DEVELOPMENT OF AN OBJECT-ORIENTED DESIGN, ANALYSIS AND SIMULATION SOFTWARE FOR A GENERIC AIR VEHICLE

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

MURAT ŞENİPEK

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
THE DEPARTMENT OF AEROSPACE ENGINEERING

SEPTEMBER 2017

Approval of the thesis:

DEVELOPMENT OF OBJECT-ORIENTED DESIGN, ANALYSIS AND SIMULATION SOFTWARE FOR A GENERIC AIR VEHICLE

submitted by **MURAT ŞENİPEK** in partial fulfillment of the requirements for the degree of **Master of Science in Aerospace Engineering Department, Middle East Technical University** by,

Prof. Dr. Gülbin Dural Ünver ______________________
Director, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Ozan Tekinalp ______________________
Head of Department, **Aerospace Engineering**

Asst. Prof. Dr. Ali Türker Kutay ______________________
Supervisor, **Aerospace Engineering Dept., METU**

Examining Committee Members:

Prof. Dr. Ozan Tekinalp ______________________
Aerospace Engineering Dept., METU

Asst. Prof. Dr. Ali Türker Kutay ______________________
Aerospace Engineering Dept., METU

Prof. Dr. Yusuf Özyörük ______________________
Aerospace Engineering Dept., METU

Assoc. Prof. Dr. İlkay Yavrucuk ______________________
Aerospace Engineering Dept., METU

Asst. Prof. Dr. Mustafa Kaya ______________________
Aeronautical Engineering Dept., AYBU

Date: 08.09.2017

I hereby declare that all the information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name:

Signature:

ABSTRACT

DEVELOPMENT OF AN OBJECT-ORIENTED DESIGN, ANALYSIS, AND SIMULATION SOFTWARE FOR A GENERIC AIR VEHICLE

Şenipek, Murat
M.S., Department of Aerospace Engineering
Supervisor : Asst. Prof. Dr. Ali Türker Kutay

September 2017, 181 pages

In this thesis, comprehensive software is developed to predict the performance of generic air vehicles. The word “generic” stands for the variety of the air vehicles that this Generic Air Vehicle Model (GAVM) encloses. GAVM software includes multiple disciplines such as flight dynamics, aerodynamics, propulsion, rotational dynamics, flight stability and control, numerical multi-variable optimization and object oriented programming. Helicopters, airplanes, compound helicopters, multi-rotor and tilting rotor vehicles can be designed, analyzed and simulated. In order to achieve this ability of modeling each aerodynamic component of a given air vehicle must be mathematically modeled in a generic manner. Therefore, object oriented programming principles are implemented while developing the code such that each modeled component can be populated and used wherever necessary. GAVM software is written in C++ programming language and is used both as a standalone application and shared library. After modeling an air vehicle there are different types of analysis options. Trim analysis, dynamic flight simulation, point performance predictions are available. Beyond the inherently available analysis options, GAVM can be used by an external application as a plant model for an air vehicle. Therefore, wide-variety of studies is possible to conduct such as control system design, flight simulation, and design optimization. The fidelity and complexity of the mathematical models of the components are compromised such that the balance between the computational time cost and analysis requirements is sustained. Shared library version of the code provides the ability to simulate the flight of different air vehicles in same environment which enables the designer to handle swarm-like problems for different air vehicles.

In the first chapter, introductory information is provided about the problem and current requirements of aerospace designs and analyses. Next chapter consists of implemented mathematical theories behind the modeled components in detail. Each component and each sub-model is prescribed and relations between the modeled objects are outlined. Trim point optimization algorithm which uses mainly the classical Newton’s optimization theory is explained. Moreover, implemented Engine model to simulate the available power and consumed energy to see the limits of the designed vehicle and conduct the point performance calculations is explained. Concepts related to software design are also mentioned in that chapter and brief information is provided about the algorithm. Next chapter includes a flight simulation example of a quad-rotor which is modeled via GAVM and abilities of the software are depicted. Throughout the next two chapters, micro-scale and macro-scale validation cases are implemented and results are compared with the available test data. Each sub-component is validated in the first chapter and conventional helicopter; airplane, tilting-rotor and quad-rotor configurations are analyzed and compared with the flight test data. Results show consistency and each modeled component is validated with the available aerodynamic data successfully. Last chapter includes the conclusion and appendix provides input/output sample files.

To sum up, this work includes detailed multi-disciplinary analysis software for air vehicles and brings the ability to accelerate and facilitate the design and analysis of aerial vehicles and new concepts.

Keywords: Helicopter, aircraft, airplane, object oriented, simulation, control, tilt-rotor, multi-rotor, performance, inflow

ÖZ

JENERIK HAVA ARAÇLARI İÇİN NESNE YÖNELİMLİ TASARIM, ANALİZ, VE BENZETİM YAZILIMI GELİŞTİRME

Şenipek, Murat
M.S., Havacılık ve Uzay Mühendisliği Departmanı
Danışman: Yard. Doç. Dr. Ali Türker Kutay

Eylül 2017, 181 sayfa

Bu tez çalışmasında, "jenerik" hava araçlarının performans hesaplamalarını yapabilmek amacıyla geniş kapsamlı bir yazılım geliştirilmiştir. "Jenerik" sözcüğü GAVM yazılımının kapsadığı çeşitli tip hava araçlarını ifade etmektedir. GAVM yazılımı kendi bünyesinde uçuş dinamiği, aerodinamik, itki, rotasyonel dinamik, uçuş kararlılığı ve kontrol, nümerik çoklu-değişkenli optimizasyon ve nesne yönelimli programlama gibi çoklu disiplinler içermektedir. Uçak, helikopter, çoklu-rotor, bileşik helikopter ve eğilen rotorlu hava araçları gibi araçlar tasarlanıp, analiz edilip, benzetimi yapılabilmektedir. Bahsedilen analiz kabiliyetine erişmek için her bir aerodinamik parçanın jenerik bir yaklaşım ile matematiksel olarak modellemesi gerekmektedir. Bu nedenle nesne yönelimli programlama yaklaşımı kullanılmıştır. Bu sayede her bir parça otomatik çoğaltılarak gereken yerlerde kullanılabilmektedir. GAVM yazılımı C++ programlama dili kullanılarak hem konsol uygulaması hem de paylaşılan kütüphane olarak kullanılacak şekilde geliştirilmiştir. Bir hava aracı modellendikten sonra denge koşulu analizi, dinamik uçuş benzetimi ve çeşitli performans analizleri yapılabilmektedir. Yazılım, kendi içindeki mevcut algoritmalar dışında, bir hava aracı modeli olarak herhangi bir analiz ortamına dahil edilebilir. Bu sayede uçuş benzetimi, kontrol sistem tasarımı ve tasarım optimizasyonu gibi çeşitli çalışmalarda kullanılmaya uygun hale gelmiştir. İçerisindeki matematik modellerin derinlik seviyesi ve karmaşıklığı konusunda analiz süresi ve kapsamı çerçevesinde uygulanabilir yöntemler izlenmiştir.

İlk bölümde probleme giriş yapılmıştır ve havacılık sektöründeki güncel tasarım ihtiyaçlarının altı çizilmiştir. Sonraki bölümde ise her modül için kullanılan teorik yaklaşımlar detaylıca açıklanmıştır. Temelde Newton'un optimizasyon teorisini kullanan denge koşulu tespit algoritması hakkında bilgi verilmiştir. Yazılıma eklenen motor güç ihtiyacı ve enerji tüketimi modeli açıklanmıştır. Bu model ile yapılması mümkün olan performans analizleri hakkında bilgi verilmiştir. Yazılım mimarisi açıklanıp genel bilgiler verilmiştir. Sonraki bölüme GAVM ile modeli oluşturulan örnek bir dört pervaneli hava aracının dinamik benzetimine yer verilmiştir. Bu bölümdeki amaç GAVM yazılımının nokta analizler dışında dinamik benzetimler de yapabildiğini göstermektir. Bundan sonraki iki bölüm boyunca her parçanın izole olarak ya da tam hava aracı şeklinde deneysel veri ile doğrulanması çalışmalarına yer verilmiştir. Bu çalışmalarda helikopter ve dikey iniş kalkış yapabilen uçak modelleri kullanılmıştır. Bu karşılaştırma sonuçları eldeki aerodinamik verilerin yeterliliği ölçüsünde başarılı bir şekilde sonuçlandırılmıştır. Son bölümde de sonuç paragrafı yazılmış olup ekler kısmında da örnek girdi/çıktı dosyalarına yer verilmiştir.

Sonuç olarak bu tez çoklu disiplinli ve detaylı bir hava aracı analiz yazılımı içermektedir ve yeni konseptlerin tasarımı, geliştirmesi ve analizi çalışmalarında hızlı bir çalışma ortamı sunmaktadır.

Anahtar Kelimeler: Helikopter, uçak, nesne yönelimli, benzetim, kontrol, çoklu-rotor, performans, iç akış

To my love,

ACKNOWLEDGEMENTS

I wish to express my deepest gratitude to my Thesis Supervisor Asst. Prof. Dr. Ali Türker Kutay for his guidance, advice and insight throughout the research done.

I would like to thank to the members of dissertation committee for their valuable feedback; Prof. Dr. Ozan Tekinalp, Prof. Dr. Yusuf Özyörük, Assoc. Prof. Dr. İlkay Yavrucuk, Asst. Prof. Dr. Ali Türker Kutay, and Asst. Prof. Dr. Mustafa Kaya.

I would like to express my gratitude to Prof. Dr. Ozan Tekinalp for his support during the graduate education program in Aerospace Engineering Department.

I especially would like to express my gratitude to Assoc. Prof. Dr. İlkay Yavrucuk for his valuable helicopter dynamics course which significantly contributed to me in M.Sc. education and for his significant feedbacks.

I am thankful to my company TAI for supporting me to take courses in Master's program and I am highly grateful to my manager in TAI Mr. Yüksel Ortakaya for his initial support and guidance which shapes my thesis plan and development process.

I especially would like to thank my friend Osman Güngör for his endless support and cooperation.

My deepest thanks to my friend Metehan Yayla who helped me whenever I need and gives valuable advices.

I would also like to thank to my colleague Arda Yücekayalı for his support and feedback during this dissertation.

I would like to thank to my colleague Anıl Öztürk for his valuable support in object oriented programming concepts.

I would also like to thank to my friends Afif Umur Limon, Talha Mutlu, Anıl Sami Önen, Levent Cevher, Batuhan Hoş, Yosheph Yang, Ramin Rouzbar, İsmail Özdemir Uzunlar, and Saffet Gökuç for their support and friendship in several design competitions and projects during the education period in METU AEE Department.

I would like to thank my friend and colleague Burak Sarsılmaz for his brotherhood, wisdom, and support and valuable moments we shared since 2004.

I would especially like to thank my father Besim, and my mother Ayşe for their support, guidance and endless love during every moment in my life.

Finally, I would like to thank to my spouse Gözde for her endeavor, support, patience and love during this dissertation period and valuable moments we shared from high-school years.

TABLE OF CONTENTS

CHAPTERS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

μ	*Advance Ratio*
α	*Angle of Attack*
M	*Mach Number*
CW	*Clockwise*
CCW	*Counter Clockwise*
u, v, w	*Body axis velocities*
$\dot{u}, \dot{v}, \dot{w}$	*Body axis accelerations*
p, q, r	*Body angular rates*
$\dot{p}, \dot{q}, \dot{r}$	*Body angular accelerations*
ϕ, θ, ψ	*Euler angles* $(roll, pitch, yaw)$
$\dot{\phi}, \dot{\theta}, \dot{\psi}$	*Euler rates* $(roll, pitch, yaw)$
V_N, V_E, V_D	*Earth velocities* $(north, east, down)$
X_N, X_E, X_D	*Earth positions* $(north, east, down)$
$u_{air}, v_{air}, w_{air}$	*Total air velocity components*
g	*Earth Gravitational Acceleration*
g_x, g_y, g_z	*Earth Gravitational Accelerations in body coordinates*
FS, BL, WL	*Stationline, Buttline, Waterline*
$\lambda, \varphi, \mathrm{h}$	*Longitude, Lattitude, Altitude*
$\beta, \dot{\beta}, \ddot{\beta}$	*Blade Flapping angle, rate and acceleration*
$\delta, \dot{\delta}, \ddot{\delta}$	*Blade Lagging angle, rate and acceleration*

LIST OF ABBREVIATIONS

CPU	Central Processing Unit
GAVM	Generic Air Vehicle Model
GPU	Graphics Processing Unit
GUI	Graphical User Interface
IBC	Individual Body Coordinates
ISA	International Standard Atmosphere
MAC	Mean Aerodynamic Chord
MBC	Multi-Body Coordinates
MTOW	Maximum Take-Off Weight
NED	North-East-Down
NACA	National Advisory Committee for Aeronautics
OOP	Object Oriented Programming
RPM	Revolution per Minute
SL	Sea-Level
TOW	Take-Off Weight
VTOL	Vertical Take-Off and Landing
XML	eXtensible Markup Language

CHAPTER 1

INTRODUCTION

1.1 Motivation of the Study

Development in the technology of aerospace increases the motivation to design and manufacture both conventional and unconventional air vehicle configurations such as tilting rotors, multiple rotors, and compound air vehicles. Therefore, an environment for the quick analysis and simulation of these kinds of vehicles are required especially for the conceptual and preliminary design phases.

In this thesis, generic software for the analysis and simulation is developed by using the object-oriented programming (OOP) principles. Each aerodynamic and dynamic component of a general air vehicle is mathematically modeled and implemented in the software. Developed software is a useful tool for design, analysis and flight simulation purposes.

1.2 Limitation of the Study

The study conducted in this thesis is limited by the fidelity and correctness of the data provided for the air vehicle components. Aerodynamic coefficient tables, correlation of parameters and structural parameters affects the accuracy of the results.

Rotor inflow models are limited to actuator disk approach and effects of tip vortices; wake-contraction and blade vortex interaction is not physically modeled but simulated by correction factors. Flight regime is limited to subsonic airspeeds; however, due to its object oriented nature, these kinds of modifications are easy to implement for further researches. Although trim results are validated by available experimental data, dynamic responses for time simulation of flight dynamics are not validated and needs detailed validation work.

1.3 Outline of the Thesis

Chapter 2 provides information about the mathematical modeling of air vehicles and introduces similar software. In this chapter, required analysis types and methodologies are discussed during the design phase of an air vehicle.

Chapter 3 gives detailed background information about the theory behind of each component modeled in the software. Coordinate systems, axis transformations, mathematical and analytical formulations behind the aircraft component models, software class diagrams, assumptions and corrections are explained.

Chapter 4 contains the comparison and validation works for isolated components. Isolated rotor, wing and propeller models are validated with available experimental data.

In Chapter 5 there are validation cases by using flight test data for full helicopter and tilting rotor vehicle. With the availability of geometric and aerodynamic data, analysis results are obtained and compared with the experimental data. Although the fidelity of the available geometric and aerodynamic data is limited, the results show consistency.

Chapter 6 includes a sample flight simulation model of LYNX XZ170 helicopter, XV-15 rotorcraft and a quadrotor air vehicle in order to depict the ability of providing non-linear flight dynamics simulation model. LYNX and XV-15 open loop simulation results are depicted. A sample controller is designed for quadrotor to show that the trim analysis and non-linear simulation of multiple rotor aircrafts is possible with the software.

Chapter 7 finalizes the thesis with a conclusion to outline the completed work and provides possible areas for further researches.

CHAPTER 2

LITERATURE REVIEW

Advancements in the aviation technology enable the designers to develop wide range of designs. Conventional propeller airplanes and conventional helicopters are the pioneers of the aviation history. Flying vehicles are designed for different purposes. Fixed wing aircrafts are designed to minimize the required time to travel to a destination. As the years passed, aircrafts with turbofan, turbojet and ramjet engines are developed and flight speed increased to super-sonic levels. Airplanes are being used in transportation, scientific researches, military, surveillance, fire-fight, and several fields in recent years. On the other hand, rotating wing concepts such as conventional helicopters are emerged and widely being used for similar operations in low speeds where the hovering is required. Developments in the aviation industry lead the engineers to design platforms which are able to hover and fly at high forward velocities. Tilting rotors and VTOL aircrafts are the examples of these concepts since they can hover like a conventional helicopter and can be transformed into fixed wing configuration.

In this age, the computation power of machines rises rapidly. High computation power makes it available for designers to design aircrafts and analyze aerodynamics, structure, stability, vibration and flight dynamics during even the conceptual design phase. Therefore, design optimization studies are no longer conducted for low levels (i.e. single-disciplinary) such as providing a higher lift airfoil shape. In contrast, airfoil optimization analyses may take into account multiple disciplines such as the stability, mission performance, noise characteristics and project requirements. As a result, design and analysis tools emerged which are capable of doing multi-disciplinary optimizations in aircraft design phases. With the existence of an aircraft mathematical model, wide variety of aircraft parameters related to overall performance are possible to be calculated and possible to be appended into design optimization.

In this scope, requirement of a generic comprehensive analysis tools gains importance to minimize the time cost to predict the performance of the new designs. The word of “Comprehensive” is utilized for a single software to conduct the aerodynamics, dynamics, structure, stability and control analyses tasks which requires almost similar level of modern technology [1]. Since the problems related to the advanced designs described above are multidisciplinary, generic models of each discipline is required such as wing aerodynamics, rotor aerodynamics and dynamics, engine modeling and flight dynamics.

Object oriented programming (OOP) concept provides a feasible environment to create mathematical models of sub-components that can be populated, derived and subordinated. Each type of aircraft component is modeled with OOP concepts to be populated and different types of aircrafts can be generated by populating the component objects. Commonly used components such as rotor, propeller, wing and fuselage are the main components exist in an airplane, helicopter or compound aircrafts.

After the computer technology matured and used by aerospace engineers the ancestors of the rotorcraft analysis tools are emerged in the 1960s as given in Figure 2-1. Several computer programs are developed by the companies and institutes since 1960s, among these codes there are comprehensive analysis software as well as specific codes for rotorcraft aerodynamics, structures, handling qualities and simulations.

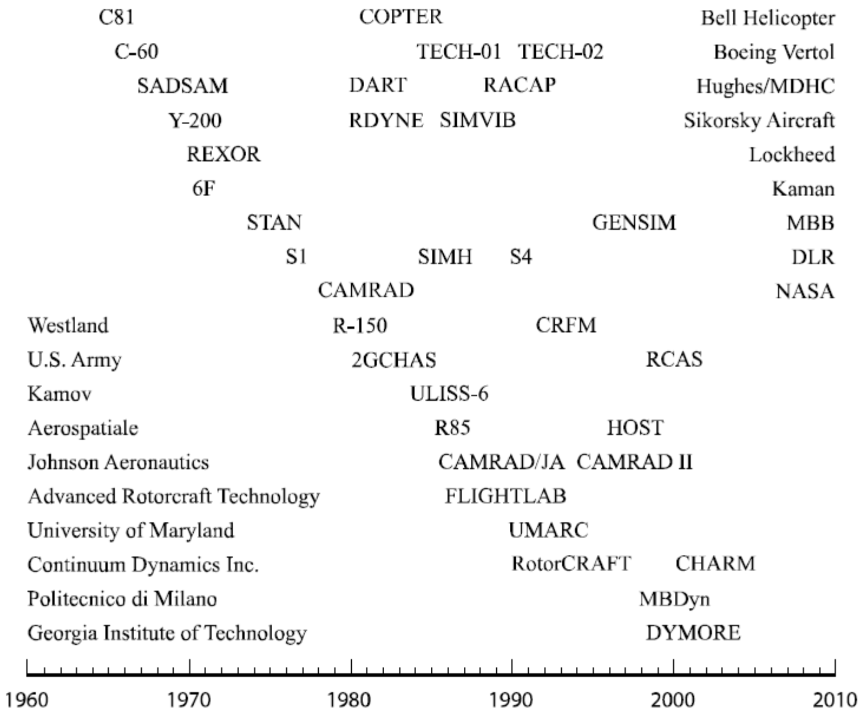


Figure 2-1 Rotorcraft analysis tools and developers in the last fifty years [1]

Correct modeling of the air vehicle before prototyping is crucial to reduce the time cost, reduce the probability of errors in design phase and enable to implement latter steps such as control design and handling qualities characteristics. In the design phase, where the designer shows that the design meets the requirements and objectives, these kinds of programs plays a significant role. If the designer owns validated aircraft analysis software, the cogency of the correctness of the calculations is probably high according to the authorities. Therefore, since the early times of availability of computers the manufacturers invest into the development of analysis tools.

Another point which is “good to have” is the user-friendliness of the code. The word user-friendly means that being easy to run, configure and modify. Since the availability of the advanced software technologies in today’s World, it is much easier to develop codes which are user-friendly. Configuration files, analysis and output files, plotting capabilities, and diagnosis opportunities should be implemented well into the analysis software.

As a result, to meet the technology requirements of the new era in aviation industry, designers focusing on new unusual designs mostly in unmanned air vehicle systems. Therefore, the main purpose in this thesis is to provide a user-friendly, object oriented, and generic design, analysis and simulation software which is able to model a wide range of air vehicles.

2.1 Literature Survey, Problem Definition and Applications

Beginning from the mid-1960s as shown in Figure 2-1, several analysis tools are developed for aircrafts. In this chapter, brief information is provided for some most popular software which is used for modeling aerodynamics and structural dynamics, flight simulation and design.

HELI-DYN

Heli-Dyn is a commercial tool which is developed by Aerotim Engineering Ltd. to design and analyze helicopters and perform a flight simulation [2]–[4].Heli-Dyn enables users to build verified dynamic models fast and easy. Model libraries includes sophisticated mathematical models for the components such as Peters-He inflow models, ground effect models, flapping dynamics, and Free-Vortex Wake models which are solved by GPU and boosts 100x than CPU as given in Figure 2-2.

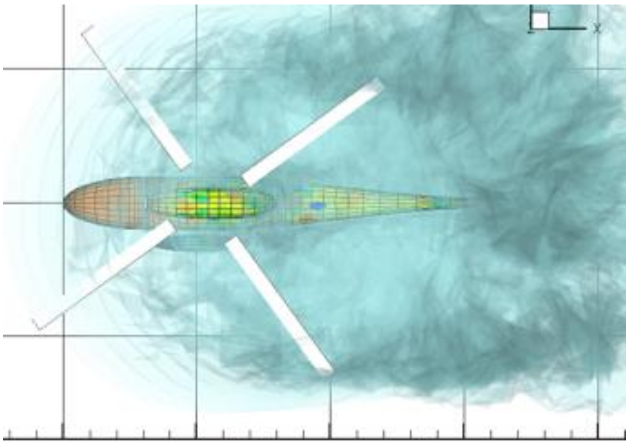


Figure 2-2 Rotor wake visualization from Heli-Dyn

Heli-Dyn has a user friendly graphical user interface to generate the model of the aircraft, obtain trim condition and perform some analysis as shown in Figure 2-3. It is capable of trimming the rotorcraft and providing a linear model around a trim point. Moreover, GUI of Heli-Dyn provides opportunity to perform dynamic analysis for a step input, sinusoidal inputs or a custom input if desired.

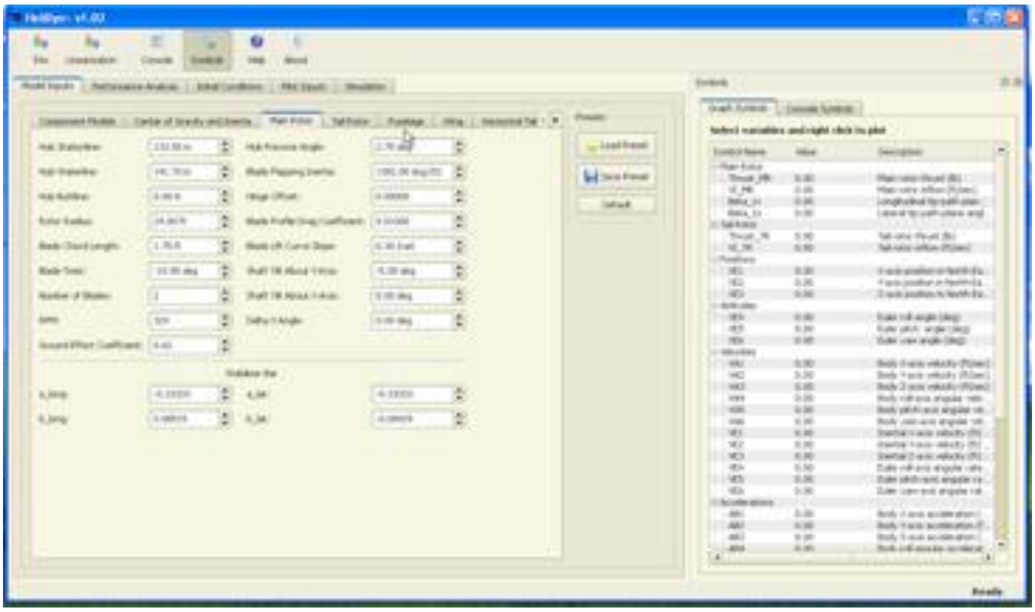


Figure 2-3 Graphical user interface of Heli-Dyn

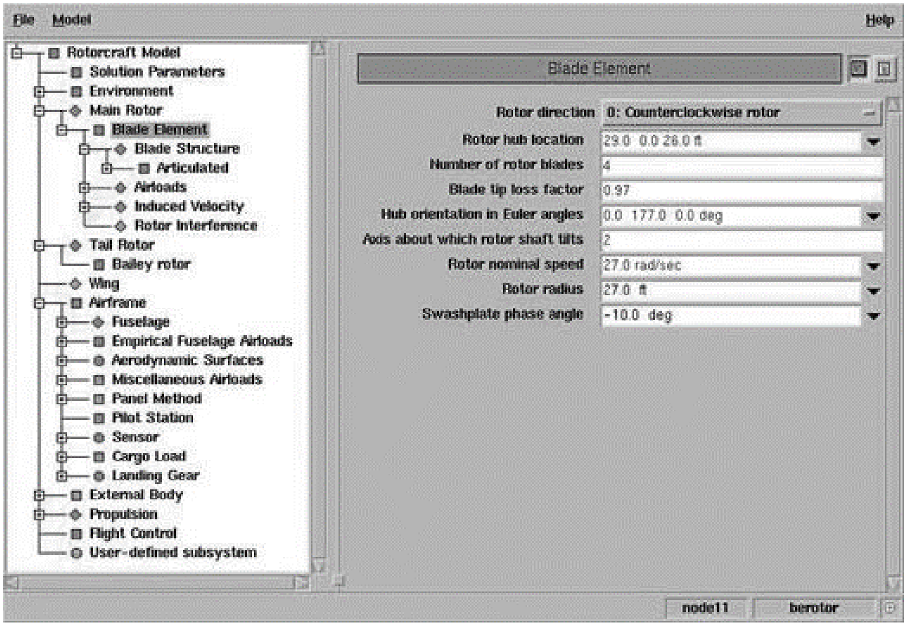
Heli-Dyn provides a DLL version of the generated helicopter model which can be used in MATLAB Simulink for simulation and control design analyses. Heli-Dyn student version is being used in METU Aerospace Engineering Control Design and Helicopter dynamics courses interactively.

Heli-Dyn+ is the advanced version of the Heli-Dyn which includes the muti-rotor and tilting rotor analysis of rotorcrafts, and integrated to a reconfigurable simulator.

Finally, Heli-Dyn is a comprehensive design, analysis and simulation tool for helicopters and multi-rotors which has integrated GUI and Flight Simulator which can be used in wide range of areas such as education and professional use.

FLIGHTLAB

FlightLab is a Linux operating system based commercial software which was developed by Advanced Rotorcraft Technologies Inc. [5]–[8].



Rotor direction	0: Counterclockwise rotor
Rotor hub location	29.0 0.0 26.0 ft
Number of rotor blades	4
Blade tip loss factor	0.97
Hub orientation in Euler angles	0.0 177.0 0.0 deg
Axis about which rotor shaft tilts	2
Rotor nominal speed	27.0 rad/sec
Rotor radius	27.0 ft
Swashplate phase angle	-10.0 deg

Figure 2-4 Screenshot of GUI from FlightLab Model Editor (FLME)

FlightLab allows users to generate air vehicle models interactively from a pre-defined model library by arbitrarily selecting the modeling components, relating each of them into a user-defined architecture. It is a computer based engineering software having graphical user interface as given in Figure 2-4 tool for analysis of flight dynamics [9].FlightLab enables designer to build each section of project separately and then gather all the separate pieces and disciplines which are depicted in Figure 2-5 together under a common framework [5].

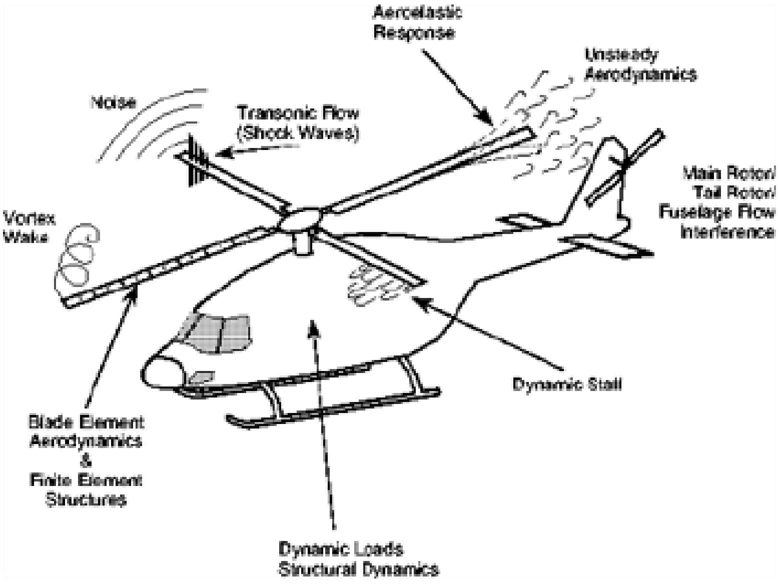

Aeroelastic Response
Unsteady Aerodynamics
Noise
Transonic Flow (Shock Waves)
Main Rotor/ Tail Rotor/ Fuselage Flow Interference
Vortex Wake
Dynamic Stall
Blade Element Aerodynamics & Finite Element Structures
Dynamic Loads Structural Dynamics

Figure 2-5 Multi-disciplinary phenomenon in helicopters

FlightLab key features may be summarized as;

- Multi-Body Dynamics
- Graphical User Interface for Modeling
- Design and Analysis Capabilities
- Test and Evaluation Capabilities
- Real Time Simulations
- Open Architecture

FLIGHTLAB includes finite state dynamic inflow models[10] and vortex wake models for rotor aerodynamics, it consists finite element model for structure dynamics and a non-linear beam model.

Above features of FLIGHTLAB makes it widely used in rotorcraft flight dynamics. Several universities, US Army and leading rotorcraft companies utilize this tool in their research, designs and simulators. Numerous publications exist in literature using this software.

CAMRAD/JA and CAMRAD II

CAMRAD (Comprehensive Analytical Model of Rotorcraft Aerodynamics and Dynamics) was developed for NASA and U.S. Army. [1][11][12] The purpose of the development was to generate a tool for the analysis of wide range of air vehicles. Available analysis technology was not utilized as desired in those years. Different analysis tools were being used for different types of analysis. Therefore, CAMRAD is developed during 1978-1979 with a Scully vortex wake model [13] for a general double rotor helicopter with a single load path elastic blade model.

CAMRAD is improved as CAMRAD/JA during 1986-1989 by Johnson Aeronautics. Previous version of the software are extensively modified and updated, integrating dual-peak blade circulation model, and second order lifting line [14], and wake roll-up models. Loose coupling with CFD was implemented. Free wake model developed for CAMRAD/JA was adopted into the analysis tools COPTER, UMARC, and 2GCHAS.

CAMRAD II is a computer software for aeromechanical analysis of helicopters and rotorcraft developed by Johnson Aeronautics which is the latest version of CAMRAD [15], [16]. CAMRAD II incorporates advanced technology which are multibody dynamics, non-linear finite elements, rotorcraft aerodynamics and wakes [17]. This software provides a comprehensive analysis environment for the design, testing and evaluation of systems for all stages including research, conceptual design, detailed design, and development. CAMRAD family is programmed in FORTRAN software language. CAMRAD II calculates performance, loads, vibration response, and stability with a high level of technology in a single computer program. CAMRAD II is applicable to a wide range of problems, and a wide class of configurations.

CAMRAD II has building block approach which leads to more general, more rigorous models. It provides separate physical and logical pieces; separate structural and aerodynamic models and each piece are capable of general analysis. It is one of the most comprehensive tool for rotorcraft design and analysis which combines finite element structural model with free wake model enabling the designer to obtain a full rotorcraft trim and transient analysis tasks. CAMRAD II is widely used in rotorcraft industry and rotorcraft research areas.

CHARM

Comprehensive Hierarchical Aeromechanics Rotorcraft Model (CHARM) is a comprehensive analysis software which utilizes vortex wake and panel methods together to analyze rotary wing aerodynamics, rotor wakes, and rotor body interactions and capable different multiple rotor concepts as given in Figure 2-6.

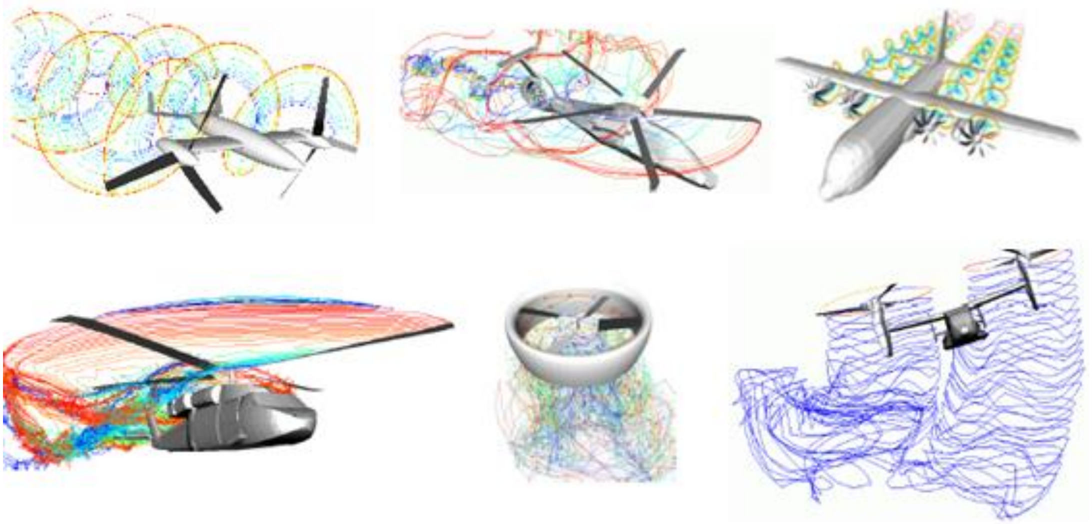


Figure 2-6 Vortex wake visualizations from CHARM [18]

Several validation studies are conducted to increase the accuracy of the prediction of performance parameters for rotating-wing aerodynamics. CHARM provides Vortex Wake Model for rotors, Panel Method for fuselage, wing, ground structures, ships etc. with a fully coupling of Wake/Panel solutions for modeling of aerodynamic interactions. CHARM is able to analyze rotors with free-wake in real-time [19]. Moreover, to capture better three dimensional flow effects it provides Lifting Surface Blade Model for chord wise loading. These models are also providing a feasible environment for predicting aero-acoustic noise of rotating blades. While there are several advantages, CHARM has some drawbacks when compared with other comprehensive tools. CHARM does not support full aircraft trim in maneuvering flights, there is no flutter and stability analyses.

JSB SIM

JSBSim is an open source code, flight dynamics simulation model written in C++ [20]. JSBSim is a 6 DoF, high fidelity, and generic flight dynamics model library. It can be integrated into simulation environments such as (Flight Gear [21], and OpenEagles[22])

In JSBSim physical components for simulating flight dynamics are modeled such as atmosphere, control system, propulsion system, and aerodynamics. JSBSim model can be used as a “black box” within some simulation and control design environments such as MATLAB Simulink by S-Function blocks. Aircrafts are configured by using a configuration XML file in which the user defines the mass, ground reactions, propulsion, system, autopilot, flight control, aerodynamics, input, and output configurations for the flight dynamics model.

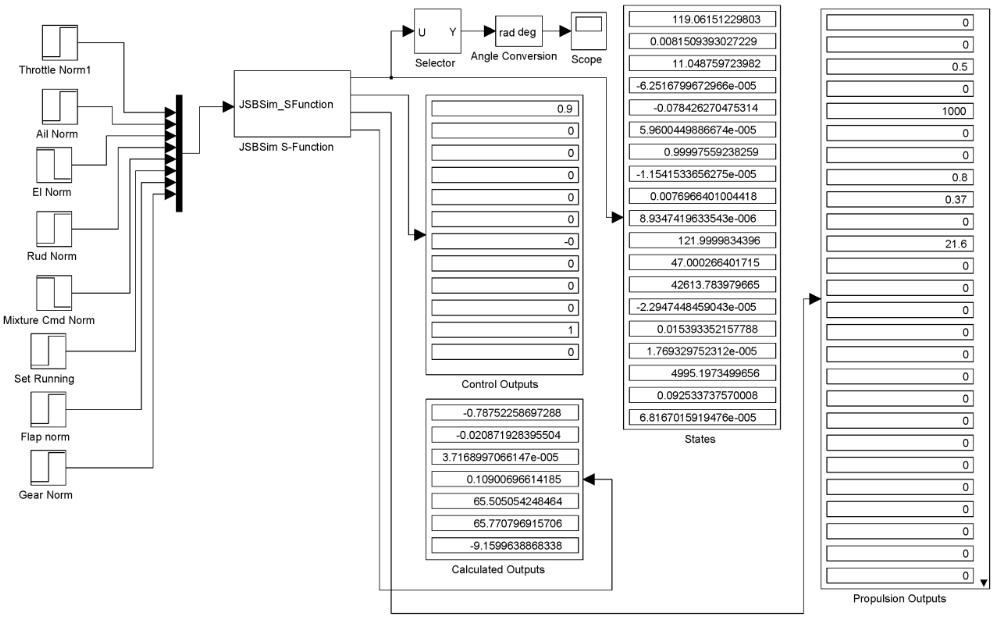


Figure 2-7 JSBSim S-Function Interface [23]

JSBSim can be used as standalone or batch analysis mode by using an external script. JSBSim has got variety of uses in scientific researches and industry. JSBSim has been used in Hardware-in-the-Loop (HIL) tests, air traffic studies, human pilot mathematical models, and several flight simulators.

DYMORE

Dymore is multi-body dynamics software based on finite element method which is being used for modeling of flexible multi-body dynamics systems. [24] In structural models of Dymore, there are damper, spring, flexible joint, rigid body mass, beam, cable, membrane, and shell elements. By using these objects, a helicopter rotor or wind turbine can be modeled with flexible structural dynamics. Moreover, in aerodynamics Dymore provides 2-D and 3-D finite state generalized dynamic wake theory for rotors [25]. Modeling of wing aerodynamics includes lifting line theory. Unsteady aerodynamics are modeled with Leishman-Beddoes unsteady aerodynamics and dynamic stall theory and ONERA dynamic stall model [26]–[28].

2.1.1 Classical Aircraft Modeling

Flying vehicles contains several disciplines due to their nature. This multidisciplinary character of aircrafts includes high risks during the early design phase of the aircraft. A false decision for a component which interacts with other disciplines during the preliminary design phase may end up with huge costs in later phases. Therefore, simulation modeling of either the aircraft or the individual components provides an estimate of aircraft before prototyping. This simulation method should be composed of models from multiple disciplines. In this thesis dynamic, aerodynamic and flight mechanics modeling of air vehicles are presented by following an object-oriented approach.

Analysis of the dynamic systems can be conducted in two ways. First way is developing mathematical methods to calculate the vehicle's performance. Analytical and numerical methods have several assumptions in air vehicle simulations. Second way is to conduct experiments to gather information about the aerodynamic and structural dynamics of components. The data collection is done by system identification and parameter estimation. Therefore, number of assumptions and approximations are less than the first method. Obviously, experimental analyses are the most reliable way of predicting performance of the air vehicle, however, these experimental analyses are costly and scheduling the entire test into the project calendar often becomes impossible. The optimum way should be decided by the designers. Therefore, there is a need for an environment where mathematical methods and experimental data are nested and preliminary simulation and performance analyses can be conducted. Air vehicle modeling provides solutions in various fidelities either full mathematical methods or hybrid methodologies with experimental and mathematical models. The best way is to start designing the aircraft with mathematical models and correlate the models with the experimental data. Analysis fidelity should be increased as the design matures.

2.1.2 Object Oriented Modeling Perspective

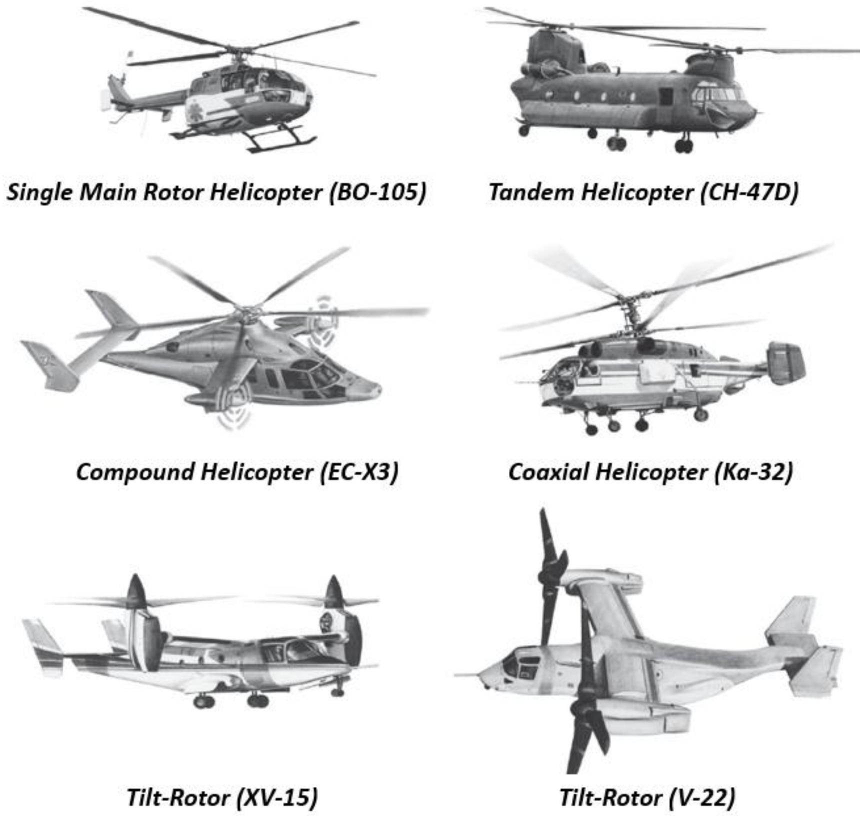
Object oriented programming refers to a software architecture in which the programmer defines both the type of the data structure and the methods (functions) that is implemented into a data structure. Therefore, designed data structure is called as “object/class” which includes both data and methods. These objects are generally used to model the objects which are found in everyday life. These methods operate as the primary tool for object-to-object communication [29].

For air vehicle modeling, this object hierarchy proposes a convenient root for mathematical modeling of generic air vehicle configurations by modeling each common component as an individual object and communicating each object by their specific methods. For instance, airfoil object is modeled to include airfoil lift, drag and moment coefficients and their polars. This object can be used with both the blade profile of the Rotor object and wing profile of the Wing object. OOP approach is useful to populate the individual objects, define relations and manage the common functions.

Since the modeled aircraft is generated by using OOP principles, addition of new methods and models are uncomplicated. Each modeled object can be archived to use further works and with different configurations. For example, it is possible to generate a library having different rotor designs and use them with different rotorcrafts. Moreover, rotorcrafts flying in an environment are possible to be flied simultaneously for collective simulations.

2.1.3 Unconventional Configurations

In recent years, the need for the modeling and analysis of unconventional configurations is increased. That is because the flying vehicles are being used to solve the major problems of the life. Different requirements, different sizes and different capabilities are required by the people.



Single Main Rotor Helicopter (BO-105)

Tandem Helicopter (CH-47D)

Compound Helicopter (EC-X3)

Coaxial Helicopter (Ka-32)

Tilt-Rotor (XV-15)

Tilt-Rotor (V-22)

Figure 2-8 Different Types of Rotorcraft Configurations [30]

There are several types of air vehicles which may have multiple rotor and propeller combinations called as compound rotorcrafts, tilting rotor rotorcrafts, and multi-rotor unmanned aerial vehicles. Such configurations are depicted in Figure 2-8 whose number of rotors, lifting surfaces, number of fuselages may vary. This variation in components of rotorcrafts increases the need for an analysis tool in which these unconventional vehicles could be modeled and simulated.

Moreover, with the rapid advances in unmanned flying vehicles which are called as “drones”, several types of flying vehicles come into daily life. They are available in almost every shopping website or hobby shops. Their sub-components and replacement parts are also available. One can easily design, manufacture and make it fly. However, to simulate different configurations and design flight controllers one requires generic flight dynamics software.

2.1.4 Simulation and Stability Analysis and Control Design

In a design process of an air vehicle, simulation is one of the most significant processes that the designer needs. Before prototyping, the dynamic simulation of the design enables the designer to predict the performance and gives idea about the design whether it meets the requirements or not. Flight simulation is a valuable ability to assess the controller and optimize the controller gains. Therefore, real time non-linear flight simulation ability is useful for air vehicle design and analysis software.

Aircraft stability and control is another crucial point. From the preliminary design phase of an air vehicle, stability and handling quality of the air vehicle should be checked. Linear state-space representation of the air vehicle is required to determine handling quality and controllability. State-space representation of an air vehicle which includes stability derivatives is obtained around a trim point with numerical differentiation in these non-linear analysis tools. Therefore, availability of state space representation is required for the user to assess the overall performance and stability.

2.1.5 Point Performance Calculations

There are main performance characteristics that define the air vehicle. Hover ceiling, cruise velocities, fuel consumption, payload and range values, and takeoff and landing distances are some of the critical point performance parameters. Air vehicle designer should also evaluate the performance parameters of the aircraft. A good design program should be able to provide such an environment that the designer could easily predict the performance parameters of the air vehicle.

2.1.6 Design Optimization

Design optimization is another significant concept in aircraft design. Rotor planform geometry, empennage geometries, wing geometry for an aircraft, placement of control surfaces, center of gravity envelope are some of the most critical topics for an aircraft design. Design optimization can be conducted for some of the topics described above by using an optimization tool with the generic aircraft model, or it can be done by using some methodologies that the designer proposes. For each case the software should be capable of defining an input sets, calculating the influence coefficients and should provide an interface for the communication of different programs. By using such a generic and user friendly software designer could easily optimize the platform.

CHAPTER 3

THEORY OF COMPONENT MODELS

This chapter includes the theory behind the whole architecture of the code and description of mathematical theories for each component implemented in this code. For each component there will exists general information about modeling of the component and currently available models, sign conventions and frame of references, mathematical theory behind the model and its implementation, software class diagram belonging the component describing the relationship between parent and child classes, fields and methods (i.e. variables and functions). Finally, each component will be supplemented by a validation case with experimental data if available.

In the components theory of this thesis there is a “BaseObject” Class which is the parent class of all sub-components such as Rotor, Wing, Body, Slung, and Propeller. This base class holds the common information of these objects such as position, orientation, body accelerations, velocities, rates, forces and moments, and required functions. This BaseObject class is generated to store fields and methods which are common in all child objects and not to define same variables again for each component. Related class diagram is given in Figure 3-1 for BaseObject class. And Rotor, Body, Wing, Propeller, Slung and MainFrame are child classes derived from BaseObject class.

Besides the base object there exist different types of objects related to trim, simulation, and performance analyses. Each dynamic component is inherited from the BaseObject class.

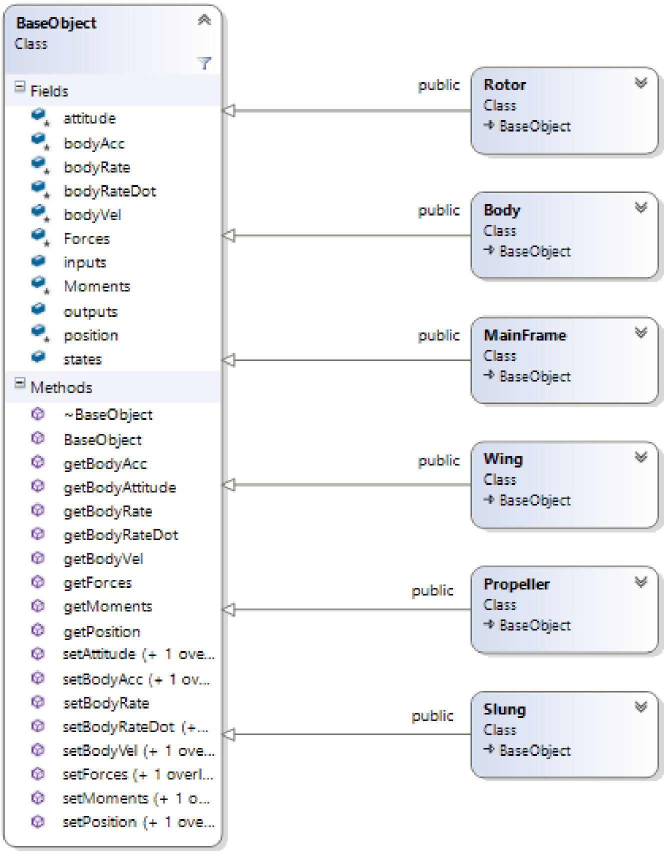


Figure 3-1 Class Diagram of BaseObject Class and its child objects

Each of the dynamic components derived from base object includes mathematical models simulating own dynamics. Therefore, each of the objects can be described by a state space representation as $d\vec{x}/dt = Ax + Bu \; and \; y = Cx + Du$. They all can be represented by a set of inputs u, states x, and outputs y. Class diagram of the dynamic system is depicted in Figure 3-2. Collective and cyclic controls are the default inputs of rotor object. Moreover, RPM and tilt angle may be defined as input. Control surface deflections are the inputs of the Wing object. Other classes do not have inputs. 6-DOF dynamics have 12 states for MainFrame object. Rotor object has dynamic states of rotational speed, flapping and lagging dynamics, and inflow dynamics. Outputs of a component may be anything desired to be existed in the linear state space representation of whole system like forces and moments, velocities, specific loads, and any type of available local properties.

