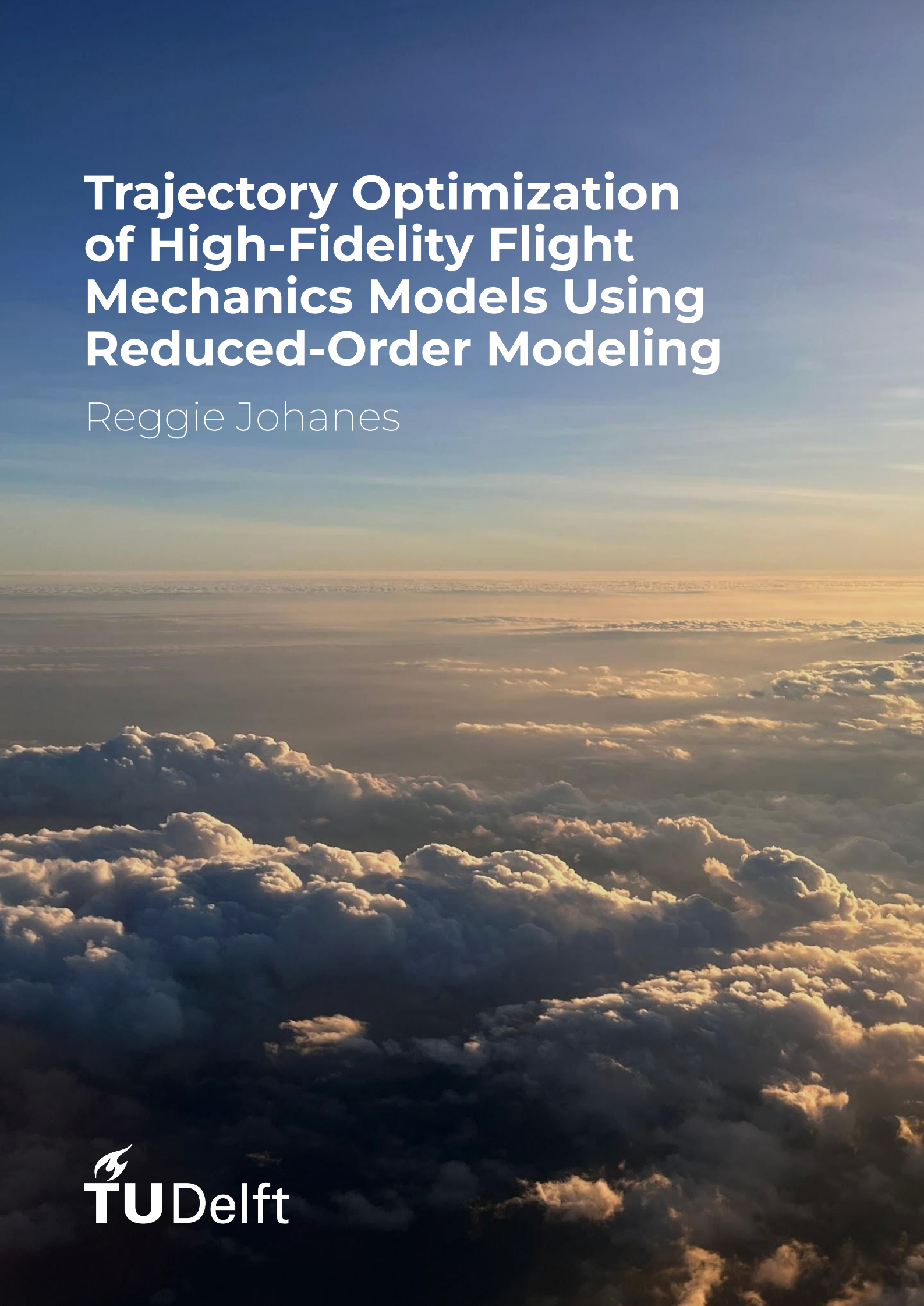

Trajectory Optimization of High-Fidelity Flight Mechanics Models Using Reduced-Order Modeling

Reggie Johanes

TU Delft

Trajectory Optimization of High-Fidelity Flight Mechanics Models Using Reduced-Order Modeling

by

Reggie Johanes

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Monday August 19, 2024.

Student number: 5477506
Project duration: September 11, 2023 – August 19, 2024
Supervisors: Dr. ir. C. Varriale, TU Delft
Dr. ir. F. Oliviero, TU Delft
Ir. J. Soikkeli, Pipistrel Aircraft

TU Delft

Contents

Nomenclature

Symbols

Symbol	Definition	Unit
c	Reference chord length	m
C_D	Drag coefficient	-
C_L	Lift coefficient	-
C_M	Pitching moment coefficient	-
C_T	Thrust coefficient	-
D	Drag force	N
F	Force	N
g	Gravitational acceleration	m/s^2
h	Altitude	m
I	Moment of inertia	kg m^2
J	Advance ratio	-
K	Controller gain	-
L	Lift force	N
M	Moment	Nm
m	Mass	kg
n	Propeller rotational speed	1/s
p	Rotational velocity in the x-axis	rad/s
q	Rotational velocity in the y-axis	rad/s
Q	Dynamic pressure	Pa
r	Rotational velocity in the z-axis	rad/s
Re	Reynolds number	-
S	Reference area	m^2
T	Thrust force	N
t	Time	s
u	Translational velocity in the x-axis	m/s
V	Airspeed	m/s
v	Translational velocity in the y-axis	m/s
W	Weight	N
w	Translational velocity in the z-axis	m/s
x	Position in the x-axis	m
y	Position in the y-axis	m
z	Position in the z-axis	m
α	Angle of attack	rad
δ	Control deflection	rad
γ	Flight path angle	rad
ϕ	Roll angle	rad
ψ	Yaw angle	rad
ρ	Density	kg/m^3
θ	Pitch angle	rad

Abbreviations

Abbreviation	Definition
CFD	Computational Fluid Dynamics
DEP	Distributed Electric Propulsion
DoF	Degree of Freedom
FCS	Flight Control System
FDD	Fault Detection and Diagnosis
GPOPS	Gauss Pseudospectral Optimization Software
HFM	High-Fidelity Model
HTU	Horizontal Thrust Unit
ICLOCS	Imperial College London Optimal Control Software
IPOPT	Interior Point Optimizer
LSED	Lock-Step Euclidean Distance
NN	Neural Network
NLP	Non-Linear Programming
NZE	Near-Zero Emission
PID	Proportional-Integral-Derivative
RoC	Rate of Climb
RoD	Rate of Descent
ROM	Reduced-Order Model
SNOPT	Sparse Nonlinear Optimizer
SoC	State of Charge
SOS	Sparse Optimization Suite
SQP	Sequential Quadratic Programming
TUD	Technische Universiteit Delft
UNIFIER	Community Friendly Miniliner

List of Figures

List of Tables

Introduction

Background and Motivation

A trajectory optimization problem is an optimal control problem that aims to find the best trajectory for a dynamic system based on a certain measure of performance while subject to a set of constraints [1]. Practical applications of trajectory optimization have been developed for a variety of fields, such as robotics [2], spacecraft [3], aircraft [4], and marine navigation [5]. In aviation, trajectory optimization has been widely utilized in the effort to minimize the negative environmental impacts associated with aircraft operations [6]. Prime examples include studies on the minimization of fuel consumption, greenhouse gas emissions, and noise impact throughout all flight phases. As the aviation industry continues to innovate towards the future of green aviation [7], trajectory optimization remains an indispensable tool to meet ambitious climate goals.

Despite its proven utility, one constraining aspect of trajectory optimization is the difficulty in performing such an analysis with a high-fidelity model, such as a 6-DoF flight dynamics model [4, 8]. This issue is exacerbated when more degrees of freedom are introduced in addition to their aircraft-level dynamic states, such is the case when subsystems such as control surface actuators are also modeled with their own degrees of freedom. In these cases, transcribing the equations of motion to solve the trajectory optimization problem becomes intractable. As a result, many studies have been conducted with lower order models such as point mass models [9, 10]. This compromise comes at the expense of physical accuracy, as lower-order models may not account for complex aerodynamic phenomena. This research project addresses this limitation by proposing a methodology incorporating reduced-order modeling that retains the most critical aerodynamic characteristics from a full-order model while allowing the resulting simplification to be manageable for a trajectory optimization solver.

The UNIFIER19 C7A aircraft model, which was previously developed as part of the UNIFIER19 project, will be the main subject of this study. The UNIFIER19 project was a joint research project commissioned by the European Union, with work being done collaboratively by Pipistrel Vertical Solutions, Politecnico di Milano, and TU Delft. The project aimed to develop an environmentally friendly 19-passenger hybrid electric aircraft with a novel propulsion system design. As such, it is envisioned to provide “an innovative near-zero emission (NZE) air mobility solution for our communities‘’ [11]. Specifically, this study will investigate the optimal trajectories for the UNIFIER C7A aircraft during landing.

Research Objective and Questions

This study aims to investigate optimal landing trajectories for the UNIFIER19 C7A aircraft. As the high-fidelity model cannot directly be used to calculate optimal trajectories, a reduced-order model of the same aircraft will be derived. The reduced-order flight mechanics model, which comprises a simplified aerodynamics and a 3-DoF flight dynamics model, will then be used to represent the system dynamics in the trajectory optimization process. To confirm that the reduced-order model maintains a sufficient degree of physical accuracy with respect to the high-fidelity model, the resulting trajectory will be tracked by the high-fidelity model. The *trackability* of the optimal trajectory will then serve as a representation of the reconstruction error of the reduced-order modeling process.

In the process of constructing the reduced-order model, a choice can be made as to the number of aerodynamic dependencies to be accounted for. For instance, in reality, the overall lift coefficient of the aircraft may be a function of a number of parameters such as its angle of attack, flap deflection, DEP advance ratio, Reynolds number, and elevator deflection. However, one or more of these dependencies may be left out to simplify the aerodynamic model and reduce computational costs. The

effect that these omissions have on the reconstruction accuracy of the reduced order model will also be investigated.

With these objectives in mind, the research question has been formulated as the following:

What is the optimal terminal approach trajectory for a hybrid-electric aircraft with a novel propulsion system such as the UNIFIER19 C7A?

To further refine this question, the following sub-questions have also been chosen:

- How can reduced-order flight mechanics models be utilized to calculate trackable optimal trajectories?
- How sensitive are the results with respect to the number of aerodynamic dependencies included in the reduced-order model?

Report Outline

This report is organized into three main parts. The first part contains the academic paper, which covers all parts of the study including results and conclusions in a journal article format. The literature review report, which was previously graded, is included in Part II. Lastly, the appendix is included to provide further practical details on the models and methods used in this study.

Academic Paper

Trajectory Optimization of High-Fidelity Flight Mechanics Models Using Reduced-Order Modeling

MSc Thesis

Reggie Johanes
Faculty of Aerospace Engineering
TU Delft

Abstract

Trajectory optimization has proven to be a powerful tool in solving a wide variety of optimal control problems in the aerospace field. However, in many cases, numerical complexities prevent the analysis of optimal trajectories for high-fidelity models, particularly due to the inherent difficulty of transcribing high-order dynamic systems. This research project proposes a methodology incorporating reduced-order modeling that retains the most critical dynamic characteristics from a full-order model while allowing the resulting simplification to be manageable for a trajectory optimization solver. The study applies this methodology to evaluate optimal landing trajectories for the UNIFIER19 C7A, a hybrid-electric aircraft equipped with a distributed electric propulsion system that was previously developed under the UNIFIER19 project. Results show that the reduced-order models generated for the aircraft can be used to generate flyable trajectories, verified by tracking the resulting landing approach paths using the base high-fidelity model. It is envisioned that this methodology will also be applicable to other aircraft models and mission phases.

Keywords

Trajectory optimization, reduced-order modeling, distributed electric propulsion

Symbols and Abbreviations

α	Angle of attack
C_L	Total lift coefficient
C_D	Total drag coefficient
C_M	Pitching moment coefficient
D	Drag force
δ	Control surface deflection
g	Gravitational acceleration
L	Lift force
m	mass
M	Pitching moment
q	Pitch rate
t	Time
θ	Pitch angle
u	Horizontal velocity, body axis
w	Vertical velocity, body axis
x	Horizontal pos., NED ref. frame
z	Vertical pos., NED ref. frame
h	Altitude, positive up
Va	True airspeed
CFD	Computational Fluid Dynamics
DEP	Distributed Electric Propulsion
DoF	Degree of Freedom
HFM	High-Fidelity Model
HTU	Horizontal Thrust Unit
$ICLOCS$	ICL Optimal Control Software
$IPOPT$	Interior Point Optimizer
$LSED$	Lock-Step Euclidean Distance
NLP	Nonlinear Programming
ROC	Rate of Climb
ROM	Reduced-Order Model

1. Introduction

A trajectory optimization problem is an optimal control problem that aims to find the best trajectory for a dynamic system based on a certain measure of performance while subject to a set of constraints [1]. Practical applications of trajectory optimization have been developed for a variety of fields, such as robotics [2], spacecraft [3], aircraft [4], and marine navigation [5]. In aviation, trajectory optimization has been widely utilized in the effort to minimize the negative environmental impacts associated with aircraft operations [6]. Prime examples include studies on the minimization of fuel consumption, greenhouse gas emissions, and noise impact throughout all flight phases. As the aviation industry continues to innovate towards the future of green aviation [7], trajectory optimization remains an indispensable tool to meet ambitious climate goals.

Despite its proven utility, one constraining aspect of trajectory optimization is the difficulty in performing such an analysis with a high-fidelity model, such as a 6-DoF flight dynamics model [4, 8]. This issue is exacerbated when more degrees of freedom are introduced in addition to aircraft-

level dynamic states, such is the case when subsystems such as control surface actuators are also modeled with their own degrees of freedom. In these cases, transcribing the equations of motion to solve the trajectory optimization problem becomes intractable. As a result, many studies have been conducted with lower order models such as point mass models [9, 10]. This compromise comes at the expense of physical accuracy, as lower-order models may not account for complex aerodynamic phenomena. This research project addresses this limitation by proposing a methodology incorporating reduced-order modeling that retains the most critical aerodynamic characteristics from a full-order model while allowing the resulting simplification to be manageable for a trajectory optimization solver.

The objective of this study is to investigate the optimal terminal approach trajectory for a hybrid-electric aircraft, in this case, the UNIFIER C7A. The reference high-fidelity model (HFM) for the UNIFIER19 C7A was previously developed as part of the UNIFIER19 project, which was a joint research project commissioned by the European Union [11]. The project aimed to develop an environmentally friendly 19-passenger hybrid electric aircraft with a novel propulsion system design, utilizing a Distributed Electric Propulsion (DEP) system.

As the high-fidelity model cannot directly be used to calculate optimal trajectories, a reduced-order model of the same aircraft will be derived. The reduced-order flight mechanics model will then be used to represent the system in the trajectory optimization process.

Multiple methods have been developed and implemented to construct these reduced-order models [12, 13, 14, 15]. A popular example among many is the use of neural networks to represent complex nonlinear systems. In the context of flight dynamics modeling, neural networks can be used to approximate aerodynamic coefficients, effectively replacing cumbersome lookup tables that contain data obtained from computational fluid dynamics (CFD) analysis or wind tunnel testing. By doing so, memory requirements are reduced while allowing differentiable approximations to be used and nonlinearities to be modeled. For this study, however, a simpler concept of reduced-order modeling is employed to allow the methodology to be easily applied to any existing high-fidelity model.

In this study, the base model is reduced in two ways. First, the aerodynamic model of the aircraft is significantly simplified. While the use of lookup tables is not eliminated, the model is reduced from a segment-based model to a singular representation of the aircraft as a whole. Second, the six-degree-of-freedom flight dynamics model is replaced with a three-degree-of-freedom representation of the system.

In the process of constructing the reduced-order model, a choice can be made as to the number of aerodynamic dependencies to be accounted for. For instance, in reality, the overall lift coefficient of the aircraft may be a function of a number of parameters such as its angle of attack, flap deflection, DEP advance ratio, and elevator deflection. However, one or more of these dependencies may be left out to simplify the aerodynamic model and reduce computational costs. The effect that these omissions have on the accuracy of the reduced order model will also be investigated.

To confirm that the reduced-order model maintains a sufficient degree of physical accuracy with respect to the high-fidelity model, the resulting trajectory will be tracked by the high-fidelity model. The *trackability* of the optimal trajectory will then serve as a representation of the so-called reconstruction error of the reduced-order modeling process.

2. Models

2.1. Reference Aircraft

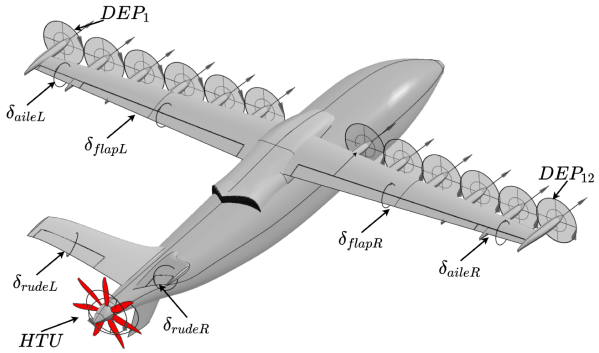

Figure 1: UNIFIER19 C7A effectors. [16].

The subject of this study is the UNIFIER19 C7A aircraft, a hybrid electric fixed-wing aircraft designed to carry nineteen passengers for commuter operations [11]. Its propulsion system comprises twelve distributed electric propulsors and one Horizontal Thrust Unit (HTU) propulsor mounted on the aft of the fuselage. It has a high-mounted, unswept main wing and a v-tail empennage configuration that houses its six control surfaces. These control surfaces are the right and left ailerons and flaps on the main wing, as

well as the pair of ruddervators for pitch and yaw control on the empennage. Figure 1 illustrates the layout of all effectors on the UNIFIER19 C7A and its top-level specifications are presented in Table 1.

Metric	Value	Unit
MTOW	7,954	kg
Passenger Capacity	19	-
Wingspan	20.11	m
Cruise Range	350	km
Cruise Altitude	1,219	m
Cruise Speed	72.74	m/s

Table 1: UNIFIER19 top-level aircraft specifications [11]

The highlight of this design is its use of the DEP. By positioning the twelve propulsors along the entire span of the main wing, the blown-wing effect is utilized to increase the aircraft's lift coefficient. Combined with the use of trailing edge flaps, this allows the aircraft to produce an increased amount of lift at relatively low airspeeds. This enhances the aircraft's short-field performance, which is especially important as the aircraft is designed as a commuter aircraft that may have to operate from smaller airfields with short runways. Additionally, the tail-mounted horizontal thrust unit (HTU) can also act as a drag generator by producing negative thrust at certain conditions. This allows the aircraft to maintain flight at even lower airspeeds, as the DEP is activated to increase lift (by increasing the dynamic pressure over the wing, i.e. producing the blown-wing effect) while the HTU produces additional drag to offset the horizontal acceleration produced by the high DEP setting. At cruise, the use of the DEP is inefficient. Therefore the DEP propellers are designed to be folded when not in takeoff/landing conditions while the HTU alone provides the necessary thrust.

The design of the UNIFIER19 C7A aircraft, including the methods used in the design process, is thoroughly documented in the project's design report [11]. As part of the project, a Simulink-based flight dynamics simulator was also constructed. The simulator is complemented with detailed aerodynamic and propulsive models, which are described in detail in the simulator documentation [17].

2.1.1. Aero-Propulsive Model

The core of the UNIFIER19 C7A flight dynamics simulator is its aerodynamics and propulsive models. Aerodynamic data was collected computationally using FlightStream, a panel method-based program. As opposed to analyzing the aircraft geometry as a whole, aerodynamic analysis was conducted on the empennage, fuselage, and main wing separately. Each component was then further broken down into segments that were analyzed individually. As a result, multi-dimensional lookup tables collected from the analysis can be used to evaluate the forces generated by each segment. The total forces and moments acting on the aircraft's center of gravity are then derived as a summation of the forces and moments generated by all individual segments. A primary advantage of this method is that it allows the forces for each segment to be evaluated during simulation based on the specific local flow conditions at that segment. By doing so, intra-component interactions, such as the effect of having the main wing blown by the DEP, can be more accurately accounted for. This capability is especially important for the UNIFIER19 C7A aircraft, as its unconventional configuration means that its aerodynamic characteristics are highly coupled. If these dependencies were to be captured by a full-model analysis without component splitting, vastly more computational resources would be needed. Therefore, this method was also done as an effort to reduce computational time.

The propulsive model comprises the DEP and HTU systems. For the DEP system, an electric motor model and propeller model are used to represent each propulsor. A unitless value termed the *activity factor* is used as the input to the model. The value of the activity factor ranges from 0 to 1, corresponding to a thrust request of zero and maximum thrust, respectively. A lookup table is then used to evaluate the proper RPM needed to achieve the desired thrust level. As the propeller operates at a fixed pitch, only one RPM will generate a specific thrust for a given airspeed. The RPM command is then fed to the electric motor model, which produces the actual RPM. Subsequently, the propeller model calculates the torque and thrust based on the motor's RPM. The advance ratio is also calculated as an input to the aerodynamic model. This allows for the aero-propulsive interactions over the main wing to be accounted for.

The HTU is modeled similarly to the DEP with two key differences. First, the activity factor for the HTU may have a negative value to represent the use of the propeller as a drag generator. Second, the HTU employs a variable pitch propeller. Consequently, calculating the optimal RPM from the activity factor is more complex

than the DEP model. To solve this, XROTOR was used to generate a 4D lookup table for the high-fidelity model. The lookup table contains HTU torque as a function of RPM, airspeed, and thrust. The torque lookup table is subsequently used to construct the power lookup table, with which the optimum RPM can be evaluated at a given airspeed and desired thrust.

As the aerodynamic and propulsive models are constructed on a component level, a complex interpolation procedure is needed to compose the total forces and moments acting on the aircraft as a whole. In summary, this procedure involves calculating local angles of attack, sideslip, and flow velocities at numerous stations on the main wing and empennage before utilizing the collected lookup tables to produce the forces and moments at each segment. Lastly, interactions between the main wing and tail as well as the aerodynamic contributions from non-lifting components are also accounted for when summing the forces and moments.

2.1.2. Control Ganging

In total, the UNIFIER19 is equipped with nineteen independently controllable effectors. This introduces a high degree of control redundancy that creates additional complexity in designing control laws and control optimization problems such as trimming and trajectory optimization. To reduce the number of effective control variables, a ganging scheme is adopted based on that used during the control law design studies conducted by Soikkeli et al. [16].

Firstly, the two ruddervators are mixed using standard control allocation schemes for v-tails to obtain explicit yaw and pitch control. Next, the flaps are assumed to be deployed symmetrically while the ailerons are deployed in opposite directions with equal deflection angles. Hence, the main wing control surfaces are reduced to two variables δ_{aile} and δ_{flap} which are defined as:

$$\delta_{\text{aile,left}} = \delta_{\text{aile}} \tag{1}$$
$$\delta_{\text{aile,right}} = -\delta_{\text{aile}} \tag{2}$$
$$\delta_{\text{flap,left}} = \delta_{\text{flap}} \tag{3}$$
$$\delta_{\text{flap,right}} = \delta_{\text{flap}} \tag{4}$$

As for the propulsion systems, the HTU thrust level, also known as the *activity factor* is kept as a separate control variable, while the controls for the DEP propulsors are ganged. Instead of twelve independent activity factors, the controls are ganged to two variables, namely DEP_{col} and $\text{DEP}_{\text{slope}}$. DEP_{col} describes the collective activity factor of the DEP system as a whole, while $\text{DEP}_{\text{slope}}$ describes the distribution of thrust differential throughout the twelve propulsors. The activity factor for each individual propulsor, denoted by DEP_i is then described by:

$$\begin{gathered}
\text{DEP}_i = \tan(\text{DEP}_{\text{slope}} K_{\text{slope}})\zeta(i) + \text{DEP}_{\text{col}} \\
i = [1, 2, 3..., 12] \\
\text{DEP}_i \in [0, 1] \\
\text{DEP}_{\text{slope}} \in [0, 1] \\
\text{DEP}_{\text{col}} \in [0, 1] \\
K_{\text{slope}} = 0.785 \\
\zeta = [-1, -0.8, -0.4, -0.2, -0.1, 0, \\
0, 0.1, 0.2, 0.4, 0.8, 1]
\end{gathered} \tag{5}$$

By implementing this scheme, the number of control variables is vastly reduced while maintaining the functions of the DEP system for thrust, lift augmentation, and directional control.

2.2. Flight Dynamics Model

2.2.1. Assumptions

The following assumptions apply:

- The aircraft is assumed to be a rigid body.
- A flat, non-rotating Earth is assumed.
- Gravity is constant.
- Undisturbed still air is assumed.
- Weight and inertia are constant.
- The aircraft's geometry and mass distributions are assumed to be symmetric with respect to the XZ plane.

Additionally, the International Standard Atmosphere model (ISA) is utilized to model atmospheric conditions.

2.2.2. 3-DoF Flight Dynamics Model

While the reference model allows for a full 6-DoF simulation, this study is limited to longitudinal flight only. Therefore, a 3-DoF flight dynamics model is used which will simulate translational motion in the x and z axes and, naturally, rotation in the longitudinal plane.
The following variables represent the aircraft's states for this model:

- x, z: The aircraft's position in the x and z axes of the flat Earth reference frame.
- u, w: Velocities in the x and z axes of the body reference frame.
- θ: Pitch angle
- q: Angular velocity in the longitudinal plane (pitch rate)

While the following control variables have been chosen as the inputs for the model:

- δ_{elev}: Elevator deflection.
- δ_{flap}: Flap deflection.
- DEP: DEP (collective) activity factor.
- HTU: HTU activity factor.

When simulating the reference model, all other control variables as described in section 2.1.2, including $\text{DEP}_{\text{slope}}$, are set to 0.

Implementation of the equations of motion for this model utilizes MATLAB Simulink's "3DOF (Body Axes)" block from the Aerospace Blockset [18]. The classic equations of motion for a 3-DoF flight mechanics model are derived from the force balance equations and moment equation. This results in the following expressions for the force and moment equations, written in the body reference frame:

$$\frac{du}{dt} = \frac{F_x}{m} - g\sin\theta - qw \tag{6}$$

$$\frac{dw}{dt} = \frac{F_z}{m} + g\cos\theta + qu \tag{7}$$

$$\frac{dq}{dt} = \frac{M_y}{I_{yy}} \tag{8}$$

where F_x and F_z are components of the resultant external force $\overrightarrow{F}$ in the x and z axes, and M_y represents the pitching moment. The resultant external force $\overrightarrow{F}$ includes the lift and drag forces, as well as the thrust and weight. The kinematics are then expressed by the following relations:

$$\frac{dx}{dt} = u\cos\theta + w\sin\theta \tag{9}$$

$$\frac{dz}{dt} = -u\sin\theta + w\cos\theta \tag{10}$$

$$\frac{d\theta}{dt} = q \tag{11}$$

3. Methodology

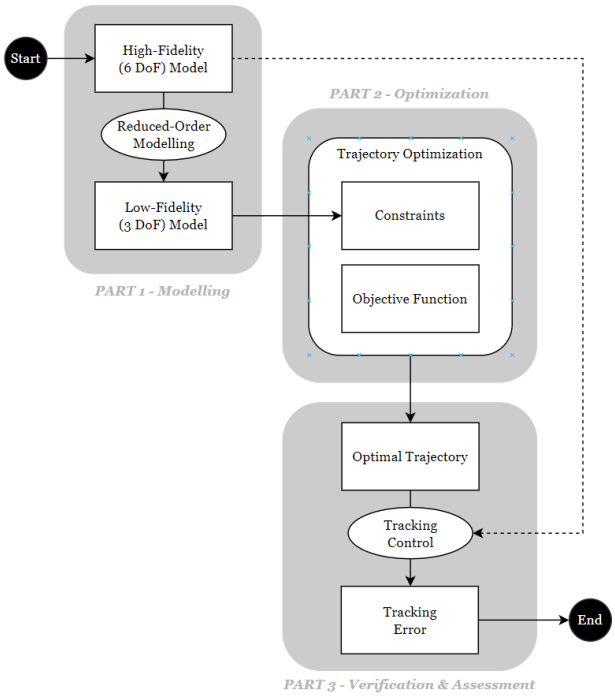

Figure 2: Research methodology

As shown in Figure 2, the research activities carried out within this thesis project are broadly grouped into three major parts: Modelling, optimization, and verification. The first part focuses on deriving the reduced-order aircraft model from the high-fidelity UNIFIER C7A model. This includes simulating the base high-fidelity model and utilizing the extracted data to construct lookup tables for the reduced-order models. The resulting reduced-order models will then serve as the dynamics constraints in the trajectory optimization problem. The optimization problem is formulated in the second part, and global trajectory optimization is performed with the ROM. The resulting trajectory is then verified in the last part, where it is tracked by the high-fidelity model using feedback control. Several versions of the reduced-order model will be constructed with varying degrees of fidelity. Hence, the entire process will be repeated for each variation.

