Multidisciplinary Design Optimization of Turbomachinery Blade

by

Nima Bahrani

A thesis submitted in conformity with the requirements for the degree of Master of Applied Science

Department of Mechanical and Industrial Engineering

University of Toronto

© Copyright by Nima Bahrani 2015
Multidisciplinary Design Optimization of Turbomachinery Blade

Nima Bahrani

Master of Applied Science
Department of Mechanical and Industrial Engineering
University of Toronto
2015

Abstract

This thesis presents an approach for designing a more-efficient and reliable turbomachinery blade. A High-Cycle Fatigue testing is conducted to evaluate fatigue behavior of the design and determine the fatigue limit. The sine-dwell approach is employed to accelerate fatigue failure by applying a base excitation at the resonance frequency of the design. A multidisciplinary design optimization architecture integrated into a stochastic and population-based optimization algorithm is developed in order to improve efficiency and maintain strain level of the blade below the maximum allowable strain determined from the fatigue testing. The NURBS surface is employed to parameterize the shape of the blade and enable the optimizer to alter the design. The design optimization process involves fluid flow analysis coupled with structural analysis to evaluate the isentropic efficiency and the maximum strain level of the blade. The optimized design is compared with the baseline design and validated by other studies.
Acknowledgments

First of all, I would like to express my deep gratitude to my supportive supervisor Professor Kamaran Behdian for his guidance and encouragement. He has always been very warm-hearted and understanding at every stage of my Master’s study.

I would also like to thank the members of the ARL-MLS at the department of the mechanical and industrial engineering for providing an enjoyable work environment. I also thank Behzad Fazel for making time in his busy schedule to assist me with technical issues. I would also like to thank Mehran Ebrahimi for his kindness.

I am also grateful to Pratt & Whitney Canada and University of Toronto for jointly sponsoring my project and supporting me during my Master’s study.

Finally, I would like to thank my family for their strong support and encouragements.
# Table of Contents

Acknowledgments.......................................................................................................................... iii

Table of Contents........................................................................................................................... iv

List of Figures ...................................................................................................................................... vii

List of Tables ....................................................................................................................................... ix

List of Appendices ........................................................................................................................... x

## Chapter 1  **Introduction** ........................................................................................................... 1

1.1 Motivation and Objectives .................................................................................................. 1

1.2 Thesis Outline ..................................................................................................................... 3

## Chapter 2  **Literature Review** ............................................................................................... 4

2.1 Structural Design Optimization .......................................................................................... 4

2.1.1 Size Optimization .................................................................................................... 5

2.1.2 Shape Optimization ................................................................................................. 6

2.1.3 Topology Optimization ........................................................................................... 6

2.2 Shape Parameterization ....................................................................................................... 7

2.2.1 Basis Vector Approach ........................................................................................... 8

2.2.2 Discrete Approach .................................................................................................. 8

2.2.3 CAD-Based Approach ............................................................................................ 9

2.2.4 Free Form Deformation (FFD) Approach ............................................................... 9

2.2.5 Polynomial and Spline Approach ......................................................................... 10

2.3 Optimization Algorithms .................................................................................................. 12

2.3.1 Gradient-Based ..................................................................................................... 12

2.3.2 Gradient-Free ........................................................................................................ 14

2.3.3 Constraint Handling .............................................................................................. 17

2.4 NASA Rotor 67 (R67) ...................................................................................................... 18
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Experimental High-Cycle Fatigue Testing</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>3.1 Overview</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>3.2 Sine-Dwell Excitation</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>3.3 Sine Sweep</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>3.4 Test Equipment</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>3.5 Test Procedure</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>3.5.1 Test Setup</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>3.5.2 Calibration Testing</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>3.5.3 Sine Sweep Testing</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>3.5.4 Sine Dwell Testing</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>3.6 Results</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>3.6.1 Calibration Results</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>3.6.2 Dwell Test Results</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>3.6.3 S-N Curve</td>
<td>33</td>
</tr>
<tr>
<td>4</td>
<td>Multidisciplinary Design Optimization</td>
<td>35</td>
</tr>
<tr>
<td></td>
<td>4.1 Overview</td>
<td>35</td>
</tr>
<tr>
<td></td>
<td>4.2 R67 Geometry</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>4.3 Parameterization</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>4.4 Reverse Engineering</td>
<td>41</td>
</tr>
<tr>
<td></td>
<td>4.5 CFD Analysis</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>4.5.1 Geometry Modeling</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>4.5.2 Pre-processing</td>
<td>46</td>
</tr>
<tr>
<td></td>
<td>4.5.3 Solving</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>4.5.4 Post-Processing</td>
<td>51</td>
</tr>
<tr>
<td></td>
<td>4.6 Structural Analysis</td>
<td>53</td>
</tr>
<tr>
<td></td>
<td>4.7 MDO Architecture</td>
<td>55</td>
</tr>
</tbody>
</table>
List of Figures

Figure 1-1 Aero-engine performance trends ................................................................. 1
Figure 2-1 A size optimization problem ......................................................................... 5
Figure 2-2 A shape optimization problem ...................................................................... 6
Figure 2-3 An example of topology design. ................................................................. 7
Figure 2-4 NASA Rotor 67 ......................................................................................... 18
Figure 3-1 Base excitation of a SDOF system ............................................................... 21
Figure 3-2 Phase vs Frequency (dwell criterion) ......................................................... 23
Figure 3-3 HCF testing configuration ............................................................................ 27
Figure 3-4 Correlation between g level and maximum strain level of the blade .......... 30
Figure 3-5 Displacement spectrum obtained from sine sweep test at 10g. ................. 31
Figure 3-6 Displacement spectrum obtained from sine sweep test at 13g. ................. 32
Figure 3-7 Displacement spectrum obtained from sine sweep test at 15g. ................. 32
Figure 3-8 Strain vs the logarithmic scale of cycles to failure (S-N curve) ................. 33
Figure 4-1 Meridional blade coordinate ...................................................................... 37
Figure 4-2 Control points of the NURBS surface ....................................................... 40
Figure 4-3 Flow chart of Reverse Engineering for finding control points ................. 43
Figure 4-4 Boundaries of the single blade passage .................................................... 45
Figure 4-5 h−s diagram of the actual and isentropic process ....................................... 53
Figure 4-6 Flowchart of the automated MDO architecture ........................................... 56
Figure 4-7 Fitness vs iteration associated with proposed reverse engineering............................. 62
Figure 4-8 Experimented efficiency of the R67 obtained from................................................... 64
Figure 4-9 Total pressure contour of the pressure side................................................................. 65
Figure 4-10 Total pressure contour of the suction side................................................................. 65
List of Tables

Table 2-1 Basic design specifications of NASA Rotor 67 ........................................................... 19
Table 3-1 Testing equipment provided by ARL-MLS lab.............................................................. 25
Table 3-2 Dwell test Results ......................................................................................................... 30
Table 4-1 Blade Coordinate Parameters ....................................................................................... 38
Table 4-2 Notation of design variables ........................................................................................... 41
Table 4-3 Reverse engineering parameters ..................................................................................... 44
Table 4-4 CFD boundary conditions .............................................................................................. 49
Table 4-5 CFD solver Control Options .......................................................................................... 51
Table 4-6 FEM solver parameters .................................................................................................. 54
Table 4-7 PSO parameters ............................................................................................................ 57
Table 4-8 Design variables bounds .............................................................................................. 58
Table 4-9 Design variables associated with control points of the R67 baseline ......................... 61
Table 4-10 Design variables associated with control points of the optimized R67 ................. 63
Table 4-11 Design evaluation of R67 ........................................................................................... 63
List of Appendices

**Appendix A:** Navier-Stokes Equations Applied to Turbomachinery ........................................ 74

**Appendix B:** Matlab Code ........................................................................................................... 77
Chapter 1
Introduction

1.1 Motivation and Objectives

Several studies have been performed to improve the turbomachinery technology for aerospace propulsion, such as aircraft engines, or power generation, such as gas turbines. Figure 1-1 demonstrates engine improvements at Rolls-Royce where the aircraft fuel burn was reduced by 60% approximately and the engine specific fuel consumption (sfc) was improved by 45% [1]. Besides the reduction in fuel burn, designers are always concerned about the engine safety since engine failure results in catastrophic aircraft disaster. Therefore, a lot of research has been conducted to investigate the cause of engine failure.

![Aero-engine performance trends](image)

Figure 1-1 Aero-engine performance trends [1]

Blade design plays a vital role in modern high-performance aircraft engines as small variations in the shape of the blade can result in higher efficiency of the turbomachinery. Moreover, blade failure is one of the significant causes of engine failure. Turbomachinery design involves several
analyses such as CFD analysis, stress analysis, fatigue analysis and heat transfer analysis. The goal of this thesis is to increase turbomachinery efficiency while the maximum strain of the blade must not exceed maximum allowable strain. The fatigue life of the blade is calculated using stress analysis, depending on the material and the maximum stress or strain of the blade in operating condition. To satisfy strain limit, maximum strain or stress of design should not exceed the fatigue limit of the material obtained from the S-N curve using experimental testing on the identical material. Additionally, maximizing isentropic efficiency is the objective function which is calculated using fluid analysis for a specified blade design. The NASA Rotor 67, which is a transonic axial fan of an aircraft engine, has been selected as a case study in this research to evaluate the proposed MDO methodology.

The major objectives of this project are as follows:

- Investigate the fatigue behavior of the blade to determine the strain fatigue limit for blade design by performing high-cycle fatigue testing.

- Parameterize the geometry of the blade to enable the optimizer to implement shape optimization.

- Perform flow analysis to evaluate the efficiency of the turbomachinery and finite element based analysis to evaluate the strain levels of the design.

- Define MDO architecture to integrate the optimization disciplines into an optimization algorithm.

- Compare the optimized design with the baseline and other references.
1.2 Thesis Outline

The thesis consists of five chapters described by the following paragraphs in overview of each chapter.

Chapter 2 presents the literature review of the structural optimization and optimization algorithms. A brief review of the NASA R67, which has been selected as a case study to implement the multidisciplinary design optimization, is also presented in this chapter.

The fatigue limit of the material should be determined based on the material behavior and the shape effect. Considering this fact, a high-cycle fatigue (HCF) test should be set up to evaluate number of cycles to failure and the strain level of the blade for generating S-N curve which is presented in Chapter 3.

The implementation of multidisciplinary design optimization is discussed in Chapter 4. The design optimization consists of the geometrical parameterization, CFD, and structural analysis integrated with a MDO architecture which defines the communications of all multidisciplinary design optimization components. The optimization results are presented in this chapter as well.

Chapter 5 summarizes the concluded remarks of the thesis and recommends further actions in order to steer the direction of future research.
Chapter 2
Literature Review

2.1 Structural Design Optimization

Structural design optimization originates from these two terms, namely structural design and optimization. A structure in mechanical engineering is defined as “any assemblage of material which is intended to sustain loads”[2]. Designing a structure may require several analyses such as stress analysis, fluid analysis, fatigue analysis, heat transfer analysis, acoustic analysis, etc. depending on purposes and restrictions. Optimization is a process that aims to achieve the best solution for a problem by changing parameters. Traditionally, designers changed their parameters according to best solution acquired by means of trial and error, which was computationally expensive and did not guarantee the global best solution. By developing new optimization algorithms and methods, the design of a structure can be optimized automatically within iterative loops. In order to combine optimization with structural design, structural parameters should be defined mathematically so that they can be used as inputs for optimization algorithms.

An optimization problem is represented by design variables, state variables, objective functions and constraint functions. Design variables describe the design parameters which can be changed during the optimization process by the optimizer. Varying the design variables results in a new design, and by searching several new designs, the optimum solution can be found. The design variables may represent simple geometrical parameters such as length, thickness, etc. may relate to the complex interpolation of the geometry of the design or may include the material properties, external loads, environmental parameters, etc. Analyzing the design may consist of solving a system of equations, such as the Navier-Stokes equations in fluid mechanics or static equilibrium equations in structural mechanics, which computes a set of responses, known as the state variables. For instance, the displacement, stress, strain, force, pressure, and velocity which are calculated in analysis equations could be deemed as state variables.
In general, the objective function is a function of all variables which evaluates design and could be used for the optimizer in order to minimize or maximize the value of the objective function. For instance, in turbomachinery design, maximizing efficiency can be an objective function while, in truss design, minimizing the weight of the structure can be an objective function. In other words, minimizing or maximizing the objective function is the goal of the optimization problem which should be chosen by the designer depending on the design purposes. An optimization problem can have multi-objective functions. The constraint function generally is a function of all variables which should not exceed a certain limit in order to restrict the design or response of design such as length, displacement, stress, etc.

Depending on the optimization purpose and the geometrical capabilities of the structure, structural optimization problems are classified into three categories: size, shape, and topology optimization.

### 2.1.1 Size Optimization

In the size optimization approach, geometrical parameters which describe the geometry of the design are chosen as design variables. These parameters are defined simply in both CAD programs and mathematical representation. It could be the length of a rectangle or the thickness of a bar or the cross-sectional areas of a truss. Therefore, variation of design variables in this approach modifies the design in terms of geometrical dimensions and allows the optimizer to search for the optimum solution by modifications. A size optimization example for a truss structure has been shown in Figure 2-1 [3]. In this case the cross-sectional areas of the truss are used as design variables. Size optimization does not redesign the topology of the structure such as number of holes, beams, etc., and hence, the topology and boundaries of the structure are determined prior to the optimization process.