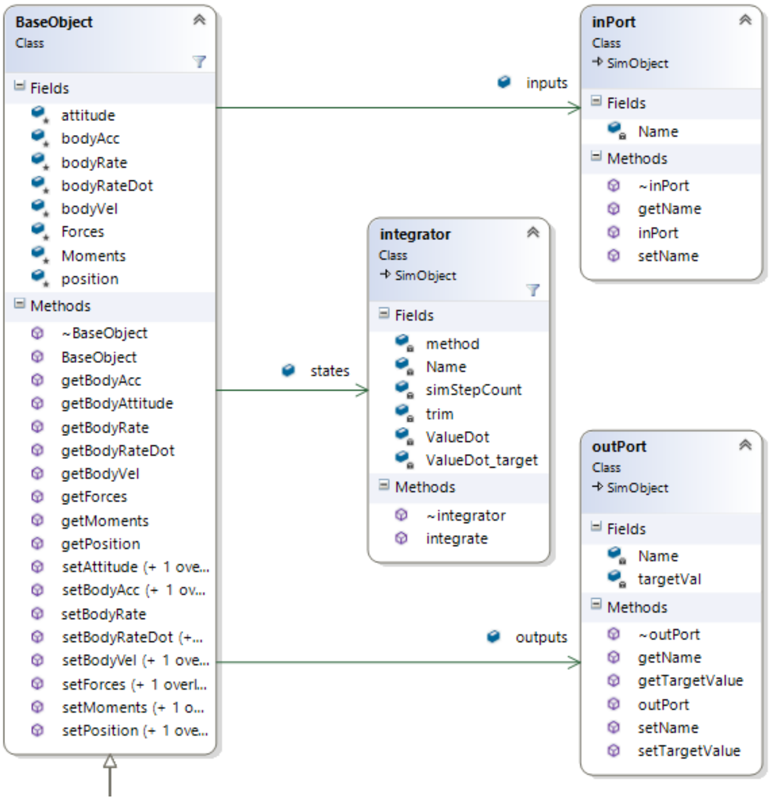


Figure 3-2 Class diagram of a BaseObject class having dynamic system inputs, integrator (state), and output classes

There exist other types of classes which are related to trim, simulation, performance, file output and plot related classes. These classes build up the environment of design and analysis and manage the dynamic objects, holding the information of whole aircraft, include the trim and simulation algorithms. Moreover, XML parser routines exist for input file parsing and model generation. In the following chapters, deeper details exist about the code itself.

3.1 Coordinate Systems, Transformations and Conventions

Multiple coordinate systems are required to be used within this current study. There exist mainly three coordinate systems to describe the motion of the aircraft which are, earth fixed frame, body fixed frame and wind frame. These frames are used to describe the 6-DOF motion, forces and moments acting on the components and/or aircraft. Flight dynamics parameters such as accelerations, rates, velocities and aerodynamic or inertial forces can be obtained for each frame and there exist transformations between them.

Some of the coordinate systems are used to represent each components 6-DOF loads and some of them are used for flight dynamics equations of motion. Body axis, wind axis and earth axis are the systems which are used most commonly in flight dynamics modeling.

3.1.1 Body-Fixed Coordinate System

Body fixed coordinate system is used to define the equations of motion of the aircraft relative to its own "body". Origin of the body frame is the center of gravity of the air vehicle and moves and rotates with the air vehicle. X axis of the body frame points to the nose of the air vehicle. Z axis of the body frame points downwards and it is perpendicular to the X axis. Y axis in body frame could be obtained by right hand rule and it is pointing to the starboard.

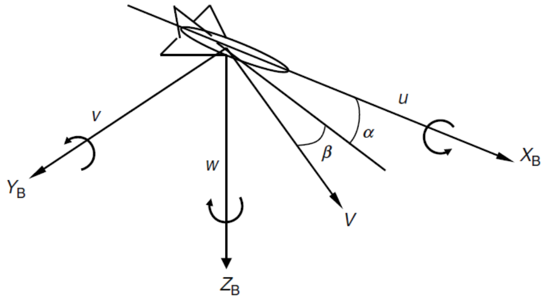


Figure 3-3 Body Axis System [31]

In this thesis, this body axis system is utilized for all sub components, as well. These sub components are wings, attached bodies, slung load, and rotor. All forces and moments acting on each component are represented in the body frame finally and moved to the center of gravity of the air vehicle to solve the equations of motion since the body-fixed frame moves and rotates with the aircraft.

In Figure 3-3 X_B, Y_B, Z_B represents the body frame axis and u, v, w are the body velocities of the aircraft. V is the velocity vector which is defined in wind axis which is defined by angle of sideslip β and angle of attack α with respect to body fixed frame of reference.

3.1.2 Wind Axis Coordinate System

Wind axis system is a coordinate system which moves with the relative air velocity with respect to aircraft. It is not a fixed coordinate system on aircraft. It only translates with the aircraft but rotates with the wind vector. It can be defined by the angles of wind relative to the earth considering the air vehicle is stationary or can be defined by angle of sideslip β and angle of attack α with respect to body frame.

The origin of wind axis is the center of gravity of the aircraft and X axis pointing forward as given in Figure 3-3. Positive Y axis is pointing to starboard and Z axis is pointing downward.

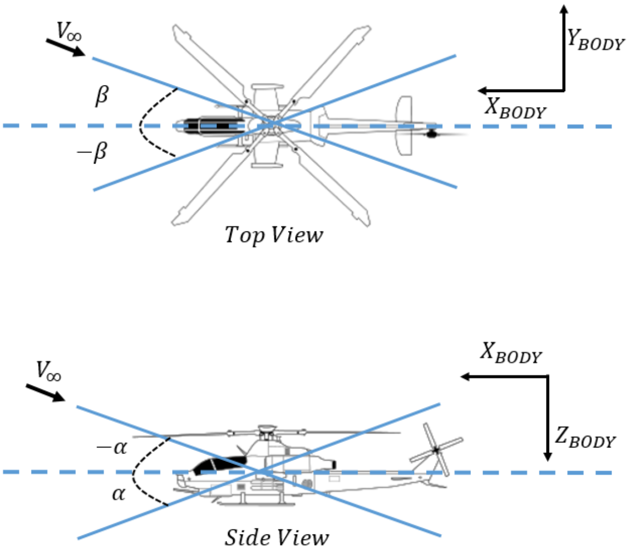


Figure 3-4 Wind coordinate system and axis conventions

This axis system is mainly used to define wind velocity and transform it into body velocities to calculate the aerodynamic forces and moments. Moreover, since the aerodynamic force and moment coefficient tables are given with respect to α and β this frame and wind to body transformation is used. Definitions of wind frame angles are illustrated in Figure 3-4.

3.1.3 Earth-Fixed (North-East-Down) Coordinate System

Earth fixed frame of reference assumes a flat earth surface and lies on the ground of earth. X axis pointing to North and Y axis pointing to East as depicted in Figure 3-5. Z axis is pointing downwards to the center of the earth.

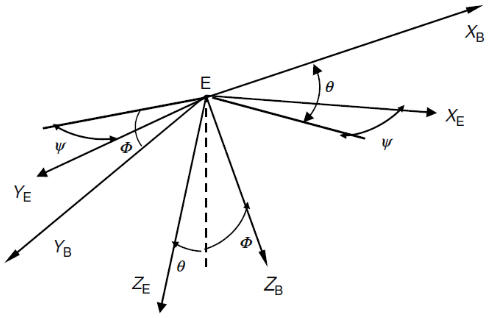


Figure 3-5 Earth axis system [31]

“For air vehicle simulation and modeling problems this frame is called as NED frame and assumed to be the inertial frame of reference where Newton laws are valid [31]”.

3.1.4 Vehicle Carried (North-East-Down) frame of reference

Vehicle carried system is located on the air vehicle and its origin is positioned at the center of gravity of air vehicle. Vehicle Carried NED frame moves with the aircraft and X axis points toward the geodetic north, Y axis points to the east and Z axis points downward and perpendicular to the ellipsoidal normal.

Ground speed, true airspeed, and inertial coordinates are defined with respect to this frame. Moreover, Euler angles defining the orientation of the air vehicle are defined in body axis with respect to NED frame of reference.

The Euler angles are roll ϕ, pitch θ, and yaw ψ angles. This Euler angles move the reference frame to the referred frame with the Z-Y-X order which are yaw, pitch and roll respectively.

3.1.5 Earth-Centered Earth-Fixed (ECEF) and Geodetic Coordinate Systems

ECEF coordinate system is located at the center of the earth and rotates with the Earth around its axis of spin [32]. The Z axis points to the North Pole and lies on the spin axis of Earth as given in Figure 3-6. X axis intersects the earth at zero-degree latitude and zero-degree longitude. Y axis is determined by the usual right hand rule since it is orthogonal to the X-Z plane. Position in ECEF coordinate system is denoted by subscript e as X_e, Y_e, Z_e.

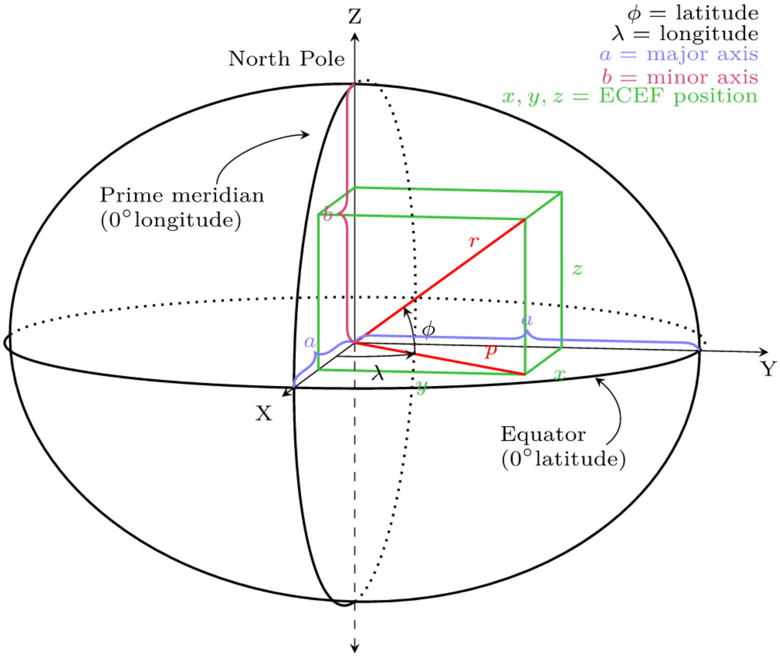


Figure 3-6 ECEF and Geodetic coordinate systems on Earth

Geodetic coordinate system is generally used in navigation which is GPS based. This coordinate system describes a point near the earth surface in terms of latitude(φ), longitude(λ) and altitude(h) as shown in Figure 3-6. Longitude is ranging from -180° to +180° starting from prime meridian. Latitude ranges from -90° to +90° between the equator and poles. The altitude is the normal distance between the reference ellipsoid and the point. Position in Geodetic frame is denoted by φ, λ, h.

3.1.6 Airplane Coordinate Axes

Airplane coordinate axes are composed of Station Line (FS), Water Line (WL) and Butt Line (BL) whose origin can be defined at any point close to aircraft. This coordinate system is used to locate the all components on the aircraft. Positive WL is directed upwards, positive FS is directed backward and positive BL is directed starboard on the aircraft as seen in Figure 3-7. In this thesis location of all the components attached to the aircraft and the center of gravity of the aircraft are identified by using the airplane coordinate axis.

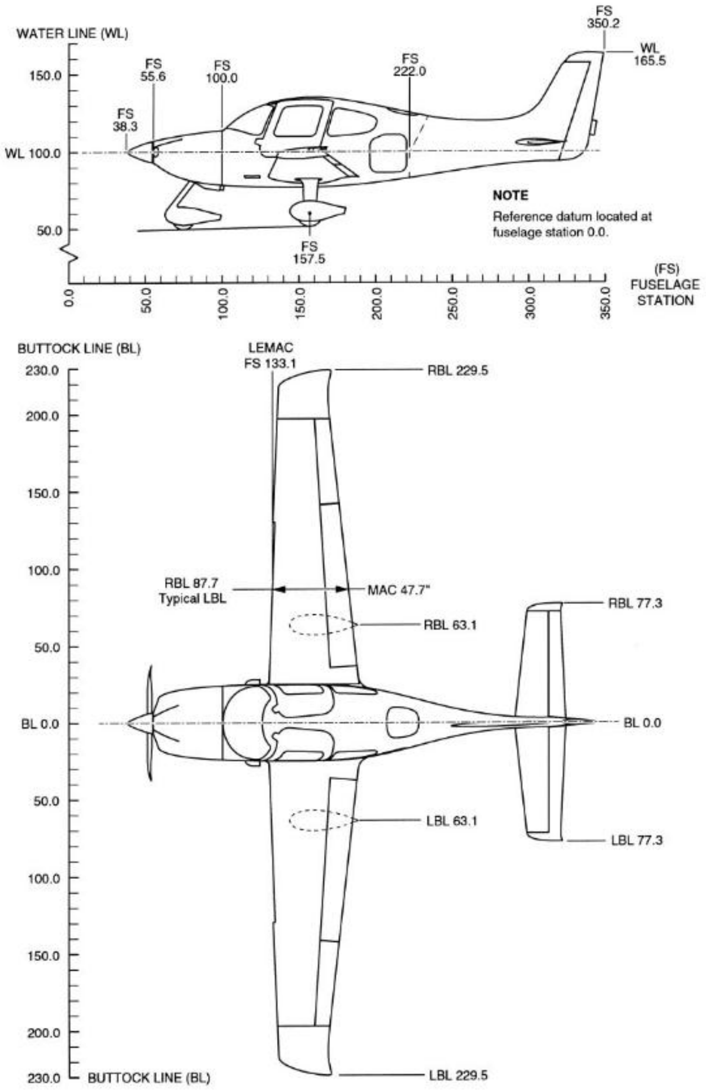


Figure 3-7 Locations of the components in terms of airplane coordinate system

3.1.7 Axis Transformations

There exist some axis transformations which are important in terms of flight dynamics modeling, flight control and spatial navigation.

North-East-Down and Body Coordinate Systems

For flight dynamics modeling kinematical relationship between NED frame and Body frame is crucial. Total accelerations are calculated in body coordinate system and integrated into body velocities. These velocities are then transformed into NED frame for simulation, control and navigation purposes. Inverse transformation is required to describe a trim condition given in NED coordinates with respect to body coordinates. As described before, the relationship between these two frames are represented by Euler angles and the transformation matrix is obtained by rotating the NED axis with the order Z-Y-X.

Rotation about Z axis for ψ angle is defined by the matrix:

$$R_{\psi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\psi & sin\psi \\ 0 & -sin\psi & cos\psi \end{bmatrix} \qquad 3\text{-}1$$

Similarly, rotation about Y axis for θ degrees:

$$R_{\theta} = \begin{bmatrix} cos\theta & 0 & -sin\theta \\ 0 & 1 & 0 \\ sin\theta & 0 & cos\theta \end{bmatrix} \qquad 3\text{-}2$$

Finally, rotation about X axis by an angle of ϕ:

$$R_{\phi} = \begin{bmatrix} cos\phi & sin\phi & 0 \\ -sin\phi & cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad 3\text{-}3$$

Transformation from NED frame to Body frame can be defined as the multiplication of these matrices with the order Z-Y-X as:

$$X_B = R(\psi)R(\theta)R(\phi)\, X_{NED}$$

$$G = \begin{bmatrix} cos\theta cos\phi & cos\theta sin\phi & -sin\theta \\ -cos\psi sin\phi + sin\psi sin\theta cos\phi & cos\psi cos\phi + sin\psi sin\theta sin\phi & sin\psi cos\theta \\ sin\psi sin\phi + cos\psi sin\theta cos\phi & -sin\psi cos\phi + cos\psi sin\theta sin\phi & \cos\psi cos\theta \end{bmatrix}$$

$$X_B = GX_{NED} \qquad 3\text{-}4$$

Since the transformation in 3-4 is orthonormal, its inverse is equal to its transpose; therefore following transformation is also valid.

$$X_{NED} = [R(\psi)R(\theta)R(\phi)]^T X_B \qquad 3\text{-}5$$

NED and Geodetic Coordinate Systems

In simulation visualization and spatial navigation, the relationship between NED and Geodetic coordinate systems is absolutely necessary. Simulation models calculates the body velocities firstly and transforms them into vehicle carried NED coordinate system. Afterwards, these velocities are integrated to obtain positions in NED frame which are represented by X_E, Y_E, Z_E. These positions represent the position of the aircraft with respect the initial point in vehicle carried NED frame. However, since aerospace navigation and simulation environments require latitude, longitude and altitude a transformation is required into Geodetic Coordinate System.

The derivatives of longitude (λ) , latitude (φ), and altitude (h) are defined as follows [32];

$$\dot{\lambda} = \frac{V_{east}}{(N_E + h)cos\varphi}$$

$$\dot{\varphi} = \frac{V_{north}}{M_E + h}$$

$$\dot{h} = -V_{down}$$

where;

$$R_{EA} = 6{,}378{,}137.0\ m,$$

$$f = \frac{1}{298.257223563}, \tag{3-6}$$

$$R_{EB} = R_{EA}(1 - f) = 6{,}356{,}752.0\ m,$$

$$e = \frac{\sqrt{R_{EA}^2 - R_{EB}^2}}{R_{EA}} = 0.08181919,$$

$$M_E = \frac{R_{EA}(1 - e^2)}{(1 - e^2 \sin^2 \phi)^{\frac{3}{2}}}$$

$$N_E = \frac{R_{EA}}{\sqrt{1 - e^2 \sin^2 \phi}}$$

Parameters defined above are;

- the major radius of Earth R_{Ea}
- the minor radius of Earth R_{Eb}
- the factor of flattening f
- the first eccentricity e
- the meridian radius of curvature M_E
- the prime vertical radius of curvature N_E

Above parameters are obtained from World Geodetic System (WGS) 84 model [33]. Integrating the above derivatives gives the position in terms of geodetic coordinate system.

3.2 Rotor Model

Rotor system is the most significant part of a rotorcraft which produces the required lift force in order to make the rotorcraft stay and move in the air. Therefore, the main source of aerodynamic forces acting on the rotorcraft is the rotating rotor.

Rotor system consists of rotor blades, hub and mast as shown in Figure 3-8. The mast is a metal cylindrical shaft which extends upwards from and is driven and sometimes supported by transmission [34]. Rotor hub is the point of attachment for blades and placed at the top of the mast. Classification of the rotor systems are done according to how the rotor is connected and move with respect to hub. These classifications include articulated, semi-rigid, or rigid or combinations of these. In this code fully articulated type of rotor hub is modeled, however rigid or semi rigid types of rotors are able to be modeled by defining an effective hinge offset [35]. Effective hinge offset is used to describe the available moment capability of rotor at the rotor hub. Therefore, hingeless rotors may be simulated by defining a proper effective hinge offset in this code.

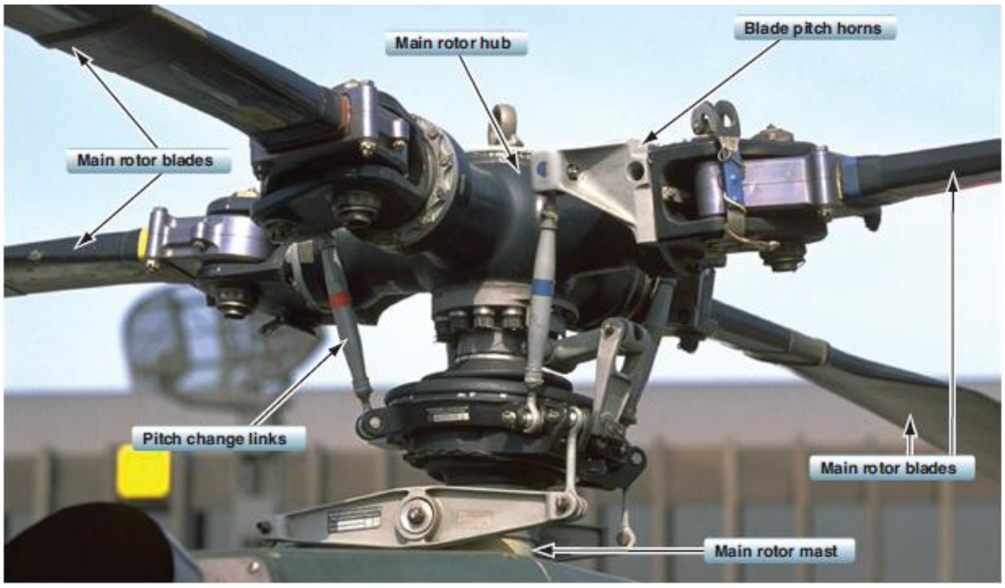


Figure 3-8 Four bladed hingeless main rotor system

For conventional helicopters there are two types of rotors which are main rotor and tail rotor. Tail rotor is also called as anti-torque device [36] since it is used for balancing the torque generated by main rotor. Tail rotor may either be a pusher configuration or a puller configuration. It is connected to main transmission system by a drive shaft and a gearbox placed at the end of the tail boom.

While it is generally used as a main rotor and tail rotor, different configurations exist such as tilt-rotor, tandem rotor, co-axial rotor and compound configurations.

Aerodynamic and dynamic properties of the rotor have a significant effect on rotorcraft performance. This thesis includes aerodynamics and dynamics modeling of a rotor component. Rotor component is modeled in detail. Nonlinear spanwise distribution of geometric properties such as twist, chord, taper, airfoil type, and sweep are available for users to define. Moreover rotor hub properties such as hinge offset, flap pitch coupling δ_3, flap and lag springs and dampers are modeled and implemented into the code.

Aerodynamics includes inflow and wake model, and dynamics includes flapping and lagging motion. Due to the necessity of being generic rotor model, all axes and transformations are applicable to any type of rotor such as CW or CCW rotating rotor, and pusher or puller type of rotor.

3.2.1 Coordinate Systems

There exists different type of axis systems for components and several axis transformations are required before calculating the total forces produced by each component. Figure 3-9 depicts the main frame body axis, rotor hub axis, and rotor shaft axis system. Transformations are done successively for each translational and angular degree of freedom. Body axis parameters are first transformed into rotor hub axis and then transformed into rotor shaft axis by using 3-7. Note that rotor hub and shaft axes are non-rotating axis systems. There exists lateral and longitudinal shaft tilt angle between hub axis and shaft axis.

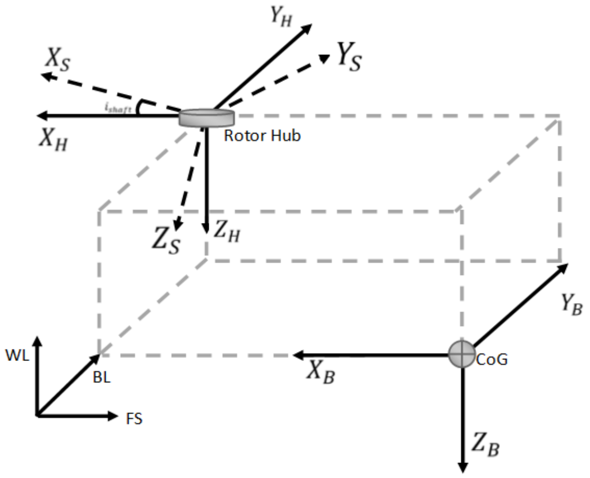


Figure 3-9 Representative figure of Body axes and Rotor Shaft Axes

$$T_{S/B} = \begin{bmatrix} \cos i_\theta & 0 & -\sin i_\theta \\ \sin i_\theta \sin i_\phi & \cos i_\phi & \cos i_\theta \sin i_\phi \\ \sin i_\theta \cos i_\phi & -\sin i_\phi & \cos i_\theta \cos i_\phi \end{bmatrix} \qquad 3\text{-}7$$

where angles i_θ and i_ϕ are the Euler angles from hub frame to rotor shaft frame. Body translational accelerations, angular accelerations, translational velocities and angular velocities are transformed into shaft axis firstly.

After obtaining the parameters in non-rotating shaft axis, they are required to be transformed into rotating axes. Rotating axes system for rotor model is defined in Figure 3-10.

Axis parameters are transformed into rotating shaft axis which is denoted as subscript s' by rotating about Z_S by $(\pi/2\ -\psi)$ degrees. In this axis, relative rotating air velocities for each blade can be obtained. The rotation matrix becomes as follows;

$$T_{s'/s} = \begin{bmatrix} \sin\psi & \cos\psi & 0 \\ -\cos\psi & \sin\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad 3\text{-}8$$

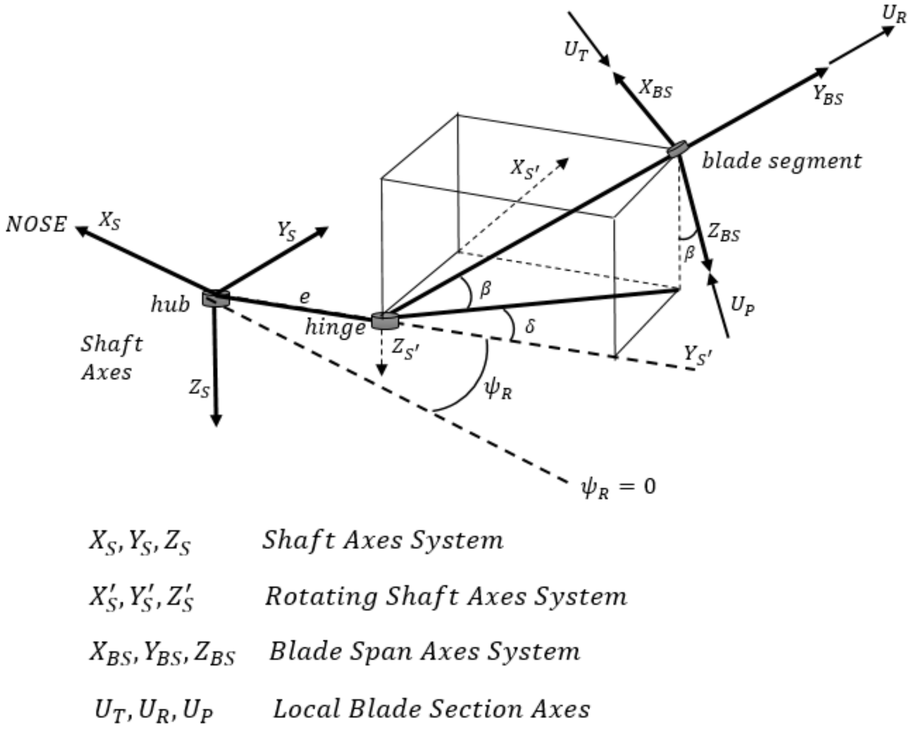

U_T
U_R
X_{BS}
Y_{BS}
$X_{S'}$
blade segment
NOSE
X_S
Y_S
Z_{BS}
β
hub
e
hinge
δ
U_P
Shaft Axes
$Z_{S'}$
$Y_{S'}$
Z_S
ψ_R
$\psi_R = 0$
X_S, Y_S, Z_S Shaft Axes System
X_S', Y_S', Z_S' Rotating Shaft Axes System
X_{BS}, Y_{BS}, Z_{BS} Blade Span Axes System
U_T, U_R, U_P Local Blade Section Axes

Figure 3-10 Shaft axes to Rotating Blade Span Axes Transformation

In order to obtain local air velocities in blade span, flapping degree of freedom should also be taken into account. For blade span axes, firstly δ rotation is required about $Z_{S'}$ axis with the matrix given in 3-9 and afterwards $-\beta$ rotation about rotated X axis is required as given in 3-10.

$$T_{e/\delta} = \begin{bmatrix} \cos\delta & sin\delta & 0 \\ -\sin\delta & \cos\delta & 0 \\ 0 & 0 & 1 \end{bmatrix} T_{\delta/e} = \begin{bmatrix} \cos\delta & -sin\delta & 0 \\ \sin\delta & \cos\delta & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad 3\text{-}9$$

$$T_{\beta/\delta} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\beta & -sin\beta \\ 0 & sin\beta & cos\beta \end{bmatrix} T_{\delta/\beta} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\beta & sin\beta \\ 0 & -sin\beta & cos\beta \end{bmatrix} \qquad 3\text{-}10$$

Blade span axes system which is denoted by subscript BS defines the local blade segment reference axis system. By using this axis system local parameters can be defined and calculated for each blade segment.

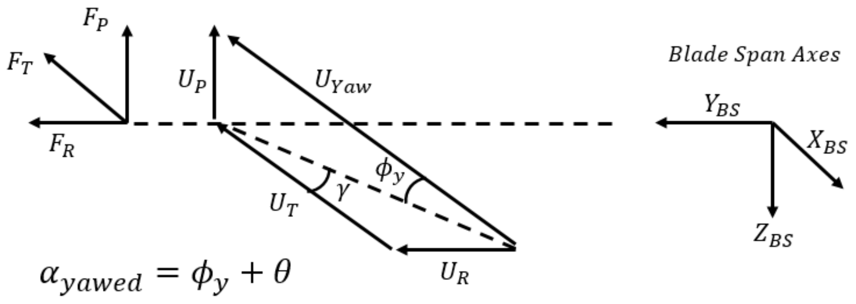


Figure 3-11 Definition of local blade section axes, angle of attack and blade pitch

In Figure 3-11 local blade section axes are shown. Sign conventions and reference planes of blade section pitch angle θ, yawed flow angle γ, and inflow angle ϕ_Y and total angle of attack α_{yawed} including three dimensional components of relative air velocity vector is described.

The airspeed axis system is the axis system is located in each blade section which defines the relative airspeed components as tangential U_T, radial U_R and perpendicular U_P. Tangential component defines the coordinate axis which is parallel to the direction of rotation of blade section, perpendicular component is perpendicular to that axis and is parallel to blade downwash, and radial axis defines the outboard flow along the rotor blade. This axis system is obtained by rotating π degrees about y_{BS} axis.

$$T_{air/BS} = -1 \times \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \qquad 3\text{-}11$$

3.2.2 Rotor Object Hierarchy

Rotor component is also modeled by utilizing OOP principles. Each individual component of a rotor is modeled separately. Beneath the rotor object, there exists Hub, Blade and Inflow sub-classes.

Rotor is modeled by initiating from the smallest part of rotor which is blade segment. Blade segment having class name as **BlSegment** holds the information of local air velocities, airfoil data, local chord, local sweep, local taper and twist, local aerodynamic and inertial forces and moments. Therefore, each section can be defined with different chord length, sweep angle, airfoil shape and twist angle.

Besides between each location where properties are defined in input XML file, there exists a linear interpolation routine for blade-wise distribution of blade segments. Therefore, blade can be defined as any spanwise distribution of chord length, sweep angle, airfoil shape and twist angle.

At the top of the **BlSegment** there exists **Blade** object. Rotor blade is composed by these blade segment components. Blade object has the information of flapping and lagging related angles and rates, rotational speed, aerodynamic and inertial integrated forces and moments. **Blade** object includes the information for a blade along the rotor disk. Therefore, azimuth-wise distribution of blade properties is available at each time in this object.

Another object is the **Hub** which includes hinge offset, flap hinge spring and damper related parameters, shaft tilt angle, pre-cone angle, and flap-pitch coupling. Moreover, hub velocities, forces and moments are also stored in the Hub object.

Another object which is required by a rotor model is the **Inflow** class. This class includes the mathematical models representing the rotor downwash. In **Inflow** object, algorithms and parameters related to rotor downwash calculation are included. Induced velocity distribution is calculated by Inflow object and the data is passes to the blade element routine. Pitt Peters' three state dynamic inflow [37] formulation and Peter's He finite state dynamic inflow model are included in this object [10], [38]. Class diagram of the inflow model is given in Figure 3-12.

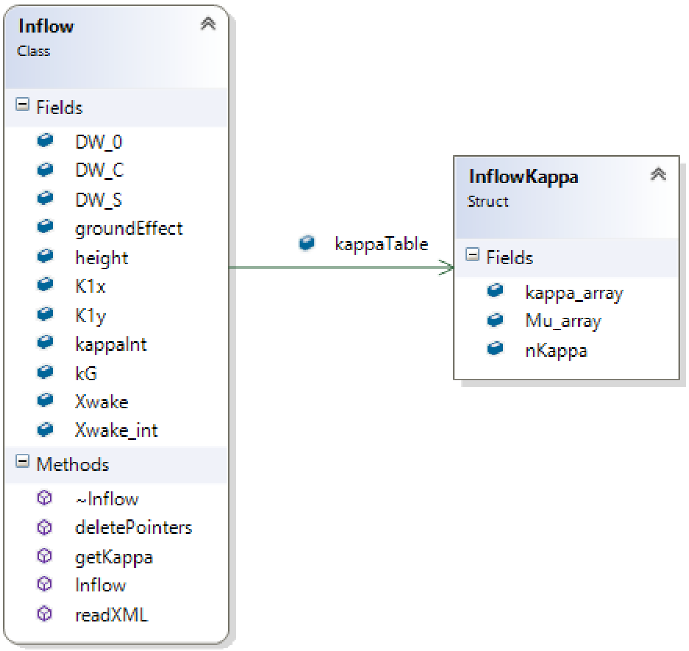
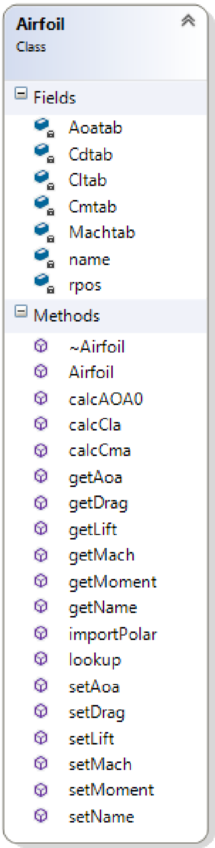


Figure 3-12 Class diagram of Inflow Object

Hub together with blades and inflow composes the **Rotor** which includes all blades, hub information and inflow models. Rotor object gathers information from different objects. One of them is **Airfoil** object which hold aerodynamic database including lift, drag, and moment coefficient databases, related interpolation and extrapolation functions, and instantaneous aerodynamic coefficients. Airfoil class is used by any object requires airfoil data. Therefore, it is used by Rotor, Propeller and Wing objects. Class diagram of airfoil and representative figure is given in Figure 3-13.



Airfoil
Class

Fields

- Aoatab
- Cdtab
- Cltab
- Cmtab
- Machtab
- name
- rpos

Methods

- ~Airfoil
- Airfoil
- calcAOA0
- calcCla
- calcCma
- getAoa
- getDrag
- getLift
- getMach
- getMoment
- getName
- importPolar
- lookup
- setAoa
- setDrag
- setLift
- setMach
- setMoment
- setName

Figure 3-13 Class diagram

Rotor object holds an input set as collective, lateral and longitudinal cyclic inputs for a given standard main rotor. For tail rotor, input settings can be arranged to lock the cyclic inputs to enable only the collective input. Moreover, for tandem, and multiple rotor configurations rotor controls can be coupled by a **control matrix** in order to have a converged trim solution for a given control definition. Other types of available inputs are RPM, longitudinal and lateral tilt control for airplanes and tilting rotor air vehicles.

Rotor object degrees of freedom are composed of flapping and lagging dynamics states, inflow states, and rotational motion states. If some of them are selected as non-active, then the number of states is automatically decreases in the generic trim algorithm.

Rotor object outputs including 6-DOF aerodynamic and inertial rotor forces and moments, rotor inflow, aerodynamic thrust and torque values. Rotor derived properties and disk distributions are dumped out in **Output** class. Following chapters provides detailed information about the possible set of inputs and outputs related to a rotor object.

3.2.3 Blade Element Model

The Blade Element Theory (BET) is one of the most widely used methodologies for predicting rotor aerodynamics since it includes azimuthal and radial distributions of aerodynamic loads over the rotor disk [36]. BET assumes each blade segment as a quasi-steady 2-D airfoil in order to represent aerodynamic loads since the aspect ratio of a rotor blade is relatively higher and 2-D airfoil assumption is applicable to the inner span-wise parts of the blade. Tip losses, root losses and other empirical correction parameters are applicable to simulate 3-D effects on the blade for momentum theory based inflow models. Blade element analysis is applicable to vortex methods since the blade is discretized into pieces which can be used as bound vortex sources. Integrated airloads on each radial blade segment over the rotor disk plane provides the rotor 6-DOF forces and moments on the rotor hub. The architecture of BET enables the integration of flapping and lagging dynamics, finite state dynamic inflow models, dynamic stall models and prescribed/free wake models.

Total forces and moments acting on the blade are derived by using total acceleration and velocity components on the blade together with control inputs. Accelerations are made up of whole body motion and motion of the blade. Velocity components are estimated from body velocities, wind velocities, rotor inflow velocities and blade motion.

Firstly, total hub accelerations, velocities and angles are calculated by translating the total parameters about the center of gravity of the vehicle to the hub coordinate system as illustrated in Figure 3-9. [39] Gravitational accelerations are represented in body frame with the following transformation from gravitational *frame* to aircraft body frame;

$$\begin{aligned} g_x &= g\sin\theta \\ g_y &= -g\cos\theta\sin\phi \\ g_z &= -gcos\,\theta\cos\phi \end{aligned} \qquad 3\text{-}12$$

Body accelerations at the rotor hub are;

$$\begin{aligned} \dot{U}_{HUB} &= \dot{u} - rv + qw - \Delta X(q^2 + r^2) + \\ &\Delta Y(pq - \dot{r}) + \Delta Z(pr + \dot{q}) + g_z \\ \dot{V}_{HUB} &= \dot{v} - pw + ru + \Delta X(pq + \dot{r}) - \\ &\Delta Y(p^2 + r^2) + \Delta Z(qr + \dot{p}) + g_y \\ \dot{W}_{HUB} &= \dot{w} + pv - qu + \Delta X(pr - \dot{q}) + \\ &\Delta Y(qr + \dot{p}) - \Delta Z(p^2 + q^2) + g_z \end{aligned} \qquad 3\text{-}13$$

where $\Delta X, \Delta Y, \Delta Z$ are stationline, buttline, and waterline distances of rotor hub from center of gravity of aircraft.

Translational body velocities at the rotor hub are calculated as in 3-14.

$$\begin{aligned} U_{HUB} &= u - r\Delta Y + q\Delta Z \\ V_{HUB} &= v - p\Delta Z + r\Delta X \\ W_{HUB} &= w + p\Delta Y - q\Delta X \end{aligned} \qquad 3\text{-}14$$

The transformation in 3-7 is applied to transform the translational accelerations, velocities, angular accelerations and rates from the rotor hub body axis to rotor shaft axis.

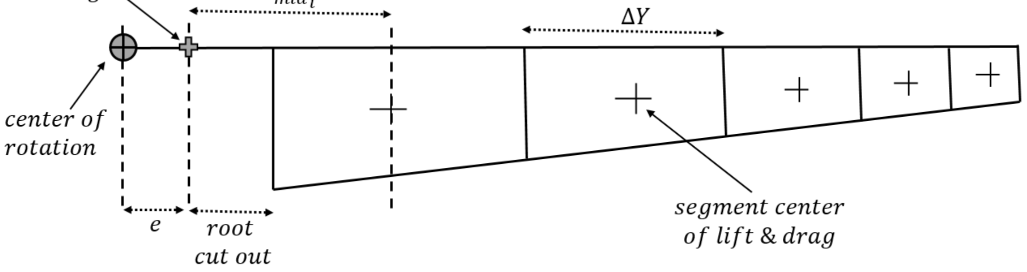


Figure 3-14 Figure showing the blade segments, root cut out and hinge offset

Each blade is discretized into blade segments (Figure 3-14) by using the equal annuli rule which allows the number of segments to be minimized and distributes the blade segments towards the higher dynamic pressure area which is the blade tips.

Figure 3-11 shows a sketch of a representative blade element, blade segment velocities, related angles and aerodynamic forces and moments. U_T represents the tangential flow which mainly includes the component of forward velocity and rotational speed, U_P represents the perpendicular flow in which induced velocity, axial velocity and relative velocity due to flapping motion exists, U_R symbolizes the span-wise air velocity component which includes the drag force especially in forward flight. ϕ angle symbolizes the inflow angle and α depicts the angle of attack of the blade section. Lift and drag forces are perpendicular to the freestream component and F_x and F_z are the forces with respect to the no-feathering plane as given in Figure 3-15.

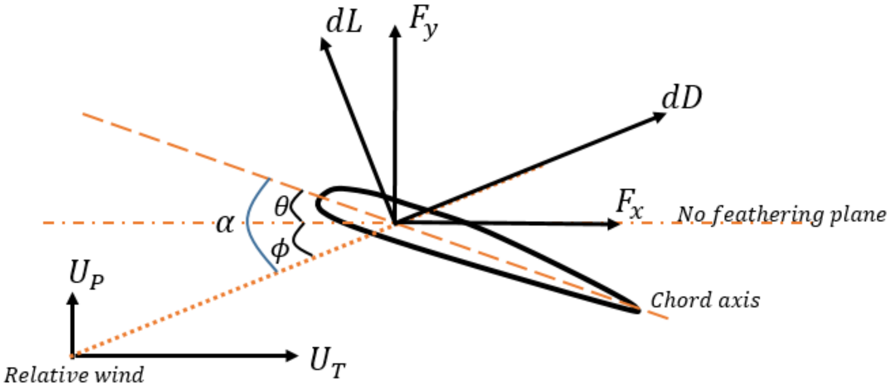


Figure 3-15 Figure showing air velocities, forces and angles in a blade segment

After moving the airframe accelerations and velocities to the rotor hub, they are required to be transformed to rotating shaft axis, then blade hinge point, and finally to the blade segment. Transformation from hub axis to shaft axis is given in 3-7.After they are moved into rotor shaft axis, they all should be moved to rotor blade hinge. Translational velocities are transformed into the rotating shaft frame with the equation given in Figure 3-14 where subscript e stands for hinge offset.

$$\begin{bmatrix} U_e \\ V_e \\ W_e \end{bmatrix}_{tr} = \begin{bmatrix} \sin\psi & \cos\psi & 0 \\ -\cos\psi & \sin\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} U_S \\ V_S \\ W_S \end{bmatrix} \qquad 3\text{-}15$$

Angular velocities at the rotor shaft axis create a translational velocity at the blade hinge. Taking the cross product of angular rates, translational velocity at a distance $\vec{r}$ is calculated for the hinge point.

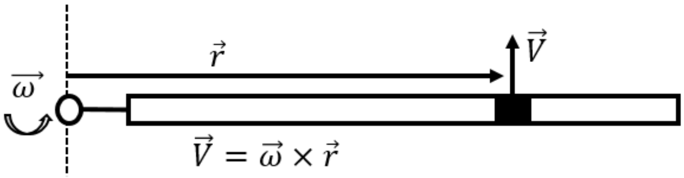


Figure 3-16 Figure representing the cross product of angular rates on rotating rotor shaft axis

$$\vec{\omega} = \begin{bmatrix} \sin\psi & \cos\psi & 0 \\ -\cos\psi & \sin\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_S \\ q_S \\ r_S \end{bmatrix} \; \vec{r} = \begin{bmatrix} 0 \\ r_e \cos\beta_p \\ -r_e \sin\beta_p \end{bmatrix} \qquad 3\text{-}16$$

r_{BS} represents the position vector for the blade segment and r_e is used as the position vector for the rotor blade hinge offset. Cross product represented in Figure 3-16 for hinge point is calculated as given in 3-17.

$$\overrightarrow{\omega_s} \times \overrightarrow{r_e} = \begin{bmatrix} (\tilde{\Omega} - r_s) r_e cos\beta_p + r_e\, sin\beta_p (p_s cos\psi - q_s sin\psi) \\ r_e\, sin\beta_p\, (q_s cos\psi + p_s sin\psi) \\ r_e\, cos\beta_p\, (p_s\, sin\,\psi + q_s\, cos\,\psi) \end{bmatrix} \qquad 3\text{-}17$$

Firstly, translational velocities due to angular rates are calculated at hinge offset (3-17) and directly appended into the translational velocity vector (3-15) as shown in equation 3-18.

$$\begin{bmatrix} U_e \\ V_e \\ W_e \end{bmatrix}_{tot} = \begin{bmatrix} U_e \\ V_e \\ W_e \end{bmatrix}_{tr} + \overrightarrow{\omega_s} \times \overrightarrow{r_e} \qquad 3\text{-}18$$

Afterwards, beyond the hinge point flapping and lagging rates are taken into account. Total velocities at a blade segment is found similarly as in equation 3-18 by taking the cross product of the rotational velocities at blade hinge with the position vector of blade segment $\vec{r}_{BS}$.

Lagging rate $\dot{\delta}$ and flapping rate $\dot{\beta}$ are added into the Z and X components of the total translational velocity respectively by taking the proper cross products due to blade segment position. Lag angle transformation is done (Equation 3-9) if lagging degree of freedom is active. Flapping rate is added into the cross product after the flap-wise axis rotation is employed (Equation 3-10). Lastly, transformation from blade segment velocities to air velocities as given in equation 3-11 is implemented.

Finally, all components of the rates and velocities are transformed and summed up for a blade segment in the blade element methodology. Translational and angular rates with the flapping and lagging rates are included in the total air velocity calculations. Last velocity component remaining is the rotor induced velocity (downwash). Induced velocity vector is defined as given in Figure 3-17 which requires a flapping degree of freedom rotation to be defined in the blade segment.

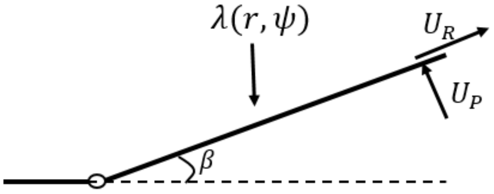


Figure 3-17 Figure representing the relationship between U_P, U_R and downwash velocity

This velocity contribution of downwash is a function of blade radial position and rotor disk azimuth. In Figure 3-17 definitions of the tangential and perpendicular air velocities in local blade section axis. Therefore, downwash contribution to the blade segment air velocities are represented as follows;

$$\begin{aligned} U_{P_\lambda} &= -\lambda(r,\psi)\cos\beta \\ U_{R_\lambda} &= -\lambda(r,\psi)\sin\beta \\ U_{T_\lambda} &= 0 \end{aligned} \qquad 3\text{-}19$$

In order to obtain the total velocity at any blade segment on the blade, translational and rotational velocities on the blade hinge should be transformed, rotated and translated properly. After hinge offset there are two degrees of freedom which are lag angle and flap angle respectively. Lag degree of freedom is defined as a positive rotation for Z axis for CCW rotating rotors and flap degree of freedom is defined as negative rotation at X axis. Translational components due to the cross product of angular velocities and radial position vectors are appended and required axis rotations are applied. Required transformations and inverse transformations are given in equations 3-9 and 3-10.

The position vector at the blade segment is defined in the blade span axes system which is already rotated for lag and flap degrees of freedom.

$$\begin{aligned} \boldsymbol{r_{mid}} &= \begin{bmatrix} 0 \\ y_{mid} \\ 0 \end{bmatrix} \\ \boldsymbol{r_{mid,\delta}} &= T_{\delta/\beta}\begin{bmatrix} 0 \\ y_{mid} \\ 0 \end{bmatrix}, \\ \boldsymbol{r_{mid,e}} &= T_{e/\delta}T_{\delta/\beta}\begin{bmatrix} 0 \\ y_{mid} \\ 0 \end{bmatrix} \end{aligned} \qquad 3\text{-}20$$

Lagging rate produces an extra translational velocity component on the blade segment when taking the cross product with the radial position vector. Total blade segment velocities after the lag rotation are defined as:

$$\boldsymbol{V_\delta} = T_{\delta/e}\boldsymbol{V_e} + \left(T_{\delta/e}\boldsymbol{\omega} \times \boldsymbol{r_{mid,\delta}}\right) + \left(\begin{bmatrix} 0 \\ 0 \\ \dot{\delta} \end{bmatrix} \times \boldsymbol{r_{mid,\delta}}\right) \qquad 3\text{-}21$$

Similarly, flapping rate creates a perpendicular component of translational velocity on the blade segment. Total blade segment velocities after the flap rotation which are defined in blade span axes are obtained as:

$$\boldsymbol{V_\beta} = T_{\beta/\delta}\boldsymbol{V_\delta} + \left(\begin{bmatrix}\dot{\beta}\\0\\0\end{bmatrix} \times \boldsymbol{r_{mid}}\right) \qquad \text{3-22}$$

Finally, transformation from blade span axes to blade segment relative air velocities U_T, U_R, U_P is implemented as:

$$\boldsymbol{V_{air}} = \begin{bmatrix}U_T\\U_R\\U_P\end{bmatrix} = \begin{bmatrix}-1 & 0 & 0\\0 & 1 & 0\\0 & 0 & -1\end{bmatrix}\boldsymbol{V_\beta} \qquad \text{3-23}$$

Total blade segment velocity and Mach number is then defined as:

$$\begin{aligned} U_{TOT} &= \sqrt{U_T^2 + U_R^2 + U_P^2} \\ M &= \frac{\Omega R}{V_{sound}}\sqrt{U_P^2 + U_T^2} \end{aligned} \qquad \text{3-24}$$

Defining a yawed flow angle to be used for yawed flow corrections as described in Figure 3-11:

$$cos\gamma = \frac{|U_T|}{\sqrt{U_T^2 + U_R^2}} \qquad \text{3-25}$$

Angle of attack is defined including the yawed angle of attack effects as described in Figure 3-11:

$$\alpha(r,\psi) = \tan^{-1}\left\{\frac{[U_T\tan\theta + U_P]cos|\gamma|}{[U_T - U_P\tan\theta\cos^2\gamma]}\right\} \qquad \text{3-26}$$

Then angle of attack and Mach number are used to interpolate the aerodynamic coefficient of a given blade section. Tip loss, sweep and stall delay effects are appended into the coefficients if required. Aerodynamic forces acting on a single blade element is calculated as:

$$
\begin{aligned}
dF_T &= \frac{1}{2}\rho\Omega^2R^3(c\Delta r)U_{TOT}\left[C_l\frac{U_T}{|\cos\gamma|}+C_dU_P\right] \\
dF_P &= \frac{1}{2}\rho\Omega^2R^3(c\Delta r)U_{TOT}[C_dU_T-C_lU_P|\cos\gamma|] \\
dF_R &= \frac{1}{2}\rho\Omega^2R^3(c\Delta r)U_{TOT}\left[C_d-C_l\frac{U_P}{U_T}|\cos\gamma|\right]U_R
\end{aligned}
\qquad 3\text{-}27
$$

Sectional forces are then integrated along the blade and total blade forces are obtained:

$$
\begin{aligned}
F_T &= \sum_{i=0}^{N_{rad}} dF_T(i) \\
F_P &= \sum_{i=0}^{N_{rad}} dF_P(i) \\
F_R &= \sum_{i=0}^{N_{rad}} dF_R(i)
\end{aligned}
\qquad 3\text{-}28
$$

Flapping and lagging aerodynamic moments are obtained by taking the moments of each blade segment with respect to hinge point as given in 3-29. These moments are directly inserted into the flapping and lagging dynamics equations as external moments.

$$
\begin{aligned}
M_{FLAP} &= R\sum_{i=0}^{N_{rad}} dF_P(i)\,y_{mid}(i) \\
M_{LAG} &= R\sum_{i=0}^{N_{rad}} dF_T(i)\,y_{mid}(i)
\end{aligned}
\qquad 3\text{-}29
$$

Tip path plane forces and moments are calculated to feed into the inflow models. Aerodynamic roll and pitch moments are defined as:

$$M_{Xaero} = -\frac{N_{blades}}{N_{azimuth}} \sum_{i=0}^{N_{rad}} dF_P(i)\, y_{mid}(i)\ \ sin\,\psi\,(i)$$

$$M_{Yaero} = -\frac{N_{blades}}{N_{azimuth}} \sum_{i=0}^{N_{rad}} dF_P(i)\, y_{mid}(i)\ \ cos\,\psi\,(i) \qquad 3\text{-}30$$

Rotor shaft forces and moments are obtained from the total blade forces as:

$$\begin{bmatrix} F_{XA} \\ F_{YA} \\ F_{ZA} \end{bmatrix} = T_{\delta/e} T_{\delta/\beta} \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} F_T \\ F_R \\ F_P \end{bmatrix} \qquad 3\text{-}31$$

Aerodynamic thrust force which is perpendicular to rotor hub is defined as:

$$T_{Zaero} = -\frac{N_{blades}}{N_{azimuth}} \sum_{i=0}^{N_{rad}} F_{ZA}(i) \qquad 3\text{-}32$$

Inertia shears at the hinge is directly obtained from the reference [39], and then total forces in rotating frame are obtained by summing up the aerodynamic forces and inertial forces:

$$\begin{bmatrix} F_{XTot} \\ F_{YTot} \\ F_{ZTot} \end{bmatrix} = \begin{bmatrix} F_{XA} \\ F_{YA} \\ F_{ZA} \end{bmatrix} + \begin{bmatrix} F_{XI} \\ F_{YI} \\ F_{ZI} \end{bmatrix} \qquad 3\text{-}33$$

Total rotor forces at non-rotating shaft axes which exclude the total blade mass obtained by transforming from the rotating shaft axes to non-rotating shaft axes as:

$$F_{X_{HUB}} = \frac{N_{blades}}{N_{azimuth}} \sum_{i=0}^{N_{rad}} F_{YTot}(i)\cos\psi - F_{XTot}(i)\sin\psi$$

$$F_{Y_{HUB}} = \frac{N_{blades}}{N_{azimuth}} \sum_{i=0}^{N_{rad}} F_{XTot}(i)\cos\psi + F_{YTot}(i)\sin\psi \qquad \text{3-34}$$

$$F_{Z_{HUB}} = \frac{N_{blades}}{N_{azimuth}} \sum_{i=0}^{N_{rad}} F_{ZTot}(i)$$

Rotor hub moments are calculated as:

$$M_{X_{HUB}} = \frac{N_{blades}}{N_{azimuth}} \sum_{i=0}^{N_{rad}} eF_{ZTot}(i)\sin\psi$$

$$M_{Y_{HUB}} = \frac{N_{blades}}{N_{azimuth}} \sum_{i=0}^{N_{rad}} eF_{ZTot}(i)\cos\psi \qquad \text{3-35}$$

$$M_{Z_{HUB}} = -\frac{N_{blades}}{N_{azimuth}} \sum_{i=0}^{N_{rad}} eF_{XTot}(i) = Torque$$

After the determination of all the components of blade section air velocities, angle of attack and Mach number is obtained for each blade section. Lift, drag and pitching moment coefficients are calculated for each blade section. These coefficients are dimensionalized according to the dynamic pressure on each segment. Forces and moments are superimposed from the blade tip to blade root. The calculated forces and moments are transferred into the flapping and lagging dynamic loops as well as inflow model. Since there is a hinge, the moment is not translated into the hub, only the forces are translated to the rotor hub. Generated moments due to hinge offset, and distance from the center of gravity is taken into account while translating the total forces and moments to the vehicle center of gravity. Converged inflow, flapping and lagging dynamics provides total forces and moments acting on the center of gravity of the aircraft, and they are calculated and stored at each iteration to be used on the 6-DOF dynamics model.