3.1. Reduced-Order Modeling

The base High-Fidelity Model described in section 2.1.1 (henceforth referred to as the *HFM*) is simplified by reducing the aerodynamic model, while the propulsive model is used as is. This approach was chosen based on the relative complexity of the aerodynamic model, which is structured

as a summation of significantly more components compared to the propulsive model, which only requires calculations for twelve (identical) DEP propulsors and one HTU. Additionally, to account for the blown-wing effect the advance ratios of the DEP propulsors need to be calculated. As the HFM already directly converts the advance ratio to the thrust coefficient using a lookup table, it is not possible to further simplify the DEP model without losing the ability to account for aero-propulsive interactions.

To build the reduced aerodynamic model, three lookup tables are used to evaluate the lift, drag, and pitching moment coefficients. These lookup tables are constructed by simulating the HFM at varying conditions and recording the instantaneous total aerodynamic forces and pitching moment at each condition. The coefficients themselves are then obtained by simply non-dimensionalizing the total lift, drag, and pitching moment by the dynamic pressure at the condition at which the HFM is simulated, as well as the chord length in the case of the pitching moment. The execution of this procedure can be intuitively understood as conducting a "virtual wind tunnel" experiment, where the aircraft is placed at varying angles of attack and configured to a range of control surface and thrust configurations while the resulting forces are recorded. Algorithm 1 shows a simplified representation of the procedure. The conditions and inputs that are varied for this evaluation, along with their ranges and increments are shown in Table 2.

Algorithm 1 ROM Generation

Define V_a
Define h
for each δ_{flap} (index i)
  for each α (index j)
    for each DEP (index k)
      for each δ_{elev} (index l)
        $C_L[i,j,k,l] \leftarrow HFM$
        $C_D[i,j,k,l] \leftarrow HFM$
        $C_M[i,j,k,l] \leftarrow HFM$
      $J[k] \leftarrow HFM$

Variable		Unit	Min	Max	Increment
α	Angle of attack	deg	-40	40	1
δ_{elev}	Elevator deflection	deg	-30	30	5
δ_{flap}	Flap deflection	deg	0	25	5
DEP	DEP activity factor	-	0	1	0.1

Table 2: ROM evaluation input variables

While the DEP activity factor is used as an input to the ROM evaluation procedure, the advance ratio J will be used in its place as an input to the resulting lookup tables. The advance ratio produced at each DEP setting is therefore calculated and stored to be used as breakpoints in the resulting lookup tables, along with the original ranges of α, δ_{elev}, and δ_{flap}. This is done to better represent the aero-propulsive interactions as they are physically linked to the advance ratio as opposed to simply the DEP setting. A limitation of this method is that the range of advance ratios produced, and therefore the range of advance ratios at which the ROM is able to evaluate the aerodynamic coefficients, is determined by the (fixed) airspeed at which the ROM is evaluated. Varying airspeed was also considered in order to obtain a more exhaustive range of advance ratios, however, this was observed to increase the computational cost of evaluating the ROM significantly. Therefore, a compromise was chosen by evaluating the ROM at an airspeed of 50m/s, roughly halfway between the cruise speed of 72.74m/s and the stall speed of 35.85m/s, resulting in a range of advance ratios sufficient to cover the landing scenario simulated in this study.

Accounting for as many dependencies that are modeled in the high-fidelity model as possible is expected to reduce the reconstruction error of the reduced-order model. However, including more dependencies will have an effect on computational efficiency as it retains more complex behaviors of the HFM. To further investigate this trade-off, multiple versions of the ROM will be constructed with varying numbers of dependencies accounted for. The resulting reconstruction error obtained with these models will then be compared in order to gain insights into the optimal level of fidelity needed to achieve satisfactory accuracy while minimizing computational costs and allowing convergence during trajectory optimization.

The variations in dependencies will be produced by fixing the value of one or more of the independent parameters, effectively using only a subset from the complete collected dataset for a particular coefficient. Three ROM versions are generated with version 1 being the base dataset with all dependencies, while version 3 is the simplest. A summary of the different ROM versions produced and the dependencies included for each coefficient is shown in Table 3. The values at which some of the parameters are fixed for the simplified models are also shown. For both versions 2 and 3, the advance ratio is fixed to a value that would correspond to a 0.5 DEP activity factor setting.

ROM Version	Coeff.	Dependencies Included			
		α	δ_{elev}	δ_{flap}	J
v1	C_L				
	C_D				
	C_M				
v2	C_L				
	C_D				
	C_M				$J = 1.67$
v3	C_L		$\delta elev = 0$ deg	$\delta flap = 5$ deg	
	C_D				
	C_M				

Table 3: Dependencies for each ROM version

3.2. Trimming

The initial conditions for both the trajectory optimization procedure and simulation require the aircraft to be trimmed at steady, straight, and level flight. Therefore the trim states and controls need to be calculated for the HFM as well as all ROM versions. In order to do this, the trim problem can be formulated as an unconstrained optimization problem [19]. The elements of the decision vector $\mathbf{z}$ are defined as the state variables $\mathbf{x_z}$, and control variables $\mathbf{u_z}$, which are subsets of the full state and ganged control variables outlined in section 2.1.2:

$$\mathbf{z} = \begin{bmatrix} \mathbf{x_z} \\ \mathbf{u_z} \end{bmatrix} \tag{12}$$

where

$$\mathbf{x_z} = \begin{bmatrix} u \\ w \\ \theta \end{bmatrix} \tag{13}$$

$$\mathbf{u_z} = \begin{bmatrix} \delta_{\text{elev}} \\ \text{DEP}_{\text{col}} \\ \delta_{\text{HTU}} \end{bmatrix} \tag{14}$$

From the full state vector, x is removed as the horizontal position of the aircraft has no bearing on its dynamics with the model being used. By definition, the aircraft should have zero pitch rate in trimmed conditions, therefore q is also removed. z and δ_{flap} are then removed from the state and control vectors respectively as their values are imposed based on the desired condition at which the aircraft is to be trimmed.

The conditions for equilibrium flight can then be expressed as:

$$\dot{\mathbf{x_d}} = [\dot{z} \quad \dot{u} \quad \dot{w} \quad \dot{\theta} \quad \dot{q}]^T = f(\mathbf{z}) = \mathbf{0} \tag{15}$$

$\mathbf{x_d}$, a subset of the full state variable vector $\mathbf{x}$, represents the states which should be constant under steady-level flight conditions. $\dot{\mathbf{x}}_d$ are then the accelerations of those variables. f describes the dynamics of the system, which are a function of the decision variables $\mathbf{z}$. With this in mind, the cost function can then be formulated using the so-called penalty method:

$$\min_{\mathbf{z}} f_o = \mathbf{Q}^T \mathbf{H} \mathbf{Q} \tag{16}$$

where

$$\mathbf{Q} = \begin{bmatrix} \dot{\mathbf{x_d}} \\ \mathbf{c} \end{bmatrix} \tag{17}$$

$\mathbf{Q}$ comprises the state accelerations $\dot{\mathbf{x_d}}$ as well as additional relationships $\mathbf{c}$ which are imposed to achieve a specific flight condition. For steady-level longitudinal flight at a target airspeed V^*, the following flight-condition-specific relationship is applied:

$$\mathbf{c} = \begin{bmatrix} V - V^* \end{bmatrix} \tag{18}$$

This addition to the objective function is necessary as it is not possible to fix the airspeed at a certain value without also bounding the angle of attack due to the choice of state variables.

The cost function is also complemented by the penalty matrix $\mathbf{H}$. The penalty matrix is a diagonal matrix containing scalar values, which are multiplied with each element of $\mathbf{Q}$ to indicate the importance of each element towards the final objective value f_o.

$$\mathbf{H} = \begin{bmatrix} h_1 & 0 & 0 & \dots & 0 \\ 0 & h_2 & 0 & \dots & 0 \\ 0 & 0 & h_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & h_n \end{bmatrix} \tag{19}$$

Therefore, equation 16 can be rewritten as:

$$f_o = h_1 q_1^2 + h_2 q_2^2 + \cdots + h_n q_n^2 \tag{20}$$

Where each penalty matrix element h_i corresponds to element q_i of the $\mathbf{Q}$ vector. The numerical values of h_i are initially set equal to one. They are then adjusted manually based on the specific flight conditions to drive the optimizer to improve convergence behavior.

3.2.1. Solver

This optimization problem is solved using MATLAB's FMINCON, a nonlinear programming

solver [20]. The bounds for the decision variables are determined based on physical constraints (for example, the altitude must be a positive value) and control surface saturation limits. The following solver settings are used

- Algorithm : interior-point
- FinDiffType : forward

while the tolerances and the DiffMinChange setting are adjusted on a case-by-case basis depending on the target trim conditions and model used in order to improve convergence results.

3.3. Stall Speed Approximation

In order to set constraints for trajectory optimization, the stall speed of the aircraft must be determined. While the design document for the reference aircraft [11] specified the stall speed at 35.85m/s, it must be verified that the ROMs also share this limit. To achieve this, the trim routine described in section 3.2 is executed for each model at a range of airspeeds. The lowest airspeed at which the aircraft can be trimmed at steady level flight is then considered to be the approximate airspeed. The criteria for the condition to be considered as 'trimmable' is that the norm of the residual acceleration vector $\dot{\mathbf{x}}_\mathbf{d}$ must be smaller than $1e^{-3}$. It should be noted that this calculation, while similar to the objective function as described by equation 16, differs in two key ways: First, the objective function merely squares the accelerations without taking the square root. Second, the objective function includes the penalty matrix, while the norm calculation used for the stall speed approximation does not. This is done to avoid the numerical effect of the penalty factors affecting the trim criteria as described above, as different ROMs may be trimmed with different values for the penalty matrix.

3.4. Trajectory Optimization

3.4.1. Solver

The computing of the optimal trajectories itself is done using the open-source ICLOCS [21] software, which utilizes the Herminte-Simpson collocation method as its transcription method. The transcribed problem is then solved using IPOPT [22] as the NLP solver.

3.4.2. Initial and Terminal Conditions

The landing sequence begins at cruise conditions. Therefore, the initial conditions of the aircraft are as follows:

- Altitude : 1219 m
- Airspeed : 72.74 m/s
- δ_{flap} : 5 deg

while the initial conditions for the rest of the state variables are determined based on the trim solution at the above conditions. Simulations done as part of the design study for the UNIFIER C7A [11] utilized discrete jumps in the flap deflection at different phases of the landing sequence. As this would require much more complex multiphase trajectory optimization to emulate, a compromise was chosen to fix the flap deflection at a constant value throughout the trajectory.

The landing trajectory is considered complete when the aircraft reaches an altitude of 5m. Additionally, the initial guess for the terminal conditions utilizes the states and controls of the aircraft trimmed for steady level flight at 5m altitude with an airspeed of 50 m/s, after descending at an average glide slope of 1 deg.

3.4.3. Bounds

The bounds for the controls are simply the system saturation and rate limits as described by the HFM simulator documentation [17]. For the state variables, the bounds are as follows:

Variable	**Bounds**		**Unit**
	Lower	Upper	
x	0	inf	m
z	-inf	0	m
u	35cos(20)	75	m/s
w	75sin(20)	75sin(20)	m/s
θ	-5	5	deg
q	-2	2	deg/s

Table 4: Trajectory optimization bounds

The bounds for u and w were chosen such that any combination of the two variables could not result in airspeeds lower than the stall speed and/or angles of attack beyond the range of the ROM breakpoints.

3.4.4. Cost Function

For this study, two boundary cost functions will be used. First, a minimum time objective is used.

Second, a minimum distance objective will also be evaluated.

In addition to the boundary cost functions, a stage cost function is also used. The aim of the stage cost function is to promote lower airspeeds at low altitudes. This is done by taking the sum of the instantaneous airspeeds multiplied by penalty factors which increase exponentially as the aircraft descends. The stage cost function is formulated as follows:

$$\min_{\mathbf{x},\mathbf{u}} stageCost = \mathrm{kp}\frac{\mathrm{V}_a}{\mathrm{V}_{a_0}} \tag{21}$$

where

$$p = e^{\left(\frac{z-z_0}{z_0}\right)^2} \tag{22}$$

$$k = 0.1 \tag{23}$$

Fig 3 shows the change in the stage cost weight variable p as the aircraft approaches the terminal altitude.

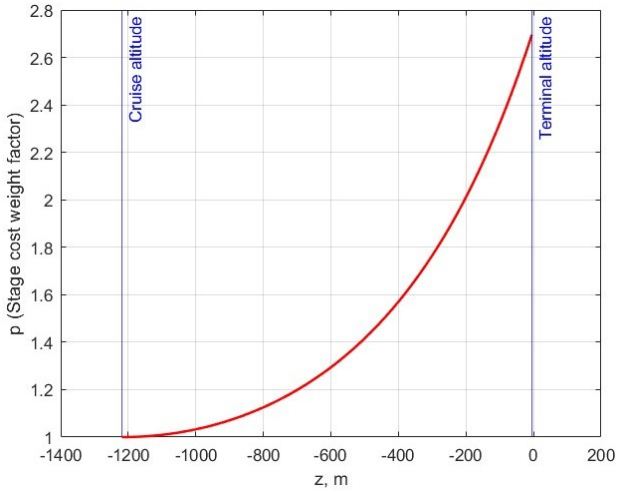

Figure 3: Stage cost weight as a function of altitude

3.4.5. Constraints

Multiple path and boundary constraints are also imposed on the problem in order to ensure a realistic solution. These constraints are summarised by Table 5.

Constraint Type	Variable	Unit	Range Min	Range Max
Path	Rate of climb	ft/min	-350	0
	Angle of attack	deg	-15	10
	True airspeed	m/s	39.44	80
Boundary	True airspeed	m/s	39.44	46.61

Table 5: Trajectory optimization bounds

The rate of climb is limited to a minimum of -350 m/s to promote ride quality by preventing excessively quick descents. It is also limited to a maximum of 0 m/s to prevent the aircraft from climbing at any point in the trajectory. Additionally, the angle of attack range was chosen in order to stay within the linear part of the lift curve slope. The airspeed is limited by two constraints. First, an inequality constraint is imposed to limit the aircraft to fly between 1.1x the stall speed to 80 m/s. This is done to prevent the aircraft from flying too close to the stall speed and to prevent excessive increases in airspeed beyond the initial cruise speed. Lastly, a narrower airspeed range is applied on the final instant, where a maximum of 1.3x of the stall speed is imposed in order to obtain realistically slow approach speeds. The boundary constraint, combined with the stage cost function described above was implemented to promote the usage of the DEP, which is a key feature of the aircraft allowing for enhanced low-speed performance.

3.5. Tracking

The final part of the study involves tracking the resulting optimal trajectory using the HFM. The trackability of the trajectory, or in other words, how closely the high-fidelity model can follow the trajectory, will then be used as a measure of the reduced-order model's reconstruction error. To achieve this, feedback control will be used to construct a tracking control system.

For the purposes of trajectory verification and assessment within the scope of this study, a relatively simple linear PID-controller-based control system will be employed.

In order to track the two-dimensional reference trajectory, the main components of the tracking system will be the longitudinal controllers, namely the airspeed and altitude tracking systems. As with the aircraft model, the flight control system will be implemented in the MATLAB Simulink environment.

3.5.1. Altitude Tracking

The altitude tracking system consists of multiple parallel control loops, which ultimately manipulate the aircraft's altitude by controlling its elevator. Figure 4 shows a block diagram for a typical altitude control system. The outermost loop receives the current altitude as feedback and compares it to the reference altitude. The altitude controller, a proportional-integral (PI) controller, then receives the altitude discrepancy, or

error and produces a target pitch angle for the inner loops. The two inner loops constitute the pitch attitude control system. Pitch angle feedback also provides an effective method to stabilize the phugoid mode of the aircraft. Therefore, the pitch attitude control system also acts as a phugoid damper.

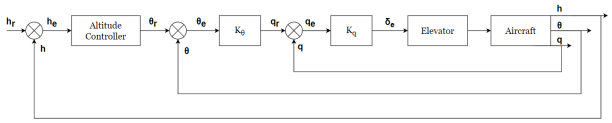

Figure 4: Overview of the altitude control system

The two inner loops were tuned using Simulink's PID Tuner App, while the outermost loop was tuned manually by trial and error, as the step response from the PID Tuner App for the outer loop produced an excessively slow settling time.

3.5.2. Airspeed Tracking

The airspeed controller is much simpler compared to the other two loops, as shown by Figure 5. Here, only a single loop is used as the airspeed typically has slow variations. A single PI controller is used to eliminate steady-state error [23]. The controller receives the airspeed error and computes the appropriate throttle input, which is then fed to the aircraft model. A limitation of this control system is that the same commands are given to both the DEP and HTU. This means that it is not possible for the controller to fully utilize the UNIFIER C7A's unique propulsion configuration, as it would require different controllers for the two independent systems to allow for better utilization during landing (e.g. by increasing the DEP activity factor while using the HTU in regenerative mode).

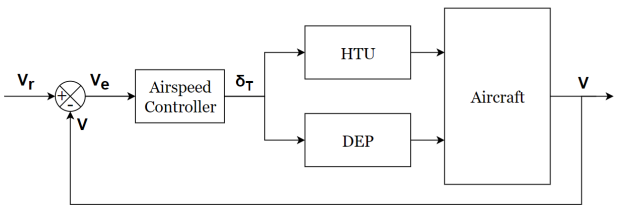

Figure 5: General structure for an airspeed control system

The airspeed controller was also tuned manually for similar reasons as the altitude controller above.

3.5.3. Tracking Error Calculation

To quantify how closely the trajectory is tracked by the HFM, a variant of the Lock-Step Euclidean Distance (LSED) similarity measurement is used. This calculation measures the average of the Euclidean distance between the reference trajectory and tracking result at the same points in time throughout the entire trajectory [24]. The main consideration for choosing LSED over other similarity measures is the fact that both the reference and tracking trajectories are aligned in time. As there are no temporal dissimilarities, there is no need for more advanced similarity measures that compensate for it. Additionally, LSED also comes with the advantage of a relatively straightforward physical interpretation.

$$\mathrm{LSED}_{\mathrm{avg}} = \frac{1}{n}\sum_{i=1}^{n} \mathrm{dist}\left(\mathrm{ref}_i, \mathrm{res}_i\right) \tag{24}$$

$$\mathrm{dist}\left(\mathrm{ref}_i, \mathrm{res}_i\right) = \sqrt{\left(x_{ref_i} - x_{res_i}\right)^2 + \left(z_{ref_i} - z_{res_i}\right)^2} \tag{25}$$

Where ref and res refer to the reference trajectory and tracking result respectively. While both the altitude and airspeed are used as top-level controller references, only the error in the geometric position of the aircraft is considered.

4. Results & Discussion

4.1. Reduced-Order Modeling

Figures 6 - 8 show a sample of the resulting lift, drag, and pitching moment coefficient curves obtained from the ROM procedure outlined in section 3.1.

It can be seen that the HFM only produces accurate aerodynamic data between angles of attack of -15 to 20 degrees, and interpolation is used beyond those limits. This is consistent with the range used for aerodynamic data collection when the HFM was constructed [17]. It is also observed that the lift coefficient curve is linear between the angles of attack of -15 to 10 degrees, which are used as constraints for the trajectory optimization procedure. An increase in flap deflection is seen to simply raise both the lift and drag coefficient curves without altering its overall profile, although the amount raised is not constant at every angle of attack. On the other hand, increasing flap deflection results in an increase in the steepness of the C_{M_α} slopes, meaning longitudinal stability is improved.

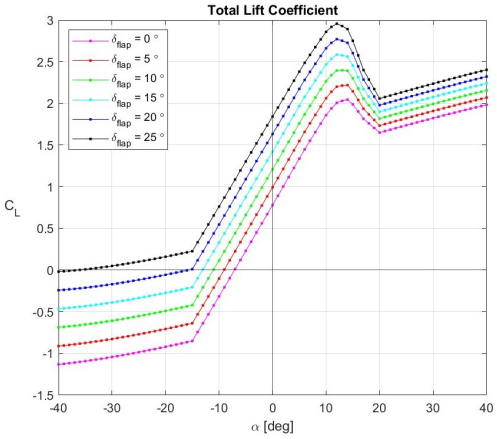

Figure 6: ROM lift coefficient curve at DEP = 0.5, $\delta_{elev} = 0$ deg

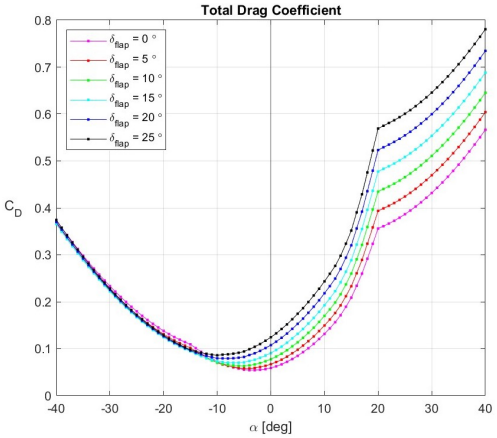

Figure 7: ROM drag coefficient curve at DEP = 0.5, $\delta_{elev} = 0$ deg

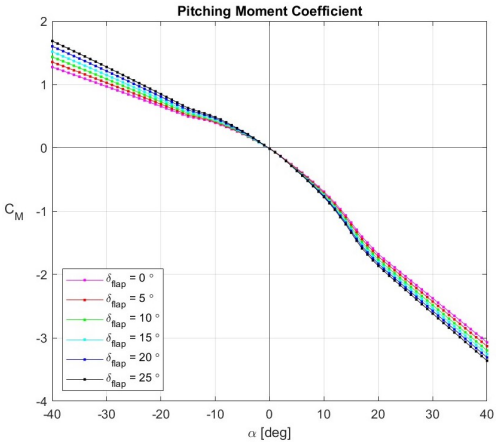

Figure 8: ROM pitching moment coefficient curve at DEP = 0.5, $\delta_{elev} = 0$ deg

Several comparisons were then conducted to study the differences between the different ROM models and also with respect to the baseline HFM. First, the aerodynamic forces and pitching moment produced with the same control inputs were compared. This comparison is done at two flight conditions: landing and cruise. The controls used are the trimmed HFM control inputs at each condition. In the case of the landing condition, the exact values for these inputs can be seen in the first row of Table 8. Tables 6 and 7 show the results of the comparisons at these two conditions respectively. In addition to the absolute values of lift, drag, and aerodynamic pitching moment, the relative error produced by each ROM is also shown as a heatmap. Cells in green indicate the smallest error produced between the three ROMs, while red indicates the largest.

ROM Version	L (N)	D (N)	$M_{y,ae}$ (Nm)
HFM	69,147.1	4,572.0	3,269.1
ROM v1	66,755.7	4,494.5	4,422.1
ROM v2	67,000.2	4,520.6	4,310.4
ROM v3	70,139.6	4,571.9	4,310.4
Error (%)			
ROM v1	-3.5%	-1.7%	+35.3%
ROM v2	-3.1%	-1.1%	+31.9%
ROM v3	+1.4%	-0.0%	+31.9%

Table 6: Comparison of lift, drag, and aerodynamic pitching moments produced by different ROM versions at landing conditions ($h = 5$ m, $Va = 50$ m/s $\delta_{flap} = 5$ deg, DEP on)

ROM Version	L (N)	D (N)	$M_{y,ae}$ (Nm)
HFM	69,101.1	5,344.6	0.0
ROM v1	69,231.3	5,423.8	-860.6
ROM v2	73,849.1	5,473.6	-4,203.5
ROM v3	91,809.3	6,234.8	-4,549.5
	Error (%)		Error (Nm)
ROM v1	+0.2%	+1.5%	-860.6
ROM v2	+6.9%	+2.4%	-4,203.5
ROM v3	+32.9%	+16.7%	-4,549.5

Table 7: Comparison of lift, drag, and aerodynamic pitching moments produced by different ROM versions at cruise conditions ($h = 1219$ m, , $Va = 72.74$ m/s $\delta_{flap} = 0$ deg, DEP off)

It is observed that while all ROM versions are able to produce reasonably similar values as the HFM, at the landing condition version 3 (the simplest) in fact most closely replicates the lift and drag forces. On the other hand, the pitching moment shows a larger relative discrepancy between the HFM and ROMs, where versions 2 and 3 produce the closest value. It is noted that the pitching moments produced by versions 2 and 3 are

identical, as the only additional dependency included for the pitching moment coefficient lookup table in version 2 (compared to version 3) is the flap deflection. While version 3 produces the most accurate results at landing conditions, the opposite is true at cruise conditions, where version 1 produces significantly more accurate values. This is most apparent when observing the values of the aerodynamic pitching moments. At cruise conditions, the DEP is deactivated. However, versions 2 and 3 are produced with a fixed 0.5 DEP activity factor setting. This means that these two versions always assume the presence of a blown-wing effect, leading to large discrepancies in conditions where the DEP is not used (or used at a different setting). Additionally, the lack of flap deflection dependency further reduces the accuracy of ROM version 3, which is produced with a fixed 5-degree flap deflection. These results show that while simplified ROM versions may indeed represent the aircraft dynamics accurately at some conditions, it comes at the cost of reduced accuracy at other conditions that are farther from the ROM's evaluation point.

A further comparison was made by simulating the HFM and ROMs with the same constant control inputs from the same initial condition (again, trim inputs for the HFM at cruise condition) for 1000 seconds. This open-loop simulation was done without any controllers, therefore the response produced is purely due to the aero-propulsive model. The results of the simulation are shown in figure 9, while figure 10 shows the same results with ROM version 3 excluded for better visibility. The first observation to be made is that the HFM is able to stay at roughly constant conditions, meaning that the aircraft is indeed trimmed. Versions 1 and 2, while producing similar responses, differ in several ways. As they produce slightly different aerodynamic forces and pitching moments under the same conditions, the trim inputs for the ROM versions are not equal to those of the HFM, nor are they equal to each other. As a result, with the same inputs the different ROM versions settle at slightly different conditions. Both versions 1 and 2 settle at constant angles of attack that are lower than the HFM. In terms of airspeed, version 1 produces a response with a constantly decreasing TAS, albeit at a very shallow slope, while the opposite is true for version 2. This results in a constantly decreasing altitude for version 1 while the simulation for version 2 shows an increase in altitude. On the other hand, the response produced by version 3 shows the loss of dynamic stability altogether, as the aircraft fails to settle at a constant condition while the amplitude of the oscillations in all states (apart from horizontal distance) continues to grow throughout the simulation.

