Improved Kirchhoff Stall Model Parameter Estimation Accuracy Through Optimal Data Slicing

Brill, P.A.R.; Pool, D.M.; de Visser, C.C.

DOI
10.2514/6.2025-1248

Publication date
2025

Document Version
Final published version

Published in
Proceedings of the AIAA SCITECH 2025 Forum

AIAA SciTech Forum
6-10 January 2025, Orlando, FL
AIAA SCITECH 2025 Forum

10.2514/6.2025-1248

Improved Kirchhoff Stall Model Parameter Estimation Accuracy through Optimal Data Slicing

Pieter A.R. (Patrick) Brill*, Daan M. Pool† and Coen C. de Visser‡
Delft University of Technology, Delft, Zuid-Holland, The Netherlands

To improve the safety of commercial air transport, pilots are required to train on simulators to recognize the characteristics of an impending stall and subsequently correctly recover from it. To prevent negative training, it is important that the accuracy of the used simulation models is sufficiently high. A key approach for modeling the nonlinear, unsteady aerodynamic effects during the stall is by using Kirchhoff's theory of flow separation. However, widespread difficulties exist in correctly estimating the stall-related parameters of nonlinear flow separation models from flight test data. Therefore, the research in this paper aims to increase the obtained model accuracy by making optimal use of already existing flight data via introduction of a slice-based modeling method. This is done by analyzing the change in the parameter estimate values when applying the system identification procedure to sliced partitions of simulated flight data, for both the pre-stall and post-stall phases. These partitions incrementally increase in size with time from the stall initiation. The simulation data is generated to be representative of the available flight test data, but with known 'truth' values for all estimated model parameters. The estimated value for each partition was compared to the true parameter setting in the simulation model used to create the data. It was also investigated whether this coincided with points of increased Fisher information in the data. Manually, an optimal window was found for each parameter for which the estimated value and truth value were equal and sufficient Fisher information was present. For the stall-related parameters the optimal window is often not more than 10 s wider than the stall. For the linear stability and control derivatives it is found that using more data generally results in a better estimate. Finally, the optimal window sizes were used for parameter estimation on the real flight test data. Even though this method represents a prototype, in more than half of the validation cases a decrease in MSE of 10 % to 35 % was achieved. This shows that the new slice-based modeling method is able to improve the accuracy of nonlinear stall models without the need to gather more flight data and may have applications that reach beyond the realm of stall modeling.

Nomenclature

Roman symbols

a_1	=	stall abruptness parameter
A_x	=	longitudinal acceleration
A_z	=	vertical acceleration
c	=	chord length
C_D	=	drag coefficient
C_L	=	lift coefficient
C_L^{nl}	=	nonlinear unsteady C_L at high angles of attack
C_X	=	longitudinal force coefficient
C_Z	=	vertical force coefficient
D	=	dispersion matrix
J	=	nonlinear estimation objective function
k	=	timestamp
M	=	Fisher information matrix
m	=	aircraft mass
MSE	=	mean squared error
N_o	=	total number of outputs
N_p	=	total number of parameters
Q	=	quartile
R	=	noise covariance matrix
$RRMS$	=	relative root mean squared error
S	=	wing surface area

*MSc student (graduated), Control and Simulation section, Faculty of Aerospace Engineering, P.O. Box 5058, 2600GB Delft, The Netherlands; patrickbrill@outlook.com

†Assistant Professor, Control and Simulation section, Faculty of Aerospace Engineering, P.O. Box 5058, 2600GB Delft, The Netherlands; d.m.pool@tudelft.nl. Associate Fellow AIAA.

‡Associate Professor, Control and Simulation section, Faculty of Aerospace Engineering, P.O. Box 5058, 2600GB Delft, The Netherlands; c.c.devisser@tudelft.nl. Member AIAA.

$S(k)$	=	output sensitivity matrix
t	=	time
V_{TAS}	=	true airspeed
X	=	airfoil flow separation point
x	=	nonlinear regressor
y	=	measurement signal
$\hat{y}$	=	model output

Greek symbols

α	=	angle of attack
α^*	=	angle of attack for which $X = 0.5$
δ_a	=	aileron deflection
δ_e	=	elevator deflection
δ_r	=	rudder deflection
θ	=	parameter
$\hat{\theta}$	=	estimated parameter
ρ	=	air density
σ	=	standard deviation
σ^2	=	variance
τ_1	=	angle of attack rate effect time constant
τ_2	=	stall hysteresis time constant

I. Introduction

The most significant cause of fatal accidents in commercial air transport is loss of control in-flight [1]. Loss of control in-flight occurs when the pilots of an aircraft cannot recover the aircraft from an adverse flight condition outside its normal operational envelope [2]. An example of this is aerodynamic stall, which is defined as "an aerodynamic loss of lift caused by exceeding the critical angle of attack" [3]. To train pilots to handle these situations properly, the International Civil Aviation Organisation has urged member states to implement upset prevention and recovery training in their regulations [3]. In response to this, the European Aviation Safety Agency has updated its legislation on flight simulation training devices [4], stating that these must accurately model the aircraft behaviour in the stall to train pilots effectively in recognizing and handling the aircraft stall characteristics [5, 6].