Individual Blade Coordinates (IBC) and Multi-Blade Coordinates (MBC)

In rotor dynamics some properties which are identified along the tip path plane (TPP) may be defined in two notations. One is individual blade coordinates and the other is multi-blade coordinates. Individual blade coordinates namely represent the quantities for one blade for a defined azimuth location and should be defined for each blade having different azimuth locations. Each blade should keep its information and have to be processed and integrated in each time step. On the other hand, multi-blade coordinates are composed of a harmonic representation of individual blade properties such as inflow, flapping dynamics, and lead-lag dynamics. The properties are symbolized by Fourier series which provides an acceptable representation of whole tip path plane. With this method properties can be determined by any given ψ locations in rotor disk. Therefore, in some calculations this perspective provides a clear understanding.

Transformation of IBC to MBC are as follows [35]:

$$Q_0 = \frac{1}{N_b}\sum_{i=1}^{N_b} Q_i \qquad 3\text{-}36$$

$$Q_{N/2} = \tfrac{1}{N_b}\textstyle\sum_{i=1}^{N_b} Q_i(-1)^i \qquad 3\text{-}37$$

$$Q_{jc} = \frac{2}{N_b}\sum_{i=1}^{N_b} Q_i cos\, j\psi_i \qquad 3\text{-}38$$

$$Q_{js} = \frac{1}{N_b}\sum_{i=1}^{N_b} Q_i sin\, j\psi_i \qquad 3\text{-}39$$

where i is the blade id number and j is the id number of cosine and sine harmonics. And azimuth angle for the ith blade is defined as:

$$\psi_i = [\psi - \frac{\pi}{2}(i-1)] \qquad 3\text{-}40$$

Symbol Q is a rotor property which can be related to inflow, flapping angles or rates, lagging angles or rates, forces and moments. Q_0 is the first harmonic of the multi-blade coordinates which represents the coning angle when Q is the flapping angle β, similarly it represents the mean inflow ratio for $Q = \lambda$. The coefficient $Q_{N/2}$ stands for alternating component of the Fourier series which appears when number of blades is even. Number of MBC coefficients should be equal to the blade numbers. For example for a 4 bladed rotor, MBC angles for flapping are $\beta_0, \beta_{1c}, \beta_{1s}, \beta_{N/2}$, similarly for a 5 bladed rotor they are β_0, β_{1c}, β_{1s}, β_{2c}, β_{2s} . These coefficients are clearly given in Figure 3-18 for flapping angles.

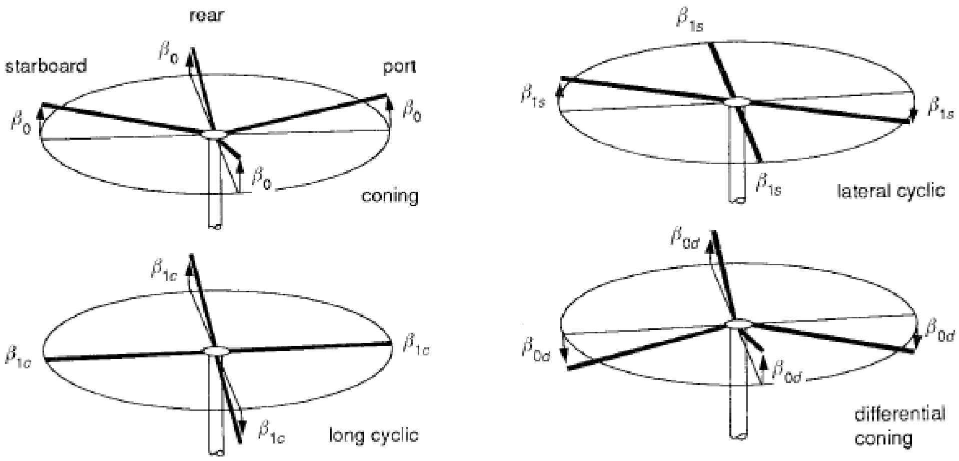


Figure 3-18 Flapping coefficients of rotor disc in MBC coordinates

Inverse transformation of MBC to IBC is done as:

$$Q_i = Q_0 + \sum_j \left(Q_{jc} \cos j\psi_i + Q_{js} \sin j\psi_i \right) + Q_{\frac{N}{2}} (-1)^i \qquad \text{3-41}$$

Where j goes from 1 to $(N_b - 1)/2$ for N_b is odd and from 1 to $(N_b - 2)/2$ for N_b is even.

In this thesis, rotor disk is discretized by also azimuth-wise and a single blade holds the calculated aerodynamic and dynamic blades for each azimuth. Flapping, lagging and inflow are distributed along the disk by Fourier series. The MBC distribution of dynamic properties are treated as dynamic states and time integration methods are applied on. After time integration, related parameters are distributed along the azimuth by the transformation given in 3-41. Therefore, although the azimuth is discretized into for example 30 pieces, number of states related to flapping angle is equal to the number of rotor blades in MBC coordinates as described above.

Since the properties are hold in a single blade for each azimuth, azimuth-wise distribution of section properties are recorded which are:

- Mach number
- Blade section air velocity components
- Blade section angle of attack, inflow angle and local pitch
- 6-DoF sectional forces and moments
- Flapping and Lagging angles and rates
- Lift, Drag and Moment coefficients

In figures Figure 3-19 and Figure 3-20 the distributions of angle of attack and Mach number are plotted for a sample helicopter main rotor from hover to high advance ratios about $\mu = 0.4$. Plotted distributions are obtained by using the Pitt Peters three state dynamic inflow model which will be described in the following chapter.

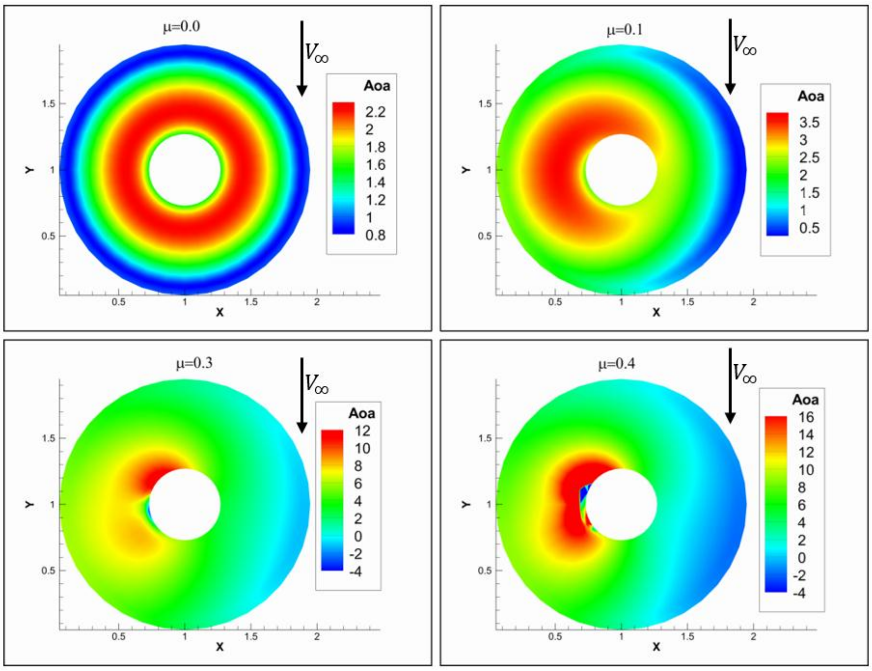


Figure 3-19 Angle of attack distribution of a sample CCW rotating rotor disk for different advance ratios

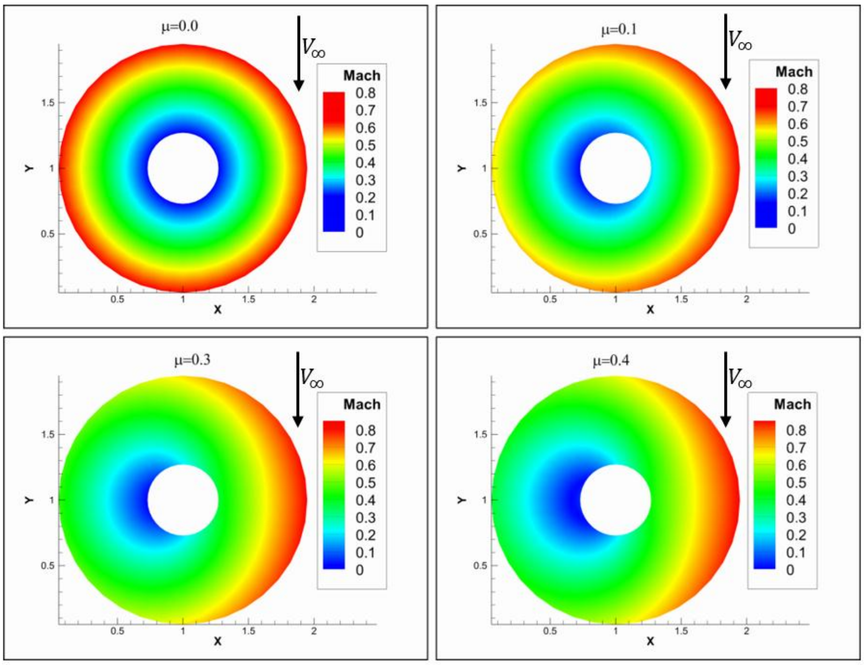


Figure 3-20 Mach number distribution of a sample CCW rotating rotor disk for different advance ratios

3.2.4 Inflow Models

In order to produce thrust, there must be a work done on the air by the rotor. Rotating rotor induces a flow field below the rotor disk which is called as rotor inflow. Inflow models are implemented into the rotor class to simulate the flow distribution onto the rotor disk. There exist several types of inflow models which search for a converged induced velocity distribution with converged disk loading distribution. Different analysis techniques exist for calculating rotor performance. Fidelity of rotor aerodynamics solution methods varies as follows:

- Classical momentum theory
- Linear inflow models
- Finite state dynamic inflow models
- Rigid wake inflow models based on lifting line theory
- Free wake inflow models based on lifting line theory
- CFD coupled actuator disk models
- CFD coupled actuator surface models
- CFD rotor solutions with moving reference

Increasing the rotor solution fidelity arises with some drawbacks one of which is basically the time cost. According to the purpose of analysis, the fidelity of rotor solution methodologies is decided. In this thesis, the purpose is to generate generic flight dynamics analysis and simulation software which is also required to give results quickly. This software is designed to analyze air vehicles quickly and provide a simulation environment which requires to be run in real time.

3.2.4.1 Pitt-Peters Inflow Models

Induced velocity distribution at the rotor disk plays a significant role in determining rotor performance and loads. Both periodic and transient blade loadings should be determined. Although the vortex wake methods propose finer representation for the rotor wake, as discussed in previous section for real-time simulations and stability analyses a finite state inflow model is favorable. One of the simplest dynamic inflow models consists of three dynamic inflow states which are;

- λ_0: uniform inflow component
- λ_{1c}: linear perturbations of the downwash (cosine)
- λ_{1s}: linear perturbations of the downwash (sine)

Downwash for a point in rotor disk is defined as;

$$\lambda(r,\psi) = \lambda_0 + \lambda_c r cos(\psi) + \lambda_s r sin(\psi) \qquad \text{3-42}$$

Pitt Peter’s dynamic inflow is represented by a first order differential equation which relates the uniform, cosine and sine components of the induced velocity with aerodynamic thrust, rolling and pitching moment. Differential equations of dynamic inflow are;

$$LM/\Omega \begin{pmatrix} \dot{\lambda_0} \\ \dot{\lambda_c} \\ \dot{\lambda_s} \end{pmatrix} + \begin{pmatrix} \lambda_0 \\ \lambda_c \\ \lambda_s \end{pmatrix} = L \begin{pmatrix} C_T \\ -C_{My} \\ C_{Mx} \end{pmatrix} \qquad \text{3-43}$$

Dynamic inflow is a set of first order differential equations composed of inflow and aerodynamic loading variables. Left hand side of the equations is the inflow state variables, and right hand side represents the aerodynamic loading coefficients in 3-43. This model is usually sufficient to observe wake effects due to low frequency blade modes and the rotorcraft flight dynamics [37]. The derivative matrix which is developed by Pitt and Peters [37] is,

$$L = \frac{1}{V_{eff}} \begin{bmatrix} \frac{1}{2} & -\frac{15\pi}{64}\sqrt{\frac{1-cos\chi}{1+cos\chi}} & 0 \\ \frac{15\pi}{64}\sqrt{\frac{1-cos\chi}{1+cos\chi}} & \frac{4\cos\chi}{1+cos\chi} & 0 \\ 0 & 0 & \frac{4\cos\chi}{1+cos\chi} \end{bmatrix} \qquad 3\text{-}44$$

With the effective velocity for the thrust terms as;

$$V_{eff} = \frac{\mu^2 + \lambda(\lambda + \lambda_i)}{\sqrt{\mu^2 + \lambda^2}} \qquad 3\text{-}45$$

And for the moment terms;

$$V_{eff} = \sqrt{\mu^2 + \lambda^2} \qquad 3\text{-}46$$

The mass matrix used in 3-43,

$$M = \begin{bmatrix} \frac{128}{75\pi} & 0 & 0 \\ 0 & \frac{64}{45\pi} & 0 \\ 0 & 0 & \frac{64}{45\pi} \end{bmatrix} \qquad 3\text{-}47$$

The values given for the mass and derivative matrices are supported by experimental data and the time lag is given by the matrix LM. The hub moments are functions of the inflow distribution which is non-uniform and it is found that the moments are reduced by a moment deficiency factor C'.

Lift deficiency factor for pitch moment;

$$C' = \frac{1}{1 + \frac{\sigma a}{8V_{eff}} \frac{2cos\chi}{1 + cos\chi}} \qquad 3\text{-}48$$

and for roll moment,

$$C' = \frac{1}{1 + \frac{\sigma a}{8V_{eff}} \frac{2}{1 + cos\chi}} \qquad 3\text{-}49$$

The wake effect reduces the aerodynamic thrust, but its influence is higher for moments due to the shed vorticity. Typical lift deficiency factors for hover is about 0.8 and for forward flight is 0.7 for thrust changes in hover; and around 0.5 for moment changes in hover. The associated deficiency factors are cited as the cause of difference between experimental data and theory.[35]

3.2.5 Flapping and Lagging Dynamics

Articulated rotors are observed in helicopters which have more than two rotor blades. In articulated rotor systems aerodynamic forces and moments produced by the pilot inputs results in a blade motion around hinges. This dependency is a crucial issue in order to understand the behavior of the response of the blade. Articulation around the flap and lead-lag hinges is the typical feature of helicopter rotors as given in Figure 3-21 and Figure 3-22. The motion is done by the help of a mechanical hinge, semi-rigid, or hingeless materials which allow the rotor blade motion.

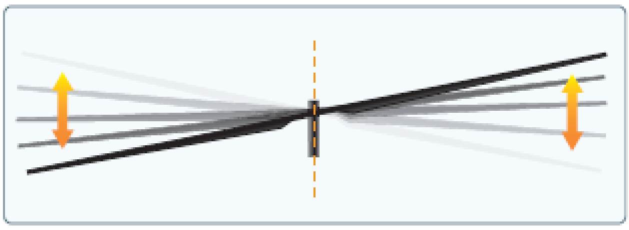


Figure 3-21 Fully articulated flapping rotor

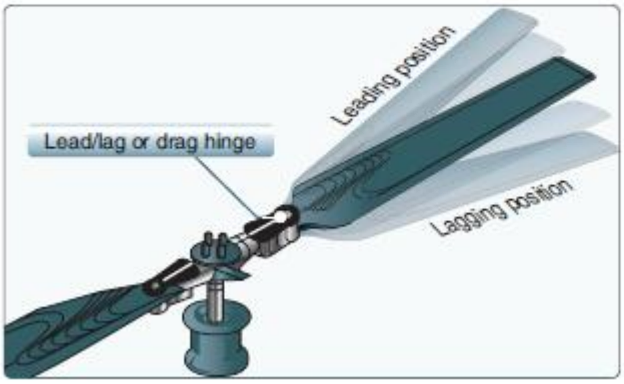


Figure 3-22 Flap and Lag hinge and Lead-Lag Motion

Because of this articulation there exists a moment capability due to the tilt of tip path plane. The line of action of integrated thrust force creates a moment on the center of gravity of the aircraft which increases the maneuverability. Flapping and lagging dynamics equations implemented in this thesis are based on Du Val’s [40] and Chen’s [41] works.

3.2.5.1 Equations of Rotor Dynamics

In this thesis a form of coupled flap-lag dynamics equations is included. The equations are derived for an articulated rotor having hinge springs and dampers with a rigid blade assumption. Lagrange method is used to derive the equations of motions. The sequence of hinges is assumed as Lag-Flap-Pitch and both flapping and lagging hinges are co-located. 6-DOF motion of air vehicle is included without any small angle assumption. Inertia dynamic terms are present in the derived equations of motion.

Articulated type rotors consist of flap, lead-lead, and pitch (feathering) hinges. There are three commonly used sequences for the hinges which are given in Figure 3-23. Selection of a sequence depends on the design experience or decision of manufacturer. For instance, Sikorsky designs generally use L-F-P sequence with co-located flapping, lagging and feathering hinges [41].

It is important to include the aircraft 6-DOF motion related parameters into the flapping and lagging dynamics equations in order to conduct a comprehensive analysis and simulation. Containing this terms in the rotor dynamics equations of motions enables to analyze stability and control properties of the air vehicle.

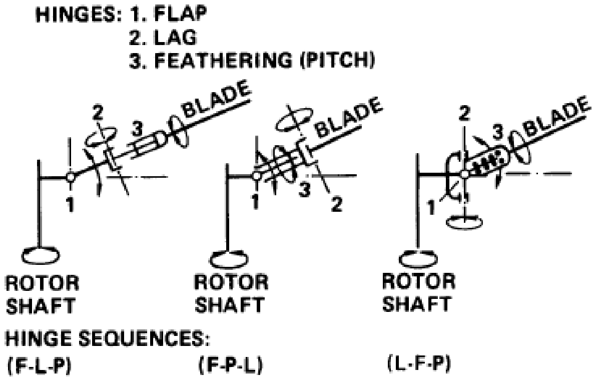


Figure 3-23 Commonly used sequences for hinges

Derived equation is based on Chen’s work [41]. Equations for flapping and lagging accelerations are as given in:

$$
\begin{aligned}
&\frac{M_\beta}{I_\beta}\{\,\sin\beta\langle -\dot{v}_x\cos(\psi-\delta)+\dot{v}_y\sin(\psi-\delta)-e_\delta\left[\dot{\Omega}sin\delta+(\Omega^2-2r\Omega)cos\delta\right]\\
&+(l_x-e_\delta cos\psi)[(\dot{r}+pq)\sin(\psi-\delta)+(q^2+r^2)\cos(\psi-\delta)]\\
&+\ \left(l_y-e_\delta sin\psi\right)[(\dot{r}-pq)\cos(\psi-\delta)-(p^2+r^2)\sin(\psi-\delta)]\\
&-l_z[(\dot{p}-qr)\sin(\psi-\delta)+(\dot{q}+pr)\cos(\psi-\delta)]]\rangle\ +cos\beta(\dot{v}_z\\
&+2\Omega e_\delta(pcos\psi-qsin\psi)-(\dot{q}-pr)(l_x-e_\delta\,cos\,\psi)+(\dot{p}+qr)\left(l_y+e_\delta sin\psi\right)\\
&-(p^2+q^2)l_z\}\}\\
&+sin\beta\left\{[(\dot{p}-qr)\sin(\psi-\delta)+(\dot{q}+pr)\cos(\psi-\delta)]sin\beta\right.\\
&\left.-\ \left\{[p\,sin(\psi-\delta)+q\cos(\psi-\delta)]^2+\ \left[r-\left(\Omega-\dot{\delta}\right)\right]^2\right\}cos\beta\right\}\\
&+cos\beta\{\ (p^2+q^2)sin\beta\\
&+\left\{2(\Omega-\delta)\left[p\cos(\psi-\delta)-q\sin\left(\psi-\dot{\delta}\right)\right]+(\dot{p}+qr)\sin(\psi-\delta)\right.\\
&\left.+(\dot{q}-pr)\cos(\psi-\delta)\right\}cos\beta\}+\frac{\Sigma M_{flap}}{I_\beta}=\frac{d^2\beta}{dt^2}
\end{aligned}
\tag{3-50}
$$

3-50 gives the flapping acceleration of an individual blade for a given azimuth angle ψ, for each blade this equation is solved once. Effects of body angular velocities and rates, translational accelerations, hinge offset and flap springs and dampers, and lagging coupled terms are included in the equation of flapping acceleration.

Similarly, lagging angle acceleration is described by the equation:

$$\begin{aligned}
&\frac{d^2\delta}{dt^2}\left(I_\beta \cos^2\beta\right) = \dot{\Omega}\left(I_\beta \cos^2\beta\right) \\
&+M_\beta \cos\beta\, \{\dot{v_x}\sin(\psi+\delta) \\
&+\dot{v_y}\cos(\psi+\delta) \\
&+e\big[(\dot{\Omega}-\dot{r})\cos\delta + (\Omega-r)^2\sin\delta - pq\cos(2\psi+\delta) - p^2\sin\psi\cos(\psi+\delta) \\
&+q^2\cos\psi\ \ \sin(\psi+\delta)\big] + l_x[(\dot{r}+pq)\cos(\psi+\delta) - (q^2+r^2)\sin(\psi+\delta)] \\
&+l_y[(pq-\dot{r})\sin(\psi+\delta) - (p^2+r^2)\cos(\psi+\delta)] \\
&-l_z[(\dot{p}-qr)\cos(\psi+\delta) - (\dot{q}+pr)\sin(\psi+\delta)]\,\} \\
&+I_\beta\cos\beta\, \Big\{\sin\beta\,\big[2\dot{\beta}(r-\Omega-\dot{\delta}) + (\dot{p}-qr)\cos(\psi+\delta) - (\dot{q}+pr)\sin(\psi+\delta)\big] \\
&-\cos\beta\, \Big\{\dot{r} + pq\cos 2(\psi+\delta) + \frac{1}{2}(p^2 \\
&-q^2)\sin 2(\psi+\delta) - 2\dot{\beta}\,[p\cos(\psi+\delta) - q\sin(\psi+\delta)]\Big\}\Big\} + \sum M_{lag}
\end{aligned}$$

3-51

Note that the positive lead-lag angle is lagging forward direction. With the equations above coupled flapping and lagging motion of the rotor blades is included in the Rotor object. These degree of freedoms are optional, the user can disable these dynamics.

3.2.5.2 Flap and Lag Motions and Integration of Acceleration Terms

Flap and lagging dynamics is modeled by firstly expressing them in the matrix form as given in the reference book “Rotorcraft Aeromechanics [30]” letting β_{rot} is equal to the vector of rotating frame flapping angles whose size is the number of blades and the vector β_{non} is the vector of multi-blade coordinates which has the length of number of blades and can be expressed as $(\beta_0, \beta_{nc}, \beta_{ns}, \beta_{N/2})^{\mathrm{T}}$;

$$\begin{aligned} \beta_{rot} &= T\beta_{non} \\ \dot{\beta}_{rot} &= T\dot{\beta}_{non} + T\beta_{non} \\ \ddot{\beta}_{rot} &= T\ddot{\beta}_{non} + 2T\dot{\beta}_{non} + T\beta_{non} \end{aligned} \tag{3-52}$$

Where m-th row of T is (where m is an integer from 1 to number of blades N);

$$t_m = (1 \quad \cos k\psi_m \quad \sin k\psi_m \quad (-1)^m) \tag{3-53}$$

This transformation is a time-varying linear transformation of angle, velocity and accelerations and can be used both for flapping dynamics and lagging dynamics. Following equations are the properties of the transformation variables;

$$\begin{aligned} DT^T T &= I \\ DT^T T &= E_1 \\ DT^T T &= E_2 = E_1^2 \end{aligned} \tag{3-54}$$

or can be defined as;

$$\begin{aligned} T^{-1} &= DT^T = S^T \\ T &= TE_1 \\ T &= TE_2 \end{aligned} \tag{3-55}$$

where;

$$D = \begin{bmatrix} 1/N & 0 & 0 & 0 \\ 0 & 2/N & 0 & 0 \\ 0 & 0 & 2/N & 0 \\ 0 & 0 & 0 & 1/N \end{bmatrix}$$

3-56

$$E_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & k\Omega & 0 \\ 0 & -k\Omega & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad E_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & -k^2\Omega^2 & 0 & 0 \\ 0 & 0 & -k^2\Omega^2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

For each azimuth step on the rotor disk flapping and lagging acceleration terms are calculated by the equations of inertial dynamics of a rotor and the distribution of flapping and lagging accelerations are obtained. These blade angular accelerations are transformed into the non-rotating multi-body coordinate system and represented by the MBC acceleration coefficients by using the following transformation [30];

$$\begin{gathered} \beta_{non} = DT^T \beta_{rot} \\ \beta_{non} = DT^T\beta_{rot} - E_1DT^T\beta_{rot} \\ \ddot{\beta}_{non} = DT^T\ddot{\beta}_{rot} - 2E_1DT^T\dot{\beta}_{rot} + E_2DT^T\beta_{rot} \end{gathered} \qquad 3\text{-}57$$

Matrix T which is the multi-body transformation matrix is defined as one multi-blade part T_0 and a rotating to non-rotating frame part R;

$$T = T_0R = \begin{bmatrix} . & . & . & . \\ . & . & . & . \\ 1 & \cos km\Delta\psi & \sin km\Delta\psi & (-1)^m \\ . & . & . & . \\ . & . & . & . \end{bmatrix} \begin{bmatrix} 1 & & & \\ & \cos k\psi & \sin k\psi & \\ & -\sin k\psi & \cos k\psi & \\ & & & 1 \end{bmatrix} \qquad 3\text{-}58$$

Since these rotation matrix is reversible, the motion in both rotating(IBC) and non-rotating (MBC) frames defines the same motion.When the rotating (IBC) accelerations for flapping and lagging degrees of freedom is calculated, they are transformed into non-rotating coordinates (MBC). Time integration of these accelerations is taken in multi-body coordinates system. For the next time-step lagging and flapping velocities and angles are distributed onto the rotor disc by transforming from MBC into IBC. Therefore, these parameters are ready for the new set of rotor inputs.

3.2.6 State-Space representation of a given rotor model

Rotor model includes different sets of input, state and output parameters. These parameters are activated or deactivated by the user before the analysis. Fidelity of the solution depends on the rotor model configuration. There are several types of pilot inputs for a rotor which are collective, longitudinal cyclic, lateral cyclic, RPM, longitudinal tilt, lateral tilt (cant). Different combinations of these input types are used for different types of rotorcrafts. For example, for a conventional single main and single tail rotorcraft, active input set is collective, longitudinal and lateral cyclic for main rotor and collective for tail rotor. There are total of 4 controls. For a tilt-rotor aircraft such as XV-15 and V-22 there are longitudinal tilt and RPM control as well. Rotor dynamic states are rotational speed of the rotor Ω, flapping angles and flapping rates in multi-body coordinate system, lagging angles and lagging rates in multi-body coordinate system, inflow dynamic states. Rotor outputs can be summarized as general rotor parameters, rotor forces and moments, angular velocities and tip path plane orientations, induced velocity distributions and so on. Detailed diagram showing the all inputs, states and outputs are given in Figure 3-24. Configuring the rotor inputs and active state depends on the user and they should be configured correctly.

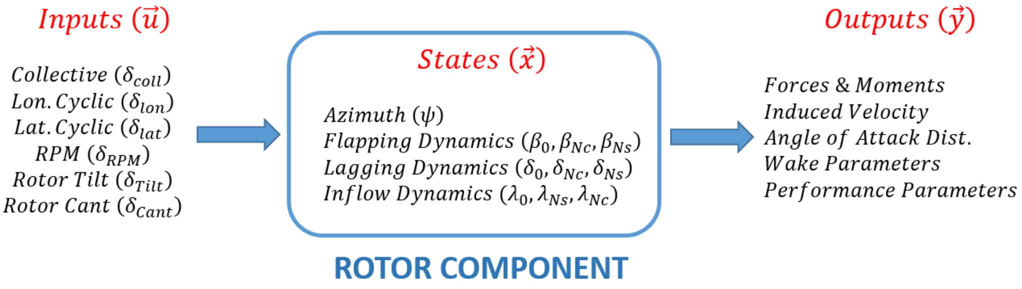

$Inputs\ (\vec{u})$
$Collective\ (\delta_{coll})$
$Lon.Cyclic\ (\delta_{lon})$
$Lat.Cyclic\ (\delta_{lat})$
$RPM\ (\delta_{RPM})$
$Rotor\ Tilt\ (\delta_{Tilt})$
$Rotor\ Cant\ (\delta_{Cant})$
$States\ (\vec{x})$
$Azimuth\ (\psi)$
$Flapping\ Dynamics\ (\beta_0, \beta_{Nc}, \beta_{Ns})$
$Lagging\ Dynamics\ (\delta_0, \delta_{Nc}, \delta_{Ns})$
$Inflow\ Dynamics\ (\lambda_0, \lambda_{Ns}, \lambda_{Nc})$
ROTOR COMPONENT
$Outputs\ (\vec{y})$
Forces & Moments
Induced Velocity
Angle of Attack Dist.
Wake Parameters
Performance Parameters

Figure 3-24 Detailed diagram for rotor object simulation parameters

3.2.7 Input Files and Algorithms

Rotor model is defined in the main rotorcraft definition XML file which is given in Appendix section. This file can include multiple numbers of rotors which are defined in the rotor chunk. Each rotor object can be specified by preparing an XML input file which holds all the required information for a rotor object.

3.2.8 Further Corrections, Models and Studies

Besides the general theory which is explained in detail in CHAPTER 3, there are some further corrections to correlate the model with the real life and approximations for the specific type of modeling issues. In this section, these small modifications and models are described.

Ground Effect

Hovering rotors produces extra thrust when operating near the ground which is called as “ground effect”. Since the ground presence constrain the rotor wake development, hover performance of a rotor parallel and close to the ground is affected. For a given power setting, rotor thrust is increased when close to the ground and a representative plot is given in Figure 3-25 to illustrate the effect of the rotor height from the ground. Presented experimental results are belonging to several test campaigns at different blade loadings which are conducted by Betz [42], Zbrozek [43], Knight and Hefner [44], Cheeseman and Bennett [45], Stepniewski and Keys [46], Prouty [47] and Hayden [48].

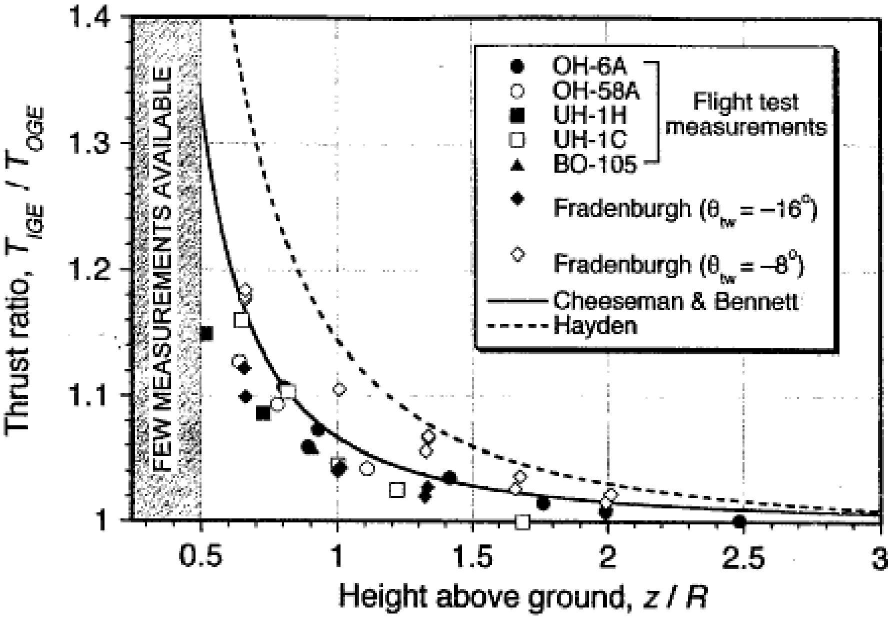


Figure 3-25 Variation of Thrust ratio vs. height above the ground

The tests show significant thrust increment when rotor is one radius above the ground. Results depend on several factors such as blade loading, aspect ratio, and blade geometry. It can be ensured that minimum 2R is required to ensure that the rotor performance can be assessed as OGE. Based on Cheeseman and Bennet an analytical approach is defined for rotor thrust such as;

$$\left[\frac{T}{T_\infty}\right] = \frac{1}{1 - \frac{(R/4z)^2}{1 + (\mu/\lambda_i)^2}} \qquad \text{3-59}$$

Equation is valid only $z/R > 0.5$. Above equation is modified by adding the effect of blade loading term.

$$\left[\frac{T}{T_\infty}\right] = \frac{1}{1 - \frac{\sigma C_{l_a}\lambda_i}{4C_T}\frac{(R/4z)^2}{1 + (\mu/\lambda_i)^2}} \qquad \text{3-60}$$

Prandtl's Tip Loss Model

Blade tip vortex formation of each rotor blade induces a high inflow at the blade tips which degrades the lifting capability. Effect of tip vortices on the local thrust is illustrated in Figure 3-26.

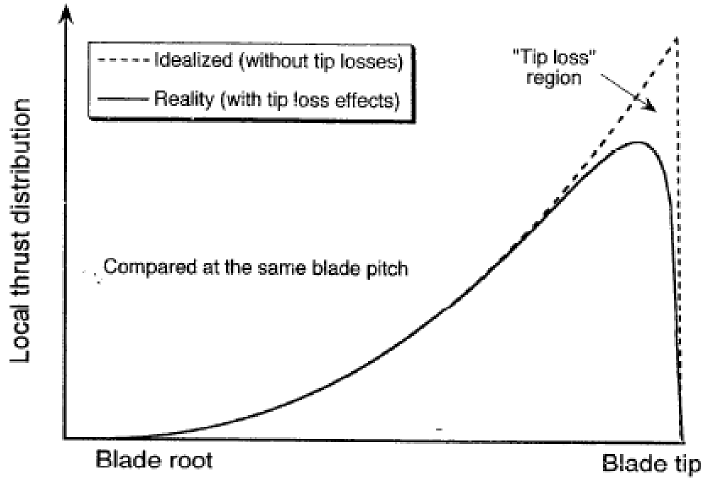


Figure 3-26 Effect of tip vortices on spanwise local thrust distribution

This phenomenon is called as "tip loss" and can be modeled by using the Prandtl's tip loss model. Prandtl's tip loss model replaces the curved helical vortex sheets of the rotor wake by a series of 2-D vortex sheets [36]. This is a good assumption for rotors but not propellers since the assumption is the radius of curvature of the vortex sheets at the blade tips is large. Tip loss is expressed as a correction factor F as in equation 3-61.

$$F = \left(\frac{2}{\pi}\right)\cos^{-1}(\exp(-f)) \qquad 3\text{-}61$$

The term f depends on the number of rotor blades, the radial position of the blade segment, and the inflow angle ϕ as in equation 3-62.

$$f = \frac{N_b}{2}\left(\frac{1-r}{r\phi}\right) \qquad 3\text{-}62$$

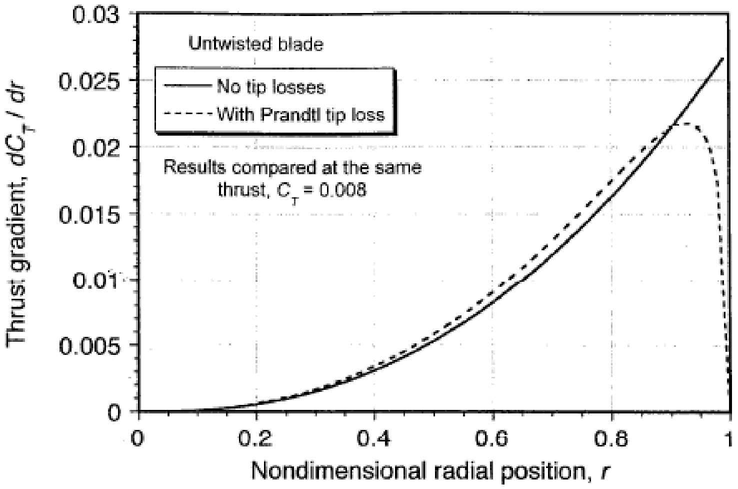


Figure 3-27 Radial thrust distribution with Prandtl's Tip Loss Model

As a result, Prandtl's tip loss correction results in a higher collective setting for the same pitch and improves the hover efficiency when the number of blades is larger. This model is approximately simulating the tip loss effects due to strong tip vortices as given in 3-24.

Induced Velocity Correction Factor

For momentum theory based inflow models, 3-D effects of rotor inflow cannot be modeled correctly. These models correlate the downwash throughout the rotor disk with the loading on the rotor disk. Therefore, some three dimensional effects such as tip vortices, blade vortex interactions, rotor swirl, and wake contraction are not taken into account. Some of these effects are significant during hover and some of them are significant during forward flight. Therefore, an induced velocity correction factor κ is implemented to reflect these effects into the model. The inflow correction factor of kappa changes rotor by rotor and according to the flight condition. Therefore in GAVM this correction factor can be defined and scheduled for advance ratio μ in configuration XML file as shown as:

```
<Kappa value="1.2" Mu="0.00"/>
<Kappa value="1.0" Mu="0.25"/>
```

The correction factor kappa is multiplied with the thrust loading term of the inflow model which requires higher induced velocity for the same level of thrust. Therefore, required power is increased to reflect the changes. As a result, inflow correction factor is used to correlate the simulation model with the test data and can be scheduled with respect to forward speed.

Static Stall Delay (Stall delay due to rotation)

Although stall delay due to rotation is a more dominant phenomenon in propellers, it has similar effects to the rotors. Experimental data belonging to propeller tests shows the lift coefficient values at the blade root is dramatically higher than 2D lift coefficients. And these tests suggests that there is a postpone in stall due to the rotation of the propeller [49]. Due to very large twist angles in root section, root stall is expected. However, the pressure distribution in 3-D flow is much more different than either the attached or the stalled 2-D pressure distribution [50]. The 3-D effect depends on the ratio of chord length to the distance which is from the axis of rotation to the blade section. The centrifugal forces of the air particles due to the rotation creates a yawed flow onto the root section and this is the main reason of the stall delay [30].

Static stall delay developed by Du and Selig [51] is implemented into the Rotor component. Boundary layer equations are solved for external velocity including with a stall delay factor K. The laminar separation point is obtained and its position is found in terms if local chord ratio. The lag in separation is detected by comparing with 2-D airfoil data as in 3-63.

$$\begin{aligned} c_l &= c_{l_{2D}} + K_{sd}(c_{l\alpha}(\alpha - \alpha_z) - c_{l2D}) \\ c_d &= c_{d_{2D}} + K_{sd}(c_{dz} - c_{d2D}) \end{aligned} \qquad 3\text{-}63$$

The functions f_l and f_d in 3-64 are developed for the separation factor which is used in 3-63. These functions are for the lift coefficient and the drag coefficients respectively. Parameters a, b and d are empirical correction factors and can be set to unity.

$$f_l = \frac{1}{2\pi}\left[\frac{1.6\left(\frac{c}{r}\right)}{0.1267}\frac{a-({}^{c}/_{r})^{\frac{dR}{\Lambda r}}}{b+({}^{c}/_{r})^{\frac{dR}{\Lambda r}}}-1\right]$$

$$f_d = \frac{1}{2\pi}\left[\frac{1.6\left(\frac{c}{r}\right)}{0.1267}\frac{a-({}^{c}/_{r})^{\frac{d\ R}{2\Lambda r}}}{b+({}^{c}/_{r})^{\frac{d\ R}{2\Lambda r}}}-1\right] \qquad 3\text{-}64$$

$$\Lambda = \Omega R/\sqrt{V_\omega^2+(\Omega R)^2}$$

This static stall delay due to rotation is integrated into the BlSegment class for each execution of Rotor object model.

Grid Dependency Study

Blade element method and Inflow model used in Rotor component is analyzed for different numbers of azimuthal and span-wise segments. Analysis is done on a sample 5-bladed rotor for high advance ratio about $\mu = 0.4$ to maximize the difference in aerodynamic and inertial loads over the disk. Analysis results shows that results are highly converged after 20 radial segments and 30 azimuthal segments. Figure 3-28 shows total number cells and the calculations of rotor thrust and torque.

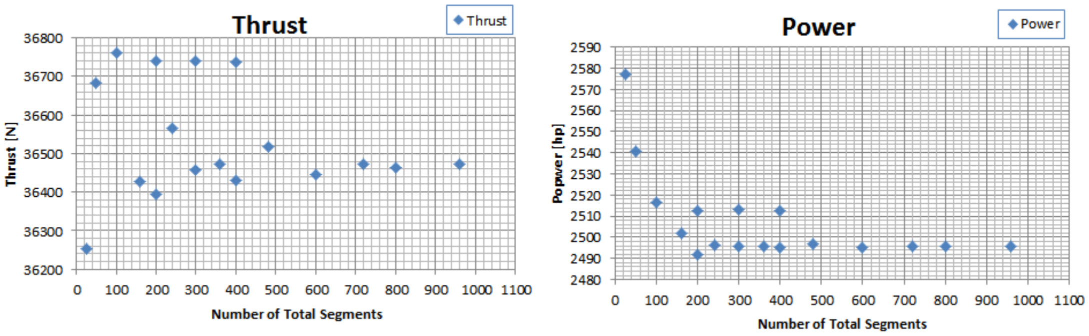


Figure 3-28 Thrust & Power variation for different number of total segments

From Figure 3-29 it can be seen that after 20 numbers of spanwise segments solution converges.

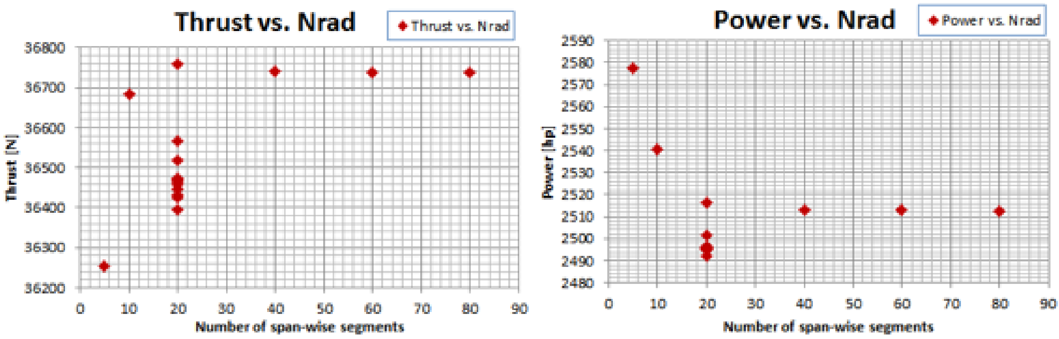


Figure 3-29 Thrust & Power for different number of spanwise segments

Analysis shown in Figure 3-30 is done for 24 spanwise distributed segments for different number azimuthal segments. It can be seen that after 30 azimuthally distributed segments, solution is converged. Moreover, all analyses are done for 24x36 rotor disk grid in this report.

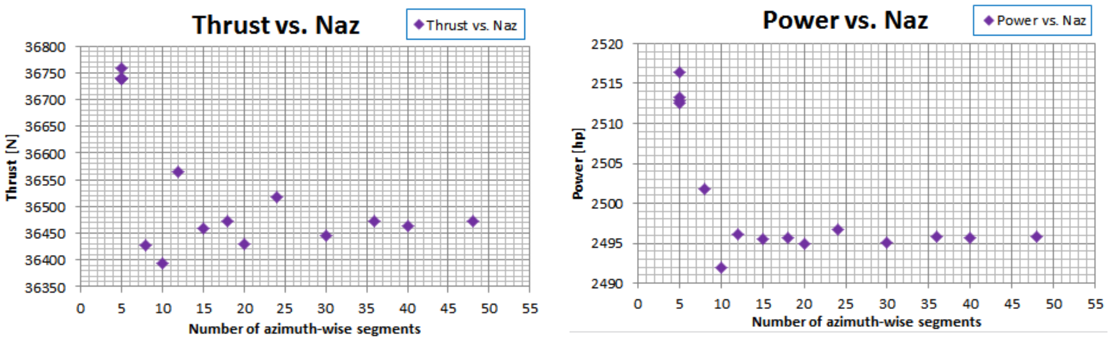


Figure 3-30 Thrust & Power for different number of azimuthal segments

3.2.9 Software Class Diagram

Reduced class diagram of rotor, blade, blade segment, hub, inflow and airfoil object is given in Figure 3-31.

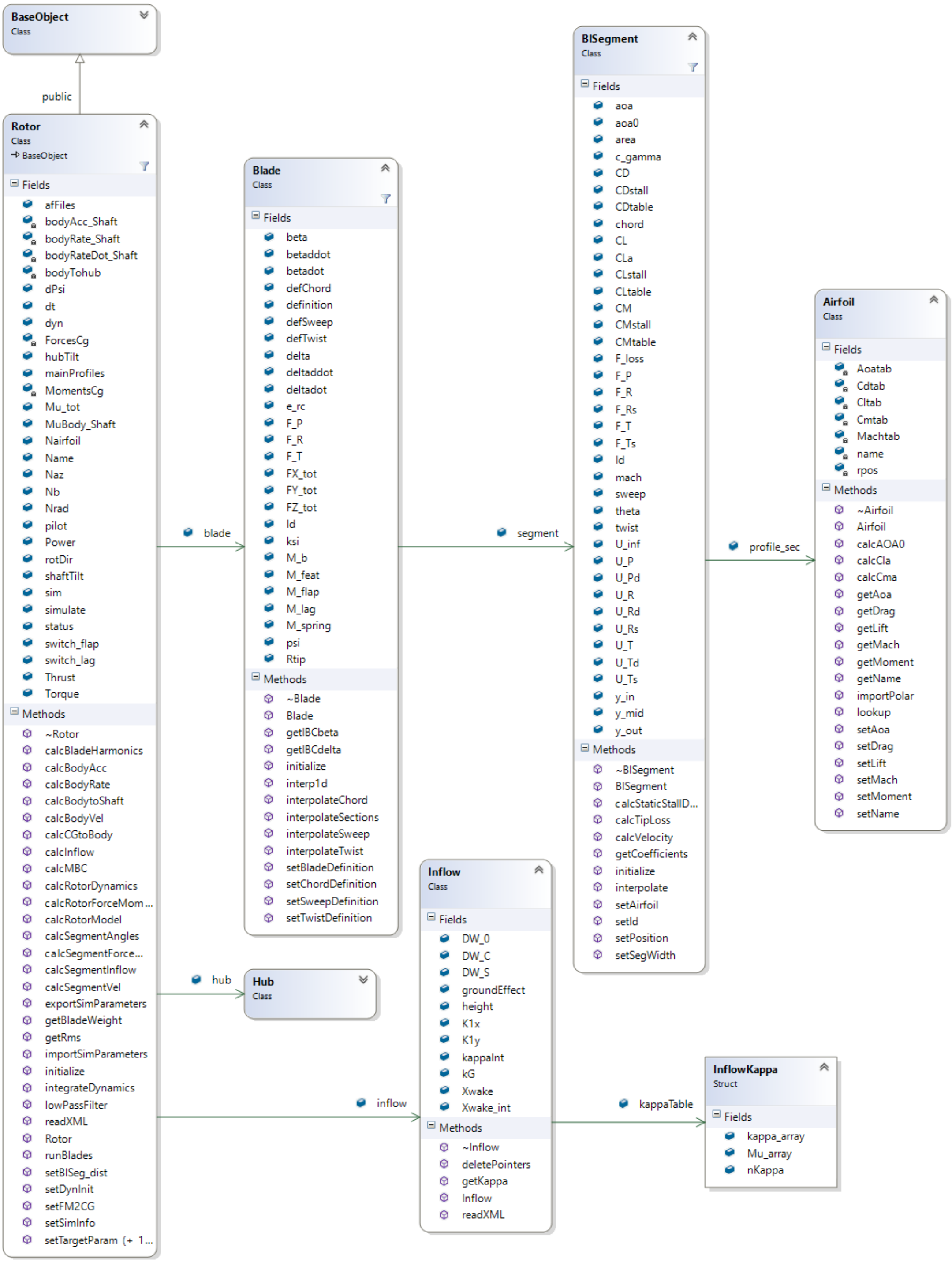


Figure 3-31 Reduced form of Class Diagram of Rotor and related objects (Not all fields and methods are displayed)

3.3 Body Model

For general air vehicles there are some components having different aerodynamic shapes mounted on the vehicle. These components generate aerodynamic forces and moments which cannot be calculated by analytical methods. The aerodynamic forces and moments of these components must either be known or be calculated. Calculation of the aerodynamics of these objects is costly and some quick and useful methodologies are required for such software. The methodology implemented in this object is the general table-lookup methodology for aerodynamic forces and moments.