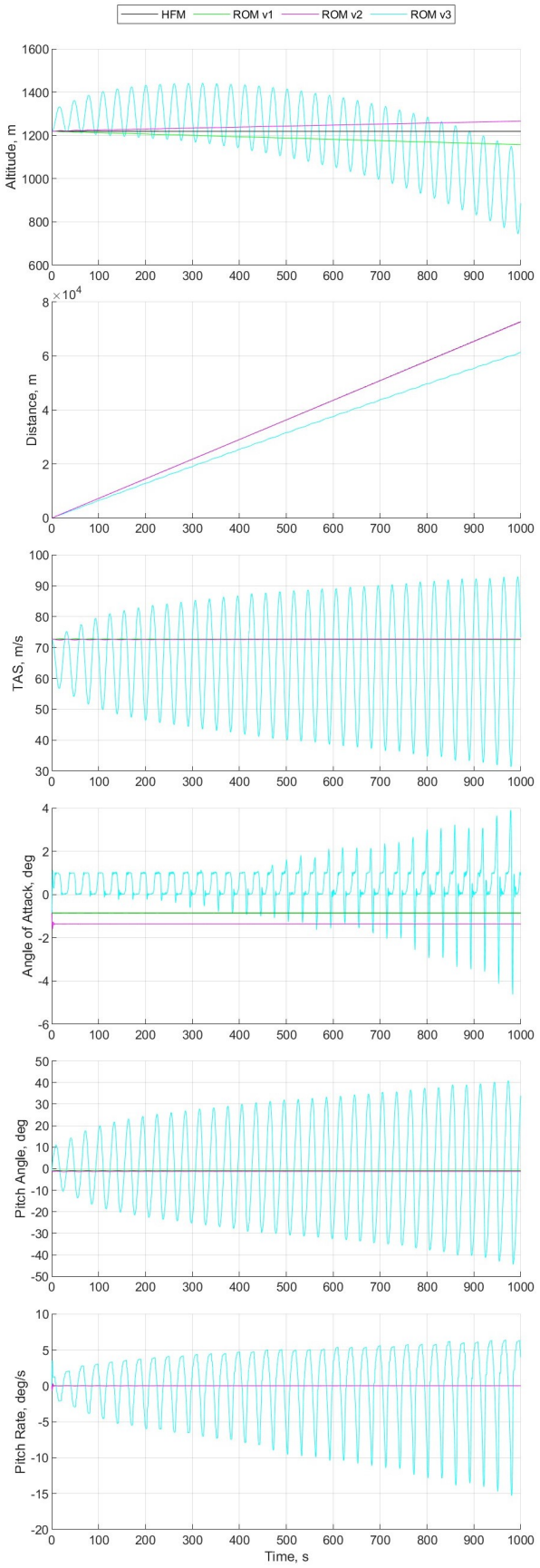

Figure 9: Comparison of simulation results produced by ROM versions 1-3

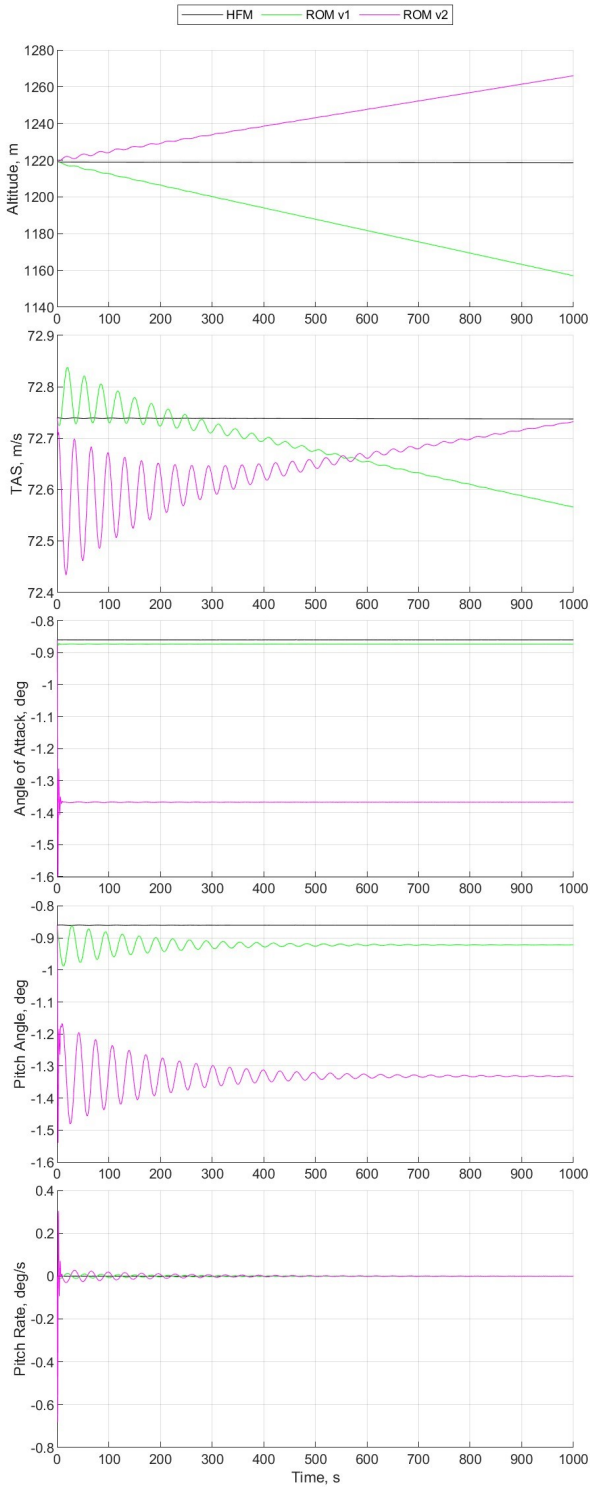

Figure 10: Comparison of simulation results produced by ROM versions 1 and 2

Each ROM version was also trimmed to compare the different inputs required to achieve the same steady-level flight at terminal landing conditions. The results are shown in Table 8. It is shown that versions 1 and 2 require a higher angle of attack compared to the HFM, while the opposite is true for version 3. This is consistent with the results shown in Fig. 6 where versions 1 and 2 produce less lift under the same conditions while version 3 produces more lift compared to the HFM. The lack of the elevator deflection dependency on the lift and drag coefficients for version 3 also results in the largest discrepancy for the elevator setting in trimmed conditions, where versions 1 and 2 are trimmed at much closer elevator deflection values.

ROM Version	α (deg)	**Controls** $\delta_{\textbf{elev}}$ (deg)	DEP (0-1)	HTU (0-1)
HFM	5.3992	-8.3459	0.4453	0.0300
ROM v1	5.9123	-8.6563	0.4263	0.0580
ROM v2	5.8320	-8.5535	0.4084	0.0754
ROM v3	5.1971	-7.6390	0.4065	0.0598

Table 8: Comparison of the angle of attack and control inputs at trimmed conditions for different ROM versions in landing conditions ($h = 5$ m, $Va = 50$ m/s $\delta_{flap} = 5$ deg, DEP on)

Based on the three comparison exercises discussed above, a few conclusions can be made. The fidelity of a reduced order model (in terms of the number of aerodynamic dependencies taken into account) determines how closely it can mimic the magnitudes of the lift, drag, and pitching moment produced by the high-fidelity model at the same flight conditions. Discrepancies in lift, drag, and pitching moment will then manifest in the aircraft's physical behavior by altering its control response. This is apparent in two ways when observing the aircraft's open-loop response. First, the four models exhibit dissimilar equilibrium points. Naturally, this also implies that their trim solutions are not interchangeable. Second, differences in the aircraft's dynamic stability are apparent. In the case of ROM v3, complete loss of dynamic stability is clearly indicated by diverging oscillations on all state variables. while this is a clear indication of reduced modeling accuracy, it should be noted that it would not necessarily create an issue in trajectory optimization. As it is obvious that unstable flight should not contribute positively toward minimizing any well-defined cost function, the solver would naturally be incentivized to find solutions that avoid this behavior, effectively resulting in stability augmentation.

Lastly, an additional comparison was done to compare the stall speeds of the different ROM versions, as outlined by the procedure described in section 3.3. The trim routine was executed for all models in a range of airspeeds at cruise (h = 1219 m, DEP off) and landing (h = 5 m, DEP on) conditions. The norms of the residual accel-

erations are then plotted and shown in figures 11 and 12.

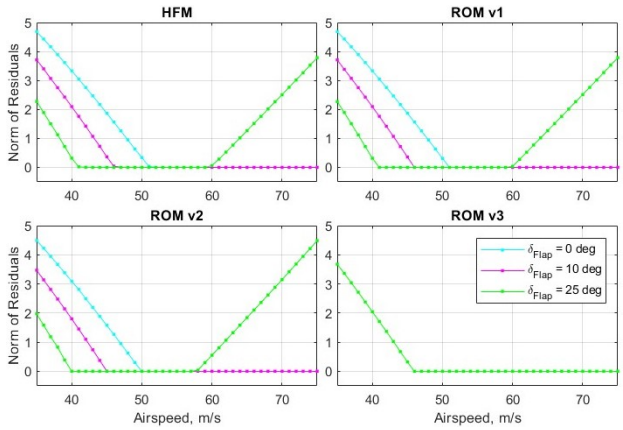

Figure 11: Norm of trim acceleration residuals, cruise conditions

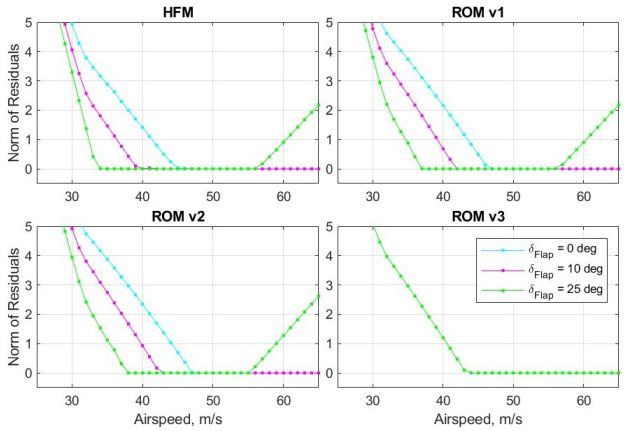

Figure 12: Norm of trim acceleration residuals, landing conditions

Based on the residual accelerations, the stall speed is evaluated. For instance, in figure 11 the HFM with a 25-degree flap deflection setting is shown to be able to produce near-zero residual accelerations down to an airspeed of 41 m/s, therefore this airspeed is determined to be the stall speed for this model at this flap deflection setting. The same procedure is repeated for all models and for all evaluated flap deflection settings. This results in a summary of the stall speed as a function of flap setting for each ROM version under cruise and landing conditions, which are shown by figures 13 and 14 respectively.

All models under both conditions show an overall trend of decreasing stall speed with increased flap deflection, which should come as no surprise. At landing conditions with the DEP on, the overall stall speed is also lowered.

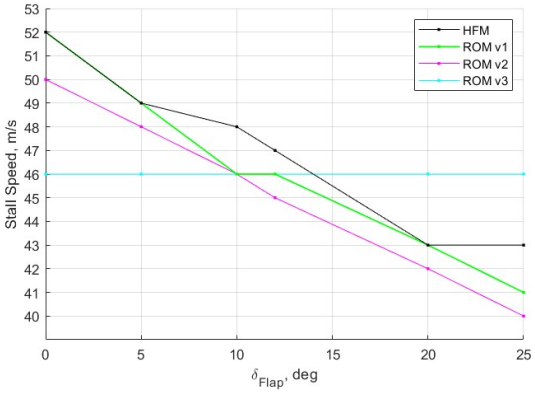

Figure 13: Stall speeds at cruise conditions (h=1219m, DEP off)

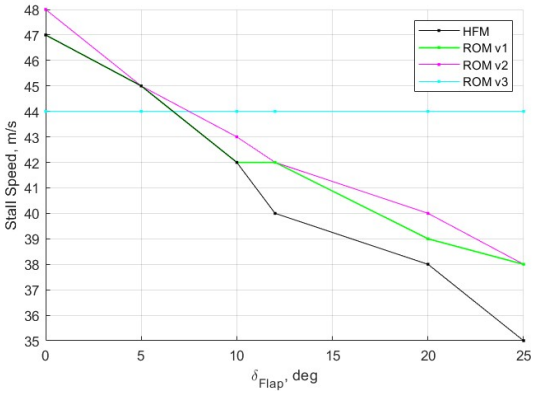

Figure 14: Stall speeds at landing conditions (h=5m, DEP on)

While not exactly equal, it is shown that the lower stall speed achievable by the HFM is 35 m/s (that is with maximum flaps and with the DEP on), similar to the stall speed indicated by the UNIFIER design report [11] of 35.85 m/s. Another observation to note is that the stall speed for ROM version 3 is not affected by the flap deflection setting, as this ROM version omits the flap deflection dependency for all aerodynamic coefficients altogether.

It should be noted that while this comparison provides a useful indication for the purposes of verifying appropriate airspeed constraints for the trajectory optimization procedure, it is by no means an exact calculation of the stall speeds. First, the precision of this method is determined by the granularity at which the airspeeds are evaluated. In this case, the trim routine is executed at increments of 1 m/s. Second, as the trim routine is done using numerical optimization methods, its results are also dependent on the robustness of the solver setup. For instance, it is possible that different numerical settings are needed for each airspeed condition in order to ob-

tain the minimum value of acceleration residuals. However, for this study, a constant numerical setting was used for all evaluations for practicality.

4.2. Trajectory Optimization

Optimal trajectories were then evaluated using the three ROM versions with the same objective functions, bounds, and constraints as outlined in section 3.4. Throughout the trajectory optimization exercise, it was observed that the convergence of the ICLOCS solver was highly sensitive to the numerical settings, bounds, and constraints that were imposed. A key numerical setting that influenced the convergence the most was the IPOPT convergence tolerance. The default value for this setting of $1e^{-8}$ was found to be too low, preventing the solver from terminating its iterations at a feasible solution. Adjustments for this setting were needed on a case-by-case basis for each optimization run, aiming to find the best trade-off between optimality and convergence by incrementally lowering the tolerance throughout multiple runs. Unfortunately, despite increasing the tolerance setting to values as high as $1e^{0}$, no converged results could be produced with the minimum distance objective function while imposing the constraints specified above. This was the case for all ROM versions. An additional measure was attempted by removing the stage cost function, however this still did not result in a converged solution. Therefore, only results produced with the minimum time cost function are shown. Additionally, multiple warnings indicating insufficient available computation memory were received, prompting a limit to 250 mesh points in order to maintain reliable convergence with acceptable solution granularity. The resulting state and control trajectories are shown in figures 15 and 16 respectively. Table 9 shows a summary of all three trajectories.

Metric	ROM v1	ROM v2	ROM v3
Final time, s	1,745.4	1064.2	684.2
Final distance, km	103.3	65.4	47.0
Avg. glide slope, deg	-0.67	-1.06	-1.48
Computation time, s	83	21	54

Table 9: Summary of trajectory optimization Results

It is observed that using ROM version 3 to represent the aircraft aerodynamics model resulted in the best trajectory, terminating the landing approach at $t_f = 684.2s$, while versions 2 and 3 resulted in considerably longer landing times. This is also reflected by the differences in the horizontal distance at which the aircraft reaches the prescribed terminal altitude as well as the average glide slopes achieved between the three ROM versions.

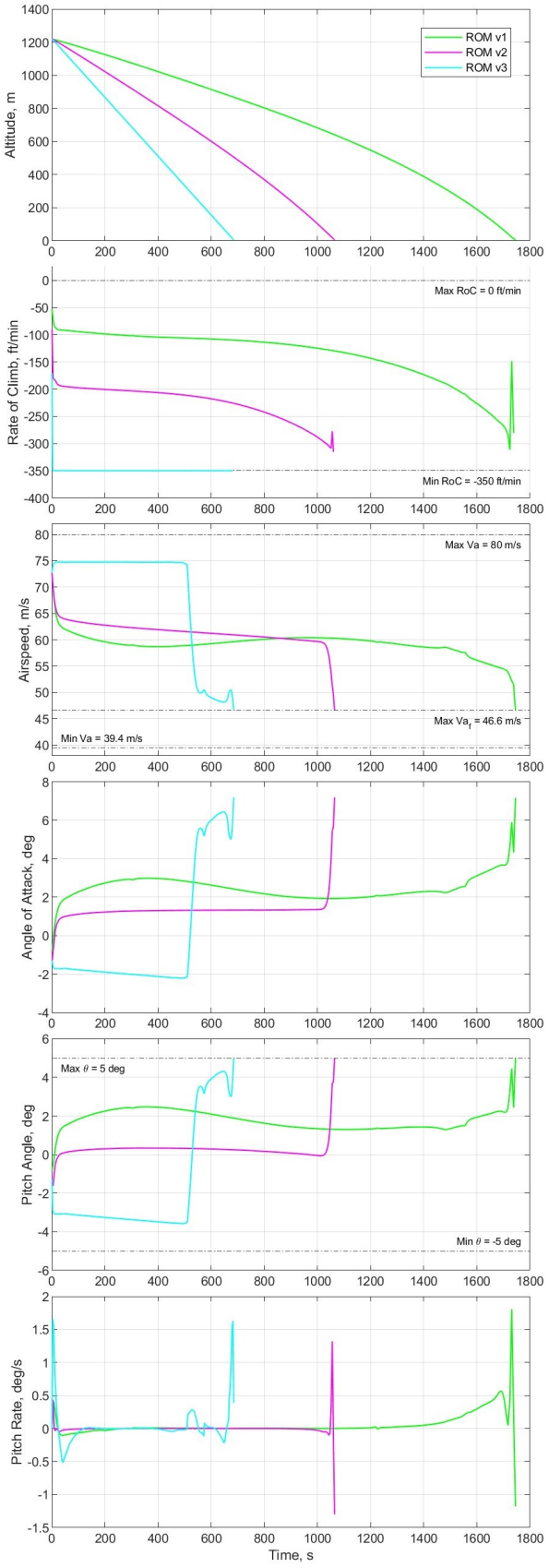

Figure 15: Optimal trajectories produced with all ROM versions, state variables

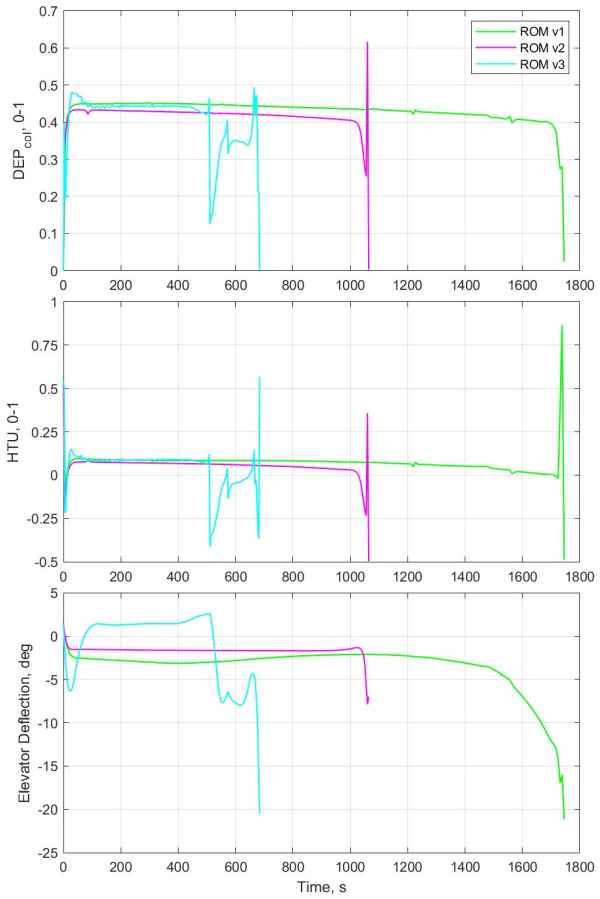

Figure 16: Optimal trajectories produced with all ROM versions, control variables

DEP and HTU usage for all three solutions present similar overall trends, where the DEP is set much higher than the HTU, which is used to produce negative thrust at the early and late stages of the trajectory. For all solutions, it is seen that the solver struggles to produce a smooth transition between the approach and terminal condition, as can be seen by the sharp drop in airspeed in the final parts of the trajectory in order to satisfy the boundary condition. The pitch angle bounds were also observed to be a limiting value. As the aircraft approaches the terminal altitude, a sharp flare maneuver is seen by deflecting the elevator which causes the aircraft to pitch up to the maximum allowable pitch angle. This maximum pitch angle was chosen to limit the aircraft to a reasonable attitude and improve passenger comfort, preventing the solver from using excessively high pitch angles.

4.3. Tracking

Finally, the three optimal trajectories presented above were tracked using the HFM equipped with the altitude and airspeed control systems as outlined in section 3.5. Figures 18 - 23 show the full tracking results for all three trajectories.

The tracking error, represented by the Euclidean distance defined by equation 25, is plotted for all three trajectories as a function of time and shown in figure 17. It is shown that while the HFM manages to track the references closely throughout a majority of the approach, sharp changes towards the start and end of the trajectories result in steep jumps in tracking error. This is an indication that the optimal trajectories resort to unrealistic maneuvers, particularly at the end of the approach as the solver tries to satisfy the imposed terminal conditions.

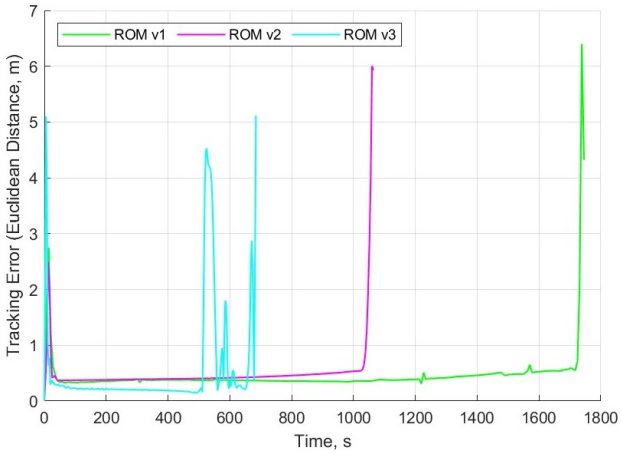

Figure 17: Tracking error comparison for all three ROM verions

Mean Tracking Error (Euclidean Distance, m)	**ROM v1** 0.4653	**ROM v2** 0.5218	**ROM v3** 0.5688

Table 10: Summary of mean tracking error for all three ROM versions versions

Lastly, the average LSED as defined by equation 24 for each of the three trajectories is shown in Table 10. As expected, there is an increase in tracking error as the number of aerodynamic dependencies is reduced in the ROM used when producing the optimal trajectories. With that being said, all trajectories are still successfully tracked relatively closely, as indicated by the $< 1m$ average tracking error values. This shows that while it is indeed true that including more aerodynamic dependencies allows for the calculation of more flyable optimal trajectories, the differences in terms of tracking error are marginal, and even the simplest ROM version was still able to produce a reasonably realistic result.

