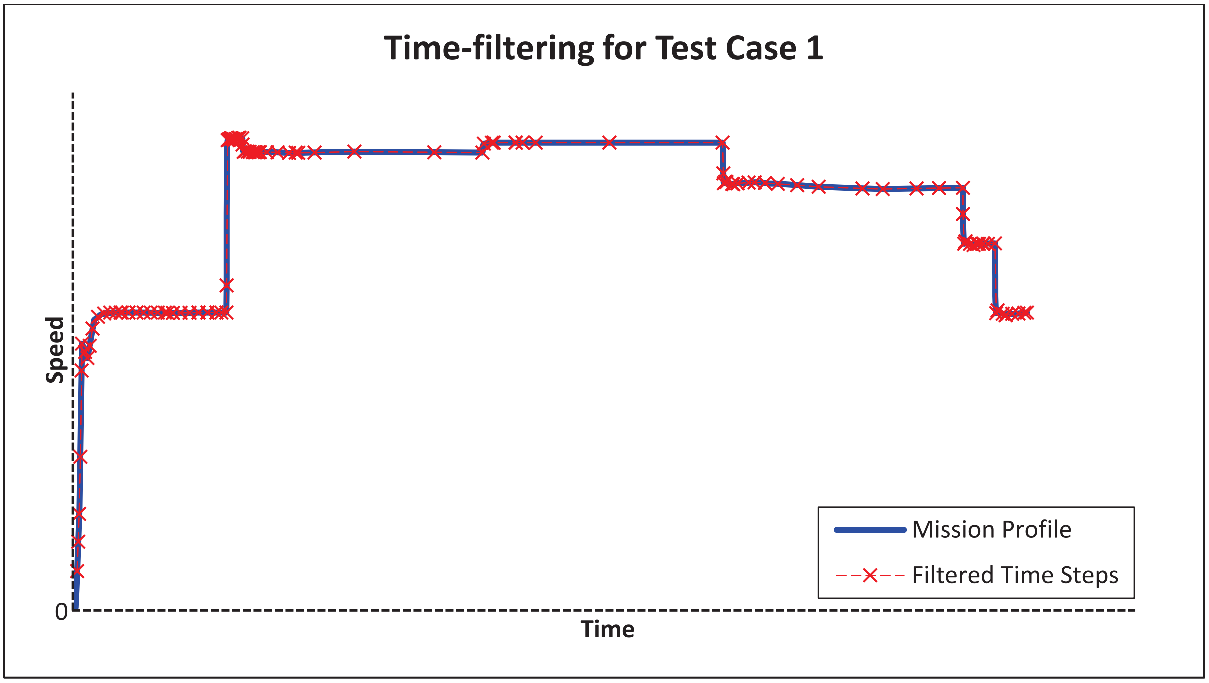

Figure 4.3 Mission profile and time-filtering comparison graph for Test Case 1

In the third phase of AMAP, the automated structural analysis is executed, generating both the structural simulation result file and a set of images used to validate the applied loading conditions. These images are compared with predefined baseline files and reviewed visually, confirming that AMAP has correctly applied the intended boundary conditions. In addition, the main stress result file is manually validated by the engineers. Six of the nodes previously selected during the thermal analysis phase are further examined in terms of their structural response. The stress values at these nodes are plotted and presented in Figure 4.4.

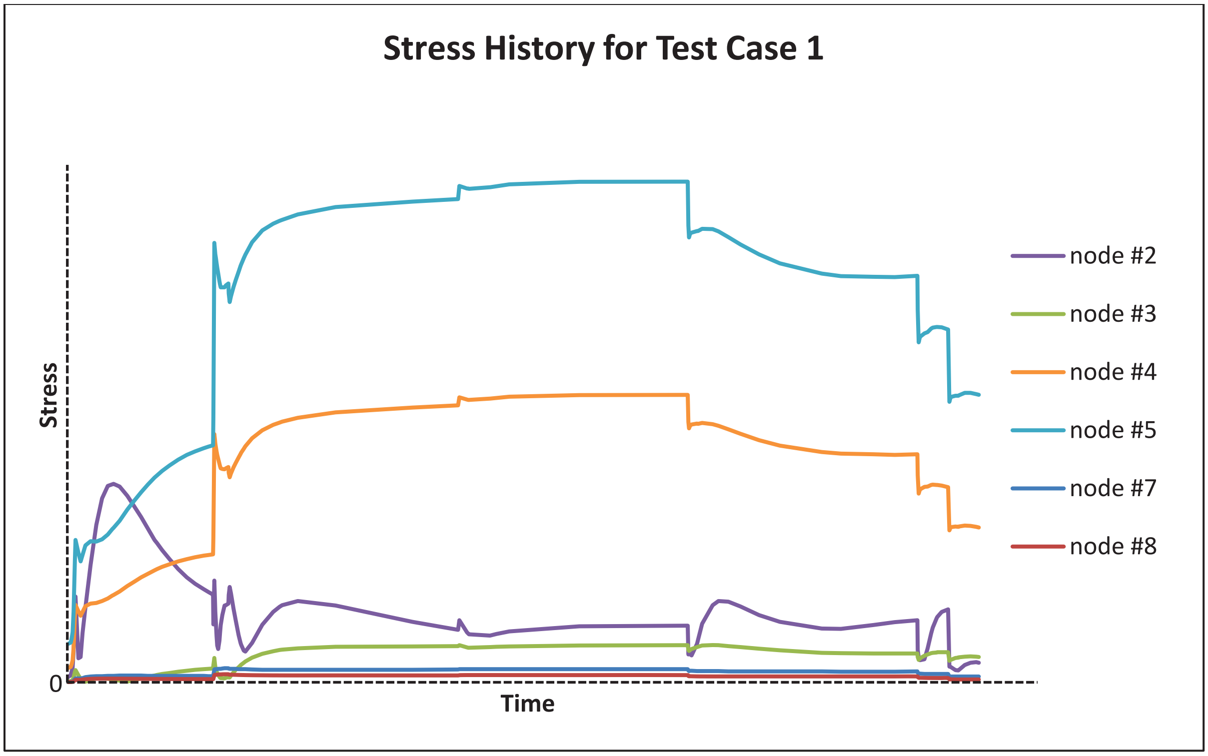

Figure 4.4 Stress history of selected nodes graph for Test Case 1

As shown in Figure 4.4, Node #5 has the highest stress values throughout the mission, indicating a critical region subjected to the highest mechanical loads. Node #4 also shows significantly high stress levels, showing a similar trend. In contrast, Nodes #3, #7 and #8 remain at very low stress levels during the entire mission profile. Node #2 is exposed to stress fluctuations at the specific parts of the simulation, which may would be associated with transient effects in the mission definition. Overall, the results for all nodes show a trend matching the mission profile presented in Figure 4.3. Finally, the stress results indicate that Node #5 and #4 may be critical in terms of lifing, since they endure the highest stress values.

In the final phase of AMAP, the lifing analysis is run automatically, and summary tables are generated through the automated workflow. Although these results cannot be presented here due to being proprietary information, they have been validated by engineers, and the automatically generated tables were found to be sufficient.