3.3.1 Table Lookup Methodology

Aerodynamic coefficient tables are required to express the aerodynamic characteristics of the component. These tables are generated either by experiment or by CFD analyses. In order to include the aerodynamic effects due to attached surfaces like fuselage, tails, external loads, pods, and table look-up model architecture is generated and the object is called as **Body** component. Several components may be attached to main frame by using the class of **Body**. Aerodynamic forces and moments are provided by 6-DOF aerodynamic coefficients either in wind axis, or in body axis. Aerodynamic load acting point, reference area and reference length are the main parameters to be determined by the user.

Aerodynamic tables included in this component should be provided for angle of attack α, and angle of sideslip β. Coefficients include lift, drag and side forces and roll, pitch, and yaw moment information either in body frame or in wind frame. Required transformations are available for each axis of reference. Table lookup procedure includes linear interpolation and binary search. There is no extrapolation exists for the values which are out of bound. Only the end values are used.

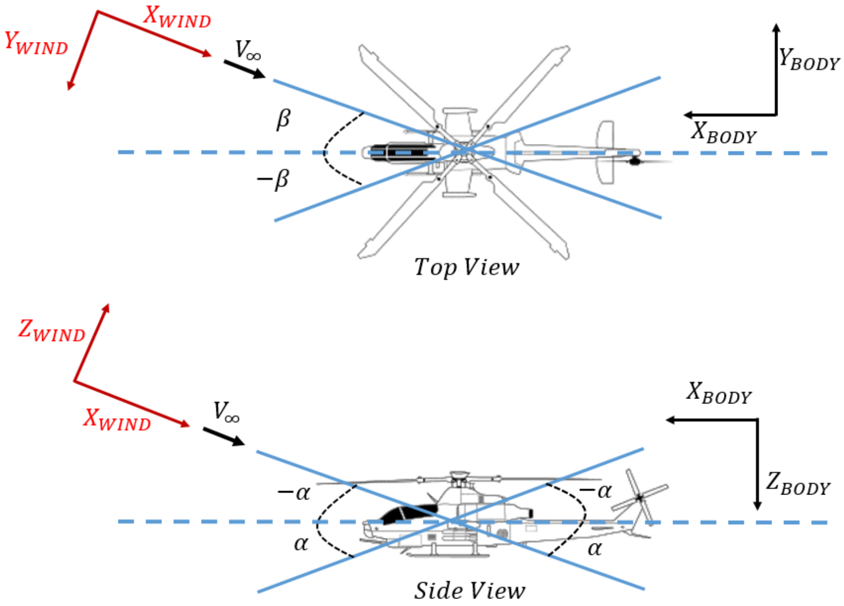


Figure 3-32 Body and wind axis definitions of a Body object

Angle of attack is defined by using the components of the total air velocity $u_{air}, v_{air}, w_{air}$ as:

$$\alpha = \tan^{-1}\left(\frac{w_{air}}{|u_{air}|}\right) [-\frac{\pi}{2}, \frac{\pi}{2}] \tag{3-65}$$

And the angle of sideslip is defined as:

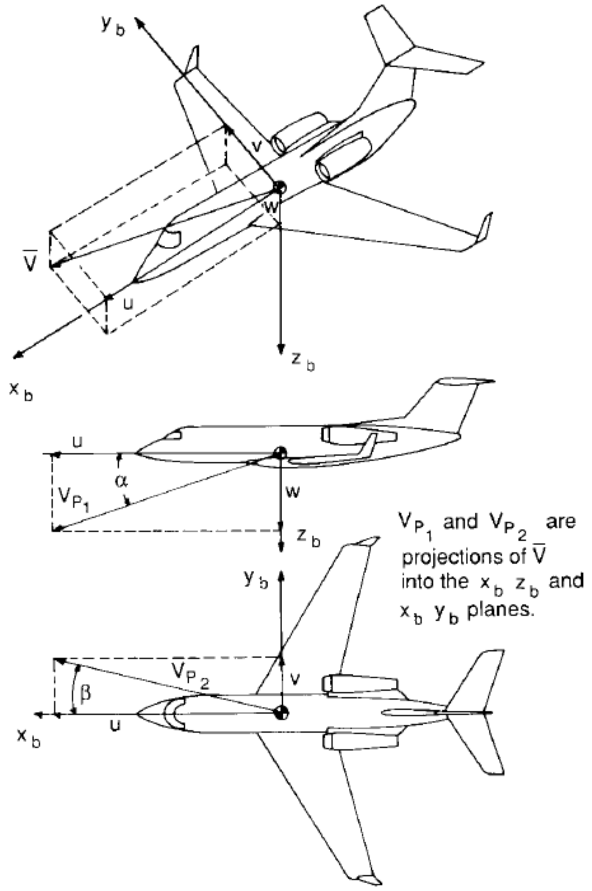
$$\beta = \tan^{-1}\left(\frac{-v_{air}}{sign(-u_{air})\sqrt{u_{air}^2 + w_{air}^2}}\right) [-\pi, \pi] \tag{3-66}$$

And the aerodynamic coefficients are defined as a function of angle of sideslip β and angle of attack α;

$$\begin{array}{c} C_X = C_X(\alpha,\beta) \\ C_Y = C_Y(\alpha,\beta) \\ C_Z = C_Z(\alpha,\beta) \\ C_L = C_L(\alpha,\beta) \\ C_M = C_M(\alpha,\beta) \\ C_N = C_N(\alpha,\beta) \\ Body\ Axis \end{array} \qquad \begin{array}{c} C_L = C_L(\alpha,\beta) \\ C_D = C_D(\alpha,\beta) \\ C_S = C_S(\alpha,\beta) \\ C_R = C_R(\alpha,\beta) \\ C_M = C_M(\alpha,\beta) \\ C_Y = C_Y(\alpha,\beta) \\ Wind\ Axis \end{array}$$

3-67
Aerodynamic Coefficients
in body and wind axes

Definitions of angle of attack and sideslip angles are given in Figure 3-33 according to body velocities.



V_{P_1} and V_{P_2} are projections of $\bar{V}$ into the x_b z_b and x_b y_b planes.

Figure 3-33 Definition of angle of attack and sideslip angle

For a given angle of attack and beta set 6-DOF aerodynamic forces acting on the object is found by the following formulas:

$$F = \frac{1}{2}\rho V_{\infty} S_{ref} C_F$$
$$F = \frac{1}{2}\rho V_{\infty} S_{ref} C_M L_{ref} \qquad 3\text{-}68$$

where

$$V_{\infty} = \sqrt{v_x^2 + v_y^2 + v_z^2} \qquad 3\text{-}69$$

S_{ref}: $reference\ area$

L_{ref}: $reference\ length$

C_F : $aerodynamic\ force\ coefficients$

C_M: $aerodynamic\ moment\ coefficients$

After 6-DOF forces and moments acting on the body is calculated, they are moved to the vehicle center of gravity. If the calculated forces and moments are in the wind frame, following transformation is used to transform the forces from wind frame to body frame:

$$T_{b/w} = \begin{bmatrix} \cos\beta\cos\alpha & \sin\beta\cos\alpha & -\sin\alpha \\ \sin\beta & -|\cos\beta| & 0 \\ |\cos\beta|\sin\alpha & \sin\beta\sin\alpha & \cos\alpha \end{bmatrix}$$

3-70

$$\begin{bmatrix} C_{F_x} & C_{F_y} & C_{F_z} \end{bmatrix}^T = T_{b/w}[-C_d \quad -C_y \quad -C_l]^T$$
$$\begin{bmatrix} C_{M_x} & C_{M_y} & C_{M_z} \end{bmatrix}^T = T_{b/w}[C_r \quad -C_m \quad C_n]^T$$

3.3.2 Software Class Diagram

Software class diagram of the Body object is depicted in Figure 3-34.

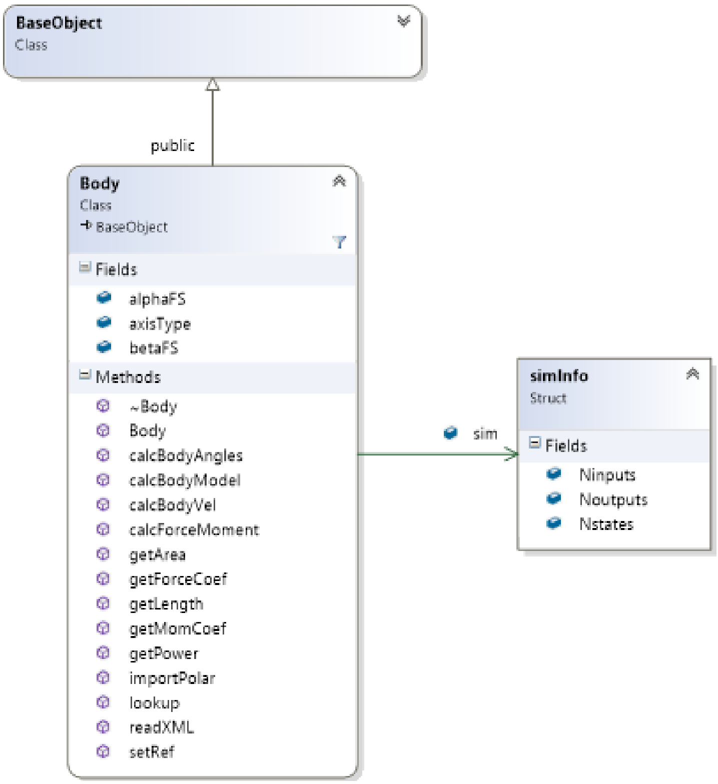


Figure 3-34 Software class diagram of Body object. (Not all fields are displayed)

3.4 Wing Model

There are different methodologies to simulate wing aerodynamics in mathematical models. The type of the method should be selected by the purpose of modeling. For initial design of an aircraft where there require several iterative loops, there should be an analytical or numerical method that provide compromise between accuracy and computational cost [52]. This model should provide the wing performance for varying chord, taper, twist and sweep. In this case a modified version of Weissinger's non-linear lifting line method is implemented [53], [54]. On the other hand, if the objective is the time marching dynamic flight simulation and controller design, the model should provide solution real-time. In such a situation pre-analyzed viscous aerodynamic coefficient tables are used in determining wing aerodynamic performance. The use of aerodynamic coefficient tables which require table generation for all domains either with CFD or experiment are mostly suggested since they reduce the computational cost. This table lookup methodology is also provided in the wing model for simulation purposes.

In this thesis, both methodologies are developed and implemented into the **Wing** object. Wing object can be modeled with aerodynamic coefficient tables or numerical lifting line methods. Numerical lifting line method is computationally more expensive than table lookup methodology, however is useful in conceptual design phase.

3.4.1 Table Lookup Methodology

Similar to **Body** object a table lookup methodology is implemented into the **Wing** object as explained in Chapter 3.3. Table lookup methodology is quick and useful if whole domain of angle of attack and sideslip is covered. Additional dimensions may arise due to the necessity of modeling control surfaces. Aerodynamic force and moment coefficients represents six degree of freedom which are lift, drag, side, roll, yaw, and pitch moments. Wing object may be considered as an aerodynamic component having a flap deflection degree of freedom. If there is not a flap control, Body model can also be used to model a wing without any control surfaces. In table lookup methodology, there are two ways to model the aerodynamic effects of a flap component. First one is providing new sets of 6-DOF aerodynamic coefficient tables for different flap deflection angles so that a 3-D interpolation algorithm which includes flap deflection angle δ_f, angle of attack α, and angle of sideslip β as the dimensions of interpolation calculates the aerodynamic forces on the wing. Fidelity of the mathematical model of wing is determined by the table itself. Tables may be generated by some inviscid methods, viscous CFD methods or experimental data.

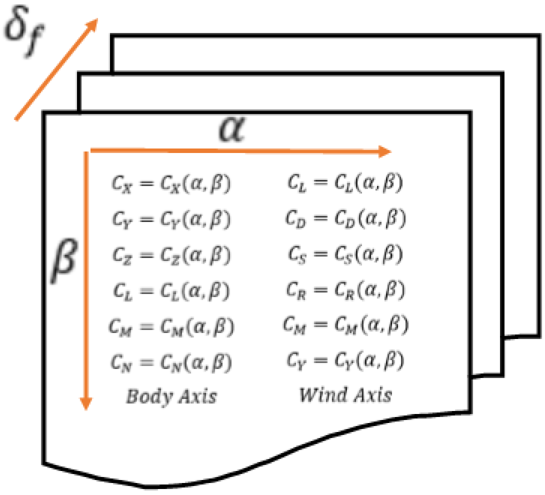

δ_f
α
β
$C_X = C_X(\alpha, \beta)$
$C_Y = C_Y(\alpha, \beta)$
$C_Z = C_Z(\alpha, \beta)$
$C_L = C_L(\alpha, \beta)$
$C_M = C_M(\alpha, \beta)$
$C_N = C_N(\alpha, \beta)$
Body Axis
$C_L = C_L(\alpha, \beta)$
$C_D = C_D(\alpha, \beta)$
$C_S = C_S(\alpha, \beta)$
$C_R = C_R(\alpha, \beta)$
$C_M = C_M(\alpha, \beta)$
$C_Y = C_Y(\alpha, \beta)$
Wind Axis

Figure 3-35 Diagram showing the 3-D interpolation and aerodynamic tables

Second approach is the aerodynamic derivatives which defines a linear relationship for 6-DOF forces and moments in case of a flap deflection. These derivatives produce delta force and moment changes onto the 2-D aerodynamic coefficient tables. Therefore, the effects of flap degree of freedom are modeled. Derivatives may be defined for either in wind frame or in body frame in which coordinate system the 2-D aerodynamic tables are defined. The derivatives may be summarized in Table 3-1.

Table 3-1 Summary of aerodynamic force and moment derivatives with respect to angle of flap

Body Frame	Wind Frame	Description
$\frac{\partial \boldsymbol{CF_x}}{\boldsymbol{\partial_{flap}}}$	$\frac{\partial C_D}{\partial_{flap}}$	X force or Drag force derivative vs. flap angle
$\frac{\partial \boldsymbol{CF_y}}{\boldsymbol{\partial_{flap}}}$	$\frac{\partial C_S}{\partial_{flap}}$	Y force or Side force derivative vs. flap angle
$\frac{\partial \boldsymbol{CF_z}}{\boldsymbol{\partial_{flap}}}$	$\frac{\partial C_L}{\partial_{flap}}$	Z force or Lift force derivative vs. flap angle
$\frac{\partial \boldsymbol{CM_x}}{\boldsymbol{\partial_{flap}}}$	$\frac{\partial C_R}{\partial_{flap}}$	Roll moment derivative vs. flap angle
$\frac{\partial \boldsymbol{CM_y}}{\boldsymbol{\partial_{flap}}}$	$\frac{\partial C_M}{\partial_{flap}}$	Pitch moment derivative vs. flap angle
$\frac{\partial \boldsymbol{CM_z}}{\boldsymbol{\partial_{flap}}}$	$\frac{\partial C_Y}{\partial_{flap}}$	Yaw moment derivative vs. flap angle

Wings can be modeled as a single wing with aerodynamic derivatives for a symmetric aileron deflection or can be modeled as double wings which have separate flaps. Flaps should be connected to each other as a single aileron input in configuration XML file. Therefore, for a control surface like aileron, pilot input will be the flap deflection angle, and for a symmetrical wing it could be aileron.

In case of a planform design or in lack of aerodynamic derivatives and 3D aerodynamic data, wings can also be modeled mathematically. Wing platform with variable chord, sweep, twist and section profiles can be defined and solved by using the Non-Linear Second Order Lifting Line Theory.

3.4.2 Numerical Lifting Line

Classical lifting line theory of Prandtl's [55] calculates the circulation distribution and induced velocity distribution over the wing for the angle of attack values where the lift curve slope is linear. Therefore, lifting line method does not cover the stall and post stall region of the wing aerodynamics.

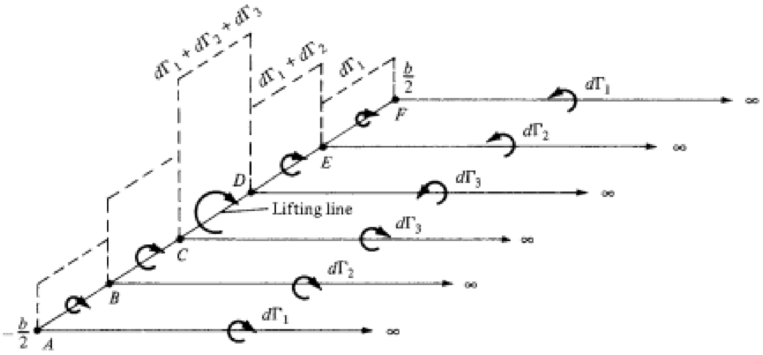


Figure 3-36 Finite number of horseshoe vortex sheets and their superposition in Prandtl's Method [56]

There is another methodology which is implemented in GAVM Wing Model is developed by Weissinger [14] which calculates the three dimensional circulation distribution along the wing by using vortex sheets. This LLT is modified and used by the Blackwell [57] for wings having dihedral and sweep. This method is called as second order lifting line method since it assumes a lifting line composed by vortex system at c/4 and a second line for flow tangency condition which passes from 3c/4. Along this line induced downwash angle is equal to the airfoils angle of attack according to Pistolesi's [14] three-quarter chord condition. Weissinger's method of lifting line substitutes the c/4 line with a spanwise distributed horseshoe vortices. Bound vortices are located at c/4 line and the flow tangency condition is implemented at 3c/4 line. Figure 3-37 illustrates an example of swept back wing having bound and trailing vortex system for each aerodynamic panel.

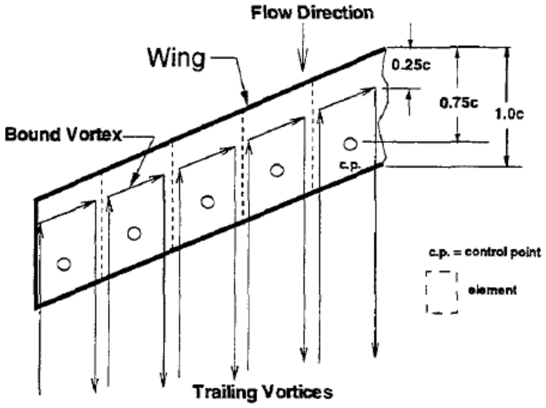


Figure 3-37 Horseshoe vortex sheets in Weissinger method [58]

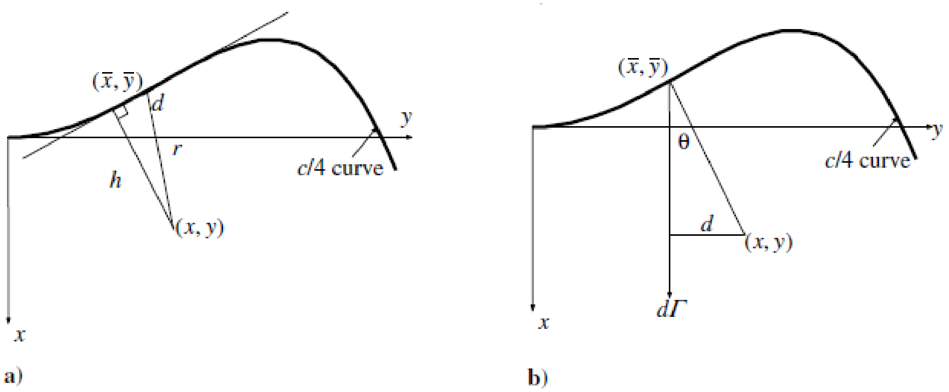


Figure 3-38 Schematic of the induced velocity by a segment of a) bound vortex b) trailing vortex system [54]

After the wing geometry such as chord distribution, sweep angle, twist and span length are distributed the aerodynamic panels are generated each of which holds one bound vortex and two trailing vortices. The downwash at each control point is the summation of the induced velocity contributions of all horseshoe vortex system having lifting vortex and trailing vortex system as shown in Figure 3-38. The downwash velocity is calculated by using Biot-Savart Law:

$$v = \frac{1}{4\pi}\int \frac{\Gamma \times r}{|r|^3} ds \qquad \text{3-71}$$

Biot-Savart Law calculates the velocity of the fluid due to vortex having strength Γ at a point having a distance r from the lifting vortex. Induced velocity caused by the bound vortex given in Figure 3-38 a) at a segment ds is given as;

$$dw(x,y) = \frac{\Gamma h ds}{4\pi r^3} \qquad \text{3-72}$$

In terms of local coordinate system of the wing (x, y) and quarter chord line coordinates $(\bar{x}, \bar{y})$, 3-72 results in;

$$dw(x,y) = \mathit{\Gamma}(\bar{y})\frac{\big[x - \bar{x}(\bar{y}) + \overline{x'}(\bar{y})(\bar{y} - y)\big]d\bar{y}}{4\pi\{[x - \bar{x}(\bar{y})]^2 + (y - \bar{y})^2\}^{3/2}} \tag{3-73}$$

Induced velocity due to the trailing vortex system given in Figure 3-38 b) at a segment is given as;

$$dw(x,y) = \frac{\mathrm{d}\Gamma}{4\pi d}(\cos\theta + 1) \tag{3-74}$$

terms of local coordinate system of the wing (x, y) and quarter chord line coordinates $(\bar{x}, \bar{y})$, 3-74 results in;

$$dw(x,y) = \frac{\Gamma'(\bar{y})}{4\pi(y - \bar{y})}\left(\frac{x - \bar{x}(\bar{y})}{\sqrt{[x - \bar{x}(\bar{y})]^2 + (y - \bar{y})^2}} + 1\right)d\bar{y} \tag{3-75}$$

3-73 and 3-75 are added up for each horseshoe vortex system to calculate the induced velocity at control point at $3c/4$. In other words, integral of the induced velocity from $-y_0$ to y_0 provides the total induced velocity for a point (x, y);

$$\begin{aligned}
w(x,y) &= \frac{1}{4\pi}\int_{-y_0}^{y_0}\frac{\Gamma'(\bar{y})}{y - \bar{y}}d\bar{y} \\
&+ \frac{1}{4\pi}\int_{-y_0}^{y_0}\frac{\Gamma'(\bar{y})}{y - \bar{y}}\frac{x - \bar{x}(\bar{y})}{\sqrt{[x - \bar{x}(\bar{y})]^2 + [y - \bar{y}]^2}}d\bar{y} \\
&+ \frac{1}{4\pi}\int_{-y_0}^{y_0}\Gamma(\bar{y})\frac{x - \bar{x}(\bar{y}) + \overline{x'}(\bar{y})(\bar{y} - y)}{\{[x - \bar{x}(\bar{y})]^2 + [y - \bar{y}]^2\}^{3/2}}d\bar{y}
\end{aligned} \tag{3-76}$$

First term of the integral in 3-76 represents the downwash caused by the bound vortex, and the remaining terms represent the downwash caused by the trailing vortices. There are two singularities at $y = \bar{y}$ in first and second integrals, but the second integral diverges only. Therefore, 3-76 is modified to remove divergence in the second integral as given in [59] by adding and subtracting the first integral term, results as;

$$\begin{aligned} w(x,y) &= \frac{1}{2\pi}\int_{-y_0}^{y_0}\frac{\Gamma'(\bar{y})}{y-\bar{y}}d\bar{y} \\ &+\frac{1}{4\pi}\int_{-y_0}^{y_0}\frac{\Gamma'(\bar{y})}{y-\bar{y}}\left[\frac{x-\bar{x}(\bar{y})}{\sqrt{[x-\bar{x}(\bar{y})]^2+[y-\bar{y}]^2}}-1\right]d\bar{y} \\ &+\frac{1}{4\pi}\int_{-y_0}^{y_0}\Gamma(\bar{y})\frac{x-\bar{x}(\bar{y})+\overline{x'}(\bar{y})(\bar{y}-y)}{\{[x-\bar{x}(\bar{y})]^2+[y-\bar{y}]^2\}^{3/2}}d\bar{y} \end{aligned} \qquad \text{3-77}$$

This final equation is the dimensional representation of downwash velocity of extended Weissinger's method. Pistolesi condition dictates that total wind velocity should be tangent to the plane of wing at a half chord aft of the quarter chord line. To include the wing sweep angle into the equation, 3c/4 point is represented as [53];

$$x=\bar{x}(y)+\frac{c(y)}{2}\frac{1}{\cos\Lambda(\mathrm{y})}=\bar{x}(y)+\frac{c(y)}{2}\sqrt{[\overline{x'}(y)]^2+1} \qquad \text{3-78}$$

To obtain the non-dimensional form of the downwash velocity following variables are used;

$$\eta=\frac{y}{y_0},\quad \bar{\eta}=\frac{\bar{y}}{y_0},\quad G=\frac{\Gamma}{y_0U_\infty},\quad \xi=\frac{x}{c},\quad \alpha=\frac{\omega}{U_\infty} \qquad \text{3-79}$$

In order to simplify the integral the non-dimensional vortex distribution $G(\bar{\eta})$ is represented as sine series [60];

$$\begin{gathered} G(\phi)=\sum_{n=1}^{m}G_nf_n(\phi) \\ f_n(\phi)=\frac{2}{m+1}\sum_{k=1}^{m}\sin(k\phi_n)\sin(k\phi),\qquad G_n=G(\phi_n) \\ \phi_n=\frac{n\pi}{m+1},\qquad \phi=\cos^{-1}(\eta) \end{gathered} \qquad \text{3-80}$$

The induced velocity 3-77 is equal to the wing angle of attack according to the Pistolesi tangency condition. Making the required substitutions into the downwash equation, it gives the induced velocity angle at$\phi_v = \cos^{-1}(\eta)$. Calculating the induced velocity angle at ϕ_n locations provides a system of linear equations $\overrightarrow{\boldsymbol{\alpha_{eff}}} = \overrightarrow{\boldsymbol{AG}}$, as follows;

$$A_{vn} = \frac{1}{\pi(m+1)\sin\phi_v}\sum_{k=1}^{m} k\sin(k\phi_n)\sin(k\phi_v)$$
$$+\frac{1}{2\pi(m+1)}\int_0^{\pi}\left[-P(\phi_v,\phi)\sum_{k=1}^{m} k\sin(k\phi_n)\cos(k\phi)\right. \qquad \text{3-81}$$
$$\left.+\left(\frac{y_0}{c(\phi_v)}\right)^2 R(\phi_v,\phi)\sin\phi\sum_{k=1}^{m}\sin(k\phi_n)\sin(k\phi)\right]d\phi$$

Extended forms of $P(\phi_v,\phi)$ and $R(\phi_v,\phi)$ is obtained from the reference [54]. Solving this system and reconstructing the series representation provides a circulation distribution for a wing having 2π lift curve slope. In order to extend this method to include the viscous airfoil data for the wings having different airfoil shapes an iterative process should be implemented. There are two methodologies for implementation of viscous airfoil data for 3-D wing model which are;

- Vortex based correction methods
- Angle of attack correction methods

Vortex based correction methodologies uses the vortex strength distribution along the wing and estimates the induced angle of attack for each section throughout the wing. Viscous section force coefficients are obtained according to the angle of attack at each section which leads to a new lift distribution. According to the difference for each approach a correction is calculated. Until the convergence is achieved, the iterative process is performed.

Secondly, there are angle of attack based correction approaches to calculate the lift distribution along the wing [61]. Angle of attack of wing sections is modified and integrated into the inviscid solution instead of implementing a correction on the distribution of lift force. The required change in angle of attack $\Delta\alpha$ is obtained by using the 2π lift curve slope based on the changes in inviscid and viscous lift coefficients in each wing section. This approach eliminates the lack of uniqueness in the post-stall calculations. Van Dam [62] uses the control points at $3c/4$ which are modified to provide the viscous slope of lift. This method calculates the lift distribution by using the Weissinger method. Then, induced angle of attack is calculated for every wing section;

$$\alpha_e = \frac{C_{l_{inv}}}{C_{l_\alpha}} + \alpha_{0l} - \Delta\alpha \tag{3-82}$$

As described in 3-82 effective angle of attack is calculated by using the inviscid lift coefficient, lift curve slope and zero lift angle of attack for each panel. Sectional viscous lift coefficient is then obtained for a pre-calculated effective angle of attack α_e. Afterwards, angle of attack correction is calculated by;

$$\Delta\alpha = \frac{C_{l_{visc}} - C_{l_{inv}}}{C_{l_\alpha}} \tag{3-83}$$

This correction is applied into the effective angle of attack and Weissinger's method is solved until the residual between the two lift coefficients are minimized.

Sample results are obtained by using the above formulation and converged circulation distribution for a rectangular untwisted NACA0012 wing is given in Figure 3-39 for angle of attack at $\alpha = 5°$.

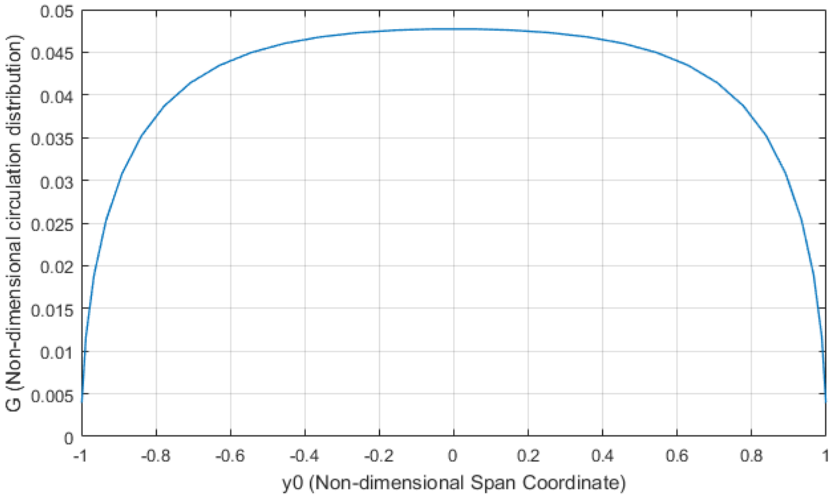


Figure 3-39 Non-dimensional circulation distribution of a rectangular NACA0012 wing at $\alpha = 5°$

Note that 2-D viscous data used in this circulation distribution having lift coefficient $C_l = 0.536$ and sectional distribution of lift coefficient is given in Figure 3-40.

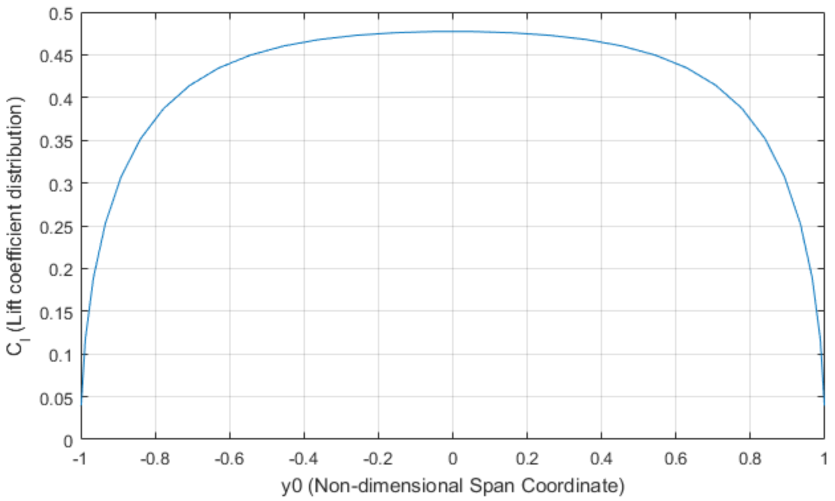


Figure 3-40 C_l distribution of a rectangular NACA0012 wing at $\alpha = 5°$

Similar to the lift coefficient, drag coefficients are available in viscous tables for interpolation with respect to angle of attack of each section. Moreover, the pitching moment coefficient may be interpolated similar to the lift and drag coefficients for each section and may be integrated throughout the three dimensional wing. In order to rotate the coefficients into plane of no twist the downwash angle ε is used,

$$\epsilon(\phi_v) = \frac{1}{2(m+1)} \sum_{n=1}^{m} G_n \sum_{k=1}^{m} \frac{k \sin(k\phi_n) \sin(k\phi_v)}{\sin \phi_v} \qquad \text{3-84}$$

If the blade section is producing lift velocity field around the airfoil of that section is defined as the closed line integral of velocity around the airfoil gives the circulation Γ:

$$\Gamma \equiv \oint \boldsymbol{V} \cdot \boldsymbol{ds} \qquad \text{3-85}$$

From the Kutta-Joukowski theorem [56], sectional lift coefficient is given by;

$$c_l = \frac{\rho_\infty U_\infty \Gamma}{\frac{1}{2}\rho_\infty U_\infty^2 c} = \frac{2\Gamma}{U_\infty c} = \frac{2y_0 G}{c} \tag{3-86}$$

After determining the sectional lift and drag coefficients rotation matrix is given as;

$$\begin{bmatrix} Cz \\ Cx \end{bmatrix} = \begin{bmatrix} \cos\epsilon & -\sin\epsilon \\ \sin\epsilon & \cos\epsilon \end{bmatrix} \begin{bmatrix} C_l \\ C_d \end{bmatrix} \tag{3-87}$$

Total three dimensional lift and drag coefficients are calculated by taking the integral from one tip to another;

$$C_L = \frac{1}{S}\int_{-1}^{1} C_l c \, d\eta \qquad\qquad C_D = \frac{1}{S}\int_{-1}^{1} C_d c \, d\eta \tag{3-88}$$

In this analysis side forces are assumed as zero and moment coefficients are calculated by non-dimensionalizing the total moments on the wing;

$$C_M = \frac{M}{QSc} \ , \qquad C_L = \frac{L}{QSb} \ , \quad C_N = \frac{N}{QSb} \tag{3-89}$$

Total moments are pitch, roll and yaw moments respectively. Pitching moment for a given point is composed of the internal pitching moment of airfoil section and moment caused by lift forces with respect to the wing center if moment arm exists. Distribution of lift forces mainly determines the total roll moment acting at the center of wing. Similarly, distribution of drag forces may create yawing moment. Center of pressure of the wing in x coordinate system is calculated by equating the moment caused by the total lift force to total moment on the wing as given in the following equation;

$$Lx_{cp} = M \quad and \ x_{cp} = \frac{1}{L}\int_{-b/2}^{b/2} l x_{c/4} dy \tag{3-90}$$

In this methodology the definition of control surfaces may be implemented in two ways. First way is to use the viscous 2-D data for the deflected airfoils where the flap region exists. Airfoil polars are defined as control surface and they may be populated for several deflection angles as plotted in Figure 3-41. For each deflection angle the lift coefficients is interpolated and integrated into Weissinger method to find total circulation distribution.

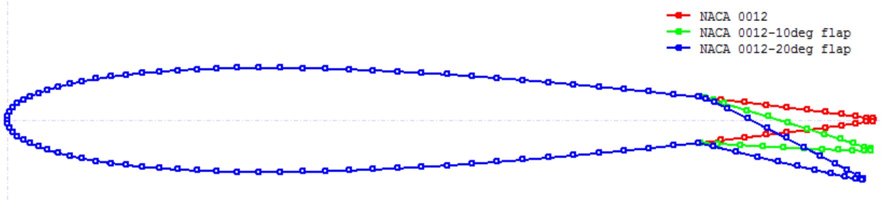


Figure 3-41 Sample flapped airfoils for a control surface

Second way of implementing the control surfaces in LLT is to define a relative angle of attack change due to the flap deflection. Flap deflection angle is incorporated with an effective angle of attack change and defined in the input files in configuration XML. Therefore, for each deflection angle there becomes a delta change in airfoil lift and drag characteristics. In Figure 3-42 lift coefficient distribution is given for a rectangular NACA0012 wing having a 15 degrees deflected flap from $y_0 = 0.55\ to\ y_0 = 0.94$ span locations. This asymmetry in lift distribution creates an integrated roll moment at the center of the wing.

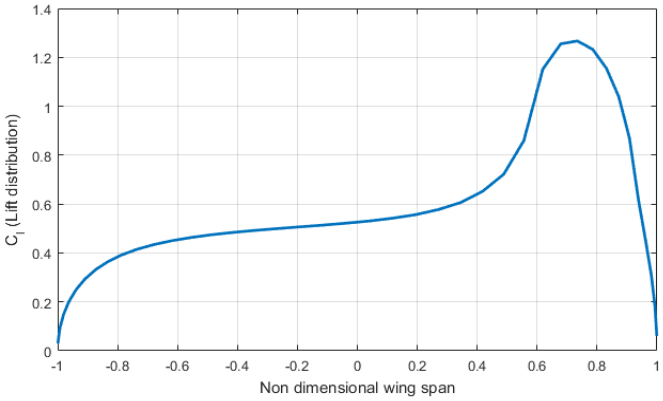


Figure 3-42 Lift distribution of NACA 0012 wing with 15 degrees deflected flap

Noting that the mathematical model implemented for a generic wing component which may be a tail surface with and without a control surface, vertical tail (half wing) with taper and asymmetric wing. Sample configuration file and description is given in Appendix.

3.4.3 Software Class Diagram

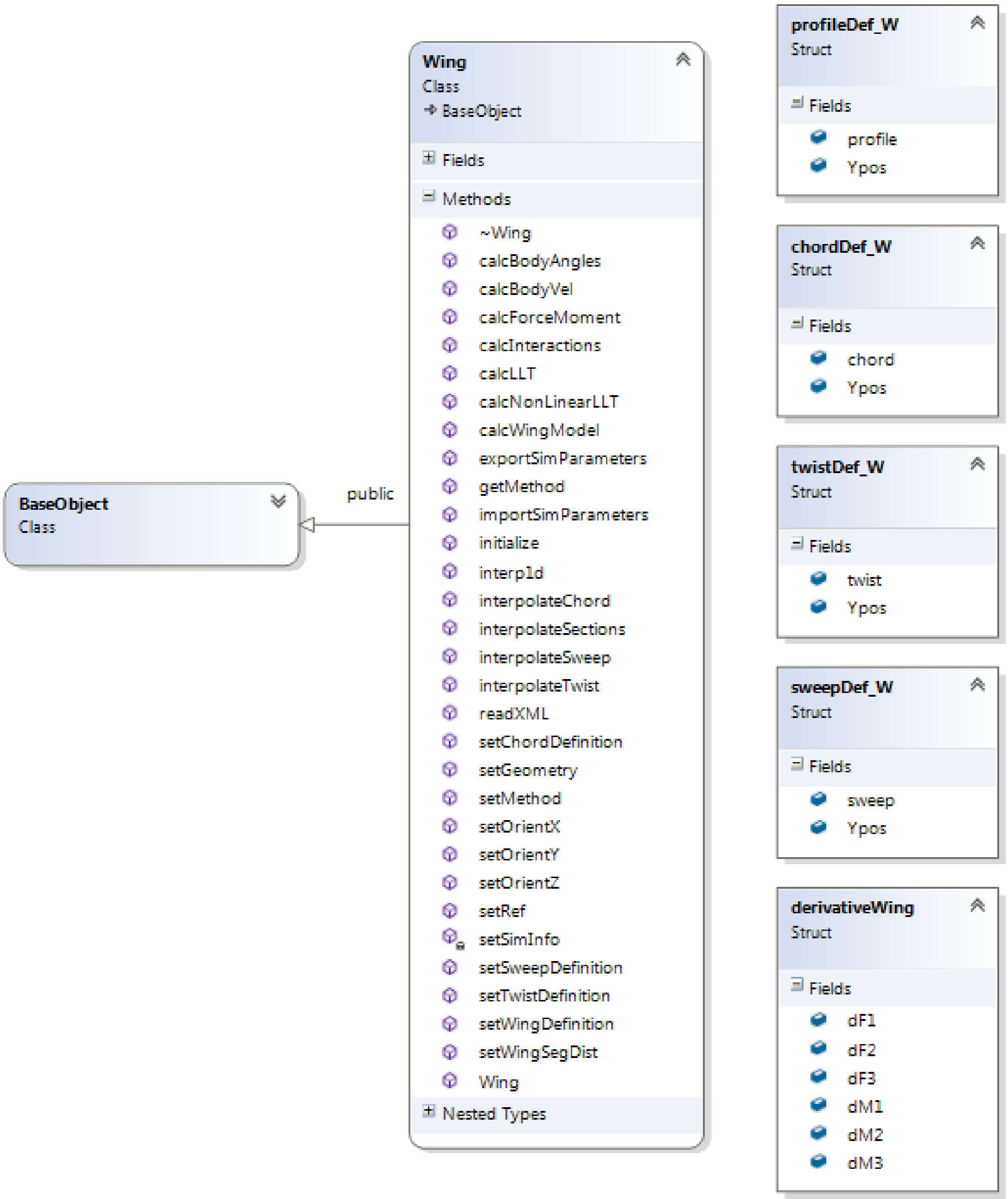


Figure 3-43 Software class diagram of Wing object (not all fields are displayed)

3.5 Propeller Model

Propellers have been used since the dawn of the engine driven aviation and being widely used in conventional airplanes and futuristic designs [63]. The propeller theory and methods implemented in engineering design and flight dynamics simulations remained unchanged since the book of Theodorsen "Theory of Propellers" which is published in 1948 [64]. Recent improvements in simulations of propeller mainly focus on unsteady aerodynamics and propellers at an incidence.

Propellers can be modeled by using the **Rotor** class, however; there are some differences due to the nature of the physics of two components. Rotors are designed to provide lift and control moments with a relatively higher in-plane freestream velocity. However, propellers are designed to provide thrust at high levels of axial velocities with relatively small in-plane freestream velocity [63]. Propellers have generally high twist angles at the root section which may be about 40 or 50 degrees and the distribution is highly non-linear, on the other hand helicopter rotors generally have 15 to 20 degrees of twist and linear distribution. Moreover, helicopter rotors operate at relatively lower disk loadings than propellers. Due to these differences obtained results with **Rotor** object generally cannot be validated. Therefore, mathematical model of propeller is implemented into the generic model as a separate component from rotor. **Propeller** class is derived from the **Rotor** class since there are several common fields and methods. Some functions are overridden related to I/O functions and different theory is implemented into the forces and moments calculations.

3.5.1 QPROP Based Methodology

Mark Drela's QPROP [65] is modified to include 2-D viscous airfoil polars and implemented as a **Propeller** component. QPROP is an analysis program for predicting the performance characteristics of propeller and electric motor combinations. QPROP uses an extension of the classical blade-element/vortex formulation, developed originally by Betz [66], Goldstein[67] and Theodorsen [64], and reformulated by Larrabee [68]. The extensions to the theory include;

- Self-induction velocity which varies radially shows consistent results with heavily-loaded actuator disk limit.
- Solution of the overall system by a global Newton method, which includes the self-induction effects.

Only the aerodynamic theory of QPROP is implemented into the **Propeller** model with including the 2-D viscous tables. Instead of using lift curve slope for calculating the propeller performance, table look-up procedure is implemented to calculate drag and lift coefficients for each blade section. Tables are two dimensional where coefficients vary with angle of attack and Mach number which includes the post stall effects, viscosity and compressibility effects. 2-D viscous data is implemented with the subroutines available in the **Rotor** and **Airfoil** objects.

Local Swirl and Local Circulation Relationship

Tangential induced velocity is associated with the torque exerted by the propeller on the fluid. Moreover, it is possible to relate the swirl with the circulation on the propeller by using the Helmholtz’s Theorem which states that a vortex filament cannot end in a fluid. Therefore, it can be assumed that the vortex filament continues as two free trailing vortices downstream up to infinity. Total circulation at radius r for all blades is $B\Gamma(r)$ where B is the number of blades, and this circulation is totally shed on the sections that are inboard of the blade segment at r.

$$\begin{aligned} 2\pi r\overline{v_t} &= \frac{1}{2}B\Gamma \\ \overline{v_t} &= \frac{B\Gamma}{4\pi r} \end{aligned} \qquad \text{3-91}$$

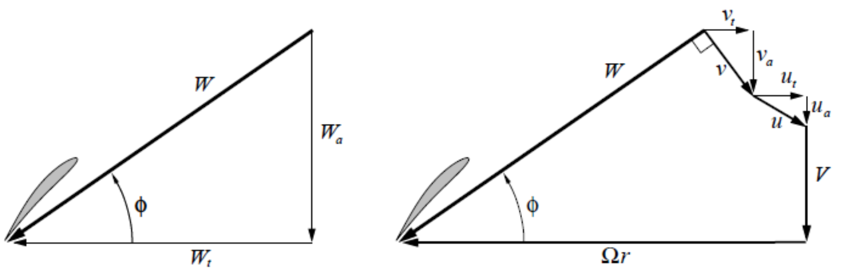


Figure 3-44 Decomposition of blade relative velocity W at a given radial position

Thus, induced tangential velocity is related with the circulation as given in 3-91. Division of ½ in 3-91 is due to the circumferential circuit seeing semi-infinite trailing vortices instead of infinite vortices as illustrated in Figure 3-45.

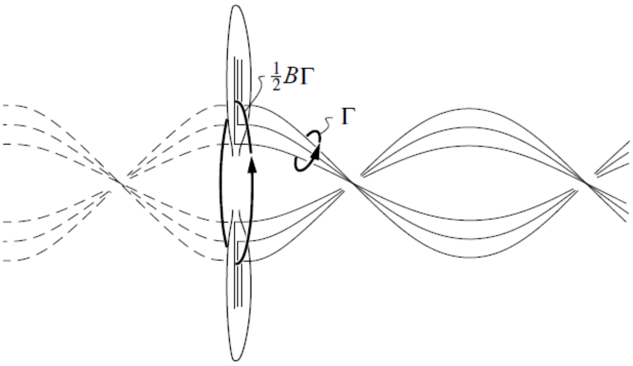


Figure 3-45 Circuits of circulation for swirl/circulation relation[65]

$\overline{v_t}$ is the circumferential-averaged induced tangential velocity and it is assumed to be related by [65]

$$\overline{v_t} = v_t F\sqrt{1+\left(\frac{4\lambda_\omega R}{\pi B r}\right)^2} \qquad \text{3-92}$$

F is the tip loss factor of Prandtl and becomes significant up to the blade tip, and the term in square root becomes significant at the root. Prandtl's tip loss factor is defined as follows;

$$F = \frac{2}{\pi}\arccos(e^{-f})$$
$$f = \frac{B}{2}\left(1 - \frac{r}{R}\right)\frac{1}{\lambda_\omega} \qquad 3\text{-}93$$
$$\lambda_\omega = \frac{r}{R}\tan(\phi) = \frac{r}{R}\frac{W_a}{W_t}$$

By using the relation between 3-91 and 3-92 the relation between local swirl velocity and local circulation is obtained as follows;

$$v_t = \frac{B\Gamma}{4\pi r}\frac{1}{F\sqrt{1 + \left(\frac{4\lambda_\omega R}{\pi B r}\right)^2}} \qquad 3\text{-}94$$

Assuming that the velocity v is perpendicular to W, following relation is obtained;

$$v_a = v_t\frac{W_t}{W_a} \qquad 3\text{-}95$$

After relating the local swirl and the local circulation and reaching a converged solution, local lift and drag coefficients can be calculated by using the 2-D viscous tables. Propeller blade section velocities and angles are illustrated in Figure 3-46 which states that the local angle of attack for a given blade section at a radial location r is defined as;

$$\alpha(r) = \beta - \phi = \beta - \operatorname{atan}\left(\frac{W_a}{W_t}\right) \qquad 3\text{-}96$$

which can be used to obtain the local blade profile lift and drag coefficients;

$$c_l = c_l(\alpha, M) \qquad c_d = c_d(\alpha, M) \qquad 3\text{-}97$$

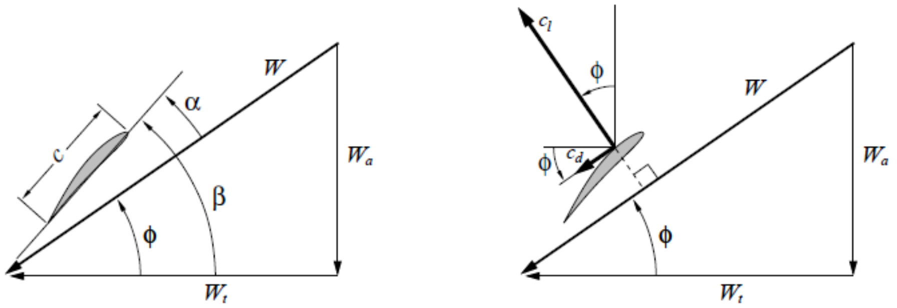


Figure 3-46 Blade section geometry and velocity triangle for a given radial location

Then the local circulation is defined as;

$$\Gamma = \frac{1}{2} W c c_l \tag{3-98}$$

Local solution of circulation with Newton Method

The radial distribution of circulation is calculated for each radial location separately for a given blade chord, twist, lift and drag coefficients, and axial and rotational speeds of propeller. This calculation is achieved by solving the non-linear circulation equation by Newton's method instead of iterating the circulation directly. A dummy variable ψ is defined for the Newton's method as illustrated in Figure 3-47.

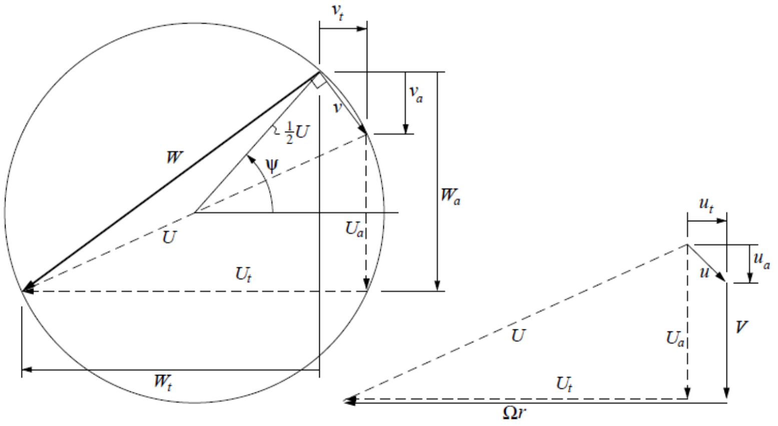


Figure 3-47 Parametrization of velocities by a dummy variable ψ

It is possible to define all the required variables with the parameter ψ. After expressing the all velocity components in terms of the dummy variable ψ finally following equation is obtained [65];

$$R(\psi) = \frac{v_t 4\pi r}{B} F \sqrt{1 + \left(\frac{4\lambda_\omega R}{\pi B r}\right)^2} - \frac{1}{2} W c_l(\alpha, M) \qquad \text{3-99}$$

The Newton residual $R(\psi)$ provides the relation between local lift coefficient and the bound circulation and should be minimized to zero. The newton update of the residual is given as;

$$\Delta\psi = -\frac{R}{\frac{dR}{d\psi}} \qquad \text{3-100}$$

$$\psi \leftarrow \psi + \Delta\psi$$

The convergence of the method is quadratic and the residual R is driven to machine zero quickly.

The relation of Thrust and Torque

Newton iteration is performed for each radial location and overall distribution of circulation $\Gamma(y)$ is determined. Then, it makes the calculation of overall thrust and torque of the propeller possible. The direction pf thrust and torque with respect to inflow angle ϕ is depicted in Figure 3-48.

