PERFORMING HELICOPTER MANEUVERS WITH OPTIMIZATION METHODS

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

BY

FATİH TOSUN

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

AUGUST 2020

Approval of the thesis:

THESIS TITLE

submitted by **FATİH TOSUN** in partial fulfillment of the requirements for the degree of **Master of Science** i**n Aerospace Engineering, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. İsmail Hakkı Tuncer
Head of the Department, **Aerospace Engineering** _____

Prof. Dr. Yavuz Yaman
Supervisor, **Aerospace Engineering, METU** _____

Examining Committee Members:

Assoc. Prof. Dr. Ercan Gürses
Aerospace Engineering, METU _____

Prof. Dr. Yavuz Yaman
Aerospace Engineering, METU _____

Asst. Prof. Dr. Mustafa Perçin
Aerospace Engineering, METU _____

Prof. Dr. Ender Ciğeroğlu
Mechanical Engineering, METU _____

Prof. Dr. Metin Uymaz Salamcı
Mechanical Engineering, Gazi University _____

Date: 28.08.2020

I hereby declare that all 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 : Fatih Tosun

Signature :

ABSTRACT

PERFORMING HELICOPTER MANEUVERS WITH OPTIMIZATION METHODS

Tosun, Fatih
Master of Science, Aerospace Engineering
Supervisor : Prof. Dr. Yavuz Yaman

August 2020, 277 pages

In order to certificate a helicopter, aviation safety agencies must know that all designed helicopter configurations can withstand all loads resulting from maneuvers defined in certification standards. In other words, the maneuvers defined in regulations must be performed for each appropriate combination of weight and center of gravity. Then, designers have to prove that the helicopter can fly safely across the entire flight spectrum. Therefore, load engineers perform all maneuvers defined in the helicopter usage spectrum in order to analyze all possible load values. In traditional methods, a trial and error approach is used to reach the maneuver. However, performing these maneuvers with a trial and error approach not only causes expensive computing but also requires more engineering effort. Additionally, it may cause some defects in the maneuvers. Therefore, this thesis study aims to achieve the desired helicopter maneuvers using optimization methods in order to reduce the calculation cost and engineering efforts and perform the maneuvers accurately. For this purpose, only selected maneuvers are performed in this thesis. In addition, various optimization methods in different configurations have been applied

to solve these maneuvers. Thus, the most useful one among them has been decided by making the necessary comparisons.

Finally, the most useful optimization method has been applied to maneuvers frequently performed by helicopters throughout its lifetime.

Keywords: Maneuver Optimization, Design Variable Vector, Constraints, Objective Function, Pilot Control Inputs

ÖZ

OPTİMİZASYON YÖNTEMLERİ İLE HELİKOPTER MANEVRALARININ GERÇEKLEŞTİRİLMESİ

Tosun, Fatih
Yüksek Lisans, Havacılık ve Uzay Mühendisliği
Tez Yöneticisi: Prof. Dr. Yavuz Yaman

Ağustos 2020, 277 sayfa

Bir helikopteri sertifiye etmek için, havacılık güvenliği kurumları, tasarlanmış tüm helikopter konfigürasyonlarının sertifikasyon standartlarında tanımlanan manevralardan kaynaklanan tüm yüklere dayanabileceğini bilmek zorundadır. Başka bir deyişle, yönetmelikte tanımlanan manevralar, her uygun ağırlık ve ağırlık merkezi kombinasyonu için gerçekleştirilmek zorundadır. Ardından, tasarımcılar helikopterin tüm uçuş spektrumunda güvenli bir şekilde uçabileceğini kanıtlamak zorundadır. Bu nedenle, yük mühendisleri tüm olası yük değerlerini analiz etmek için helikopter kullanım spektrumunda tanımlanan tüm manevraları gerçekleştirir. Geleneksel yöntemlerde, manevraya ulaşmak için deneme yanılma yaklaşımı kullanılır. Ancak, bu manevraları deneme yanılma yaklaşımı ile gerçekleştirmek sadece pahalı hesaplamaya neden olmakla kalmaz, aynı zamanda daha fazla mühendislik çabası gerektirir. Ek olarak, manevralarda bazı kusurlara neden olabilir. Bu sebeple, bu tez çalışması, hesaplama maliyetini ve mühendislik çabalarını azaltmak ve manevraları doğru bir şekilde gerçekleştirmek için optimizasyon yöntemlerini kullanarak istenen helikopter manevralarına ulaşmayı amaçlamaktadır. Bu amaçla, bu tezde sadece seçilmiş manevralar yapılmaktadır. Ayrıca bu manevraları çözmek için farklı konfigürasyonlarda çeşitli optimizasyon yöntemleri

uygulanmıştır. Böylelikle gerekli karşılaştırmalar yapılarak aralarından en faydalı olanına karar verilmiştir.

Son olarak, en kullanışlı optimizasyon yöntemi, kullanım ömrü boyunca helikopterler tarafından sıklıkla gerçekleştirilen manevralara uygulanmıştır.

Anahtar Kelimeler: Manevra Optimizasyonu, Tasarım Değişkeni Vektörü, Kısıtlamalar, Amaç Fonksiyonu, Pilot Kontrol Girdileri

To my family

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my supervisor Prof. Dr. Yavuz Yaman for giving me an opportunity to work with him, sharing his invaluable comments and experiences with me and his endless patience during my study.

I am greatly indebted to İlhan Ozan Tunçöz for his endless supports, advices, helps and tolerances throughout my academic life.

I sincerely thank to my invaluable colleagues Uğur Kalkan and Harun Tıraş for their helpful advices, kind supports, valuable discussions and their indispensable friendship.

I would also like to thank to my coworkers, Pınar Arslan Dülgar, Murat Şenipek, Mehmet Melih Atalay, Yağmur Bulut and Gürkan Sertsoy for cooperation and friendship, and helping me in all the possible ways.

I also would like to express my thanks to Derya Gürak and Fatih Mutlu Karadal for always promoting me and believing in me.

I would like to thank to my company Turkish Aerospace for supporting and encouraging me to write this thesis.

I offer my endless gratitude to my brothers Barış Usta and Mehmet Gökçay Kabataş for their inexhaustible support and invaluable friendships throughout my life.

I wholeheartedly thank to İlayda Kuru for her dedication and help in the writing of this thesis, her love, support, trust, understanding, or briefly for making every day of my life meaningful.

I would like to extend my thanks to my uncle Aydın Tosun and my aunt Halise Tosun, who never made me feel lonely during my university life. I also thank to my cousins Kaan Tosun and Burcu Tosun for their advices and friendship.

My greatest thanks go to my parents, Ayhan Tosun and Hanife Tosun for their moral support, guidance and inspiration all through my life, my brother Onur Tosun, my twin sister Esra Çınar who are always there for me. This study would never be possible without their help. I also would like to thank Çınar baby for bringing excitement and joy to our lives.

And finally, I would like to thank everyone who supported and believed me in every aspect during the thesis.

TABLE OF CONTENTS

APPENDICES

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

ABBREVIATIONS

METU	Middle East Technical University
cg	Center of Gravity
NLG	Nose Landing Gear
MLG	Main Landing Gear
MR	Main Rotor
TR	Tail Rotor
CCW	Counter Clockwise
SAS	Stability Augmentation System
CPU	Central Processing Unit

LIST OF SYMBOLS

SYMBOLS

δ_{co}	Collective cyclic input
δ_{lo}	Longitudinal cyclic input
δ_{la}	Lateral cyclic input
δ_{ap}	Anti-torque pedal input
t	Time
p	Penalty parameter
g	Constraint of the objective function
γ	Main rotor blade flapping angle
θ	Helicopter pitch angle
$\emptyset$	Helicopter roll angle
Ψ	Helicopter yaw angle
$\dot{\Psi}$	Helicopter yaw angle rate
P	The power of the helicopter supplied from engine
U	Helicopter longitudinal velocity with positive sign in forward direction
V	Helicopter lateral velocity with positive sign in rightward direction
W	Helicopter vertical velocity with positive sign in upward direction
h	Helicopter altitude
α	Step size
ε	Perturbation constant
n_z	Load factor in body z axis

CHAPTER 1

INTRODUCTION

In the design process of a helicopter, fuselage load analysis is of great importance. If the external loads acting on the outer surface of the helicopter are known, the internal loads resulting from these loads can be calculated. These calculated internal loads such as axial force, shear force, bending moment, and torsional moment are used in structural analysis to examine the helicopter's load limits.

The external loads differ depending on the type of maneuver and flight conditions. Therefore, each maneuver type and flight combination must be analyzed to be able to perform appropriate structural analysis.

1.1 Objective of the Study

According to the design purpose, a helicopter may need to fly in wide range of flight conditions with different configurations. Also, it has to perform various maneuvers. Both to design and to certify a rotorcraft, external and internal loads due to maneuvers defined in standards should be calculated. Therefore, all maneuvers defined in the helicopter usage spectrum have to be performed at each appropriate combination of weight and center of gravity. The types of maneuver expected to be analyzed are specified in the relevant standards according to the military or civilian configuration of the helicopter. To illustrate this, it can be said that CS 27 is acceptable as a civilian standard for small helicopters (with maximum weights of 3175 kg (7 000 lbs) or less and 9 or fewer passenger seats) [1] while CS 29 is acceptable as a civilian standard for larger ones [2].

The standards require that each maneuver in different conditions can be performed for all helicopter configurations. This situation needs thousands of maneuver

analysis. In the aerospace industry, one of the main duties of a load engineer is to perform the necessary maneuvers to analyze loads on the aircraft. Traditionally, the desired maneuvers are carried out by changing the pilot control inputs with a trial and error approach. In this approach, the pilot control inputs are applied to the helicopter system, then it is investigated whether the desired movement is accurately performed or not. If it is not, the inputs are modified manually until the target is achieved. However, these repetitive analyses both take a lot of time and need more computational effort. Moreover, it may cause some deficiencies in the maneuvers. Therefore, some of the optimization methods will be explored in this study to overcome these defects. Thus, the desired maneuver will be realized in the most economical way with high quality.

1.2 Layout of Thesis

Chapter 1 is the introduction chapter. In this chapter, the objective of the study is introduced.

Chapter 2 is reserved for the literature review. This chapter starts with what optimization is and its elements. Then, it continues with its use in aviation and helicopter maneuvering. Finally, it ends with what the comprehensive analysis tool is and the explanation of FLIGHTLAB as the tool for analysis.

Chapter 3 explains some of the gradient-based optimization techniques. The chapter also contains the formulation of each method with their assumptions.

Chapter 4 defines the axis system and the mathematical model of the rotorcraft used in this thesis.

Chapter 5 describes the optimization methodology of helicopter maneuvers. Also, the mathematical formulation of maneuver optimization is described in this section.

Chapter 6 contains the maneuvering optimization results of various optimization methods in different configurations. In addition, the comparison of each analysis is

described in this chapter. Thus, the most useful optimization method is determined with its best configuration.

Chapter 7 discusses the development process of the optimization code and examines the importance of time step and initial condition choices.

Chapter 8 shows that the decided optimization method is also useful for other maneuvers.

Chapter 9 summarize the results obtained in this thesis and presents the benefits of the developed optimization method.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter gives the literature view of this thesis. In this section, optimization and its elements are explained with their mathematical representations. It also mentions the use of optimization in aviation and helicopter maneuvers in the literature. Finally, the comprehensive analysis tools and one of them, FLIGHTLAB software, are also described.

2.2 What is Optimization?

Optimization is a process of search in order to reach the best objective among some set of alternatives by satisfying the required restrictions [3]. In nature, animals and plants use optimization instinctively to achieve their basic vital requirements faster and with minimum energy. Since people are guided and influenced by their natural environment, they also instinctively apply optimization that occurs in nature all over in their lives. Several examples can be found in daily experiences. For example, people try to find alternative routes to minimize commuting time and fuel consumption or do market research to get cheaper products of the same quality. In today's world where natural resources are exhausted, the importance of optimization is increasing day by day. For this reason, optimization should continue to find its place in various life areas, especially in scientific fields.

2.3 Elements of Optimization Problems

Optimization problems need three fundamental elements to be able to construct their mathematical models. These elements are design variables, objective function and constraints on the problem formulation [3]. Design variables and objective function may differ depending on the type and needs of the defined problem. Constraints are usually identified according to physical and material limitations.

2.3.1 Design Variables

Design variables in an optimization problem are the inputs of the optimization process and directly affect the objective of the problem. In an aircraft maneuver problem, pilot control inputs are defined as design variables because aircraft respond according to the control inputs applied. Two types of design variables exist; continuous and discrete design variables. Continuous design variables can take any value within the predefined interval. On the other hand, discrete design variables can only take a fixed number of distinct values in the range.

2.3.2 Objective Function

An objective function is a function whose inputs are design variables and whose outputs show the quality of the solution. In other words, the objective function is a mathematical equation that shows how close the design variables get to the optimum solution. Objective functions target to reach maximum value or minimum value according to the type of the optimization problem. If a problem wants to find maximum speed of the aircraft in limitation of available engine power and maneuverability of the airplane, such problems are called the maximization problems. In addition, if a problem target to find the lightest aircraft configuration to be able to perform the desired mission, such problems are called minimization problems.

2.3.3 Constraints

In an optimization problem, the best objective is tried to be achieved under some restrictions. These restrictions are named as constraints. The constraints can be classified as equality, inequality and geometric constraints. Constraints have to be satisfied to be able to consider the solution is valid and acceptable. If constraints are not be met, the results cannot be accepted as the solution of the optimization problem.

2.4 Mathematical Modelling of Optimization

The main purpose of an optimization problem is to reach the optimum solution of many combinations of design variables defined in a mathematical problem. The general form of a typical mathematical model for an optimization problem can be defined as follows:

Find the values of design variables $\boldsymbol{X}^T = [x_1, x_2, \dots, x_n]$ to minimize or maximize objective function $f(\boldsymbol{X}^T)$

subject to

$$g_i(x_1, x_2, \dots, x_n) \leq 0 \qquad i = 1,2, \dots, m$$

$$h_j(x_1, x_2, \dots, x_n) = 0 \qquad j = 1,2, \dots, l$$

$$x_k^l \leq x_k \leq x_k^u \qquad k = 1,2, \dots, d$$

where, n is the number of total design variables. Constraints are inequality constraints g_i, equality constraints h_j and geometric constraints with lower x_k^l and upper x_k^u bounds.

2.5 Optimization in Aerospace

Optimization in the aerospace industry focuses on solving some typical aviation problems using mathematical control techniques. In history, various different mathematical control techniques have been developed to obtain solutions. The application field of these techniques in the aerospace industry is quite wide. Like structural mechanics control problems, propulsion systems control problems, helicopter rotor structure, flight mechanics control problems, space flight control problems, vibration suppression, passive control, active dynamic control, smart materials for control applications, thermal efficiency, acoustic transmission optimization, jet engine control, transmission control, combustion efficiency, aeroelastic coupling problem, re-entry orbit problem, transfer orbit problem, etc. Therefore, various different techniques have been developed as an optimization approach for the solution of these kind problems. For example, the Explorer-Settler Optimization algorithm developed by combining the advantageous aspects of Particle Swarm Optimization and Nelder-Mead Optimization algorithms to minimize rotor vibrations by optimizing the rotor structure that requires a highly accurate solution that takes time for vibration predictions [4]. The other example, to study the re-entry of the space shuttle to the atmosphere, the geometric optimal control problem developed based on various concepts of different geometries is used. [5]. The other one, the homotopy method, which consists of constantly simplifying a complex problem to a number of simpler parameterized problems is employed for the transfer orbit problems [6]. In addition, the optimization has been used in very important applications in the helicopter design. Nowadays, the optimization techniques are used in many areas of helicopter manufacturing such as aerodynamic design, geometric design, helicopter performance, helicopter maneuvering, control system design, etc.

2.6 Helicopter Maneuver Control System

A helicopter is a rotary-wing aircraft capable of performing hover, vertical takeoff and landing besides of low-speed flight. On the other hand, it also has unstable and heavy coupling characteristics. Since the controllability of a helicopter is not easy, many undesirable situations can arise during the flight. Maneuverability is an important factor that determines whether a helicopter can successfully complete certain flight missions.

Helicopter maneuvers are performed by using pilot control inputs which are applied through cyclic and collective inputs and anti-torque pedals as shown in Figure 2.1.

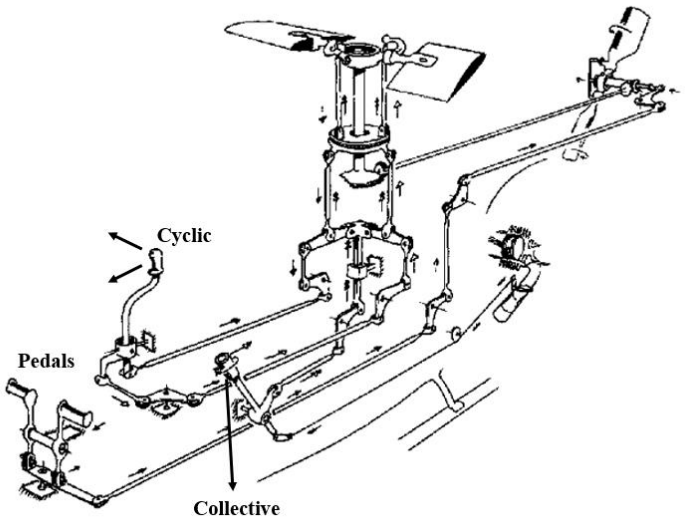


Figure 2.1: Cyclic, Collective and Anti-Torque Pedal Control Inputs [7]

The cyclic has a direct effect on the pitch of the rotor blades through each rotation of the main rotor system to produce unequal thrust. The rotor disc, which is tilted in a particular direction due to the effect of the cyclic, produces thrust in that direction. In other words, when the cyclic is pushed forward direction (longitudinal cyclic), the rotor disk tilt to forward. Thus, the rotor thrust is produced in the forward direction. Similarly, if the pilot pulls the cyclic laterally (lateral cyclic), the rotor disc will lean in this direction. Thus, the rotor thrust will be produced in that direction. In addition,

the collective cyclic causes change on the pitch angle of all main rotor blades at the same time independently of their position. This means that when collective cyclic position is changed, the total thrust on the main rotor blades will increase or decrease. Therefore, the altitude and airspeed of the helicopter will be affected. For example, if the pilot pulls the collective cyclic towards himself or herself, the total thrust on the main rotor blades will increase. If the collective cyclic is pushed, the total thrust will decrease. In addition, the pedal input has an effect on the pitch angles of the tail rotor blades. It causes increasing or decreasing tail rotor thrust, so the helicopter can perform a nose to yaw in a particular direction. If the pilot presses the left pedal, the helicopter turns to the left. Similarly, if the pedal presses the right pedal, the helicopter turns to the right.

Although each different combination of pilot control inputs results in different helicopter reactions, a reaction can also be obtained with different combinations. This is caused by the high coupling between roll, pitch, and yaw motions of a rotorcraft. In other words, helicopters show distinctive reactions for different combinations of control inputs. Therefore, pilot control inputs should be satisfied together to be able to perform the desired maneuver.

2.7 Optimization in Helicopter Maneuvering

In the aerospace industry, aerodynamic and inertial loads caused during the maneuvers defined in regulations must be calculated to design and to certify an aircraft. For large rotorcraft proof of compliance, all requirements in the flight subsection of Certification Specifications for small and large rotorcraft must be met for each proper combination of weight and center of gravity [1][2]. Therefore, the maneuvers must be performed accurately by using a mathematical model of a helicopter for certification.

Traditionally, the desired maneuvers are carried out by trial and error approach. The basis of this approach is based on manipulating pilot control inputs manually

according to obtained results. The pilot control inputs are changed manually, then, obtained results for these inputs are examined if the maneuver can be performed as in the desired form or not. If not, this process is repeated until the requested maneuvering is achieved. However, this manual approach brings some imperfections on maneuvering as:

- Uncertainty of the direction and the rate at which the pilot inputs should be applied
- Waste of time
- Cost of re-analysis
- Computational effort
- Difficulty in obtaining accurate results to satisfy maneuver conditions under helicopter limitations

To overcome these deficiencies, an automated methodology called inverse simulation method is needed to obtain the pilot control inputs that provide the intended maneuver. At that point, the inverse simulation approach is used to estimate the required time history of control inputs to achieve a particular system outputs. In helicopter flight dynamic, inverse simulation answers the question of which pilot control inputs are needed to perform the aimed maneuver as desired under the piloting workload, maneuverability and performance limits of the helicopter [8]. This approach uses the pilot controllers as design variables to satisfy objective function. In this approach, one of the most important parameters is the objective function. The objective function refers to the conditions and constraints of maneuver flight. Hence, the objective function should be structured very carefully in order to be able to perform the desired maneuver accurately.

After seeing that the dynamic flight simulation approach is useful for obtaining control in fixed-wing aircraft, many attempts have been made to use this method in helicopter applications [9][10]. In the literature, there are different approaches to the use of inverse simulation for helicopter maneuvering. These approaches can be

divided into three main categories which are numerical differentiation approach, numerical integration approach and global optimization methods. Robert Thomas, who is one of the first to apply inverse simulation methodology, was concerned with the problem of representing a gust disturbance or control manipulation [11]. Thomas explained the application of Heaviside's operational method in improving aircraft dynamic problems using the numerical differentiation approach. Also, particular graphical methods and logarithmic formulas that reduce the amount of calculation are explained in his study. Other researchers used the inverse simulation methodology for helicopter maneuvers were Thomson and Bradley [12]. They also used a numerical differentiation approach to develop a nonlinear six degrees of freedom mathematical model "HELINV" for rotorcraft applications. They used the numerical differentiation of situations at each time step to calculate the time derivatives of the situations. Thus, the governing differential equations could be transformed into algebraic equations for the solution of the control angles. Moreover, this model provides achieving the predefined trajectory path for a variety of helicopter configurations in a wide range of helicopter maneuvers [12]. Although the approach depends on the model and is time step size, it has emphasized the usefulness of the inverse simulation method in helicopter flight dynamics.

Instead of numerical differentiation based inverse, Hess et al. [13] preferred the numerical integration based inverse simulation method. They also aimed to reach control inputs that necessary to follow a predefined trajectory. This method benefits from the combination of the numerical integration of the equations of motion and the Newton Raphson method to reach desired states. Even though this approach is not as fast as the numerical differentiation based approach, it is not affected by the rotor dynamic model and allows its development without the need for any changes in the solution procedure [13]. Other pioneers for numerical integration based inverse simulation methodology were Raghavendra and de Abhishek [14]. They derived the methodology to estimate the pilot control inputs and shaft orientation angles in the widest range of maneuvering that can be performed by helicopters. In addition, they

verified and validated their analysis results with available flight test and simulated results for helicopter data in the literature [14].

Contrary to those approaches, Celi [15] used the global optimization based-inverse simulation method to operate a whole family of possible trajectories. In other words, his derived technique is applicable for all familiar trajectories, not only a single one. He aimed to obtain the best trajectory and corresponding pilot control inputs for helicopter slalom maneuvers in the required performance criteria. The trajectory is tried to be reached by using the gradient-based Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm.

Similar to Celi's work, Guglieri and Mariano [16] also benefited from global optimization base inverse simulation methodology for the helicopter slalom maneuver optimization by tracking of the predefined trajectory. Yet, they also examined the pirouette maneuver and used a genetic algorithm, unlike Celi. Moreover, they employed the Genetic Algorithm developed by D.L.Caroll [17] into their own algorithm to follow the predicted trajectories with reasonable control inputs. They used Genetic Algorithm to start optimization, then, switch to the Quasi-Newton method to reach the desired trajectory.

Global optimization can also be achieved by using stochastic algorithms instead of gradient-based methods. Unfortunately, stochastic algorithms are not as fast as gradient-based ones due to their lack of direction to follow up [18]. But, they are more likely to achieve a global minimum solution because this algorithm also combines possible random design values. Furthermore, they do not need any initial design variable at the beginning of the optimization. Unlike this method, gradient-based ones need the starting point as an initial design variable. Therefore, this method can stick at a local minimum point according to the taken initial reference point.

In this thesis, the main purpose is to perform the helicopter maneuver in the shortest time and in the most efficient way in terms of the small objective function. The global

optimum solution is not sought indeed. Therefore, gradient-based optimization methods were preferred instead of stochastic or a combination of stochastic and gradient-based methods. However, the initial design variables have to be estimated accordingly in order to avoid losing speed advantage. Therefore in this study, the initial point estimation will also be examined in detail.

2.8 Rotorcraft Comprehensive Analysis Programs

There are several computer programs to understand aeromechanical characteristic of the rotorcraft. These programs perform trim and transient analysis using the mathematical model of the rotor vehicle. In the mathematical model, airframe and rotor systems may be created with rigid or elastic equations of motion. The described computer programs are called as comprehensive analysis programs. These programs should combine the most advanced geometric models, structure, dynamics and existing aerodynamics [19]. In addition, comprehensive analysis programs can compute aeroelastic stability, flight dynamic, structural load, vibration, rotorcraft performance and trim [19].

The comprehensive analysis tools play a very important role during the rotorcraft design and performance process by computing rotor performance and maneuver loads. Also, these programs analyze all the rotorcraft during the maneuver. The aeromechanics of a stand-alone rotor under a stable operating condition is complex, yet the ability to analyze multiple rotors and maneuvers is crucial [19].

A comprehensive analysis has its origins in the 1960s when digital computers became available in engineering applications [19]. Comprehensive analyses and developers that have been improved throughout history since the 1960s are given in Figure 2.2 [19].

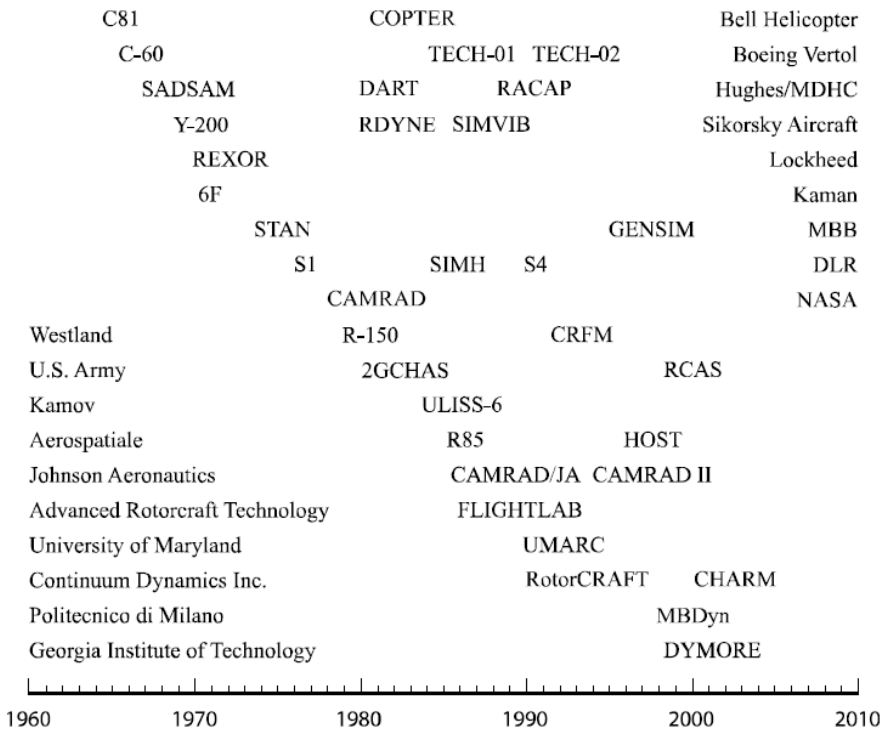


Figure 2.2: Comprehensive Analysis Tools throughout the History [19]

The comprehensive analysis tools that started with the helicopter simulation program C81 has added many members to its family throughout history. In this thesis, FLIGHTLAB software is used as a comprehensive analysis tool for helicopter maneuvers.

2.8.1 FLIGHTLAB

Advanced Rotorcraft Technology, Inc. (ART) has developed FLIGHTLAB as real-time simulation of the blade element helicopter [20][21][22]. In early 1985, ART reconfigured the GENHEL blade element model which is used in NASA [23] to show a real-time simulation potential using parallel processing on affordable computers [19]. These experiences provided ART with the groundwork to develop a generic, modular, reconfigurable analysis for real-time simulation in 1986 [19]. FLIGHTLAB emerged as commercial software in 1990 and this version was

supplied the quick prototyping ambiance to model and analyze a rotorcraft [19]. During the development process, FLIGHTLAB gave more attention to handling qualities analysis and real-time simulation [19]. FLIGHTLAB was not restricted with these topics, it was also acquired various abilities in this process. In 1995, FLIGHLAB possessed the vortex wake aerodynamic, finite element structural dynamics and a non-linear beam model [24].

FLIGHTLAB simulation library supplies a generic object-oriented environment with modal components [20]. The components include linear and non-linear control blocks, engine and drive train components, finite state dynamic inflow, prescribed and free vortex wake aerodynamics, aerodynamic interference, rigid masses, non-linear springs and dampers etc.[23]. Hence, all rotorcraft configurations can be structured with this object-oriented modeling methodology.

FLIGHTLAB simulation analysis consists of two main parts which are trim analysis and transient analysis. Trim analysis can be performed as both accelerated and unaccelerated flight. In the accelerated trim analysis, the flight maneuver is solved at predefined linear and angular accelerations or at a specific parameter value. This solution also provides other helicopter characteristic parameters such as Euler's angle, flight velocity etc. In the unaccelerated trim analysis, the flight maneuver is solved at zero linear and angular acceleration conditions. Hence, the rotorcraft is in either steady condition or constant speed flight condition. The transient analysis uses the trim flight parameters as a starting point. In the transient analysis process, pilot control inputs are applied to perform the desired maneuver. Thus, all parameters of a rotorcraft concerning time are obtained by starting trim analysis and continuing with transient analysis.

CHAPTER 3

REVIEW OF SOME OPTIMIZATION METHODS

3.1 Introduction

This section describes some useful optimization techniques for helicopter maneuvers. Also, their prerequisites, advantages and deficiencies are mentioned.

3.2 Optimization Techniques

The usefulness of optimization techniques often depends on the optimization problem. Therefore, an appropriate optimization algorithm should be preferred according to the problem type.

The approach of optimization algorithms is similar to the search for water in a rugged desert. Imagine a Bedouin man consumes all his water and needs water to survive. One of the conjectures is that if the Bedouin does not know where to find water, he tries to find water by pure random search. However, this is not an effective way as seen obviously. In another conjecture, if the man knows that he can reach the water at the lowest place in the desert, he goes directly to the lowest point of his region. In general, the optimization problems are structured between these two hypotheses. In an optimization search, even though the algorithm is not blind, it still does not know where to look. It takes time to look at all the possible points. Therefore, the algorithm collects useful hints when searching for possible points randomly. The algorithm tends to search for a more reasonable place in each iteration step using these tips. Such random research processes occur on the basis of modern search algorithms. If there is no time constraint and inaccessible region, it is possible to reach the required

water theoretically. Therefore, while the optimization algorithms are being developed, they are refined a little bit further to eliminate these kinds of deficiencies.

In literature, there are two different classes of optimization algorithms which are deterministic algorithms and stochastic algorithms. Deterministic algorithms take the initial condition as a reference point to start optimization. During the optimization process, they do not use a random input selection procedure. Therefore, they always get the same solutions for the same initial points. To illustrate, if Bedouin's direct orientation to the lowest point of his region is accepted as the deterministic algorithm, he always follows the same path when he starts from the same starting point. Unlike the deterministic algorithms, the stochastic algorithms are always based on the randomization process. Because of this randomness, the used numbers and followed solution paths may be different for each optimization runs at different times. On the other hand, the final optimization solution may not be very different from each other. For example, if Bedouin's pure random research is represented as a stochastic algorithm, he may behave differently in the optimization process to reach water at different times.

In history, gradient-based deterministic algorithms have traditionally been used to find the optimum solution [25][26][27]. These algorithms tend to be fast, but they require a logical initial guess. Moreover, they need the calculation of derivatives and may get stuck in local minima.

Stochastic algorithms have gained popularity in the past two decades. In these years, several stochastic algorithms have been developed and applied in many areas [28][29][30]. Such algorithms have been developed with inspiration from nature. Moreover, they are more likely to achieve a global minimum solution and do not get stuck at the local minimum point. However, even though they do not require any gradient computations, they are still slower than gradient-based optimization algorithms.

In recent years, there are many relevant studies on the comparison of deterministic algorithms and stochastic algorithms [31][32][33]. In these studies, the pioneers select several sample algorithms from these methods and compare them to each other in solving a particular problem. Sometimes, they also compare the combination of both as well as comparing these two algorithms. The general idea obtained from these studies states that deterministic methods are more effective than stochastic methods in terms of cost and time. In addition, although the combination obtained from deterministic and stochastic methods is more effective than stochastic methods, it is not as effective as deterministic methods in terms of cost and time.

The main purpose of this thesis is to realize the helicopter maneuver in the shortest time and in the most efficient way. It is not aimed to reach global minimum value. Therefore, gradient-based optimization methods are preferred instead of stochastic or a combination of the stochastic and gradient-based methods. However, the first value should be chosen logically in order to obtain the most efficient solution from these methods. The necessary examinations for the initial value estimation is also a subject of this thesis and will be explained in the following chapters.

3.2.1 Gradient-Based Methods

Gradient-based methods are iterative methods where the gradient information of the objective function is widely used throughout iteration [34]. To minimize the objective function *f(x)*, these methods are based on the following description:

$$x^{(k+1)} = x^{(k)} + \alpha^{(k)}\, g\left(\nabla f, x^{(k)}\right) \tag{1}$$

where α represents the step size and can change with each iteration. Moreover, $x^{(k)}$ is the current design point of the function and $g\left(\nabla f, x^{(k)}\right)$ represents the function of gradient ∇f. There are various gradient-based methods which use the different forms of $g\left(\nabla f, x^{(k)}\right)$. In this thesis, some of these gradient-based methods are examined for helicopter maneuver optimization.

3.2.1.1 Newton's Method

The Newton method is known to be one of the most popular iterative methods used to find zeros of a nonlinear objective function [35]. It is based on finding the root of the objective function *f(x)* while fulfilling the necessary restrictions. Therefore, it uses the first derivative of the objective function *f'(x)* in order to achieve the roots of the function. Due to continuous differentiability of the function, Taylor expansion of a function *f(x)* about point $x^{(k)}$ can be written for $\Delta x = x^{(k+1)} - x^{(k)}$as

$$f\left(x^{(k+1)}\right) = f\left(x^{(k)}\right) + \nabla f\left(x^{(k)}\right)\Delta x + \frac{\Delta x^T \ \nabla^2 f\left(x^{(k)}\right)\Delta x}{2!} + \cdots \qquad (2)$$

Third term of the expansion is not included in following calculations because it has a quadratic form. The value of Δx which is the root of the following linear equation is also solution of function *f(x)*.

$$\nabla f\left(x^{(k)}\right) + \nabla^2 f\left(x^{(k)}\right)\Delta x = 0 \qquad (3)$$

The expanded form of the equation is

$$x^{(k+1)} = x^{(k)} - \left[H\left(x^{(k)}\right)\right]^{-1} \nabla f\left(x^{(k)}\right) \qquad (4)$$

where the Hessian matrix $H\left(x^{(k)}\right)$ is defined as

$$H\left(x^{(k)}\right) = \begin{bmatrix} \frac{\partial^2 f}{\partial^2 x_1^2} & \cdots & \frac{\partial^2 f}{\partial^2 x_1 x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial^2 x_n x_1} & \cdots & \frac{\partial^2 f}{\partial^2 x_n^2} \end{bmatrix} \qquad (5)$$

The Newton's Method is updated by including step size α to increase convergence speed of the algorithm as:

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} \left[H\left(x^{(k)}\right)\right]^{-1} \nabla f\left(x^{(k)}\right) \qquad (6)$$

As known, the calculation of the Hessian matrix needs second derivatives, which can lead to time loss and increased computational efforts. Therefore, identity matrix (***I***)

is a good alternative to approximate the Hessian matrix by using $\left[H\left(x^{(k)}\right)\right]^{-1} = I\left(x^{(k)}\right)$, so the method is modified as:

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} \left[I\left(x^{(k)}\right)\right]^{-1} \nabla f\left(x^{(k)}\right) \tag{7}$$

where $\alpha \in (0,1)$ is a step size. Actually, this formulation represents the steepest descent method.

3.2.1.2 Steepest Descent Method

The steepest descent method, first suggested by Cauchy in 1847 [36], is one of the simplest and best-known methods used throughout history. Moreover, the method purposes to reach a minimum of an objective function [37]. Suppose that $x^{(k)}$ is the current design point of the objective function $f(x^{(k)})$. Then, it is aimed to achieve a better point $x^{(k+1)}$ in each step. Therefore, a better point is sought by moving along the search direction. In the Steepest Descent Method, this search direction is obtained by negative gradient as $-\nabla f(x^{(k)})$. Hence, the formulation is:

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} \nabla f\left(x^{(k)}\right) \tag{8}$$

where α represents the step size of the method. This method needs a line search algorithm to calculate the step size.

The steepest descent method has a well-known convergence theory [38][39]. In addition, the applicability of the algorithm is not complicated for minimization problems. However, the method converges to the optimum solution very slowly even for slightly nonlinear problems. Therefore, various attempts have been made throughout history to achieve the solution of a nonlinear equation system [40][41][42]. Nowadays, quasi-Newton methods are preferred to overcome this deficiency [43] [44][45]. In this thesis study, some of these methods are preferred and examined.

3.2.2 Quasi-Newton Methods

Quasi-Newton methods are the most effective and practical methods which are used for nonlinear systems of equations. These methods are preferred in cases where Hessian matrix calculation causes increased cost, computational difficulty or time loss. Therefore, they aim to approach the Hessian matrix with the best accuracy. The iterative formula of the method is

$$x^{(k+1)} = x^{(k)} + \alpha^{(k)} d^{(k)} \text{ where } d^{(k)} = -\left[\boldsymbol{B}^{(k)}\right]^{-1} \nabla f\left(x^{(k)}\right) \quad (9)$$

where $\boldsymbol{B}^{(k)} = B\left(x^{(k)}\right)$, positive definite matrix, is updated at each iteration to approximate the Hessian matrix of the objective function $f(x)$. The real question is here how to calculate this? You can start with identity matrix as $\boldsymbol{B}^{(1)} = I_n$, but then how $\boldsymbol{B}^{(k)}$ should be updated at each iteration? To answer these questions, a Taylor expansion is applied on the gradient at current point $x^{(k)}$, so the gradient and the Hessian are such that

$$\nabla f\left(x^{(k+1)}\right) = \nabla f\left(x^{(k)}\right) + \nabla^2 f\left(x^{(k)}\right) \Delta x + \epsilon \Delta x \quad (10)$$

If f is a well-defined quadratic function up to second degree:

$$\nabla f\left(x^{(k+1)}\right) - \nabla f\left(x^{(k)}\right) \approx \nabla^2 f\left(x^{(k)}\right) \Delta x \quad (11)$$

which leads to the quasi-Newton relation. Moreover, the approximate Hessian matrix $\boldsymbol{B}^{(k+1)}$supports this relation if

$$q^{(k)} \equiv \nabla f\left(x^{(k+1)}\right) - \nabla f\left(x^{(k)}\right) \approx \boldsymbol{B}^{(k)} p^{(k)} \quad (12)$$

where $p^{(k)} = x^{(k+1)} - x^{(k)}$. Since the main idea behind the quasi-Newton relation is based on this equation, it is defined as *quasi-Newton equation* or *secant equation.*

After this point, there remains the answer to the question of how to update $B\left(x^{(k)}\right)$. Therefore, many quasi-Newton methods have been developed to be able to answer this question. In this thesis study, Broyden's Method, Symmetric Rank-One (SR1)

method, Davidon-Fletcher-Powell (DFP) method and Broyden-Fletcher-Goldfarb-Shanno (BFGS) method are executed as a quasi-Newton method. These methods are used to approximate Hessian matrix during helicopter maneuver optimization.

3.2.2.1 Broyden's Method

Broyden's method, developed by Broyden in 1965 [46], is one of the quasi-Newton methods to solve the nonlinear optimization problems. Furthermore, this method is applicable for the solution of nonlinear equation systems. The method supplies the approximate Hessian matrix $\left(\boldsymbol{B}^{(k+1)} \cong \nabla^2 f\left(x^{(k+1)}\right)\right)$. It is denoted with $\boldsymbol{B}^{(k+1)}$. This method is based on two assumptions as:

i. Secant equation must be met as
$\boldsymbol{B}^{(k)} p^{(k)} = q^{(k)}$ where $p^{(k)} = x^{(k+1)} - x^{(k)}$ and $q^{(k)} = \nabla f\left(x^{(k+1)}\right) - \nabla f\left(x^{(k)}\right)$

ii. No change condition: Since it is not known the change of the function along the direction of $q^{(k)}$, it is not possible to make a change in that section $\left(\boldsymbol{B}^{(k+1)} - \boldsymbol{B}^{(k)}\right) q^{(k)} = 0$, means that:
$\boldsymbol{B}^{(k+1)} q^{(k)} = \boldsymbol{B}^{(k)} q^{(k)}$ whenever $\left(p^{(k)}\right)^T q^{(k)} = 0$

Hence, Broyden developed the Hessian approximate method according to these assumptions as:

$$\boldsymbol{B}^{(k+1)} = \boldsymbol{B}^{(k)} + \frac{q^{(k)} - \boldsymbol{B}^{(k)} p^{(k)}}{(p^{(k)})^T p^{(k)}} \left(p^{(k)}\right)^T \tag{13}$$

The main advantage of this method is to obtain the approximate Hessian matrix with less function calculation because there is no need to evaluate the partial derivatives. Thus, evaluation number deceases from n^2+n to n for each iteration [47].

3.2.2.2 Symmetric Rank-One (SR1) Method

Symmetric Rank-One (SR1) method is one of the most popular quasi-Newton methods for the solution of nonlinear optimization problems. In history, many studies have been done using this method [48][49][50][51]. As with other quasi-Newton methods, the symmetric rank-one method aims to approximate the Hessian matrix $\boldsymbol{B}^{(k+1)}$. It is known that secant equation is

$$\boldsymbol{B}^{(k)}p^{(k)} = q^{(k)} \text{ where } p^{(k)} = x^{(k+1)} - x^{(k)} \text{ and } q^{(k)} = \nabla f\left(x^{(k+1)}\right) - \nabla f\left(x^{(k)}\right)$$

If initial $\boldsymbol{B}^{(1)}$ is symmetric and positive definite, SR1 method can compute the approximate Hessian matrix $\boldsymbol{B}^{(k+1)}$. For this approximation, the method uses the approximate Hessian matrix $\boldsymbol{B}^{(k)}$ at k^{th} iteration and uses the function gradient of these consecutive iterations. Finally, the Hessian approximation of SR1 method is formulated as

$$\boldsymbol{B}^{(k+1)} = \boldsymbol{B}^{(k)} + \frac{\left(q^{(k)} - \boldsymbol{B}^{(k)}\, p^{(k)}\right)\left(q^{(k)} - \boldsymbol{B}^{(k)}\, p^{(k)}\right)^{T}}{(q^{(k)} - \boldsymbol{B}^{(k)}\, p^{(k)})^{T}\, p^{(k)}} \qquad (14)$$

Approximation of Hessian matrix for next iteration is not a complicated procedure by using SR1 method since it feeds from the current approximated matrix. Thus, the method decreases the computational effort and time.

3.2.2.3 Davidon-Fletcher-Powell (DFP) Method

Davidon-Fletcher-Powell (DFP) method is one of the iterative quasi-Newton methods. The method was first suggested by Davidon in 1959 [52]. Then, Fletcher and Powell improved and finalized the method in 1963 [53]. The method is also named as the variable metric method. It is known that the secant equation is

$$\boldsymbol{B}^{(k)}p^{(k)} = q^{(k)} \text{ where } p^{(k)} = x^{(k+1)} - x^{(k)} \text{ and } q^{(k)} = \nabla f\left(x^{(k+1)}\right) - \nabla f\left(x^{(k)}\right)$$

This method approximates the Hessian matrix $\boldsymbol{B}^{(k)}$ at each iteration. Suppose that *f* is quadratic and differentiable function and $\boldsymbol{B}^{(1)}$ is the initial symmetric and positive definite matrix. Finally, the Hessian approximation of DFP method is formulated as

$$\boldsymbol{B}^{(k+1)} = \left(I - \frac{q^{(k)}\left(p^{(k)}\right)^T}{(q^{(k)})^T\, p^{(k)}}\right)\boldsymbol{B}^{(k)}\left(I - \frac{p^{(k)}\left(q^{(k)}\right)^T}{(q^{(k)})^T\, p^{(k)}}\right) + \frac{q^{(k)}\left(q^{(k)}\right)^T}{(q^{(k)})^T\, p^{(k)}} \qquad (15)$$

DFP algorithm is mathematically very similar to Broyden-Fletcher-Goldfarb-Shanno (BFGS) method. When it is discovered, it was of great importance in the solution of nonlinear[29] optimization problems.

3.2.2.4 Broyden-Fletcher-Goldfarb-Shanno (BFGS) Method

Broyden-Fletcher-Goldfarb-Shanno (BFGS) method is one of the quasi-Newton methods commonly used for the solution of nonlinear objective function *f*. As other quasi-Newton methods described above, this method also computes approximate Hessian matrix $\boldsymbol{B}$ instead of the exact Hessian matrix.

The BFGS method uses the current approximate Hessian matrix to estimate the approximate Hessian matrix in the next iteration. In other words, the BFGS method updates the approximate Hessian matrix $\boldsymbol{B}^{(k)}$ at k^{th} iteration to $\boldsymbol{B}^{(k+1)}$ at $(k+1)^{th}$ iteration as:

$$\boldsymbol{B}^{(k+1)} = \boldsymbol{B}^{(k)} + \frac{q^{(k)}\left(q^{(k)}\right)^T}{(q^{(k)})^T\, p^{(k)}} - \frac{\boldsymbol{B}^{(k)}\, p^{(k)}\left(p^{(k)}\right)^T \boldsymbol{B}^{(k)}}{(p^{(k)})^T\, \boldsymbol{B}^{(k)}\, p^{(k)}} \qquad (16)$$

If the first approximate Hessian matrix $\boldsymbol{B}^{(1)}$ is chosen as positive definite, all consecutive approximate Hessian matrices will be positive definite. Therefore, identity matrix or some multiple of identity matrix can be taken as initial approximate Hessian matrix.

3.2.3 Numerical Differentiation Methods

The general formula of the quasi-Newton methods is defined in Section 3.2.2 as

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} \left[\boldsymbol{B}^{(k)}\right]^{-1} \nabla f\left(x^{(k)}\right) \tag{17}$$

As it is seen from the formulation that the optimization algorithms need the derivative of the objective function ∇f. Therefore, Taylor expansion of a function f about point x is used as

$$f(x+\varepsilon) = f(x) + \varepsilon\, \nabla f(x) + \varepsilon^2 \frac{\nabla^2 f(x)}{2!} + \cdots \tag{18}$$

where the perturbation constant is represented by $\boldsymbol{\varepsilon}$. By rearranging the equation

$$\nabla f(x) = \frac{f(x+\varepsilon) - f(x)}{\varepsilon} - \varepsilon \frac{\nabla^2 f(x)}{2!} + \cdots \tag{19}$$

which shows that the approximation error is proportional to the first power of ε, so this is a *"first order approximation"*.

If the value of ε is positive number, then

$$\nabla f(x) = \frac{f(x+\varepsilon) - f(x)}{\varepsilon} \tag{20}$$

which is called *the forward difference approximation* of $\nabla f(x)$.

If the value of ε is negative number, then

$$\nabla f(x) = \frac{f(x) - f(x-\varepsilon)}{\varepsilon} \tag{21}$$

which is called *the backward difference approximation* of $\nabla f(x)$.

If the two Taylor series expansions are subtracted,

$$f(x+\varepsilon) = f(x) + \varepsilon\, \nabla f(x) + \varepsilon^2 \frac{\nabla^2 f(x)}{2!} + \varepsilon^3 \frac{\nabla^3 f(x)}{3!} + \cdots$$

$$f(x-\varepsilon) = f(x) - \varepsilon\,\nabla f(x) + \varepsilon^2 \frac{\nabla^2 f(x)}{2!} - \varepsilon^3 \frac{\nabla^3 f(x)}{3!} + \cdots$$

it is obtained as

$$f(x+\varepsilon) - f(x-\varepsilon) = 2\varepsilon\,\nabla f(x) + 2\varepsilon^3 \frac{\nabla^3 f(x)}{3!} + \cdots$$

By rearranging the equation

$$\nabla f(x) = \frac{f(x+\varepsilon) - f(x-\varepsilon)}{2\,\varepsilon} - \varepsilon^2 \frac{\nabla^3 f(x)}{3!} + \cdots \qquad (22)$$

which shows that the approximation error is proportional to the second power of ε, so this is a *"second order approximation".* Finally, the formulation of

$$\nabla f(x) = \frac{f(x+\varepsilon) - f(x-\varepsilon)}{2\,\varepsilon} \qquad (23)$$

which is called *the central difference approximation* of $\nabla f(x)$.

In this thesis study, these three different approximations are used in order to calculate the derivative of the objective function. They are applied for each optimization method and the most useful one is chosen as a derivative computation approach.

3.3 Conclusion

In this chapter, the gradient-based optimization methods, stochastic and combination of the stochastic and gradient-based methods are compared with each other in terms of advantages and disadvantages. At the end of these comparisons, it is decided that it is appropriate to use the gradient-based optimization methods in this study because this study aims to realize the helicopter maneuver in the shortest time and most efficient way, not to go to the global minimum. Afterwards, the Newton's method, one of the most used gradient-based methods, was examined. It is seen that this method needs the calculation of the Hessian matrix. Since the calculation will take a

lot of time, some alternative methods that obtain this matrix by spending less time and effort are examined. Moreover, forward difference approximation, backward difference approximation and central difference approximation are described to be able to compute the gradient of the objective function during the optimization process.

Finally, it is decided to obtain the most useful optimization configuration by using these selected optimization methods and numerical differentiation algorithms.

CHAPTER 4

AXIS SYSTEM USED IN THE STUDY AND THE DEVELOPED MATHEMATICAL MODEL

4.1 Introduction

All axis systems used in this thesis are explained in this section. Moreover, the structural and aerodynamic properties of the helicopter mathematics model are given with necessary explanations.

4.2 Axis System Used in the Study

The axis systems used in this thesis are shown schematically on a rotorcraft model as seen in Figure 4.1.

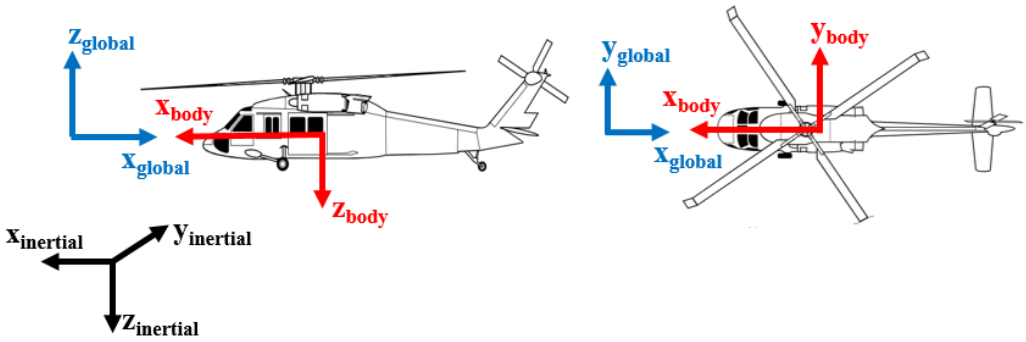


Figure 4.1: Global, Body and Inertial Axis System

4.2.1 Global Axis System

In this thesis, the global axis system is used to define the geometric dimensions of the helicopter model. The origin of the axis system is centered 0.5 meters in front of the nose of the helicopter. Moreover, the global axis system is defined as:

- x_{global} points out the backward direction of the helicopter
- y_{global} points out the starboard direction of the helicopter
- z_{global} points out the upward direction of the helicopter

4.2.2 Body Axis System

The body axis system is the axis system fixed to the rotorcraft. The origin of the axis system is at the helicopter center of gravity. Therefore, this axis system rotates and translates with the helicopter motion. Moreover, the body axis system is defined as:

- x_{body} points out the nose of the helicopter
- y_{body} points out the starboard direction of the helicopter
- z_{body} points out the downward direction of the helicopter

4.2.3 Inertial Axis System

The inertial axis system is the axis system fixed to the earth. This axis system does not rotate or translate with helicopter motion. Moreover, the inertial axis system is defined as:

- $x_{inertial}$ points out the north direction
- $y_{inertial}$ points out the east direction
- $z_{inertial}$ points out the downward direction

4.3 Mathematical Model

To able to analyze an aircraft maneuver in any software program, the aircraft must be modelled mathematically. In this thesis study, the used mathematical model of the rotorcraft is explained in detail.

4.3.1 Airframe Model

Main dimensions of the helicopter are given in Table 4.1.

Table 4.1: Helicopter Parameters

Parameter Name	Information	Unit
Length	13.87	m
Maximum Height	2.71	m
Maximum Width	2.60	m

The airframe is modelled as rigid in the mathematical model, but this does not mean it is just a single item. On the contrary, it consists of many items. These items have their masses and are located in many places on the structure. Therefore, the total mass of the helicopter is calculated by summing the masses of all these items. In addition, the center of gravity of the helicopter is computed by the accounting location of these items. To illustrate, the center of gravity calculation formula for the x-axis is

$$x_{cg} = \frac{\sum_{i=1}^{N}(x_i * m_i)}{\sum_{i=1}^{N} m} \tag{24}$$

Mass and center of gravity have a direct influence on the dynamic character of the rotorcraft. Therefore, the mass and center of gravity locations in global axis for each helicopter component are given in Table 4.2 - Table 4.11:

Table 4.2: Fuselage Mass Items

Mass Items	Mass [kg]	x_{cg} [m]	y_{cg} [m]	z_{cg} [m]
Item 1	12.0	0.40	0.00	1.00
Item 2	24.0	0.70	0.00	1.10
Item 3	34.0	1.30	0.00	1.10
Item 4	42.5	1.50	0.00	1.16
Item 5	46.4	1.70	0.00	1.22
Item 6	55.7	2.10	0.00	1.34
Item 7	68.1	2.40	0.00	1.45
Item 8	76.5	2.85	0.00	1.62
Item 9	75.5	3.40	0.00	1.82
Item 10	87.0	3.80	0.00	1.82
Item 11	60.0	4.60	0.00	1.82
Item 12	78.0	5.00	0.00	1.82
Item 13	83.8	5.30	0.00	1.82
Item 14	81.0	5.40	0.00	1.82
Item 15	71.0	5.60	0.00	1.82
Item 16	66.0	6.00	0.00	1.82
Item 17	55.0	6.55	0.00	1.90
Item 18	46.0	6.92	0.00	2.00
Item 19	34.0	7.35	0.00	2.10
Item 20	33.0	8.75	0.00	2.19
Item 21	31.0	9.16	0.00	2.23
Item 22	31.0	9.82	0.00	2.26
Item 23	26.0	10.08	0.00	2.29
Item 24	20.2	10.48	0.00	2.32
Item 25	18.0	10.80	0.00	2.34
Item 26	18.0	11.44	0.00	2.35
Item 27	17.5	11.88	0.00	2.35
Item 28	17.5	12.45	0.00	2.37
Item 29	16.8	13.05	0.00	2.38
Item 30	14.5	13.35	0.00	2.38
Item 31	8.0	13.68	0.00	2.38
Item 32	4.0	13.96	0.00	2.38

Table 4.3: Vertical Fin Mass Items

Mass Items	Mass [kg]	x_{cg} [m]	y_{cg} [m]	z_{cg} [m]
Item 41	5.0	13.00	0.00	3.00
Item 42	5.0	13.20	0.00	3.25
Item 43	5.0	13.40	0.00	3.40
Item 44	5.0	13.60	0.00	3.60

Table 4.4: Horizontal Tail Mass Items

Mass Items	Mass [kg]	x_{cg} [m]	y_{cg} [m]	z_{cg} [m]
Item 33	3.0	13.05	-1.20	2.10
Item 34	3.0	13.05	-0.90	2.10
Item 35	3.0	13.05	-0.30	2.10
Item 36	3.0	13.05	3.00	2.10
Item 37	3.0	13.05	6.00	2.10
Item 38	3.0	13.05	1.20	2.10

Table 4.5: Rotor System Component Masses

Mass Items	Mass [kg]	x_{cg} [m]	y_{cg} [m]	z_{cg} [m]
MR Hub	150.0	5.10	0.00	3.80
MR Gearbox	260.0	5.30	0.00	3.00
TR Hub	50.0	14.10	0.35	4.00
TR Gearbox	40.0	14.10	-0.10	3.93
Middle Gearbox	20.0	12.50	0.00	2.80

Table 4.6: Landing Gear Component Masses

Mass Items	Mass [kg]	x_{cg} [m]	y_{cg} [m]	z_{cg} [m]
Left MLG	65.0	6.20	-0.60	1.10
Right MLG	65.0	6.20	0.60	1.10
NLG	40.0	1.40	0.00	1.10

Table 4.7: Engine Masses

Mass Items	Mass [kg]	x_{cg} [m]	y_{cg} [m]	z_{cg} [m]
Left Engine	195.0	5.70	-0.45	2.90
Right Engine	195.0	5.70	0.45	2.90

Table 4.8: Fuel Tank Masses

Mass Items	Mass [kg]	x_{cg} [m]	y_{cg} [m]	z_{cg} [m]
Left Fuel Tank	180.0	5.45	-0.50	1.80
Right Fuel Tank	180.0	5.45	0.50	1.80

Table 4.9: External Equipment Masses

Mass Items	Mass [kg]	x_{cg} [m]	y_{cg} [m]	z_{cg} [m]
FLIR	100.0	1.20	0.00	0.70
Cooling Systems	30.0	4.10	0.00	2.75
All Antennas	40.0	6.05	0.30	2.50

Table 4.10: Pilots Masses with Seats

Mass Items	Mass [kg]	x_{cg} [m]	y_{cg} [m]	z_{cg} [m]
Pilot with seat	110.0	2.60	-0.42	1.50
Co-pilot with seat	110.0	2.60	0.42	1.50

Table 4.11: Passenger Masses

Mass Items	Mass [kg]	x_{cg} [m]	y_{cg} [m]	z_{cg} [m]
Passenger 1	110.0	3.40	-0.75	1.50
Passenger 2	110.0	3.40	0.75	1.50
Passenger 3	110.0	3.70	-0.50	1.50
Passenger 4	110.0	3.70	0.50	1.50
Passenger 5	110.0	3.90	-0.50	1.50
Passenger 6	110.0	3.90	0.50	1.50
Passenger 7	110.0	4.30	-0.75	1.50
Passenger 8	110.0	4.30	0.75	1.50

Hence, the mass and center of gravity location for the helicopter are obtained as in Table 4.12.

Table 4.12: Airframe Mass and Center of Gravity Location in Global Axis

Mass [kg]	**x_{cg} [m]**	**y_{cg} [m]**	**z_{cg} [m]**
4100.0	5.11	0.01	1.99

4.3.2 Main Rotor Model

4.3.2.1 Main Rotor Structural Model

The fundamental dimensions of the main rotor structural model are given in Table 4.13 with the necessary explanations.

Table 4.13: Main Rotor Structural Parameters

Parameter Name	**Information**	**Unit**	**Additional Description**
Type	-	-	Fully Articulated
Number of Blades	5	-	-
Rotation Direction	CCW	-	When seen from top view
Rotor Diameter	13.6	m	-
Disc Area	145.27	m^2	-
Hub Center Location	5.1	m	x in global axis system
	0	m	y in global axis system
	3.8	m	z in global axis system
Shaft Tilt Angle	6	degree	Forward in body axis
Rotational Speed	300	rpm	-
Hinge Offset	5	%	Fraction from the rotor hub center to tip of the blade [56]
Torque Offset	0	degree	No offset between feathering axis and the rotor hub center
Hub Pre-cone Angle	0	degree	Predefined conning angle [56]

The helicopter has a fully articulated main rotor system. Such rotor systems allow each blade to lead/lag, flap up/down and feather motion independent of the other blades [54]. These motions are shown in Figure 4.2.

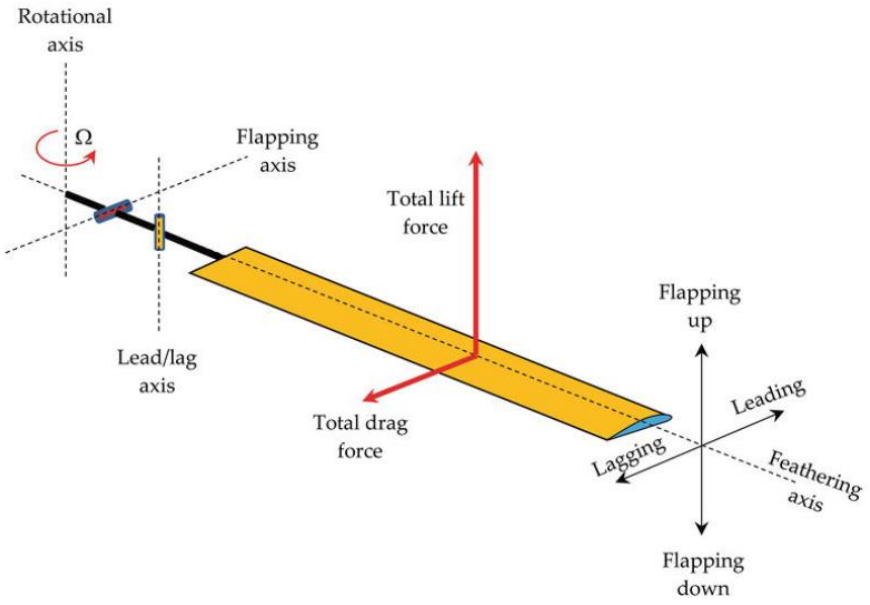


Figure 4.2: Three Main Motions of the Rotor Blade [55]

Main rotor blades which have a radius of 6.8 meters turn counter-clockwise when viewed from the top of the helicopter. Moreover, helicopter main rotor blades are modelled as a rigid blade.

4.3.2.2 Main Rotor Aerodynamic Model

The fundamental dimensions of the main rotor aerodynamic model are given in Table 4.14 with the necessary explanations.

Table 4.14: Main Rotor Aerodynamic Parameters

Parameter Name	Information	Unit	Additional Description
Airfoil	NACA 0012	-	Same throughout the span
Root Cutout	20	%	Fraction from the hub center to tip of the blade

Table 4.14 (cnt'd)

Chord Length	0.45	m	Same throughout the span
Twist Angle	0	degree	Same throughout the span
Sweep Angle	0	degree	Same throughout the span

The main rotor blades consist of NACA 0012 airfoil. This airfoil is symmetrical and has a 12% thickness to chord length. Moreover, the root cutout information shows that the lifting section of the blades starts at 20% of the blade radius. The blades retain their current structure along the span, do not show any rotation.

4.3.3 Tail Rotor Model

4.3.3.1 Tail Rotor Structural Model

The fundamental dimensions of the tail rotor structural model are given in Table 4.15 with the necessary explanations.

Table 4.15: Tail Rotor Structural Parameters

Parameter Name	Information	Unit	Additional Description
Type	-	-	Fully articulated
Number of Blades	4	-	-
Rotation Direction	CCW	-	When seen from right view
Rotor Diameter	2.7	m	-
Disc Area	5.73	m^2	-
Hub Center	14.1	m	x in global axis system
	0.35	m	y in global axis system
	4	m	z in global axis system
Shaft Tilt Angle	11	degree	Cant up
Rotational Speed	1430	rpm	-
Hinge Offset	11	%	Fraction from the rotor hub center to tip of the blade [56]
Torque Offset	0	m	No offset between feathering axis and the rotor hub center
Hub Pre-cone Angle	0	degree	Predefined conning angle [56]

Like the main rotor, the tail rotor has a fully articulated rotor system. The radius of tail rotor blades is 1.35 meters. They turn counterclockwise when viewed from the right of the helicopter. Moreover, helicopter tail rotor blades are also modelled as a rigid blade.

4.3.3.2 Tail Rotor Aerodynamic Model

The fundamental dimensions of the tail rotor aerodynamic model are given in Table 4.16 with the necessary explanations.

Table 4.16: Tail Rotor Aerodynamic Parameters

Parameter Name	Information	Unit	Additional Description
Airfoil	NACA 0012	-	Same throughout the span
Root Cutout	30	%	Fraction from the hub center to tip of the blade
Chord Length	0.22	m	Same throughout the span
Twist Angle	0	degree	Same throughout the span

The tail rotor blades have the same type of airfoil as the main rotor. In addition, their lifting section of the blades starts at 30% of the blade radius.

4.4 Conclusion

A mathematical model of the rotorcraft is developed for this thesis study. In this chapter, the developed mathematical model is described in three main structures which are airframe, main rotor and tail rotor model. In addition to total mass and CG calculation, the fundamental dimensions and properties of each model are explained in their sections.

CHAPTER 5

OPTIMIZATION METHODOLOGY USED IN HELICOPTER MANEUVERING

5.1 Introduction

In this chapter, the optimization methodology for helicopter maneuvering used in this thesis study is explained in detail. In addition, the mathematical formulation of the optimization code is described step by step.

5.2 Optimization Methodology for Helicopter Maneuvering

An optimization algorithm has three main elements, which are design variables, problem constraints and objective function as described in Section 2.3. In this thesis, the generation of high-quality helicopter maneuvers is defined as an optimization problem. Therefore, three main elements are specified for helicopter maneuver optimization as:

Design variables	:	Pilot control inputs
Constraints	:	Helicopter design limitations and maneuver requirements
Objective function	:	Helicopter maneuvering accuracy under constraints

Helicopters give a reaction according to pilot control inputs, so these inputs are the source of the optimization. During the maneuvering process, different pilot control inputs should be applied to produce the desired maneuver. Moreover, the pilot control inputs are time-dependent parameters. This means that when they are defined as design variables, time-dependent should also be taken into account. Therefore, the design variables vector for the helicopter maneuver optimization problem consists of four main pilot control inputs. These control inputs are called collective,

longitudinal and lateral cyclic for main rotor control and anti-torque pedal for tail rotor control. Finally, the time-dependent design variable vector is defined as

$$\boldsymbol{X}^T = \left[\delta_{co}(t_1), \ldots \delta_{co}(t_n), \delta_{lo}(t_1), \ldots \delta_{lo}(t_n), \delta_{la}(t_1), \ldots \delta_{la}(t_n), \delta_{ap}(t_1), \ldots \delta_{ap}(t_n)\right]$$

where t_n represents the n^{th} time step. In this thesis, time steps are selected as equal interval, but this is not essential.

An example of the design variable vector is shown in Figure 5.1. This figure shows the first two values of the vector.

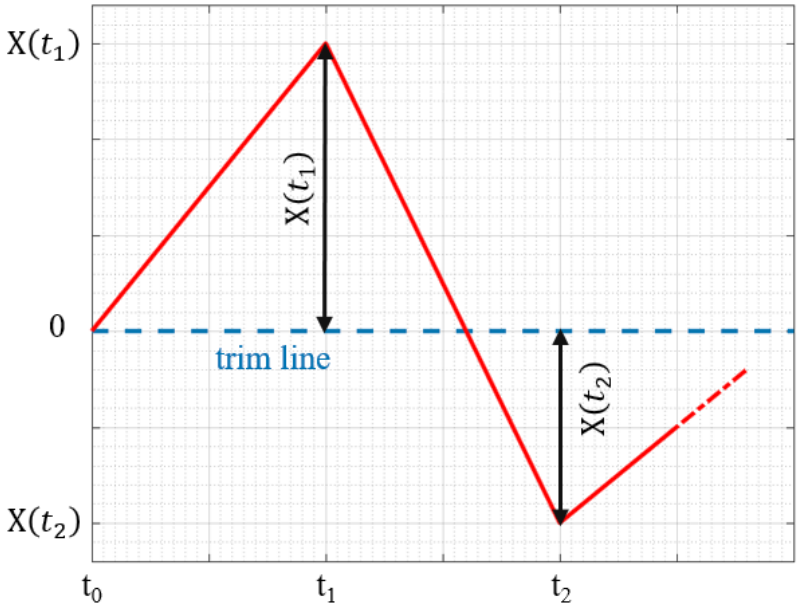


Figure 5.1: An Example of Design Variable Vector

As it is seen from the figure that the values of the design variable vector start at time t_1 because the value at time t_0 represents the trim value of the used design variable parameter. Therefore, the design variable values are delta values applied to the trim point. In addition, these values change linearly in successive time intervals.

In the FLIGHTLAB mathematical model of the helicopter, there is a Stability Augmentation System (SAS) for all pilot control inputs. This system keeps the helicopter in its current state if there is no input to a helicopter. Thus, SAS can help

the optimization algorithm to get the solution faster. In other words, the optimization process can be carried out by defining only the main pilot control inputs for particular maneuvers, other inputs can be controlled by SAS.

The pilot control inputs can be applied within a certain range. In this thesis, this range is expressed in percentages as shown in Table 5.1. Moreover, the value of 50% means that the control inputs are in a neutral position.

Table 5.1: Definition of Pilot Control Input Motion

Pilot Control Inputs	**Percentage [%]**	**Controller Movement**
Longitudinal Cyclic	0	Forward
	100	Backward
Lateral Cyclic	0	Leftward
	100	Rightward
Collective Cyclic	0	Up
	100	Down
Anti-Torque Pedal	0	Right
	100	Left

All helicopters have some limitations due to their structural, dynamic and aerodynamic designs. For example, they have a maximum power limitation due to the capacity of the engine or they have a maximum pitch angle limitation because when they exceed this limit, they might go in the stall condition. Therefore, when the optimization of helicopter maneuvering is modelled, these limitations should be considered as constraints. In addition, each maneuver has its own characteristic, which brings certain requirements. To illustrate, if it is desired to maintain the altitude of the helicopter during any maneuver, the altitude change will be one of the conditions for the maneuver. Hence, such requirements should also be considered as constraints of the optimization algorithm.

Constraints can be designated as inequality constraints, equality constraints and geometric constraints with lower and upper bounds. Moreover, the logarithmic

barrier function is also used in this thesis study in order to define some strict constraints. Figure 5.2 shows that the objective value goes to infinity when the parameter approaches or exceeds the limits. In other words, this function does not allow the defined parameter to approach the barrier limit.

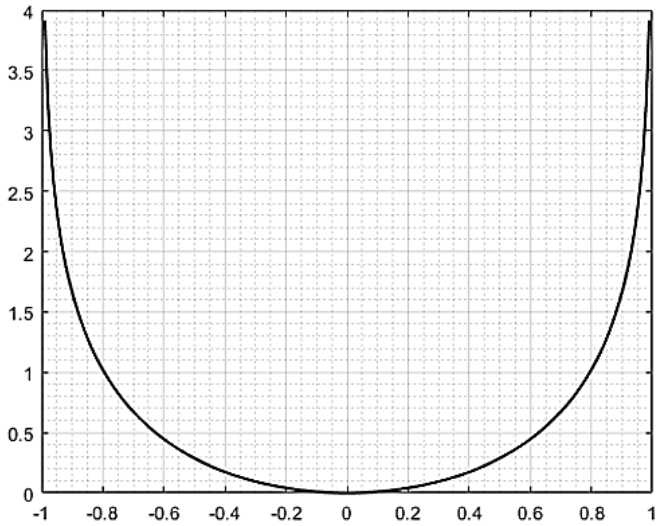


Figure 5.2: Representation of the Logarithmic Barrier Function

In the helicopter maneuver optimization problem, the objective function is configured with the combination of all these constraints. In this study, the penalty function is used to transform a constrained problem to an unconstrained one by creating an artificial penalty [57]. Thus, the weight of each constraint is determined by penalty value (p), then the objective function is formulated mathematically as,

$$f(X) = p_1 * g_1 + p_2 * g_2 + \cdots + p_n * g_n \qquad (25)$$

where the constraint and its penalty at n^{th} constraint are represented with g_n and p_n respectively. Moreover, the weight of each constraint can be controlled by penalty parameters. In other words, a higher penalty value means a significant constraint for the optimization process.

5.3 Mathematical Formulation for Helicopter Maneuvering Optimization

In this thesis study, quasi-Newton methods described in Section 3.2.2 are examined as a gradient optimization method. It is known that the main difference between them is the computational technique of the Hessian Matrix. In other words, the mathematical structure of the algorithms is the same, except for the calculation of the approximated Hessian matrix. Therefore, all optimization algorithms have the same code structure. Their only difference is the calculation formula of the approximate Hessian matrix $\boldsymbol{B}^{(k+1)}$. The general formulation of the optimization methods is

$$x^{(k+1)} = x^{(k)} - \alpha \left[\boldsymbol{B}^{(k)}\right]^{-1} \nabla f\left(x^{(k)}\right) \tag{26}$$

As it is described in Section 5.2, the optimization algorithm uses the time-dependentdesign variable vector. In order to create this vector, the pilot control inputs and the time steps in which the design variables are applied must be defined. Therefore, the optimization code is constructed as below:

Optimization Code Algorithm

Inputs: Identify the initial design variable vector $\boldsymbol{X}^T$

Identify the step size α and perturbation constant ε values

Loop: until the termination criteria is met

Calculate the current objective function $f\left(x^{(k)}\right)$

Calculate the gradient of the objective function $\nabla f\left(x^{(k)}\right)$

Calculate the approximate Hessian matrix $\boldsymbol{B}^{(k+1)}$

Apply the line search algorithm

End Loop

At the beginning of the optimization code, the pilot control inputs and the time interval are explained in detail for each maneuver because they differ according to

the maneuver type and the desired maneuver requirements. To explain better the optimization code structure, suppose that the design variables are δ_1 and δ_2. Moreover, the time steps are t_1 and t_2. Therefore, the design variable vector has four variables as:

$$\boldsymbol{X}_c^T = [\delta_1(t_1), \delta_1(t_2), \delta_2(t_1), \delta_2(t_2)]$$

In addition, the values of step size (α) and perturbation constant (ε) are also defined at the beginning of the code. Thus, the preparation part of the optimization code is completed.

The main part of the optimization code consists of repeated analysis until the termination criteria are met. In the main part, firstly, FLIGHTLAB analysis is performed for the current design variable vector and then the current objective value is obtained. Afterwards, the optimization code continues with the gradient calculation of the objective function. Therefore, each value of the current design variable vector is perturbed by the perturbation constant value in order to apply the finite divided difference approximations of first-order derivative. Then, FLIGHTLAB analyses are executed for all these perturbed vectors. Thus, their objective values are calculated for the desired finite divided difference approximation. Note that FLIGHTLAB analysis is required to calculate each objective value. Moreover, the steps of the gradient computation process are shown in Table 5.2 for the assumed example as

Table 5.2: The Process of the Gradient Computation

Steps	**Formulation**
Perturbed Vectors	$\boldsymbol{X}_{p_1}^T = [\delta_1(t_1) \pm \varepsilon,\ \delta_1(t_2),\ \delta_2(t_1),\ \delta_2(t_2)]$ $\boldsymbol{X}_{p_2}^T = [\delta_1(t_1),\ \delta_1(t_2) \pm \varepsilon,\ \delta_2(t_1),\ \delta_2(t_2)]$ $\boldsymbol{X}_{p_3}^T = [\delta_1(t_1),\ \delta_1(t_2),\ \delta_2(t_1) \pm \varepsilon,\ \delta_2(t_2)]$ $\boldsymbol{X}_{p_4}^T = [\delta_1(t_1),\ \delta_1(t_2),\ \delta_2(t_1),\ \delta_2(t_2) \pm \varepsilon]$
FLIGHTLAB analysis for each perturbed vector.	
Objective Functions	$\left[f(\boldsymbol{X}_{p_1}^T) \quad f(\boldsymbol{X}_{p_2}^T) \quad f(\boldsymbol{X}_{p_3}^T) \quad f(\boldsymbol{X}_{p_4}^T) \right]^T$

After the objective values of perturbed design variables are calculated, the desired finite divided difference approximation can be applied. Thus, the gradient of the objective function ∇f can be obtained.

After calculating the gradient of the objective function, the optimization algorithms need the approximate Hessian matrix. As known from Section 3.2.2, each optimization algorithm calculates the approximate Hessian matrix differently. Therefore, the optimization code is formulated according to the type of method used at this point.

After finding the search direction, the optimization algorithm applies the line search algorithm to achieve the minimum objective in the search direction. The line search algorithm is one of the main steps of the gradient-based optimization methods. In this thesis, the line search algorithm is formulated as

$$\alpha^{(k)} = arg\ min\ f\left(x^{(k)} + \alpha * d^{(k)}\right) \text{ where } d^{(k)} = -\left(B^{(k)}\right)^{-1} \nabla f\left(x^{(k)}\right)$$

After the line search direction is obtained, $d^{(k)}$ is normalized with its maximum absolute value. Thus, the maximum line search direction will be equal to the value of α. Hence, the lower and upper boundaries of the line search algorithm are obtained as $x^{(k)}$ and $x^{(k)} + \alpha * d^{(k)}$ respectively.

In the line search algorithm, it is aimed to achieve the minimum objective value in the defined line search direction. For this reason, the interval between the lower and upper boundaries is divided into eight equal parts and the objective values at each point are calculated. Afterwards, new boundaries are created according to the position of the minimum objective value on this line. The line search algorithm is represented schematically in Figure 5.3.

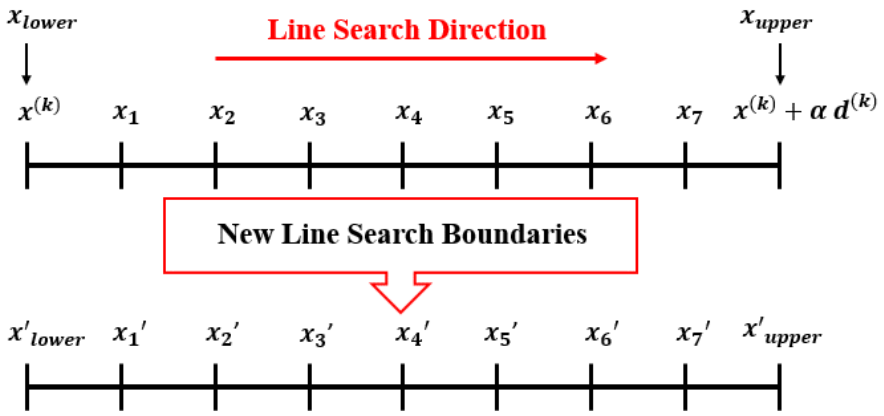


Figure 5.3: Representation of Line Search Algorithm

In order to start the line search algorithm, the current design point is defined as the lower boundary of the line search algorithm x_{lower} and the value obtained by going in the search direction as much as the constant α value is defined as the upper boundary x_{upper}. Moreover, the line search algorithm is created based on the position of the minimum objective value. Therefore, after the boundaries of the search direction have been obtained, the range of the boundaries is divided into eight equal parts at the beginning of each line search iteration. Moreover, the maximum iteration number, which is a number to terminate the line search algorithm, is assigned. After that, the following conditions are checked while the current iteration number is less than this maximum iteration number. These conditions are used in order to define new boundaries of the line search direction. Furthermore, if one of the conditions is met, the algorithm turns and starts from the beginning. Also, the line search current iteration number is incremented by 1. Hence, the conditions used by the line search algorithm to be able to define new boundaries are:

i. If a minimum objective point is obtained at the value of $f(x_{lower})$ and the current iteration number is less than the maximum iteration number, the new lower boundary x'_{lower} and the new upper boundary x'_{upper} are defined as

$$x'_{lower} = x_{lower} \quad \text{and} \quad x'_{upper} = x_1$$

Figure 5.4 is shown as an example to explain better how these new boundaries are created. However, although the figure of the new boundary criterion is not shown for each condition, each uses the same format.

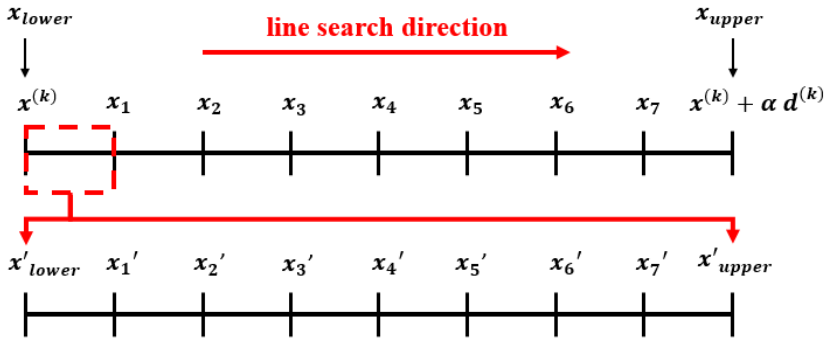


Figure 5.4: Creation of New Boundaries of Line Search Algorithm

ii. If a minimum objective point is obtained at the value of $f(x_{lower})$ and the current iteration number is equal to the maximum iteration number, the line search is done in the reverse direction by the one-time constant α value. Therefore, the new lower boundary x'_{lower} and the new upper boundary x'_{upper} are defined as,

$$x'_{lower} = x_{lower} - \alpha * d^{(k)} \quad \text{and} \quad x'_{upper} = x_1$$

In addition, the current iteration number is reset to a value of 1 and the maximum iteration number is increased to be able to search more intervals.

iii. If a minimum objective point is obtained at the value of $f(x_1)$ and the current iteration number is less than the maximum iteration number, the new lower boundary x'_{lower} and the new upper boundary x'_{upper} are defined as

$$x'_{lower} = x_{lower} \quad \text{and} \quad x'_{upper} = x_2$$

iv. If a minimum objective point is obtained at the value of $f(x_7)$ and the current iteration number is less than the maximum iteration number, the new lower boundary x'_{lower} and the new upper boundary x'_{upper} are defined as

$$x'_{lower} = x_6 \quad \text{and} \quad x'_{upper} = x_{upper}$$

v. If a minimum objective point is obtained at the value of $f(x_{upper})$ and the current iteration number is less than the maximum iteration number, the line search is expanded in the same direction to check if there is a lower objective value. This expansion is done up to the boundary range. Therefore, the new lower boundary x'_{lower} and the new upper boundary x'_{upper} are defined as

$$x'_{lower} = x_7 \quad \text{and} \quad x'_{upper} = 2 * x_{upper} - x_{lower}$$

In addition, the current iteration number is reset to a value of 1 and the maximum iteration number is increased to be able to search more intervals. Note that this expansion can only be done once for each optimization iteration.

vi. If a minimum objective point is obtained between the boundaries and the current iteration number is less than the maximum iteration number, the neighbors of the point are identified as new boundaries. Let assume that the minimum objective point is obtained at the value of $f(x_4)$. Hence, the new lower boundary x'_{lower} and the new upper boundary x'_{upper} are defined as

$$x'_{lower} = x_3 \quad \text{and} \quad x'_{upper} = x_5$$

Finally, the line search algorithm is terminated when the current iteration number reaches the maximum iteration number for the line search. Thus, one of the optimization code loop is completed for k^{th} iteration and the optimization point of $x^{(k+1)}$ is obtained. Moreover, the optimization code loop is executed until the termination criterion is met. In this thesis study, the termination criterion is based on the change of the objective value. In other words, if the objective value no longer changes significantly, the optimization code is terminated.

5.4 Conclusion

In this chapter, the optimization elements are assigned for helicopter maneuver optimization. The definitions and application of each optimization parameter are

described in detail. Then, the helicopter maneuver optimization is formulated mathematically. Moreover, the developed optimization code algorithm is explained step by step with the necessary examples.

CHAPTER 6

THE SELECTION OF THE MOST USEFUL CONFIGURATION AMONG VARIOUS OPTIMIZATION METHOD CONFIGURATIONS

6.1 Introduction

The main purpose of this section is to decide the most useful configurations of optimization methods which are Broyden's Method, Symmetric Rank-One Method, Davidon-Fletcher-Powell Method, Broyden-Flethcher-Goldfarb-Shanno Method. Therefore, Hover to Forward Flight and Hover to Sideward Flight maneuvers are performed by using different optimization configurations for each method. Then, these configurations are compared with each other to decide the most useful method for helicopter maneuver optimization. The main purpose of choosing different configurations is to find the best generic combination for optimization methods. For this comparison to be reliable, all optimization methods used the same initial design variables as the starting point and the identity matrix as the first approximate Hessian matrix. At the end of the process, the most useful method configuration for the helicopter maneuver optimization is obtained.

6.2 Comparison Process

In the comparison process, firstly, a maneuver to be studied is decided and explained in detail. Afterwards, the maneuver is modelled mathematically for optimization structure. The design variables, constraints and objective function of the optimization problem are appointed in this mathematical model. After the optimization problem is configured, the comparisons of the specified configurations are examined.

In this comparison process, all different configurations are initialized as zero values for each element of the initial design variable vector. The reason for choosing this starting point as the initial value is explained in Section 7.4. In addition, these configurations are used the identity matrix as the first approximate Hessian matrix. During the optimization process, the last updated version of the optimization code described in Section 7.2 is used.

First of all, forward, central and backward difference approximations have been researched to estimate the numerical derivative of the objective function in the most efficient way. This research is carried out at fixed step size and perturbation constant values for all methods. After determining the most effective finite divided difference approximation to use under the chosen conditions, the best combination of step size and perturbation constant is obtained for each method. Firstly, different perturbation constant values are compared for the same step size and all other fixed optimization parameters. After determining the best perturbation constant value, different step size values are investigated in this perturbation constant value. Thus, the combination of the step size and perturbation constant is obtained for each method.

In all these comparison processes, the values of the objective function along both the optimization time and the iteration number are plotted for each configuration. The objective function value represents the accuracy of the optimization. A lower value of the objective function means a better solution is achieved. Furthermore, both shorter optimization time and smaller optimization number mean the method is faster. In short, the best solution method can be expressed as reaching the lowest objective value with minimum optimization time and iteration number.

6.3 Comparison of the Optimization Methods

Hover to Forward Flight and Hover to Sideward Flight maneuvers are used to achieve the most useful optimization algorithm with its best design parameter combinations. In addition to the definition of the maneuvers, the requirements of the

maneuvers are explained by their limitations. Then, the different configurations for each method are compared in this section.

6.3.1 Hover to Forward Flight Maneuvering with Optimization Methods

6.3.1.1 Hover to Forward Flight

Hover to forward flight maneuvers start from a stationary hover position and then continue with the forward motion of the rotorcraft. These maneuvers aim to reach a constant speed and to sustain the process with almost the same altitude and heading.

How can it be performed [54]?

1. Bring helicopter to hover position at a specific altitude then move the longitudinal cyclic in a forward direction to accelerate the helicopter
2. Apply necessary collective cyclic inputs to avoid any possible altitude changes
3. Apply necessary anti-torque pedal inputs to avoid any possible excessive heading changes
4. Complete the maneuver as the desired forward speed is reached
5. All these parameters must be implemented at the same time to generate maneuver properly

6.3.1.2 Optimization Modeling of Hover to Forward Flight Maneuver

The main pilot control inputs for forward acceleration maneuvers are longitudinal and collective cyclic as mentioned in Section 6.3.1.1. Therefore, these control inputs are defined as design variables in this maneuver. Nevertheless, the lateral cyclic and anti-torque pedal inputs are not taken as design variables because they are controlled by SAS.

One of the key points for creating design variables is to decide the maneuver time and at which time points the design variable values are modified. In this optimization, the maneuver takes about 7 seconds and the pilot control inputs, which are longitudinal and collective cyclic, are modified at each one-second. Therefore, the design variable vector consists of $7 * 2 = 14$ design points as

$$\boldsymbol{X}^T = [\delta_{co}(t=1), \delta_{co}(t=2), \dots \delta_{co}(t=7), \delta_{lo}(t=1), \delta_{lo}(t=2), \dots \delta_{lo}(t=7)]$$

The constraints of this optimization problem are defined according to helicopter design limits and maneuver requirements. In this maneuver, it is targeted to reach 30 Knot forward velocity while the helicopter maintains its almost constant altitude. Furthermore, helicopter design limitations must be taken into account when maneuvering. A helicopter cannot fly at or outside the design limits because its stability is impaired as it approaches its limits. Therefore, these design limitations are mathematically constrained with the logarithmic barrier function. Hence, the constraints of the maneuver optimization problem are created as:

Constraint 1 - Blade Flap Up/Down Constraint

While helicopter main rotor blades rotate, they tend to move up and down around a hinge in the rotor system. This motion of the blades is defined as blade flapping motion. Helicopter main rotor design permits the blade to move up and down between pre-limited angles, so this is one of the helicopter design limitations. Therefore, it is mathematically formulated for this optimization problem as:

$$g_1 = \begin{cases} -log(\gamma_{max} - max(\gamma)) - log(\gamma_{min} + min(\gamma)) & for \quad max(\gamma) < \gamma_{max} \; and \; min(\gamma) > \gamma_{min} \\ Infinity & for \quad otherwise \end{cases}$$

where the main rotor blade flapping angle is represented by $\boldsymbol{\gamma}$. Also, the maximum and minimum blade flapping angle limits are represented by $\boldsymbol{\gamma_{max}}$ and $\boldsymbol{\gamma_{min}}$ respectively.

Constraint 2 – Pitch Angle Constraint

The pitch angle is the angle between the helicopter's longitudinal axis and horizontal plane. The pitch angle is directly affected by the longitudinal cycle movement. All helicopters have maximum and minimum pitch angle limitations where they can maneuver. These limits come from aerodynamic design. If they are exceeded, the helicopter enters into stall condition and loses its lifting force. Therefore, this angle should be limited in constraints when the longitudinal cycle is used as a design variable. To prevent the stall condition, these limitations are mathematically formulated as

$$g_2 = \begin{cases} -log\left(\theta_{max} - max(\theta)\right) - log\left(\theta_{min} + min(\theta)\right) & for \quad max(\theta) < \theta_{max} \; and \; min(\theta) > \theta_{min} \\ Infinity & for \quad otherwise \end{cases}$$

where the pitch angle of the helicopter is represented by $\boldsymbol{\theta}$. Also, the maximum and minimum pitch angle limits are represented by $\boldsymbol{\theta_{max}}$ and $\boldsymbol{\theta_{min}}$ respectively.

Constraint 3 – Engine Power Constraint

Due to the capacity of the engine, the helicopter has limited maximum power to use it during maneuvers. In addition, the minimum power level can be defined as zero because there are some power-free maneuvers, such as autorotation flight. The power limitation is also one of the design limitations, so it is formulated as

$$g_3 = \begin{cases} -log\left(P_{max} - max(P)\right) - log\left(P_{min} + min(P)\right) & for \quad max(P) < P_{max} \; and \; min(P) > P_{min} \\ Infinity & for \quad otherwise \end{cases}$$

where the power of the helicopter supplied from engine is represented by $\boldsymbol{P}$. Also, the maximum and minimum engine power limits are represented by $\boldsymbol{P_{max}}$ and $\boldsymbol{P_{min}}$ respectively.

Constraint 4 – Forward Velocity Constraint

In this maneuver, it is targeted that the helicopter reaches the 30 Knot velocity in the forward direction between 6.3^{th} and 6.7^{th} seconds. The constraint will be met when

the desired velocity is reached and the helicopter no longer accelerates. This goal is one of the maneuvering requirements, not the helicopter design limitation. This constraint is defined mathematically in this problem as:

$$g_4 = \int_{t_s=6.3}^{t_f=6.7} |U - 30|^2 \, dt$$

where the helicopter longitudinal velocity is represented by $\boldsymbol{U}$ and is considered a positive sign to the forward. Also, the upper and lower time boundaries where the targeted parameter is desired to be obtained are represented with $\boldsymbol{t_s}$ and $\boldsymbol{t_f}$ respectively.

Constraint 5 – Altitude Constraint

The other maneuver requirement for this maneuver is to keep the helicopter at almost the same altitude. Although this restriction is not a helicopter design limitation, it is also formulated with a logarithmic barrier function to keep the helicopter at permitted height limits. The maneuver is performed at $100\ [ft]$ altitude and the helicopter is allowed to ascend/descend all $10\ [ft]$ maximum. Hence, the constraint is formulated as

$$g_5 = \begin{cases} -log\big(h_{max} - max(h)\big) - log\big(h_{min} + min(h)\big) & for \quad max(h) < h_{max} \; and \; min(h) > h_{min} \\ Infinity & for \quad otherwise \end{cases}$$

where the helicopter altitude is represented by $\boldsymbol{h}$. Also, the allowed maximum and minimum altitude limits are represented with $\boldsymbol{h_{max}}$ and $\boldsymbol{h_{min}}$respectively. In this maneuver, the maximum and minimum limitations are defined as $\boldsymbol{h_{max}} = 110\ [ft]$ and $\boldsymbol{h_{min}} = 90\ [ft]$.

Finally, the objective function is created by multiplying these constraints with penalties to arrange the importance of the constraints. In this maneuver, the penalty parameters p_1, p_2, p_3 and p_5 are taken as 5 and p_4 is taken 100. Then, the objective function is structured as:

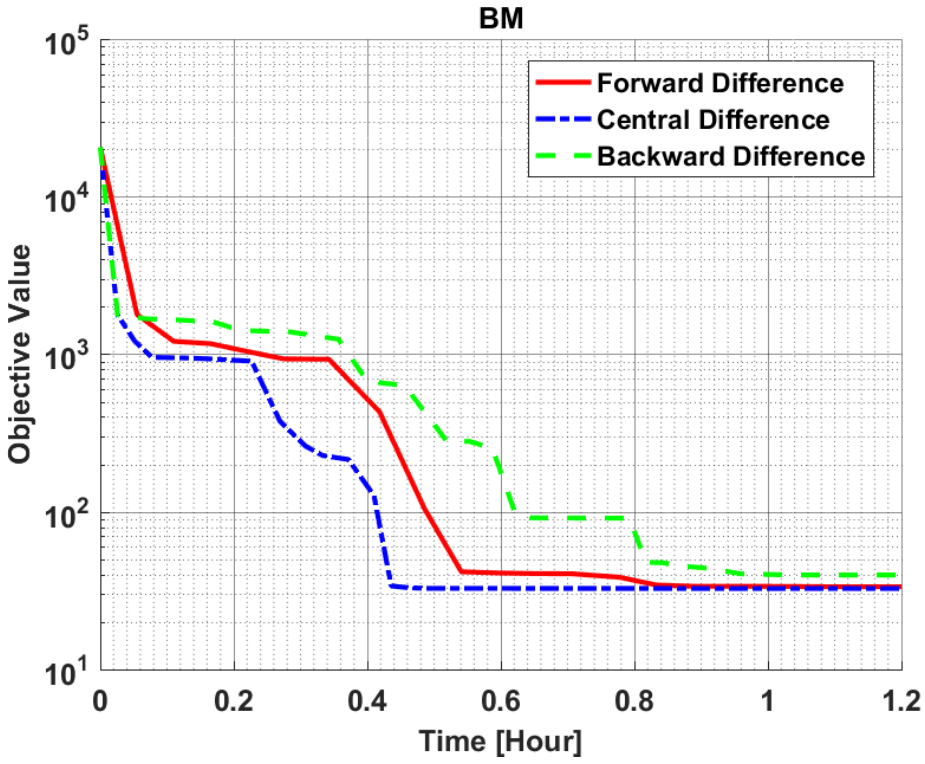
$$f(X) = p_1 g_1 + p_2 g_2 + p_3 g_3 + p_4 g_4 + p_5 g_5 \text{ where } p_{1,2,3,5} = 5 \,\&\, p_4 = 100$$

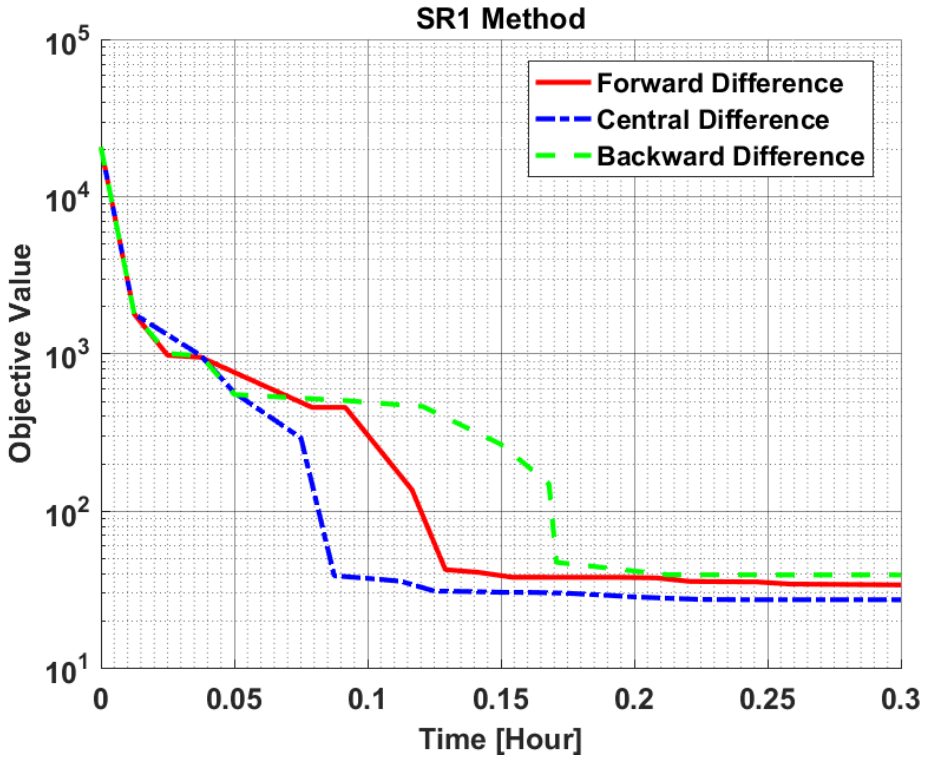
In this objective function definition, Constraint 4 is defined as the most critical constraint.

6.3.1.3 Decision of Finite Divided Difference Approximation for Hover to Forward Flight Maneuver

As it is described in Section 3.2.2, all the methods used in this thesis need the derivative of the objective function. Since the most useful optimization method is tried to be obtained, the gradient of the objective function must also be calculated most effective way. In order to calculate the gradient, forward, central and backward difference approximations are preferred and applied to each optimization method. Then, the best gradient calculation method for each method is decided by examining the change of objective function value.

To compare the finite divided difference approximations for hover to forward flight maneuver, the optimization methods are solved with constant parameter values. For example, while the perturbation constant is taken as a fixed value of 0.1, the value for the step size parameter is taken as 5. Afterwards, the changes of the objective function along both the optimization time and the iteration number are plotted for all methods as seen in Figure 6.1 and Figure 6.2, respectively.