Aerodynamic models valid in nominal flight phases are not immediately valid near the stall, where nonlinear and unsteady effects are present [7]. Therefore, the model needs to be extended with nonlinear terms, but this is not trivial. Often, the conventional stability and control derivatives are estimated separately for the nominal flight phase and near the stall, requiring parameter scheduling, e.g. as in [8, 9].

An elegant yet powerful method to model some of the characteristic behaviour of an aircraft in the stall without the need for scheduling is through Kirchhoff's theory of flow separation [10]. Flow separation models can be used to identify aircraft stall models from flight data for both the longitudinal and lateral motions of the aircraft [11–13]. However, in practice, difficulties still arise in estimating and validating some of the stall model parameters with sufficient reliability [14, 15]. This issue is usually accounted to a lack of dynamic excitation present in the available flight data in combination with the short duration of the stall event.

Kirchhoff's theory is also used in stall modeling research at the Faculty of Aerospace Engineering of the Delft University of Technology. This research is centered around the Delft University Aircraft Simulation Model and Analysis Tool (DASMAT) [16], which models the nominal flight dynamics of TU Delft's Cessna Citation II laboratory aircraft (PH-LAB). Multiple research efforts have improved and extended this model from the normal flight envelope to the stall [17–21]. It has been shown that this model, in combination with the incorporated buffet model [18, 22], provides aircraft stall behaviour in a full flight simulator that matches the actual aircraft well [22, 23]. However, estimation accuracy could still be improved [19, 21], especially for the flow separation model parameters describing the nonlinear and unsteady effects.

Sufficient dynamic excitations in flight testing are crucial as this increases the amount of information that is present in the data for the to-be-estimated parameters. This is usually achieved by applying specific control inputs throughout flight test runs. These may be conventional inputs applied by the pilots of the aircraft [24, 25] or may be optimal inputs that are specifically tailored to increase the amount of information for a certain model parameter [26–28]. However, this approach is only possible if *a priori* information about that parameter is available. The optimal-input approach is based on the concept of Fisher information [24, 29]. However, introducing this approach also requires additional flight testing. Furthermore, a drawback is that simply gathering more flight test data is often not an option due to high costs, lack of time, or both.

The research in [8, 9] makes use of a second, distinctively different method with the same goal of increasing stall model accuracy, but without the need for additional flight tests. This is the practice of partitioning data, introduced in [30]. This can result in an improved stall model, but requires scheduling according to the associated angle of attack of the partition on which the estimation was applied. Apart from the model accuracy increase, this research also showed that different partitions associated with different angles of attack can deliver different estimates for the same parameter.

However, it is has never been demonstrated whether the slicing of data also has a similar positive effect on any model that also involves nonlinear terms and associated parameters.

In summary, often research efforts have difficulties in reliably identifying the nonlinear parameters of stall models from the limited data available, i.e., for models using Kirchhoff's theory of flow separation [19, 21]. In these efforts, as much data as is available is generally used in the estimation procedure. No attention is paid towards which parts of the data may be valuable to the nonlinear Kirchhoff stall parameters and which parts may be valuable (only) to the linear stability and control derivatives. This may cause nonlinear stall parameters to model parts of the linear nominal behaviour in the data and vice versa, which is not in line with their intrinsic purpose. On the other hand, there is proof that different parts of already available flight test data deliver different estimates, as seen in [30]. This raises the question: can model accuracy be improved by choosing specific parts of data to which to apply the estimation of a specific parameter, in contrary to the usual practice of simply using all data for all parameters regardless of the circumstances under which the data were gathered?

The main contribution of this paper is to improve stall model accuracy through optimal data slicing by explicitly analyzing Kirchhoff stall parameter estimate behaviour when different slices of data are used for model identification. A new slice-based modeling method is introduced and new simulation data is generated from a simulation with the model structure of [19] with known parameter values set to objectively assess estimation accuracy. The simulated flight trajectory is designed to closely resemble available real stall flight test data. Via data slicing, partitioning and Fisher information analysis, it is for the first time possible to directly observe which part of the simulated data causes a change in the value of a parameter estimate. This is named '*parameter estimate behaviour*' in the remainder of this paper. Though not part of the modeling method, this analysis can also be performed on flight test data for more insights that can validate the method. By comparing the parameter estimate behaviour of the simulation data to the actual simulation model value, an optimal time window can be found for all input types given during the stall, for the pre-stall and post-stall phases, and for every separate parameter. The optimal window sizes are subsequently applied to the estimation of every parameter using the real flight test data. The accuracy achieved with the slice-based method is compared to that achieved when the normal modeling method from [19] is applied where all available data is used.