Once all analysis results are examined and validated, time consumption for the Test Case 1 can be evaluated. The execution times for these tasks are measured and examined. The results indicate that AMAP automated all the possible tasks and minimized the time consumption for Test Case 1. To better evaluate the tool's performance and quantify the time-saving benefits, a more detailed analysis will be provided in Section 4.3.

4.2 AMAP Execution for Test Case 2: PT Geometry

The power turbine (PT) blade of a turboprop engine is selected for the second tests case. The rotor has a two-blade configuration. Two distinct missions are selected for the second test case. This allows testing both the iteration functionality of AMAP and its mission comparison capabilities. Mission 1 consists of a set of steady state points, forming a shorter mission profile, while Mission 2 has a transient mission definition. Automated workflow is executed for the two missions of the second test case, and the results are presented and validated in this section.

The BCG results of both missions obtained from the AMAP execution are manually compared with the files which are generated by the manual workflow, like the first test case. The boundary condition files produced by the automated workflow are found to be same with the baseline files. This confirms that the BCG automation is fully functional and reliable.

In the context of the transient thermal analysis, the temperature results file was validated by the engineering team for both missions, similar to that was done for Test Case 1. Subsequently, the temperature histories at 11 selected nodes were extracted from the thermal analysis results. These temperature distributions are presented in Figure 4.5 for Mission 1 and in Figure 4.6 for Mission 2. In Figure 4.5, the curves show a rapid temperature increase at the beginning, followed by a stabilization. While most nodes follow a similar thermal trend, minor differences can be observed in their peak values. It is worth noting that Node #4 reaches the highest temperature, suggesting a region subjected to the most severe thermal conditions during the mission. Figure 4.6 also introduces a quick increase in temperature for all nodes. Node #4 and Node #5 are exposed to most severe conditions, and they have higher peaks where Node #9 has considerably lower temperature values.

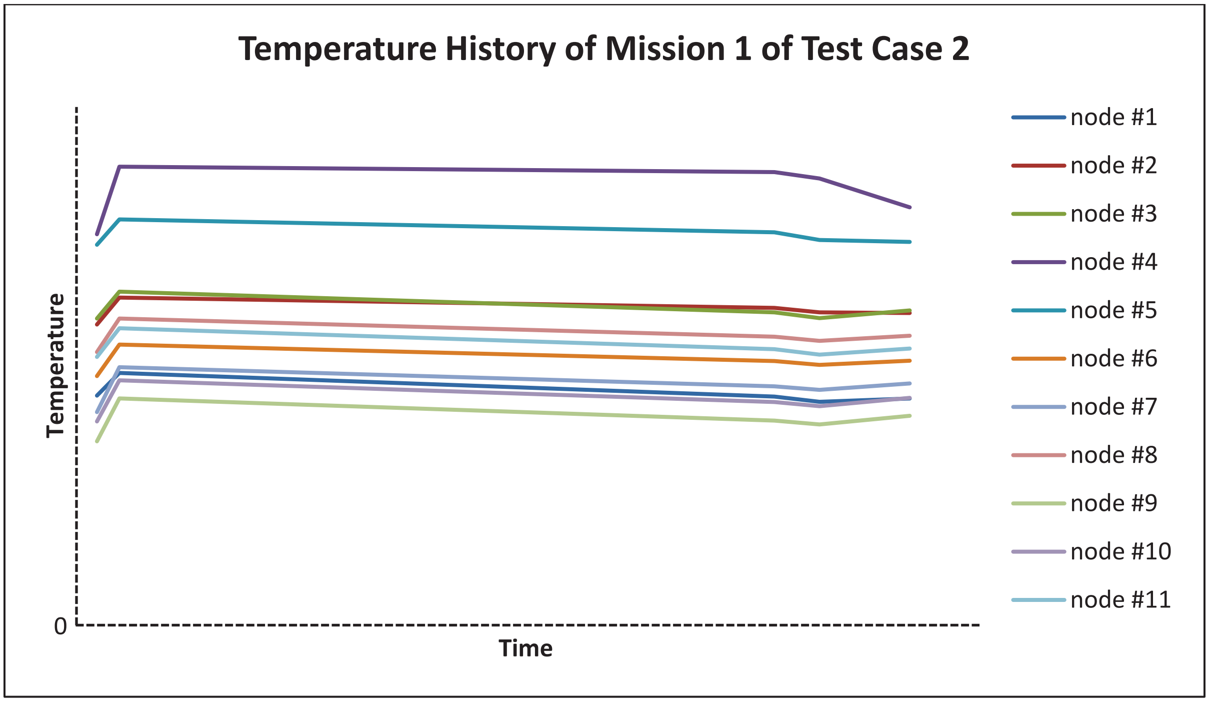

Figure 4.5 Temperature history of selected nodes graph for Mission 1 of Test Case 2

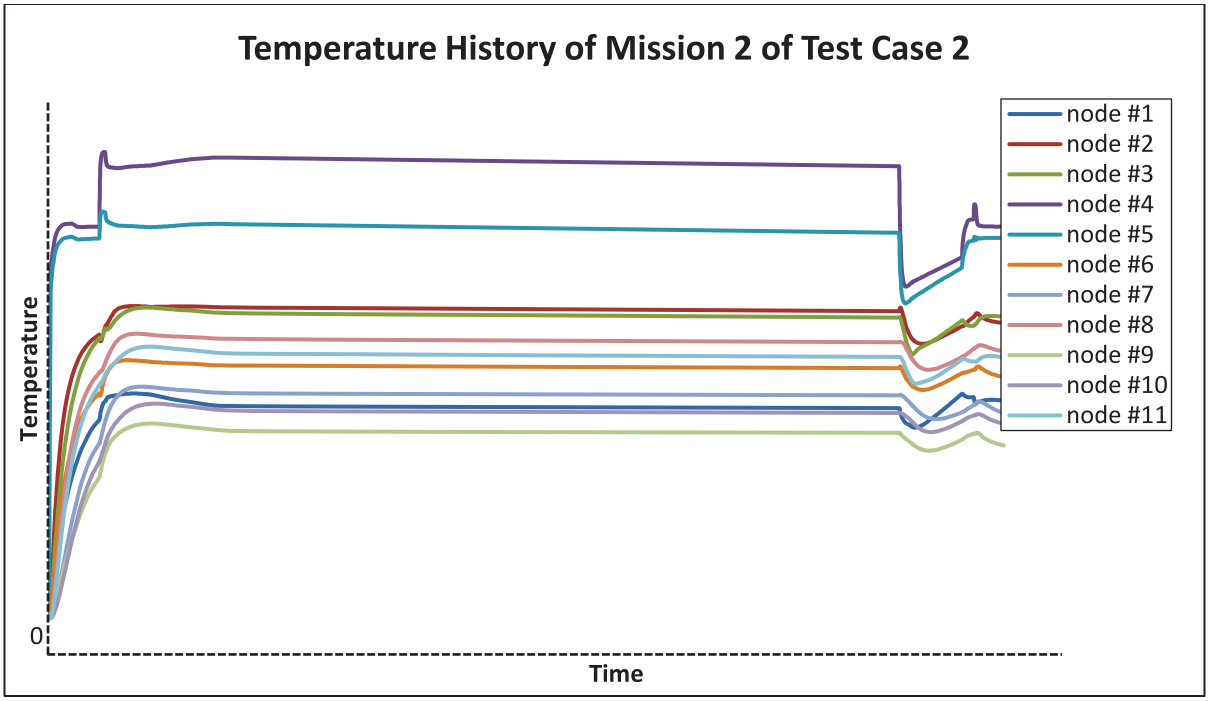

Figure 4.6 Temperature history of selected nodes graph for Mission 2 of Test Case 2

