AeroBest 2023
Lisboa, 19-21 July 2023

FIXED-WING UAV MODEL IDENTIFICATION FOR LONGITUDINAL MOTION USING FIRST-ORDER MODELS AND LIMITED FLIGHT TESTING

Nuno M. B. Matos[1,2*] and André C. Marta [2]

1: Research and Development
Tekever UAS
Tekever
2500-750 Caldas da Rainha
nuno.matos@tekever.com, www.tekever.com

2: IDMEC
Insituto Superior Técnico
Universidade de Lisboa
1049-001 Lisboa, Portugal
nuno.matos, andre.marta@tecnico.ulisboa.pt

Abstract. *System identification plays an important role in the determination of an aircraft behaviour that helps predict and simulate different scenarios crucial for control, mission or safety assurance analysis. This work describes the system identification process of a medium sized UAV through the usage of limited flight test data and a non-linear model dynamic simulator. The proposed solution uses parameter based first-order models to describe the various aerodynamic properties of the UAV. The parameter estimation is based on a least square error optimization algorithm in a time-domain formulation starting from a low-fidelity aerodynamic analysis solution. The work focuses on the longitudinal motion by using routine flight test data of pitch down and pitch up manoeuvres to excite the longitudinal dynamics. The optimization geared towards parameter tuning used a combination of pitch and altitude UAV model response as measure of accuracy. Very significant improvements in the UAV model response are obtained with the resulting optimizer found relevant longitudinal aerodynamic and control derivatives. The pitching moment derivatives proved to be the most important parameters, as expected. The process hereby described is meant to be usable on any fixed-wing UAV with limited planned flight test data achieving reasonable accuracy.*

Keywords: aircraft design, optimization, aerospace, system identification, aircraft dynamics

1 INTRODUCTION

System identification is a field of study that encompasses the search for the optimized characterization of a system, from which, a simulated system can be created to not only explore simulation environments but also gather important information about the system qualities. In aircraft design, it can be used to determine its dynamic characteristics. By using different identification schemes, one can develop a model to either simulate flight conditions or evaluate the overall static and dynamic stability of the aircraft.

Work has been done in this field and several different approaches have been used to properly set a system identification methodology. This includes pure geometric based system identification [1], classical flight testing analysis used to do comparison between model and flight data using non-linear dynamic models [2–6], studies on the usage of big data and artificial neural networks [7, 8], and machine learning techniques [9] that train a model to behave as the real one.

Furthermore, extensive knowledge and studies have been produced on the best ways to conduct flight testing experiments for a simpler and direct construction of a classical aircraft dynamic model [10–12] where parameter estimation plays an important role. In a similar manner, the data acquisition implications also play an important role to build a complete system identification routine with a high degree of accuracy.

This work focuses on the longitudinal dynamic model characterization and optimization through the usage of limited flight data for a medium sized fixed-wing UAV (Tekever AR5 [13]). Due to the fast growing market and demands, little can be known about medium-sized UAV's, especially when flight testing is incompatible with product demands due to its slow paced data acquisition and treatment.

Here, a solution is presented to cover the gaps in the system identification of an aircraft with the usage of only routine factory flight verification data. Using consecutive pitch up and pitch down manoeuvres (used in structural flight testing), a dynamic model constructed is used to optimize and minimize the error between observable data and simulated data.

Due to data sensitivity of the studies conducted to Tekever, the results are shown in non-dimensional form. However, they are believed to be reproducible on any aircraft with similar low flight database.

2 BACKGROUND

For a contextualization of the work, a brief review of the most prominent subjects is presented.

2.1 Fixed Wing Aircraft Equations of Motion

The equations of motion that describe the behaviour of a rigid body motion in a 6 DOF system in time can be expressed as

$$\dot{X} = f(X, U) \quad , \tag{1}$$

where X, U and f are, respectively, the state vector, control vector and the vector-values nonlinear function of the individual states and controls [14].

The foundation for the formulation in flight mechanics is Newton's second law [15]. It states, for translational motion, that the sum of external forces $\vec{F}$ is equal to the change

of linear momentum and, for rotational motion, that the sum of external moments $\vec{M}$ is equal to the rate of change of angular momentum,

$$\vec{F} = \frac{d}{dt}(m\vec{U})_E = \frac{d}{dt}(m\vec{U})_B + \vec{\Omega} \times (m\vec{U})_B \tag{2}$$

$$\vec{M} = \frac{d}{dt}([I]\vec{\Omega})_E = \frac{d}{dt}([I]\vec{\Omega})_B + \vec{\Omega} \times ([I]\vec{\Omega})_B \tag{3}$$

where m, $\vec{U}$, $[I]$ and $\vec{\Omega}$ represent the mass, velocity vector, inertia tensor and angular velocity of the rigid body, respectively. The subscripts E and B refer to the Earth and body (aircraft) references frames.

The notation used for the position, aircraft velocity, aircraft angular velocity, forces and moments, is respectively,

$$R^E = \begin{bmatrix} x^E \\ y^E \\ z^E \end{bmatrix}, \quad U^B = \begin{bmatrix} u \\ v \\ w \end{bmatrix}, \quad \Omega^B = \begin{bmatrix} p \\ q \\ r \end{bmatrix}, \quad F^B = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}, \quad M^B = \begin{bmatrix} L \\ M \\ N \end{bmatrix}$$

An inertial Earth-fixed frame is used where the x, y and z axes are oriented towards North, East and Down, respectively. For the aircraft, a body-fixed frame is used where the body x, y and z axis point towards the nose, right wing and the aircraft bottom, respectively. The model used neglects the structural dynamic behaviours (rigid-body assumption) and uses the complete 6 degrees of freedom equations - the attitude or kinematic equations, which provide the aircraft rotation rate equations as

$$\begin{cases} \dot{\phi} = p + q\sin\phi\tan\theta + r\cos\phi\tan\theta \\ \dot{\theta} = q\cos\phi - r\sin\phi \\ \dot{\psi} = q\frac{\sin\phi}{\cos\theta} + r\frac{\cos\phi}{\cos\theta} \end{cases}, \tag{4}$$

where ψ, θ and ϕ are the Euler angles that correspond to yaw, pitch and roll, the force equations, which describe the translational motion as

$$\begin{cases} X - mg\sin\theta = m(\dot{u} + qw - rv) \\ Y + mg\sin\phi\cos\theta = m(\dot{v} + ru - pw) \\ Z + mg\cos\phi\cos\theta = m(\dot{w} + pv - qu) \end{cases}, \tag{5}$$

and the moment equations, which describe the rotational motion assuming a symmetric aircraft (xz-plane), as

$$\begin{cases} L = I_{xx}\dot{p} + I_{xz}\dot{r} + (I_{zz} - I_{yy})qr + I_{xz}pq + h_z q - h_y r \\ M = I_{yy}\dot{q} + (I_{xx} - I_{zz})rp + I_{xz}(r^2 - p^2) + h_x r - h_z p \\ N = I_{xz}\dot{p} + I_{zz}\dot{r} + (I_{zz} - I_{xx})pq + -I_{xz}rq + h_y p - h_x q \end{cases} \tag{6}$$

The aircraft model is constructed through the description of the forces and moments due to the state variables, which is the description of the force and moment coefficients. Generally, the coefficients are a non-dimensional characterization of the forces and moments, as

$$C_X = \frac{X}{q_d S}, \quad C_Y = \frac{Y}{q_d S}, \quad C_Z = \frac{Z}{q_d S}, \quad C_l = \frac{L}{q_d S b}, \quad C_m = \frac{M}{q_d S c}, \quad C_n = \frac{N}{q_d S b} \quad , \quad (7)$$

where $q_d = \frac{1}{2}\rho V^2$, S, c and b represent the dynamic pressure, reference area, reference chord and reference span, respectively. The pursuit of this work is the identification of the derivatives of these coefficients with respect to the state variables.

2.2 System Identification

System identification is the process of determining the mathematical model of a real system that best predicts the outputs of a system given certain inputs [10]. Generally, this process is composed by four elements [16]: data collection, selection of the model structure, selection of the system Identification method and optimization of the model.

For the data collection, different types of sensors and data acquisition techniques can be developed and used to better retrieve the needed dataset for the best possible validation and/or creation of a model. Generally, the data collection aims to retrieve the data in a time domain or frequency domain. Knowing *a priori* an approximation of the dynamic characteristics of the UAV can help excite the needed frequencies to determine the natural dynamic response of the system, enabling a better estimation of the model. A time domain analysis is, however, simpler and more direct to use.