BM
Forward Difference
Central Difference
Backward Difference
Objective Value
Time [Hour]
10^5
10^4
10^3
10^2
10^1
0
0.2
0.4
0.6
0.8
1
1.2


SR1 Method
Forward Difference
Central Difference
Backward Difference
Objective Value
Time [Hour]
10^5
10^4
10^3
10^2
10^1
0
0.05
0.1
0.15
0.2
0.25
0.3

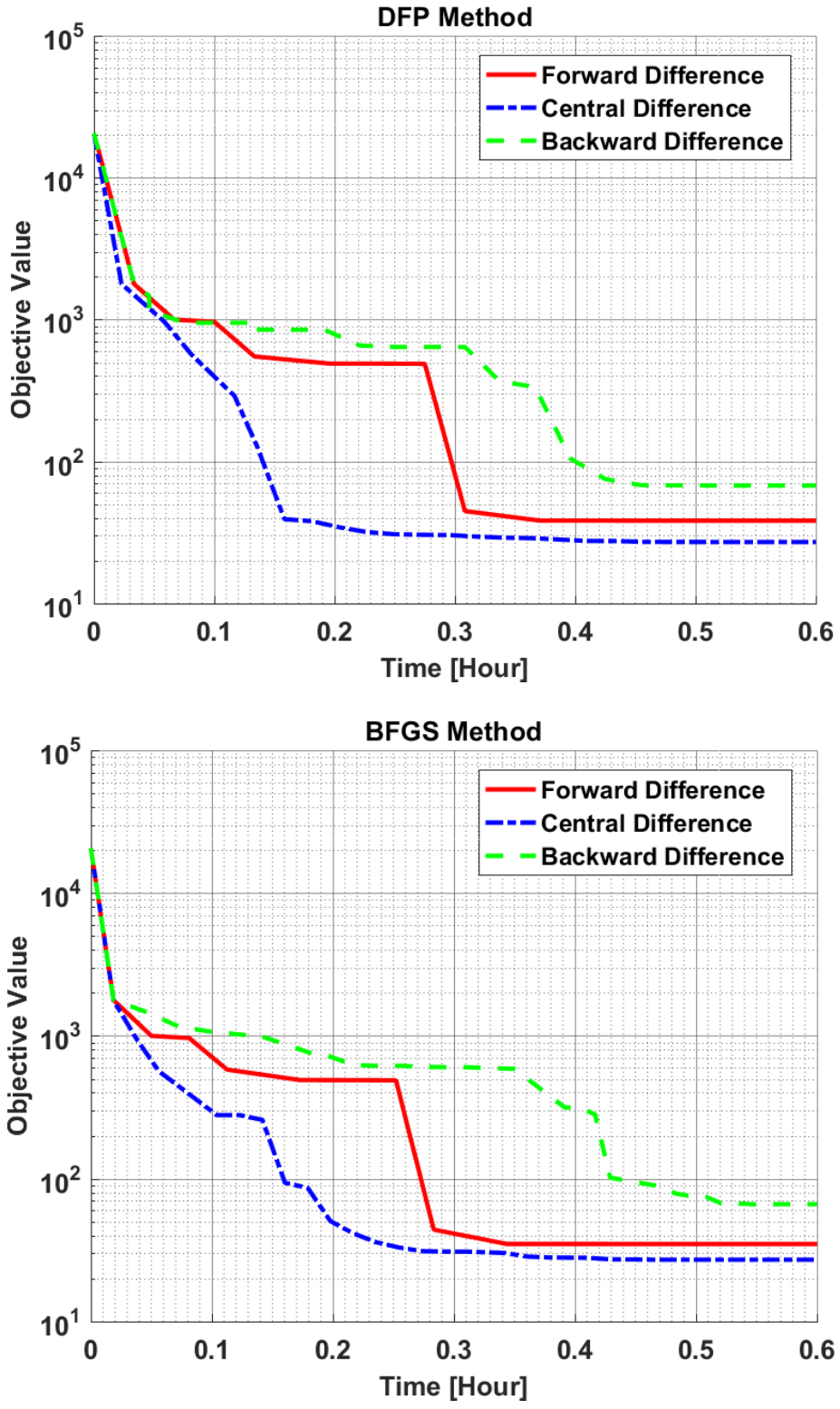
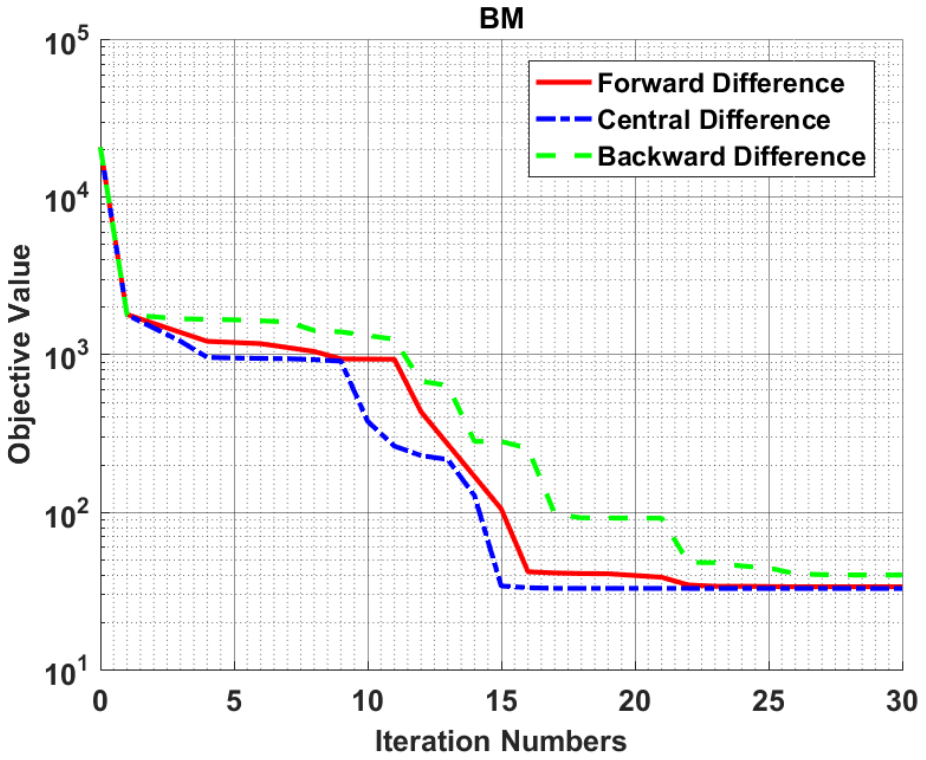
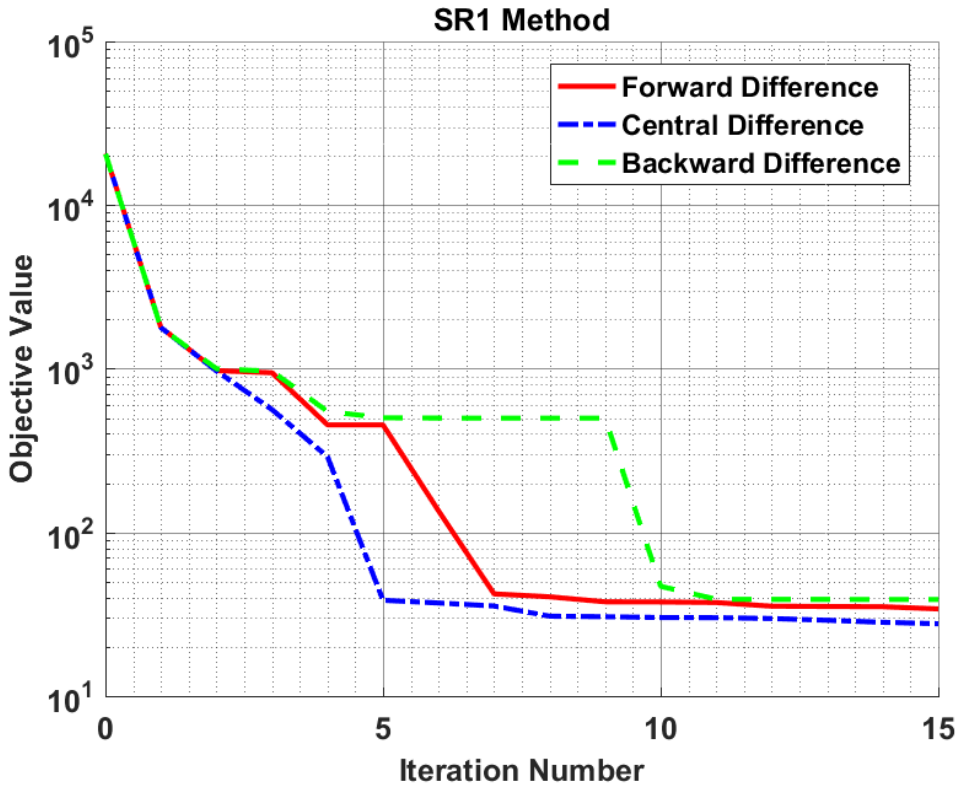


Figure 6.1: Comparison of Objective Change along Time for the Finite Divided Difference Approximations in Hover to Forward Flight Maneuver


BM
Forward Difference
Central Difference
Backward Difference
Objective Value
Iteration Numbers
10^5
10^4
10^3
10^2
10^1
0
5
10
15
20
25
30


SR1 Method
Forward Difference
Central Difference
Backward Difference
Objective Value
Iteration Number
10^5
10^4
10^3
10^2
10^1
0
5
10
15

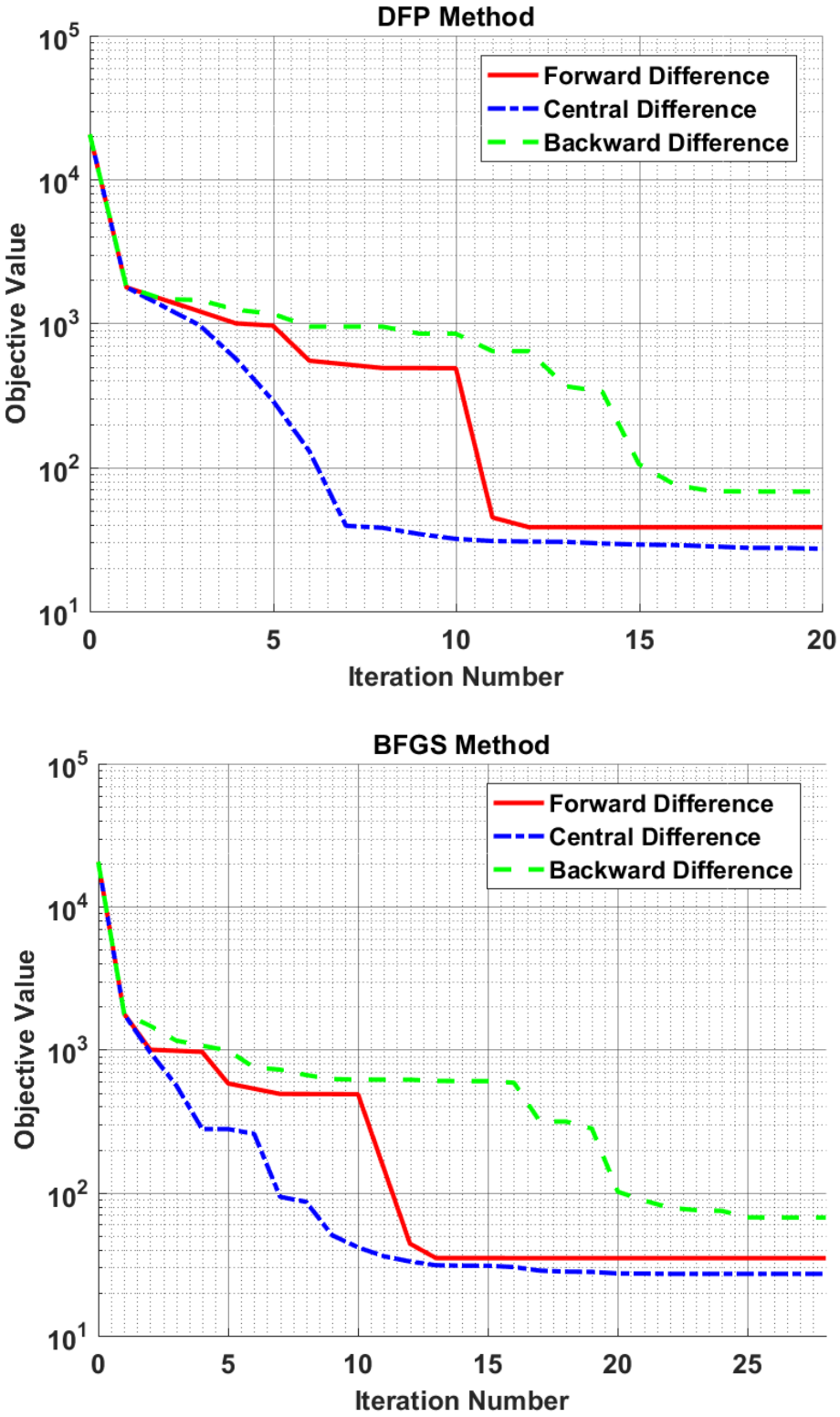


Figure 6.2: Comparison of Objective Change along Iteration Number for the Finite Divided Difference Approximations in Hover to Forward Flight Maneuver

Figure 6.1 and Figure 6.2 show that the finite divided difference approximations have a great effect over problem convergence for each algorithm. As it is mentioned in Section 6.2, all methods use the same initial design variables. Therefore, the objective value of each solution in Figure 6.1 and Figure 6.2 start from the same initial objective value as expected.

It can be concluded that the slowest difference approximation for all optimization methods is the backward difference approximation and the fast one is the central difference approximation. Although the central difference approximation needs two FLIGHTLAB analyses compared to other methods, it converges faster than others because it provides a more accurate search direction than others. Therefore, the central difference approximation has been decided as the most useful difference approximation under defined conditions in the calculation of the objective gradient. Namely, this difference approximation is accepted as the gradient calculation method for hover to forward flight maneuvering problems in all these optimization methods.

For all unique solutions in this comparison process, the constraint responses of the maneuver and corresponding pilot control inputs are given in Appendix A1.

6.3.1.4 Sensitivity Analysis of Step Size and Perturbation Constant Parameters for Hover to Forward Flight Maneuver

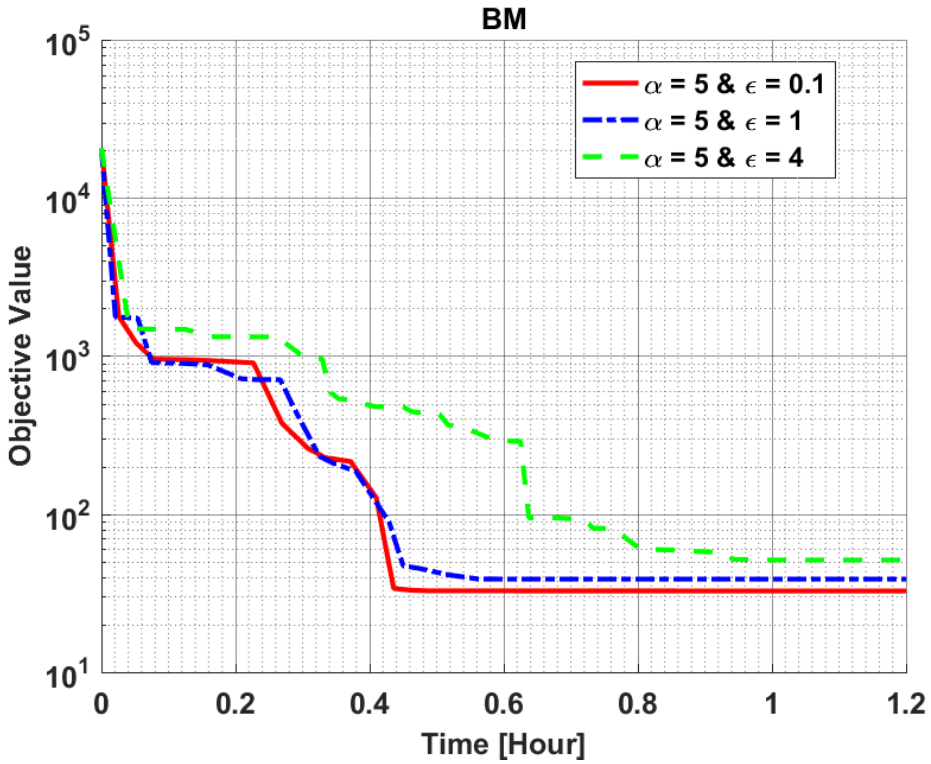
While deciding on the most useful optimization method, different step size and perturbation constant values should be examined to obtain the best combination. Firstly, the different perturbation constants are compared for the same step size and all other fixed optimization parameters. After deciding the best perturbation constant value, some different step size values are investigated at the best perturbation constant value. Thus, the combination of the step size and perturbation constant is obtained for each method. These sensitivity analyses are executed using the most effective finite divided difference approximation obtained in Section 6.3.1.3. To be able to make this controlled investigation, the values of the objective function along

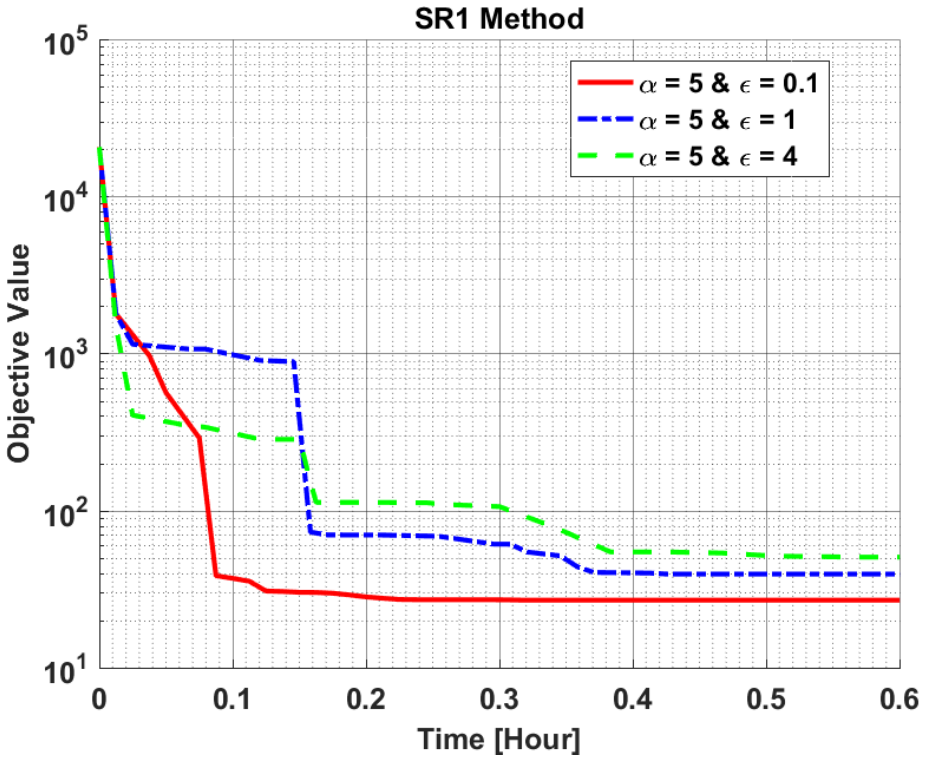
both the optimization time and the iteration number are plotted for each comparison condition.

In the sensitivity analysis figures, the changes in the objective function along both the optimization time and the iteration number give the efficiency of each algorithm. Even though some of the analyses are terminated earlier, the results of the final analysis are expanded to be comparable to other analyses. Moreover, it can be seen from comparison figures that some objective values dropped more sharply at the beginning of the optimization. However, the best parameter value is decided according to the lowest objective achieved earlier. Moreover, the objective values could not converge to zero because it is highly affected by optimization parameters and constraints. For example, if the design variables are applied at smaller intervals, the algorithm will be more able to control the problem. Thus, the constraints could be met more accurately. The detailed investigations on the importance of the time interval selection are explained in Section 7.3.

In the process of obtaining the best combination for hover to forward flight maneuver optimization, while the values of 2, 5 and 10 are taken as step size value, the values of 0.1, 1 and 4 are taken as perturbation constant value. Moreover, the central difference approximation is applied for these sensitivity analyses because it is accepted as the gradient computation method in Section 6.3.1.3. For the same reason as in Figure 6.1 and Figure 6.2, each figure in this subchapter also starts from the same initial objective value.

For the sensitivity analysis of perturbation constant, the optimization methods are executed with predefined perturbation constant values while step size value is taken 5 and all other optimization parameters are kept constant. Then, the changes of objective values along both the optimization time and the iteration number for these perturbation constant values are obtained as seen in Figure 6.3 and Figure 6.4.


BM
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
Objective Value
10^5
10^4
10^3
10^2
10^1
0
0.2
0.4
0.6
0.8
1
1.2
Time [Hour]


SR1 Method
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
Objective Value
10^5
10^4
10^3
10^2
10^1
0
0.1
0.2
0.3
0.4
0.5
0.6
Time [Hour]

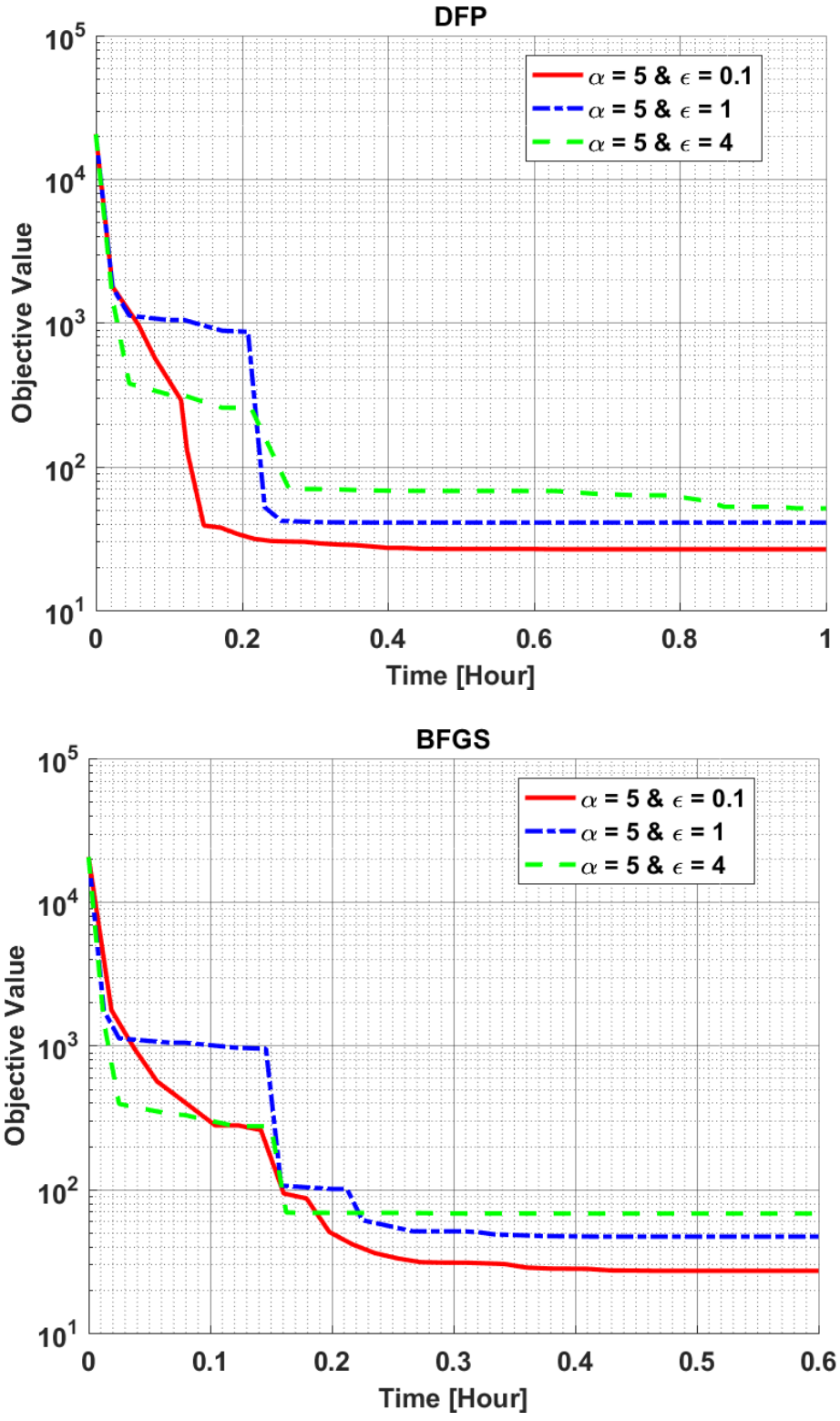
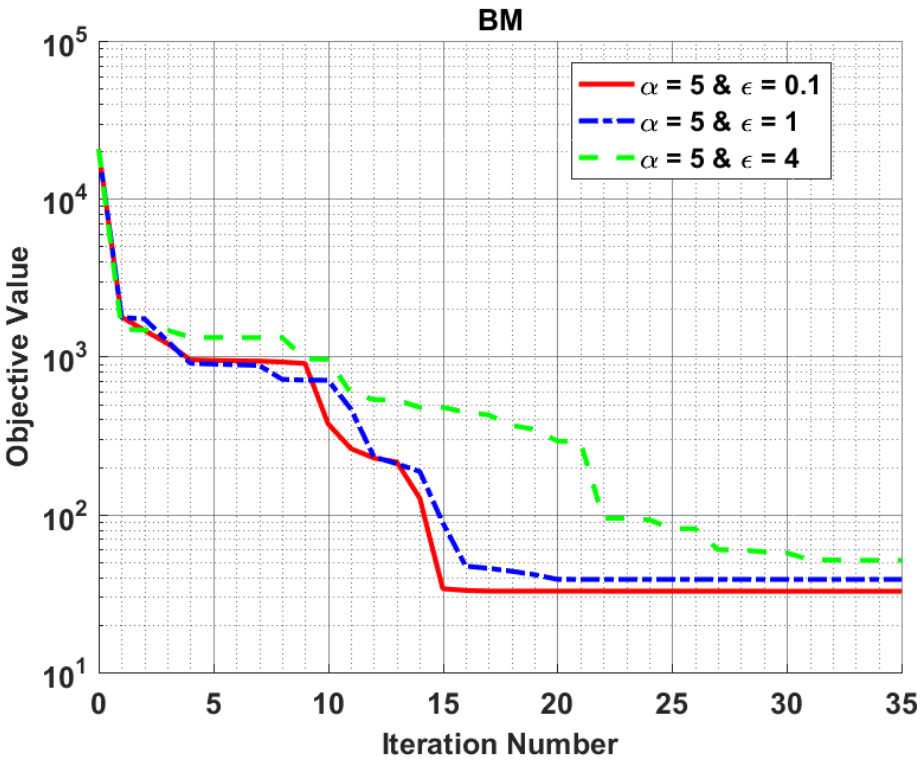
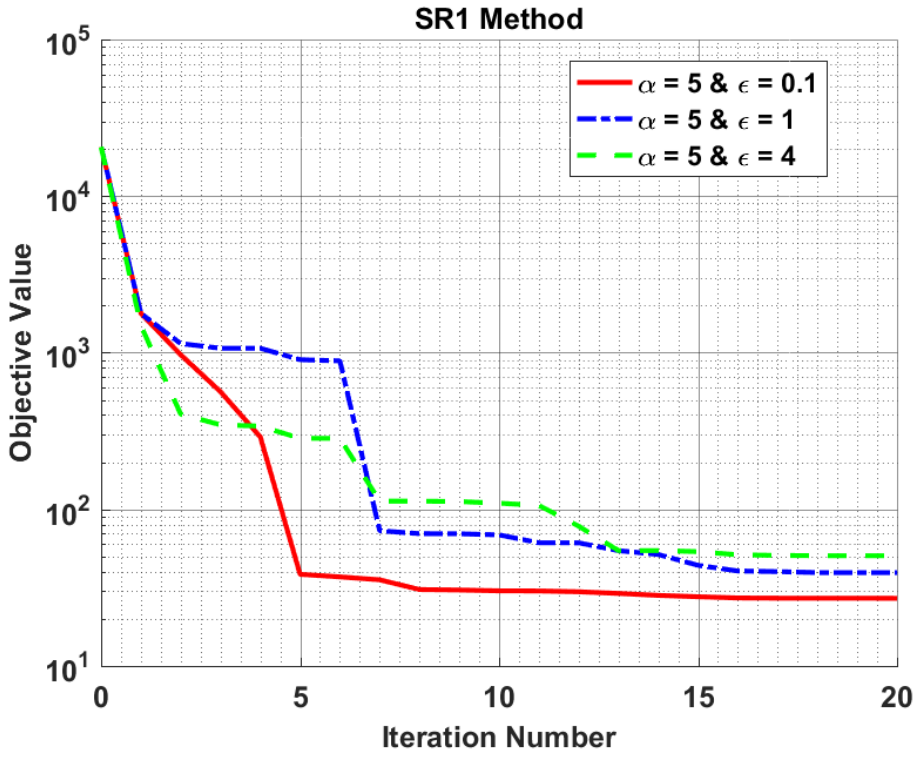


Figure 6.3: Comparison of Objective Change along Time for Perturbation Constant Sensitivity Analysis in Hover to Forward Flight Maneuver


BM
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
Objective Value
Iteration Number
10^5
10^4
10^3
10^2
10^1
0
5
10
15
20
25
30
35


SR1 Method
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
Objective Value
Iteration Number
10^5
10^4
10^3
10^2
10^1
0
5
10
15
20

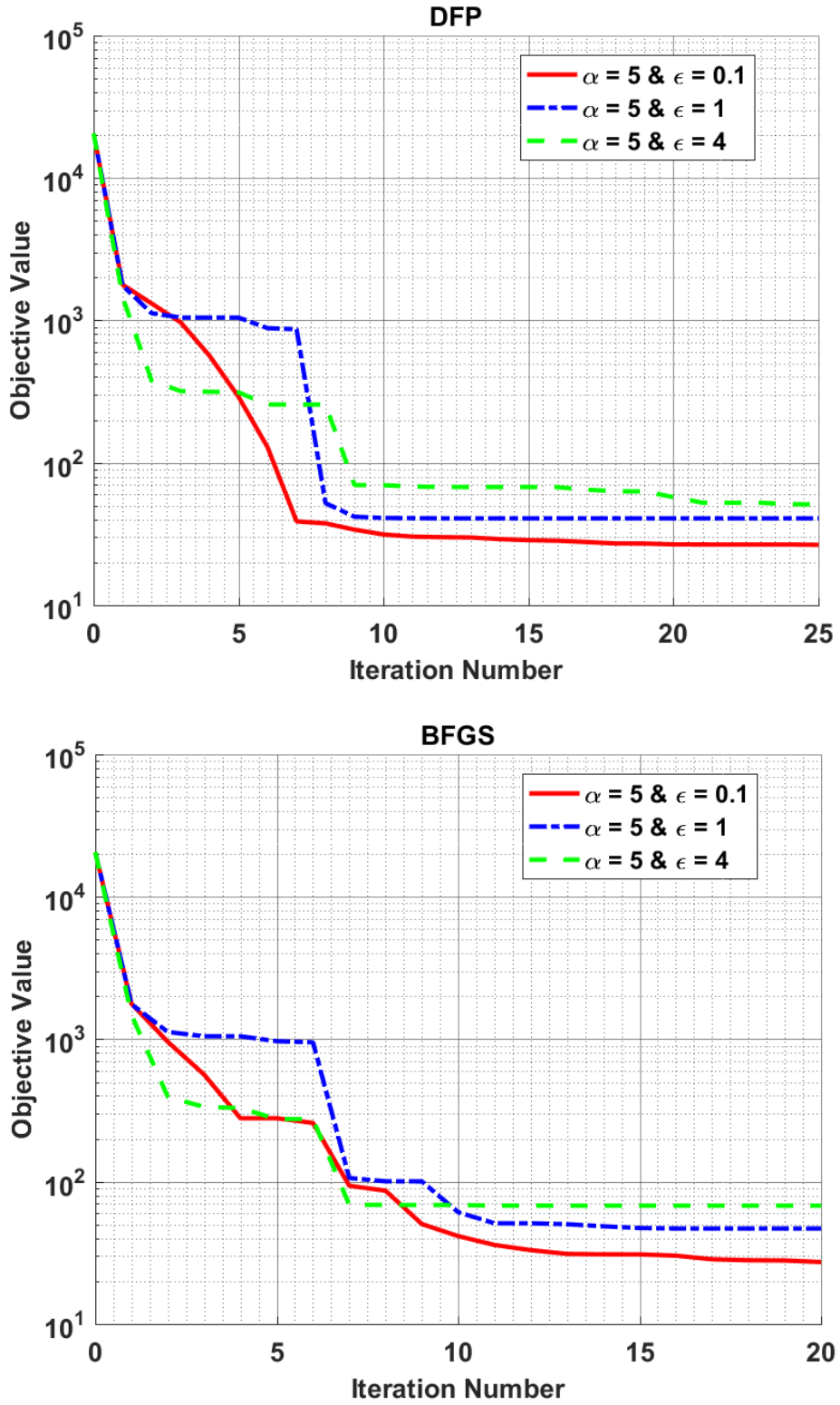
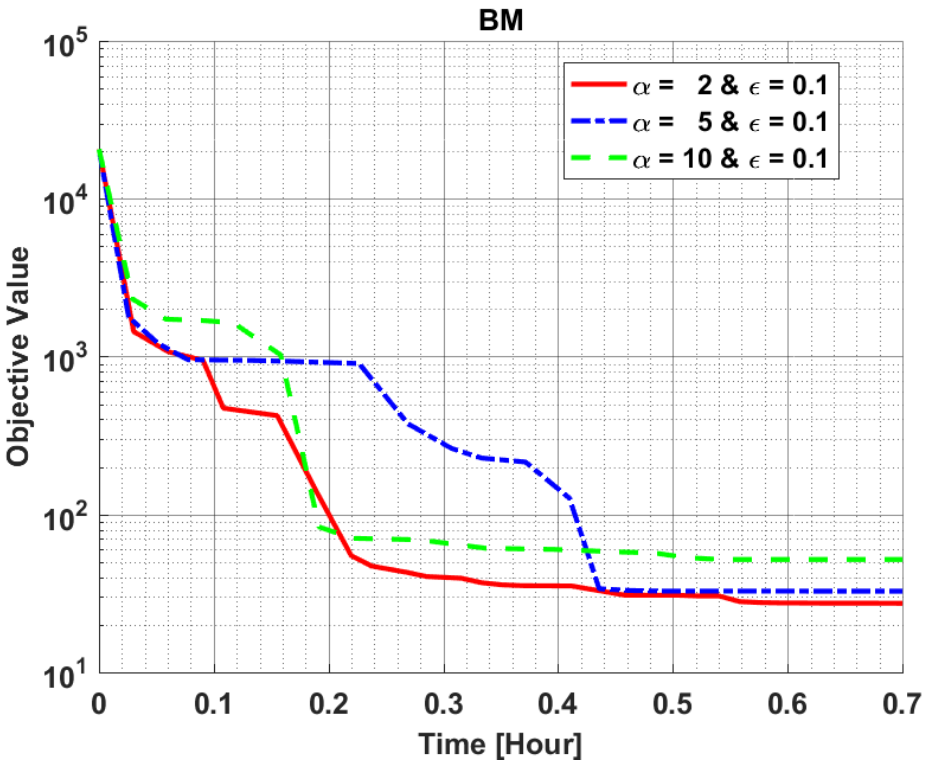


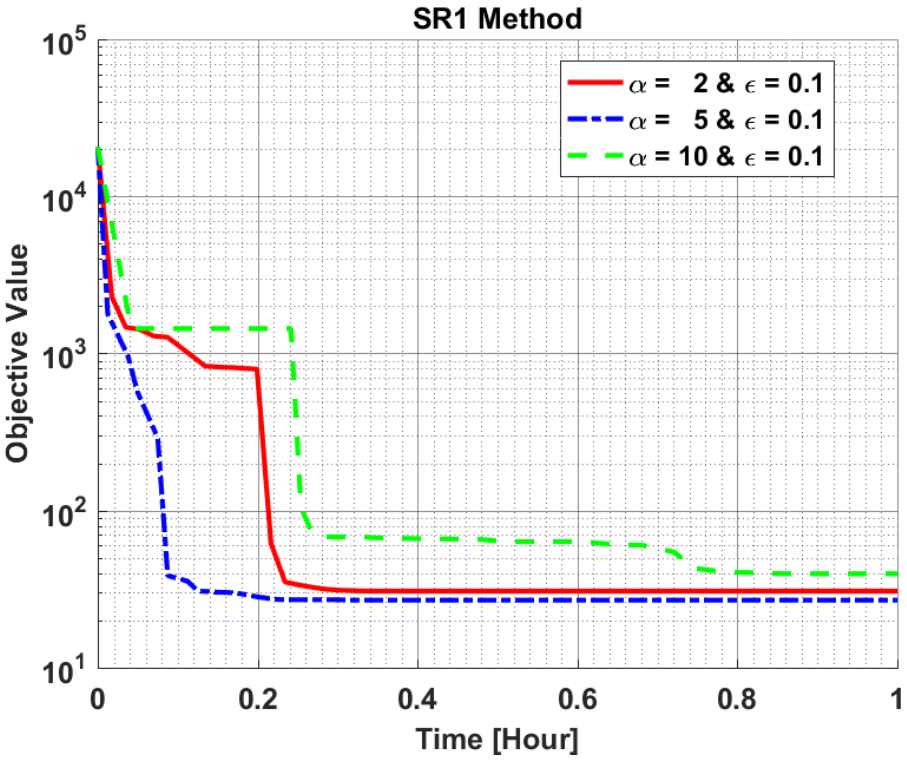
Figure 6.4: Comparison of Objective Change along Iteration Number for Perturbation Constant Sensitivity Analysis in Hover to Forward Flight Maneuver

Figure 6.3 and Figure 6.4 show the effect of perturbation constant on the convergence of the optimization problem for each algorithm. For all these optimization methods, it is clearly seen that while the perturbation constant values of 1 and 4 could not obtain a better solution after a point, the perturbation constant values of 0.1 converged to the lowest objective solution in performed optimization time. Therefore, the most useful value among others is the 0.1 value of perturbation constant for each method. Namely, this value is accepted as the perturbation constant value for hover to forward flight maneuvering problem in all these optimization methods.

For the sensitivity analysis of the perturbation constant, the constraint responses of the helicopter in all configurations and corresponding pilot control inputs are given in Appendix A2.

After determining the most useful perturbation constants, step size sensitivity analysis is performed. All optimization methods are executed with 2, 5 and 10 step size values while the perturbation constant is taken 0.1 and all other optimization parameters are still kept constant. Then, the changes of objective value along both the optimization time and the iteration number for these step size values are obtained as seen in Figure 6.5 and Figure 6.6.


BM
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1
Objective Value
Time [Hour]


SR1 Method
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1
Objective Value
Time [Hour]

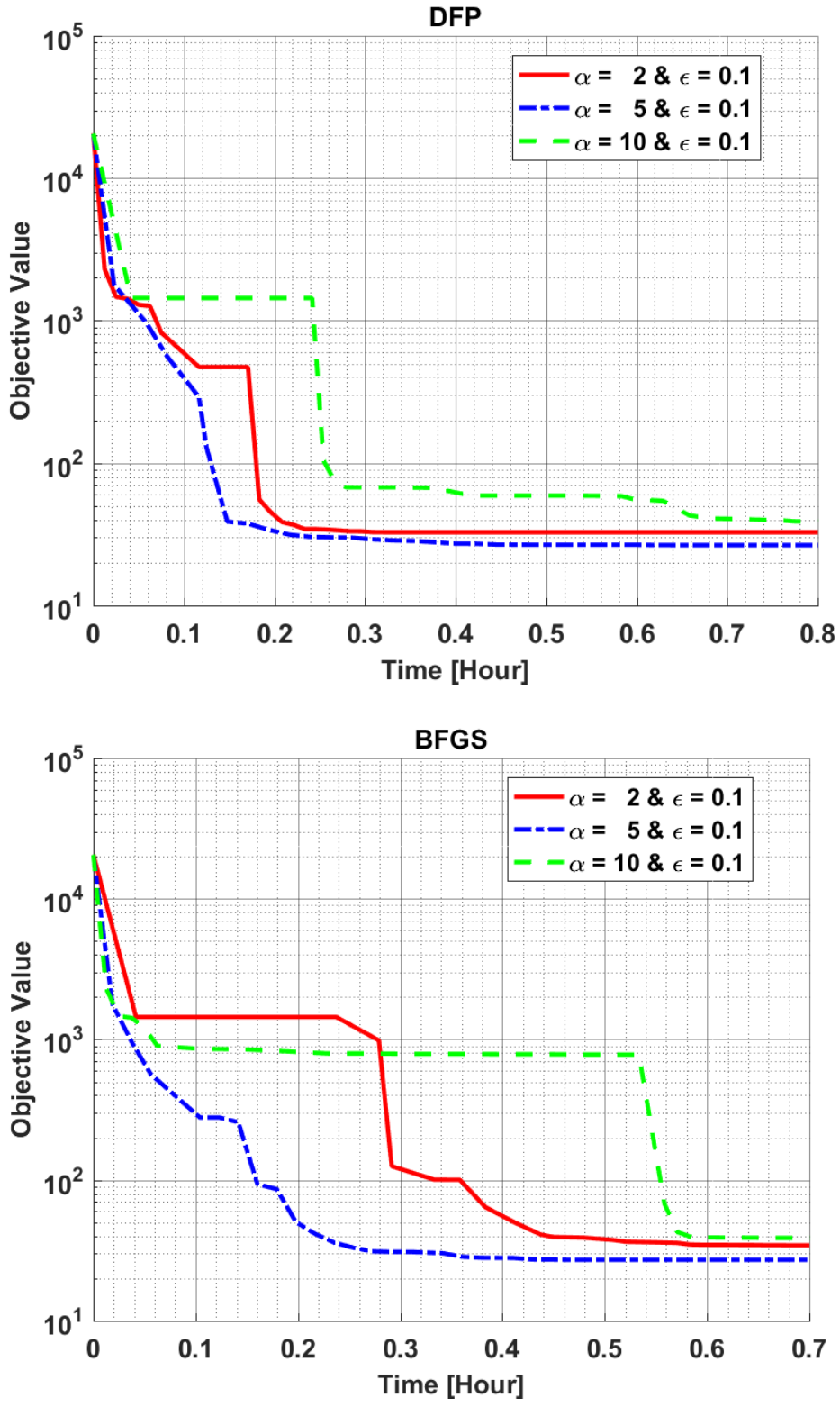
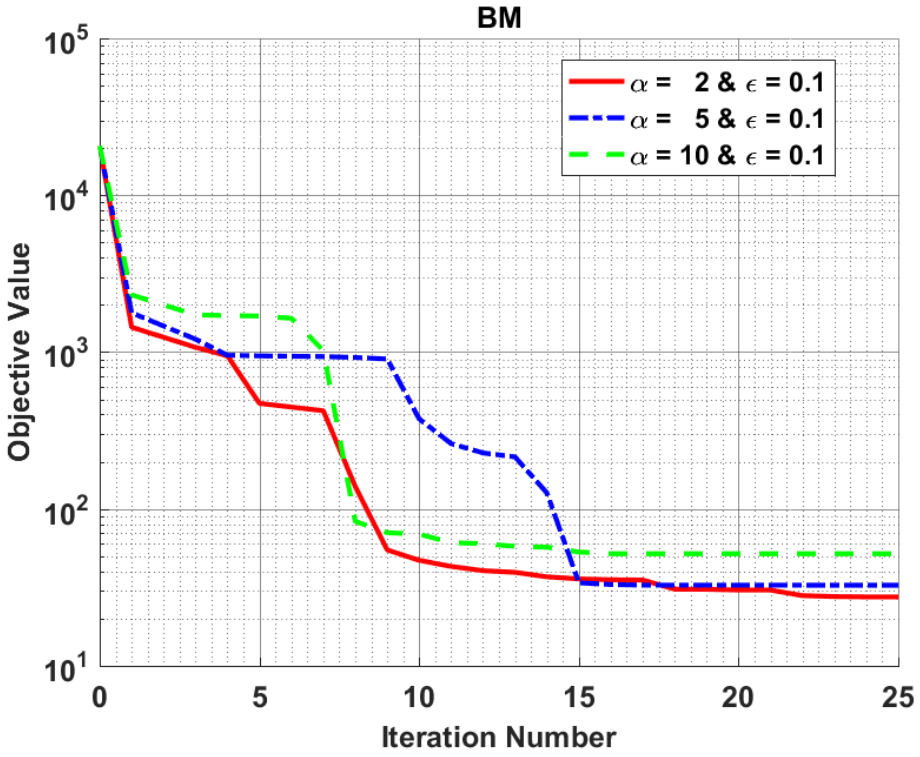
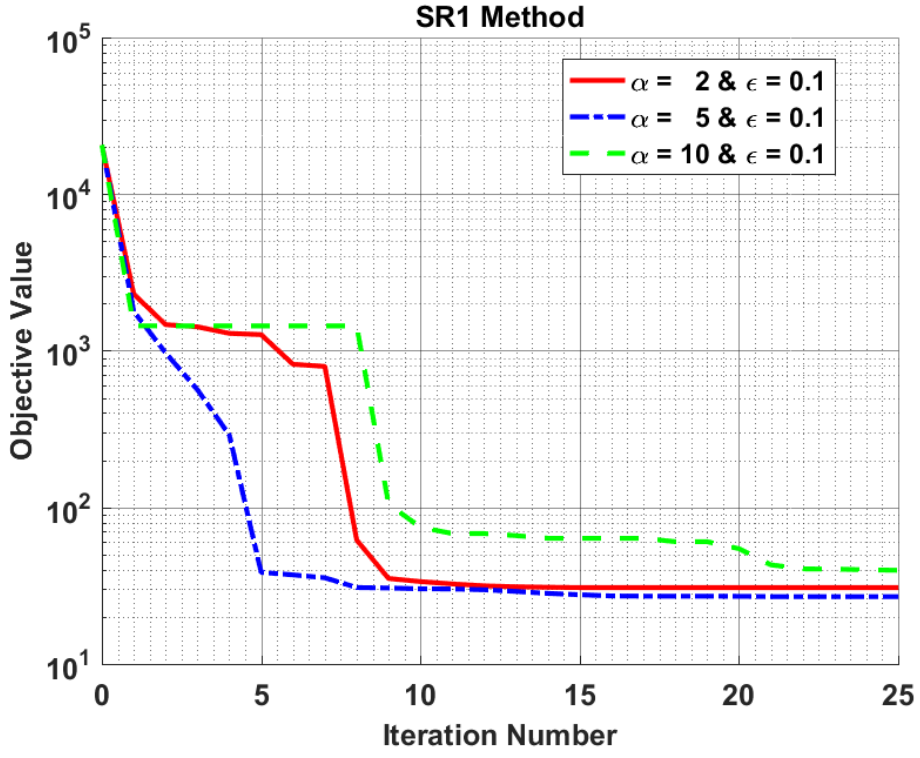


Figure 6.5: Comparison of Objective Change along Time for Step Size Sensitivity Analysis in Hover to Forward Flight


BM
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1
Objective Value
Iteration Number
10^5
10^4
10^3
10^2
10^1
0
5
10
15
20
25


SR1 Method
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1
Objective Value
Iteration Number
10^5
10^4
10^3
10^2
10^1
0
5
10
15
20
25

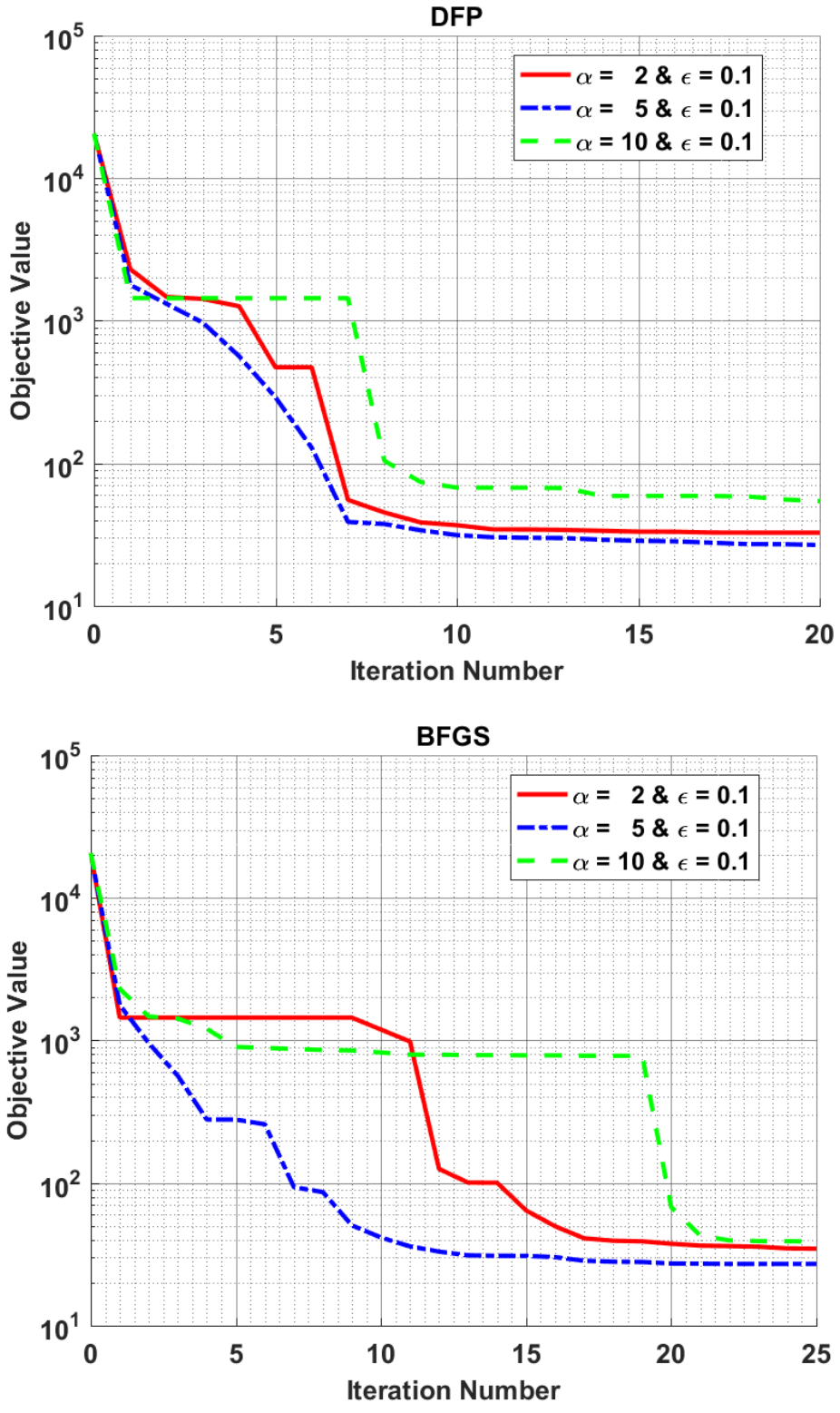


Figure 6.6: Comparison of Objective Change along Iteration Number for Step Size Sensitivity Analysis in Hover to Forward Flight

Figure 6.5 and Figure 6.6 show the effect of the step size parameter on the optimization problem convergence for each optimization algorithm. This examinations shows that the step size value of 5 converges faster to the lowest objective function in a shorter time and smaller iteration number for all optimization methods except Broyden's Method. In Broyden's Method, this value appears to be 2. Therefore, while the step size value of 2 is taken as the most useful value among the others for Broyden's Method, this value is taken as 5 for other optimization methods. Namely, these values are accepted as the most useful step size values for hover to forward flight maneuvering problem in these optimization methods.

For these sensitivity analyses, the constraint responses of the helicopter in all configurations and corresponding pilot control inputs are given in Appendix A3.

For hover to forward flight maneuver optimization under defined conditions, it is concluded that the central difference approximation is the most useful gradient computation method for all optimization methods. In addition, SR1, DFP and BFGS methods for this optimization problem have the best combination of step size and perturbation constant at $\alpha = 5$ and $\varepsilon = 0.1$. However, BM has the best combination of step size and perturbation constant at $\alpha = 2$ and $\varepsilon = 0.1$.

6.3.1.5 Comparison of the Optimization Methods for Hover to Forward Flight Maneuvering

For hover to forward flight maneuver optimization, the central difference approximation is decided as the gradient calculation method in Section 6.3.1.3. Then, the sensitivity analysis of perturbation constant and step size parameters are performed for each selected optimization method. Thus, the most effective combination of perturbation constant and step size is obtained for each one. In this section, the best combination solution obtained from each method is compared in order to decide the most effective method for this maneuvering problem. Therefore, the best solutions of the methods are shown in Figure 6.7 and Figure 6.8.

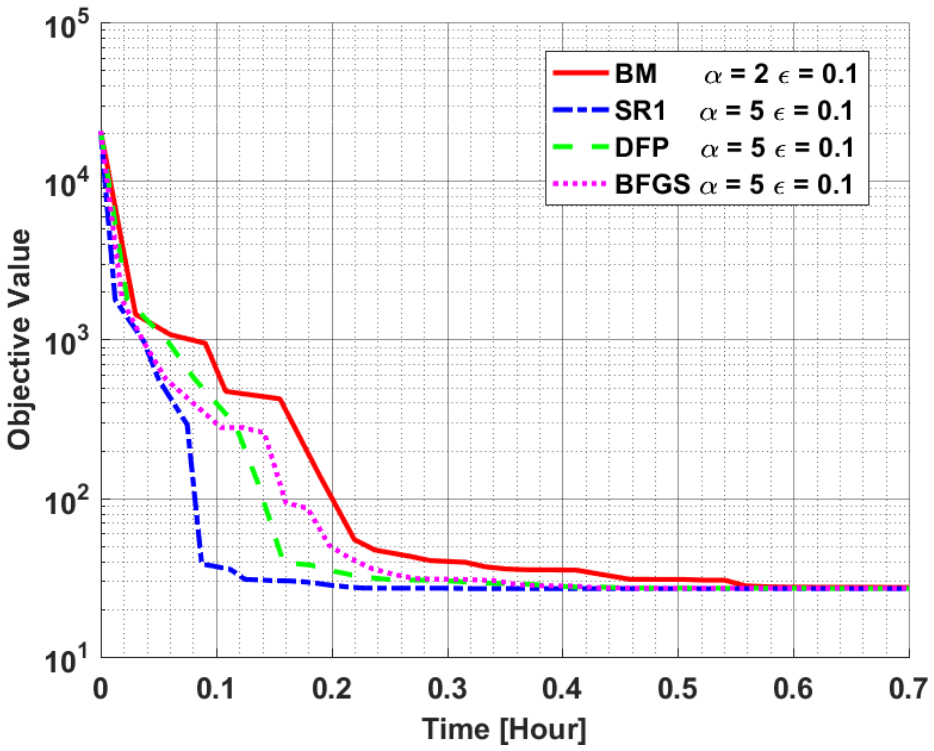


Figure 6.7: Comparison of Objective Change along Time for the selected quasi-Newton Methods in Hover to Forward Flight Maneuver Optimization

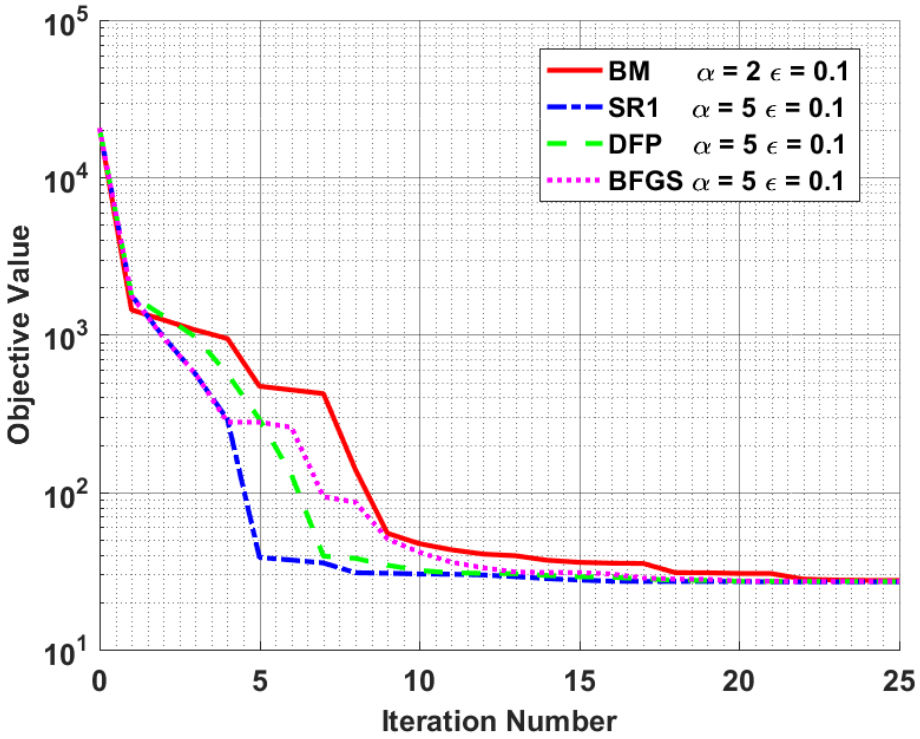
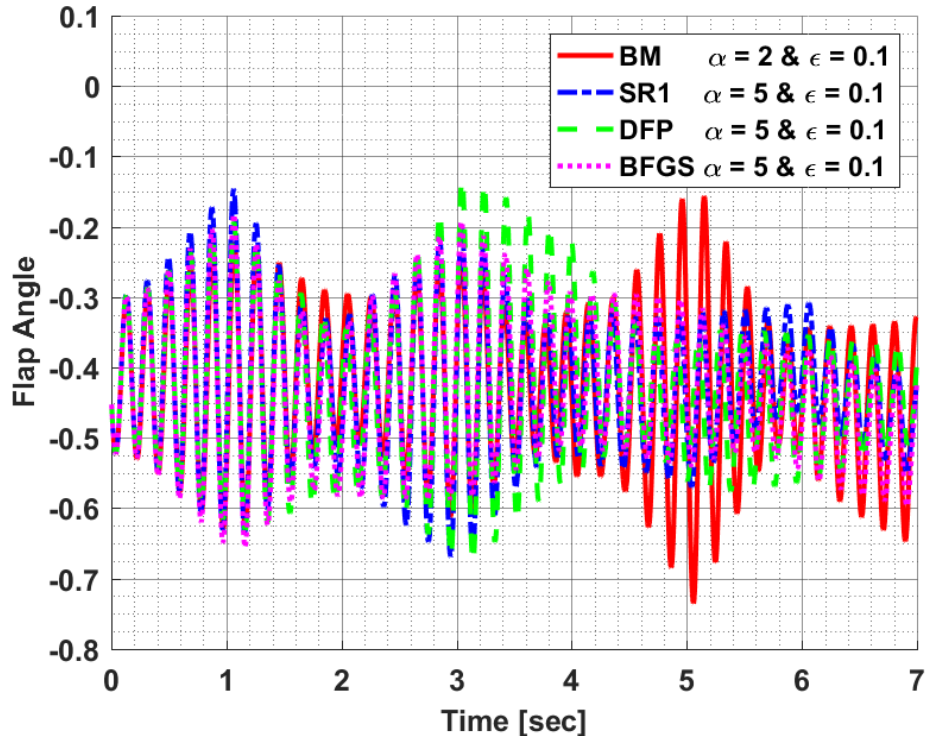


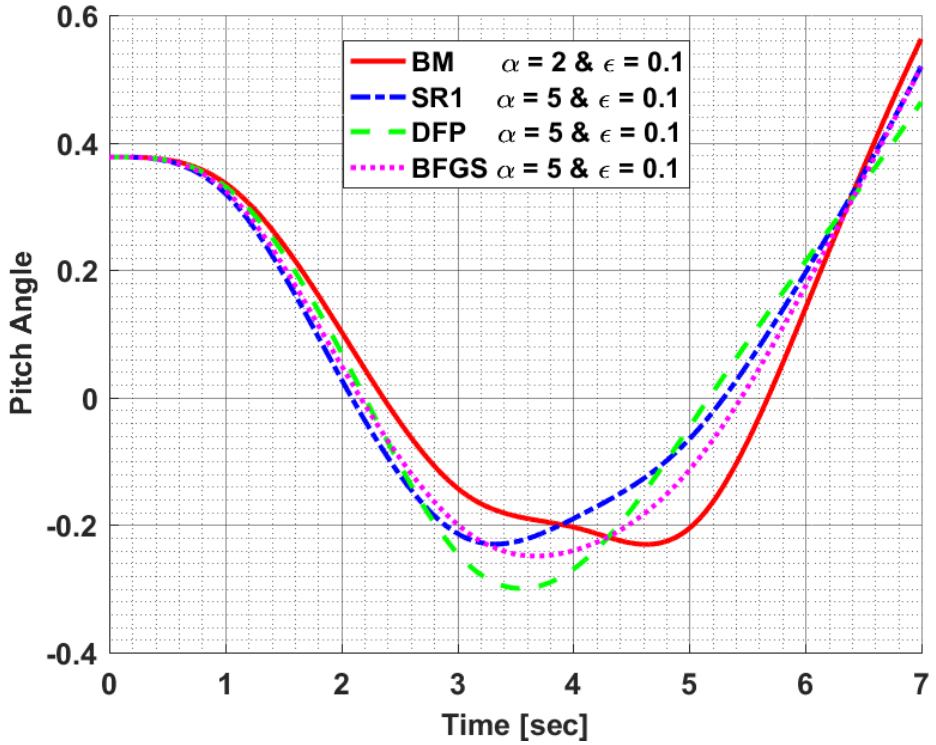
Figure 6.8: Comparison of Objective Change along Iteration Number for the selected quasi-Newton Methods in Hover to Forward Flight Maneuver

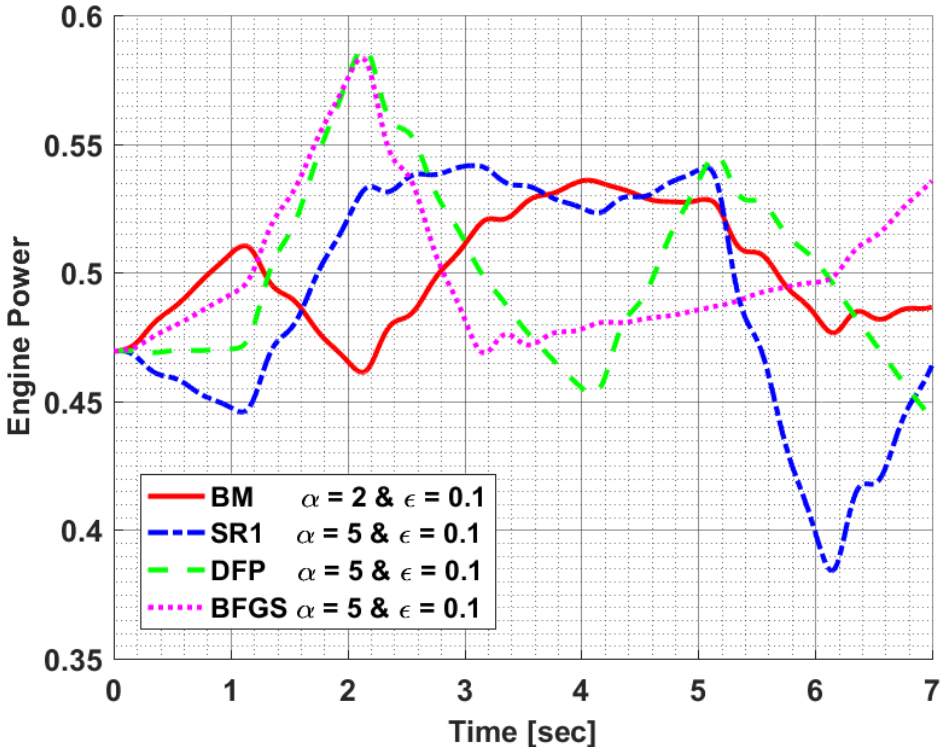
It is clearly seen from these figures that although the same objective value is reached at the end of all methods, their convergence speeds are different. Therefore, both the convergence time and the iteration number are defined as the main parameter to evaluate the efficiency of the methods.

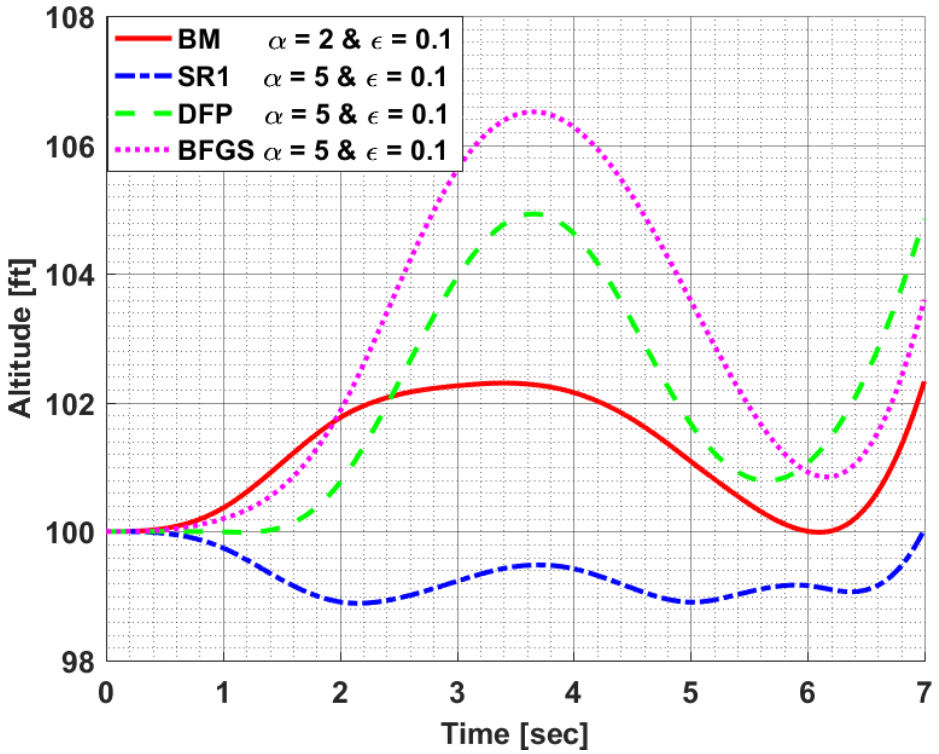
Figure 6.7 and Figure 6.8 show that the method reaching the target at the latest is the Broyden's Method. After the BM, the slowest method is the BFGS method. According to Figure 6.7 and Figure 6.8, the fastest method among the other is SR1 method. Therefore, it is concluded that the symmetric rank-one method is the most useful method among others for hover to forward flight maneuver optimization under defined conditions.

In Figure 6.9, the constraint results of hover to forward flight maneuver are plotted for each method. These charts show if the constraints are met or not. Moreover, while the constraints of flap angle and pitch angle are scaled from -1 to 1, the engine power constraint is scaled from 0 to 1. In other words, the minimum and maximum limits of flap and pitch angles are represented by the value of -1 and 1, respectively. Similarly, the value of 1 in the engine power chart represents the maximum engine power value. Afterwards, the values between the parameter limits are scaled according to these ranges. However, there is no scaling for forward velocity and altitude charts.


BM α = 2 & ϵ = 0.1
SR1 α = 5 & ϵ = 0.1
DFP α = 5 & ϵ = 0.1
BFGS α = 5 & ϵ = 0.1
Flap Angle
Time [sec]


BM α = 2 & ϵ = 0.1
SR1 α = 5 & ϵ = 0.1
DFP α = 5 & ϵ = 0.1
BFGS α = 5 & ϵ = 0.1
Pitch Angle
Time [sec]


Engine Power
Time [sec]
BM α = 2 & ϵ = 0.1
SR1 α = 5 & ϵ = 0.1
DFP α = 5 & ϵ = 0.1
BFGS α = 5 & ϵ = 0.1


Altitude [ft]
Time [sec]
BM α = 2 & ϵ = 0.1
SR1 α = 5 & ϵ = 0.1
DFP α = 5 & ϵ = 0.1
BFGS α = 5 & ϵ = 0.1

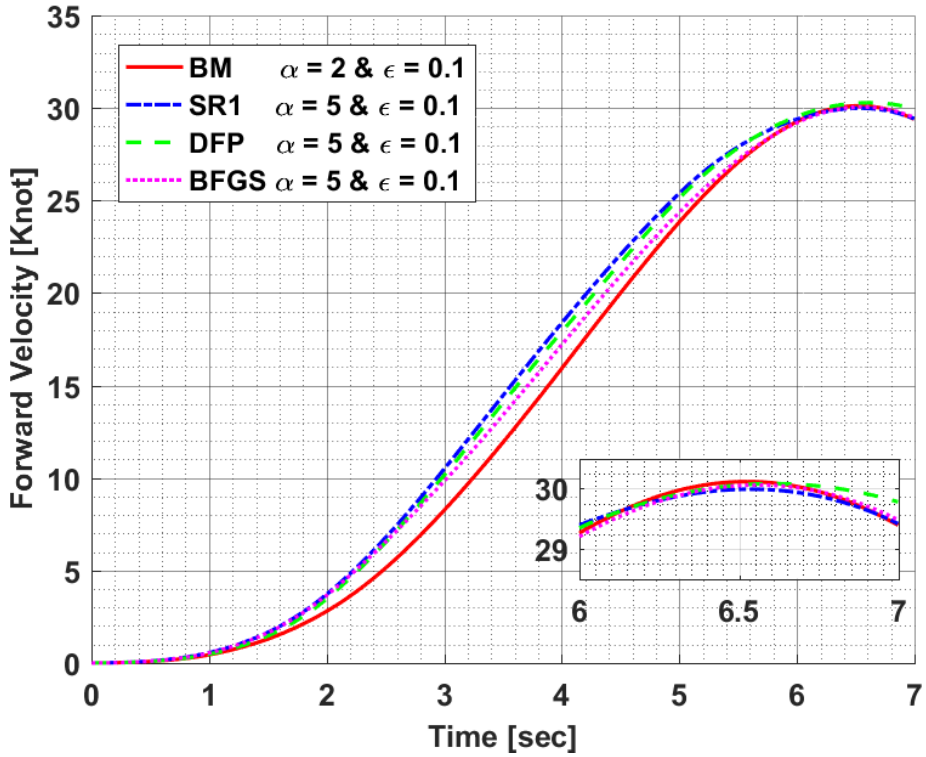


Figure 6.9: Constraint Results of Hover to Forward Flight Maneuver in Optimization Methods

The charts in Figure 6.9 show that each optimization method fulfilled all the constraints of hover to forward flight maneuver. It is seen from the forward velocity graph that all optimization methods have reached almost the target value within the specified time interval. In addition, the pitch angle graph says that the helicopter's nose is turned down during maneuvering. This is an expected result because the thrust force increases in the forward direction so that the helicopter accelerates in this direction. Moreover, although other constraint results varied, they remained within the specified limits.

Since the methods have different solution paths, they got different optimization results. However, they still met at almost the same objective value due to the high coupling between pitch, roll and yaw motion of the helicopter. Namely, a helicopter can perform the same maneuver with different pilot control inputs.

Figure 6.10 shows the required longitudinal and collective cyclic inputs which are defined as design variables in this maneuvering problem to be able to perform hover to forward flight maneuver whose results are given in Figure 6.9.

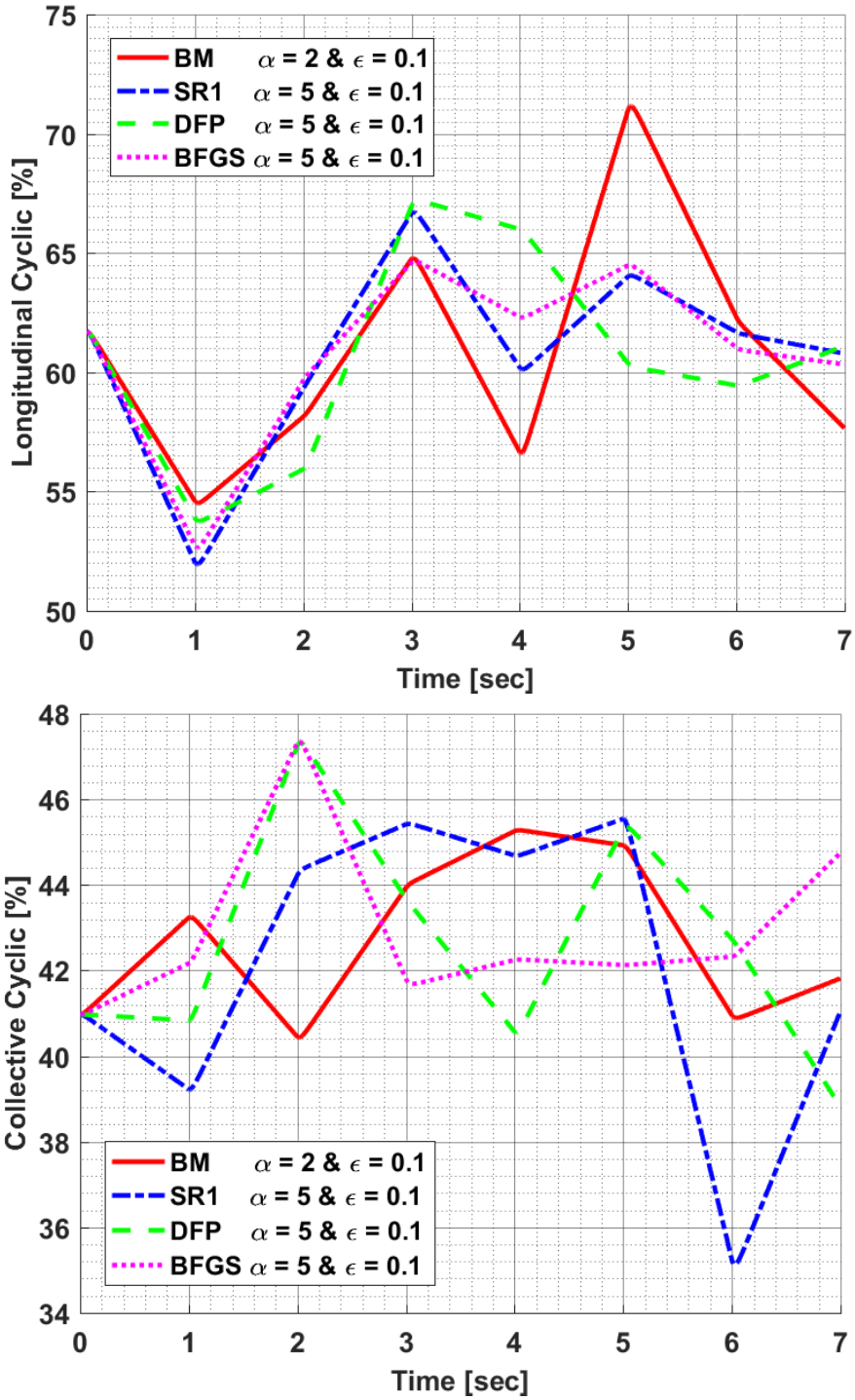


Figure 6.10: Obtained Design Variable in Hover to Forward Flight Maneuvering Problem

6.3.2 Hover to Sideward Flight Maneuvering with Optimization Methods

6.3.2.1 Hover to Sideward Flight

In hover to sideward flight maneuvers, a rotorcraft comes to hover position and then continues with its lateral movement. These maneuvers target to reach a constant speed and keep the process in almost the same altitude and heading.

How can it be performed [54]?

1- Bring helicopter to hover position at a specific altitude then move the lateral cyclic in the desired sideward direction to accelerate the helicopter in that direction
2- Apply necessary collective cyclic inputs to avoid any possible altitude changes
3- Apply necessary anti-torque pedal inputs to avoid any possible excessive heading changes
4- Complete the maneuver as the desired lateral speed is reached
5- All these parameters must be implemented at the same time to generate maneuver properly

6.3.2.2 Optimization Modeling of Hover to Sideward Flight Maneuver

The main pilot control inputs for sideward acceleration maneuvers are lateral and collective cyclic as mentioned in Section 6.3.2.1. Therefore, these control inputs are defined as design variables in this maneuver. Besides, the longitudinal cyclic and anti-torque pedals are controlled by SAS, so they are not modified by the optimization algorithm. Like hover to forward flight maneuver, this optimization of the maneuver also takes about 7 seconds. Moreover, the pilot control inputs, which

are lateral and collective cyclic, are modified at intervals of every second. Therefore, the design variable vector consists of $7 * 2 = 14$ design points as

$$\boldsymbol{X}^T = [\delta_{co}(t=1), \delta_{co}(t=2), \dots \delta_{co}(t=7), \delta_{la}(t=1), \delta_{la}(t=2), \dots \delta_{la}(t=7)]$$

As it is mentioned in Section 6.3.1.2 that the helicopter design limits and maneuver requirements should be taken into account to create the constraints of the optimization problem. To create these constraints, the rightward flight is chosen as a sideward flight maneuver. In this maneuver, the main purpose is to reach the speed of 30 Knot rightward velocity while maintaining the almost constant altitude of the helicopter. Moreover, the helicopter design limitations should also be included as constraints. Therefore, Constraints 1, 3 and 5 are accepted the same as in Section 6.3.1.2. In this maneuver, the lateral cyclic is used as a design variable, so the rolling angle should be kept within the design limits. Therefore, Constraint 2 is created to limit the rolling angle. Moreover, Constraint 4 is similar to hover to forward flight maneuver, but this maneuver targeted to rightward flight. Therefore, Constraint 4 in this maneuver is modified according to the requirements of this maneuver. Hence, the Constraints 2 and 4 of this maneuver optimization problem are adjusted as:

Constraint 2 – Roll Angle Constraint

The roll angle is the angle to be turned around the longitudinal axis of an airplane to bring its lateral axis to a horizontal plane. All helicopters have the maximum and minimum roll angle limitations where they can fly. These limitations result from aerodynamic design. If these limit points are exceeded, the helicopter enters into stall condition and loses its lifting force. Therefore, this angle should be limited in constraints when the lateral cycle is used as a design variable. To prevent the stall condition, this limitation is also formulated with the logarithmic barrier function as

$$g_2 = \begin{cases} -log(\emptyset_{max} - max(\emptyset)) - log(\emptyset_{min} + min(\emptyset)) & for \ \ max(\emptyset) < \emptyset_{max} \ and \ min(\emptyset) > \emptyset_{min} \\ Infinity & for \quad otherwise \end{cases}$$

where the roll angle of the helicopter is represented by ∅. Also, the maximum and minimum roll angle limits are represented with $\emptyset_{max}$ and $\emptyset_{min}$ respectively.

Constraint 4 – Rightward Velocity Constraint

In this maneuver, it is desired that the helicopter reaches the 30 Knot rightward velocity between 6th and 7th seconds. The constraint will be met when the desired velocity is reached and the helicopter no longer accelerates. This target is one of the maneuvering requirements. This constraint is defined formulated as

$$g_4 = \int_{t_s=6.3}^{t_f=6.7} |V - 30|^2 \, dt$$

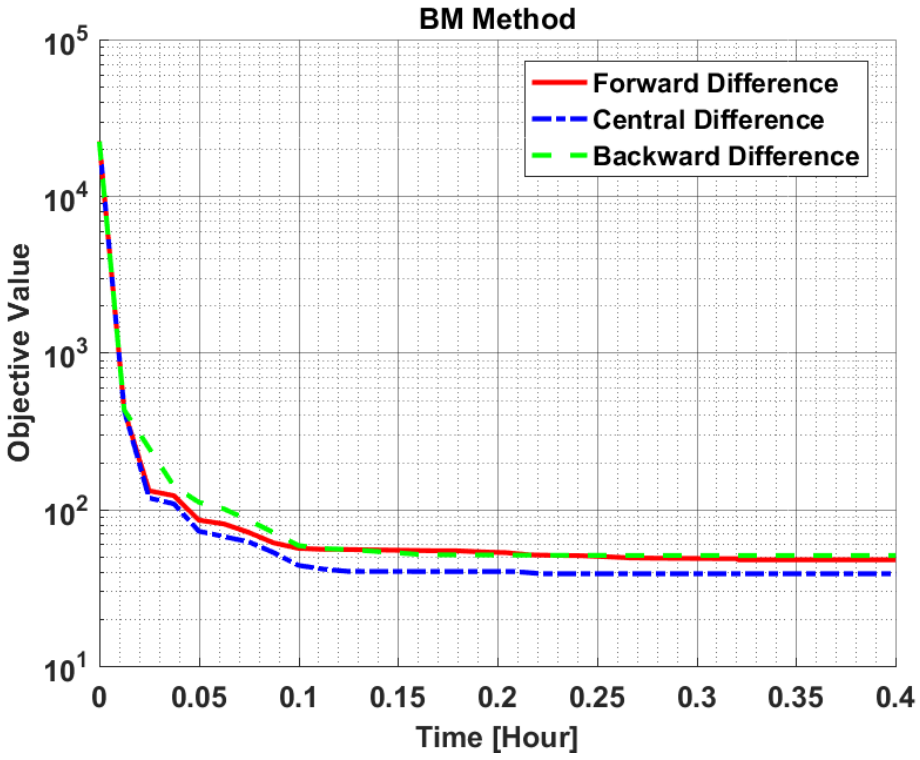
where the helicopter lateral velocity is represented by $\boldsymbol{V}$ and is considered a positive sign to the right. Furthermore, the upper and lower time boundaries where the targeted parameter is desired to be achieved are represented with $\boldsymbol{t_s}$ and $\boldsymbol{t_f}$, respectively.

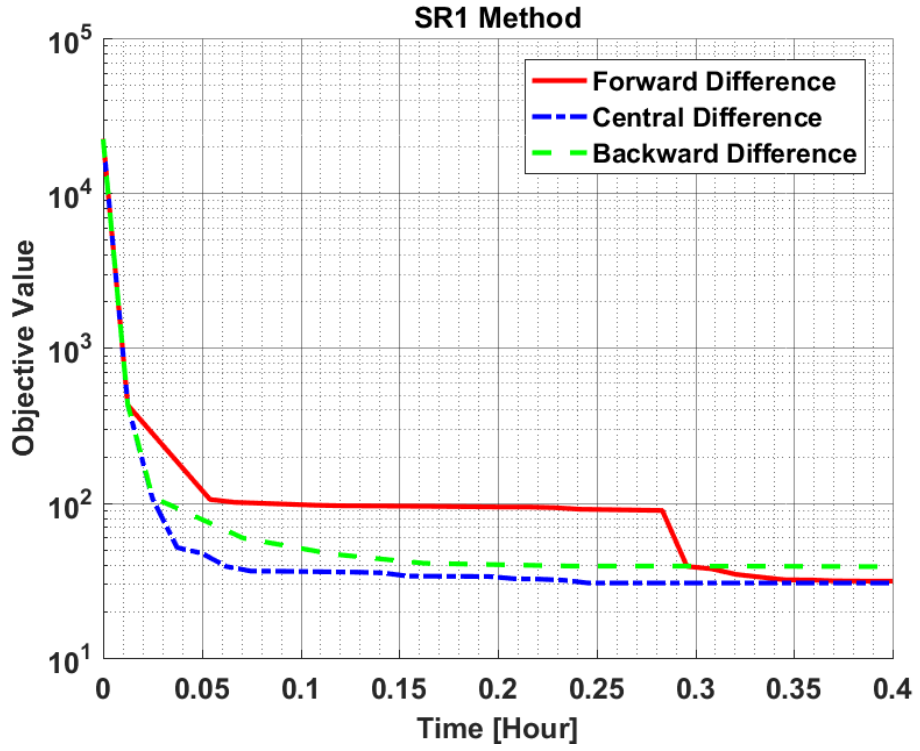
Finally, the objective function is created in exactly the same way as described in Section 6.3.1.2. In addition, the same penalty parameter values are used in this objective function.

6.3.2.3 Decision of the Finite Divided Difference Approximation for Hover to Rightward Flight Maneuver

In hover to rightward flight maneuver optimization, forward, central and backward difference approximations are applied to each optimization method. Afterwards, the solution of these difference approximations is compared to decide the most effective gradient calculation method. To compare the finite divided difference approximations, the maneuver is carried out with constant optimization parameter values. The same parameter values described in Section 6.3.1.3 are used in this maneuver. Then, the change of the objective function along both the optimization

time and the iteration number is plotted for all methods as seen in Figure 6.11 and Figure 6.12.





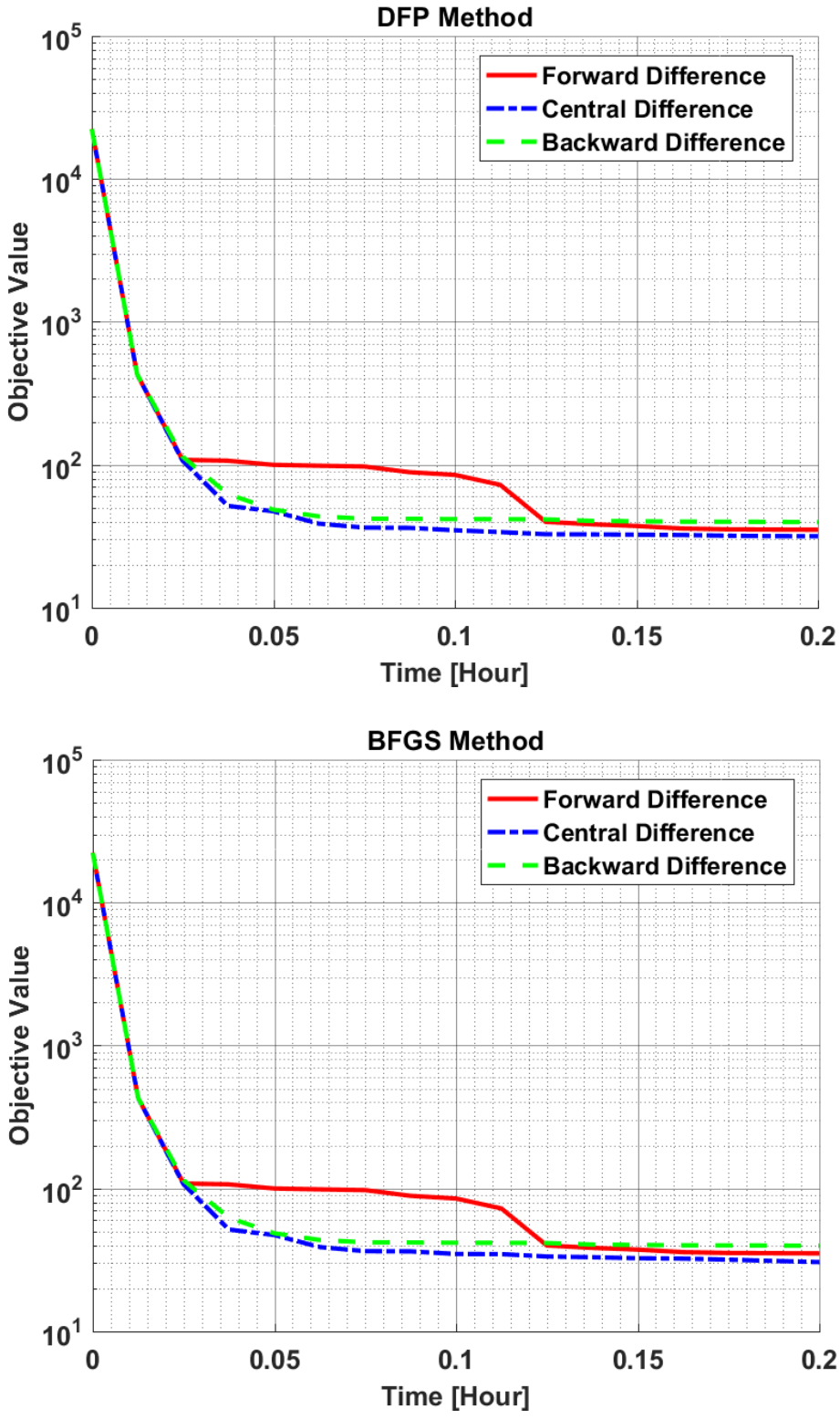
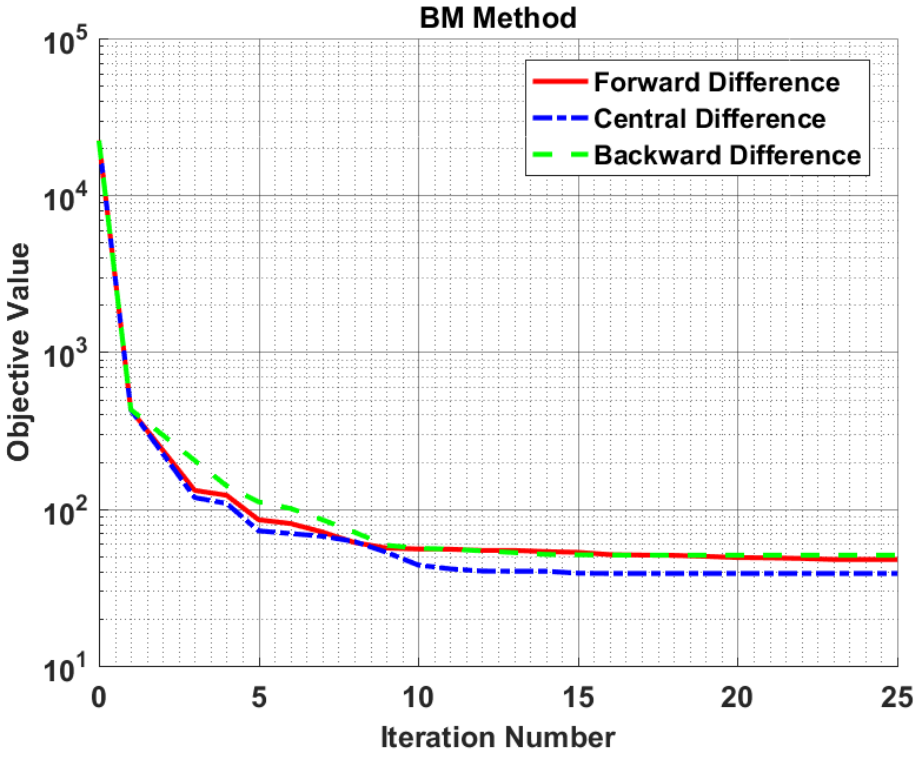
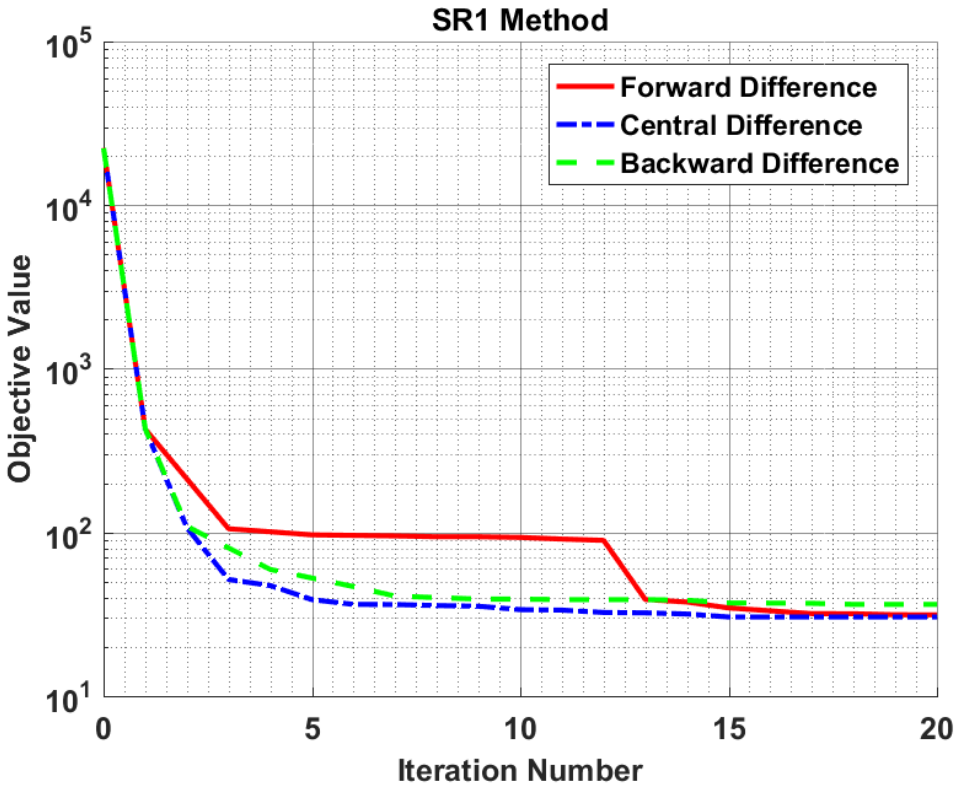


Figure 6.11: Comparison of Objective Change along Time for the Finite Divided Difference Approximations in Hover to Rightward Flight Maneuver


BM Method
Forward Difference
Central Difference
Backward Difference
Objective Value
Iteration Number
10^5
10^4
10^3
10^2
10^1
0
5
10
15
20
25


SR1 Method
Forward Difference
Central Difference
Backward Difference
Objective Value
Iteration Number
10^5
10^4
10^3
10^2
10^1
0
5
10
15
20

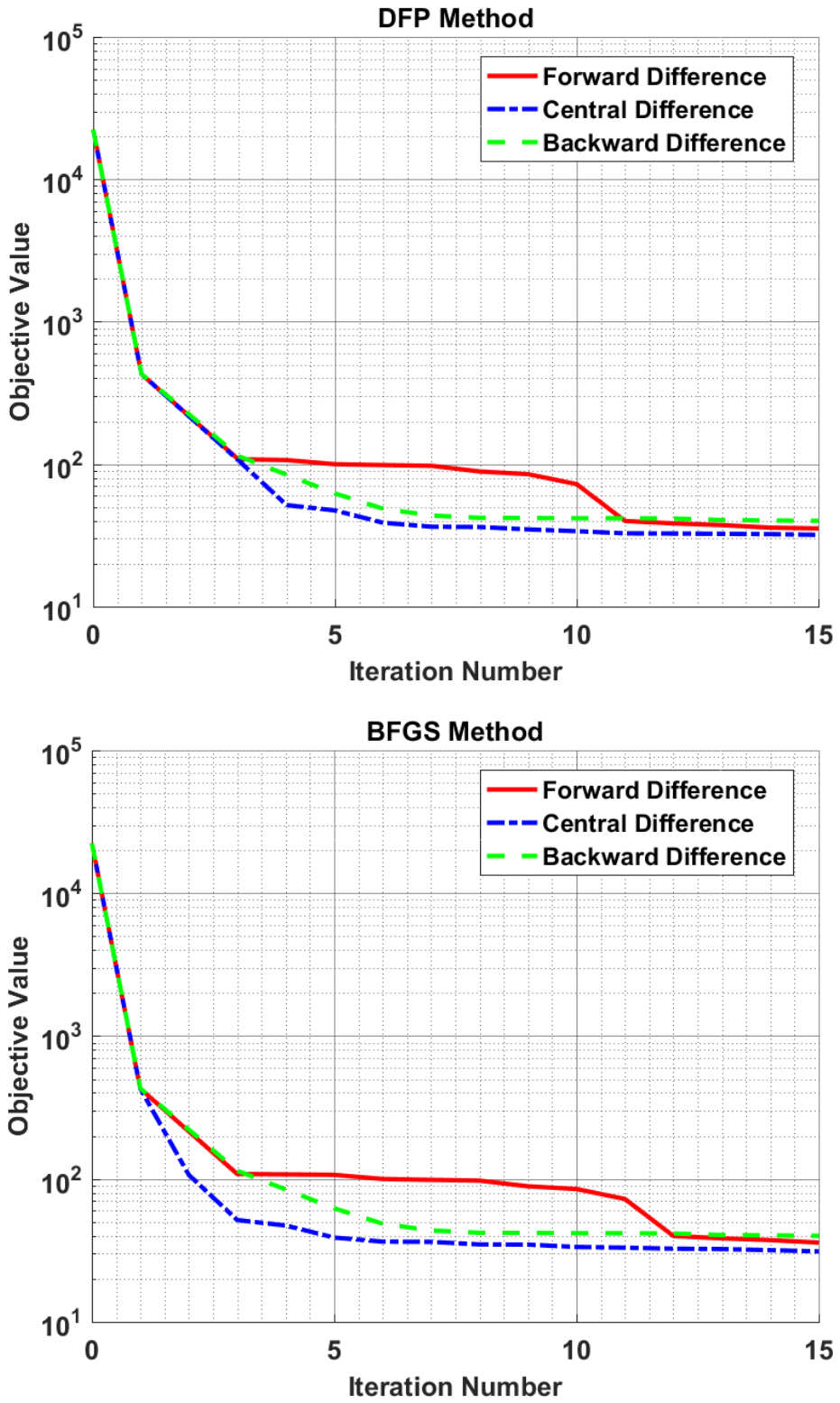


Figure 6.12: Comparison of Objective Change along Iteration Number for the Finite Divided Difference Approximations in Hover to Rightward Flight Maneuver

Figure 6.11 and Figure 6.12 show that the finite divided difference approximations directly affect the convergence of the maneuver optimization for each algorithm. Moreover, each chart starts from the same objective because the same initial design variables are used for each solution.

The comparison process of the finite divided difference approximations reveals that the backward difference is the slowest approximation while the central difference is the fastest one for all optimization algorithms. Therefore, the central difference approximation is accepted as the most effective gradient computation method for hover to rightward flight maneuver problem in all these optimization methods.

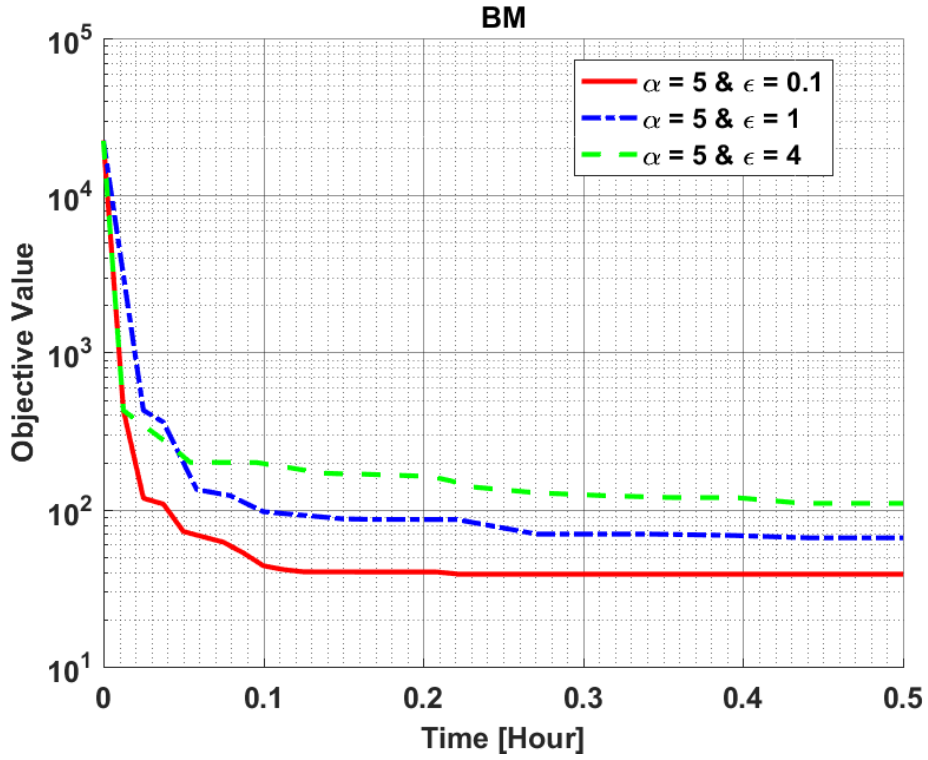
For all unique solution in this comparison process, the constraint responses of the maneuver and corresponding pilot control inputs are given in Appendix B1.

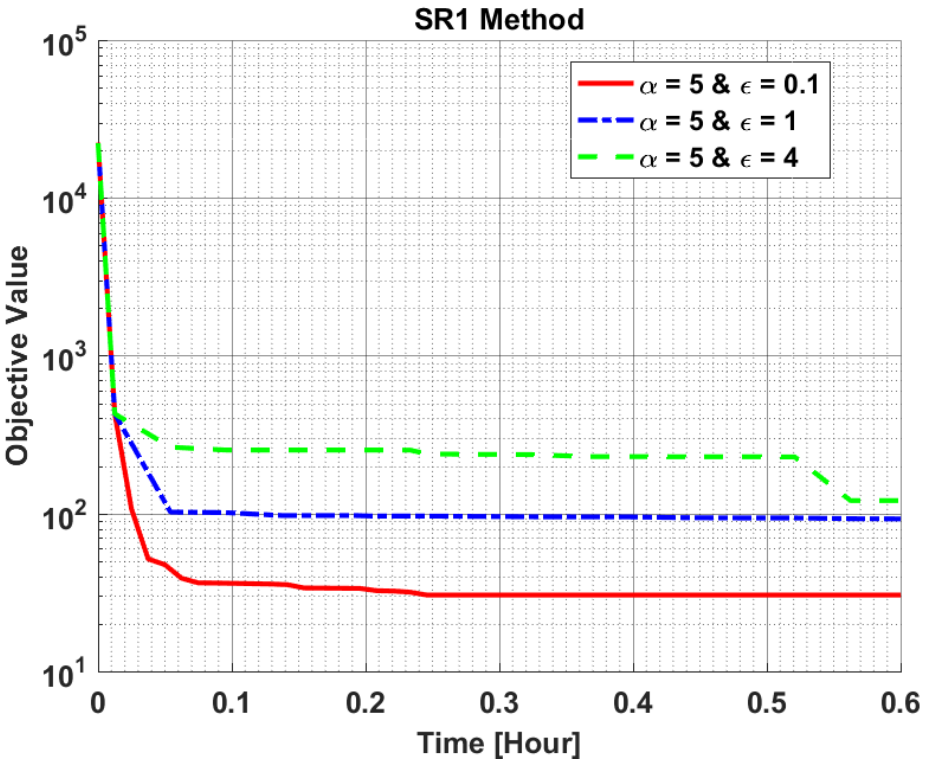
6.3.2.4 Sensitivity Analysis of Step Size and Perturbation Constant Parameters for Hover to Rightward Flight Maneuver

The comparison procedure and all explanations about sensitivity analysis of step size and perturbation constant parameters are described in Section 6.3.1.4. All these descriptions are valid for this section. In this maneuver, the same step size values which are 2, 5 and 10 are used. Similarly, the values of 0.1, 1 and 4 are taken as perturbation constant values. Moreover, all unique analyses are carried out using the central difference approximation which is set out in Section 6.3.2.3. For the same reason as in Figure 6.3 and Figure 6.4, each figure in this subchapter also starts from the same initial objective value.

The sensitivity analysis of perturbation constants is also performed here for all selected optimization methods. The same procedure as described in Section 6.3.1.4 is followed for the perturbation constant sensitivity analysis. Therefore, while the optimization methods are applied for 0.1, 1 and 4 perturbation constant values, the step size value is taken 5 and all other optimization parameters are kept constant. Then, the changes of objective value along both the optimization time and the

iteration number for these perturbation constant values are obtained as seen in Figure 6.13 and Figure 6.14.