To examine the temperature gradients on the blade geometry, 6 node pairs were selected for both missions. AMAP automatically extracted the temperature histories from the thermal analysis results, computed the temperature differences for the specified node pairs, and generated the corresponding graphs. These graphs are presented in Figure 4.7 for Mission 1 and Figure 4.8 for Mission 2. In Mission 1 (See Figure 4.7), the temperature differences between node pairs remain nearly constant throughout the transient analysis, indicating a steady-state thermal behavior and minimal fluctuations in the mission definition. On the other hand, results from Mission 2 (See Figure 4.8) seems to be more dynamic, particularly during the initial phase. Some node pairs, such as Delta #1 and Delta #6, show sharp changes in temperature difference, indicating high transient thermal loading. Then all temperature gradients stabilize, suggesting a steady-state condition in the mission profile. Overall, Mission 2 demonstrates higher temperature gradients, which may contribute to higher thermal stresses on the blade.

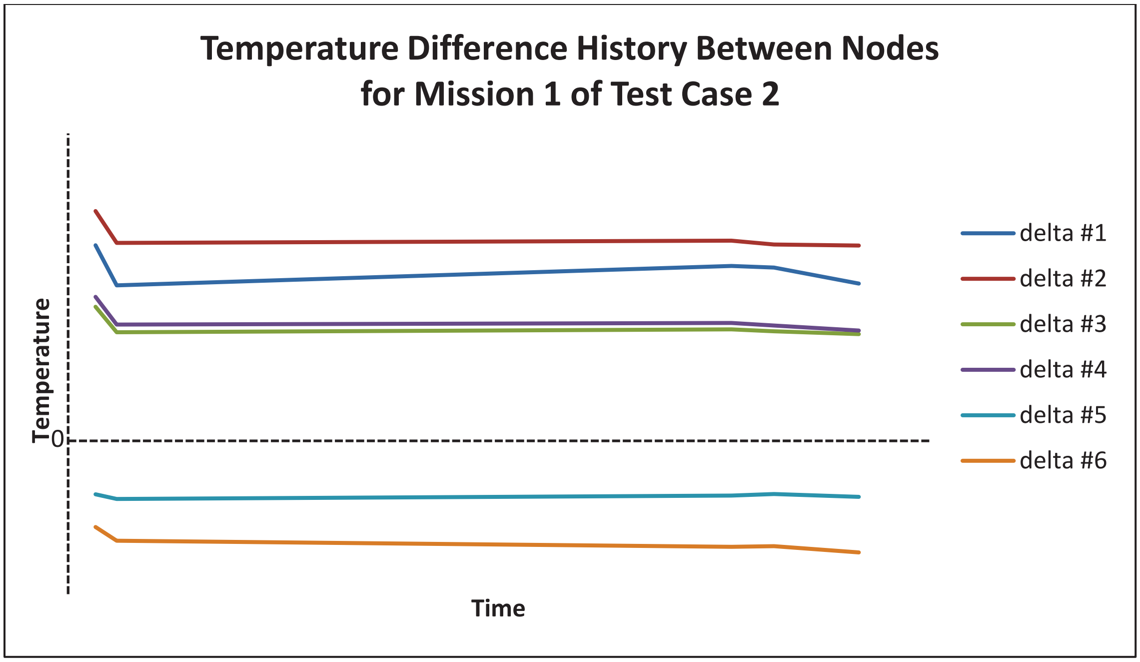

Figure 4.7 Temperature difference history of selected node pairs graph for Mission 1 of Test Case 2

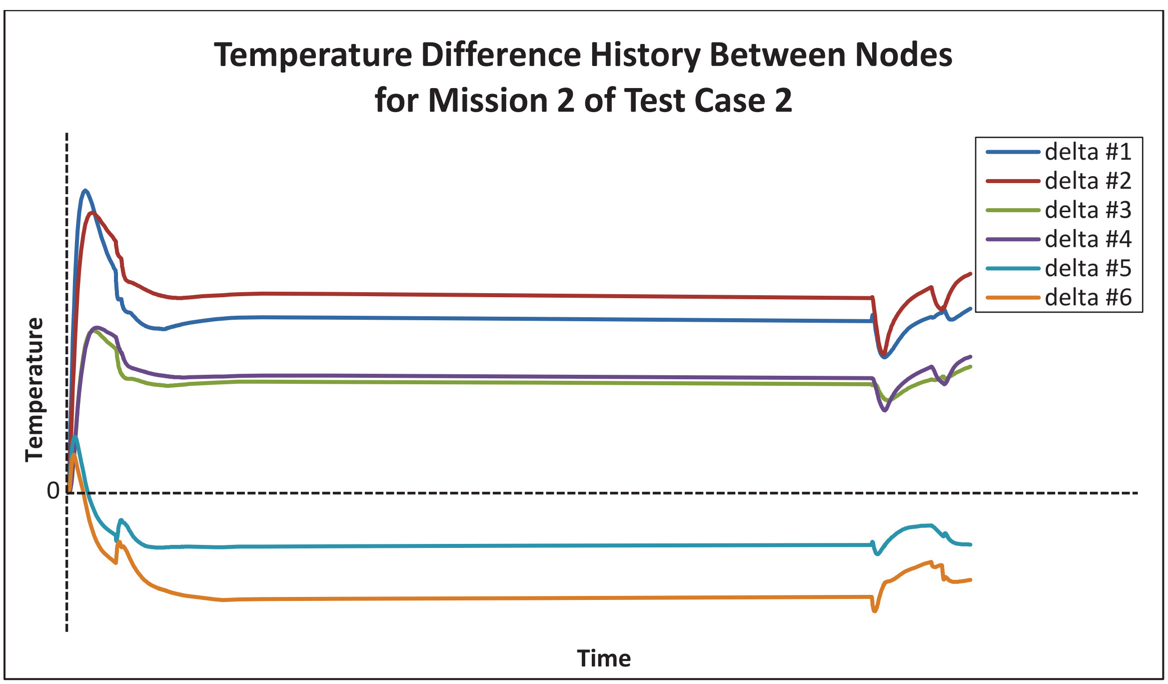

Figure 4.8 Temperature difference history of selected node pairs graph for Mission 2 of Test Case 2

When AMAP conducted time-filtering based on the thermal analysis results, different numbers of time steps were selected for each mission. The mission profile and the corresponding filtered time steps are shown in Figure 4.9 for Mission 1 and Figure 4.10 for Mission 2. Mission 1 consists of only 5 time steps, which is a relatively low number for structural analysis. In accordance with it, all 5 time steps were selected by the AMAP algorithm and confirmed by the responsible engineers as all of them required to capture the critical loading conditions. In contrast, Mission 2 includes 437 time steps in the thermal analysis, which is too computationally expensive to be used entirely for structural analysis. AMAP filtered these steps down to 111 representative time steps, providing a good trade-off between computational cost and result accuracy. The selected time steps, shown in Figure 4.10, are distributed mainly around transient regions where significant changes occur in the mission profile. This approach effectively ignores long steady-state time intervals, which would offer minimal variation in stress results. This proves that the filtering strategy successfully focuses on significant time intervals and minimizes computational cost for the next steps of the mission analysis.