The model structure is the mathematical principles used to build the computational model aimed to predict the behaviour of the real system. It is one of the most important decisions as choosing an incorrect model can lead to a partially described system [16]. Several works have been done with different model approaches for the description of the system: linear force and moment coefficient models with non-linear dynamic models [2–6], inclusion of structural deformation due to flexible wings [17] and a physics-based neural network approach [8]. This work focuses on the usage of a non-linear dynamic model with polynomial based equations to describe the influence of the state variables in the forces and moments.

The system identification method will determine the parameter estimation error usage. Common methods for aircraft estimation include the output-error method and the equation-error method [10]. The equation-error typically uses a least-squares regression to minimize the difference between forces and moments computed through the data acquired and the ones predicted by the model. On the other hand, the output-error method minimizes the weighted sum of squared differences between measured outputs and model outputs. This work uses a version of the latter with a choice on the observable variables as well as the weight given to each state variable output-error.

The final step is the optimization and validation model. Using the chosen model and identification method, the model is incrementally refined until the desired response accuracy is achieved. This process can involve the usage of different flight test data and optimization routines for an automatic correction of the model parameters. Figure 1 shows a summarized overview of the system identification process used in this work.

3 PRELIMINARY AIRCRAFT IDENTIFICATION

An initial model is created using low-fidelity aerodynamic tools, propulsion gathered data treatment and point mass inertia assessment. The aerodynamic and control co-

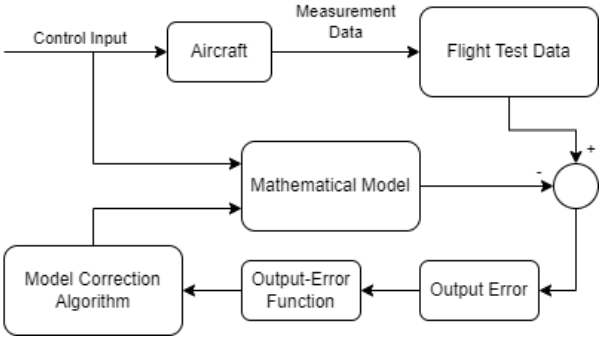


Figure 1: System identification methodology.

efficients in this model will then be optimized to achieve a better identification of the UAV.

3.1 Aerodynamics

The first aerodynamic estimate is provided using Athena Vortex Lattice (AVL) [18] analysis through the usage of vortice lattice method (VLM). This aerodynamic model is based on potential flow theory and, thus, is an initial low-fidelity estimation of the aircraft properties. Figure 2 previews the aircraft in the AVL software. A gapped wing was used to better represent the lift reduction zone due to the fuselage presence.

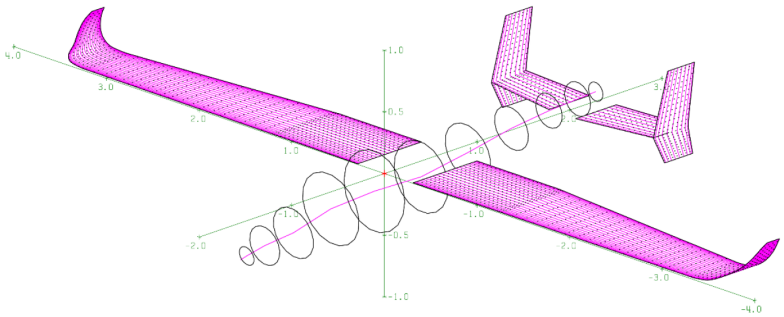


Figure 2: Tekever AR5 model in AVL.

The purpose is to build a longitudinal (and lateral) model of the aircraft to characterize the dependence of aerodynamic forces and moments with the longitudinal (and lateral) dynamic and control variables.

From an aerodynamic perspective, it is convenient to use the stability frame, in which the x axis points towards the longitudinal velocity, the y axis towards the right wing and the z axis following the right hand rule. To transform the lift C_L and drag C_D coefficients from the Earth-fixed frame to the stability frame, assuming zero heading ψ and roll ϕ,

$$\begin{bmatrix} C_X \\ C_Z \end{bmatrix} = \begin{bmatrix} -\cos(\theta) & -\sin(\theta) \\ \sin(\theta) & -\cos(\theta) \end{bmatrix} \begin{bmatrix} C_D \\ C_L \end{bmatrix} \quad , \tag{8}$$

In this work, each force and moment coefficient is described as a sum of functions for each state variable and control position. For the longitudinal motion, a generic aerodynamic force or moment coefficient is written as

$$C_A = f_0(\alpha) + f_1(\dot{\alpha}) + f_2(q) + f_3(\delta_e) + f_4(\delta_t) \quad , \tag{9}$$

where each function f_i can be presented as a linear model for simplicity, or more complex, such as higher order polynomial or non-polynomial functions.

3.2 Propulsion

The Tekever AR5 has a twin piston engine configuration where both engines rotate in the clockwise direction when seen from the aircraft tail. To model these, the engine and propeller modules in *JSBSim* were used along with a correction of the propeller curves for both power and thrust. This model includes the induced moments generated by the engine and propeller rotations.

The lack of wind tunnel data created a gap in the understanding of the engine and propeller behaviour under a free stream velocity. Thus, little can be known other that the static thrust. However, using initial flight data for different flight regimes (cruise, climb and descent), different working points for the propulsion unit can be particularly and analysed. This, coupled with the aerodynamic forces characterization (drag) can be sufficient to build a propulsion model that, for the same conditions of flight, produces the same outputs.

Using simple trimmed longitudinal flight conditions, and the coefficients for thrust force $C_T = \frac{T}{q_d S}$ and weight $C_W = \frac{W}{q_d S}$, the equations of motion for the translation in the longitudinal x and vertical z aircraft axes lead to

$$\begin{cases} C_X \sin(\theta) - C_Z \cos(\theta) + C_T \sin(\theta) - C_W = 0 \\ C_X \cos(\theta) - C_Z \sin(\theta) + C_T \cos(\theta) = 0 \end{cases} . \tag{10}$$

By using the aerodynamic model, one can determine the lift C_L and drag C_D coefficients, from which, the C_X and C_Z force coefficients can be determined using Eq.(8). Knowing the weight W of the UAV, the thrust coefficient C_T generated can then be calculated.

For each data point available and knowing the amount of fuel consumed, this approach can lead to an accurate propulsion model deeply connected to the UAV aerodynamic model. However, the propulsion model is only correct if the aerodynamic assumptions are exact. Nonetheless, this assures that the thrust response will be the same in terms of power consumption and RPM. Furthermore, it assures that, for the same data points used, the same equilibrium would be found with the model.

This model is ultimately encapsulated in two propeller curves - the power C_P and thrust C_T coefficient curves, used by the *JSBSim* model to calculate both thrust and power as as well as all the consequent forces and moments generated by each engine unit.

3.3 Inertia

The inertia of the aircraft is built with two different types of mass data: fixed point masses and variable point masses. The first describe all static payload and the airframe weight, the latter describe those that area variable, such as fuel, or whose position change. The description of the aircraft in point masses assumes that the mass body inertia I_C about its centre of mass is negligible when compared to the parallel theorem component of the inertia, resulting for the moments of inertia

$$I_{ii} = I_{C_{ii}} + m d_i^2 \approx m d_i^2 \ , \tag{11}$$

and for the products of inertia

$$I_{ij} = I_{C_{ij}} - md_i d_j \approx -md_i d_j \ , \tag{12}$$

where m is the point mass and d_i and d_j represent the distance of the point mass to the i and j axis, respectively.

3.4 Dynamic Model

The dynamic model is derived from the *JSBSim* capabilities. This software allows the choice of the force and moment axes, in which any dependency can be formulated, from constant contributions to n-variable functions as described in Eq.(9).

Using the concept formulated in Section 3.1, each longitudinal moment and forces are described by setting a function for each state and control variable. Furthermore, the model allows for the definition of the inertia of the body. Meaning, each mass or inertia contribution can be fed into the model during optimization cycles. The lateral contributions of the system are neglected for this case study, as the interest lays on the longitudinal motion.

The model is effectively a non-linear 6 degree of freedom construction of the equations presented in Section 2.1. The way the software proceeds with each step calculation is shown in Figure 3, where, from the current state variable solution, each force and moment contribution is calculated and fed into the non-linear equations scheme, from which the state variables are retrieved and used once again to begin a new step calculation cycle.