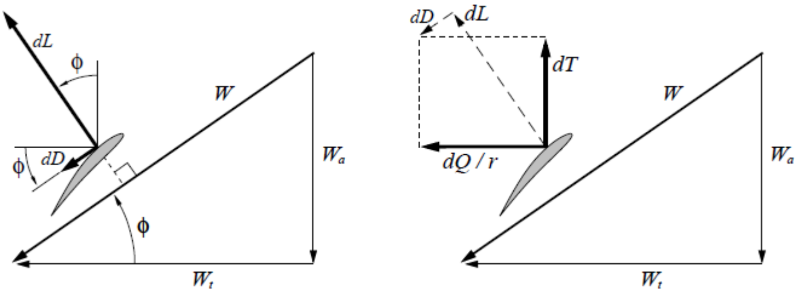


Figure 3-48 Blade section velocities, angles and the local thrust and torque representation

Then, blade profile lift and drag can be represented as

$$dL = B\frac{1}{2}W^2 c_l c dr \tag{3-101}$$

$$dD = B\frac{1}{2}W^2 c_d c dr \tag{3-102}$$

$$dT = B\frac{1}{2}\rho W^2 (c_l cos\phi - c_d sin\phi) c\ dr \tag{3-103}$$

$$dQ = B\frac{1}{2}\rho W^2 (c_l sin\phi - c_d cos\phi) cr\ dr \tag{3-104}$$

As represented in Figure 3-48 the tangential and axial component of total velocity is decomposed as;

$$\begin{gathered} W\ cos\phi = W_t \\ W\ sin\phi = W_a \end{gathered} \tag{3-105}$$

Then, the thrust and torque is represented as in terms of thrust and velocity components as;

$$\begin{gathered} dT = \rho B\Gamma\left(W_t - \frac{c_d}{c_l}W_a\right)dr \\ dQ = \rho B\Gamma\left(W_t - \frac{c_d}{c_l}W_a\right)r\ dr \end{gathered} \tag{3-106}$$

Finally, the total thrust and torque of the propeller can be expressed by taking the integral of 3-106 throughout the blade span;

$$T = \sum_{r=0}^{r=R} dT\ \ , \qquad Q = \sum_{r=0}^{r=R} dQ \tag{3-107}$$

This finalized forces and moments are represented in body frame of the propeller and transformed and translated to the center of gravity in order to obtain the total 6-DoF effects on the air vehicle.

3.5.2 Software Class Diagram

In Figure 3-49 representative sample class diagram of propeller object is provided. As described in the previous chapter Propeller class is inherited from the Rotor class since there are common fields and methods in both models. In the class diagrams all the details are not provided since there are huge numbers of variables and methods.

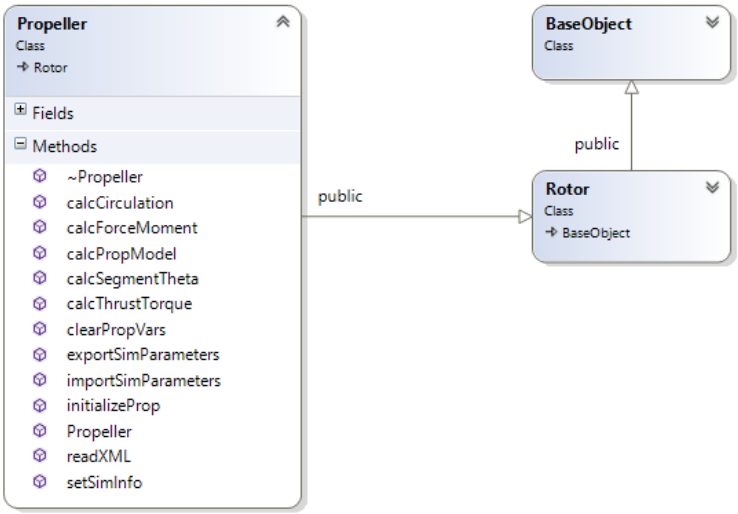


Figure 3-49 Class diagram of Propeller object (All fields are not displayed)

3.6 Interference Models

Aerodynamic performance of components is usually predicted. However, when these components are gathered together as a system their aerodynamic performance is affected by the others due to the effect of interactions. Rotor downwash changes the flow field and other components such as fuselage, wing, and empennage components are affected. Dynamic pressure values and angle of attack of the flow are changed for these components. Interactions occur between individual elements of helicopter and often their effects are nonlinear and dependent on flight condition and thrust level. Due to rotor – aerodynamic component interference the direction and magnitude of the relative air velocity on the aerodynamic component is different from the interference-free condition. This difference in velocity causes sudden forces and moments with respect to aircraft center of gravity which results in different trim condition. For example, interference between main rotor and horizontal tail changes the helicopter pitch attitude in low speed flight regimes.

In GAVM interaction is modeled as a pre-defined lookup tables. By using these look-up tables, the effect of downwash velocity is simulated on the aerodynamic surfaces. It is evaluated that rotor interaction depends on various flight and state parameters. “i”, being any interaction magnitude, is defined as;

$$i = f\{GW, hp, T, V, \alpha, \beta, \theta, \phi, [\beta_0, \beta_{1c}, \beta_{1s}], [\theta_0, \theta_{1s}, \theta_{1c}]\} \qquad \text{3-108}$$

On the other hand, in order to develop a practical mathematical model, the interaction magnitude definition is updated and defined as;

$$i = f\{\lambda, \chi, \alpha\}, \qquad \text{3-109}$$

where;

$$\lambda = f\{GW, hp, T, [\beta_0, \beta_{1c}, \beta_{1s}], [\theta_0, \theta_{1c}, \theta_{1s}]\} \qquad \text{3-110}$$

$$\chi_{wake} = f\{V, \lambda, \alpha, \beta, \theta, \varphi, [\beta_0, \beta_{1c}, \beta_{1s}], [\theta_0, \theta_{1c}, \theta_{1s}]\}, \qquad \text{3-111}$$

where;

$$\chi_{wake} = tan^{-1}(\overline{u_{TPP}}/_{\overline{\lambda_0}}) \qquad \text{3-112}$$

This equation covers level flight interaction thus eliminating α (flight path angle) from the equation. Consequently, the developed interaction models only require wake skew angle and mean inflow velocity as input. “i” represents any interaction magnitude such as dynamic pressure ratios or angle of attack variation.

Wake skew angle is estimated as:

$$\chi_{wake} = tan^{-1}(\overline{u_{TPP}}/_{\overline{\lambda_0}}) \qquad \text{3-113}$$

Mean inflow velocity is defined as depicted with Figure 3-50.

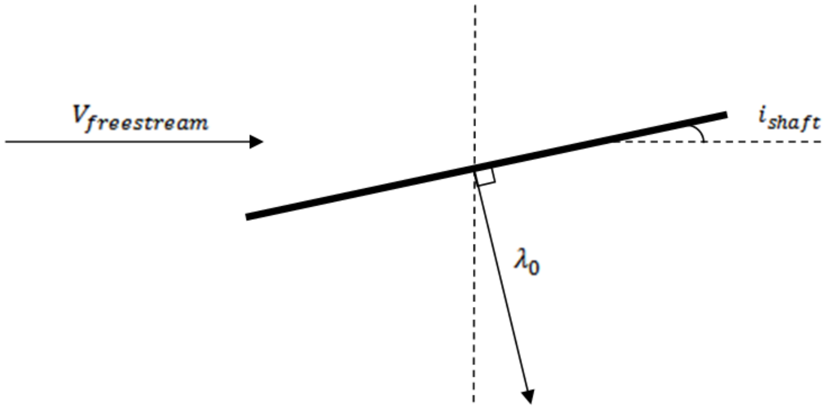


Figure 3-50 Inflow velocity($\boldsymbol{\lambda_0}$) representation

Developed models utilize interaction measures as local dynamic pressure(q) and effective angle of attack ($\alpha_{effective}$). Therefore; when wake skew angle and mean inflow velocity is supplied, change of effective angle of attack and dynamic pressure variation with respect to isolated conditions are obtained. This methodology mainly works for helicopters and requires previous flowfield solutions by using some external flow solvers.

3.7 6-DOF Dynamics Model

The general equations of motion for a rigid air vehicle are derived in the six degree of freedom model and translational and rotational accelerations are obtained, as well as kinematics. There is two axis systems used in this part; the earth fixed axis system $X'Y'Z'$ and the rotorcraft body axis system XYZ as depicted in Figure 3-51.

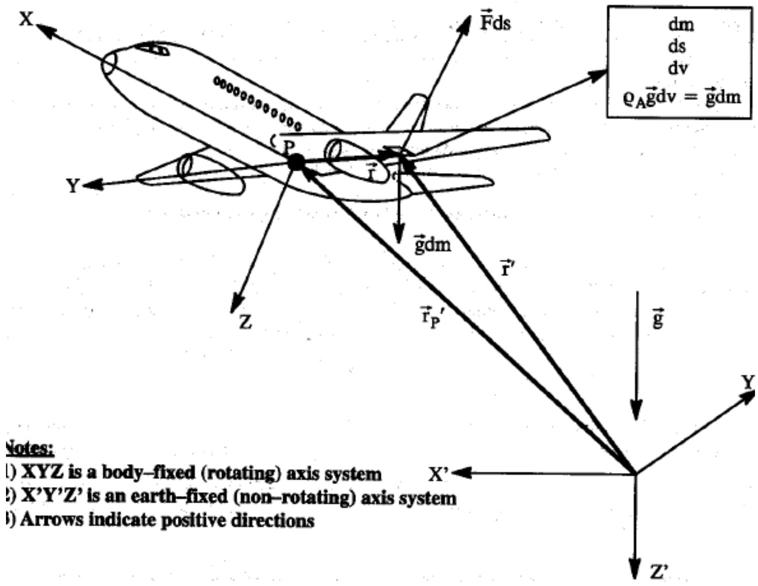

X
$\vec{F}ds$
dm
ds
dv
$\varrho_A \vec{g} dv = \vec{g} dm$
P
Y
$\vec{r}$
$\vec{g}dm$
$\vec{r}'$
Z
$\vec{r}_P{}'$
$\vec{g}$
Y
X'
Z'
Notes:
1) XYZ is a body-fixed (rotating) axis system
2) X'Y'Z' is an earth-fixed (non-rotating) axis system
3) Arrows indicate positive directions

Figure 3-51 Earth fixed and Body fixed Coordinate systems. [69]

Equations of motion derived from linear and angular momentum equations for a rigid body in body fixed frame of reference are summarized as follows:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} (X - mgsin\theta)/m & -qw & rw \\ (Y + mgcos\theta)/m & -ru & pw \\ (Z + mgcos\theta cos\phi)/m & -pv & qu \end{bmatrix} \tag{3-114}$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{yz} & I_{zz} \end{bmatrix}^{-1} \begin{bmatrix} L + I_{yz}(q^2 - r^2) + I_{zx}pq - I_{xy}rp + (I_{yy} - I_{zz})qr \\ M + I_{zx}(r^2 - p^2) + I_{xy}qr - I_{yz}pq + (I_{zz} - I_{xx})pr \\ N + I_{xy}(p^2 - q^2) + I_{xy}pr - I_{zx}qr + (I_{xx} - I_{yy})pq \end{bmatrix} \tag{3-115}$$

And kinematics equations as:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} p + \dot{\psi}\sin\theta \\ qcos\theta - rsin\theta \\ (r\cos\phi + q\sin\phi)\sec\theta \end{bmatrix} \tag{3-116}$$

All the equations above are used the total forces and moments acting on the vehicle center of gravity and dynamic state derivatives (accelerations and rate derivatives) are obtained for rigid air vehicle. These accelerations, rate derivatives and kinematics angles compose nine dynamic state variables in trim algorithm which are body translational velocities $u, v, w,$ body angular rates $p, q, r,$ and Euler angles $\phi, \theta, \psi.$ After that with the help of kinematics and translational velocities, earth fixed states may be obtained as in equation 3-5 with the following equation:

$$V_{NED} = [R(\psi)R(\theta)R(\phi)]^T V_B$$
$$X_{NED} = \int V_{NED}\, dt \qquad 3\text{-}117$$

Therefore with these three velocities and their integral position states X_E, Y_E, Z_E total number of states in dynamic motion of air vehicle becomes twelve. The state vector becomes:

$$[x]=\begin{bmatrix} u \\ v \\ w \\ p \\ q \\ r \\ \phi \\ \theta \\ \psi \\ X_E \\ Y_E \\ Z_E \end{bmatrix}, \qquad [\dot{x}]=\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \\ \dot{p} \\ \dot{q} \\ \dot{r} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ V_N \\ V_E \\ V_D \end{bmatrix}, \qquad 3\text{-}118$$

Positions in NED frame is transformed into the Geodetic coordinates for simulation and navigation purposes.

3.8 Control Allocation Model

In GAVM there are several types of inputs available for each component. There are six types of inputs for a standard rotor object, and four types of inputs for the Wing object and three types for the propeller object. Besides, unconventional configurations such as tilting rotors and multiple rotor air vehicles require allocation of control inputs in order to produce a meaningful input in terms of flight dynamics. For instance, tandem helicopters may use differential lateral cyclic for yawing and quadrotors use collective RPM increment to climb. This nature of unconventional configurations brings the need of a control allocation. Therefore, a control allocation class is introduced into the GAVM class architecture.

ControlMixer class is derived from **BaseObject** class and includes only the **Input** data structure. This class is used to generate upper-level input definitions for the advanced or unconventional configurations. Once an upper level control description is employed in the control mixed class, individual inputs are deactivated and does not taken into trim algorithm. However, their values are updated according to the description in the control mixer class. Control Mixer brings the ability to define the required number of upper-level inputs so that the total system of equations becomes determinate. In other words, if the system has over-determined number of available inputs, the control mixer brings the ability to reduce the active set of inputs such that the trim algorithm achieves a unique solution, instead of infinitely many solutions.

The logic in the **ControlMixer** object is based a simple connection between the available inputs with a proportional gain coefficient. For each mixed control input a gain is determined which reflects the behavior of each input merged into a parent control mixer input. For instance, throttle control is generated for a quadcopter which is expected to increase rotational speed of all propellers collectively, therefore, each propeller is gathered together under a high level control input “throttle” and same gain coefficient is determined for each rotational speed input to provide maximum RPM for each propeller when throttle is set to maximum. This approach is useful for multiple propeller aircrafts when throttle stick is required to send commands to all propellers. If a differential RPM is required for a different type of input, the sign of gain coefficients should be reversed. On the other hand, these high level controls can be defined between different types of components having similar sets of inputs. Given propeller or rotor can be coupled by a wing to tilt together when tilt command is provided. Therefore, trim point during transition phase can be obtained.

Table 3-2 Control definition for a tilting rotor rotorcraft (XV-15)

```
    <ControlMixer active="1">
        <Input name="Colective" value="10" max="40" min="0">
            <Rotor Id="1" input="Collective" K="1.0" />
            <Rotor Id="2" input="Collective" K="1.0" />
        </Input>
        <Input name="Longitudinal" value="0" max="20" min="-20">
            <Rotor Id="1" input="LonCyclic" K="1.0" />
            <Rotor Id="2" input="LonCyclic" K="1.0" />
        </Input>
        <Input name="Lateral" value="0" max="20" min="-20">
            <Rotor Id="1" input="LatCyclic" K="1.0" />
            <Rotor Id="2" input="LatCyclic" K="1.0" />
        </Input>
        <Input name="Pedal" value="0" max="20" min="-20">
            <Rotor Id="1" input="LonCyclic" K="1.0" />
            <Rotor Id="2" input="LonCyclic" K="-1.0" />
        </Input>
    </ControlMixer>
```

In Table 3-2 input system definition of a tilt-rotor rotorcraft is illustrated. This control mixer creates a Collective input which directly actuates the both rotors swash plates to produce the same collective. Longitudinal and lateral cyclic commands are also coupled. Pedal input is defined by a differential longitudinal cyclic for two rotors. Therefore, with this definition total control inputs for a tilt rotor is reduced to four while it was six.

Table 3-3 Control definition for a 8-kg Quadrotor

```xml
<ControlMixer active="1">
    <Input name="throttle" value="1.0" max="1.25" min="0">
        <Propeller Id="1" input="RPM" K="12000.0" />
        <Propeller Id="2" input="RPM" K="12000.0" />
        <Propeller Id="3" input="RPM" K="12000.0" />
        <Propeller Id="4" input="RPM" K="12000.0" />
    </Input>
    <Input name="roll" value="0" max="3" min="-3">
        <Propeller Id="3" input="RPM" K="-1000.0" />
        <Propeller Id="4" input="RPM" K="1000.0" />
    </Input>
    <Input name="pitch" value="0" max="3" min="-3">
        <Propeller Id="1" input="RPM" K="1000.0" />
        <Propeller Id="2" input="RPM" K="-1000.0" />
    </Input>
    <Input name="yaw" value="0" max="3" min="-3">
        <Propeller Id="1" input="RPM" K="1000.0" />
        <Propeller Id="2" input="RPM" K="1000.0" />
        <Propeller Id="3" input="RPM" K="-1000.0" />
        <Propeller Id="4" input="RPM" K="-1000.0" />
    </Input>
</ControlMixer>
```

Similar control definition is defined for the quadrotor control in Table 3-1 by providing a collective RPM for throttle input; differential RPM for roll pitch and yaw inputs. In this case total number of inputs is not changed; control mixer is used to generate meaningful types of inputs in terms of flight dynamics.

3.9 Trim Algorithm

Condition of “trim” can be defined as a state of dynamic equilibrium which means that a trimmed vehicle sustain its dynamic equilibrium when unperturbed. State of dynamic equilibrium requires that the total forces and moments acting on the center of gravity of aircraft is zero [70]. For any static or dynamic system there should be an equilibrium point at which the designer desires to evaluate the system performance, specifications and response. The generic rotorcraft model provides creation of isolated rotor, propeller, or wing component or a dynamical system like airplane, rotorcraft or compound air vehicles. Trim algorithm is designed as a generic routine which provides the desired state variables, and inputs for a pre-defined target trim conditions under the constraints. Isolated rotor and propeller can be trimmed for a target thrust and moment values, and an air vehicle can be trimmed for a given flight condition.

Trim algorithm implemented in this thesis perceives the designed system as a one total system formed by small sub-components. Each system component has its own inputs (u), states (x), and outputs (y) as depicted in Figure 3-52. These objects are gathered together and trim algorithm uses classical Newton’s method of unconstrained optimization to make the system converged into the desired condition.

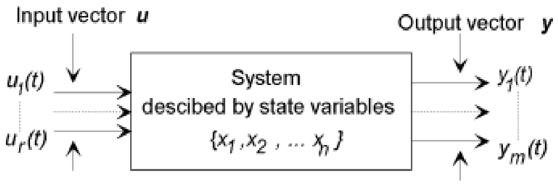


Figure 3-52 Standard input, output and state (DoF) schematic for a component model [71]

For instance, rotor has inputs as collective and cyclic controls, thrust, and torque and hub moments for outputs. Wing object similarly has flap and aileron inputs and several outputs parameters as forces and moments and induced velocity distributions. Mainframe object to which all components are attached has 6-DOF Euler system states and several outputs as velocities, accelerations and translational and angular positions. Whole system objects are merged and total system of nonlinear equations is composed. Total system is interpreted as a system of equations when the target and variable trim parameters are defined. These parameters should be defined properly to ensure the number of equations and number of unknowns is equal.

Parameter definition for the trim analysis is done by using a configuration XML file. Moreover, trim configuration can also be changed between two successive trim analyses.

Trim variables are selected between the system states and inputs, in other hands; trim targets are selected between the system state derivatives and outputs. Cross derivatives in 3-119 are calculated by perturbing the states and inputs and represent the sensitivity of state derivatives and outputs.

$$\begin{bmatrix} dx/dt \\ y \end{bmatrix}_{target} = \begin{bmatrix} \partial\dot{x}/\partial x & \partial\dot{x}/\partial u \\ \partial y/\partial x & \partial y/\partial u \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix}_{variable} \tag{3-119}$$

For Newton optimization, a function $G(\tilde{x})$ is defined which represents the difference between the model and the reference. Trim function should optimize $G(\tilde{x})$ to zero. $\tilde{x}$ is defined as the vector of variable inputs and states.

$$G(\tilde{x}) = \begin{bmatrix} \partial\dot{x}/\partial x & \partial\dot{x}/\partial u \\ \partial y/\partial x & \partial y/\partial u \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix}_{variable} - \begin{bmatrix} dx/dt \\ y \end{bmatrix}_{desired} \tag{3-120}$$

Defining a feasible set $\tilde{x}'$ where $G(\tilde{x}') = 0$ and applies Newton's method, where $\Delta\tilde{x}$ is the array of delta states and inputs through each iterate.

$$\Delta\tilde{x} = -\left(DG(\tilde{x})\right)^{-1} \left(G(\tilde{x}) - \begin{bmatrix} dx/dt \\ y \end{bmatrix}_{desired} \right) \tag{3-121}$$

In order to achieve convergence, number of equations should not be less than number of unknowns. Newton iteration continues until the convergence is achieved.

$$\tilde{x}_{k+1} = \tilde{x}_k + \Delta\tilde{x} \tag{3-122}$$

3.10 Engine Model

Engine object is modeled in order to analyze point and mission performance parameters. Engine model is made up from look-up tables for available power and specific fuel consumption (SFC) for varying altitude, atmospheric temperature, airspeed and different power settings. There may be defined different engine ratings for different available power and fuel consumption values.

Table lookup methodology requires an interpolated available shaft power and specific fuel consumption tables. By using these values GAVM is able to estimate:

- Service and Absolute Ceiling altitudes,

- Rate of Climb curves,
- Payload and Range chart,
- Fuel consumption curves,
- Best range and best endurance velocities,
- Maximum velocity,

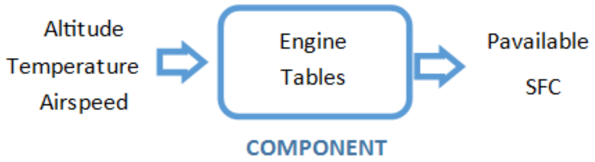


Figure 3-53 Diagram showing the Engine Model

3.10.1 Engine Table Lookup Methodology

Table lookup procedure is implemented for calculation of available power values and fuel consumption rates in order to perform point performance analysis. Besides a mathematical model of a given air vehicle, the designer need an engine model to determine aircraft's performance parameters such as ceiling altitudes, cruise velocities and range etc. All of the point performance parameters described in the previous chapter needs available power and specific fuel consumption. Therefore, generated engine deck model should cover the flight envelope of the air vehicle. Implemented engine model requires 3-D tables for available power. The dimensions are altitude, temperature and airspeed respectively. Sample schematic is shown for available power tables as in Figure 3-54. Available power tables may be generated for different ratings and inoperative conditions for multi-engine air vehicles.

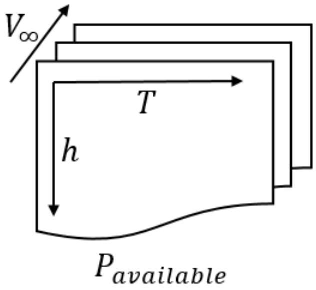


Figure 3-54 Representative sketch of 3-D available power tables

Another required parameter which represents the amount of fuel burned for a specific required power is the Specific Fuel Consumption (SFC). SFC values vary with altitude, temperature, airspeed and produced power. Therefore SFC tables are 4-D tables as in Figure 3-55 which is required to calculate the amount of fuel consumed.

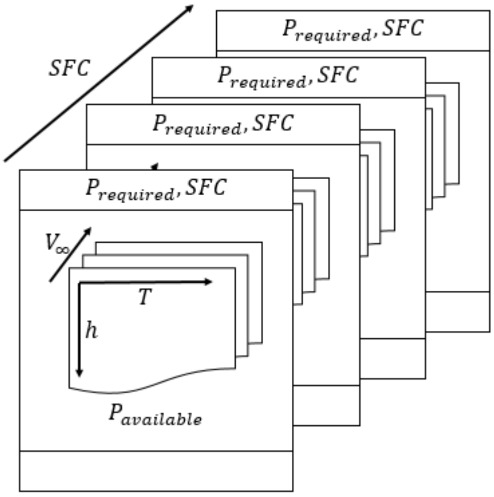
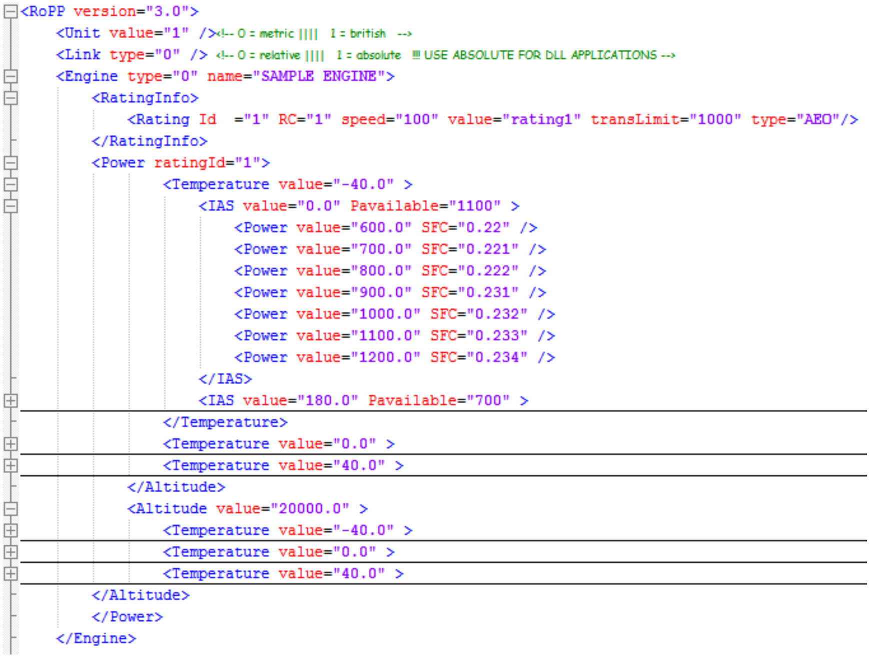


Figure 3-55 Representative 4-D Specific Fuel Consumption Tables

These generated tables are used for 3-D and 4-D table lookup for a given altitude, temperature, airspeed and power setting respectively. Example XML file is of the format as in Figure 3-56. In Figure 3-56 different rating values are defined with different mechanical limits. Mechanical limits are defined to simulate the mechanical limit of the available power and these limits are applied into the performance analysis if the available power is higher than the mechanical limit.



```
<RoPP version="3.0">
    <Unit value="1" /><!-- 0 = metric ||||  1 = british  -->
    <Link type="0" /> <!-- 0 = relative ||||  1 = absolute  !!! USE ABSOLUTE FOR DLL APPLICATIONS -->
    <Engine type="0" name="SAMPLE ENGINE">
        <RatingInfo>
            <Rating Id  ="1" RC="1" speed="100" value="rating1" transLimit="1000" type="AEO"/>
        </RatingInfo>
        <Power ratingId="1">
                <Temperature value="-40.0" >
                    <IAS value="0.0" Pavailable="1100" >
                        <Power value="600.0" SFC="0.22" />
                        <Power value="700.0" SFC="0.221" />
                        <Power value="800.0" SFC="0.222" />
                        <Power value="900.0" SFC="0.231" />
                        <Power value="1000.0" SFC="0.232" />
                        <Power value="1100.0" SFC="0.233" />
                        <Power value="1200.0" SFC="0.234" />
                    </IAS>
                    <IAS value="180.0" Pavailable="700" >
                </Temperature>
                <Temperature value="0.0" >
                <Temperature value="40.0" >
            </Altitude>
            <Altitude value="20000.0" >
                <Temperature value="-40.0" >
                <Temperature value="0.0" >
                <Temperature value="40.0" >
        </Altitude>
        </Power>
    </Engine>
```

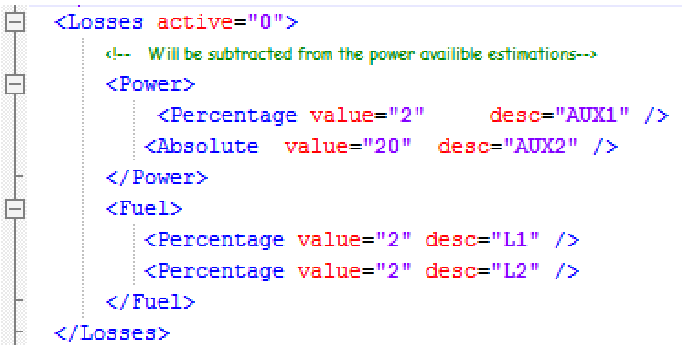
Figure 3-56 Engine Table XML file

3.10.2 Engine Losses Model

In every air vehicle there exists engine related losses which may occur due to the for example, aerodynamic losses in inlet exhaust, and power consumed by the accessories.

The engine related losses may be supplied by the manufacturer and may be integrated into the engine model. In such a case, it is more convenient to generate lookup tables for available power and fuel consumption values where losses are existed in these tables.

If the engine related tables are generated without any losses, they should be defined externally. In GAVM, the losses can be defined either as percentage losses or absolute losses as given in Figure 3-57. The implementation order of the losses will be successively.



```
<Losses active="0">
    <!--   Will be subtracted from the power availible estimations-->
    <Power>
         <Percentage value="2"      desc="AUX1" />
        <Absolute  value="20"  desc="AUX2" />
    </Power>
    <Fuel>
        <Percentage value="2" desc="L1" />
        <Percentage value="2" desc="L2" />
    </Fuel>
</Losses>
```

Figure 3-57 XML configuration file for Engine related losses

3.10.3 Software Class Diagram

Software class diagram of Engine object is illustrated as in Figure 3-58.

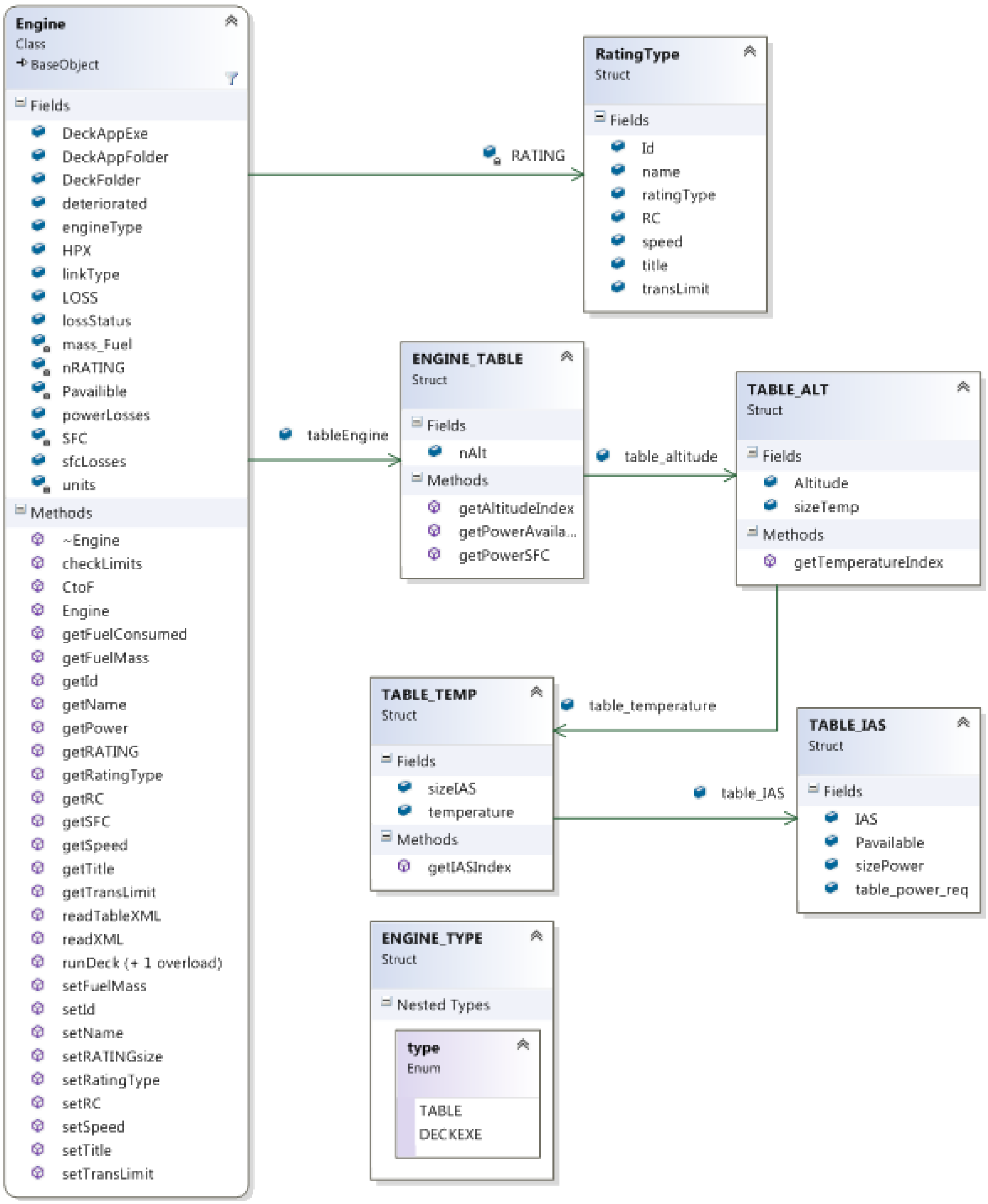


Figure 3-58 Class diagram of Engine object

3.11 Main Class: Simulation

There are two main classes implemented in GAVM software which are **Simulation** and **Performance. Performance** class is derived from **Simulation** class and it includes the required algorithms for performance analyses.

Simulation class is defined to connect each component with each other to perform the required analyses. In Simulation class there are several objects and methods which compose the main architecture of GAVM software. Import functions load the required XML files, count the objects, allocate required amount of memory. There are instances of all components of the air vehicle in Simulation class. They are populated and connecting with each other and final air vehicle is stored in the Simulation class for further analysis. Trim algorithm is implemented into the Simulation class in which the total system inputs, states and outputs are managed.

Performance class is also derived from the simulation class since they both requires common functions and features. Differences in the Performance class are the specific algorithms for performance analyses and engine related structures. **Performance** class uses the functions related to trim and one-step model execution in **Simulation** class. There are several performance algorithms to calculate hover and service ceilings, cruise velocities, range estimations, best rate of climb values and maximum rate of climb values. Every performance parameter is estimated by trimming the full non-linear mathematical model of the aircraft and uses the **Engine** class where available power values are imported.

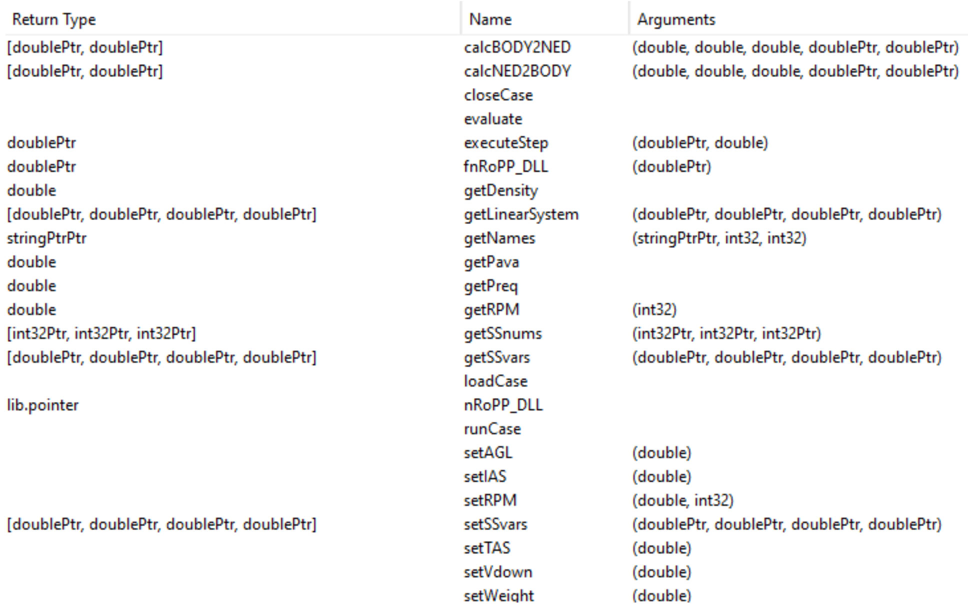
3.12 Shared Library Versions

Standard GAVM is compiled as a standalone executable project which requires input files for analysis definitions and executes the analyses described in the input files. Standalone application is useful since it does not require extra software and tools to work with. However, the dynamic simulation environment is not implemented into the standalone project since the standalone project provides limited authority to the user and requires coding of several I/O routines.

Therefore, GAVM is modified as a shared library (DLL) in Windows OS to be used with simulation purposes interactively. C++ DLL version of the software is compiled with several export functions. Exported functions are generated for users to generate, modify the rotorcraft model and execute the simulation. Aircraft model can be created and all analyses, which can be done by standalone application, can be conducted by using this shared library version. This DLL requires calling by any proper language or tool.

There are some functions in DLL which are available as a wrapper to communicate the core software with the other platforms which loads the DLL. Available shared library functions are used for:

- Loading the aircraft model file
- Trimming for a given analysis XML
- Setting or getting the inputs, states and outputs of whole air vehicle
- Evaluation of all dynamics for one time-step simulation
- Integration of states for a time-step
- Dumping state space representation
- Running Engine model for available power and fuel consumption
- Getting required power



Return Type	Name	Arguments
[doublePtr, doublePtr]	calcBODY2NED	(double, double, double, doublePtr, doublePtr)
[doublePtr, doublePtr]	calcNED2BODY	(double, double, double, doublePtr, doublePtr)
	closeCase	
	evaluate	
doublePtr	executeStep	(doublePtr, double)
doublePtr	fnRoPP_DLL	(doublePtr)
double	getDensity	
[doublePtr, doublePtr, doublePtr, doublePtr]	getLinearSystem	(doublePtr, doublePtr, doublePtr, doublePtr)
stringPtrPtr	getNames	(stringPtrPtr, int32, int32)
double	getPava	
double	getPreq	
double	getRPM	(int32)
[int32Ptr, int32Ptr, int32Ptr]	getSSnums	(int32Ptr, int32Ptr, int32Ptr)
[doublePtr, doublePtr, doublePtr, doublePtr]	getSSvars	(doublePtr, doublePtr, doublePtr, doublePtr)
	loadCase	
lib.pointer	nRoPP_DLL	
	runCase	
	setAGL	(double)
	setIAS	(double)
	setRPM	(double, int32)
[doublePtr, doublePtr, doublePtr, doublePtr]	setSSvars	(doublePtr, doublePtr, doublePtr, doublePtr)
	setTAS	(double)
	setVdown	(double)
	setWeight	(double)

Figure 3-59 Available exported functions from shared library version of GAVM

3.13 Analysis Types

3.13.1 Point Trim and Simulation

Point trim and simulation analysis is defined as a single flight condition by defining the required set of inputs and outputs. Generally, the target outputs are airspeed, climb rate, sideslip angle and climb rate. Mass and environmental conditions are the parameters which need to be defined for a trim condition. Generic trim algorithm which is based on Newton’s optimization searches the dynamic equilibrium around the target point within the defined tolerance. Required outputs are printed out and whole system parameters are stored into the code for further analyses.

In shared library version of the code, this trim type is used to initiate a simulation or use a core air vehicle model by calling required functions for different type of analyses.

3.13.2 Sweep Trim

Sweep trim option is composed of several point trim conditions which are analyzed successively. Each trim dataset is used as an initial condition for the next trim condition in order to increase the convergence and execution time of the algorithm.

Several trim conditions can be defined and analyzed by the trim XML file. Initial conditions and target parameters can be changed any time between two target condition XML nodes. Therefore, quasi-steady maneuver conditions can be analyzed by this option. Output of each trim condition is printed out to a brief text file, detailed text file and plot files for the distribution of some significant parameters on the objects.

3.13.3 Point Performance Analysis

Point performance analysis includes several types of performance analyses in **Performance** class which is derived from the parent **Simulation** class. Point performance analyses are used to determine some performance parameters of a modeled aircraft for a given point condition. These analysis types are summarized as:

- HOGE/HIGE Ceiling for hovering vehicles
- Maximum Vertical and Oblique Rate of Climb
- Required Power Sweep
- Fuel Consumption Sweep
- Cruise Velocities (Best Range - Best Endurance - Maximum)
- Payload and Range Estimations
- Absolute Ceiling and Service Ceiling

3.14 Output Object

Air vehicles have different components for generating propulsive force, creating control forces and moments, and surfaces to be modeled mathematically to calculate aerodynamic, dynamic and structural parameters. Each component of the air vehicle has different effect on the overall performance. Therefore, one may need detailed input sets for air vehicle components. For instance, peak to peak rotor loads, lift and drag distribution along the wing, rotor induced velocity distribution and angle of attack distribution are some of the required data for a specified analysis condition. In this context, each modeled component has an instance in the **Output** class which represents the detailed set of data either obtained by analysis or derived for just dumping out.

There are aerodynamic, structural, flight dynamics, loads, and vibration and control parameters on the flying vehicle which need to be provided to the user in order to assess the required data set. Therefore, air vehicle architecture is modeled in **Output** class similar to the **Simulation** class. After the analysis is finalized, output class generates parameters for each component and prints the output files. Following table summarizes the available output parameters for each component to observe.

Table 3-4 Definition of Output Parameters of Components

CLASS	OUTPUT PARAMETERS
ROTOR / PROPELLER	• Azimuthal distributions of: o Induced velocity o Mach number o Effective Angle of Attack o Flapping and Lagging angles and rates o Blade sectional forces and moments • Figure of Merit, Thrust weighted chord and solidity, Lock Number, Tip speed, Inflow correction factor, Geometric parameters, mass and inertia properties • Trim condition and active trim controls, • Hub and CG forces and moments, • Wake skew angle and mean induced velocity • Body velocities, rates and accelerations, forces and moments at hub and CG • Trust and Power coefficients and dimensional values
WING	• Geometric parameters of the wing object • Active trim inputs. (Control surface deflections, X-Y-Z tilting orientations) • Body velocities, rates and accelerations, forces and moments at aerodynamic center and CG • Lift, drag, angle of attack and downwash distributions if LLT is selected. • Reference length and reference area if the wing table is provided.
MAINFRAME	• Total weight, CG, and Inertia information • 6-DOF dynamics related parameters • Angle of attack and angle of sideslip • True, and Indicated Airspeed • Rate of climb, true heading, flight path angle, • Atmosphere related parameters
BODY	• Body velocities, rates and accelerations, forces and moments at aerodynamic center and CG • Reference length and reference area that the coefficients are dimensionalized.
TRIM	• Trim related parameters such as trim variables and trim targets. Trim condition and trim solution. Linearized state space representation of whole system.

CHAPTER 4

VALIDATION OF ISOLATED COMPONENTS

4.1 Rotor Model Validation

The validation study of the rotor model is conducted by comparing the analysis results with the available experimental data in the literature. Data obtained from wind tunnel test campaign of Sikorsky S-76 Helicopter Main Rotor is used as an experimental validation case for GAVM software. Experimental test campaigns cover hover and forward flight validations. The case provides the power coefficient versus thrust coefficient curves for different collective sets and different wind conditions.

S-76 main rotor has four blades, and 22-feet tip radius and SC1095 [72] airfoil geometries distributed along the span with -10° degrees linear twist. Figure 4-1 depicts the main rotor blade parameters and blade geometry. Table 4-1 summarizes the properties of S-76 main rotor geometry and this model is generated in a proper format as GAVM input XML file.

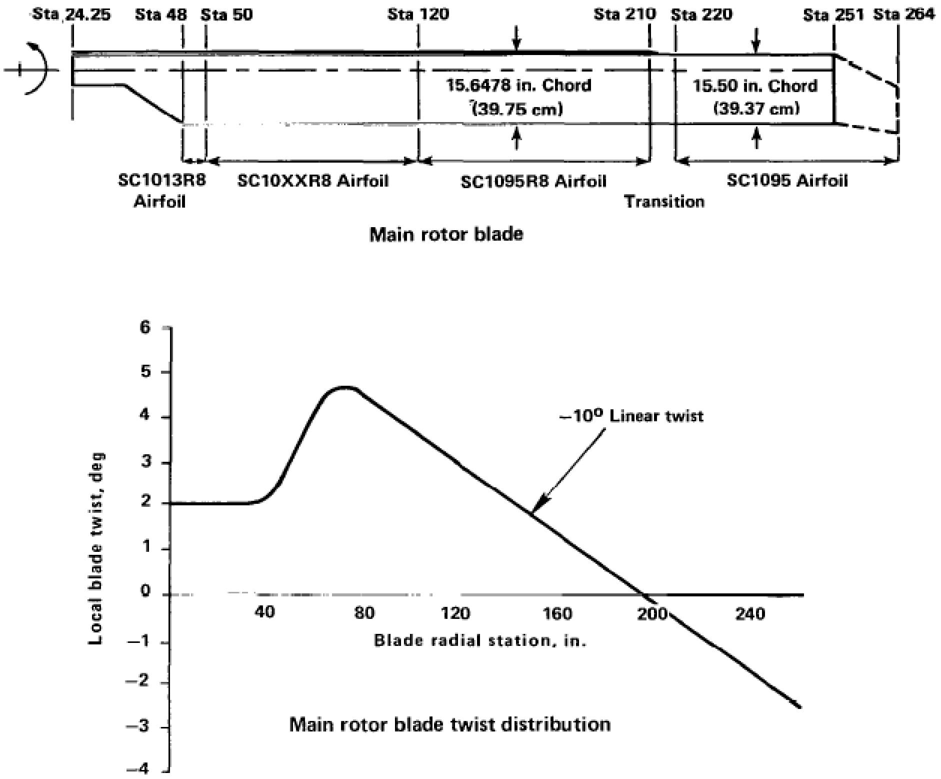


Figure 4-1 Geometry details of S-76 Main Rotor Blade [73]

Table 4-1 Geometric Properties of Main Rotor Blade

Main Rotor Parameters

Radius [m]	6.7
Nominal Chord [cm]	39
Solidity Ratio	0.0748
# of blades	4
Airfoils	SC1095 & SC1095R8
Flapping Hinge Offset	3.79%
Lock Number	11.6
100% RPM	293
100% ΩR	675 fps

After modeling the main rotor, wind tunnel trim option is activated and several conditions are analyzed with GAVM to compare and validate the **Rotor** object.

First set of data is obtained from a full scale helicopter rotor test in the NASA AMES 40 by 80 feet Wind Tunnel [74]. Second set of data is again obtained from the NASA AMES 80 by 120 feet Wind Tunnel [75]. Lastly, whirl tower test results are obtained from Sikorsky Aircraft Division [73].

Test case is S76 rotor model which is tested at NASA AMES Wind Tunnels and Sikorsky Whirl Tower. In NASA AMES 40x80 wind tunnel a full-scale helicopter rotor was tested to measure the performance, loads and noise characteristics of rotors with various tip geometries. For this report only the performance results of these tests are used. In NASA AMES 80x120 wind tunnel a full-scale rotor is tested to measure performance and loads. In Sikorsky Whirl Tower tests a full scale rotor is used for the estimations of main rotor performance.

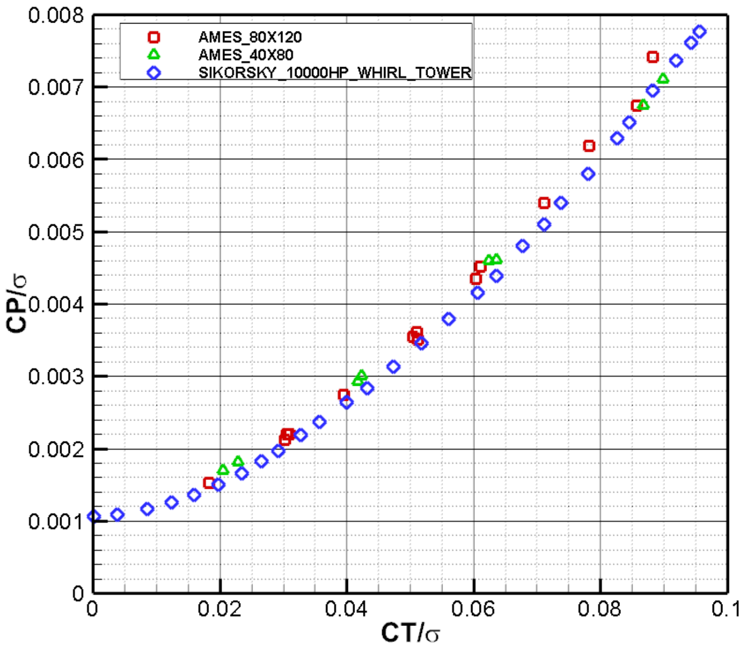


Figure 4-2 Hover case for S-76 MR Blade AMES Wind Tunnel & Sikorsky Whirl Tower Tests

Test results are compared among themselves and it is realized that wind tunnel tests over predicts the power required and there is also more scatter in the wind tunnel data compared to whirl tower in Figure 4-2. It is assessed that the over estimation of hover power required in wind tunnel tests is caused by the effect of ground and/or ceiling of the tunnel on thrust of rotors [76]. Approximate streamlines in wind tunnel hover tests are similar as in Figure 4-3.

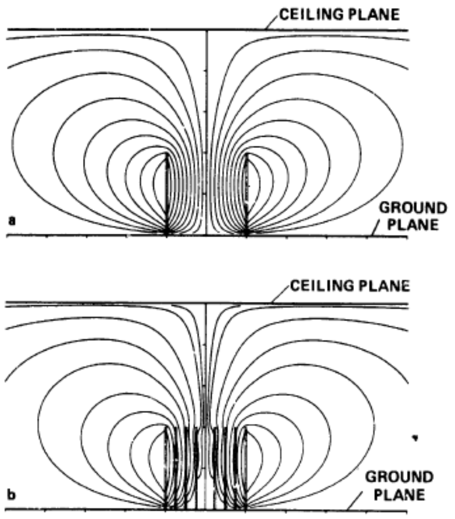


Figure 4-3 Streamline patterns of axisymmetric flow fields induced by vortex wakes when ground and ceiling planes are present. (a) One vortex wake; (b) four vortex wakes [76]

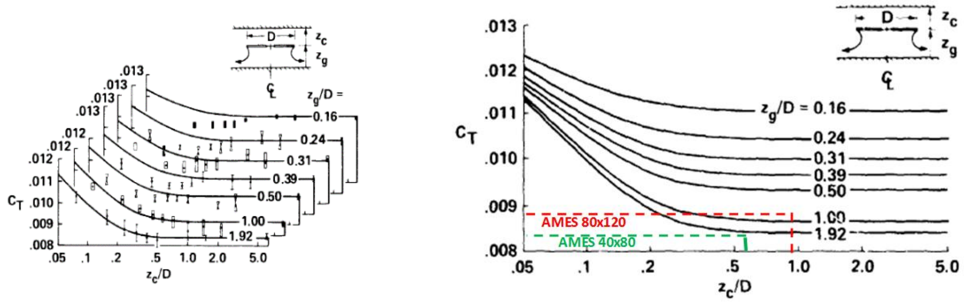


Figure 4-4 Comparison of thrust measured on 0.324-m-dia rotor in Wind Tunnel with values predicted by theory based on approximate image system [76]

According to the Figure 4-4, AMES 80x120 wind tunnel tests are examined and concluded that the overestimation in hover case due to tunnel effects is ~15% for power and ~10% for thrust. When compared to the Sikorsky whirl tower data the value comes up as 7-8% which was expected.