![Figure 2-1 A size optimization problem [3]](image-url)
2.1.2 Shape Optimization

In the shape optimization approach, the shape of boundaries is considered to be controlled by the optimizer. The shape of the design is changed by varying boundaries and enables the optimizer to search for the optimum solution of the problem. Allowing boundaries to be changed during the optimization dramatically ameliorates design space and offers more design possibilities in comparison with the size optimization. The boundaries are usually represented by curves or surfaces which require parameterization before it can be applied to the optimization process [4]. Therefore, for sophisticated geometries, extra mathematical tools should be involved to adjust alteration of boundaries to CAD programs or other geometrical representation tools. In this approach, a new boundary is not permitted to be created or defined. Instead, only the shape of existing boundaries can be altered. For instance, creating a hole inside the boundaries of a sheet will make a new boundary which is not acceptable in this approach. A two-dimensional shape optimization problem has been shown in Figure 2-2 [3].

![Figure 2-2 A shape optimization problem [3.](image)](image)

2.1.3 Topology Optimization

In the topology optimization approach, the topology of the structure is allowed to vary. This capability makes the topology optimization the most general form of structural optimization and improves the design space by considering more design possibilities. Theoretically, shape optimization is a subclass of topology optimization which forbids defining a new boundary. However, in practice, implementing topology optimization requires relatively different techniques [3]. Several methods have been proposed for topology optimization [5]–[7]. Generally these methods can be categorized into two approaches: discrete element approach and continuum approach. In the discrete element approach, the design domain of the structure is
discretized into a finite set of possible locations of discrete structural members or substructures. By changing the thickness of each substructure between zero and a limited value, the topology of the design can be found during the optimization. In the continuum approach, the design domain of structure is considered a continuum mixture of a material with different distribution of material density which is allowed to vary throughout the optimization process to search for the optimal distribution. A two dimensional topology optimization problem has been shown in Figure 2-3 [8].

![Figure 2-3 An example of topology design, a) design domain b) optimized solution [8].](image)

### 2.2 Shape Parameterization

Since the boundary of the geometry is remodeled during the optimization, a mathematical method should be considered to determine how the boundaries of the shape are modeled by design variables. Shape parameterization methods enable the optimizer to link design variables to geometry modeling. Therefore, changing the design variables results in remodeling the geometry and allows for searching different possible geometries to find the optimum design. The purpose of the shape parameterization is to mathematically define interactive representation of shape and provide geometric transformation to reduce the point cloud into a compact set of data.
Several shape parameterization techniques have been proposed which can be categorized into eight approaches [9]: Basis Vector, Domain Element, Discrete, Analytical, Free Form Deformation (FFD), Partial Differential Equation (PDE), Polynomial and Spline, and Computer Aided Design (CAD). One of these approaches is chosen depending on the application and then utilized in a shape optimization problem. The compatibility of the chosen approach with the geometry of the design and the analysis disciplines should be investigated in terms of efficiency, ease of implementation, sensitivities for grid models, and feasibility of computational time. A short review for some of the most popular approaches is presented here.

2.2.1 **Basis Vector Approach**

In this approach, the shape changes can be defined as [10]

$$G_{\text{new}} = G_{\text{initial}} + \sum_{i} T_{ji} X_i + Q \quad (2.1)$$

where $G_{\text{new}}$ is the design shape, $G_{\text{initial}}$ is the initial shape, $X_i$ is the design variable vector, the columns of matrix $T_{ji}$ are the basis shapes associated with the design variables, the vector $T_{ji} X_i$ give the perturbation of the designed grids, and the constant term vector $Q$ is computed to perform requisite modification. In practice this technique is only applicable for relatively simple geometries due to high computational setup time, inconsistency in generating a set of basis vectors, and the possibility of modeling rough geometry. On the other hand, grid regeneration and deformation can be performed conveniently in this approach.

2.2.2 **Discrete Approach**

The coordinates of the boundary points are utilized as design variables in the discrete approach. It is simplest method for parameterizing geometry of the design. However, it is arduous to control the smoothness of the geometry, and it also generates a huge number of design variables which effectively worsens computational cost. For multidisciplinary applications, different grid generation is required for each discipline, which may result in inconsistent parameterization. Multipoint constraints and dynamic adjustment of lower and upper bounds on the design variables are used to prevent generating rough geometry.
2.2.3 CAD-Based Approach

Commercial CAD software enables designers to represent a physical solid geometry of the design by using the boundary representation method (B-Rep) or the constructive solid geometry method (CSG) which can provide a fully defined mathematical representation of the solid object. Equipping CAD systems with feature-based solid modeling (FBSM) techniques allows designers to construct some topological features such as holes, bosses, fillets, sweeps, and shells by using Boolean operations such as intersections and unions. These features are dimension-driven that can be modified by varying the dimension of the feature. Therefore, designers can create a parametric model and define relations and constraints for each feature in the model in order to enable the optimizer to modify the model and search for the optimum solution. In this approach, the dimension of the feature is chosen as a design variable. The optimizer particularly changes the dimensions of the features which results in the modification of features for an existing model. Although, design changes are not time consuming in this approach, meshing the geometry for finite element or computational fluid dynamic purposes may increase the total computation time of the optimization process due to mesh regenerating for each modified model. In addition, it is challenging to create consistent parameterization and connect the CAD systems to analysis software such as FE software and CFD software. However, utilizing this approach creates smoother surfaces for geometry and provides a compact set of design variables.

2.2.4 Free Form Deformation (FFD) Approach

Free Form Deformation (FFD) technique originates from Soft Object Animation (SOA) in the computer graphics field [11] which provides algorithms for deforming an object regardless of its geometrical description. FFD defines a parallelepiped lattice embedding an object which is assumed to be deformable. The object is deformed by moving control points of the lattice from their initial positions. The object is deformed by moving control points of the lattice from their initial positions. The deformation function mathematically is defined by a trivariate Bernstein polynomial [11] as

\[
X_{\text{ffd}} = \sum_{i=0}^{\eta_i} \sum_{j=0}^{\eta_j} \sum_{k=0}^{\eta_k} C_i^{n_i} s^i (1-s)^{n_i-i} C_j^{n_j} t^j (1-t)^{n_j-j} C_k^{n_k} u^k (1-u)^{n_k-k} P_{ijk} \tag{2.2}
\]
where $X_{\beta\alpha}$ is a vector containing the Cartesian coordinates of a displaced point, $P_{ijk}$ is a vector containing the Cartesian coordinates of the control point, $(s,t,u)$ is a local coordinate system defined in the lattice, $0 < s < 1$, $0 < t < 1$, $0 < u < 1$, $C$ is the combination operator, $n_i+1$, $n_j+1$, and $n_k+1$ is the number of grid points in the $s$, $t$, and $u$ direction, respectively.

2.2.5 Polynomial and Spline Approach

In this approach, curves or surfaces of the design are parameterized by defining adequate control points and using polynomial and spline functions in order to integrate the optimizer algorithm into interactive shape design. Numerous types of polynomial and spline functions have been utilized for shape parameterization to define a parametric curve or surface such as Bezier, B-spline and Non-uniform rational B-spline (NURBS).

A given Bezier surface of $n \times m$ degree is defined using $(n+1) \times (m+1)$ control points and Bernstein polynomial function as

$$S(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} B_i^n(u)B_j^m(v)P_{i,j}$$

(2.3)

$$B_i^n(u) = \binom{n}{i} u^i (1-u)^{n-i}$$

where $P_{i,j}$ is coordinates of a given control point, $S(u,v)$ is the parametric surface. By considering the control points’ coordinates as design variables, the optimizer is authorized to alter the shape of the design. Bezier representation is efficient and accurate for shape optimization of simple surfaces or curves. However, for more sophisticated geometries, Bezier representation requires a high-degree form in order to manipulate the formulation and ameliorate accuracy. Increasing the degree of Bezier representation causes longer computation time of coefficients and larger round-off errors.

The given B-spline surface is defined using $(n+1) \times (m+1)$ control points, two knot vectors, and the product of the univariate B-spline function as
\[ S(u, v) = \sum_{i=0}^{n} \sum_{j=0}^{m} N_{i,p}(u) N_{j,q}(v) P_{i,j} \quad (2.4) \]

where \( P_{i,j} \) is the control point and \( N_{i,p} \) is the p-degree spline basis function. Several methods have been proposed for computing a basis function [12]. DeBoor and Cox [13]-[14] have developed a recurrence formula which is most efficient for computer programming purposes. By supposing a knot vector as a set of a non-decreasing sequence of real numbers \( (u_i) \), the \( i^{th} \) B-spline basis function of p-degree is formulated as

\[
N_{i,0}(u) = \begin{cases} 
1 & \text{if } u_i \leq u \leq u_{i+1} \\
0 & \text{otherwise} 
\end{cases} 
\]

\[
N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) 
\]

(2.5)

While Bezier and B-spline representations provide many advantages, complex geometries such as circular, elliptical, conical, cylindrical, or spherical shape, cannot be parameterized accurately by using these functions [12]. As a result, the rational function, which is well-known from classical mathematics for representing all conical curves, has been combined with polynomials or B-spline functions in order to obtain a more accurate representation of curves or surfaces. The non-uniform rational B-spline (NURBS) is one of the examples of using the rotational function, which is presented in the following formulation for a surface of degree \( p \) in the \( u \) direction and degree \( q \) in the \( v \) direction.

\[ S(u, v) = \sum_{i=1}^{n} \sum_{j=1}^{m} R_{i,j}(u,v) P_{i,j} \quad (2.6) \]

\[ R_{i,j}(u,v) = \frac{N_{i,p}(u) N_{j,q}(v) \omega_{i,j}}{\sum_{k=1}^{n} \sum_{l=1}^{m} N_{k,p}(u) N_{l,q}(v) \omega_{k,l}} \quad (2.7) \]
$P_{i,j}$ is control point, $\omega_{i,j}$ is the weight coefficient, $N_{i,p}(u)$ is the non-rational B-spline basis function defined on the knot vector, and $S$ is the parameterized surface.

Using the NURBS function yields efficient geometric representation, accurate shape and compact data storage [12]. It can be utilized in homogenous coordinates in $n$ dimensional space. Since both control point movement and weight modification can be utilized to alter the shape of the NURBS surface, it is a suitable parameterization for shape optimization.

2.3 Optimization Algorithms

A general optimization problem is defined by objective functions, constraints, and design variables. Optimization algorithms manage to find an optimum solution of the objective functions by changing the design variables in a design space subjected to the constraints. The optimization algorithms have a strong background in the mathematics. Several algorithms have been proposed to search for the optimum solution of the problem. These algorithms can be classified under two groups: gradient-based optimization and gradient-free optimization. Gradient-based optimization algorithms utilize derivatives of the objective function to update design variables while gradient-free algorithms usually use artificial intelligence to search globally using observed stochastic phenomena in nature.

2.3.1 Gradient-Based

Gradient-based methods require evaluation of the derivatives of the objective function. The gradients for single-objective problems or the Jacobian matrix for multi-objective problems are used to determine the search direction. These methods are often computationally expensive since the gradient should be computed in each iteration of the optimization. Additionally, discontinuities in the objective functions result in inefficient gradient computation as the function should be continuous for most numerical approximations of derivatives.

Various gradient-based methods have been developed. A brief recap of the most commonly used algorithms is given here.
2.3.1.1 Steepest Descent Method

The steepest descent method is a line search method which employs the gradient vector of the objective function \( g(\vec{x}_k) \) at the vector of the design variables \( \vec{x}_k \) for the major iteration \( k \) as the search direction. Therefore, the vector of the design variables for next iteration \( \vec{x}_{k+1} \) can be expressed as

\[
\vec{x}_{k+1} = \vec{x}_k + \alpha_k p_k
\]

\[
p_k = \frac{-g(\vec{x}_k)}{\|g(\vec{x}_k)\|}
\]

where \( p_k \) is the normalized gradient vector, and the positive scalar \( \alpha_k \) represents the step length, which can be calculated by assuming the first-order variation in the vector of the design variables same as the variation associated with the previous step. Therefore:

\[
\alpha_k = \alpha_{k-1} \frac{g_{k-1}^T P_{k-1}}{g_k^T p_k}
\]

The steepest descent direction is considered as the simplest choice for search direction of the line search method [15]. However, this method requires a large number of evaluations to converge to an optimum point [16]. Also, the initial design variables remarkably affect the number of evaluations. Moreover, the updated design variables may be trapped into a local optimum instead of the global optimum point as the gradient is zero at any local optimum.

2.3.1.2 Quasi-Newton Method

The quasi-Newton method originates from Newton’s method which utilizes a second-order Taylor’s series expansion of the objective function in order to determine the Newton’s direction for line search. The objective function can be approximated by Taylor’s series as

\[
f(\vec{x}_k + p) \approx f(\vec{x}_k) + p^T \nabla f(\vec{x}_k) + \frac{1}{2} p^T \nabla^2 f(\vec{x}_k) p
\]
By differentiating the objective function with respect to \( p \) which is called the Newton step, the Newton direction is obtained as \( p_k = -H_k^{-1}g_k \) where \( H \) is the Hessian matrix and \( g \) is the gradient vector. Newton’s method requires the calculation of the first and second derivatives of the function which decreases the computational efficiency significantly, while the quasi-Newton method utilizes the first order information to approximate the second order information based on an evaluation of the function and its gradients. Davidon proposed the first Quasi-Newton method [17] which has been modified by Fletcher and Powell [18] and known as the Davidon-Fletcher-Powell (DFP) method. DFP method approximates the Hessian matrix by

\[
H_{k+1} = H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \frac{s_k s_k^T}{y_k^T s_k} \tag{2.11}
\]

where \( y_k \) is calculated in terms of gradient difference as \( y_k = g_{k+1} - g_k \), and \( s_k \) is the Newton step towards the optimum point as \( s_k = \alpha_k p_k \). Several methods have been proposed for approximation of the Hessian matrix such as Broyden–Fletcher–Goldfarb–Shanno algorithm (BFGS) [19] or Symmetric rank-one (SR1) [20] in order to develop the line search efficiently.