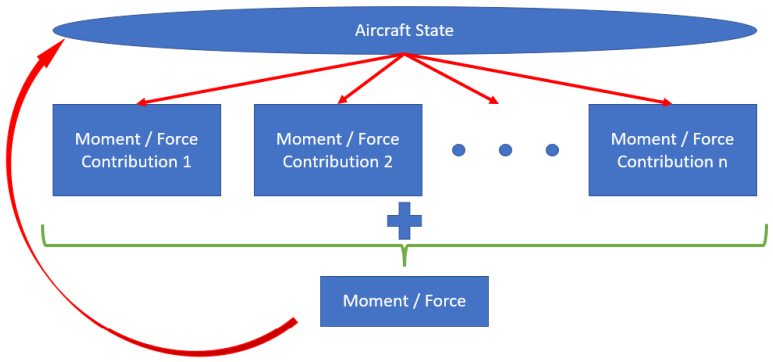


Figure 3: Force and moment calculations overview on *JSBSim*.

The *JSBSim* [19] software package includes Runge-kutta (RK) and Linear Multistep Methods (LMMs) to solve the dynamic system of equations. The default methods depend on the equation set being numerically integrated in time. Translational and rotational rate equations (Eqs.(5) and (6)) use a second-order Adams-Bashforth-Moulton (ABM) method

$$X_{ABM}(t+\Delta t) = X(t) + \Delta t \left[\frac{3}{2}\dot{X}(t) - \frac{1}{2}\dot{X}(t-\Delta t)\right] \ , \tag{13}$$

and the kinematic equation (Eq.(4)) uses a second order RK method

$$\begin{cases} X_{RK}(t+\Delta t) = X(t) + \frac{1}{2}(k_1 + k_2) \\ k_1 = \Delta t f(X,t) \\ k_2 = \Delta t f(X + k_1, t + \Delta t) \end{cases} . \tag{14}$$

The time step Δ_t was set to 0.0083 seconds, which corresponds to a frequency of $120Hz$, believed to capture sufficient detail of the dynamics of the system.

4 EXPERIMENT AND SIMULATION DESIGN

An important part of the problem is the experimental data available. In this work, routine structural test flight data is used as flight database for optimization purposes. The experiment consists of trimmed flight followed by pitch up and pitch down manoeuvres where the longitudinal motion is highly predominant. An example test layout is shown in Figure 4, where six plots are presented: on the top side, from left to right, the pitch θ, pitch rate q and estimated angle of attack α are presented; and on the bottom side, the elevator deflection delta from the initial trim state $\Delta\delta_e$, calibrated airspeed V_{CAS} and altitude h are presented.

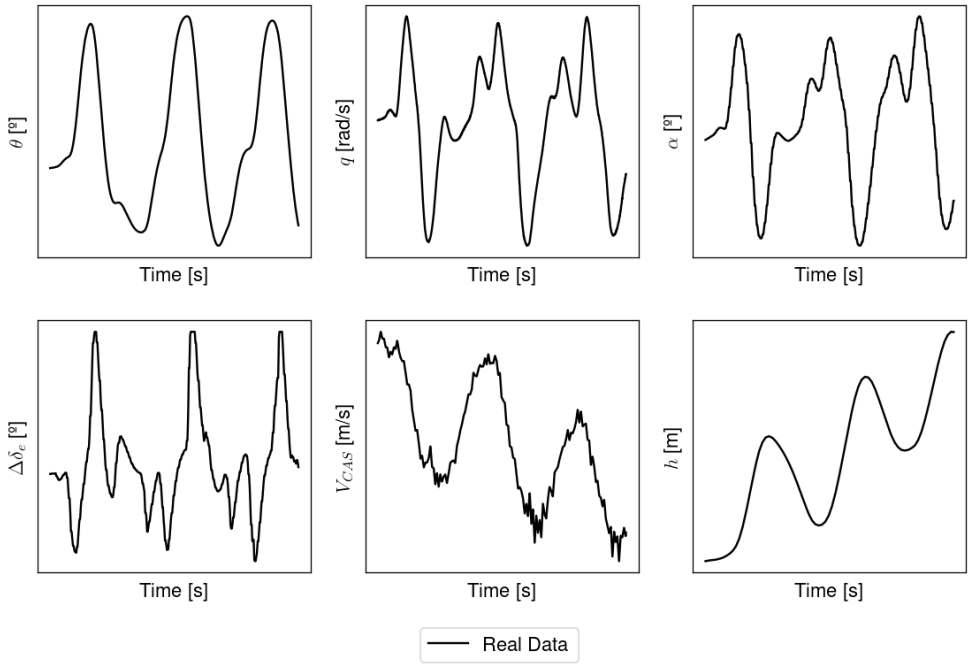


Figure 4: Structural test layout manoeuvre example for a generic Tekever AR5 aircraft.

4.1 Aircraft Data Collection

In this work, the data collection is heavily based on gyroscopic and pitot tube data to determine the angular position and speed of the aircraft, respectively. The UAV carries a data collection sensor system capable of retrieving longitudinal velocity, angular position, rates and accelerations. Angle of attack and sideslip angle estimators exist but their data is only used for data comparison and not error definition. The control data is also registered as output commands from the auto-pilot system, from which, the throttle command and elevator position can be deduced.

The autopilot works at $400Hz$, however, the control actuation is limited to $50Hz$. Thus, the control output of the autopilot is only evaluated at the control actuation frequency. In this work, the *JSBSim* simulation has a higher frequency than the control actuation, both to accurately represent it and capture more precisely the fast dynamic behaviours of the UAV. Thus, a linear interpolation is used between each control data point to evaluate,

for each simulation step, both the control positions as well as the state of the aircraft. These are used for error computation and control equivalence between simulation and real log data.

Although prone to error, the observable data is considered exact. The analysis of the degree of uncertainty in relation to a specific dataset is out of the scope of this work. Nonetheless, proven the method, an inclusion of certainty degree for each observable data enhances the system identification capabilities.

4.2 Simulation Development

This work focuses on the longitudinal simulation of the UAV based on the dynamic model created. For this, the control commands are fed into the model and the behaviour of the system is compared with the flight data available. Two important aspects arise from this analysis: the initial conditions set-up and the control input commands.

For the first, an approximation is used assuming the aircraft starts from a trimmed levelled flight condition. Using a simulated auto-pilot system, the simulated UAV is levelled and, after meeting certain convergence criteria, the experiment is started. The convergence criteria are based on the evaluation of the first derivative of a group of state variables, namely calibrated airspeed V_{CAS}, vertical velocity $\dot{h}$, pitch rate q, roll rate p and yaw rate r.

For the second, the available control input data consists of distributed time points of throttle or elevator commands logged by the system in a specific frequency that does not match the control output frequency. For this reason, as mentioned, a linear interpolation method is used to evaluate the value of each control in each simulation time step. For the throttle command, the important data is the RPM of the engines, from which the linear interpolation is used to calculate the simulated propeller RPM, thus producing, according to the proposed model, the same thrust and power consumption connected to the model. As for the elevator command, the important data is the elevator deflection difference from a trimmed elevator position.

To assure equal dynamic excitation on both the real and the simulated model, delta (Δ) control inputs are used instead of absolute control inputs. This methodology minimizes the trimmed state control input error and focuses on the delta impact on the overall dynamics of the UAV. Thus, the elevator command fed is the difference between the previous trimmed state condition. The throttle command, since it is validated with prior experimental data, will have similar responses for both delta and absolute implementations.

Since the position of the elevator is only known through an approximation - notice that the position is calculated by knowing the auto-pilot command and the mechanism itself (there is no sensor for the angle deflection) - most relevant data is obtained relative to the trim state, thus discarding the trim point convergence optimization for the pitch moment (C_{m_0} is constant throughout).

5 PARAMETER ESTIMATION RESULTS

In this section the parameter estimation results are discussed. First, the problem at hand is formulated and the algorithm used is validated using a test-case scenario with the convergence between a generic model the modified version of it. Then, the error function is formulated based on different definitions and the optimization problem is finally defined.

Single-manoeuvre and multi-manoeuvre optimizations are conducted where, in the latter, training and testing data sets are used to both fit the model and evaluate its generalized error based on a maximum likelihood error estimation.