This paper is structured as follows. In section II, the research methodology is explained extensively. In section III, the results of this method are presented, after which they are analyzed and discussed in section IV. This paper concludes with section V.

II. Methodology

A. Aerodynamic Stall Model

Before discussing the full slice-based modeling method, the assumed aerodynamic stall model must be introduced. In order to capture the unsteady and nonlinear behaviour in the stall, Kirchhoff's theory of flow separation is used [10, 11]. This theory states that the nonlinear and unsteady behaviour of the lift coefficient at high angles of attack C_L^{nl} is a function of angle of attack α and flow separation point X:

$$C_L^{nl}(\alpha, X) = C_{L_\alpha} \left(\frac{1 + \sqrt{X}}{2} \right)^2 \alpha \tag{1}$$

The movement of the flow separation point X along the chord of the wing is itself also a function of the angle of attack α and the angle of attack rate $\dot{\alpha}$, such that $X = X(\alpha, \dot{\alpha})$. Variations in X can be described by an ordinary differential equation [11]:

$$\tau_1 \frac{dX}{dt} - X = \frac{1}{2}\{1 - \tanh\left[a_1(\alpha - \tau_2\dot{\alpha} - \alpha^*)\right]\} \tag{2}$$

The value of X is 1 when the flow is fully attached (separation point at trailing edge) and 0 when it is fully separated (separation point at leading edge). The parameters a_1, α^*, τ_1 and τ_2 are the nonlinear stall model's parameters. The a_1 parameter controls the abruptness of the stall, α^* is the angle of attack where $X = 0.5$, τ_1 influences the aerodynamic lag in the flow's separation, and τ_2 models the hysteresis effect. The value of these parameters depends on aircraft type and are ultimately the parameters that need to be estimated correctly to result in highest stall model accuracy.

The stall model investigated in this paper is the model as proposed and derived in [19]. The equation for the lift coefficient C_L of this model is given in Eq. (3). This model structure was identified from flight test data by use of orthogonal function modeling [31]. Only the C_L model is considered in this paper as it was found in preliminary research that this contains most information on the stall's flow separation parameters [19, 21].

$$C_L = C_{L_0} + C_{L_\alpha}\left(\frac{1+\sqrt{X}}{2}\right)^2 \alpha + C_{L_{\alpha^2}}(\alpha - 6^\circ)_+^2 \tag{3}$$

In Eq. (3), $(\alpha - 6^\circ)_+^2$ is a spline function that activates only when $\alpha \geq 6^\circ$. The stability and control derivatives C_{L_0}, C_{L_α} and $C_{L_{\alpha^2}}$ are the linear parameters of the model. The parameter values estimated by [19] and the absolute lower and upper bounds of the search space applied during estimation for these parameters are given in Table 1. These settings are used as the main reference in this paper.

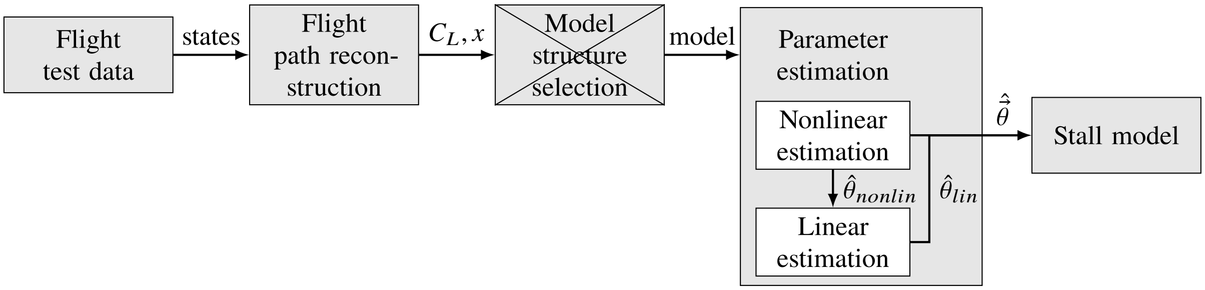
Table 1 Estimated parameter values and used search space bounds for estimation of the parameters, as in [19].