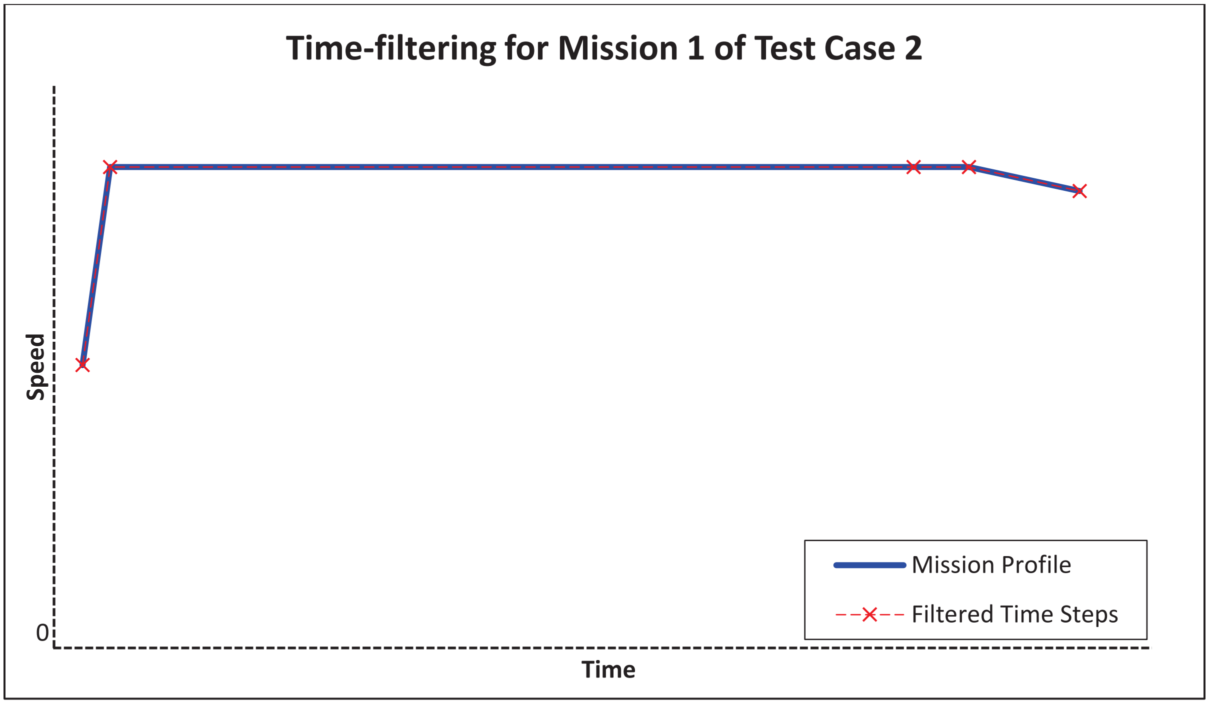

Figure 4.9 Mission profile and time-filtering comparison graph for Mission 1 of Test Case 2

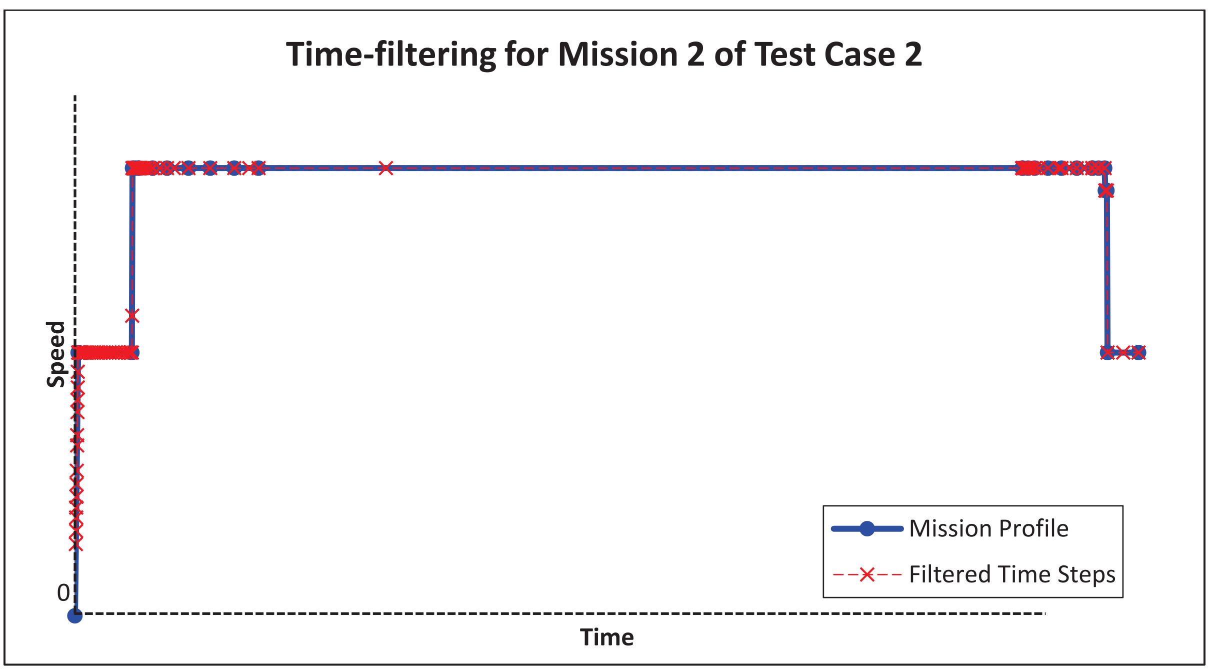

Figure 4.10 Mission profile and time-filtering comparison graph for Mission 2 of Test Case 2

To evaluate the mission comparison functionality, the reports generated by AMAP was examined. As an explanatory example, the mission profile comparison graph is presented in Figure 4.11. Mission 1 consists of only 5 time steps, while Mission 2 includes 437 time steps. As clearly seen in Figure 4.11, Mission 1 is significantly shorter and faster compared to Mission 2. All previously obtained results were consistent with this observation, consequently confirming that AMAP successfully generates the Excel file required for mission comparison.