5.1 Problem Formulation

This work uses the *JSBSim* software package to simulate a manoeuvre scenario with throttle and elevator control inputs equivalent to a real log data manoeuvre. The purpose is to verify the error between the constructed model and the real log data to better identify the system at hand.

For this purpose, a python-based framework was constructed to handle both the simulation calling as well as the data analysis between model and real data. This framework allows to additionally compare two *JSBSim* models and do optimization cycles to converge one into the other. This allowed for the validation of the method, which will be seen in Section 5.1.1.

The problem is the evaluation of the error between the calculated model data and the collected real log data. The error in this work is handled with four different variables - calibrated airspeed (V_{CAS}), altitude (h), pitch angle (θ) and pitch rate (q). These are used to estimate, through maximum likelihood methods, the similarities between the model and the real flight data. This work used the root mean squared error (RMSE) to calculate, for each variable, the associated error. This is further explored in Section 5.1.2.

Additionally, for the optimization, the *openMDAO* [20] tool was implemented within the python framework to efficiently handle the optimization process. The *SLSQP* (Sequential Least Squares Programming) optimizer was chosen along with first order finite differences method to address the objective function (RMSE) vector gradient with respect to the design variables.

The design variables considered can be found in Table 1. Note that throughout the parameter estimation analysis, some design variables were not included due to the results found in Section 5.2. Additionally, two variables are left out from the analysis, the zero angle of attack pitch moment coefficient (C_{m_0}) and all drag contributions. The former is left out since it only affects the initial trim point and the latter is not included since modifying the drag contributions would make the propulsion component unreliable.

The optimization loop is as follows: a first run is executed to determine the initial objective function value; the vector gradient is calculated through a first order finite difference method; the *SLSQP* calculates a first step towards modifying the design variables; the model is updated according to the new set of variables; the process is repeated until a certain convergence criterion is met. The optimization process can also be visualized in Figure 5, where τ represents the vector of design variables.

5.1.1 Validation

To prove the effectiveness of the method formulated, a *JSBSim* model identification using another *JSBSim* model was done for a modified set of initial design variables. The initial model is a generic model (the first iteration model from the aerodynamic, propulsion and inertia analyses data) from which modifications were made to verify if the optimizer could successfully return the model to its original state, both in design variable values, as well as dynamic behaviour. Figure 7 provides data on the error for the conducted analysis

Table 1: Design variables considered for the optimization problem.

Variable	Definition
C_{m_α}	Pitching moment coefficient partial derivative in relation to the angle of attack α
C_{m_q}	Pitching moment coefficient partial derivative in relation to the pitch rate q
$C_{m_{\dot{\alpha}}}$	Pitching moment coefficient partial derivative in relation to the derivative in time of the angle of attack $\dot{\alpha}$
$C_{m_{\delta_e}}$	Pitching moment coefficient contribution of the elevator control surfaces. In this work it is described as either a linear function or a quadratic function. Positive and negative deflections have different terms.
$C_{L_{\delta_e}}$	Lift contribution of the elevator control surfaces. In this work it is described as either a linear function or a quadratic function. Positive and negative deflections have different terms.
C_{L_α}	Lift coefficient partial derivative in relation to the angle of attack α
C_{L_0}	Zero angle of attack α lift coefficient
$C_{L_{\dot{\alpha}}}$	Lift coefficient partial derivative in relation to the derivative in time of the angle of attack $\dot{\alpha}$
C_{L_q}	Lift coefficient partial derivative in relation to the pitch rate q

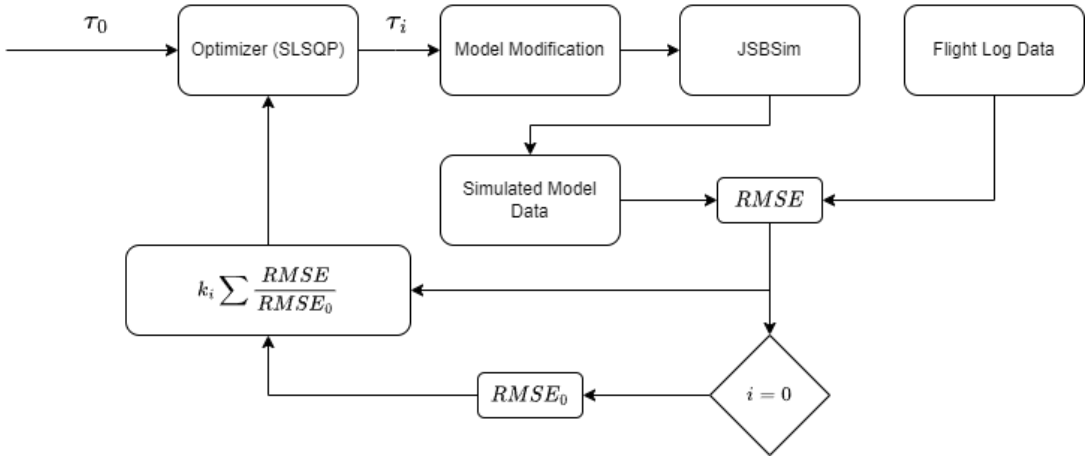


Figure 5: System identification optimization algorithm.

and Figure 6 provides a graphical overview of the conducted optimization obtained using a combined pitch, pitch rate, calibrated airspeed and altitude error formulation (see Section 5.1.2).

As seen, the optimized results converges to the same dynamic behaviour as the desired solution. The improvement is also clear as, for all shown curves, the real data and final solution data are equivalent.

5.1.2 Error Formulation

This work uses two different formulations to calculate the error between real and model data. For a single variable error, the formulation is the standard root mean square error (RMSE) normalized with the initial RMSE given as

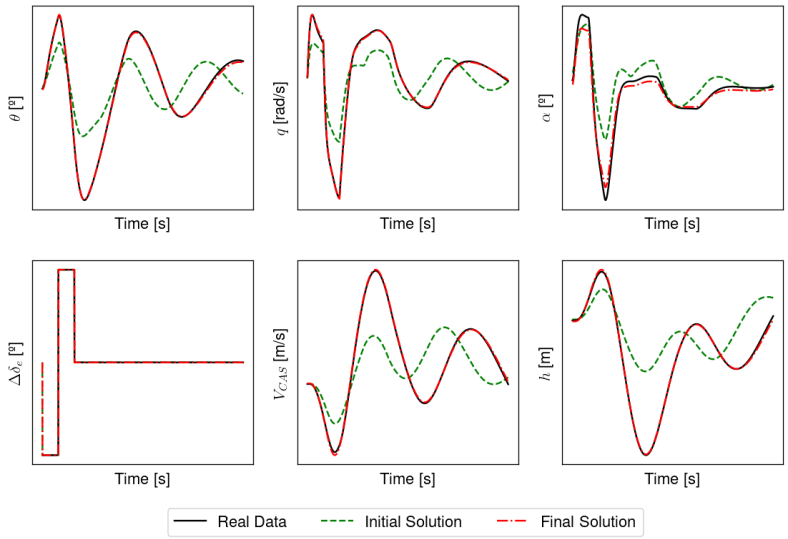


Figure 6: Methodology validation using two *JSBSim* models to prove the convergence of one towards the other.

$$E = \frac{RMSE_i}{RMSE_0} = \frac{\sqrt{\sum_{j=1}^{N}(y_j - \hat{y_j})^2}}{\sqrt{\sum_{j=1}^{N}(y_{j_0} - \hat{y_{j_0}})^2}} \quad , \tag{15}$$

where N is the number of data points in the manoeuvre, y_i the real values and $\hat{y}_i$ the simulated model values, both for a desired error variable.

For a multiple variable error, this work uses an equal based weight formulation, where the error is the normalized sum of all errors (formulated as stated previously) for all variables intended,

$$E = \sum_i \frac{RMSE_{\tau_i}}{RMSE_{\tau_{i_0}}} \quad , \tag{16}$$

where τ_i represents a generic designated error variable.