### 2.3.2 Gradient-Free

Gradient-free methods search for the optimum solution without evaluation of the derivatives. These methods imitate stochastic structures witnessed in nature such as molecular evolution of genes or stylized movement of bees. Gradient-based methods are incapable of handling discontinuous and noisy functions as they steer search direction based on derivatives. Gradient-free methods provide the ability to solve problems which are impractical to optimize by Gradient-based methods. Some of the most popular gradient-free methods have been discussed here.

#### 2.3.2.1 Genetic Algorithm

The genetic algorithm (GA) is a stochastic-based search technique which employs the pattern of natural evolution of genes to update design variables. The first effort for using genetic algorithm in an optimization problem was performed by John Holland [21]. A genetic algorithm evaluates
the fitness function for a set of individual design variables known as a generation at each iteration and updates the generation until the optimum solution is achieved. The genetic algorithm updates the generation of the design variables using three main operators including generic selection, reproduction (crossover), and variation (mutation).

- Selection

During each iteration a valued portion of the present population is selected to generate a new population. Individuals (design variables) are valued based on the accumulated normalized fitness known as the cumulative probability. Several approaches can be used for selecting individuals. In the roulette-wheel approach a set of random numbers between zero and one is generated to choose the first individual which has a cumulative probability greater than the associated random number. In the tournament approach, repeatedly, the best individual in a random subset of the population is selected.

- Crossover

Selected individuals are used as parents for breeding in order to generate children as new population. Crossover techniques create a new population by sharing characteristics of the parents as the biological observations suggest. Usually, in genetic algorithms each individual is modeled as a binary string. One-point crossover technique defines a point on both parents’ strings and exchanges all string data beyond the point between parents to generate children. Two-point crossover technique uses two points for each parent to exchange data while the cut and splice technique defines a separated point for each parent.

- Mutation

Generic mutation adds randomness in new population in order to maintain diversity in the generation. The mutation is imposed by changing a bit string at a point which is usually selected randomly.
2.3.2.2 Particle Swarm Optimization

Particle swarm optimization (PSO) originates from the concept of collective behavior of a system known as swarm intelligence (SI) which was expressed in context of cellular robotic system by G. Beni and J. Wang [22]. J. Kennedy and R. Eberhart developed the first particle swarm and population-based optimization algorithm inspired by swarm intelligence and simulations of bird flocking [23]. PSO considers each design point as a particle which moves in design space towards the optimum point. The algorithm modifies the movement of particle using its velocity as presented by

$$x_{k+1}^i = x_k^i + v_{k+1}^i \Delta t$$

(2.12)

and the velocity is updated by

$$v_{k+1}^i = w v_k^i + c_1 r_1 \frac{(p_k^i - x_k^i)}{\Delta t} + c_2 r_2 \frac{(p_{gb}^i - x_k^i)}{\Delta t}$$

(2.13)

where \(x_k^i\) is the vector of the design variables associated with \(i^{th}\) particle of the population at iteration \(k\), \(v_k^i\) represents its velocity, \(r_1\) and \(r_2\) are random numbers between zero and one, \(c_1\) and \(c_2\) represent the cognitive and social coefficient, respectively. \(w\) is inertia factor, and \(\Delta t\) is the time step factor.

The cognitive term in formulation 2.13 attracts the present position of the \(i^{th}\) particle towards the best ever position associated with itself \(p_k^i\) which has been obtained for all iterations before. On the other hand, the social term attracts the current position towards the fittest position obtained from all populations and generations. These terms play a significant role in the convergence and stability of particle swarm optimization. R. Perez and K. Behdinan [24] rearranged the position and velocity equations by combining them into a matrix form as
which can be considered as a discrete-dynamic representation. They developed the stability criteria for the PSO algorithm by solving the dynamic matrix characteristic equation as

\[
0 < c_1 + c_2 < 4
\]

\[
\frac{(c_1 + c_2)}{2} -1 < w < 1
\]

These criteria make the PSO algorithm practical and ideal for structural optimization as they promise convergence to an equilibrium point [25]. The algorithm can also be modified for using parallel computing which enables users to employ more processors similar to other population-based algorithm.

### 2.3.3 Constraint Handling

Structural optimization problems are constrained frequently while most of the optimization algorithms are developed based on unconstrained problems. Therefore, several methods have been suggested in order to cope with different types of constraints and accommodate the constrained problems with unconstrained-based optimization algorithms. The penalty function approach is widely used and able to handle all types of constrained problems. In this approach, a penalty function dependent on constraints is augmented to the objective function in order to define a fitness function. There are several methods that define the penalty function. Some of the most famous methods have been presented here.

- **Quadratic Penalty Function**

The quadratic penalty function is defined as

\[
Fitness = f(x) + \sum_{i}^{m} \rho(\max[0,c_i(x)])^2
\]
where $f$ is the objective function, $c_i$ represents the constraints of the problem, and $\rho$ is a positive scalar known as the penalty parameter. When the design point violates the constraints, the objective function is penalized by adding a relatively great value to the objective function.

- **Logarithmic Barrier Function**

The logarithmic barrier function $\psi$ is defined as

$$
\psi(x) = \begin{cases} 
-\mu \sum_{i=1}^{m} \log(-c_i(x)) & c_i(x) < 0 \\
+\infty & \text{otherwise (constraint violation)} 
\end{cases}
$$

where $\mu$ is a positive scalar known as the barrier parameter. The fitness value is obtained by adding the logarithmic barrier function to the objective function.

### 2.4 NASA Rotor 67 (R67)

NASA Rotor 67 (R67) is a low-aspect-ratio and transonic axial-flow rotor which is used in the first-stage rotor of a two-stage fan (Figure 2-4 [26]). R67 was designed in the 1970s to develop efficient and lightweight engines for short-haul aircrafts [27]. Aerodynamic design specifications of the R67 have been published in NASA technical papers [28], [29]. Some major design specifications have been presented in Table 2-1.

![Figure 2-4 NASA Rotor 67](C-88-05331)

Figure 2-4 NASA Rotor 67 [26]
NASA Rotor 67

<table>
<thead>
<tr>
<th>Number of Rotor Blades</th>
<th>22</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rotational Speed</td>
<td>16043 [rpm]</td>
</tr>
<tr>
<td>Design Pressure ratio</td>
<td>1.63</td>
</tr>
<tr>
<td>Design Mass Flow</td>
<td>33.25 [kg/s]</td>
</tr>
</tbody>
</table>

Table 2-1 Basic design specifications of NASA Rotor 67

Several studies have been conducted on the R67 through experiment and simulation. The NASA Lewis research center obtained the detailed flow field measurements for the R67 using laser anemometers [30]. Based on the experimental measurements, they developed a three-dimensional flow analysis code to compute the flow field in the transonic fan rotor and compared the results with experimental data for R67 [26]. They optimized the efficiency of the R67 using B-spline shape parameterization, the evolutionary algorithms, and TRAF3D code which solved the three-dimensional, full Reynolds-averaged Navier-Stokes equations of the fluid flow [27]. Individual research has also been performed on the R67 involving several areas such as flow analysis, structural analysis and vibration analysis. W. Tiow and M. Zangeneh redesigned the R67 blade by varying the static pressure loading characteristics to improve the strong shock formation near the tip [31]. C. Zhang, Z. Ye and F. Liu evaluated the aeroelastic characteristics of R67 and studied the self-excitation of the blade known as flutter in the second bending mode [32]. V.J Fidalgo, C. A. Hall, and Y. Colin applied computational fluid analysis methods to study the interaction between a transonic fan and an inlet total pressure distortion using a full-annulus R67 model [33]. G. Abate performed an aerodynamic optimization to maximize the efficiency of the R67 using computational fluid analysis and Bezier curves to parameterize sections of the blade [34].
Chapter 3
Experimental High-Cycle Fatigue Testing

3.1 Overview

As presented in the introduction, an objective of this study is to maintain maximum strain of the blade below fatigue limit which has been considered in terms of a constraint. The fatigue limit of the material should be determined in order to satisfy the definition of the constraint, which states that the maximum strain level of the blade in operating condition should not exceed the fatigue limit. Fatigue behavior of a material essentially depends on material properties. However, geometrical effects can alter fatigue behavior of the material. Therefore, the most reliable method to determine fatigue behavior of the design is to obtain experimentally the strain-life (S-N) curve of the material using a specimen which has been fabricated from an identical material.

High-cycle fatigue (HCF) had been identified as the most important cause of blade failure in gas turbine engines [35]. As a result, an experimental high-cycle fatigue testing is required to be conducted to investigate the fatigue behavior. High-cycle fatigue testing is a type of fatigue test which evaluates the fatigue behavior in high cycle regime approximately below $10^7$-$10^8$ cycles [36]. The sine-dwell excitation technique is one of the HCF testing approaches which has been implemented in this research to study the fatigue characteristics of the blade.

3.2 Sine-Dwell Excitation

Sine-dwell excitation accelerates fatigue failure by applying a vibrating force on a part at its resonant frequency due to the vibration amplitude of the part being at maximum at the resonance frequency which results in higher stress or strain levels. In addition, the testing time will be reduced if the resonance frequency is selected at higher frequencies especially after 100 Hz. In other words, choosing higher resonance frequency makes the high-cycle fatigue testing practical where $10^7$ cycles of vibration is achieved in lower testing time at higher frequencies.

The excitation force is applied to the support points known as base. A generalized base excitation problem has been shown in Figure 3-1 schematically for a single degree-of-freedom (SDOF) system.
The base excitation \( y(t) = Y \sin \omega t \) causes the mass \( m \) to vibrate as a function of excitation frequency and amplitude \( x(t) = X \sin(\omega t + \phi) \). By solving the differential equation of the system the ratio of the response amplitude \( X \) to base excitation amplitude \( Y \) known as the displacement transmissibility and the phase difference \( \phi \) can be expressed as

\[
\frac{X}{Y} = \sqrt{\frac{1 + (2\zeta r)^2}{(1-r^2)^2 + (2\zeta r)^2}}
\]

\[
\phi = \tan^{-1}\left[\frac{2\zeta r^3}{1 + (4\zeta^2 - 1)r^2}\right]
\]

where \( \zeta = \frac{c}{2m\omega_n} \), \( r \) is given by \( r = \frac{\omega}{\omega_n} \), \( \omega \) and \( \omega_n \) represent the excitation and the natural frequency, respectively.

Sine-dwell tests excite structural resonances with a sinusoidal force which is applied at a fixed frequency or at a tracking resonance frequency with respect to vibration amplitude or phase [37]. In the fixed frequency method, the vibrating force is applied at the predetermined resonance...
frequency which is fixed throughout the test while in the tracking mode the resonance frequency is updated. As mentioned, the sine-dwell test is utilized to promote fatigue failure. Damage progression and thermal effects can cause variations in mechanical properties such as stiffness, which results in changes in resonance frequency [38]. Consequently, the resonance frequency of the structure shifts progressively during the test. Therefore, the resonance frequency should be tracked and modified during the test, in order to obtain maximum cumulative fatigue in the structure. In the tracking method, the frequency of the vibrating force is updated to instantaneous resonant frequency using a computerized feedback control system which updates these frequencies according to amplitude or phase criterion.

Amplitude dwell criterion has been presented as

\[
\begin{align*}
\omega_{n+1} &= \omega_n + \hat{\omega}_s \quad \text{if } x_n > x_{n-1} \\
\omega_{n+1} &= \omega_n - \hat{\omega}_s \quad \text{if } x_n < x_{n-1}
\end{align*}
\]

(3.2)

where \( \hat{\omega}_s \) is calculated by \( \hat{\omega}_s S = \frac{\Delta \omega}{S} \), \( \Delta \omega \) is the frequency bandwidth which is predetermined manually, \( S \) is the sine-dwell tracking step factor, \( \omega_n \) demonstrates the excitation frequency at \( n \) cycles, and \( x \) represents the response amplitude. The frequency is updated depending on the vibration amplitudes of previous steps. The purpose of this criterion is to maintain the amplitude at maximum as it is imposed in the formulation 3.2.

Phase dwell criterion has been presented as

\[
S_p = \frac{\phi_n - \phi_{n-1}}{\omega_n - \omega_{n-1}} , \quad \omega_T = \frac{1}{S_p} (\phi_T - \phi_{n-1}) + \omega_{n-1}
\]

\[
\begin{align*}
\omega_{n+1} &= \omega_n + \hat{\omega}_s \quad \text{if } \omega_T - \omega_n > \hat{\omega}_s \\
\omega_{n+1} &= \omega_T \quad \text{if } \omega_T - \omega_n \leq \hat{\omega}_s
\end{align*}
\]

(3.3)

where \( \hat{\omega}_s \) is calculated by \( \hat{\omega}_s S = \frac{\Delta \omega}{S} \), \( \Delta \omega \) is the frequency bandwidth which is predetermined manually, \( S \) is the sine-dwell tracking step factor, \( S_p \) is the slop of the plot of \( \phi \) versus \( \omega \), \( \phi_T \)
is the target phase, $\omega_n$ demonstrates excitation frequency at $n$ cycles, and $\phi$ represents input-response phase difference.

The phase dwell criterion updates the excitation frequency to maintain the input-output phase difference at a targeted value. The control system compares the phase difference with the target phase and updates the frequency with respect to the formulation 3.3 as shown in Figure.3-2.