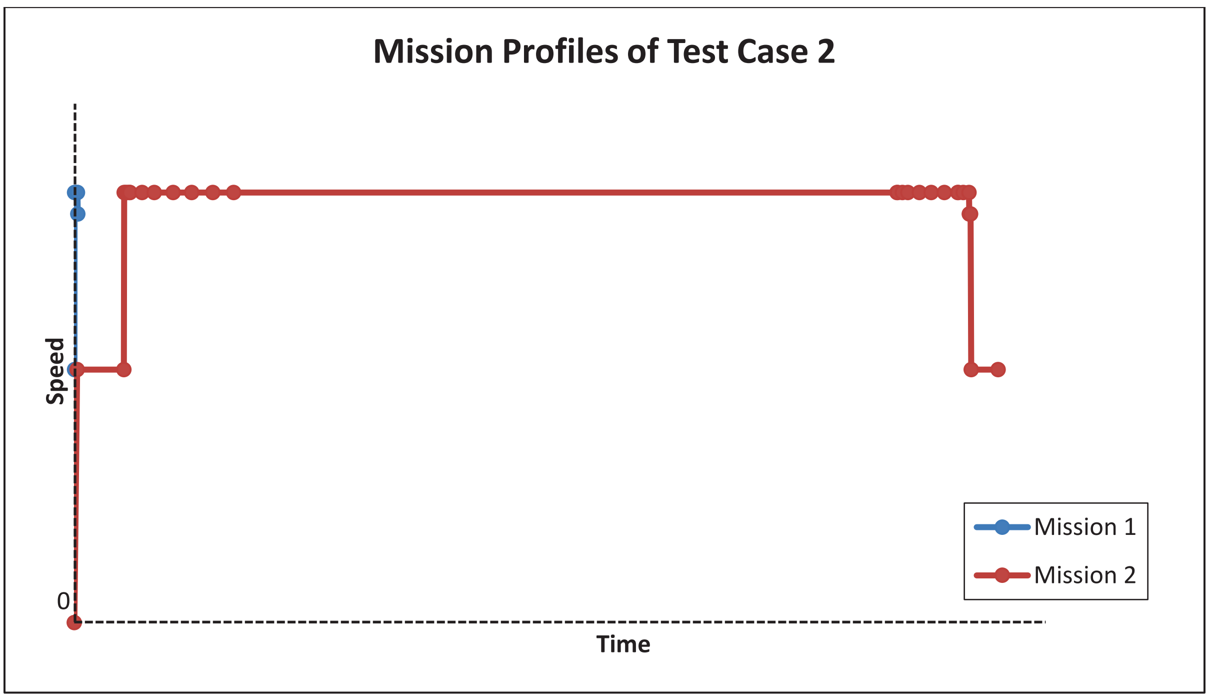

Figure 4.11 Mission profile comparison for Mission 1 and Mission 2 of Test Case 2

In the context of the transient structural analysis, the stress results file was validated by the engineering team for both missions, and the images of loading conditions are confirmed visually, similar to that was done for Test Case 1. To examine the stress response of the turbine blade, 7 of the 11 selected nodes during the thermal post-processing are picked for further analysis. The stress histories of these 7 nodes are extracted from the automatically created result files and are depicted in Figure 4.12 for Mission 1 and in Figure 4.13 for Mission 2. For both missions, Nodes #2 and #3 are exposed to a rapid increase in stress, before stabilizing in later phases of the mission. To review the overall stress values throughout the mission profile, Node #1 endured the highest stress values, while Nodes #9, #10 and #11 experienced the lowest. These findings suggest that Node #1 may be critical for lifing, as it was subjected to the high stresses over long periods, whereas Node #2 and #3 may be critical for in terms of life considerations due to their rapid exposure to peak stresses.