Additionally, two different approaches to the determination of y_i and $\hat{y}_i$ were used. One classical method was to use the absolute value of the variable chosen for the error calculation

$$\delta_{y_i} = y_i - \hat{y}_i = \tau_i - \hat{\tau}_i \quad . \tag{17}$$

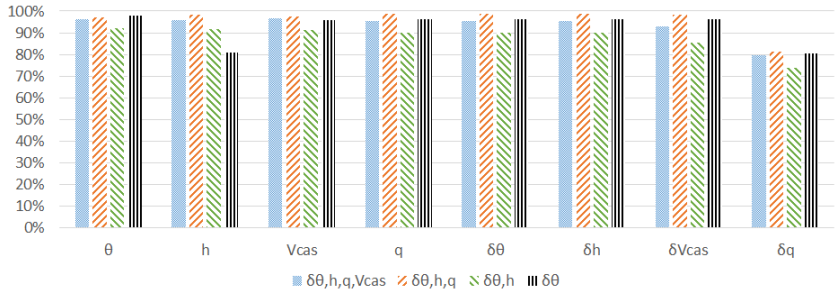
The other was to use the step difference between each state during the flight manoeuvre

$$\delta_{y_i} = y_i - \hat{y}_i = (\tau_{j+1} - \tau_j) - (\hat{\tau}_{j+1} - \hat{\tau}_j) \quad , \tag{18}$$

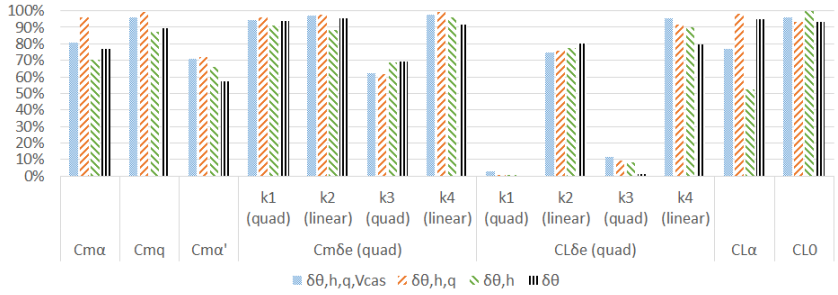
where, in this case, i represents the step index and j the state variable index. The first formulation, combined with the formulated error expressions, will evaluate the similarity between the absolute values of the state variables. The latter will evaluate the similarities between shapes.

To address which error formulation to use, different approaches were conducted using the aforementioned *JSBSim* model validation. Four error formulations were tested where each one was obtained by incorporating an additional variable, starting with the pitch step error δ_θ and then including altitude δ_h, pitch rate δ_q and calibrated airspeed $\delta_{V_{CAS}}$

errors. The results are compiled in Figure 7, where, for each optimization analyses, the results for all single variable errors are shown (Fig. 7a) and the error for each variable is also shown (Fig. 7b).



(a) Improvement in each single error score.



(b) Similarity between validation model variables and optimized model design variables.

Figure 7: Results for the different error formulations with the *JSBSim* validation scheme.

Results show that increasing the variables in the error formulation helps by averaging the improvement score in all variables, whilst having better single score in some cases. Note the improvement in the pitch step error score ($\delta\theta$) when the optimization error variables are the pitch step, altitude and pitch rate. Additionally, the similarity between real values and optimized values also improves due to the usage of a higher number of error sources. The only exception to improvement are the quadratic terms for the elevator surface lift coefficient. This can be due to their small influence in the overall longitudinal dynamics of the system being modelled, where all the other variables seem to have higher influence in the error minimization.

For these reasons and, since the real optimization problem is more complex, this work opted for the use of all error variables in the objective function formulation, where the pitch is used with the step error formulation and all the other variables (altitude, pitch rate and calibrated airspeed) are used with the absolute formulation. This augments the capture of all needed data for a best longitudinal model fit.

5.1.3 Problem Definition

Using the previous results as basis, the optimization problem is formulated as

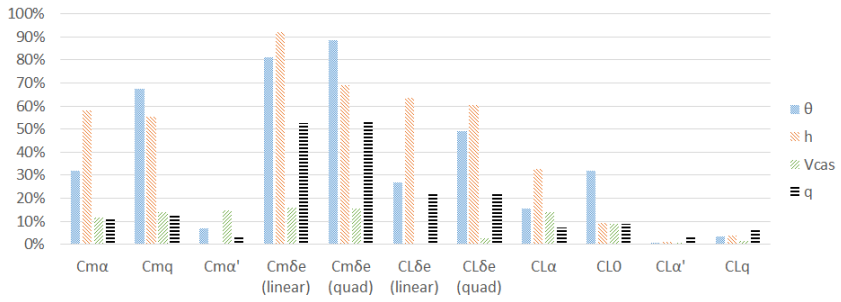
$$\begin{aligned} min \quad & \sum \frac{RMSE_{\tau_i}}{RMSE_{\tau_{i_0}}} \\ w.r.t \quad & f_{\tau_i} \end{aligned} \quad , \tag{19}$$

where the τ and f_{τ_i} are the error variables (θ, h, V_{CAS} and q) and the functions f for each design variable τ considered (as presented in Eq.(9)), respectively. The optimization algorithm is the *SLSQP* and the optimization parameters included a tolerance of $1e-3$ and a maximum number of iterations of 200 for convergence criteria.

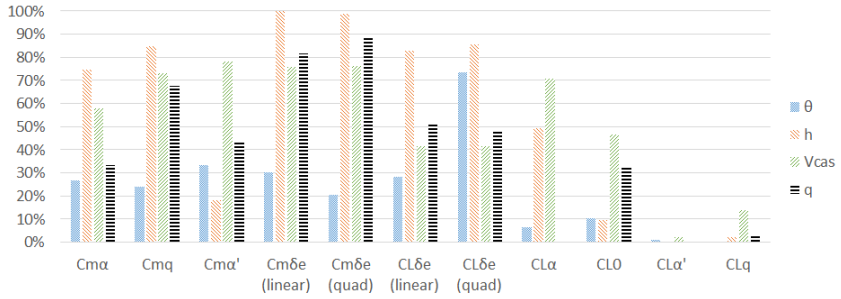
The following results use these convergence parameters and the same error formulation. The only exception is the sensitivity analysis of the design variables (Section 5.2), where the selection of the design variables for the final optimization problem is done using a single variable error approach.

5.2 Sensitivity Analysis

To assess the impact of the design variables on the optimization problem and system identification procedure, a sensitivity analysis was conducted. This analysis also allowed for the selection of the design variables for the parameter estimation optimizations conducted. For each variable and each type of single variable error considered, an optimization was conducted. The compilation of the results are presented in Figure 8, including the error improvements for each data variable using absolute (Fig. 8a) and step (Fig. 8b) error formulations in single variable optimizations.



(a) Using the absolute error formulation.



(b) Using the step error formulation.

Figure 8: Improvement in each error variable for all single variable optimization cases.

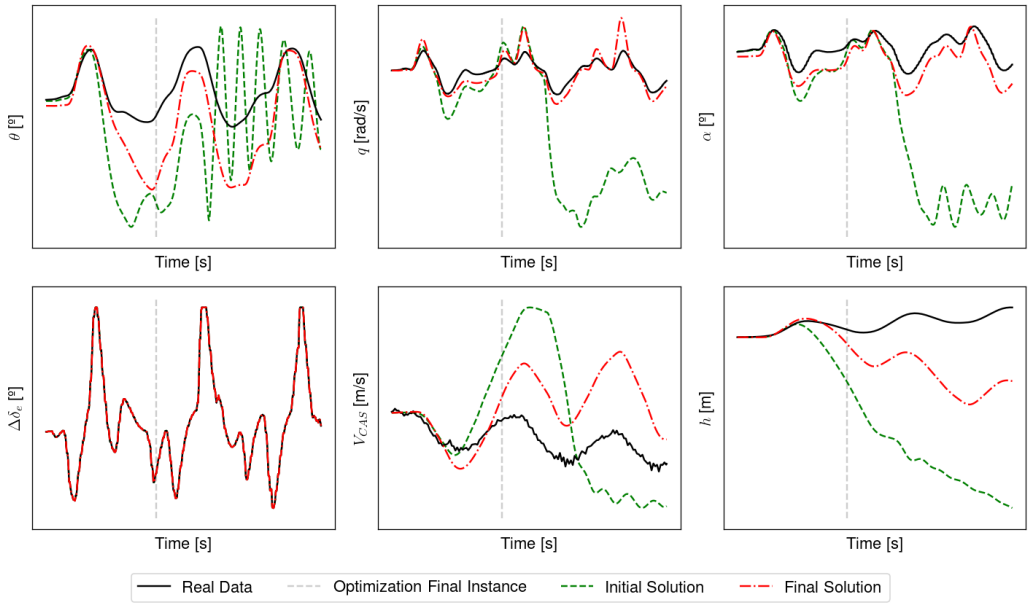
As seen, the optimization results depend not only on the designated variable but also on the type of objective (error) function used to evaluate the performance of each solution. In both cases, the variables C_{m_α}, C_{m_q}, $C_{L_{\delta_e}}$ and $C_{m_{\delta_e}}$ seem to have more impact on the final results. Variables $C_{m_{\dot{\alpha}}}$, C_{L_α} and C_{L_0} have sufficient impact on the optimization results to be considered in the optimization ($> 20\%$). As for the variables $C_{m_{\dot{\alpha}}}$ and C_{L_q}, their impact on the overall results is minimal, thus, these variables will be discarded.