		Bounds	
θ_i	Value	Lower	Upper
a_1, –	27.6711	15	40
α^*, rad	0.2084	0.1	0.35
τ_1, s	0.2547	0.001	0.8
τ_2, s	0.0176	0	0.5
C_{L_0}, –	0.1758	0.1	0.4
C_{L_α}, –	4.6605	2	6
$C_{L_{\alpha^2}}$, –	10.7753	0	20

B. Overview of the Slice-based Modeling Method

The sliced-based modeling method introduced in this paper is a further development of the method introduced in [19]. The modeling method used there, in the remainder of this paper called the 'normal modeling method', is depicted in Figure 1. In the normal method, the gathered flight data is directly and completely passed to the flight path reconstruction. From the reconstructed states, the lift coefficient C_L is calculated. This C_L is passed to the model structure selection containing the orthogonal function modeling, together with any other reconstructed state that may be needed as a potential regressor x. In this current paper, the model structure selection is not performed, but the final model structure of Eq. (3) is always used. This makes it possible to perform an analysis on a distinct set of known parameters, including calculation of the Fisher information. A nonlinear and linear estimation are performed, resulting in the final parameter estimate vector $\hat{\hat{\theta}}$. This completes the stall model.

Fig. 1 Overview of the normal modeling method, as in [19]. The model structure selection that takes place in [19] has not been incorporated in the research described in this paper.



The slice-based modeling method is an extended version of the normal method described above. It is depicted in Figure 2. The slice-based method contains all but one of the steps of the normal estimation procedure, represented by the boxes 'Flight test data', 'Flight path reconstruction', a slightly different parameter estimation called 'Slice-based parameter estimation', and finally the estimated 'Stall model'. As mentioned, the model structure selection is not included, as the model structure is set *a priori*. The main addition of the new slice-based method is that the parameter estimation now uses an optimal time window for each separate parameter, indicated by the optimal slice number $n_{slice_{opt}}$.

Finding this optimal window starts with generating a simulation data set. The parameter estimate behaviour that results from this simulation data is used as a substitute for the expected, but unknown, parameter estimate behaviour that may result from flight test data. The simulated data is also passed through the same flight path reconstruction step also applied to flight test data [19]. This data is then sliced and partitioned in three different manners. The nonlinear parameters $\hat{\theta}_{nonlin}$ and the linear parameters $\hat{\theta}_{lin}$ in the model are estimated for all three slicing types and every candidate data partition within these types. This makes it possible to analyze the behaviour of the parameter estimates in time by use of the median of the parameter estimates $\tilde{\hat{\theta}}$, the quartiles Q of their distribution and the associated Fisher information content M. From this behaviour, the sought-after optimal slice number $n_{slice_{opt}}$ can be selected for each separate parameter for each type of stall present in the flight test data, both for pre-stall and post-stall phases. In the following subsections, all of the above steps are discussed in more detail.