Therefore it is concluded that for small advance ratios (i.e. $\mu < 0.055$) whirl tower tests are less affected from test environment than the wind tunnel tests.

4.1.1 Hover Analysis

Figure 4-5 and Figure 4-6 compare the GAVM rotor model results and experimental data, finite state inflow models (FLIGHTLAB) and Free Wake results (CAMRAD II and CHARM). Results show that GAVM Rotor Model with Pitt Peter's dynamic state inflow model shows consistency with the experimental data.

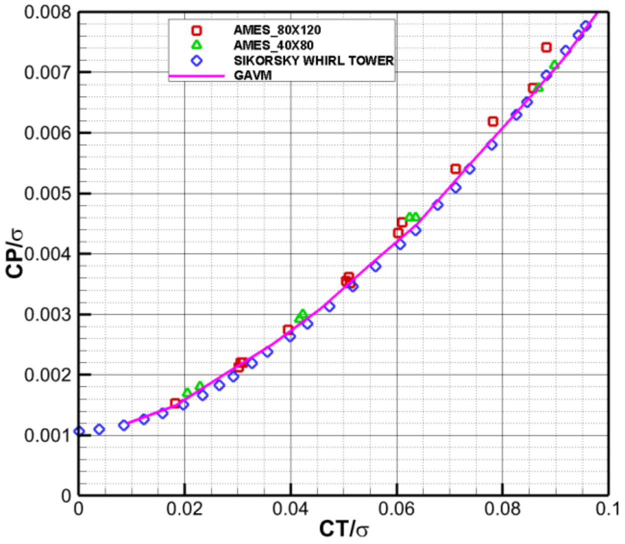


Figure 4-5 Hover comparison of GAVM S-76 Rotor Model with test data

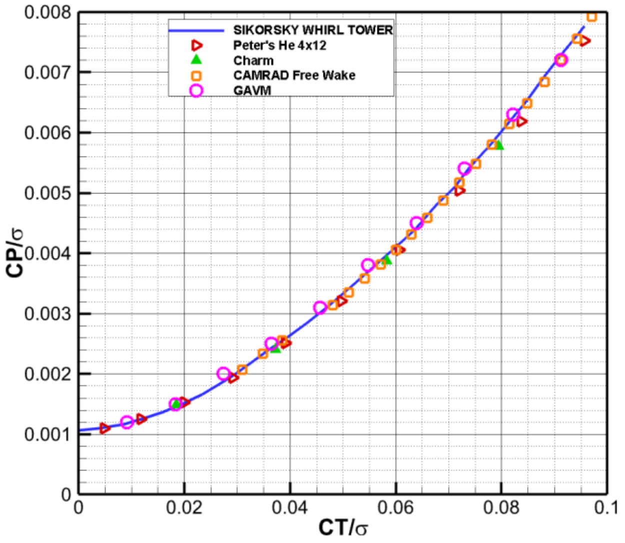


Figure 4-6 Comparison of blade element methods having different inflow & wake models

As stated in inflow modeling chapter Free Vortex-Wake methods (CAMRAD II) gives more precise hover power required estimations than the other blade element methods. Only 4x12 Peters-He solutions in FLIGHTLAB which is computationally expensive stay close to test data. This difference in these methods requires the use of inflow correction factors (κ) up to 1.15. This factor is applied directly applied directly to induced velocity in FLIGHTLAB and GAVM. As a result, GAVM Rotor Model hovering rotor performance results provides matched data when compared with other commercial software and experimental data.

4.1.2 TAI HC-1 Main Rotor Whirl Tower Validation

HC-1 is an indigenous unmanned helicopter development project of Turkish Aerospace Industries Inc. (TAI). HC-1 has a 3 meter two bladed main rotor and specifications are given in

Table 4-2. There is whirl tower test data which is conducted in TAI Whirl Tower facility [77], [78]. Isolated rotor validation with HC-1 Main Rotor is conducted and results are compared. Scatter in the test results is because there is more than one test conducted at different times.

Table 4-2 Specifications of HC-1 Main Rotor

Blade Span	3 m
Nominal rotational speed	540 rpm
Tip Mach # Interval	0.5-0.8
Twist Angle (linear)	16°
Cone Angle	2°
Root-Cut Out	10%

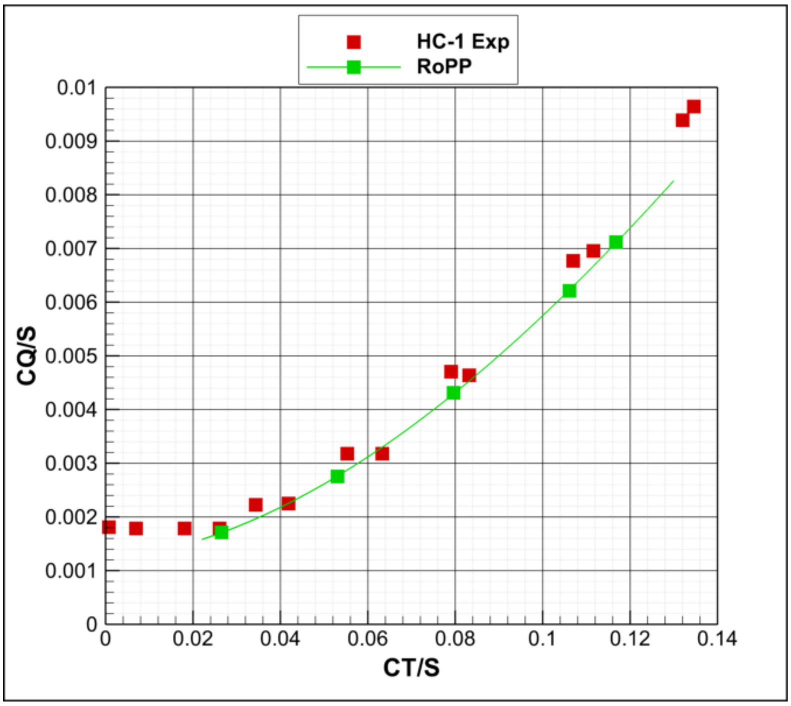


Figure 4-7 HC-1 Isolated Rotor analysis results comparison with Experimental Data

In Figure 4-7 thrust coefficient vs. torque coefficient results are obtained from RoPP HC-1 Isolated rotor analysis. For low loaded cases results shows more consistency than high loading as expected since the implemented inflow model cannot capture the 3-D flow effects such as tip vortices and wake contraction. However, for the overall assessment RoPP Isolated Model is validated and the difference between the test data is evaluated as negligibly small and may be correlated with the small corrections.

4.1.3 Forward Flight Analysis

Forward flight analysis is conducted for the advance ratios in AMES 80x120 Wind Tunnel tests. GAVM rotor model is trimmed and analyzed for different shaft tilt angles (α_s) and different forward velocities. Moreover, corrections are applied onto the analyses results to reflect the ground and ceiling effects.

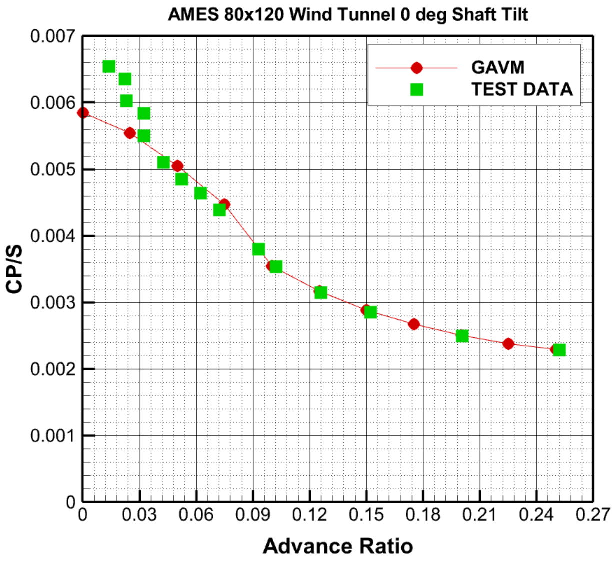


Figure 4-8 Rotor Power Coefficient function of advance ratio $\alpha_s = 0$, $\frac{C_T}{\sigma} = 0.080$

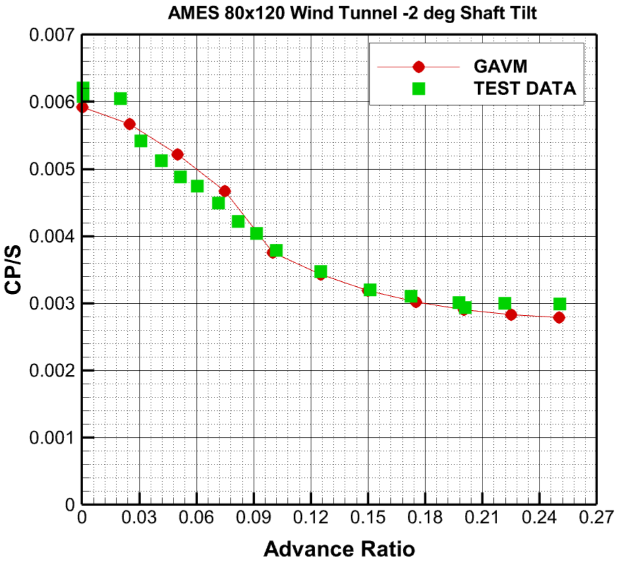


Figure 4-9 Rotor Power Coefficient vs. advance ratio $\alpha_s = -2°$, $\frac{C_T}{\sigma} = 0.080$

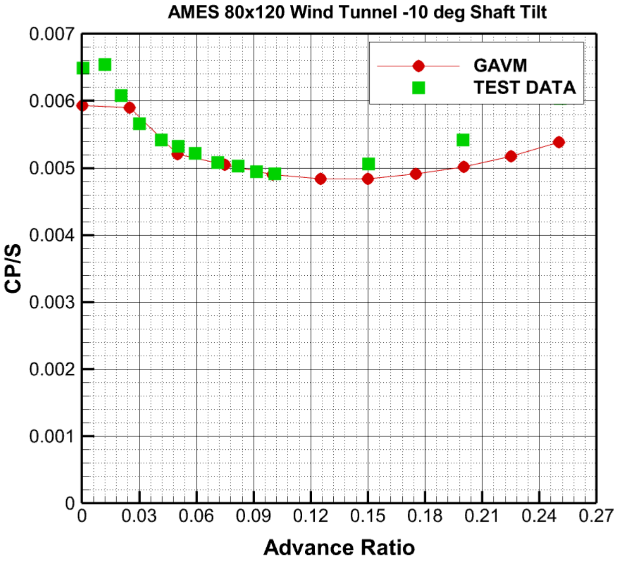


Figure 4-10 Rotor Power Coefficient vs. of advance ratio $\alpha_s = -10°$, $\frac{C_T}{\sigma} = 0.080$

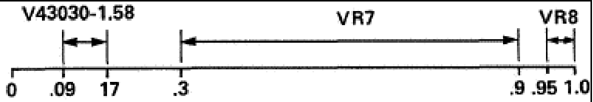
In Figure 4-8, Figure 4-9, and Figure 4-10 forward flight performance of the isolated S-76 main rotor is analyzed and compared with the experimental data. Forward flight analyses results are corrected for wind tunnel effects for small advance ratios (i.e. $\mu < 0.055$). Results are obtained by trimming the isolated rotor in wind tunnel for different shaft tilt angles. Trim target is taken as the thrust coefficient divided by the solidity which equals to $\left(\frac{\mathbf{C_T}}{\boldsymbol{\sigma}} = \mathbf{0.080}\right)$. For all cases, results obtained by GAVM Rotor Model in forward flight shows consistency.

In this validation work different methods are discussed and different tools are used in order to compare GAVM with the Sikorsky S-76 Helicopter Main Rotor performance in hover and forward flight cases. However, it is concluded that in wind tunnel tests hover power required is overestimating due to ground and ceiling effects. Therefore, these results are required some corrections before being used. The most reliable test data belongs to Sikorsky whirl tower and it is taken into account while comparing the hover performance test results.

4.1.4 XV-15 Advanced Technology Rotor Blades Hover Analysis

XV-15 is a tilting rotor aircraft having several available experimental data for validation studies. An isolated rotor performance test campaign is selected as a validation case for rotor model of GAVM [79]. XV-15 has 25 feet diameter main rotor having three blades. Specifications of XV-15 rotor blade are given in Table 4-3. Structural and rotor dynamics properties of XV-15 rotor blade is found in the reference of XV-15 CAMRAD model [80].

Table 4-3 Specifications of XV-15 Tilt Rotor Blade

Proprotor		
# of blades	3	
RPM (Helicopter)	589	[rev/min]
Geometry		
Radius	3.81	[m]
Mean Chord	0.411	[m]
Solidity	0.103	
Airfoil Profiles (VR7 & VR8 tip)	 V43030-1.58 VR7 VR8 0 .09 17 .3 .9 .95 1.0	
Twist	-47.0	[deg]
Precone angle	2.5	[deg]

Hover tests of XV-15 rotor are conducted in Ames Research Center. The main purpose of the tests is to measure the hover performance of the rotor system. Rotational speed of the rotor ranges from such that tip Mach number varies from 0.35 Mach to 0.73 Mach. In Figure 4-11 hover tests of rotor blade are compared with GAVM generated XV-15 rotor model and open source rotor analysis software XRotor [81]. Test results show consistency when compared with both GAVM and XRotor. GAVM calculates the required power slightly lower than test results for a specific thrust level. This small difference may be caused due to un-modeled 3-D flow effects which may be correlated by using inflow correction factor. As a result, behavior of CT/S and CP/S curve for XV-15 rotor is consistent with the test data.

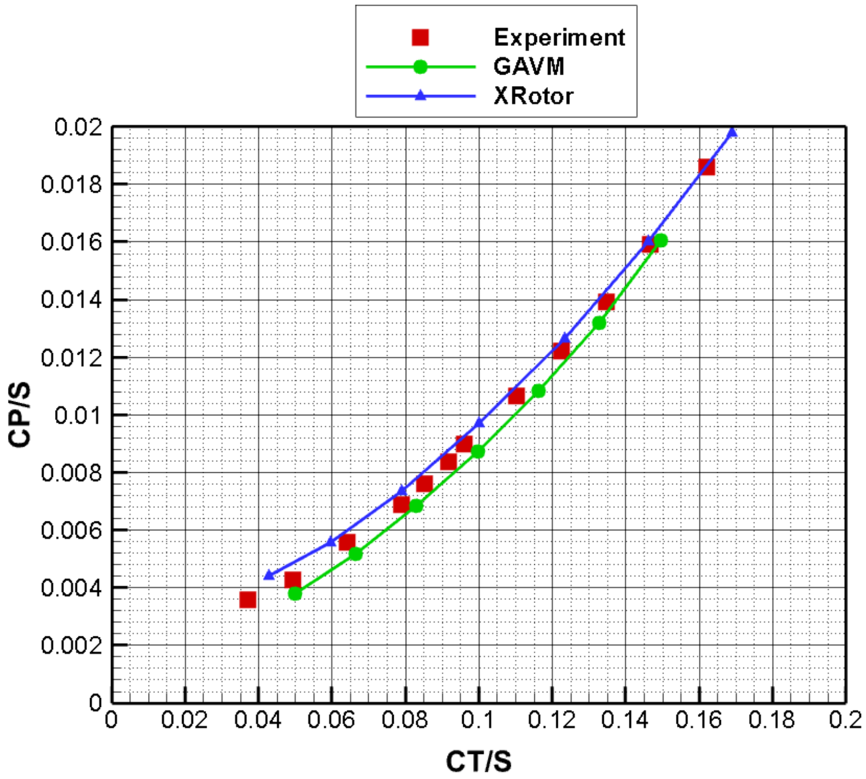


Figure 4-11 XV-15 Rotor Blade Isolated Hover Tests compared with GAVM Isolated Rotor Analysis

4.2 Wing Model

There are two different types of mathematical models of Wing component in GAVM. First one is table look-up methodology. In this methodology the accuracy and the precision of the mathematical model is highly depending on the imported coefficient tables generated by CFD tools. Therefore, no study is done for validation of table-lookup methodology. Second method is based on the modified second order Lifting Line Theory of Weissinger. Post stall region for a wing is modeled by the α-correction methods. This second methodology needs validation. Therefore, an experimental case is employed to validate the Wing model. [82]

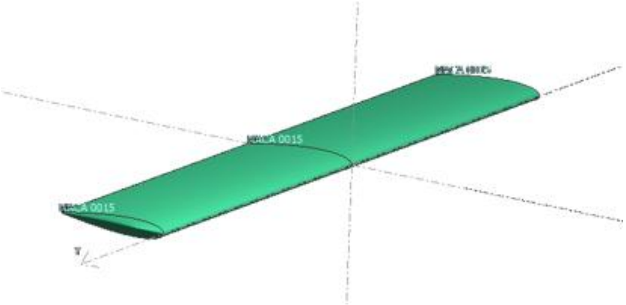


Figure 4-12 Rectangular NACA 0015 Wing

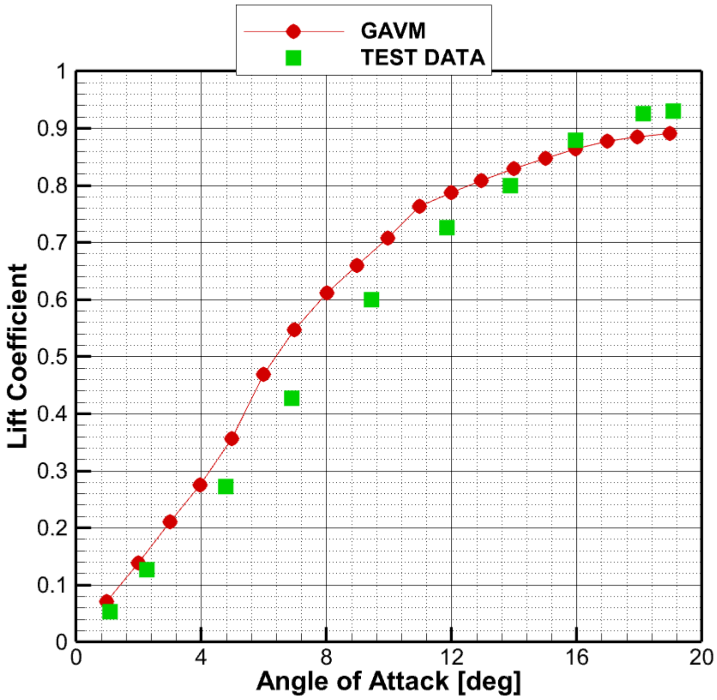


Figure 4-13 Wing model comparison with experimental data

4.3 Propeller Model

A test campaign conducted by National Advisory Committee for Aeronautics (NACA) [83] is taken as an experimental case for isolated propeller model validation. In this GAVM code propeller can be modeled in two ways. First way is to use the QPROP based model and using the propeller object type. Second way of modeling the propeller is using the rotor class and defining the RPM as input and deactivating the lag and flap dynamics.

In the referenced work above the tests of several blades are conducted whose tip speeds are changing from 150 to 300 m/sec and tip radii are 1.45 meter. Two blades are used as a validation case which is having Clark-Y airfoils and Raf-6 airfoils.

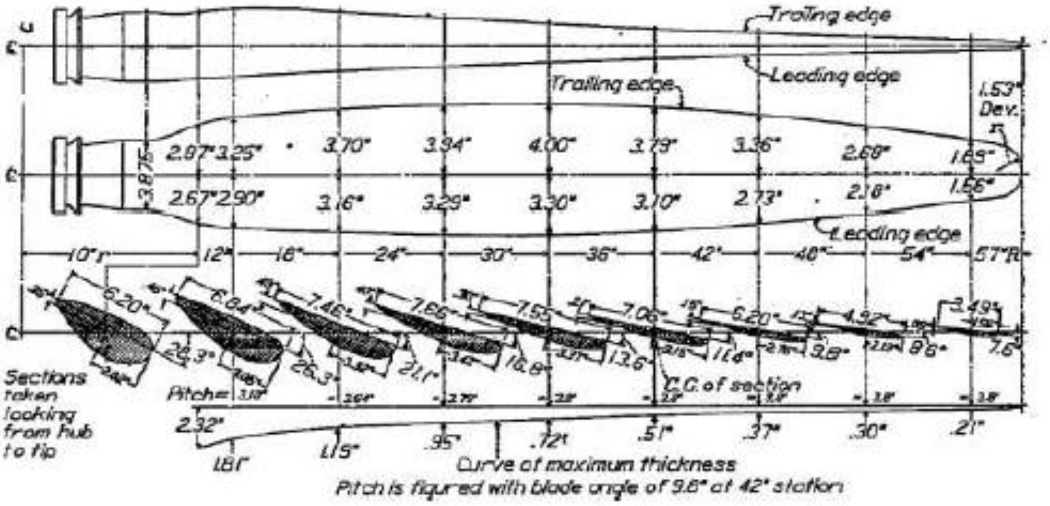
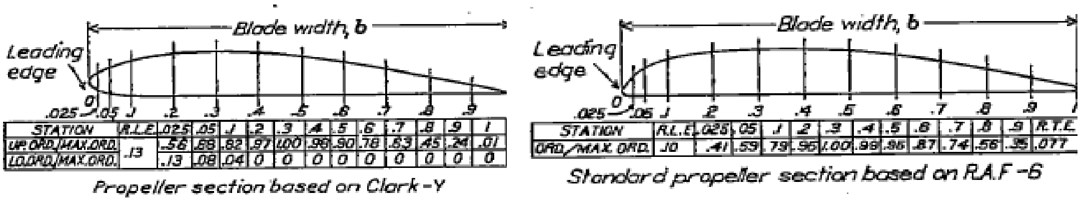


Figure 4-14 Propeller geometry having Clark-Y and Raf-6 airfoils

Figure 4-14 depicts the twist, and chord distribution of experimental propellers. Same twist and chord ratios are used for both airfoils and thrust and required power values are obtained. Airfoil geometries are given in Figure 4-15.



STATION	R.L.E.	.025	.05	.1	.2	.3	.4	.5	.6	.7	.8	.9	1
UP. ORD./MAX. ORD.	.13	.56	.68	.82	.97	1.00	.98	.90	.78	.63	.45	.24	.01
LO. ORD./MAX. ORD.		.13	.08	.04	0	0	0	0	0	0	0	0	0

Propeller section based on Clark-Y

STATION	R.L.E.	.025	.05	.1	.2	.3	.4	.5	.6	.7	.8	.9	R.T.E.
ORD./MAX. ORD.	.10	.41	.59	.79	.96	1.00	.99	.96	.87	.74	.56	.35	.077

Standard propeller section based on R.A.F-6

Figure 4-15 Propeller section profile geometries of RAF-6 and Clark-Y

Propellers are modeled in GAVM and airfoil lift and drag polars are generated by using viscous CFD solvers.

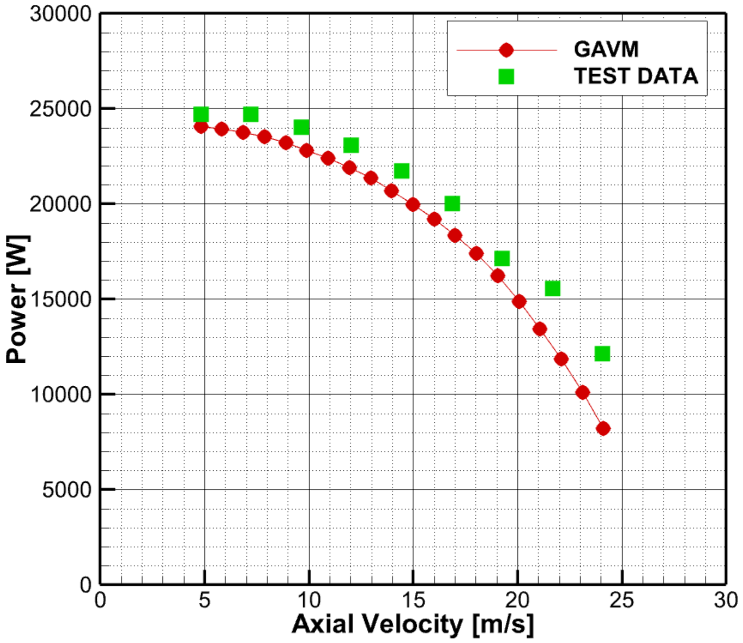


Figure 4-16 CLARK-Y Propeller comparison of Power vs. Airspeed

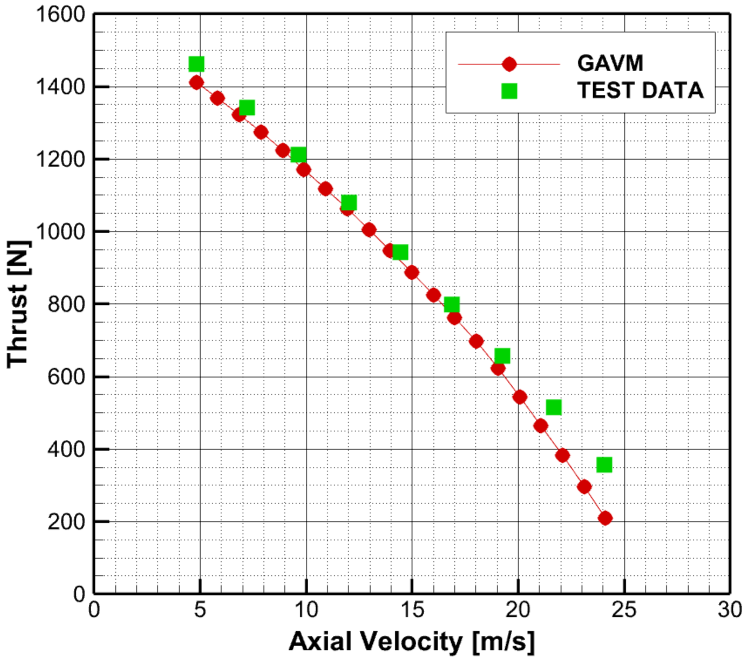


Figure 4-17 CLARK-Y Propeller comparison of Thrust vs. Airspeed

Figure 4-16 and Figure 4-17 illustrates the comparison of propeller model with the experimental test cases. GAVM propeller model includes the modified mathematical model of QROP code which includes the viscous airfoil tables. Analysis results show adequate compromise with the experimental data in estimating the Thrust force especially. Analysis results slightly under-estimates the required power for especially low axial airspeed, however; required power results matches the experimental data with the increasing forward speed.

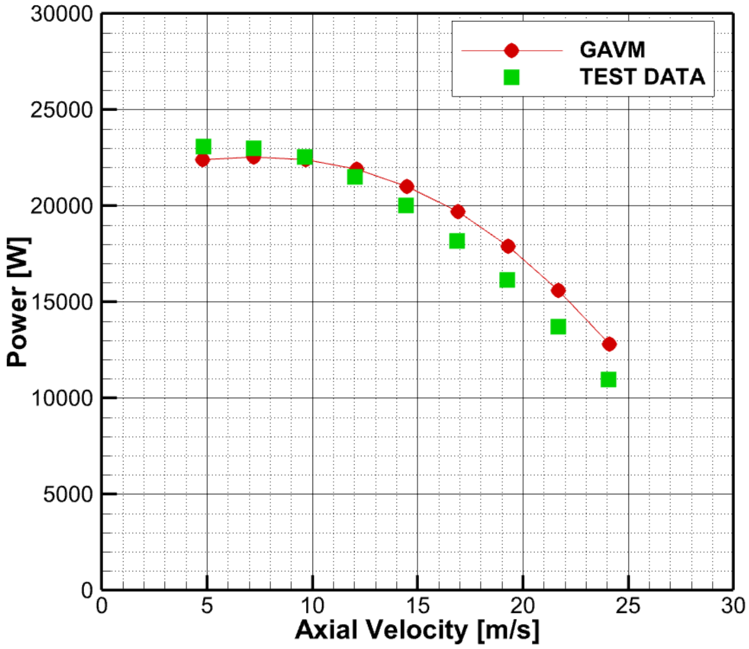


Figure 4-18 R.A.F-6 Propeller comparison of Power vs. Airspeed

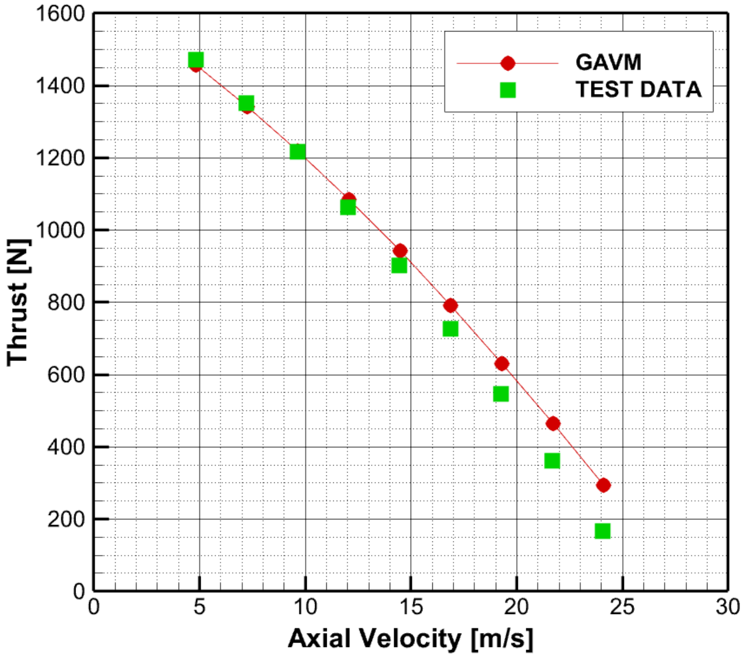


Figure 4-19 R.A.F-6 Propeller comparison of Thrust vs. Airspeed

Figure 4-18 and Figure 4-19 belongs to the experimental results of the propeller which has the R.A.F-6 airfoils as blade profiles. Comparison of experimental results with the mathematical model of propeller shows that the GAVM propeller model shows consistency with the experimental test cases.

As a result, mathematical model inside the **Propeller** class is compared and validated with the experimental data obtained from NACA propeller test campaign. Analysis results adequately fit the experimental results; therefore, methodology implemented in GAVM Propeller object is validated to be used as an airplane propeller in case of only axial freestream exists since the implemented propeller theory does not cover asymmetric loading on the propeller disk.

CHAPTER 5

COMPARISON AND VALIDATION

In this part, different comparison and validation works are conducted to illustrate the generic rotorcraft model works correctly. For a conventional helicopter validation case the flight test campaigns conducted for LYNX XZ170 helicopter is taken as reference [84]. Total required power, body attitudes and pilot inputs are compared for a given flight condition where these data is obtained. Moreover, XV-15 tilt-rotor aircraft is modeled to use in validation both hovering helicopter mode and airplane mode [85].

As stated in the first chapter as limitations of the study, the correctness and the fidelity of aerodynamic and geometric data provided into GAVM determines the accuracy of the results. In these validation cases, firstly wind tunnel test results are used for airfoils, fuselage and wing surfaces. If these parameters are not obtained or do not cover the required envelope, CFD and similar tools are used to obtain useful data and some extrapolation algorithms are used. Assumptions are stated for each case and results are displayed.

5.1 Conventional Helicopter

In order to validate a full rotorcraft model generated in this code, a flight test campaign conducted at Ames Research Center for Lynx XZ170 helicopter, which is shown in Figure 5-1, is used [84]. This flight tests are carried out for the British Experimental Rotor Programme (BERP). Performance and load data is acquired during the tests and flights beyond the stall boundaries are conducted.

Lynx XZ170 helicopter is a single main rotor and a single tail rotor helicopter having rectangular blades. Main rotor has hingeless hub and the helicopter has an empty weight of approximately 2570 kg with a capability of MTOW as 4535 kg.

Figure 5-1 Westland Lynx AH9 · Royal Navy · RNAS Yeovilton (YEO / EGDY) · Stephen John Rendle - 07/07/2007

LYNX XZ170 has two 800-hp engines and has a four-bladed main rotor and a four bladed tail rotor as shown in Figure 5-2. Main rotor shaft tilt is 4° forward and tail rotor has no cant or shaft tilt. Horizontal tail is located at the starboard of the helicopter. Further details are summarized in Figure 5-3.

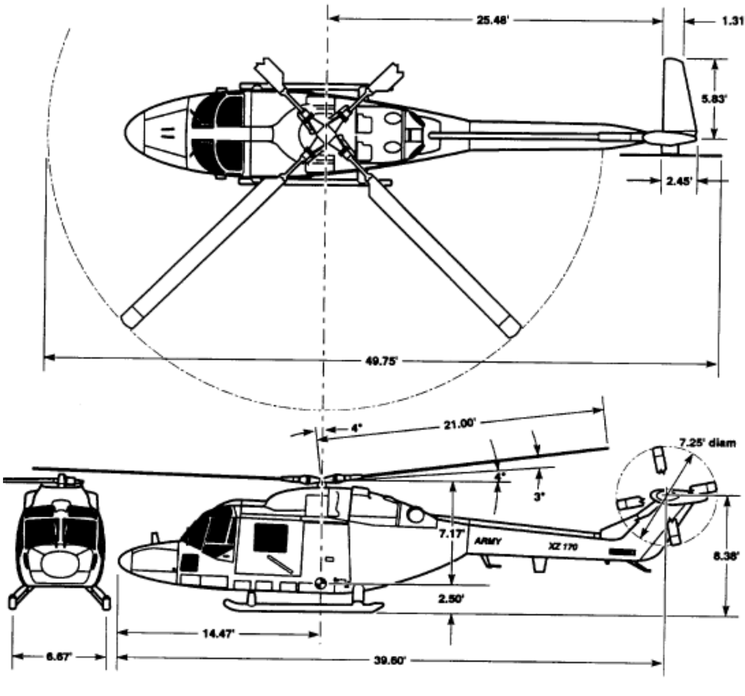


Figure 5-2 3-D view drawing of LYNX-170 Helicopter

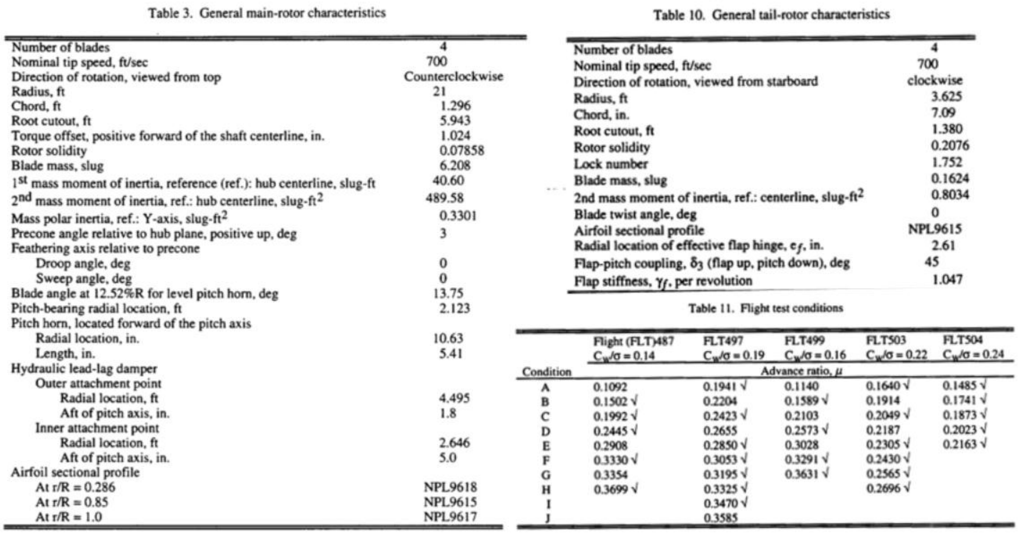


Table 3. General main-rotor characteristics

Number of blades	4
Nominal tip speed, ft/sec	700
Direction of rotation, viewed from top	Counterclockwise
Radius, ft	21
Chord, ft	1.296
Root cutout, ft	5.943
Torque offset, positive forward of the shaft centerline, in.	1.024
Rotor solidity	0.07858
Blade mass, slug	6.208
1st mass moment of inertia, reference (ref.): hub centerline, slug-ft	40.60
2nd mass moment of inertia, ref.: hub centerline, slug-ft^2	489.58
Mass polar inertia, ref.: Y-axis, slug-ft^2	0.3301
Precone angle relative to hub plane, positive up, deg	3
Feathering axis relative to precone	
Droop angle, deg	0
Sweep angle, deg	0
Blade angle at 12.52%R for level pitch horn, deg	13.75
Pitch-bearing radial location, ft	2.123
Pitch horn, located forward of the pitch axis	
Radial location, in.	10.63
Length, in.	5.41
Hydraulic lead-lag damper	
Outer attachment point	
Radial location, ft	4.495
Aft of pitch axis, in.	1.8
Inner attachment point	
Radial location, ft	2.646
Aft of pitch axis, in.	5.0
Airfoil sectional profile	
At r/R = 0.286	NPL9618
At r/R = 0.85	NPL9615
At r/R = 1.0	NPL9617

Table 10. General tail-rotor characteristics

Number of blades	4
Nominal tip speed, ft/sec	700
Direction of rotation, viewed from starboard	clockwise
Radius, ft	3.625
Chord, in.	7.09
Root cutout, ft	1.380
Rotor solidity	0.2076
Lock number	1.752
Blade mass, slug	0.1624
2nd mass moment of inertia, ref.: centerline, slug-ft^2	0.8034
Blade twist angle, deg	0
Airfoil sectional profile	NPL9615
Radial location of effective flap hinge, e_f, in.	2.61
Flap-pitch coupling, δ_3 (flap up, pitch down), deg	45
Flap stiffness, γ_f, per revolution	1.047

Table 11. Flight test conditions

	Flight (FLT)487 $C_W/\sigma = 0.14$	FLT497 $C_W/\sigma = 0.19$	FLT499 $C_W/\sigma = 0.16$	FLT503 $C_W/\sigma = 0.22$	FLT504 $C_W/\sigma = 0.24$
Condition	Advance ratio, μ				
A	0.1092	0.1941 √	0.1140	0.1640 √	0.1485 √
B	0.1502 √	0.2204	0.1589 √	0.1914	0.1741 √
C	0.1992 √	0.2423 √	0.2103	0.2049 √	0.1873 √
D	0.2445 √	0.2655	0.2573 √	0.2187	0.2023 √
E	0.2908	0.2850 √	0.3028	0.2305 √	0.2163 √
F	0.3330 √	0.3053 √	0.3291 √	0.2430 √	
G	0.3354	0.3195 √	0.3631 √	0.2565 √	
H	0.3699 √	0.3325 √		0.2696 √	
I		0.3470 √			
J		0.3585			

Figure 5-3 General specifications of LYNX XZ170 and Flight test conditions

Mathematical model of LYNX XZ170 helicopter generated in GAVM and flight test conditions are analyzed. Following assumptions and losses are employed;

- Pitt-Peters dynamic inflow model is selected with coupled flapping and lagging dynamics.
- Excrescence drag is assumed to be 30% greater than clean fuselage drag obtained from wind tunnel test and appended into the fuselage database. Therefore, total flat plate drag area becomes 14.15 ft^2.
- Inflow correction factor κ is assumed as 1.16 for both rotors to account for the 3-D effects and modeling discrepancies.
- Transmission losses are assumed as 6% and accessory losses are assumed as 35hp.
- Blade airfoil wind tunnel aerodynamic coefficients are extrapolated between -180° and +180° angle of attack by using Viterna and Montgomerie extrapolation methods. [86], [87].

After imposing all the assumptions, Figure 5-4, Figure 5-5, Figure 5-6, Figure 5-7, Figure 5-8, and Figure 5-9 depict the comparisons of the analytical model with the experimental data for required power, pilot inputs and body attitudes.

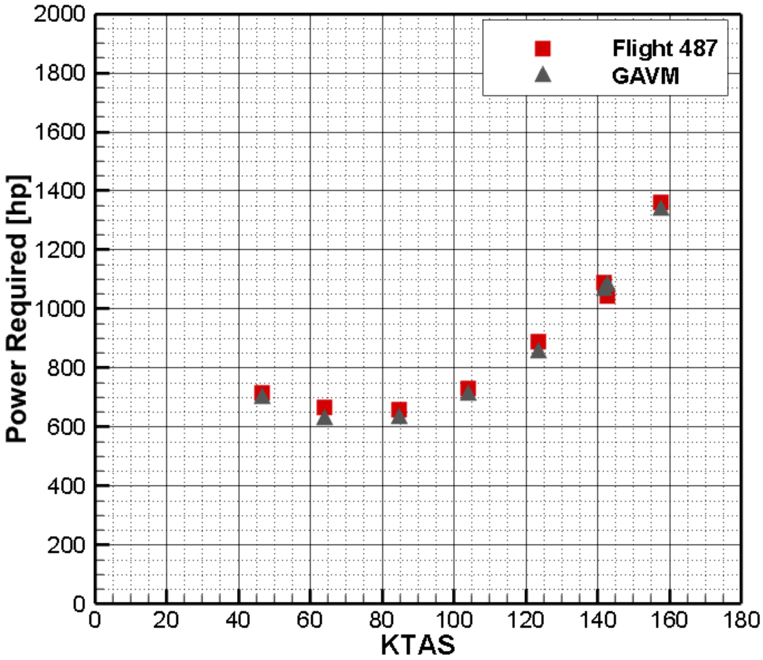


Figure 5-4 LYNX XZ170 Flight 487 Required power vs. True airspeed comparison with GAVM

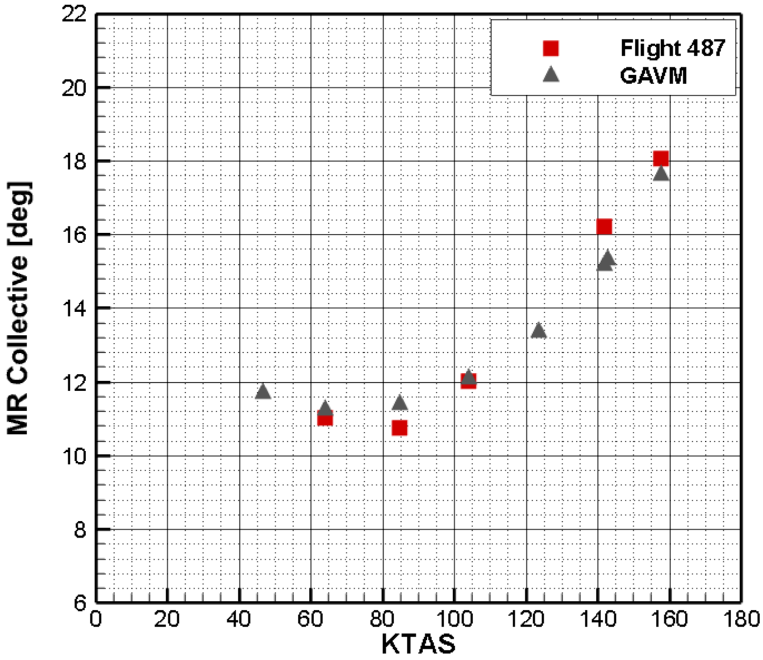


Figure 5-5 LYNX XZ170 Flight 487 Main Rotor Collective vs. True airspeed comparison with GAVM

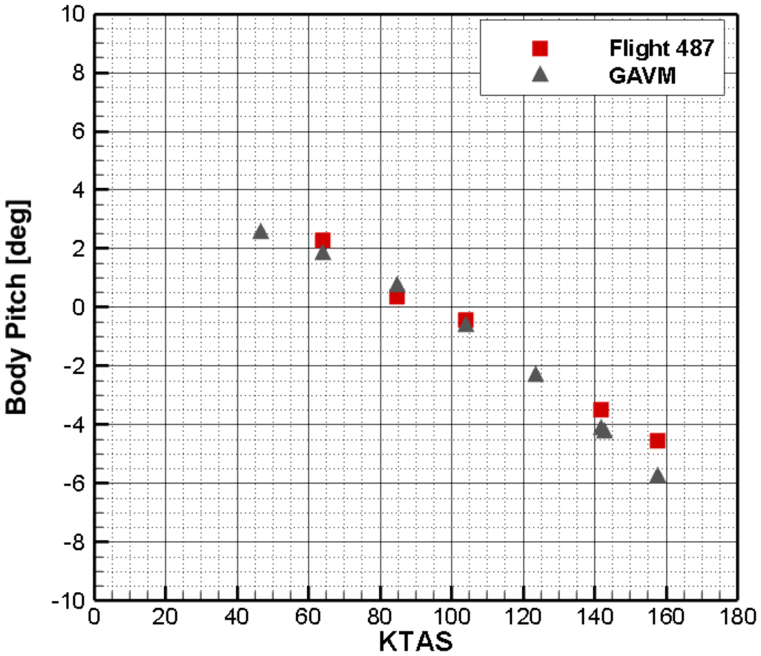


Figure 5-6 LYNX XZ170 Flight 487 Body Pitch Angle vs. True Airspeed comparison with GAVM

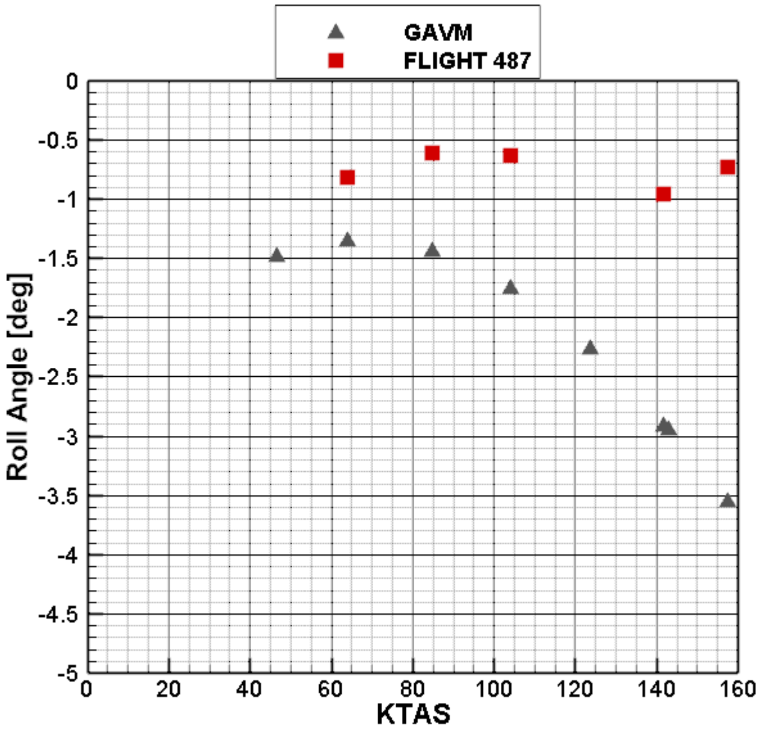


Figure 5-7 LYNX XZ170 Flight 487 Body Roll Angle vs. True Airspeed comparison with GAVM

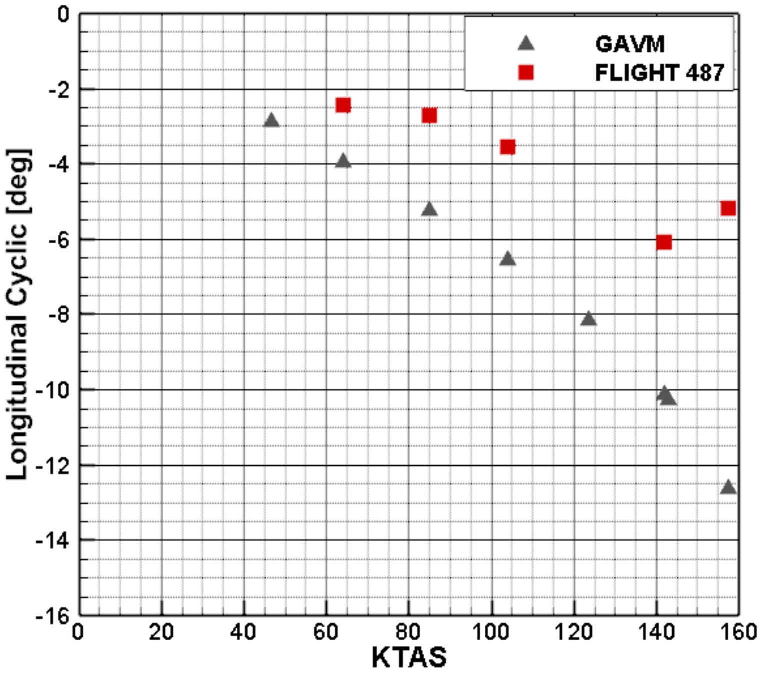


Figure 5-8 LYNX XZ170 Flight 487 Longitudinal Cyclic Input vs. True Airspeed comparison with GAVM

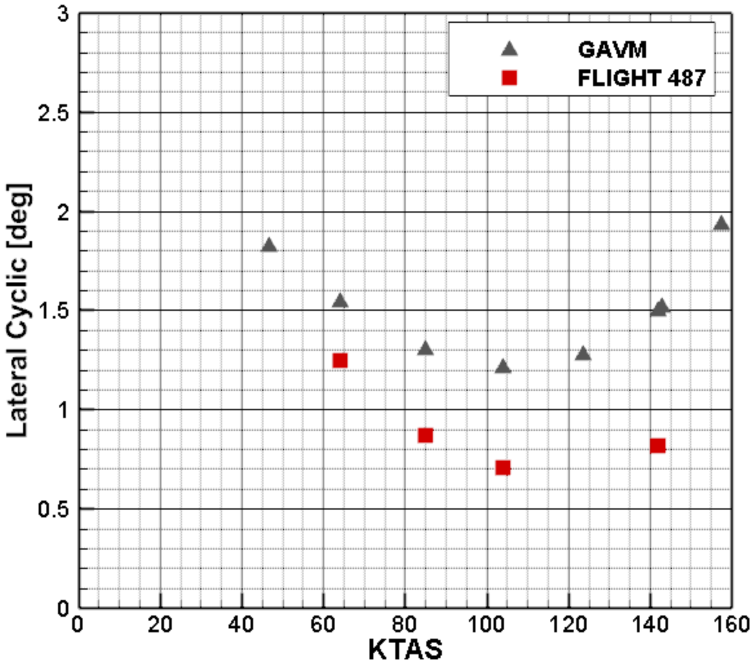


Figure 5-9 LYNX XZ170 Flight 487 Lateral Cyclic Input vs. True Airspeed comparison with GAVM

Under the assumptions listed above, comparison with flight test data shows that the maximum difference in required power is approximately 40 hp which corresponds to 1% to 3% of percentage error. Noting that the interactional effects are not modeled and other uncertainties exists in the flight test, the difference can be evaluated as acceptable and general trend of power and pitch attitude shows consistency with the flight tests. Roll angle comparisons shows a little bit difference but similar behavior as forward speed is increased. That may be due to the type of the forward flight, trim condition implies zero sideslip flight; however, pilot may fly with a small sideslip which reduces the roll angle. Moreover, differences in rotor dynamics and inertial properties, elastic motion may generate the difference. Longitudinal and lateral cyclic inputs shows similar behavior as the velocity increased. The difference in these pilot inputs may be explained in similar manner as in the difference in roll attitude values.

5.2 Tilt Rotor

Since GAVM is capable of analyzing tilting rotors a proper validation case is implemented. Validation case is obtained from the experimental tilting rotor aircraft XV-15. XV-15 is a Vertical Take-off and Landing (VTOL) research aircraft program from the earliest efforts by the U.S. army designing new concepts for supporting field operations [85].