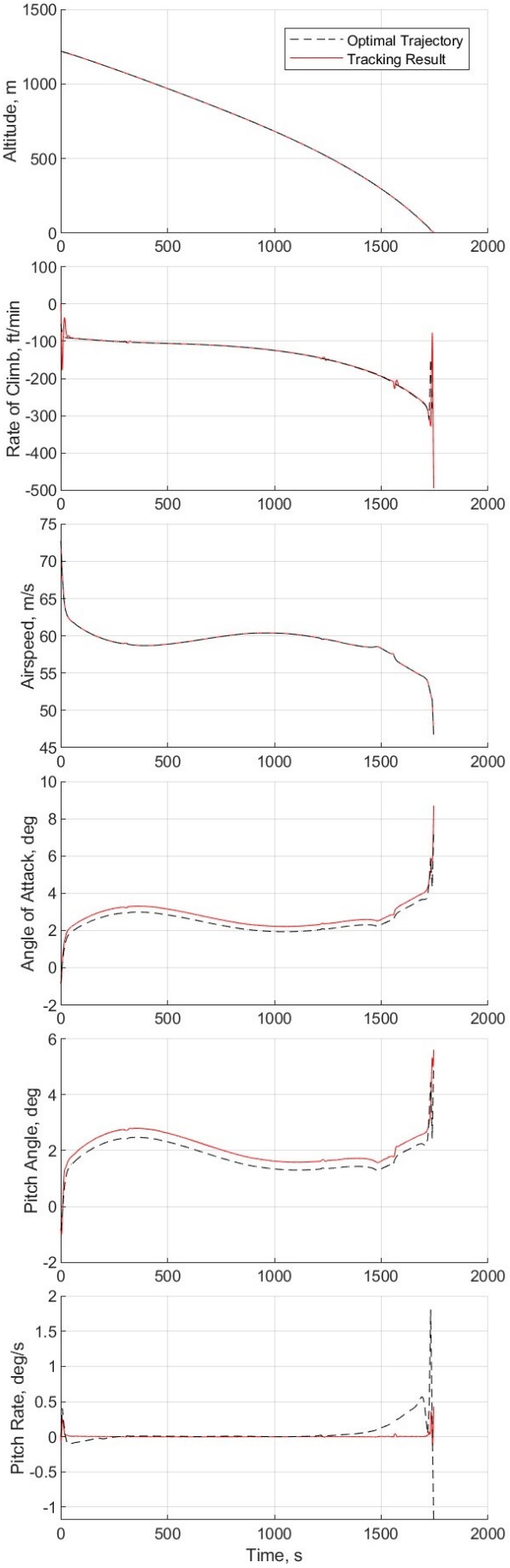

Figure 18: Tracking results for ROM version 1, state variables

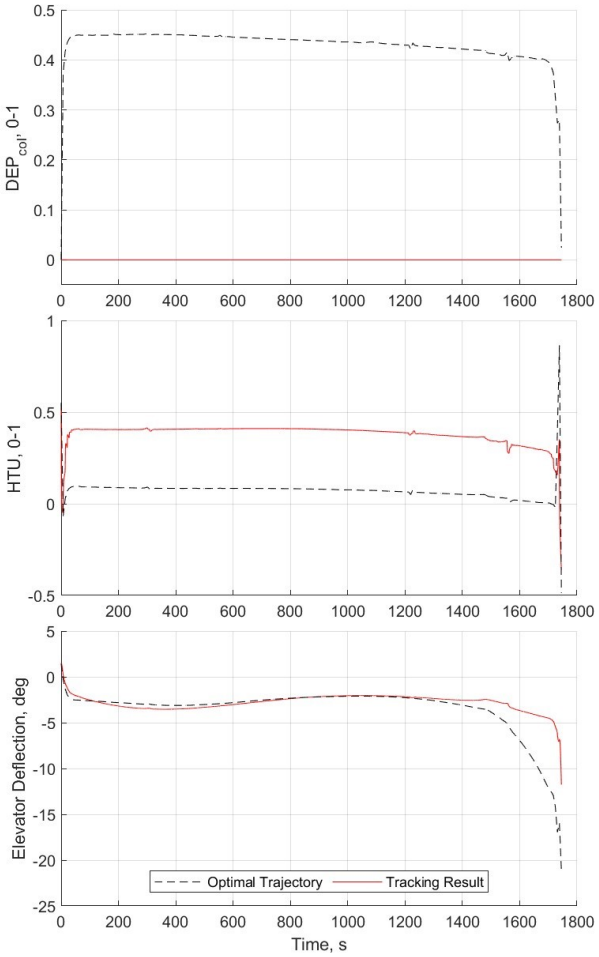

Figure 19: Tracking results for ROM version 1, control variables

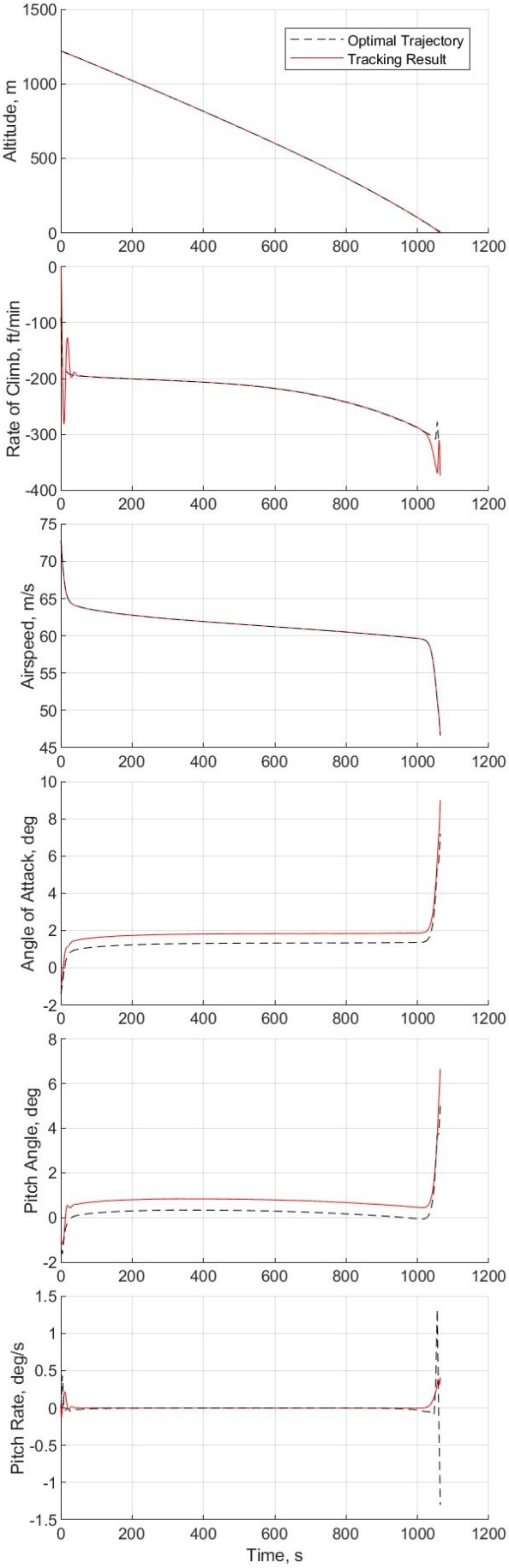

Figure 20: Tracking results for ROM version 2, state variables

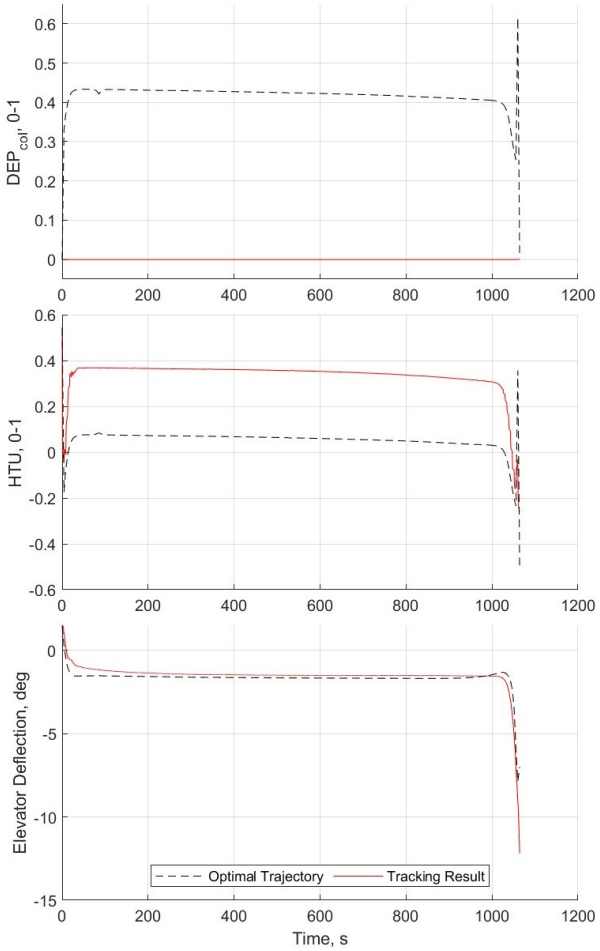

Figure 21: Tracking results for ROM version 2, control variables

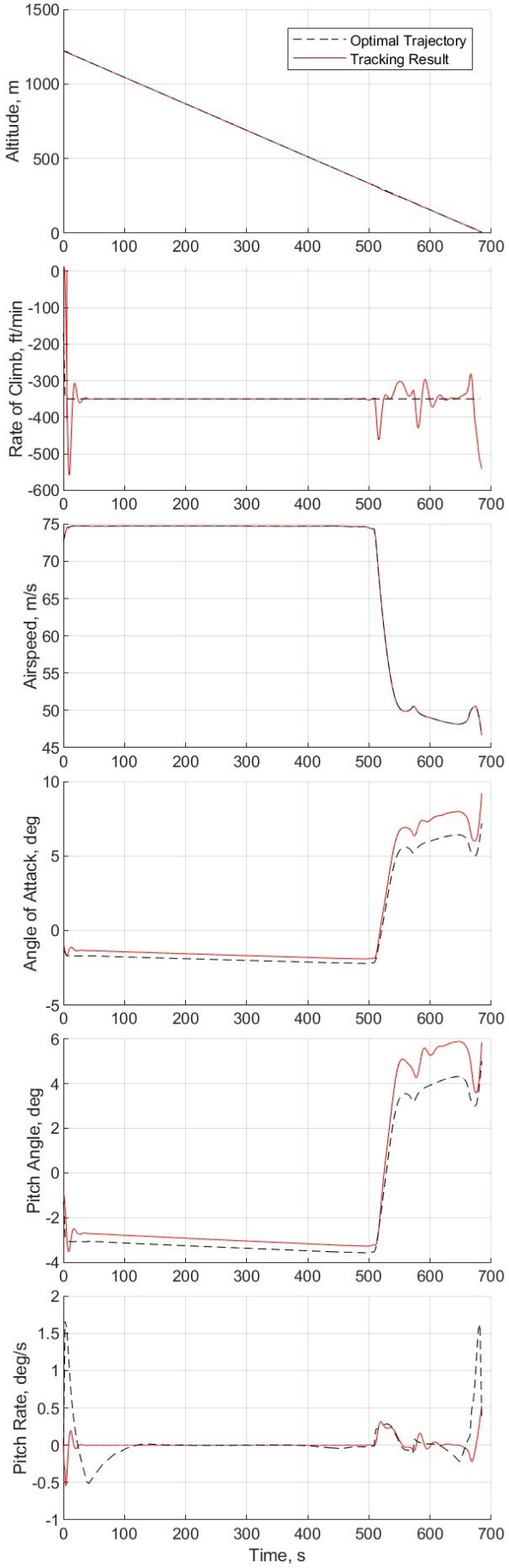

Figure 22: Tracking results for ROM version 3, state variables

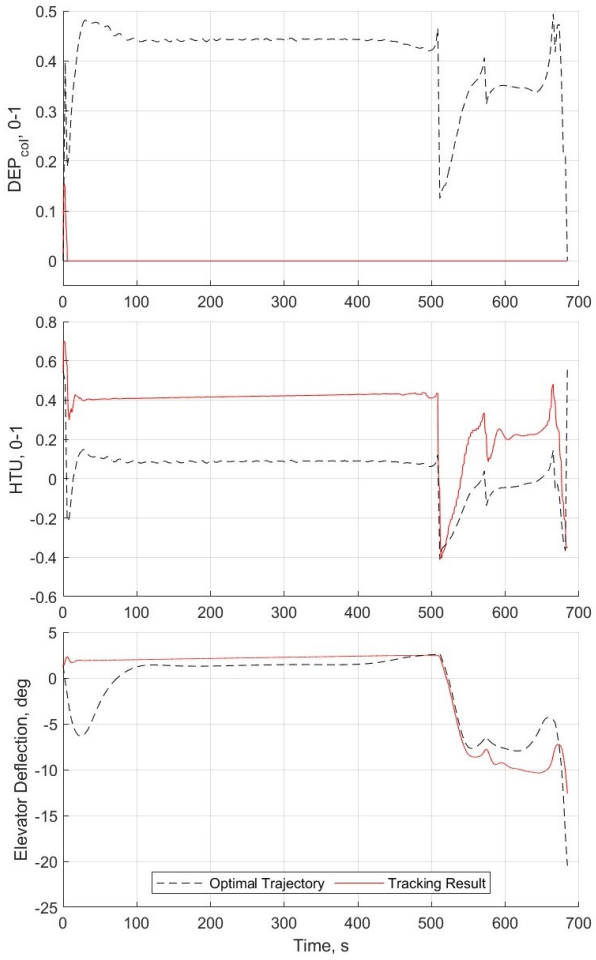

Figure 23: Tracking results for ROM version 3, control variables

5. Conclusions & Recommendations

The optimal terminal approach trajectory for the UNIFIER C7A was evaluated with a minimum time objective. Three different ROM versions with varying levels of accuracy with respect to the number of aerodynamic dependencies accounted for were used, resulting in significantly different results in terms of the time needed for the aircraft to reach the terminal landing conditions from cruise conditions. The simplest ROM version with the fewest aerodynamic dependencies resulted in the shortest approach trajectory while increasing aerodynamic dependencies (and therefore increasing model accuracy) also increased the time needed for the aircraft to complete the landing approach. While there are variations in computation time, all three solutions were generally produced quickly, with the longest calculation only taking 83 seconds.

When the resulting optimal trajectories were tracked using the HFM equipped with a PID-control-based tracking control system, it was observed that the trajectories produced with more

accurate ROMs (i.e. with more dependencies) were able to be tracked more closely. With that being said, all three resulting trajectories were still successfully tracked closely, with a mean tracking error (represented by the average LSED) of around 0.5 m for all ROM versions. The differences in trackability of the three solutions can therefore be considered marginal. Additionally, it should be noted that only a relatively simple tracking control system is used for this study. In reality, a more advanced automatic flight control system may be implemented with better tracking performance. This means that the trackability of the trajectories measured can be considered conservative.

In conclusion, it is shown that in this case, the simplest ROM version with the fewest aerodynamic dependencies is the best option as it produces the best results (in terms of landing time minimization) by a considerable margin while still maintaining a reasonably realistic trajectory.

Further research is recommended in comparing the results with different objective functions. Additionally, as the tracking results are strongly dependent on the quality of the tracking control system, more advanced flight control system designs should also be explored. In particular, for the UNIFIER C7A, it would be beneficial to utilize a more advanced throttle control system that can independently control the DEP and HTU, allowing for better utilization of the unique propulsion configuration. Applying this methodology to other aircraft models and flight scenarios is also recommended in order to verify its practicality for a wider range of applications.

References

[1] D.E. Kirk. *Optimal Control Theory: An Introduction*. Dover Books on Electrical Engineering Series. Dover Publications, 2004. ISBN: 978-0-486-43484-1.

[2] Mariana Ratiu and Mariana Adriana Prichici. "Industrial robot trajectory optimization- a review". In: *MATEC Web of Conferences* 126 (2017). Ed. by G. Grebenisan, p. 02005. ISSN: 2261-236X. DOI: 10.1051/matecconf/201712602005.

[3] Abolfazl Shirazi, Josu Ceberio, and Jose A. Lozano. "Spacecraft trajectory optimization: A review of models, objectives, approaches and solutions". In: *Progress in Aerospace Sciences* 102 (Oct. 2018), pp. 76–98. ISSN: 03760421. DOI: 10.1016/j.paerosci.2018.07.007.

[4] Runqi Chai et al. *Advanced trajectory optimization, guidance and control strategies for aerospace vehicles: methods and applications*. Singapore: Springer, 2023. ISBN: 978-981-9943-10-4.

[5] Hongchu Yu et al. "Ship Path Optimization That Accounts for Geographical Traffic Characteristics to Increase Maritime Port Safety". In: *IEEE Transactions on Intelligent Transportation Systems* 23.6 (June 2022), pp. 5765–5776. ISSN: 1524-9050, 1558-0016. DOI: 10.1109/TITS.2021.3057907.

[6] Ahmed W.A. Hammad et al. "Mathematical optimization in enhancing the sustainability of aircraft trajectory: A review". In: *International Journal of Sustainable Transportation* 14.6 (June 2020), pp. 413–436. ISSN: 1556-8318, 1556-8334. DOI: 10.1080/15568318.2019.1570403.

[7] Harijono Djojodihardjo. "Green Aircraft Technology Imperatives for Environmental Sustainability". In: *Applied Mechanics and Materials* 815 (Nov. 2015), pp. 273–281. ISSN: 1662-7482. DOI: 10.4028/www.scientific.net/AMM.815.273.

[8] Prasun N. Desai and Bruce A. Conway. "Six-Degree-of-Freedom Trajectory Optimization Using a Two-Timescale Collocation Architecture". In: *Journal of Guidance, Control, and Dynamics* 31.5 (Sept. 2008), pp. 1308–1315. ISSN: 0731-5090, 1533-3884. DOI: 10.2514/1.34020.

[9] Robert D. Falck et al. "Trajectory Optimization of Electric Aircraft Subject to Subsystem Thermal Constraints". In: *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*. Denver, Colorado: American Institute of Aeronautics and Astronautics, June 2017. ISBN: 978-1-62410-507-4. DOI: 10.2514/6.2017-4002.

[10] Fabrizio Oliviero and Kilian Swannet. "Optimal Energy Management Strategies and Mission Profiles for a generic Hybrid Aircraft". In: *AIAA AVIATION 2023 Forum*. San Diego, CA and Online: American Institute of Aeronautics and Astronautics, June 2023. ISBN: 978-1-62410-704-7. DOI: 10.2514/6.2023-4224.

[11] David Erzen et al. *UNIFIER19: Final Concurrent Design Report*. unifier19.eu. July 2021.

[12] Jack P C Kleijnen and Robert G Sargent. "A methodology for fitting and validating metamodels in simulation". In: *European Journal of Operational Research* (2000).

[13] D. Galbally et al. "Non-linear model reduction for uncertainty quantification in large-scale inverse problems". In: *International Journal for Numerical Methods in Engineering* 81.12 (2010), pp. 1581–1608. ISSN: 1097-0207. DOI: 10.1002/nme.2746.

[14] Carlo Bottasso, Alessandro Croce, and Domenico Leonello. "Two-level model-based control of flexible multibody systems". In: (Nov. 2023).

[15] S. Chen and S. A. Billings. "Neural networks for nonlinear dynamic system modelling and identification". In: *International Journal of Control* 56.2 (Aug. 1992), pp. 319–346. ISSN: 0020-7179, 1366-5820. DOI: 10.1080/00207179208934317.

[16] Johannes S. Soikkeli, Drago Matko, and Thomas Koopman. "Cascaded nonlinear dynamic inversion applied to a fixed-wing distributed electric propulsion aircraft". In: *AIAA AVIATION 2023 Forum*. American Institute of Aeronautics and Astronautics, June 2023. DOI: 10.2514/6.2023-4048.

[17] Johannes Soikkeli, Drago Matko, and Thomas Koopman. *UNIFIER19 Flight Dynamics Simulator Model Documentation*. https://github.com/soikkelij/unifier_fds. Aug. 2022.

[18] Mathworks. *MATLAB 3DOF (Body Axes)*. 2023.

[19] Christopher Lum. *AE512 Lecture Notes - Trimming a Model of a Dynamic System Using Numerical Optimization*. github.com/clum/YouTube/tree/main /FlightMech20. May 2020.

[20] Mathworks. *MATLAB Optimization Toolbox*. 2014.

[21] Imperial College London. *ICLOCS2 Optimal Control Software*. ee.ic.ac.uk/ICLOCS/default.htm.

[22] Andreas Wächter and Lorenz T. Biegler. "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming". In: *Mathematical Programming* 106.1 (Mar. 2006). DOI 10.1007/s10107-004-0559-y, pp. 25–57. ISSN: 0025-5610, 1436-4646. DOI: 10.1007/s10107-004-0559-y.

[23] E.J.J. Smeur and A Jamshidnejad. *AE4301 Automatic Flight Control System Design Course Materials*. 2023.

[24] Yaguang Tao et al. "A comparative analysis of trajectory similarity measures". In: *GIScience & Remote Sensing* 58.5 (July 2021), pp. 643–669. ISSN: 1548-1603. DOI: 10.1080/15481603.2021.1908927.

II

Literature Study Report (Previously Graded)

1 Introduction

1.1. Background and Motivation

As with any iterative design process, the ability to assess the performance of each design is crucial when designing an aircraft. A good understanding of key performance parameters allows engineers to direct their efforts more effectively to achieve design requirements. On the other hand, inaccurate assessments may lead to significant technical and commercial issues. As such, aerospace engineers are continuously working to develop tools and methodologies that can more accurately and efficiently predict aircraft dynamics. Prototyping and flight testing provide the most accurate information regarding the dynamics of a system. However, flight testing is expensive, potentially dangerous, and often not practically feasible in the early design phases. Consequently, aircraft models are constructed to predict their dynamic characteristics computationally. A key component of any aircraft model is its aerodynamics model, which cannot be constructed without aerodynamic data that may be obtained either experimentally or numerically. Additionally, Subsystem models, which represent various mechanisms of the aircraft itself, as well as environmental models also play an essential role in creating an accurate flight dynamics model. Combined, a full aircraft dynamics model can be used to simulate an aircraft in any operational situation, providing an invaluable tool for engineers to investigate its mission performance.

One implementation of aircraft dynamics models is for the analysis of optimal trajectories. Trajectory optimization is a type of optimal control problem that aims to calculate the control law needed as a function of time to obtain the best trajectory with respect to a specific objective function. Some examples of real-world aerospace applications for this type of analysis include finding the fastest climb trajectory for an interceptor aircraft to reach a certain altitude and determining a fuel-optimal ascent trajectory for a space plane. Flight trajectories obtained as a result of this type of analysis reflect the best performance an aircraft could theoretically achieve given a specific mission and are, therefore, a good indication of its performance.

A challenging aspect of trajectory optimization is the computational resources required to perform such an analysis with a high-fidelity model, such as a 6-DoF flight dynamics model [8]. This issue is exacerbated when more degrees of freedom are introduced in addition to aircraft-level dynamic states, such is the case when subsystems such as control surface actuators are also modeled with their own degrees of freedom. In these cases, transcribing the equations of motion to solve the trajectory optimization problem becomes intractable. As a result, many studies have been conducted with lower order models such as point mass models [9, 10]. This compromise comes at the expense of physical accuracy, as lower-order models may not account for more complex aerodynamic phenomena. This research project addresses this limitation by proposing a methodology incorporating reduced-order modeling that retains the most critical dynamic characteristics from a full-order model while allowing the resulting simplification to be manageable for a trajectory optimization solver.

The UNIFIER19 C7A aircraft model, which was previously developed as part of the UNIFIER19 project, will be the main subject of this study. The UNIFIER19 project was a joint research project

commissioned by the European Union, with work being done collaboratively by Pipistrel Vertical Solutions, Politecnico di Milano, and TU Delft. The project aimed to develop an environmentally friendly 19-passenger hybrid electric aircraft with a novel propulsion system design. As such, it is envisioned to provide “an innovative near-zero emission (NZE) air mobility solution for our communities‘’ [11]. Specifically, this study will investigate the optimal trajectories for the UNIFIER C7A aircraft during landing.

1.2. Research Objective and Questions

This study aims to investigate optimal landing trajectories for the UNIFIER19 C7A aircraft. As the high-fidelity model cannot directly be used to calculate optimal trajectories, a reduced-order model of the same aircraft will be derived. The reduced-order flight mechanics model, which comprises simplified aerodynamics and a 3-DoF flight dynamics model, will then be used to represent the system dynamics in the trajectory optimization process. To confirm that the reduced-order model maintains a sufficient degree of physical accuracy with respect to the high-fidelity model, the resulting trajectory will be tracked by the high-fidelity model. The *trackability* of the optimal trajectory will then serve as a representation of the so-called reconstruction error of the reduced-order modeling process.

In the process of constructing the reduced-order model, a choice can be made as to the number of aerodynamic dependencies to be accounted for. For instance, in reality, the overall lift coefficient of the aircraft may be a function of a number of parameters such as its angle of attack, flap deflection, DEP advance ratio, Reynolds number, and elevator deflection. However, one or more of these dependencies may be left out to simplify the aerodynamic model and reduce computational costs. The effect that these omissions have on the reconstruction accuracy of the reduced order model will also be investigated.

With these objectives in mind, the research question has been formulated as the following:

> *What is the optimal terminal approach trajectory for a hybrid-electric aircraft with a novel propulsion system such as the UNIFIER19 C7A?*

To further refine this question, the following sub-questions have also been chosen:

- How can reduced-order flight mechanics models be utilized to calculate trackable optimal trajectories?
- How sensitive are the results with respect to the number of aerodynamic dependencies included in the reduced-order model?

In conclusion, a methodology is proposed to couple reduced-order modeling and trajectory optimization. A reduced-order model with simplified aerodynamics will be derived from the base high-fidelity model, retaining only essential aerodynamic characteristics. Doing so significantly simplifies the trajectory optimization problem, enabling an off-the-shelf solver to produce a converged result. While the subject of this study is the UNIFIER19 C7A aircraft specifically, it is envisioned that this methodology should be applicable to other aircraft and in investigating other performance parameters or during different mission phases.

1.3. Document Outline

The reference aircraft is described in further detail in the following chapter. Chapters 3 - 5 will then cover the disciplines relevant to the methodology employed for this study. Subsequently, the research activities are synthesized into a structured research plan in Chapter 6, where a timeline containing project and content milestones are provided. Lastly, Chapter 7 provides a conclusion on the research plan.

2

Reference Aircraft

The subject of this study is the UNIFIER19 C7A aircraft, which was designed as part of the UNIFIER19 project. It is a hybrid electric fixed-wing aircraft designed to carry nineteen passengers for commuter operations. Its propulsion system comprises a distributed electric propulsion (DEP) system with twelve propulsors and one Horizontal Thrust Unit (HTU) propulsor mounted on the aft of the fuselage. It has a high-mounted, unswept main wing and a v-tail empennage configuration that houses its six control surfaces. These control surfaces are the right and left ailerons and flaps on the main wing, as well as the pair of ruddervators for pitch and yaw control on the empennage. The figure below illustrates the layout of all effectors on the UNIFIER19 C7A.

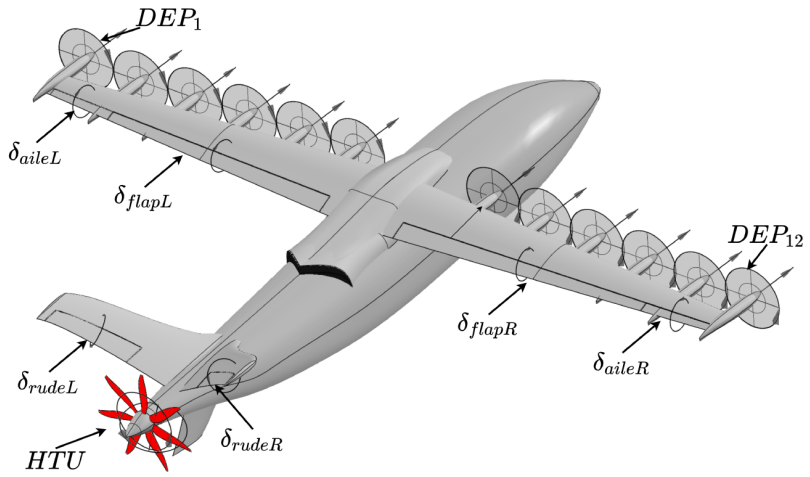

Figure 2.1: UNIFIER19 C7A effectors [12].

The design of the UNIFIER19 C7A aircraft, including the methods used in the design process, is documented in the project's design report [11]. As part of the project, a Simulink-based flight dynamics simulator was also constructed. The simulator is complemented with detailed aerodynamic and propulsive models, which are described in detail in the simulator documentation [13]. This chapter will provide an overview of the aero-propulsive model of the aircraft to establish the basis for the reduced-order modeling methodology utilized in this study. Additionally, the control ganging will also be described as it pertains to the formulation of the optimization problem.

2.1. Aero-Propulsive Model

The core of the UNIFIER19 C7A flight dynamics simulator is its aerodynamics and propulsive models. Aerodynamic data was collected computationally using FlightStream, a panel method-based program. As opposed to analyzing the aircraft geometry as a whole, aerodynamic analysis was conducted on

the empennage, fuselage, and main wing separately. Each component was then further broken down into segments that were analyzed individually. As a result, multi-dimensional lookup tables collected from the analysis can be used to evaluate the forces generated by each segment. The total forces and moments acting on the aircraft's center of gravity are then derived by combining the forces and moments generated by all individual segments. A primary advantage of this method is that it allows the forces for each segment to be evaluated during simulation based on the specific local flow conditions at that segment. By doing so, intra-component interactions, such as the effect of having the main wing blown by the DEP, can be more accurately accounted for. This capability is especially important for the UNIFIER19 C7A aircraft, as its unconventional configuration means that its aerodynamic characteristics are highly coupled. If these dependencies were to be captured by a full-model analysis without component splitting, vastly more computational resources would be needed. Therefore, this method was also done as an effort to reduce computational time.

The propulsive model comprises the DEP and HTU systems. For the DEP system, an electric motor model and propeller model are used to represent each propulsor. A unitless value termed the *activity factor* is used as the input to the model. The value of the activity factor ranges from 0 to 1, corresponding to a thrust request of zero and maximum thrust, respectively. A lookup table is then used to evaluate the proper RPM needed to achieve the desired thrust level. As the propeller operates at a fixed pitch, only one RPM will generate a specific thrust for a given airspeed. The RPM command is then fed to the electric motor model, which produces the actual RPM. Subsequently, the propeller model calculates the torque and thrust based on the motor's RPM. The advance ratio is also calculated as an input to the aerodynamic model. This allows for the aero-propulsive interactions over the main wing to be accounted for.

The HTU is modeled similarly to the DEP with two key differences. First, the activity factor for the HTU may have a negative value to represent the use of the propeller as a drag generator. Second, the HTU employs a variable pitch propeller. Consequently, calculating the optimal RPM from the activity factor is more complex compared to the DEP model. To solve this, XROTOR was used to generate a 4D lookup table which contains HTU torque as a function of RPM, airspeed, and thrust. The torque lookup table is subsequently used to construct the power lookup table, with which the optimum RPM can be evaluated at a given airspeed and desired thrust. A simplified representation of both the DEP and HTU models is shown in figure 2.2

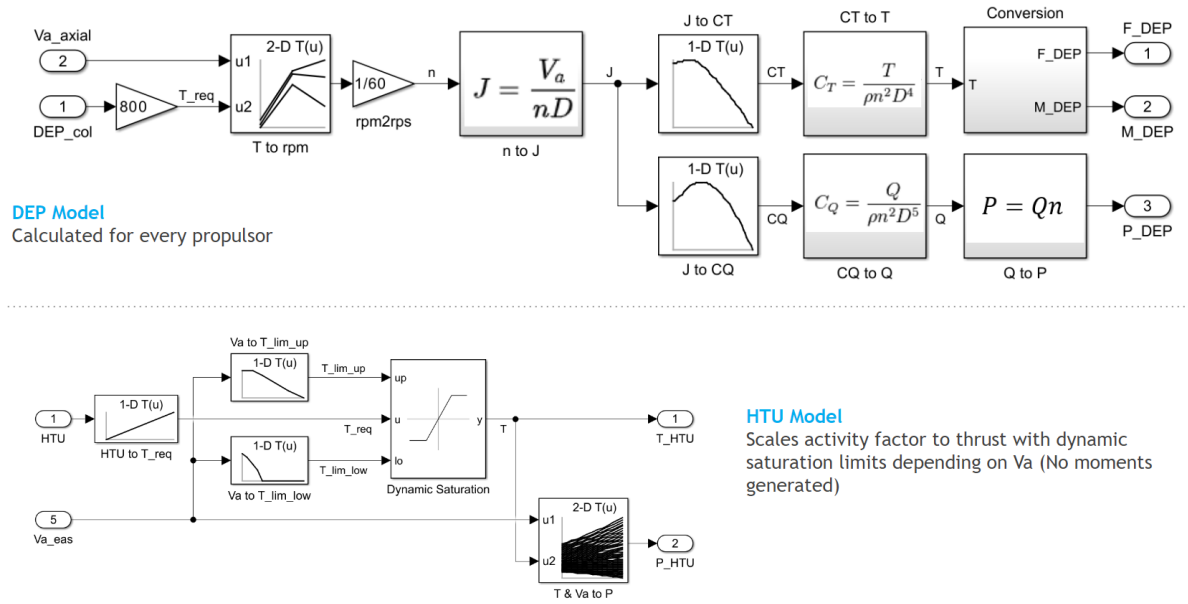

Figure 2.2: Simplified Simulink diagram of DEP and HTU models

As the aerodynamic and propulsive models are constructed on a component level, a complex interpolation procedure is needed to compose the total forces and moments acting on the aircraft as a whole. An overview of this process is shown in Figure 2.3. In summary, this procedure involves calculating local angles of attack, sideslip, and flow velocities at numerous stations on the main wing and empennage before utilizing the collected lookup tables to produce the forces and moments at each segment. Lastly, interactions between the main wing and tail as well as the aerodynamic contributions from non-lifting components are also accounted for when summing the forces and moments.