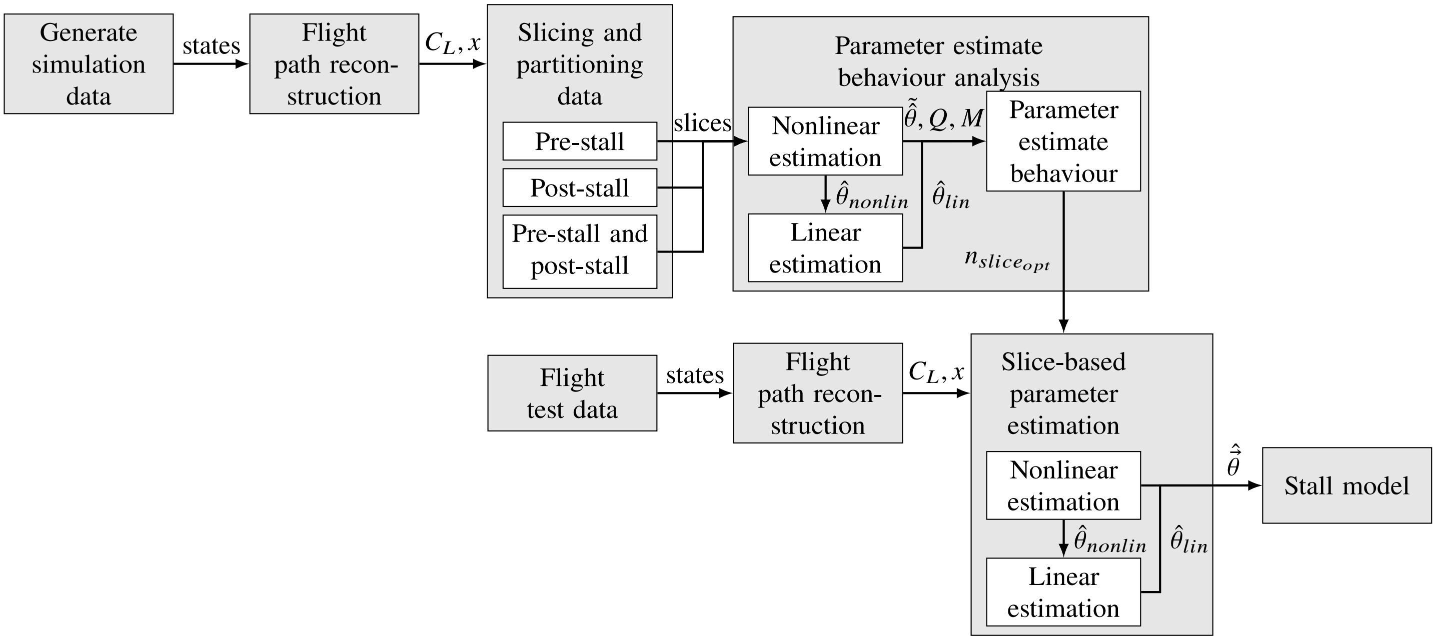


Fig. 2 Overview of the slice-based modeling method, as introduced in this paper.

C. Data Sets

In the research described in this paper two types of data are used: simulated flight data and real test flight data. This is different from existing stall modeling research [18, 19, 21], in which only real flight data is used. The use of simulated flight data creates the possibility to directly compare the estimated parameters with their true known values, rather than only model output validation. This can deliver more direct insight into the performance of the parameter estimation procedure. Especially, because the simulation data is generated to mimic the available real flight test data. After the analysis of the simulated data, the real flight data is then used for final parameter estimation and validation. The two types of data sets in this paper are discussed below.

1. Flight Test Data

Real flight data was gathered in two stall test flights by the TU Delft Cessna Citation II research aircraft in 2018 [19] and 2019 (not connected to a specific research paper). Three different types of stalls are chosen that differ from each other by the control input given within the stall:

- *No inputs.* To provide a baseline for this research, the most straightforward type of stall was chosen: the symmetric stall with no inputs. There are three such stalls in the 2018 dataset.
- *Inputs $-\delta_a$ 3-2-1-1 and $-\delta_e$ 3-2-1-1.* The 3-2-1-1 stalls were chosen to investigate the usability of this widely-applied inputs in stall modeling. The combination of the two surfaces was chosen as there are no stalls in the data with solely a δ_e input. There are thirteen such stalls in the 2019 dataset.

- *Inputs $-\delta_a$ wiggle and $-\delta_e$ wiggle.* The wiggle input was specifically developed in [25] and used in [19] as a pilot-applied input for use in stall flight modeling, to apply large deflections without leaving the desired stall state of the aircraft. This type with two control surfaces is chosen to be able to compare the parameter estimate behaviour in this stall type to the behaviour of the 3-2-1-1 stall type. There are six such stalls in the 2018 dataset.

To make the terminology for the real flight data consistent with the simulated flight data, for a certain stall run of a certain input type the term 'realization' is used. Thus, there are three, thirteen and six realizations for the different input types in the flight test data. The aim of the simulation data is to mimic the data gathered in the real flight tests, with the three input types present therein.

2. Simulation Data

The simulated flight data was created by use of the aerodynamic model of the PH-LAB Cessna Citation II research aircraft as developed by [19], which also contains the stall buffet model as developed by [18]. Simulated stall maneuvers are performed using the 'stall autopilot' developed in [22, 23] for piloted flight simulator experiments.

The simulated flight data starts with the aircraft trimmed in steady, straight, symmetric flight. To improve the unscented Kalman filter's convergence in the flight path reconstruction, successive excitations are introduced during this initial nominal flight phase. A 3-2-1-1 input followed by return to symmetric flight is given successively for the aileron, elevator and rudder. This introduces excitations in all six degrees of freedom.

After this a simulated 'stall run' is included, which is used in the parameter estimation procedure. First, steady straight symmetric flight is maintained for 120 s. A wings-level hold controller is active during this time to prevent the aircraft from entering its slightly unstable spiral eigenmode. Hereafter, the stall autopilot is engaged to perform a symmetric stall. The stall entry and recovery procedure are discussed in detail in [22]. The stall autopilot is turned off when the original altitude is almost reacquired. Then, the aircraft is steered back to steady straight symmetric flight, which is retained for 120 s by use of an altitude hold and a roll attitude hold controller.

An overview of the simulated flight and the definition of the different phases contained therein is given in Table 2 and visualized in Figure 3. The importance of Phase 7-12 to the current research is discussed further in subsection II.E.