Figure 3-2 Phase vs Frequency presenting calculation of the phase dwell criterion for updating frequency
3.3 Sine Sweep

The resonance frequency of the structure must be determined before performing a sine-dwell test since the specimen is excited at resonance frequency. The resonance frequency can be obtained using either a free vibration test such as an impact hammer test or a forced vibration test such as a sine sweep test. In the sine sweep test, the specimen is excited with a sinusoidal loading which is swept through a defined range of frequencies. By measuring the response of the specimen over the defined range of frequencies and calculating the frequency response function (FRF), the resonance frequency is determined. The base excitation of sine sweep has been shown by

$$y(t) = A\sin(2\pi\left(\frac{f_1(2^N -1)}{R \ln(2)}\right))$$  \hspace{1cm} (3.4)

where $y(t)$ is a linear sine sweep forced displacement for base excitation, $R$ is the sweep rate which is calculated by $R = \frac{N}{t_2 - t_1}$ in terms of Octaves, $N$ is the number of Octaves in the frequency range which is calculated by $N = \frac{\ln(f_2/f_1)}{\ln(2)}$, $f_1$ is start frequency, $f_2$ is stop frequency, $t$ represents time duration of the sine sweep, and $A$ demonstrates forced displacement amplitude.

At the resonance frequency, the frequency response function of the specimen has a peak. Therefore, by detecting the peak within FRF, the resonance frequency is determined.

3.4 Test Equipment

Sine-Dwell test requires an electrodynamic shaker to excite the specimen, an amplifier to support the shaker, a controller which is connected to a computer to utilize a computerized feedback control system, a data acquisition system to measure signals of sensors, a vibrometer to measure the vibration of the specimen, and some complementary sensors such as strain gauges and accelerometers.
The Advanced Research Laboratory for Multifunctional Lightweight Structures (ARL-MLS) at the University of Toronto is equipped with the instruments required for high-cycle fatigue testing. The equipment, which has been utilized in this research, has been listed in Table 3-1.

<table>
<thead>
<tr>
<th>Equipment Type</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electrodynamic Shaker</td>
<td>2110E MODALSHOP</td>
</tr>
<tr>
<td>Power Amplifier</td>
<td>2050E09 MODALSHOP</td>
</tr>
<tr>
<td>Cooling Unit</td>
<td>AMD 71ZBA2 LAFE</td>
</tr>
<tr>
<td>Digital-to-Analog Convertor</td>
<td>LMS</td>
</tr>
<tr>
<td>SCADAS Mobile Controller</td>
<td>LMS</td>
</tr>
<tr>
<td>Software</td>
<td>LMS Test.lab 12</td>
</tr>
<tr>
<td>Strain Gauge</td>
<td>Model #EA-06-031CF-120</td>
</tr>
<tr>
<td></td>
<td>Vishay Intertechnology</td>
</tr>
<tr>
<td>Accelerometer</td>
<td>PCB 353B04 &amp; 352C22</td>
</tr>
<tr>
<td>Vibrometer</td>
<td>Polytec OFV 503</td>
</tr>
</tbody>
</table>

Table 3-1 Testing equipment provided by ARL-MLS lab

3.5 Test Procedure

The proposed fatigue testing has several steps including test setup, calibration testing, the sine sweep test and the dwell test. These steps have been discussed comprehensively.

3.5.1 Test Setup

The first step to perform a high-cycle fatigue test is to set up the required equipment. A strain gauge is placed in the maximum stress spot of the blade which is determined before the testing by finite element analysis as the maximum strain level should be monitored for plotting the S-N curve. The specimen is attached to a fixture which is mounted on the shaker that simulates the fixed-free (cantilever) boundary condition. An accelerometer is attached to the fixture in order to evaluate and control the excitation force. The laser beam of the vibrometer, which is mounted on a tripod and placed at an optimum stand-off distance from the specimen, is pointed on the specimen as close as possible to its tip. The vibrometer measures the velocity or displacement.
response of the forced excitation for the specimen. A schematic for the testing setup has been shown in Figure 3-3. All sensors including the accelerometer, the strain gauge, and the vibrometer are connected to software through a data acquisition system to process the measured signals. The software is also connected to an amplifier, which provides power for the shaker, through a controller. As discussed, the software controls the frequency of excitation by changing the input signal of the amplifier using frequency tracking criterion. The shaker is bolted down to an optical table and supported with a cooling unit.
Figure 3-3 HCF testing configuration
3.5.2 **Calibration Testing**

Occasionally, at high g-level, strain gauges can break during the HCF fatigue test. As a result, a calibration test is required to establish the relationship between strain level, displacement, and excitation g-level by performing several sine sweep tests at different g-levels between 5g and 35g. FRF, strain level and corresponding g-levels are recorded in order to plot g-level versus strain and displacement graphs.

3.5.3 **Sine Sweep Testing**

To search for resonance frequency, a sine sweep test is required to be performed through a range of frequencies which includes the resonance mode of interest. In this study, first bending mode of the blade has been selected for the dwell test. The frequency range has been set between 100 Hz and 150 Hz due to the first bending mode for this specimen being within this frequency range. The g-level is determined with respect to the required g-level for the sine-dwell test.

3.5.4 **Sine Dwell Testing**

The specimen is excited by the shaker at resonance frequency which is determined by the sine sweep test in previous steps. The resonance frequency is tracked by phase dwell criterion until the blade cracks or 10 million cycles is reached. Blade cracking will result in a substantial shift in the resonance frequency because of the change in the structure stiffness after crack initiation. The test is stopped either by detecting crack on the surface or by observing a considerable shift in the resonance frequency. Consequently, the software has been set to stop excitation when the frequency shift exceeds 1.5%. The accelerometer, strain gauge and vibrometer are used throughout the test to enable the controller to track the resonance frequency and evaluate the strain level. As soon as the specimen cracks, number of cycles is recorded in order to use it for plotting the S-N curve.
3.6 Results

Three specimens have been tested to evaluate the fatigue behavior of the material. These specimens are axial-fan blades of an aircraft engine which were fabricated from aluminum. As the R67 is also an axial-fan and possesses relatively similar blade shape in compare to the tested blades, the fatigue behavior of both blades, which are fabricated from identical material, has been considered to be approximately the same.

3.6.1 Calibration Results

The calibration test was performed before the main high-cycle fatigue test to correlate the results obtained from different g-levels. The velocity and displacement of a point near to the tip was also measured to evaluate the magnitude of the vibration by the vibrometer. The g-level was increased from 5g to 35g. The values for measured velocity, displacement, and strain levels are related to the amplitude of these parameters at the fundamental frequency. As the amplitude of the base excitation (g-level) was increased, displacement, velocity, and strain value were increased with respect to the g-level. The correlation between the acceleration of the applied excitation in terms of g-level as the input and maximum strain level of the blade as the response has been presented in Figure 3-4. The calculated correlation coefficient is relatively close to 1 which implies that the g-level and strain levels have a linear correlation.
3.6.2 Dwell Test Results

The dwell tests were conducted at different g-levels. Specimens were excited at their resonance frequency at 10g, 13g, and 15g until the blade cracked. The resonance frequency of the blade was observed to a shift away from its initial position when the blade cracked. The results summary which is required for plotting the S-N curve has been presented in Table 3-2.

<table>
<thead>
<tr>
<th>g level</th>
<th>Displacement (mm)</th>
<th>Strain level (µ)</th>
<th>Number of Cycles (cracking)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>1.08</td>
<td>2157.08</td>
<td>9.14862×10^6</td>
</tr>
<tr>
<td>13</td>
<td>1.45</td>
<td>2395.6</td>
<td>1.299052×10^6</td>
</tr>
<tr>
<td>15</td>
<td>1.95</td>
<td>2714.64</td>
<td>0.345439×10^6</td>
</tr>
</tbody>
</table>

Table 3-2 Dwell test Results

Figure 3-5 represents the comparison of displacement spectrum before and after crack initiation when the excitation was applied at 10g. These spectrums were obtained by performing the sine sweep test before and after the crack initiation.
Figure 3-5 Displacement spectrum before and after crack obtained from sine sweep test at 10g

The resonance frequency of the specimen has been shifted due to the crack initiation in the blade. The crack initiation causes reduction in the stiffness of the blade which essentially results in a frequency shift and a higher level of the vibration as it shown in Figure 3-5.

The displacement spectrum before and after the crack initiation have been presented for excitation at 13g and 15g in Figure 3-6 and Figure 3-7, respectively.
Figure 3-6 Displacement spectrum before and after crack obtained from sine sweep test at 13g

Figure 3-7 Displacement spectrum before and after crack obtained from sine sweep test at 15g
The crack initiation results in a frequency shift, reduction in stiffness, higher level of the vibration similar to the sine sweep excitation at 10g as it can be observed from both Figure 3-6 and Figure 3-7.

### 3.6.3 S-N Curve

Based on the results of three sine-dwell tests at different g-levels, the S-N curve, which is the graph of the level of harmonic strain versus the logarithmic scale of cycles to failure, has been presented in the Figure 3-8.

![S-N Curve](image)

Figure 3-8 Strain vs the logarithmic scale of cycles to failure (S-N curve)

Three points shown in the S-N curve represent the results of the high-cycle fatigue tests using sine-dwell excitation at 10g, 13g, and 15g. Unlike the steel and titanium alloys, aluminum alloys do not exhibit well-determined endurance limits. These materials demonstrate a gradual decrease in strain level in the S-N curves as the number of the cycles to failure increases. However, an effective fatigue limit for these materials can occasionally be defined at the stress or strain which results in failure at $10^8$ or $5\times10^8$ cycles [39], [40]. An interesting investigation has been
performed by M. J. Caton [41] which demonstrates a well-determined endurance limit at super-
high-cycle fatigue regime ($10^9 \sim 10^{10}$) for an aluminum alloy using ultrasonic fatigue testing. In
this study, the effective fatigue limit has been approximately determined by extrapolating the S-
N curve up to $10^8$ cycles and applying a safety factor of two in order to cope with the variation of
fatigue behavior which may occur by changing the blade geometry or increasing the cycles. As
shown in the Figure 3-8, the extrapolation demonstrates how the applied sinusoidal strain with
amplitude of 1850 $\mu$-strain will result in failure at $10^8$ cycles. Therefore, the maximum allowable
strain (MAS) is calculated by

$$MAS = \frac{1850}{SF} = 925 \mu \text{strain}$$

(3.5)

where $SF = 2$ represents the safety factor. The maximum allowable strain has been imposed into
the optimization problem as the limit of the constraint relation.
Chapter 4  
Multidisciplinary Design Optimization

4.1 Overview

For solving any optimization problem, a mathematical algorithm is employed to alter the design variables and search for the optimum solution. Therefore, the first step for an optimization problem would be defining the problem with mathematical representation in order to make the problem feasible in correlation with the mathematical optimization algorithm. Objective functions, constraints functions, and design variables should be determined in terms of mathematical notations.

Accordingly, the mathematical problem definition is presented by

\[
\begin{align*}
\text{maximize} & \quad \eta = f(x_1,\ldots,x_n) \\
\text{with respect to} & \quad x_1,\ldots,x_n \\
\text{subject to} & \quad \varepsilon_{\text{max}} = C(x_1,\ldots,x_n) \leq MAS = 925 \mu
\end{align*}
\]

(4.1)

where \( \eta \) is the isentropic efficiency, \( \varepsilon_{\text{max}} \) is maximum strain in the operating system, MAS is the maximum allowable strain for the material, and \( x_1,\ldots,x_n \) are design variables.

A baseline design for the case study should be modeled in CAD format since the optimization algorithm utilizes the baseline as an initial point to reduce the number of evaluations and total computation time. In the R67 Geometry section, obtaining the NSAS Rotor 67 point cloud and modeling the geometry are discussed. For establishing a link between geometry modeling and optimizer, design variables should be determined by using a shape parameterization method which is presented in the Parameterization and Reverse Engineering sections.

For computing isentropic efficiency, a CFD analysis is required which is an essential part of this research due to its complexity. For CFD analysis, specific geometry should be modeled and appropriate boundary conditions should be applied which are presented in the CFD Analysis
section. In order to obtain both isentropic efficiency and pressure load, which is utilized for stress analysis, unprocessed data obtained from CFD simulation should be processed and transformed to the useable data. Subsequently, maximum stress and maximum strain are computed using finite element method (FEM) by applying pressure load and boundary conditions which are explained in the Structural Analysis section.

In MDO Architecture section, a multidisciplinary design optimization (MDO) architecture is described to illustrate the correlation between all disciplines including CFD analysis, structural analysis, geometrical parameterization, and the optimizer. MDO architecture demonstrates how the analyses and simulations couple with the optimizer to create an optimization loop which leads to an automated design process.
4.2 **R67 Geometry**

NASA Rotor 67 (R67) has been selected as a case study in this research. A survey for R67 design specification has been presented in the Literature Review chapter. A baseline design of R67 is required to be modeled for both optimization and simulation setups. Creating the baseline also reduces the computational time dramatically due to the optimizer initiating the design optimization process from a suitable point. NASA has released the geometry of R67 in the meridional and radius × angular coordinate system [30].

The NASA Lewis Research Center had undertaken to measure point cloud for R67 blade surface coordinate using an advanced laser anemometer (LA) system. Blade coordinates are published for surfaces of revolution for 14 blade sections (spanwise direction) where each blade section has 35 points (chordwise direction). Therefore, the blade geometry is determined by 14×35 points in meridional and radius × angular coordinate surface whose parameters have been defined in Table 4-1 and shown in Figure 4-1.