These results also present the comparison of linear and quadratic representation of the

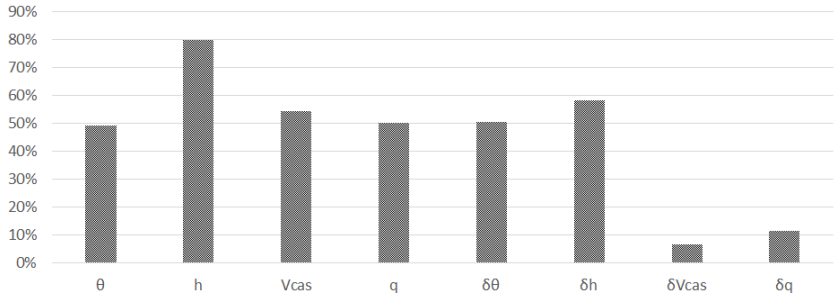
elevator δ_e influence on the aerodynamic forces. Since the quadratic option, in some cases, benefits the optimizer with increased performance, these coefficients will be used.

5.3 Single-Manoeuvre Optimization

Using the problem definition presented in Section 5.1.3 and the design variable choice from Section 5.2 the initially built model was optimized using a single flight test manoeuvre. Figure 9 shows the solution (Fig. 9a) and the error improvement (Fig. 9b).



(a) Single manoeuvre optimization graphical results.



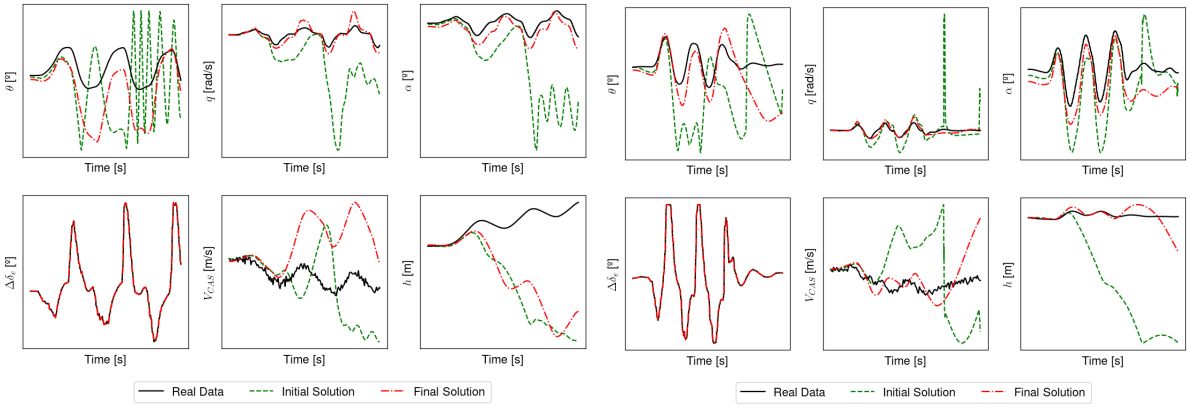
(b) Single manoeuvre optimization error improvement score.

Figure 9: System identification using a single manoeuvre for the Tekever AR5.

Both the response and improvement scores show that the optimization was successful in modifying the model to reduce the overall error between log and simulated data. Clearly, the initial model was divergent and after a certain period of time it evolves into an unstable condition (high pitch oscillations, very low angle of attack and high altitude loss). On the other hand, the optimized solution, even if only optimized during a single cycle of the manoeuvre, is capable of not only approximate the dynamics, but also maintain the aircraft within acceptable margins of the real values found. The largest improvements were found for pitch and altitude where the error was also maximum on the initial solution.

5.3.1 Solution Application

A single manoeuvre optimization is prone to error due to the lack of variability and different condition manoeuvres. For this reason, first, the solution found is tested with two example flight test manoeuvres with the same UAV. Figure 10 yield the results for the manoeuvres tested (10a and 10b), where the initial solution provided to the optimizer is also presented to verify the improvements visually.



(a) First manoeuvre. (b) Second manoeuvre.

Figure 10: Single manoeuvre optimization validation with two separate independent manoeuvres of the same Tekever AR5 aircraft.

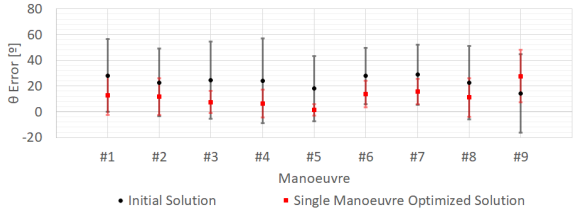
Similarly to the optimized manoeuvre case, the initial and final solution have different behaviours. The former quickly diverges for the real solution whereas the latter is capable of maintaining the dynamic of the real UAV. The second manoeuvre seems to be well fitted, however, for the first manoeuvre, both altitude and calibrated airspeed results have larger differences and seem to be more prone to error as the simulation time increases.

Nonetheless, the optimization was capable of modifying the model to be more coherent with the real log data and also maintain stability in multiple cycles of the same manoeuvre, reducing the gap between the modelled and real systems.

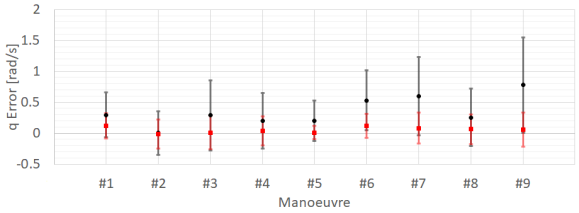
For a statistical analysis, the optimized solution was simulated with eight additional different similar manoeuvres and the mean and standard deviation of the error was calculated, as shown in Figure 11, for the used and observable error data: pitch (Fig. 11a), pitch rate (Fig. 11b), altitude (Fig. 11c) and calibrated airspeed (Fig. 11d).

As shown, the optimized solution improves the modelled dynamics by not only having a lower mean value for the majority of the errors, but also lower standard deviations, meaning the output of the optimized model is more accurately representing the real log data found. Additionally, some error cases are better fitted than the optimized manoeuvre scenario (for example comparing manoeuvre #3 with the used in optimization manoeuvre #1). This means that the optimized solution is capturing the general longitudinal dynamic qualities of the real aircraft. However, the mean error and standard deviations are not ideal, where room for improvement still exists.

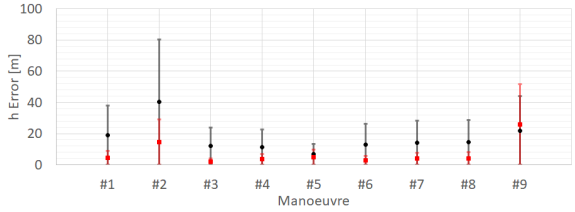
Expectedly, overall results vary from the single manoeuvre optimization shown in Section 5.3. To further increase the performance of the method, an increased number of observable data is expected to augment the system identification capabilities by introducing new dynamics of the same aircraft and reducing the case specific elements of the



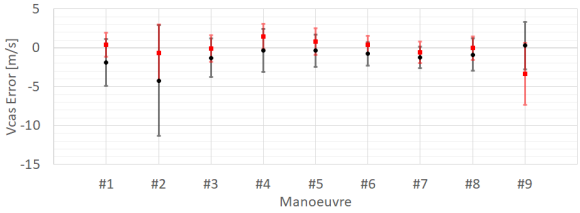
(a) Pitch θ error.



(b) Pitch rate q error.



(c) Altitude h error.



(d) Calibrated airspeed V_{CAS} error.

Figure 11: Average error and standard deviation of the error results for nine manoeuvres for the initial and final solution of the single manoeuvre optimization. Manoeuvre #1 was the used manoeuvre for the optimization.

single manoeuvre optimization.

5.4 Multi-Manoeuvre Optimization

To mitigate the single manoeuvre optimization case specific nature, a multi-manoeuvre approach to the problem was adopted. To achieve this, multiple flight test manoeuvres were collected and used to better fit the parameter estimated model.