Table 2 Phases in the simulated flight data.

Phase	t_{begin}, s	t_{end}, s	Description
0	0	30	steady straight symmetric flight
1	30	50	$-\delta_a$ 3-2-1-1 (t_{input} = 40 s)
2	50	90	$-\delta_a$ 3-2-1-1 recovery
3	90	110	$-\delta_e$ 3-2-1-1 (t_{input} = 100 s)
4	110	150	$-\delta_e$ 3-2-1-1 recovery
5	150	170	$+\delta_r$ 3-2-1-1 (t_{input} = 160 s)
6	170	210	$+\delta_r$ 3-2-1-1 recovery
7	210	330	steady straight symmetric flight
8	330	369	stall entry
9	369	382	stall (buffet activated)
10	382	440	stall recovery
11	440	500	return to trimmed flight
12	500	560	steady straight symmetric flight

Three different stall types were simulated to mimic the real flight data. These three input types in the simulation data are shown in Figure 4. The first stall type includes no additional inputs and is the standard stall as flown by the stall autopilot. This is the baseline to which the other types can be compared. The second type has a 3-2-1-1 input on both the elevator and the aileron. To achieve the desired high angles of attack, a constant δ_e input of -0.11 rad was set at the start of the stall, and on top of this a 0.02 rad 3-2-1-1 input was applied. For the δ_a, a 0.04 rad 3-2-1-1 input was applied to the trim deflection. The last type is the wiggle input, also on the elevator and aileron. A MATLAB `sawtooth()`-signal was used to manually tweak the inputs to imitate the inputs from the test flight data as closely as possible. Important to note is the different 'frequency' for the δ_a and δ_e inputs.

Finally, a random noise signal with the same standard deviation as in the unscented Kalman filter noise and measurements covariance matrices in the flight path reconstruction is added to all states and measurements. Also, a known bias is added to the input signals (linear accelerations and angular velocities). In reality, the α and β are

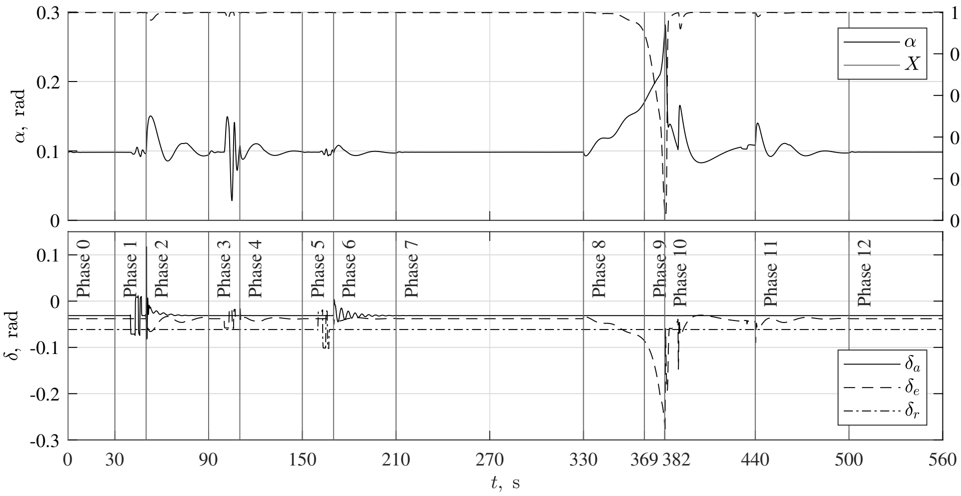


Fig. 3 Overview of the phases from Table 2 in the simulated data.