![Unwrapped conical surface](image)

**Figure 4-1 Meridional blade coordinate [42]**
<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z</td>
<td>Axial Coordinate</td>
</tr>
<tr>
<td>R</td>
<td>Radial Coordinate</td>
</tr>
<tr>
<td>THSP1</td>
<td>Angular Coordinate of suction surface</td>
</tr>
<tr>
<td>THSP2</td>
<td>Angular Coordinate of pressure surface</td>
</tr>
<tr>
<td>TNPC</td>
<td>Thickness normal to the mean camber line on the surface</td>
</tr>
</tbody>
</table>

Table 4-1 Blade Coordinate Parameters

These notations have been obtained from the NASA technical paper [30], noting that the blade is rotated from $R = 0$. In addition to the blade coordinates, the hub coordinates and the shroud coordinates, which are required for CFD analysis, have been released in the NASA technical paper [30].

All coordinates should be transformed from the meridional coordinate system into the Cartesian coordinate system due to most commercial CAD software supporting the Cartesian system. A set of formulations has been derived to transform blade coordinates into the Cartesian system which have been shown in the following formulations for suction surface. These formulations are obtained using transformation of reference [43] and applying adjustments for this case.

$$
\begin{align}
    x^1 &= (\rho + \text{TNPC} / 2) \sin \phi_0 \cos (\text{THSP1}) \\
    x^2 &= (\rho + \text{TNPC} / 2) \sin \phi_0 \sin (\text{THSP1}) \\
    x^3 &= Z - (\text{TNPC} / 2) \cos \phi_0 \\
\end{align}
$$

(4.2)

Similarly, for pressure surface, transformation yields to

$$
\begin{align}
    x^1 &= (\rho - \text{TNPC} / 2) \sin \phi_0 \cos (\text{THSP2}) \\
    x^2 &= (\rho - \text{TNPC} / 2) \sin \phi_0 \sin (\text{THSP2}) \\
    x^3 &= Z + (\text{TNPC} / 2) \cos \phi_0 \\
\end{align}
$$

(4.3)
where $\phi$ is determined using $\phi_0 = \tan^{-1}\left(\frac{R_o - R_i}{Z_o - Z_i}\right)$ formulation, $\rho$ is calculated by $\rho = \frac{R}{\sin \phi_0}$.

$R_o$ and $Z_o$ describe the blade coordinates for the trailing edge of the blade, $R_i$ and $Z_i$ describe blade coordinates for the leading edge of the blade. Other notations are the same as the blade coordinates parameters presented in Table 4-1.

The geometry of R67 has been obtained and the point cloud has been transformed to the Cartesian system which can be modeled in CAD software as a baseline design.

### 4.3 Parameterization

Design variables in an optimization problem should be determined in order to establish an explicit relation between the optimizer and the alteration in design. For this study, only the geometrical parameters have been allowed to change within the optimization process. Therefore, an appropriate shape parameterization is required to determine design variables and enable the optimizer to alter the geometry of the design. In the Literature Review chapter several shape parameterization methods have been presented. For this study, the polynomial and spline approach has been selected to parameterize the geometry of R67 using the non-uniform rational b-spline (NURBS) surface formulation due to its advantages such as accuracy, and compact data storage which have been mentioned earlier. The NURBS surface function can be formulated as

$$S(u, v) = \sum_{i=1}^{n} \sum_{j=1}^{m} R_{i,j}(u, v)P_{i,j}$$

(4.4)

where $S$ demonstrates surface coordinates, $P_{i,j}$ includes control point coordinates, and $R_{i,j}$ is calculated using B-spline basis function which has been presented in the Literature Review chapter.

The geometry of R67 has been parameterized by $3\times3$ control points where both chordwise and spanwise directions have been divided to 3 segments. By considering $u$ as the chordwise direction of the blade, $v$ as the spanwise direction, and applying $3\times3$ control points to the NURBS formulation the blade geometry has been calculated in terms of $u$ and $v$ which vary in the interval between 0 and 1 ($0 \leq u \leq 1, 0 \leq v \leq 1$).
Hence NASA has released R67 by 14×35 points, 35 equidistant numbers within the interval between 0 and 1 have been selected for $u$ in chordwise direction. Similarly, 14 numbers have been selected for $v$ in spanwise direction. Therefore, the exact number of points in comparison with NASA’s number of points is produced by plugging all combinations of $u$ and $v$ into the NURBS formulation.

As mentioned earlier, the aim of this study is shape optimization of the R67 blade component and not the whole axial fan. Thus, the parameterization of the R67 blade should be restricted in order to avoid creating a geometry that may not suitable for the original fan design. Consequently, all corners of the R67 blade, which are included in four of the control points as shown in Figure 4-2, have been fixed. As a result, the coordinates of only 5 control points from 9 control points have been permitted to change during the optimization process. Each control point has 5 coordinates, which are represented in meridional coordinates. Therefore, 25 design variables have been identified which have been indicated in Table 4-2.

Figure 4-2 Control points of the NURBS surface
Table 4-2 Notation of design variables

<table>
<thead>
<tr>
<th>$P_{ij}$</th>
<th>$Z$ (cm)</th>
<th>$R$ (cm)</th>
<th>THSP1 (rad)</th>
<th>THSP2 (rad)</th>
<th>TNPC (cm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1,2)</td>
<td>$x_1$</td>
<td>$x_2$</td>
<td>$x_3$</td>
<td>$x_4$</td>
<td>$x_5$</td>
</tr>
<tr>
<td>(2,1)</td>
<td>$x_6$</td>
<td>$x_7$</td>
<td>$x_8$</td>
<td>$x_9$</td>
<td>$x_{10}$</td>
</tr>
<tr>
<td>(2,2)</td>
<td>$x_{11}$</td>
<td>$x_{12}$</td>
<td>$x_{13}$</td>
<td>$x_{14}$</td>
<td>$x_{15}$</td>
</tr>
<tr>
<td>(2,3)</td>
<td>$x_{16}$</td>
<td>$x_{17}$</td>
<td>$x_{18}$</td>
<td>$x_{19}$</td>
<td>$x_{20}$</td>
</tr>
<tr>
<td>(3,2)</td>
<td>$x_{21}$</td>
<td>$x_{22}$</td>
<td>$x_{23}$</td>
<td>$x_{24}$</td>
<td>$x_{25}$</td>
</tr>
</tbody>
</table>

Although, the point cloud of R67 has been determined as a baseline design, the control points of the baseline have not been determined yet. As it is represented in the NURBS formulation, the geometry of R67 is a function of its control points. Thus, acquiring these control points is required in order to initiate the optimization from the baseline design. In the next section, an approach has been developed to determine control points and initial design variables.

### 4.4 Reverse Engineering

In the Parameterization section 9 control points are considered to define the parameterization of the R67 geometry. The four corners of the blade have been considered as 4 control points. To supplement the parameterization and determine all initial design variables, other 5 control points should be determined as well.

The geometry of the design is a function of control points in the NURBS formulation. Consequently, a reverse engineering approach should be performed in order to determine all control points from the geometry since the only available information is from NASA’s point cloud. An optimum NURBS curve fitting method has been discussed for reverse engineering in reference [44]. They performed an optimization to find the control points of an airfoil which has been parameterized by the NURBS curve formulation. Therefore, this concept has been taken from reference [44] to develop a reverse engineering approach for finding control points of R67 which is parameterized by the NURBS surface.
The process of reverse engineering has been presented in a flow chart in Figure 4-3. Some initial values should be guessed for coordinates of control points in order to start reverse engineering. Subsequently, the coordinates of geometry are calculated by applying initial control points into the NURBS formulation which is specified to generate $14 \times 35$ points. Through comparison of the generated points with original points of R67, the accuracy of initial values would be assessed. An optimization algorithm is utilized to update the coordinates of the control points with respect to the difference of the calculated coordinates by NURBS and original coordinates in order to minimize this difference.

Therefore, this reverse engineering method results in an optimization problem defined as

$$
\begin{align*}
\text{min} & \quad f(x_1, x_2, \ldots, x_{25}) = \sum_{k=0}^{35} \sum_{l=0}^{14} (S_{\text{NURBS}}(u_k, v_l) - S_{\text{R67}}(u_k, v_l))^2 \\
\text{with respect to} & \quad x_1, x_2, \ldots, x_{25} \\
\text{subject to} & \quad a_i \leq x_i \leq b_i
\end{align*}
$$

(4.5)

where $f$ is the objective function, which is a function of design variables and is defined in terms of summation of all $14 \times 35$ distances between calculated and original points, $x_1, \ldots, x_{25}$ are design variables which are defined in Table 4-2, $a_i$ and $b_i$ are the lower and upper bound for design variables, respectively.
As shown in Figure 4-3, three Matlab codes have been utilized to implement the reverse engineering method. A Matlab function has been written to calculate the NURBS surface formulation and generate the blade coordinates using control points coordinates as input. Another Matlab function has been written to evaluate objective function in formulation 4.5 by calculating the difference of the original coordinates and generated blade coordinates. An open source particle swarm optimization (PSO) algorithm [45] has been employed as an optimizer to update control points based on objective function evaluation. The optimizer continues updating the design variables and creates a loop process until optimum control points are found.
The optimization parameters, which have been used in reverse engineering, have been summarized in Table 4-3

<table>
<thead>
<tr>
<th>Reverse Eng. Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Control Points</td>
<td>3x3</td>
</tr>
<tr>
<td>Number of Design Variables</td>
<td>25</td>
</tr>
<tr>
<td>Number of Points generated by NURBS</td>
<td>35x14</td>
</tr>
<tr>
<td>Optimizer</td>
<td>PSO</td>
</tr>
<tr>
<td>Population Size</td>
<td>80</td>
</tr>
</tbody>
</table>

Table 4-3 Reverse engineering parameters

4.5 CFD Analysis

Computational Fluid Dynamics (CFD) analysis is required to simulate fluid flow within the turbomachinery design. CFD utilizes numerical methods and algorithms to analyze the interaction of liquids and gases with boundary conditions and solve the Navier-Stokes equations. In the design optimization of the turbomachinery blade, as the shape of the blade is redesigned throughout optimization iteration, the efficiency of the turbomachinery should be calculated for each design optimization loop using CFD analysis. Moreover, the surface pressure loads should be evaluated for fluid–structure interaction (FSI) analysis in order to apply these pressure loads in structural analysis. CFD analysis can be categorized into four tasks including geometry modeling, pre-processing, solving, and post-processing.

4.5.1 Geometry Modeling

The first step for any CFD analysis is defining physical bounds of the problem. The volume occupied by the fluid should be modeled to define the bounds for fluid simulation. One approach for an axial turbomachinery simulation is to analyze overall flow induced by a whole annulus passage. However, this approach is computationally expensive and complicated to implement in automated design optimization. In most cases, the flow can be assumed symmetric in axial direction which implies that the mean flow does not vary blade-to-blade [46]. Therefore, instead
of analyzing the whole axial fan, a single blade passage called cascade can be modeled to evaluate the properties of the flow and calculate the turbomachinery efficiency. The fluid domain is enclosed by its boundaries including solid walls, inlet, outlet and periodicity [47] which are illustrated in Figure 4-4.

![Figure 4-4 Boundaries of the single blade passage](Image)

- **Periodicity**

In single blade passage approach, two periodic boundaries should be defined to model the symmetry of flow between each circular sector of the cascade which is imposed by angular rotation around the axial fan calculated by $P = \frac{2\pi}{N}$ where $P$ is the central angle of the circular sector and $N$ is the number of blades of the whole fan stage.

- **Solid Walls**

Solid walls represent solid boundaries consisting of a hub, shroud, and blade in the cascade model. The blade geometry has been determined in the Parameterization section by $14 \times 35$
points. Blades rotate between two end-walls which are referred to as the hub and shroud. The hub and shroud design are modeled by original coordinates for R67 as they are fixed during the optimization. Furthermore, the blade is restricted to have 0.1016 cm tip clearance at shroud.

- **Outlet and Inlet**

The Inlet and outlet represent inflow and outflow of cascade, respectively.

### 4.5.2 Pre-processing

One approach for solving Navier-Stokes equations of three-dimensional flow is the finite volume method, which requires discretizing the spatial domain into finite volumes or grid cells by meshing. The fluid behavior and properties at each boundary condition should be defined and imposed at nodes before solving the equations.

#### 4.5.2.1 Meshing

Meshing is one of the most significant requirements of a CFD analysis as a coarse mesh can drastically affect the results and a highly fine mesh can dramatically increase the computational time. Traditionally, some control points which are modified manually, should be set to create a topology for meshing the blade passage appropriately. However, this method is hard to implement in an automated design optimization. Therefore, in this study, the automatic topology and meshing (ATM optimized) method, which automatically creates high-quality meshes with minimal effort, has been applied.

TruboGrid software has been employed to model the geometry and mesh the fluid domain by applying the ATM method. Mesh sensitivity analysis on the initial design should be performed in order to both avoid creating elements with a sharp edge angle and to find the optimum mesh size for blade passage which does not affect the results. The mesh refinement study suggests 200000 number of elements for the baseline. Therefore, the target passage mesh size during optimization has been set as 200000 number of elements. As the geometry changes during the optimization process, TurboGrid automatically creates and meshes the fluid domain with respect to variation of blade geometry. In addition, the software is prescribed to create recommended intermediate surface between hub and shroud to decrease mesh distortion.
4.5.2.2 Boundary Conditions

The fluid domain is enclosed by its boundaries which have been modeled and meshed in TurboGrid. The fluid behavior and properties at all boundaries should be determined in order to impose these boundary conditions into the Navier-stokes equations. Boundary condition types have been discussed in the Geometry Modeling section. The boundary conditions’ properties are mentioned here:

- **Outlet and Inlet Conditions**

The outlet and inlet are considered as the far-field boundary conditions which are imposed to the calculation of inviscid flux term $F^I_K$ in Navier-Stokes equations at the boundary node $K$ by solving the following equation

$$ F^I_K = \frac{1}{2}(F^I_K(U_K) + F^I_K(U_\infty) - |A_K|(U_\infty - U_K)) $$

(4.6)

where $U$ represents the vector of conservative flow variables which is fully defined in Appendix B, $A_K = \frac{\partial F^I_K}{\partial U_K}$, and $U_\infty$ describes the far-field state which can be determined in several ways. Commonly in literature [46], [47], [34], [27] total pressure, total temperature, and turbulence parameters are prescribed at the inlet and static pressure at the outlet in order to determine the far-field state in the axial turbomachinery blade passage.