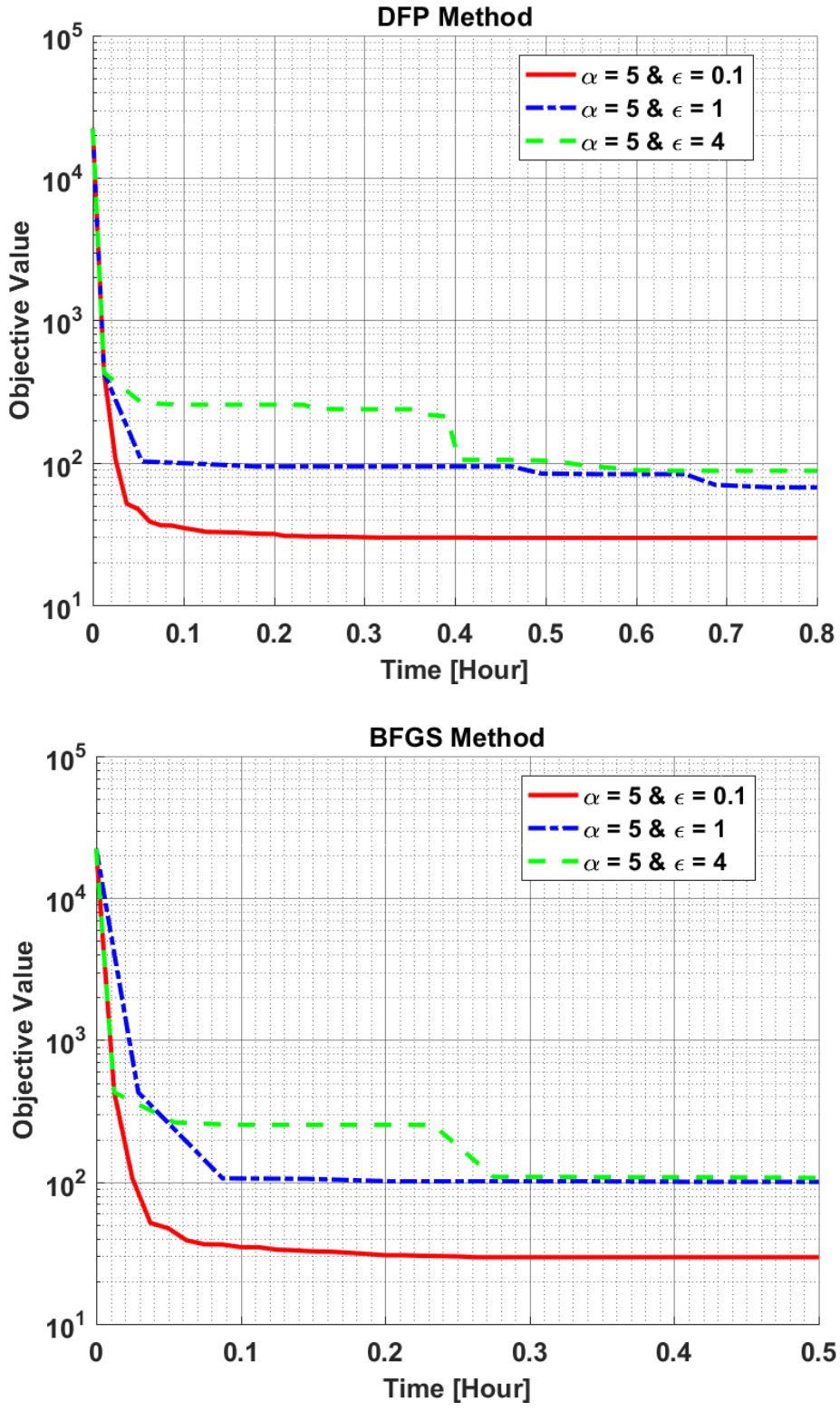
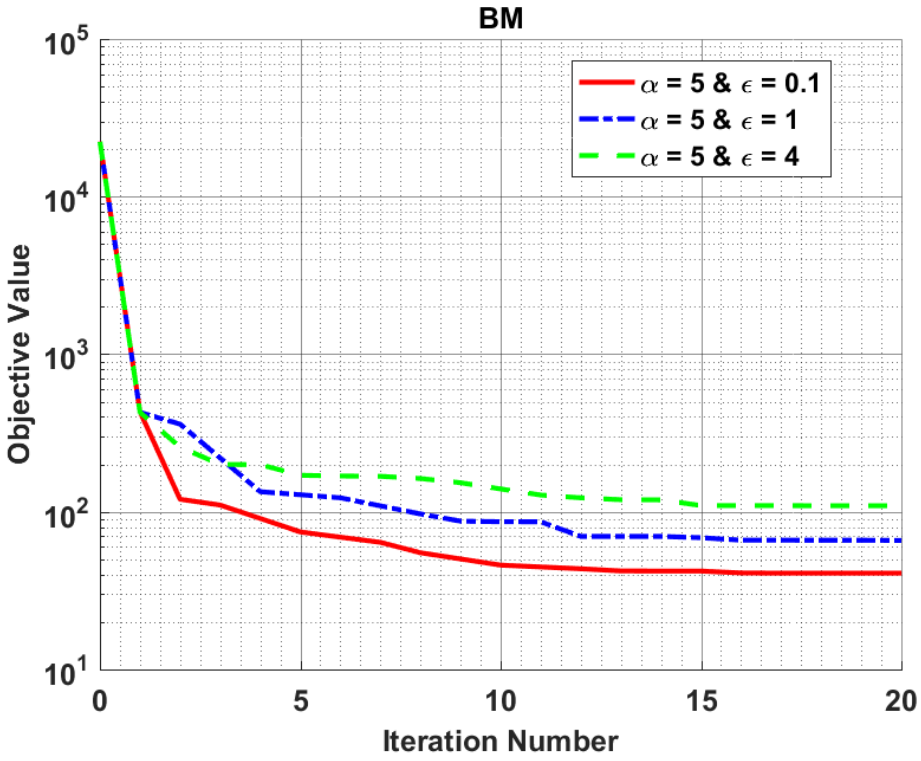
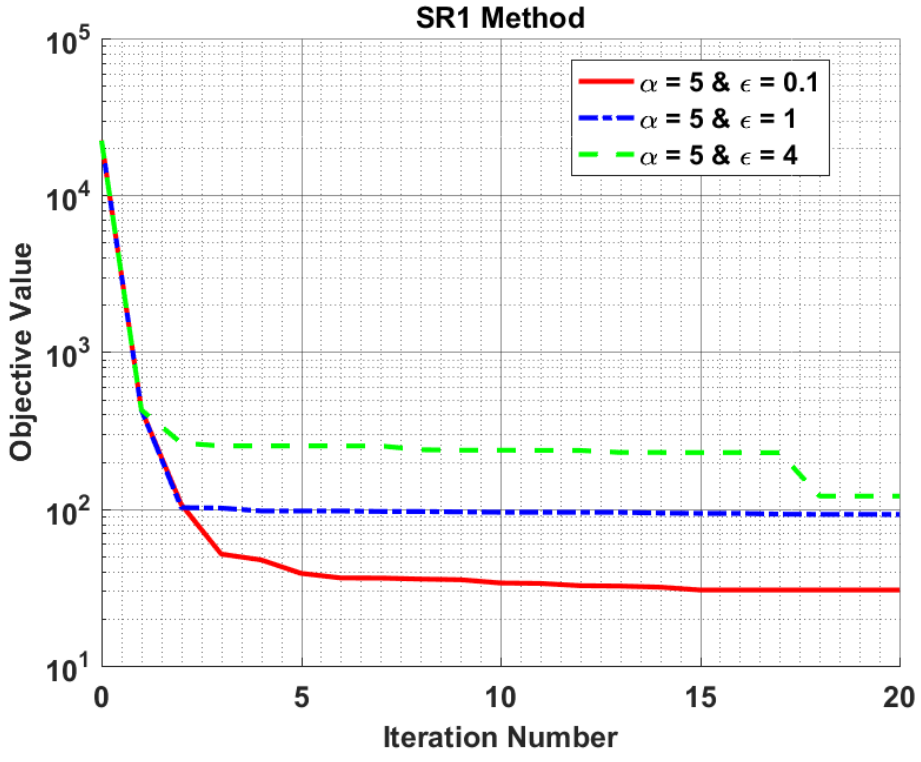


Figure 6.13: Comparison of Objective Change along Time for Perturbation Constant Sensitivity Analysis in Hover to Rightward Flight Maneuver


BM
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
Objective Value
Iteration Number
10^5
10^4
10^3
10^2
10^1
0
5
10
15
20


SR1 Method
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
Objective Value
Iteration Number
10^5
10^4
10^3
10^2
10^1
0
5
10
15
20

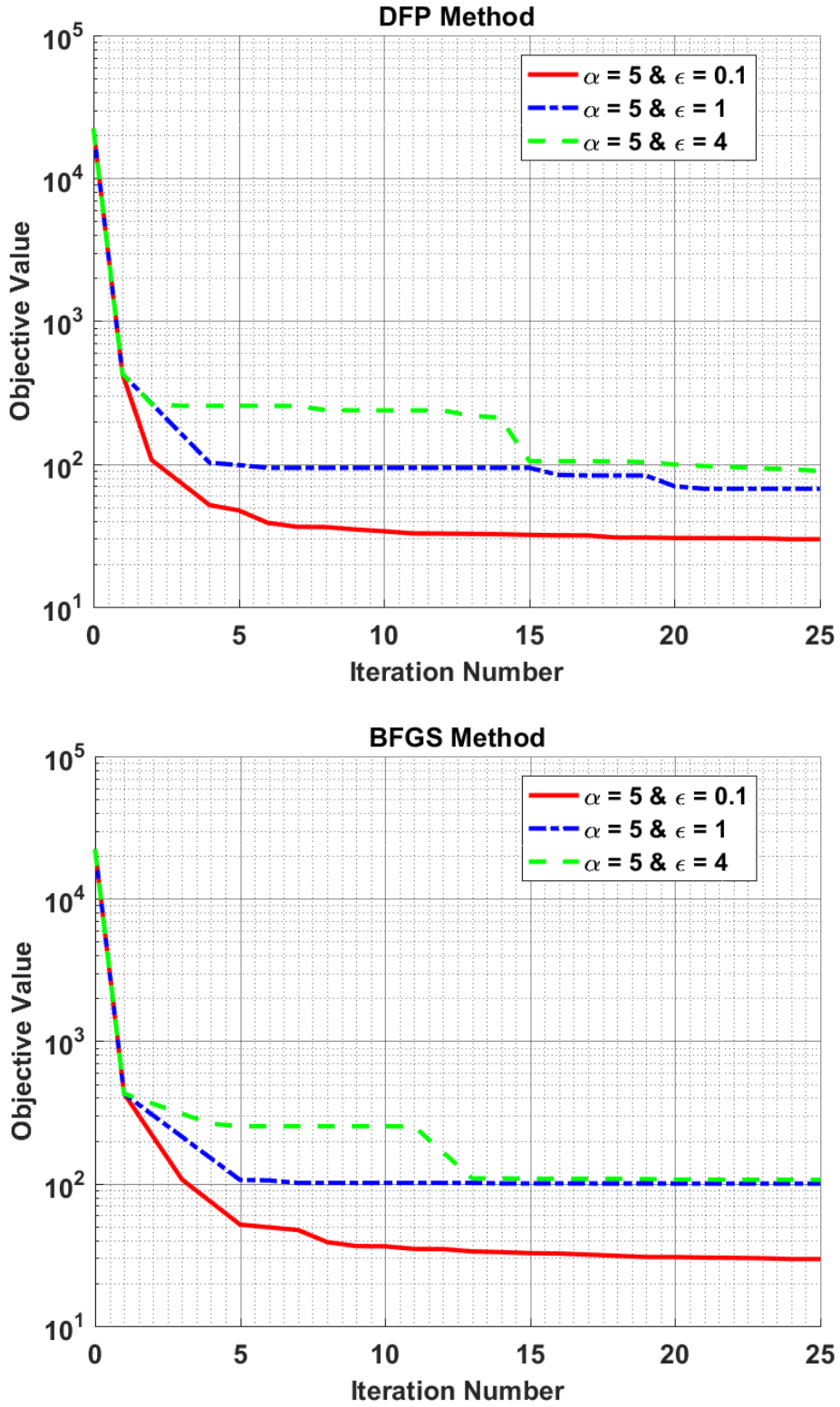
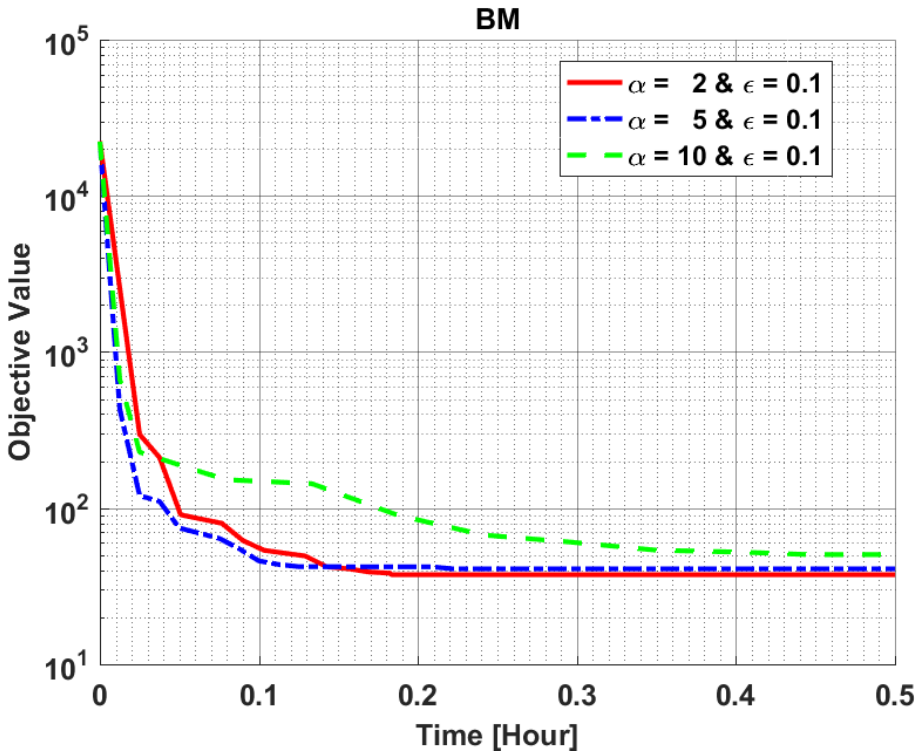


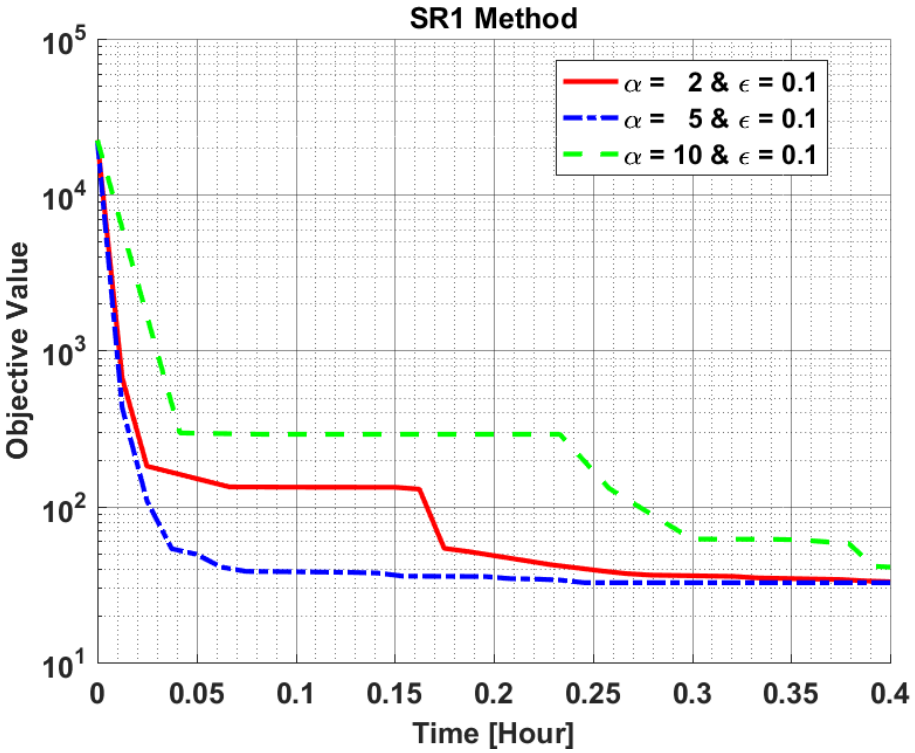
Figure 6.14: Comparison of Objective Change along Iteration Number for Perturbation Constant Sensitivity Analysis in Hover to Rightward Flight Maneuver

Figure 6.13 and Figure 6.14 show how the perturbation constant affects the optimization problem convergence for each algorithm. Like in hover to forward flight maneuver sensitivity analysis, the perturbation constant values of 1 and 4 are stuck high objective levels in this maneuver and could not achieve a better solution for any method. In the same way, the perturbation constant values of 0.1 converges to the lowest objective value. Therefore, so this value is accepted as the perturbation constant value for hover to rightward flight maneuvering problem in all these optimization methods.

For the sensitivity analysis of the perturbation constant, the constraint responses of the helicopter in all configurations and corresponding pilot control inputs are given in Appendix B2.

The same procedure described in Section 6.3.1.4 for the step size sensitivity analysis is also followed in this section. Therefore, while the optimization methods are applied for 2, 5 and 10 step size values, the perturbation constant value is taken as 0.1 and all other optimization parameters are still kept constant. Then, the changes of objective value along both the optimization time and the iteration number for these step sizes are obtained as seen in Figure 6.15 and Figure 6.16.


BM
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1
Objective Value
10^5
10^4
10^3
10^2
10^1
0
0.1
0.2
0.3
0.4
0.5
Time [Hour]


SR1 Method
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1
Objective Value
10^5
10^4
10^3
10^2
10^1
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
Time [Hour]

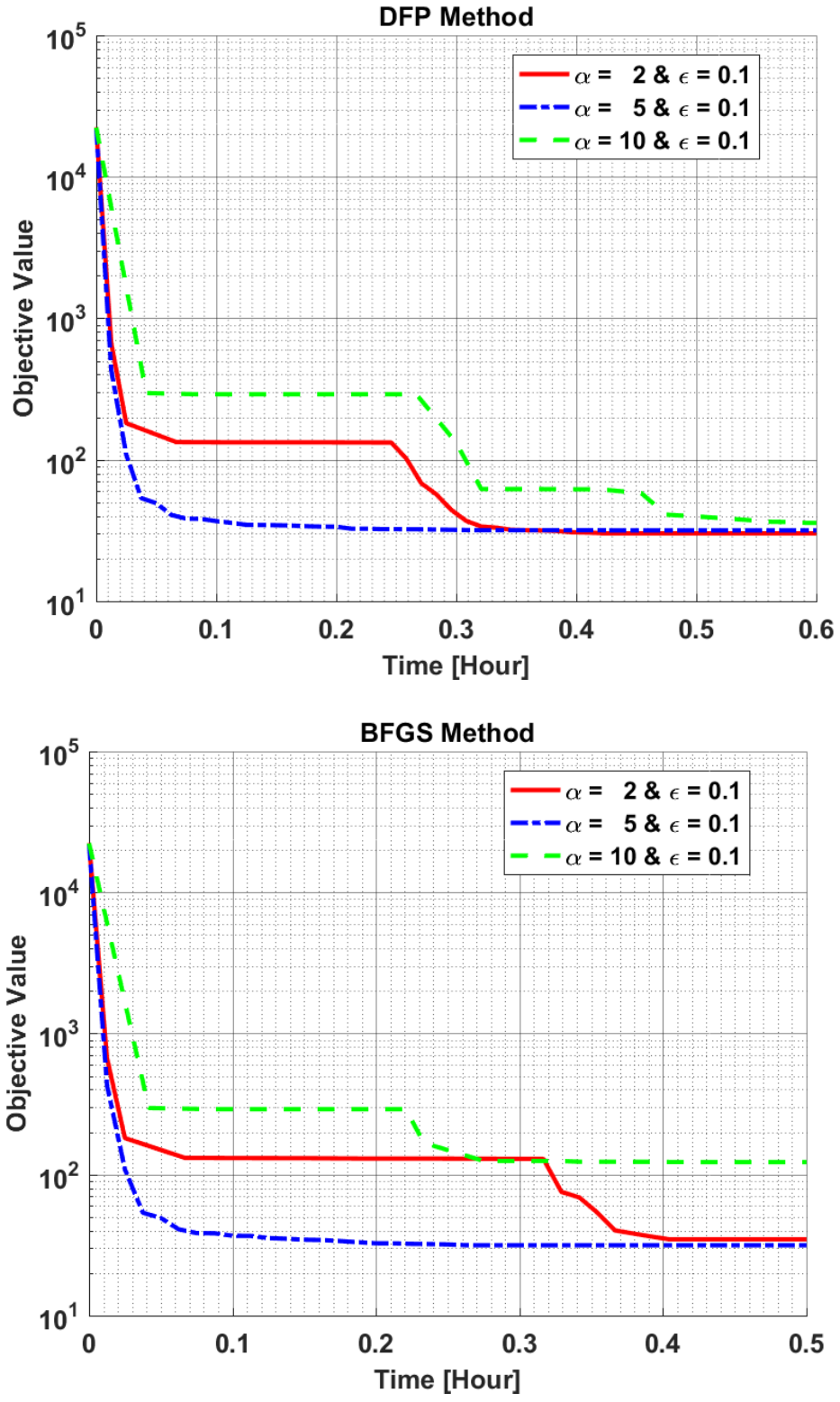
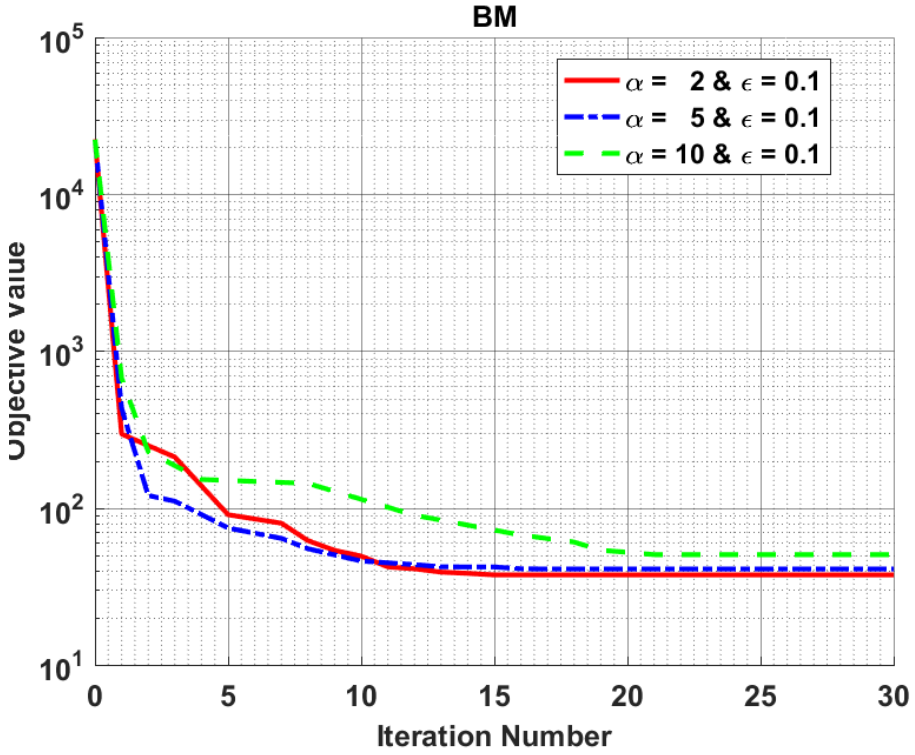
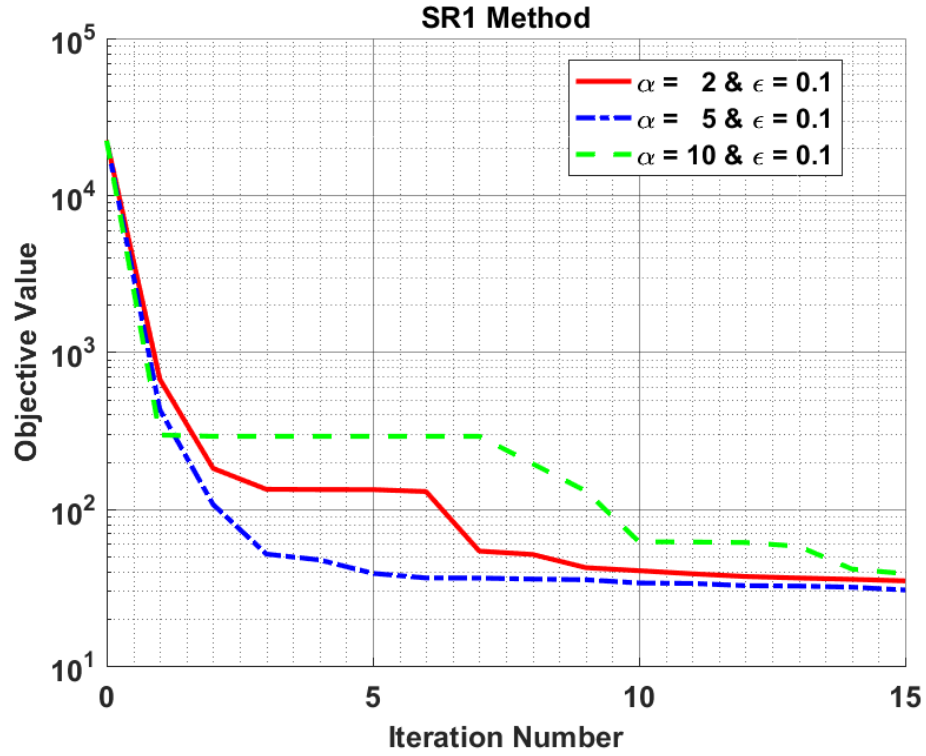


Figure 6.15: Comparison of Objective Change along Time for Step Size Sensitivity Analysis in Hover to Rightward Flight Maneuver


BM
Objective Value
Iteration Number
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1
10^5
10^4
10^3
10^2
10^1
0
5
10
15
20
25
30


SR1 Method
Objective Value
Iteration Number
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1
10^5
10^4
10^3
10^2
10^1
0
5
10
15

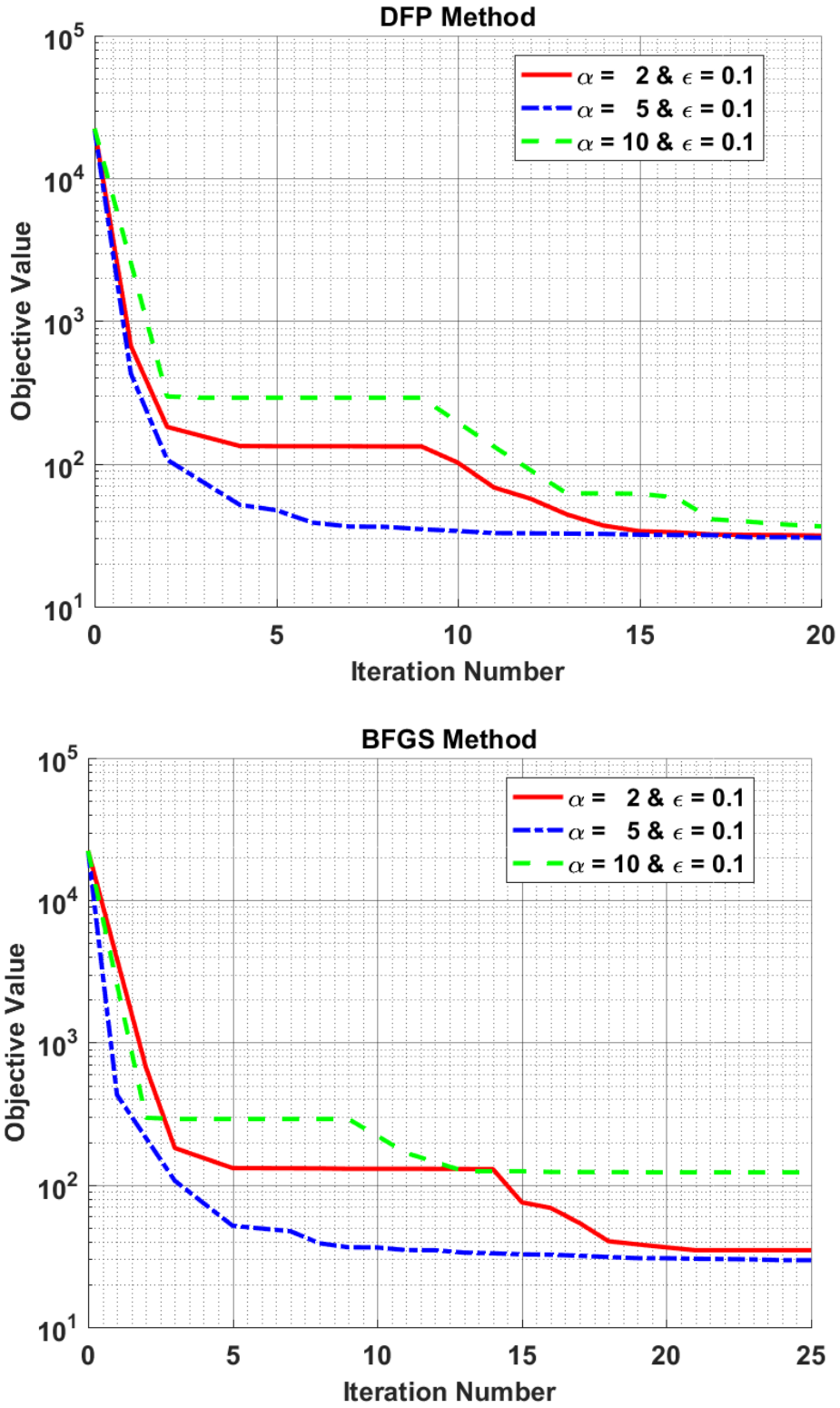


Figure 6.16: Comparison of Objective Change along Iteration Number for Step Size Sensitivity Analysis in Hover to Rightward Flight Maneuver

Figure 6.15 and Figure 6.16 show the importance of the step size parameter on the convergence of the optimization problem for each optimization algorithm. Also, they set out that the 5 value of step size reached the best solution in earlier for all optimization methods except Broyden's Method. In Broyden's method, even though the solution of this value seems close to the solution of 2 step size value, the value of 2 has a faster solution path. Therefore, while the step size value of 2 is taken as the most useful value for Broyden's Method, the step size value of 5 is taken for other methods. Hence, these values are considered as the most useful step size values for hover to rightward flight maneuvering problem in these optimization methods.

For these sensitivity analyses, the constraint responses of the helicopter in all configurations and corresponding pilot control inputs are given in Appendix B3.

Finally, it can be deduced that SR1, DFP and BFGS methods for this optimization problem have the best combination of step size and perturbation constant at $\alpha = 5$ and $\varepsilon = 0.1$. In addition, BM has the best combination of step size and perturbation constant at $\alpha = 2$ and $\varepsilon = 0.1$.

6.3.2.5 Comparison of the Methods for Hover to Rightward Flight Maneuvering

As in Section 6.3.1.5, the solution of the best combination for each method is compared here in order to decide the most effective method for this maneuver optimization. Therefore, the solution of best combination for each method is plotted in Figure 6.17 and Figure 6.18.

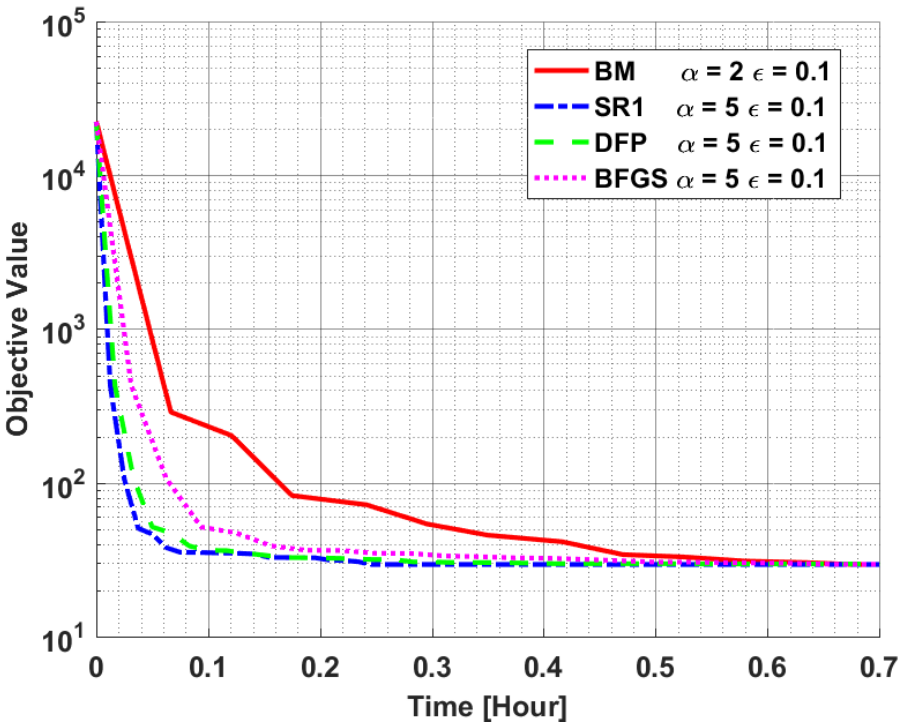


Figure 6.17: Comparison of Objective Change along Iteration Number for the selected quasi-Newton Methods in Hover to Forward Flight Maneuver

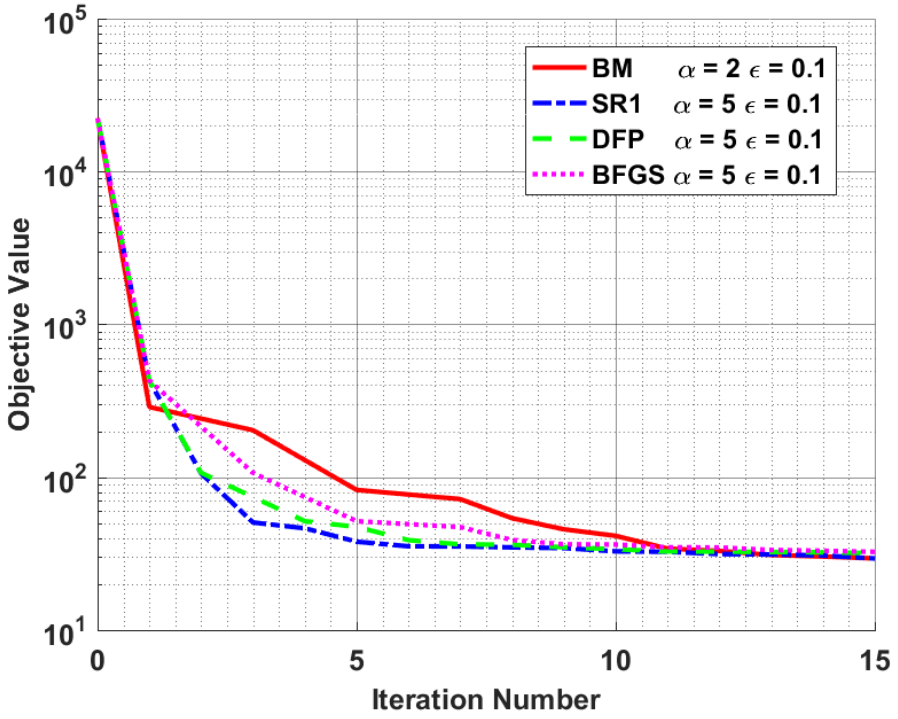
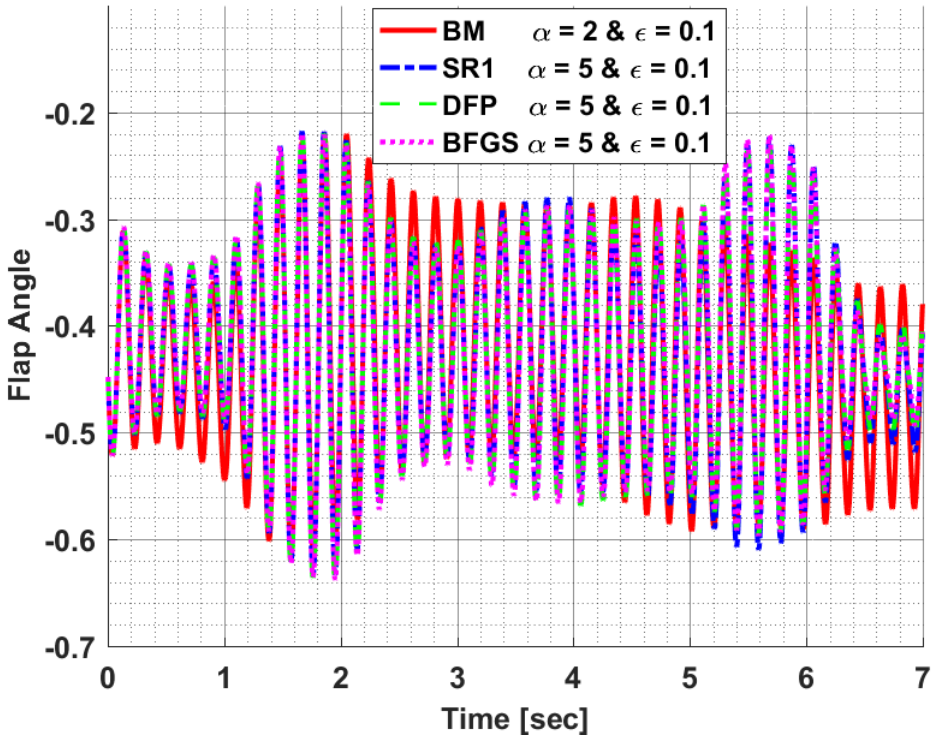
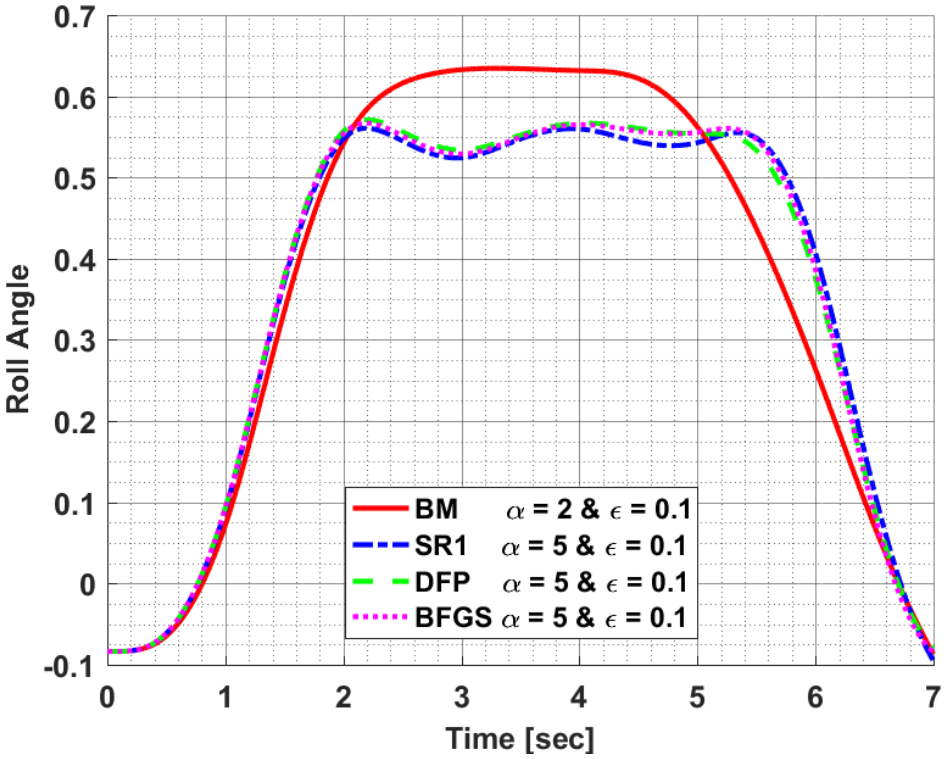


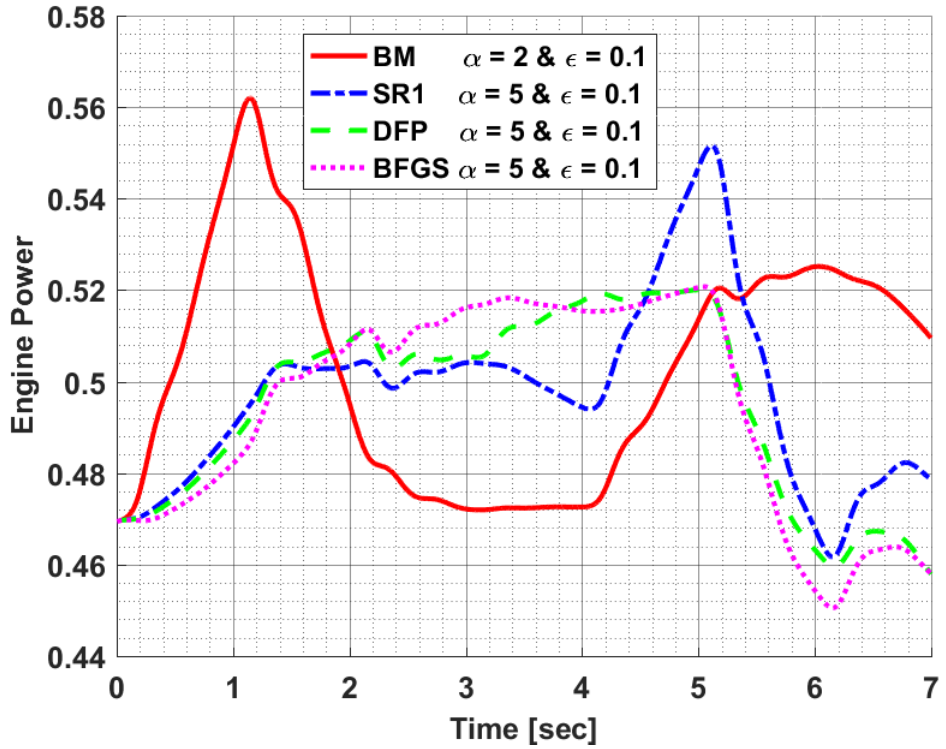
Figure 6.18: Comparison of Objective Change along Iteration Number for the selected quasi-Newton Methods in Hover to Forward Flight Maneuver

Figure 6.17 and Figure 6.18 show that the Broyden's Method is the method that achieves the lowest target at the latest. Again, the second slowest method is the BFGS method, and the third slowest is the DFP method. Namely, the SR1 method is the fastest method for this maneuver optimization. Therefore, it is concluded that SR1 method is the most useful method among others for hover to rightward flight maneuver optimization.

Figure 6.19 shows how much the constraint results of hover to rightward flight maneuver could meet for each optimization method. The constraint results are taken from the final iteration solution of each optimization algorithm. Moreover, while the constraints of flap angle and roll angle are given in the range of -1 and 1, the engine power constraint is given from 0 to 1. In other words, the minimum and maximum limits of flap and roll angles are represented by the value of -1 and 1, respectively. Similarly, the value of 1 in the engine power chart represents the maximum engine power value. Then, the values between the parameter limits are scaled according to these ranges. However, there is no scaling for rightward velocity and altitude charts.




Roll Angle
Time [sec]
BM α = 2 & ϵ = 0.1
SR1 α = 5 & ϵ = 0.1
DFP α = 5 & ϵ = 0.1
BFGS α = 5 & ϵ = 0.1


Engine Power
Time [sec]
BM α = 2 & ϵ = 0.1
SR1 α = 5 & ϵ = 0.1
DFP α = 5 & ϵ = 0.1
BFGS α = 5 & ϵ = 0.1

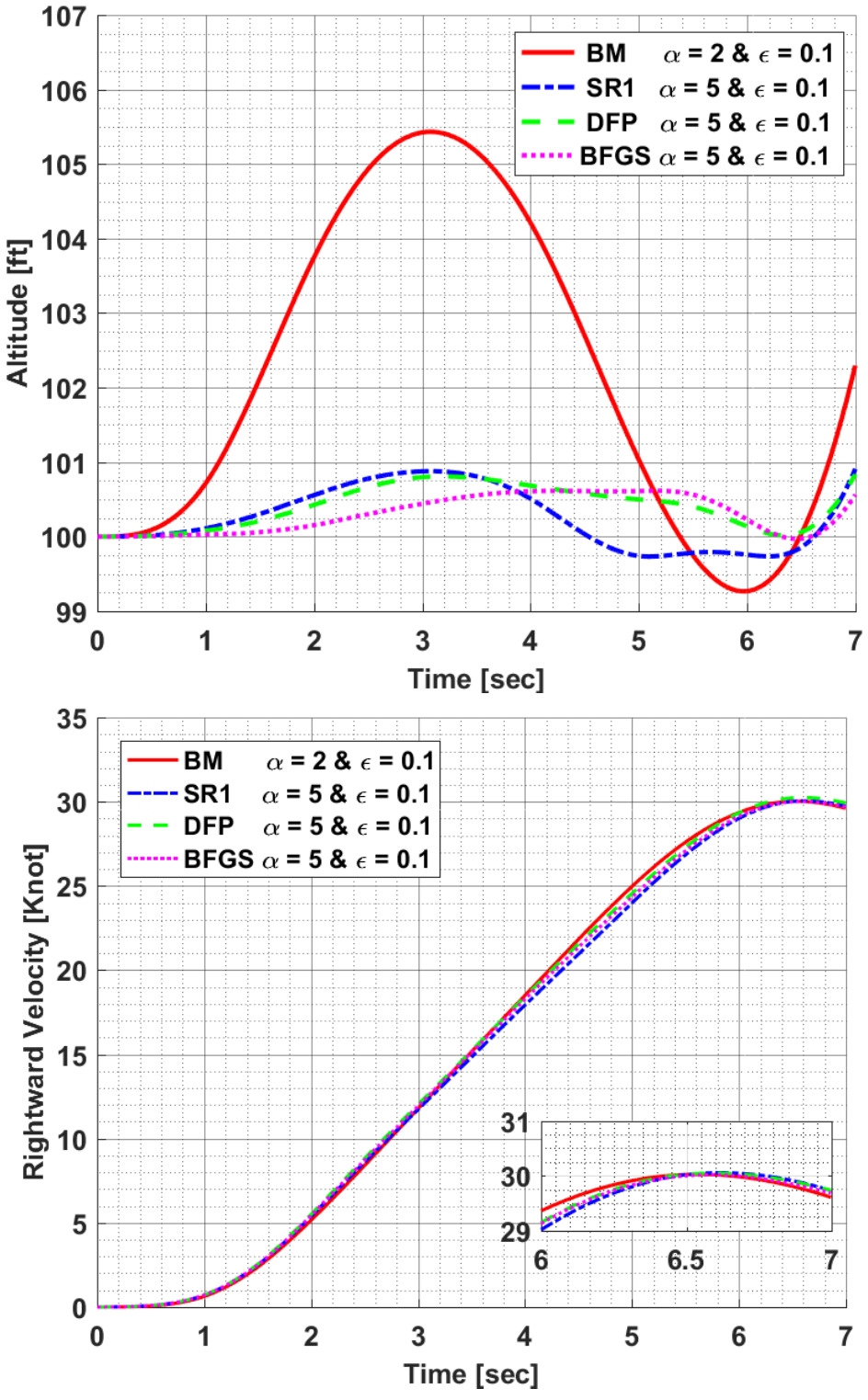


Figure 6.19: Constraint Results of Hover to Rightward Flight Maneuver in Optimization Methods

Figure 6.19 shows that each optimization method met all the constraints of hover to rightward flight maneuver. The graph of the rightward velocity says that the targeted speed value in the specified time interval is obtained by all optimization methods. Moreover, the roll angle graph shows that the helicopter rotates clockwise on the flight direction. Thus, the effect of the thrust force is increased in this direction. Furthermore, although other constraints have different solutions, they have been achieved within the specified limits.

Since each optimization method has its own character, they follow different ways to solve the problem. Nevertheless, although they converged at almost the same objective value as seen in Figure 6.17, the constraint graphics are not identical since a helicopter has a high coupling between pitch, roll and yaw motion. This means that the same maneuver can be performed with different pilot control inputs.

The lateral and collective cyclic are defined as design variables in this maneuvering problem. Figure 6.20 shows the necessary lateral and collective cyclic input to be able to perform hover to rightward flight maneuver whose results are given in Figure 6.19.

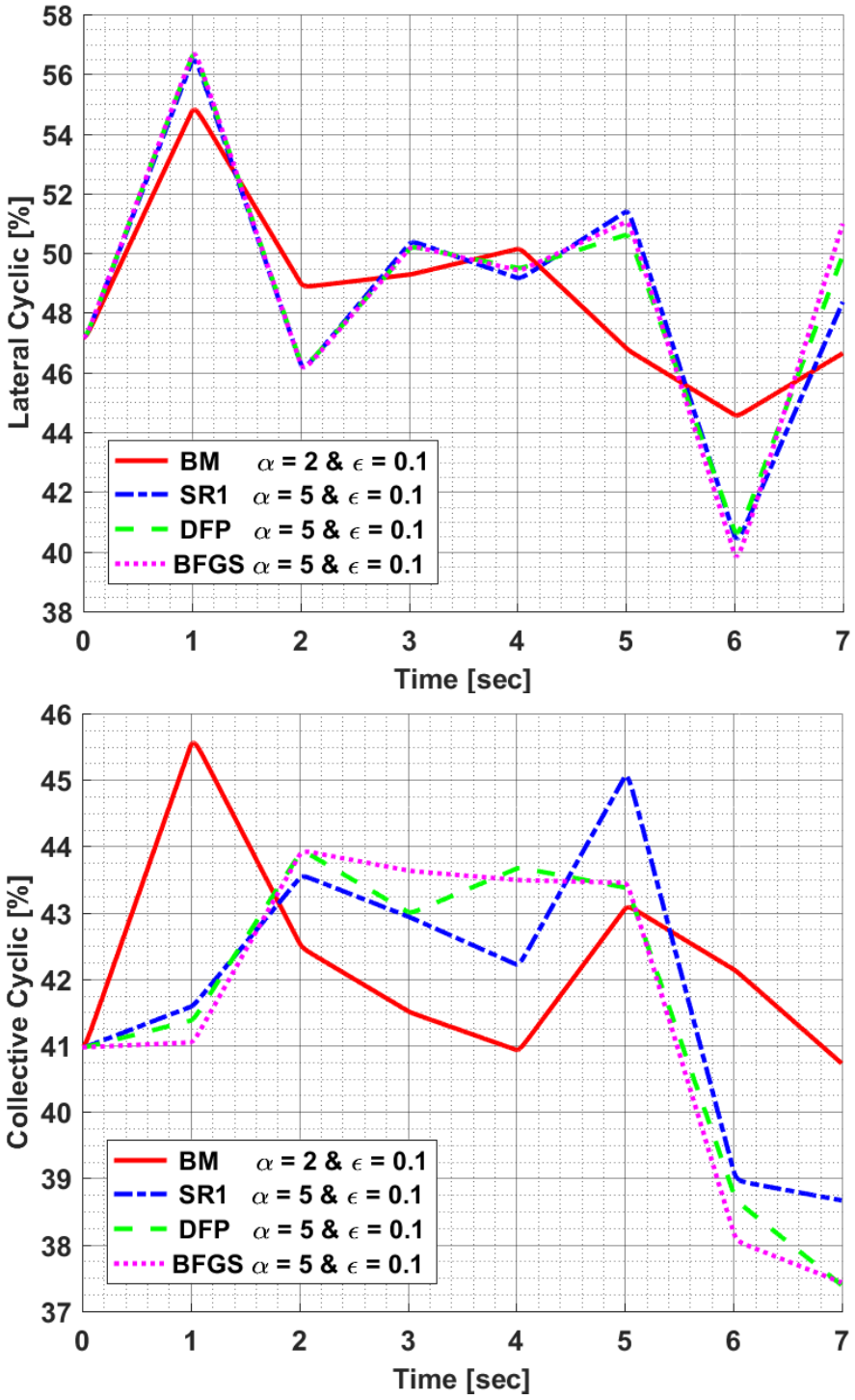


Figure 6.20: Obtained Design Variable in Hover to Rightward Flight Maneuvering Problem

An increase in percentage means that the lateral cyclic is pushed rightward and the collective cyclic is pulled.

6.4 Conclusion

The main purpose of this chapter is to decide on the most useful optimization method among those chosen. For this reason, two different maneuvers which are Hover to Forward Flight and Hover to Sideward Flight are investigated to find the most useful optimization method. Therefore, forward, central and backward difference approximations are applied as a gradient computational method. After the gradient computational method is decided, the sensitivity analysis of perturbation constant and step size parameters are examined for the selected values.

At the end of the comparison process of the maneuver optimizations, the results should be compared with each other. Thus, the best generic optimization method for maneuver optimization can be obtained. According to these results, it is seen that the central difference approximation is the most effective gradient computation method among the finite divided difference approximation. Moreover, the sensitivity analysis of perturbation constant and step size parameters shows that the SR1 method, whose perturbation constant value is 0.1 and step size value is 5, is the most useful method configuration among others.

As a result, it is decided that the SR1 method, whose gradient computation method is the central difference approximation, perturbation constant value is 0.1 and step size value is 5, is the most useful method for maneuver optimization. Therefore, all future comparisons and reviews are carried out with this method.

CHAPTER 7

THE DEVELOPED OPTIMIZATION CODE AND THE EFFECT OF THE OPTIMAZITION PARAMETERS

7.1 Introduction

This chapter describes the development stages of the optimization code, the effects of the starting point and time step on the optimization process. In addition, the different perturbation constants on the finite divided difference approximations were also studied. For these investigations, the hover to forward flight maneuver has been studied as an example to explain these issues better.

7.2 Development of the Optimization Code

An optimization code for helicopter maneuver optimization has been developed that can be applied to all methods. This code needs FLIGHTLAB analysis at three steps of each iteration. As it is described in Section 5.3, each iteration starts with the calculation of the objective function for the current design variable vector. This step requires only one FLIGLAB analysis. The next steps that need FLIGTLAB analysis are gradient calculation and line search algorithm, respectively. In the gradient calculation step, FLIGHTLAB analyses are performed with as many as the number of the perturbed design variables. Moreover, the line search algorithm calls FLIGHTLAB software to calculate the objective values of each point in the line search interval.

As defined in Section 2.8.1, FLIGHTLAB simulation analysis performs the trim analysis first and then continues the transition analysis. Therefore, the helicopter mathematical model is trimmed to the desired flight maneuver. Then, the transient

part of the simulation analysis is performed by applying pilot control inputs with respect to time. Thus, FLIGHTLAB resolves the helicopter maneuver for the defined parameters. In addition, the solution of these two stages requires different times. FLIGHTLAB software consumes more time in the trim stage. For example, it consumes about 85% of the seven-second maneuver in the trim stage, and the rest in the transient phase. The applied pilot control input values do not significantly change the duration of the transient phase. In other words, the analyses using different pilot inputs consume approximately the same time in the transient phase.

One of the main purposes of using optimization methods to solve a problem is to get the best solution in the shortest time. Therefore, the optimization code must also be improved in the most efficient way. In this thesis, the following modifications are applied in order to increase the efficiency of the optimization code. Each version of the modified optimization code is represented by the name 'Code' and its modification number. To illustrate, the first version is named ‘Code 1’, the second one is named ‘Code 2’ and so on.

Code 1 is the purest form of the optimization code. In this version, the code first calculates the objective value of the current design variables. Then, it computes the gradient of the objective function. For this purpose, it obtains the objective value of each perturbed design variables separately. During these computations, it waits for the completion of the previous one before proceeding to the other perturbation analysis. In the same way, the objective values of each point in the line search interval are calculated one by one.

Code 2 is the modified form of Code 1. The points where Code 2 differs from Code 1 are the gradient calculation of the purpose function and the line search algorithm. Unlike Code 1, it executes FLIGHTLAB analysis in parallel at these stages. In other words, it does not wait for the completion of the previous one to start the other analysis. On the contrary, the perturbation analyses are performed simultaneously in analysis time. Similarly, analyses of all points on the line search are carried out at the same time. Therefore, each line search cycle takes a single analysis time.

Code 3 has also modification in the gradient calculation of the objective function and the line search algorithm. In these stages, this code performs the trim part once, then it applies the sequential transient analyses by using this trim solution. To express this better, let's explain the gradient calculation section as an example. This code first obtains the desired trim flight of the helicopter. Then, the transient analyses of each perturbed design variable are applied to this trim solution one after the other. This means that there is no need for any trim analysis for each perturbation. Thus, the most time-consuming part of FLIGHTLAB analysis is performed only once. However, the previous transient analysis must be completed to start a new one. Moreover, the same procedure is followed for the points in the line search interval.

Code 4 is the final form of the optimization code in this study. The change made in Code 3 is adapted to the whole by this code. In this study, it is known that the helicopter mathematical model is trimmed to the same flight condition during maneuver optimization. The only thing that changes is the pilot control inputs applied in the transition analysis. Therefore, this repeated trim analysis is a waste of time. To overcome this loss, Code 4 calculates and stores the trim point at the beginning of the optimization code once. Afterwards, Code 4 gives this point as input in all FLIGHTLAB analysis and resolves the transition analysis part. Moreover, the analyses in gradient calculation and line search algorithm are still executed in parallel. Thus, the trim solution is not performed during the optimization process. This last modification is a big step in saving time because trim analysis takes a lot of time in the FLIGHTLAB simulation analysis.

To show the effect of these code versions on optimization, hover to forward flight maneuver described in Section 6.3.1 is carried out for all these code versions. For the solution of this maneuver, the SR1 method, which is the most useful method in helicopter maneuver optimization, is applied with its best configuration. Finally, the change of the objective function along the time is plotted for all these code modifications as seen in Figure 7.1.

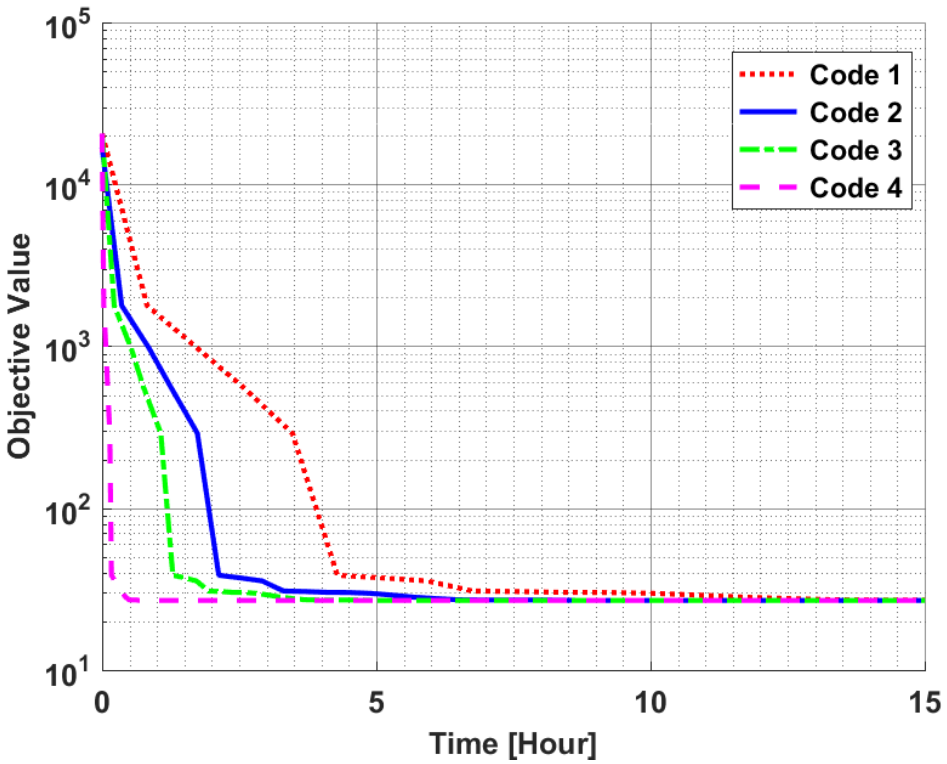


Figure 7.1: Comparison of Optimization Code Versions for Hover to Forward Flight Maneuver with SR1 Method

Figure 7.1 shows that each change greatly increases the convergence speed of the optimization code. As a result, Code 4 reveals that the optimization code is getting much faster in terms of convergence speed.

The improvement of the optimization code algorithm only affects the optimization time, not the number of iterations. Therefore, the change of the objective function along the iteration number is not given for this comparison. Moreover, the constraint results of this maneuver and the corresponding design variables are not given for these codes because the code versions only change the speed of the problem, not the results. This means that they are the same as the results obtained by the SR1 method at the 5 size step and 0.1 perturbation constant.

7.3 Effect of the Time Step on the Optimization Problem

As it is defined in Section 5.2, the pilot control inputs are time dependent parameters. Therefore, the design variable vector is created depending on time as

$$\boldsymbol{X}^T = \left[\delta_{co}(t_1), \dots \delta_{co}(t_n), \delta_{lo}(t_1), \dots \delta_{lo}(t_n), \delta_{la}(t_1), \dots \delta_{la}(t_n), \delta_{ap}(t_1), \dots \delta_{ap}(t_n)\right]$$

where t_n represents the n^{th} time step in which the design variables are modified. Although time steps can be selected at desired time points, they are selected at equal intervals in this thesis study. In order to examine the effect of the time step on the optimization problem, hover to forward flight maneuver described in Section 6.3.1 is taken as an example. This maneuver is performed with four different time steps, which are 0.25, 0.5, 1 and 1.4. Moreover, the maneuver is accomplished with the best configured SR1 method. In the end, the changes of the objective function along both the optimization time and the iteration number are plotted for these time step configurations as seen in Figure 7.2 and Figure 7.3.

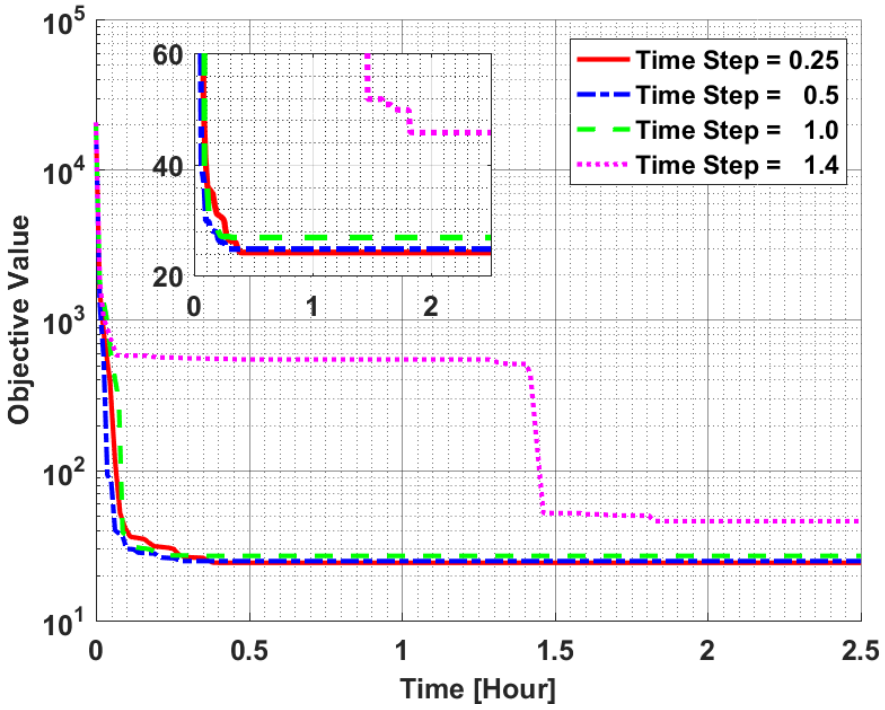


Figure 7.2: Effect of the Time Step Selection on Objective Change along Time for Hover to Forward Flight Maneuver with SR1 Method

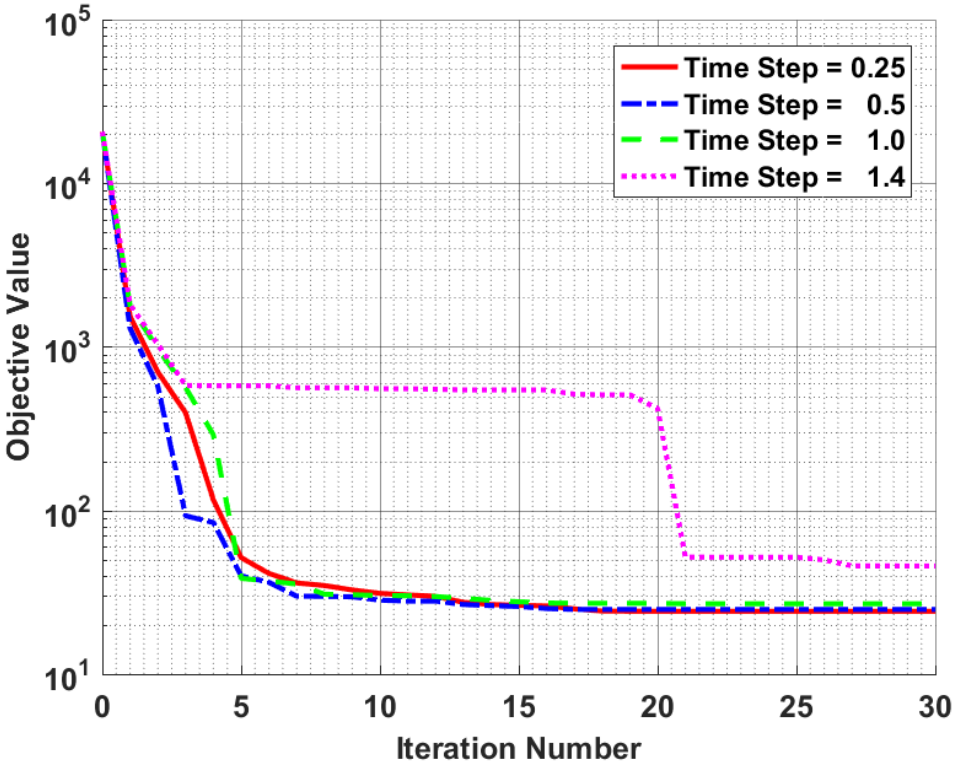
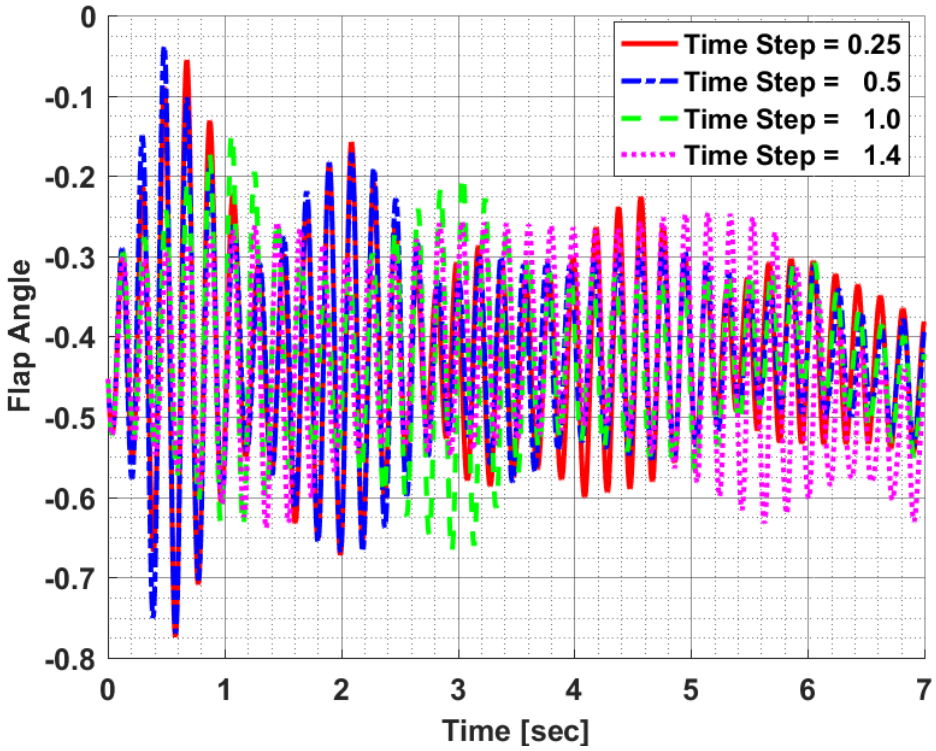
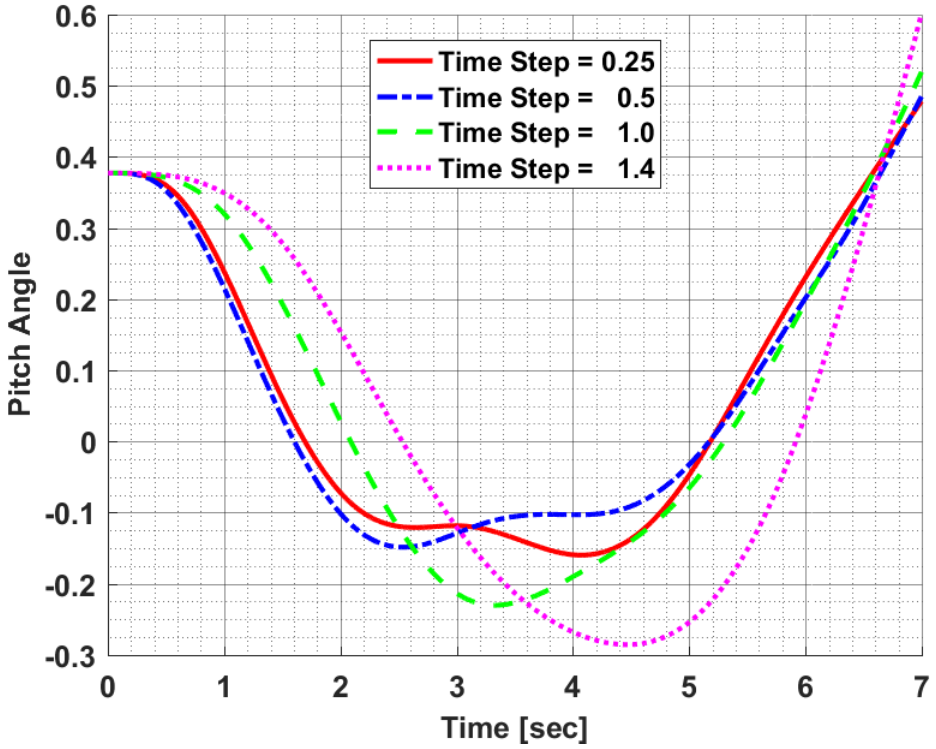


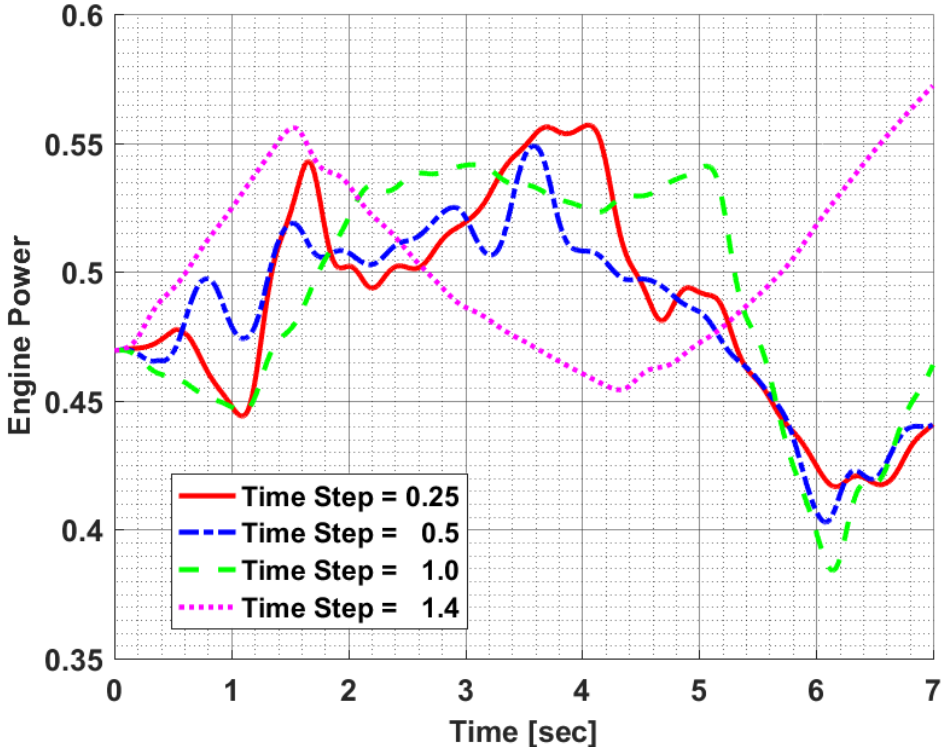
Figure 7.3: Effect of the Time Step Selection on Objective Change along Iteration Number for Hover to Forward Flight Maneuver with SR1 Method

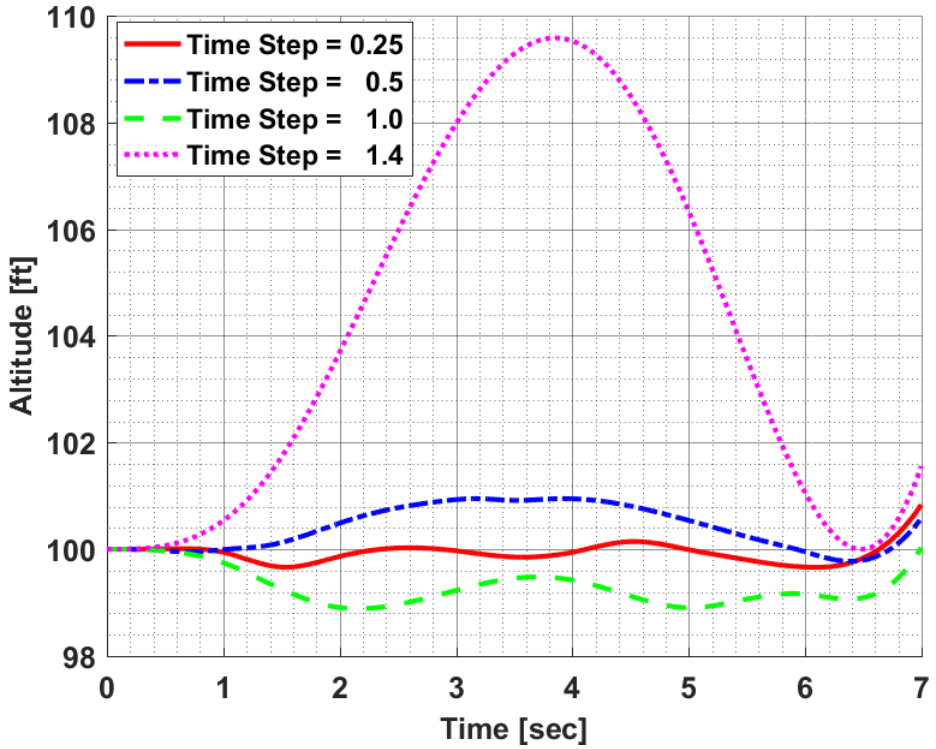
Figure 7.2 and Figure 7.3 show that the time step significantly affects the speed of the optimization problem. It is seen that the objective value converges to a smaller value as the time step gets smaller. These are the expected results because the optimization algorithm can fine-tune with small time steps to better meet problem constraints. Therefore, the 1.4 time step optimization is completed at a larger objective value. In addition, this time step value caused the optimization time to be extended as it had difficulty in finding better solutions during the optimization period. Furthermore, although the analyses with smaller time steps converge to smaller objective value, the smaller time steps mean more FLIGHTLAB analysis needs to be done at the same time due to the parallel run process. These parallel analyses can slow down the optimization speed because of CPU performance of the computer. Namely, if the computer CPU is slow, the optimization problem may converge later. In this optimization problem, although 0.25 time step optimization has reached the smallest objective value, it took more time than others.

In Figure 7.4, the constraint results of this maneuver are plotted for each time step. The scaled constraints are also included in this figure.






Engine Power
Time [sec]
Time Step = 0.25
Time Step = 0.5
Time Step = 1.0
Time Step = 1.4


Altitude [ft]
Time [sec]
Time Step = 0.25
Time Step = 0.5
Time Step = 1.0
Time Step = 1.4

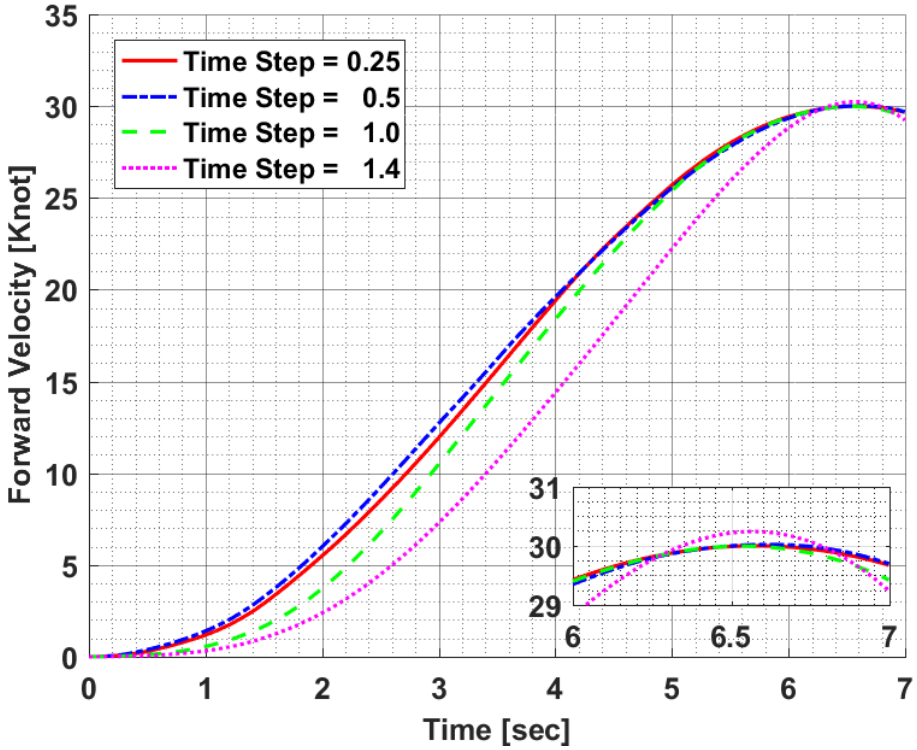


Figure 7.4: Constraint Results of Time Steps in Hover to Forward Flight Maneuver with SR1 Method

Figure 7.4 shows that the analyses of the time steps meet all the constraints of hover to forward flight maneuvers. In addition, it is seen that the constraints are met more accurately as the time step decreases. However, it should be noted that the decreasing time step may increase the optimization time. Therefore, the time step should be chosen depending on whether it is important to complete the maneuver as soon as possible or to achieve the best result. In other words, if the optimization accuracy is more important than the optimization time, a smaller time step can be preferred. If the optimization time is more important, a relatively larger time step can be used.

The design variables in this maneuvering problem are longitudinal and collective cyclic. Therefore, Figure 7.5 shows the longitudinal and collective cyclic inputs corresponding to the results given in Figure 7.4.

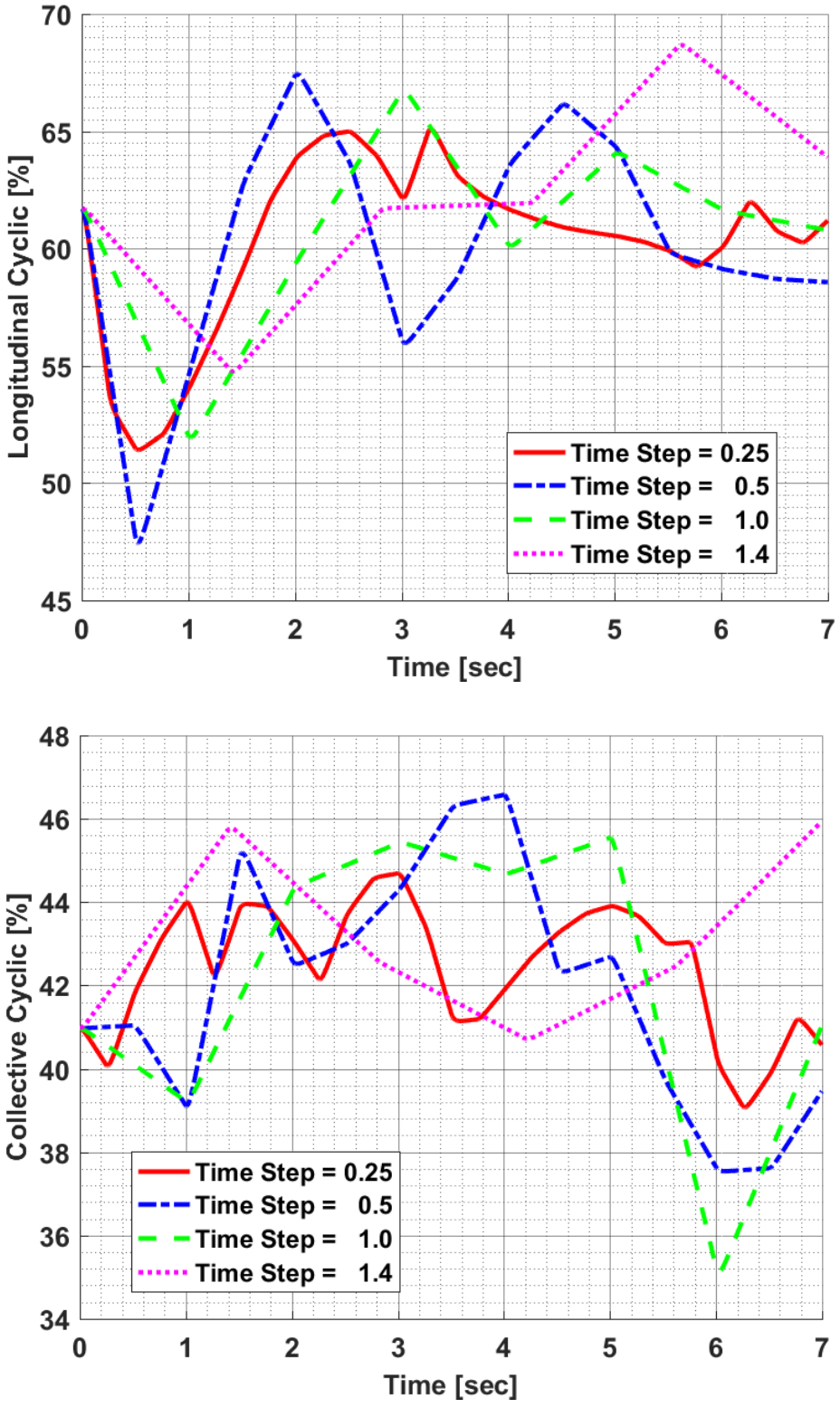


Figure 7.5: Obtained Design Variable of Time Steps in Hover to Forward Flight Maneuver with SR1 Method

As it is seen in Figure 7.5, each design variable changes at defined time step points. Namely, the variables of the 0.25 time step are modified every 0.25 seconds, while in step 0.5 they are modified every 0.5 seconds, and the others are in the same logic.

7.4 Effect of the Initial Condition on the Optimization Problem

As it is defined in Section 3.2, a deterministic algorithm needs an initial point to start optimization. Starting from a logical initial condition is crucial for the convergence of optimization. If an optimization problem starts from an unreasonable initial condition, it either diverges or takes more time to reach the optimum result. Therefore, the initial values of the design variables should be chosen logically.

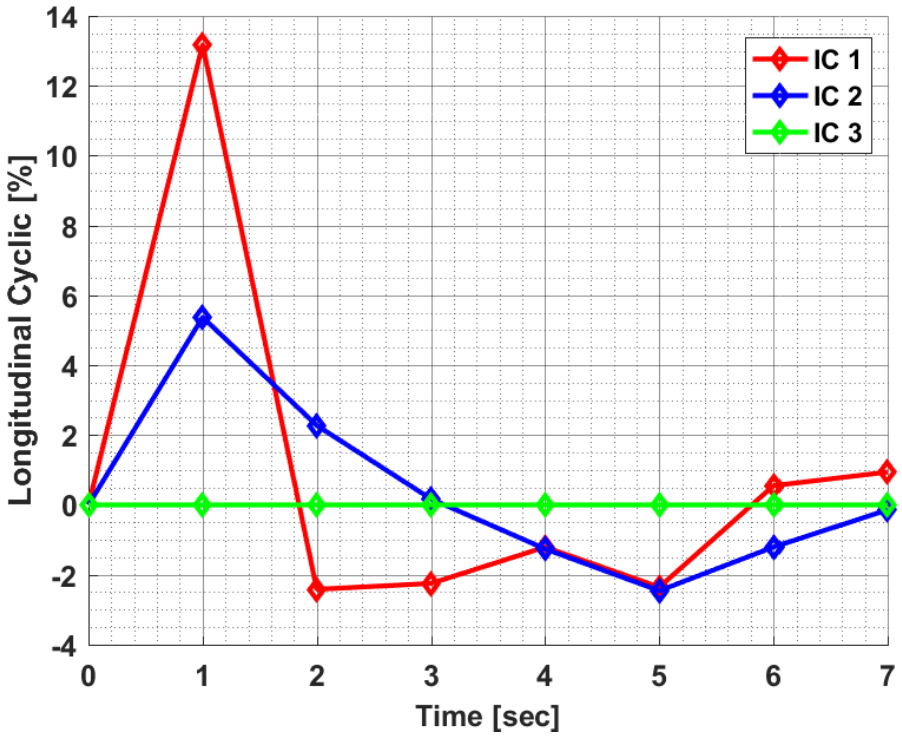
To examine the importance of the initial condition in the optimization problem, hover to forward flight maneuver described in Section 6.3.1 is considered as an example. For the solution of this maneuver, the best configuration of the symmetric rank-one method is used. Moreover, the design variable vector is created to be modified at each one-second. In addition, three different initial conditions are used for this comparison.

Initial condition 1 (**IC1**) represents the pilot control inputs required to perform hover to 30 Knot backward flight maneuver. To achieve these inputs, the backward flight maneuver is solved by the best configuration of the symmetric rank-one method. The optimization procedure of this maneuver is described in Section 8.3.1.

Initial condition 2 (**IC2**) is similar to IC1, but this uses the pilot control inputs required to perform hover to 20 Knot backward flight maneuvers. The same procedure in hover to 30 Knot backward flight maneuver is followed for the IC2 values. The only difference in the analysis procedure between them is their target speeds. But, the results of the analysis are not presented in this thesis.

Initial condition 3 (**IC3**) is the most logical one. This condition represents the pilot control inputs on the trim flight of the helicopter. In other words, IC3 defines each element of the initial design variable vector as zero.

In the comparison process of the initial conditions, it is aimed to realize hover to forward flight maneuver. For this purpose, the pilot control input values obtained from backward flight maneuvers are defined as IC1 and IC2. Logically, they are not correct approaches because these starting points are far from the intended maneuver. On the other hand, IC3 uses the pilot control input obtained at the trim point. This is both a real and logical choice because a pilot starts maneuvering from the trim flight condition in real life. This means that if the helicopter has the ability to perform the targeted maneuver, it can be carried out when the maneuver is started from the trim point. As it is stated in Section 5.2, the design variable values represent delta values applied to the trim points. Therefore, IC3 takes each element of the initial design variable vector as zero at all time point. Finally, the variation of these initial design variables over time is given in Figure 7.6.



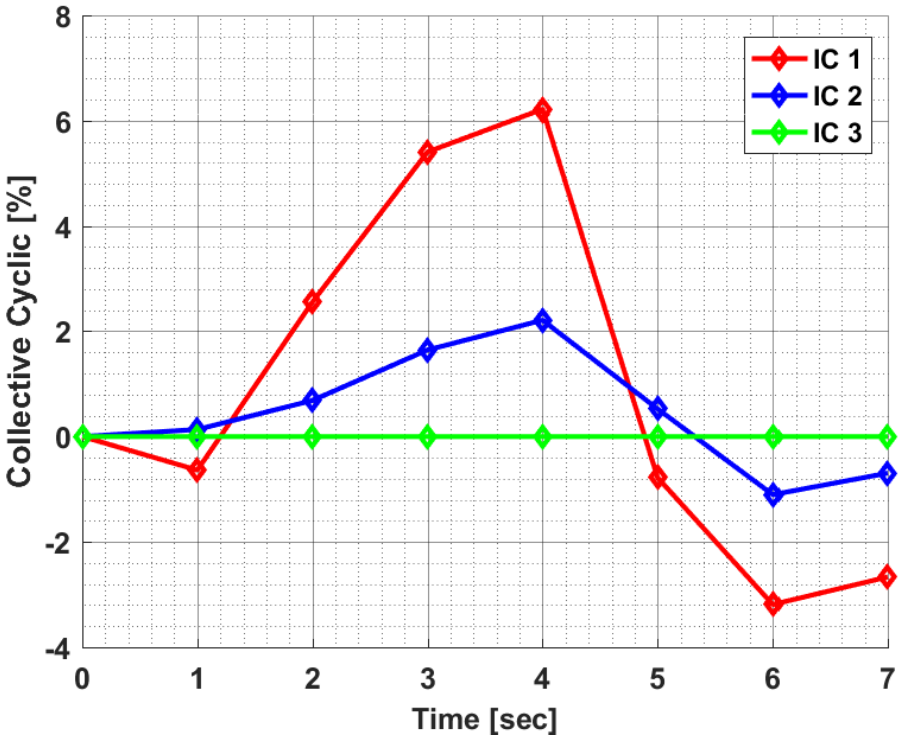


Figure 7.6: Defined Initial Design Variables for Hover to Forward Flight Maneuver with SR1 Method

The hover to forward flight maneuver optimization problem has been performed by using these initial conditions as a starting point. Afterwards, the changes of the objective function along both the optimization time and the iteration number are given in Figure 7.7 and Figure 7.8.

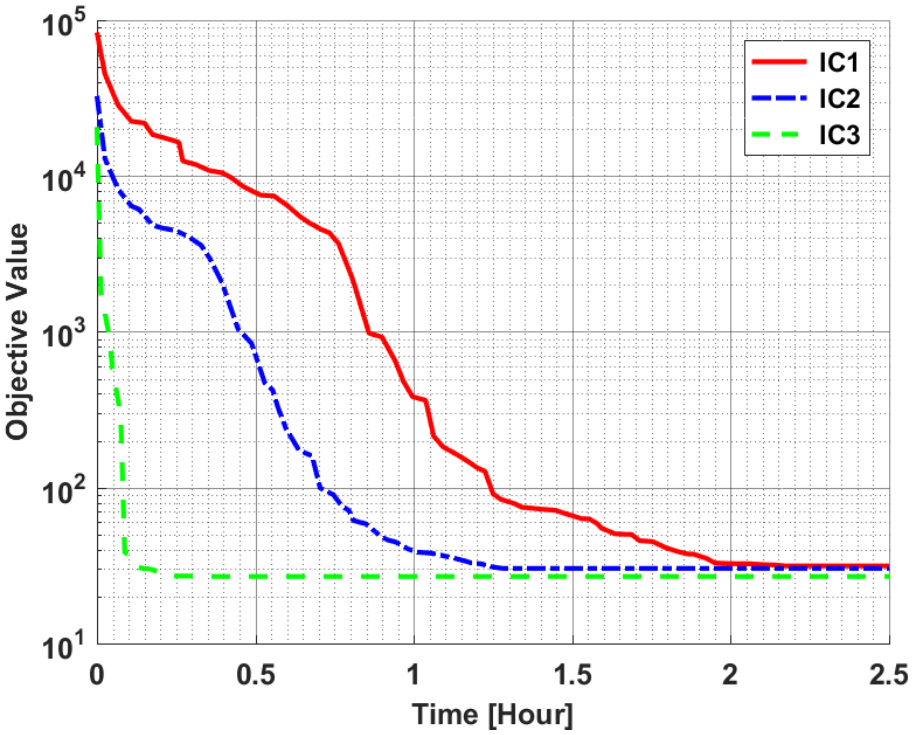


Figure 7.7: Effect of the Initial Condition Selection on Objective Change along Time for Hover to Forward Flight Maneuver with SR1 Method

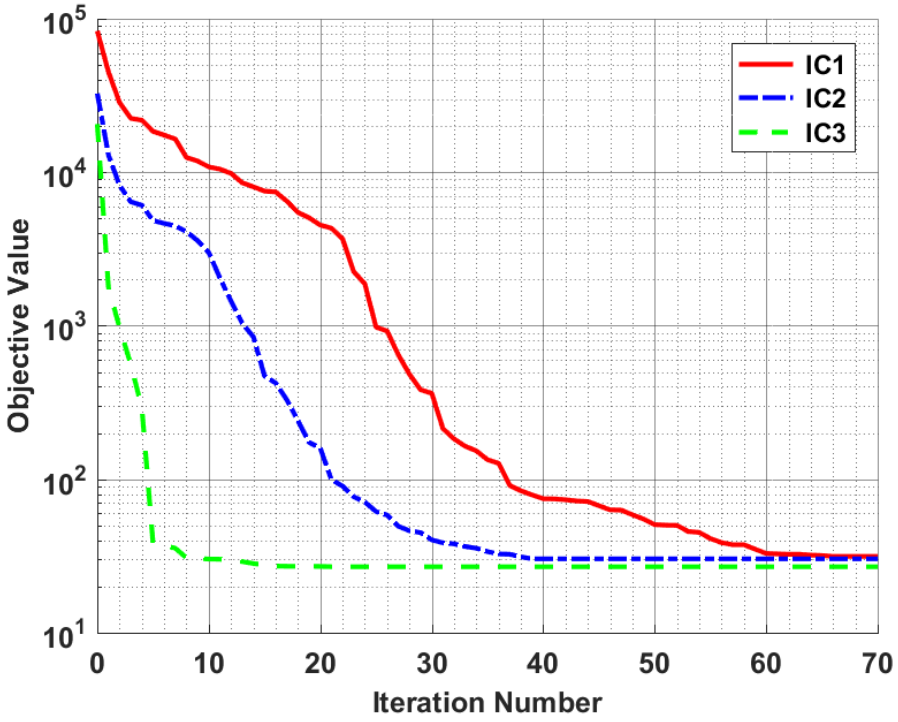
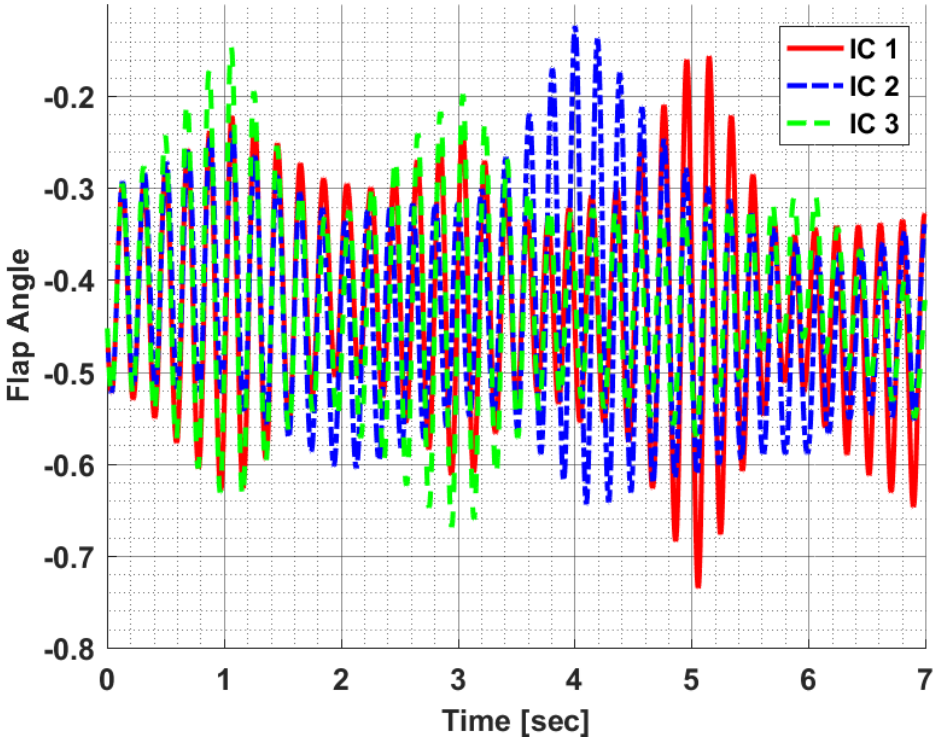
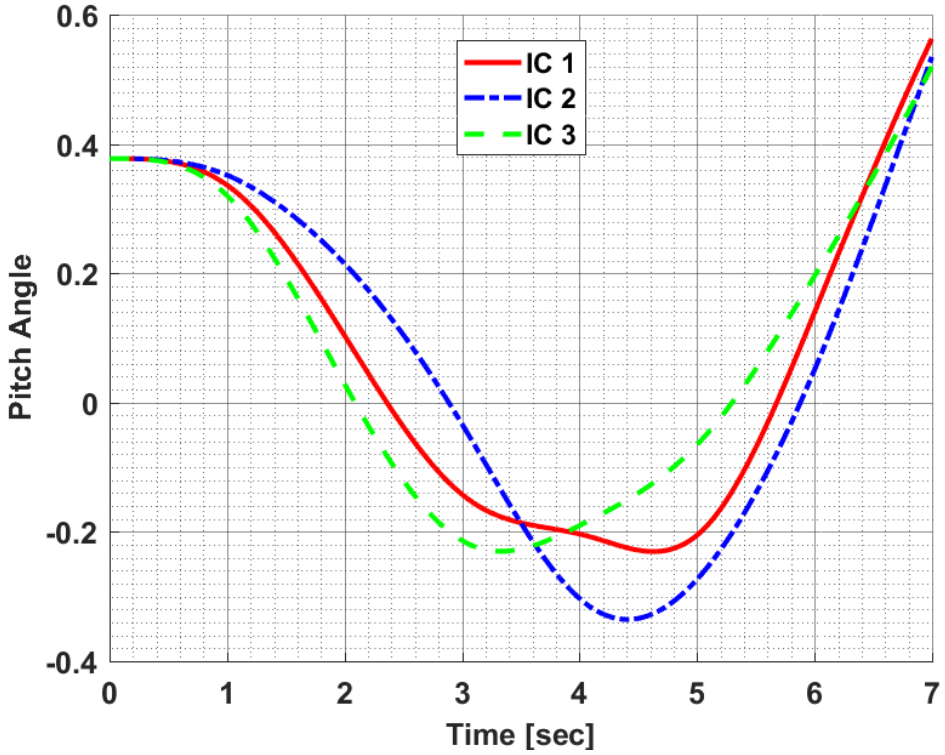
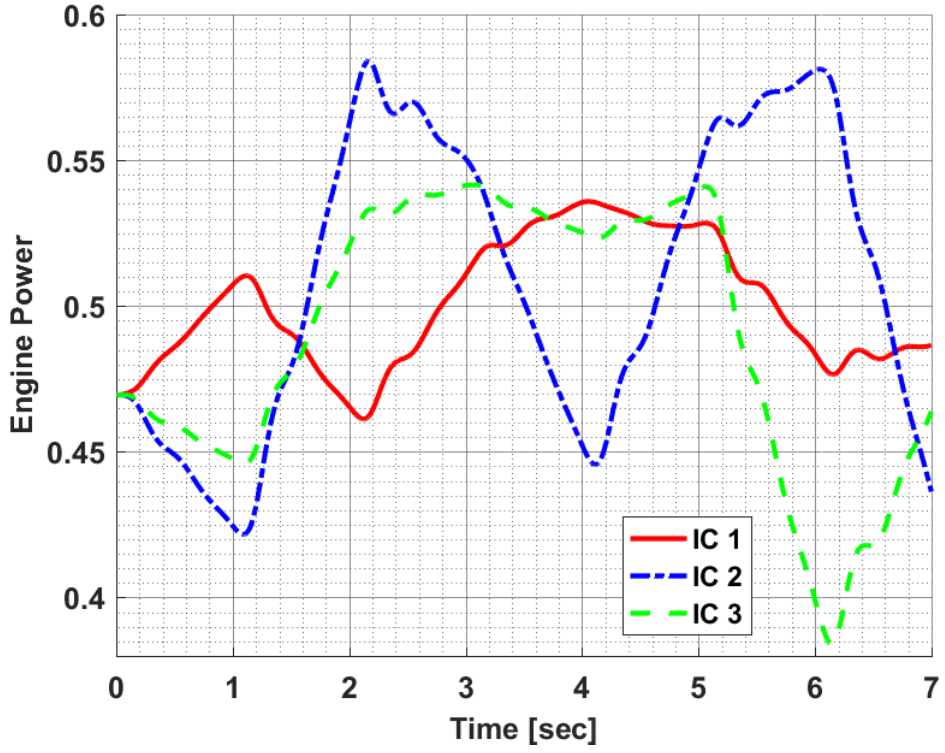


Figure 7.8: Effect of the Initial Condition Selection on Objective Change along Iteration Number for Hover to Forward Flight Maneuver with SR1 Method

Figure 7.7 and Figure 7.8 show the importance of the selection of the initial conditions on the convergence speed of the optimization problem. It can be seen that each optimization analysis starts from different objective values. While the objective value of the analysis starting from IC1 has the highest value, which of the analysis starting from IC3 has the lowest value. This is the expected result because IC1 is the farthest initial condition and IC3 is the most logically selected starting condition among them. For the same reason, the analysis starting from IC3 converges faster than the analysis starting from IC1. Furthermore, the analysis starting from IC2 takes place between other analyses both in terms of convergence speed and initial objective value. Although the converged objective values are very close to each other, the analysis starting from logically selected IC3 converges faster. Finally, the constraint results of these analyses with different initial values are given in Figure 7.9.




IC 1
IC 2
IC 3
Pitch Angle
Time [sec]
0.6
0.4
0.2
0
-0.2
-0.4
0
1
2
3
4
5
6
7


IC 1
IC 2
IC 3
Engine Power
Time [sec]
0.6
0.55
0.5
0.45
0.4
0
1
2
3
4
5
6
7

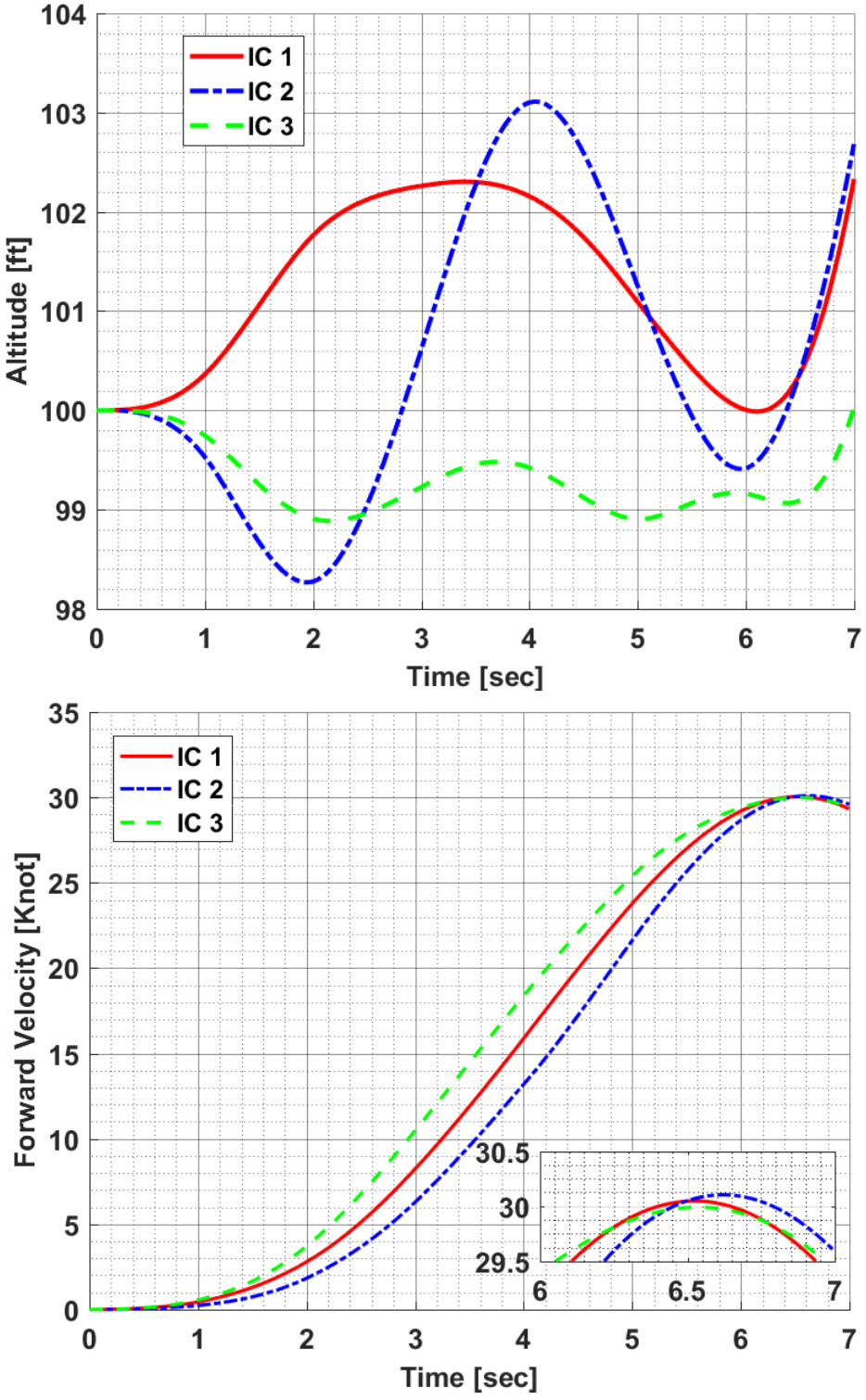
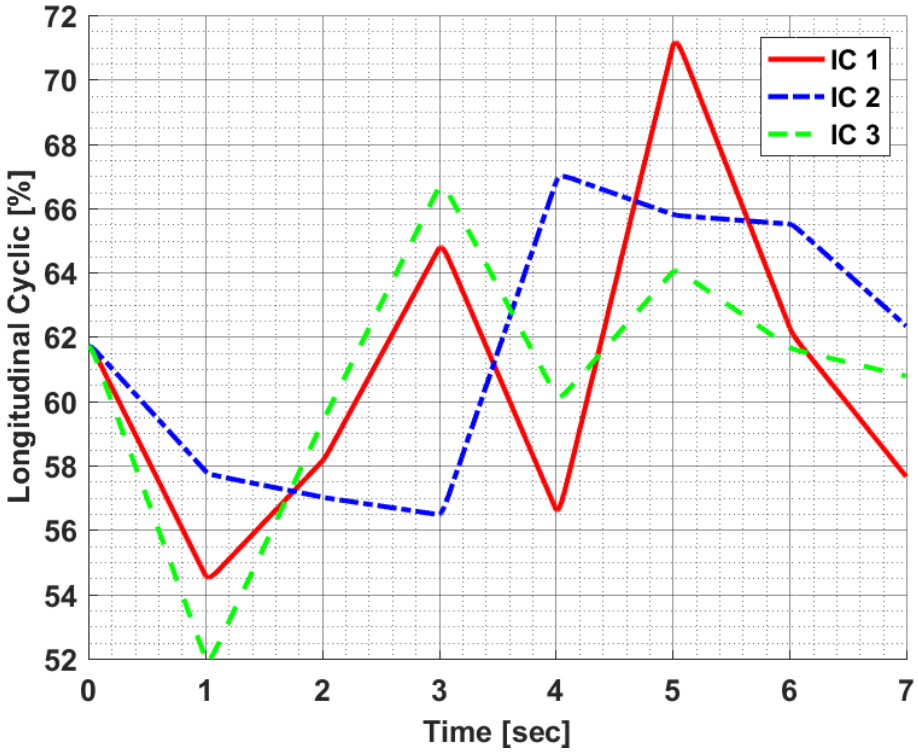


Figure 7.9: Constraint Results of Initial Condition in Hover to Forward Flight Maneuver with SR1 Method

Figure 7.9 shows that although each analysis with different initial values follows different solution paths, they meet the constraints of hover to forward flight maneuver. It can be said that the pitch angle chart has a similar trend for each analysis. Moreover, each optimization analysis completed the maneuver at almost the same altitude. According to the velocity chart, even though these analyses follow different acceleration paths, they eventually reached the targeted value.

The corresponding design variables in this maneuvering problem, which are longitudinal and collective cyclic are given in Figure 7.10.