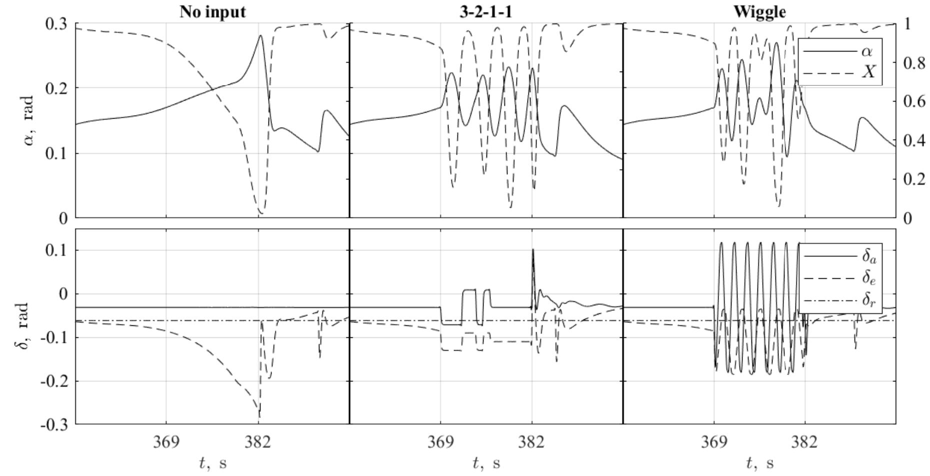


Fig. 4 Overview of the three simulated input types within the stall (phase 9), showing the angle of attack α and flow separation X, as well as the inputs δ_a, δ_e and δ_r. Note the non-zero aileron and rudder deflections in trimmed flight, caused by the non-zero C_{ℓ_0} and C_{n_0} coefficients as found by [19] present in the simulation model.

measured by the air data boom installed on the nose of the aircraft. Therefore, the α and β signals are passed through the formula for the air data boom corrections [32], such that the final signals behave as if they were measured by the vanes on the air data boom. The simulations are run 30 times to acquire 30 differently seeded realizations in order to obtain a statistically relevant sample size, i.e., reach the central limit theorem's minimum sample size.

D. Flight Path Reconstruction

The flight path reconstruction is performed to filter out noise and bias from the measured aircraft states and measurements prior to parameter estimation. It is needed for both the simulation data as well as the flight test data. Flight path reconstruction is performed using an Unscented Kalman filter (UKF) [33, 34]. It has been shown that the UKF achieves higher reliability in estimation of the states of nonlinear systems when compared to the Extended Kalman filter, although at higher computational cost [35–37]. As the Kalman filter procedure only has to be performed once per dataset of a flight, the higher reliability is more important than the added computational cost, as in earlier TU Delft stall modeling research [17–19]. It is shown in [37] that the UKF is also effective for sensor fusion, which is also required in this research. The kinematic and measurement model applied in the UKF are taken directly from [19].

Then, using the reconstructed states, the longitudinal force coefficient C_X and vertical force coefficient C_Z in the aircraft reference frame are calculated via:

$$C_X = \frac{mA_x}{\frac{1}{2}\rho V_{TAS}^2 S} \tag{4}$$

$$C_Z = \frac{mA_z}{\frac{1}{2}\rho V_{TAS}^2 S} \tag{5}$$

In these equations, m is the aircraft mass calculated from the available mass model and A_x and A_z are the reconstructed accelerations of the aircraft center of gravity in longitudinal and vertical direction, respectively. The calculated air density at the reconstructed altitude is denoted by ρ and the reconstructed true airspeed by V_{TAS}. S is the aircraft wing surface. Thereafter, the lift coefficient C_L is calculated via:

$$C_L = C_X \sin\alpha - C_Z \cos\alpha \tag{6}$$

This C_L is then the reference model data for the nonlinear and linear parameter estimation, in combination with the reconstructed states for the angle-of-attack α and its derivative in time $\dot{\alpha}$, i.e., the regressors x.

E. Slicing and Partitioning Data

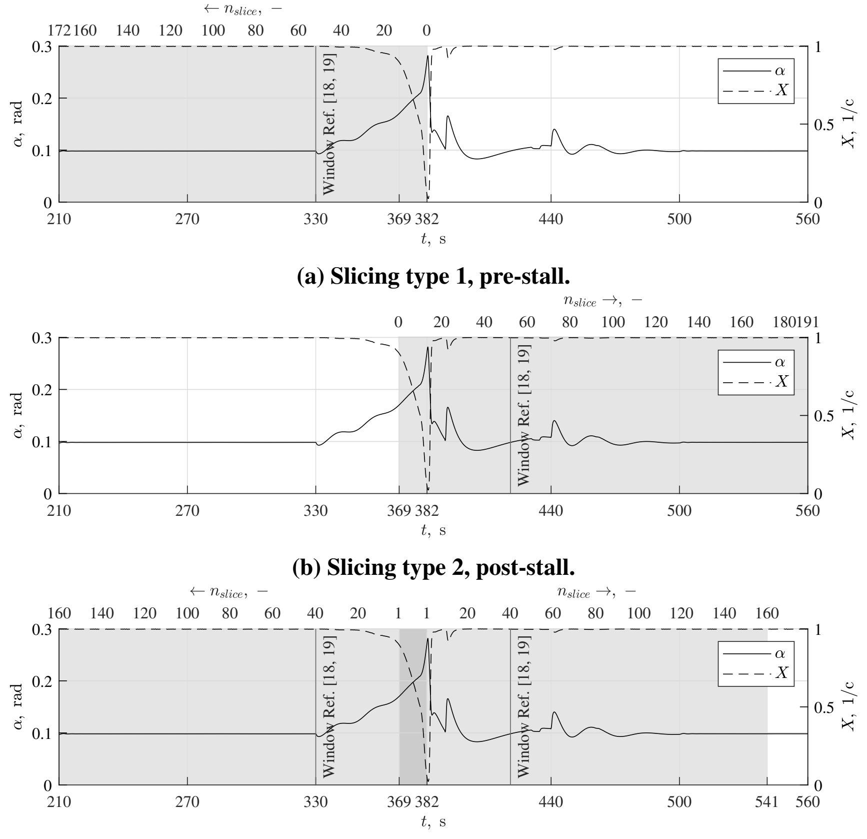
With data slicing, the simulation data is partitioned in time into slices of different widths. For this, Phase 7 to 12 from Table 2 are of importance as they are the basis of the different slicing types considered in this paper. Phase 7 is the start of the steady straight symmetric flight foregoing the stall. Phase 8 is the start of the deceleration into the stall. Phase 9 is the stall, whose entry is defined by the beginning of the stall buffet. The start of phase 10, the stall recovery, is when the angle of attack has reached its maximum value and airspeed begins to increase again. The stall buffet may still be active at that point. The recovery ends and phase 11 starts when the original altitude is reacquired. Phase 11 and 12 together contain 120 s of (quasi) steady straight symmetric flight. Phases 7 to 12 together add up to 350 s of data.