- **Solid Walls Conditions**

There are three solid walls in the blade passage: the hub, shroud, and blade surfaces. Depending on the flow type and assumption around the solid wall, solid wall boundary condition can be classified into two groups: slip boundary condition, and no-slip boundary condition.

  ✓ **Slip Boundary Conditions**

Slip boundary condition is applied at a solid wall where the flow around the wall is considered as inviscid flow. In this condition the fluid can slip on the solid wall which implies that the flow direction remains tangential to the wall surface. Therefore, the dot product of fluid velocity...
vector $u_i$ and the normal vector to the wall surface $n_k$ at node $K$ belonging to solid wall must be zero as expressed by

$$u_i \cdot n_k = 0.$$ \hfill (4.7)

In this study, the flows around the shroud and hub have been assumed to be inviscid. Thus, the slip boundary condition has been imposed to the Navier-Stokes equations at nodes which belong to the shroud and hub surfaces.

✓ No-Slip Boundary Condition

Slip boundary condition is applied at a solid wall where the flow around the wall is considered viscous flow. In this condition, the fluid at the wall boundary is not permitted to slip on the solid wall. In other words, the fluid at the wall moves at the same velocity as the wall. Therefore, the fluid velocity vector $u_i$ is equal to the wall velocity vector $u_{Wall}^K$ at node $K$ belonging to the solid wall.

$$u_i = u_{Wall}^K.$$ \hfill (4.8)

In this study, the no-slip boundary condition has been imposed to the Navier-Stokes Equations at the blade surfaces.

• Periodicity Conditions

The single blade passage approach requires the evaluation of mean flow over the passage by imposing periodic boundary conditions. The single blade passage has two periodic boundaries. The boundary located at the lower angular coordinate $\theta$ is called the lower periodic boundary and the other boundary located at the angular coordinate $\theta + P$ is called the upper periodic boundary where $P$ is the central angle of the circular sector associated to the passage. $P$ is determined by the number of blades of the whole fan stage $N$ using $P = \frac{2\pi}{N}$. Since the flow is considered symmetric in axial direction and does not vary in blade passages, the flow parameters at node $K$ belonging to the lower periodic surface should be the same as flow parameters at the equivalent node $K'$ belonging to the upper periodic surface which can be expressed mathematically as
\[ U_K'(\theta + P) = U_K'(\theta) \]  

(4.9)

where \( U \) is the vector of conservative flow variables defined in Appendix B. Periodicity condition is imposed on the Navier-Stokes equations.

All boundary conditions have been imposed using CFX-Pre software by importing the mesh file of fluid domain from TurboGrid. The list of boundaries and their types is presented in Table 4-4. CFX-Pre is employed to provide the definition of the problem and its boundary conditions in order to enable the solver (CFX-Solve) to simulate the flow. The fluid is assumed to be air and exhibit ideal gas behavior.

<table>
<thead>
<tr>
<th>Boundary Name</th>
<th>Boundary Type</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inlet</td>
<td>Far-Field</td>
<td>( P_t = 101325 \text{ Pa} )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( T_t = 288.2 \text{ K} )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Subsonic</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Stationary</td>
</tr>
<tr>
<td>Outlet</td>
<td>Far-Field</td>
<td>( P_s = 114500 \text{ Pa} )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Subsonic</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rotating</td>
</tr>
<tr>
<td>Blade</td>
<td>Solid Wall</td>
<td>No-Slip (Viscous flow)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rotating</td>
</tr>
<tr>
<td>Hub</td>
<td>Solid Wall</td>
<td>Slip (Inviscid flow)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rotating</td>
</tr>
<tr>
<td>Shroud</td>
<td>Solid Wall</td>
<td>Slip (Inviscid flow)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rotating</td>
</tr>
<tr>
<td>Lower Periodic</td>
<td>Periodicity</td>
<td>Rotating</td>
</tr>
<tr>
<td>Upper Periodic</td>
<td>Periodicity</td>
<td>Rotating</td>
</tr>
</tbody>
</table>

Table 4-4 CFD boundary conditions, \( P_t \) and \( T_t \) represent the total pressure and total temperature, respectively and \( P_s \) represents static pressure.
4.5.3 Solving

Flow analysis requires solving three-dimensional Navier-Stokes equations which are generally derived from the conservation of mass, momentum, and energy for viscous flow. All Navier-Stokes equations can be regrouped into a single equation by integrating around the control volume. The steady-state form of the integrated Navier-Stokes equation can be obtained by eliminating time-dependent terms as [46]

\[
R(U) = \int_{V} C(U) dV - \oint_{S} F(U, \nabla U).n dS = 0
\]

(4.10)

where \(R\) is the vector of flow residual, \(C\) is the vector of centrifugal and Coriolis sources, \(F\) is the vector of convective and viscous fluid fluxes, \(V\) is the control volume of interest, \(S\) is the boundary surface, and \(U\) is the vector of conservative flow variables. All terms have been determined and defined in Appendix B.

The Navier-Stokes equation has an analytical solution only for some simple flows under ideal conditions. Therefore, a numerical method should be implemented by substituting algebraic approximations for the Navier-Stokes equation in order to solve the equation for real flows. In this study, CFX-Solve has been utilized to solve the Navier-Stokes equations numerically. CFX uses a finite volume method for solving equations. In the finite volume method, the spatial fluid domain is discretized into finite volumes and the Navier-Stokes is applied to each of these volumes. The Navier-Stokes equation results in the following formulation for each volume, which is assumed to have relatively small dimensions, by applying the numerical integration method.

\[
R_I = \frac{1}{V_I} \left[ v_I C_I - \sum_{j \in E_I} (F_{IJ}) n_{IJ} \Delta S_{IJ} \right] = 0
\]

(4.11)

Where \(v_I\) is the value of the control volume associated to node \(I\), \(E_I\) is the set of all nodes connected to node \(I\), \(R_I\) is the residual of node \(I\), \(C_I\) is the centrifugal and Coriolis term at node \(I\), \(\Delta S_{IJ}\) is the area of the surface associated with node \(I\) and \(J\), \(n_{IJ}\) is the normal of the
surface, and $F_{ij}$ represents the summation of the discretized inviscid and viscous fluxes which are defined in Appendix B.

CFX-Solve has been set to solve the problem by parallelization of calculations using four processor cores. The solution is obtained when the flow residuals converge to a threshold value. The solver control options have been shown in the Table 4-5.

<table>
<thead>
<tr>
<th>Control Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Residual Target</td>
<td>$10^{-4}$</td>
</tr>
<tr>
<td>Maximum Iterations</td>
<td>2500</td>
</tr>
<tr>
<td>Heat Transfer Model</td>
<td>Total Energy</td>
</tr>
<tr>
<td>Turbulence Model</td>
<td>Shear Stress Transport (SST)</td>
</tr>
<tr>
<td>Morphology</td>
<td>Continuous Fluid</td>
</tr>
</tbody>
</table>

Table 4-5 CFD solver Control Options

4.5.4 Post-Processing

After obtaining the solution of the Navier-Stokes equation, the results should be translated into useable data which is desired for the design optimization problem. Primarily, the surface pressure loads must be determined as an input for the finite element analysis in order to perform unidirectional fluid-structure interaction (FSI) and evaluate the maximum strain of the blade. On the other hand, the isentropic efficiency should be evaluated as the objective function of the optimization problem.

4.5.4.1 Pressure loads

In this study, the effect of the blade deformation on the flow analysis is considered to be negligible. Therefore, performing unidirectional FSI is adequate to couple the structural analysis with the flow analysis. The unidirectional FSI has been performed using CFD-Post software by importing the surface mesh of the blade created by FEM software. The software reads the mesh file and interpolates the pressure loads obtained from flow analysis results and maps them onto
the associated nodes. It is necessary to apply the imported mesh file into the related boundary surface.

### 4.5.4.2 Isentropic Efficiency

The efficiency of the turbomachinery must be calculated to evaluate the objective function of the optimization problem. Isentropic efficiency indicates the degree of degradation of energy in turbomachinery devices. The actual performance of the turbomachinery is compared with the idealized performance which is obtained under an isentropic process using the same boundary conditions and fluid state. The minimum required work for operating an adiabatic compressor would be achieved by the isentropic process due to this process involving change in fluid state without entropy variation. Practically, the compressor is operated by greater work than the isentropic work as shown in Figure 4-5. Therefore, the ratio of isentropic work to the actual work is defined as the isentropic efficiency of compressors.

\[
\eta = \frac{\text{Isentropic work}}{\text{Actual work}} \approx \frac{h_{2a} - h_1}{h_{2a} - h_1}
\]  

(4.12)

Where \( \eta \) represents the isentropic efficiency, \( h_1 \) is the enthalpy of the system at the inlet, \( h_{2a} \) is the actual enthalpy at the outlet, and \( h_{2s} \) represents the enthalpy at outlet which would be obtained by the isentropic process. By considering Polytropic relations of the ideal gas, the isentropic efficiency can be simplified into the following formulation.

\[
\eta = \frac{\left(\frac{P_{2t}}{P_{1t}}\right)^{\gamma-1} - 1}{\frac{T_{2t}}{T_{1t}} - 1}
\]  

(4.13)

Where \( \eta \) is the isentropic efficiency, \( P_{2t} \) and \( P_{1t} \) represent total pressure at the outlet and inlet, respectively, \( T_{2t} \) and \( T_{1t} \) is total temperature at the outlet and inlet, respectively, and \( \gamma \) is the heat capacity ratio of the fluid.
Evaluation of both pressure loads and isentropic efficiency has been performed by CFD-Post using the solution of the Navier-Stokes computed by CFX-Solve.

4.6 Structural Analysis

As discussed earlier, the optimization problem has been subjected to a constraint which implies that the maximum strain of the blade in operating condition must not exceed the maximum allowable strain. The maximum allowable strain has been determined in the High Cycle Fatigue Testing chapter. Therefore, as the design is updated during the optimization process, the maximum strain of the design should be calculated in each optimization loop to evaluate the constraint of the optimization.

The solid geometry of the blade has been modeled using SolidWorks commercial software by importing the point cloud which is determined by $14\times35$ points coordinates. ANSYS APDL has been utilized as a finite element solver in this study. The CAD file which includes the solid geometry of the blade has been imported to the ANSYS APDL. The blade geometry has been discretized into a finite number of elements using Solid187 element type. The solid elements have been overlaid with the SURF154 element type which enables the software to import the surface loads from the CFD analysis and perform FSI analysis [48]. The hub surface of the blade is fixed and pressure loads are applied onto the pressure and suction sides. The material properties of the blade (Aluminum) and other solver parameters have been defined as presented in Table 4-6.
### Table 4-6 FEM solver parameters

<table>
<thead>
<tr>
<th>Solver Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Young’s Modulus</td>
<td>$70 \times 10^9 \dfrac{N}{m^2}$</td>
</tr>
<tr>
<td>Poisson's Ratio</td>
<td>0.33</td>
</tr>
<tr>
<td>Density</td>
<td>$2700 \dfrac{kg}{m^3}$</td>
</tr>
<tr>
<td>Elements</td>
<td>Solid187, SURF154</td>
</tr>
<tr>
<td>Solver approach</td>
<td>Small displacement static</td>
</tr>
</tbody>
</table>

The software solves the structural equation as presented in the following formulation by applying boundary conditions, nodal forces $\{F\}$, calculating the global stiffness matrix $[K]$, and the consistent mass matrix $[M]$ in order to obtain the vector of nodal displacement $\{u\}$ and nodal acceleration $\{\ddot{u}\}$

$$[M]\{\ddot{u}\} + [K]\{u\} = \{F\}$$  \hspace{1cm} (4.14)

The vector of the nodal strain $\{\varepsilon\}$ and the vector of the nodal stress $\{\sigma\}$ can be calculated using the nodal displacements solution obtained from formulation 4.14 and two following relations

$$\{\varepsilon\} = [B]\{u\}$$  \hspace{1cm} (4.15)

$$\{\sigma\} = [D]\{\varepsilon\}$$  \hspace{1cm} (4.16)

where $[B]$ represents the strain-displacement matrix computed using the shape function of the element, and $[D]$ denotes the stress-to-strain matrix obtained by generalizing Hooke’s law in three dimensions.
4.7 MDO Architecture

A multidisciplinary design optimization (MDO) problem involves optimization of a design with multiple disciplines such as fluid analysis, structural analysis, and heat transfer. Organizing all disciplines appropriately in cooperation with the optimizer and the problem formulation is significant in MDO implementation. Therefore, a combination of problem definition and organizational strategy for utilizing each discipline in an optimization algorithm, which is referred to as the MDO architecture, is required to specify how the different disciplines are coupled and how the optimizer controls the process. The MDO architecture for this case has been presented in Figure 4-6.