The requirement for a design of VTOL aircraft comes up from the urgent field operations where there is no runway to land and take-off. Tilting rotor aircrafts has lower hovering efficiency than helicopters as in Figure 5-10; however, their maximum velocities are much higher than helicopters which ensure much faster operations on field.

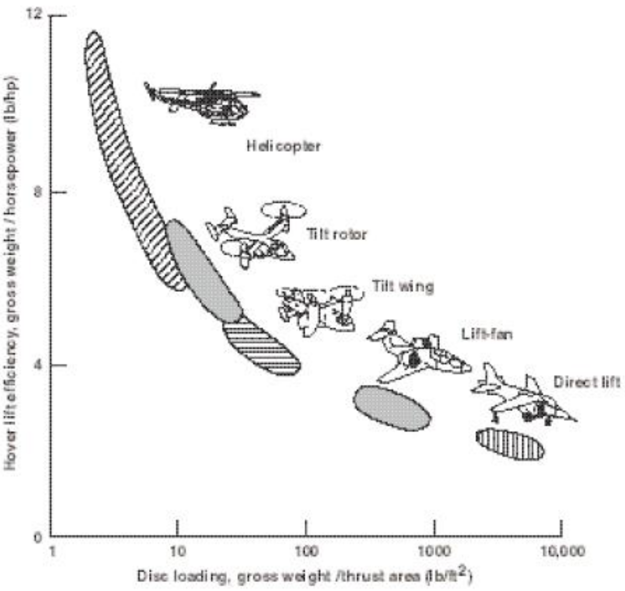


Figure 5-10 Hover vertical lift efficiency as a function of disc loading [85]

Rotor design of XV-15 is different from conventional helicopters and conventional propeller aircrafts. There were several analyses and test campaigns to assess the performance of XV-15 prop-rotor. In this chapter GAVM model of XV-15 rotorcraft is generated by using the available data and trim conditions are compared with the available flight test data [88] [89].

5.2.1 Analysis

XV-15 rotorcraft is a tilt rotor aircraft having VTOL capability. XV-15 has wing tip mounted, interconnected 3-bladed two prop-rotors, forward swept wing, conventional horizontal tail and two end-plate type vertical fins. Aircraft has collective and cyclic controls as pilot controls in hover and flaps, flaperons, elevator and rudders as pilot controls for forward flight. XV-15 has design gross weight of 6000kg.

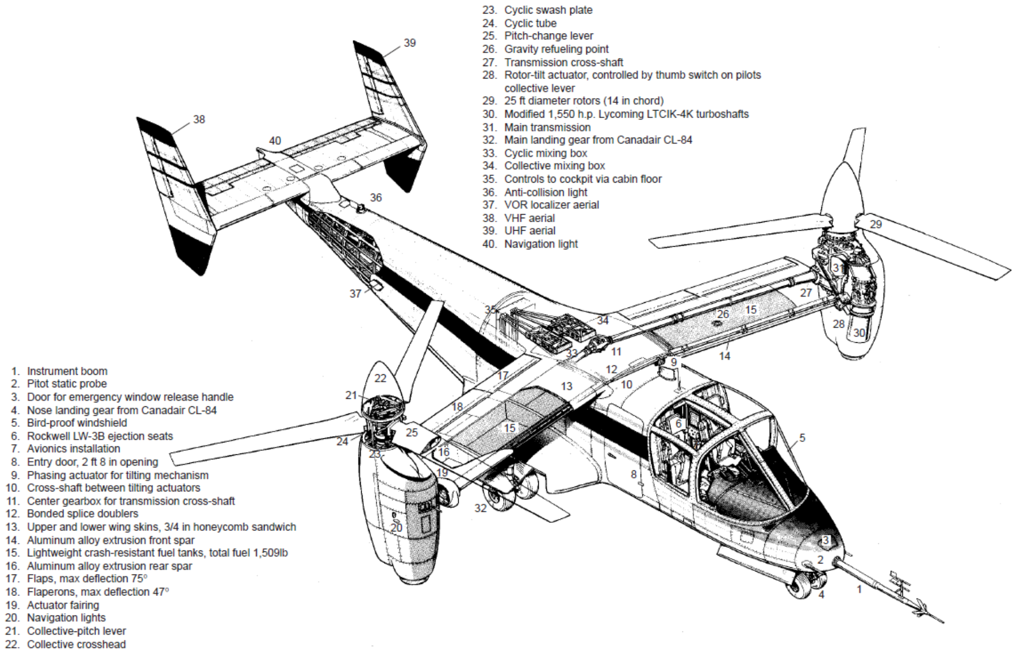

1. Instrument boom
2. Pitot static probe
3. Door for emergency window release handle
4. Nose landing gear from Canadair CL-84
5. Bird-proof windshield
6. Rockwell LW-3B ejection seats
7. Avionics installation
8. Entry door, 2 ft 8 in opening
9. Phasing actuator for tilting mechanism
10. Cross-shaft between tilting actuators
11. Center gearbox for transmission cross-shaft
12. Bonded splice doublers
13. Upper and lower wing skins, 3/4 in honeycomb sandwich
14. Aluminum alloy extrusion front spar
15. Lightweight crash-resistant fuel tanks, total fuel 1,509lb
16. Aluminum alloy extrusion rear spar
17. Flaps, max deflection 75°
18. Flaperons, max deflection 47°
19. Actuator fairing
20. Navigation lights
21. Collective-pitch lever
22. Collective crosshead
23. Cyclic swash plate
24. Cyclic tube
25. Pitch-change lever
26. Gravity refueling point
27. Transmission cross-shaft
28. Rotor-tilt actuator, controlled by thumb switch on pilots collective lever
29. 25 ft diameter rotors (14 in chord)
30. Modified 1,550 h.p. Lycoming LTCIK-4K turboshafts
31. Main transmission
32. Main landing gear from Canadair CL-84
33. Cyclic mixing box
34. Collective mixing box
35. Controls to cockpit via cabin floor
36. Anti-collision light
37. VOR localizer aerial
38. VHF aerial
39. UHF aerial
40. Navigation light

Figure 5-11 General layout and list of components of XV-15 rotorcraft [85]

Helicopter controls in hover mode is similar to the control of a tandem helicopter through lateral axis. Differential collective results in rolling, collective longitudinal cyclic for two rotors leads to pitching motion and differential cyclic results yaw motions. Fixed wing aircraft controls remain active all the times. During the conversion mode from helicopter to airplane, helicopter controls are phased out and airplane controls provide all control forces and moments. When the nacelles reach airplane configuration, collective stick is transferred into the automatic rpm governor.

Table 5-1 XV-15 geometric specifications

Weight		**Horizontal Tail**	
Design	5896 [kg]	Span	3.91 [m]
Empty	4574 [kg]	Area	4.67 [m^2]
Gross	6010 [kg]	Chord	1.19 [m]
Wing		Airfoil	NACA 64A015
Span	9.8 [m]	Aspect Ratio	3.27
Chord	1.6 [m]	**Vertical Tail**	
Area	15.7 [m^2]	Area	4.69 [m^2]
Airfoil	NACA 64A223	Airfoil	NACA 0009
Aspect Ratio	6.12	MAC	1.13 [m]
Sweep	-6.5 [deg]	Aspect Ratio	2.33
Dihedral	2 [deg]	**Proprotor**	
Length	12.827 [m]	No of rotors	2 (interconnected)
Width	17.424 [m]	RPM (helicopter)	589
Height	3.861 [m]	RPM (airplane)	517

XV-15 has three bladed, 25 feet dia, stiff-inplane and gimbal mounted rotors [90].

XV-15 tilting rotor aircraft has 9.8-meter wing with a forward sweep which produces lift for the high forward velocities. Wing has flaps and flaperons and in hover mode flaps are deflected to reduce the rotor wing download force due to wake-wing interactions. Flaps are deflected not only to reduce the exposed area for the interactions but also reduce the effect of separation [46]. Therefore, flaps are deflected for hover and for transition flight as well. H-type tail section is designed to provide stability and control for the transition and forward flight regimes as a conventional airplane.

In NASA Ames wind tunnel facilities, there are numerous available test data either full-scale or scaled version of XV-15 aircraft [89]. Therefore, for the wing, control surfaces and fuselage aerodynamic specifications wind tunnel test results are used. There are tests for the aerodynamic coefficients of XV-15 for different angle of attack, angle of sideslip, nacelle tilt, and flap deflection. Therefore, these tests provide valuable data for the GAVM to model the tilting rotor wing, fuselage and empennage aerodynamics.

Rotor dynamics and inertia properties are obtained from available literature data belonging to a developed tilt-rotor simulation model and CAMRAD model developed for aeroelastic stability analyses [88][80].

5.2.2 Validation

As given in the previous chapter XV-15 main rotor was modeled by using the **Rotor** class and validated with the isolated hover test data. In this part, XV-15 prop-rotors are modeled by using the **Propeller** object and trim conditions are obtained for conventional airplane mode. Experimental results are belonging to the test campaign which is explained in the reference [91].

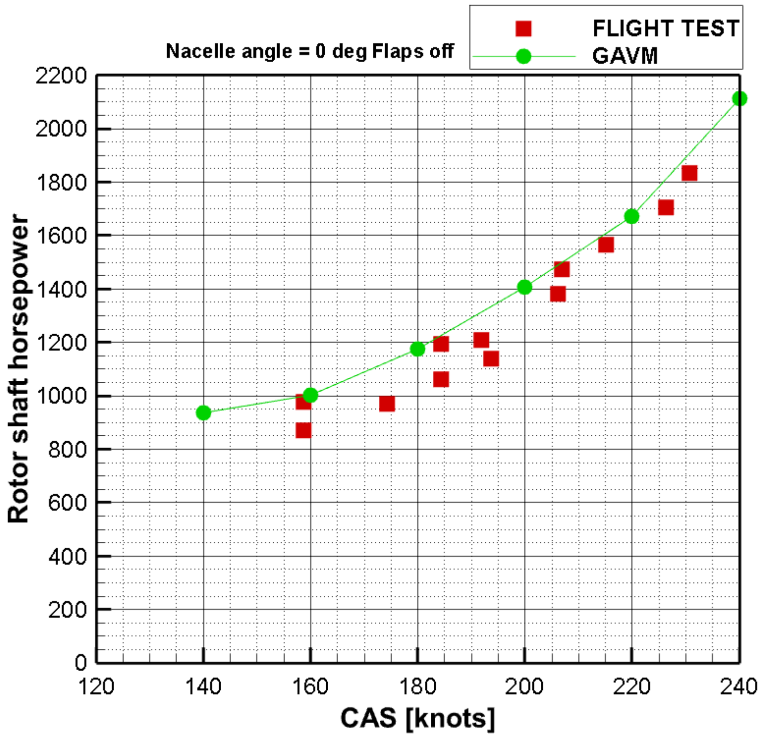


Figure 5-12 XV-15 comparison of shaft power with airspeed in Airplane mode

In Figure 5-12 airplane mode analysis results show consistency with the flight test data with small over-estimation. The over estimation may be due to the provided aerodynamic data for airfoils and assumed excrescence drag which is about 30%.

In Figure 5-13 comparison of GAVM model shaft power with the flight test data shows good consistency with 30° of nacelle tilt and 20° of flap.

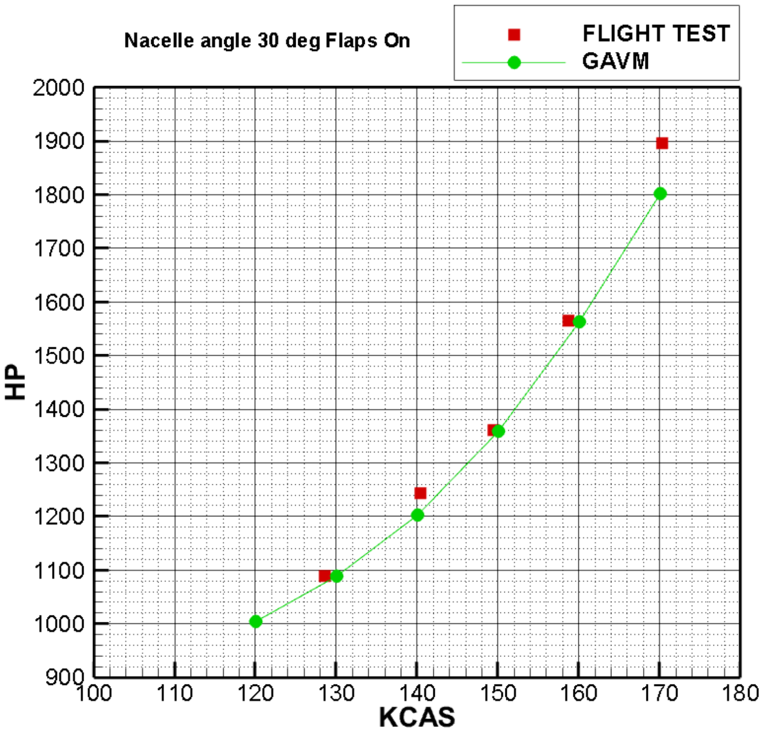


Figure 5-13 XV-15 comparison of shaft power with airspeed in Transition mode

Hover required power validation case is analyzed for sea level standard ISA conditions for MTOW by modeling the full rotorcraft by using **Rotor** object and results are given in Table 5-2. Inflow correction number is assumed as 1.4 and download drag is assumed as 7% of the total required power. Under the current assumptions hover required power is calculated with only 3 hp difference.

Table 5-2 XV-15 Hover required power comparison with available test data

	Flight Test Data	GAVM
Required Power [hp]	1824.8	1821.2

As a result, within the available data and under the given assumptions, GAVM provides promising results when compared with an experimental case for a tilt-rotor aircraft configuration. Due to the generic input definition and ControlMixer object, it is observed that a given trim condition can be reached by using different set of input configurations during the model generation phase of the work. As stated previously, these results are obtained by using the limited set of aerodynamic data and better validations are possible with the higher fidelity of aerodynamic data support.

CHAPTER 6

SIMULATION AND CONTROL ANALYSES

For simulation and control analysis GAVM shared library version is utilized. By calling the DLL export functions model can be generated, trim condition can be obtained and simulation is initiated. During the time simulation, inputs and states can be modified by the caller environment. Any type of environment which supports importing C++ shared libraries may be utilized for simulation and control analysis. One of the most common tools used in control design and analysis is MATLAB© Simulink. Air vehicle model is compiled by calling the library before initiating the simulation model; moreover, trim condition can be obtained by executing the exported functions. Required export functions are defined as initialization callback functions in Simulink model properties interface. Time integration of simulation is done within the GAVM DLL which means Simulink only provides the integration timestep.

6.1 LYNX XZ170 Simulation Example

After the validation work and comparisons, dynamic response of the helicopter is investigated. Shared library version of GAVM is utilized with MATLAB environment to obtain the trim condition simulation and dynamic responses of the helicopter for a step input given to a trimmed flight condition. Simulations for 10 seconds are executed and results are shared in the following pages.

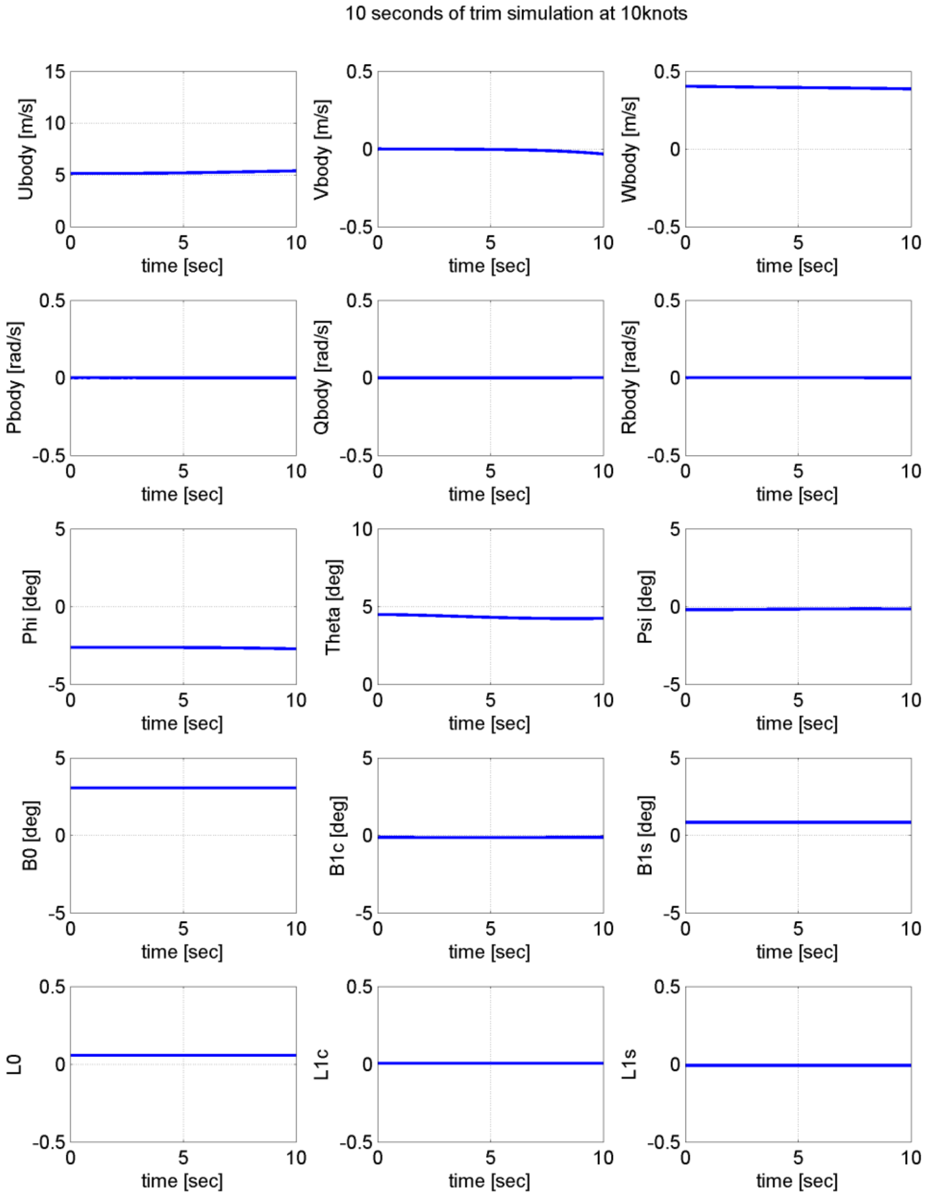


Figure 6-1 Simulation of 10 knots forward flight trim condition

In Figure 6-1 open loop simulation of LYNX XZ170 helicopter is plotted. Trim condition is at SL ISA 4500kg TOW at 10 knots forward speed. As seen in the results, trim condition is almost sustained for 10 seconds of open loop simulation.

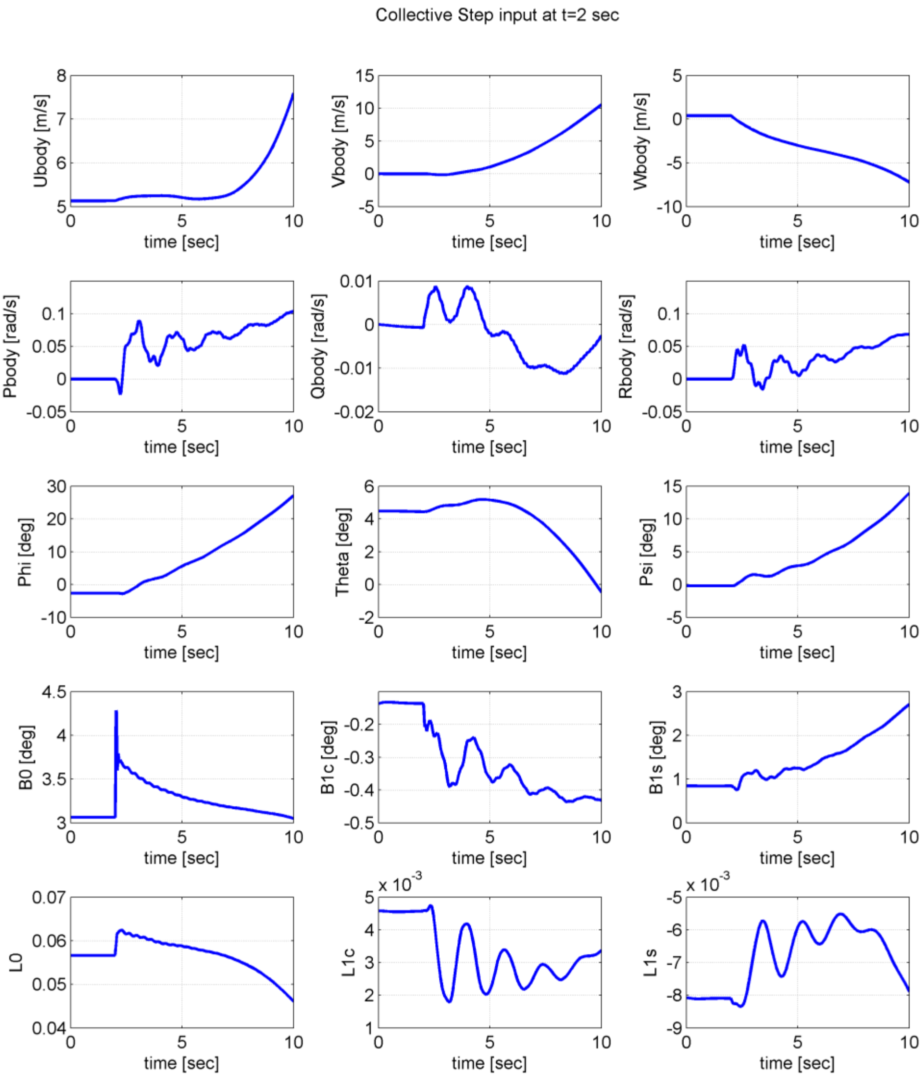


Figure 6-2 Simulation of 10 knots forward flight trim condition for a 2 degrees step input into main rotor collective

In Figure 6-2 step input of 2 degrees of collective is given to the main rotor at t=2 seconds. This input increases the rotor thrust which results in increased induced velocity and flap cone angle. Moreover, as seen from the z-axis body velocity, the helicopter gains altitude and fuselage pitch angle decreases. Moreover, the torque imbalance due to increased collective creates positive yaw rate and positive heading angle since the tail rotor does not produce required anti-torque thrust.

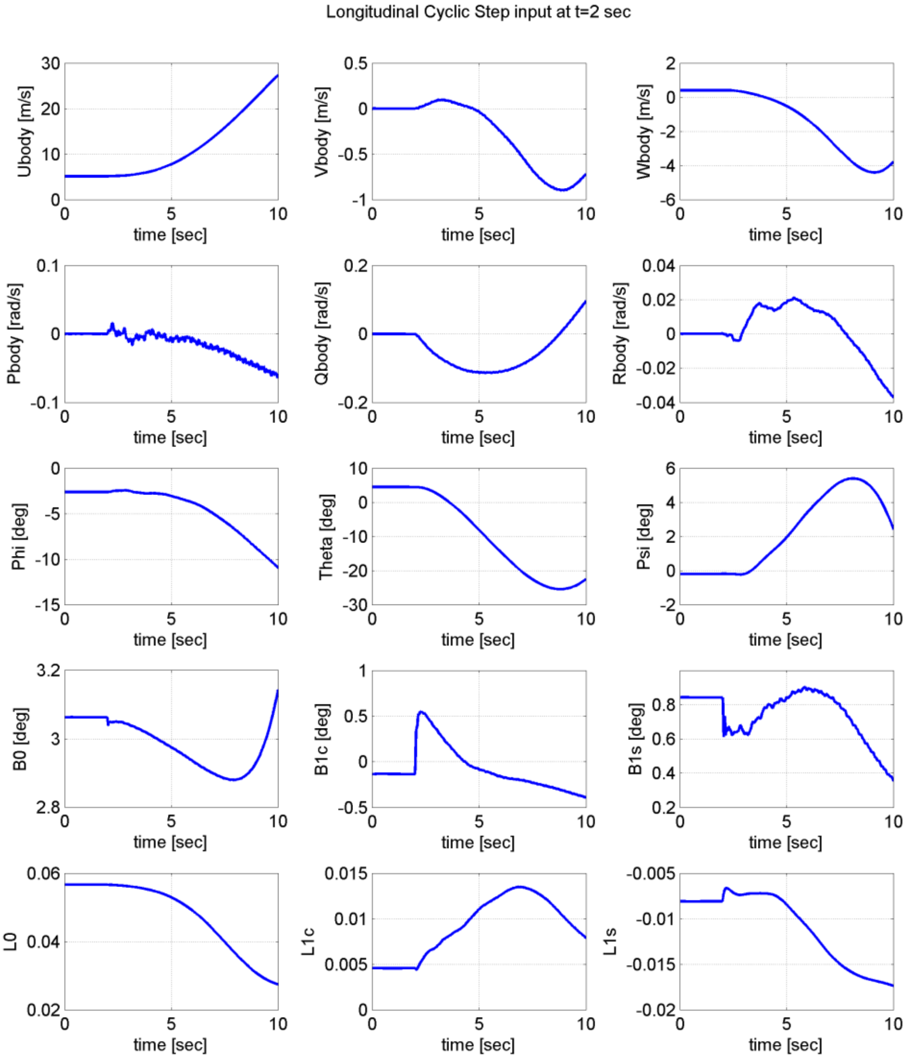


Figure 6-3 Simulation of 10 knots forward flight trim condition for a 1 degrees step input into longitudinal cyclic

In Figure 6-3 the same trim condition is simulated with a 1 degrees of longitudinal cyclic step input which is given in t=2 seconds. Longitudinal cyclic input creates a moment on the rotor disk which tilts the TPP of main rotor. This tilt generates pitch down moment and forward acceleration as expected. At the end of 10 seconds helicopter gains velocity and reaches about 50 knots forward velocity.

6.2 XV-15 Simulation Example

Modeled XV-15 rotorcraft case is adapted into MATLAB Simulink environment to simulate the trim condition and open loop step input responses. Full aircraft having two rotors is trimmed in hover at 6000 kg TOW 100 ft. ISA+0 condition. This simulation case is employed to show the open-loop dynamic response of the tandem tilting rotor rotorcraft. During the simulation nacelle tilt angle is kept constant at 90°.

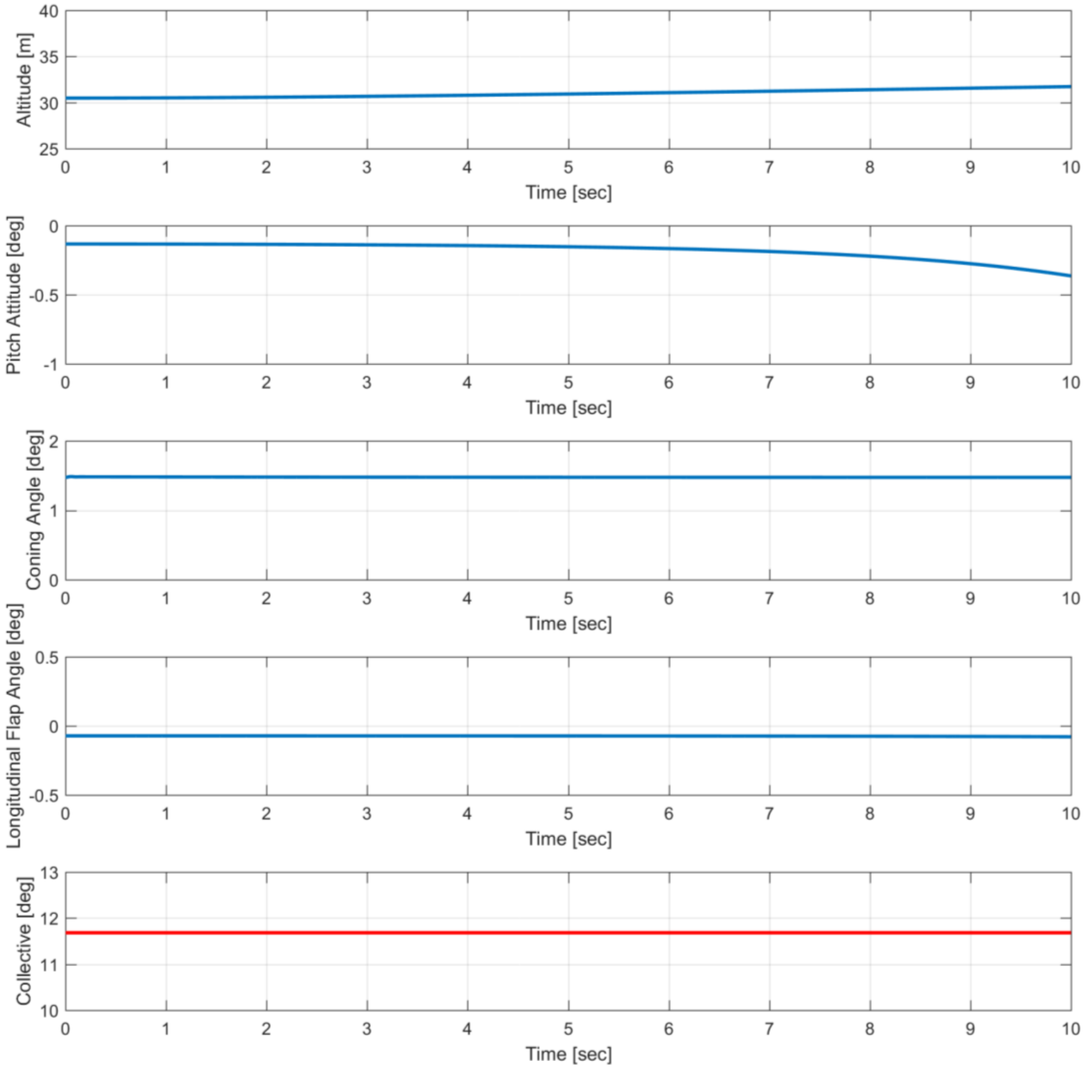


Figure 6-4 Simulation of 10 seconds started from trim condition at hover

In Figure 6-4 open loop simulation is performed for a trimmed helicopter to see whether it can stay at the trim condition or not. Results shows a 11.8 degrees of collective as a trim condition and there is a slight change in the trimmed condition during ten seconds of simulation. Altitude changes about 1 foot and pitch attitude changes about 0.3 degrees.

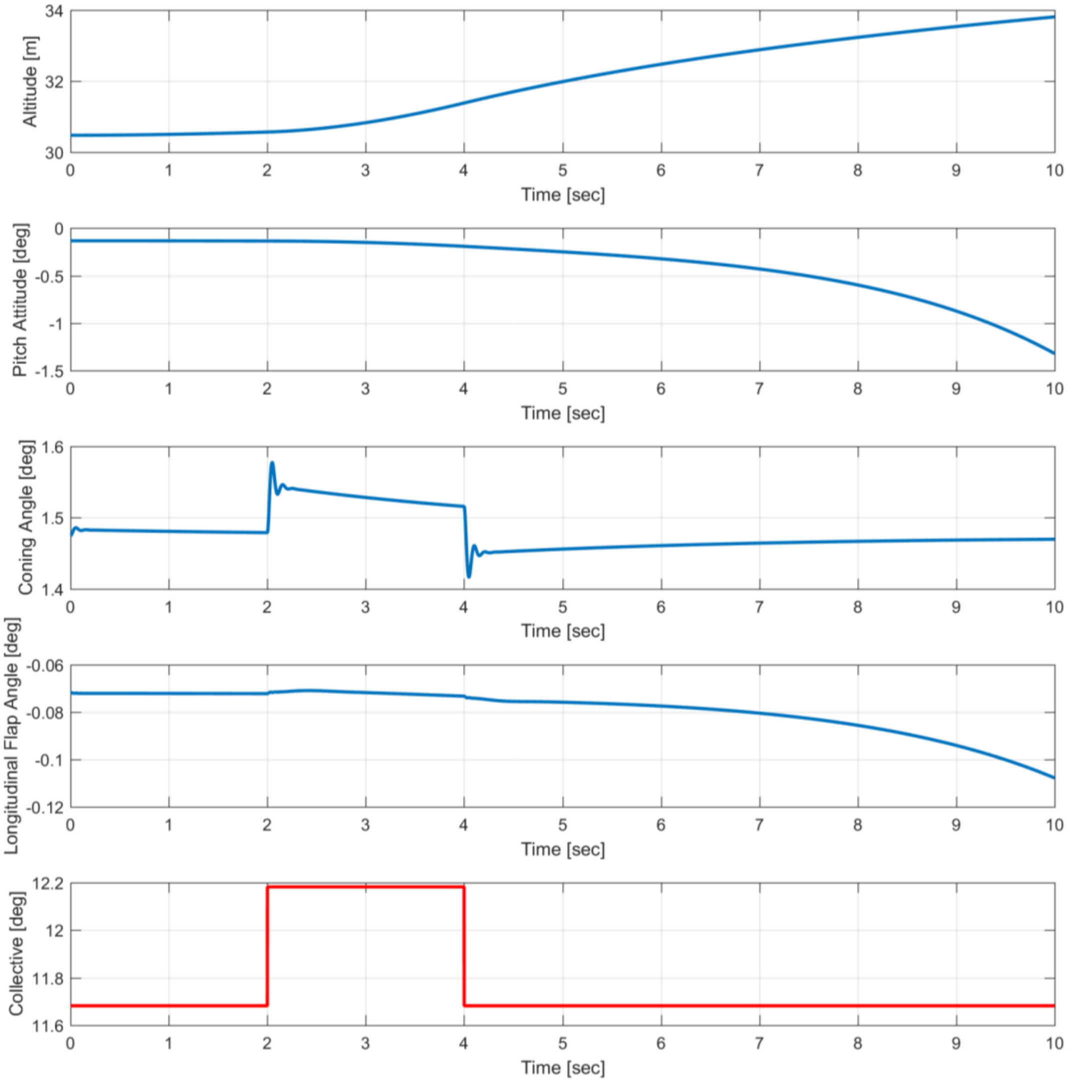


Figure 6-5 Dynamic response of aircraft to a 0.5 degrees Collective pulse input

In Figure 6-5 a pulse input is defined to the collective channel for both rotors starting at t=2 seconds to t=4 seconds. This input creates an increase in the coning angle and altitude as expected. At the end of 10 seconds altitude of the aircraft increases about 4 feet and body pitch angle decreases about 1.3 degrees. Note that in this simulation there is not any controller on any channels.

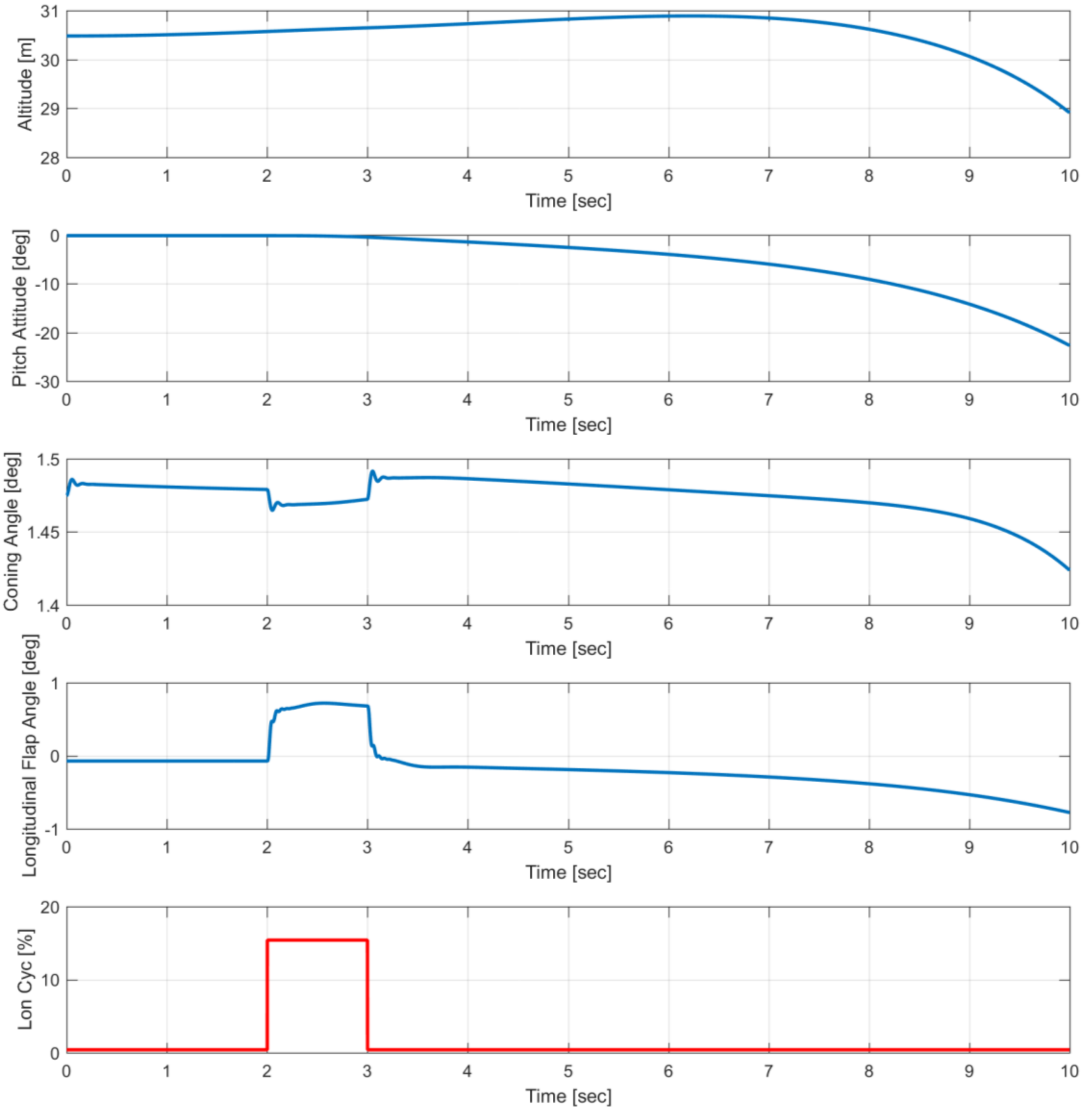


Figure 6-6 Response of aircraft to a 15 percent Longitudinal Cyclic pulse input

Figure 6-6 belongs the same trim condition and open loop responses are plotted for a pulse input of longitudinal cyclic input. Longitudinal flap angle response is similar to the longitudinal cyclic input and pitch angle reaches -23 degrees at the end of 10 seconds of open loop response. Altitude at the end of 10 seconds of simulation decreases about two feet.

6.3 Sample Quad-Rotor Simulation Case

A sample GAVM quadrotor model is implemented into the Simulink as shown in Figure 6-7. By using the state space representation of Quadrotor model a controller is designed and integrated into the Simulink model. Designed Quadrotor is composed of four 11 inches' propellers having Clark-Y airfoils and MTOW as 8 kg;

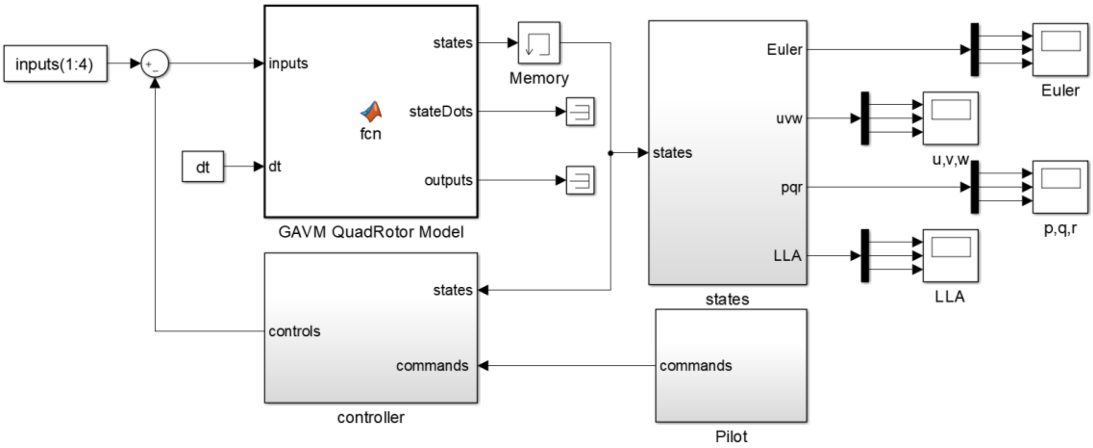


Figure 6-7 Sample Simulink model for a Quadrotor simulation and control analyses

Table 6-1 illustrates the GAVM Quadrotor Model embedded function which includes one step execution of Quadrotor simulation model.

Table 6-1 Sample simulation DLL one step execution algorithm

```
function [states,stateDots,outputs] = fcn(inputs,dt)
%#codegen
coder.extrinsic('calllib')
states=zeros(16,1);
stateDots=zeros(16,1);
outputs=zeros(66,1);

[inputs] = calllib('GAVM_DLL','executeStep',inputs,dt);
[~,states,stateDots,outputs] =
calllib('GAVM_DLL','getSSvars',inputs,states,stateDots,outputs);
```

Separate subsystem block is generated for controller implementation in Simulink which is fed back states and outputs into the non-linear quadrotor model. Controller is designed by Linear Quadratic Regulator (LQR) for attitude Stability Augmentation System (SAS). Body rates $p, q, and\ r$ are defined as errors and proportional and integral feedback loops are designed. Four control inputs are defined which is throttle (heave), longitudinal stick (pitch), lateral stick (roll) and heading stick (yaw). Throttle stick directly changes the collective RPM of each propeller, lateral and longitudinal stick inputs generates differential RPM. Heading stick creates a differential RPM between the two CW and CCW motors by keeping the total thrust constant and changing the total torque as illustrated in Figure 6-8.

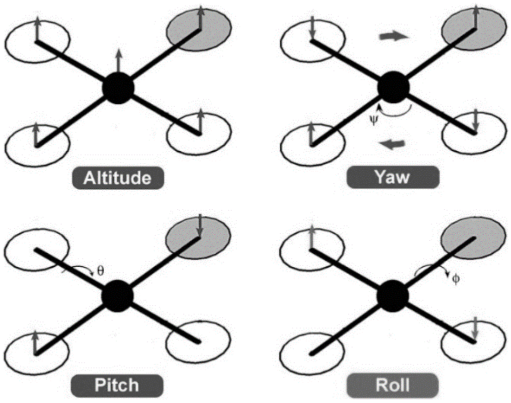


Figure 6-8 Quadrotor controls, (filled disk shows nose)

Throttle, pitch, roll and yaw stick inputs are defined by using the **ControlMixer** object. Throttle is defined by providing the same RPM for all propellers. Pitch and roll inputs provide differential RPM for neighbor propellers. Yaw input changes the heading by supplying higher RPM for two reciprocal propellers. State space and control matrices are obtained for these upper level inputs and a simple controller is designed to stabilize the quadrotor.

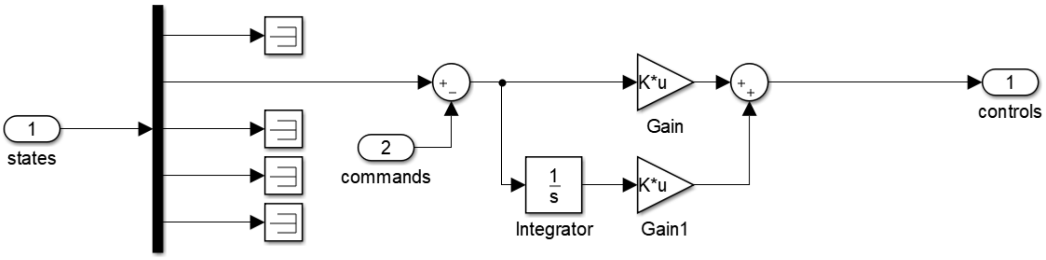


Figure 6-9 Simulink block diagram of the SAS controller

Controller is designed by using the state space matrices which are numerically calculated in GAVM model and Linear Quadratic Regulator (LQR) design is employed directly. Body rates are fed back to the controller as an error and rate reference commands are produced by the pilot. Proportional and integral gains are defined in the controller.

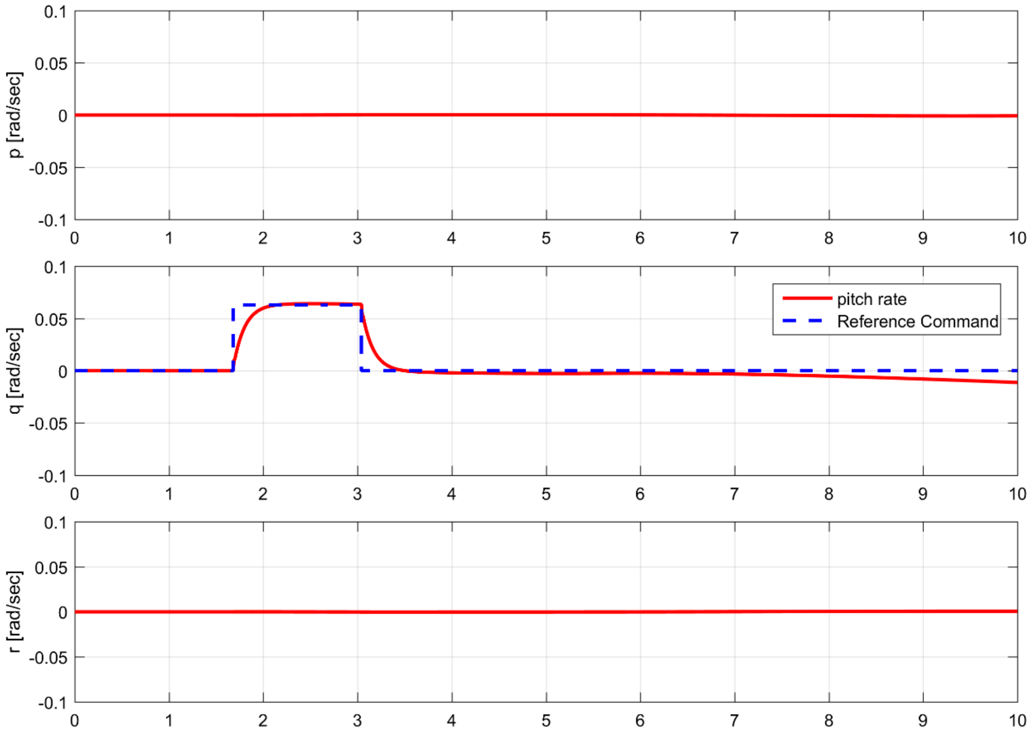


Figure 6-10 Dynamic response of a step input of rate SAS LQR PI controller

In Figure 6-10 Dynamic response of body rates for a step input in body pitch rate is plotted. Trim condition was 5 knots forward flight and defined rate reference is followed by the controller. Steady state error may be caused due to the non-linearity of the flight dynamics model.

The purpose of this chapter is to show the ability of simulating air vehicles and designing controllers. DLL version of the software is coupled with MATLAB Simulink by calling the library externally. To sum up, the architecture of the software enables user to integrate the dynamic simulation responses with the initial design parameters. Therefore, users can design air vehicles by considering the handling qualities together.

CHAPTER 7

CONCLUSION

In this thesis, flight dynamics software is developed to model, analyze and simulate both conventional and futuristic air vehicle designs flying at subsonic airspeeds and is possible to use for air vehicle flying at supersonic airspeeds with a little work. Since there are several types of air vehicles, creating generic software includes difficulties. Therefore, object oriented programming principles are utilized into the code which is written in C++ language. Each object is modeled as an individual component having internal properties and methods. These objects are communicating with each other and can be used in different models such as using **Airfoil** object for rotor, propeller and wings.

Each component of an air vehicle is modeled and validation works are presented for Rotor, Propeller and Wing components. Rotor model validation is conducted with the available S-76 helicopter Main Rotor test data from different test campaigns for hover and forward velocities. Moreover, commercial and/or open-source tools such as CAMRAD II, QPROP, XRotor and FLIGHTLAB are also used in the validation work. 2nd order lifting line theory used in wing model is validated for a NACA0015 wing experimental test data by comparing the lift coefficients from linear region to post stall regions.

Overall validation for the GAVM software is done by conducting a comparison work for LYNX helicopter, XV-15 tilting rotor aircraft. In order to show the control modeling and simulation capabilities of GAVM software, a sample Quad-Rotor configuration is modeled. Simple controller is designed and simulation results are provided.

Object oriented nature of the GAVM software enables the users to store individual components of different aircrafts within an XML file and analyze very quickly. The architecture itself provides the advantage of batch analyses for design optimization. Comparison of performance of isolated rotors and wings are possible. ControlMixer class is generated for arranging and coupling the required set of inputs for advanced and unconventional configurations. Trim module includes several algorithms to achieve single point trim, sweep trim, maneuvering trim and performance analyses.

Moreover, with the integration of Engine Deck via available power and fuel consumption tables, overall performance of the designed air vehicle may be analyzed by using the Performance module. Ceiling altitudes, payload and range calculations, best cruise velocities and best rate of climb algorithms are included in the performance module.

Besides an analysis model, GAVM provides a simulation environment for generated air vehicles by calling it as a dynamic library DLL. GAVM enables the user to simulate numerous helicopter and/or aircraft flights within one simulation environment if desired. For control design, it provides the numerically linearized state space representation of the air vehicle. Controllability and handling qualities studies are possible to be assessed.

As a result, a comprehensive design, analysis and simulation software is developed for air vehicle design. It can be used as either a standalone program or a shared library for several different purposes.

REFERENCES

[1] W. Johnson, “A History of Rotorcraft Comprehensive Analyses,” *NASA Tech. Note TP-2012-216012*, pp. 1–22, 2012.

[2] “Rotorcraft Simulation with Heli-Dyn.” [Online]. Available: http://www.heli-dyn.com/Helidyn/Home.html.

[3] G. Gursoy, O. Tarimci, and I. Yavrucuk, “Helicopter Slung Load Simulations Using Heli-Dyn+,” in *AIAA Modeling and Simulation Conference*, 2012.

[4] I. Yavrucuk, O. Tarimci, M. Katircioglu, E. Kubali, and D. Yilmaz, “A NEW HELICOPTER SIMULATION AND ANALYSIS TOOL: HELIDYN+,” in *European Rotorcraft Forum 2010*, 2010.

[5] “FLIGHTLAB - Theory Manual,” 2008.

[6] R. W. Du Val, “A Real-Time Blade Element Helicopter Simulation For Handling Qualities Analysis,” *15th Eur. Rotorcr. Forum*, 1989.

[7] R. W. Du Val, “A Low Cost - High Fidelity Real-Time Rotorcraft Simulation for Control and Handling Qualities Analysis,” in *Heli Japan 98: AHS International Meeting on Advanced Rotorcraft Technology and Disaster Relief*, 1998.

[8] R. W. Du Val, “A Real-Time Multi-Body Dynamics Architecture for Rotorcraft Simulation,” *AHS R. Aeronaut. Soc. Int. Conf. Chall. Realis. Rotorcr. Simul.*, 2001.

[9] “FLIGHTLAB Development Software Datasheet.” .

[10] D. A. Peters and C. J. He, “Finite State Induced Flow Models Part II : Three-Dimensional Rotor Disk,” vol. 32, no. 2, pp. 323–333, 1995.

[11] W. Johnson, “A Comprehensive Analytical Model of Rotorcraft Aerodynamics and Dynamics,” 1980.

[12] W. Johnson, “Development of a Comprehensive Analysis for Rotorcraft .1. Rotor Model and Wake Analysis,” *Vertica*, 1981.