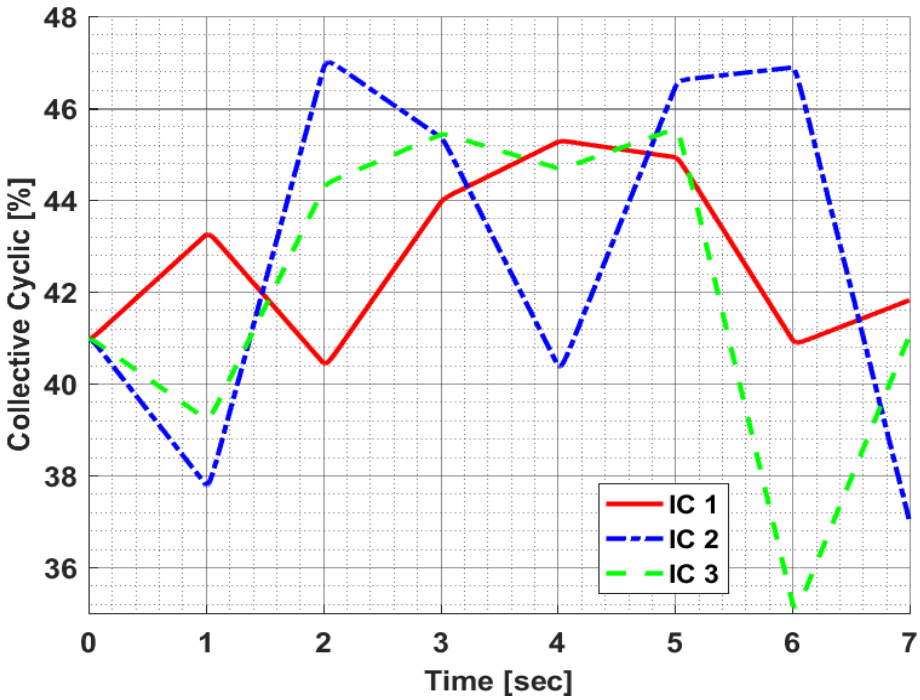


Figure 7.10: Obtained Design Variable of Initial Condition in Hover to Forward Flight Maneuver with SR1 Method

Figure 7.10 shows that the targeted maneuver can be achieved with different longitudinal and collective cyclic inputs due to the high coupling between roll, pitch, and yaw motions. As a result, when the trim point is taken as the initial value, the desired maneuver can be accomplished as long as the helicopter's ability. Therefore, IC3 has decided the logical initial condition for the maneuver optimization problems.

7.5 Effect of Perturbation Constant Selection for the Finite Divided Difference Approximations on the Optimization Problem

As it is defined in Section 3.2.1, a gradient-based optimization method needs the gradient computation of the objective function. For this purpose, the finite divided difference approximations were used for the gradient calculation of the objective function. Moreover, they were compared to each other at the same perturbation constant value of 0.1 and then the most useful one was decided according to this comparison condition.

In order to investigate the perturbation constant selection on the finite divided difference approximations, the hover to 30 Knot forward flight maneuver defined in Section 6.3.1.2 was realized by the forward and backward difference approximations with perturbation constant values of 0.01 and the central difference approximation with perturbation constant values of 0.1. For the solution of this maneuver, the SR1 method is applied. Then, the changes of the objective function along both the optimization time and the iteration number are plotted for these conditions as seen in Figure 7.11 and Figure 7.12.

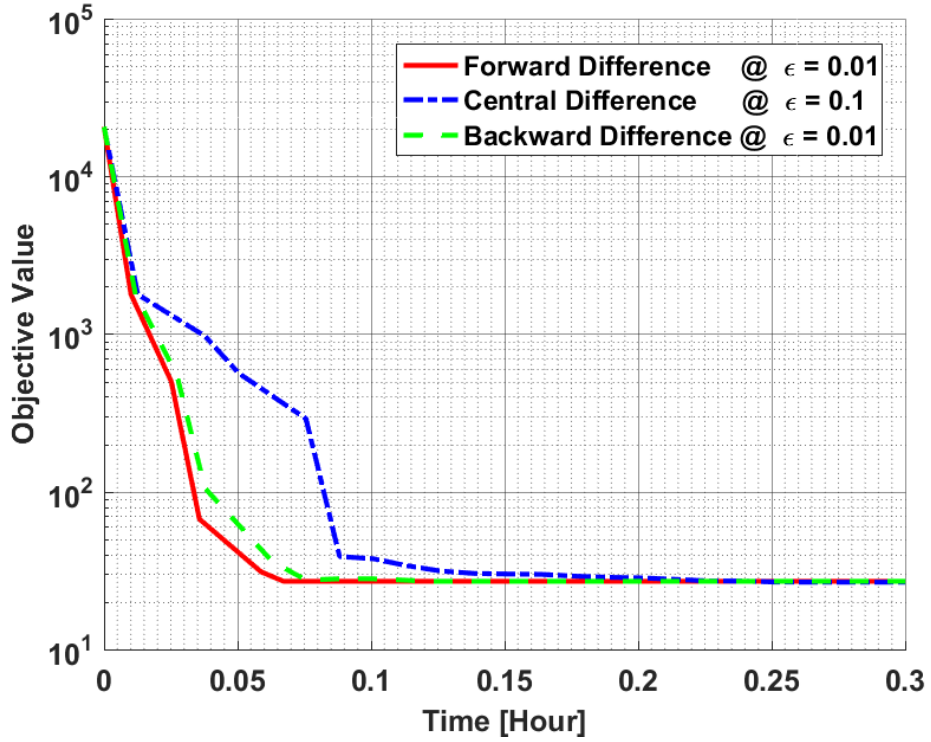


Figure 7.11: Different Perturbation Constants for the Finite Divided Difference Approximations along Time in Hover to Forward Flight Maneuver

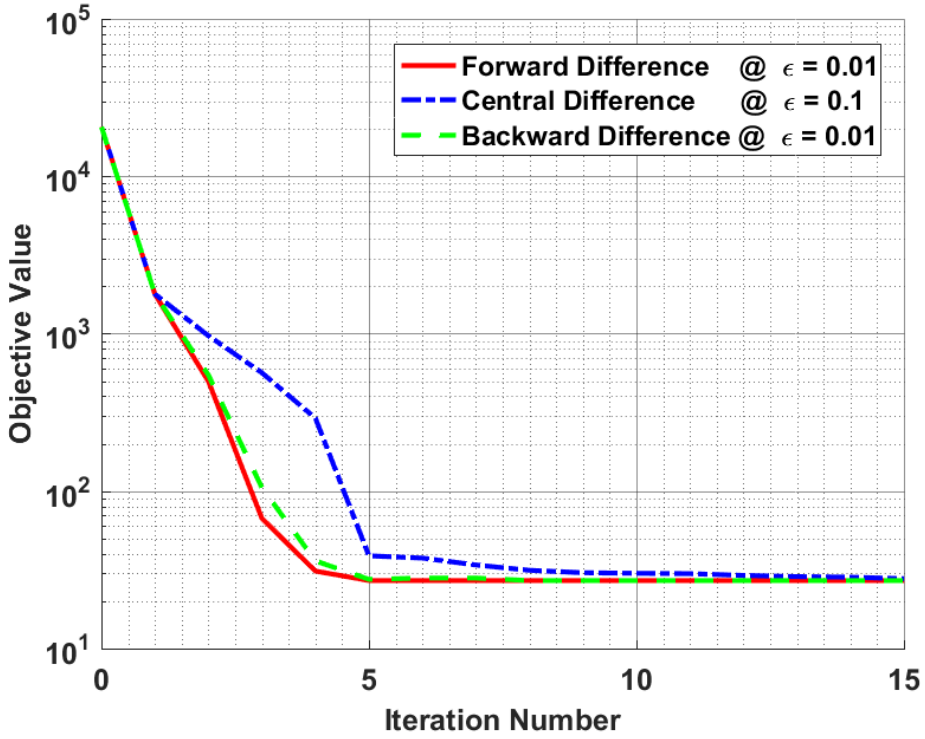


Figure 7.12: Different Perturbation Constants for Finite Divided Difference Approximations along Iteration Number in Hover to Forward Flight Maneuver

Figure 7.11 and Figure 7.12 show that the solutions obtained by the forward and backward difference approximations with perturbation constant values of 0.01 are faster than the central difference approximation with perturbation constant values of 0.1. In addition, the forward and backward difference approximations followed a similar trend throughout the optimization. At the end of the optimization problem, the forward and backward difference approximations converged to nearly the same objective value as the central difference approximation. As a result, it can be concluded that although the forward and backward difference approximations with perturbation constant values of 0.01 achieved faster almost the same accurate solution as the central difference approximation with perturbation constant value 0.1.

7.6 Conclusion

In this chapter, the improvement process of the developed optimization code algorithm is explained in detail. Furthermore, the efficiency of this improvement is supported by comparing the convergence speed of a maneuver performed with different code versions. Afterwards, the importance of choosing the time step and initial value is showed separately by examining maneuvers which are performed different values. Finally, the significance of perturbation selection on finite divided difference approximations is studied by example maneuver.

CHAPTER 8

MANEUVER OPTIMIZATION ANALYSIS

8.1 Introduction

In addition to the maneuvers used in the comparison analysis, different helicopter maneuvers were solved with the symmetric rank-one optimization method and their optimization procedures are given in this chapter. These maneuvers are the maneuvers the helicopter is most exposed to in its lifetime. However, pull up and pushover maneuvers have been configured at limit flight values to prove that the optimization method can be applied to more difficult maneuvers. Then, the results of the defined constraints and corresponding pilot control inputs were plotted for each optimization maneuver.

8.2 Optimization of Hover to Backward Flight Maneuver

8.2.1 Hover to Backward Flight

The technique of hover to backward flight maneuver is the same as that of hover to forward flight maneuver as defined in Section 6.3.1.1. The only differences between them are the direction of the rotorcraft. In other words, the backward maneuver also starts from hover position at a specific altitude. Then, the longitudinal cyclic is applied backwards to accelerate the rotorcraft in that direction. Moreover, this maneuver targets to reach a constant speed and to perform the maneuver at the almost same altitude and heading. The rest of the maneuver requirements are the same as hover to forward flight maneuver.

8.2.2 Optimization Modeling of Hover to Backward Flight Maneuver

Like hover to forward flight maneuver optimization, hover to backward flight maneuver also uses the longitudinal and collective cyclic as the design variables. In addition, the lateral cyclic and anti-torque pedal inputs are controlled by SAS.

In this maneuver, the maneuvering time takes about 7 seconds and the design variables are modified every second. Moreover, it is aimed to reach 30 Knot backward velocity at the almost constant altitude. Therefore, the constraints of this optimization problem are created taking into account the maneuver requirements and the design limits of the helicopter. The constraints of this maneuver optimization are the same as those of hover to forward flight optimization, except the direction of the velocity. Therefore, Constraint 4 is modified for the direction of the velocity as

Constraint 4 – Backward Velocity Constraint

This maneuver also aims to achieve the 30 Knot backward velocity between 6.3th and 6.7th seconds. However, since it is intended to move in the opposite direction of hover to forward flight maneuver, the sign of this constraint is changed as:

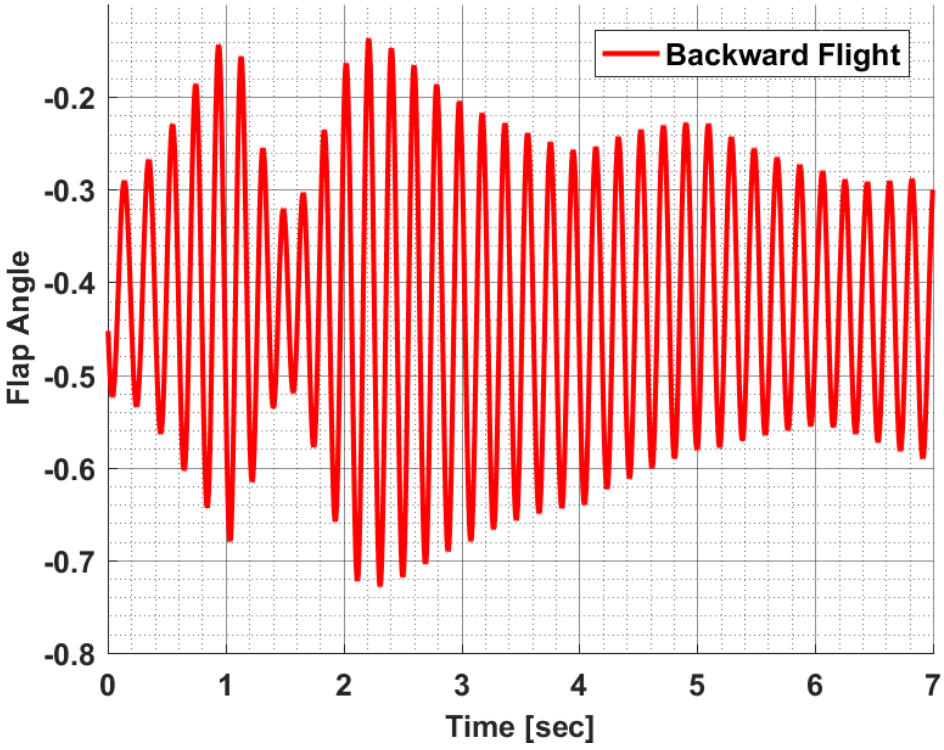
$$g_4 = \int_{t_s=6.3}^{t_f=6.7} |U + 30|^2 \, dt$$

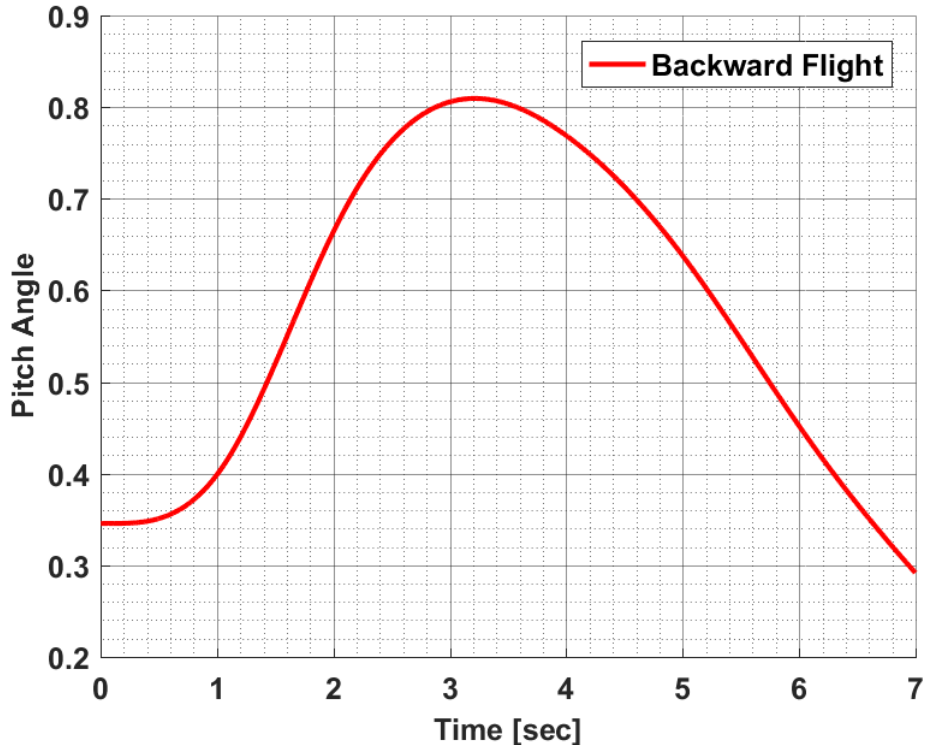
where the helicopter longitudinal velocity is represented by $\boldsymbol{U}$ and is considered a positive sign to the forward. Also, the upper and lower time boundaries where the targeted parameter is desired to be obtained are represented with $\boldsymbol{t_s}$ and $\boldsymbol{t_f}$ respectively.

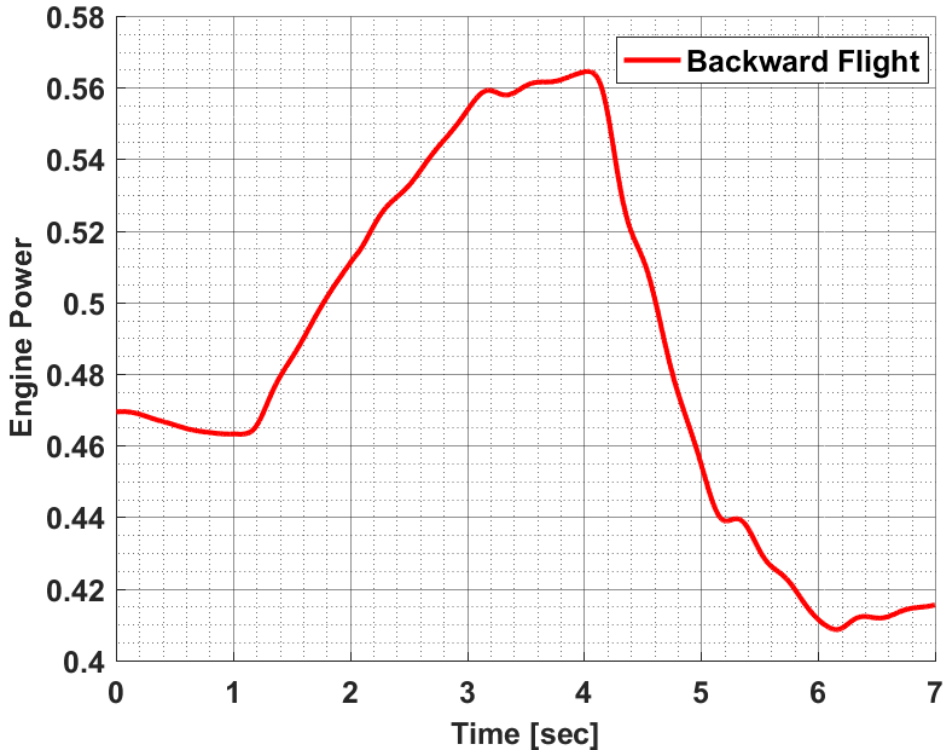
8.2.3 Optimization Solution of Hover to Backward Flight Maneuver

For the solution of hover to backward flight maneuver optimization, the symmetric rank-one method is applied with its best configuration which is at the 5 size step and 0.1 perturbation constant. Moreover, the central difference approximation is used as the gradient calculation method.

The maneuver optimization is performed under these conditions and the constraint results are given in Figure 8.1.




Backward Flight
Pitch Angle
Time [sec]
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
1
2
3
4
5
6
7


Backward Flight
Engine Power
Time [sec]
0.58
0.56
0.54
0.52
0.5
0.48
0.46
0.44
0.42
0.4
0
1
2
3
4
5
6
7

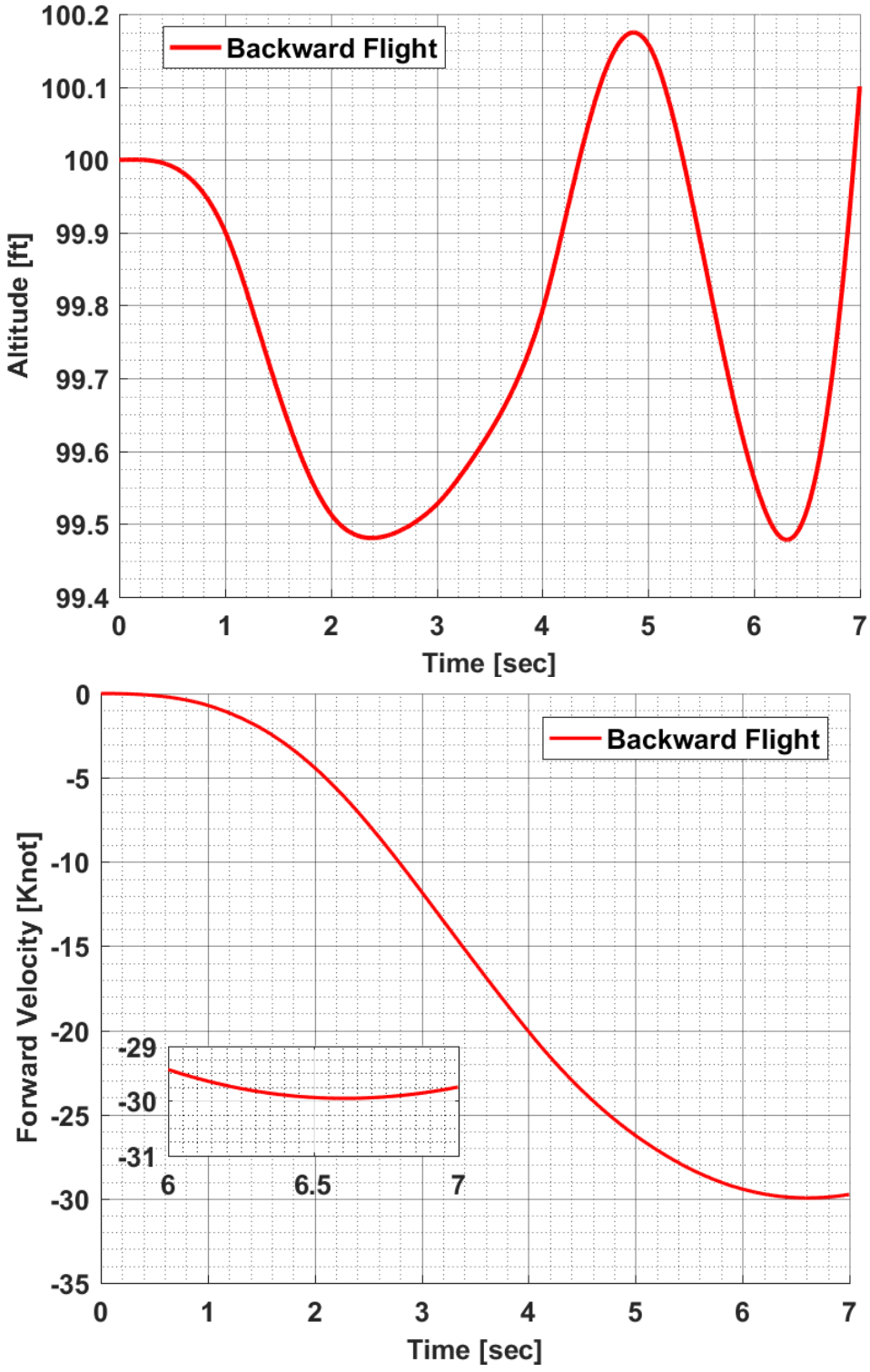
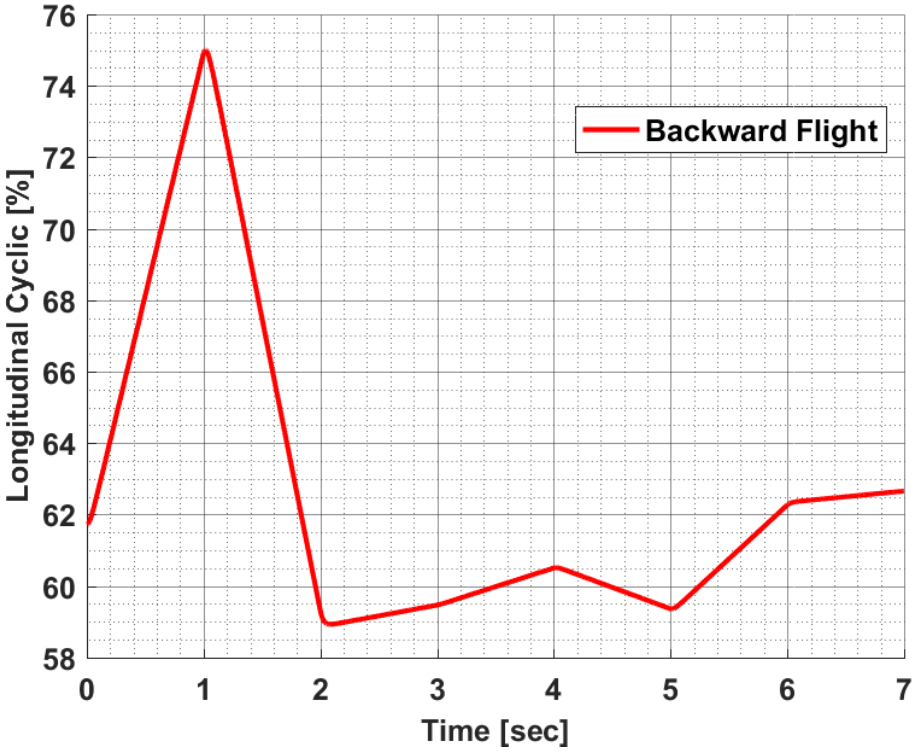


Figure 8.1: Constraint Results for Hover to Backward Flight Maneuver with SR1 Method

The charts in Figure 8.1 shows that all the constraints of hover to backward flight maneuver are met. It is seen in the altitude chart that the maneuver is performed at the almost same altitude level. In addition, the pitch angle is increased to accelerate the helicopter backwards as expected. The flap angle and engine power curves indicate that the maneuver is performed within the design limits. Furthermore, the velocity graph, the main restriction parameter, shows that the helicopter has reached 30 [Knot] backwards velocity within the specified time interval. Finally, the corresponding values of longitudinal and collective cyclic inputs, which are design variables, are given in Figure 8.2 for this maneuvering results.



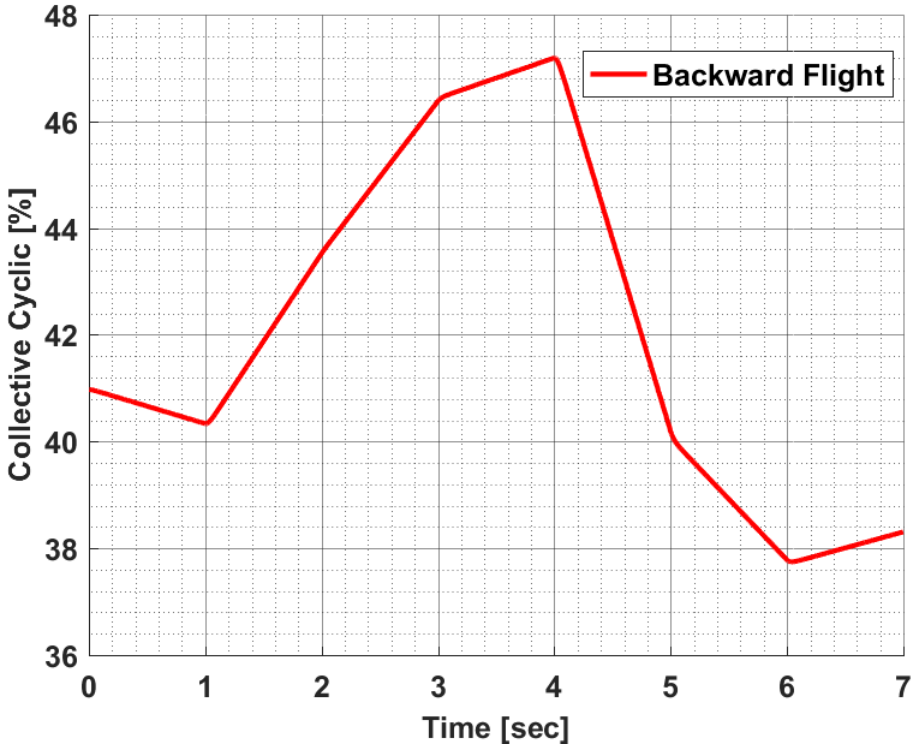


Figure 8.2: Obtained Design Variable for Hover to Backward Flight Maneuver with SR1 Method

8.3 Optimization of Hover to Leftward Flight Maneuver

8.3.1 Hover to Leftward Flight

Hover to left flight maneuver follows the same procedure which is defined in Section 6.3.2.1. The optimization process of this maneuver is same as hover to right flight maneuver. The only differences between them are the direction in which the pilot control inputs are applied and the direction of the targeted speed. In this maneuver, the helicopter is brought to hover position before accelerating in the targeted direction. Moreover, the lateral cyclic is applied to the left, so the helicopter is accelerated in that direction. The remaining maneuvering requirements are the same as hover to rightward flight maneuver.

8.3.2 Optimization Modeling of Hover to Leftward Flight Maneuver

Since hover to leftward flight maneuver is also one of hover to sideward flight maneuvers, it also used the lateral and collective cyclic as the main design variables. In addition, other pilot control inputs are controlled by SAS in this maneuver optimization.

The maneuvering time is defined as approximately 7 seconds, and design variables change every second. The main purpose of this maneuver is to reach 30 Knot leftward velocity at the almost same altitude. In addition, the helicopter design limits and maneuver requirements are defined as optimization constraints. This maneuver uses the same constraint as hover to rightward flight optimization which is defined in Section 6.3.2.2. However, the velocity constraint of this maneuver is updated according to the direction of the targeted velocity. Therefore, Constraint 4 is modified as

Constraint 4 – Leftward Velocity Constraint

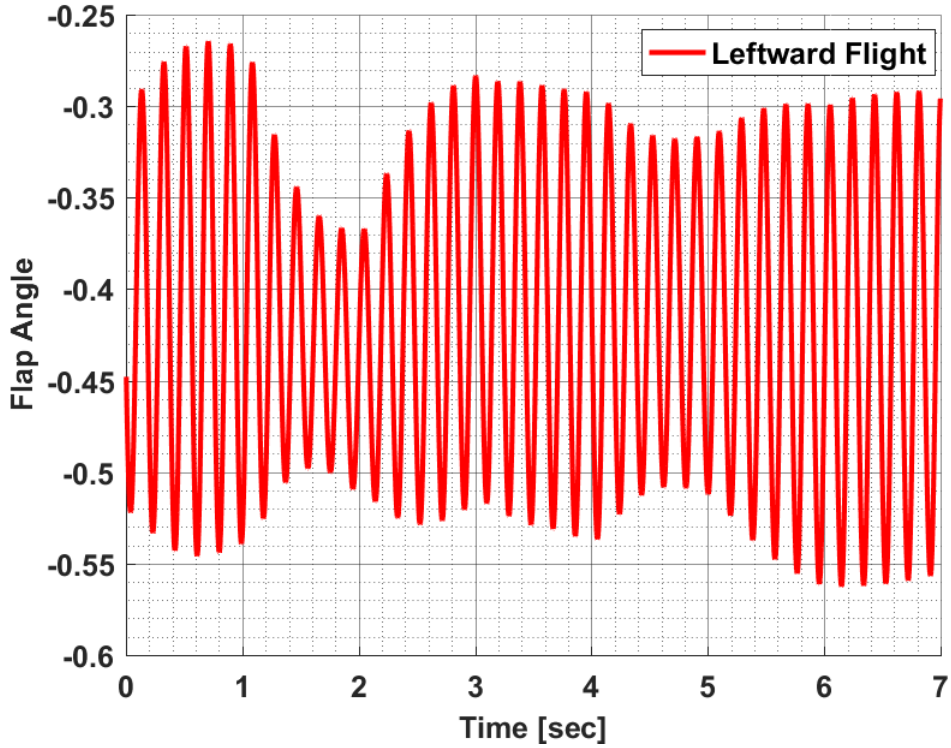
Unlike the hover to rightward flight maneuver optimization, this maneuver aims to reach 30 Knot speed in the left direction at the same time intervals. Therefore, the sign of the direction of movement is taken into account when creating the constraint as

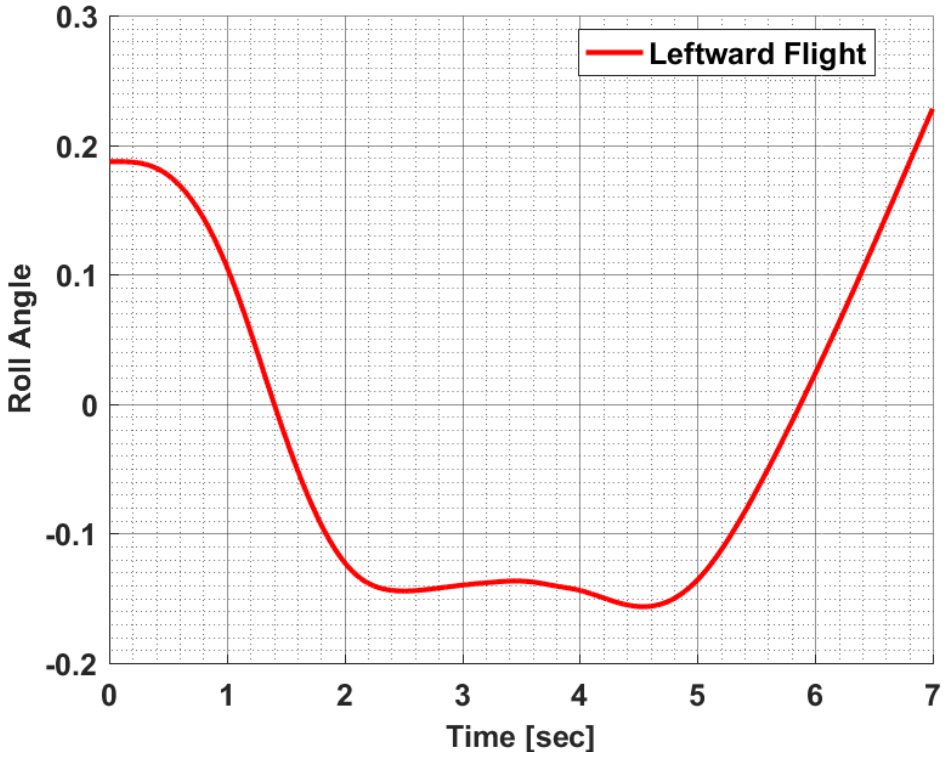
$$g_4 = \int_{t_s=6.3}^{t_f=6.7} |V + 30|^2 \, dt$$

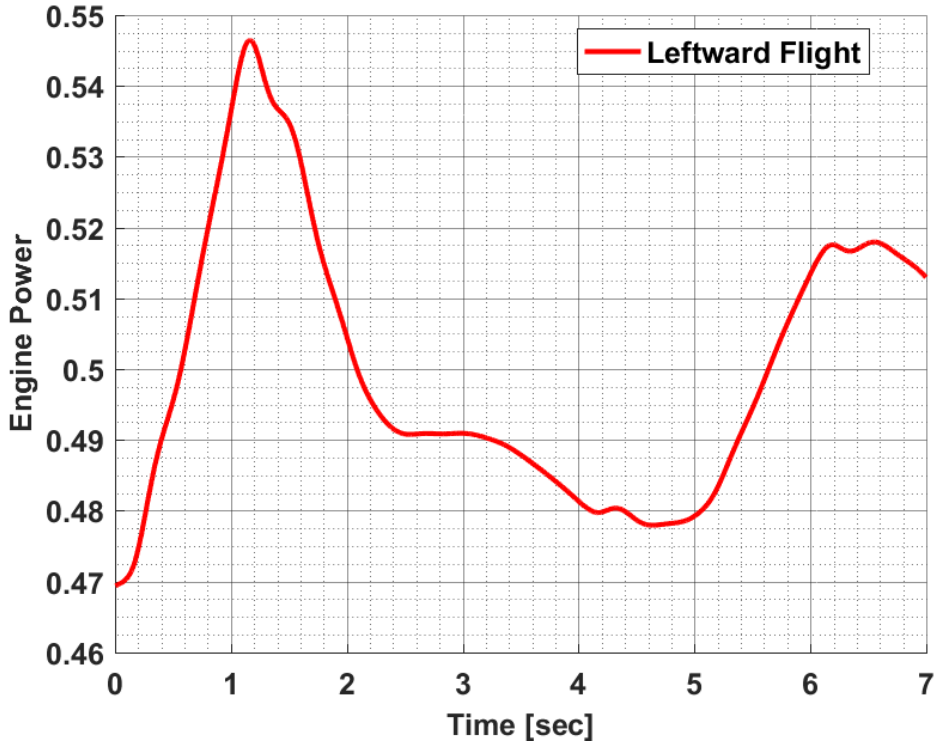
where the helicopter lateral velocity is represented by $\boldsymbol{V}$ and is considered a positive sign to the right. Furthermore, the upper and lower time boundaries where the targeted parameter is desired to be achieved are represented with $\boldsymbol{t_s}$ and $\boldsymbol{t_f}$, respectively.

8.3.3 Optimization Solution of Hover to Leftward Flight Maneuver

Hover to leftward maneuver has been carried out with the best configuration of the symmetric rank-one method and its constraints were given in Figure 8.3.




Leftward Flight
Roll Angle
0.3
0.2
0.1
0
-0.1
-0.2
0
1
2
3
4
5
6
7
Time [sec]


Leftward Flight
Engine Power
0.55
0.54
0.53
0.52
0.51
0.5
0.49
0.48
0.47
0.46
0
1
2
3
4
5
6
7
Time [sec]

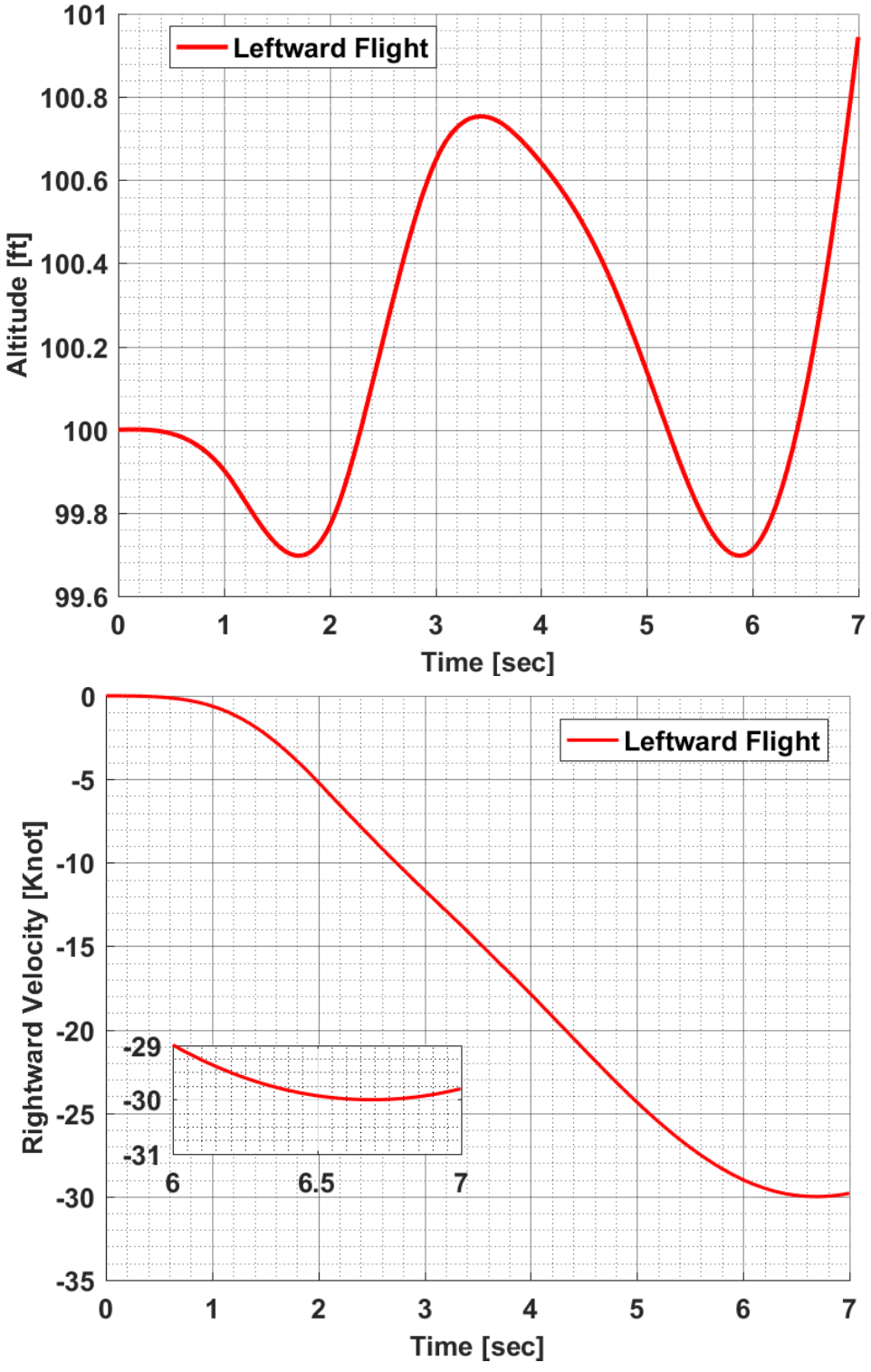
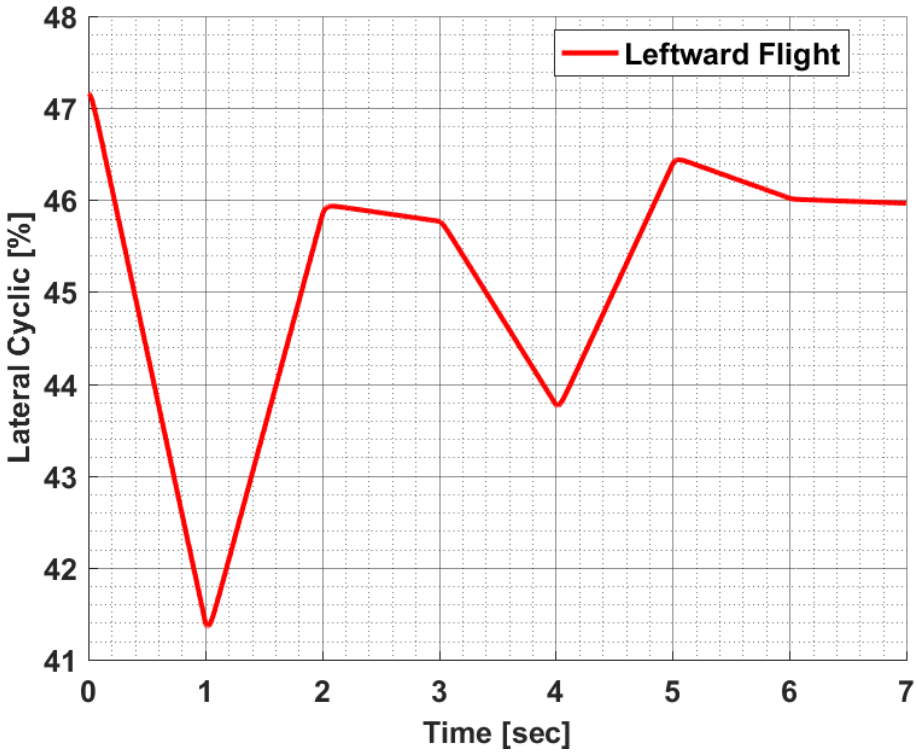


Figure 8.3: Constraint Results for Hover to Leftward Flight Maneuver with SR1 Method

The charts in Figure 8.3 shows that all the constraints of hover to leftward flight maneuver are fulfilled. The rightward velocity graph says that the helicopter has reached -30 Knot speed. The minus sign means that the helicopter is moving in the opposite direction. In other words, the helicopter has reached 30 Knot speed to the left. In addition, the roll angle is obtained as a negative value. This means that the helicopter rotates counterclockwise in the body x-axis to accelerate the helicopter to the left. Moreover, while the altitude of the helicopter is kept almost the same level, the helicopter design constraints are within the defined limits. Lastly, the corresponding design variables, which are the longitudinal and collective cyclic inputs, are given in Figure 8.4.



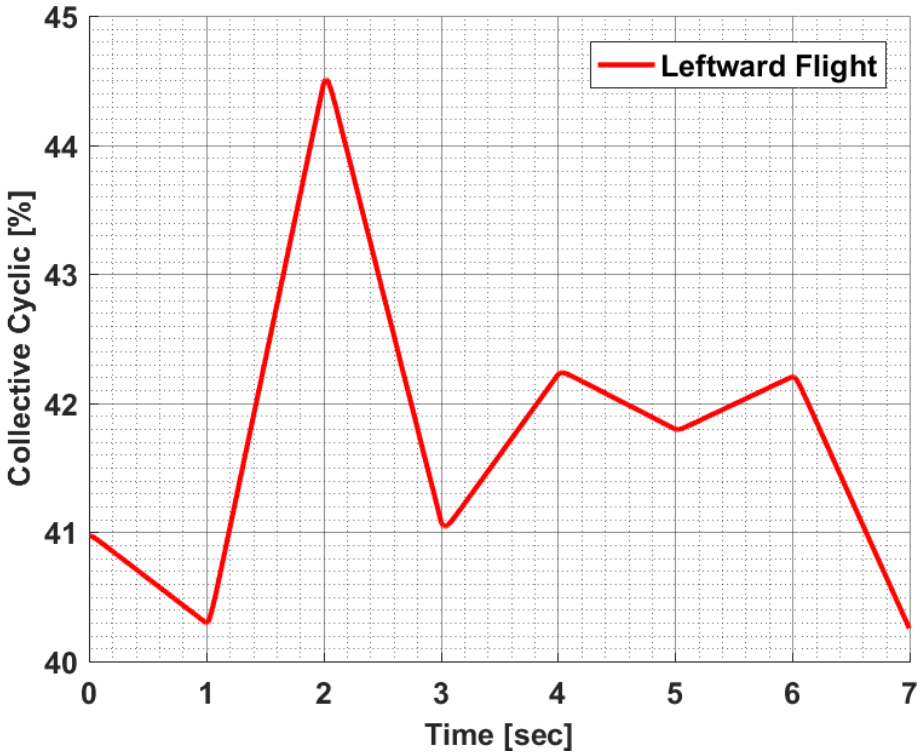


Figure 8.4: Obtained Design Variable for Hover to Leftward Flight Maneuver with SR1 Method

8.4 Optimization of Pull Up Maneuver

8.4.1 Pull Up Maneuver

Pull up maneuvers start from a level flight at a certain speed and then continue with the dive motion of the rotorcraft. The main parameter that defines this maneuver is the load factor in the helicopter body z-axis. When the target load factor is reached, the maneuver is ended and the helicopter starts to climb. The schematic representation of the pull up maneuver is given in Figure 8.5.

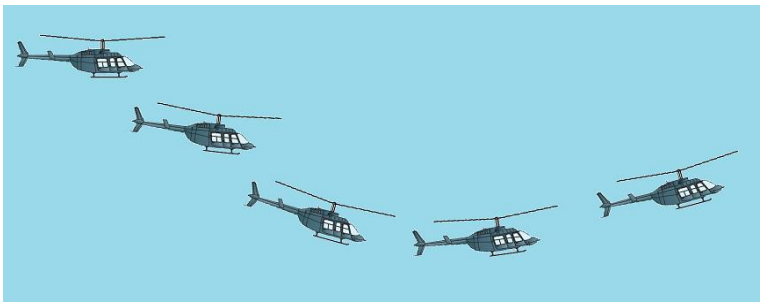


Figure 8.5: Flight View for Helicopter Pull Up Maneuver [30]

How can it be performed [30]?

1- Bring helicopter to level flight at a certain speed and an altitude
2- Apply longitudinal cyclic in the forward direction to dive. Then, it is applied in the backward direction to perform the pull up maneuver
3- Pull the collective cyclic if the aft longitudinal cyclic is inadequate in order to carry out the maneuver.
4- Apply necessary anti-torque pedal inputs to avoid any possible excessive heading changes
5- Complete the maneuver as the aimed load factor is achieved
6- All these parameters must be implemented at the same time to generate maneuver properly

8.4.2 Optimization Modeling of Pull Up Maneuver

Pull up maneuvers use the longitudinal and collective cyclic as the main pilot control inputs as defined in Section 8.4.1. Therefore, these inputs are accepted as design variables for this maneuver optimization [58]. Nevertheless, the lateral cyclic and anti-torque pedal inputs are not taken as design variables, so they are controlled by SAS.

At the beginning of this pull up maneuver optimization, the helicopter model is trimmed to the level flight at 200 [ft] altitude and 100 [Knot] forward speed. Then, the optimization process is applied to obtain the design variables over time. In order

to correctly define the design variable vector, it is necessary to decide the maneuver time and at which time points to change the design variable values. In this optimization, the maneuver takes about 6 seconds, and the design variables are changed every half second. Therefore, the design variable vector has $2 * 6 * 2 = 24$ design points as

$$\boldsymbol{X}^T = [\delta_{co}(t = 0.5), \delta_{co}(t = 1), \dots \delta_{co}(t = 6), \delta_{lo}(t = 0.5), \delta_{lo}(t = 1), \dots \delta_{lo}(t = 6)]$$

This maneuver optimization constraint the helicopter design limits and maneuver requirements. Therefore, the constraints of the blade flapping and the engine power are defined as the helicopter design limitations in the same format as the previous optimizations. In addition, since the longitudinal cyclic is one of the design variables, the pitch angle constraint is also used as described in Section 6.3.1.2. These parameters are targeted to remain within the defined limits, so the same structure in previous optimizations is also used in this optimization. Furthermore, because the pull up maneuver aims to reach the desired load factor in the body z-axis, this load factor value should be defined as a constraint. Therefore, the load factor constraint is defined as

Constraint 4 – Load Factor Constraint

Certification Specification for Large Rotorcraft says that the rotorcraft must be designed for limit maneuvering load factor ranging from a positive limit of 3.5 to a negative limit of -1. However, if the helicopter design load factors do not reach these limit load factors, the probability of being exceeded must be shown by analysis and flight tests to be extremely remote [2]. In order to demonstrate the usefulness of the symmetric rank-one method on this type of limit maneuver, this maneuver optimization targets to reach 3.5 load factor in the body z-axis between 4.4^{th} and 4.6^{th} seconds. When the targeted load factor is achieved, the constraint will be met. Even though this parameter seems helicopter design limitation, it is actually one of the maneuvering constraints. Hence, the constraint is formulated as

$$g_4 = \int_{t_s=4.4}^{t_f=4.6} |n_z - 3.5|^2 \, dt$$

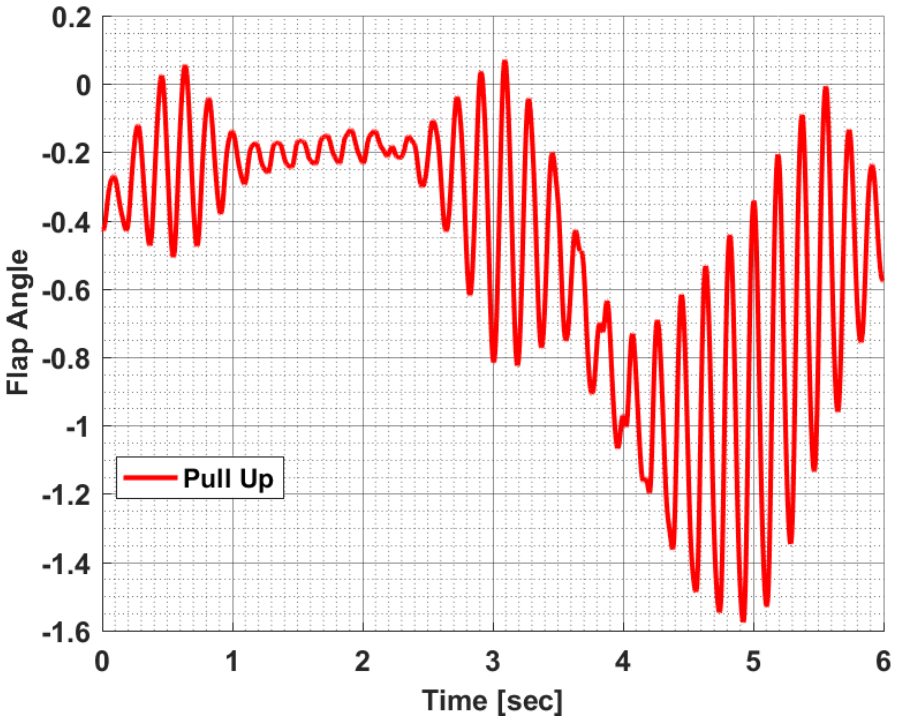
where the helicopter load factor in body z axis is represented by $\boldsymbol{n_z}$. Also, the upper and lower time boundaries where the targeted parameter is desired to be obtained are represented with $\boldsymbol{t_s}$ and $\boldsymbol{t_f}$ respectively.

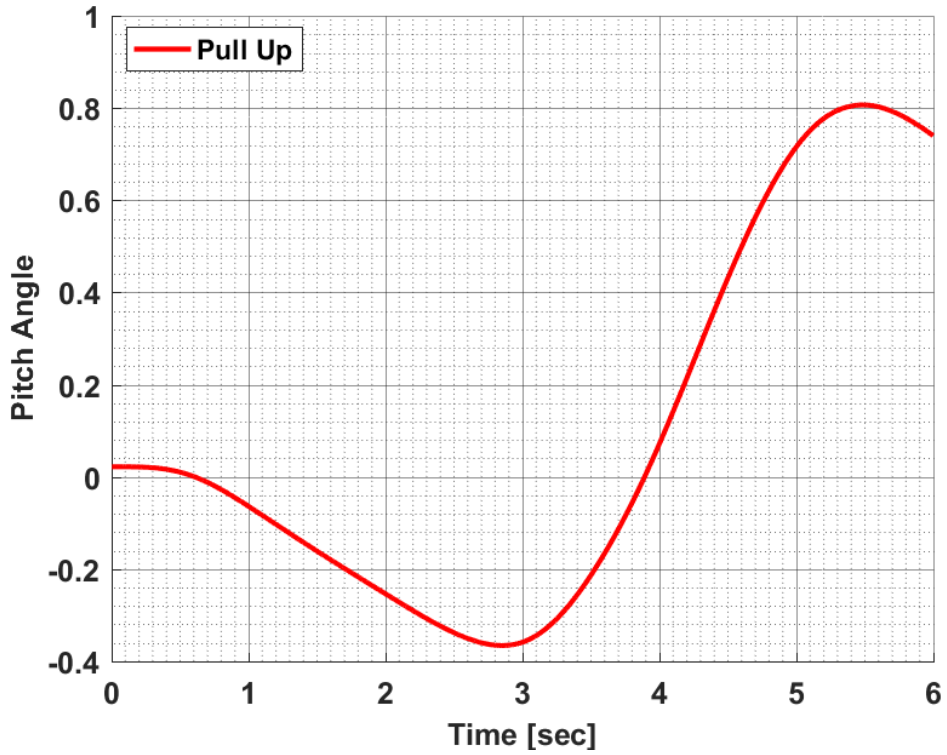
While creating the objective function, the penalty parameters p_1, p_2 and p_3 are taken as 5 and p_4 is taken 100. Thus, the load factor constraint is defined as the most critical constraint. Then, the objective function is structured as:

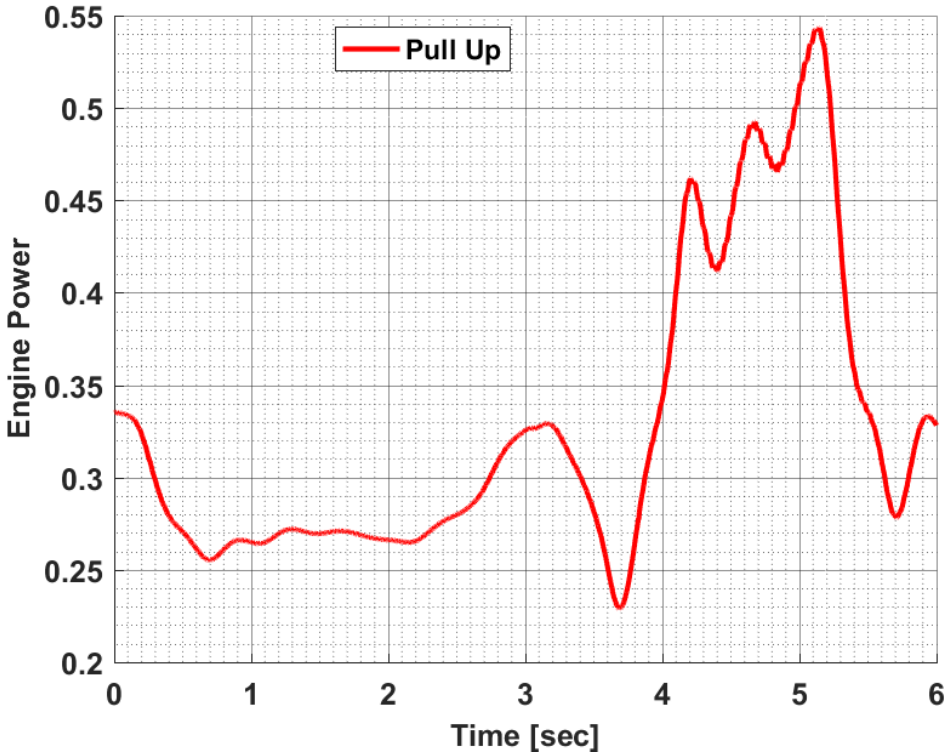
$$f(X) = p_1 g_1 + p_2 g_2 + p_3 g_3 + p_4 g_4$$

8.4.3 Optimization Solution of Pull Up Maneuver

The pull up maneuver is executed with the best configuration of the symmetric rank-one method, which is defined in Section 6.4. Then, the constraint results and altitude change of the maneuver for this maneuvering optimization are given in Figure 8.6.




Pull Up
Pitch Angle
Time [sec]
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
0
1
2
3
4
5
6


Pull Up
Engine Power
Time [sec]
0.55
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0
1
2
3
4
5
6

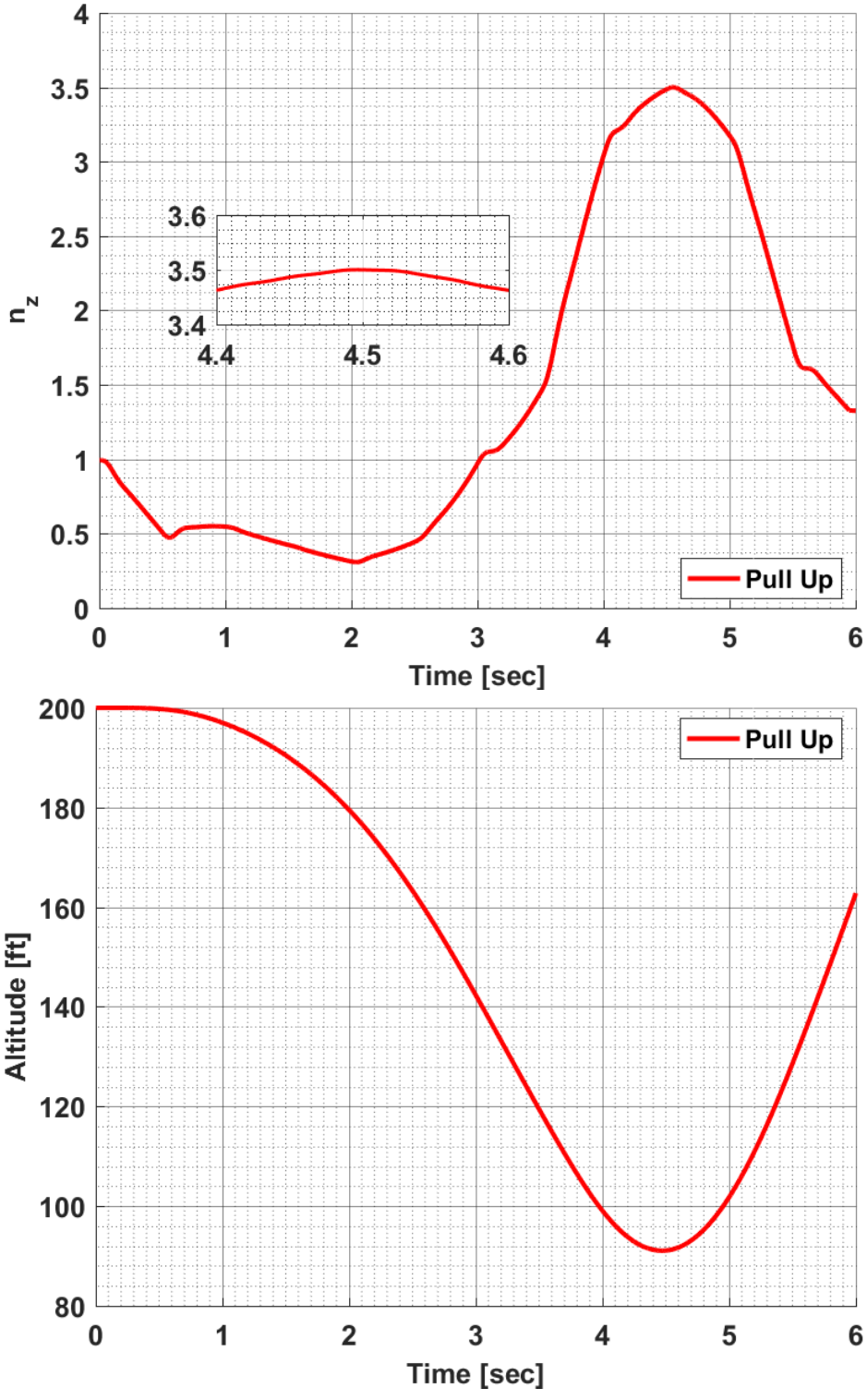


Figure 8.6: Constraint Results for Pull Up Maneuver with SR1 Method

Figure 8.6 shows that all the constraints of the pull up maneuver are within the defined range. The load factor chart includes its zoomed view at the time interval where the constraint is defined. According to this chart, it is seen that the targeted load factor is obtained within the defined range. In addition, starting the pitch angle chart in a negative direction means that the nose of the helicopter is turned downward for diving at the beginning of the maneuver. Then, it finishes diving and starts to climb. Moreover, the constraints of the flap angle and the engine power are within the design limits. However, as seen from the engine power chart, the helicopter had to use more power to reach the target load factor value. This means that the collective cyclic is pulled because it does not reach the desired value with the longitudinal cyclic. Furthermore, the altitude change graph says that the helicopter started to maneuver at 200 [ft] altitude. Then, it followed a similar flight path as shown visually in Figure 8.5.

In the pull up maneuver optimization, the longitudinal and collective cyclic are used as design variables. Hence, the corresponding design variable inputs are given in Figure 8.7.

Figure 8.7 says that the collective cyclic is pulled simultaneously with the longitudinal cyclic to reach the desired load factor within the defined range.

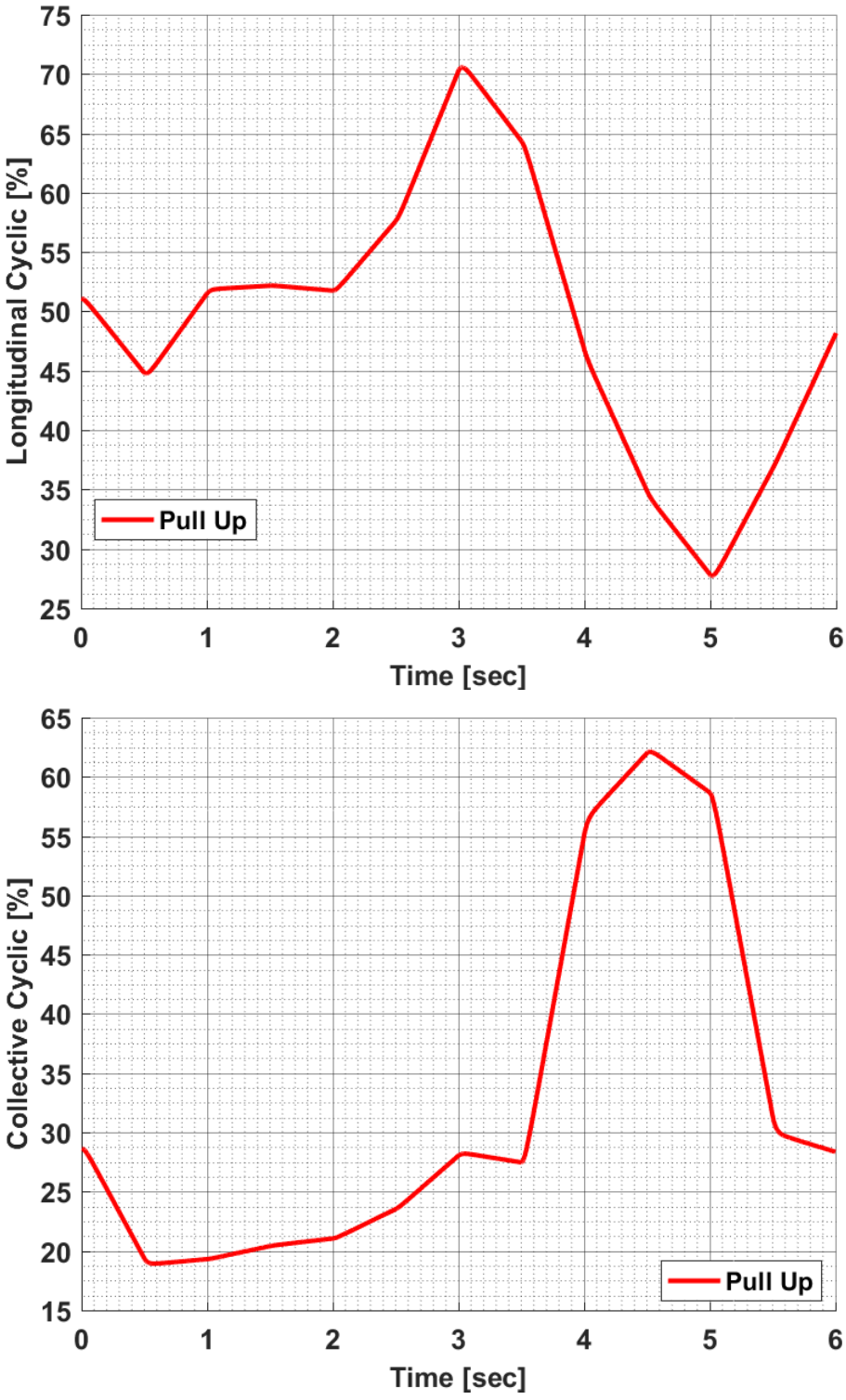


Figure 8.7: Obtained Design Variable for Pull Up Maneuver with SR1 Method

8.5 Optimization of Pushover Maneuver

8.5.1 Pushover Maneuver

Similar to the pull up maneuvers, pushover maneuvers take the level flight at a certain speed as a starting point. However, these maneuvers climb instead of diving to reach the desired load factor value in the helicopter body z-axis. Once the target load factor is reached, the helicopter terminates the climbing trend and starts to descend. Moreover, pushover maneuvers are carried out with the same pilot control inputs used to pull up maneuvers. However, the pilot control inputs are applied in different directions in these maneuvers. In short, the pushover maneuvers are similar to the pull up maneuvers except for the target load value and the flight path of the helicopter. Therefore, optimization structure is created in a similar way by taking these parameters into consideration.

8.5.2 Optimization Modeling of Pushover Maneuver

Since pushover maneuvers are performed with the same pilot control inputs which are used in pull up maneuvers, the design variables are also selected as the same inputs. Namely, the longitudinal and collective cyclic are defined as the design variables in pushover maneuver optimization. Moreover, SAS controls the lateral cyclic and anti-torque pedal inputs during the optimization.

In order to start the optimization process, the helicopter model is first trimmed to the level flight at 200 [ft] altitude and 100 [Knot] forward speed. Then, the required pilot control inputs are sought to perform the targeted maneuver. For this purpose, the maneuver time is set at 3 seconds and the design variables are changed every half second. Therefore, the design variable vector has $2 * 3 * 2 = 12$ design points as

$$\boldsymbol{X}^T = [\delta_{co}(t = 0.5), \delta_{co}(t = 1), \dots \delta_{co}(t = 3), \delta_{lo}(t = 0.5), \delta_{lo}(t = 1), \dots \delta_{lo}(t = 6)]$$

As it is explained in Section 8.5.1, the main difference between pushover and pull up maneuvers is the targeted load factor value and the required pilot control input values for this. Therefore, while the same pilot control inputs are defined as the design variables, the load factor constraint is modified according to pushover maneuver requirement as

Constraint 4 – Load Factor Constraint

In the pull up maneuver optimization, it is targeted to reach 3.5 load factor in the body z-axis according to CS 29 [2]. Due to the same reason, this pushover optimization problem aims to achieve -1 load factor in the body z-axis between 1.78th and 1.82th seconds. Therefore, the constraint formula is modified by taking into account the positive sign direction of the parameter as

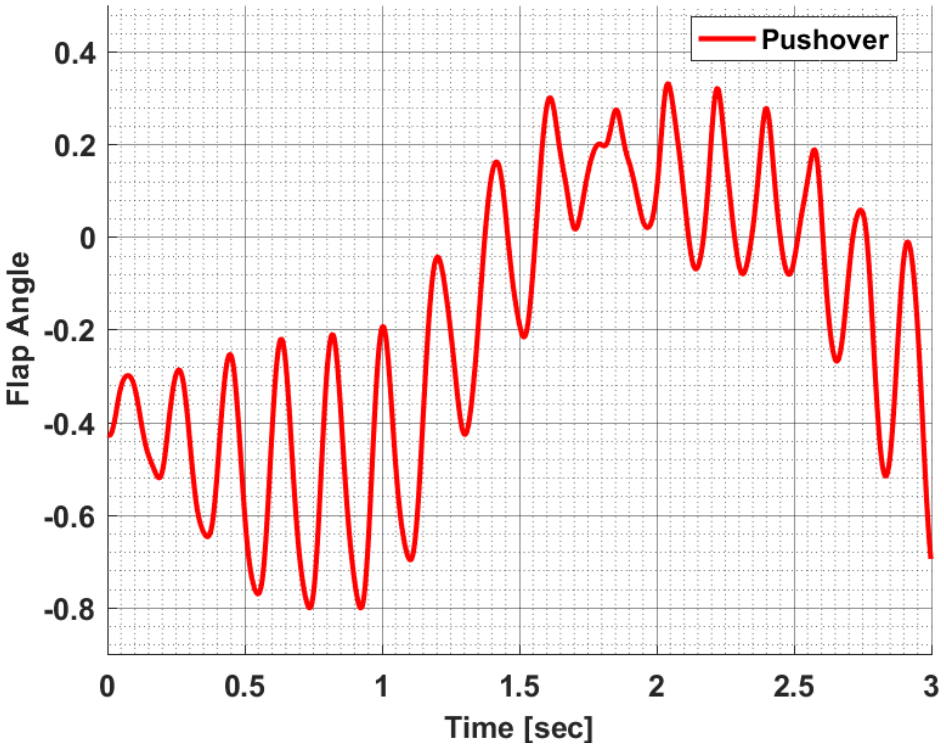
$$g_4 = \int_{t_s=1.78}^{t_f=1.82} |n_z + 1|^2 \, dt$$

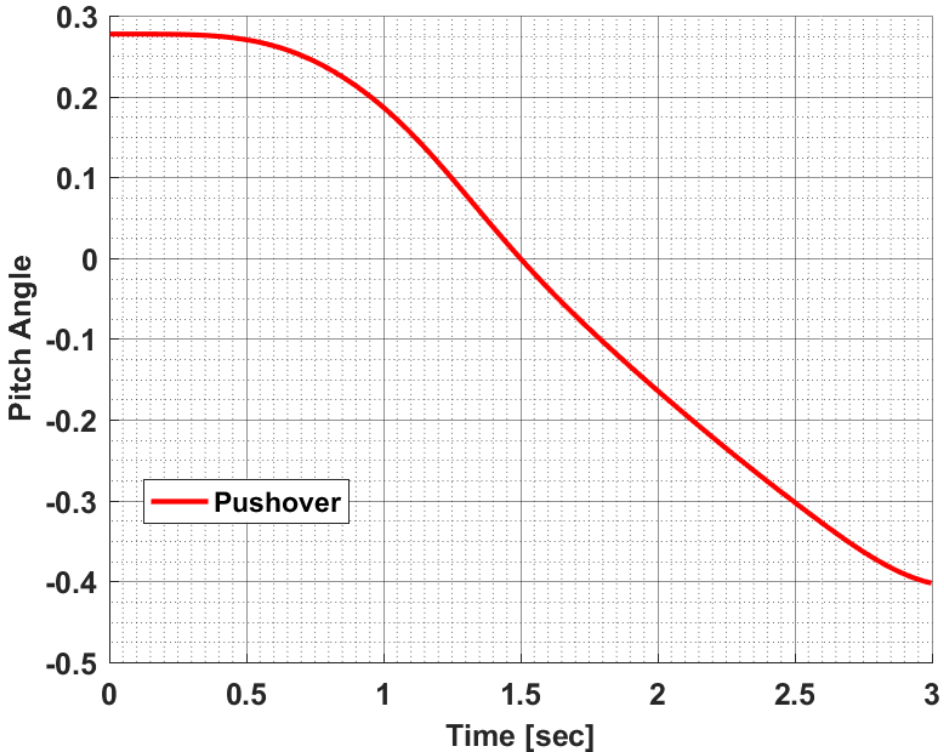
where the helicopter load factor in body z axis is represented by $\boldsymbol{n_z}$. Also, the upper and lower time boundaries where the targeted parameter is desired to be obtained are represented with $\boldsymbol{t_s}$ and $\boldsymbol{t_f}$ respectively.

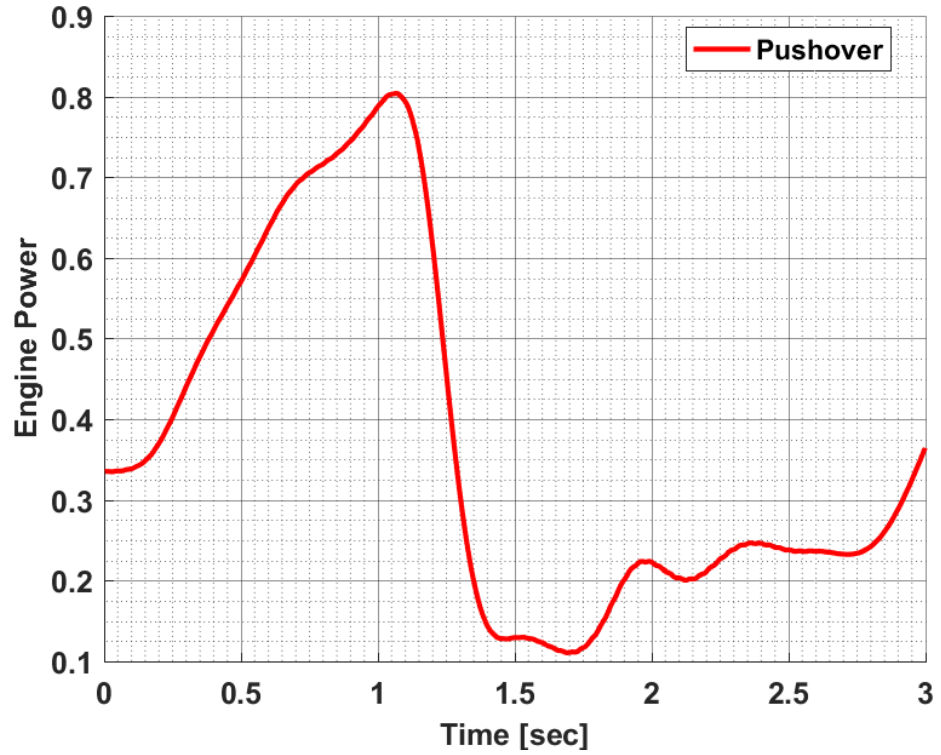
The other constraints and objective structure are used in the same way as defined in the pull up maneuver.

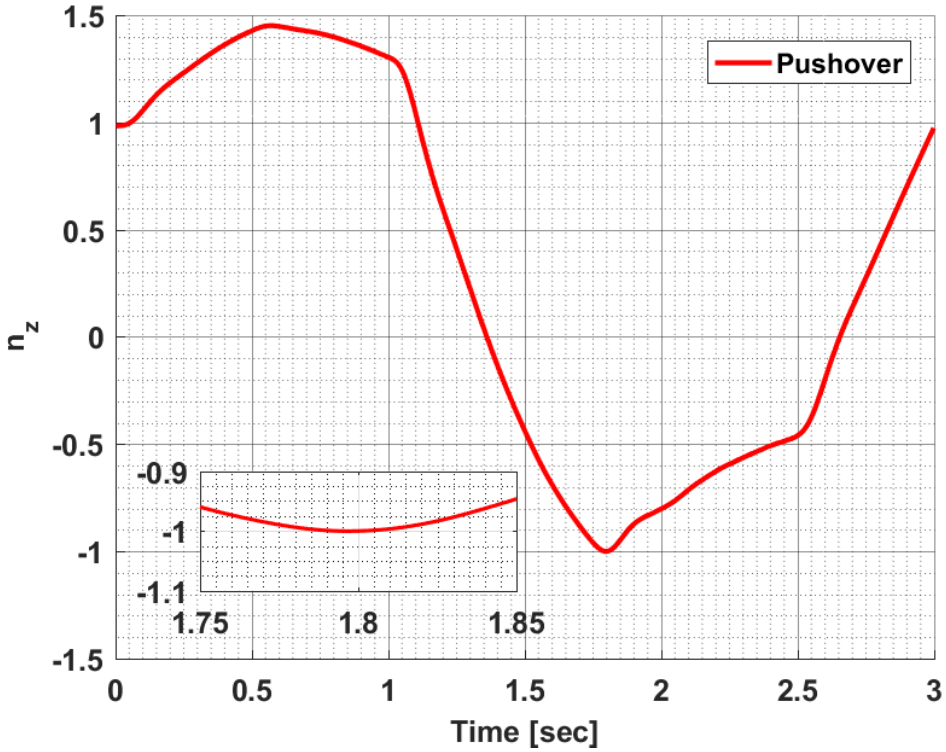
8.5.3 Optimization Solution of Pushover Maneuver

The pushover maneuver is carried out with the best configuration of the symmetric rank-one method, which is defined in Section 6.4. After the optimization process is completed, the constraint results and altitude change of the maneuver for this maneuvering optimization are given in Figure 8.8.


Pushover
Flap Angle
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
0
0.5
1
1.5
2
2.5
3
Time [sec]


Pushover
Pitch Angle
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
0
0.5
1
1.5
2
2.5
3
Time [sec]


Pushover
Engine Power
Time [sec]
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.5
1
1.5
2
2.5
3


Pushover
n_z
Time [sec]
1.5
1
0.5
0
-0.5
-1
-1.5
-0.9
-1
-1.1
1.75
1.8
1.85
0
0.5
1
1.5
2
2.5
3

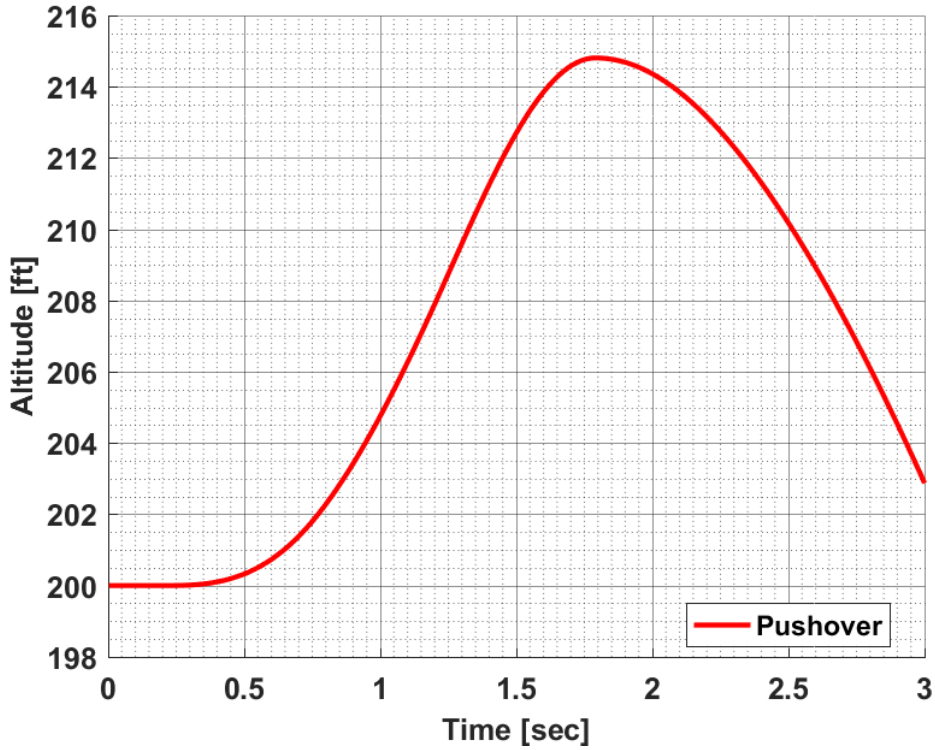


Figure 8.8: Constraint Results for Pushover Maneuver with SR1 Method

Figure 8.8 shows that any of the constraints of the pull up maneuver does not exceed the defined limits. Moreover, it can be seen from the zoomed view of the load factor chart that the targeted load factor values are achieved within the defined range. In addition, the engine power graph shows that the helicopter reduces the used power to reach the target negative load value. There is a parallel relationship between motor power and collective cyclic inputs. In other words, when the collective cyclic is pushed, the used engine power decreases. Therefore, the decreasing engine power in the specified range means that the collective cyclic is pushed to achieve the desired load factor value. In addition, the altitude change graph says that the helicopter starts maneuvering at an altitude of 200 [ft] and climb in contrast to pull up to reach the desired value.

The design variable of the pushover maneuvers is the longitudinal and collective cyclic. Therefore, the corresponding design variable inputs along time are plotted in Figure 8.9.

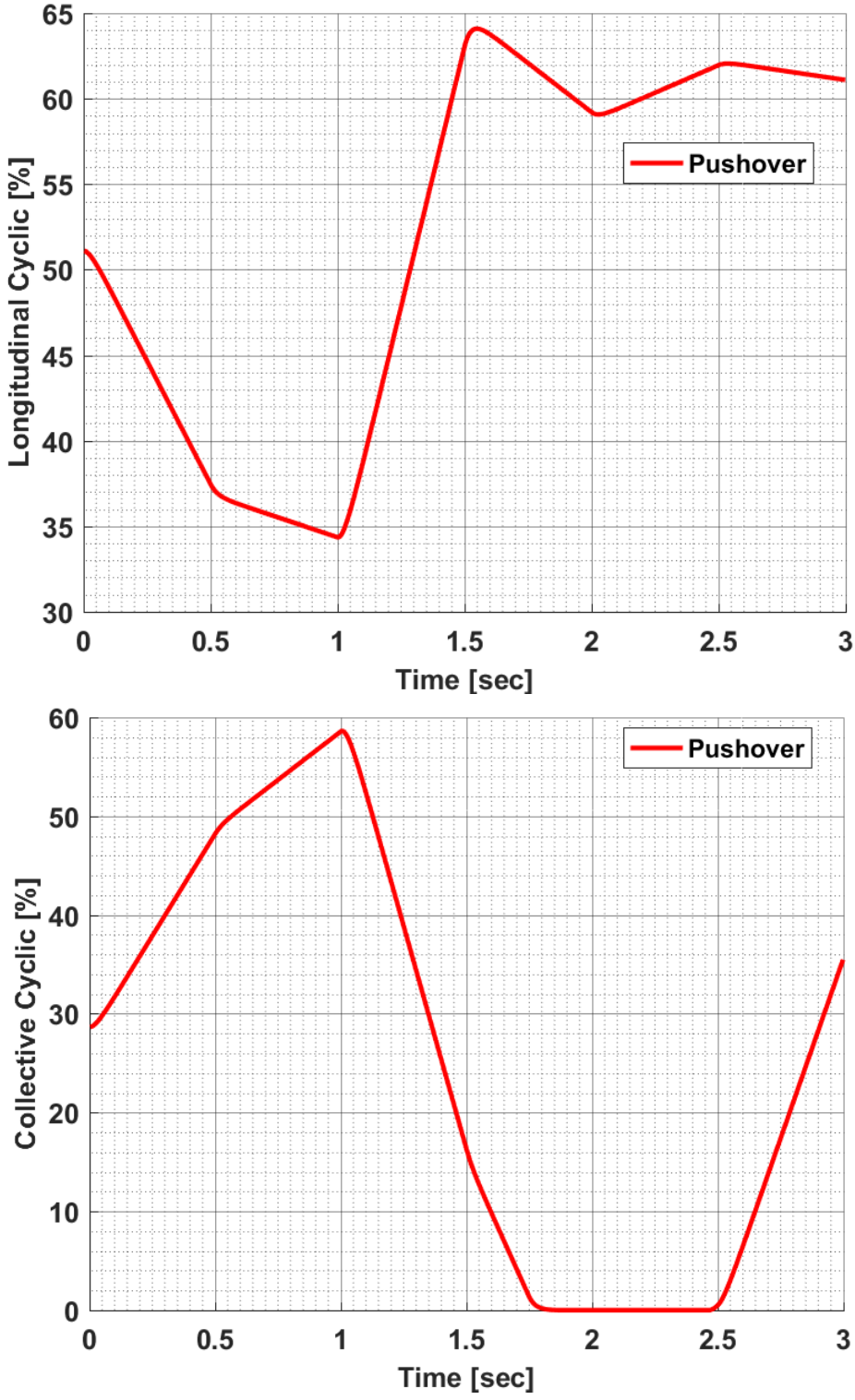


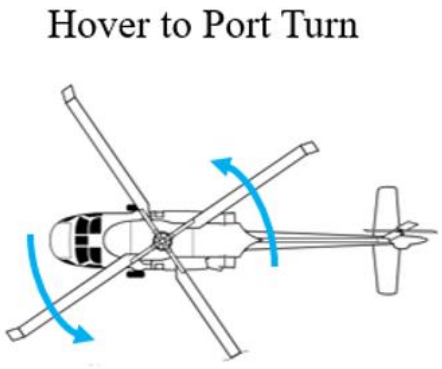
Figure 8.9: Obtained Design Variable for Pushover Maneuver with SR1 Method

According to Figure 8.9, the collective cyclic has been pushed to its limit position within the defined range. This means that the pilot must eliminate the influence of collective input to achieve the target value in defined restrictions. Since this maneuver is the reverse of the pull up maneuver, the collective cycle must be pushed in this maneuver.

8.6 Optimization of Hovering Turn Maneuvers

Hovering turn maneuvers are the maneuvers performed at a certain altitude to rotate the nose of the helicopter left or right as seen in Figure 8.10. In other words, these maneuvers aim to turn around their axis at a certain altitude without displacement. The maneuvers are carried out by the coordination of all flight controls. Moreover, the maneuver should be maintained at a fixed altitude and a constant turn rate.

In hovering turn maneuver optimization, the helicopter model is trimmed to a hover position at a certain altitude. Then, the rotation of the helicopter in the desired direction is achieved by providing the required pilot control inputs. However, it must be remembered that the altitude must be maintained during the turn.



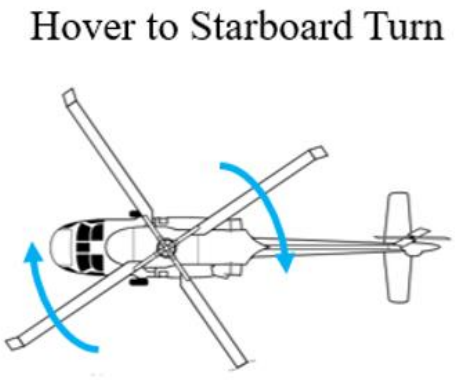


Figure 8.10: Schematic Representation of Hovering Turns

How can it be performed [54]?

1- Bring helicopter to hover position at a specific altitude
2- Apply the required anti-torque pedal inputs to rotate the helicopter at the desired turn rate in the planned direction
3- Apply required collective cyclic inputs to maintain the altitude
4- Apply necessary longitudinal and lateral cyclic inputs to avoid any possible accelerations
5- Complete the maneuver when the desired turn is achieved
6- All these parameters must be implemented at the same time to generate maneuver properly

The implementation of the anti-torque pedal inputs varies according to the direction the helicopter is intended to turn. Therefore, the optimization structure is configured separately for hover to port turn and hover to starboard turn maneuvers.

8.6.1 Optimization Modeling of Hover to Port Turn Maneuver

The main purpose of hovering turn maneuvers is to maintain altitude while turning the nose of the helicopter in the desired direction. Therefore, the anti-torque pedal and collective cyclic are used as the design variables in the optimization of these maneuvers. Furthermore, the lateral and longitudinal cyclic are controlled by SAS,

so they are not taken as the design variables. Hence, the pilot control inputs are defined in accordance with this logic for hover to port turn maneuver optimization.

In this optimization, the maneuver is targeted to perform in 8 seconds. Also, the design variables are modified every half second. Hence, the design variable vector is formulated with $2 * 8 * 2 = 32$ design points as

$$\boldsymbol{X}^T = \left[\delta_{co}(t = 0.5), \delta_{co}(t = 1), \dots \delta_{co}(t = 8), \delta_{ap}(t = 0.5), \delta_{ap}(t = 1), \dots \delta_{ap}(t = 8)\right]$$

In the optimization of this maneuver, the constraints of the blade flap and the engine power are defined as helicopter design limits. Therefore, this maneuver optimization also uses the same formulation for them as created in previous maneuver optimizations. Moreover, since the maneuver requires a fixed altitude and a constant turn rate, they are described as the constraints of the maneuver requirements. In summary, Constraints 1 and 3 are assigned as the constraints of the blade flap and the engine power, respectively. The other constraints are created as

Constraint 2 – Turn Rate Constraint

Yaw angle is the angle represents how many degrees the helicopter turns left or right on its own vertical axis. Therefore, the yaw angle rate shows how fast the helicopter rotates. In this maneuver optimization, it is targeted to reach the turn rate value of -60 [°/s] at 1.5th second and then continue the maneuver at this value. The minus sign originates from the axis system and represents that the helicopter turns to port. Therefore, the constraint is mathematically formulated as

$$g_2 = \int_{t_s=1.5}^{t_f=8} |\dot{\Psi} + 60|^2 \, dt$$

where the yaw angle rate of the helicopter is represented by $\dot{\Psi}$**.** Also, the upper and lower time boundaries where the targeted parameter is desired to be obtained are represented with $\boldsymbol{t_s}$ and $\boldsymbol{t_f}$ respectively.

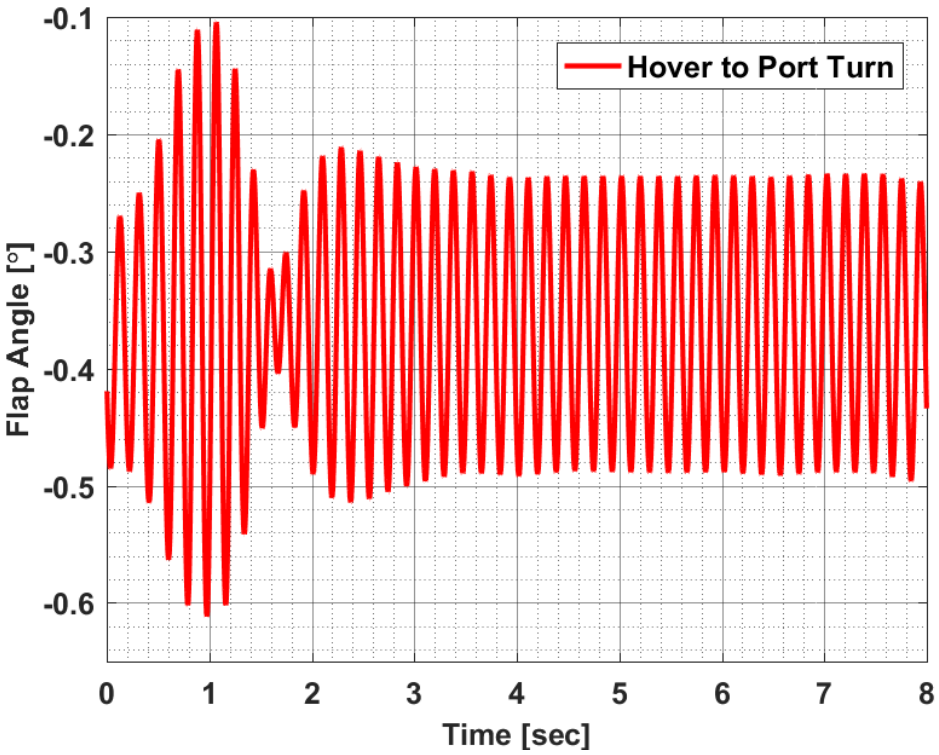
Constraint 4 – Altitude Constraint

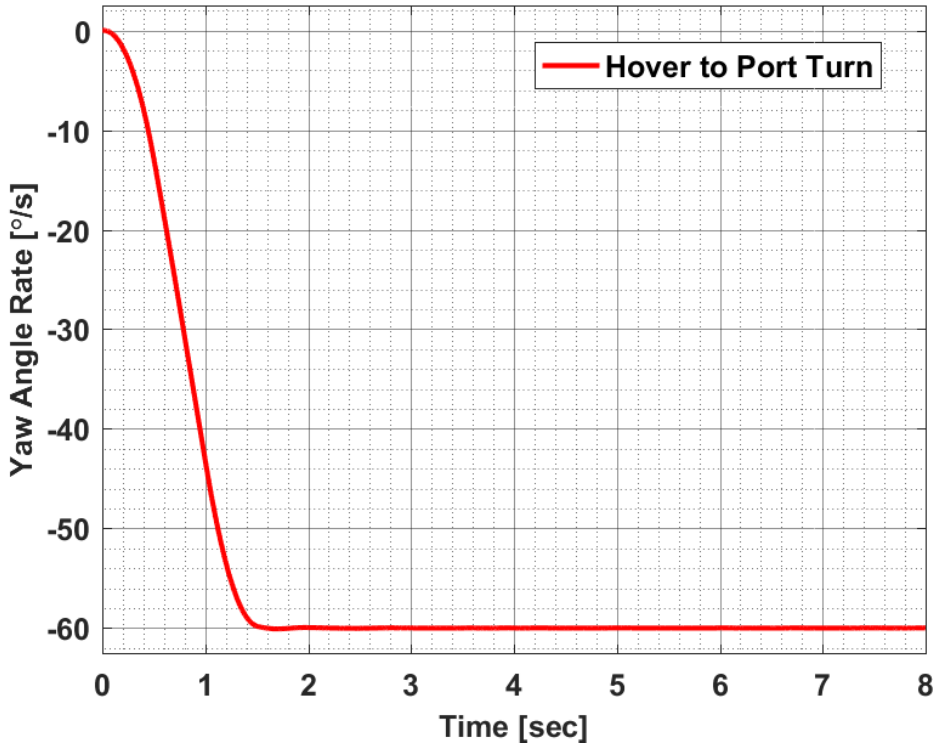
Another maneuver condition for this maneuver is to perform the maneuver at almost the same altitude. Therefore, the same constraint structure is used as described in Section 6.3.1.2. However, this optimization is carried out at 200 [ft] altitude and the helicopter is allowed to ascend/descend at most 5 [ft] maximum. Therefore, the allowed maximum altitude limits ($\boldsymbol{h_{max}}$) is defined as $\boldsymbol{h_{max}} = 105\ [ft]$ and the minimum altitude limits ($\boldsymbol{h_{min}}$) is defined as $\boldsymbol{h_{min}} = 95\ [ft]$. Finally, the same constraint formulation is used for the helicopter altitude $\boldsymbol{h}$ as

$$g_5 = \begin{cases} -log\big(h_{max} - max(h)\big) - log\big(h_{min} + min(h)\big) & for \quad max(h) < h_{max}\ and\ min(h) > h_{min} \\ Infinity & for \quad otherwise \end{cases}$$

8.6.2 Optimization Solution of Hover to Port Turn Maneuver

Hover to port turn maneuver is performed with the best configuration of the SR1 method, which is defined in Section 6.4. Then, the constraint results of the maneuver for this maneuvering optimization are given in Figure 8.11.


Hover to Port Turn
Flap Angle [°]
-0.1
-0.2
-0.3
-0.4
-0.5
-0.6
0
1
2
3
4
5
6
7
8
Time [sec]


Hover to Port Turn
Yaw Angle Rate [°/s]
0
-10
-20
-30
-40
-50
-60
0
1
2
3
4
5
6
7
8
Time [sec]

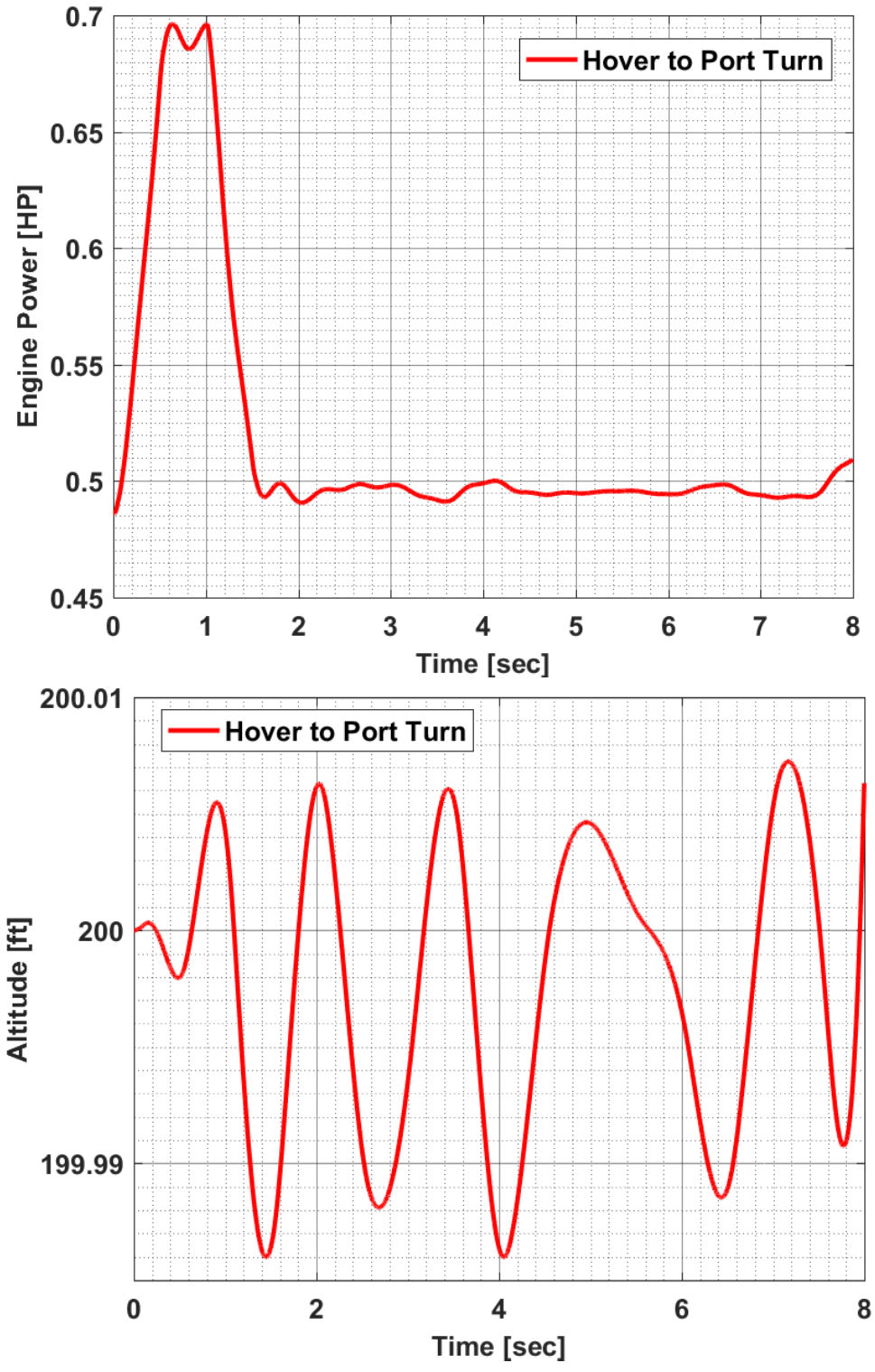
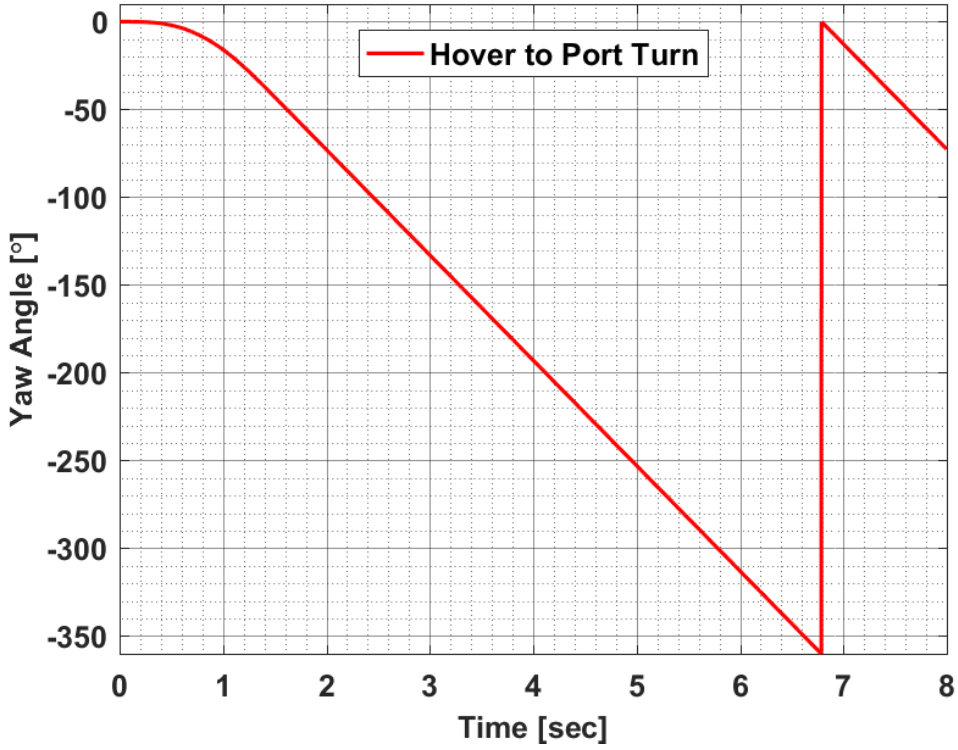


Figure 8.11: Constraint Results for Hover to Port Turn Maneuver with SR1 Method

Figure 8.11 shows that all the constraints of this maneuver optimization are fulfilled. In addition, the flap angle and engine power graphs are between specified limit values. According to the altitude chart, it can be said that the altitude of the helicopter has almost never changed. Furthermore, the yaw angle rate graph represents that the helicopter reached the port turn rate value of 60 [°/s] at 1.5th second and then continue the maneuver at that value. In addition to these constraint graphs, the changes of the velocity values in three directions and the yaw angle are plotted in Figure 8.12.