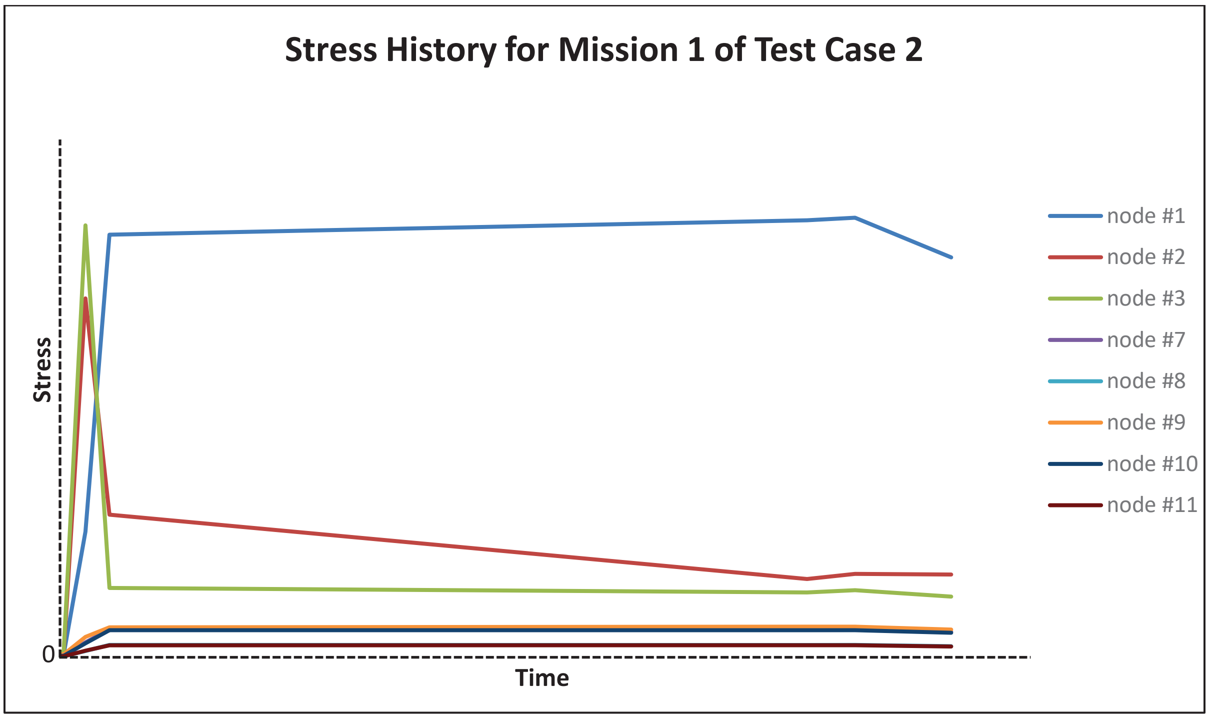

Figure 4.12 Stress history of selected nodes graph for Mission 1 of Test Case 2

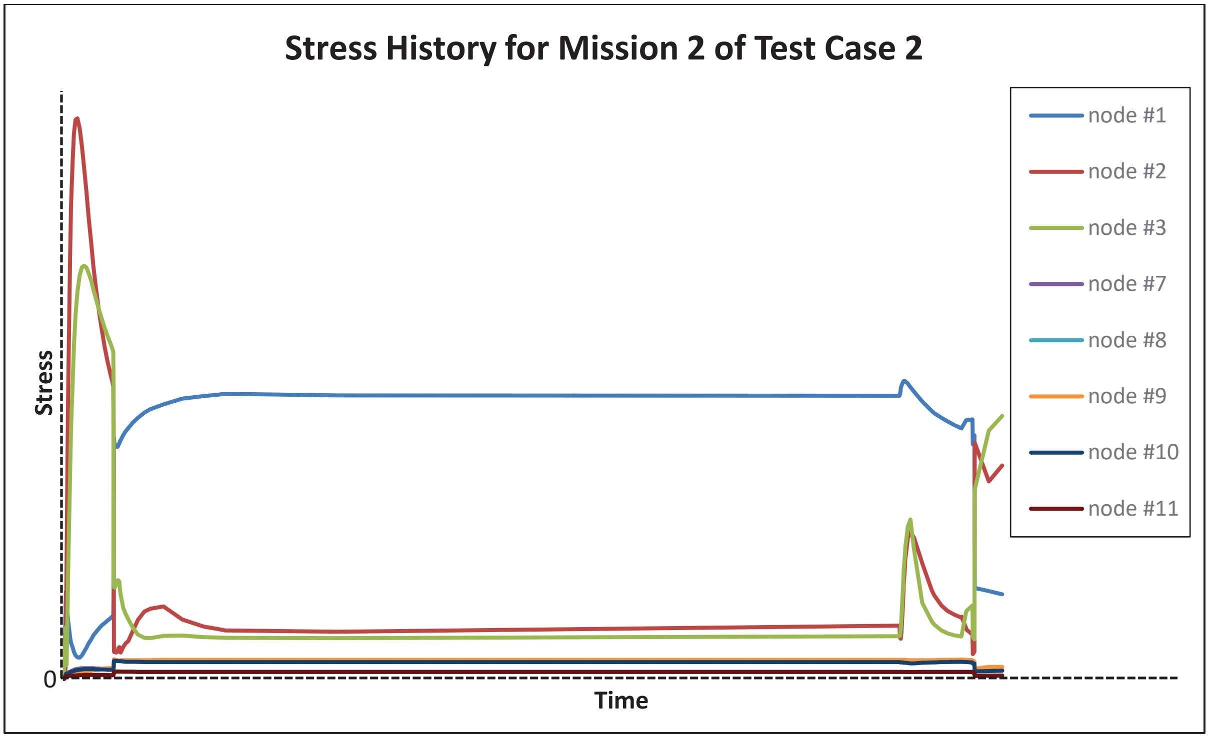

Figure 4.13 Stress history of selected nodes graph for Mission 2 of Test Case 2

Finally, AMAP executed the lifing analysis through the automated workflow for both missions of Test Case 2. Similar to the previous test case, the results and the generated tables were verified and found sufficient by the engineers. However, the results cannot be presented due to proprietary information.

Time-consumptions of AMAP jobs, as well as the transient thermal, structural and lifing analyses was measured for both missions of Test Case 2, to evaluate the efficiency of AMAP, like that was done for Test Case 1. These results indicate that AMAP jobs are executed quickly for both missions, effectively automating the overall process. This is consistent with the observations from Test Case 1, which suggests that the execution time of AMAP does not depend on the mission definition or the test case geometry. Additionally, this further demonstrates that the automation was implemented with maximum efficiency.

4.3 Productivity Analysis on Time Reduction

The impact of AMAP on time efficiency through automation is evaluated and presented in this section.

- The BCG automation process results in a time saving of 95% for each mission. This improvement is primarily provided by the elimination of manual operations previously conducted via the GUI, which were both time-consuming and prone to human errors. In the manual workflow, users had to manually select input files and verify their correctness for each run.
- The thermal automation handles both the execution of the thermal analysis and the associated post-processing tasks. This automation reduced the total processing time by approximately 50% for each mission and significantly minimized the risk of manual errors, thus prevented unnecessary repetitions. Time savings were achieved in both analysis submissions and thermal post-processing, as its inefficiency was discussed in Section 3.1.1.2. By eliminating the need of tedious manual tasks with traditional GUIs and introducing a standardized time-filtering approach, the automation provided an effective and reliable engineering solution.
- The structural automation avoided the need for a separate structural pre-processing phase. Additionally, the submission of the FEA, which was previously performed manually, has also been automated. Overall, this resulted in average time saving of around 5% for each mission.
- The lifing automation manages both the execution of the lifing analysis and the post-processing of the obtained results. Previously, these tasks required manual work and GUI operations. Automation has eliminated these steps, saving approximately 60% of the time required for each mission.

The results indicate that AMAP reduces the total time consumption by approximately 30% for a typical test case.

Considering these results and those obtained from AMAP executions for the two test cases, the percentage time gains can be calculated for the two test cases. The time consumptions were measured for both test cases were obtained from the automated workflow. Using these values along with the time saved through automation, the percentage time gain compared to the manual workflow are calculated and presented in Table 4.1. The results highlight that the time reduction for the mission analysis of Test Case 1 is 45%, whereas it 65% for Mission 1 and 20% for Mission 2 of Test Case 2. These calculations also confirm that the previously estimated a time gain of 30% is sufficiently accurate for an average test case.

Table 4.1 Percentage time gain

Analysis Execution	**Percentage Time Gain**
Test Case 1	45%
Mission 1 of Test Case 2	65%
Mission 2 of Test Case 2	20%

It is worth noting that, thanks to AMAP's capability to run on HPC, each mission analysis is conducted in parallel. This makes the total required time for one test case equivalent to the time taken by the longest mission analysis. Moreover, the time savings increase linearly with the number of missions, since manual work had to be repeated for each mission in the manual workflow. This further enhances the practical time gain while allowing the automated workflow to complete in a minimal amount of time.

For all analysis phases, manual processes were not only increasing time consumption but also more prone for human error. These errors could cause repetitions of analyses, further extending project timelines. The AMAP framework effectively eliminated these manual errors and prevents unnecessary repetitions, thus it ensured data integrity and consistency throughout the workflow. This did not only save significant time but also improved overall reliability. Furthermore, AMAP's automated data management efficiently handled folder structuring and file organization, contributing additional reductions in total time consumption. As a result, the actual time saved by automation is even greater in practice than the conducted theoretical calculations. Moreover, since the analysis duration is decreased, it becomes possible to evaluate a larger number of missions within the same period of time, further improving the productivity and decision-making in turbine rotor design.

4.4 User Experience Evaluation

User experience was at the core of development of AMAP. The requirements were driven by the goal of improving usability. To assess user satisfaction, this section investigates user feedback and their experiences with AMAP.

The agile development philosophy was adopted throughout the development phase. The process was divided in short development cycles, and continuous delivery was provided to users. As outlined in Section 2.4, a functional version of AMAP was delivered to users for testing and feedback at the end of each cycle. This approach embraced changes in the code and enabled continuous improvement of the tool.

Engineering best practices were adopted to enhance the reliability of AMAP. For instance, data management is handled in accordance with the internal folder structuring methodology, which eases understanding of analysis outputs. Additionally, AMAP automatically updates file names based on mission-specific naming conventions, saving time and improving clarity for users. Finally, the reporting of results follows templates and procedures defined by the engineers, supporting familiarity with results and providing ease of use.

AMAP also allows engineer intervention when necessary. For example, if the user prefers to bypass the time-filtering step during thermal post-processing, a custom times file can be manually provided to the tool, skipping the times file generation by the AMAP algorithm.