As it is shown, the analysis part consists of the structural analysis coupled with CFD analysis. Optimizer updates the values of the design variables with respect to the evaluation of the objective function and constraint until the optimum design is achieved. The new design of the blade is mathematically defined by the NURBS calculation using control points associated with design variables as determined in the Parameterization section. The solid geometry of the blade is modeled in SolidWorks for structural analysis purposes and the geometry of the fluid domain is modeled in TurboGrid for fluid analysis. The pressure loads are transferred to the structural solver after the CFD analysis is performed. A fitness value is computed with respect to the analysis results in order to use the evaluated objective function and constraint for updating the design variables which is controlled by the optimizer. This process is iterated to construct an automated multidisciplinary design optimization.
Figure 4-6 Flowchart of the automated MDO architecture

All components of the MDO structure have been discussed earlier in terms of theory. Here, some computer programming concerns have been discussed in order to build the automated design optimization process.
4.7.1 Optimizer (PSO)

The optimizer controls the optimization process by providing the design variables \((x_1, x_2, \ldots, x_{25})\) as input for the NURBS calculation. The design variables are updated at each iteration with respect to the fitness function in order to search for the optimum design. The particle swarm optimization (PSO) algorithm has been utilized as the optimizer in this study due to its efficiency. An open source PSO toolbox written in the Matlab language has been selected from an online source [45] to integrate with the computerized design analysis. The PSO toolbox uses the evaluated fitness function as input and calculates the new design variables as the output. The parameters which have been used for PSO algorithm are presented in Table 4-7. The optimization has been started by initial design variables which have been calculated in the Reverse Engineering section. The lower and upper bound for design variables have been set as indicated in Table 4-8.

<table>
<thead>
<tr>
<th>PSO Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Design Variables</td>
<td>25</td>
</tr>
<tr>
<td>Population Size</td>
<td>35</td>
</tr>
<tr>
<td>Generations</td>
<td>500</td>
</tr>
<tr>
<td>Cognitive Coefficient</td>
<td>2</td>
</tr>
<tr>
<td>Social Coefficient</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 4-7 PSO parameters
Table 4-8 Design variables bounds

<table>
<thead>
<tr>
<th>( P_{i,j} )</th>
<th>( Z ) (cm)</th>
<th>( R ) (cm)</th>
<th>THSP1 (rad)</th>
<th>THSP2 (rad)</th>
<th>TNPC (cm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1,2)</td>
<td>0 ≤ ( x_1 ) ≤ 9</td>
<td>9.5 ≤ ( x_2 ) ≤ 11.8</td>
<td>0 ≤ ( x_3 ) ≤ 1</td>
<td>0 ≤ ( x_4 ) ≤ 0.7</td>
<td>0.15 ≤ ( x_5 ) ≤ 2</td>
</tr>
<tr>
<td>(2,1)</td>
<td>0 ≤ ( x_6 ) ≤ 2.48</td>
<td>12 ≤ ( x_7 ) ≤ 20</td>
<td>−0.2 ≤ ( x_8 ) ≤ 0.2</td>
<td>−0.2 ≤ ( x_9 ) ≤ 0.2</td>
<td>0.02 ≤ ( x_{10} ) ≤ 0.1</td>
</tr>
<tr>
<td>(2,2)</td>
<td>1 ≤ ( x_{11} ) ≤ 7.8</td>
<td>12 ≤ ( x_{12} ) ≤ 20</td>
<td>0 ≤ ( x_{13} ) ≤ 0.5</td>
<td>0 ≤ ( x_{14} ) ≤ 0.5</td>
<td>0.02 ≤ ( x_{15} ) ≤ 1</td>
</tr>
<tr>
<td>(2,3)</td>
<td>0 ≤ ( x_{16} ) ≤ 9</td>
<td>12 ≤ ( x_{17} ) ≤ 20</td>
<td>0 ≤ ( x_{18} ) ≤ 0.5</td>
<td>0 ≤ ( x_{19} ) ≤ 0.5</td>
<td>0.02 ≤ ( x_{20} ) ≤ 0.1</td>
</tr>
<tr>
<td>(3,2)</td>
<td>2.5 ≤ ( x_{21} ) ≤ 6.5</td>
<td>24.8 ≤ ( x_{22} ) ≤ 25.5</td>
<td>0 ≤ ( x_{23} ) ≤ 0.6</td>
<td>0 ≤ ( x_{24} ) ≤ 0.6</td>
<td>0.1 ≤ ( x_{25} ) ≤ 0.8</td>
</tr>
</tbody>
</table>

4.7.2 Geometrical Parameterization

Geometrical parameterization generates a point cloud of the blade associated with updated design variables. An open source Matlab code obtained from the reference [49] has been modified in order to deal with the NURBS calculation using defined control points. This Matlab function code is fed by design variables as coordinates of the control points and generates 14×35 points in meridional coordinates. Another Matlab code has been written to transform the generated point cloud into the Cartesian coordinate system. The transformation code stores the coordinates of points in several text files where their formats are adjusted to be importable to the analysis software. These text files can generally be categorized into two groups. One group is used for modeling the fluid domain in TurboGrid, and the other one for modeling the solid geometry of the blade in SolidWorks.

4.7.3 CFD Configuration

As mentioned in the CFD analysis section, the fluid analysis involves four tasks: Modeling, Pre-processing, Solving, and Post-processing.

- TurboGrid

TurboGrid deals with fluid domain modeling by using the single blade passage method. Also, the fluid domain is discretized into a finite number of volumes by the software. TurboGrid uses the
CFX Command Language (CCL is the internal communication and command language of ANSYS TurboGrid and other CFX software packages). CCL supports object and parameter definition, command actions which perform a specific task such as meshing, and power syntax which provides programming with loops and logic macros. A CCL code has been written to model the blade passage by importing three text files that contain the blade coordinates, the hub coordinates, and the shroud coordinates. Subsequently, the fluid domain is discretized using the Automatic Topology and Meshing (ATM optimized) command and the mesh file is stored in a specific file in order for use in the next step.

- **CFX-Pre**

Another CCL code has been written for the pre-processing part of the fluid analysis. This code deals with imposing boundary conditions and specifying the fluid state and properties. The mesh file created by TurboGrid, is imported to the CFX-Pre and a definition file which consists of the determination of the problem is created in order to import to the solver.

- **CFX-Solve**

The CFX-Solve only needs the definition file to solve the equation and perform the analysis. Thus, the solving part does not require an individual code. The solver is run by invoking an Operating System (OS) command in Matlab using batch mode and the definition file. After solving the problem, the solution is stored in a result file.

- **CFD-Post**

CFD-Post calculates the required parameters such as the isentropic efficiency and the pressure loads. CFD-Post can be run by a CFX Expression Language (CEL) code without graphical interface. A CEL code has been written to evaluate the efficiency and export the pressure load for the structural analysis. The resulting file is imported to the CFD-Post and the surface mesh created by structural software is applied to the associated boundary condition. Finally, the efficiency and pressure loads are saved in separated text files.
4.7.4 FEM Configuration

The structural analysis involves solid modeling of the blade, meshing, applying the loads and boundary conditions. A visual basic code has been written to model the solid geometry of the blade by 14×35 points using SolidWorks. This code reads the text file which contains the coordinates of points generated by NURBS in the software. By creating spline curves and lofting these curves the solid model is obtained and saved in a CAD file. Two ANSYS Parametric Design Language (APDL) codes have been written for structural analysis. The first one is used for meshing the solid model and exporting the mesh into a Mechanical Coded Database (CDB) file in order to perform the FSI analysis. The second one deals with reading the pressure loads evaluated by CFD, applying boundary conditions and solving the finite element equations. The maximum strain of the blade is calculated and stored in a text file subsequently.

4.7.5 Fitness Function Evaluation

A Matlab code has been written to run each associated software or codes by invoking them at an appropriate time. The external software is invoked by using Operating System (OS) commands. This code also computes the fitness value of the optimization which is required for the optimizer in each iteration. The fitness value is computed using the quadratic penalty function method which appends the constraint value to the objective function value by

\[
\text{Fitness} = \frac{10}{\eta} + 100(pf - 1)^2
\]

\[
pf = \begin{cases} 
1 & \text{if } \varepsilon_{\text{max}} \leq \text{MAS} \\
\frac{\varepsilon_{\text{max}}}{\text{MAS}} & \text{if } \varepsilon_{\text{max}} > \text{MAS}
\end{cases}
\]

(4.17)

where \( \eta \) is isentropic efficiency calculated by fluid analysis, \( \varepsilon_{\text{max}} \) is maximum strain of the blade computed by structural analysis, \( \text{MAS} \) is the maximum allowable strain obtained from testing and \( pf \) represents the penalty function coefficient.
4.8 Results

4.8.1 Initial Design

The unknown coordinates of control points have been determined using the proposed reverse engineering method. The design variables for R67 have been presented in Table 4-9.

<table>
<thead>
<tr>
<th>P_{i,j}</th>
<th>Z (cm)</th>
<th>R (cm)</th>
<th>THSP1 (rad)</th>
<th>THSP2 (rad)</th>
<th>TNPC (cm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1,2)</td>
<td>x_1 = 4.9189</td>
<td>x_2 = 11.1193</td>
<td>x_3 = 0.4182</td>
<td>x_4 = 0.2713</td>
<td>x_5 = 1.6776</td>
</tr>
<tr>
<td>(2,1)</td>
<td>x_6 = 1.1893</td>
<td>x_7 = 17.4920</td>
<td>x_8 = -0.0290</td>
<td>x_9 = -0.0370</td>
<td>x_{10} = 0.0544</td>
</tr>
<tr>
<td>(2,2)</td>
<td>x_{11} = 4.6944</td>
<td>x_{12} = 18.1774</td>
<td>x_{13} = 0.2054</td>
<td>x_{14} = 0.1567</td>
<td>x_{15} = 0.8290</td>
</tr>
<tr>
<td>(2,3)</td>
<td>x_{16} = 7.7809</td>
<td>x_{17} = 18.3087</td>
<td>x_{18} = 0.4195</td>
<td>x_{19} = 0.4183</td>
<td>x_{20} = 0.0839</td>
</tr>
<tr>
<td>(3,2)</td>
<td>x_{21} = 4.3336</td>
<td>x_{22} = 25.2811</td>
<td>x_{23} = 0.2038</td>
<td>x_{24} = 0.1602</td>
<td>x_{25} = 0.4376</td>
</tr>
</tbody>
</table>

Table 4-9 Design variables associated with control points of the R67 baseline.

The calculated objective function (formulation 4.5) at each iteration has been used as the fitness function of the optimizer. The fitness value converged to an optimum point which was found relatively small as shown in Figure 4-7. The fitness value (formulation 4.5) represents the summation of differences between calculated geometry and original geometry.
The converged fitness value demonstrates 1.5% averaged difference at each point between the original R67 blade coordinates and the baseline which was obtained from proposed reversed engineering. As the error value is relatively small at each point, these calculated control points can be utilized as the initial design variables of the main optimization problem. By setting initial values for design variables, the computation time of the main optimization was reduced drastically.

4.8.2 Optimum Design

Based on the initial design variables obtained from the proposed reverse engineering, the optimum design of the blade was determined using the proposed multidisciplinary design optimization (MDO) architecture according to the defined fitness function. The design variables associated with the optimum design have been presented in Table 4-10.
The design optimization was based on the maximization of the isentropic efficiency while the maximum strain of the blade must not exceed the experimented maximum allowable strain. The evaluated objective function (isentropic efficiency) and evaluated constraint (maximum strain) for both the optimized design and initial design have been presented in Table 4-11. These parameters compare the performance of the R67 baseline with the optimized design. The isentropic efficiency has been significantly improved 1.9% from baseline while the maximum strain has remained below the maximum allowable strain. The maximum strain of the optimized blade was evaluated relatively close to the constraint limit. Improvement in the efficiency results in higher performance and saving energy and maintaining the maximum strain below the maximum allowable strain ensures the high-cycle fatigue life for the blade and improves the safety of the design.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>MDO</th>
<th>Baseline</th>
<th>Abate [34]</th>
<th>NASA [27]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Isentropic Efficiency</td>
<td>0.9088</td>
<td>0.8919</td>
<td>0.8917</td>
<td>0.9189</td>
</tr>
<tr>
<td>Max Strain (µ-strain)</td>
<td>909.6</td>
<td>505.9</td>
<td>--------</td>
<td>--------</td>
</tr>
</tbody>
</table>

Table 4-11 Design evaluation of R67
Table 4-11 also shows the isentropic efficiency of the R67 evaluated by other researchers. NASA evaluated the isentropic efficiency of R67 using their own code and imposing non-uniform total pressure at the inlet as the boundary condition, along with non-uniform static pressure at the outlet as the boundary condition based on the experimental results [27]. Abate [34] imposed the averaged uniform total pressure at the inlet and averaged uniform static pressure at the outlet using CFX solver similar to this study. Therefore the evaluated isentropic efficiency for R67 baseline is approximately the same as the Abate evaluated. However, the efficiency shows relatively small difference in comparison with NASA’s evaluation since their imposed boundary conditions were non-uniform which yields a higher efficiency. The experimented isentropic efficiency has been presented in Figure 4-8 obtained from NASA’s measurements.

![Figure 4-8](image)

**Figure 4-8** Experimented efficiency of the R67 obtained from [30].

As shown in Figure 4-8 the computed efficiency is in the range of the experimental results. Therefore, the experimental measurements can validate the computed efficiency. These experiments were conducted in different total pressures at the inlet and static pressures at the outlet (different boundary conditions), while in this study the averaged pressure was imposed at the outlet and inlet. Therefore, the difference between NASA’s experimental results and the computed isentropic efficiency is reasonable.