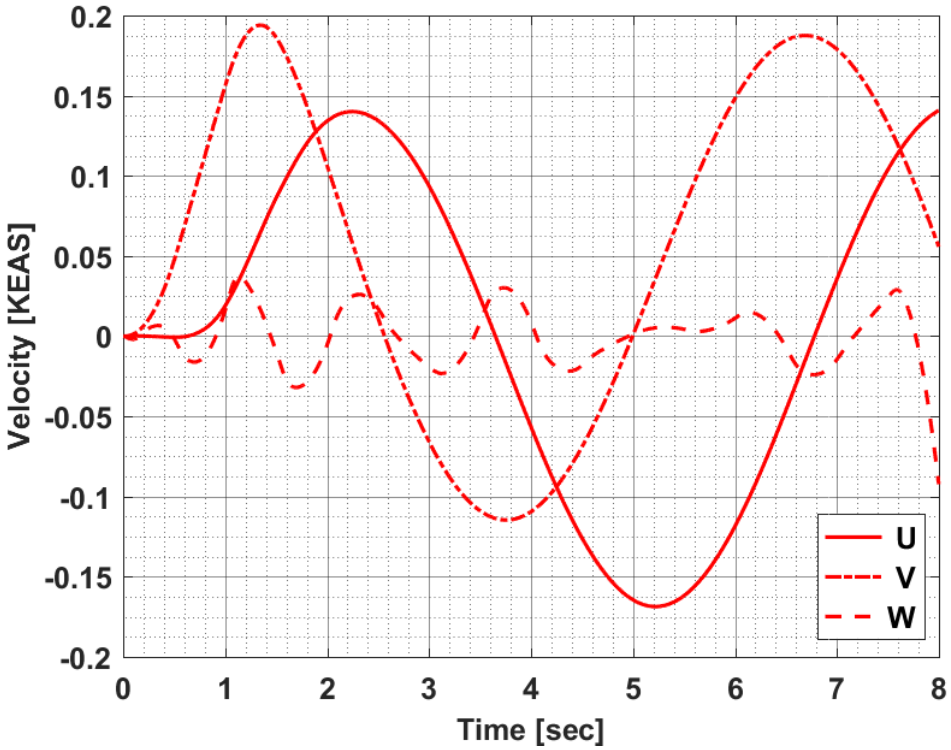


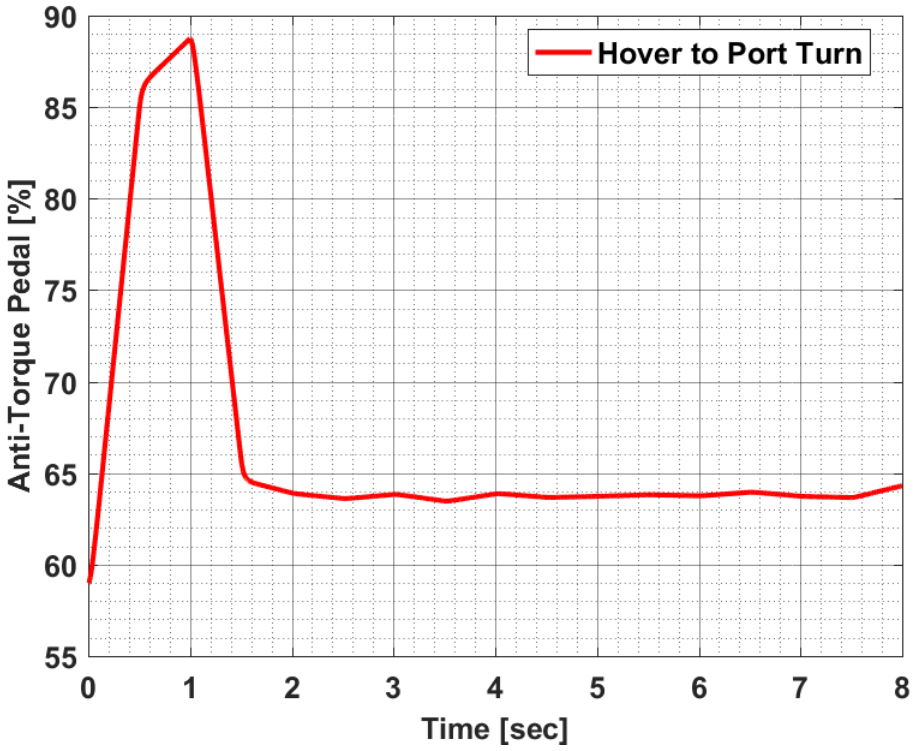
Figure 8.12: Maneuver Parameters for Hover to Port Turn Maneuver with SR1 Method

The velocity values in three directions and the yaw angle are not constrained in this optimization because the velocity is controlled by the SAS and the yaw angle is a natural result of the maneuver. Nevertheless, the results of these parameters are investigated to check the accuracy of the maneuver optimization. The negative sign on the yaw angle graph means that helicopter is turning to the left. According to this graph, it is observed that the yaw angle increases to 360° with a constant slope up to about 6.8^{th} second. This constant slope proves that the rate of rotation does not change. When the yaw angle reaches 360°, the helicopter will be back to the starting direction. Since the yaw angle takes the starting point as a reference, the yaw angle graph jumps to zero when the helicopter takes a full turn. Then, the graph continues to increase with the same slope because the turn rate is still the same. Moreover, this maneuver aims to rotate the helicopter around its own vertical axis at a certain altitude without moving. The velocity graph shows that the velocity change in three

directions is small enough to be neglected. This means that the helicopter does not move in any direction.

The anti-torque pedal and collective cyclic are defined as design variables in this maneuvering problem. Figure 8.13 shows the required anti-torque pedal and collective cyclic input for hover to port turn optimization whose results are given in Figure 8.11 and Figure 8.12.

As it is mentioned in Section 5.2 that the anti-torque pedal input must be increased in percent to turn the helicopter to the left. Also, the helicopter has been turned in the direction of the port by applying the anti-torque pedal value higher than the trim value.



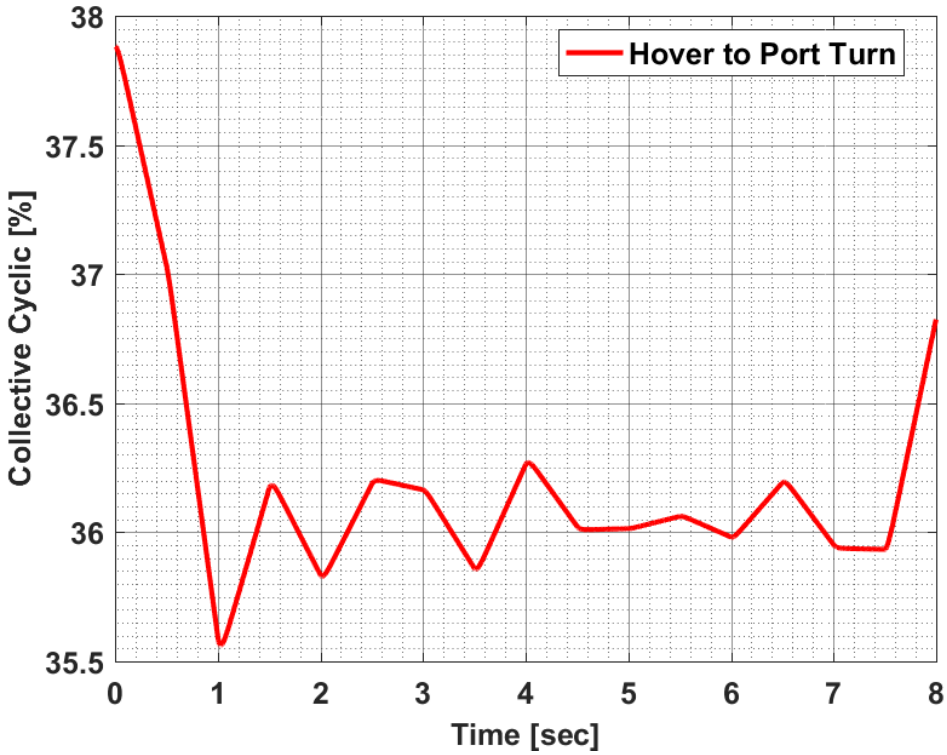


Figure 8.13: Obtained Design Variable for Hover to Port Turn Maneuver with SR1 Method

8.6.3 Optimization Modeling of Hover to Starboard Turn Maneuver

The technique of hover to starboard turn maneuver is the same as that of hover to port turn maneuver. The only difference between them is that hover to starboard turn maneuver aims to turn the helicopter right, unlike hover to port turn maneuver. Therefore, the implementation of the pilot control inputs required for this maneuver will naturally be different. In short, this optimization maneuver also uses the anti-torque pedal and collective cyclic as the design variables. Moreover, the lateral and longitudinal cyclic are controlled by SAS.

In this maneuver optimization, the maneuver time, design variables and constraints are formulated with the same values and structure as hover to port turn maneuver

optimization. However, the constraint of the turn rate is modified according to the right turn rate as

Constraint 2 – Turn Rate Constraint

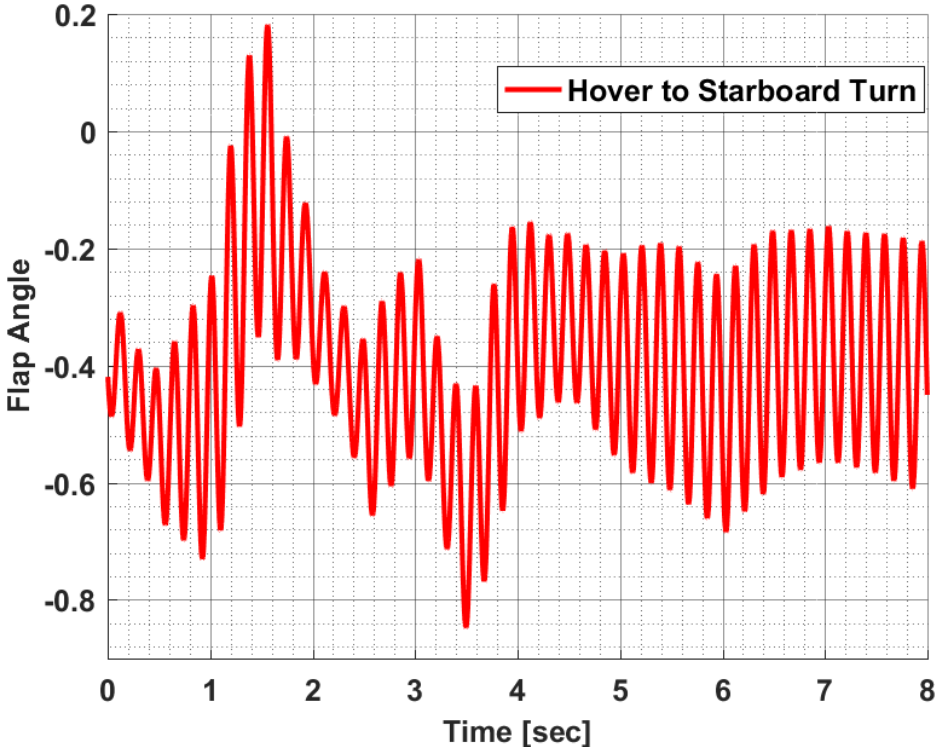
The maneuver aims to achieve the turn rate value of 60 [°/s] at 1.5th second and then continue the maneuver at this value. Since the right turn rate is defined as a positive sign, the constraint formula has been updated accordingly. Hence, the mathematical formula of the constraint updated as

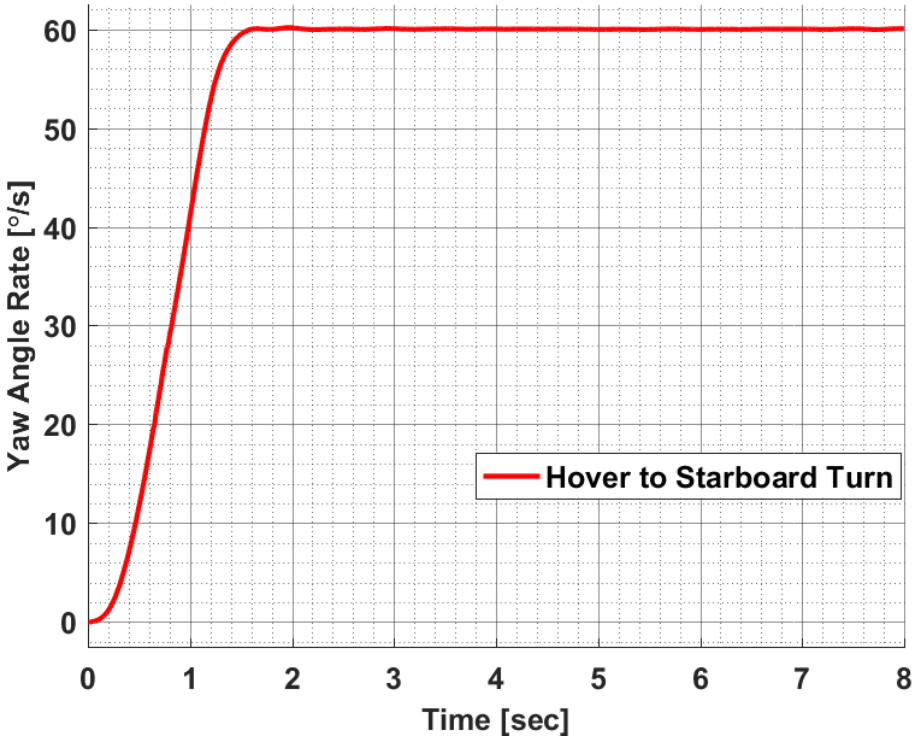
$$g_2 = \int_{t_s=1.5}^{t_f=8} |\dot{\Psi} - 60|^2 \, dt$$

where the yaw angle rate of the helicopter is represented by $\dot{\Psi}$**.** Also, the upper and lower time boundaries where the targeted parameter is desired to be obtained are represented with $\boldsymbol{t_s}$ and $\boldsymbol{t_f}$ respectively.

8.6.4 Optimization Solution of Hover to Starboard Turn Maneuver

This maneuver optimization is performed with the best configuration of the symmetric rank-one method, which is defined in Section 6.4. Then, the constraint results of the maneuver for this maneuvering optimization are given in Figure 8.14.


Hover to Starboard Turn
Flap Angle
Time [sec]
0.2
0
-0.2
-0.4
-0.6
-0.8
0
1
2
3
4
5
6
7
8


Hover to Starboard Turn
Yaw Angle Rate [°/s]
Time [sec]
60
50
40
30
20
10
0
0
1
2
3
4
5
6
7
8

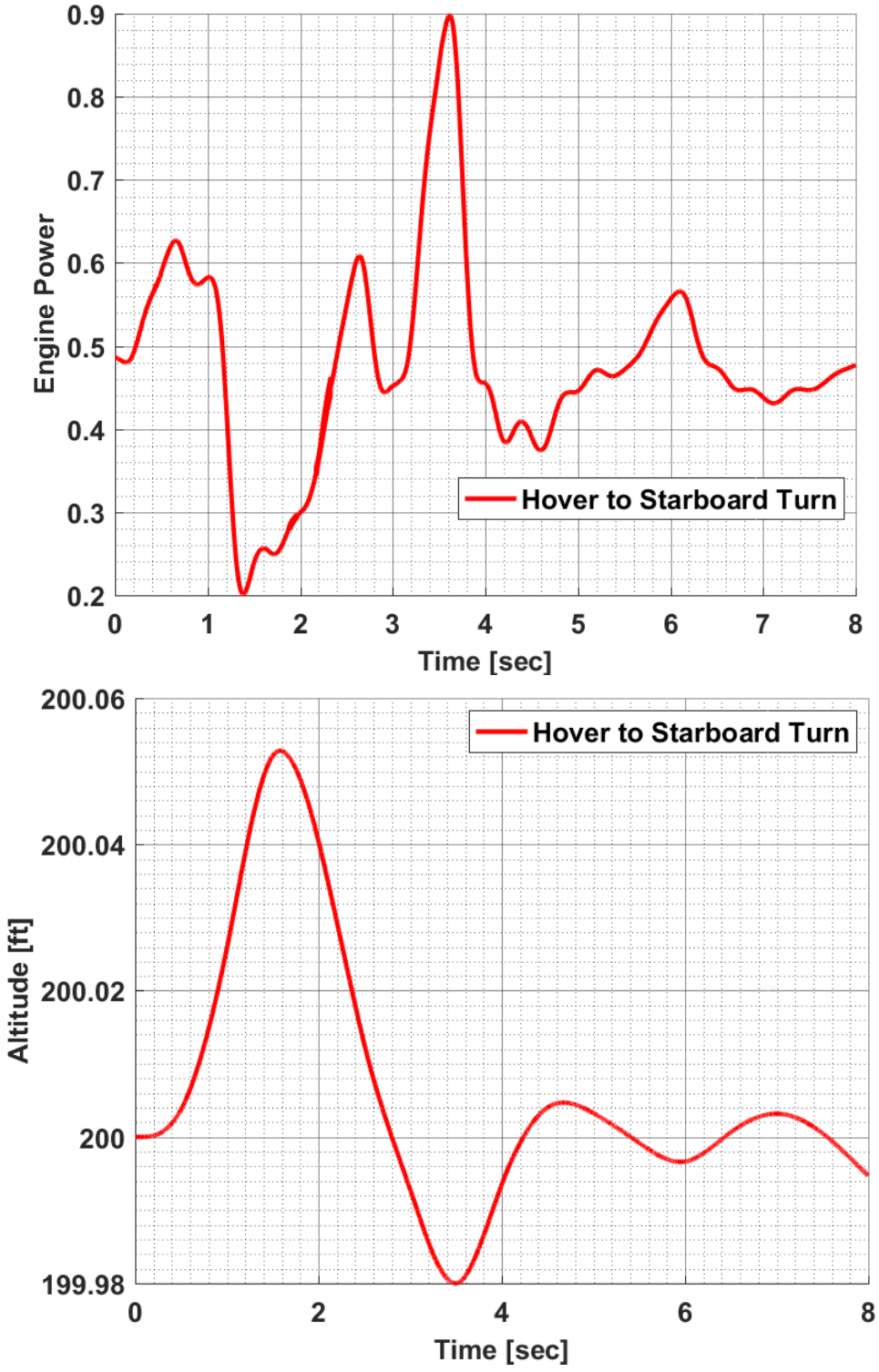
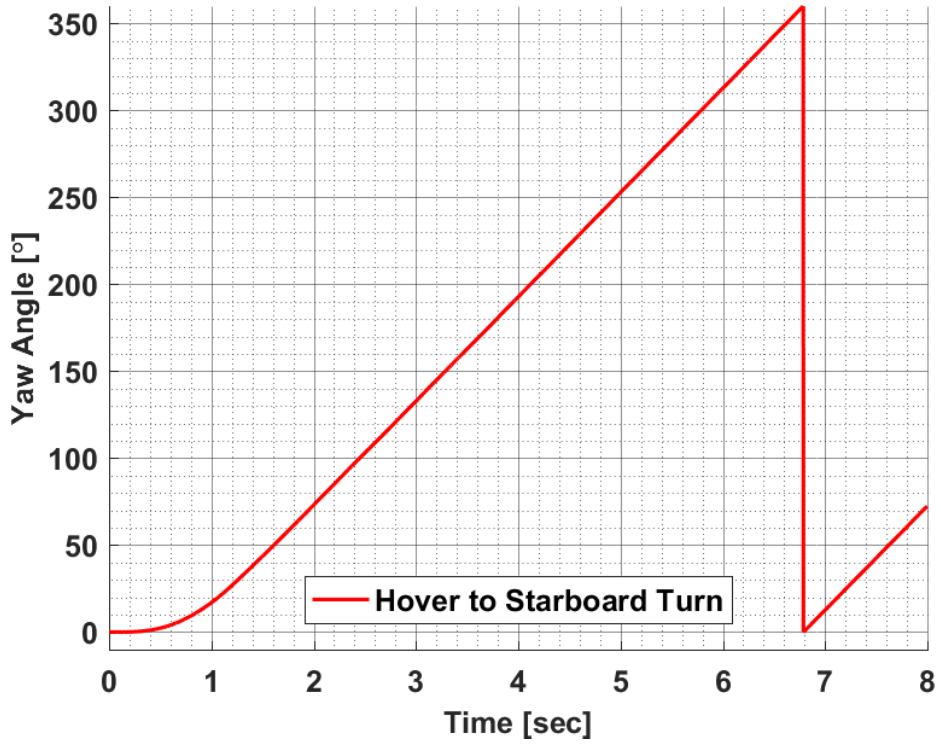


Figure 8.14: Constraint Results for Hover to Starboard Turn Maneuver with SR1 Method

According to Figure 8.13, it can be said that all the constraints of this maneuver optimization are met. In other words, the maneuvering requirements have been fulfilled while the flap angle and engine power plots are within defined design limits. In addition, the chart of the yaw angle rate shows that the helicopter maneuvered at the targeted turn rate from the 1.5th second as defined. Furthermore, the altitude chart says that the helicopter does not move on the vertical axis. Still, the velocity changes in three directions is given in Figure 8.15. This figure also includes the yaw angle chart to show the rotation of the helicopter on its vertical axis.



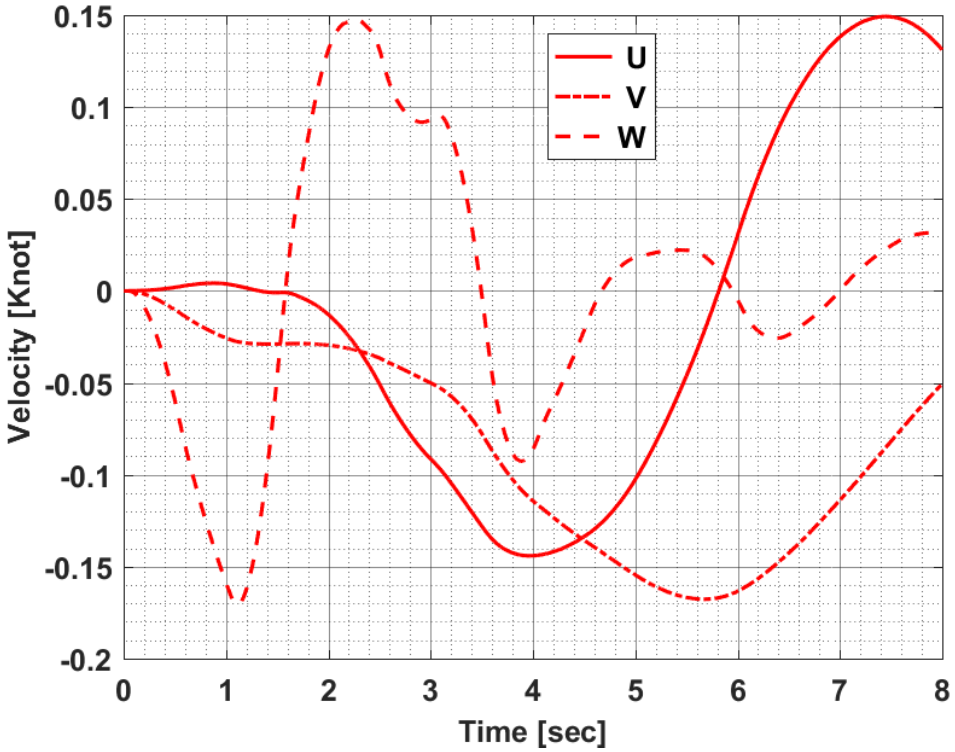


Figure 8.15: Maneuver Parameters for Hover to Starboard Turn Maneuver with SR1 Method

The yaw angle occurs as a natural result of this maneuver. It is seen that the chart of the yaw angle increases with a constant slope from zero to 360° up to about 6.8th second. This positive constant slope means that the helicopter is turning right at a constant turn rate. When the yaw angle reaches the value of 360°, the graph returns to zero since the helicopter takes the starting point as a reference direction. After 6.8th second, the graph continues to increase with the same slope because the helicopter maintains to rotate at the same turn rate. In addition, the velocity chart shows the velocity change in three directions. It can be deduced that the helicopter maintains its current position since the variation of each velocity value in this graph is quite small.

This maneuver optimization uses the anti-torque pedal and collective cyclic are defined as design variables. Therefore, the required design variable inputs are given in Figure 8.16 for hover to starboard turn optimization whose results are given in Figure 8.15.

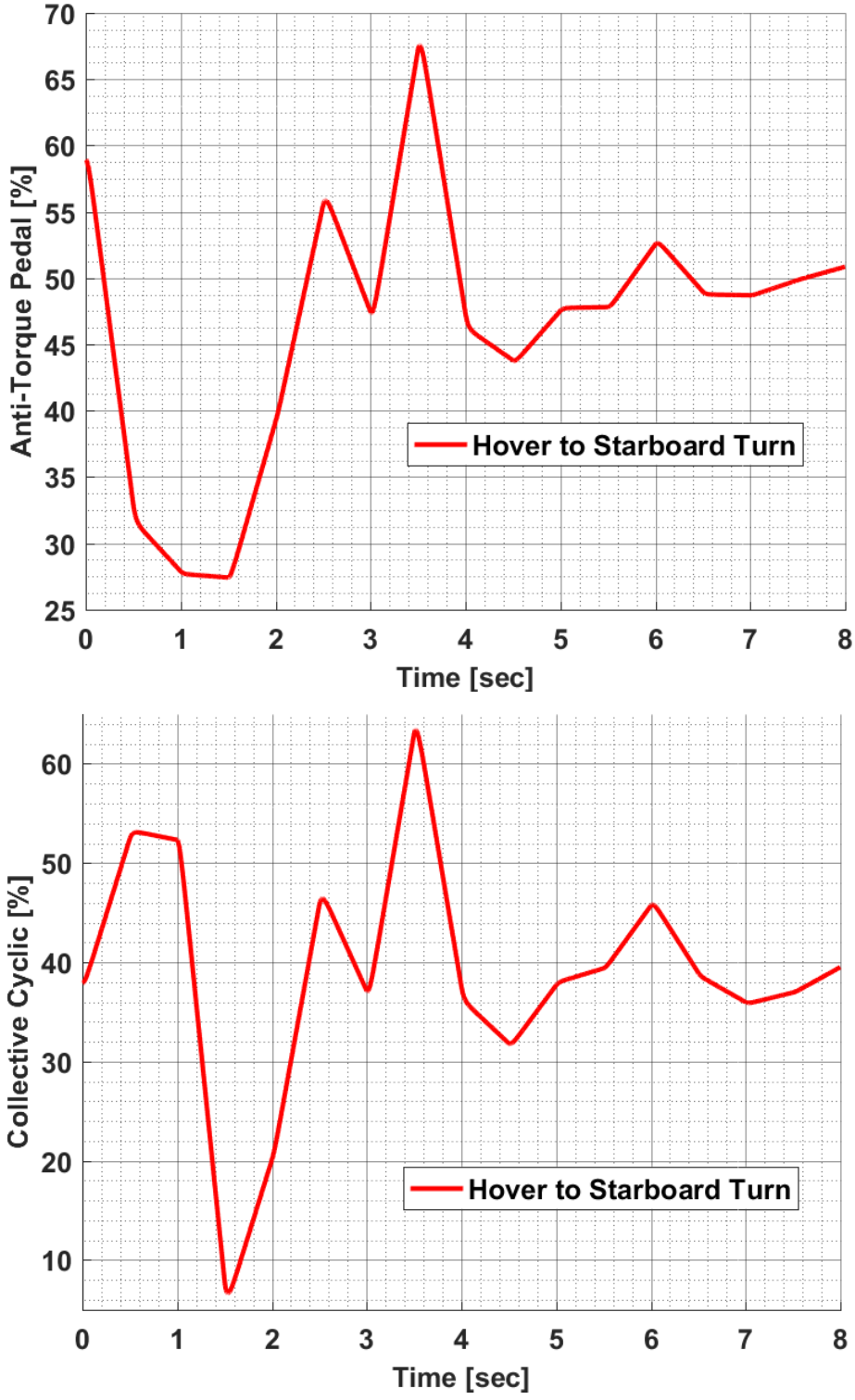


Figure 8.16: Obtained Design Variable for Hover to Starboard Turn Maneuver with SR1 Method

As it is mentioned in Section 5.2 that the anti-torque pedal input must be decreased in percent to turn the helicopter to the right. Moreover, Figure 8.16 says that the anti-torque pedal inputs are below the trim value, except for a single time step. Shortly, the helicopter has been rotated in the starboard direction by applying the design variable inputs shown in Figure 8.16.

8.7 Conclusion

In this chapter, different helicopter maneuvers were performed by using the symmetric rank-one optimization method, whose perturbation constant value is 0.1 and step size value is 5. The maneuver results prove that this method configuration is useful for helicopter maneuvering optimization.

CHAPTER 9

CONCLUSION

9.1 General Conclusions

In order to provide flight certification, aviation safety agencies must ensure that the designed helicopter can safely perform all defined maneuvers for each appropriate combination of weight and center of gravity. For this reason, the designers have to prove that the helicopter can fly in the entire flight spectrum without any breakage. Therefore, load engineers perform the maneuvers defined in the helicopter usage spectrum and calculate the internal and external loads caused by the maneuvers. Since performing these maneuvers with a trial and error approach causes both time loss and some defects in the maneuvers, this thesis study aims to perform the maneuvers using the optimization method. Thus, higher quality maneuvers can be achieved with less engineering time and less cost.

In the first part of the thesis, the optimization methods used in the literature have been researched and their applicability for maneuver optimization has been evaluated. In the literature, it is seen that the most used optimization methods are stochastic and gradient-based optimization methods. Unlike the gradient-based optimization methods, the stochastic methods are more likely to achieve a global minimum solution because it scans the entire possible solution domain. However, this comprehensive search means more time is spent getting the optimum solution. The main purpose of this thesis is not to reach the global minimum value, but to realize the helicopter maneuver in the shortest time and in the most efficient way. Therefore, it has been decided to use gradient-based optimization methods for helicopter maneuvering optimization. In accordance with this purpose, Broyden's Method, Symmetric Rank-One (SR1) method, Davidon-Fletcher-Powell (DFP)

method and Broyden-Fletcher-Goldfarb-Shanno (BFGS) method are determined to be examined as the gradient-based optimization methods.

In the second part of the thesis, the helicopter mathematical model and the optimization methodology for helicopter maneuvering are described in detail. Moreover, after explaining how the optimization parameters are defined and configured, the developed optimization code is explained step by step in detail.

In the third part of the thesis, the different configurations of the defined optimization methods were applied for two different maneuvers chosen as examples. Thus, the most useful configurations of the optimization method are determined among these configurations. During this comparison process, hover to forward flight and hover to sideward flight maneuvers were used as example maneuvers. The main purpose of performing the same operations in two different maneuvers is to make sure that the chosen configuration is the best. Moreover, the optimization parameters are defined for each maneuver. Then, maneuver optimization is executed to obtain the most useful configuration in terms of the gradient calculation method, perturbation constant and step size values. Finally, the optimization result of each configuration is compared to each other. As a result, it is concluded that one of the symmetric rank-one method configurations is more useful among the others. This most useful configuration uses the central difference approximation as the gradient computation method. Moreover, it is obtained at the values of 0.1 perturbation constant and 5 step size. Shortly, it is decided that the symmetric rank-one method, whose gradient computation method is the central difference approximation, perturbation constant value is 0.1 and step size value is 5, is the most useful method for helicopter maneuvering optimization.

In the fourth part of the thesis, the main modifications are made while developing optimization code and their effects on optimization are examined. This examination shows that the last updated version of the code is more effective than other versions. In addition, the importance of time step selection was tested by analyzing different time step values. According to these analyses, as the value of the time step decreases,

optimization converges to a smaller objective value. However, if the value of the time step is too small, this can cause the optimization algorithm to perform many FLIGHTLAB analyses at the same time due to the parallel run process. Therefore, optimization time may increase if the computer CPU slows down due to these parallel studies. Consequently, the time step has a direct effect on both optimization time and optimization accuracy. Therefore, its value should be chosen depending on which one is more important. In addition to time step selection, the different initial conditions are also compared in this part to decide the most logical choice. The result of these comparisons shows that the values of pilot inputs at the trim point are the best choice. In addition, since the helicopter starts the maneuver from the trim point, it is both a realistic and logical approach to take the trim values as a reference. Lastly, perturbation selection on finite divided difference approximations has been investigated. This investigation shows that the accuracy of the central difference approximation with perturbation constant value 0.1 is equal to the accuracy of both forward and backward difference approximations with perturbation constant values of 0.01. Moreover, the forward and backward difference approximations have been achieved the solution earlier. In addition, it should be noted that they cover only the hover phase of the helicopter maneuvers.

In the last part of the thesis, the types of maneuvers performed by helicopters very frequently throughout their lifetime have been solved as an example. In addition to these maneuvers, pull up and pushover maneuvers are solved in limited flight conditions. These exemplary maneuver optimizations show that the selected symmetric rank-one configuration is suitable and useful for helicopter maneuvers.

In summary, this thesis has been studied on some gradient-based optimization methods in different configurations to perform helicopter maneuvers. At the end of a whole study, the symmetric rank-one method with the best configuration has been decided for helicopter maneuver optimization. Finally, solving desired helicopter maneuvers with this optimization method not only reduces engineering time and cost but also provides a higher quality maneuver.

9.2 Recommendation for Future Studies

In this thesis study, the values of step size, perturbation constant and penalty parameters are assigned as a fixed value. This study covers only the hover phase of the helicopter flight. All the analysis have been performed, and all the computations have been calculated for that phase. Should any other phase is intended to analyze, the required analyses should be developed and conducted for that phase. Moreover, it is clear that they have effects on optimization efficiency. Therefore, these parameter values can be guided more intelligently with an algorithm during optimization. In addition, if the pilot control inputs are stored for each maneuver optimization, optimization speed can be increased by giving these values as the initial condition in similar maneuver optimizations. Finally, the accuracy of the FLIGHTLAB software used as a helicopter simulation program can be achieved by gradually changing the value of the perturbation constant. Thus, the perturbation constant value can be defined by considering this accuracy. The necessary accuracy should be determined prior to the computations.

REFERENCES

[1] European Aviation Safety Agency (EASA), “Certification Specification and Acceptable Means of Compliance for Small Rotorcraft CS 27”. Ed 6, December 17, 2018.

[2] European Aviation Safety Agency (EASA), “Certification Specification and Acceptable Means of Compliance for Large Rotorcraft CS 29”. Ed 7, July 15, 2016.

[3] Aydoğdu, İ., “Optimum Design of 3-D Irregular Steel Frames Using Ant Colony Optimization And Harmony Search Algorithms”, Middle East Technical University, 2010.

[4] Bilen M. E., “Development Of A Hybrıd Global Optımızatıon Algorıthm And Its Applıcatıon To Helıcopter Rotor Structural Optımızatıon”, Middle East Technical University, May 2019.

[5] Zhu J., Trélat, E., & Cerf, M., “Geometric Optimal Control and Application to Aerospace”, *Pasific Journal of Mathematics for Industry*, Vol. 9, No. 1, January 22, 2017.

[6] Gerguad, J., & Haberkom, T., “Homotopy Method for Minimum Consumption Orbit Transfer Problem”, *ESAIM: Control, Optimization and Calculus of Variations*, Vol. 12, No. 2, pp. 294-310, April 2006.

[7] Imani, K. & Novinzadeh, A. B., & Habibi, M., “FDI and Adaptive Threshold Detection of Pilot by Application Development of Available Models Using Bond Graph Method”, *Journal of Fundamental and Applied Science*, Vol. 10, No. 2S, pp. 245, February 2018.

[8] Thomson, D., & Bradley, R., “Inverse simulation as a tool for flight dynamics research-Principle and applications,” *Progress in Aerospace Sciences*, Vol. 42, No.3, pp. 174-210, May 2006.

[9] Kato, O., & Sugiura, I., “An Interpretation of Airplane General Motion and Control as Inverse Problem”, *Journal of Guidance, Control, and Dynamics*, Vol. 9, No. 2, pp. 198–204, 1985.

[10] Lane, S. H., & Stengel, R., “Flight Control Design Using Non-linear Inverse Dynamics”, *Automatica*, Vol. 24, No. 4, pp.471-48, 1988.

[11] Jones, R. T., “A Simplified Application of the Method of Operators to the Calculation of Disturbed Motions of an Airplane”, *Annual Report-National Advisory Committee for Aeronautics*, Vol. 560, 1936.

[12] Thomson, D. G., & Bradley, R., “Development and Verification of an Algorithm for Helicopter Inverse Simulations”, *Vertica*, Vol. 14, No. 2, pp. 185-200, 1990.

[13] Hess, R. A., & Gao, C., “A Generalized Algorithm for Inverse Simulation Applied to Helicopter Maneuvering Flight”, *Journal of the American Helicopter Society*, Vol. 38, No. 4, pp. 3–15, October 1993.

[14] Prasad, R., “Development and Validation of Inverse Flight Dynamics Simulation for Helicopter Maneuver”, Presented at the 4th Asian/Australian Rotorcraft Forum. India, Institute of Science, November 16-18, 2015.

[15] Celi, R., “Optimization-Based Inverse Simulation of a Helicopter Slalom Maneuver”, *Journal of Guidance, Control, and Dynamics*, Vol. 23, No.2, March-April 2000.

[16] Guglieri, G. & Mariano, V., “Optimal Inverse Simulation of Helicopter Maneuvers”, In *Communications to SIMAI Congress*, Vol. 3, pp. 261-272, August 2009.

[17] Carroll, D. L., “Genetic Algorithms and Optimizing Chemical Oxygen-Iodine Lasers”, *Development in Theoretical and Applied Mechanics*, Vol.18, No. 3, pp. 411-424, 1996.

[18] Haupt, R., "Comparison between genetic and gradient-based optimization algorithms for solving electromagnetics problems", *IEEE Transactions on Magnetics*, Vol. 31, No. 3, pp. 1932-1935, May 1995.

[19] Johnson, W., “A History of Rotorcraft Comprehensive Analysis”, Ames Research Center, Moffett Field, California, 2013.

[20] Du Val, R. W., “A Real-Time Blade Element Helicopter Simulation For Handling Qualities Analysis”, Presented at 15th European Rotorcraft Forum, Amsterdam, The Netherlands, September 12-15, 1989.

[21] Du Val, R. W., “A Low Cost.—High Fidelity Real-Time Rotorcraft Simulation for Control and Handling Qualities Analysis”, In *Heli Japan*

1998: AHS International Meeting on Rotorcraft Technology and Disaster Relief, Gifu, Japan, April 1998.

[22] Du Val, R. W., "A Real-Time Multi-Body Dynamics Architecture for Rotorcraft Simulation", In *The Challenge of Realistic Rotorcraft Simulation: Proceeding*, London, UK, November 7-8, 2001.

[23] Howlett, J. J., "UH-60A Black Hawk Engineering Simulation Program", NASA CR 166309, CR 166310, December 1981.

[24] Saberi, H.-A., Jung, Y. C. & Anastassiades, T., "Finite Element and Modal Method in Multibody Dynamic Code", In *Proceeding of 2nd International Aeromechanics Specialists' Conference*, Bridgeport, CT, October 1995.

[25] Kalkan, U., & Tosun, F., "Optimization Based Inverse Simulation Method for Helicopter Pull Up Maneuver", Presented at 45th European Rotorcraft Forum, Warsaw, Poland, September 17-20, 2019.

[26] Amari, S., "Synthesis of Cross-Coupled Resonator Filters Using an Analytical Gradient-Based Optimization Technique", *IEEE Transactions on Microwave Theory and Techniques*, Vol. 48, Vo. 9, pp. 1559-1564, September 2000.

[27] Conn, A. R, Elfadel, I. M, Molzen, Jr, W. W, O'Brien, P. R, Strenski, P. N., Visweswariah, C. & Whan, C. B., "Gradient-Based Optimization of Custom Circuits Using a Static-Timing Formulation", In *Proceeding of the 36th annual ACM/IEEE Design Automation Conference*, pp. 452-459, June 1999.

[28] Çelik, U. & Yurtay, N., "An Ant Colony Optimization Algorithm-Base Classification for the Diagnosis of Primary Headaches Using a Website Questionnaire Expert System", *Turkish Journal of Electrical Engineering & Computer Sciences*, Vol. 25, No. 5, pp. 4200-4210, 2017.

[29] Alobaedy, M. M. & Ku-Mahamud, K. R., "Scheduling Jobs in Computational Grid using Hybrid ACS and GA Approach", In *2014 IEEE Computer, Communications and IT Applications Conference*, pp. 223-228, 2014.

[30] Tosun, F., Kalkan, U., Tıraş, H. & Yaman, Y., "Generation of Helicopter Pull-Up Maneuver with Harmony Search Optimization Method", Presented at 8th Asian/Australian Rotorcraft Forum, Ankara, Turkey, October 30–November 2, 2019.

[31] Bezat, A., "Comparison of the Deterministic and Stochastic Approaches for Estimating Technical Efficiency on the Example of Non-parametric DEA and SFA Methods", *Metodyilościowe w badaniachekonomicznych*, Vol. 10, No. 1, pp. 20-29, 2009.

[32] Francisco, M., Revollar S., Lamanna R. & Vega, P.A., *"Comparative Study of Deterministic and Stochastic Optimization Method for Integrated Design of Processes."* Proceedings 16th IFAC World Congress, Prague, Czech Republic, July 3-8 2005.

[33] Wetter, M. and Wright, J., "A Comparison of Deterministic and Probabilistic Optimization Algorithms for Nonsmooth Simulation-Based Optimization", *Building and Environment*, Vol. 39, No. 8, pp. 989-999, 2004.

[34] Yang, X.-S., *"Introduction to Algorithm for Data Mining and Machine Learning",* Academic Press, pp. 45-65, 2019.

[35] Yang, X.-S., *"Engineering Mathematic with Examples and Applications",* Academic Press, pp. 267-283, 2017.

[36] Cauchy, M. A., "Methodes generales pour la resolution des syst'emes dequations simultanees", Comptes Rendus de Academie des Sciences, Vol. 25, No. 2, pp. 536-539, 1847.

[37] Meza, J. C., "Steepest Descent", *Wiley Interdisciplinary Reviews: Computational Statistics Journal*, Vol. 2, No. 6, pp. 719-722, November 2010.

[38] Luenberger, D. G. & Ye, Y., *"Linear and Nonlinear Programing"*, Springer International Publishing Switzerland, 4th ed., 2016.

[39] Nash, S. G. & Sofer, A., *"Linear and Nonlinear Programing."* International Series in Operations Research & Management Science. Springer US, New York, 2008.

[40] Amini, K. & Rostami, F., "Three-steps modified Levenber-Marquardt method with a new line search for systems of nonlinear equations", *Journal of Computational and Applied Mathematics*, Vol. 300, pp. 30-42, July 2015.

[41] Broyden, C. G., Dennis, Jr, J. E. & Moré J. J, "On the Local and Superlinear Convergence of Quasi-Newton Methods", *IMA Journal of Applied Mathematics*, Vol. 12, No. 3, pp. 223-245, December 1, 1973.

[42] Sayevand, K. & Jafari, H., "On systems of nonlinear equations: some modified iteration formulas by the homotopy perturbation method with accelerated fourth- and fifth- order convergence", *Applied Mathematical Modelling*, Vol. 40, No. 2, pp. 1467-1476, 2016.

[43] Dennis, Jr, J. E., *"On Some Methods Based on Broyden's Secant Approximation to the Hessian, in Numerical Methods for Nonlinear Optimization",* edited by F. A. Lootsma, Academic Press, New York, pp. 19-34, 1972.

[44] Berenguer, L. & Tromeur-Dervout, D., "Development on the Broyden procedure to solve nonlinear problems arising in CFD", *Computers & Fluids*, Vol. 88, pp. 891-896, December 2013.

[45] Pfrommer, B. G., Côté, M., Louie, S. G. & Cohen, M. L., "Relaxation of Crystals with the Quasi-Newton Method", *Journal of Computational Physics*, Vol. 131, No. 1, pp. 233-240, February 1997.

[46] Broyden, C. G, "A Class of Methods for Solving Nonlinear Simultaneous Equations" *Mathematics of Computation*, Vol. 19, No. 92, pp. 577-593, October 1965.

[47] Al-Towaiq, M. & Hour, Y. A., "Two improved classes of Broyden's methods for solving nonlinear systems of equations", *Journal of Mathematics and Computer Science*, Vol. 17, No. 1, pp. 22-31, 2017.

[48] Modarres, F., Hassan, M. A. & Leong, W. J., "Structured symmetric rank-one method for unconstrained optimization", *International Journal of Computer Mathematics*, Vol. 88, No. 12, pp. 2608-2617, August 2011.

[49] Ip, C. M. & Todd, M. J., "Optimal Conditioning and Convergence in Rank One Quasi-Newton Updates", *SIAM Journal on Numerical Analysis*, Vol. 25, No. 1, pp. 206-221, February 1988.

[50] Khiyabani, F. M. & Leong, W. J., "On the performance of a new symmetric rank-one method with restart for solving unconstrained optimization

problems", *Computers and Mathematics with Applications*, Vol. 64, No. 6, pp. 2141-2152, September 2012.

[51] Modarres, F., Hassan, M. A. & Leong, W. J., "A symmetric rank-one method based on extra updating techniques for unconstrained optimization", *Computers and Mathematics with Applications*, Vol. 62, No. 1, pp. 392-400, July 2011.

[52] Davidon, W. C., "Variable Metric Method for Minimization", A.E.C. Research and Development Report, ANL-5990 (Rev.), November 1959.

[53] Fletcher, R. & Powell, M. J. D., "A rapidly convergent descent method for minimization", The *Computer Journal*, Vol. 6, No.2, pp. 163-168, 1963.

[54] Federal Aviation Administration, *"Helicopter Flight Handbook:FAA-H-8083-21A"*, Washington, 2019.

[55] Rotaru, C. & Todorov, M., "Helicopter Flight Physics", *Flight Physic – Models, Techniques and Technologies*, Konstantin Volkov, December 20 2017.

[56] Jhonson W., *"Helicopter Theory."* Dover Publications, New York, NY, U.S., 1994.

[57] Yeniay, Ö., "Penalty function methods for constrained optimization with genetic algorithm", *Mathematics and Computational Applications*, Vol. 10, No. 1, pp. 45-56, 2005.

[58] Gül S., Bilen M.E., Şahin M., Gürak D. & Yaman Y., "A Convergence Study for a Longitudinal Maneuver by Using Various Tools", AHS 72nd Annual Forum, Florida, USA, 17-19 May 2016.

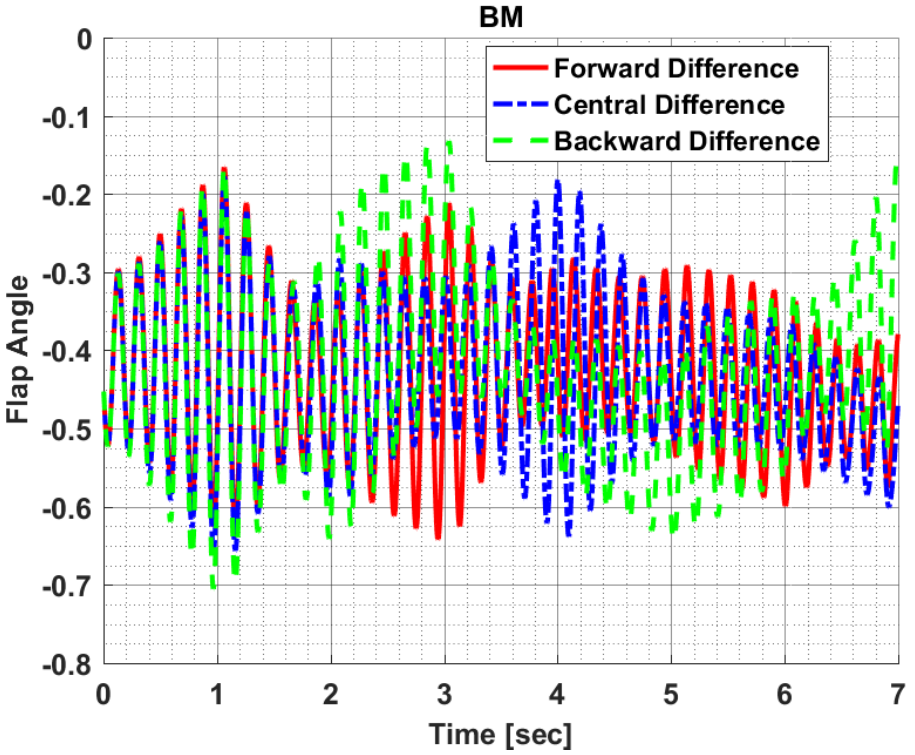
APPENDICES

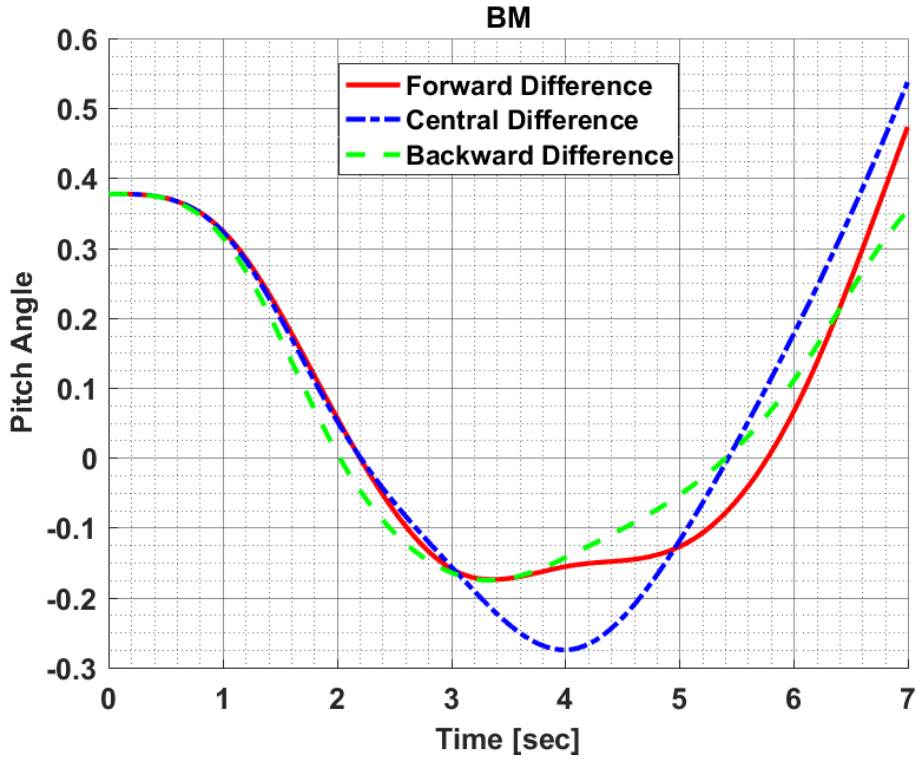
In the appendix section, all constraint results are plotted according to last solution for each configuration. Moreover, while the constraints of flap angle and pitch angle are scaled from -1 to 1, engine power constraint are scaled from 0 to 1. In other words, the minimum and maximum limits of flap and pitch angles are represented by the value of -1 and 1, respectively. Similarly, the value of 1 in engine power chart represents the maximum engine power value. Afterwards, the values between the parameter limits are scaled according to these intervals. However, there is no any scaling for the velocity and the altitude charts. The velocity chart includes its zoomed view at time interval where the constraint is defined.

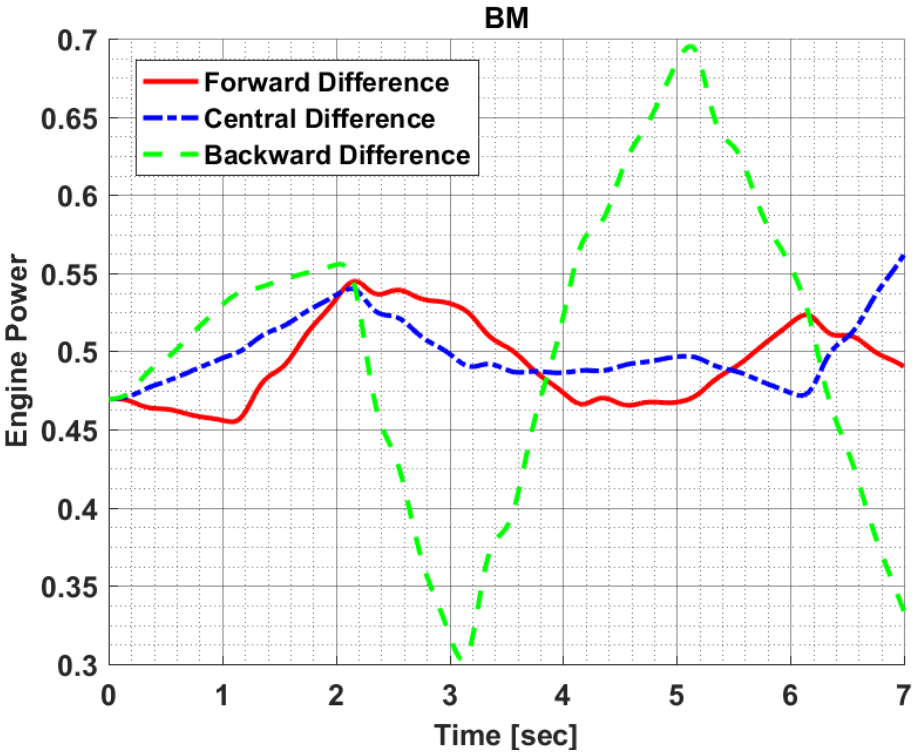
APPENDIX A1: Comparison - The Finite Divided Difference Approximation for Hover to Forward Flight Maneuver Optimization

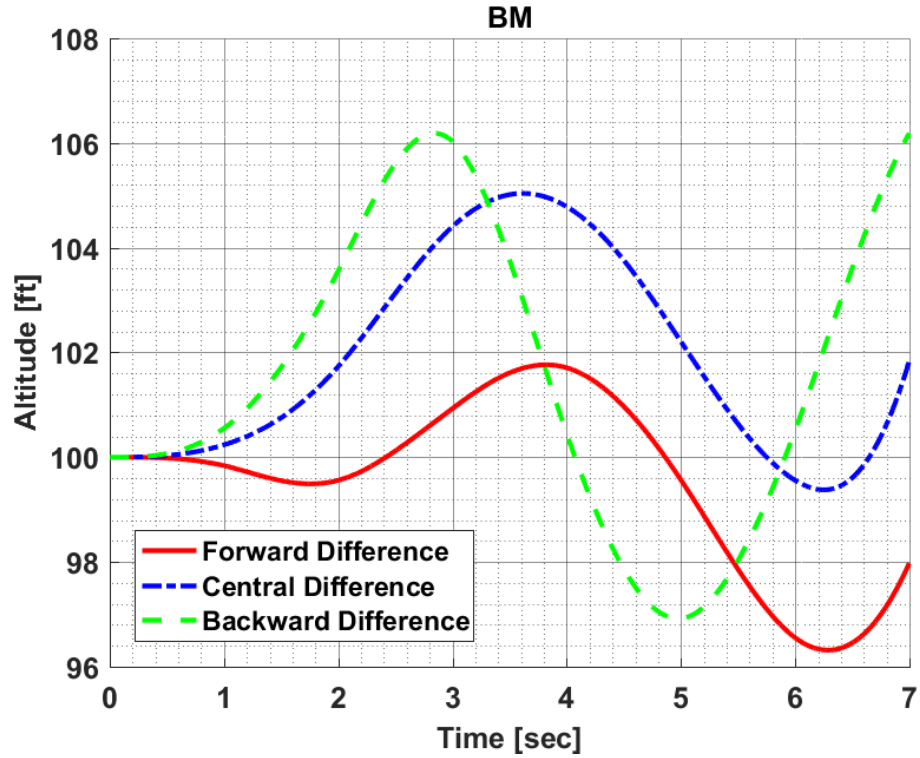
In this appendix section, constraint results and corresponding pilot control inputs are provided for each optimization method configuration which are executed for the comparison of the finite divided difference approximation. Moreover, they are the results of hover to forward flight maneuver optimization.

Results of BM:






BM
Forward Difference
Central Difference
Backward Difference
Engine Power
Time [sec]
0.7
0.65
0.6
0.55
0.5
0.45
0.4
0.35
0.3
0
1
2
3
4
5
6
7


BM
Forward Difference
Central Difference
Backward Difference
Altitude [ft]
Time [sec]
108
106
104
102
100
98
96
0
1
2
3
4
5
6
7

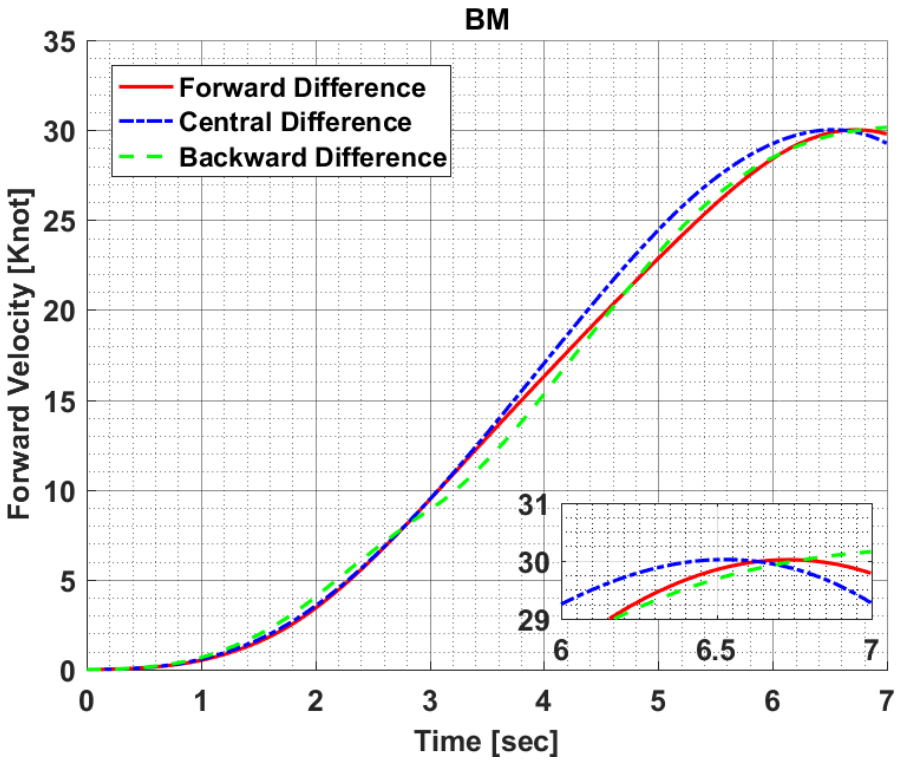


Figure 10.1: Constraint Results of Hover to Forward Flight Maneuver Optimization in the Finite Divided Difference Approximation Comparison for BM

Figure 10.1 shows the constraint results of hover to forward flight maneuver obtained in comparing the finite divided difference approximation for BM. Moreover, Figure 10.2 shows the required longitudinal and collective cyclic input to be able to carry out hover to forward flight maneuver as given in Figure 10.1.

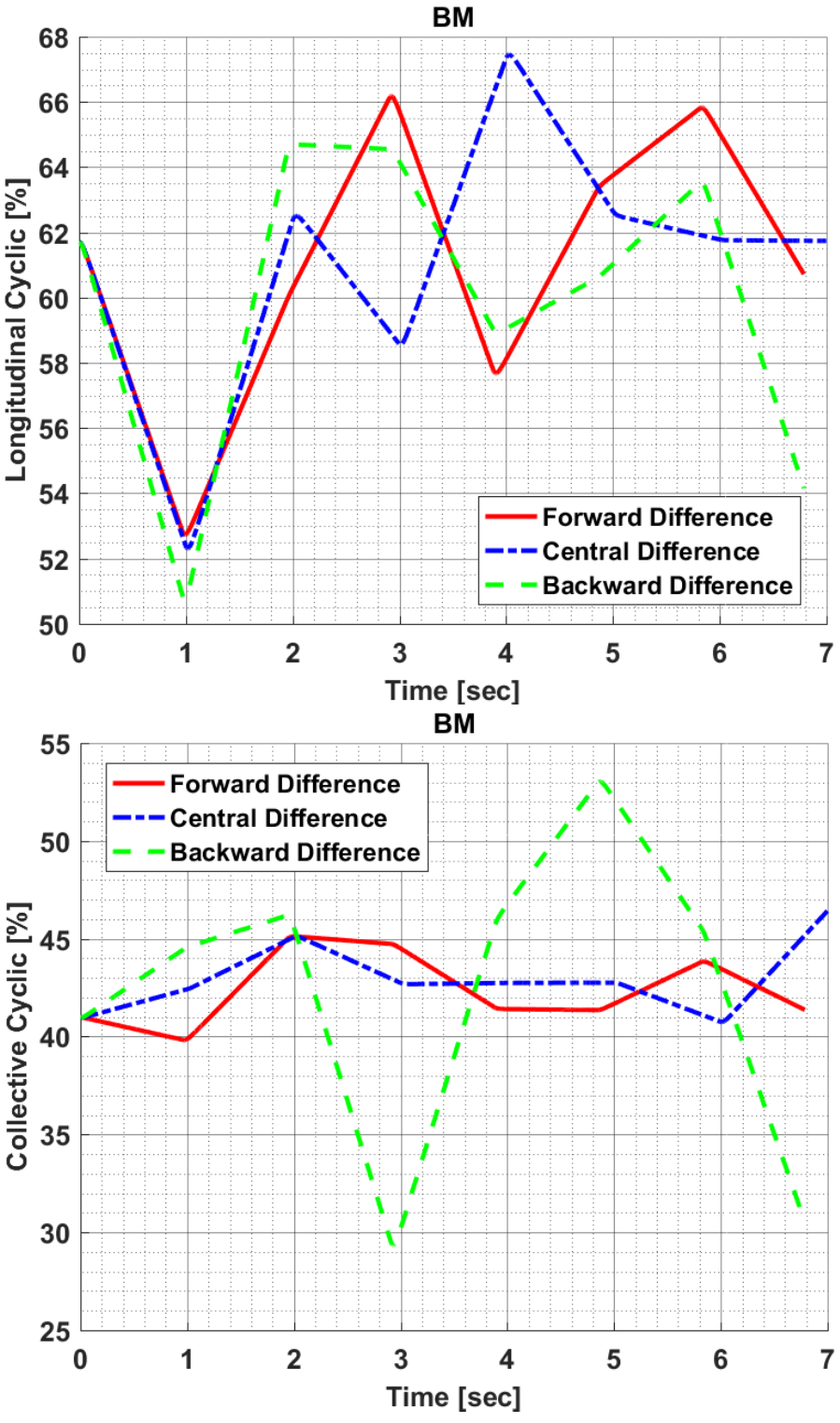
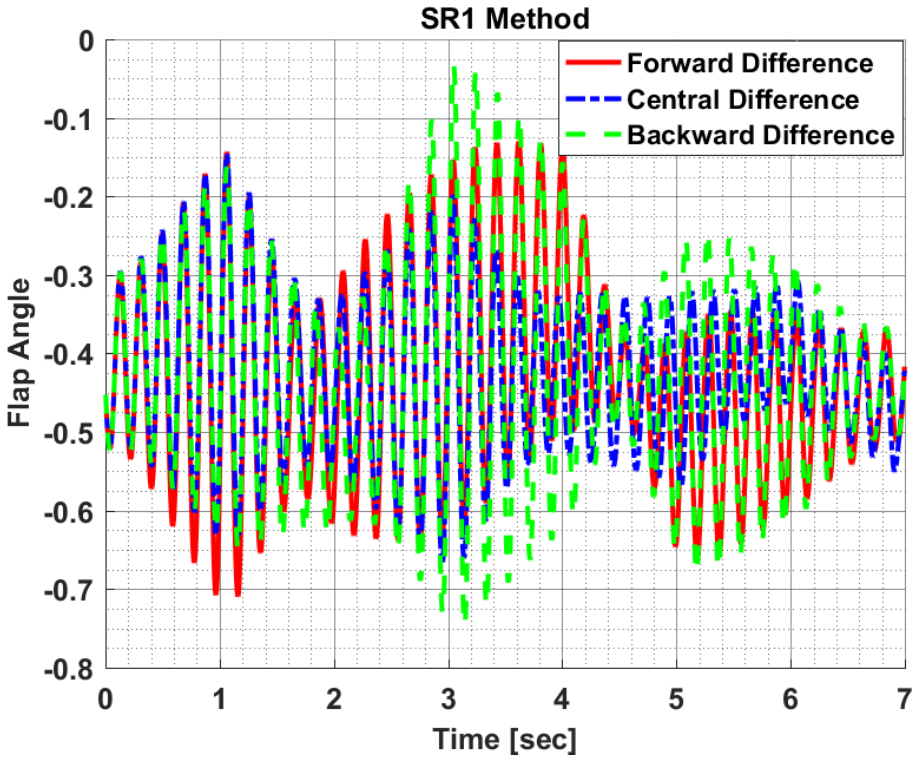
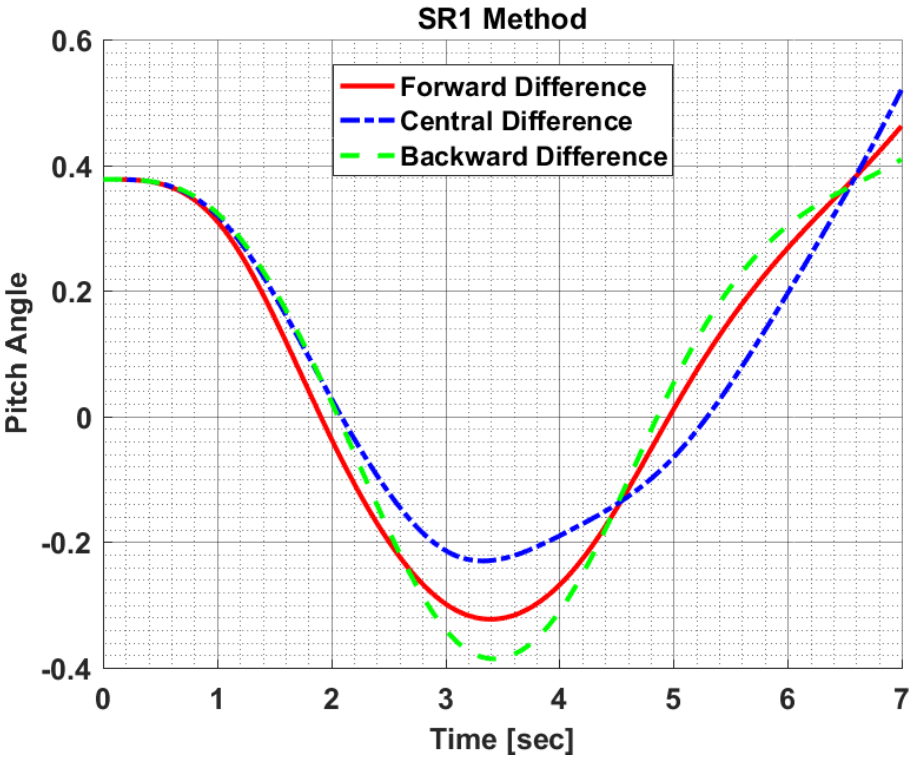
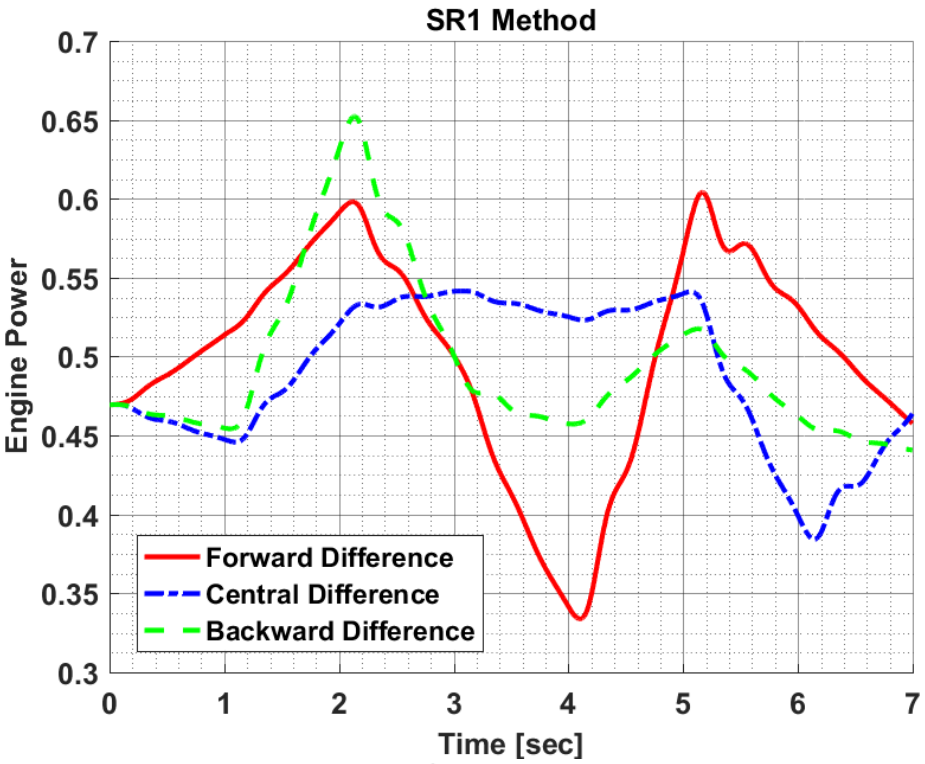


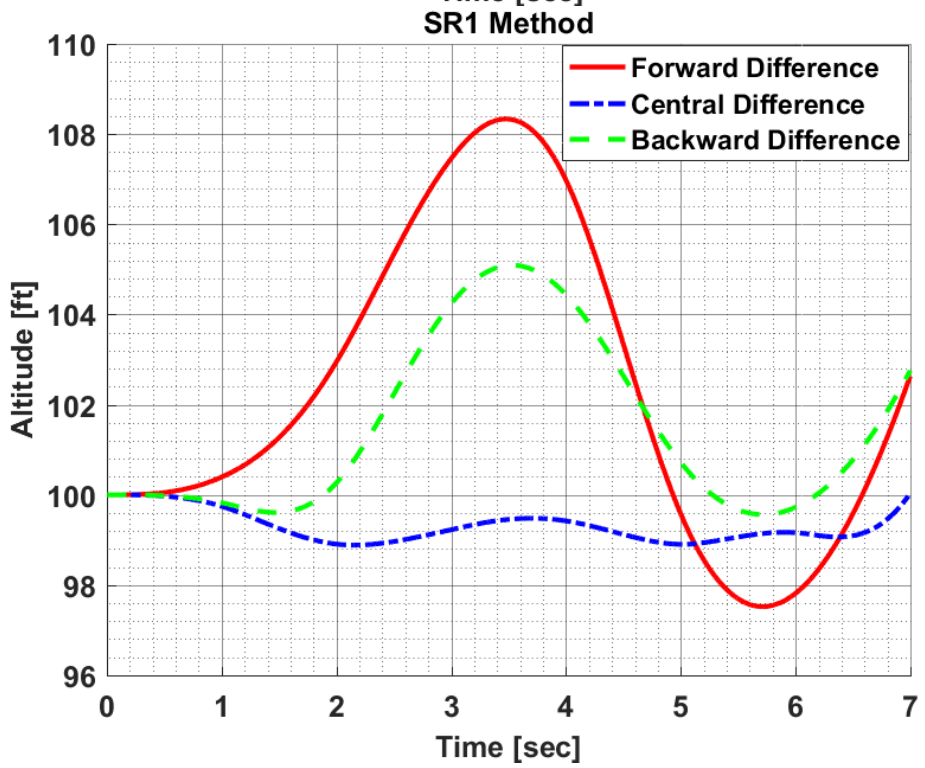
Figure 10.2: Design Variables of Hover to Forward Flight Maneuver Optimization in the Finite Divided Difference Approximation Comparison for BM

Results of SR1 Method:






SR1 Method
Engine Power
Time [sec]
Forward Difference
Central Difference
Backward Difference
0.7
0.65
0.6
0.55
0.5
0.45
0.4
0.35
0.3
0
1
2
3
4
5
6
7


SR1 Method
Altitude [ft]
Time [sec]
Forward Difference
Central Difference
Backward Difference
110
108
106
104
102
100
98
96
0
1
2
3
4
5
6
7

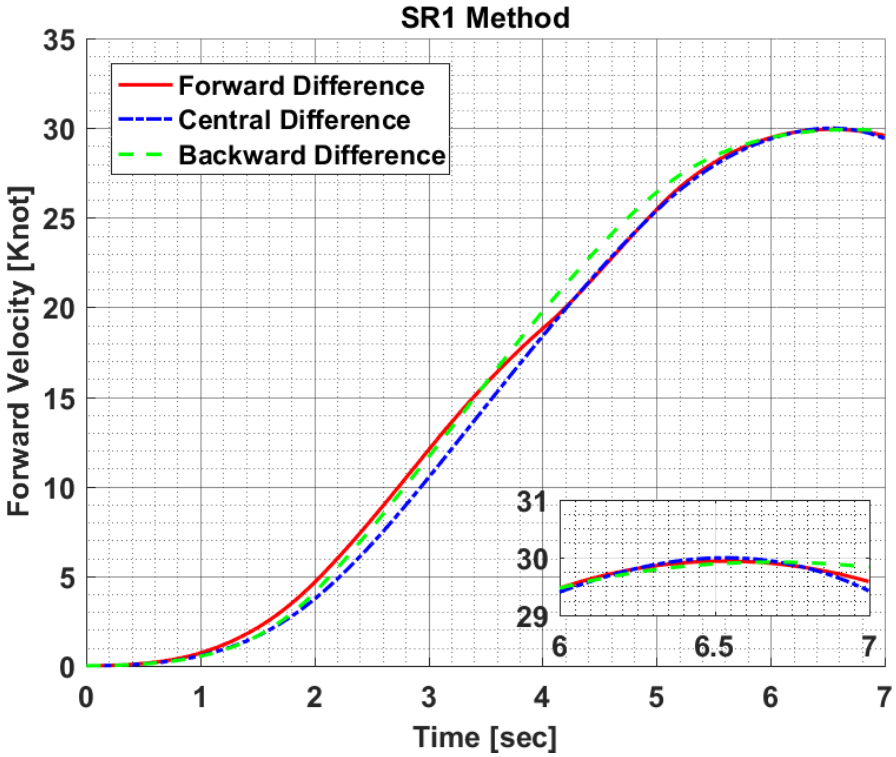


Figure 10.3: Constraint Results of Hover to Forward Flight Maneuver Optimization in the Finite Divided Difference Approximation Comparison for SR1 Method

Figure 10.3 shows the constraint results of hover to forward flight maneuver obtained in comparing the finite divided difference approximation for symmetric rank-one method. Moreover, Figure 10.4 shows the required longitudinal and collective cyclic input to be able to carry out hover to forward flight maneuver as given in Figure 10.3.

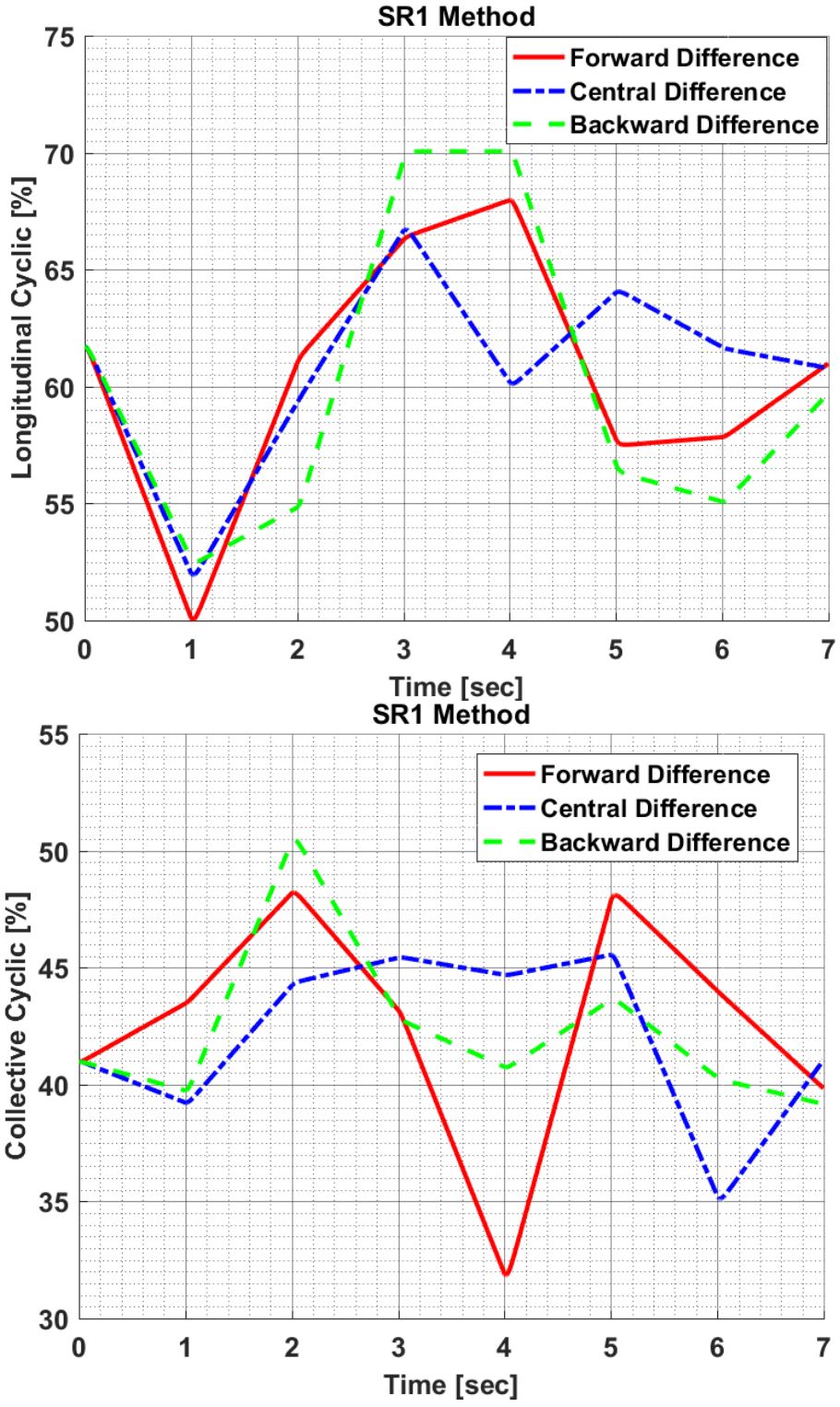
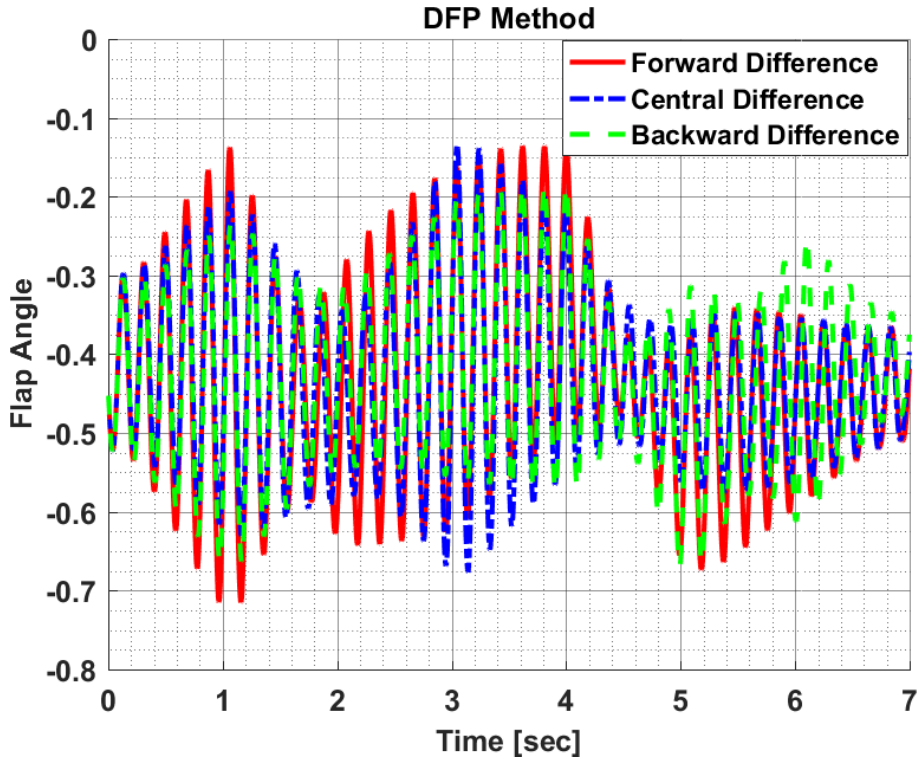
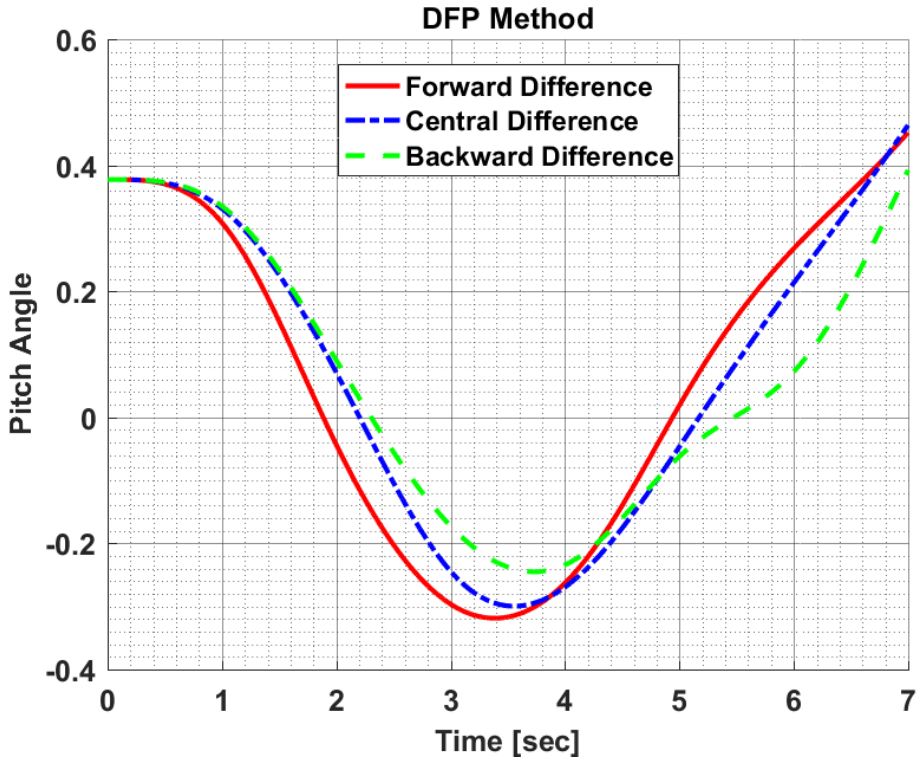
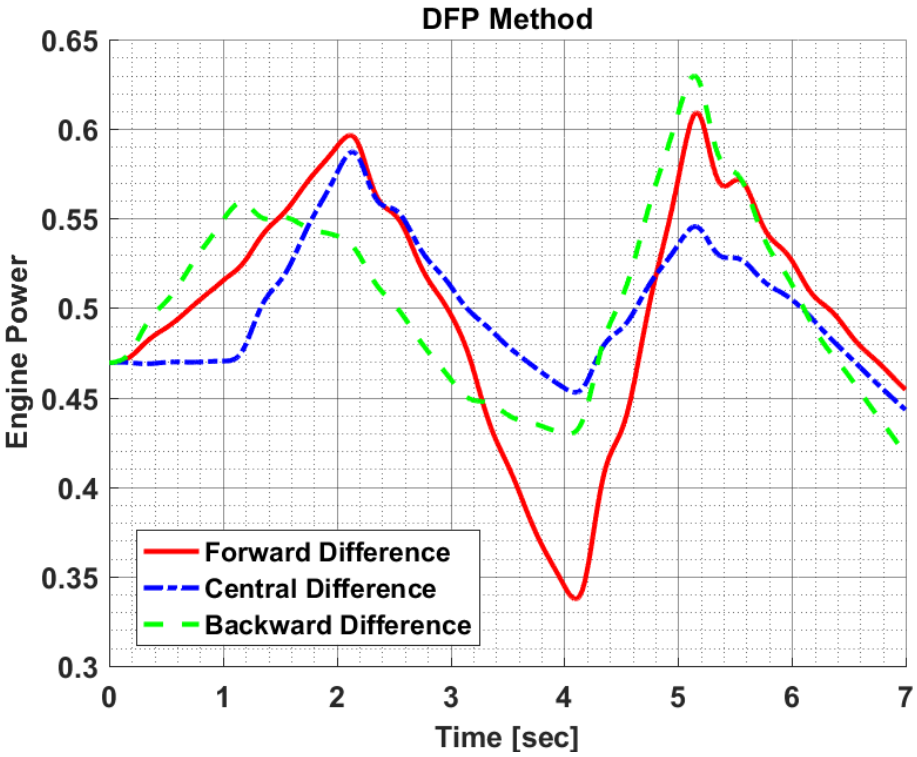


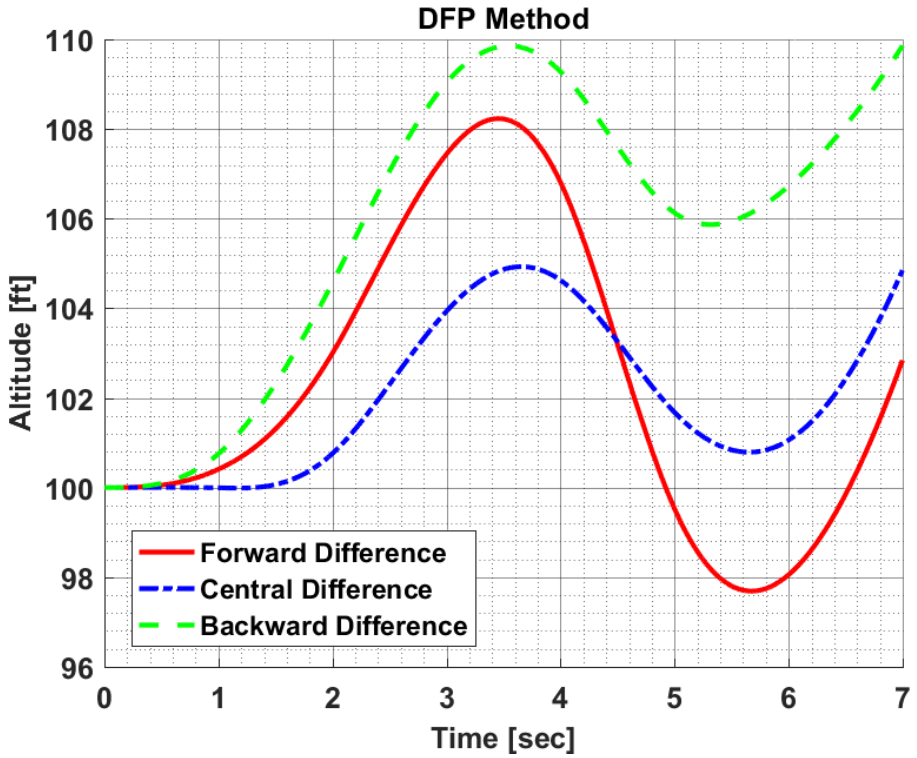
Figure 10.4: Design Variables of Hover to Forward Flight Maneuver Optimization in the Finite Divided Difference Approximation Comparison for SR1 Method

Results of DFP Method:






DFP Method
Engine Power
Time [sec]
Forward Difference
Central Difference
Backward Difference
0.65
0.6
0.55
0.5
0.45
0.4
0.35
0.3
0
1
2
3
4
5
6
7


DFP Method
Altitude [ft]
Time [sec]
Forward Difference
Central Difference
Backward Difference
110
108
106
104
102
100
98
96
0
1
2
3
4
5
6
7

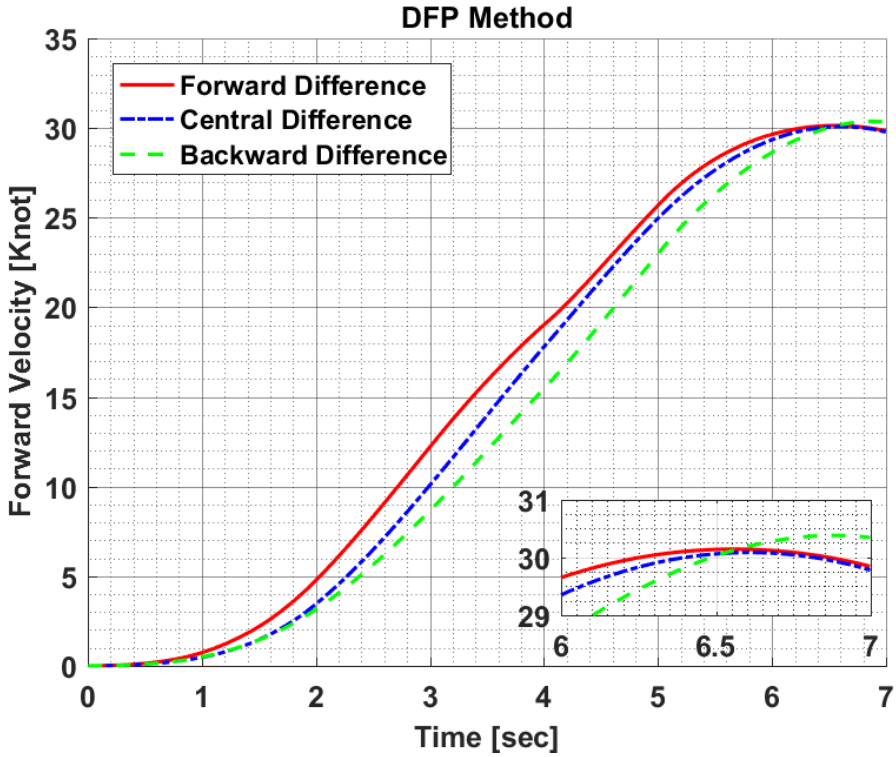


Figure 10.5: Constraint Results of Hover to Forward Flight Maneuver Optimization in the Finite Divided Difference Approximation Comparison for DFP Method

Figure 10.5 shows the constraint results of hover to forward flight maneuver obtained in comparing the finite divided difference approximation for DFP method. Moreover, Figure 10.6 shows the required longitudinal and collective cyclic input to be able to carry out hover to forward flight maneuver as given in Figure 10.5.

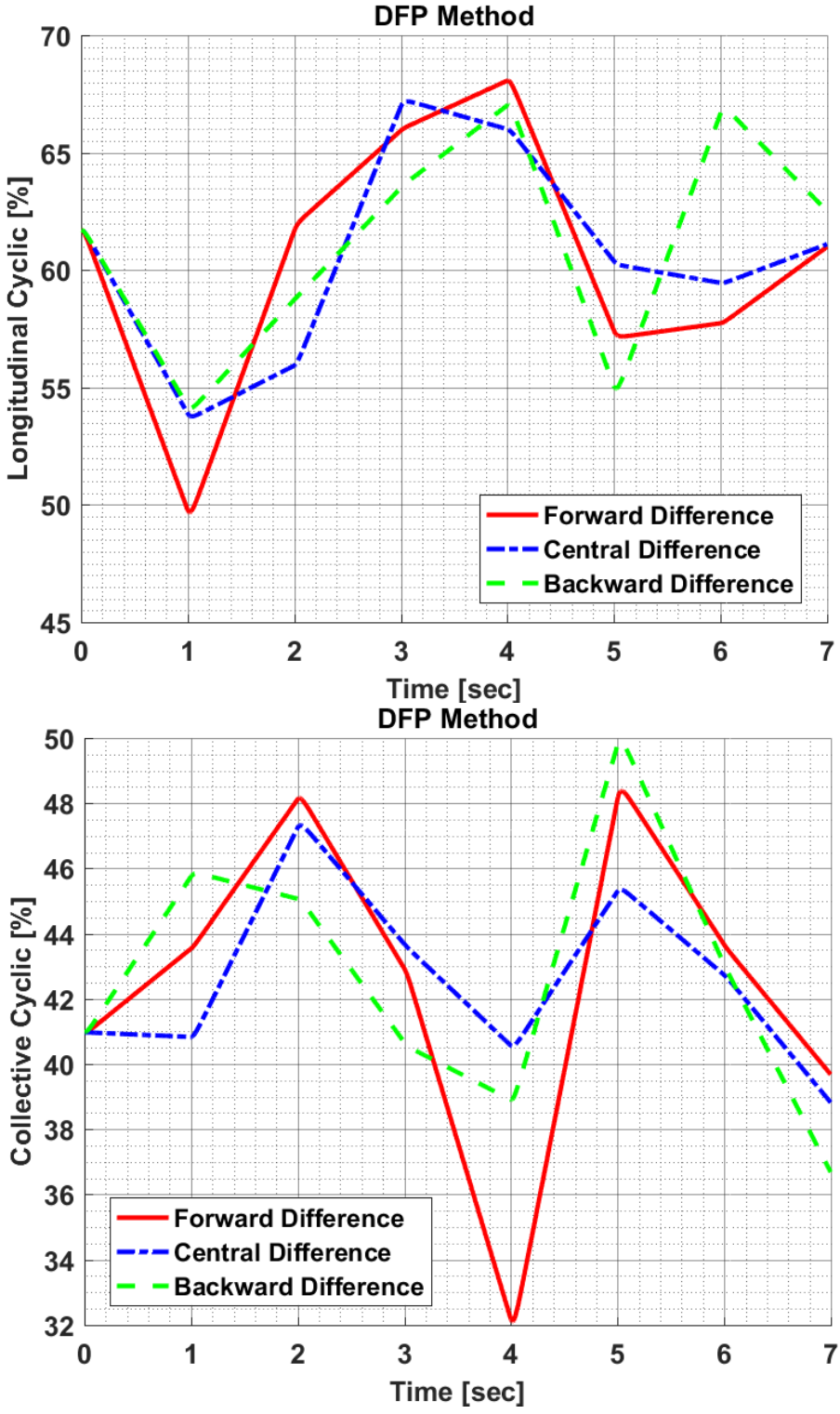
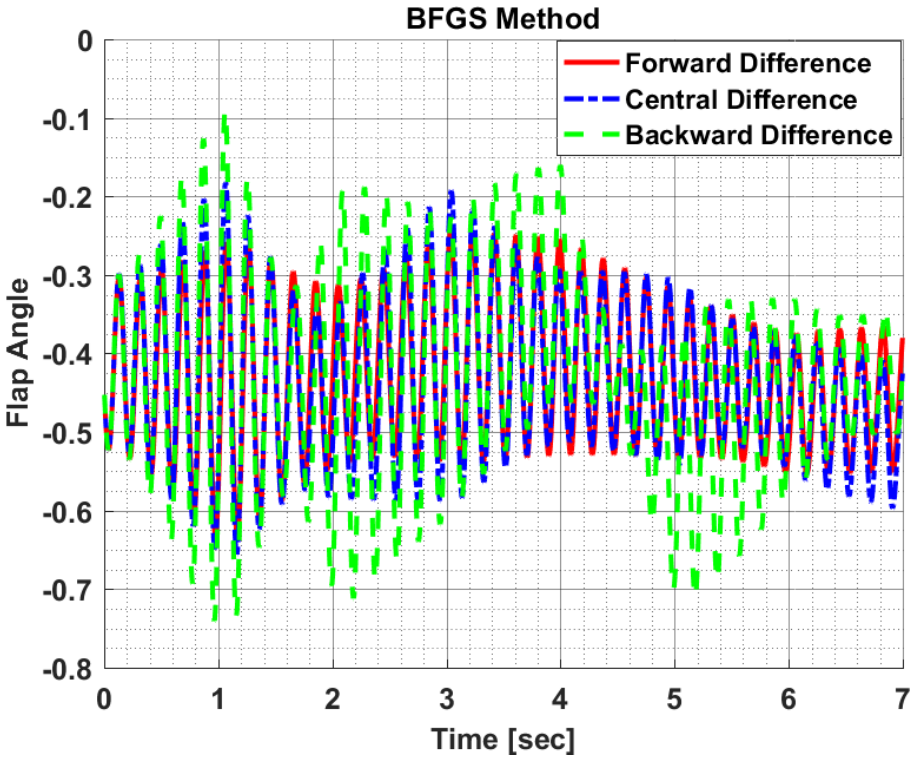
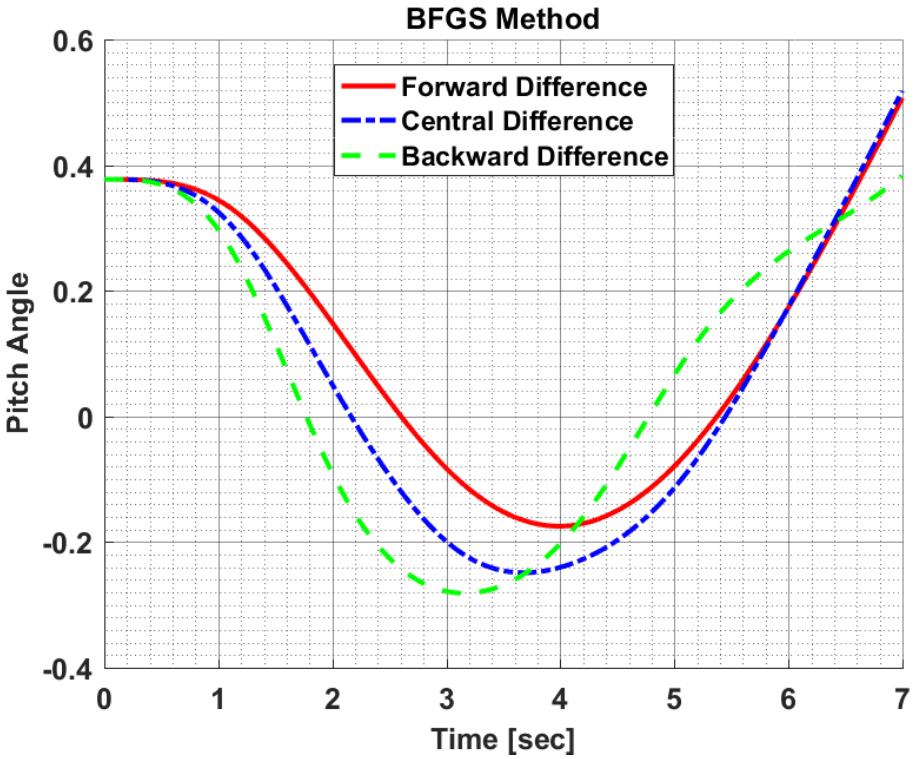
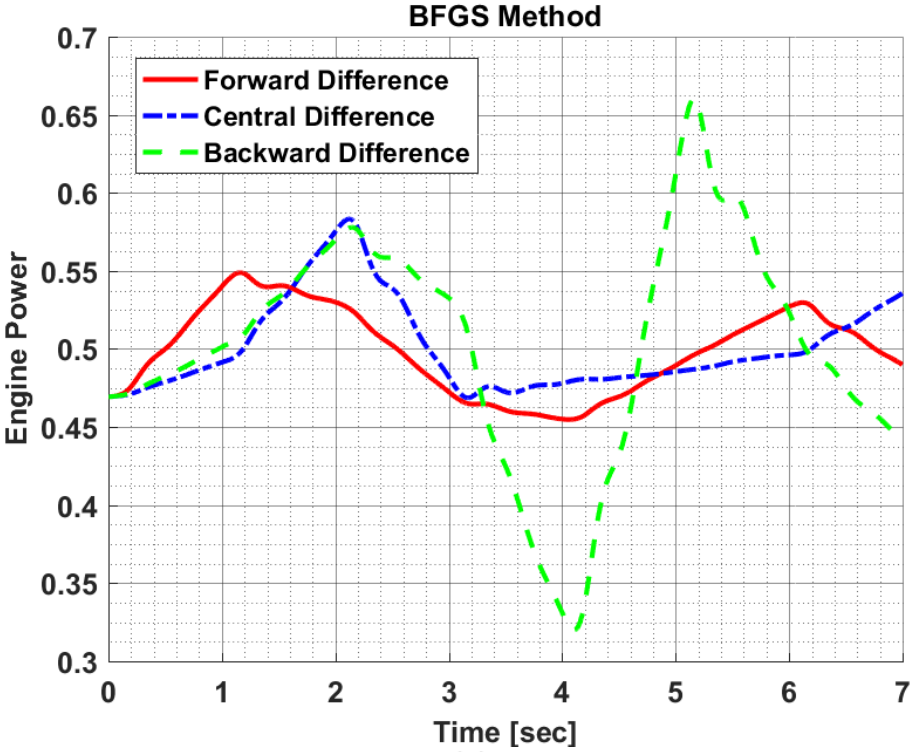


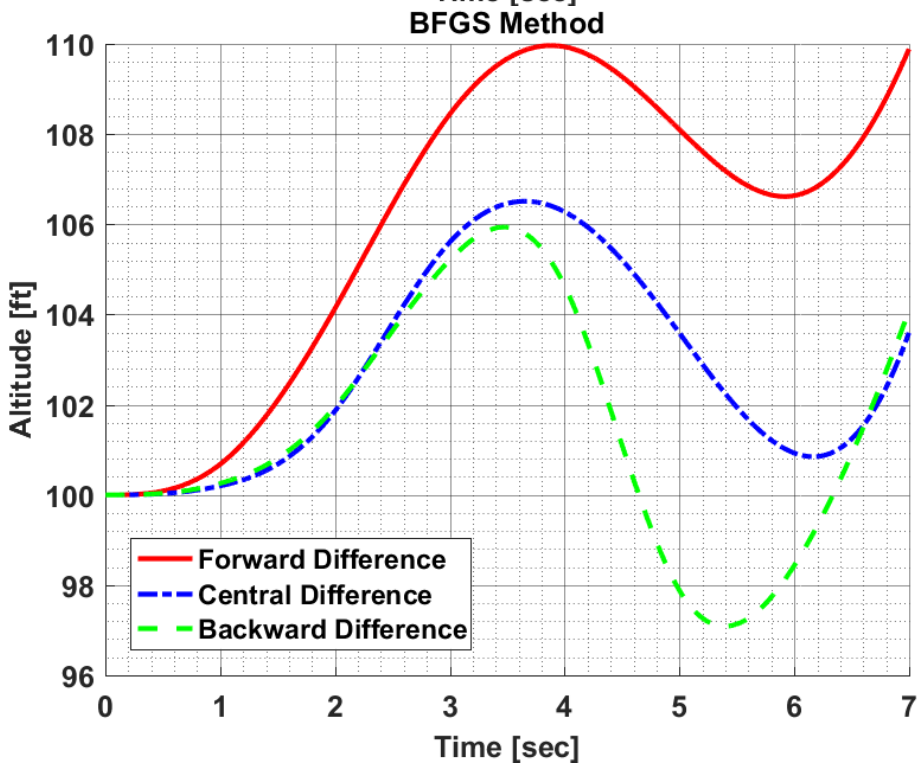
Figure 10.6: Design Variables of Hover to Forward Flight Maneuver Optimization in the Finite Divided Difference Approximation Comparison for DFP Method

Results of BFGS Method:






BFGS Method
Forward Difference
Central Difference
Backward Difference
Engine Power
0.7
0.65
0.6
0.55
0.5
0.45
0.4
0.35
0.3
0
1
2
3
4
5
6
7
Time [sec]


BFGS Method
Forward Difference
Central Difference
Backward Difference
Altitude [ft]
110
108
106
104
102
100
98
96
0
1
2
3
4
5
6
7
Time [sec]

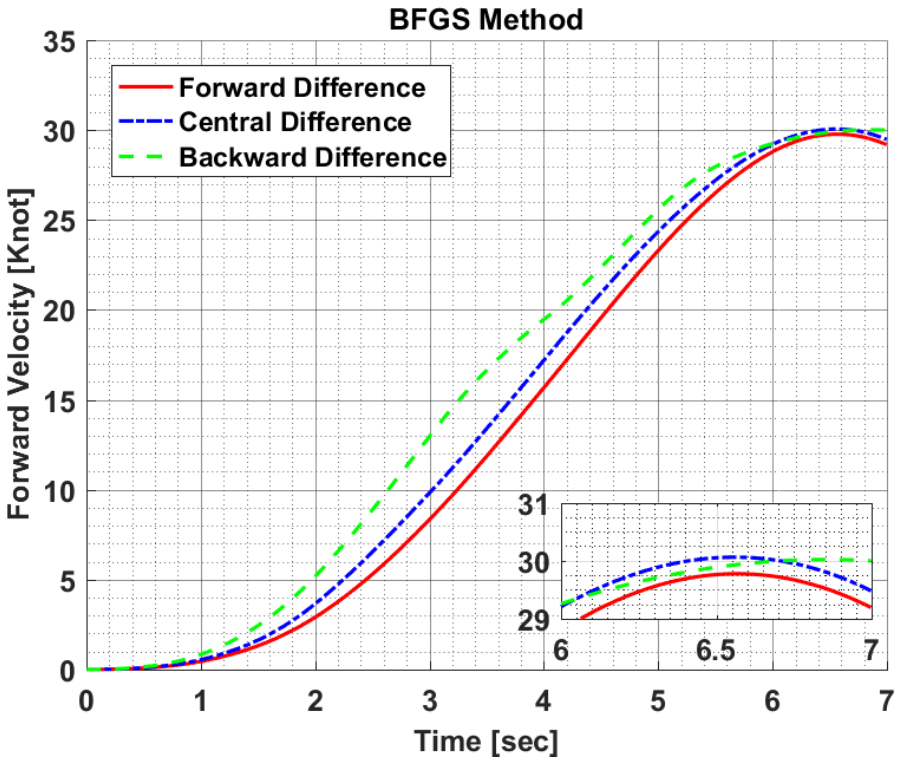


Figure 10.7: Constraint Results of Hover to Forward Flight Maneuver Optimization in the Finite Divided Difference Approximations Comparison for BFGS Method

Figure 10.7 shows the constraint results of hover to forward flight maneuver obtained in comparing the finite divided difference approximation for BFGS method. Moreover, Figure 10.8 shows the required longitudinal and collective cyclic input to be able to carry out hover to forward flight maneuver as given in Figure 10.7.