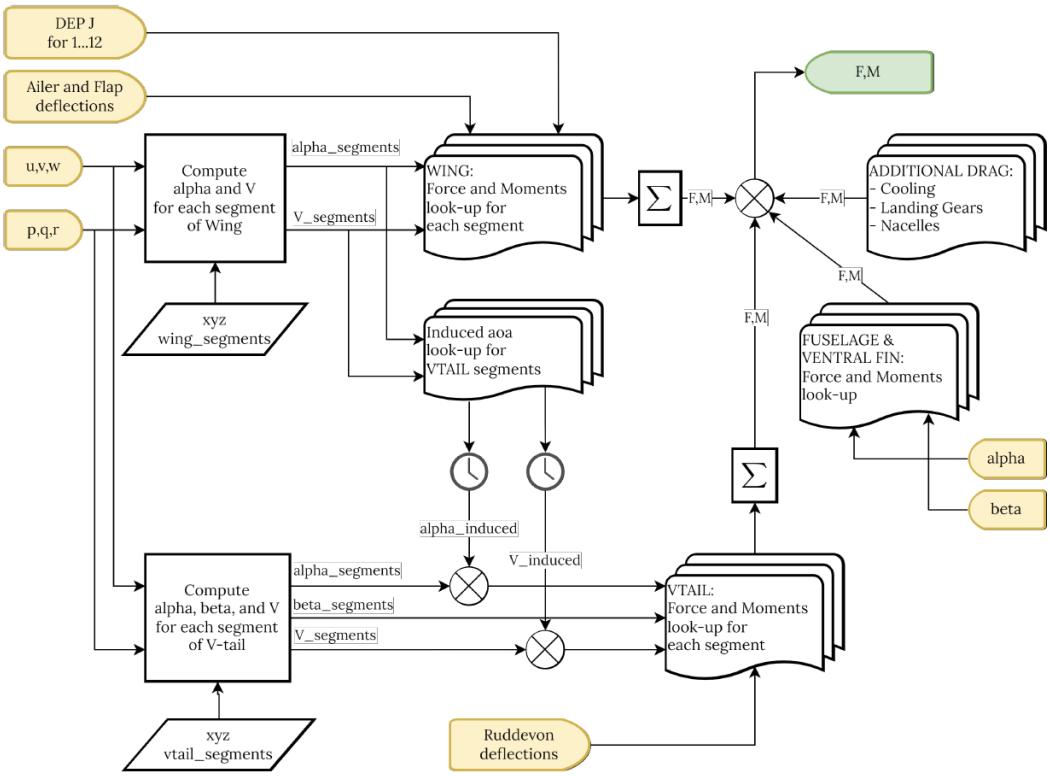

Figure 2.3: Overview of the aerodynamic data interpolation process [13]

2.2. Control Ganging

In total, the UNIFIER19 is equipped with nineteen independently controllable effectors. This introduces a high degree of control redundancy that creates additional complexity in designing control laws and control optimization problems such as trimming and trajectory optimization. To reduce the number of effective control variables, a ganging scheme is adopted based on that used during the control law design studies conducted by Soikkeli et al. [12].

Firstly, the two ruddervators are mixed using standard control allocation schemes for v-tails to obtain explicit yaw and pitch control. Next, the flaps are assumed to be deployed symmetrically while the ailerons are deployed in opposite directions with equal deflection angles. Hence, the main wing control surfaces are reduced to two variables δ_{aile} and δ_{flap} which are defined as:

$$\delta_{\text{aile,left}} = \delta_{\text{aile}} \tag{2.1}$$

$$\delta_{\text{aile,right}} = -\delta_{\text{aile}} \tag{2.2}$$

$$\delta_{\text{flap,left}} = \delta_{\text{flap}} \tag{2.3}$$

$$\delta_{\text{flap,right}} = \delta_{\text{flap}} \tag{2.4}$$

As for the propulsion systems, the HTU throttle level is kept as a separate control variable, while the DEP is ganged. Instead of twelve independent throttle levels, the controls are ganged to two variables, namely $\mathrm{DEP}_{\mathrm{col}}$ and $\mathrm{DEP}_{\mathrm{slope}}$. $\mathrm{DEP}_{\mathrm{col}}$ describes the collective thrust level of the DEP system as a whole, while $\mathrm{DEP}_{\mathrm{slope}}$ describes the distribution of thrust differential throughout the twelve propulsors. The thrust level for each individual propulsor, denoted by DEP_i is then described by:

$$\mathrm{DEP}_i = \tan(\mathrm{DEP}_{\mathrm{slope}} K_{\mathrm{slope}})\zeta(i) + \mathrm{DEP}_{\mathrm{col}}$$

$$\begin{gathered} i = [1, 2, 3..., 12] \\ \mathrm{DEP}_i \in [0, 1] \\ \mathrm{DEP}_{\mathrm{slope}} \in [0, 1] \\ \mathrm{DEP}_{\mathrm{col}} \in [0, 1] \\ K_{\mathrm{slope}} = 0.785 \\ \zeta = [-1, -0.8, -0.4, -0.2, -0.1, 0, 0, 0.1, 0.2, 0.4, 0.8, 1] \end{gathered} \tag{2.5}$$

By implementing this scheme, the number of control variables is vastly reduced while maintaining the functions of the DEP system for thrust, lift augmentation, and directional control.

3
Flight Dynamics

3.1. Assumptions

For all flight dynamics models utilized in this study, the following assumptions apply:

- The aircraft is assumed to be a rigid body.
- A flat, non-rotating Earth is assumed.
- Gravity is constant.
- Undisturbed still air is assumed.
- Weight and inertia are constant.
- The aircraft's geometry and mass distributions are assumed to be symmetric with respect to the XZ plane.

Additionally, the International Standard Atmosphere model (ISA) is utilized to model atmospheric conditions.

3.2. 6-DoF Flight Dynamics Model

The high-fidelity model for the UNIFIER19 C7A aircraft utilizes a nonlinear six-degrees-of-freedom (DoF) dynamic model to represent the aircraft's three-dimensional free motion. The following state variables represent the aircraft's states:

- u, v, w: Transnational velocities in the x, y, and z axes of the body reference frame.
- p, q, r: Rotational velocities in the x, y, and z axes of the body reference frame.
- θ, ψ, ϕ: The aircraft's pitch, yaw, and roll angles. Also known as Euler angles.
- x, y, z: The aircraft's position in the x, y, and z axes of the flat Earth reference frame.

As a result of the control ganging scheme described in the previous subchapter, the control variables are represented as the seven following expressions:

- δ_{aile}: Aileron deflection
- δ_{flap}: Flap deflection
- δ_{rudd}: Effective rudder deflection
- δ_{elev}: Effective elevator deflection
- δ_{HTU}: HTU thrust level
- DEP_{col}: DEP collective thrust level
- $\text{DEP}_{\text{slope}}$: DEP thrust differential distribution

3.2.1. Equations of Motion

As previously mentioned, the high-fidelity model allows free movement in three-dimensional space with six degrees of freedom. The equations of motion for a typical aircraft are well established and are provided by Ruijgrok in his textbook [14]. They are comprised of the force equations:

$$-W \sin\theta + F_x = m\left(\frac{du}{dt} + wq - vr\right) \tag{3.1}$$

$$W \cos\theta \sin\phi + F_y = m\left(\frac{dv}{dt} + ur - wp\right) \tag{3.2}$$

$$W \cos\theta \cos\phi + F_z = m\left(\frac{dw}{dt} + vp - uq\right) \tag{3.3}$$

where F_x, F_y, and F_z are components of the resultant external force $\overrightarrow{F}$ which includes the aerodynamic forces $\overrightarrow{R}$ and thrust $\overrightarrow{T}$.

$$\overrightarrow{F} = \overrightarrow{R} + \overrightarrow{T} \tag{3.4}$$

followed by the moment equations,

$$M_x = \frac{dp}{dt} I_x + (I_z - I_y)qr + I_{xz}\left(\frac{dr}{dt} + pq\right) \tag{3.5}$$

$$M_y = \frac{dq}{dt} I_y + (I_x - I_z)pr + I_{xz}\left(p^2 - r^2\right) \tag{3.6}$$

$$M_z = \frac{dr}{dt} I_z + (I_y - I_x)pq + I_{xz}\left(\frac{dp}{dt} + qr\right) \tag{3.7}$$

where M_x, M_y, and M_z represent the rolling, pitching, and yawing moments respectively. It should be noted that these moment equations are the result of simplification based on the assumption of symmetric mass distribution about the XZ plane.

Finally, the kinematic relations are expressed by:

$$p = -\frac{d\psi}{dt} \sin\theta + \frac{d\phi}{dt} \tag{3.8}$$

$$q = \frac{d\psi}{dt} \cos\theta \sin\phi + \frac{d\theta}{dt} \cos\phi \tag{3.9}$$

$$r = \frac{d\psi}{dt} \cos\theta \cos\phi - \frac{d\theta}{dt} \sin\phi \tag{3.10}$$

The computational implementation of these equations for this study utilizes MATLAB Simulink's "6DoF (Euler Angles)" block from the Aerospace blockset. The block requires the total forces and moments acting on the aircraft in the body axis as inputs, along with information regarding the aircraft's mass, inertia, and initial conditions. The states and accelerations are then given as outputs.

3.2.2. Trimming

An essential step in flight dynamics modeling and simulation is the calculation of trim conditions. Trim points are calculated by determining the appropriate state and controls of the aircraft that are required to achieve a desired instantaneous flight condition. A commonly calculated operating condition is equilibrium flight, where the aircraft is flying at constant conditions, or in other words, with no accelerations. It is necessary to calculate this condition for many applications. For instance, flight simulations are typically started in equilibrium conditions. Therefore, the trim point needs to be known before the rest of the simulation can be executed.

For simple flight dynamics models, the states and controls required for these trim points may be calculated analytically. However, for a six-DoF nonlinear model as used for the high-fidelity UNIFIER19 C7A model, the complex system of equations makes it untractable to solve for these trim points with analytical methods. Additionally, the presence of redundant controls, as is the case for the UNIFIER19 C7A aircraft, creates an under-determined trim problem, adding a layer of complexity. To overcome this, the trim problem can be solved numerically.

The trim problem can be formulated as an unconstrained optimization problem. The elements of the decision vector **z** are defined as the state variables **x**, and control variables **u**, which are the result of the control ganging scheme outlined in section 2.2:

$$\mathbf{z} = \begin{bmatrix} \mathbf{x} \\ \mathbf{u} \end{bmatrix} \tag{3.11}$$

where

$$\mathbf{x} = \begin{bmatrix} u \\ v \\ w \\ p \\ q \\ r \\ \phi \\ \theta \\ \psi \\ x \\ y \\ z \end{bmatrix} \tag{3.12}$$

$$\mathbf{u} = \begin{bmatrix} \delta_{\text{aile}} \\ \delta_{\text{flap}} \\ \delta_{\text{rudd}} \\ \delta_{\text{elev}} \\ \delta_{\text{HTU}} \\ \text{DEP}_{\text{col}} \\ \text{DEP}_{\text{slope}} \end{bmatrix} \tag{3.13}$$

The conditions for equilibrium flight can then be expressed as:

$$\dot{\mathbf{x_d}} = [\dot{u} \quad \dot{v} \quad \dot{w} \quad \dot{p} \quad \dot{q} \quad \dot{r}]^T = f(\mathbf{z}) = \mathbf{0} \tag{3.14}$$

$\mathbf{x_d}$, a subset of the full state variable vector **x**, represents the states which should be constant under steady-level flight conditions. $\dot{\mathbf{x}}_d$ are then the accelerations of those variables. f describes the dynamics of the system, which are a function of the decision variables **z**. With this in mind, the cost function can then be formulated using the so-called penalty method:

$$\min_{\mathbf{z}} f_o = \mathbf{Q}^T \mathbf{H} \mathbf{Q} \tag{3.15}$$

where

$$\mathbf{Q} = \begin{bmatrix} \dot{\mathbf{x_d}} \\ \mathbf{c} \end{bmatrix} \tag{3.16}$$

Q comprises the state accelerations $\dot{\mathbf{x}}_\mathbf{d}$ as well as additional relationships **c** which are imposed to achieve a specific flight condition. For instance, for steady-level longitudinal flight at a target airspeed V^*, the following flight-condition-specific relationship vector may be applied:

$$\mathbf{c} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ \dot{y} \\ \dot{z} \\ \phi \\ \psi \\ V - V^* \end{bmatrix} \tag{3.17}$$

The relationship vector above is formulated such that each value will be equal to zero when the desired conditions are reached. Therefore, each element represents the following flight conditions:

- Euler angle accelerations $\dot{\phi}$, $\dot{\theta}$, and $\dot{\psi} = 0$
- Lateral velocity $\dot{y} = 0$
- Climb rate $\dot{z} = 0$
- Bank angle ϕ = 0
- Heading ψ = 0
- Airspeed $V = V^*$

As the above conditions are embedded in the cost function, a drawback of this formulation is that the resulting solution may not exactly satisfy the desired conditions due to residual errors. A further refinement of this general formulation is presented in the following section which addresses this limitation while also simplifying the optimization problem.

Lastly, the cost function is also complemented by the penalty matrix **H**. The penalty matrix is a diagonal matrix containing scalar values, which are multiplied with each element of **Q** to indicate the importance of each element towards the final objective value f_o.

$$\mathbf{H} = \begin{bmatrix} h_1 & 0 & 0 & \dots & 0 \\ 0 & h_2 & 0 & \dots & 0 \\ 0 & 0 & h_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & h_n \end{bmatrix} \tag{3.18}$$

Therefore, equation 3.15 can be rewritten as:

$$f_o = h_1 q_1^2 + h_2 q_2^2 + \cdots + h_n q_n^2 \tag{3.19}$$

Where each penalty matrix element h_i corresponds to element q_i of the **Q** vector. The numerical values of h_i are initially set equal to one. They are then adjusted manually based on the specific flight conditions to drive the optimizer to improve convergence behavior.

Decision Vector Reduction

For some flight conditions, the desired values of several decision variables are known and, therefore, do not need to be calculated as part of the trim problem. For instance, by definition, the following must apply for straight, level, longitudinal flight:

$$\begin{bmatrix} v & p & q & r & \phi & \psi \end{bmatrix} = \mathbf{0} \tag{3.20}$$

The values of x and y will not affect the results of the trim problem as no geo-referenced model is used. Therefore, they may also be set equal to zero to indicate a starting position in the inertial reference frame. Additionally, if coordinated flight with symmetric thrust is assumed, δ_{rudd}, δ_{aile}, and DEP_{slope} can also be removed from the optimization problem. Lastly, z and δ_{flap} can be removed from the decision vector as their values are imposed based on the desired condition at which the aircraft is to be trimmed. This results in the following reduced decision vector $\mathbf{z}'$:

$$\mathbf{z}' = \begin{bmatrix} u \\ w \\ \theta \\ \delta_{\text{elev}} \\ \delta_{\text{HTU}} \\ \text{DEP}_{\text{col}} \end{bmatrix} \tag{3.21}$$

It can be seen that for this particular case, the number of decision variables can be vastly reduced. Doing so simplifies the optimization problem and substantially minimizes the computational effort required.

Solver

This optimization problem is solved using MATLAB's FMINCON, a nonlinear programming solver [15]. The bounds for the decision variables are determined on the basis of physical constraints (for example, the altitude must be a positive value) and control surface saturation limits. The following solver settings will initially be used. However, it should be noted that adjustments may be required to improve convergence behavior during execution.

- Algorithm : Interior Point
- TolFun : 1e-6
- TolX : 1e-6
- FinDiffType : forward
- DiffMinChange : 1e-5
- DiffMaxChange : inf

3.3. 3-DoF Flight Dynamics Model

A longitudinal flight dynamics model will be used for the reduced-order aircraft model. This model utilizes a classical derivation that will allow free two-dimensional movement with three degrees of freedom. As opposed to 6-DoF dynamics models, which account for forces and moments in all three axes and planes, this model will only account for translational motion in the x and z axes and, naturally, moments in the longitudinal plane.

The following variables represent the aircraft's states for this model:

- u, w: Velocities in the x and z axes of the body reference frame.
- q: Angular acceleration in the longitudinal plane
- x, z: The aircraft's position in the x and z axes of the flat Earth reference frame.
- θ: Pitch angle

While the following control variables have been chosen as the inputs for the model:

- δ_T: Thrust level.
- δ_{elev}: Elevator deflection.

3.3.1. Equations of Motion

Implementation of the equations of motion for this model utilizes MATLAB Simulink's "3DOF (Body Axes)" block from the Aerospace Blockset [16]. The classic equations of motion for a 3-DoF flight mechanics model are derived from the force balance equations and moment equation. This results in the following expressions for the force and moment equations, written in the body reference frame:

$$\frac{du}{dt} = \frac{F_x}{m} - g\sin\theta - qw \tag{3.22}$$

$$\frac{dw}{dt} = \frac{F_z}{m} + g\cos\theta + qu \tag{3.23}$$

$$\frac{dq}{dt} = \frac{M_y}{I_{yy}} \tag{3.24}$$

where F_x and F_z are components of the resultant external force $\overrightarrow{F}$ in the x and z axes, and M_y represents the pitching moment. The resultant external force $\overrightarrow{F}$ includes the lift and drag forces, as well as the thrust $\overrightarrow{T}$, which is assumed to act in the x-axis of the body reference frame. The kinematics are then expressed by the following relations:

$$\frac{dx}{dt} = u\cos\theta + w\sin\theta \tag{3.25}$$

$$\frac{dz}{dt} = -u\sin\theta + w\cos\theta \tag{3.26}$$

$$\frac{d\theta}{dt} = q \tag{3.27}$$

4

Control Theory

4.1. Trajectory Optimization

Trajectory optimization allows us to obtain the optimal flight path for a particular criterion (or multiple criteria). If it can be confidently established that the resulting trajectory is indeed the best possible flight path for the given aircraft, conditions, and objective, then the results will also serve as a measure of the aircraft's performance limits. Therefore, in this context trajectory optimization also serves as a performance assessment tool.

To obtain an optimal trajectory, the optimal state and control trajectories are determined to minimize the value of a specific objective function. For continuous-time problems, the optimal state and control trajectories $\mathbf{x}^*(t)$ and $\mathbf{u}^*(t)$ may be continuously varying functions of time. The cost function $J(\mathbf{x}(t), \mathbf{u}(t))$ is then referred to as a *functional*, or in other words, a function of functions, as it calculates the objective value with the state and control functions as inputs. In its general form shown in equation 4.1, the cost function comprises two terms: the boundary objective E and the integral objective L, also known as the Mayer and Lagrange terms respectively [17].

$$\min_{\mathbf{x}(t),\mathbf{u}(t)} J(t_0, t_f, \mathbf{x}(t), \mathbf{u}(t)) = E(\mathbf{x}(t_0), \mathbf{x}(t_f), \mathbf{u}(t_0), \mathbf{u}(t_f), t_0, t_f) + \int_{t_0}^{t_f} L(\mathbf{x}(t), \mathbf{u}(t))dt \tag{4.1}$$

The optimization problem is subject to several constraints, shown in equations 4.2 - 4.4. First, the system dynamics are imposed to dictate how the system changes in time:

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t)) \tag{4.2}$$

Path constraints may also be enforced along the trajectory. A common example would be limiting the horizontal position of the aircraft above the ground.

$$h(\mathbf{x}(t), \mathbf{u}(t)) \leq 0 \tag{4.3}$$

Additionally, the initial and final states of the system may also be restricted by imposing nonlinear boundary constraints:

$$g(t_0, t_f, \mathbf{x}(t_0), \mathbf{x}(t_f)) \leq 0 \tag{4.4}$$

In addition to the constraints, constant limits may be applied to the states or controls, as shown by equations 4.5 and 4.6 respectively. For example, this may represent the maximum control surface deflections that the aircraft is capable of providing.

$$\mathbf{x}_{\text{low}} \leq \mathbf{x}(t) \leq \mathbf{x}_{\text{upp}} \tag{4.5}$$
$$\mathbf{u}_{\text{low}} \leq \mathbf{u}(t) \leq \mathbf{u}_{\text{upp}} \tag{4.6}$$

Lastly, exact limits may be imposed on the initial and final time and states. This is desirable when it is required that the trajectory starts and finishes at specific points in space, or that a specific time window is adhered to.

$$t_{\text{low}} \leq t_0 \leq t_f \leq t_{\text{upp}} \tag{4.7}$$
$$\mathbf{x}_{0,\text{low}} \leq \mathbf{x}(t_0) \leq \mathbf{x}_{0,\text{upp}} \tag{4.8}$$
$$\mathbf{x}_{\text{f,low}} \leq \mathbf{x}(t_f) \leq \mathbf{x}_{\text{f,upp}} \tag{4.9}$$

A fundamental step in solving any trajectory optimization problem is transcription. Transcription refers to the process of transforming the trajectory optimization problem into a constrained parameter optimization problem. When the resulting optimization problem has nonlinear terms within its objective or constraint functions, the trajectory optimization problem is said to have been transcribed into a nonlinear program. Nonlinear programs are formulated to minimize an objective function with real numbers as its decision variables, as opposed to continuous functions as in the original trajectory optimization problem:

$$\min_z J(z) \tag{4.10}$$

Where it is subject to equality constraints, inequality constraints, and bounds:

$$f(z) = 0 \tag{4.11}$$
$$g(z) \leq 0 \tag{4.12}$$
$$z_{\text{low}} \leq z \leq z_{\text{upp}} \tag{4.13}$$

A variety of transcription methods have been developed, each with its strengths and limitations. A summary of the different types of available methods is shown in figure 4.1. Transcription methods can be broadly grouped into two categories: Direct and indirect methods. Direct methods work by discretizing the state, control, and objective functions, thereby discretizing the optimization problem itself. To achieve this, polynomial splines are used to approximate all continuous functions that define the problem statement [18]. In essence, direct methods discretize the optimization problem before optimizing. Direct collocation is distinct from indirect methods, where the optimization problem is solved by computing the necessary and sufficient conditions for optimality before discretizing [17].

During the optimizing process, collocation methods change the state and control functions during each iteration simultaneously. As such, they are also referred to as simultaneous methods. In contrast, non-simultaneous methods only vary the control functions, while the states are derived by simulating the system dynamics. A common example of a non-simultaneous method is the single-shooting method. Single-shooting methods are known to work well to solve simple problems but perform poorly when complexity is increased. [19]

Various programs are readily available to perform transcription, each utilizing a specific transcription method. Some examples of popular software packages that solve trajectory optimization problems are shown in Table 4.1. After transcription is performed, the problem is then solved by a nonlinear programming solver. Table 4.2 provides a few examples of such software packages.

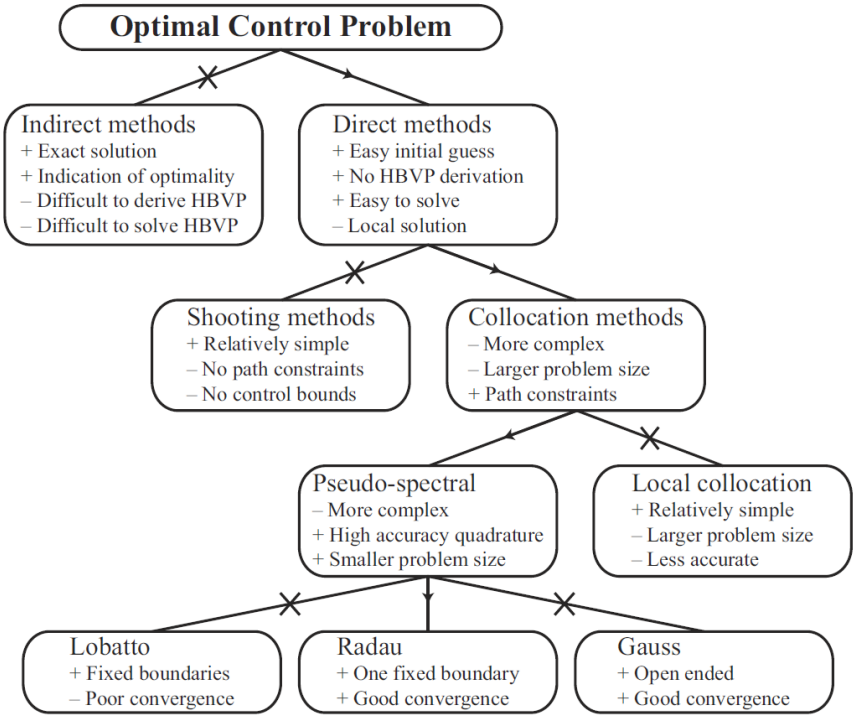

Figure 4.1: Taxonomy of different transcription methods [20].

Name	License	Interface	Method
GPOPS-II [21]	commercial	MATLAB	direct orthogonal collocation
PSOPT [22]	open source	C++	direct collocation
SOS [23]	commercial	GUI	direct collocation
DIRCOL [24]	free license	C	direct collocation
DIDO [25]	commercial	MATLAB	indirect orthogonal (pseudospectral) collocation
ICLOCS [26]	open source	MATLAB	direct collocation

Table 4.1: Examples of trajectory optimization software [17]

Name	License	Interface
FMINCON [15]	commercial (MATLAB)	MATLAB
SNOPT [27]	commercial	C++
IPOPT [28]	open source	C++

Table 4.2: Examples of nonlinear programming solvers [17]

For this study, the ICLOCS software will be used to transcribe the trajectory optimization problem. ICLOCS is a MATLAB-based, open-source optimal control tool developed by Imperial College London. The transcribed problem will subsequently be solved using IPOPT, an NLP solver. While ICLOCS is also compatible with other NLP solvers, such as FMINCON, they are known to result in longer runtimes and higher failure rates [26]. The following section will explore further details on direct collocation, the transcription method utilized by ICLOCS.

4.1.1. Direct Collocation

In direct collocation, the approximation of the state, control, and objective functions begins by breaking up the continuous curves into discrete segments, which are bounded by points known as *knot points*. For typical collocation methods, these knot points are also *collocation points*, which are points at which the dynamic constraints are applied. These dynamic constraints ensure that the system dynamics are adhered to. In other words, they ensure that the states and controls at the collocation functions match. The knot points are then connected by piecewise functions, which can be polynomial functions of different orders [17]. Figure 4.2 illustrates how a continuous function can be approximated by segments of quadratic splines.