[13] M. P. Scully, “Computation of Helicopter Rotor Wake Geometry and its Influence on Rotor Harmonic Loads,” *MIT ASRL TR*, 1975.

[14] J. Weissinger, “The Lift Distribution of Swept-Back Wings.,” *NACA Technical Memorandum No. 1120*. 1947.

[15] W. Johnson, “Technology Drivers in the Development of CAMRAD II,” in *AHS Aeromechanics Specialists Conference*, 1994.

[16] W. Johnson, “Rotorcraft Aeromechanics Applications of a Comprehensive Analysis,” in *Heli Japan 98: AHS International Meeting on Advanced Rotorcraft Technology and Disaster Relief*, 1998.

[17] W. Johnson, “CAMRAD II: Comprehensive Analytical Model of Rotorcraft Aerodynamics and Dynamics Volume I: Theory.”

[18] “CHARM - Comprehensive Hierarchical Aeromechanics Rotorcraft Model.” [Online]. Available: http://www.continuum-dynamics.com/pr-charm.html. [Accessed: 01-Jan-2017].

[19] T. R. Quackenbush, D. A. Wachspress, J. D. Keller, A. H. Boschitsch, B. J. Wasileski, and T. H. Lawrance, “Full Vehicle Flight Simulation with Real Time Free Wake Methods,” 2002.

[20] J. S. Berndt, “JSBSim: An Open Source Flight Dynamics Model in C++,” *AIAA Model. Simul. Technol. Conf. Exhib.*, no. August, pp. 1–27, 2011.

[21] “Flight Gear Flight Simulator.” [Online]. Available: http://www.flightgear.org/. [Accessed: 01-Jan-2017].

[22] D. Hodson, “OpenEaagles, An Open Source Simulation Framework,” *Am. Inst. Aeronaut. Astronaut. Model. Simul. January 2008 Newsl.*, vol. 1, no. 1, 2008.

[23] J. S. Berndt and A. De Marco, “Progress on and Usage of the Open Source Flight Dynamics Model Software Library , JSBSim,” *AIAA Model. Simul. Technol. Conf.*, no. August, pp. 1–17, 2009.

[24] O. A. Bauchau, “Dymore Solutions - Simulation Tools for Flexible Multibody Systems.” [Online]. Available: http://www.dymoresolutions.com/index.html. [Accessed: 01-Jan-2017].

[25] Dymore Solutions, “Dymore User’s Manual: Unsteady Aerodynamics.”

[26] D. Petot, “Differential equation modeling of dynamic stall,” *La Rech. Aerosp.*, vol. 5, pp. 59–72, 1989.

[27] J. G. Leishman and K. Q. Nguyen, “State-space representation of unsteady airfoil behavior,” *AIAA J.*, vol. 28, no. 5, pp. 836–844, 1990.

[28] J. G. Leishman and T. S. Beddoes, “A Semi-Empirical Model for Dynamic Stall,” *J. Am. Helicopter Soc.*, vol. 34, no. 3, pp. 3–17, 1989.

[29] “ORACLE - Java Object Oriented Programming Concepts.” [Online]. Available: https://docs.oracle.com/javase/tutorial/java/concepts/. [Accessed: 01-Jan-2017].

[30] W. Johnson, *Rotorcraft Aeromechanics*. New York: Cambridge University Press, 2013.

[31] J. R. Raol and J. Singh, *FLIGHT MECHANICS MODELING and ANALYSIS*. CRC Press, 2009.

[32] G. Cai, B. M. Chen, and T. H. Lee, *Unmanned Rotorcraft Systems, Advances in Industrial Control*. 2011.

[33] N.-N. G. Intellingence Agency of EUA, “World Geodetic System 1984,” *Wgs-84*, vol. 1984, p. 3, 1984.

[34] *Helicopter Flying Handbook*. U.S. Department of Transportation FAA.

[35] G. D. Padfield, *Helicopter Flight Dynamics*. 2007.

[36] G. J. Leishmann, “Principles of Helicopter Aerodynamics.” .

[37] D. M. Pitt and D. A. Peters, “Theoretical Prediction of Dynamic-Inflow Derivatives,” *Vertica*, vol. 5, no. 1, 1981.

[38] D. A. Peters and C. J. He, “Correlation of Measured Induced Velocities with a Finite-State Wake Model,” *J. Am. Helicopter Soc.*, vol. 36, no. 3, pp. 59–70, 1991.

[39] J. J. Howlett, “UH-60A Black Hawk Engineering Simulation Program Vol: I Mathematical Model,” California, 1981.

[40] R. W. Du Val, “Inertial Dynamics of a General Purpose Rotor Model,” 1979.

[41] R. T. N. Chen, “Flap-lag equations of motion of rigid, articulated rotor blades with three hinge sequences,” *NASA Tech. Memo. 100023*, no. November, 1987.

[42] A. Betz, “The Ground Effect on Lifting Propellers,” 1937.

[43] J. Zbrozek, “Ground Effect on the Lifting Rotor,” in *British ARC*, 1947.

[44] M. Knight and R. A. Hefner, “Analysis of Ground Effect on the Lifting Airscrew,” 1941.

[45] I. C. Cheeseman and W. E. Bennett, “The Effect on the Ground on a Helicopter in Coordinated Turns,” 1955.

[46] C. N. Stepniewski, W. Z. and Keys, *Rotary-Wing Aerodynamics*. 1979.

[47] R. W. Prouty, “Ground Effect and the Helicopter,” in *AIAA/AHS/ASEE Aircraft Design Systems and Operations Meeting*, 1985.

[48] H. J.S., “The Effect of the Ground on Helicopter Hovering Power Required,” in *32th Annual National V/STOL Forum of the American Helicopter Society*, 1976.

[49] H. Schlichting, *Boundary Layer Theory*, 7th ed. McGraw Hill - Hill Book Company, 1979.

[50] C. Tung and L. Branum, “Model Tilt-Rotor Hover Performance and Surface Pressure Measurement,” in *46th Annual Forum of the American Helicopter Society*, 1990.

[51] Z. Du and M. Selig, “A 3-D stall-delay model for horizontal axis wind turbine performance prediction,” *1998 ASME Wind Energy Symp.*, 1998.

[52] S. Gallay and E. Laurendeau, “Presentation of Modern Non Linear Lifting Line Methods,” pp. 1–13.

[53] A. M. Wickenheiser and E. Garcia, “Extended Nonlinear Lifting-Line Method for Aerodynamic Modeling of Reconfigurable Aircraft,” *J. Aircr.*, vol. 48, no. 5, pp. 1812–1817, 2011.

[54] A. Wickenheiser and E. Garcia, “Aerodynamic Modeling of Morphing Wings Using an Extended Lifting-Line Analysis,” *J. Aircr.*, vol. 44, no. 1, pp. 10–16, 2007.

[55] L. Prandtl, “Applications of modern hydrodynamics to aeronautics,” *Naca, Tech. Rep. 116*, p. 56, 1923.

[56] J. Anderson, *Fundamentals of Aerodynamics*. New York: McGraw Hill, 2001.

[57] J. a. Blackwell, “A finite-step method for calculation of theoretical load distributions for arbitrary lifting surface arrangements at subsonic speeds,” 1969.

[58] D. B. Owens, “Weissinger’s Model of the Nonlinear Lifting-Line Method for Aircraft Design,” *AIAA J.*, 1997.

[59] J. DeYoung and C. W. Harper, “Theoretical symmetric span loading at subsonic speeds for wings having arbitrary plan form,” 1948.

[60] H. Multhopp, “Die Berechnung der Auftriebsverteilung von Traflugein,” *Luftfahrtforschung*, vol. 15, no. 14, pp. 153–169, 1938.

[61] J. Tseng and C. Lan, “Calculation of aerodynamic characteristics of airplane configurations at high angles of attack,” 1988.

[62] C. P. van Dam, J. C. Vander Kam, and J. K. Paris, “Design-Oriented High-Lift Methodology for General Aviation and Civil Transport Aircraft,” *J. Aircr.*, vol. 38, no. 6, pp. 1076–1084, 2001.

[63] H. R. Smith, “ENGINEERING MODELS OF AIRCRAFT PROPELLERS AT INCIDENCE,” University of Glasgow, 2015.

[64] T. Theodorsen, *Theory of Propellers*. McGraw Hill, 1948.

[65] M. Drela, “QProp Formulation,” 2006.

[66] A. Prandtl, Ludwig and Betz, L. Prandtl, and A. Betz, “Vier Abhandlungen zur Hydrodynamik und Aerodynamik,” *III. Int. Methmatker Kongresses zu Heidelb.*, vol. 3, 1927.

[67] S. Goldstein, “On the vortex theory of screw propellers,” *Proc. R. Soc. London*, pp. 440–465, 1929.

[68] E. E. Larrabee and S. E. French, “Minimum induced loss windmills and propellers,” *J. Wind Eng. Ind. Aerodyn.*, vol. 15, pp. 317–327, 1983.

[69] J. Roskam, “Airplane Flight Dynamics and Automatic Flight Controls - Part I,” *Book*. p. 608, 1998.

[70] M. V. Cook, *Flight Dynamics Principles, Second Edition: A Linear Systems Approach to Aircraft Stability and Control*, vol. 53, no. 9. 2007.

[71] D. Rowell, “State-Space Representation of LTI Systems,” *System*, no. October, pp. 1–18, 2002.

[72] W. Bousman, “Aerodynamic Characteristics of SC1095 and SC1094 R8 Airfoils,” *Nasa*, no. December, pp. 1–50, 2003.

[73] D. Jepson, R. Moffitt, K. Hilzinger, and J. Bissell, “Analysis and Correlation of Test Data from an Advanced Technology Rotor System,” California, 1983.

[74] W. Johnson, “Performance and Loads Data from a Wind Tunnel Test of a Full-

Scale Rotor with Four Blade Tip Planforms," California, 1980.

[75] P. M. Shinoda, "Full-scale S-76 rotor performance and loads at low speeds in the NASA Ames 80- by 120-Foot Wind Tunnel," *NASA Tech. Rep. 110379,* vol. 1, pp. 1–248, 1996.

[76] V. J. Rossow, "Effect of Ground and/or Ceiling Planes on Thrust of Rotors in Hover," Caifornia, 1985.

[77] A. Yücekayalı, A. Ezertaş, and Y. Ortakaya, "Whirl Tower Testing and Hover Performance Evaluation of a 3 Meter Radius Rotor Design," *7. Aiac,* no. September, pp. 1–13, 2013.

[78] A. Ezertaş, A. Yücekayali, and Y. Ortakaya, "HOVER PERFORMANCE ASSESSMENT OF 3 METER RADIUS ROTOR ON WHIRL TOWER," in *European Rotorcraft Forum*, 2013.

[79] F. F. Felker, L. E. Young, and D. B. Signor, "Performance and Loads Data from a Hover Test of a Full-Scale Advanced Technology XV-15 Rotor," 1986.

[80] C.W.Acree, "An Improved CAMRAD Model for Aeroelastic Stability Analysis of the XV-15 With Advanced Technology Blades," *NASA Tech. Memo. 4448,* 1993.

[81] M. Drela, "XROTOR Download Page." [Online]. Available: http://web.mit.edu/drela/Public/web/xrotor/. [Accessed: 01-Jan-2017].

[82] S. J.K., "An experimental Investigation of the Effects of Leading Edge Modifications on the Post Stall Characteristics of an NACA 0015 Wing," 1979.

[83] D. H. Wood, "Full-Scale Tests of Metal Propellers at High Tip Speeds." NACA.

[84] B. H. Lau, A. W. Louie, N. Griffiths, and C. P. Sotiriou, "Performance and Rotor Loads Measurements of the Lynx XZ170 Helicopter with Rectangular Blades," *Nasa Tech. Memo. 104000,* 1993.

[85] M. D. Maisel, D. J. Giulianetti, and D. C. Dugan, "The History of The XV-15 Tilt Rotor Research Aircraft: From Concept to Flight," *NASA Spec. Publ. 4517,* p. 194, 2000.

[86] L. A. Viterna and D. C. Janetzke, "Theoretical and experimental power from large horizontal-axis wind turbines," vol. 82944, p. 19, 1982.

[87] B. Montgomerie, "Methods for root effects, tip effects and extending the angle of attack range to +-100 deg, with application to aerodynamics for blades on wind turbines and propellers.," 2004.

[88] S. W. Ferguson, "Development and validation of a simulation for a generic tilt-rotor aircraft," NASA, 1989.

[89] J. A. Weiberg and M. D. Maisel, "Wind-Tunnel Tests of the XV-15 Tilt Rotor Aircraft," California, 1980.

[90] J. P. Magee and K. G. Wernicke, "XV-15 Tilt Rotor Research Aircraft Program Report," California.

[91] W. L. Arrington, “Flight Test Report NASA CR 177406 and USAACSCOM TR-86- A-1,” 1985.

APPENDICES

A. SAMPLE FILES

Sample XV 15 rotor input file

```
    <Rotor name="Left Rotor" type="1" Id="1">
      <!--  1 = Main Rotor -->
      <Naz value="36" />
      <!--  CCW = 1 & CW = -1 -->
      <Rotation value="1" />
      <Airfoil>
        <Profile Id="0" value="C:/CODE/XV15/VR7.dat" />
        <Profile Id="1" value="C:/CODE/XV15/VR8.dat" />
      </Airfoil>
      <Inflow>
        <GroundEffect status="0" height="3.048" />
        <Kappa value="1.3" Mu="0.0"/>
      </Inflow>
      <Hub>
        <Position>
          <FS value="7.64" />
          <BL value="5.0" />
          <WL value="0.0" />
        </Position>
        <Orientation>
          <!--    Conventional Rotor Orientation is ZERO. All axes
are assumed as body axis and rotation is ZYX-->
          <X value="0.0" />
          <Y value="0.0" />
          <Z value="0.0" />
        </Orientation>
        <ShaftTilt>
          <X value="0.0" />
          <Y value="0.0" />
          <Z value="0.0" />
        </ShaftTilt>
        <hOffset    value="0.76" /> <!-- dimensional -->
        <PhaseAngle value="0.0" />
        <Delta3     value="0" />
        <FlapDyn    value="1"/>
        <LagDyn     value="0" />
        <Precone    value="0" />
      </Hub>
      <Blade>
        <BladeNum value="3" />
        <Nrad value="18" />
        <StallDelay value="0" />
        <Rtip value="3.81" />
        <Rcutout value="0.50" /> <!-- dimensional -->
```

```
  <TwistDist>
      <Twist Rpos="0.000"     angle="0.000"   />
      <Twist Rpos="0.101"     angle="31.600"  />
      <Twist Rpos="0.174"     angle="27.250"  />
      <Twist Rpos="0.258"     angle="22.300"  />
      <Twist Rpos="0.364"     angle="16.750"  />
      <Twist Rpos="0.482"     angle="10.450"  />
      <Twist Rpos="0.607"     angle="5.200"   />
      <Twist Rpos="0.736"     angle="0.250"   />
      <Twist Rpos="0.830"     angle="-2.600"  />
      <Twist Rpos="0.891"     angle="-4.250"  />
      <Twist Rpos="0.947"     angle="-6.950"  />
      <Twist Rpos="1.000"     angle="-9.200"  />
  </TwistDist>
  <I_b value="200.12" />
  <M_b value="25" />
  <W_b value="100.0" />
  <K_b value="350.0" />
  <BladeProfiles>
  <!--    Must start with 0 and end up with 1.0 -->
      <Airfoil Id="0" Rpos="0.0" />
      <Airfoil Id="0" Rpos="0.9" />
      <Airfoil Id="1" Rpos="0.95" />
      <Airfoil Id="1" Rpos="1.0" />
  </BladeProfiles>
  <ChordDist>
      <Chord Rpos="0.00"      length="0.050"  />
      <Chord Rpos="0.10"      length="0.534"  />
      <Chord Rpos="0.14"      length="0.533"  />
      <Chord Rpos="0.19"      length="0.529"  />
      <Chord Rpos="0.23"      length="0.534"  />
      <Chord Rpos="0.27"      length="0.533"  />
      <Chord Rpos="0.31"      length="0.507"  />
      <Chord Rpos="0.37"      length="0.506"  />
      <Chord Rpos="0.46"      length="0.504"  />
      <Chord Rpos="0.54"      length="0.507"  />
      <Chord Rpos="0.62"      length="0.506"  />
      <Chord Rpos="0.69"      length="0.506"  />
      <Chord Rpos="0.72"      length="0.474"  />
      <Chord Rpos="0.76"      length="0.432"  />
      <Chord Rpos="0.78"      length="0.399"  />
      <Chord Rpos="0.82"      length="0.382"  />
      <Chord Rpos="0.87"      length="0.362"  />
      <Chord Rpos="0.90"      length="0.350"  />
      <Chord Rpos="0.92"      length="0.332"  />
      <Chord Rpos="0.95"      length="0.311"  />
      <Chord Rpos="0.97"      length="0.261"  />
      <Chord Rpos="0.98"      length="0.209"  />
      <Chord Rpos="0.99"      length="0.166"  />
      <Chord Rpos="1.00"      length="0.124"  />
  </ChordDist>
  <SweepDist>
    <Sweep Rpos="0.0"     angle="0.00" />
    <Sweep Rpos="1.0"     angle="0.00" />
  </SweepDist>
</Blade>
```

```
  <Inputs>
    <Collective value="28.0"  active = "1"  max="50" min="5"/>
    <RPM       value="589"   active = "0"  max="600" min="100"/>
    <LonCyclic  value="0.0"     active = "1"    />
    <LatCyclic  value="0.0"     active = "1"    />
    <Tilt       value="0"       active = "0"    />
    <Cant       value="0"       active = "0"    />
  </Inputs>
  <States>
    <B0 value="2.5" />
    <B1C value="0" />
    <B1S value="0" />
    <D0 value="0.0" />
    <D1C value="0.0" />
    <D1S value="0.0" />
  </States>
</Rotor>
```

Sample Quadrotor File

```
<Helicopter name="Quadrotor">
    <Unit value="0" />
        <Weight>
      <Mass value="8" />
      <Ixx value="0.1" />
      <Ixy value="0.0" />
      <Ixz value="0.0" />
      <Iyx value="0.0" />
      <Iyy value="0.1" />
      <Iyz value="0.0" />
      <Izx value="0.0" />
      <Izy value="0.0" />
      <Izz value="0.1" />
      <Position>
        <!--  location of CG -->
        <FS value="0" />
        <BL value="0.0" />
        <WL value="-0.1" />
      </Position>


    </Weight>

    <Propeller name="pervaneOn" Id="1">
      <!--  CCW = 1 & CW = -1 -->
      <Rotation value="1" />
      <Airfoil>
        <Profile Id="0" value="C:/CODE/CLARKY/clarky.dat" />
      </Airfoil>
      <Inflow>
        <GroundEffect status="0" height="3.048" />
        <Kappa value="1.16" Mu="0.0"/>
      </Inflow>
      <Hub>
        <Position>
          <FS value="0.1" />
          <BL value="0" />
          <WL value="0" />
        </Position>
        <Orientation>
          <X value="0.0" />
          <Y value="0.0" />
          <Z value="0.0" />
        </Orientation>
        <ShaftTilt>
          <X value="0.0" />
          <Y value="0.0" />
          <Z value="0.0" />
        </ShaftTilt>
        <hOffset value="0.02" />
      </Hub>
      <Blade>
        <BladeNum value="2" />
```

```
  <Nrad value="50" />
  <Rtip value="0.14478" />
  <Rcutout value="0.01" />
  <BladeProfiles>
    <Airfoil Id="0" Rpos="0.0" />
    <Airfoil Id="0" Rpos="1.0" />
  </BladeProfiles>
  <ChordDist>
      <Chord Rpos="0.2"   length="0.01708" />
      <Chord Rpos="0.25"  length="0.01795" />
      <Chord Rpos="0.3"   length="0.01882" />
      <Chord Rpos="0.35"  length="0.01911" />
      <Chord Rpos="0.4"   length="0.0194" />
      <Chord Rpos="0.45"  length="0.0194" />
      <Chord Rpos="0.5"   length="0.01911" />
      <Chord Rpos="0.55"  length="0.01882" />
      <Chord Rpos="0.6"   length="0.01839" />
      <Chord Rpos="0.65"  length="0.01737" />
      <Chord Rpos="0.7"   length="0.01636" />
      <Chord Rpos="0.75"  length="0.0152" />
      <Chord Rpos="0.8"   length="0.01361" />
      <Chord Rpos="0.85"  length="0.01231" />
      <Chord Rpos="0.9"   length="0.01042" />
      <Chord Rpos="0.95"  length="0.00869" />
      <Chord Rpos="1"     length="0.00724" />
  </ChordDist>
  <SweepDist> <!-- set them to zero for now -->
    <Sweep Rpos="0.0"     angle="0" />
    <Sweep Rpos="1.0"     angle="0" />
  </SweepDist>
  <TwistDist>
      <Twist Rpos="0.2"   angle="26.9896" />
      <Twist Rpos="0.25"  angle="24.0194" />
      <Twist Rpos="0.3"   angle="21.697" />
      <Twist Rpos="0.35"  angle="19.529" />
      <Twist Rpos="0.4"   angle="17.6568" />
      <Twist Rpos="0.45"  angle="15.7984" />
      <Twist Rpos="0.5"   angle="14.2866" />
      <Twist Rpos="0.55"  angle="13.0343" />
      <Twist Rpos="0.6"   angle="11.9808" />
      <Twist Rpos="0.65"  angle="11.0829" />
      <Twist Rpos="0.7"   angle="10.3089" />
      <Twist Rpos="0.75"  angle="9.635" />
      <Twist Rpos="0.8"   angle="9.0431" />
      <Twist Rpos="0.85"  angle="8.5192" />
      <Twist Rpos="0.9"   angle="8.0523" />
      <Twist Rpos="0.95"  angle="7.6336" />
      <Twist Rpos="1"     angle="7.2561" />
  </TwistDist>
</Blade>
<Inputs>
  <Collective value="0.0"     active="0"  />
  <RPM        value="12000"   active="1"  />
  <Tilt       value="0.0"     active="0"  />
  <Cant       value="0.0"     active="0"  />
</Inputs>
```

```
    </Propeller>
    <Propeller name="pervaneArka" Id="2">
      <!--  CCW = 1 & CW = -1 -->
      <Rotation value="1" />
      <Airfoil>
        <Profile Id="0" value="C:/CODE/CLARKY/clarky.dat" />
</Airfoil>
      <Inflow>
        <GroundEffect status="0" height="3.048" />
        <Kappa value="1.16" Mu="0.0"/>
      </Inflow>
      <Hub>
        <Position>
          <FS value="-0.1" />
          <BL value="0" />
          <WL value="0" />
        </Position>
        <Orientation>
          <X value="0.0" />
          <Y value="0.0" />
          <Z value="0.0" />
        </Orientation>
        <ShaftTilt>
          <X value="0.0" />
          <Y value="0.0" />
          <Z value="0.0" />
        </ShaftTilt>
        <hOffset value="0.02" />
      </Hub>
      <Blade>
        <BladeNum value="2" />
        <Nrad value="50" />
        <Rtip value="0.14478" />
        <Rcutout value="0.01" />
        <BladeProfiles>
          <Airfoil Id="0" Rpos="0.0" />
          <Airfoil Id="0" Rpos="1.0" />
        </BladeProfiles>
        <ChordDist>
            <Chord Rpos="0.2"   length="0.01708" />
            <Chord Rpos="0.25"  length="0.01795" />
            <Chord Rpos="0.3"   length="0.01882" />
            <Chord Rpos="0.35"  length="0.01911" />
            <Chord Rpos="0.4"   length="0.0194" />
            <Chord Rpos="0.45"  length="0.0194" />
            <Chord Rpos="0.5"   length="0.01911" />
            <Chord Rpos="0.55"  length="0.01882" />
            <Chord Rpos="0.6"   length="0.01839" />
            <Chord Rpos="0.65"  length="0.01737" />
            <Chord Rpos="0.7"   length="0.01636" />
            <Chord Rpos="0.75"  length="0.0152" />
            <Chord Rpos="0.8"   length="0.01361" />
            <Chord Rpos="0.85"  length="0.01231" />
            <Chord Rpos="0.9"   length="0.01042" />
            <Chord Rpos="0.95"  length="0.00869" />
            <Chord Rpos="1"     length="0.00724" />
        </ChordDist>
        <SweepDist> <!-- set them to zero for now -->
          <Sweep Rpos="0.0"     angle="0" />
          <Sweep Rpos="1.0"     angle="0" />
        </SweepDist>
        <TwistDist>
            <Twist Rpos="0.2"   angle="26.9896" />
            <Twist Rpos="0.25"  angle="24.0194" />
            <Twist Rpos="0.3"   angle="21.697" />
            <Twist Rpos="0.35"  angle="19.529" />
            <Twist Rpos="0.4"   angle="17.6568" />
            <Twist Rpos="0.45"  angle="15.7984" />
            <Twist Rpos="0.5"   angle="14.2866" />
            <Twist Rpos="0.55"  angle="13.0343" />
            <Twist Rpos="0.6"   angle="11.9808" />
            <Twist Rpos="0.65"  angle="11.0829" />
            <Twist Rpos="0.7"   angle="10.3089" />
            <Twist Rpos="0.75"  angle="9.635" />
            <Twist Rpos="0.8"   angle="9.0431" />
            <Twist Rpos="0.85"  angle="8.5192" />
            <Twist Rpos="0.9"   angle="8.0523" />
            <Twist Rpos="0.95"  angle="7.6336" />
            <Twist Rpos="1"     angle="7.2561" />
        </TwistDist>
      </Blade>
      <Inputs>
        <Collective value="0.0"     active="0"  />
        <RPM        value="12000"   active="1"  />
        <Tilt       value="0.0"     active="0"  />
        <Cant       value="0.0"     active="0"  />
      </Inputs>
    </Propeller>


        <Propeller name="pervaneSol" Id="3">
      <!--  CCW = 1 & CW = -1 -->
      <Rotation value="-1" />
      <Airfoil>
        <Profile Id="0" value="C:/CODE/CLARKY/clarky.dat" />
      </Airfoil>
      <Inflow>
        <GroundEffect status="0" height="3.048" />
        <Kappa value="1.16" Mu="0.0"/>
      </Inflow>
      <Hub>
        <Position>
          <FS value="0" />
          <BL value="0.1" />
          <WL value="0" />
        </Position>
        <Orientation>
          <!--    Conventional Rotor Orientation is ZERO. All axes
are assumed as body axis and rotation is ZYX-->
          <X value="0.0" />
          <Y value="0.0" />
          <Z value="0.0" />
        </Orientation>
        <ShaftTilt>
          <X value="0.0" />
          <Y value="0.0" />
          <Z value="0.0" />
        </ShaftTilt>
        <hOffset value="0.02" />
      </Hub>
      <Blade>
        <BladeNum value="2" />
        <Nrad value="50" />
        <Rtip value="0.14478" />
        <Rcutout value="0.01" />
        <BladeProfiles>
          <Airfoil Id="0" Rpos="0.0" />
          <Airfoil Id="0" Rpos="1.0" />
        </BladeProfiles>
        <ChordDist>
            <Chord Rpos="0.2"   length="0.01708" />
            <Chord Rpos="0.25"  length="0.01795" />
            <Chord Rpos="0.3"   length="0.01882" />
            <Chord Rpos="0.35"  length="0.01911" />
            <Chord Rpos="0.4"   length="0.0194" />
            <Chord Rpos="0.45"  length="0.0194" />
            <Chord Rpos="0.5"   length="0.01911" />
            <Chord Rpos="0.55"  length="0.01882" />
            <Chord Rpos="0.6"   length="0.01839" />
            <Chord Rpos="0.65"  length="0.01737" />
            <Chord Rpos="0.7"   length="0.01636" />
            <Chord Rpos="0.75"  length="0.0152" />
            <Chord Rpos="0.8"   length="0.01361" />
            <Chord Rpos="0.85"  length="0.01231" />
            <Chord Rpos="0.9"   length="0.01042" />
            <Chord Rpos="0.95"  length="0.00869" />
            <Chord Rpos="1"     length="0.00724" />
        </ChordDist>
        <SweepDist> <!-- set them to zero for now -->
          <Sweep Rpos="0.0"     angle="0" />
          <Sweep Rpos="1.0"     angle="0" />
        </SweepDist>
        <TwistDist>
            <Twist Rpos="0.2"   angle="26.9896" />
            <Twist Rpos="0.25"  angle="24.0194" />
            <Twist Rpos="0.3"   angle="21.697" />
            <Twist Rpos="0.35"  angle="19.529" />
            <Twist Rpos="0.4"   angle="17.6568" />
            <Twist Rpos="0.45"  angle="15.7984" />
            <Twist Rpos="0.5"   angle="14.2866" />
            <Twist Rpos="0.55"  angle="13.0343" />
            <Twist Rpos="0.6"   angle="11.9808" />
            <Twist Rpos="0.65"  angle="11.0829" />
            <Twist Rpos="0.7"   angle="10.3089" />
            <Twist Rpos="0.75"  angle="9.635" />
            <Twist Rpos="0.8"   angle="9.0431" />
            <Twist Rpos="0.85"  angle="8.5192" />
            <Twist Rpos="0.9"   angle="8.0523" />
```

```
            <Twist Rpos="0.95"  angle="7.6336" />
            <Twist Rpos="1"     angle="7.2561" />
        </TwistDist>
      </Blade>
      <Inputs>
        <Collective value="0.0"     active="0"  />
        <RPM        value="12000"   active="1"  />
        <Tilt       value="0.0"     active="0"  />
        <Cant       value="0.0"     active="0"  />
      </Inputs>
    </Propeller>


        <Propeller name="pervaneSag" Id="4">
      <!--  CCW = 1 & CW = -1 -->
      <Rotation value="-1" />
      <Airfoil>
        <Profile Id="0" value="C:/CODE/CLARKY/clarky.dat" />
      </Airfoil>
      <Inflow>
        <GroundEffect status="0" height="3.048" />
        <Kappa value="1.16" Mu="0.0"/>
      </Inflow>
      <Hub>
        <Position>
          <FS value="0" />
          <BL value="-0.1" />
          <WL value="0" />
        </Position>
        <Orientation>
          <!--    Conventional Rotor Orientation is ZERO. All axes
are assumed as body axis and rotation is ZYX-->
          <X value="0.0" />
          <Y value="0.0" />
          <Z value="0.0" />
        </Orientation>
        <ShaftTilt>
          <X value="0.0" />
          <Y value="0.0" />
          <Z value="0.0" />
        </ShaftTilt>
        <hOffset value="0.02" />
      </Hub>
      <Blade>
        <BladeNum value="2" />
        <Nrad value="50" />
        <Rtip value="0.14478" />
        <Rcutout value="0.01" />
        <BladeProfiles>
          <Airfoil Id="0" Rpos="0.0" />
          <Airfoil Id="0" Rpos="1.0" />
        </BladeProfiles>
        <ChordDist>
            <Chord Rpos="0.2"   length="0.01708" />
            <Chord Rpos="0.25"  length="0.01795" />
            <Chord Rpos="0.3"   length="0.01882" />
            <Chord Rpos="0.35"  length="0.01911" />
            <Chord Rpos="0.4"   length="0.0194" />
            <Chord Rpos="0.45"  length="0.0194" />
            <Chord Rpos="0.5"   length="0.01911" />
            <Chord Rpos="0.55"  length="0.01882" />
            <Chord Rpos="0.6"   length="0.01839" />
            <Chord Rpos="0.65"  length="0.01737" />
            <Chord Rpos="0.7"   length="0.01636" />
            <Chord Rpos="0.75"  length="0.0152" />
            <Chord Rpos="0.8"   length="0.01361" />
            <Chord Rpos="0.85"  length="0.01231" />
            <Chord Rpos="0.9"   length="0.01042" />
            <Chord Rpos="0.95"  length="0.00869" />
            <Chord Rpos="1"     length="0.00724" />
        </ChordDist>
        <SweepDist> <!-- set them to zero for now -->
          <Sweep Rpos="0.0"     angle="0" />
          <Sweep Rpos="1.0"     angle="0" />
        </SweepDist>
        <TwistDist>
            <Twist Rpos="0.2"   angle="26.9896" />
            <Twist Rpos="0.25"  angle="24.0194" />
            <Twist Rpos="0.3"   angle="21.697" />
            <Twist Rpos="0.35"  angle="19.529" />
            <Twist Rpos="0.4"   angle="17.6568" />
            <Twist Rpos="0.45"  angle="15.7984" />
            <Twist Rpos="0.5"   angle="14.2866" />
            <Twist Rpos="0.55"  angle="13.0343" />
            <Twist Rpos="0.6"   angle="11.9808" />
            <Twist Rpos="0.65"  angle="11.0829" />
            <Twist Rpos="0.7"   angle="10.3089" />
            <Twist Rpos="0.75"  angle="9.635" />
            <Twist Rpos="0.8"   angle="9.0431" />
            <Twist Rpos="0.85"  angle="8.5192" />
            <Twist Rpos="0.9"   angle="8.0523" />
            <Twist Rpos="0.95"  angle="7.6336" />
            <Twist Rpos="1"     angle="7.2561" />
        </TwistDist>
      </Blade>
      <Inputs>
        <Collective value="0.0"     active="0"  />
        <RPM        value="12000"   active="1"  />
        <Tilt       value="0.0"     active="0"  />
        <Cant       value="0.0"     active="0"  />
      </Inputs>
    </Propeller>


    <Fuselage name="Main Fuselage ">
      <Position>
        <!--  location where the aerodynamic tables exists   - force
acting point  -->
        <FS value="0.0" />
        <BL value="0.0" />
        <WL value="0.0" />
      </Position>
      <Orientation>
        <!--    Conventional Rotor Orientation is ZERO. All axes are
assumed as body axis and rotation is ZYX-->
        <X value="0.0" />
        <Y value="0.0" />
        <Z value="0.0" />
      </Orientation>
       <!--    0 body | 1 wind-->
      <Polar value="C:/CODE/bodyQuad.dat" type="0"/>
      <References>
        <Area value="1.0" />
        <Length value="1.0" />
      </References>
    </Fuselage>


    <ControlMixer active="1">
        <Input name="throttle" value="1.0" max="1.25" min="0">
            <Propeller Id="1" input="RPM" K="12000.0" />
            <Propeller Id="2" input="RPM" K="12000.0" />
            <Propeller Id="3" input="RPM" K="12000.0" />
            <Propeller Id="4" input="RPM" K="12000.0" />
        </Input>
        <Input name="roll" value="0" max="3" min="-3">
            <Propeller Id="3" input="RPM" K="-1000.0" />
            <Propeller Id="4" input="RPM" K="1000.0" />
        </Input>
        <Input name="pitch" value="0" max="3" min="-3">
            <Propeller Id="1" input="RPM" K="1000.0" />
            <Propeller Id="2" input="RPM" K="-1000.0" />
        </Input>
        <Input name="yaw" value="0" max="3" min="-3">
            <Propeller Id="1" input="RPM" K="1000.0" />
            <Propeller Id="2" input="RPM" K="1000.0" />
            <Propeller Id="3" input="RPM" K="-1000.0" />
            <Propeller Id="4" input="RPM" K="-1000.0" />
        </Input>
    </ControlMixer>

</Helicopter>
```

Sample Trim Configuration File

```
<Trim active="1" type="rotorcraft">
    <InitialCond>  <!-- update which one exists otherwise do not 
change -->

        <Inputs> <!-- units are in degrees -->
            <Propeller id="1">
                <input RPM="9000.0" />
            </Propeller>
            <Propeller id="2">
                <input RPM="9000.0" />
            </Propeller>
            <Propeller id="3">
                <input RPM="9000.0" />
            </Propeller>
            <Propeller id="4">
                <input RPM="9000.0" />
            </Propeller>
        </Inputs>

        <States> <!-- units are in degrees and meters -->
            <state Phi="0.0"    Phidot="0.0"/>
            <state Theta="0.0"  Thetadot="0.0"/>
            <state Psi="0.0"    Psidot="0.0"/>
            <state Longitude="0.0" />
            <state Latitude="0.0" />
        </States>
    </InitialCond>

    <TargetCond id="3">
    <!-- describe a flight condition whether it is in trim or not --
>
        <GroundEffect   value="0" />
        <Altitude       value="10" />
        <dISA           value="0" />
        <Relax          value="0.4" />
        <Tolerance      value="0.0001" />
        <Weight         value="10" />
        <Outputs>
            <output KIAS="0.0"      target="0" /> <!-- KIAS or KTAS 
MUST be specified as target -->
            <output KTAS="5.1"      target="1" /> <!-- KIAS or KTAS 
MUST be specified as target -->
            <output SideSlip="0.0"  target="1" />
            <output GammaZ="0.0"    target="1" />
            <output RoC="0.0"       target="1" /> <!-- [feet/min]  -
->
        </Outputs>
    </TargetCond>
</Trim>
```

Sample Output File for Quadrotor

```
        --------####################### TRIM CASE 1 #######################---------

-----------######-------------- DETAILED OUTPUT FOR MAINFRAME -----------######--------------

   ------ TRIM INFO ------
          KIAS:       5.10   [knots]
          KTAS:       5.10   [knots]
         Alpha:     -0.003     [deg]
          Beta:      0.000     [deg]
           RoC:      -0.00  [ft/min]
        Weight:      10.00      [kg]
      Altitude:      10.00  [ft/min]
   Temperature:      14.98 [Celcius]
       Density:   1.224696   [kg/m3]
   Total Power:       2.50      [hp]

                    Parameter |        X component |        Y component |        Z component |
    ------------------------------------------------------------------------------------------
   CG Location (FS-BL-WL) [m] |                0.00|                0.00|               -0.10|
        Body Velocities [m/s] |                2.62|                0.00|               -0.00|
    Body Accelerations [m/s2] |               -0.00|                0.00|               -0.00|
           Body Rates [rad/s] |                0.00|                0.00|                0.00|
            Body Angles [deg] |                0.00|               -0.00|                0.00|
       Body Angle Rates [deg] |                0.00|                0.00|                0.00|
      Earth NED Positions [m] |                0.00|                0.00|                3.05|
               CG Forces [N] |               -0.01|                0.00|              -98.10|
   Earth NED Velocities [m/s] |                2.62|                0.00|                0.00|
              Load Factors [g] |               -0.00|                0.00|               -1.00|

 -----------######-------------- DETAILED OUTPUT FOR FUSELAGE #  -----------######--------------

 Solution method: TABLE LOOKUP

 ------------------------- Geometric Parameters for FUSELAGE  -------------------------

                    Parameter |        X component |        Y component |        Z component |
    ------------------------------------------------------------------------------------------
      Location (FS-BL-WL) [m] |            0.000000|            0.000000|            0.000000|
  Position wrt to CG(Body)[m] |            0.000000|            0.000000|           -0.100000|
            Orientation [deg] |            0.000000|            0.000000|            0.000000|

            Reference Area (S):      1.000 [m^2]
          Reference Length (l):      1.000 [m]

 ------------------------- Performance Parameters for FUSELAGE  -------------------------

                    Parameter |        X component |        Y component |        Z component |
    ------------------------------------------------------------------------------------------
             Forces at CG [N] |              -0.006|               0.000|               0.000|
            Moments at CG [N] |               0.000|               0.001|               0.000|

       Angle of Attack (alpha):     -0.003 [deg]
      Angle of Sideslip (beta):      0.000 [deg]

 -----------######-------------- DETAILED OUTPUT FOR PROP # 1 -----------######--------------

 ------------------------- Geometric Parameters for Prop 1 -------------------------
   # of blades:          2            Tip Radius:     0.145 [m]       T-Weighted Chord:       0.0137 [m]
 RPM [rev/min]:   11951.43      Tip Speed (Hover):    181.20 [m/s]            Root Chord:        0.018 [m]
      Solidity:     0.0773           Aspect Ratio:     10.59             Advance Ratio:        0.014
  T-W Solidity:     0.0601 [%]       Hinge Offset:    13.814 [%]          Root Cut Out:         6.91 [%]
 Blade Mass-x1:      0.000 [kg]          Rotation:       CCW               Lock Number:           --


                    Parameter |        X component |        Y component |        Z component |
    ------------------------------------------------------------------------------------------
      Location (FS-BL-WL) [m] |               0.100|               0.000|               0.000|
  Position wrt to CG(Body)[m] |              -0.100|               0.000|              -0.100|
            Orientation [deg] |               0.000|               0.000|               0.000|
             Shaft Tilt [deg] |               0.000|               0.000|               0.000|
               Hub Forces [N] |                0.00|                0.00|              -24.53|
             Hub Moments [Nm] |                0.00|                0.00|                0.35|
                CG Forces [N] |                0.00|                0.00|              -24.53|
              CG Moments [Nm] |                0.00|               -2.45|                0.35|
     Rotor Body Velocity [m/s] |                2.62|                0.00|               -0.00|

 # of airfoils: 1
     Airfoil 1 name: Clark-Y

 ------------------------- Performance Parameters for Prop 1 -------------------------

  ------ TRIM INFO ------
                 RPM: 11951.43 [rev/min]

        Mean Inflow (L0):     13.049 [m/s]

   Thrust Coefficient CT: 0.00926303
   Torque Coefficient CQ: 0.00090022
    Power Coefficient CP: 0.00090046
```

```
     CT/sigma: 0.15412490
     CQ/sigma: 0.01497846
     CP/sigma: 0.01498250

         Thrust:    24.5280   [N]
         Torque:     0.3451  [Nm]
    Total Power:     432.05   [W]      0.58  [hp]

-----------######-------------- DETAILED OUTPUT FOR PROP # 2 -----------######--------------

-------------------------- Geometric Parameters for Prop 2 --------------------------
  # of blades:          2             Tip Radius:     0.145 [m]        T-Weighted Chord:       0.0137 [m]
RPM [rev/min]:   11950.04      Tip Speed (Hover):    181.18 [m/s]            Root Chord:        0.018 [m]
     Solidity:     0.0773           Aspect Ratio:     10.59               Advance Ratio:        0.014
 T-W Solidity:     0.0601 [%]       Hinge Offset:    13.814 [%]            Root Cut Out:         6.91 [%]
Blade Mass-x1:      0.000 [kg]          Rotation:       CCW                 Lock Number:           --


                   Parameter |        X component |        Y component |        Z component |
   ------------------------------------------------------------------------------------------
     Location (FS-BL-WL) [m] |              -0.100|               0.000|               0.000|
 Position wrt to CG(Body)[m] |               0.100|               0.000|              -0.100|
           Orientation [deg] |               0.000|               0.000|               0.000|
            Shaft Tilt [deg] |               0.000|               0.000|               0.000|
              Hub Forces [N] |                0.00|                0.00|              -24.52|
            Hub Moments [Nm] |                0.00|                0.00|                0.35|
               CG Forces [N] |                0.00|                0.00|              -24.52|
             CG Moments [Nm] |                0.00|                2.45|                0.35|
   Rotor Body Velocity [m/s] |                2.62|                0.00|               -0.00|

# of airfoils: 1
    Airfoil 1 name: Clark-Y

-------------------------- Performance Parameters for Prop 2 --------------------------

 ------ TRIM INFO ------
                RPM: 11950.04 [rev/min]

       Mean Inflow (L0):     13.047 [m/s]

  Thrust Coefficient CT: 0.00926303
  Torque Coefficient CQ: 0.00090022
   Power Coefficient CP: 0.00090046

     CT/sigma: 0.15412490
     CQ/sigma: 0.01497846
     CP/sigma: 0.01498250

         Thrust:    24.5223   [N]
         Torque:     0.3450  [Nm]
    Total Power:     431.90   [W]      0.58  [hp]

-----------######-------------- DETAILED OUTPUT FOR PROP # 3 -----------######--------------

-------------------------- Geometric Parameters for Prop 3 --------------------------
  # of blades:          2             Tip Radius:     0.145 [m]        T-Weighted Chord:       0.0137 [m]
RPM [rev/min]:   11950.73      Tip Speed (Hover):    181.19 [m/s]            Root Chord:        0.018 [m]
     Solidity:     0.0773           Aspect Ratio:     10.59               Advance Ratio:        0.014
 T-W Solidity:     0.0601 [%]       Hinge Offset:    13.814 [%]            Root Cut Out:         6.91 [%]
Blade Mass-x1:      0.000 [kg]          Rotation:        CW                 Lock Number:           --


                   Parameter |        X component |        Y component |        Z component |
   ------------------------------------------------------------------------------------------
     Location (FS-BL-WL) [m] |               0.000|               0.100|               0.000|
 Position wrt to CG(Body)[m] |               0.000|              -0.100|              -0.100|
           Orientation [deg] |               0.000|               0.000|               0.000|
            Shaft Tilt [deg] |               0.000|               0.000|               0.000|
              Hub Forces [N] |                0.00|                0.00|              -24.53|
            Hub Moments [Nm] |                0.00|                0.00|               -0.35|
               CG Forces [N] |                0.00|                0.00|              -24.53|
             CG Moments [Nm] |                2.45|                0.00|               -0.35|
   Rotor Body Velocity [m/s] |                2.62|                0.00|               -0.00|

# of airfoils: 1
    Airfoil 1 name: Clark-Y

-------------------------- Performance Parameters for Prop 3 --------------------------

 ------ TRIM INFO ------
                RPM: 11950.73 [rev/min]

       Mean Inflow (L0):     13.048 [m/s]

  Thrust Coefficient CT: 0.00926303
  Torque Coefficient CQ: -0.00090022
   Power Coefficient CP: 0.00090046

     CT/sigma: 0.15412489
     CQ/sigma: -0.01497846
     CP/sigma: 0.01498250

         Thrust:    24.5252   [N]
         Torque:    -0.3451  [Nm]
    Total Power:     431.97   [W]      0.58  [hp]

-----------######-------------- DETAILED OUTPUT FOR PROP # 4 -----------######--------------

-------------------------- Geometric Parameters for Prop 4 --------------------------
  # of blades:          2            Tip Radius:     0.145 [m]       T-Weighted Chord:       0.0137 [m]
RPM [rev/min]:   11950.73      Tip Speed (Hover):    181.19 [m/s]           Root Chord:        0.018 [m]
     Solidity:     0.0773          Aspect Ratio:     10.59             Advance Ratio:        0.014
 T-W Solidity:     0.0601 [%]      Hinge Offset:    13.814 [%]           Root Cut Out:         6.91 [%]
Blade Mass-x1:      0.000 [kg]         Rotation:        CW                Lock Number:           --

                  Parameter |        X component |        Y component |        Z component |
   ------------------------------------------------------------------------------------------
     Location (FS-BL-WL) [m] |               0.000|              -0.100|               0.000|
 Position wrt to CG(Body)[m] |               0.000|               0.100|              -0.100|
           Orientation [deg] |               0.000|               0.000|               0.000|
            Shaft Tilt [deg] |               0.000|               0.000|               0.000|
              Hub Forces [N] |                0.00|                0.00|              -24.53|
            Hub Moments [Nm] |                0.00|                0.00|               -0.35|
               CG Forces [N] |                0.00|                0.00|              -24.53|
             CG Moments [Nm] |               -2.45|                0.00|               -0.35|
   Rotor Body Velocity [m/s] |                2.62|                0.00|               -0.00|

# of airfoils: 1
    Airfoil 1 name: Clark-Y

-------------------------- Performance Parameters for Prop 4 --------------------------

 ------ TRIM INFO ------
                RPM: 11950.73 [rev/min]

       Mean Inflow (L0):     13.048 [m/s]

  Thrust Coefficient CT: 0.00926303
  Torque Coefficient CQ: -0.00090022
   Power Coefficient CP: 0.00090046

     CT/sigma: 0.15412489
     CQ/sigma: -0.01497846
     CP/sigma: 0.01498250

         Thrust:    24.5252   [N]
         Torque:    -0.3451  [Nm]
    Total Power:     431.97   [W]      0.58  [hp]

------------------------######################## END OF CASE #######################------------------------
```

S76 Main Rotor File

```
    <Rotor name="Main Rotor" type="1" Id="1">
      <!--  1 = Genhel Main Rotor -->
      <Naz value="36" />
      <!--  CCW = 1 & CW = -1 -->
      <Rotation value="1" />
      <Airfoil>
        <Profile Id="0" value="C/CODE/S76/sc1013r8.dat" />
        <Profile Id="1" value="C:/CODE/S76/sc1095r8.dat" />
        <Profile Id="2" value="C:/CODE/S76/sc1095.dat" />
      </Airfoil>
      <Inflow>
        <GroundEffect status="0" height="3.048" />
        <Kappa value="1.2" Mu="0.0"/>
        <Kappa value="1.0" Mu="0.25"/>
      </Inflow>
      <Hub>
        <Position>
          <FS value="0" />
          <BL value="0" />
          <WL value="0" />
        </Position>
        <Orientation>
          <!--    Conventional Rotor Orientation is ZERO. All axes
are assumed as body axis and rotation is ZYX-->
          <X value="0.0" />
          <Y value="-10.0" />
          <Z value="0.0" />
        </Orientation>
        <ShaftTilt>
          <X value="0.0" />
          <Y value="0.0" />
          <Z value="0.0" />
        </ShaftTilt>
        <hOffset value="0.20" />
        <!-- dimensional -->
        <PhaseAngle value="0.0" />
        <Delta3 value="0.0" />
        <FlapDyn value="1" />
        <LagDyn value="0" />
        <Precone value="0" />
      </Hub>
      <Blade>
        <BladeNum value="4" />
        <Nrad value="20" />
        <Rtip value="6.70" />
        <Rcutout value="0.93" />
        <!-- dimensional -->
        <I_b value="1000.56" />
        <M_b value="250" />
        <W_b value="600" />
        <K_b value="1100" />
        <BladeProfiles>
          <Airfoil Id="0" Rpos="0.0" />
          <Airfoil Id="0" Rpos="0.92" />
          <Airfoil Id="1" Rpos="0.45" />
          <Airfoil Id="1" Rpos="0.795" />
          <Airfoil Id="2" Rpos="0.80" />
          <Airfoil Id="2" Rpos="1.0" />
        </BladeProfiles>
        <ChordDist>
          <Chord Rpos="0.0"     length="0.39744" />
          <Chord Rpos="0.73"    length="0.39744" />
          <Chord Rpos="0.84"    length="0.3938" />
          <Chord Rpos="0.935"   length="0.3938" />
          <Chord Rpos="0.955"   length="0.3712" />
          <Chord Rpos="0.985"   length="0.292" />
          <Chord Rpos="1.0"     length="0.266" />
        </ChordDist>
        <SweepDist>
          <Sweep Rpos="0.0"     angle="0" />
          <Sweep Rpos="0.954"   angle="0" />
          <Sweep Rpos="0.955"   angle="30" />
          <Sweep Rpos="1.0"     angle="30" />
        </SweepDist>
        <TwistDist>
            <Twist Rpos="0.14"  angle="2.5" />
            <Twist Rpos="0.7"   angle="0.0" />
            <Twist Rpos="1.0"   angle ="-2.6" />
        </TwistDist>
      </Blade>
      <Inputs>
        <Collective value="8.0"     active="1"  />
        <LonCyclic  value="0.0"     active="1"  />
        <LatCyclic  value="0.0"     active="1"  />
        <RPM        value="293"     active="0"  />
        <Tilt       value="0.0"     active="0"  />
        <Cant       value="0.0"     active="0"  />
      </Inputs>
      <States>
        <B0 value="2.5" />
        <!-- higher order coefficients are assumed 0 inside the code
-->
        <B1C value="0" />
        <B1S value="0" />
        <D0 value="0.0" />
        <D1C value="0.0" />
        <D1S value="0.0" />
      </States>
    </Rotor>
```