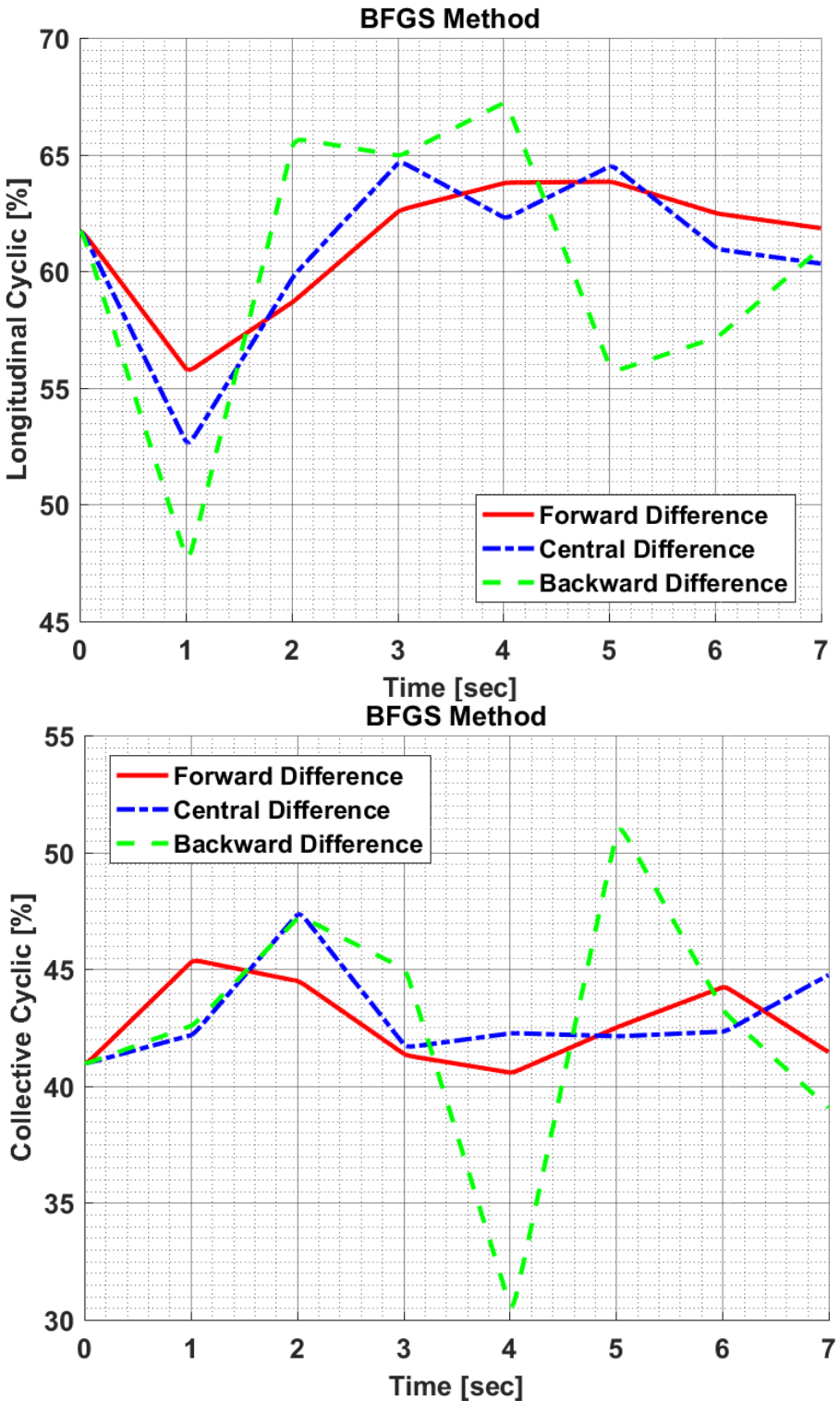


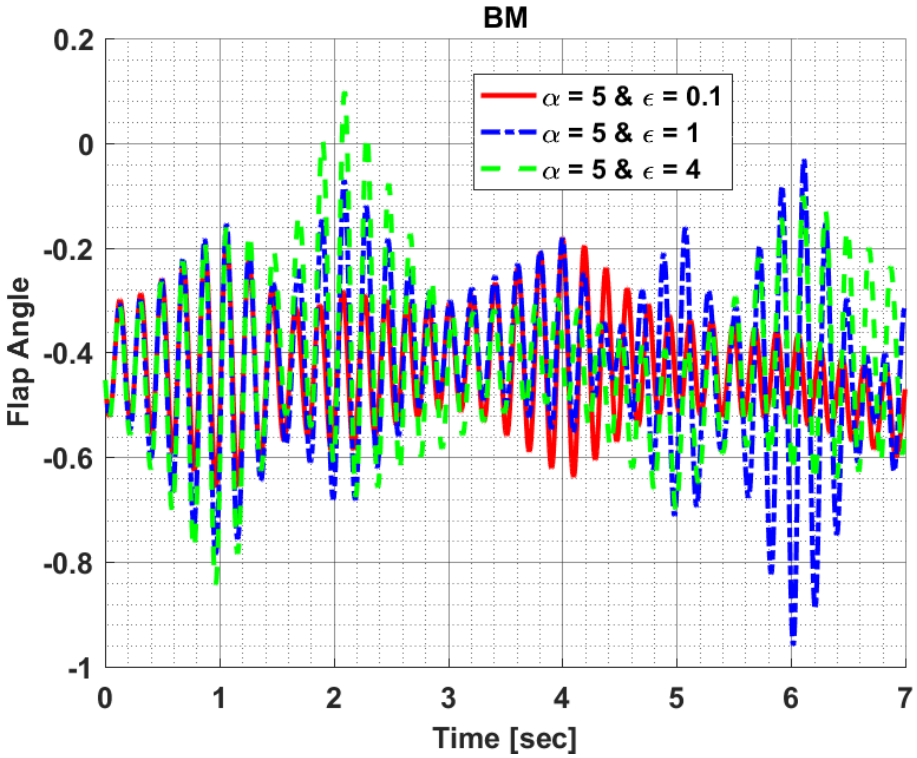
Figure 10.8: Design Variables of Hover to Forward Flight Maneuver Optimization in the Finite Divided Difference Approximation Comparison for BFGS Method

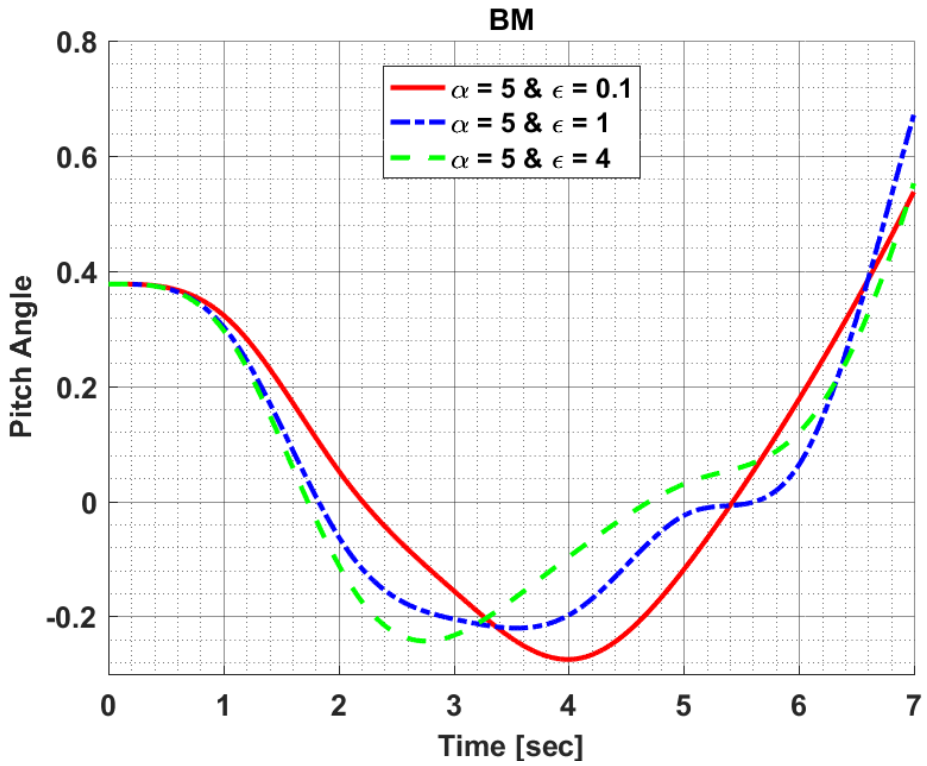
APPENDIX A2: Comparison – Sensitivity Analysis of Perturbation Constant Parameter for Hover to Forward Flight Maneuver Optimization

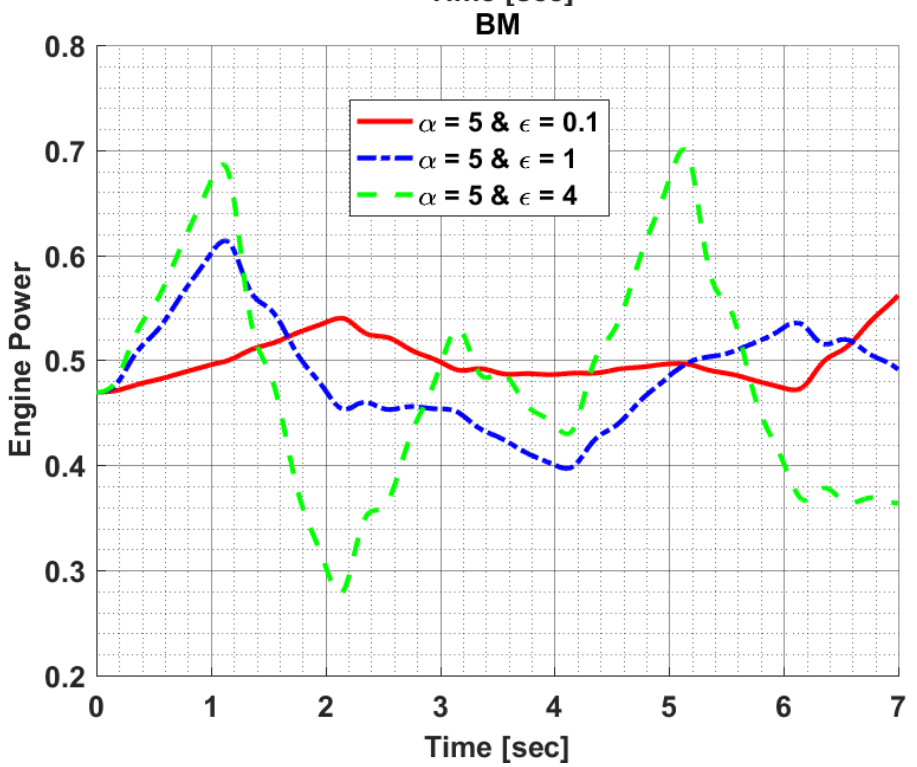
In this appendix section, constraint results and corresponding pilot control inputs are provided for each optimization method configuration which are executed for the sensitivity analysis of perturbation constant parameter. Moreover, they are the results of hover to forward flight maneuver optimization.

Results of BM:

Figure 10.9 shows the constraint results of hover to forward flight maneuver obtained in perturbation constant comparison for Broyden's Method.




BM
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
Pitch Angle
0.8
0.6
0.4
0.2
0
-0.2
0
1
2
3
4
5
6
7
Time [sec]


BM
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
Engine Power
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
1
2
3
4
5
6
7
Time [sec]

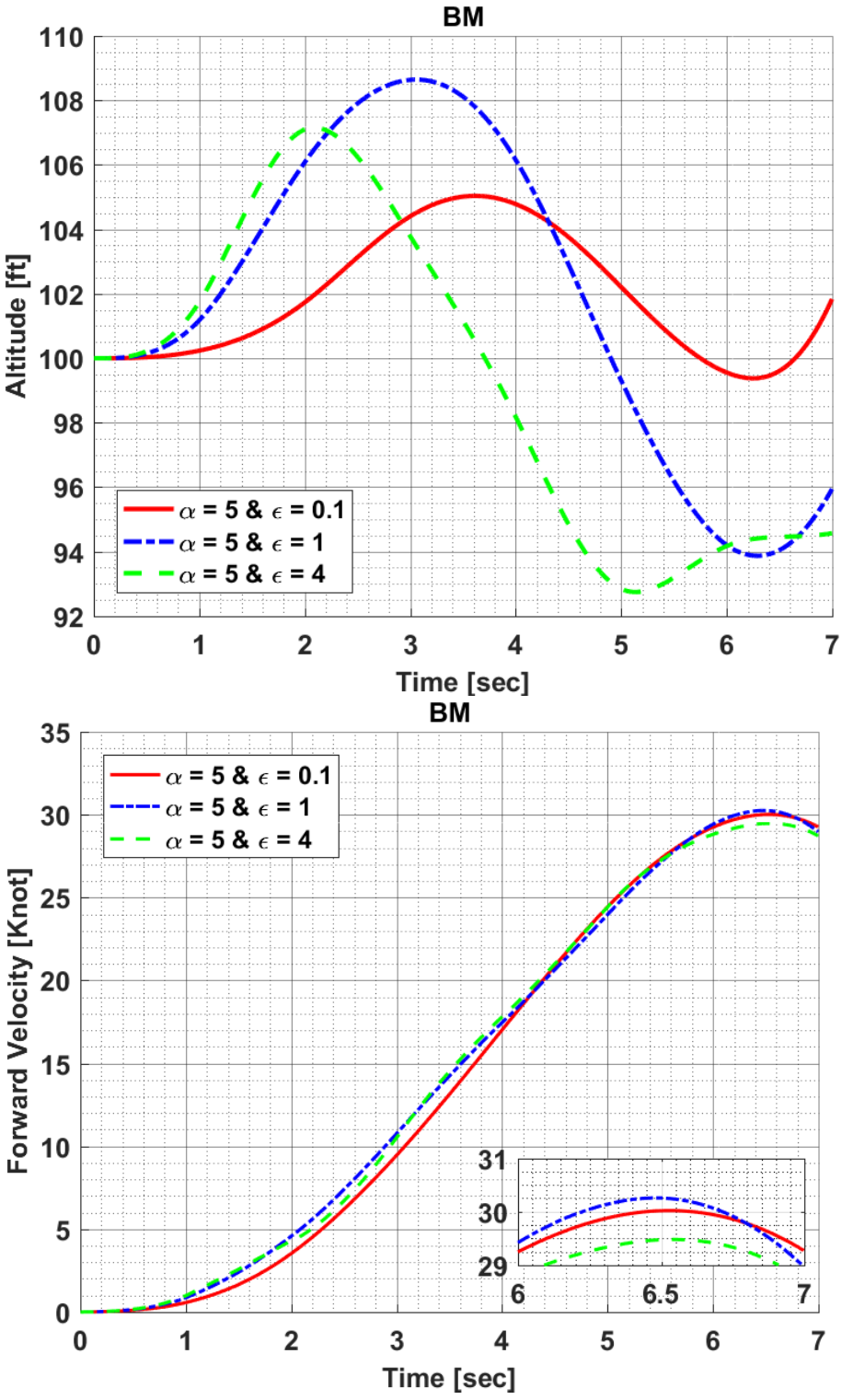


Figure 10.9: Constraint Results of Hover to Forward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for BM

Figure 10.10 shows the required longitudinal and collective cyclic input to be able to carry out hover to forward flight maneuver as given in Figure 10.9.

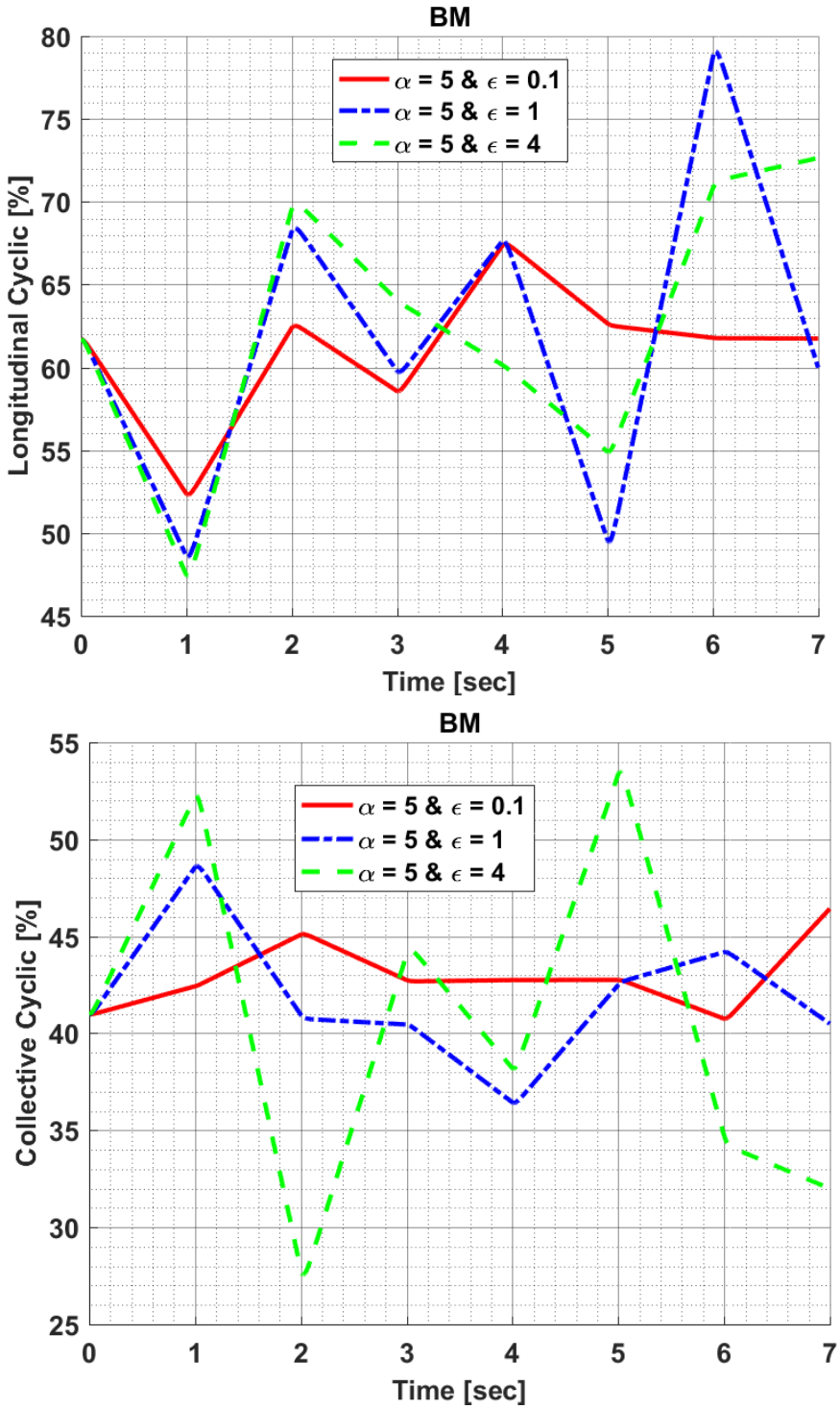
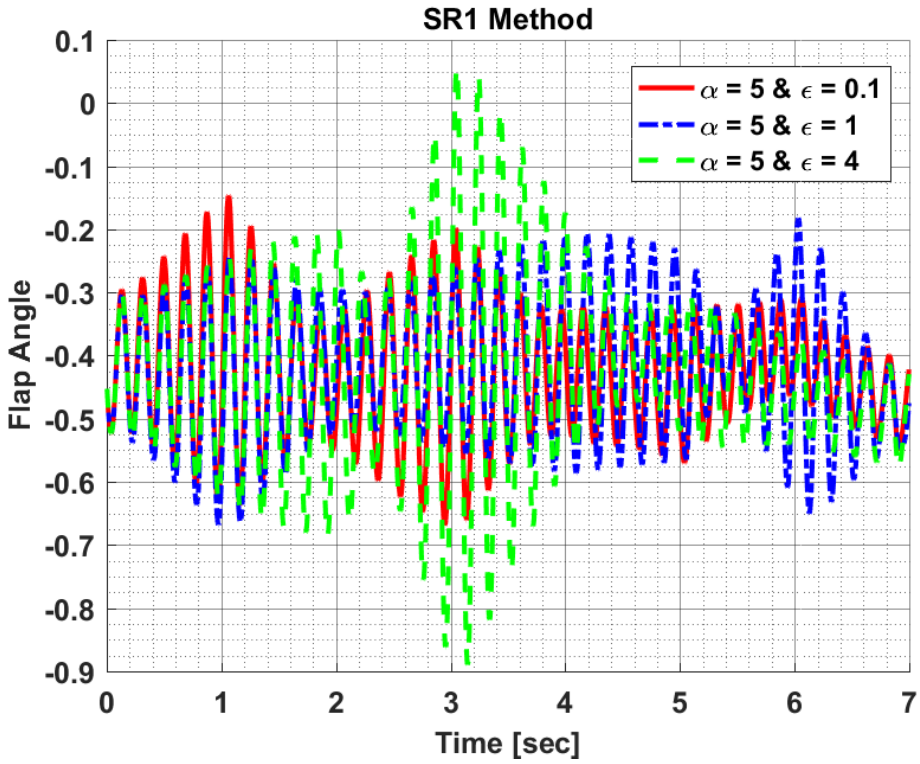
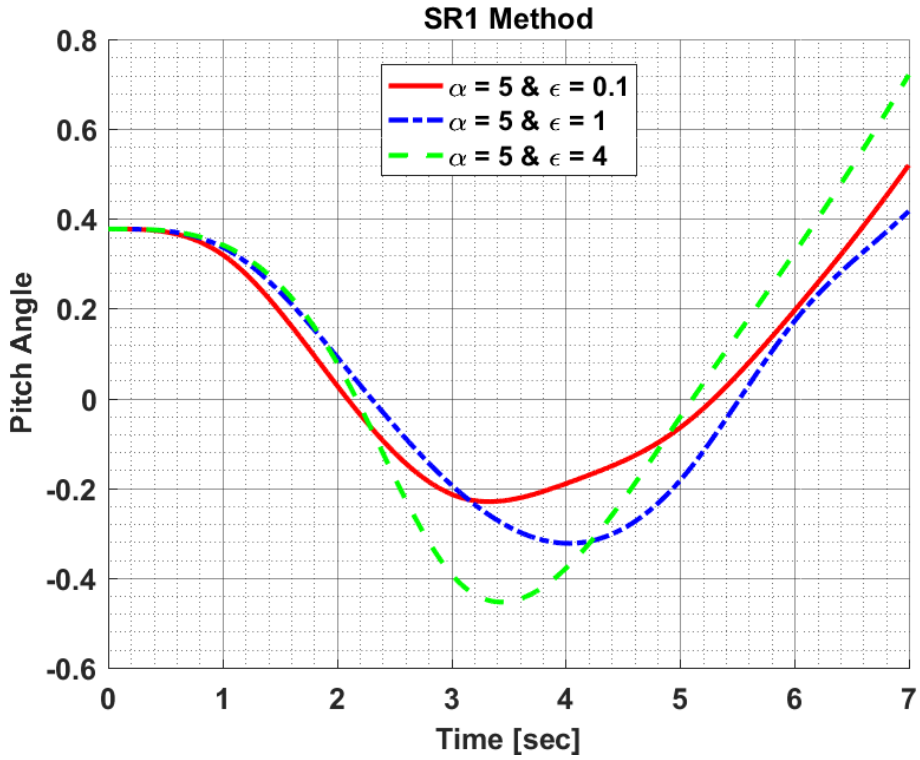
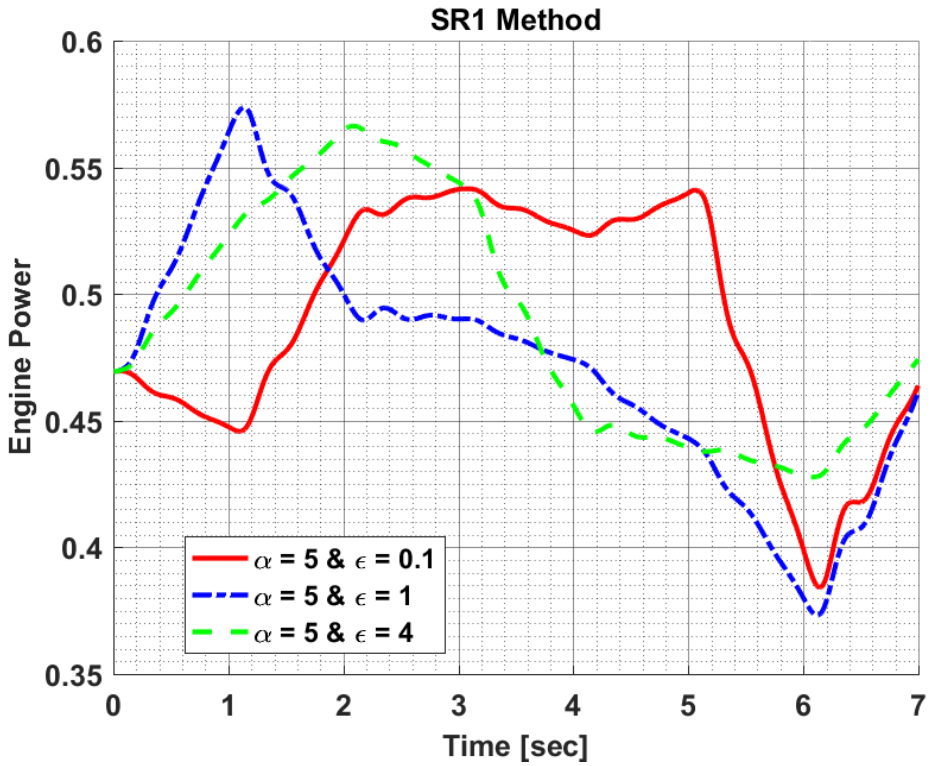


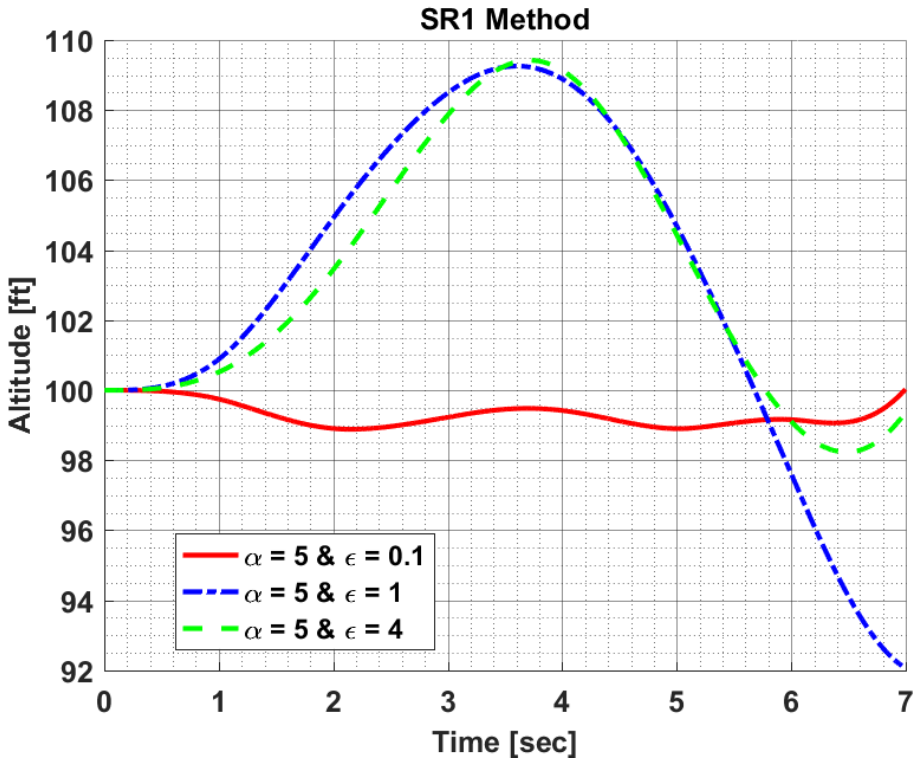
Figure 10.10: Design Variables of Hover to Forward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for BM

Results of SR1 Method:






SR1 Method
Engine Power
Time [sec]
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4


SR1 Method
Altitude [ft]
Time [sec]
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4

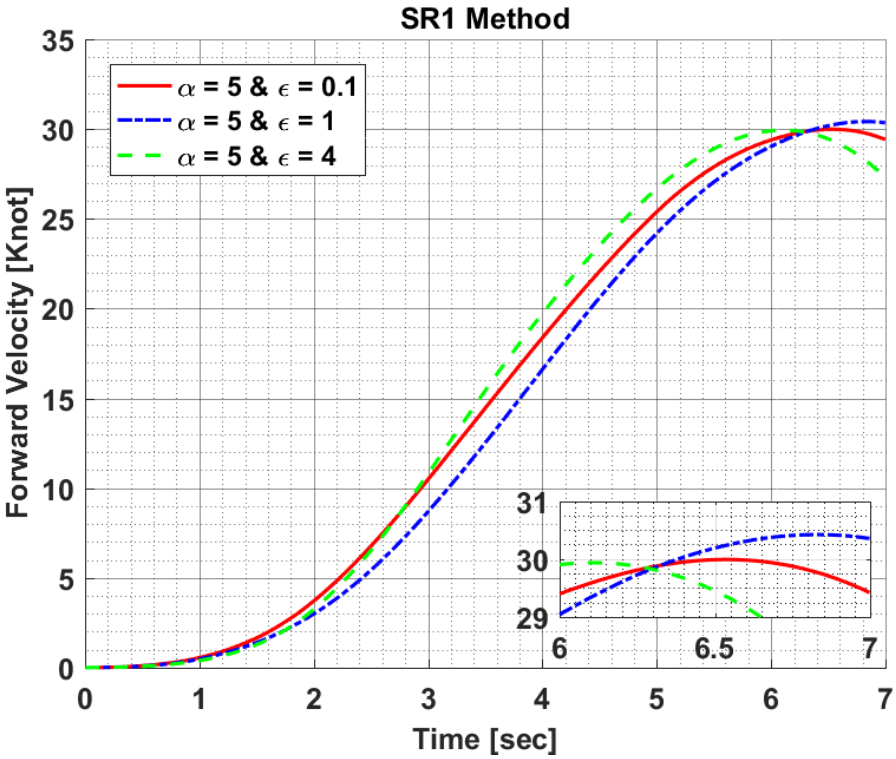


Figure 10.11: Constraint Results of Hover to Forward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for SR1 Method

Figure 10.11 shows the constraint results of hover to forward flight maneuver obtained in perturbation constant comparison for SR1. Moreover, Figure 10.12 shows the required longitudinal and collective cyclic input to be able to carry out hover to forward flight maneuver as given in Figure 10.11.

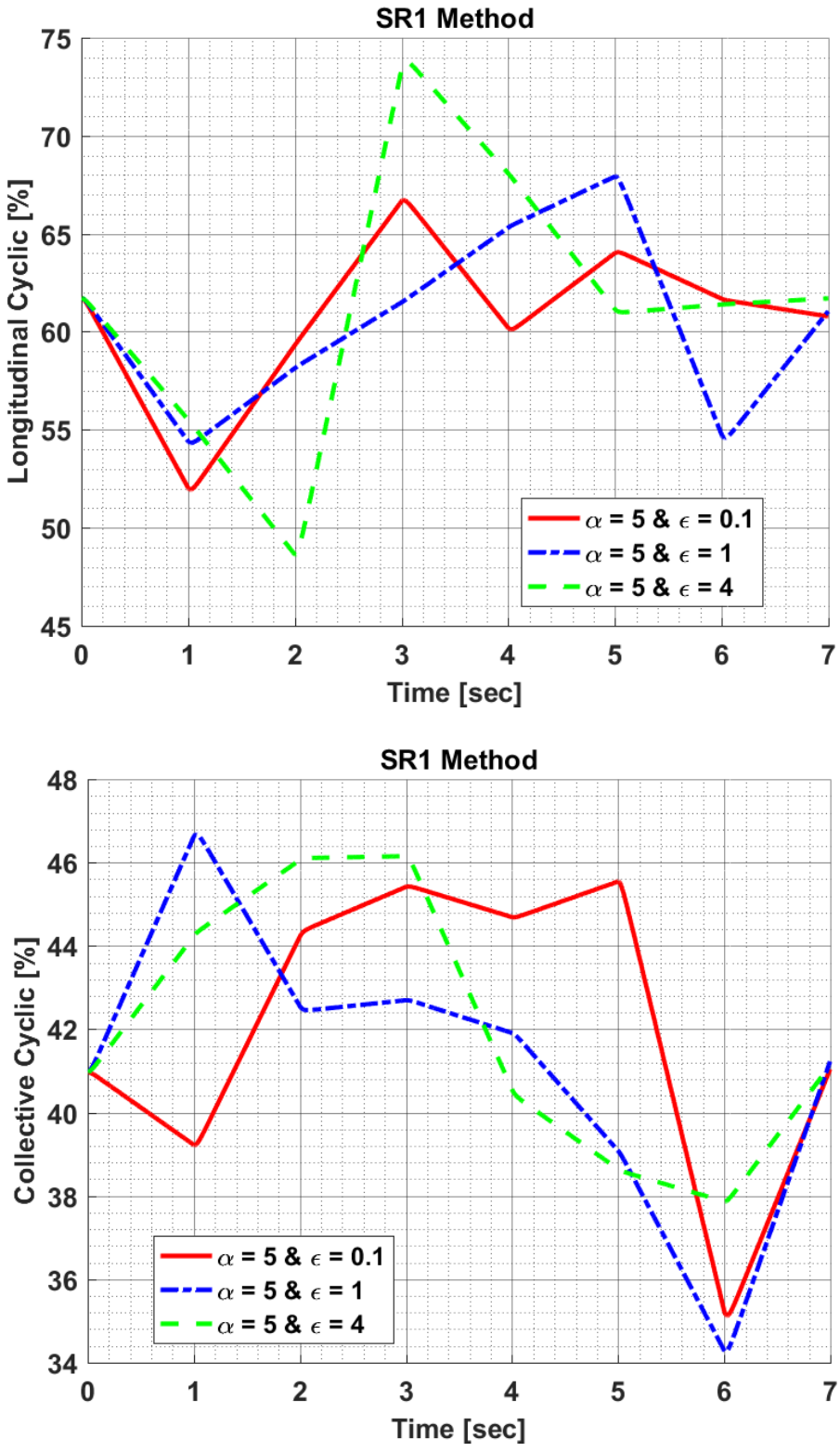
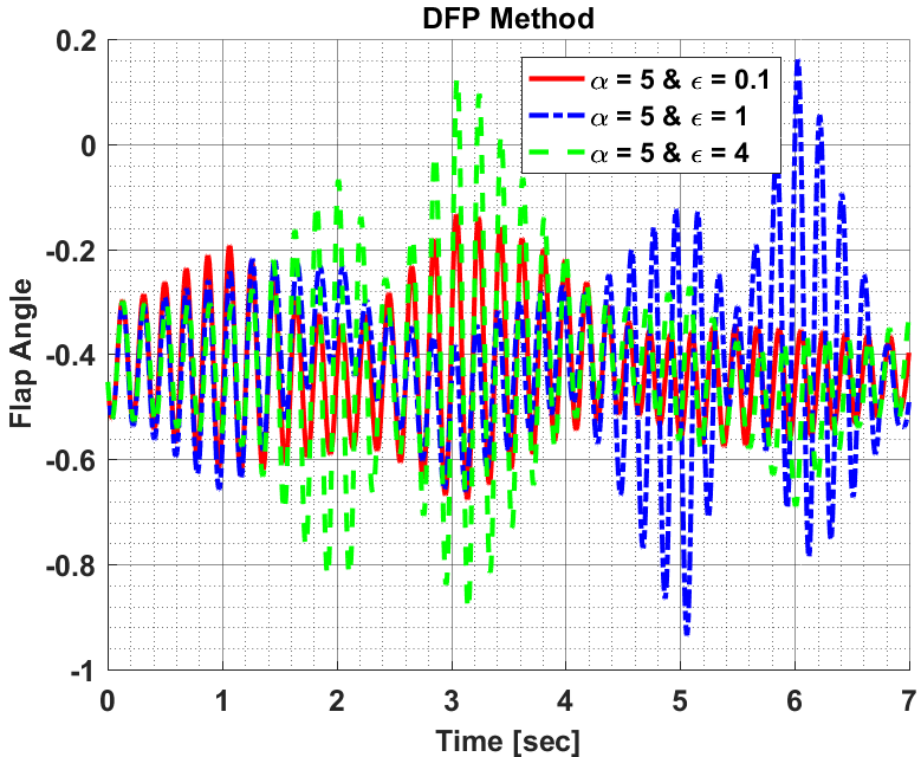
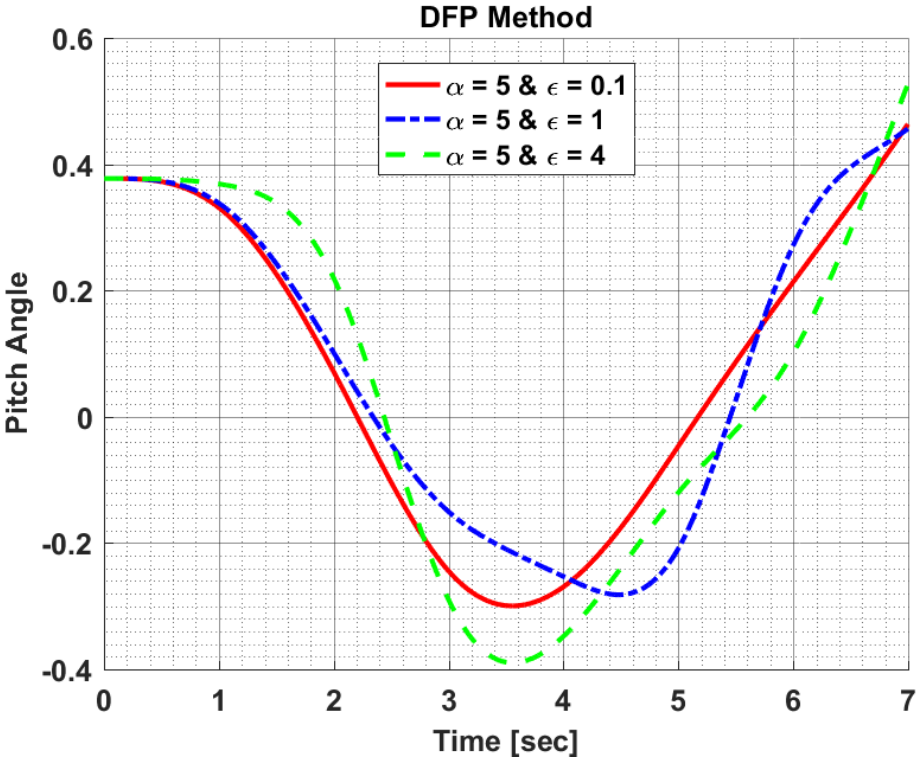
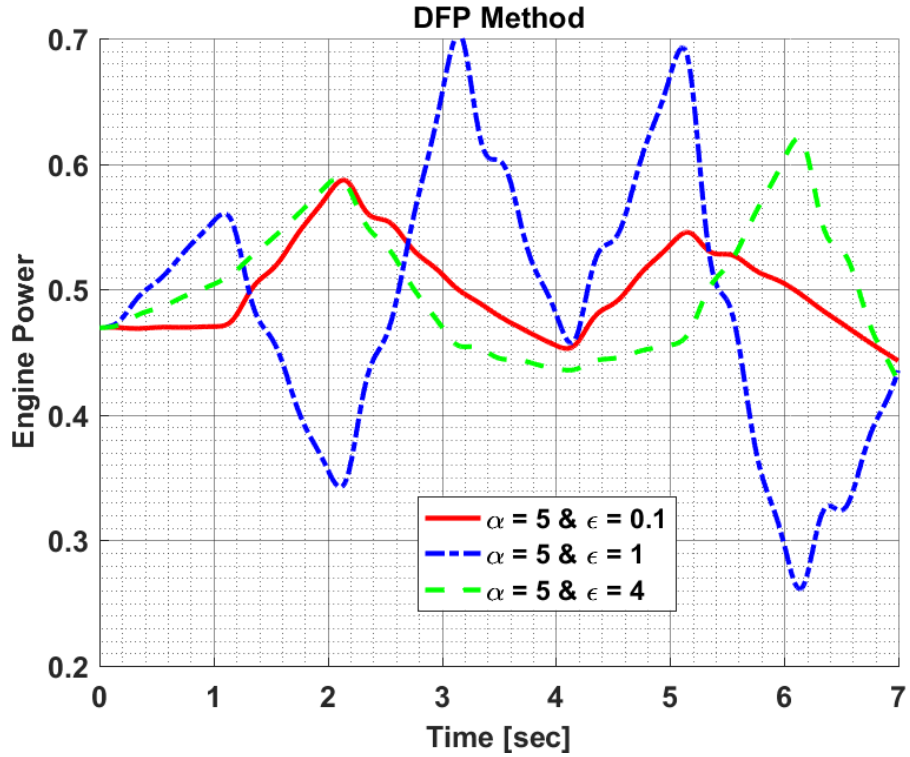


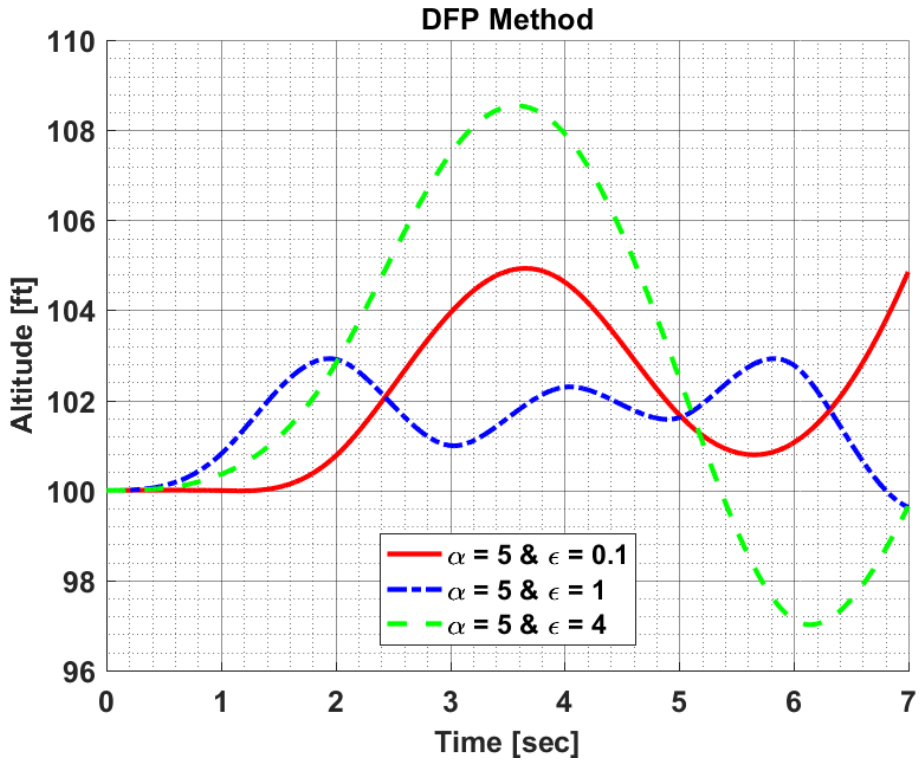
Figure 10.12: Design Variables of Hover to Forward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for SR1 Method

Results of DFP Method:






DFP Method
Engine Power
Time [sec]
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4


DFP Method
Altitude [ft]
Time [sec]
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4

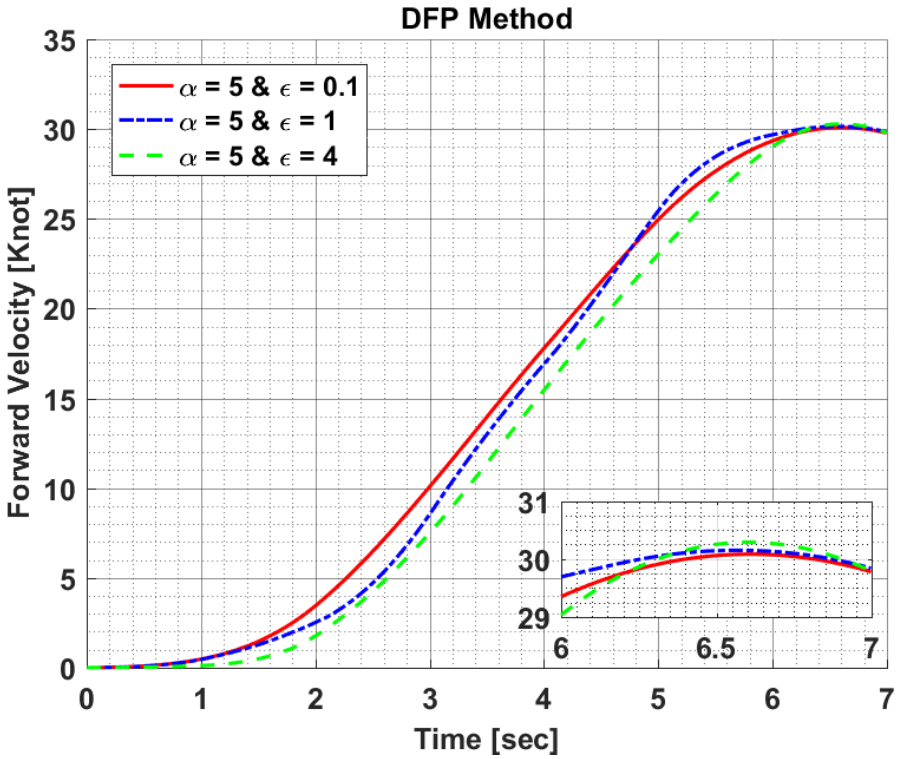


Figure 10.13: Constraint Results of Hover to Forward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for DFP Method

Figure 10.13 shows the constraint results of hover to forward flight maneuver obtained in perturbation constant comparison for DFP. Moreover, Figure 10.14 shows the required longitudinal and collective cyclic input to be able to carry out hover to forward flight maneuver as given in Figure 10.13.

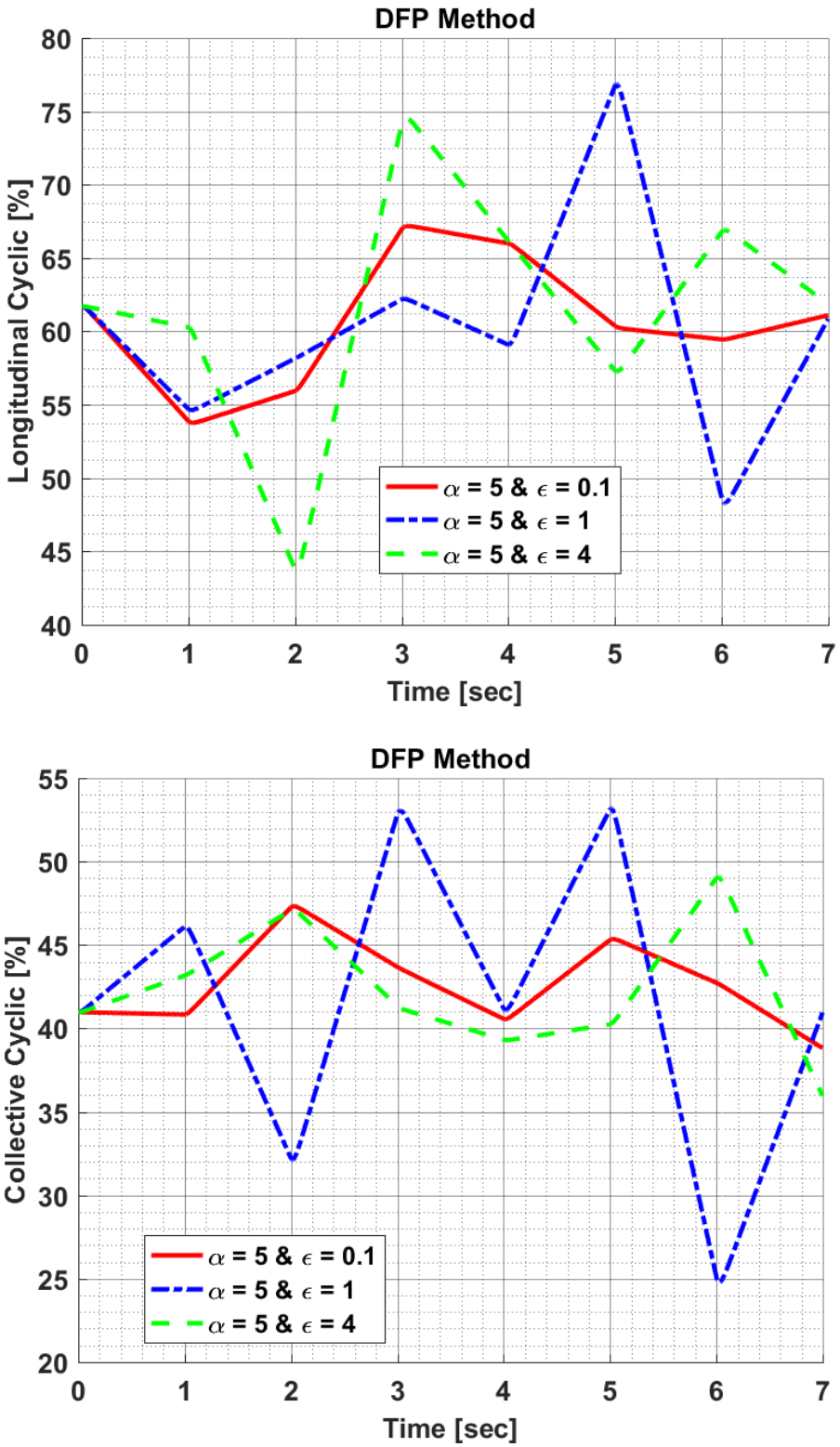
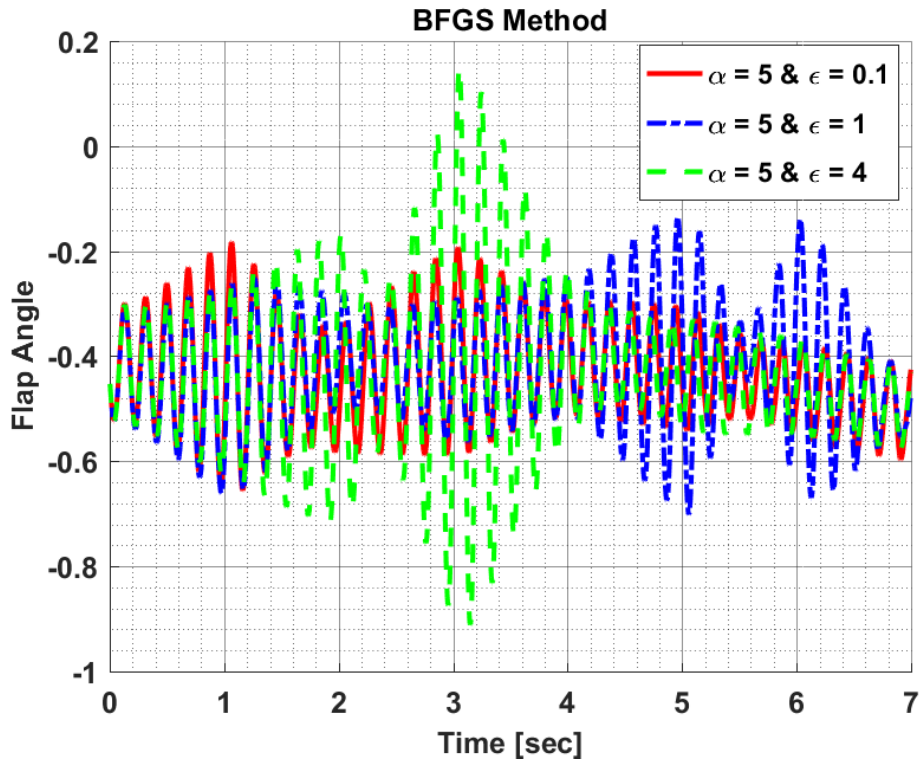
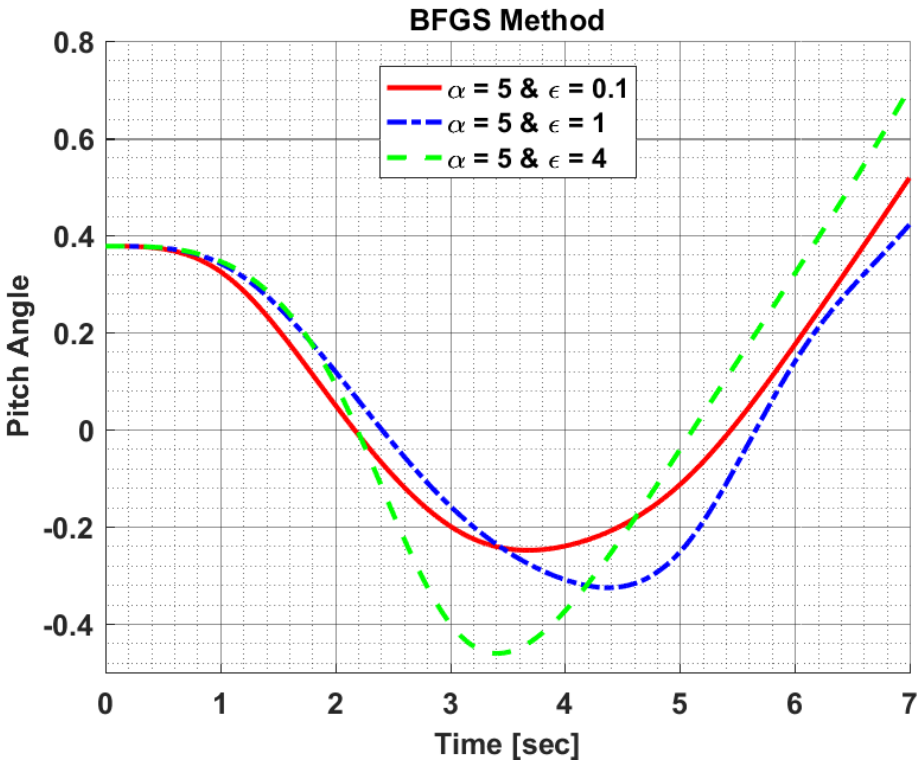
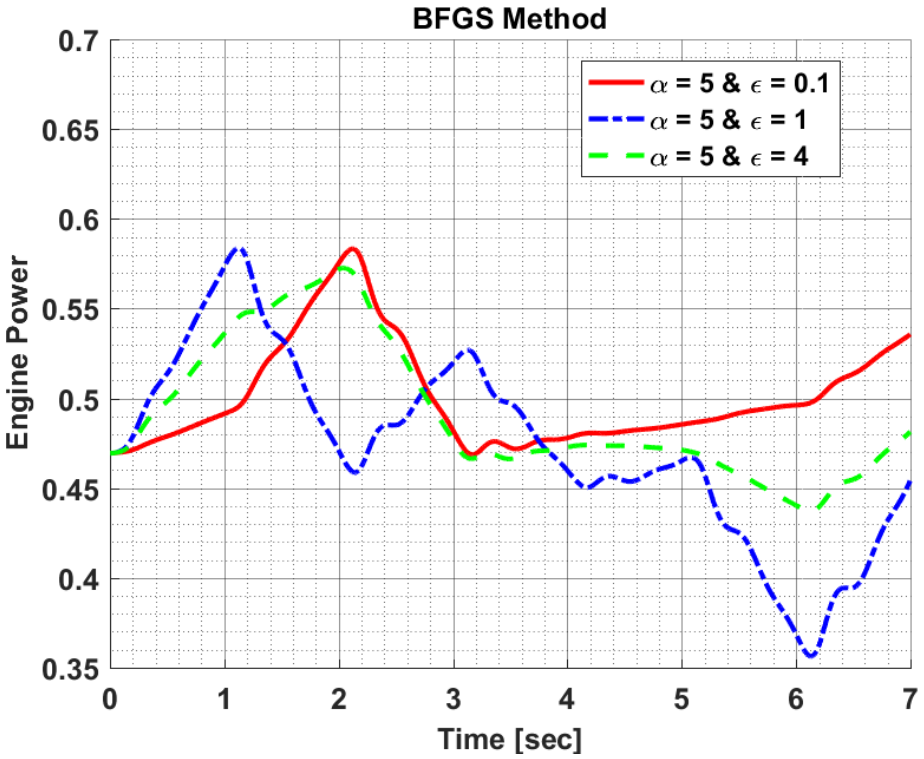


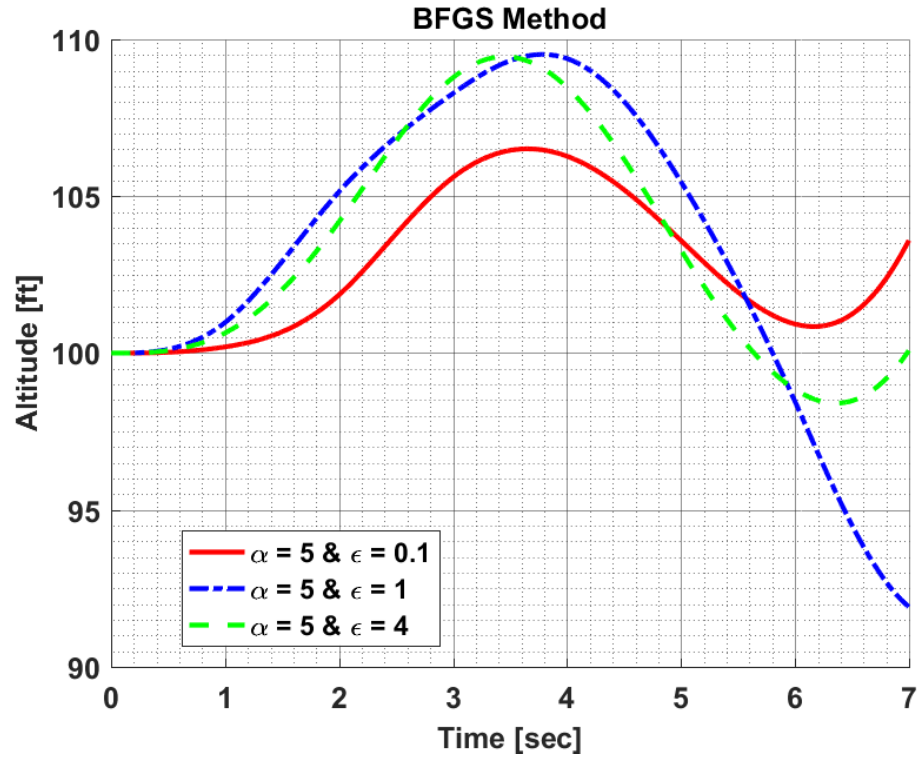
Figure 10.14: Design Variables of Hover to Forward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for DFP Method

Results of BFGS Method:






BFGS Method
Engine Power
Time [sec]
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
0.7
0.65
0.6
0.55
0.5
0.45
0.4
0.35
0
1
2
3
4
5
6
7


BFGS Method
Altitude [ft]
Time [sec]
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
110
105
100
95
90
0
1
2
3
4
5
6
7

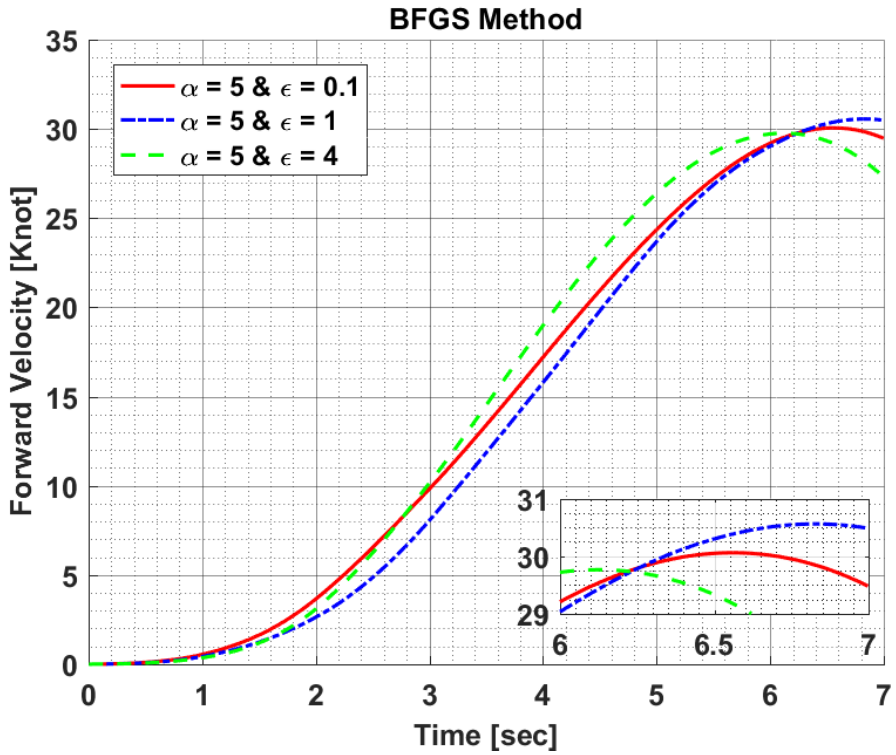


Figure 10.15: Constraint Results of Hover to Forward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for BFGS Method

Figure 10.15 shows the constraint results of hover to forward flight maneuver obtained in perturbation constant comparison for BFGS. Moreover, Figure 10.16 shows the required longitudinal and collective cyclic input to be able to carry out hover to forward flight maneuver as given in Figure 10.15.

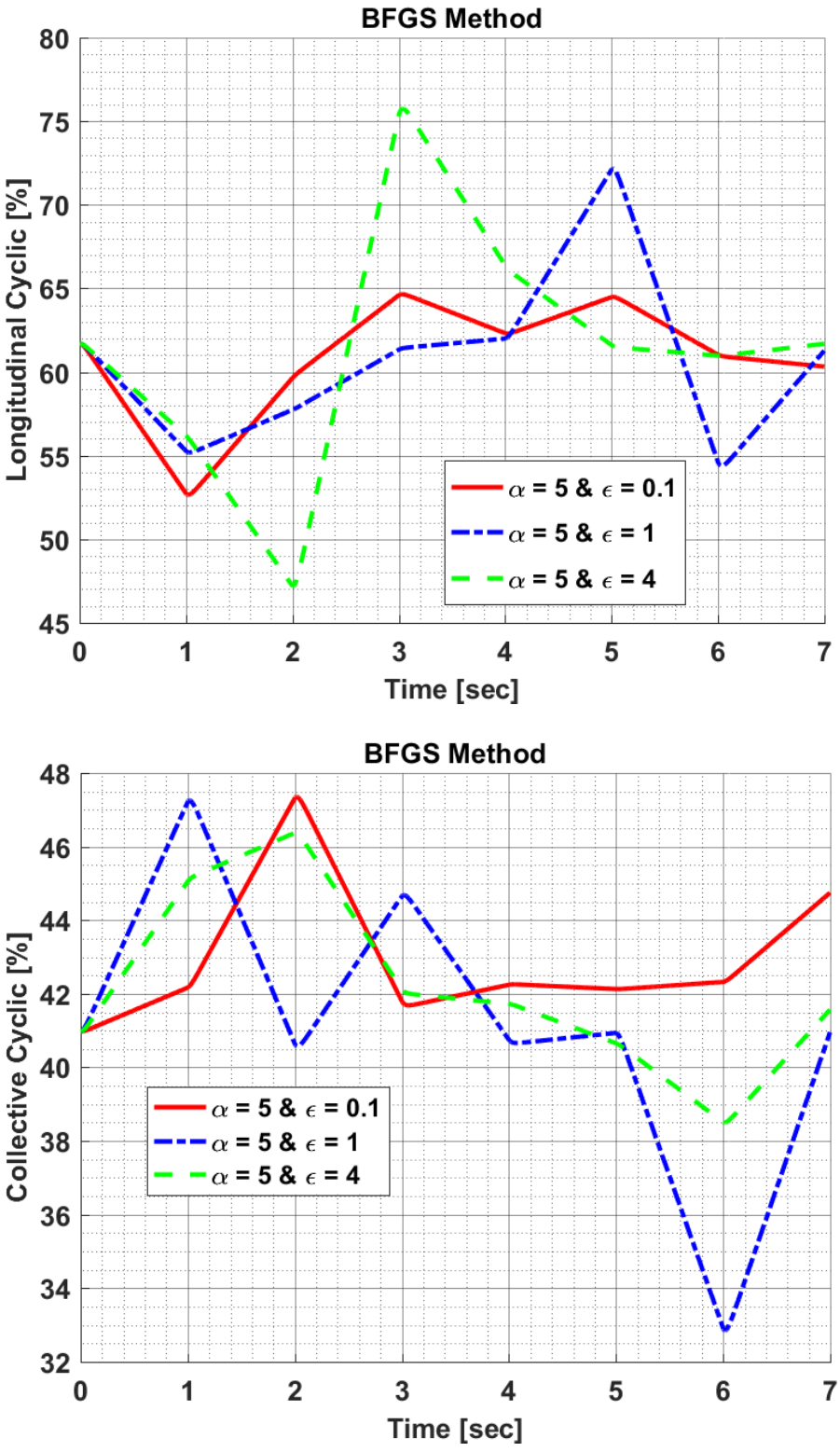


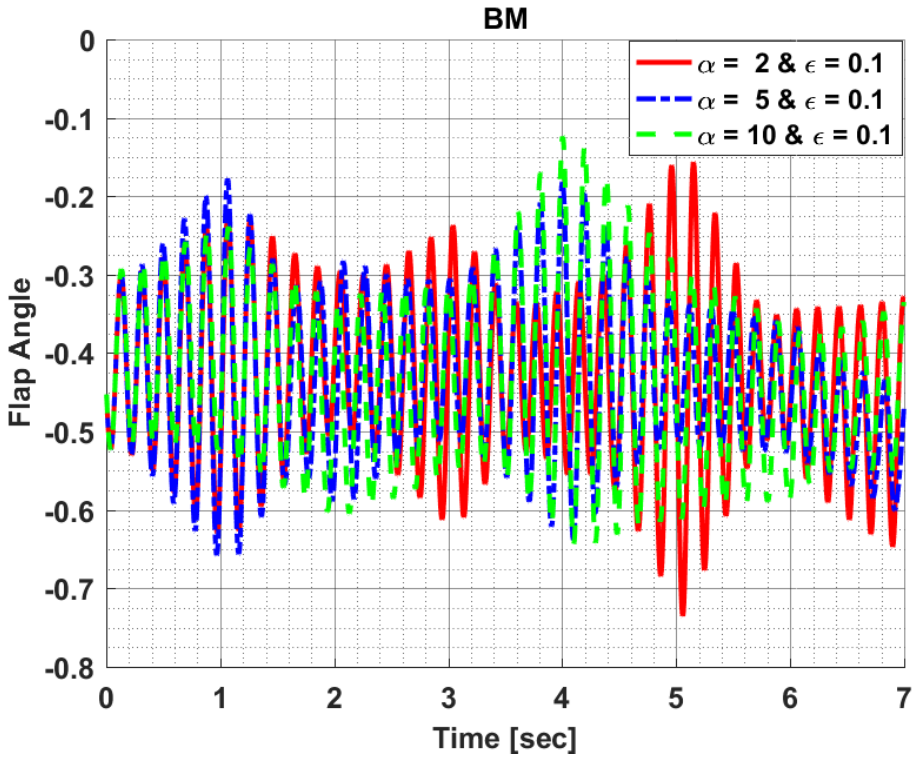
Figure 10.16: Design Variables of Hover to Forward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for BFGS Method

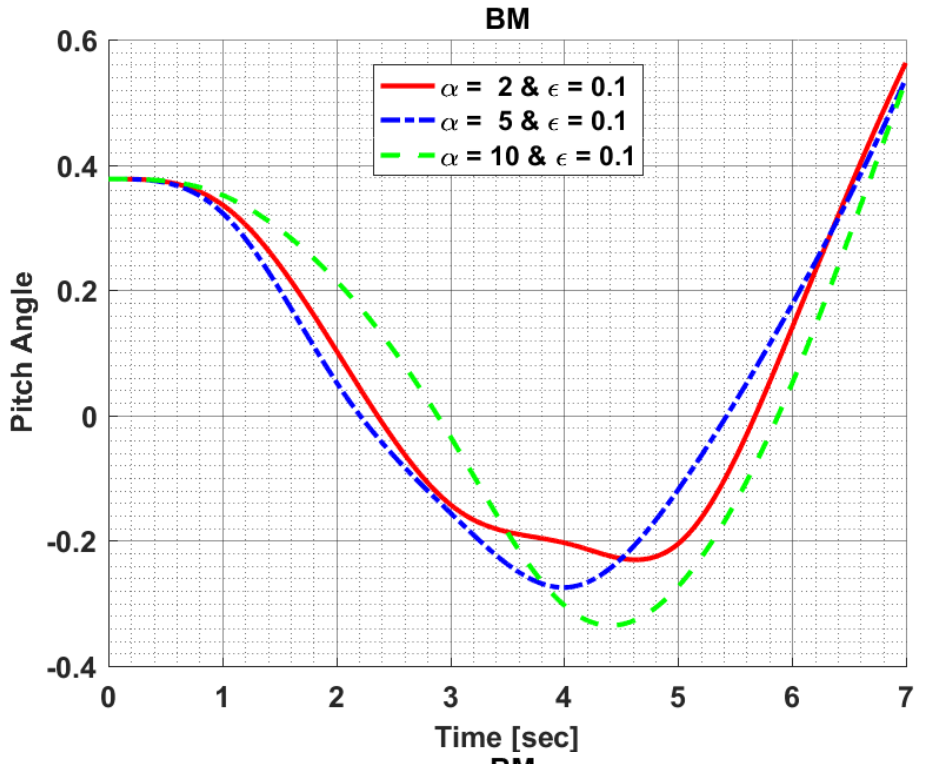
APPENDIX A3: Comparison – Sensitivity Analysis of Step Size Parameter for Hover to Forward Flight Maneuver Optimization

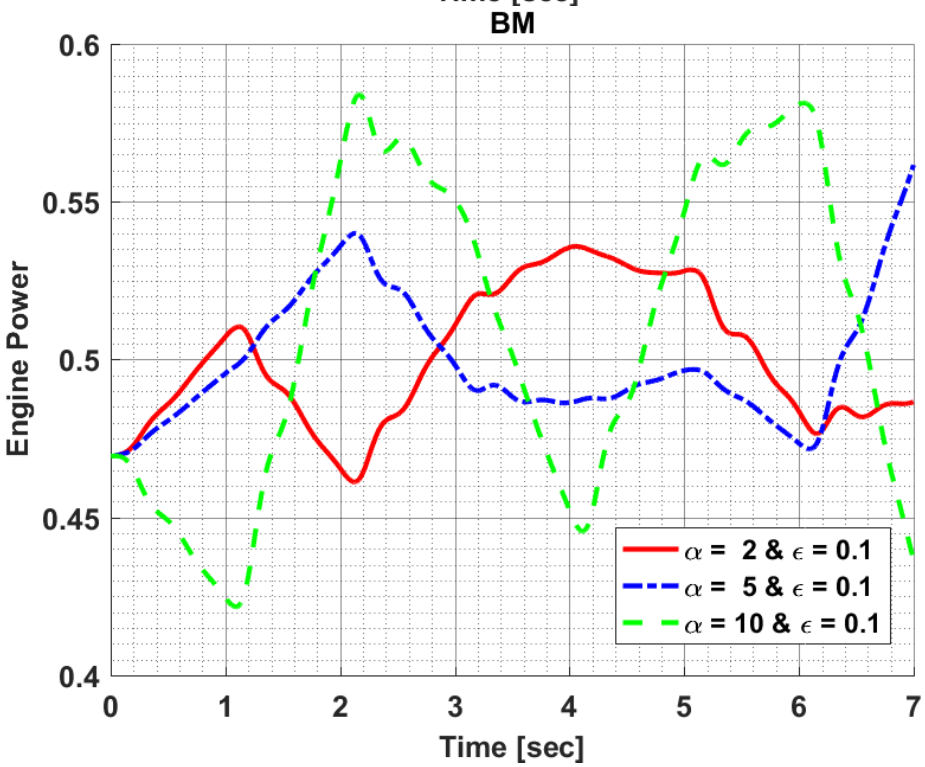
In this appendix section, constraint results and corresponding pilot control inputs are provided for each optimization method configuration which are executed for the sensitivity analysis of step size parameter. Moreover, they are the results of hover to forward flight maneuver optimization.

Results of BM:

Figure 10.17 shows the constraint results of hover to forward flight maneuver obtained in Step size comparison for Broyden's Method.




BM
Pitch Angle
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1


BM
Engine Power
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1

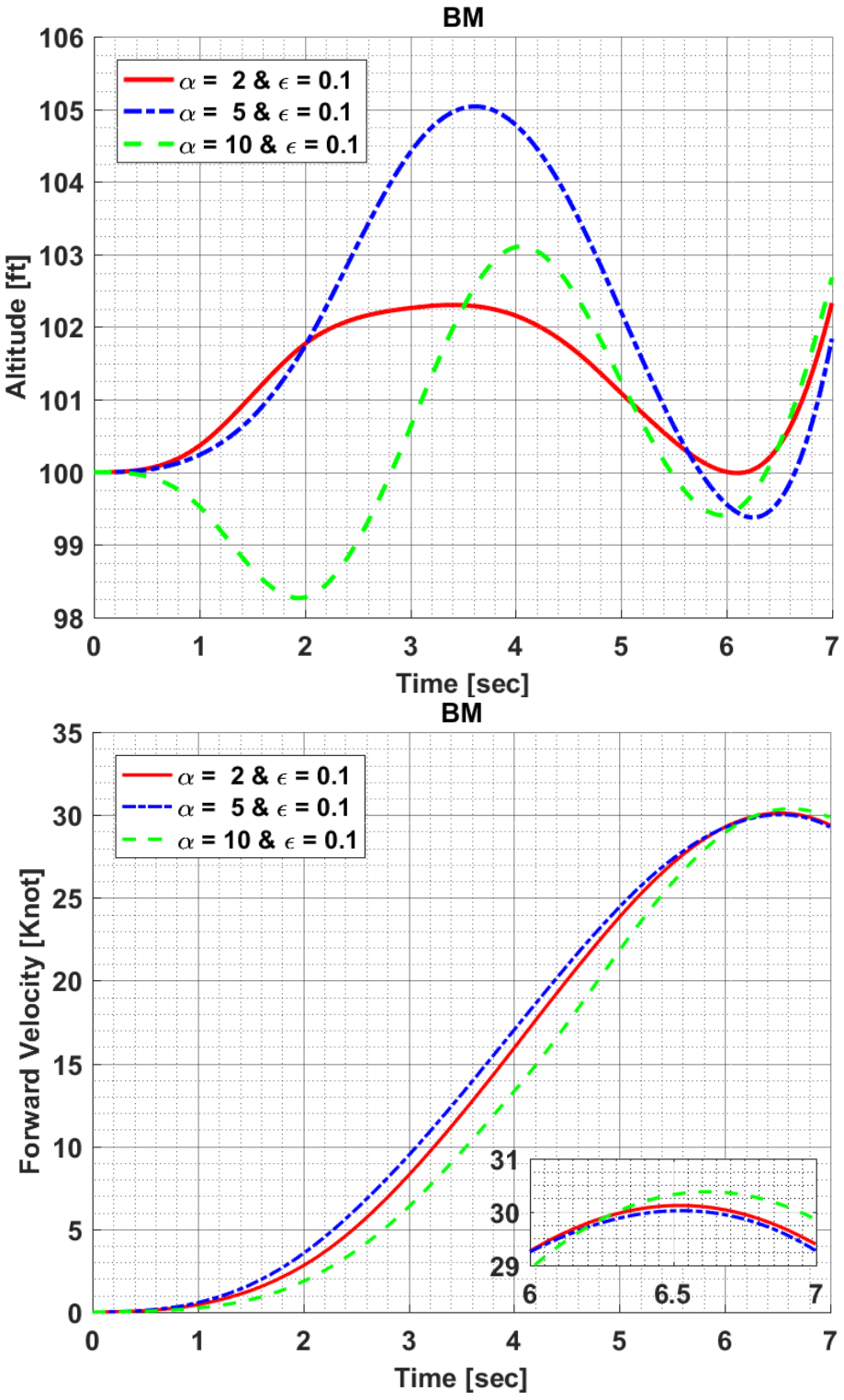


Figure 10.17: Constraint Results of Hover to Forward Flight Maneuver Optimization in Step Size Sensitivity Analysis for BM

Figure 10.18 shows the required longitudinal and collective cyclic input to be able to carry out hover to forward flight maneuver as given in Figure 10.17.

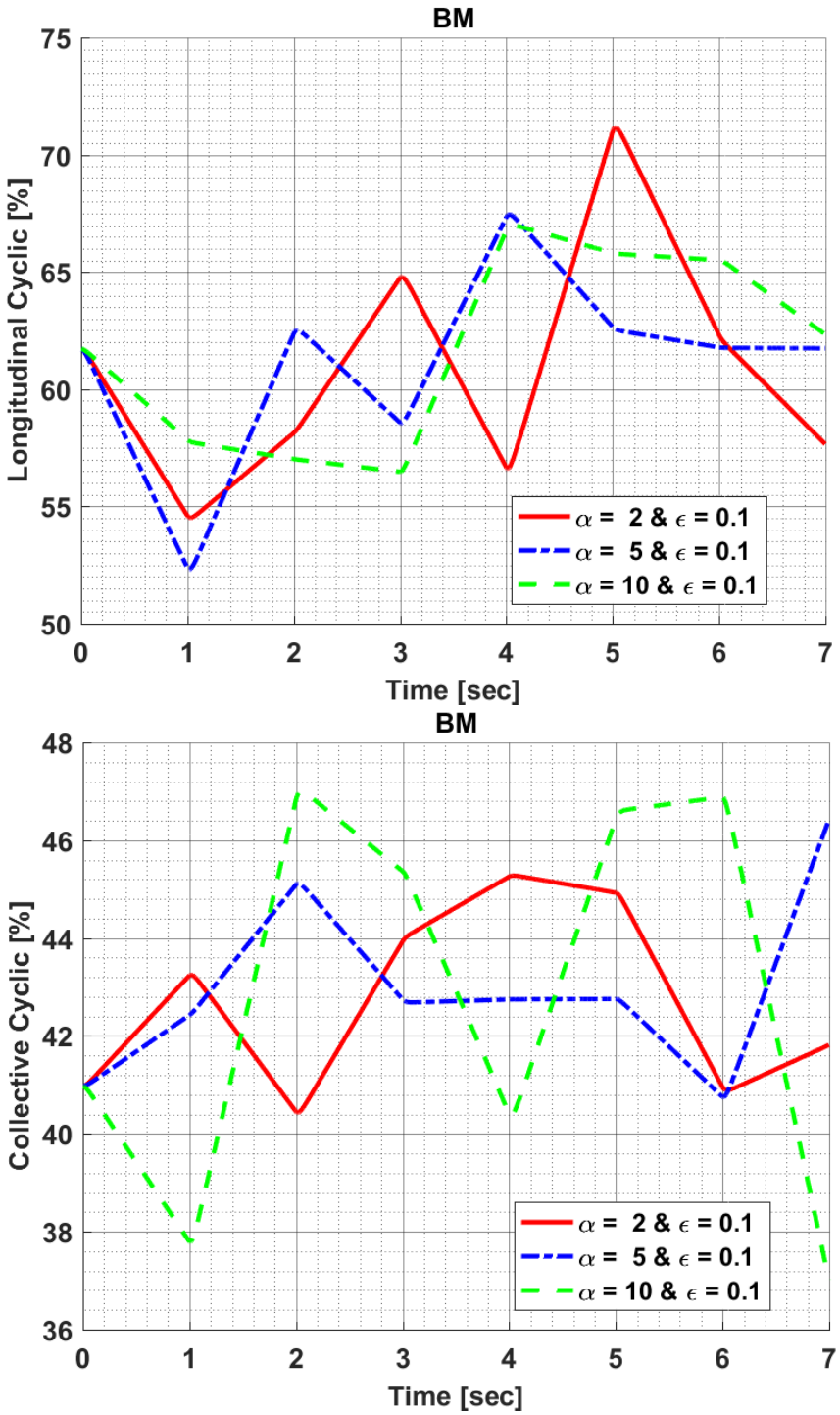
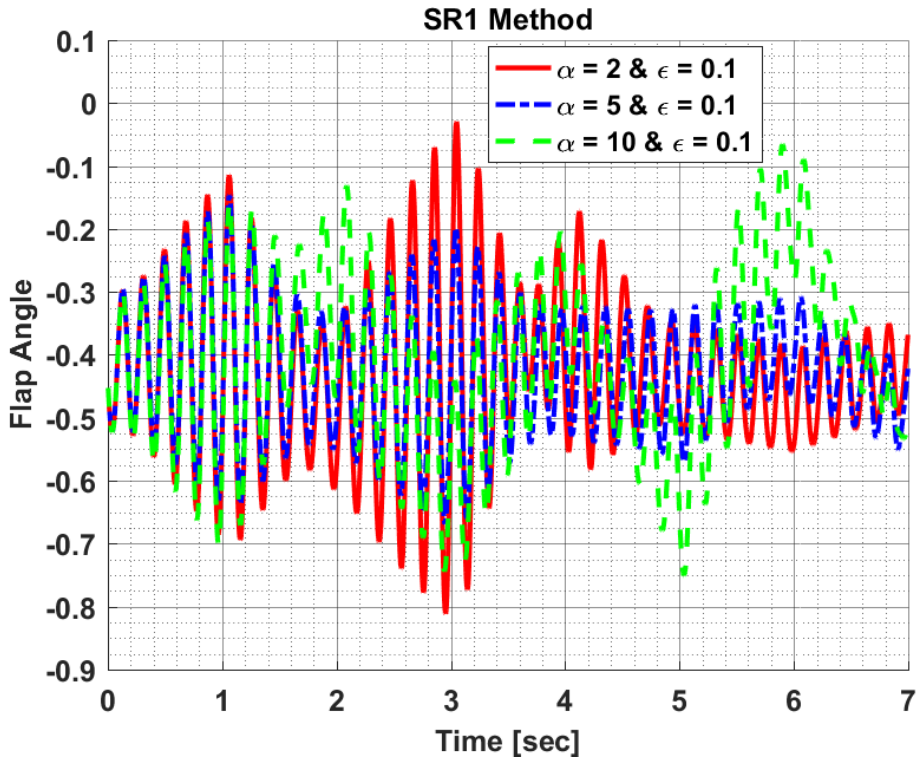
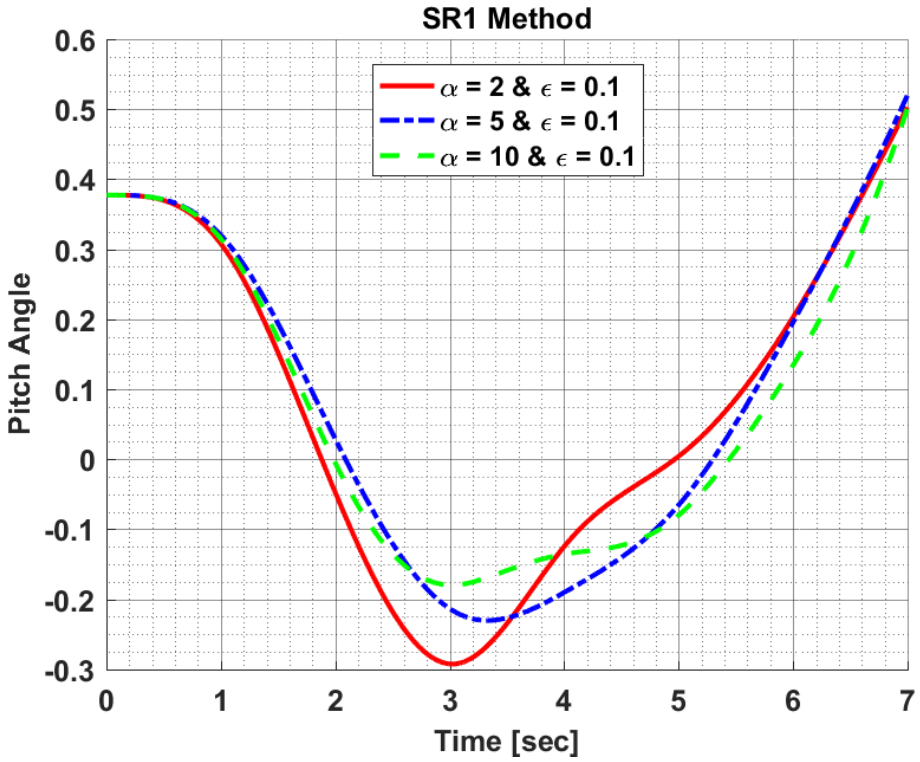
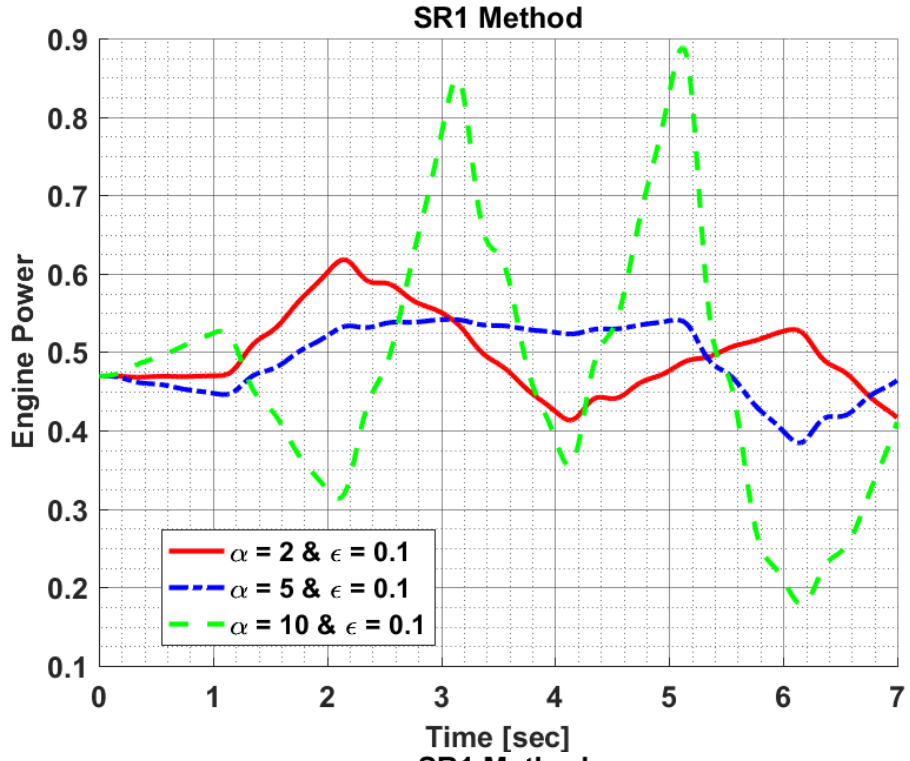


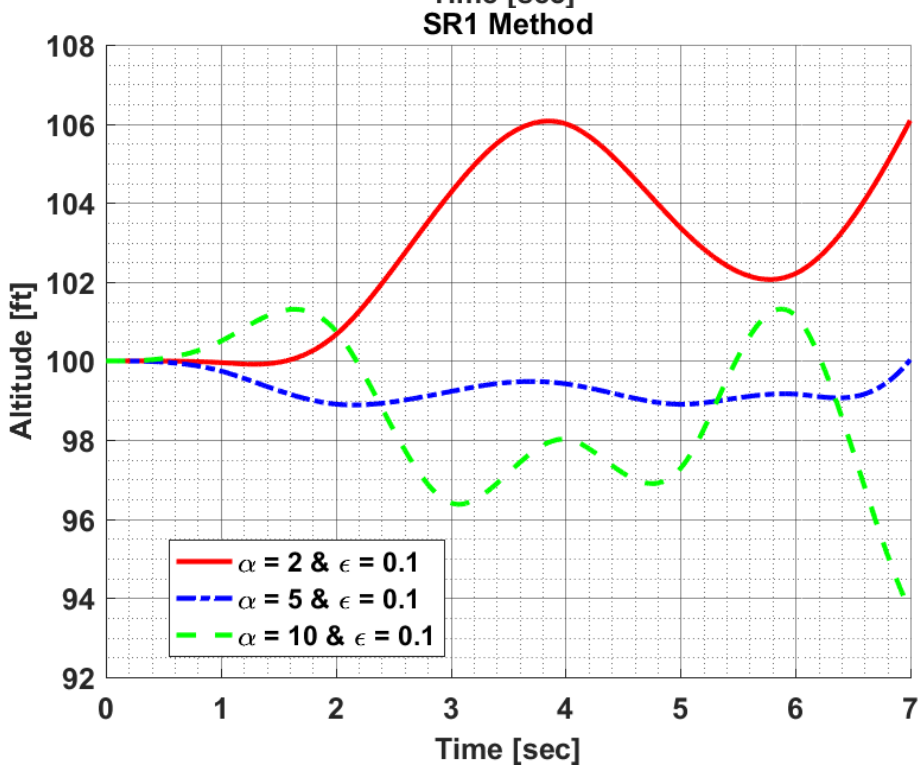
Figure 10.18: Design Variables of Hover to Forward Flight Maneuver Optimization in Steps Size Sensitivity Analysis for BM

Results of SR1 Method:






SR1 Method
Engine Power
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1


SR1 Method
Altitude [ft]
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1

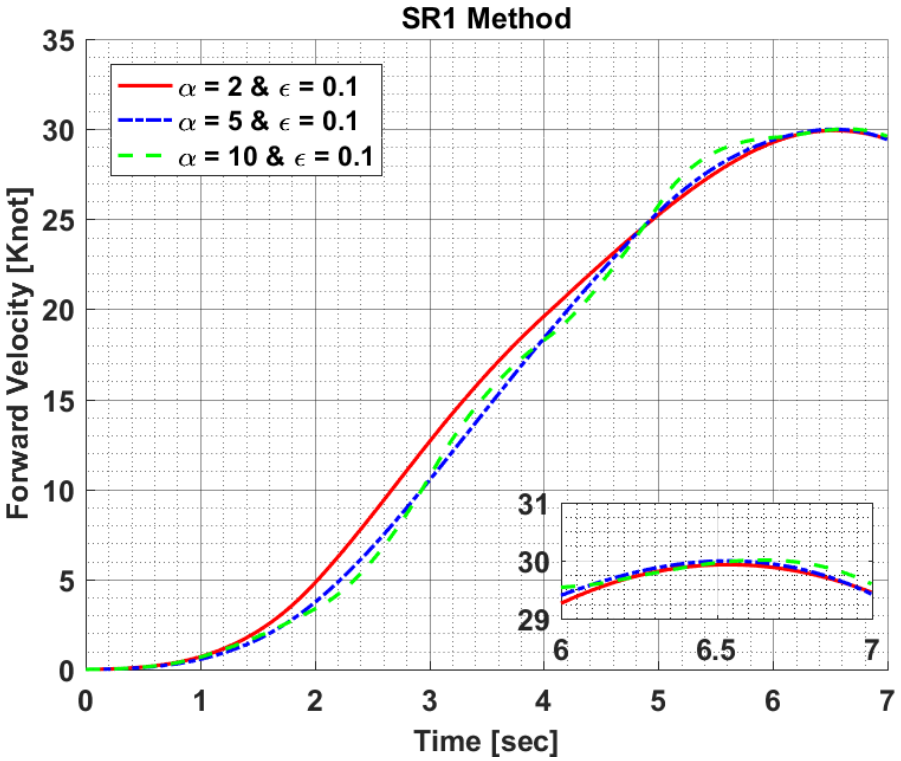


Figure 10.19: Constraint Results of Hover to Forward Flight Maneuver Optimization in Step Size Sensitivity Analysis for SR1 Method

Figure 10.19 shows the constraint results of hover to forward flight maneuver obtained in Step size comparison for SR1. Moreover, Figure 10.20 shows the required longitudinal and collective cyclic input to be able to carry out hover to forward flight maneuver as given in Figure 10.19.

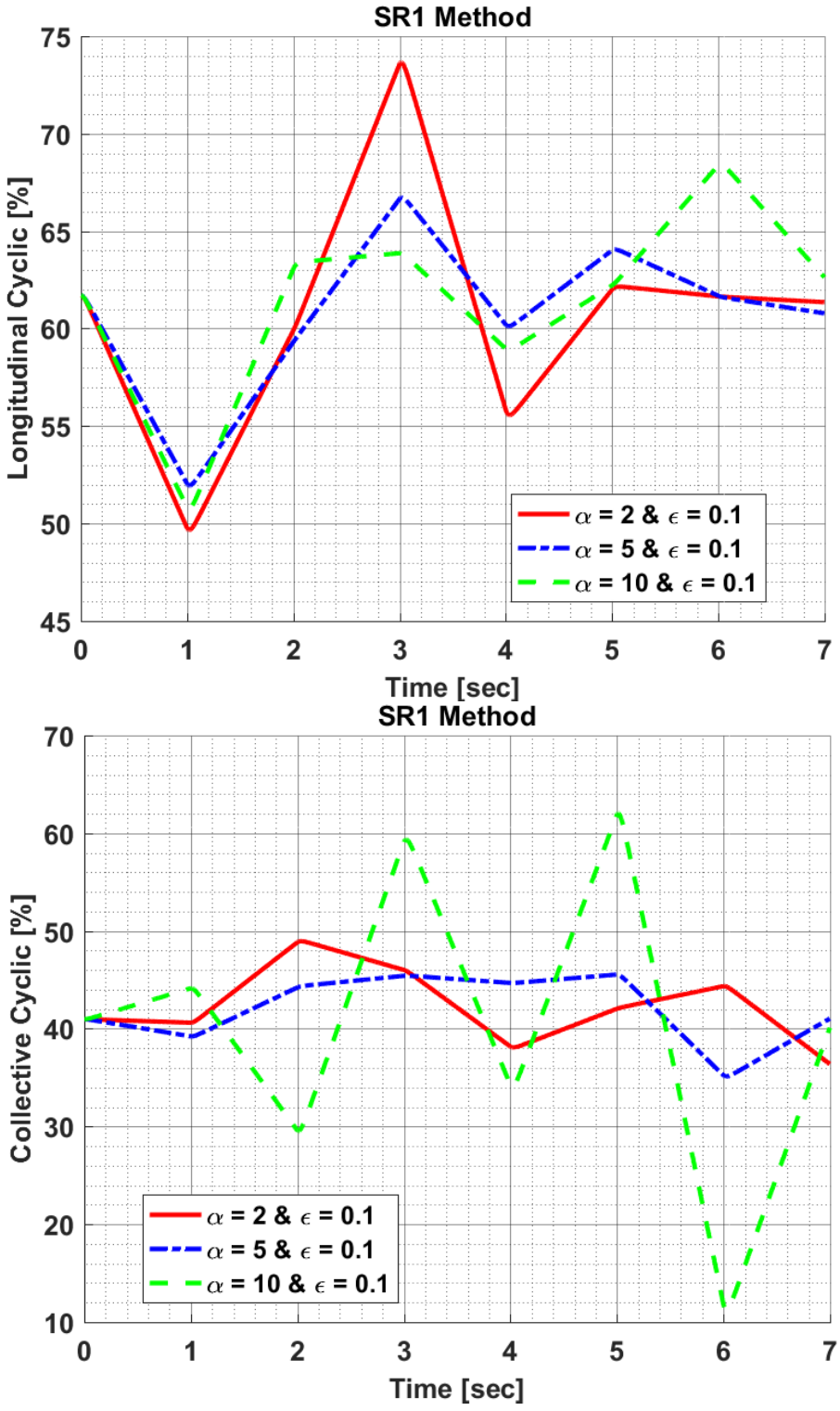
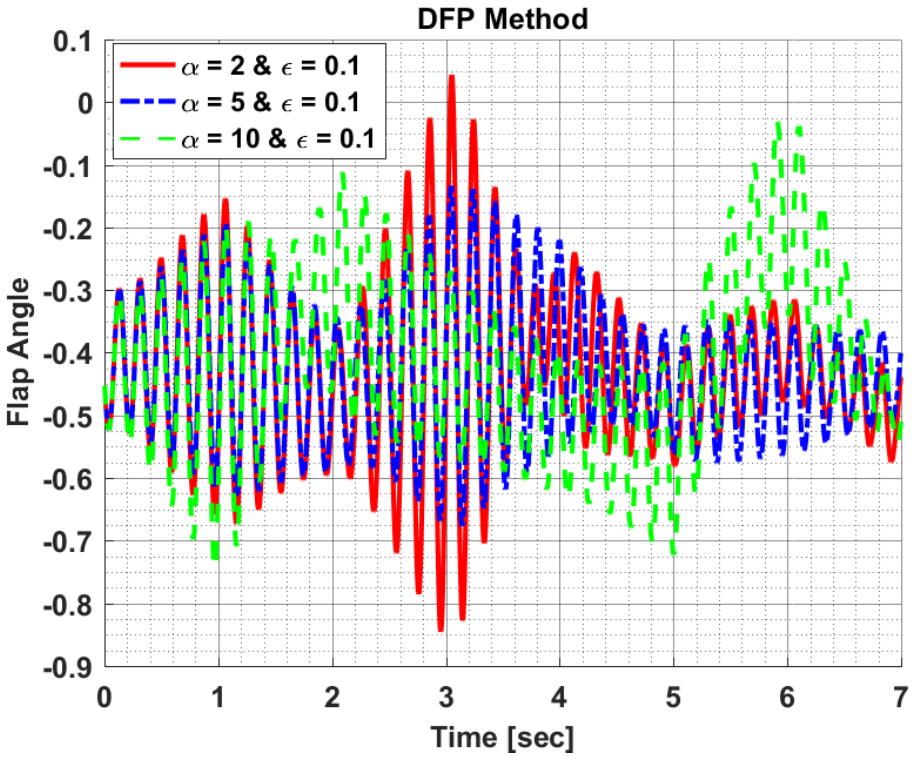
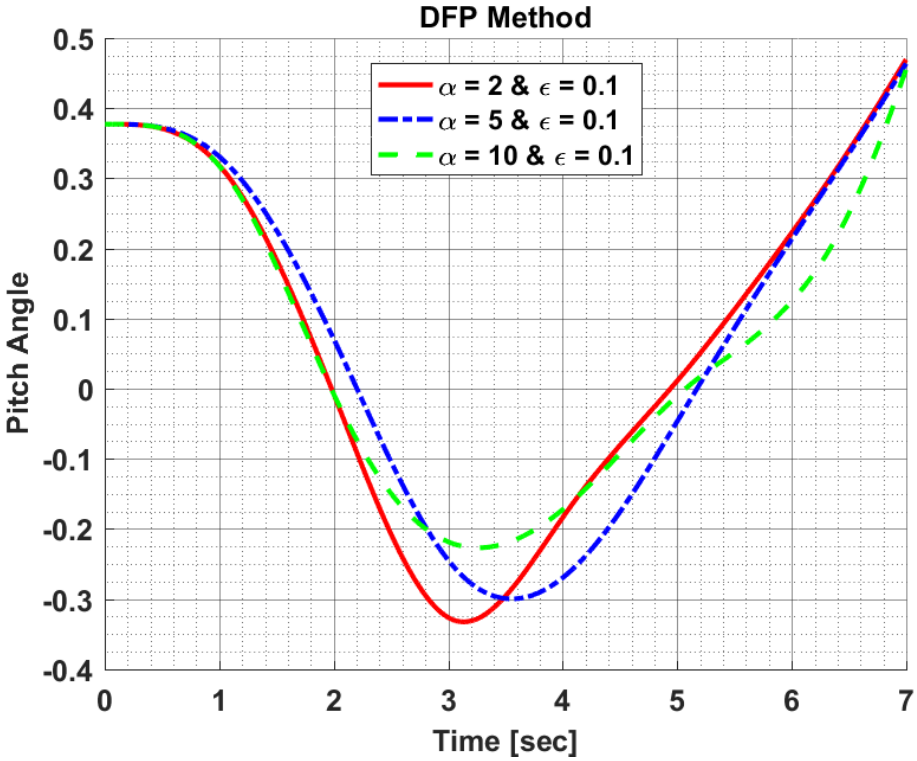
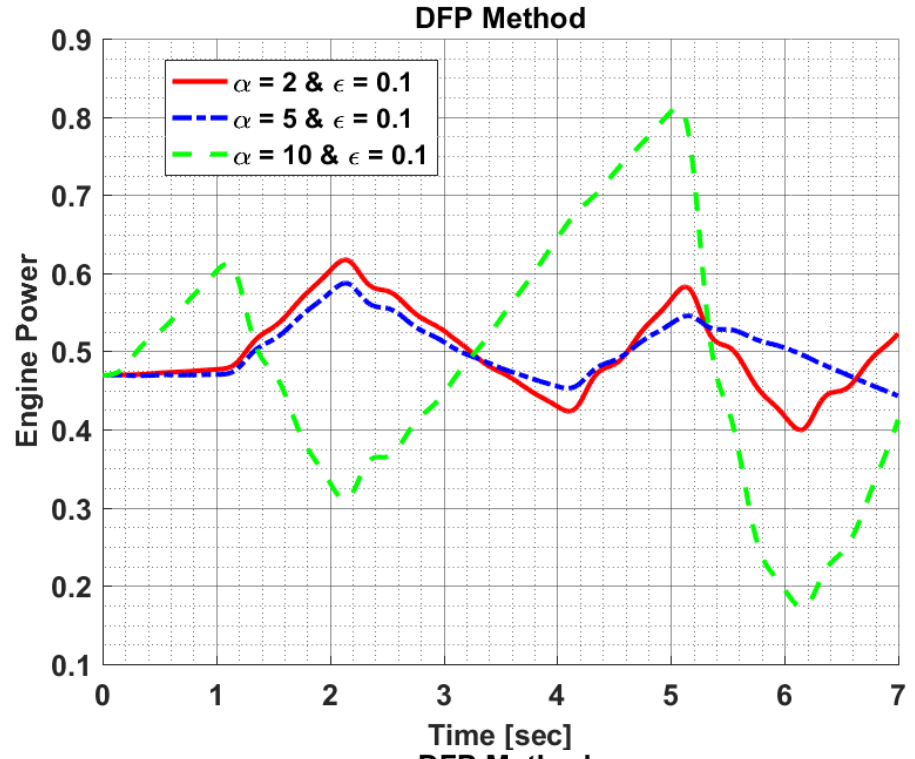


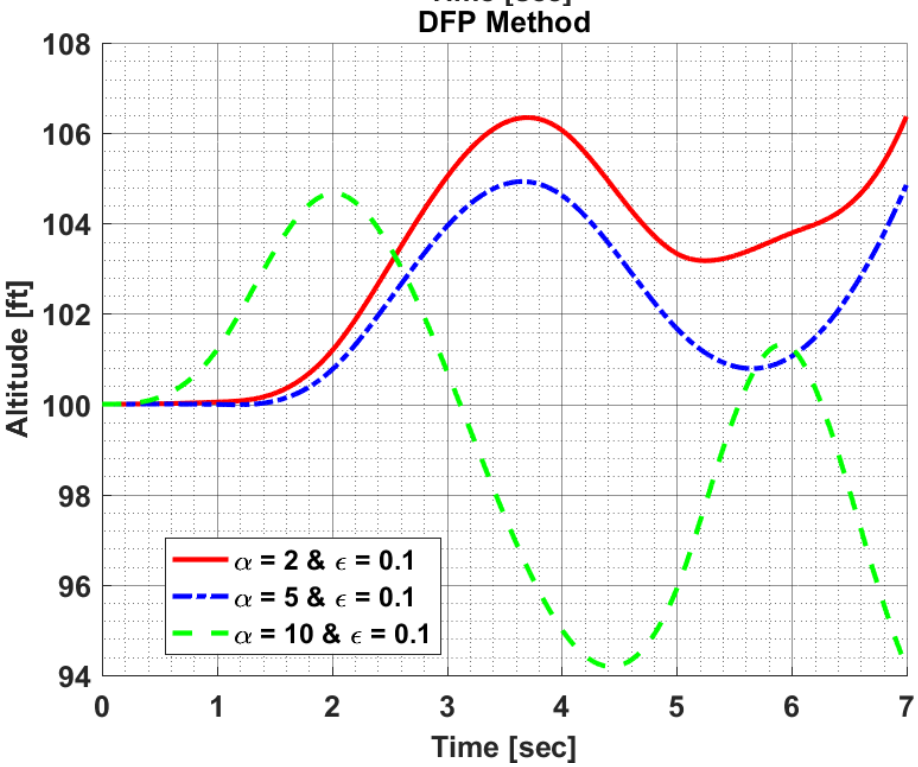
Figure 10.20: Design Variables of Hover to Forward Flight Maneuver Optimization in Step Size Sensitivity Analysis for SR1 Method

Results of DFP Method:






DFP Method
Engine Power
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
1
2
3
4
5
6
7


DFP Method
Altitude [ft]
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1
108
106
104
102
100
98
96
94
0
1
2
3
4
5
6
7

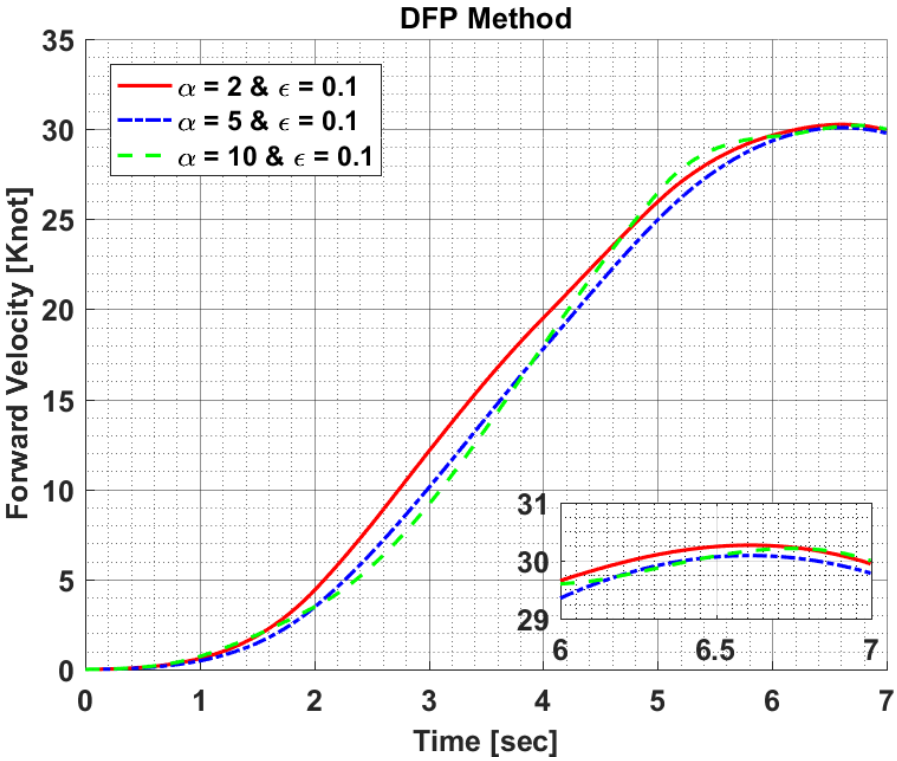


Figure 10.21: Constraint Results of Hover to Forward Flight Maneuver Optimization in Step Size Sensitivity Analysis for DFP Method

Figure 10.21 shows the constraint results of hover to forward flight maneuver obtained in Step Size comparison for DFP. Moreover, Figure 10.22 shows the required longitudinal and collective cyclic input to be able to carry out hover to forward flight maneuver as given in Figure 10.21.