Finally, mission analysis process was previously highly user-dependent, with manual procedures varying significantly among engineers. AMAP not only accelerated the mission analysis process but also streamlined the procedure among different users.

4.5 Chapter Conclusion

This chapter presented the testing and results of the developed software, AMAP. One LPT and one PT geometry were analyzed, and the results were validated for their accuracy. Then, the time reduction was calculated for general test cases to discuss productivity analysis, and time gain results for two specific cases were presented. Finally, user experience was evaluated based on the collected feedback.

CONCLUSION

This thesis has presented the development of a novel automation tool for the transient mission analysis of turbine rotors, with a particular emphasis on low-cycle fatigue (LCF) life prediction. Through this study, it has been demonstrated that mechanical and thermal loads derived from transient finite element analysis (FEA) simulations can be accurately calculated based on detailed mission definitions, and that the results can be seamlessly processed in a fully automated workflow to generate efficient and reliable life estimations. By framing the problem from both an engineering and a process-optimization perspective, this research has underscored the concrete benefits of integrating automation into the traditionally manual procedures of turbine rotor analysis.

The final artifact of this work is the AMAP tool, a robust and dependable software solution designed to meet the evolving needs of the aerospace sector. This project focused primarily on process innovation: improving the quality and consistency of analyses, reducing the risk of human error, accelerating response times to internal and external stakeholders, and facilitating better data management across multiple engineering teams. In achieving these objectives, AMAP contributes long-term value to the aerospace industry by strengthening engineering capabilities, supporting more informed decision-making, and enhancing overall competitiveness in the market.

AMAP integrates a variety of in-house and commercial tools, enabling users to compare multiple mission profiles in terms of LCF life consumption. By eliminating repetitive and non-value-added manual tasks, it automates the entire analysis process and frees engineers to concentrate on higher-level design considerations. By prioritizing user experience and transparency, the tool not only improves productivity but also contributes to reducing turbine design and validation costs.

A modular development approach was adopted on purpose. The mission analysis process was divided into four categories—BCG, Thermal, Structural, and Lifing Models—each automated individually. This architecture allows users to execute modules in standalone mode when required, while also supporting a fully integrated workflow that encompasses the entire transient mission analysis process. As a result, AMAP provides a comprehensive and flexible framework adaptable to a range of engineering contexts and project sizes.

The system was validated using two test cases and three mission scenarios involving turboprop engines. Results indicated time savings ranging from 20% to 65% compared to existing manual procedures, with an average reduction of approximately 30% for a typical test case. These efficiency gains were achieved through automated data handling, the elimination of redundant manual steps, and the integration of reliable post-processing algorithms. In practice, even greater improvements can be anticipated by minimizing iterations due to human error. This not only accelerates turnaround times but also elevates the quality of decision-making by allowing engineers to focus on innovation rather than routine data manipulation.

Beyond saving time, AMAP has improved the quality, reliability, and reproducibility of the design process by standardizing the transient mission analysis procedure across different users. By creating a consistent methodology and common data environment, it enhances collaboration between engineering teams, reduces discrepancies across analyses, and increases confidence in the results produced. In this way, the work represents a significant step forward in the automation of fatigue life assessment processes for aero-engines.

RECOMMENDATIONS

Several important areas for future work can be identified:

- In addition to the lifing of the bladed turbine rotors presented in this study, the scope of the automation tool should be expanded to include other gas turbine components such as the compressor and shaft.
- The presented tool focused on LCF lifing, the methodology can be extended cover other failure mechanisms such as HCF and creep, which can be critical for determining the service life of the components.
- Although the presented tool excludes the automation of validation steps for thermal and structural simulations, it provides a framework that allows for the easy integration of future automation modules. Automating these validation steps would be a valuable contribution to the tool.
- Implementation of a GUI to the presented tool would not only improve usability but also reduce the time spent to prepare the inputs and visualizing the results.
- The current automation focuses on a specific phase of turbine blade design: the iterative mission analysis process. The scope of the automation could be expanded to include upstream processes, such as the generation of input files or preliminary CFD analyses.
- The tool has significant potential for integration within an optimization framework. Its implementation into a multidisciplinary optimization tool would allow for the evaluation of a greater number of turbine geometries in early design phases, ultimately supporting the development of more efficient and well-optimized gas turbines.
- Even though this tool was primarily developed for the Detailed Design Phase, due to the substantial time reduction, it could be integrated into the Pre-Detailed Design Phase.

LIST OF BIBLIOGRAPHICAL REFERENCES

Ahmed, M. A. K. I. (2025). *Surrogate-Based Optimization for Evaluating Turbine Cooling Impacts on Aero-Engine Performance*.

Bannantine, J. A., Comer, J. J., & Handrock, J. L. (1990). *Fundamentals of metal fatigue analysis*. Prentice Hall.

Bathias, C., & Pineau, A. (Eds.). (2013). *Fatigue of Materials and Structures: Fundamentals* (1st ed.). Wiley. https://doi.org/10.1002/9781118623435

Bhatti, S. K., Kumari, S., Neelapu, M. L., & Kedarinath, C. (2006). *TRANSIENT STATE STRESS ANALYSIS ON AN AXIAL FLOW GAS TURBINE BLADES AND DISK USING FINITE ELEMENT PROCEDURE.*

Binder, B., Gottschall, M., Blochwittz, T., & Kirsch. (2019). *PLM- and Microservice-based Integration Approach of MBSE and System Simulation for Federated Continuous Engineering Platforms.*

Boyce, M. P. (2002). *Gas turbine engineering handbook* (2. ed). Gulf Professional Publ.

Çengel, Y. A., & Boles, M. A. (2006). *Thermodynamics: An engineering approach* (5. ed). McGraw-Hill Higher Education.

Chakrabarty, A., Mannan, S., & Cagin, T. (2016). Finite Element Analysis in Process Safety Applications. In *Multiscale Modeling for Process Safety Applications* (pp. 275–288). Elsevier. https://doi.org/10.1016/B978-0-12-396975-0.00005-X

Coito, T., Viegas, J. L., Martins, M. S. E., Cunha, M. M., Figueiredo, J., Vieira, S. M., & Sousa, J. M. C. (2019). A Novel Framework for Intelligent Automation. *IFAC-PapersOnLine*, *52*(13), 1825–1830. https://doi.org/10.1016/j.ifacol.2019.11.501

Cummings, R. M., Morton, S. A., Mason, W. H., & McDaniel, D. R. (2015). *Applied computational aerodynamics: A modern engineering approach*. Cambridge University Press.

Farahmand, B., Bockrath, G., & Glassco, J. (1997). *Fatigue and fracture mechanics of high risk parts: Application of LEFM & FMDM theory*. Chapman & Hall.

Hu, X. A., Yang, X. G., Wang, J. K., Shi, D. Q., & Huang, J. (2013). A simple method to analyse the notch sensitivity of specimens in fatigue tests. *Fatigue & Fracture of Engineering Materials & Structures*, *36*(10), 1009–1016. https://doi.org/10.1111/ffe.12053

Kurowski, P. (2004). *Finite Element Analysis for Design Engineers* (1st ed). SAE International.