The total pressure contour of the optimized design for both pressure and suction side have been presented in Figure 4-9 and Figure 4-10.
Figure 4-9 Total pressure contour of the pressure side

Figure 4-10 Total pressure contour of the suction side
Chapter 5
Conclusion and Future Work

5.1 Concluding Remarks

An automated multidisciplinary design optimization approach was developed in order to optimize the efficiency of a turbomachinery blade and maintain the maximum strain of the blade below maximum allowable strain. A vibration based HCF testing was conducted to investigate the fatigue behavior of the fan blade in high-cycle regime and evaluate the fatigue limit. The NASA Rotor 67 is a transonic axial fan of an aircraft engine which was used as a case study to evaluate the proposed design optimization methodology. The following remarks conclude the outcome of this research:

- High-cycle fatigue is the most significant cause of blade failure in aircraft engines. As the geometrical effects can influence the fatigue behavior, the most reliable method to investigate the fatigue behavior of the blade is to perform a high-cycle fatigue testing using a specimen with similar shape to the design shape.

- The sine-dwell testing accelerates fatigue failure by applying the base excitation at the resonance frequency of the part. In order to obtain the maximum cumulative fatigue, the resonance frequency of the part should be tracked and the excitation frequency should be updated to the instantaneous resonant frequency using a computerized feedback control system.

- The sine sweep test evaluates the resonance frequency and the magnitude of the vibration of an excited part. The experimental results represent a linear correlation between the level of the base excitation and the response of the part as the theoretical foundations indicate.

- Crack initiation results in stiffness reduction which causes higher vibration levels as well as a decrease in the resonance frequency as the sine-dwell experimental results represent.
Aluminum alloys do not exhibit well-determined endurance limits. An effective fatigue limit can be estimated by applying a safety factor to the strain level of the S-N curve at $10^8$ cycles.

Since turbomachinery design involves multiple analyses, traditional optimization methods are computationally expensive especially for a transonic fan design which involves complicated CFD analysis coupled with structural analysis. As a result, an automated MDO architecture is developed to reduce the computational time and integrate the design process into an optimization algorithm.

The NURBS surface can be employed in shape parameterization. It is beneficial to use the NURBS surface method as it drastically reduces the number of design variables of the shape optimization problem which causes a substantial reduction in computation time.

Initiating optimization from a baseline also significantly saves the computational cost. In the case that the control points of NURBS surface for the baseline design are unknown, a reverse engineering method should be employed to determine the control points through guessing initial control points and updating them until the closest shape with respect to the original shape is obtained.

For an axial turbomachinery, the flow can be assumed symmetric in axial direction. Therefore, a single blade passage can be modeled to simulate the flow instead of analyzing the overall flow induced by the entire annulus passage.

Preforming the FSI analysis is required to couple the structural analysis with the CFD analysis and apply pressure loads of the turbomachinery blade design. Mapping the nodes associated with structural mesh onto the CFD mesh is a critical part of the FSI analysis.

The optimized R67 demonstrates approximately 1.9% improvement in the isentropic efficiency while its maximum strain is evaluated below the maximum allowable strain determined by high-cycle fatigue testing.
5.2 Future Directions

In this study the R67 is parameterized by 3×3 control points using NURBS function. As the blade has complex shape, parameterizing the blade by more control points such as 4×4 or 5×5 is highly recommended. This design problem can be used as a case study for comparing the optimization algorithms since most comparative studies on the optimization algorithms have evaluated these algorithms by applying simple case studies such as optimization of truss-bar structures or mathematical functions. Therefore, a comparative study is needed to evaluate the performance of the most famous algorithms such as GA, PSO and EAs using a complicated case study such as the R67 optimization. In order to calculate more accurate results for flow analysis, a specialized code should be employed to simulate the flow around the turbomachinery by applying non-uniform boundary conditions. The TCGRID and SWIFT codes have been written in FORTRAN by the NASA Glenn Research Center for the analysis of flow around turbomachinery (available to universities within the United States). An alternative is to write a code for solving three-dimensional Navier-Stokes equations of flows around turbomachinery and integrate it with the optimization codes. Since the fatigue behavior shows variations in the fatigue testing, the reliability analysis can be performed on this case study in order to assess the reliability-based design optimization (RBDO) methods.
References


Appendix A: Navier-Stokes Equations Applied to Turbomachinery

A.1: Navier-Stokes Equations

General form of the Navier-Stokes equation can be expressed as [46]

$$\frac{d}{dt} \int_{V(t)} U dV + \oint_{S(t)} F(U, \nabla U).ndS = \int_{V(t)} C(U)dV + \oint_{S(t)} (Uu_p).ndS$$

The steady-state form of the equation is given by

$$R(U) = \int_{V} C(U)dV - \oint_{S} F(U, \nabla U).ndS = 0$$

Where $R$ is the vector of flow residual, $C$ is the vector of centrifugal and Coriolis sources, $F$ is the vector of convective and viscous fluid fluxes, $V$ is the control volume of interest, $S$ is the boundary surface, $U$ is the vector of conservative flow variables given by $U = (\rho, \rho u, \rho v, \rho w, \rho E)^T$, $\rho$ is the density, $u, v, w$ are the three Cartesian components of the velocity, and $E$ is the total energy per unit mass.

The vector of centrifugal and Coriolis sources for a turbomachinery case can be expressed as

$$C = (0, 0, \rho(\Omega^2 y + 2\Omega w), \rho(\Omega^2 z - 2\Omega w), 0)^T$$

Where $\Omega$ is the rotational speed of turbomachinery, $y$ and $z$ are Cartesian coordinates.

The vector of the convective and viscous fluid fluxes can be decomposed as

$$F(U, \nabla U) = F^I(U) + F^V(U, \nabla U)$$
Both the inviscid flux and viscous flux has terms from three Cartesian directions as

\[
F^i_x = \begin{pmatrix}
\rho u \\
\rho u^2 + p \\
\rho uv \\
\rho w \\
(\rho E + p) u
\end{pmatrix} \quad F^i_y = \begin{pmatrix}
\rho v \\
\rho uv \\
\rho v^2 + p \\
\rho vw \\
(\rho E + p)v
\end{pmatrix} \quad F^i_z = \begin{pmatrix}
\rho w \\
\rho uw \\
\rho vw \\
\rho w^2 + p \\
(\rho E + p)w
\end{pmatrix}
\]

\[
F^v_x = \begin{pmatrix}
0 \\
-\tau_{xx} \\
-\tau_{yx} \\
-\tau_{zx} \\
-\tau_{xz} - u\tau_{xx} - v\tau_{yx} - w\tau_{zx} + q_x
\end{pmatrix} \quad F^v_y = \begin{pmatrix}
0 \\
-\tau_{xy} \\
-\tau_{yy} \\
-\tau_{zy} \\
-\tau_{yz} - u\tau_{xy} - v\tau_{yy} - w\tau_{zy} + q_y
\end{pmatrix} \quad F^v_z = \begin{pmatrix}
0 \\
-\tau_{xz} \\
-\tau_{yz} \\
-\tau_{zz} \\
-\tau_{xz} - u\tau_{xz} - v\tau_{yz} - w\tau_{zz} + q_z
\end{pmatrix}
\]

Where

\[
\tau_{xx} = 2\mu \frac{\partial u}{\partial x} + \lambda(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}) \\
\tau_{xy} = 2\mu \frac{\partial v}{\partial y} + \lambda(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}) \\
\tau_{xz} = 2\mu \frac{\partial w}{\partial z} + \lambda(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z})
\]

And the heat fluxes are:

\[
q_x = -k_r \frac{\partial T}{\partial x}; q_y = -k_r \frac{\partial T}{\partial y}; q_z = -k_r \frac{\partial T}{\partial z}
\]

Where \( k_r = \frac{\mu c_p}{pr} \) is the coefficient of thermal conductivity, \( c_p \) specific heat, \( \mu \) is the viscosity, and \( pr \) is Parndtl number.
A.2: Finite Volume Method

In the finite volume method, the spatial fluid domain is discretized into finite volumes and the Navier-Stokes is applied to each of these volumes. The Navier-Stokes equation results in the following formulation for each volume, which is assumed to have small dimensions relatively, by using numerical integration method

\[ R_I = \frac{1}{V_I} \left[ \nu_I C_I - \sum_{J \in E_I} (F^I_{IJ} + F^V_{IJ}) n_{IJ} \Delta S_{IJ} \right] = 0 \]

Where \( \nu_I \) is the value of the control volume associated to node \( I \), \( E_I \) is the set of all nodes connected to node \( I \), \( R_I \) is the residual of node \( I \), \( C_I \) is the centrifugal and Coriolis term at node \( I \), \( \Delta S_{IJ} \) is the area of the surface associated with node \( I \) and \( J \), \( n_{IJ} \) is the normal of the surface, \( F^I_{IJ} \) and \( F^V_{IJ} \) represent the discretized inviscid and viscous fluxes, respectively.
Appendix B: Matlab Code

Fitness Function Evaluation: This Matlab Code is the core of the MDO architecture. It is responsible for calculating the Fitness value by invoking NURBS codes, CFX codes, a VB code (SolidWorks), and APDL codes.

```matlab
function pf  = fitness_func( CPs )
% CPs: control points coordinates (design variables).
% pf : fitness function for minimizing
%CPs=evalin('base','CPs');     % for standalone function
currentFolder = pwd;
addpath(genpath(currentFolder));    % considering subfolder
npts=3;    % number of control point in u direction
mpts=3;     %number of control point in w direction
k=3;         % order of spline in u direction
m=3;         %order of spline in w direction
p1=14;       % number of point will be created in u direction
p2=35;       %number of point will be created in w direction
b=zeros(3,npts,mpts);
c=zeros(3,npts,mpts);
Zaxial=evalin('base','Zaxial');
r=evalin('base','r');
THSP1=evalin('base','THSP1');
THSP2=evalin('base','THSP2');
TNPC=evalin('base','TNPC');

b(:,1,1)=[Zaxial(1,1);r(1,1);THSP1(1,1)];%1-1
b(:,1,2)=[CPs(1,1);CPs(1,2);CPs(1,3)];%17-1
b(:,1,3)=[Zaxial(35,1);r(35,1);THSP1(35,1)];%35-1
b(:,2,1)=[CPs(1,6);CPs(1,7);CPs(1,8)];%1-7
b(:,2,2)=[CPs(1,11);CPs(1,12);CPs(1,13)];%17-7
b(:,2,3)=[CPs(1,16);CPs(1,17);CPs(1,18)];%35-7
b(:,3,1)=[Zaxial(1,14);r(1,14);THSP1(1,14)];%1-14
b(:,3,2)=[CPs(1,21);CPs(1,22);CPs(1,23)];%17-14
b(:,3,3)=[Zaxial(35,14);r(35,14);THSP1(35,14)];%35-14

c(:,1,1)=[THSP2(1,1);TNPC(1,1)];
c(:,1,2)=[CPs(1,4);CPs(1,5)];
c(:,1,3)=[THSP2(35,1);TNPC(35,1)];
c(:,2,1)=[CPs(1,9);CPs(1,10)];
c(:,2,2)=[CPs(1,14);CPs(1,15)];
c(:,2,3)=[CPs(1,19);CPs(1,20)];
c(:,3,1)=[THSP2(1,14);TNPC(1,14)];
c(:,3,2)=[CPs(1,24);CPs(1,25)];
c(:,3,3)=[THSP2(35,14);TNPC(35,14)];

knot={[0 0 1 1 1] [0 0 1 1 1]};
surf1=nrbmak(b,knot);
surf2=nrbmak(c,knot);
ute=linspace(0.0,1.0,p1);
vt=linspace(0.0,1.0,p2);
```
pp1 = nrbeval(surf1,{ut vt});
pp2 = nrbeval(surf2,{ut vt});
for i=1:p1
    for j=1:p2
        Zaxial_nu(j,i)=pp1(1,i,j);
        r_nu(j,i)=pp1(2,i,j);
        THSP1_nu(j,i)=pp1(3,i,j);
        THSP2_nu(j,i)=pp2(1,i,j);
        TNPC_nu(j,i)=pp2(2,i,j);
    end
end

%Transform Coordinates
noimp  = change_coordinate( Zaxial_nu, r_nu, THSP1_nu, THSP2_nu, TNPC_nu );

%SolidWorks
cd ./solid
status1 = system('start sld.bat');
cd ..

%TurboGrid
! turbogrid.bat
if exist('cfxtg_error.log')==2
    datafile = fopen('data.dat','w+');
    fprintf(datafile,'%f',0.1);
    fclose(datafile);
    pf=0;
    delete('cfxtg_error.log');
else
    cd ./solid
    %Structural meshing (FSI)
    status1 = system('start ansys_esurf.bat');
    cd ..
    %Pre-CFX
    ! pre.bat
    %Solver-CFX
    ! solve.bat
    %Post-CFD
    ! post.bat
    %Structural analysis
    ! ./solid/ansys_analysis.bat
datafile = fopen('data.dat','r');
data = textscan(datafile,'%f %f %f %f %f ');
pf = data{1};
fclose(datafile);
fid = fopen('./solid/element-strain.txt');
read_ansys = textscan(fid,'%s','Delimiter','
');
sizeline = size(read_ansys{1,1},1);
strain_max = textscan(read_ansys{1,1}{sizeline,1},'%s %f');
constraint = strain_max{1,2};
fclose(fid);
end

if pf == 0
    pf=100;
else
    if constraint > 0.925*10^-3
        penalty = constraint/(0.925*10^-3);
    else
        penalty = 1;
    end
    pf=10/pf + 100*(penalty-1)^2;
end

fileID = fopen('result.txt','a+');
cellsize = size (CPs);
fprintf(fileID,\n %s','D.Var = ');
for i=1:cellsize(1,2)
    fprintf(fileID,'%f\t',CPs(1,i));
end

fprintf(fileID,'%s %f',' > Effi = ',10/pf);
fclose(fileID);

clc;
strtrim(sprintf('%f ',CPs))
sprintf('%f',10/pf)
sprintf('%s','Next evaluation is running ... Please wait until the current evaluation is finished :-)')
end