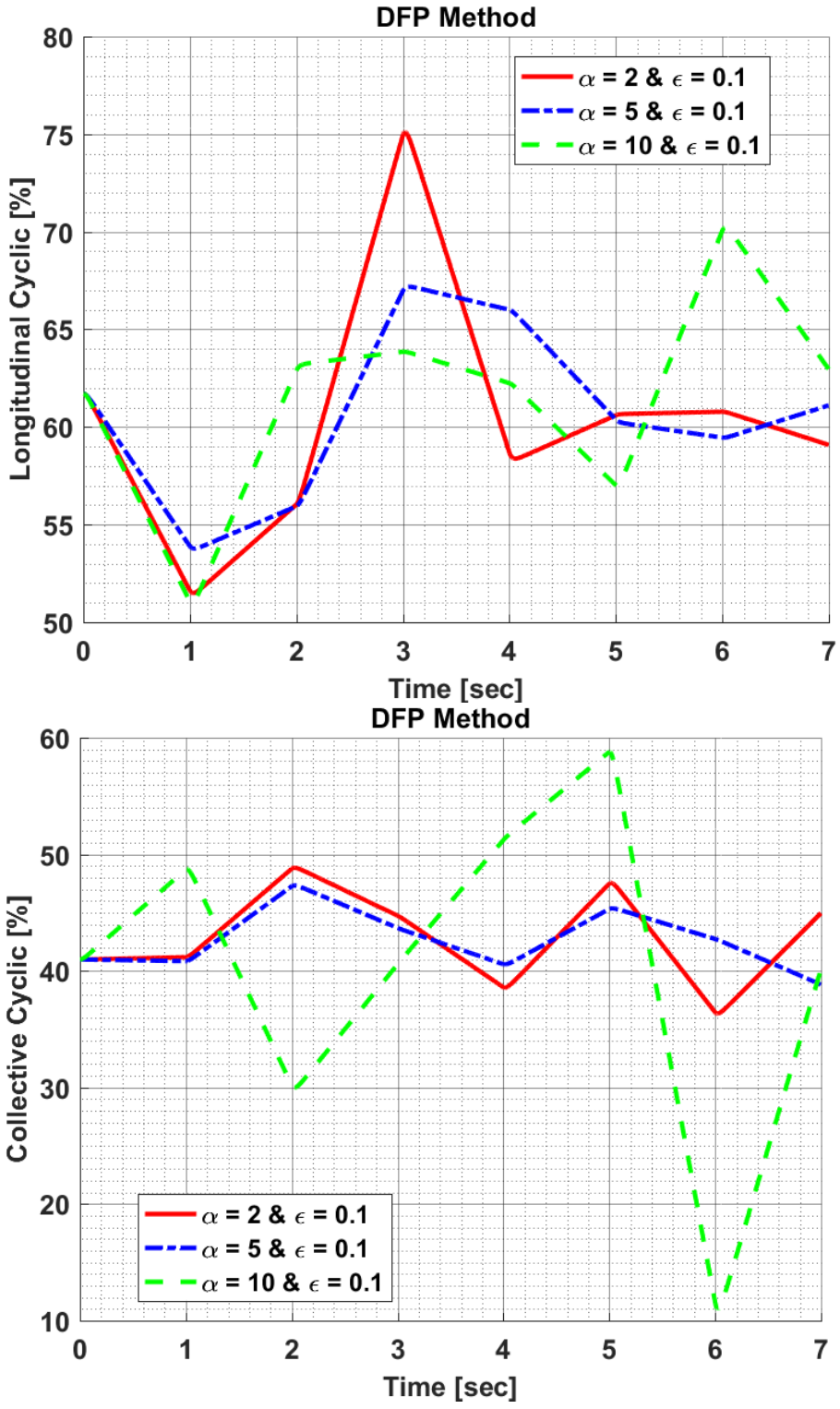
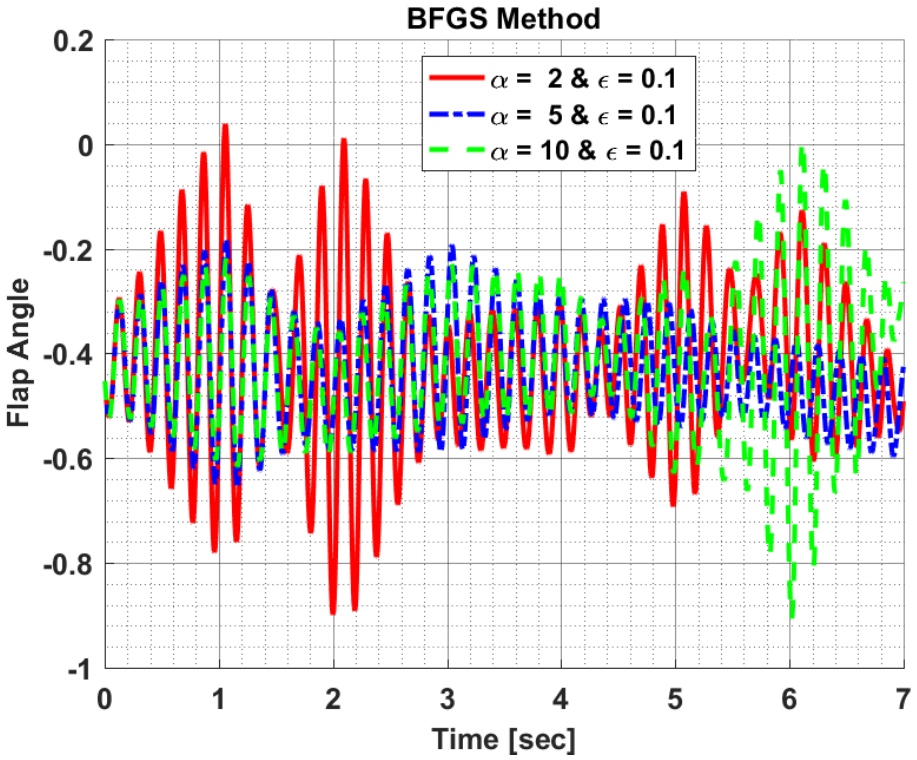
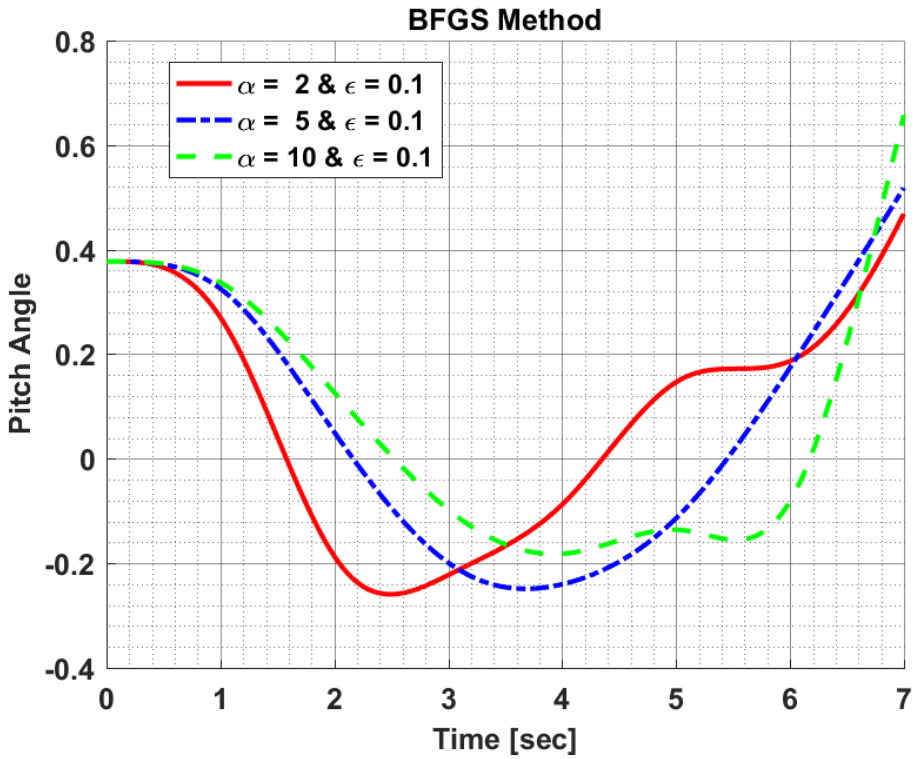
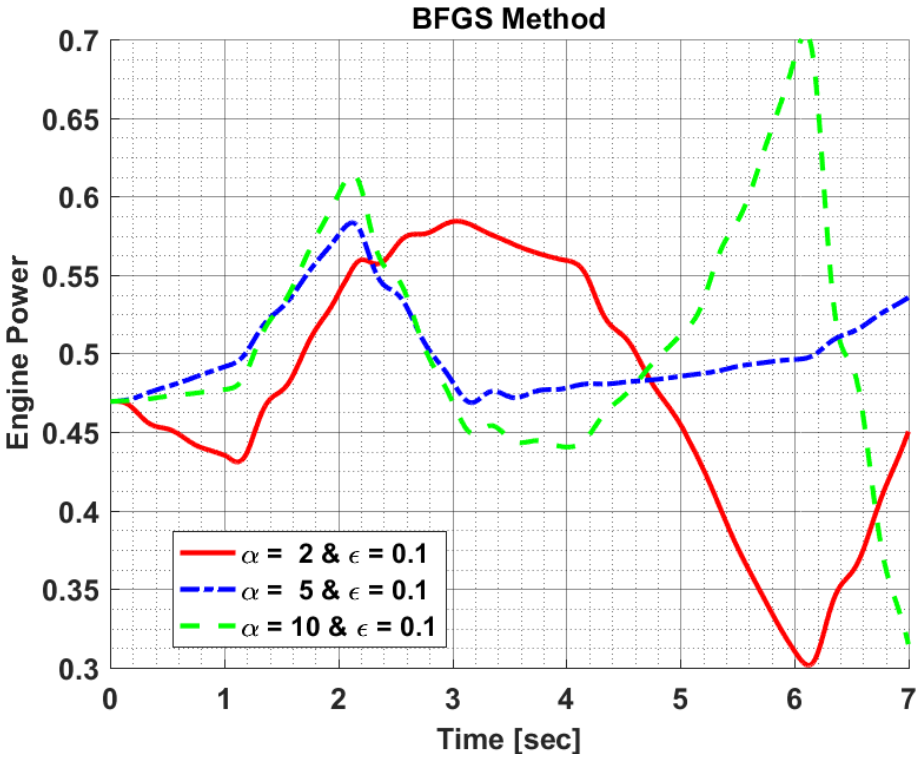


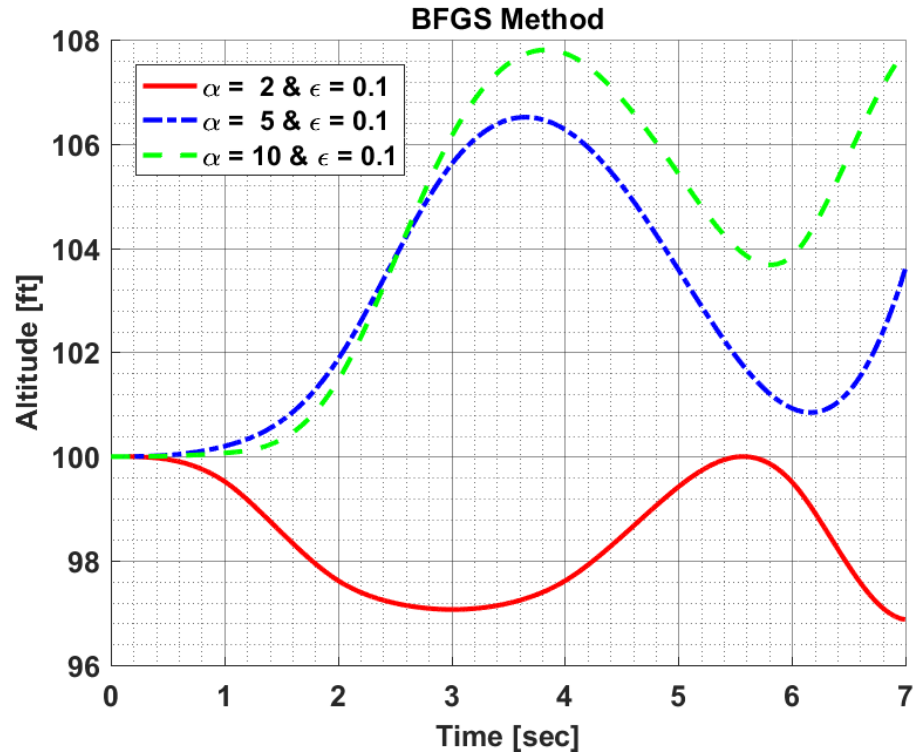
Figure 10.22: Design Variables of Hover to Forward Flight Maneuver Optimization in Step Size Sensitivity Analysis for DFP Method

Results of BFGS Method:






BFGS Method
Engine Power
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1


BFGS Method
Altitude [ft]
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1

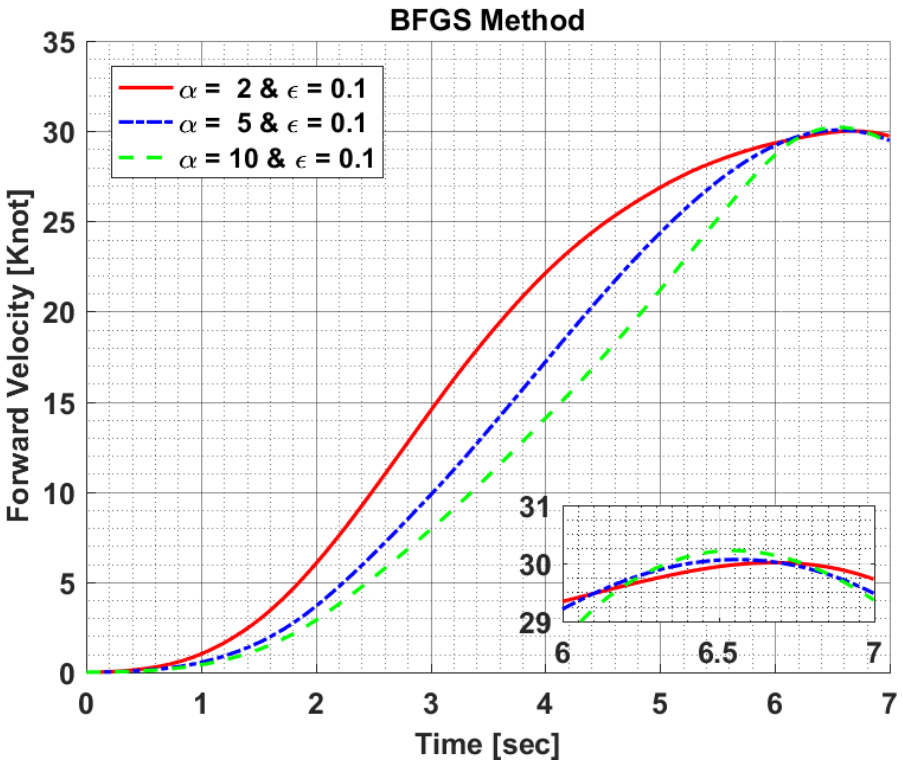


Figure 10.23: Constraint Results of Hover to Forward Flight Maneuver Optimization in Step Size Sensitivity Analysis for BFGS Method

Figure 10.23 shows the constraint results of hover to forward flight maneuver obtained in Step Size comparison for BFGS. Moreover, Figure 10.24 shows the required longitudinal and collective cyclic input to be able to carry out hover to forward flight maneuver as given in Figure 10.23.

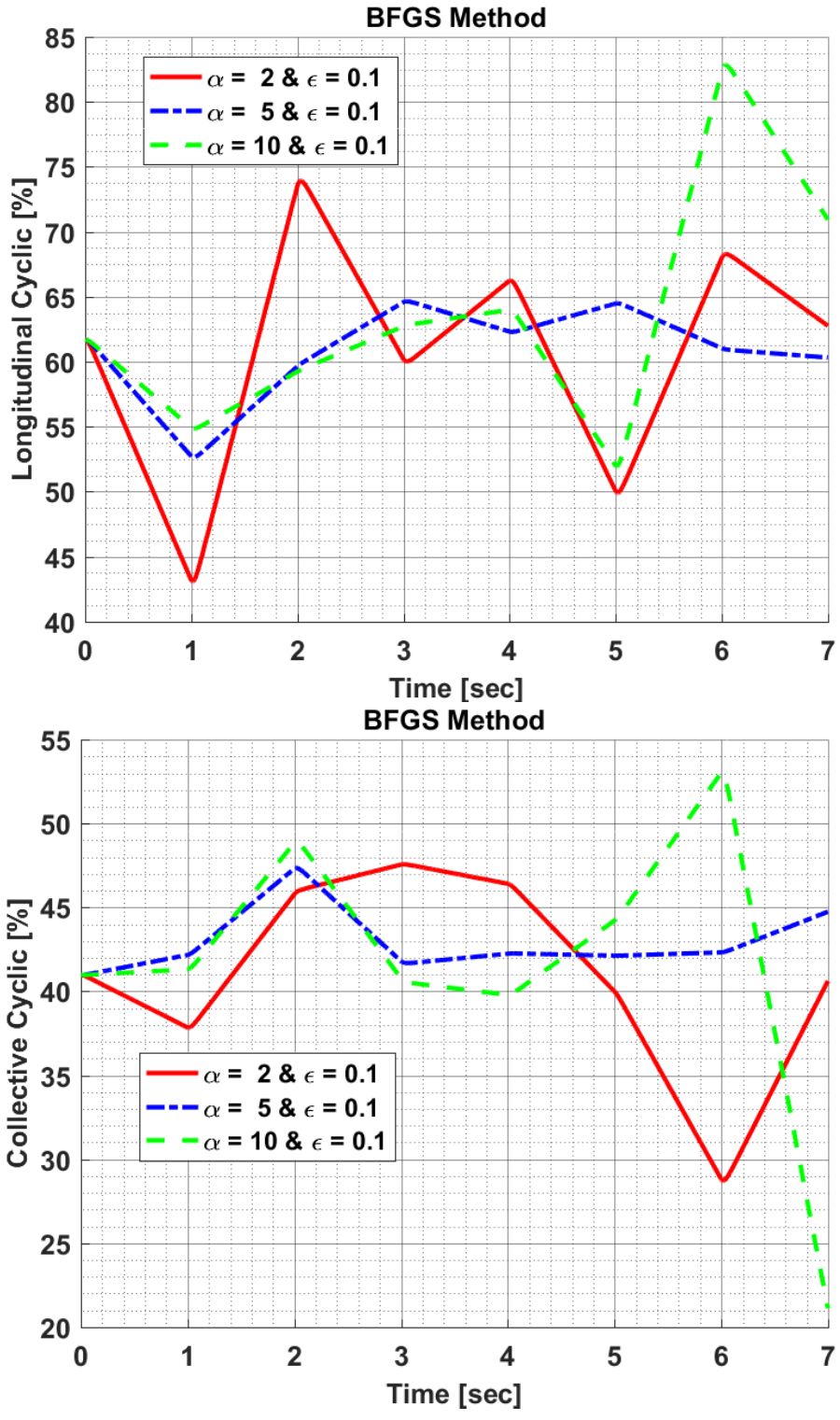
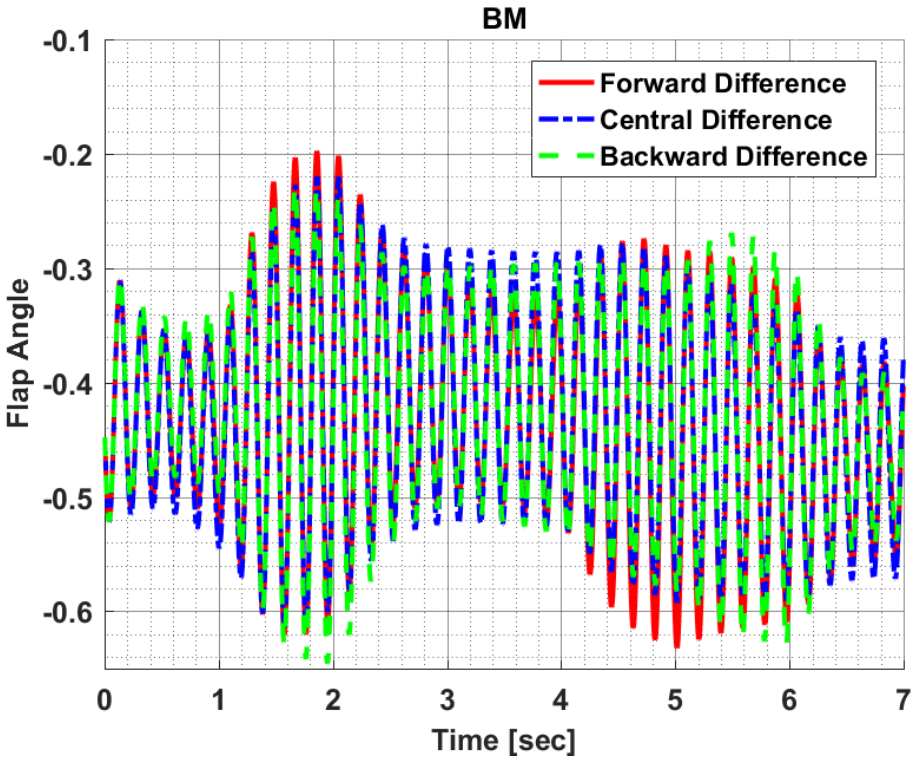


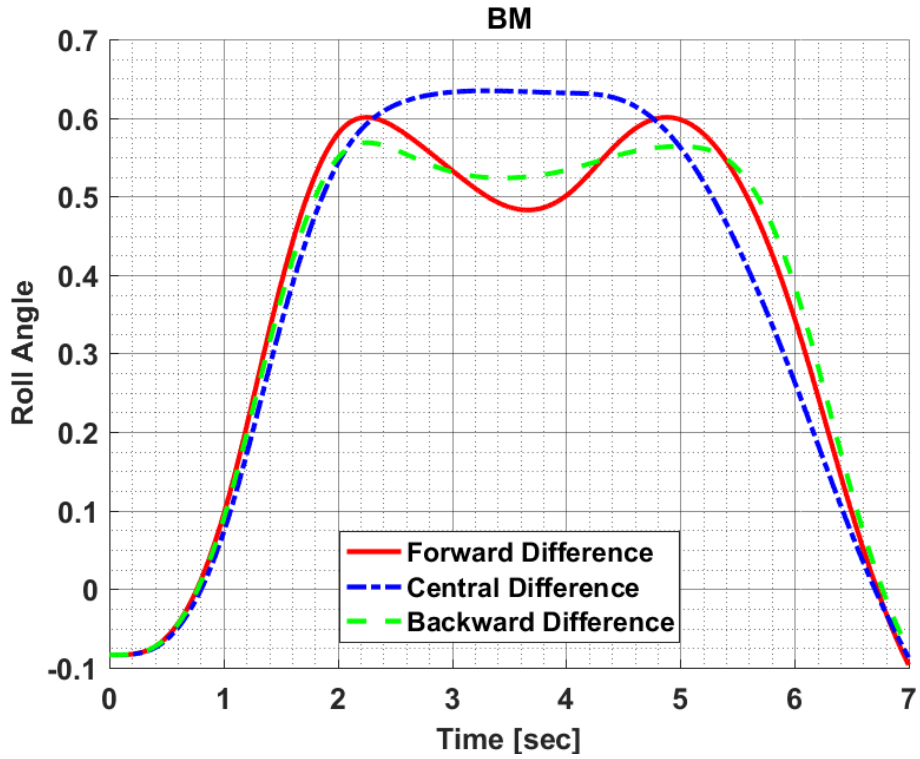
Figure 10.24: Design Variables of Hover to Forward Flight Maneuver Optimization in Step Size Sensitivity Analysis for BFGS Method

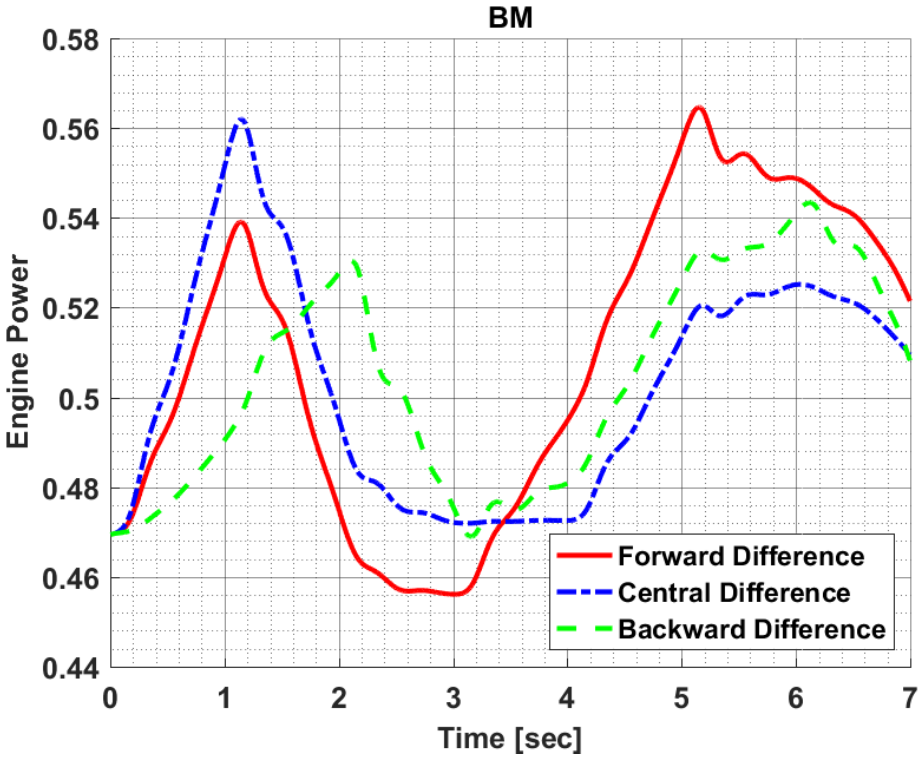
APPENDIX B1: Comparison - The Finite Divided Difference Approximations for Hover to Rightward Flight Maneuver Optimization

In this appendix section, constraint results and corresponding pilot control inputs are provided for each optimization method configuration which are executed for the comparison of the finite divided difference approximations. Moreover, they are the results of hover to rightward flight maneuver optimization.

Results of BM:




BM
Roll Angle
Time [sec]
Forward Difference
Central Difference
Backward Difference


BM
Engine Power
Time [sec]
Forward Difference
Central Difference
Backward Difference

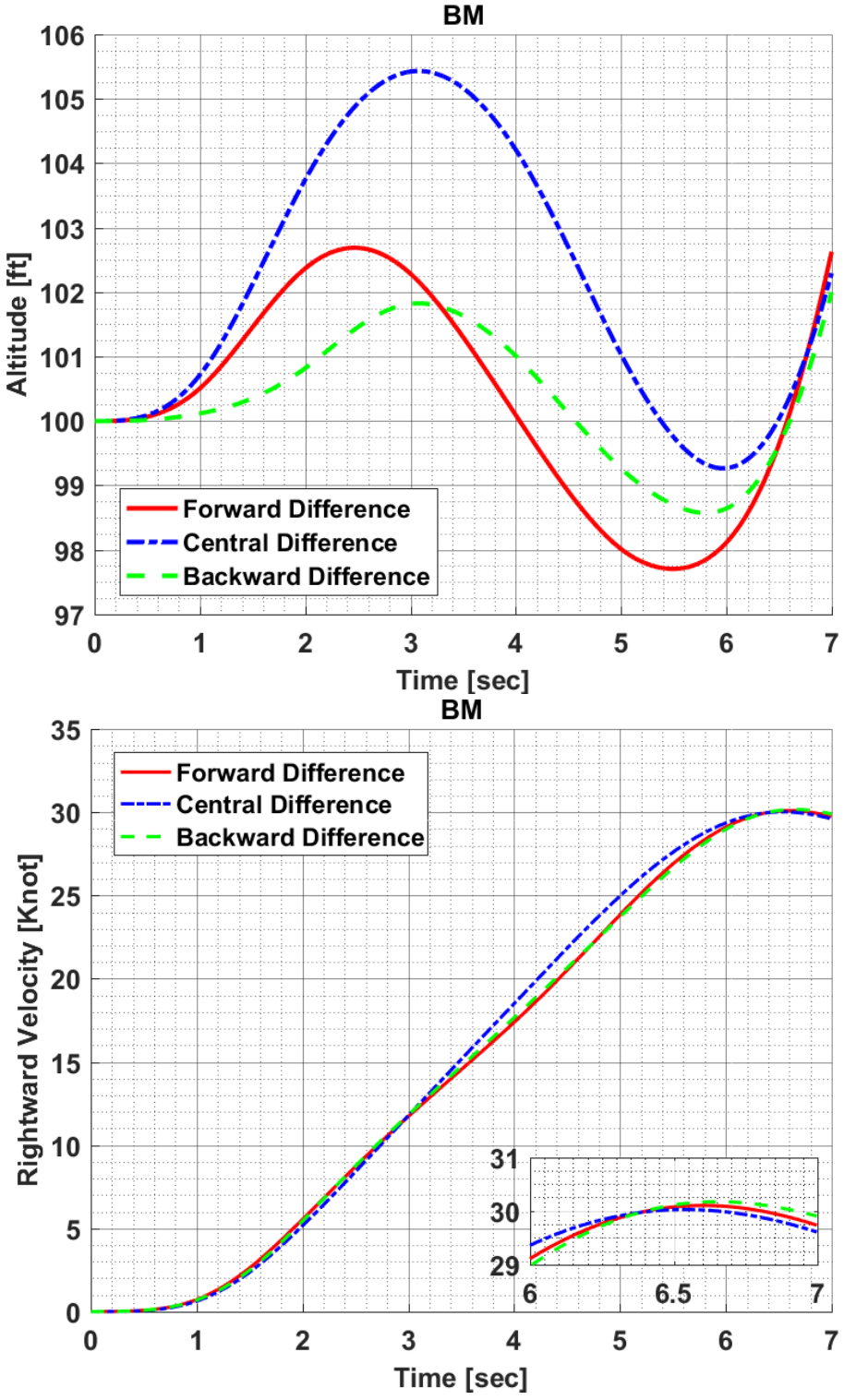


Figure 10.25: Constraint Results of Hover to Forward Flight Maneuver Optimization in the Finite Divided Difference Approximation Comparison for BM

Figure 10.25 shows the constraint results of hover to rightward flight maneuver obtained in comparing the finite divided difference approximation for BM. Moreover, Figure 10.26 shows the required longitudinal and collective cyclic input to be able to carry out hover to rightward flight maneuver as given in Figure 10.25.

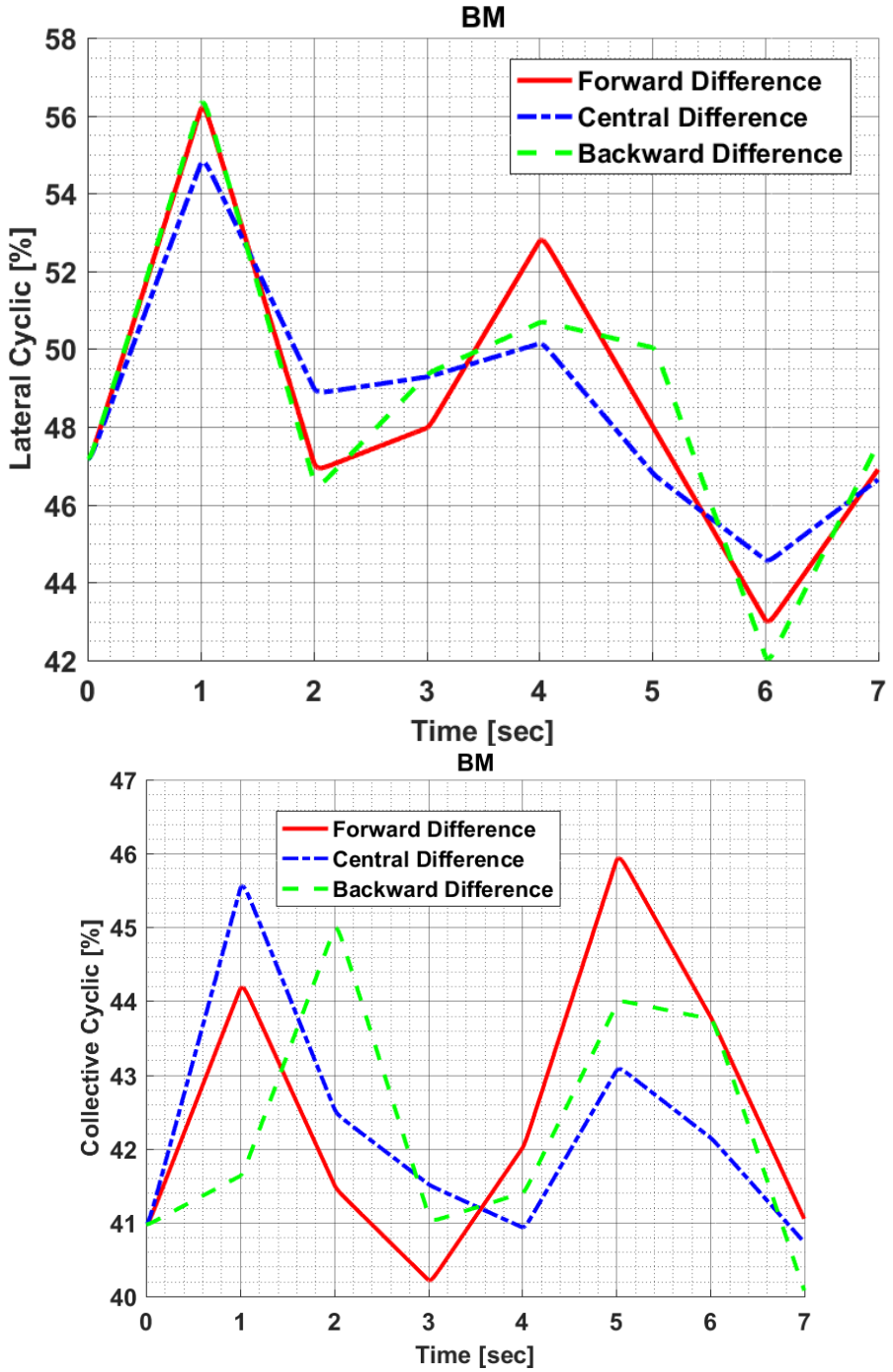
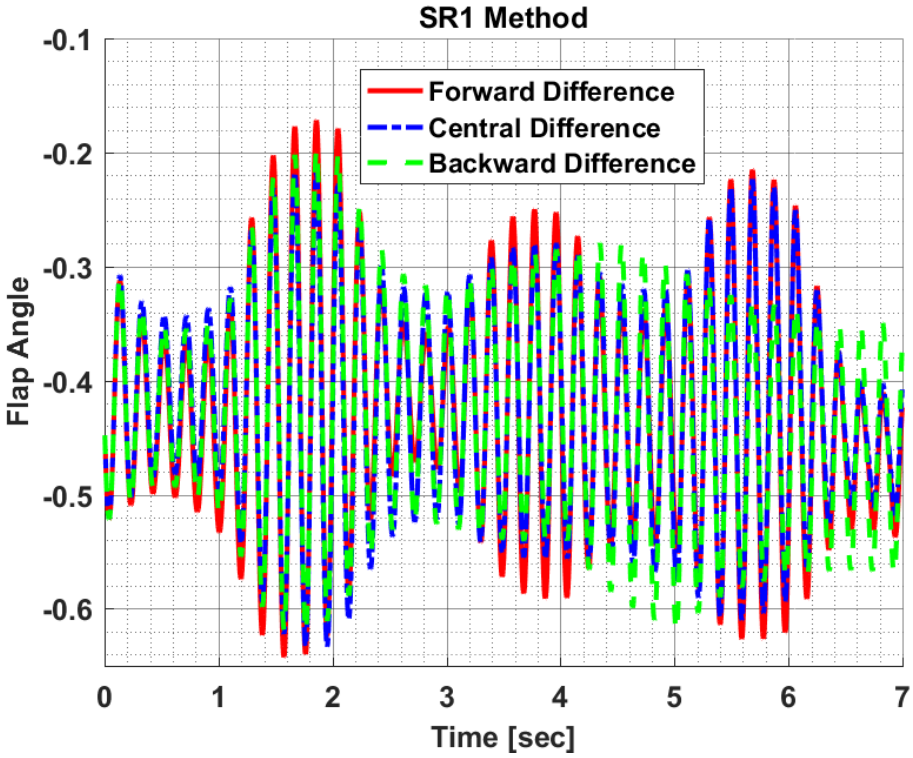
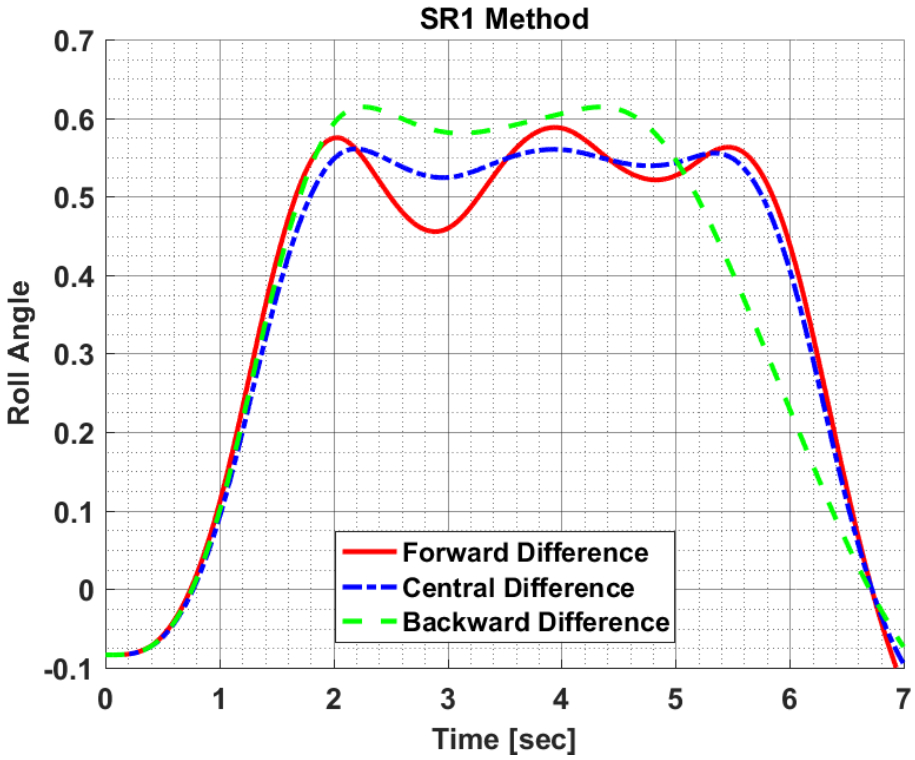
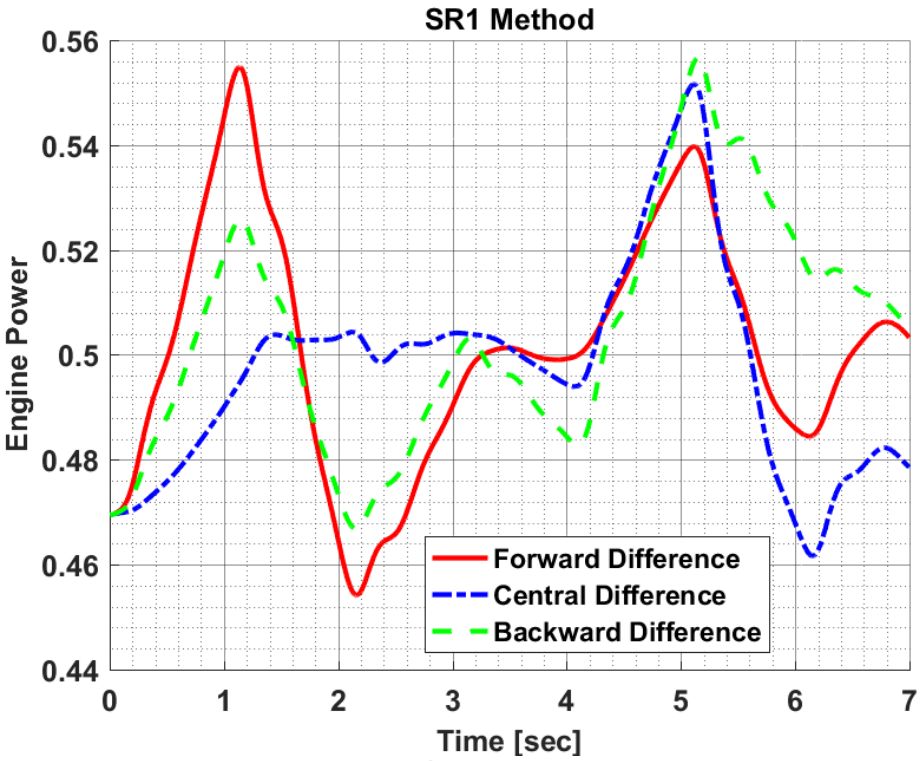


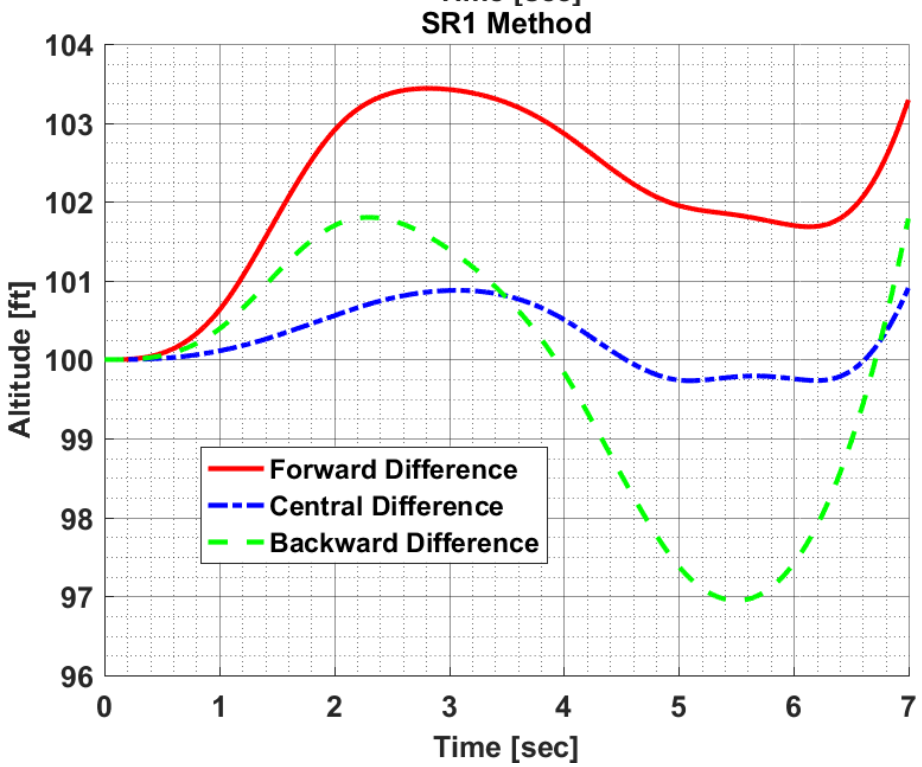
Figure 10.26: Design Variables of Hover to Rightward Flight Maneuver Optimization in the Finite Divided Difference Approximation Comparison for BM

Results of SR1 Method:






SR1 Method
Engine Power
Time [sec]
Forward Difference
Central Difference
Backward Difference


SR1 Method
Altitude [ft]
Time [sec]
Forward Difference
Central Difference
Backward Difference

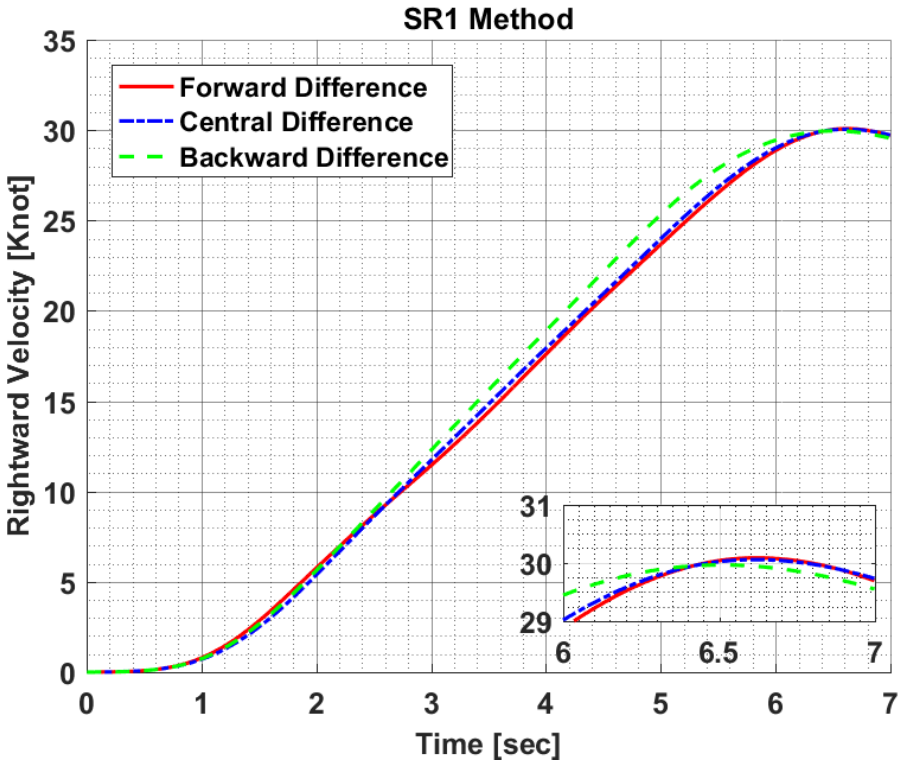


Figure 10.27: Constraint Results of Hover to Rightward Flight Maneuver Optimization in the Finite Divided Difference Approximation Comparison for SR1

Figure 10.27 shows the constraint results of hover to rightward flight maneuver obtained in comparing the finite divided difference approximation for SR1 method. Moreover, Figure 10.28 shows the required lateral and collective cyclic input to be able to carry out hover to rightward flight maneuver as given in Figure 10.27.

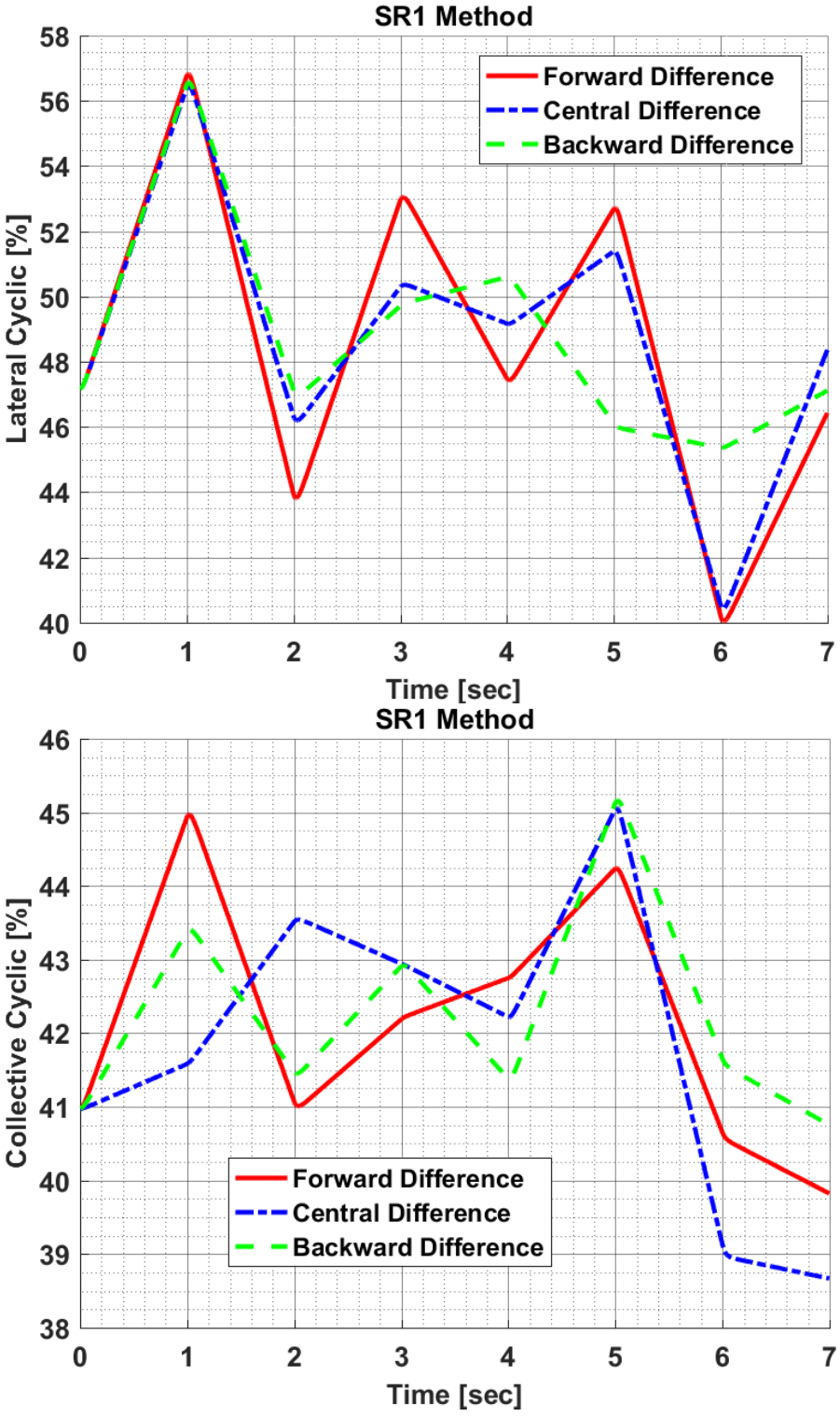
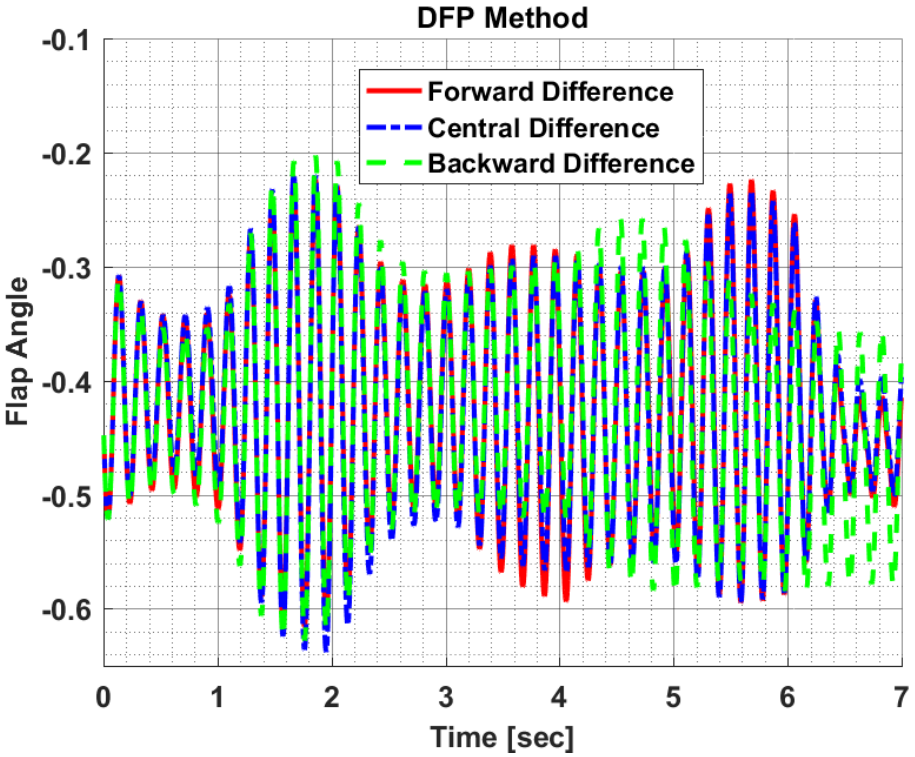
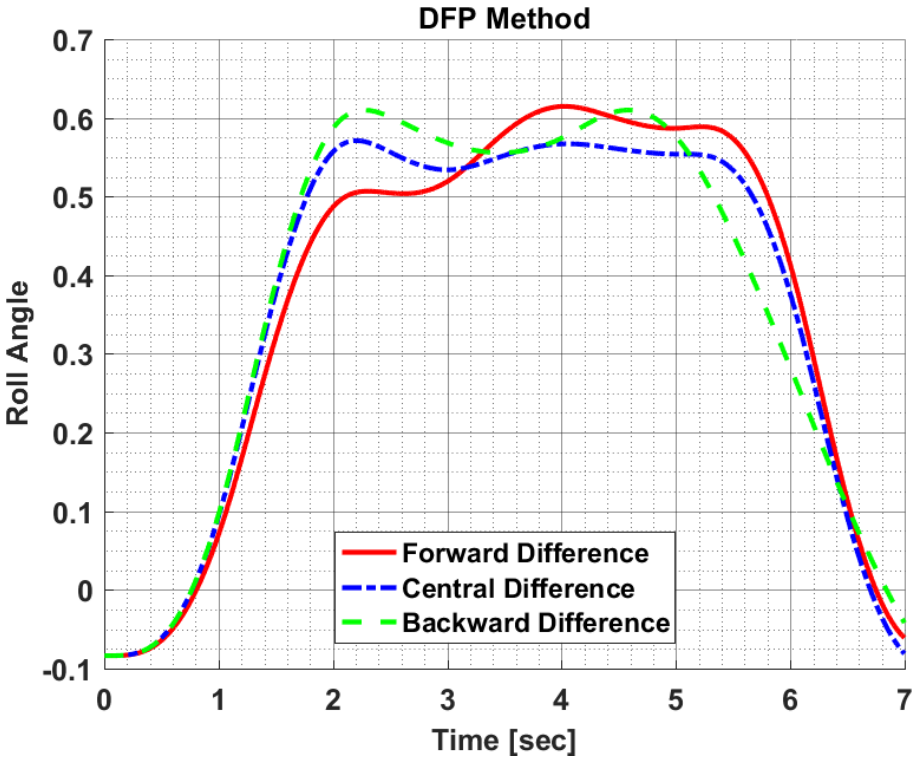
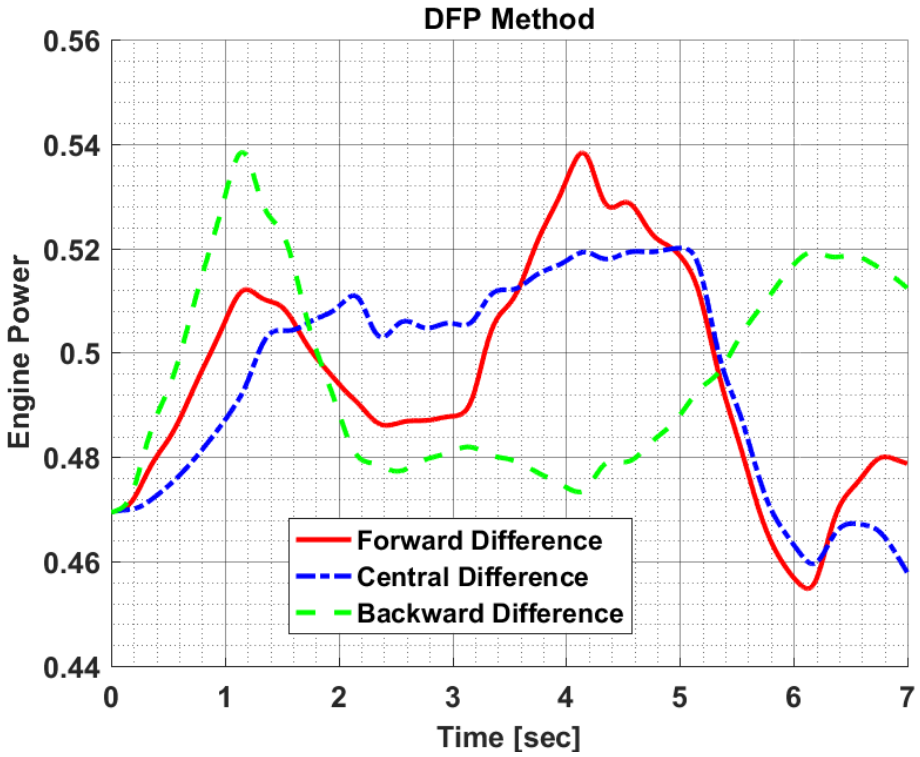


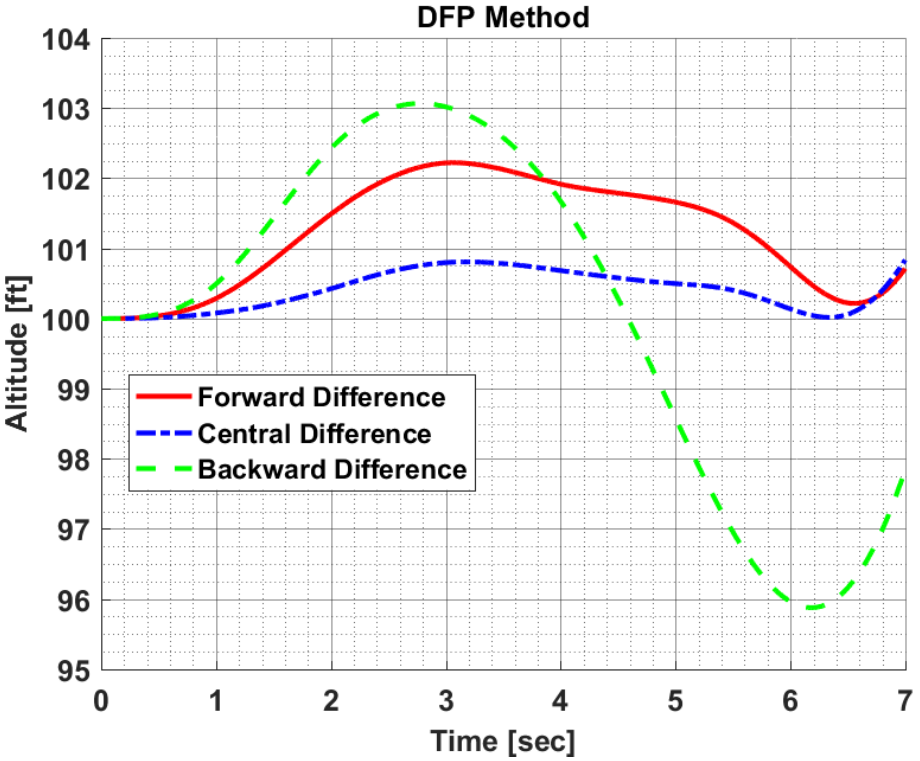
Figure 10.28: Design Variables of Hover to Rightward Flight Maneuver Optimization in the Finite Divided Difference Approximation Comparison for SR1

Results of DFP Method:






DFP Method
Engine Power
Time [sec]
Forward Difference
Central Difference
Backward Difference


DFP Method
Altitude [ft]
Time [sec]
Forward Difference
Central Difference
Backward Difference

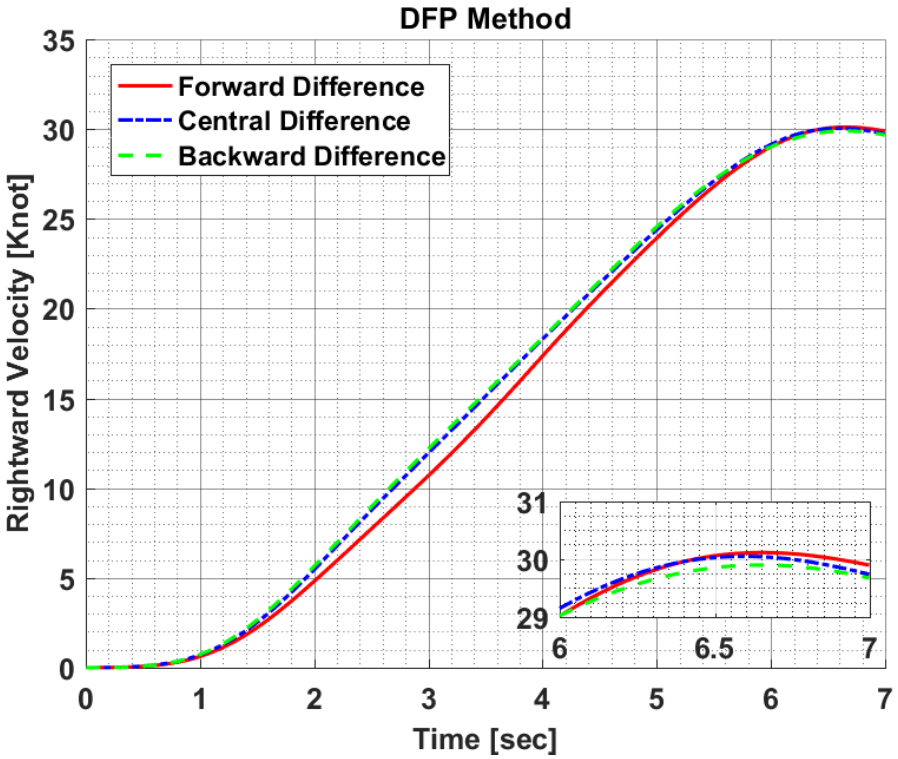


Figure 10.29: Constraint Results of Hover to Rightward Flight Maneuver Optimization in the Finite Divided Difference Approximation Comparison for DFP

Figure 10.29 shows the constraint results of hover to rightward flight maneuver obtained in comparing the finite divided difference approximation for DFP method. Moreover, Figure 10.30 shows the required lateral and collective cyclic input to be able to carry out hover to rightward flight maneuver as given in Figure 10.29.

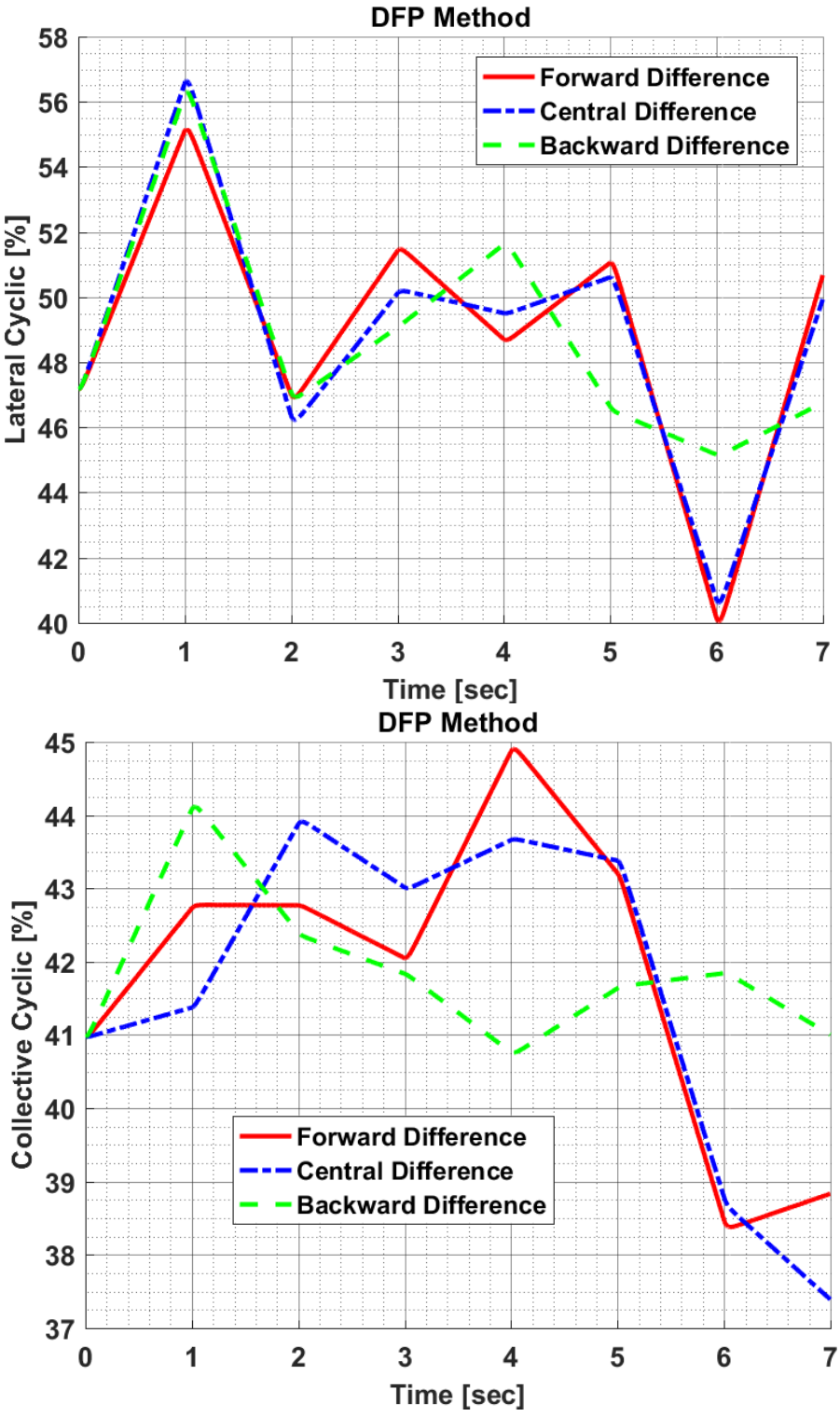
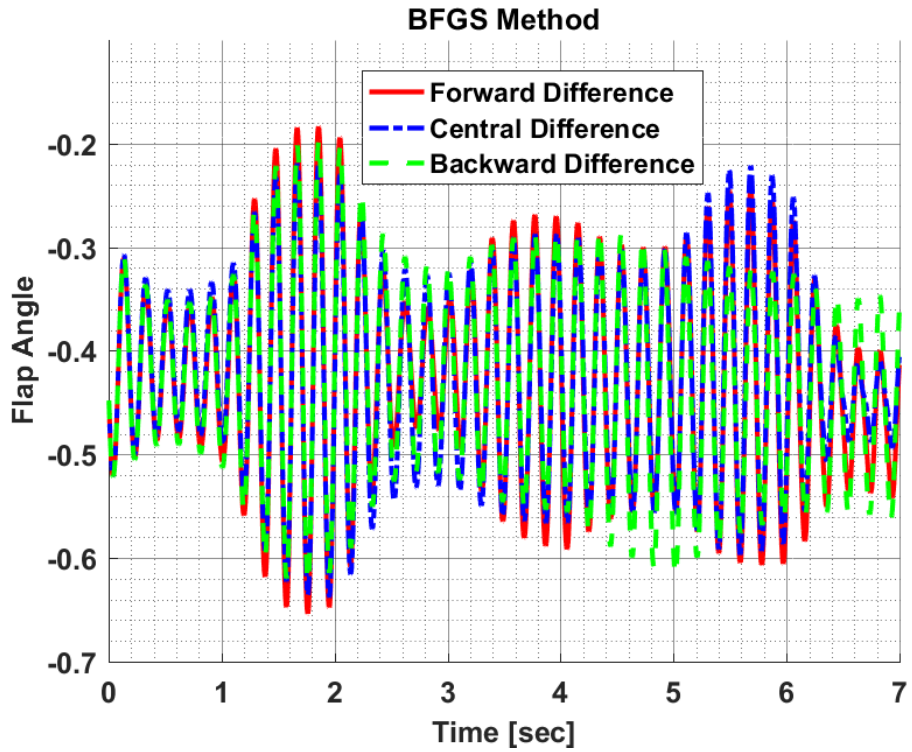
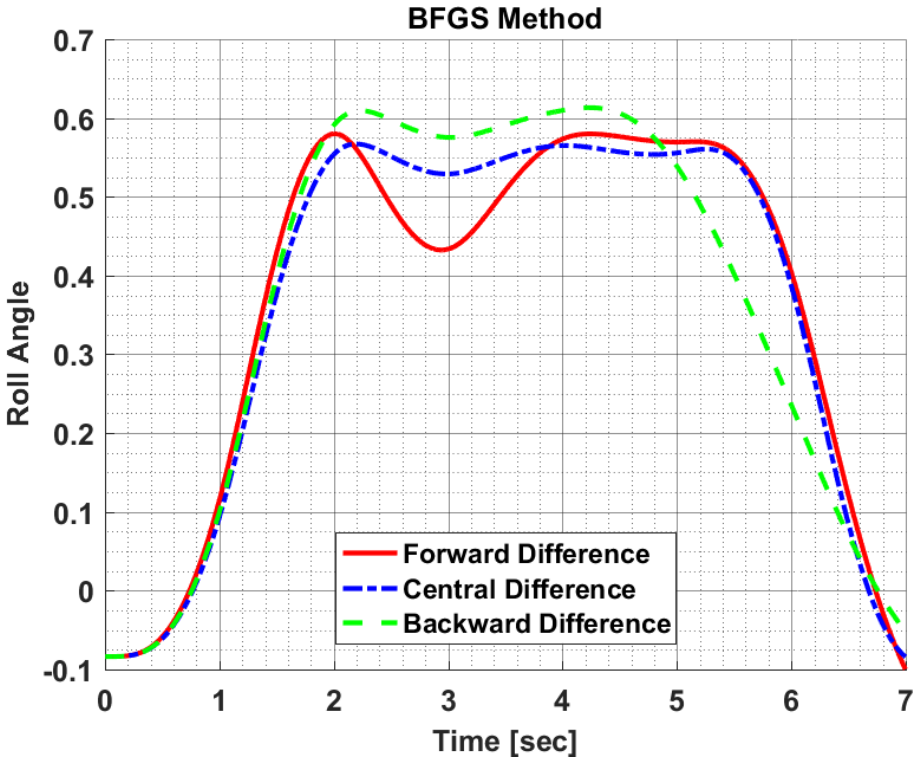
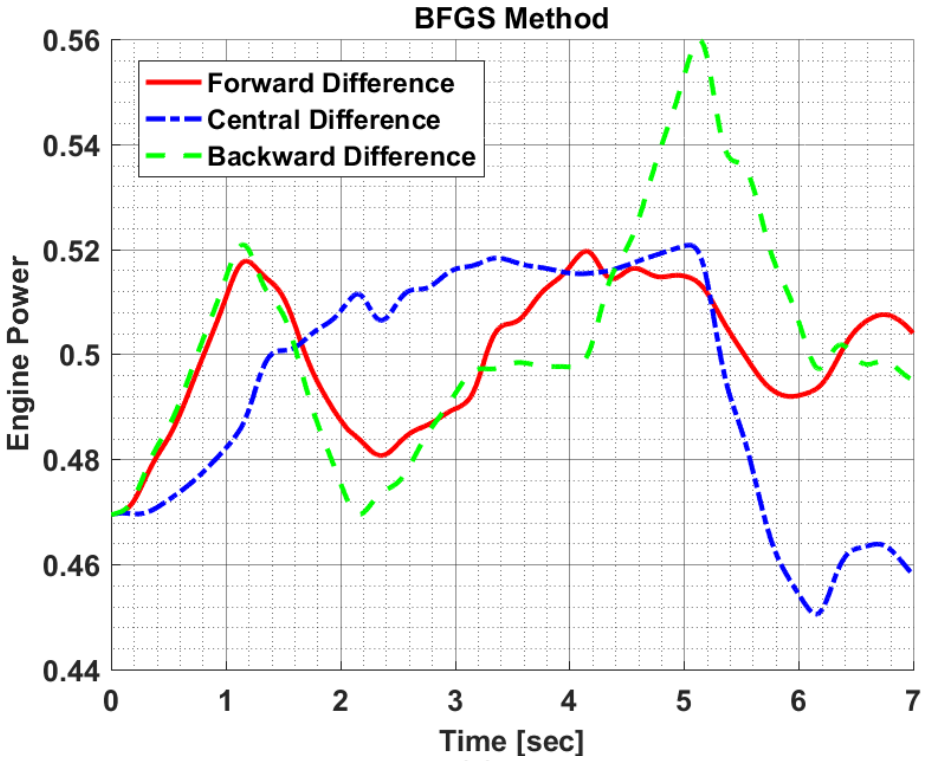


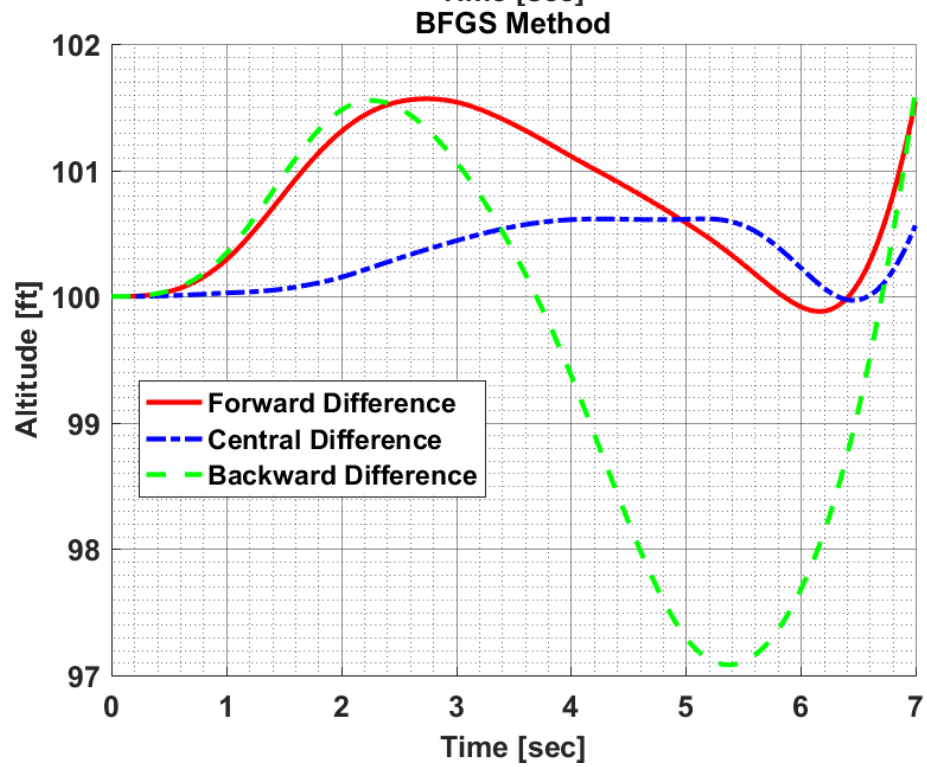
Figure 10.30: Design Variables of Hover to Rightward Flight Maneuver Optimization in the Finite Divided Difference Approximation Comparison for DFP

Results of BFGS Method:






BFGS Method
Forward Difference
Central Difference
Backward Difference
Engine Power
0.56
0.54
0.52
0.5
0.48
0.46
0.44
0
1
2
3
4
5
6
7
Time [sec]


BFGS Method
Forward Difference
Central Difference
Backward Difference
Altitude [ft]
102
101
100
99
98
97
0
1
2
3
4
5
6
7
Time [sec]

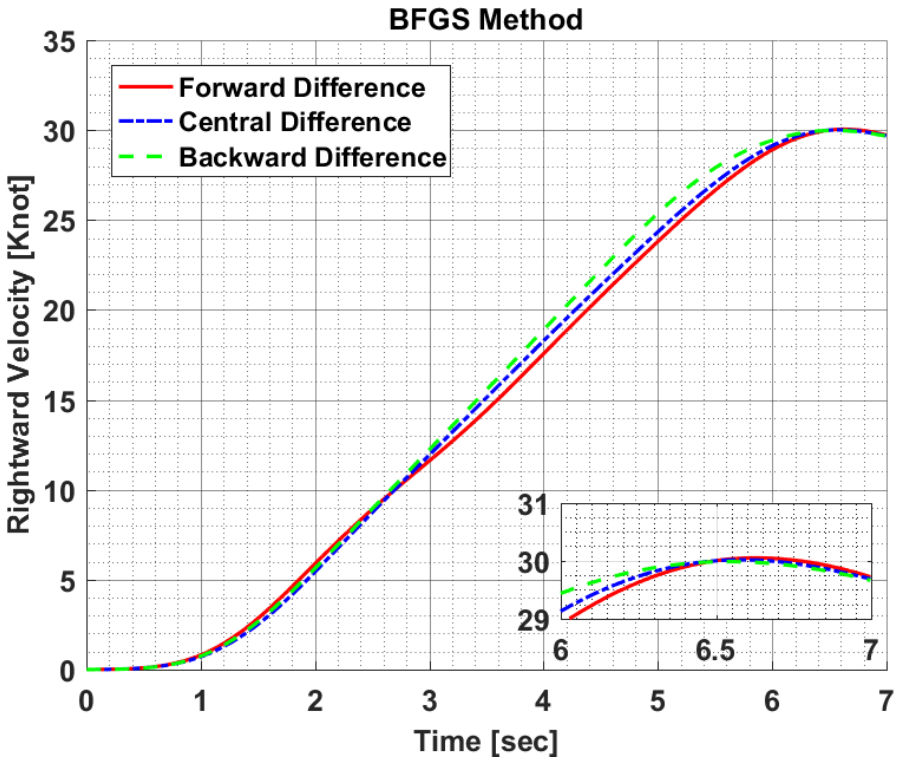


Figure 10.31: Constraint Results of Hover to Rightward Flight Maneuver Optimization in Finite Divided Difference Approximation Comparison for BFGS

Figure 10.31 shows the constraint results of hover to rightward flight maneuver obtained in comparing the finite divided difference approximation for BFGS method. Moreover, Figure 10.32 shows the required lateral and collective cyclic input to be able to carry out hover to rightward flight maneuver as given in Figure 10.31.

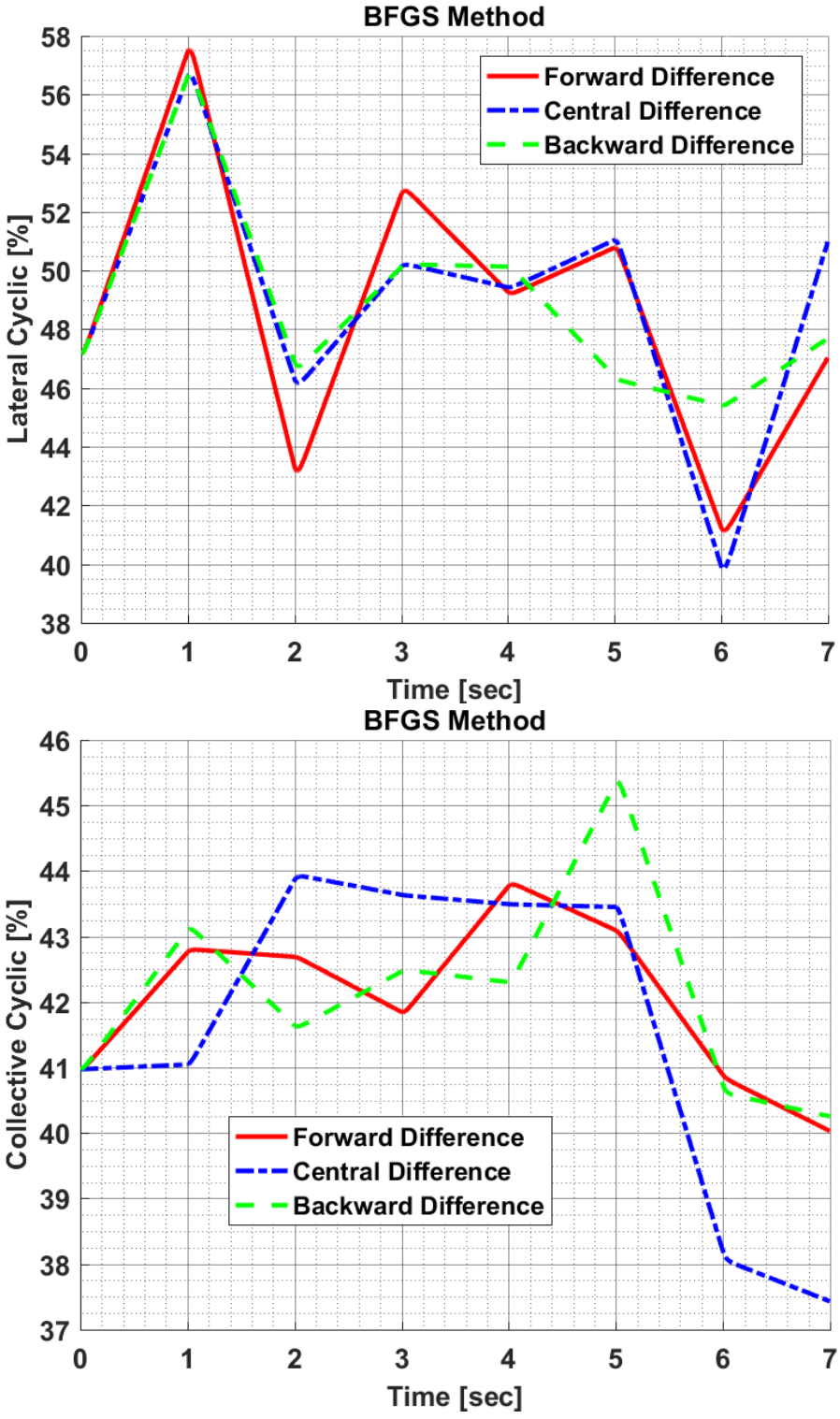


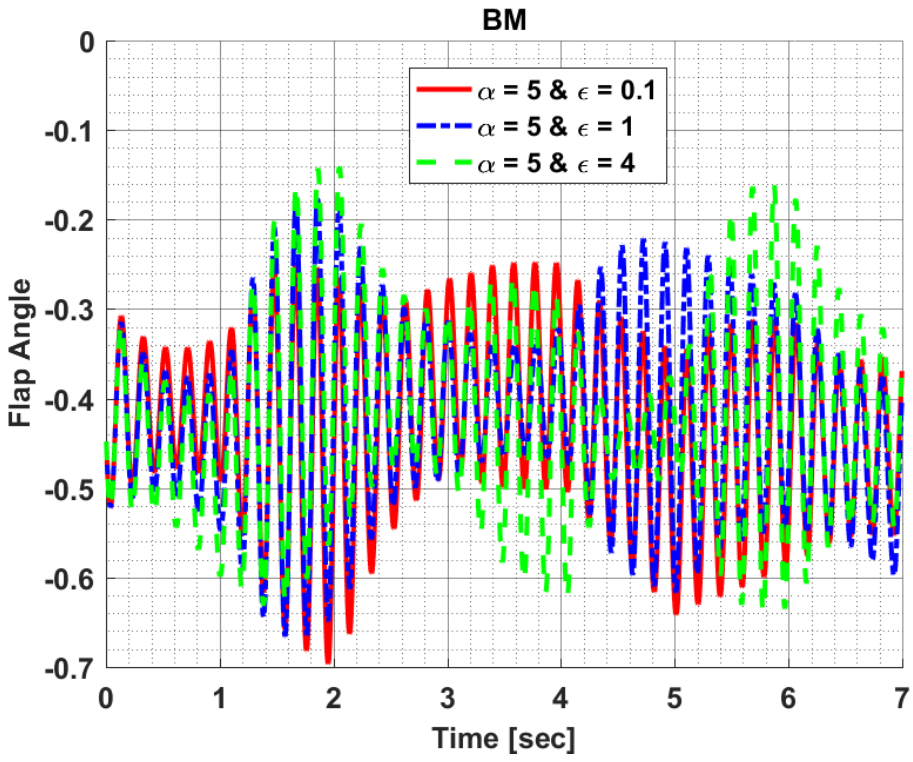
Figure 10.32: Design Variables of Hover to Rightward Flight Maneuver Optimization in Finite Divided Difference Approximation Comparison for BFGS

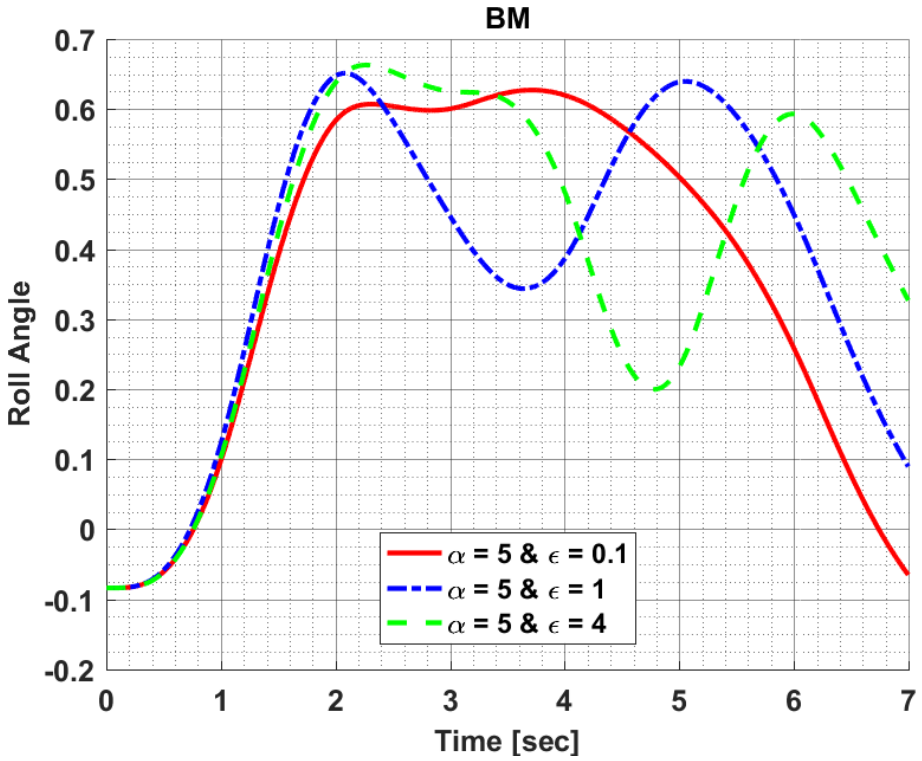
APPENDIX B2: Comparison – Sensitivity Analysis of Perturbation Constant Parameter for Hover to Rightward Flight Maneuver Optimization

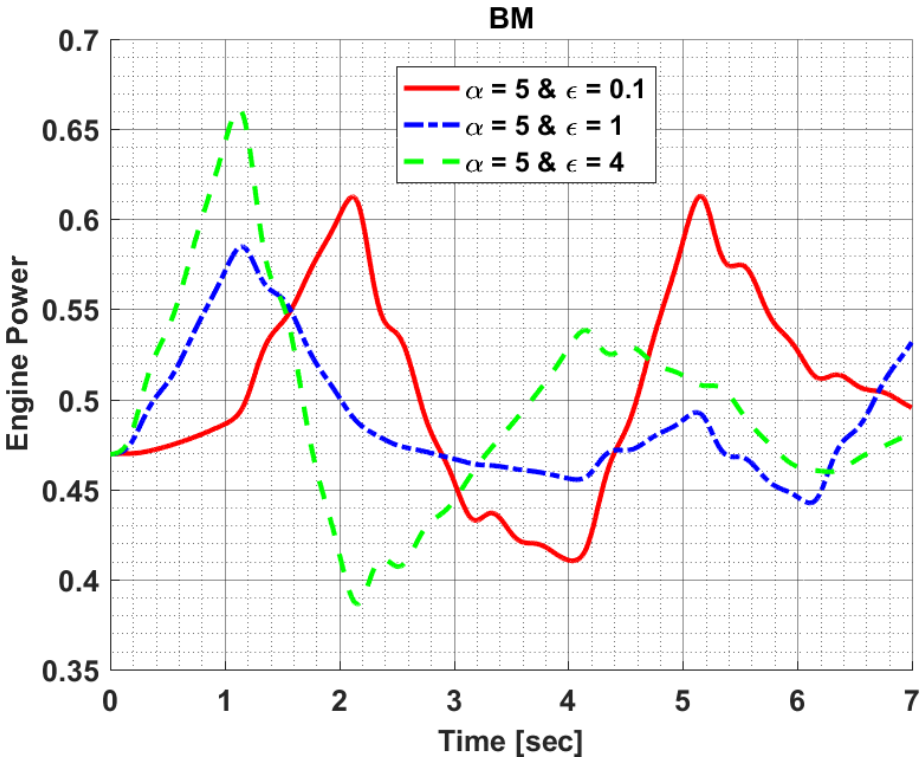
In this appendix section, constraint results and corresponding pilot control inputs are provided for each optimization method configuration which are executed for the sensitivity analysis of perturbation constant parameter. Moreover, they are the results of hover to rightward flight maneuver optimization.

Results of BM:

Figure 10.33 shows the constraint results of hover to rightward flight maneuver obtained in perturbation constant comparison for Broyden's Method.




BM
Roll Angle
Time [sec]
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4


BM
Engine Power
Time [sec]
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4

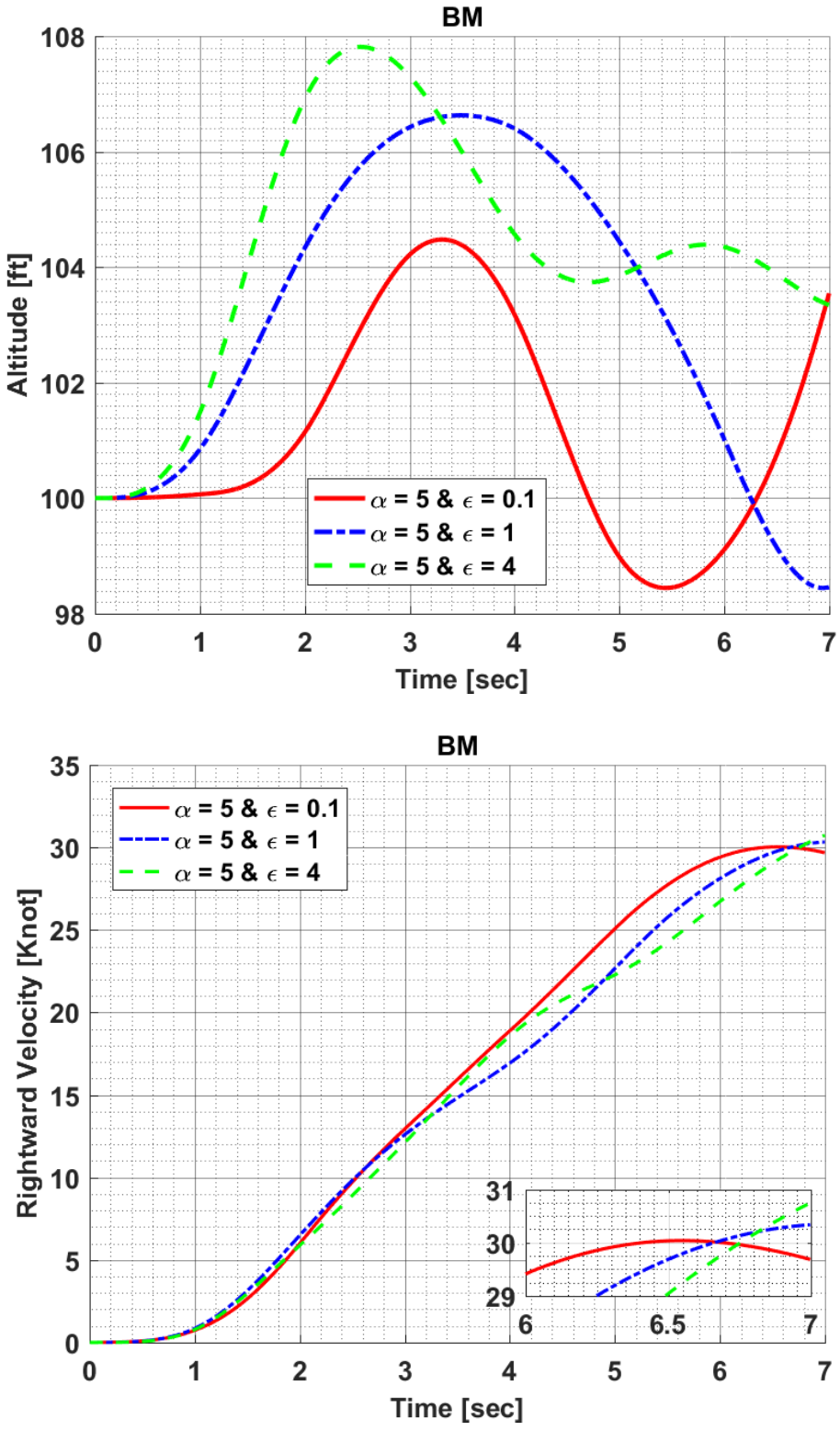


Figure 10.33: Constraint Results of Hover to Forward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for BM

Figure 10.34 shows the required longitudinal and collective cyclic input to be able to carry out hover to rightward flight maneuver as given in Figure 10.33.

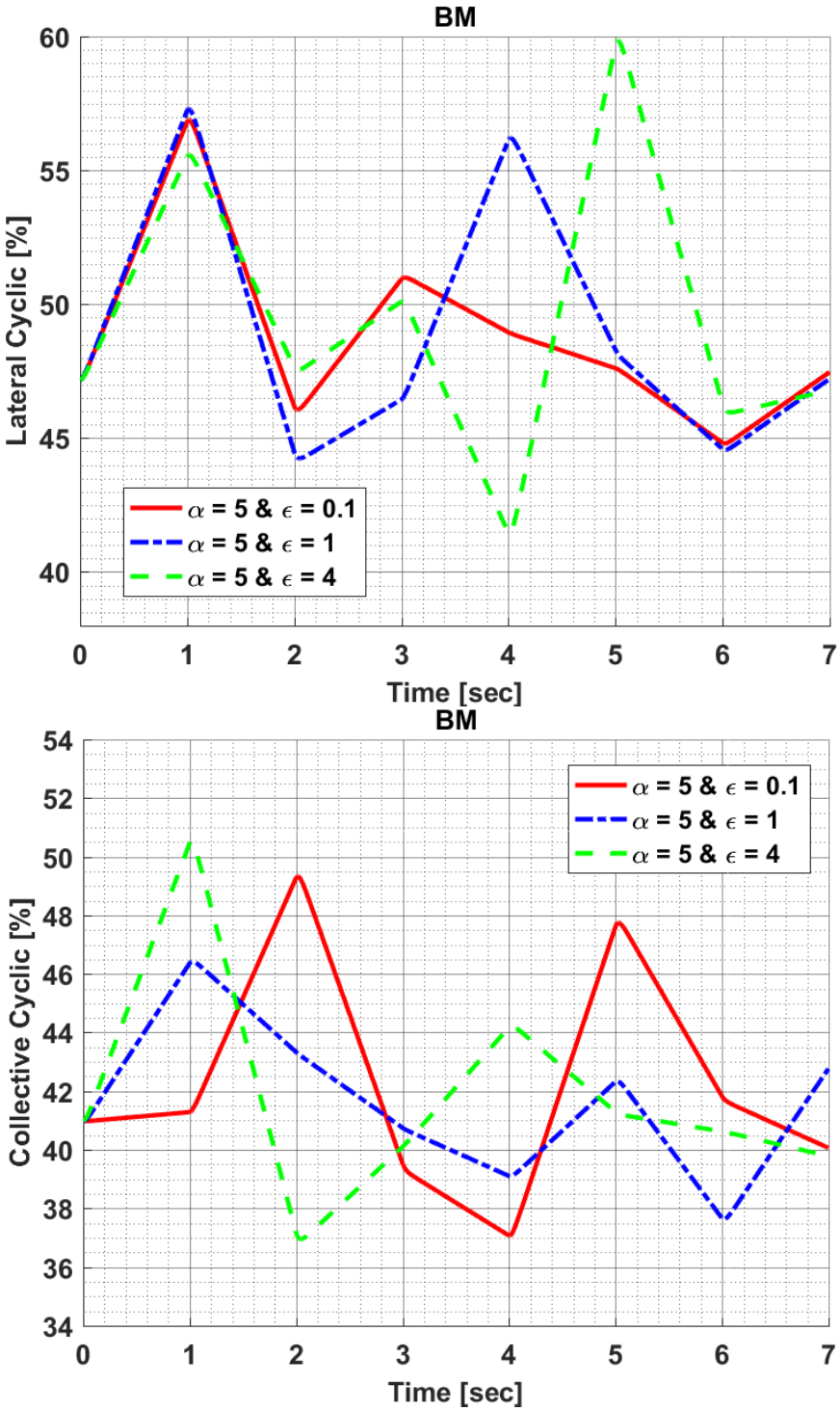
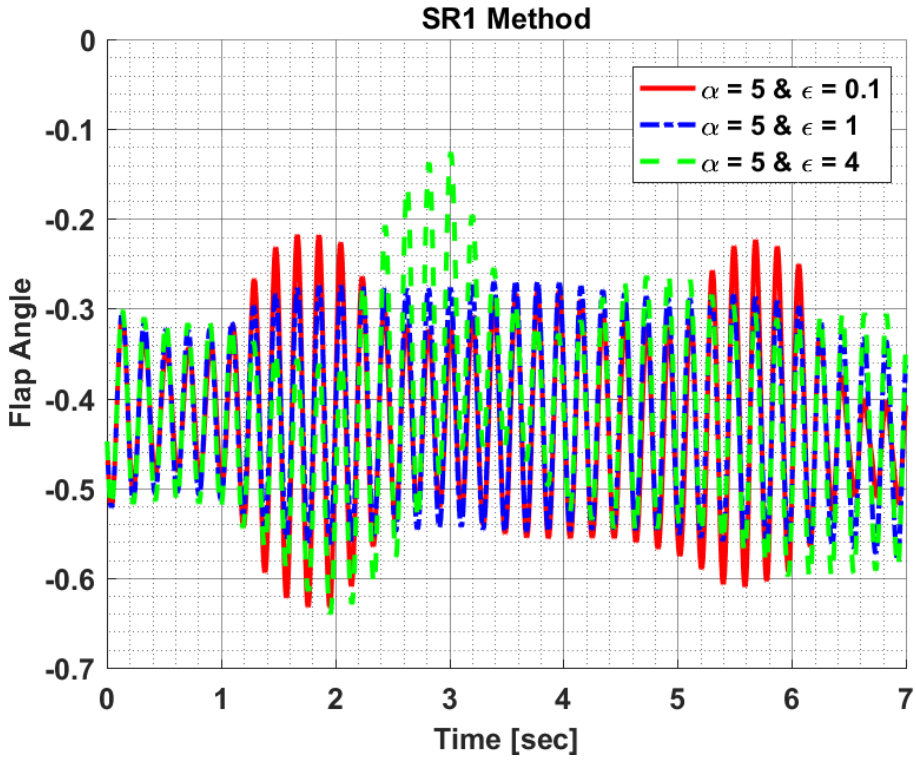
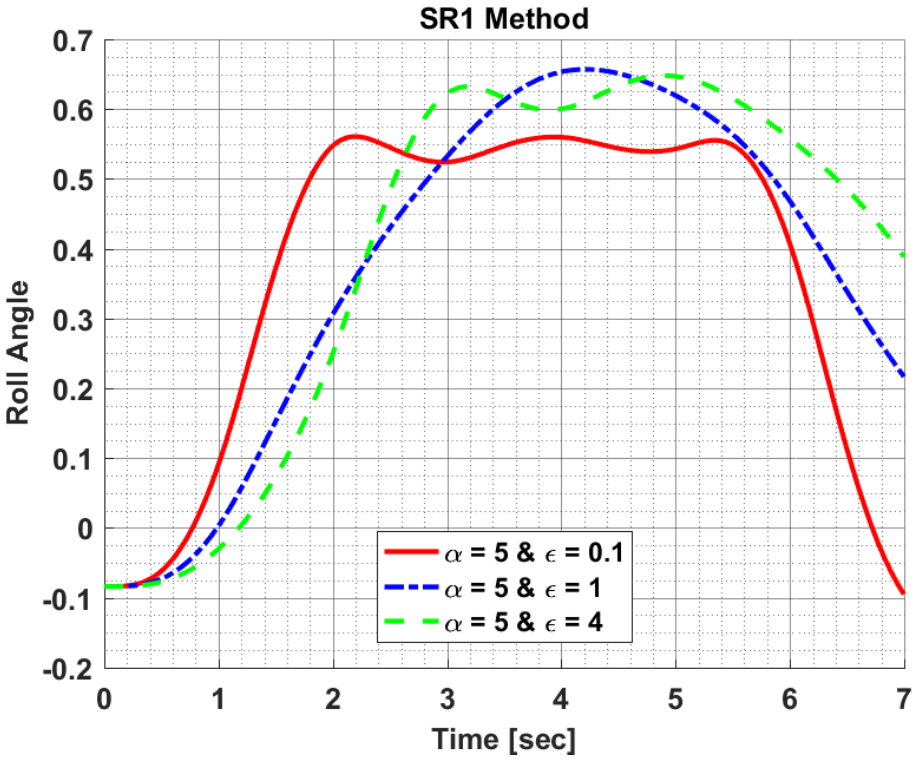
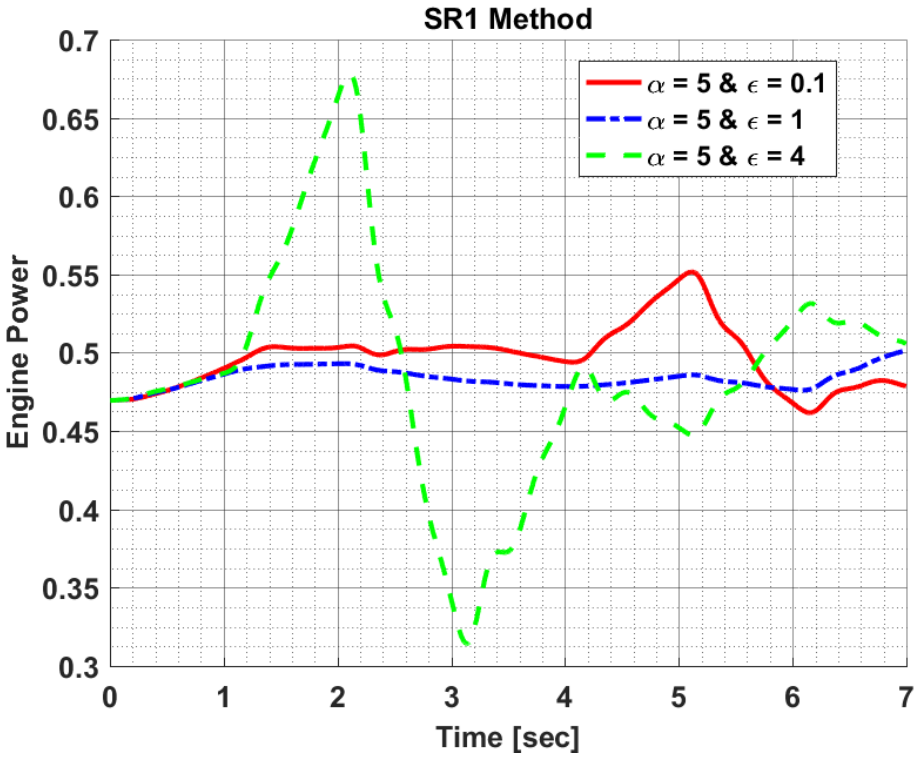


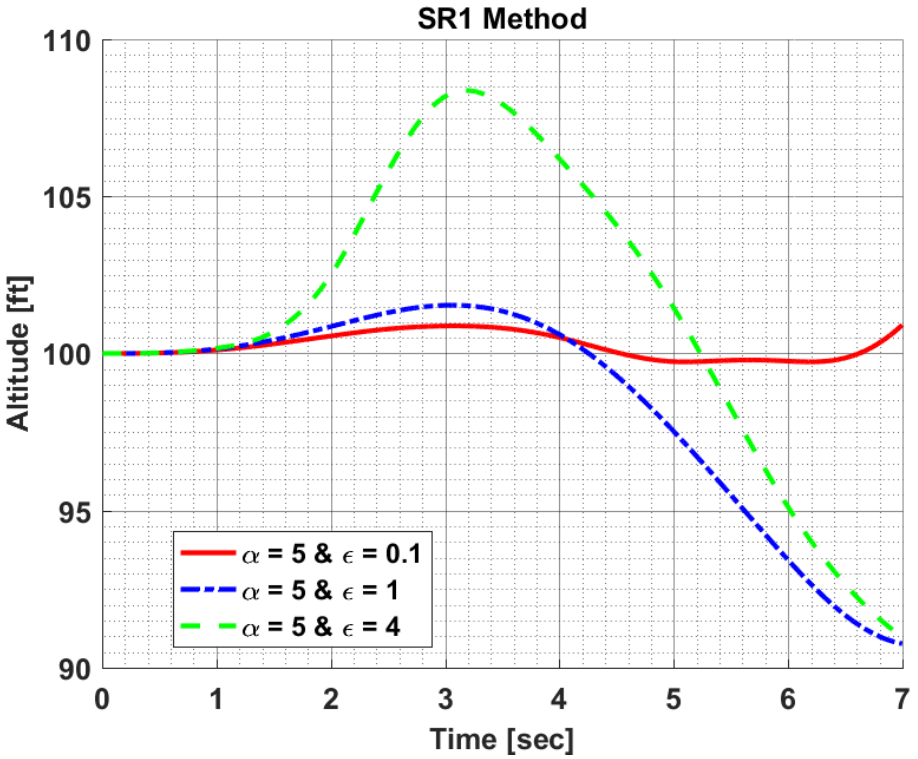
Figure 10.34: Design Variables of Hover to Rightward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for BM

Results of SR1 Method:






SR1 Method
Engine Power
Time [sec]
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
0.7
0.65
0.6
0.55
0.5
0.45
0.4
0.35
0.3
0
1
2
3
4
5
6
7


SR1 Method
Altitude [ft]
Time [sec]
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
110
105
100
95
90
0
1
2
3
4
5
6
7

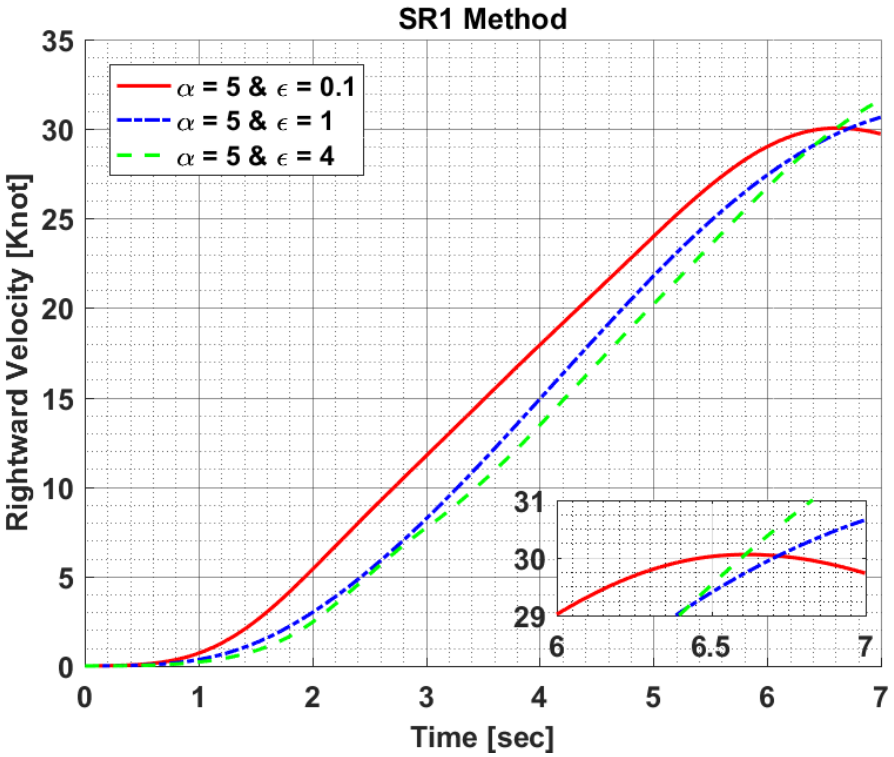


Figure 10.35: Constraint Results of Hover to Rightward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for SR1 Method

Figure 10.35 shows the constraint results of hover to rightward flight maneuver obtained in perturbation constant comparison for SR1. Moreover, Figure 10.36 shows the required lateral and collective cyclic input to be able to carry out hover to rightward flight maneuver as given in Figure 10.35.