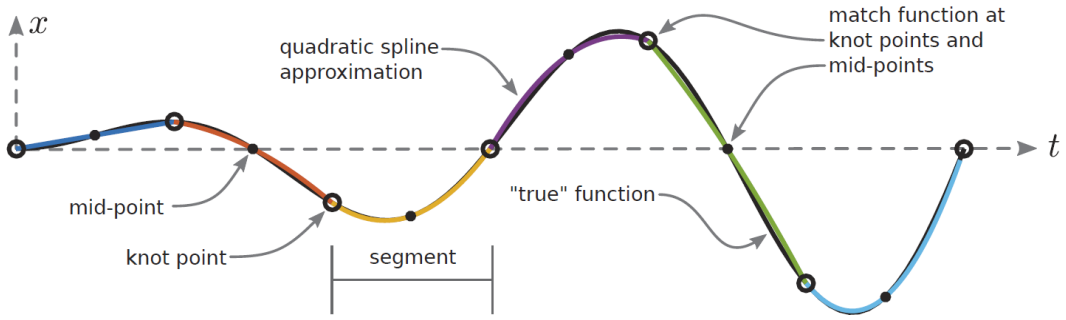

Figure 4.2: Function approximation using a quadratic spline [17].

The trajectory optimization solver to be used for this study, ICLOCS, utilizes the Hermite-Simpson collocation method. Hermite-Simpson collocation is conceptually similar to simpler methods such as trapezoidal collocation. However, it employs higher-order polynomials to interpolate between knot points, resulting in a more accurate approximation. The Simpson quadrature rule is used to approximate the integral form of the dynamics constraint, resulting in a quadratic spline approximation of the dynamics trajectory. As the states are the result of an integration of the dynamics, this naturally results in the state function being represented by a cubic spline. These splines are also known as cubic Hermite splines [29]. In addition, the control trajectory is also approximated using a quadratic interpolation formula. Figure 4.3 illustrates the quadratic and cubic spline segments used to represent the controls and states respectively. Lastly, midpoints are also added as collocation points within each segment, where the dynamics of the system are enforced as constraints, as shown in Figure 4.2.

Figure 4.3: Illustration of the quadratic and cubic spline segments that are used to approximate the control and state trajectories for Hermite–Simpson collocation. [17].

4.1.2. Objective Function Selection

To explore the effects of objective function selection on the shape and trackability of the optimal trajectory, two simple objective functions will be used. These are the minimum time and minimum horizontal distance objective functions.

$$J_{\text{time}} = t_f \tag{4.14}$$

$$J_{\text{distance}} = d(t_f) \tag{4.15}$$

Another parameter that was considered for minimization was noise emissions. As the UNIFIER C7A aircraft was designed as a commuter aircraft, it would likely be operated mostly from small airfields. With these airfields typically positioned near highly populated areas, the aircraft's noise emissions would have to be well understood to prevent excessive disturbance to the surrounding communities [30].

Difficulties were encountered when exploring viable noise models. As the model would serve within the objective function during trajectory optimization, it would be evaluated numerous times throughout the routine. A computationally efficient model would be needed to keep computation times manageable. The challenge arises from the unique configuration of the UNIFIER19 C7A's propulsion system. With multiple propellers in proximity to each other, their aerodynamic interactions change their noise emissions. Additionally, their sound fields also directly interact with each other, resulting in an overall noise footprint that would not be equal to simply the sum of each propeller's noise in isolation [31]. One propeller noise model that takes these interactions into consideration was published by Monteiro et al. [32]. However, this particular model would require computation times that would not make it suitable for use in trajectory optimization. Another option is to utilize noise models for isolated propellers [33], thereby ignoring the interaction effects entirely. This would result in a much simpler and computationally efficient model, albeit at the cost of significant reductions in accuracy.

With these factors in mind, it was concluded that the challenges associated with producing meaningful results in the evaluation of noise-optimal trajectories for the UNIFIER19 C7A would be impractical to overcome within the scope of this study. Therefore, in keeping with the primary objective of this project, the relatively simpler objective functions above were chosen to demonstrate the viability of the proposed methodology.

4.2. Tracking Control

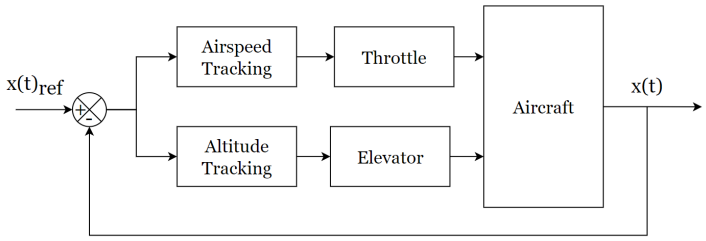

Figure 4.4: Flight control system overview

The final part of the study involves tracking the resulting optimal trajectory using the baseline high-fidelity model. The trackability of the trajectory, or in other words, how closely the high-fidelity model can follow the trajectory, will then be used as a measure of the reduced-order model's reconstruction error.

For trajectory verification and assessment within the scope of this study, a relatively simple linear PID-controller-based control system will be employed. An overview of the flight control system to be implemented on the high-fidelity UNIFIER19 C7A model is shown in Figure 4.4. The flight control system comprises two longitudinal controllers, namely the altitude and airspeed tracking systems. The two tracking systems are described in further detail in sections 4.2.1 - 4.2.2. As with the aircraft model, the flight control system will be implemented in the MATLAB Simulink environment.

4.2.1. Altitude Tracking

The altitude tracking system consists of multiple parallel control loops, which ultimately manipulate the aircraft's altitude by controlling its elevator. Figure 4.5 shows a block diagram for a typical altitude control system. The outermost loop receives the current altitude as feedback and compares it to the reference altitude. The altitude controller, typically a proportional (P) or proportional-integral (PI) controller, then receives the altitude discrepancy and produces a target pitch angle for the inner loops. The two inner loops constitute the pitch attitude control system. Pitch angle feedback also provides an effective method to stabilize the phugoid mode of the aircraft. Therefore, the pitch attitude control system also acts as a phugoid damper.

Here, the inner and outer loops both use proportional controllers, which receive the error in pitch angle and pitch rate, respectively. By forming parallel loops, they effectively act as a proportional-derivative (PD) controller, as the inner loop regulates the rate of change of the pitch angle. This arrangement has the advantage of being simpler to tune compared to a single PD controller.

When tuning each gain sequentially, the inner loop is tuned first. For the inner loop, a faster response is typically desirable as it controls a so-called fast dynamic variable, namely the pitch rate [34]. Once the inner loop is tuned, the outer (pitch angle) loop is tuned, followed lastly by the altitude loop.

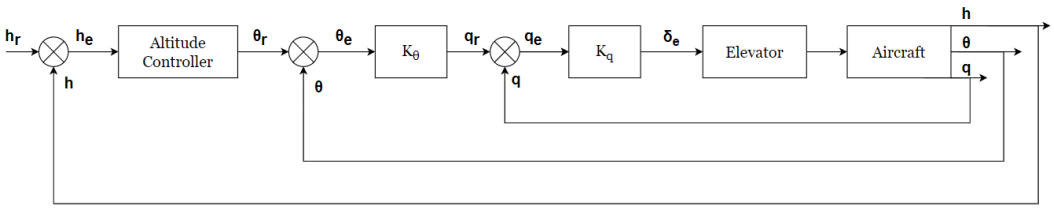

Figure 4.5: General structure for an altitude control system

4.2.2. Airspeed Tracking

The airspeed controller is much simpler compared to the other two loops, as shown by Figure 4.6. Here, only a single loop is needed as the airspeed typically has slow variations. A single controller, usually a proportional controller, is used. However, a PI controller may also be employed to eliminate steady-state error [35]. The controller receives the airspeed error and computes the appropriate throttle input, which is then fed to both the HTU and DEP.

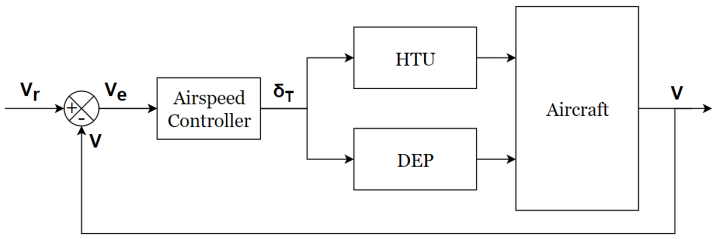

Figure 4.6: General structure for an airspeed control system

5

Reduced-Order Modeling

A reduced-order model, also often referred to as a surrogate model or metamodel, is a simplified representation of an existing high-fidelity model that is used to predict the behavior of a system [36]. Reduced-order models have been utilized in a wide variety of engineering applications where the use of high-fidelity models becomes too complex or results in impractical computation times. In the aerospace field, this is a particularly powerful tool to have. Many tasks in aerospace engineering require the use of models to represent the behavior of the aircraft. While detailed classical models (derived from aerodynamics, structural dynamics, flight mechanics, etc.) are often available, they may not be suited to be used as they are due to their high level of complexity. This may stem from nonlinearities, high order, or large memory requirements, such as the case when discrete data tables are used. To address this, simplified representations are needed to complete tasks such as optimization, embedded model implementation, and flight control system design (as shown by Figure 5.1).

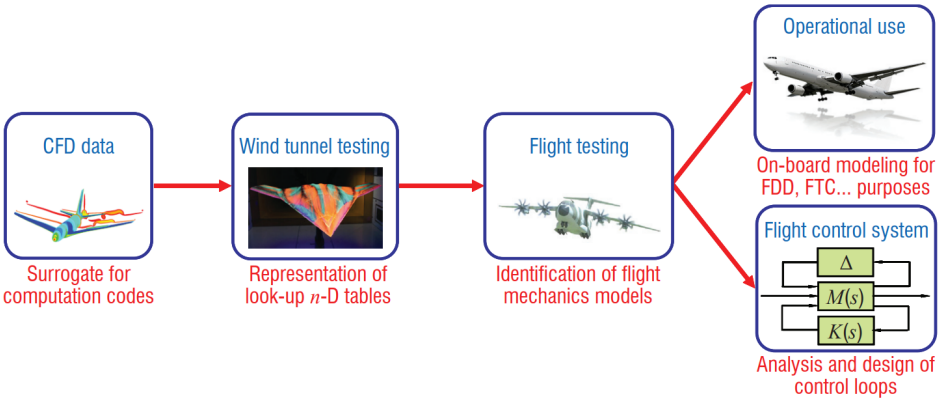

Figure 5.1: Examples of FCS development stages likely to benefit from the usage of reduced-order models [36]

Multiple methods have been developed and implemented to construct these reduced-order models [37, 38, 39, 40]. A popular example among many is the use of neural networks to represent complex nonlinear systems. In the context of flight dynamics modeling, neural networks can be used to approximate aerodynamic coefficients, effectively replacing cumbersome lookup tables that contain data obtained from CFD analysis or wind tunnel testing. By doing so, memory requirements are reduced while allowing differentiable approximations to be used and nonlinearities to be modeled. For this study, however, a simpler concept of reduced-order modeling is employed to allow the methodology to be easily applied to any existing model.

In this study, the base model is reduced in two ways. First, the aerodynamic model of the aircraft is significantly simplified. While the use of lookup tables is not eliminated, the model is reduced from a segment-based model to a singular representation of the system. Second, the six-degree-of-freedom

flight dynamics model is replaced with a three-degree-of-freedom representation of the system.

As discussed in Chapter 2, the high-fidelity UNIFIER19 model utilizes a detailed aerodynamic model based on component-level models, which are further broken down into segments. This allows for an accurate representation of local flow conditions at each component segment, thus enabling aero-propulsive interactions to be accounted for. However, a complex interpolation and summing procedure is subsequently needed to produce the total forces and moments acting on the aircraft. This naturally means that the model requires significant computational resources for each evaluation. Additionally, the model also includes various subsystems, mainly actuators, which are modeled with their own degrees of freedom. As a result, the model has more than just the six dynamic degrees of freedom, making transcription of the equations of motion intractable. To address this, the reduced order model will consider the aircraft as a single body, hence simplifying the aerodynamics from a component level to an overall aircraft level. By doing so, the optimization problem is substantially simplified.

The base High-Fidelity Model described in Chapter 2 (henceforth referred to as the *HFM*) is simplified by reducing the aerodynamic model, while the propulsive model is used as is. This approach was chosen based on the relative complexity of the aerodynamic model, which is structured as a summation of significantly more components compared to the propulsive model, which only requires calculations for twelve (identical) DEP propulsors and one HTU. Additionally, to account for the blown-wing effect the advance ratios of the DEP propulsors need to be calculated. As the HFM already directly converts the advance ratio to the thrust coefficient using a lookup table, it is not possible to further simplify the DEP model without losing the ability to account for aero-propulsive interactions.

To build the reduced aerodynamic model, three lookup tables are used to evaluate the lift, drag, and pitching moment coefficients. These lookup tables are composed by evaluating the aerodynamic forces and pitching moment produced by the HFM at varying conditions. The coefficients themselves are then obtained by simply non-dimensionalizing the lift, drag, and pitching moment by the dynamic pressure at the condition at which the HFM is simulated, as well as the chord length in the case of the pitching moment. The execution of this procedure can be intuitively understood as conducting a ”virtual wind tunnel” experiment, where the aircraft is placed at varying angles of attack and configured to a range of control surface and thrust configurations while the resulting forces are recorded. The conditions and inputs that are varied for this evaluation, along with their ranges and increments are shown in Table 5.1.

Variable		Unit	Min	Max	Increment
α	Angle of attack	deg	-40	40	1
δ_{elev}	Elevator deflection	deg	-30	30	5
δ_{flap}	Flap deflection	deg	0	25	5
DEP	DEP activity factor	-	0	1	0.1

Table 5.1: ROM evaluation input variables

While the DEP activity factor is used as an input to the ROM evaluation procedure, the advance ratio will be used in its place as an input to the resulting lookup tables. The advance ratio produced at each DEP setting is therefore calculated and stored to be used as breakpoints in the resulting lookup tables, along with the original range of α, δ_{elev}, and δ_{flap}. This is done to better represent the aero-propulsive interactions as they are physically linked to the advance ratio as opposed to simply the DEP setting. A limitation of this method is that the range of advance ratios produced, and therefore the range of advance ratios at which the ROM is able to evaluate the aerodynamic coefficients, is determined by the (fixed) airspeed at which the ROM is evaluated. Varying airspeed was also considered in order to obtain a more exhaustive range of advance ratios, however, this was observed to increase the computational cost of evaluating the ROM significantly. Therefore, a compromise was chosen by evaluating the ROM at an airspeed of 50m/s, roughly halfway between the cruise speed of 72.74m/s and the stall speed of 35.85m/s, resulting in a range of advance ratios sufficient to cover the landing scenario simulated in this study.

Accounting for as many dependencies that are modeled in the high-fidelity model as possible is expected to reduce the reconstruction error of the reduced-order model. However, including more dependencies will have an effect on computational efficiency as it retains more complex behaviors of the HFM. To further investigate this trade-off, multiple versions of the ROM will be constructed with varying numbers of dependencies accounted for. The resulting reconstruction error obtained with these models will then be compared in order to gain insights into the optimal level of fidelity needed to achieve satisfactory accuracy while minimizing computational costs and allowing convergence during trajectory optimization.

The variations in dependencies will be produced by fixing the value of one or more of the independent parameters, effectively using only a subset from the complete collected dataset for a particular coefficient. Three ROM versions are generated with version 1 being the base dataset with all dependencies, while version 3 is the simplest. A summary of the different ROM versions produced and the dependencies included for each coefficient is shown in Table 5.2. The values at which some of the parameters are fixed for the simplified models are also shown. For both versions 2 and 3, the advance ratio is fixed to a value that would correspond to a 0.5 DEP activity factor setting.

ROM Version	Coeff.	Dependencies Included			
		α	δ_{elev}	δ_{flap}	J
v1	C_L				
	C_D				
	C_M				
v2	C_L				
	C_D				
	C_M				
v3	C_L		δ_{elev} = 0 deg	δ_{flap} = 5 deg	J = 1.67
	C_D				
	C_M				

Table 5.2: Dependencies for each ROM version

6 Project Planning

6.1. Research Activities

The research activities carried out within this thesis project are broadly grouped into three major parts: Modelling, optimization, and verification.

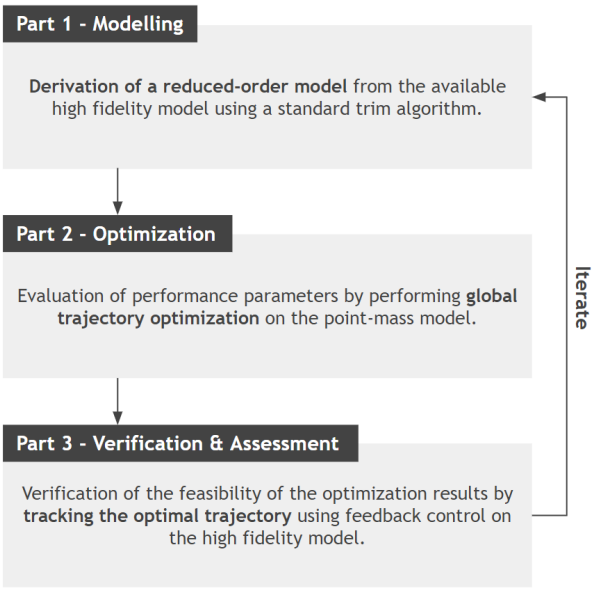

Figure 6.1: Overview of main research parts

The first part focuses on deriving the reduced-order aircraft model from the high-fidelity UNIFIER C7A model. This includes simulating the base high-fidelity model and utilizing the extracted data to construct lookup tables for the reduced-order models. The resulting reduced-order models will then serve as the dynamics constraints in the trajectory optimization problem. The optimization problem is formulated in the second part, and global trajectory optimization is performed with the ROM using a readily available solver. The resulting trajectory will then be verified in the last part, where it is tracked by the high-fidelity model using feedback control. Figure 6.2 illustrates the activities necessary within each research part. As discussed in Chapter 5, several versions of the reduced-order model will be constructed with varying degrees of fidelity. Hence, the entire process will be repeated for each variation.

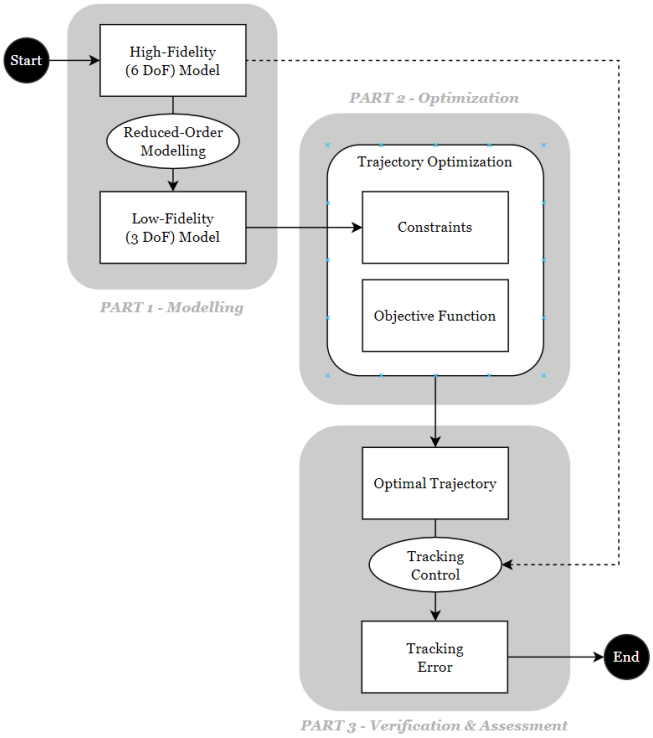

Figure 6.2: Research activity diagram

6.2. Timeline

In accordance with guidelines set by the TU Delft Faculty of Aerospace Engineering, this thesis project is organized into four stages, spanning a total of eleven months. The literature review stage was started in early September 2023 and will be followed by the research stages, which are divided into two phases and concluded by the dissemination stage. The thesis defense is aimed at August 2024. A comprehensive overview of the project timeline is included in Figure 6.3. The figure outlines the four research stages and provides dates for major project milestones as well as research content milestones.

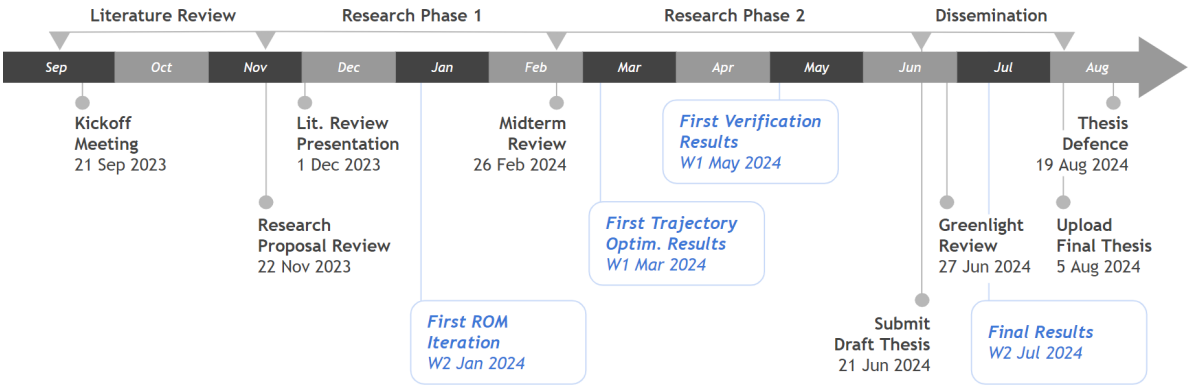

Figure 6.3: Thesis timeline

7
Conclusion

A brief background on trajectory optimization and its utility in the aircraft design process has been presented. Subsequently, the challenges faced when evaluating optimal trajectories with high-fidelity flight mechanics models were identified. These challenges mainly stem from the high order associated with high-fidelity models, making the transcription of its equations of motion intractable. To address this, a methodology is proposed that employs reduced-order modeling and tracking control. The base high-fidelity model will be reduced, resulting in a 3-DoF representation with simplified aerodynamics. An optimal trajectory can then be calculated with the use of the reduced-order model. Subsequently, tracking control will be used to track the resulting optimal trajectory using the high-fidelity model. The tracking error obtained from this exercise then serves as a proxy to represent the reconstruction error.

In addition to proving the feasibility of the methodology outlined above, the sensitivity of the results will be studied with respect to the fidelity of the reduced-order model being used. Several versions of the reduced-order model will be constructed with varying numbers of aerodynamic dependencies included in their aerodynamic models. Doing so allows insights to be gained on the trade-off between computational cost (during ROM construction as well as trajectory optimization and tracking) and model reconstruction accuracy.

The subject of this study is the UNIFIER19 C7A aircraft, of which a high-fidelity flight mechanics model was previously developed as part of the UNIFIER19 project. The aircraft is a 19-passenger fixed-wing commuter aircraft designed with a novel hybrid propulsion system. It is equipped with a distributed electric propulsion system comprising twelve propulsors along its main wing and a pusher propeller on its tail. Its unique configuration introduces interesting challenges to the optimization problem, mainly due to its redundant controls and the role of the so-called blown wing effect on its flight characteristics. For this study, the aircraft's performance during the terminal approach phase will be evaluated.

The project is organized into three main parts. The first part is concerned with the flight modeling aspects of the study, resulting in the construction of the reduced-order models. In the second part, global trajectory optimization is performed on the reduced-order models. Tracking control is then used in the third part in order to quantify the reconstruction error of the reduced-order models. The process is repeated several times with variations in ROM fidelity level, allowing for comparisons to be made from the resulting reconstruction errors produced by each variation. The project is carried out throughout an 11-month period starting in September 2023, culminating in a thesis defense aimed to be held in August 2024.

III

Appendix

A

Appendix

A.1. Simulink Models

The following provides an overview of the 3DOF UNIFIER simulator built in Simulink for this study, including the control system.

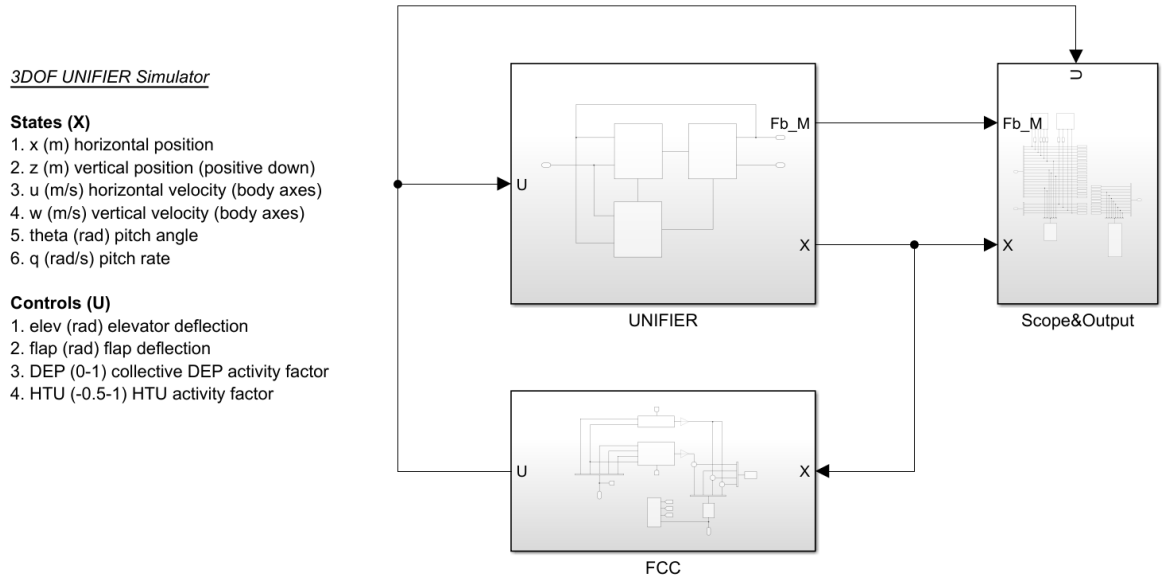

Figure A.1: 3DOF UNIFIER simulator Simulink diagram

Note that the ”Fb_M” output from the UNIFIER aircraft model subsystem is only fed to the ”Scopes&Output” subsystem and is used purely for diagnostic purposes.

The same Simulink file can be used to simulate the HFM and all ROM versions. Switching between models is done by commenting all unused models, as shown in figures A.3 and A.4. In these figures, ROM v1 is being used while the HFM, ROM v2, and ROM v3 are inactive.

The ”Propulsion” subsystem and the HFM model inside the ”Aerodynamics” subsystem are exact reconstructions of the original UNIFIER19 flight dynamics model [13]. The only modifications done to these two subsystems are that their contributions in the non-longitudinal plane are unused.