5.4.1 Model Formulation

The combined manoeuvre optimization needs two types of data subsets for the proper validation of the model. Using as reference the surrogate modelling theory [21], from all available manoeuvres, 75% are used as training data and 25% are used as validation data. The former is fed into the optimization problem and is used to calculate the error between the simulated model and the real flight data. For this, the error function was modified to

$$E = \sum_{m=1}^{M} \sum_{i} \frac{RMSE_{\tau_i}}{RMSE_{\tau_{i_0}}} \quad , \tag{20}$$

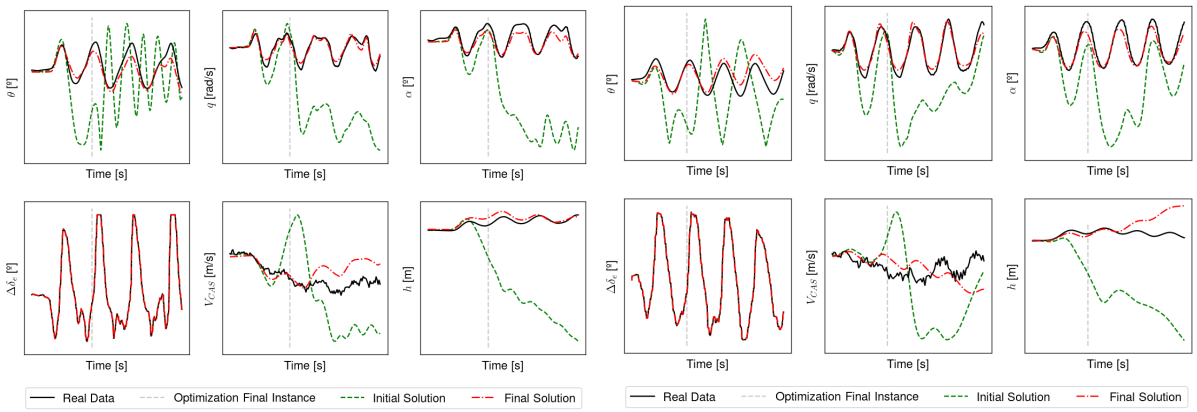
where m is the manoeuvre counter, to account for all manoeuvres in an equal weighted manner. The latter is used to prove the new found optimized model, where the model is tested with the validation data to check whether the optimization was successful.

All the manoeuvres collected correspond to equivalent flight test procedures with the same aircraft. The training and validation datasets division was done at random, where of a total of nine manoeuvres, six were selected for training and the remaining three were used for validation.

5.4.2 Optimization Results

Following the aforementioned optimization scheme, the initial model solution was optimized using six manoeuvres from a total of nine. Figure 12 presents two of these manoeu-

vres (#4 in (12a) and #7 in (12b)) with both the initial and final solutions presented.



(a) Manoeuvre #4.

(b) Manoeuvre #7.

Figure 12: Multi-manoeuvre optimization results plots for two example manoeuvres used in the optimization of the same Tekever AR5 aircraft.

Clearly, the final solution found by the optimizer is not only closer to the expected dynamics of the real flight data, but also has more stability and capability to diverge less when above the used optimization time for the manoeuvre. In the first example shown (12a), the final solution correctly models both pitch and altitude response for the shown four cycles of the nose up and nose down manoeuvres. However, the second example previews an altitude deviation as the flight time increases.

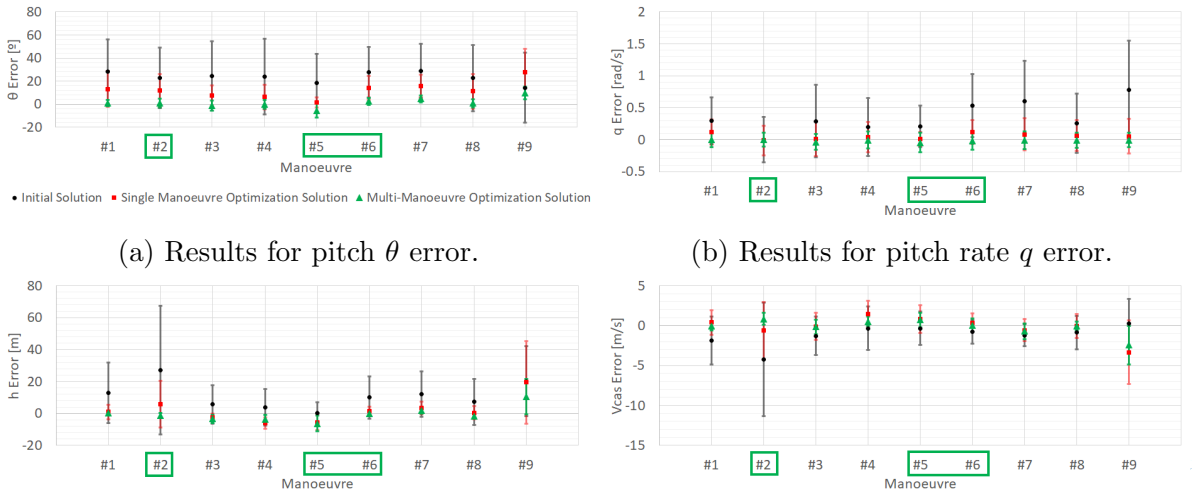
Comparing to the single-manoeuvre optimization solution, these results show that the increased number of manoeuvres inside the optimization loop revealed useful data to the algorithm that allowed for a better dynamic response convergence. The algorithm used was successful in reducing the gap between modelled and real UAV responses, diminishing the error for the system identification, for all the used manoeuvres inside the optimization process.

5.4.3 Model Validation

As seen, the final solution obtained was capable of suitably predicting the real dynamics of the aircraft, specially in the optimization flight time period used, where above this time the response of the model can deviate. However, this analysis was only for the used manoeuvres on the optimization process. To finally validate the final solution, the three left out manoeuvres for the validation process are analysed.

First, an error analysis was conducted. Figure 13 presents the mean error and standard deviations of all found solutions - initial, single manoeuvre and multi-manoeuvre - for all used observable error data: pitch (13a), pitch rate (13b), altitude (13c) and calibrated airspeed (13d). The marked manoeuvres #2, #5 and #6 are the ones used for validation purposes and the remaining were used as error source for the optimization.

Compared to Fig. 11, there is a decrease in both the mean and standard deviation errors for the final solution of the multi-manoeuvre optimization cases, where, for the used observable data, the final solution model fits well with the real flight log data. The only exception to the results found was the manoeuvre #9, where both addressed error



(a) Results for pitch θ error.

(b) Results for pitch rate q error.

(c) Results for altitude h error.

(d) Results for calibrated airspeed V_{CAS} error.

Figure 13: Average error and standard deviation of the error results for nine manoeuvres for the initial and final solutions of the multi-manoeuvre optimization.

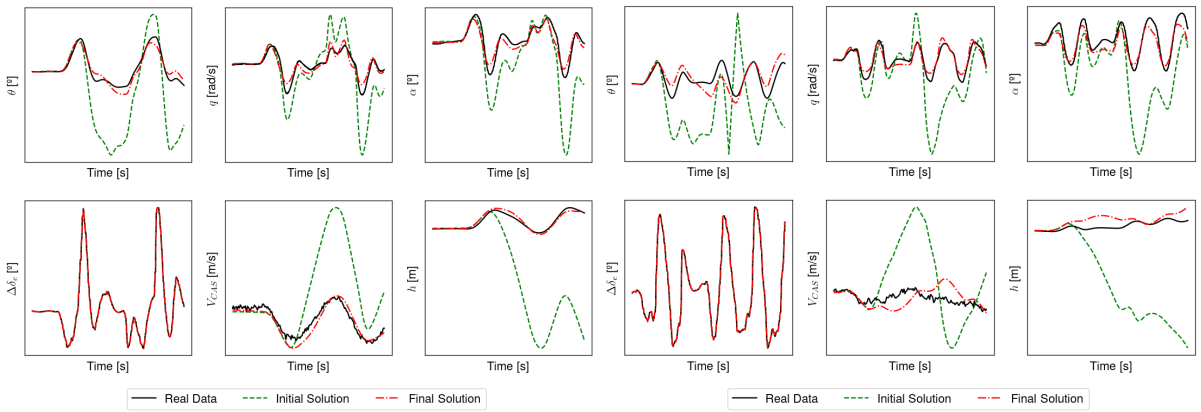
metrics were maximum. This can be due to poor flight log data of that manoeuvre or the initial conditions coherence.