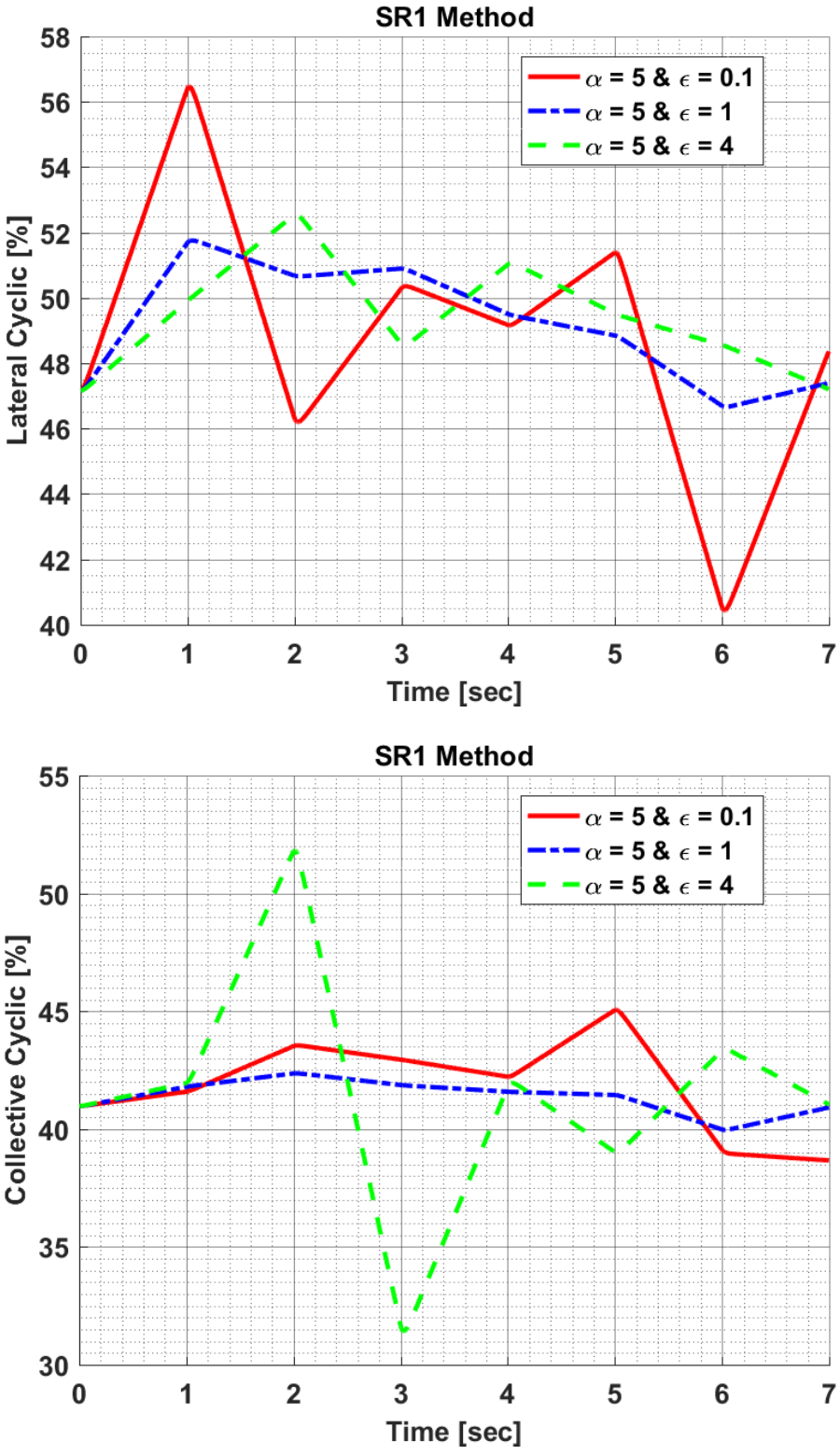
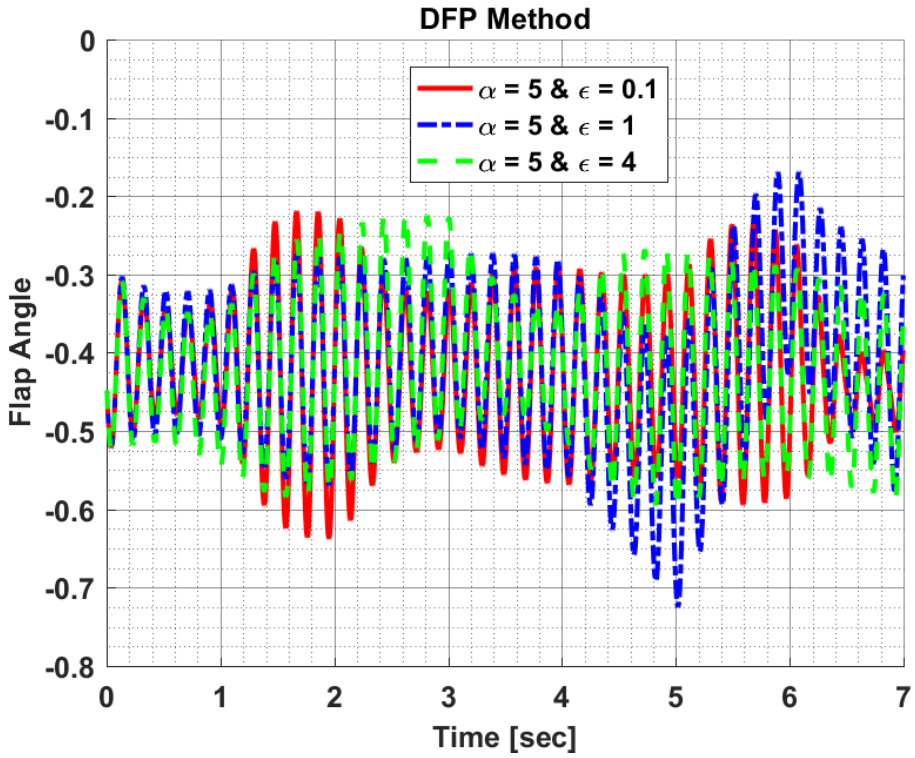
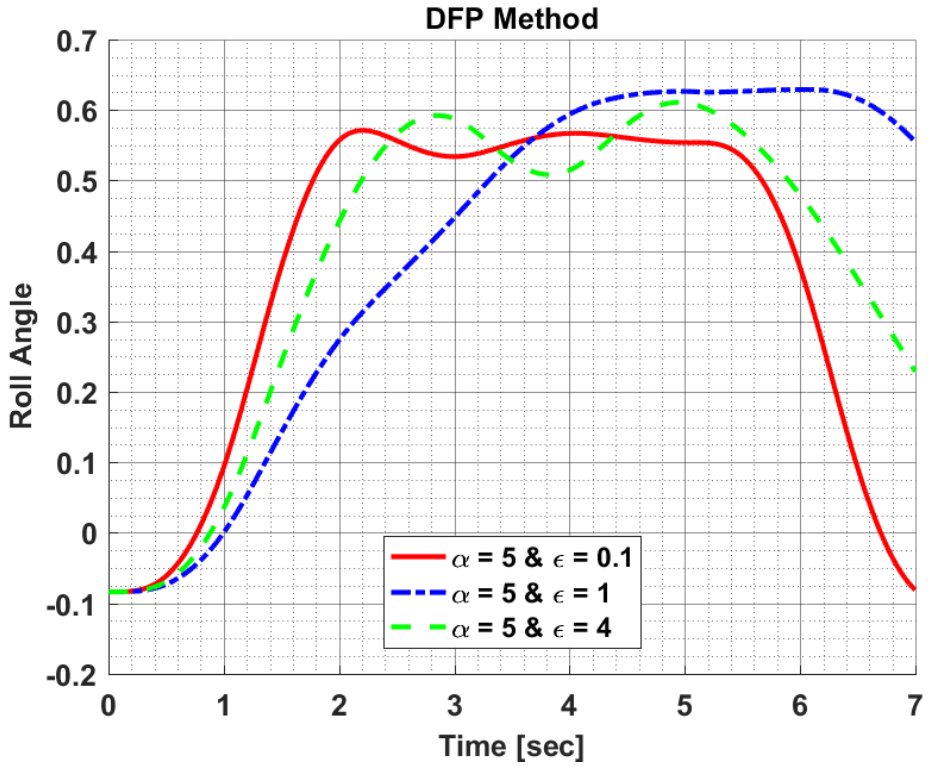
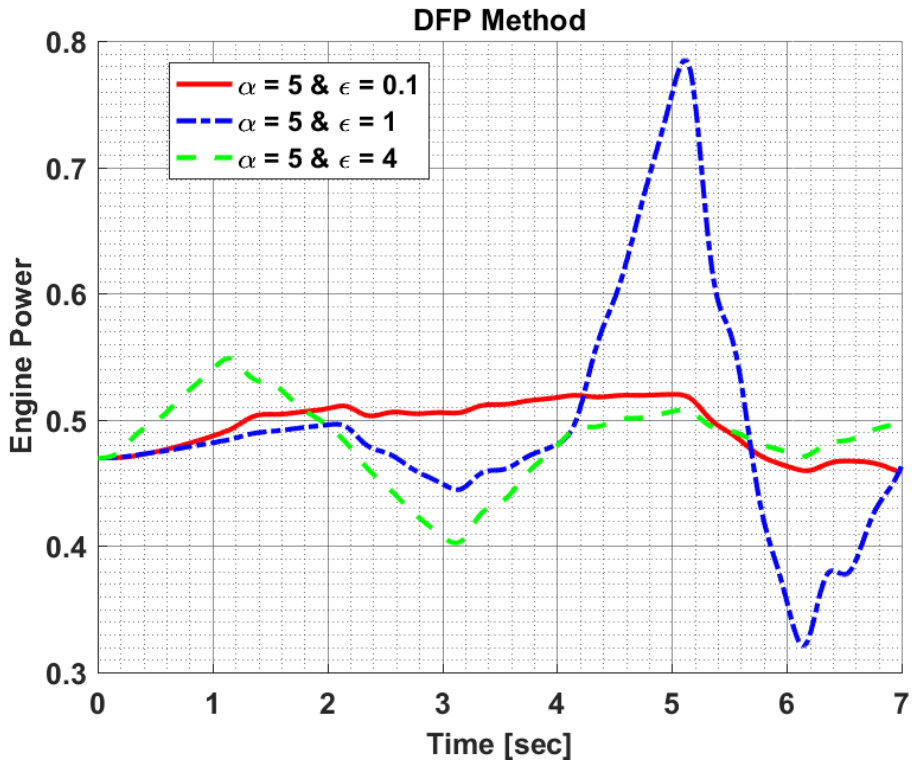


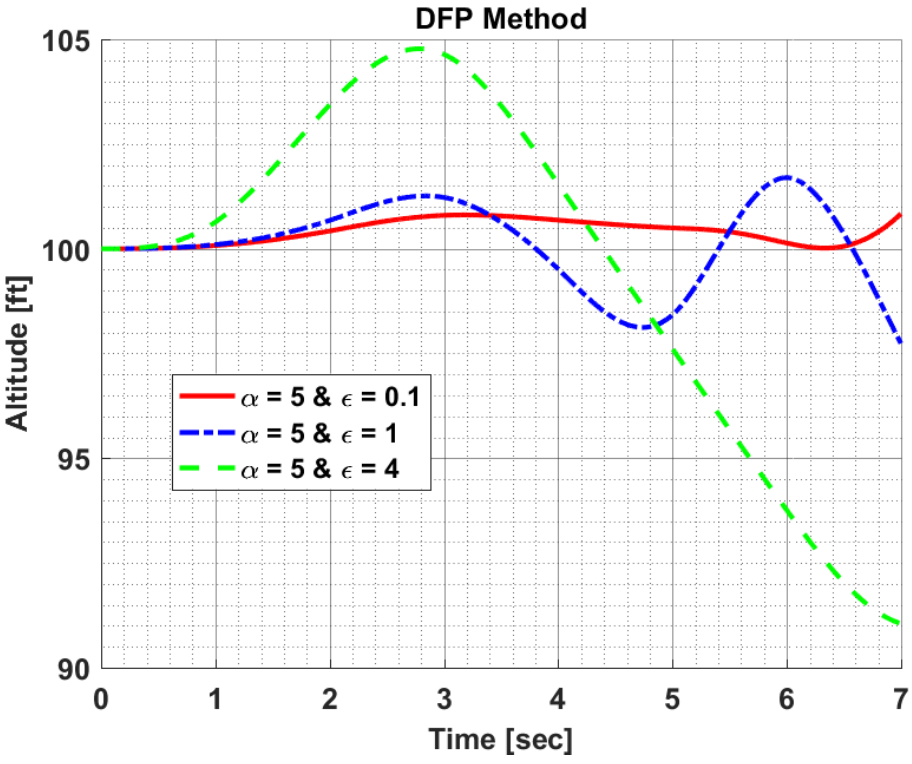
Figure 10.36: Design Variables of Hover to Rightward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for SR1 Method

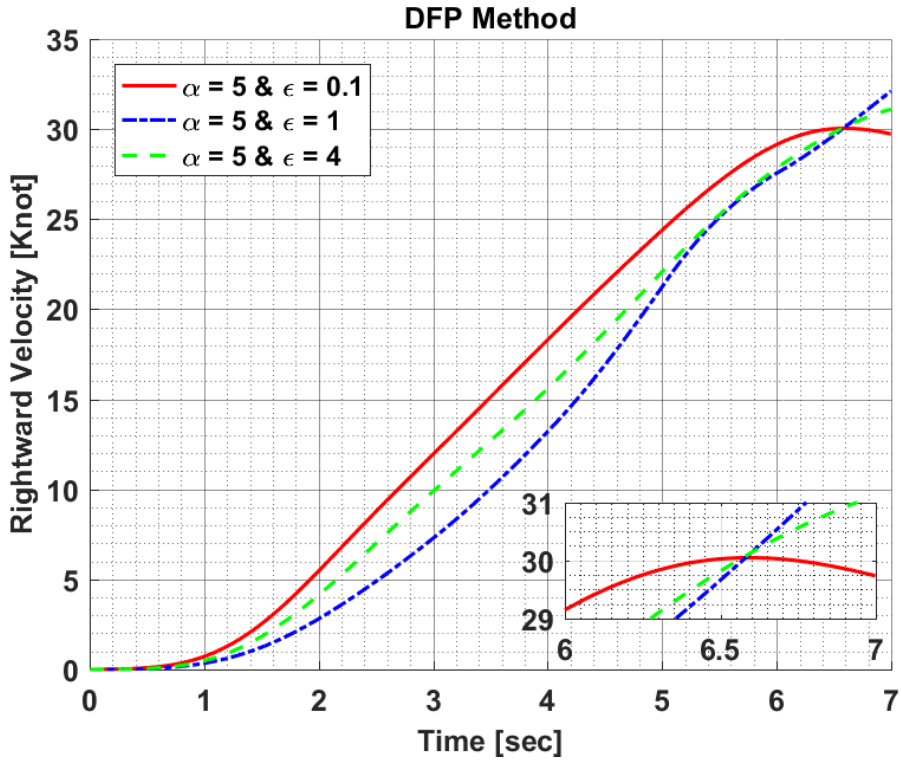
Results of DFP Method:






DFP Method
Engine Power
Time [sec]
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
0.8
0.7
0.6
0.5
0.4
0.3
0
1
2
3
4
5
6
7


DFP Method
Altitude [ft]
Time [sec]
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
105
100
95
90
0
1
2
3
4
5
6
7



,

Figure 10.37: Constraint Results of Hover to Rightward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for DFP Method

Figure 10.13 shows the constraint results of hover to rightward flight maneuver obtained in perturbation constant comparison for DFP. Moreover, Figure 10.38 shows the required lateral and collective cyclic input to be able to carry out hover to rightward flight maneuver as given in Figure 10.37.

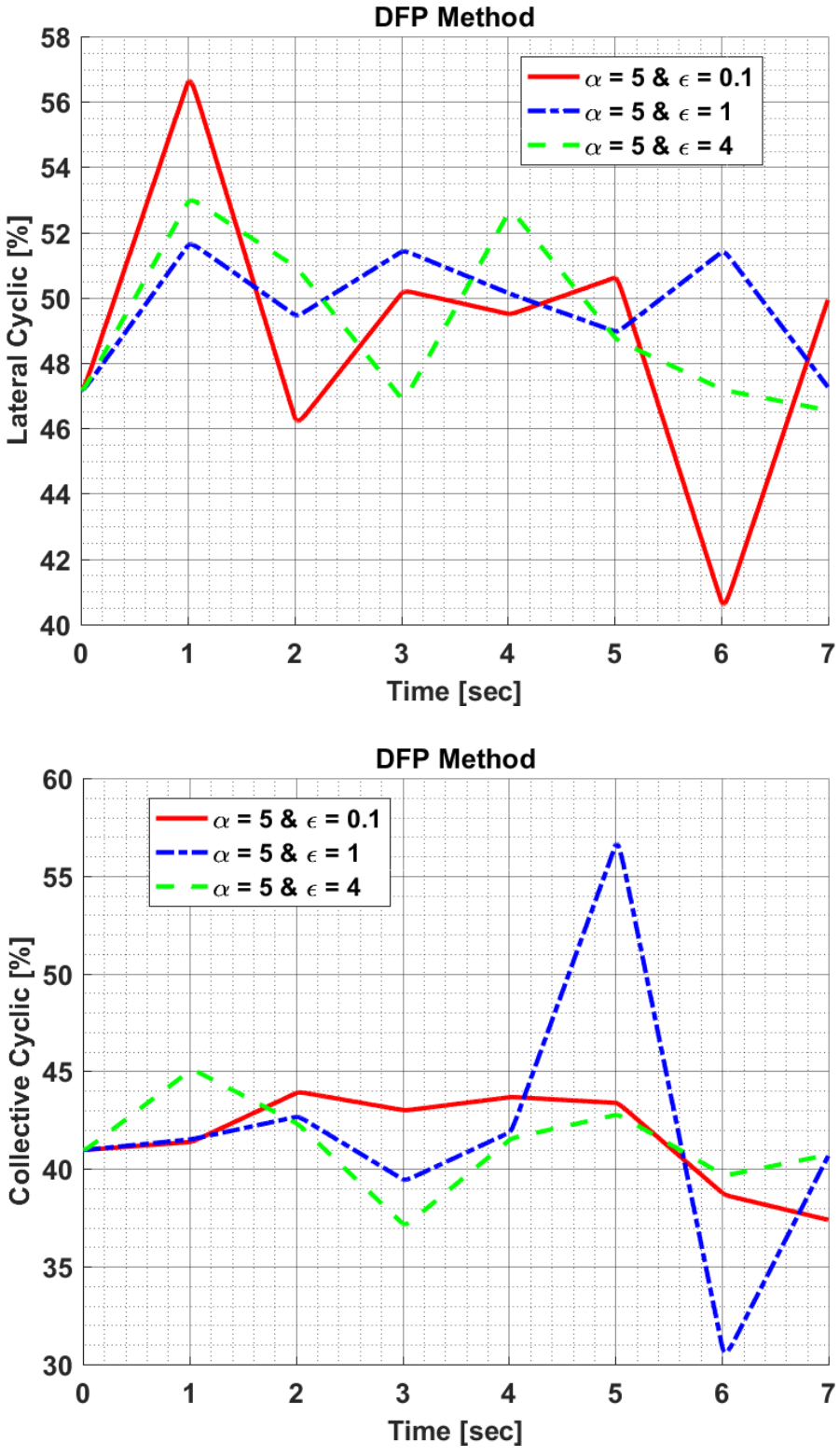
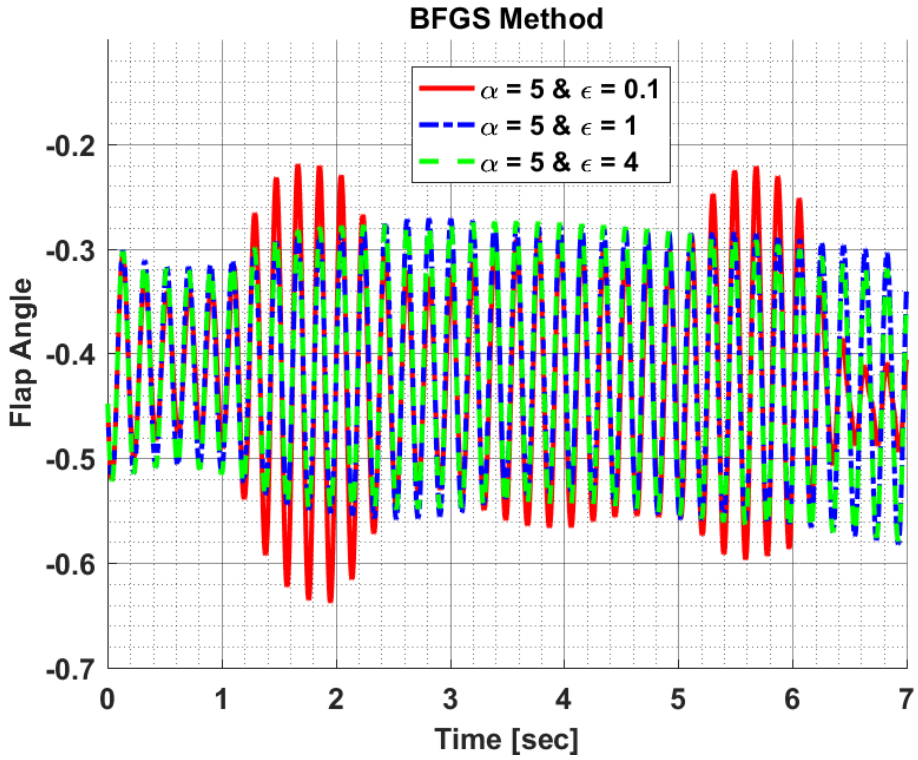
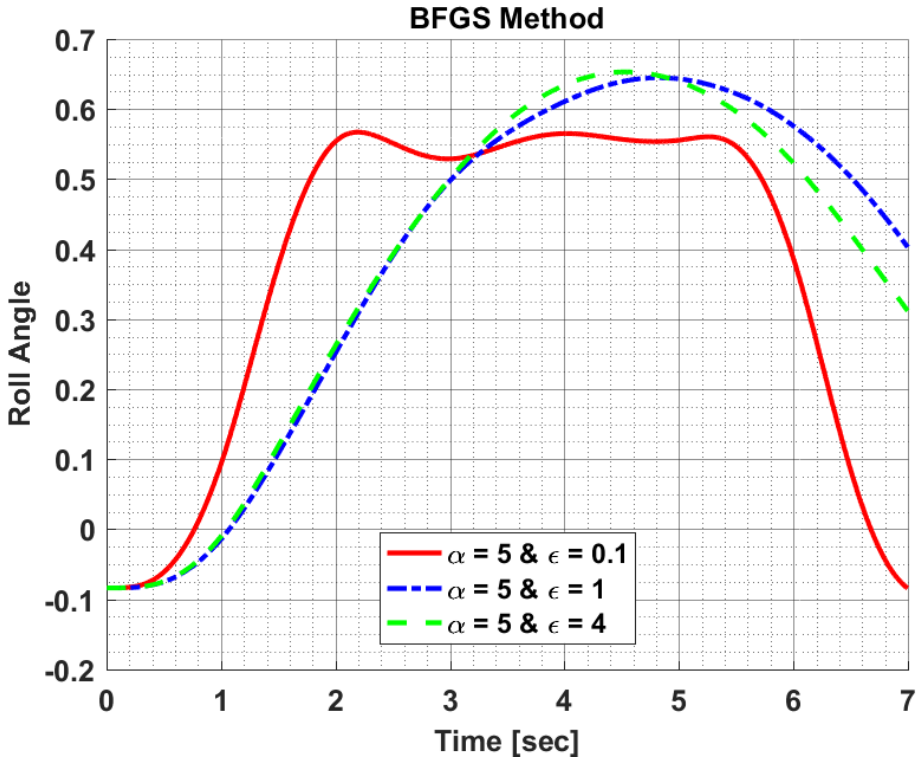
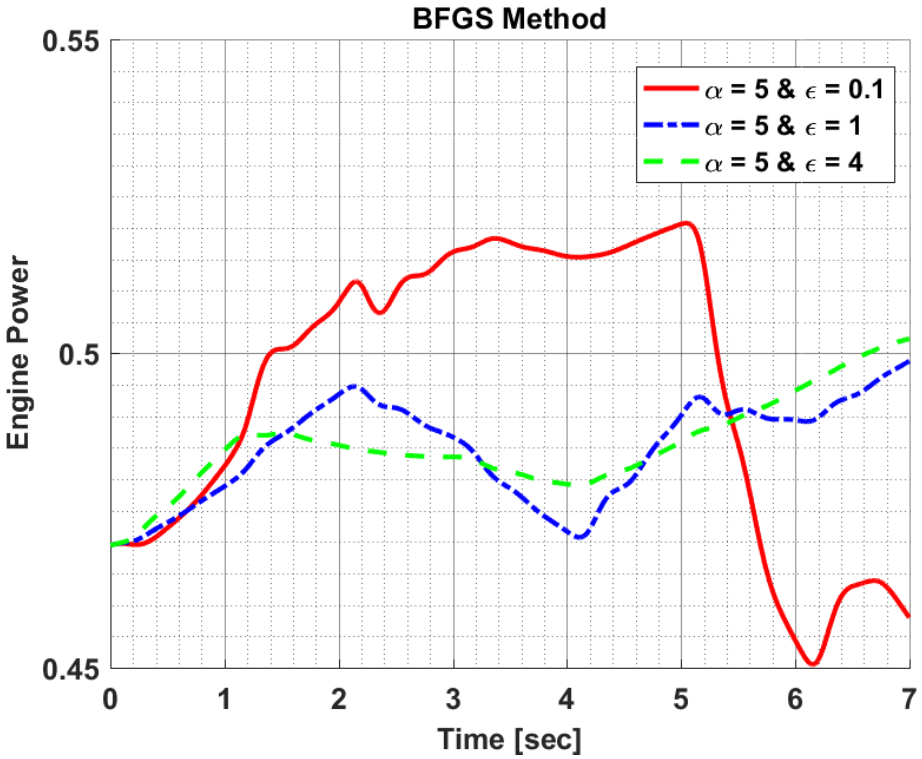


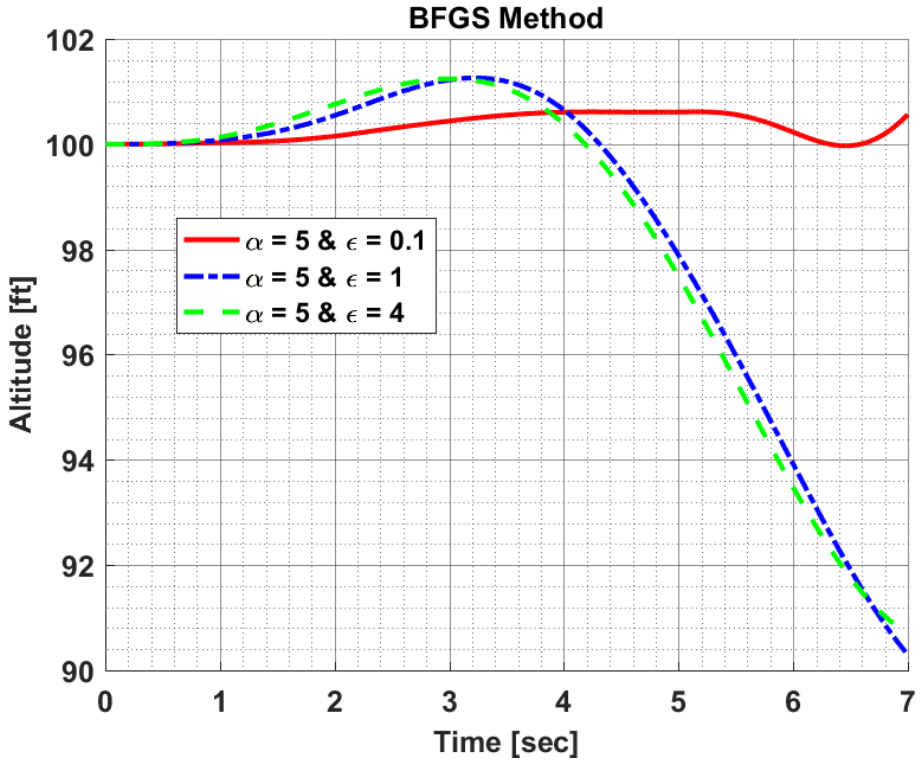
Figure 10.38: Design Variables of Hover to Rightward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for DFP Method

Results of BFGS Method:






BFGS Method
Engine Power
Time [sec]
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
0.55
0.5
0.45
0
1
2
3
4
5
6
7


BFGS Method
Altitude [ft]
Time [sec]
α = 5 & ϵ = 0.1
α = 5 & ϵ = 1
α = 5 & ϵ = 4
102
100
98
96
94
92
90
0
1
2
3
4
5
6
7

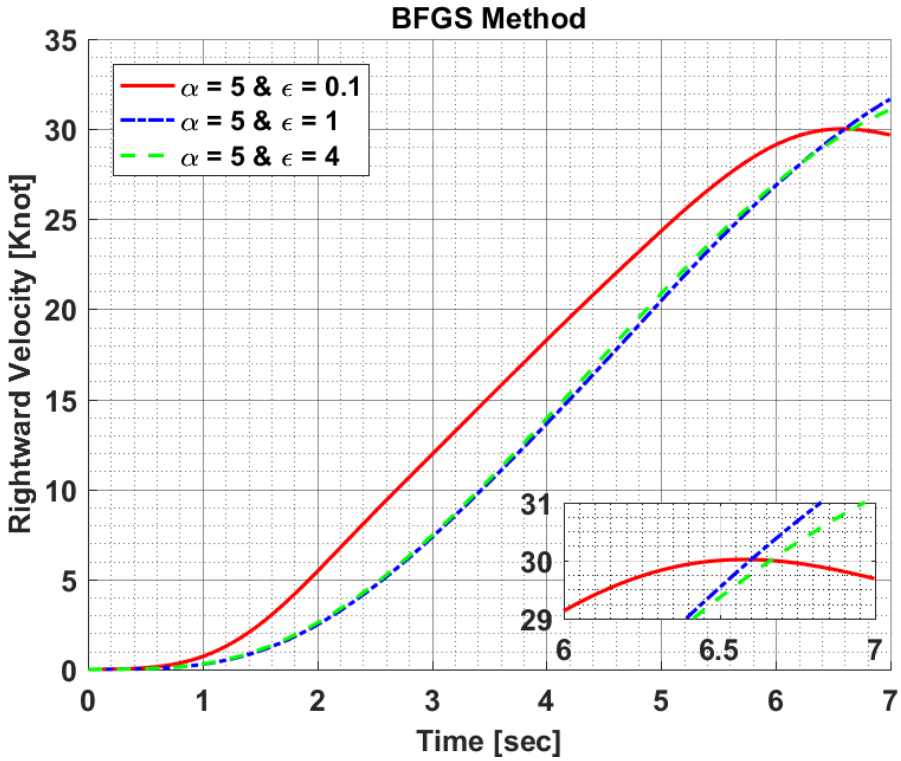


Figure 10.39: Constraint Results of Hover to Rightward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for BFGS Method

Figure 10.39 shows the constraint results of hover to rightward flight maneuver obtained in perturbation constant comparison for BFGS. Moreover, Figure 10.40 shows the required lateral and collective cyclic input to be able to carry out hover to rightward flight maneuver as given in Figure 10.39.

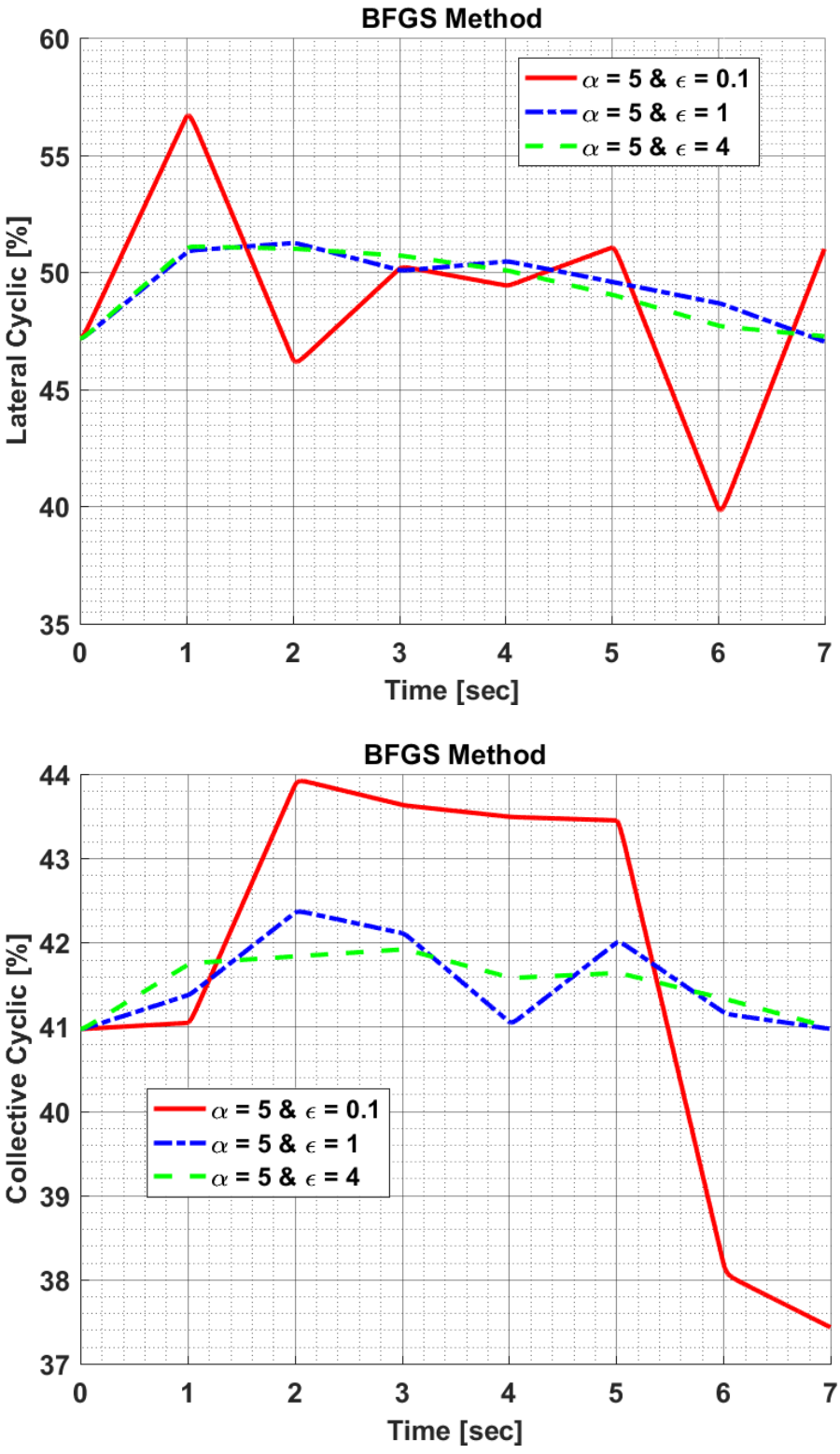


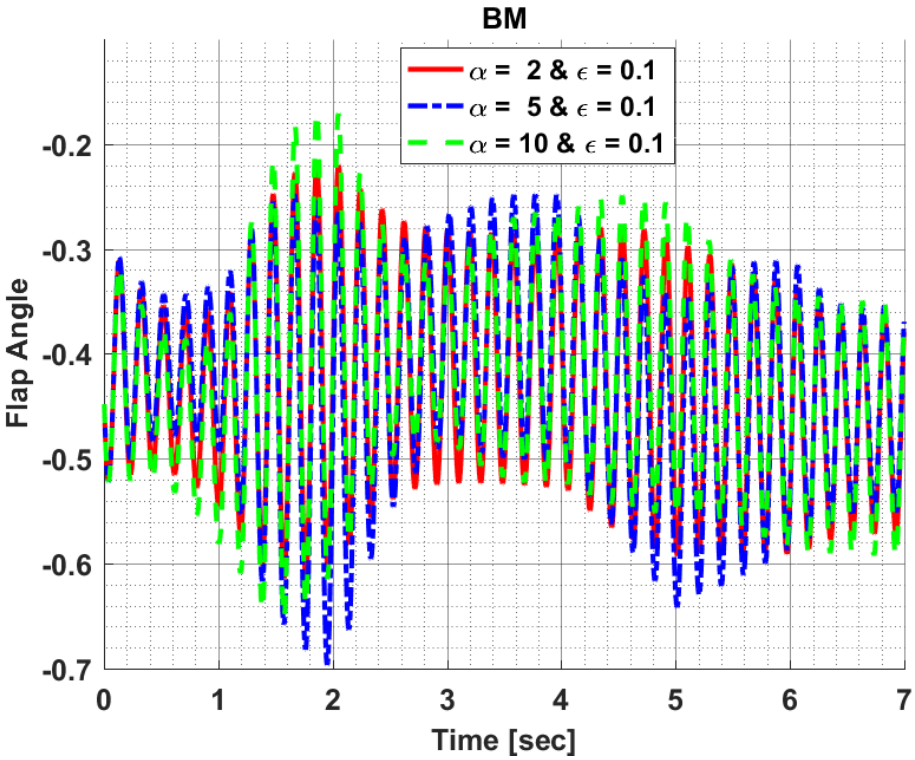
Figure 10.40: Design Variables of Hover to Rightward Flight Maneuver Optimization in Perturbation Constant Sensitivity Analysis for BFGS Method

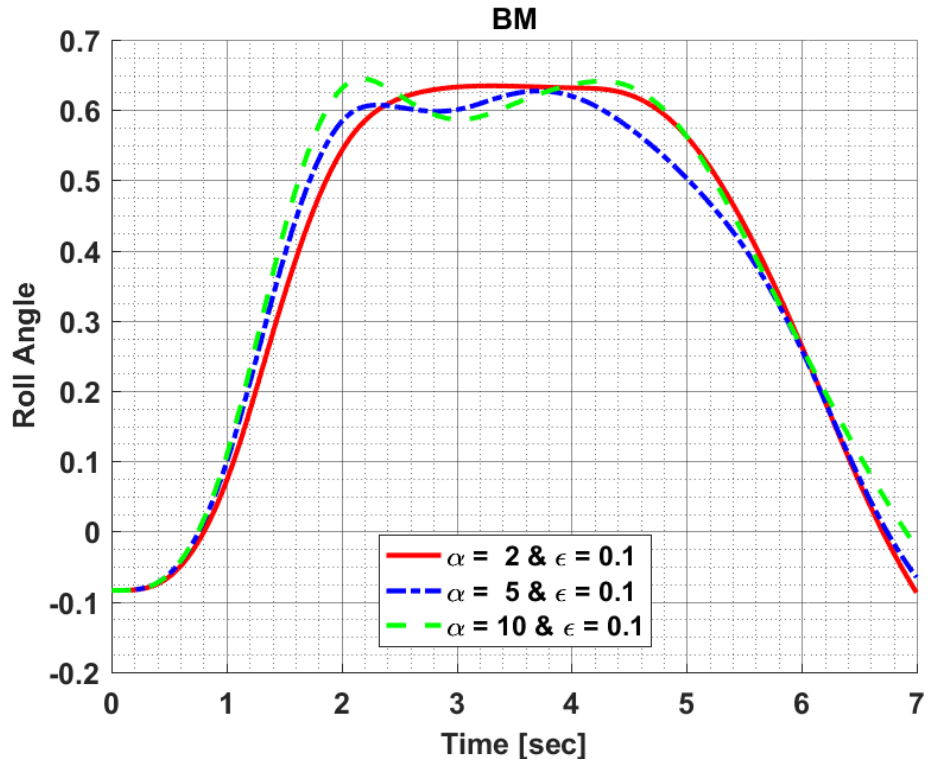
APPENDIX B3: Comparison – Sensitivity Analysis of Step Size Parameter for Hover to Rightward Flight Maneuver Optimization

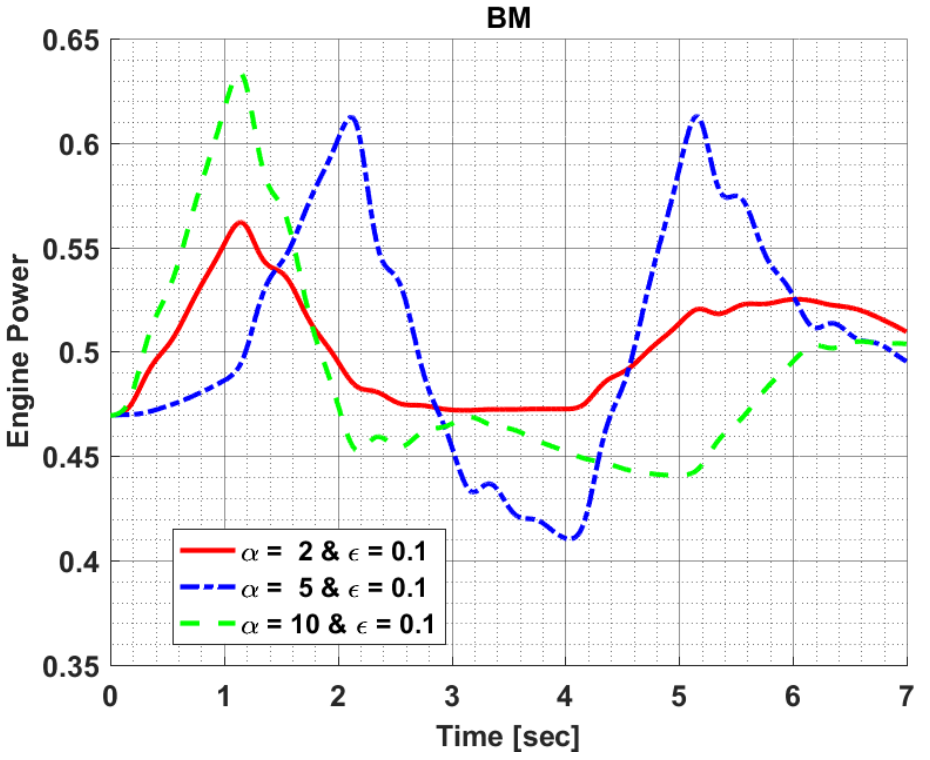
In this appendix section, constraint results and corresponding pilot control inputs are provided for each optimization method configuration which are executed for the sensitivity analysis of step size parameter. Moreover, they are the results of hover to rightward flight maneuver optimization.

Results of BM:

Figure 10.41 shows the constraint results of hover to rightward flight maneuver obtained in Step size comparison for Broyden's Method




BM
Roll Angle
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1


BM
Engine Power
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1

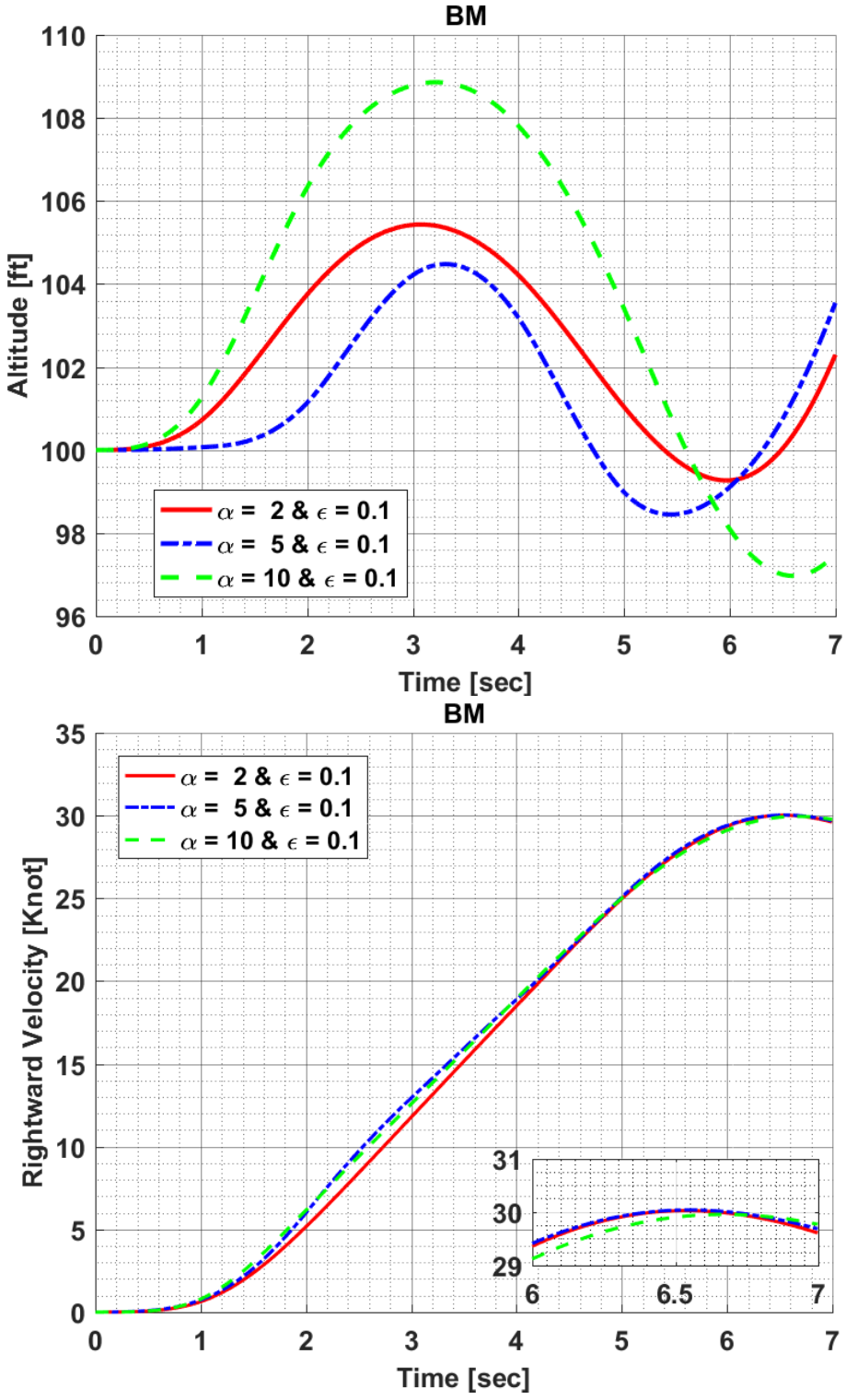


Figure 10.41: Constraint Results of Hover to Forward Flight Maneuver Optimization in Step Size Sensitivity Analysis for BM

Figure 10.42 shows the required lateral and collective cyclic input to be able to carry out hover to rightward flight maneuver as given in Figure 10.41.

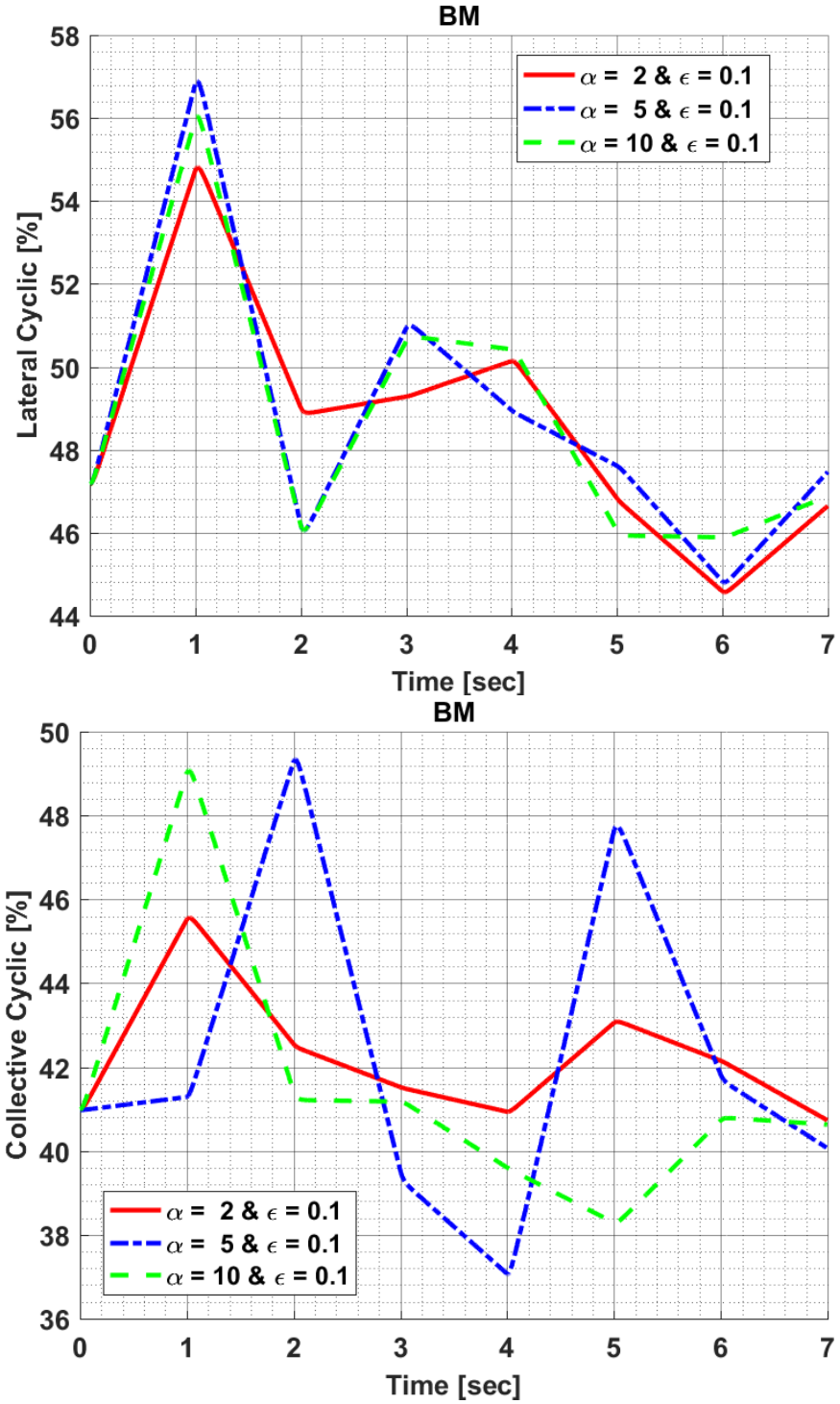
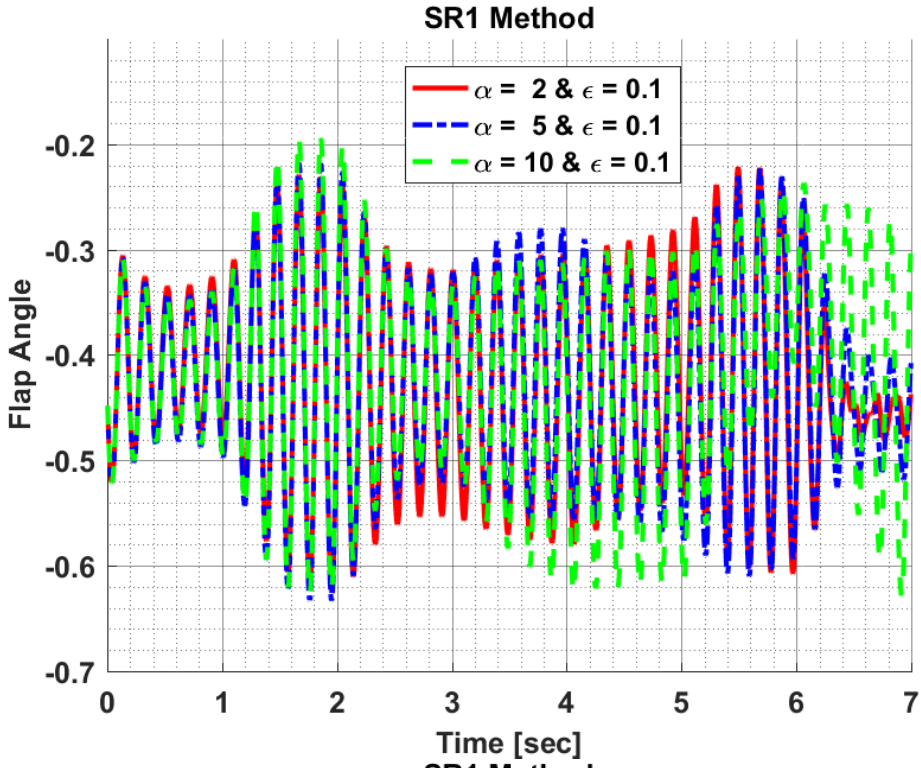
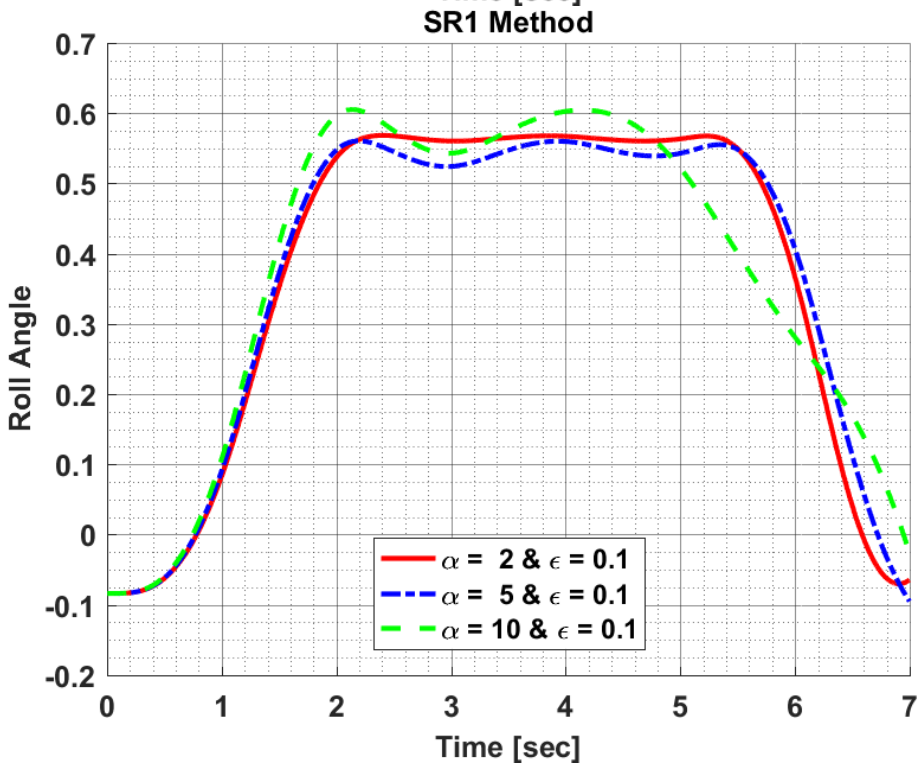
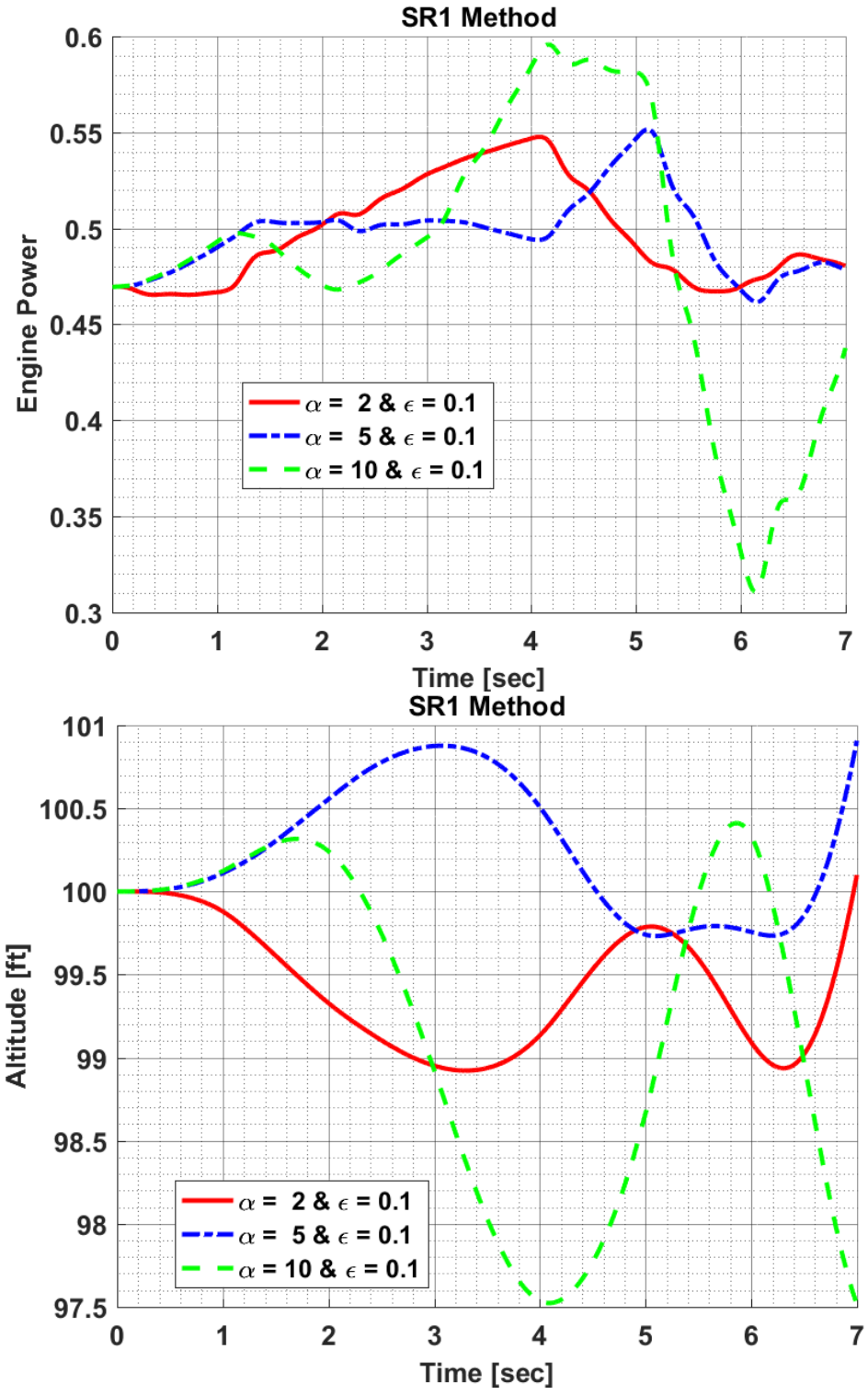


Figure 10.42: Design Variables of Hover to Rightward Flight Maneuver Optimization in Steps Size Sensitivity Analysis for BM

Results of SR1 Method:






SR1 Method
Engine Power
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1
SR1 Method
Altitude [ft]
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1

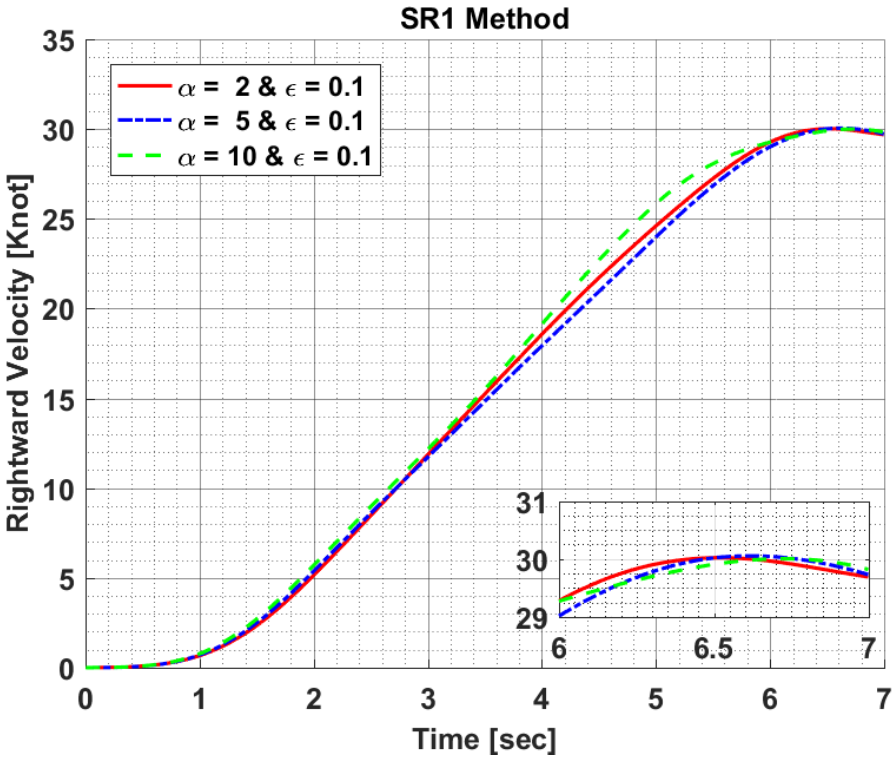


Figure 10.43: Constraint Results of Hover to Rightward Flight Maneuver Optimization in Step Size Sensitivity Analysis for SR1 Method

Figure 10.43 shows the constraint results of hover to rightward flight maneuver obtained in Step size comparison for SR1. Moreover, Figure 10.44 shows the required lateral and collective cyclic input to be able to carry out hover to rightward flight maneuver as given in Figure 10.43.

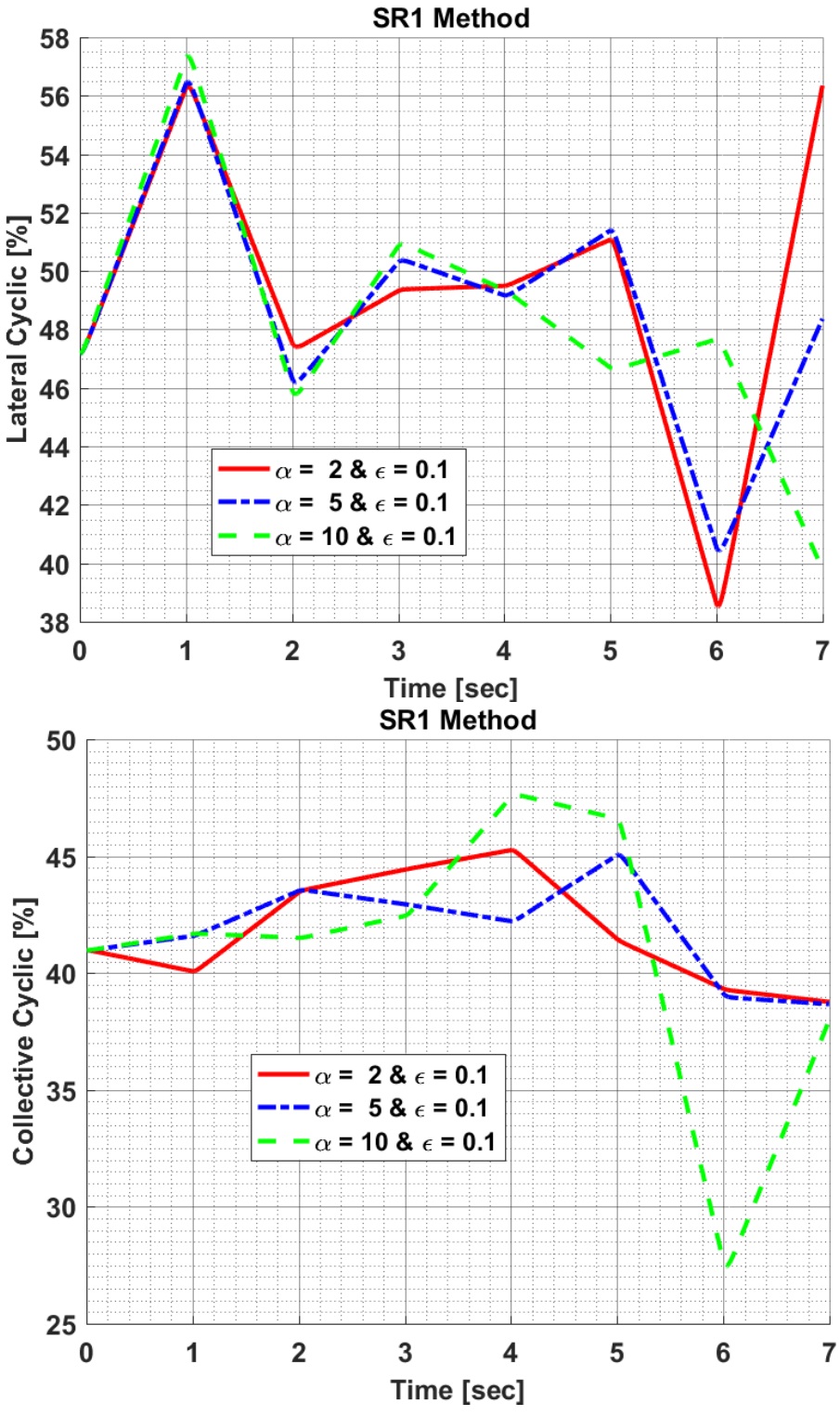
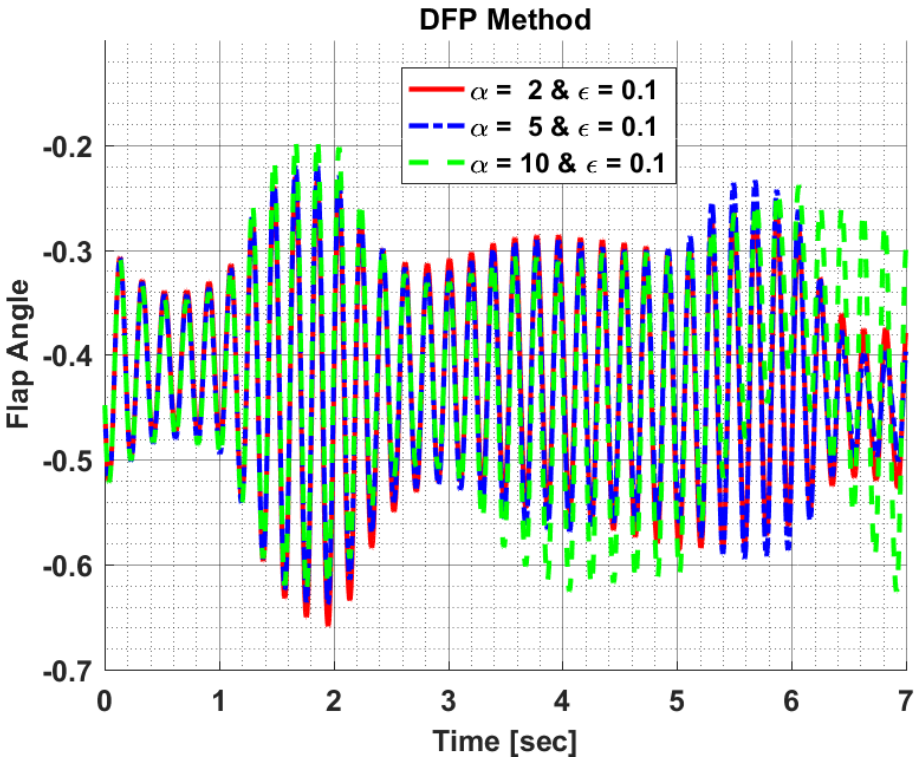
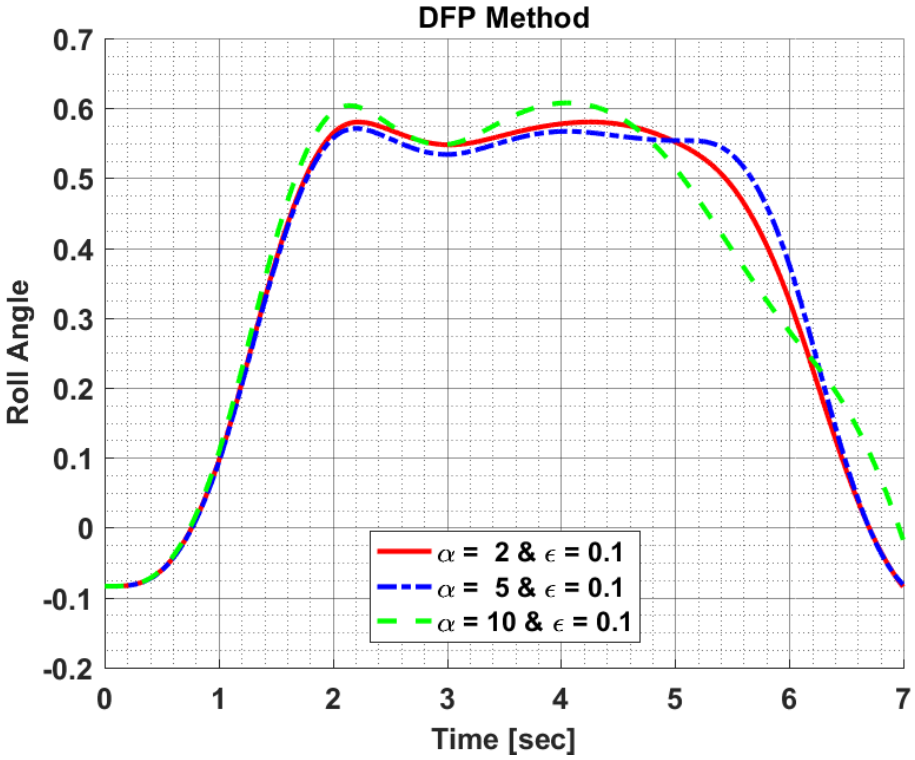
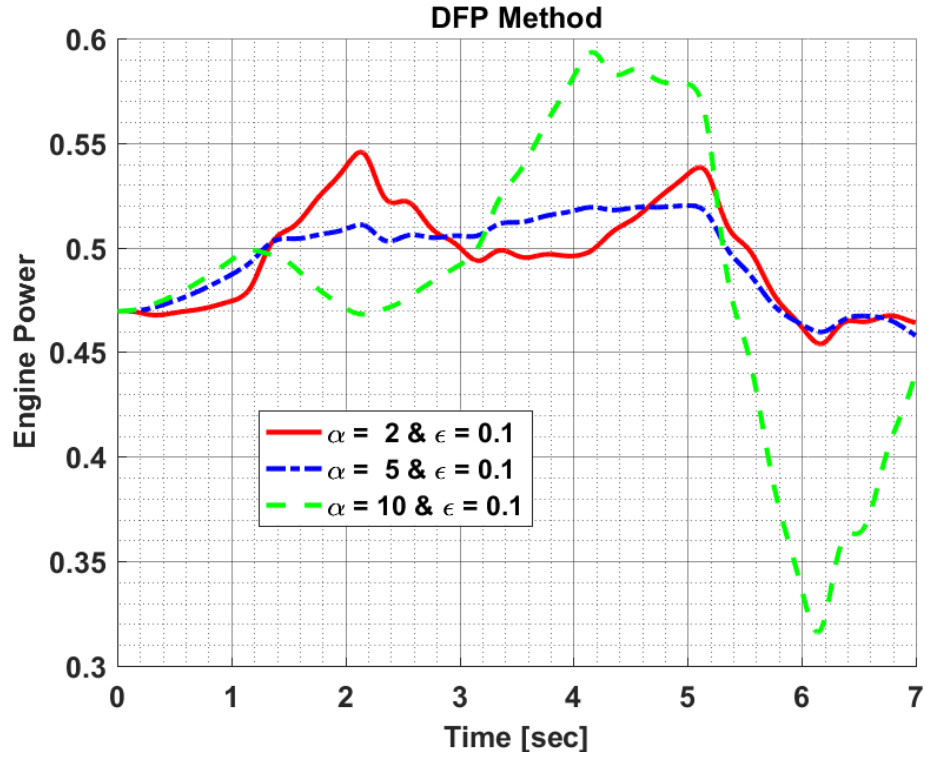


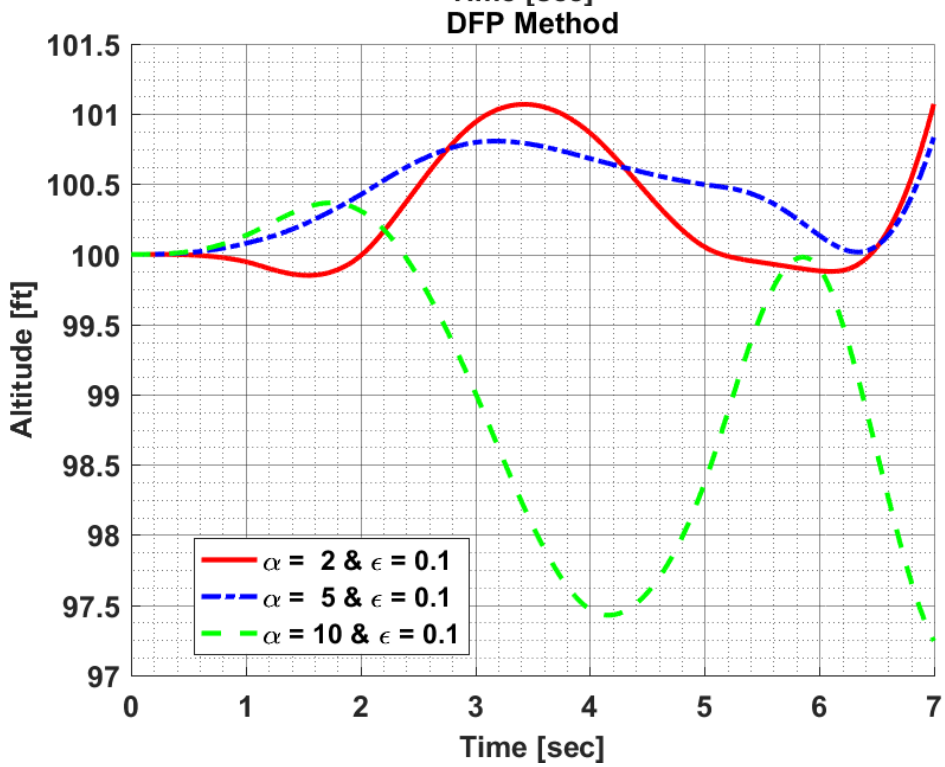
Figure 10.44: Design Variables of Hover to Rightward Flight Maneuver Optimization in Step Size Sensitivity Analysis for SR1 Method

Results of DFP Method:






DFP Method
Engine Power
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1


DFP Method
Altitude [ft]
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1

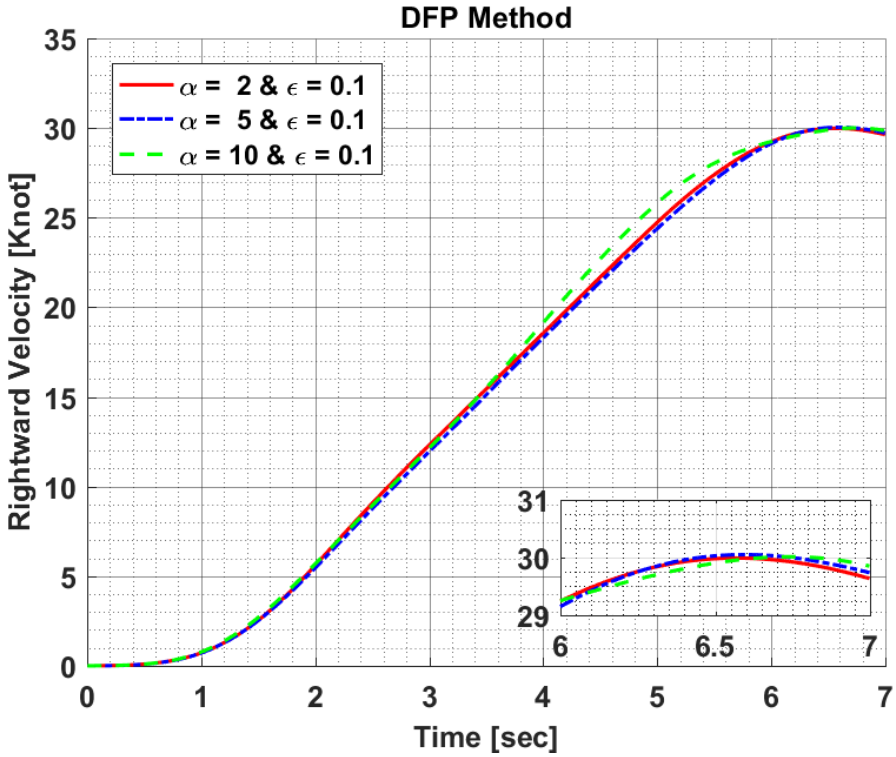


Figure 10.45: Constraint Results of Hover to Rightward Flight Maneuver Optimization in Step Size Sensitivity Analysis for DFP Method

Figure 10.45 shows the constraint results of hover to rightward flight maneuver obtained in Step Size comparison for DFP. Moreover, Figure 10.46 shows the required lateral and collective cyclic input to be able to carry out hover to rightward flight maneuver as given in Figure 10.45.

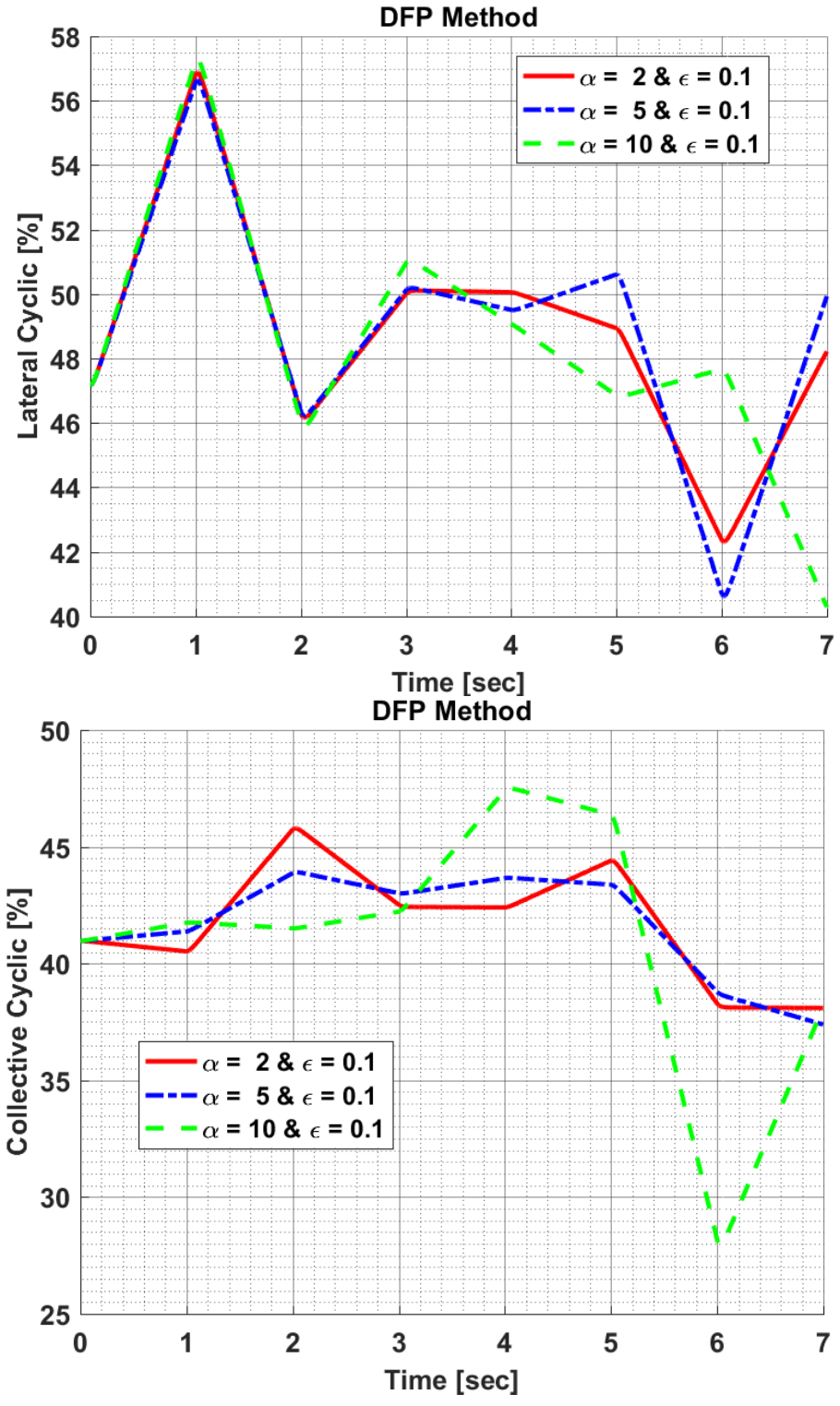
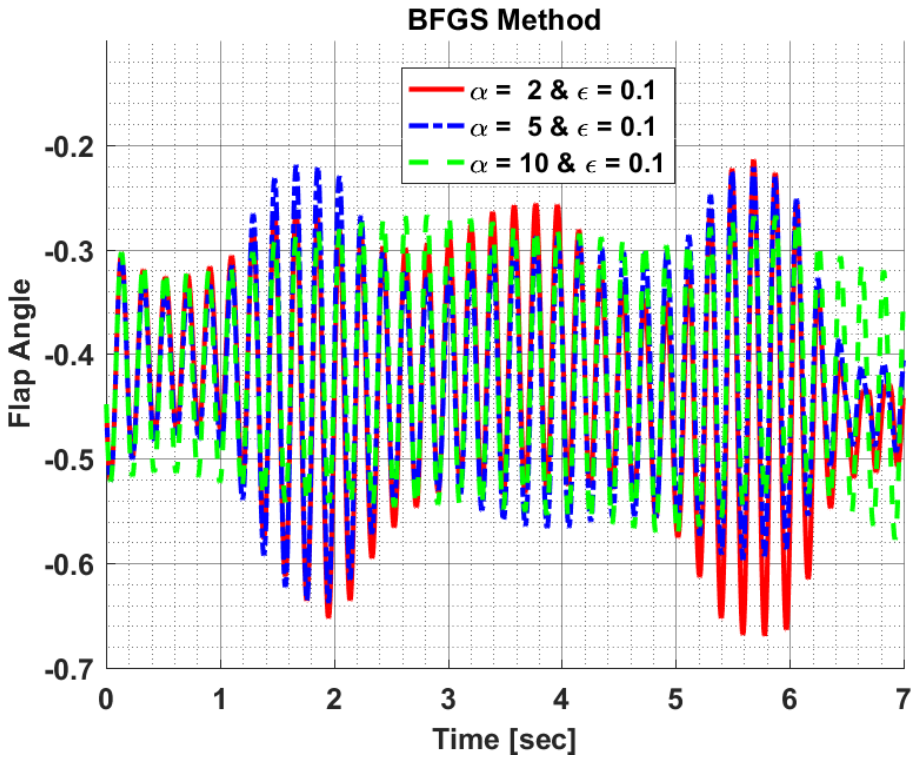
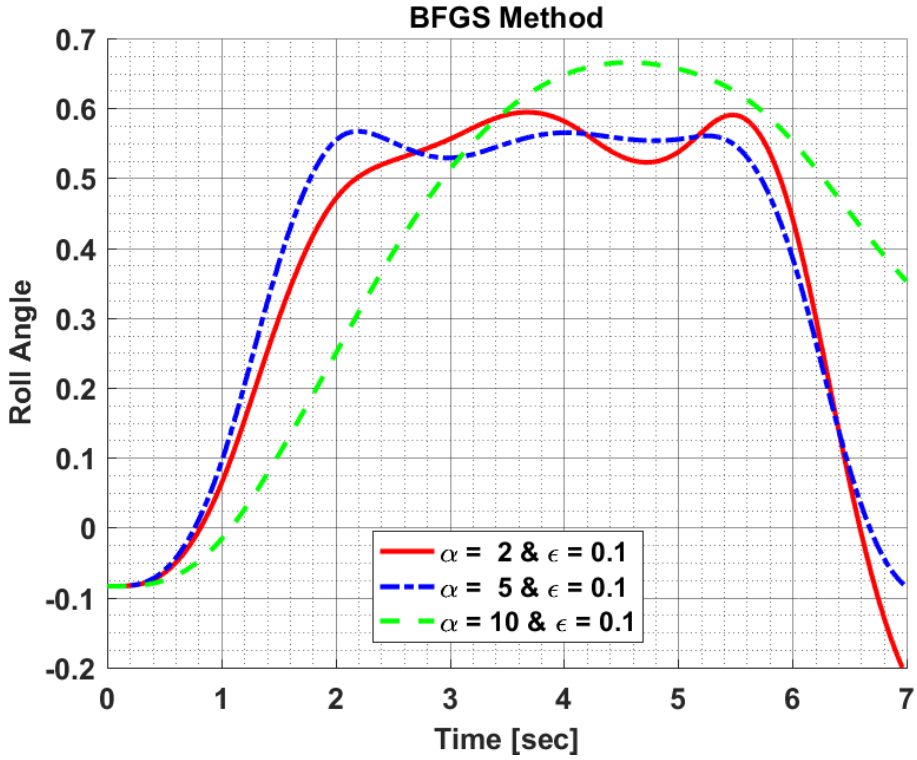
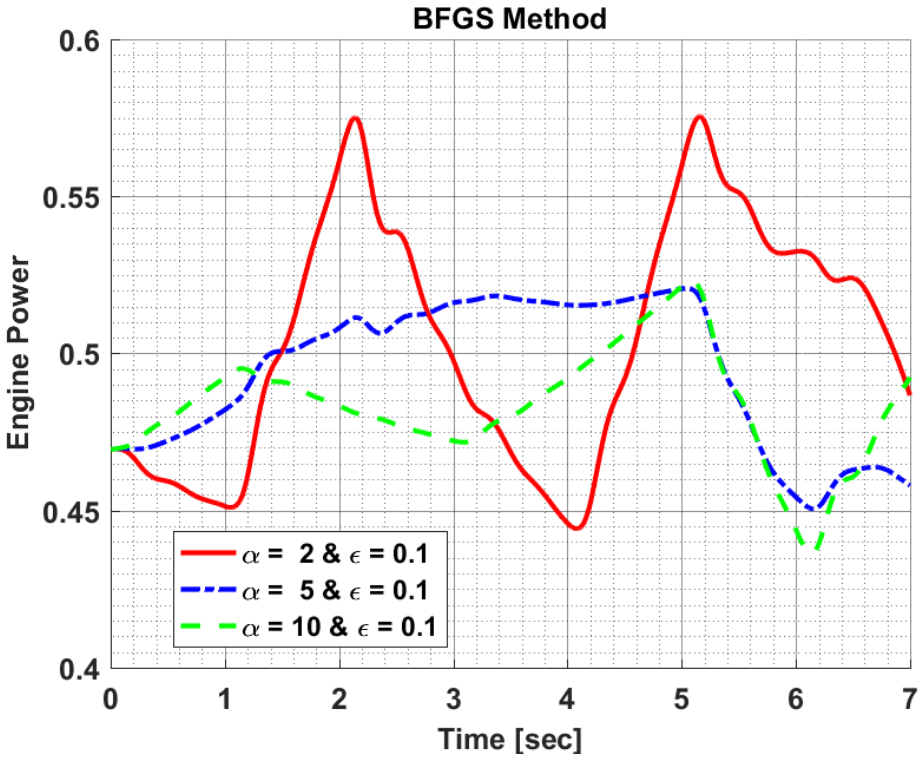


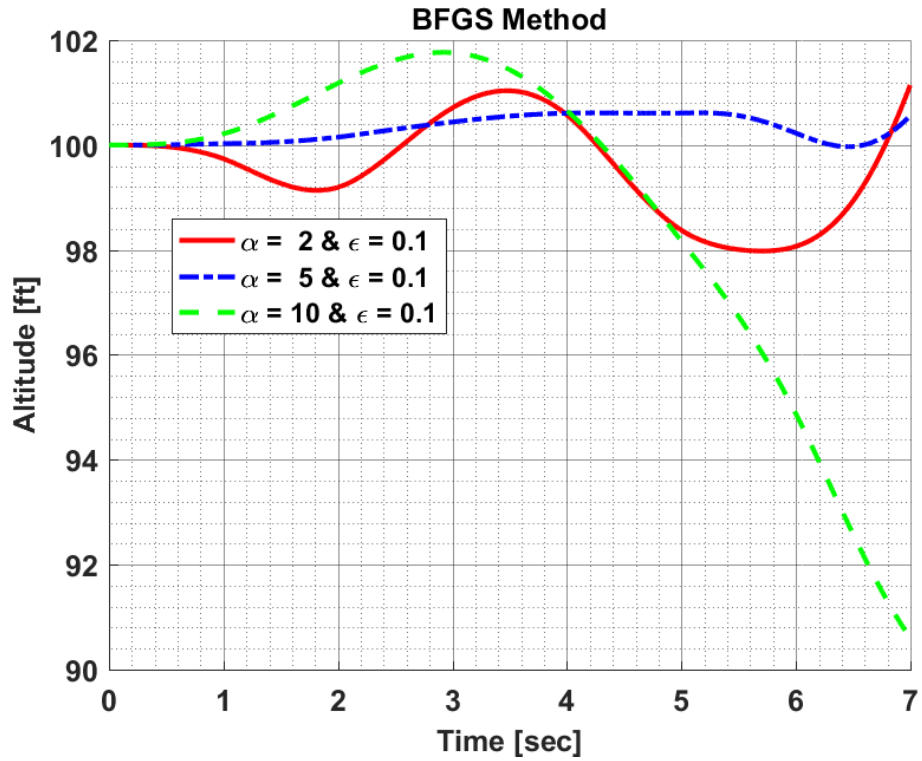
Figure 10.46: Design Variables of Hover to Rightward Flight Maneuver Optimization in Step Size Sensitivity Analysis for DFP Method

Results of BFGS Method:






BFGS Method
Engine Power
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1
0.6
0.55
0.5
0.45
0.4
0
1
2
3
4
5
6
7


BFGS Method
Altitude [ft]
Time [sec]
α = 2 & ϵ = 0.1
α = 5 & ϵ = 0.1
α = 10 & ϵ = 0.1
102
100
98
96
94
92
90
0
1
2
3
4
5
6
7

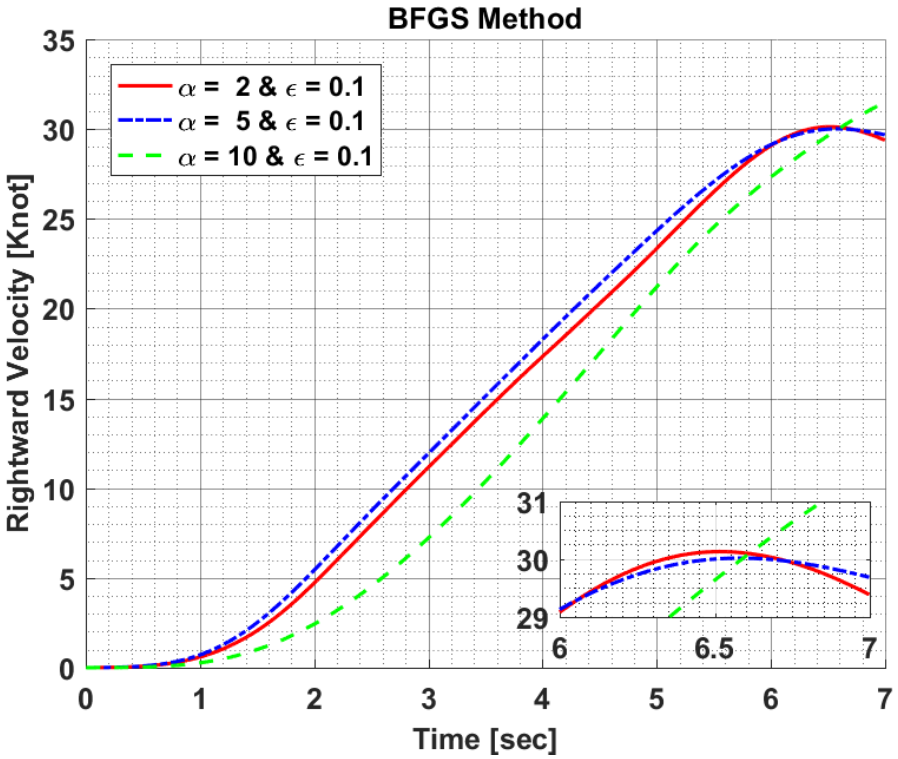


Figure 10.47: Constraint Results of Hover to Rightward Flight Maneuver Optimization in Step Size Sensitivity Analysis for BFGS Method

Figure 10.47 shows the constraint results of hover to rightward flight maneuver obtained in Step Size comparison for BFGS. Moreover, Figure 10.48 shows the required lateral and collective cyclic input to be able to carry out hover to rightward flight maneuver as given in Figure 10.47.

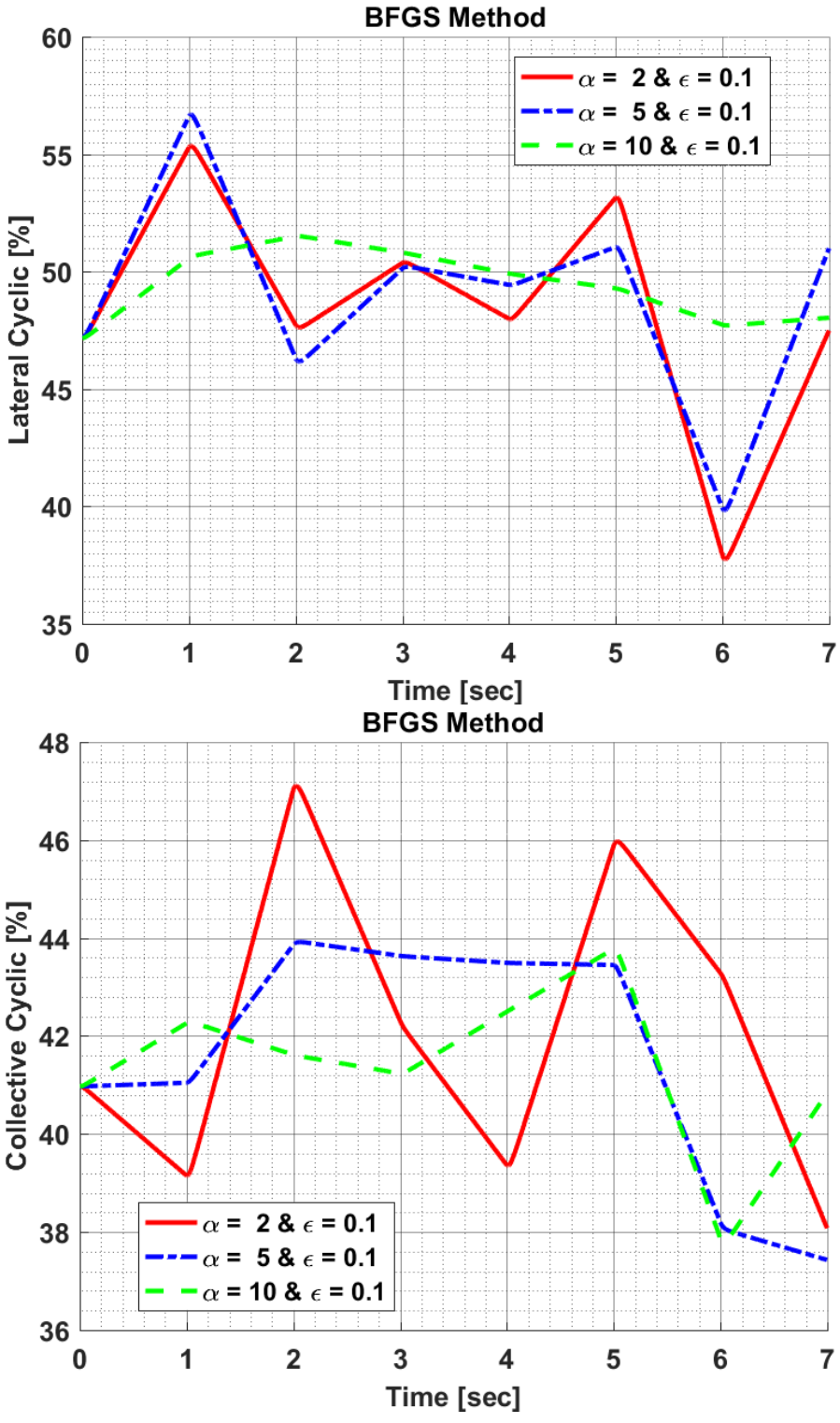


Figure 10.48: Design Variables of Hover to Rightward Flight Maneuver Optimization in Step Size Sensitivity Analysis for BFGS Method