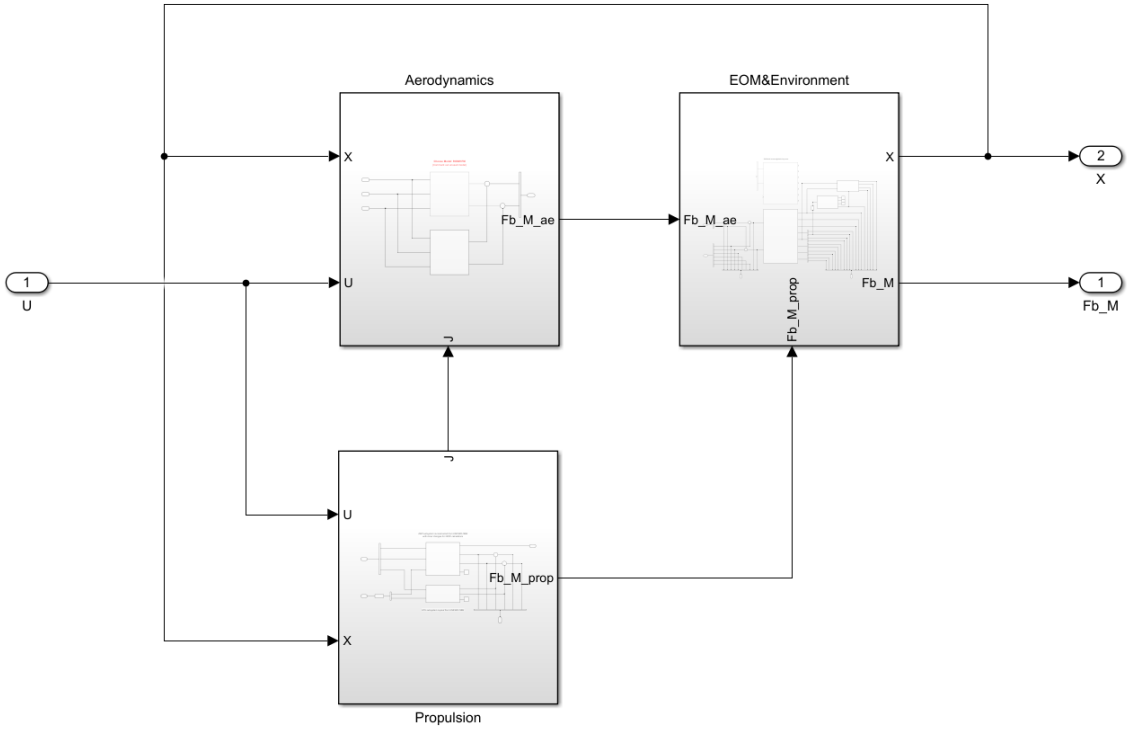

Figure A.2: ”UNIFIER” subsystem Simulink diagram

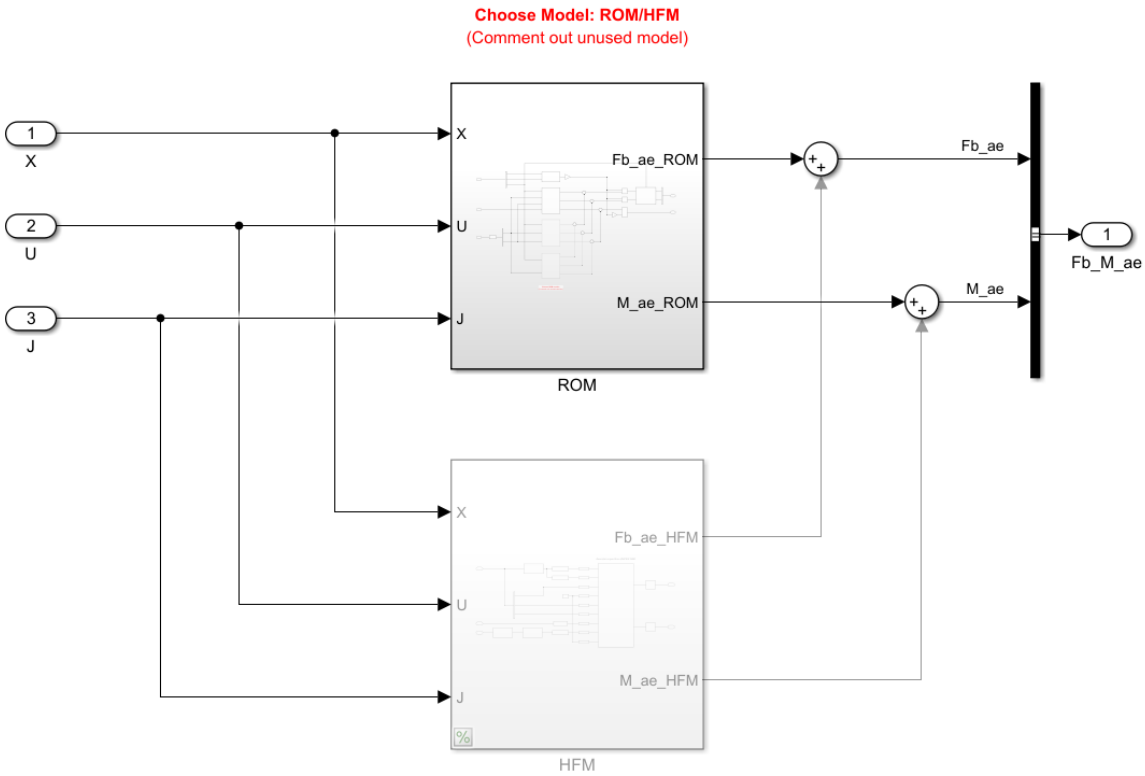

Figure A.3: ”Aerodynamics” subsystem Simulink diagram

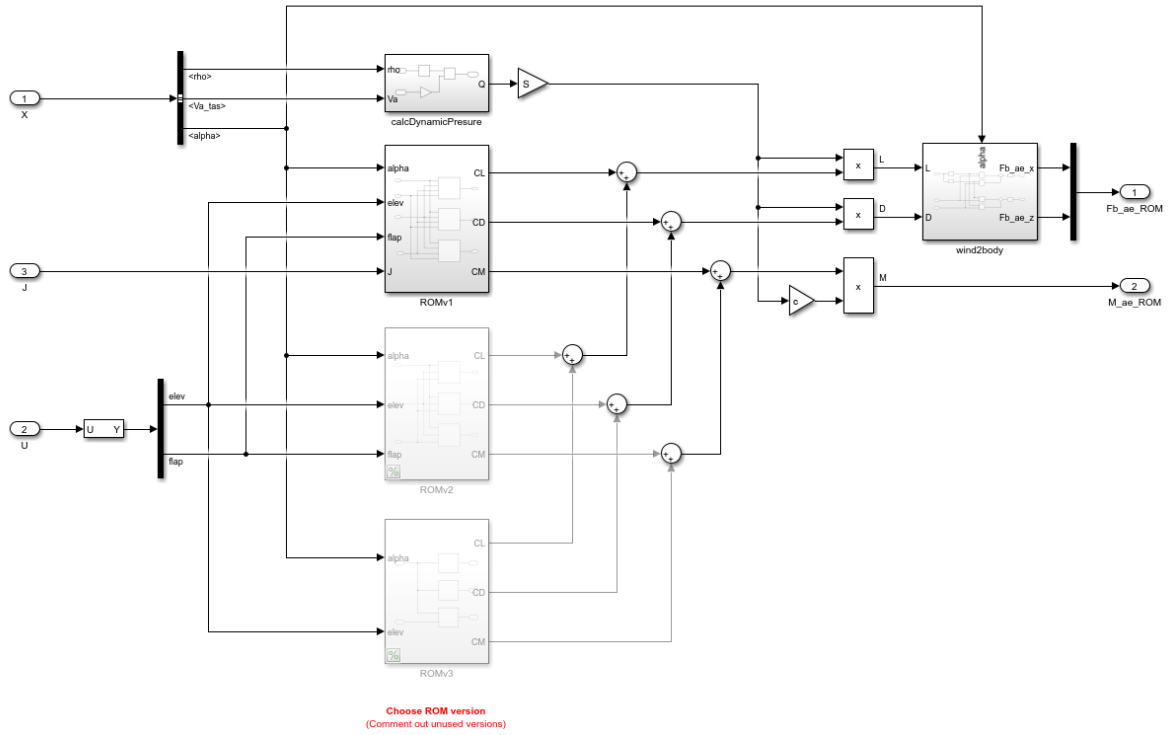

Figure A.4: ”ROM” subsystem Simulink diagram

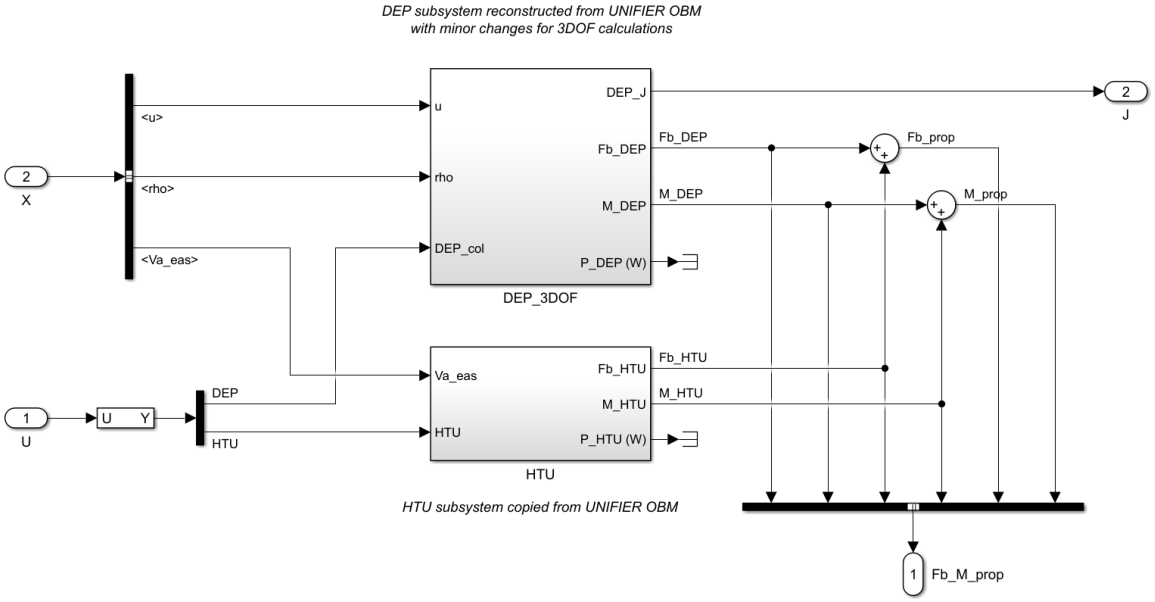

Figure A.5: ”Propulsion” subsystem Simulink diagram

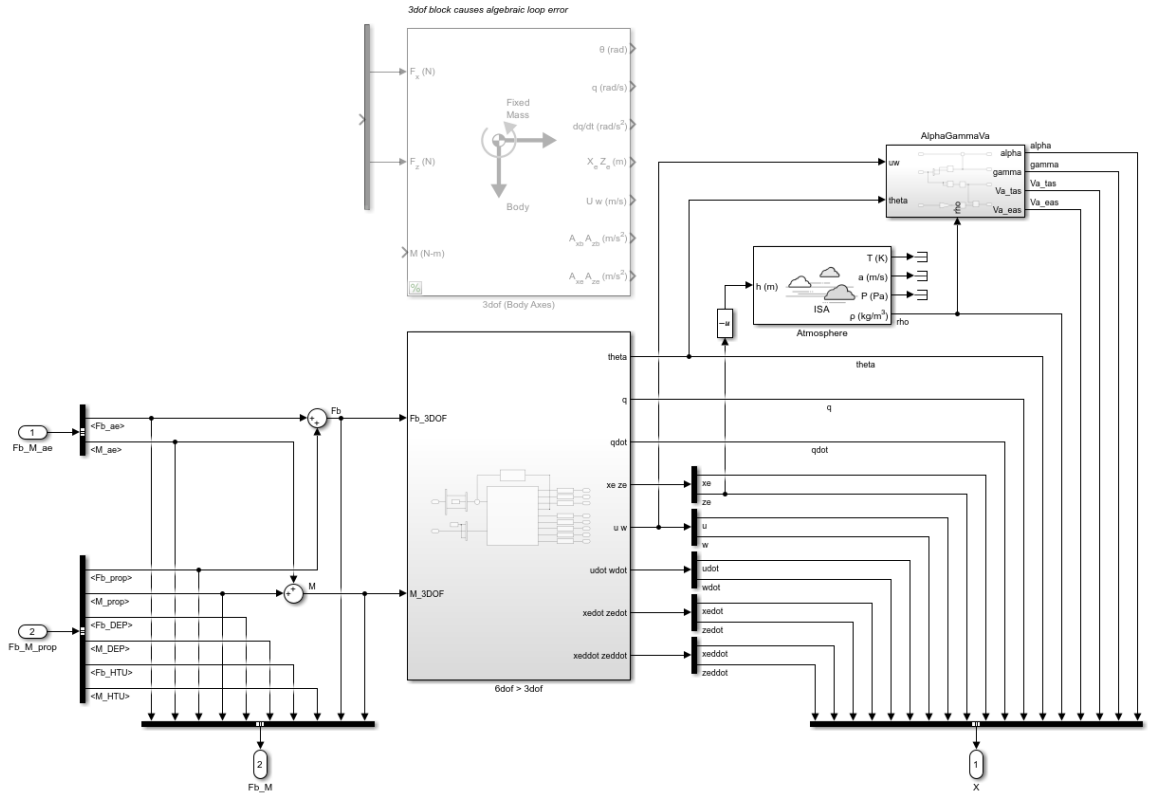

Figure A.6: ”EOM&Environment” subsystem Simulink diagram

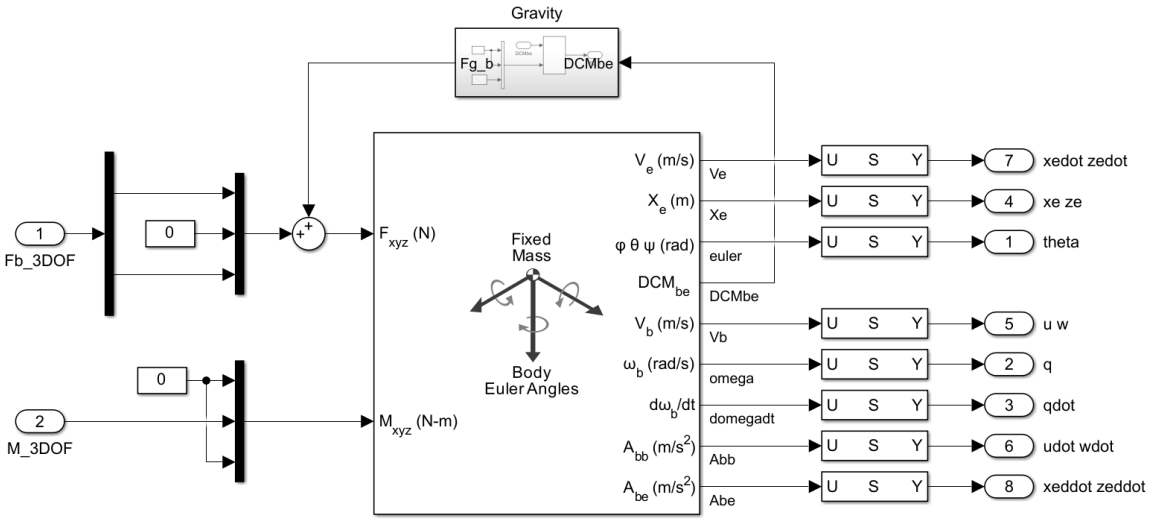

Figure A.7: ”6dof > 3dof” subsystem Simulink diagram

Note that the 6DOF EOM block is used instead of the 3DOF EOM block. During initial tests the 3DOF block was used, however, this caused an algebraic loop error which prevented the simulation from running, despite all initial conditions being defined. A workaround was devised by building a wrapper around the 6DOF EOM block to act as a 3DOF EOM block. This is done by setting the non-longitudinal forces and moments to zero and including a gravity subsystem to account for the weight acting on the aircraft, as shown by figure A.7

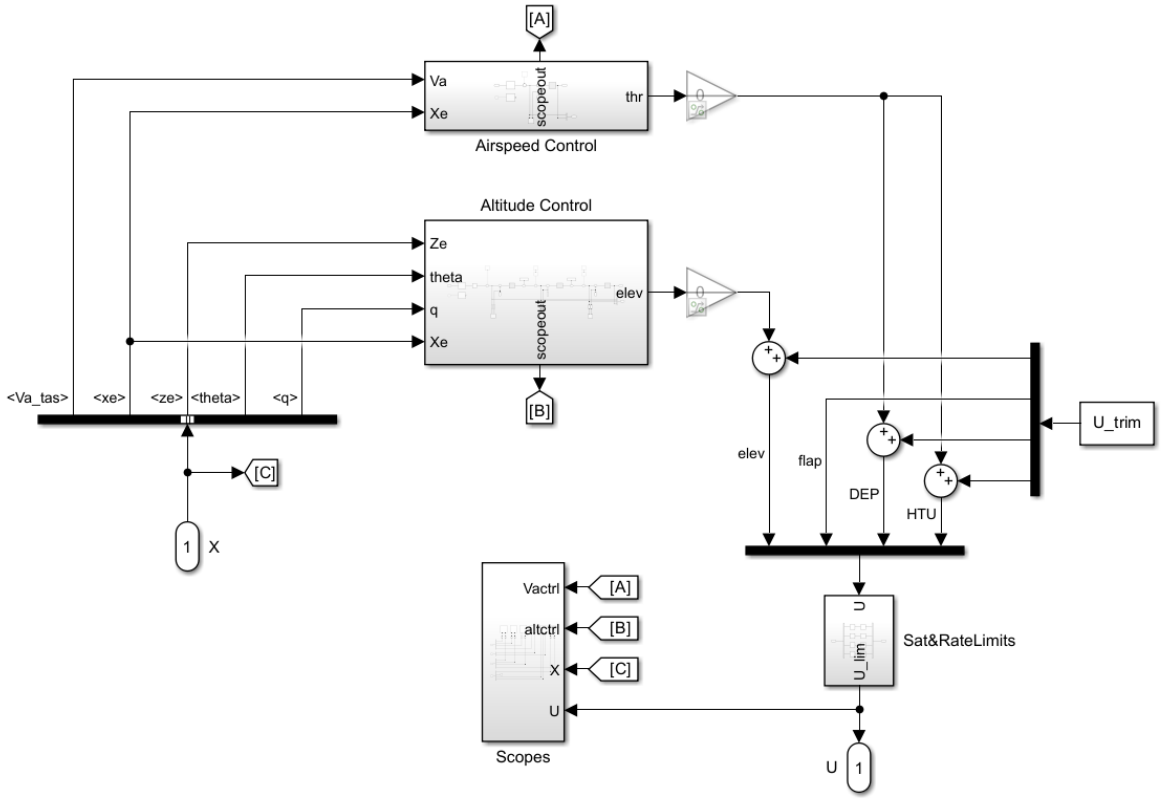

Figure A.8: ”FCC” subsystem Simulink diagram

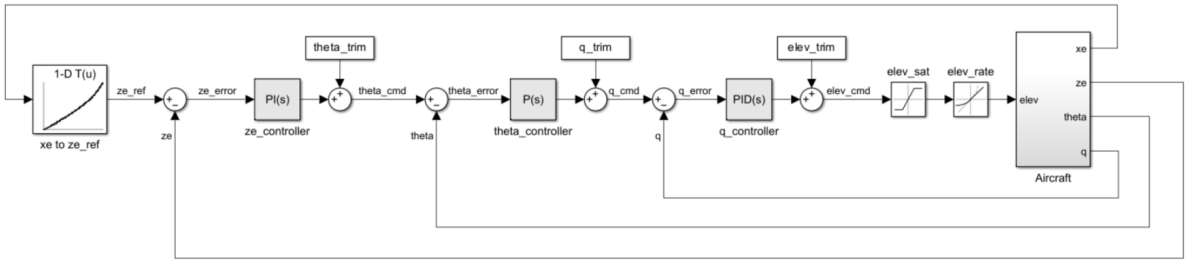

Figure A.9: Altitude controller Simulink diagram (simplified for clarity)

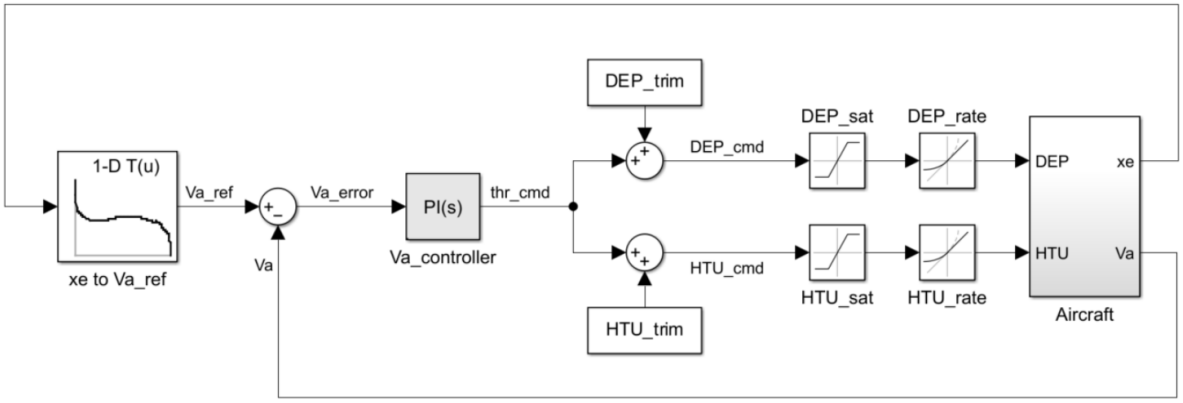

Figure A.10: Airspeed controller Simulink diagram (simplified for clarity)

Figures A.9 and A.10 show simplified versions of the Altitude and Airspeed control subsystems. These figures do not show the actual layout of the subsystems implemented in the model but are constructed to clearly show how the controllers work as standalone figures. Note that the reference values for the altitude and airspeed are interpolated from the reference trajectory based on the current horizontal distance. Another option was to interpolate the reference values based on the current time, however, it was found that this approach produced a larger tracking error.

TableA.1 shows the values used for all gains in the tracking control system. The altitude and airspeed controllers were tuned manually by trial and error, as the step response from the PID Tuner App for these controllers produced an excessively slow settling time.

Controller	**Gains** P	I	D	**Tuning Method**
z	-0.0050	-0.0010	-	Manual
θ	1.8613	-	-	Simulink PID Tuner
q	-8.7344	-54.0980	-0.0408	Simulink PID Tuner
V_a	1.5	0.5	-	Manual

Table A.1: Controller gains

A.2. Trajectory Optimization Log

Run No.	Boundary Cost	Stage cost	Inequality Constraints						Boundary Constraints				ROM Version	IPOPT Tolerance	Converged?	t_run (s)	tf (s)	xf (m)	Remarks
			RoD (ft/min)		AoA (deg)		Va (m/s)		Va (m/s)		RoD (ft/min)								
			min	max	min	max	min	max	min	max	min	max							
166			0	inf	-15	10	39.435	80	39.435	46.605	0	350		1.00E-03	N				
167			0	Inf	-15	10	39.435	80	39.435	46.605	0	350		5.00E-03	N				
168			0	Inf	-15	10	39.435	80	39.435	46.605	0	350		5.00E-03	N				
169			0	Inf	-15	10	39.435	80	39.435	46.605	0	350		5.00E-03	N				
170			0	350	-15	10	39.435	80	39.435	46.605	-	-	v1	1.00E-02	Y	94	1745	103268	
171			0	350	-15	10	39.435	80	39.435	46.605	-	-		1.00E-03	Y	98	1745	103268	
172			0	350	-15	10	39.435	80	39.435	46.605	-	-		1.00E-04	Y	99	1745	103268	
173			0	350	-15	10	39.435	80	39.435	46.605	-	-		1.00E-05	Y	92	1745	103268	
174			**0**	**350**	**-15**	**10**	**39.435**	**80**	**39.435**	**46.605**	-	-		**1.00E-05**	**Y**	**83**	**1745**	**103268**	**Final result**
175			0	350	-15	10	39.435	80	39.435	46.605	-	-		1.00E-06	N				
176	tf	Y	0	350	-15	10	39.435	80	39.435	46.605	-	-		1.00E-01	Y	23	1064	65383	
177			0	350	-15	10	39.435	80	39.435	46.605	-	-		1.00E-02	Y	21	1064	65383	
178			0	350	-15	10	39.435	80	39.435	46.605	-	-		1.00E-03	Y	20	1064	65383	
179			0	350	-15	10	39.435	80	39.435	46.605	-	-	v2	1.00E-04	Y	21	1064	65383	
180			0	350	-15	10	39.435	80	39.435	46.605	-	-		5.00E-05	Y	21	1064	65383	
181			**0**	**350**	**-15**	**10**	**39.435**	**80**	**39.435**	**46.605**	-	-		**2.50E-05**	**Y**	**21**	**1064**	**65383**	**Final result**
182			0	350	-15	10	39.435	80	39.435	46.605	-	-		2.00E-05	N				
183			0	350	-15	10	39.435	80	39.435	46.605	-	-		1.00E-01	Y	59	684	47037	
184			0	350	-15	10	39.435	80	39.435	46.605	-	-	v3	1.00E-02	Y	60	684	47037	
185			**0**	**350**	**-15**	**10**	**39.435**	**80**	**39.435**	**46.605**	-	-		**5.00E-03**	**Y**	**54**	**684**	**47037**	**Final result**
186			0	350	-15	10	39.435	80	39.435	46.605	-	-		1.00E-03	N				
187	xf	Y	0	350	-15	10	39.435	80	39.435	46.605	-	-	v1	1.00E-02	N				Failed runs w/ minimum distance objective
188			0	350	-15	10	39.435	80	39.435	46.605	-	-	v1	1.00E+00	N				
189		N	0	350	-15	10	39.435	80	39.435	46.605	-	-	v1	1.00E+00	N				
190			0	350	-15	10	39.435	80	39.435	46.605	-	-	v2	1.00E+00	N				
191			0	350	-15	10	39.435	80	39.435	46.605	-	-	v3	1.00E+00	N				
192			0	350	-15	10	39.435	80	-Inf	Inf	-	-	v1	1.00E+00	N				
193			0	350	-15	10	39.435	80	-Inf	Inf	-	-	v2	1.00E+00	N				
194			0	350	-15	10	39.435	80	-Inf	Inf	-	-	v3	1.00E+00	N				

Table A.2: Trajectory optimization run log

Table A.2 provides a log of notable trajectory optimization runs executed for this study. The log includes records of the cost function, constraints, ROM version, and IPOPT tolerance setting used for each run. Runs number 166-175 show several trajectory optimization iterations done with ROM version 1. A boundary constraint for the rate of descent was initially used, however, as this failed to produce converged results, it was removed starting from run 170 onwards. Once a converged result was obtained,

the IPOPT tolerance setting was gradually lowered to find the lowest value at which a converged result could still be produced. This procedure was also repeated for ROM versions 2 and 3, as shown by runs number 176-182 and 183-186 respectively. On the other hand, runs number 187-194 were done with the minimum distance boundary cost function. No converged results were obtained with this cost function, despite measures to promote convergence such as removing the stage function, removing boundary constraints, and increasing the IPOPT tolerance setting.