Lee, J., Cameron, I., & Hassall, M. (2019). Improving process safety: What roles for Digitalization and Industry 4.0? *Process Safety and Environmental Protection*, *132*, 325–339. https://doi.org/10.1016/j.psep.2019.10.021

Liu, H., Sun, J., Lei, S., & Ning, S. (2021). In-service aircraft engines turbine blades life prediction based on multi-modal operation and maintenance data. *Propulsion and Power Research*, *10*(4), 360–373. https://doi.org/10.1016/j.jppr.2021.09.001

Logan, D. L. (2017). *A First Course in the Finite Element Method* (6th ed.). https://istasazeh-co.com/wp-content/uploads/2022/02/Daryl-L-Logan-A-First-Course-in_the-Finite-Element-Method-CL-Engineering.pdf

McCune, R. W., Armstrong, C. G., & Robinson, D. J. (2000). Mixed-dimensional coupling in finite element models. *International Journal for Numerical Methods in Engineering*, *49*(6), 725–750. https://doi.org/10.1002/1097-0207(20001030)49:6<725::AID-NME967>3.0.CO;2-W

McManus, H., Haggerty, A., & Murman, E. (2005). Lean Engineering: Doing the Right Thing Right. *Northern Ireland*.

Mendel, A. F. (2011). Why Care About PLM? *Mechanical Engineering*, *133*(03), 42–43. https://doi.org/10.1115/1.2011-MAR-5

Miner, M. A. (1945). Cumulative Damage in Fatigue. *Journal of Applied Mechanics*, *12*(3), A159–A164. https://doi.org/10.1115/1.4009458

Mirhosseini, A. M., Adib Nazari, S., Maghsoud Pour, A., Etemadi Haghighi, S., & Zareh, M. (2020). Failure analysis of first stage nozzle in a heavy-duty gas turbine. *Engineering Failure Analysis*, *109*, 104303. https://doi.org/10.1016/j.engfailanal.2019.104303

Moret, M. (2018). *Automation of a Turbine Tip Clearance Preliminary Calculation Process*.

Moret, M., Delecourt, A., Moustapha, H., Abenhaim, A.-I., & Garnier, F. (2017). Automated thermal and stress preliminary analyses applied to a turbine rotor. *Aerospace Science and Technology*, *63*, 123–131. https://doi.org/10.1016/j.ast.2016.12.013

Moustapha, H., Zelesky, M. F., Bainas, N. C., & Japiske, D. (Eds.). (2003). *Axial and radial turbines*. Concepts NREC.

NATO (Ed.). (2006). *Integration of tools and processes for affordable vehicles*. Workshop on Integrated Airframe Design.

Oesterreich, T. D., & Teuteberg, F. (2016). Understanding the implications of digitisation and automation in the context of Industry 4.0: A triangulation approach and elements of a research agenda for the construction industry. *Computers in Industry*, *83*, 121–139. https://doi.org/10.1016/j.compind.2016.09.006

Ouellet, Y., Garnier, F., Roy, F., & Moustapha, H. (2014). *A Preliminary Design System for Turbine Discs*.

Palmgren, A. (1924). *THE SERVICE LIFE BALL BEARINGS*.

Panchenko, Y., Moustapha, H., Patel, K., Dowhan, M. J., Marie-Victorin, B., & Drive, C. P. (2002). *Preliminary Multi-Disciplinary Optimization in Turbomachinery Design*.

Papadogiannis, D. (2023). *Coupled Large Eddy Simulations of combustion chamber-turbine interactions*.

Paramassivam, N. (2020). *Integration of Finite Element Analysis for Lifing in a Design and Analysis Platform for Gas Turbines*.

Peeters, M. F., Robichaud, M., & Guevremont, G. (2005). *ANALYTICAL PROCESS CERTIFICATION FOR GAS TURBINE DESIGN*.

Peoc'H, T. (2019). *AUTOMATION & INTEGRATION OF SECONDARY AIR SYSTEM WORKFLOW FOR MULTIDISCIPLINARY DESIGN OPTIMIZATION OF GAS TURBINES*.

Robinson, T. T., Armstrong, C. G., & Fairey, R. (2011). Automated mixed dimensional modelling from 2D and 3D CAD models. *Finite Elements in Analysis and Design*, *47*(2), 151–165. https://doi.org/10.1016/j.finel.2010.08.010

Saravanamuttoo, H. I. H., Rogers, G. F. C., Cohen, H., Straznicky, P. V., & Nix, A. C. (Eds.). (2017). *Gas turbine theory* (7th ed). Pearson Prentice Hall.

Sein, Henfridsson, Purao, Rossi, & Lindgren. (2011). Action Design Research. *MIS Quarterly*, *35*(1), 37. https://doi.org/10.2307/23043488

Shim, K. W., Monaghan, D. J., & Armstrong, C. G. (2002). Mixed Dimensional Coupling in Finite Element Stress Analysis. *Engineering with Computers*, *18*(3), 241–252. https://doi.org/10.1007/s003660200021

Singh, H. P., Rawat, A., Manral, A. R., & Kumar, P. (2021). Computational analysis of a gas turbine blade with different materials. *Materials Today: Proceedings*, *44*, 63–69. https://doi.org/10.1016/j.matpr.2020.06.486

Sonsino, C. (2007). Course of SN-curves especially in the high-cycle fatigue regime with regard to component design and safety. *International Journal of Fatigue*, *29*(12), 2246–2258. https://doi.org/10.1016/j.ijfatigue.2006.11.015

Tarkian, M. (2012). *Design automation for multidisciplinary optimization: A high level CAD template approach*. Department of Management and Engineering, Linköping University.

Tinga, T., Visser, W. P. J., De Wolf, W. B., & Broomhead, M. J. (2000). Integrated Lifing Analysis Tool for Gas Turbine Components. *Volume 1: Aircraft Engine; Marine; Turbomachinery; Microturbines and Small Turbomachinery*, V001T01A020. https://doi.org/10.1115/2000-GT-0646

Twahir, A. (2021). *An Architecture for a Multi-Disciplinary Integrated Turbine Rotor System Optimizer*.

Warde, S., Goddin, J., & Naker, S. (2019). *Materials and Simulation: From CAE to ICME A 2040 Vision*. NAFEMS. https://www.nafems.org/downloads/dropbox/watermarked/resource_center/conference/nwc19/nwc19-349.pdf

Yu, D., He, A., Yang, H., Yang, J., Cheng, K., Zhou, D., Wang, G., & Gu, L. (2015). A LCF Life Assessment Method for Steam Turbine Long Blade Based on Elastoplastic Analysis and Local Strain Approach. *Volume 8: Microturbines, Turbochargers and Small Turbomachines; Steam Turbines*, V008T26A036. https://doi.org/10.1115/GT2015-43955

Zaretsky, V. (1997). *A. Palmgren Revisited_A Basis for Bearing Life Prediction*.

Zucca, B. D., Gola, M. T., Troncarelli, E., & Pasquero, G. (2002). *Component Modes Synthesis Applied to a Thermal Transient Analysis of a Turbine Disc*.