Nonetheless, taking a closer look into the validation data (manoeuvres #2, #5 and #6), the final solution was able to accurately capture the real dynamics of the aircraft using single cycle portions of flight data from the other similar manoeuvres. Furthermore, the average and standard deviation error values are similar to the ones found for the optimized manoeuvres, thus, the qualities captured within the final solution model are representing the real flight log data. For a deeper look into the validation results, Figure 14 yields the response for of each manoeuvre using the initial, real and final solutions for comparison.

As seen, the final solution results have similar dynamics in relation to the real flight data in the duration of the first manoeuvre cycle and also maintain a reasonable accuracy as the number of cycles increases. The available data clearly shows an improvement in the model dynamic response and assures equivalence for the initial moments of the manoeuvre. Thus, the validation of the algorithm and final solution has found that the multi-manoeuvre optimization approach is able to accurately capture the real UAV dynamics with enough broadness that similar manoeuvres can be evaluated without recurring to new optimization analysis. The model can be used to test different manoeuvres or evaluate the aircraft longitudinal dynamics.

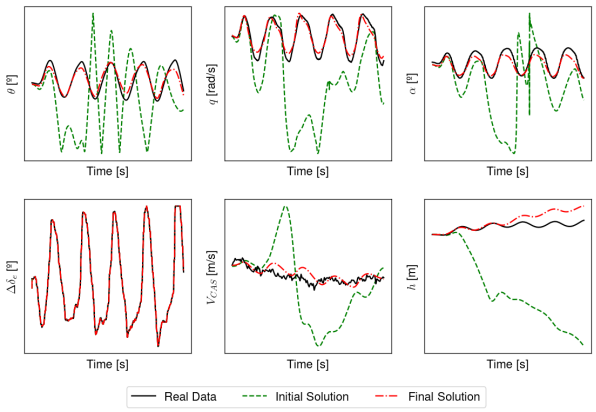
6 CONCLUSIONS

This work focused on the development of a system identification optimization framework where a six degree non-linear dynamic model (formulated in *JSBSim* and based in simple polynomial functions) was successfully modified, using the *SLSQP* gradient algorithm, to accurately predict and emulate real flight longitudinal manoeuvres. Furthermore, it has been shown that the usage of a low number of manoeuvres was sufficient to accurately correct and modify the model towards an equivalent longitudinal response, at least for a single cycle of nose up and nose down manoeuvres. The usage of routine structural flight testing was shown to be sufficient for the optimization of a longitudinal model that estimates the longitudinal dynamic characteristics of the aircraft.



(a) Results for the manoeuvre #2.

(b) Results for the manoeuvre #5.



(c) Results for the manoeuvre #6.

Figure 14: Multi-manoeuvre results using the three validation manoeuvres of the same Tekever AR5 aircraft.

The comparison between single and multi-manoeuvre environments concluded that a broader use of manoeuvres successfully captures more accurately the dynamic response of a system. However, the single manoeuvre optimization has shown to decrease the error found in the initial solution and improve the response in other equivalent manoeuvres. The training/validation subset division has shown to be important for the correct evaluation and validation of a final solution, where the multi-manoeuvre algorithm proved to be effective at determining the real UAV longitudinal qualities.

The usage of single cycle portions of complete manoeuvres was shown to be enough to correctly model the UAV and to acquire sufficient data to allow the final solution to reasonably follow n cycle manoeuvres.

ACKNOWLEDGEMENTS

The authors would like to thank Tekever for supporting the production of this article by offering all the needed data to successfully produce the analyses here shown. This work was supported by FCT, through IDMEC, under LAETA, project UIDB/50022/2020.

REFERENCES

[1] A. Klöckner. Geometry based flight dynamics modelling of unmanned airplanes. *AIAA Modeling and Simulation Technologies Conference*, August 2013. doi:10.2514/6.2013-5154. Boston, Massachusetts, USA.

[2] A. Kamal, A. M. Aly, and A. Elshabka. Tuning of airplane flight dynamic model using flight testing. *AIAA Modeling and Simulation Technologies Conference*, Januray 2015. doi:10.2514/6.2015-1595. Dallas, Texas, USA.

[3] P. Lichota and M. Lasek. Aircraft system identification from flight data recorder data. *ML-XV Mechanika w Lotnictwie (ML-XV Mechanics in Aviation)*, May 2012. Warszawa, Poland.

[4] R. Jategaonkar, D. Fischenberg, and W. von Gruenhagen. Aerodynamic modeling and system identification from flight data-recent applications at DLR. *Journal of Aircraft*, 41(4):681–691, 2004. doi:10.2514/1.3165.

[5] S. J. Koeberle, A. E. Albert, L. H. Nagel, and M. Hornung. Flight testing for flight dynamics estimation of medium-sized UAVs. *AIAA Scitech 2021 Forum*, January 2021. doi:10.2514/6.2021-1526. Virtual Event.

[6] B. M. Simmons, H. G. McClelland, and C. A. Woolsey. Nonlinear model identification methodology for small, fixed-wing, unmanned aircraft. *Journal of Aircraft*, 56(3): 1056–1067, 2019. doi:10.2514/1.C035160.

[7] J. Harris, F. Arthurs, J. Henrickson, and J. Valasek. Aircraft system identification using artificial neural networks with flight test data. *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 679–688, June 2016. doi:10.1109/ICUAS.2016.7502624. Arlington, Virginia, USA.

[8] T. Stachiw, A. Crain, and J. Ricciardi. A physics-based neural network for flight dynamics modelling and simulation. *Advanced Modeling and Simulation in Engineering Sciences*, 9(13), April 2022. doi:10.1186/s40323-022-00227-7.

[9] R. Cao, Y. Lu, and Z. He. System identification method based on interpretable machine learning for unknown aircraft dynamics. *Aerospace Science and Technology*, 126:107593, July 2022. doi:10.1016/j.ast.2022.107593.

[10] V. Klein and E. Morelli. *Aircraft System Identification: Theory and Practice*. AIAA education series. American Institute of Aeronautics and Astronautics, 2006. ISBN 9781563478321.

[11] R. Jategaonkar. *Flight Vehicle System Identification: A Time Domain Methodology*. 2nd edition, 2015. ISBN 978-1-62410-278-3.

[12] K.-S. Wang and K. Iliff. Retrospective and recent examples of aircraft parameter identification at NASA dryden flight research center. *Journal of Aircraft*, 41:752–764, July 2004. doi:10.2514/1.332.

[13] Tekever. URL `https://www.tekever.com/`.

[14] B. Stevens and F. Lewis. *Aircraft Control and Simulation*. Wiley, 1992. ISBN 9780471613978.

[15] W. Phillips. *Mechanics of Flight*. Engineering case studies online. Wiley, 2004. ISBN 9780471334583.

[16] N. Hoffer, C. Coopmans, A. Jensen, and Y. Chen. Small low-cost unmanned aerial vehicle system identification: A survey and catagorization. *2013 International Conference on Unmanned Aircraft Systems*, May 2013. doi:10.1109/ICUAS.2013.6564775. Atlanta, Georgia, USA.

[17] G. Dussart, V. Portapas, A. Pontillo, and M. Lone. Flight dynamic modelling and simulation of large flexible aircraft. In K. Volkov, editor, *Flight Physics*, chapter 3. IntechOpen, Rijeka, 2018. doi:10.5772/intechopen.71050.

[18] M. Drela and H. Youngren. Athena vortex lattice. URL `https://web.mit.edu/drela/Public/web/avl/`.

[19] J. Berndt. JSBSim: An open source flight dynamics model in C++. *AIAA Modeling and Simulation Technologies Conference and Exhibit*, August 2004. doi:10.2514/6.2004-4923. Providence, Rhode Island, USA.

[20] J. S. Gray, J. T. Hwang, J. R. R. A. Martins, K. T. Moore, and B. A. Naylor. OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization. *Structural and Multidisciplinary Optimization*, 59(4):1075–1104, April 2019. doi:10.1007/s00158-019-02211-z.

[21] A. Forrester, A. Sobester, and A. Keane. *Engineering Design Via Surrogate Modelling: A Practical Guide*. Progress in astronautics and aeronautics. Wiley, 2008. ISBN 9780470060681.