A.3. Trim Log

Run No.	DiffMin Change	Va (m/s)	ze (m)	dFlap (deg)	Penalty Factors					DEP on?	ROM ver.	Results				Accelerations				Airspeed (m/s)		Remarks
					zedot	udot	wdot	qdot	Va			Optimality	Iterations	Func. Evals.	t_run (s)	zedot (m/s)	udot (m/s)	wdot (m/s)	qdot (deg/s2)	Actual	Error	
264	1.00E-05	72.74	1,219	0	1	1	1	100	1	N	HFM	3.78E-04	18	178	6.40	0.0004	-0.0001	0.0018	-0.0026	72.7399	-0.0001	HFM, cruise, 0 deg flap
265	1.00E-06	72.74	1,219	0	1	1	1	100	1	N	HFM	6.45E-03	25	231	4.50	0.0000	-0.0001	0.0001	-0.0004	72.7399	-0.0001	
266	1.00E-08	72.74	1,219	0	1	1	1	100	1	N	HFM	3.69E-05	87	625	10.98	0.0000	0.0000	0.0000	0.0000	72.7400	0.0000	
267	1.00E-09	72.74	1,219	0	1	1	1	100	1	N	HFM	3.69E-05	87	625	10.88	0.0000	0.0000	0.0000	0.0000	72.7400	0.0000	
268	1.00E-09	72.74	1,219	0	1	1	1	10	1	N	HFM	5.34E+01	6	99	2.25	0.4638	0.0291	0.5929	-2.8851	72.7313	-0.0087	
269	1.00E-09	72.74	1,219	0	1	1	1	1,000	1	N	HFM	8.68E-04	157	1,161	17.54	0.0000	0.0000	0.0000	0.0000	72.7400	0.0000	
270	1.00E-08	72.74	1,219	0	1	1	1	1,000	1	N	HFM	8.68E-04	157	1,161	17.87	0.0000	0.0000	0.0000	0.0000	72.7400	0.0000	
271	1.00E-08	72.74	1,219	0	1	1	1	1,000	1	N	HFM	8.68E-04	157	1,161	17.93	0.0000	0.0000	0.0000	0.0000	72.7400	0.0000	
272	1.00E-08	72.74	1,219	0	1	1	1	1	1	N	HFM	8.70E-05	141	1,054	15.92	0.0000	0.0000	0.0000	-0.0010	72.7400	0.0000	
273	1.00E-08	72.74	1,219	0	1	1	1	10	1	N	HFM	5.34E+01	6	99	2.12	0.4638	0.0291	0.5929	-2.8851	72.7313	-0.0087	
274	**1.00E-08**	**72.74**	**1,219**	**0**	**1**	**1**	**1**	**100**	**1**	**N**	HFM	**3.69E-05**	**87**	**625**	**11.18**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**72.7400**	**0.0000**	
275	1.00E-07	72.74	1,219	0	1	1	1	100	1	N	HFM	2.84E-04	29	268	5.44	0.0000	0.0000	0.0000	0.0000	72.7400	0.0000	
276	1.00E-08	72.74	1,219	5	1	1	1	100	1	N	HFM	3.19E-01	141	1,204	17.65	0.0001	-0.0085	0.0014	0.0066	72.7357	-0.0043	HFM, cruise, 5 deg flap
277	1.00E-07	72.74	1,219	5	1	1	1	100	1	N	HFM	6.35E-03	24	205	3.76	0.0000	-0.0017	-0.0002	0.0003	72.7391	-0.0009	
278	1.00E-05	72.74	1,219	5	1	1	1	100	1	N	HFM	2.94E-02	20	199	3.48	-0.0005	0.0000	-0.0016	0.0030	72.7390	-0.0010	
279	**1.00E-06**	**72.74**	**1,219**	**5**	**1**	**1**	**1**	**100**	**1**	**N**	HFM	**1.25E-03**	**18**	**177**	**3.47**	**0.0000**	**-0.0001**	**-0.0002**	**0.0003**	**72.7399**	**-0.0001**	
280	1.00E-06	72.74	1,219	5	1	1	1	10	1	N	HFM	3.12E-04	25	248	4.09	0.0000	0.0000	-0.0002	0.0015	72.7399	-0.0001	
281	1.00E-06	72.74	1,219	5	1	1	1	1	1	N	HFM	2.27E-03	23	203	3.98	0.0000	-0.0001	-0.0001	0.0108	72.7399	-0.0001	
282	1.00E-06	72.74	1,219	5	1	1	1	1,000	1	N	HFM	3.66E-03	15	137	3.15	0.0000	0.0000	-0.0004	0.0001	72.7398	-0.0002	
283	**1.00E-06**	**72.74**	**1,219**	**12**	**1**	**1**	**1**	**100**	**1**	**N**	HFM	**1.04E-03**	**36**	**286**	**6.46**	**-0.0001**	**-0.0001**	**-0.0002**	**0.0003**	**72.7398**	**-0.0002**	HFM, cruise, 12 deg flap
284	1.00E-08	72.74	1,219	12	1	1	1	100	1	N	HFM	7.42E-01	269	1,988	30.39	-0.0040	-0.0101	-0.0055	0.0069	72.7375	-0.0025	
285	1.00E-07	72.74	1,219	12	1	1	1	100	1	N	HFM	7.77E-02	39	308	5.49	-0.0040	-0.0102	-0.0050	0.0087	72.7341	-0.0059	
286	1.00E-05	72.74	1,219	12	1	1	1	100	1	N	HFM	1.72E-03	25	222	4.43	-0.0004	0.0000	-0.0017	0.0026	72.7389	-0.0011	
287	1.00E-06	72.74	1,219	12	1	1	1	1,000	1	N	HFM	2.23E-02	22	197	4.44	-0.0008	-0.0020	-0.0012	0.0004	72.7388	-0.0012	
288	1.00E-06	72.74	1,219	12	1	1	1	10	1	N	HFM	5.87E-03	43	353	5.98	-0.0001	0.0000	-0.0001	0.0019	72.7399	-0.0001	
289	**1.00E-07**	**50.00**	**5**	**0**	**1**	**1**	**1**	**100**	**1**	**Y**	HFM	**3.95E-03**	**160**	**1,372**	**18.66**	**0.0001**	**0.0003**	**0.0002**	**-0.0002**	**50.0001**	**0.0001**	HFM, landing, 0 deg flap
290	1.00E-07	50.00	5	0	1	1	1	100	1	Y	HFM	3.96E-03	158	1,343	19.20	0.0001	0.0003	0.0002	-0.0002	50.0001	0.0001	
291	1.00E-08	50.00	5	0	1	1	1	100	1	Y	HFM	3.78E+00	44	448	6.29	0.0136	0.0217	0.0312	-0.0452	50.0203	0.0203	
292	1.00E-06	50.00	5	0	1	1	1	100	1	Y	HFM	1.18E-03	53	524	7.09	0.0001	0.0002	0.0004	-0.0006	50.0001	0.0001	
293	1.00E-06	50.00	5	0	1	1	1	10	1	Y	HFM	1.91E-03	36	379	5.26	0.0001	0.0003	0.0004	-0.0049	50.0001	0.0001	
294	1.00E-06	50.00	5	0	1	1	1	1,000	1	Y	HFM	9.77E-03	48	419	6.63	0.0001	0.0002	0.0007	-0.0001	50.0003	0.0003	
295	1.00E-07	50.00	5	5	1	1	1	100	1	Y	HFM	2.10E-03	89	800	10.54	0.0000	0.0000	0.0000	-0.0002	50.0000	0.0000	HFM, landing, 5 deg flap
296	1.00E-06	50.00	5	5	1	1	1	100	1	Y	HFM	6.55E-03	44	403	5.90	0.0000	0.0001	0.0002	-0.0003	50.0000	0.0000	
297	1.00E-08	50.00	5	5	1	1	1	100	1	Y	HFM	2.31E+00	190	1,829	21.25	0.0008	0.0296	0.0504	-0.0594	50.0129	0.0129	
298	**1.00E-07**	**50.00**	**5**	**5**	**1**	**1**	**1**	**1,000**	**1**	**Y**	HFM	**1.63E-03**	**127**	**1,066**	**14.66**	**0.0000**	**0.0000**	**0.0001**	**0.0000**	**50.0000**	**0.0000**	
299	1.00E-07	50.00	5	5	1	1	1	10	1	Y	HFM	1.65E-03	98	866	11.81	0.0000	0.0000	0.0000	-0.0015	50.0000	0.0000	
300	1.00E-07	50.00	5	5	1	1	1	10,000	1	Y	HFM	1.49E-02	199	1,606	21.97	0.0002	0.0006	0.0011	0.0000	50.0003	0.0003	
301	1.00E-07	50.00	5	12	1	1	1	10,000	1	Y	HFM	8.57E-03	161	1,308	18.90	-0.0001	0.0003	0.0000	0.0000	50.0000	0.0000	HFM, landing, 12 deg flap
302	**1.00E-07**	**50.00**	**5**	**12**	**1**	**1**	**1**	**100**	**1**	**Y**	HFM	**2.19E-04**	**35**	**299**	**5.37**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**50.0000**	**0.0000**	

Table A.3: Trim procedure log, HFM

Run No.	DiffMin Change	Va (m/s)	ze (m)	dFlap (deg)	Penalty Factors					DEP on?	ROM ver.	Results				Accelerations				Airspeed (m/s)		Remarks
					zedot	udot	wdot	qdot	Va			Optimality	Iterations	Func. Evals.	t_run (s)	zedot (m/s)	udot (m/s)	wdot (m/s)	qdot (deg/s2)	Actual	Error	
303	1.00E-08	72.74	1,219	0	1	1	1	100	1	N	v1	4.01E-07	20	157	6.52	0.0000	0.0000	0.0000	0.0000	72.7400	0.0000	ROMv1, cruise, 0 deg flap
304	1.00E-08	72.74	1,219	5	1	1	1	100	1	N	v1	4.01E-07	17	138	3.80	0.0000	0.0000	0.0000	0.0000	72.7400	0.0000	ROMv1, cruise, 5 deg flap
305	1.00E-08	72.74	1,219	12	1	1	1	100	1	N	v1	8.88E-06	22	178	4.36	0.0000	0.0000	0.0000	0.0000	72.7400	0.0000	ROMv1, cruise, 12 deg flap
306	1.00E-08	50.00	5	0	1	1	1	100	1	Y	v1	6.17E-05	28	251	5.41	0.0000	0.0000	0.0000	0.0000	50.0000	0.0000	ROMv1, landing, 0 deg flap
307	1.00E-08	50.00	5	5	1	1	1	100	1	Y	v1	4.20E-05	30	265	5.75	0.0000	0.0000	0.0000	0.0000	50.0000	0.0000	ROMv1, landing, 5 deg flap
308	1.00E-08	50.00	5	12	1	1	1	100	1	Y	v1	1.23E-04	21	198	3.99	0.0000	0.0000	0.0000	0.0000	50.0000	0.0000	ROMv1, landing, 12 flap
309	1.00E-08	72.74	1,219	0	1	1	1	100	1	N	v2	4.00E-07	18	154	2.91	0.0000	0.0000	0.0000	0.0000	72.7400	0.0000	ROMv2, cruise, 0 deg flap
310	1.00E-08	72.74	1,219	5	1	1	1	100	1	N	v2	4.01E-07	17	129	2.73	0.0000	0.0000	0.0000	0.0000	72.7400	0.0000	ROMv2, cruise, 5 deg flap
311	1.00E-08	72.74	1,219	12	1	1	1	100	1	N	v2	1.65E-08	20	159	3.26	0.0000	0.0000	0.0000	0.0000	72.7400	0.0000	ROMv2, cruise, 12 deg flap
312	1.00E-08	50.00	5	0	1	1	1	100	1	Y	v2	4.80E-05	30	255	4.35	0.0000	0.0000	0.0000	0.0000	50.0000	0.0000	ROMv2, landing, 0 deg flap
313	1.00E-08	50.00	5	5	1	1	1	100	1	Y	v2	9.07E-06	32	294	4.60	0.0000	0.0000	0.0000	0.0000	50.0000	0.0000	ROMv2, landing, 5 deg flap
314	1.00E-08	50.00	5	12	1	1	1	100	1	Y	v2	5.62E-06	33	304	5.01	0.0000	0.0000	0.0000	0.0000	50.0000	0.0000	ROMv2, landing, 12 deg flap
315	1.00E-08	72.74	1,219	0	1	1	1	100	1	N	v3	2.05E-04	15	125	2.39	0.0000	-0.0001	0.0000	0.0000	72.7400	0.0000	ROMv3, cruise, 0 deg flap
316	1.00E-08	72.74	1,219	5	1	1	1	100	1	N	v3	2.05E-04	15	125	2.43	0.0000	-0.0001	0.0000	0.0000	72.7400	0.0000	ROMv3, cruise, 5 deg flap
317	1.00E-08	72.74	1,219	12	1	1	1	100	1	N	v3	2.05E-04	15	125	3.77	0.0000	-0.0001	0.0000	0.0000	72.7400	0.0000	ROMv3, cruise, 12 deg flap
318	1.00E-08	50.00	5	0	1	1	1	100	1	Y	v3	7.96E-05	33	284	4.82	0.0000	0.0000	0.0000	0.0000	50.0000	0.0000	ROMv3, landing, 0 deg flap
319	1.00E-08	50.00	5	5	1	1	1	100	1	Y	v3	7.96E-05	33	284	4.52	0.0000	0.0000	0.0000	0.0000	50.0000	0.0000	ROMv3, landing, 5 deg flap
320	1.00E-08	50.00	5	12	1	1	1	100	1	Y	v3	7.96E-05	33	284	5.35	0.0000	0.0000	0.0000	0.0000	50.0000	0.0000	ROMv3, landing, 12 deg flap

Table A.4: Trim procedure log, ROM

Tables A.3 and A.4 show records of trim runs executed for the HFM and ROM respectively. For the HFM, multiple runs were done for each flight condition and flap deflection combination to find the best numerical settings in terms of penalty factors and DiffMinChange value. It is shown that the best nu-

merical settings may vary according to the particular condition. The best results are chosen based on the smallest acceleration residuals and are indicated by a bold font.

It was observed that ROMs could be trimmed more easily, meaning that the solver was able to minimize the residual accelerations more quickly and reliably. This meant that fine-tuning the numerical values was less critical toward the trim results. As a result, similar numerical settings were used for the final trim runs done on all ROMs.

A.4. Alternative Tracking Control Method

The basic concept for this method is to use numerical optimization to evaluate the control inputs needed to progress from one point in the reference trajectory to the next. Each point along the trajectory would be separated by a certain time step. With the states at each point known, the accelerations can be derived. An optimization problem can then be formulated for each point to obtain the controls in order to generate the accelerations needed to follow the trajectory. The decision variables are, therefore, the controls **u**, with the objective of minimizing the error between the resulting accelerations (calculated with the high-fidelity dynamics) and the reference accelerations. The optimization problem can then be solved at each time step similarly to a trim problem using FMINCON.

The main advantage of this method compared to feedback control-based tracking is that the resulting tracking error would provide a more accurate proxy to the reduced-order modeling reconstruction error. For this study, the objective is not necessarily to create an online tracking controller but rather to assess how feasible it is for the reference trajectory to be tracked by the high-fidelity model. Although feedback control is relatively simpler and requires substantially less computational resources, the performance of a feedback control-based tracking system is obviously highly dependent on the value of its gains. This may result in difficulties in quantifying how much of the final tracking error can be attributed to the control error (due to sub-optimal gains) and how much can be attributed to the actual reconstruction error of the reduced-order model. By formulating the tracking exercise as a type of optimal control problem, this issue can potentially be diminished. The tracking error would be affected to a much lesser extent by sub-optimal controls and, therefore, provide a better representation of the reduced-order modeling reconstruction error.

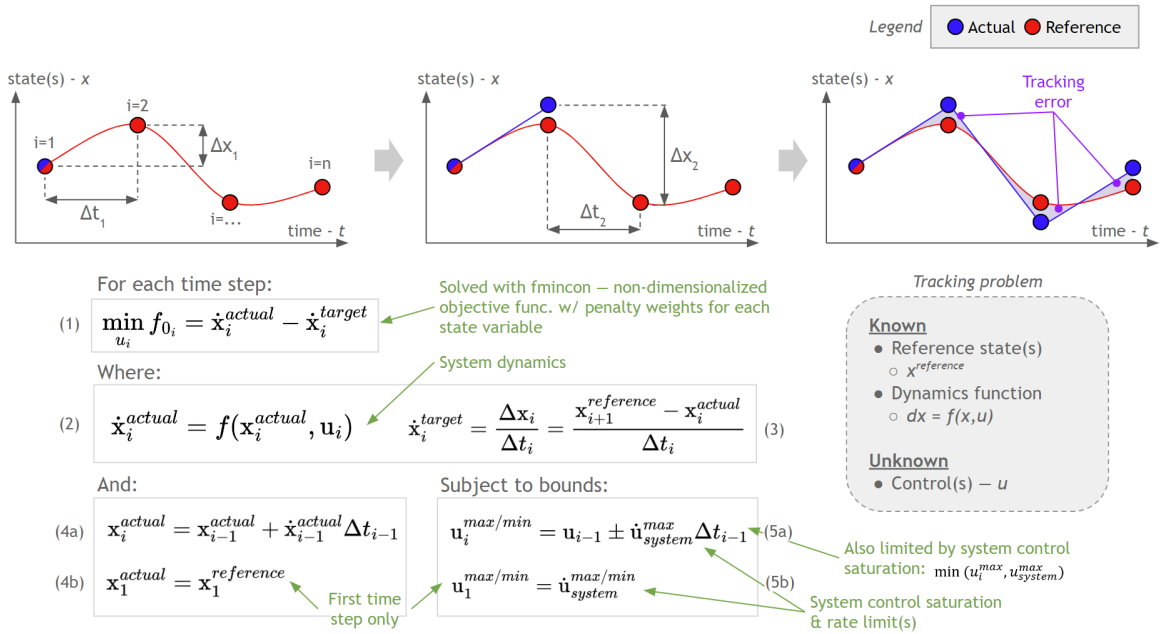

Figure A.11: Graphical representation and mathematical definition of the proposed tracking control method

This tracking method was tested using preliminary trajectory optimization outputs, producing mixed results. While some states were able to be tracked closely, other states diverged. A potential improvement could be made by only considering some state accelerations as targets (e.g. altitude, airspeed), instead of forcing all states to match the reference trajectory exactly. Additionally, it is possible that tracking all states of the preliminary trajectory optimization result was not physically possible, as the trajectory was not produced from a converged run, and therefore may not completely satisfy system dynamics constraints.

In conclusion, an alternate method to minimizing the tracking error is presented which formulates the tracking problem into an optimal control problem. This method is proposed for its potential to reduce the contribution of control sub-optimality toward the tracking error, thereby providing a better measure of the reduced-order reconstruction error. However, due to the complexity associated with the formulation of this particular optimal control problem, combined with uncertainties regarding computational costs, this method was removed from the scope of this study.

A.5. GitHub Repository

A complete repository containing all code and supporting files used for this research project is available at `https://github.com/reggiejohanes/UNIFIER_ROM_TrajOpt`.

Bibliography

[1] D.E. Kirk. *Optimal Control Theory: An Introduction*. Dover Books on Electrical Engineering Series. Dover Publications, 2004. ISBN: 978-0-486-43484-1.

[2] Mariana Ratiu and Mariana Adriana Prichici. “Industrial robot trajectory optimization- a review”. In: *MATEC Web of Conferences* 126 (2017). Ed. by G. Grebenisan, p. 02005. ISSN: 2261-236X. DOI: 10.1051/matecconf/201712602005.

[3] Abolfazl Shirazi, Josu Ceberio, and Jose A. Lozano. “Spacecraft trajectory optimization: A review of models, objectives, approaches and solutions”. In: *Progress in Aerospace Sciences* 102 (Oct. 2018), pp. 76–98. ISSN: 03760421. DOI: 10.1016/j.paerosci.2018.07.007.

[4] Runqi Chai et al. *Advanced trajectory optimization, guidance and control strategies for aerospace vehicles: methods and applications*. Singapore: Springer, 2023. ISBN: 978-981-9943-10-4.

[5] Hongchu Yu et al. “Ship Path Optimization That Accounts for Geographical Traffic Characteristics to Increase Maritime Port Safety”. In: *IEEE Transactions on Intelligent Transportation Systems* 23.6 (June 2022), pp. 5765–5776. ISSN: 1524-9050, 1558-0016. DOI: 10.1109/TITS.2021.3057907.

[6] Ahmed W.A. Hammad et al. “Mathematical optimization in enhancing the sustainability of aircraft trajectory: A review”. In: *International Journal of Sustainable Transportation* 14.6 (June 2020), pp. 413–436. ISSN: 1556-8318, 1556-8334. DOI: 10.1080/15568318.2019.1570403.

[7] Harijono Djojodihardjo. “Green Aircraft Technology Imperatives for Environmental Sustainability”. In: *Applied Mechanics and Materials* 815 (Nov. 2015), pp. 273–281. ISSN: 1662-7482. DOI: 10.4028/www.scientific.net/AMM.815.273.

[8] Prasun N. Desai and Bruce A. Conway. “Six-Degree-of-Freedom Trajectory Optimization Using a Two-Timescale Collocation Architecture”. In: *Journal of Guidance, Control, and Dynamics* 31.5 (Sept. 2008), pp. 1308–1315. ISSN: 0731-5090, 1533-3884. DOI: 10.2514/1.34020.

[9] Robert D. Falck et al. “Trajectory Optimization of Electric Aircraft Subject to Subsystem Thermal Constraints”. In: *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*. Denver, Colorado: American Institute of Aeronautics and Astronautics, June 2017. ISBN: 978-1-62410-507-4. DOI: 10.2514/6.2017-4002.

[10] Fabrizio Oliviero and Kilian Swannet. “Optimal Energy Management Strategies and Mission Profiles for a generic Hybrid Aircraft”. In: *AIAA AVIATION 2023 Forum*. San Diego, CA and Online: American Institute of Aeronautics and Astronautics, June 2023. ISBN: 978-1-62410-704-7. DOI: 10.2514/6.2023-4224.

[11] David Erzen et al. *UNIFIER19: Final Concurrent Design Report*. unifier19.eu. July 2021.

[12] Johannes S. Soikkeli, Drago Matko, and Thomas Koopman. “Cascaded nonlinear dynamic inversion applied to a fixed-wing distributed electric propulsion aircraft”. In: *AIAA AVIATION 2023 Forum*. American Institute of Aeronautics and Astronautics, June 2023. DOI: 10.2514/6.2023-4048.

[13] Johannes Soikkeli, Drago Matko, and Thomas Koopman. *UNIFIER19 Flight Dynamics Simulator Model Documentation*. https://github.com/soikkelij/unifier_fds. Aug. 2022.

[14] G. J. J. Ruijgrok. *Elements of Airplane Performance*. ISBN 978-90-6275-608-7. Delft University Press, 1990. ISBN: 978-90-6275-608-7.

[15] Mathworks. *MATLAB Optimization Toolbox*. 2014.

[16] Mathworks. *MATLAB 3DOF (Body Axes)*. 2023.

[17] Matthew Kelly. “An Introduction to Trajectory Optimization: How to Do Your Own Direct Collocation”. In: *SIAM Review* 59.4 (Jan. 2017). DOI 10.1137/16M1062569, pp. 849–904. ISSN: 0036-1445, 1095-7200. DOI: 10.1137/16M1062569.

[18] C.R. Hargraves and S.W. Paris. “Direct trajectory optimization using nonlinear programming and collocation”. In: *Journal of Guidance, Control, and Dynamics* 10.4 (July 1987). DOI 10.2514/3.20223, pp. 338–342. ISSN: 0731-5090, 1533-3884. DOI: 10.2514/3.20223.

[19] Matthew P. Kelly. *Transcription Methods for Trajectory Optimization: a beginners tutorial*. arxiv.org /abs/1707.00284. July 2017.

[20] S. Hartjes. “An Optimal Control Approach to Helicopter Noise and Emissions Abatement Terminal Procedures”. https://doi.org/10.4233/uuid:ca4b93ac-6a39-4c89-8699-e0a351e7fe2f. Doctoral Thesis. Delft, The Netherlands: TU Delft, 2015.

[21] Michael A. Patterson and Anil V. Rao. “GPOPS-II: A MATLAB Software for Solving Multiple-Phase Optimal Control Problems Using hp-Adaptive Gaussian Quadrature Collocation Methods and Sparse Nonlinear Programming”. In: *ACM Transactions on Mathematical Software* 41.1 (Oct. 2014). DOI 10.1145/2558904, pp. 1–37. ISSN: 0098-3500, 1557-7295. DOI: 10.1145/2558904.

[22] V.M. Becerra. *PSOPT Optimal Control Solver User Manual*. http://www.psopt.net/. 2011.

[23] J.T. Betts. *SOS: Sparse Optimization Suite–User’s Guide*. https://www.astos.de/ products/sos. 2013.

[24] Oskar von Stryk. *User’s Guide for DIRCOL: A Direct Collocation Method for the Numerical Solution of Optimal Control Problems*. https://www.informatik.tu-darmstadt.de/sim/forschung_sim/software_sim/dircol_sim/dircol_informationen/index.en.jsp. 1999.

[25] I. M. Ross. *DIDO*. http://www.elissarglobal.com/academic/products/. 2001.

[26] Imperial College London. *ICLOCS2 Optimal Control Software*. ee.ic.ac.uk/ICLOCS/default.htm.

[27] P.E. Gill, W Murray, and M.A. Saunders. *User’s Guide for SNOPT Version 7: Software for Large-Scale Nonlinear Programming*. http://www.sbsi-sol-optimize.com/ manuals/SNOPT%20Manual.pdf. 2006.

[28] Andreas Wächter and Lorenz T. Biegler. “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming”. In: *Mathematical Programming* 106.1 (Mar. 2006). DOI 10.1007/s10107-004-0559-y, pp. 25–57. ISSN: 0025-5610, 1436-4646. DOI: 10.1007/s10107-004-0559-y.

[29] Christoph Rösmann, Artemi Makarow, and Torsten Bertram. *Time-optimal control with direct collocation and variable discretization*. arXiv 2005.12136. May 2020.

[30] Lorenzo Trainelli et al. *UNIFIER19: The design framework for an NZE 19-seater*. unifier19.eu/www.unifier19.eu/index5508.html?page_id=245. Mar. 2020.

[31] Reynard De Vries et al. “Aerodynamic interaction between propellers of a distributed-propulsion system in forward flight”. In: *Aerospace Science and Technology* 118 (Nov. 2021), p. 107009. ISSN: 12709638. DOI: 10.1016/j.ast.2021.107009.

[32] Fernanda D. Monteiro et al. “Low-order acoustic prediction tool for estimating noise emissions from distributed propeller configurations”. In: *AIAA AVIATION 2023 Forum*. San Diego, CA and Online: American Institute of Aeronautics and Astronautics, June 2023. ISBN: 978-1-62410-704-7. DOI: 10.2514/6.2023-4180.

[33] Donald B. Hanson. “Helicoidal Surface Theory for Harmonic Noise of Propellers in the Far Field”. In: *AIAA Journal* 18.10 (Oct. 1980), pp. 1213–1220. ISSN: 0001-1452, 1533-385X. DOI: 10.2514/3.50873.

[34] Mohammad Sadraey. *Automatic Flight Control Systems*. ISBN 978-1-68173-729-4. Morgan & Claypool Publishers, Feb. 2020. ISBN: 978-1-68173-729-4.

[35] E.J.J. Smeur and A Jamshidnejad. *AE4301 Automatic Flight Control System Design Course Materials*. 2023.

[36] Georges Hardier et al. “Surrogate Models for Aircraft Flight Control: Some Off-Line and Embedded Applications”. In: *AerospaceLab Journal* Issue 12 (2016). DOI 10.12762/2016.AL12-14, 21 pages. DOI: 10.12762/2016.AL12-14.

[37] Jack P C Kleijnen and Robert G Sargent. “A methodology for fitting and validating metamodels in simulation”. In: *European Journal of Operational Research* (2000).

[38] D. Galbally et al. “Non-linear model reduction for uncertainty quantification in large-scale inverse problems”. In: *International Journal for Numerical Methods in Engineering* 81.12 (2010), pp. 1581–1608. ISSN: 1097-0207. DOI: 10.1002/nme.2746.

[39] Carlo Bottasso, Alessandro Croce, and Domenico Leonello. “Two-level model-based control of flexible multibody systems”. In: (Nov. 2023).

[40] S. Chen and S. A. Billings. “Neural networks for nonlinear dynamic system modelling and identification”. In: *International Journal of Control* 56.2 (Aug. 1992), pp. 319–346. ISSN: 0020-7179, 1366-5820. DOI: 10.1080/00207179208934317.