These 350 s of data are divided into 350 ‘slices’, each containing 1 s of data. Combinations of multiple slices are called a ‘partition’, which in this paper are constructed from the slices in three different manners. These three manners are called the ‘slicing types’. In a later step, the parameters can be estimated for every partition, which is the core of the parameter estimate behaviour analysis. Whenever from this analysis an optimal pre-stall and post-stall partition are chosen with a starting time and end time expressed in a corresponding slice number n_{slice}, they comprise a ‘window’.

The three different slicing types considered in this research are shown graphically in Figure 5. This figure indicates the direction and numbering of the slices n_{slice} and the respective times t they are located at. The vertical lines in the figure indicated by "Window Ref. [18, 19]" correspond approximately to the data window used in [18, 19] to perform the estimation procedure of each stall. In these research efforts, the cutoff points were usually arbitrarily chosen to be around the beginning of the stall entry and somewhere during stall recovery. It is worthwhile to investigate if this may have influenced the estimated stall parameters, which is discussed in section III.

Slicing type 1 contains partitions that start with the last second of data in the stall and that increase in size towards the pre-stall phase. Slicing type 2 contains partitions that start with the first second of data in the stall and that increase in size towards the post-stall phase. For slice types 1 and 2, the first partition contains 1 s of data and the last 172 s and 191 s, respectively. The first partition of slicing type 3 contains all slices that make up the stall and the partitions increase in size towards both sides of the stall (each increment adds 2 slices of 1 s). For slice type 3, the first partition contains 13 s of data and the last 331 s.

In the slice-based modeling method, the parameter estimate behaviour analysis is not performed for the real flight test data. This has no added value as there is no actual parameter value that the estimates can be compared with, in contrast to the simulation data where the values in the model are set to a known true value. However, as part of the proof of concept, the parameter estimate behaviour of the real flight data is discussed in the results of section III. It is discussed to what extent the simulation data is representative of the real flight test data. In the case of the flight test data, the phases as in Table 2 can also be identified by closely examining the behaviour of the aircraft in time and assigning parts around the stall to a certain stall phase. Note that some phases before stall entry and after stall recovery may be of variable length or may not be present at all.



(a) Slicing type 1, pre-stall.

(b) Slicing type 2, post-stall.

(c) Slicing type 3, pre-stall and post-stall.

Fig. 5 Overview of the three slicing types.

F. Parameter Estimate Behaviour Analysis

The parameter estimate behaviour analysis of the simulation data consists of two main parts. The first part is the parameter estimation for all partitions of the three slicing types. The second part is visualizing the parameter estimate behaviour by use of metrics $\tilde{\hat{\theta}}$ and Q describing the median and distribution of the parameter estimates across the 30 data realizations and by calculating the Fisher information M.

1. Nonlinear and Linear Parameter Estimation

The parameter estimation procedure is based on the methodology presented in [19] and consists of a nonlinear estimation of the stall parameters, followed by a linear estimation of the stability and control derivatives.

The nonlinear estimation is performed by use of the `fmincon` optimizer in MATLAB. The parameter estimates are found by minimizing the objective function $J(\vec{\theta}, x)$, which is the mean squared error (MSE) of the lift coefficient:

$$\hat{\vec{\theta}} = \arg\min_{\vec{\theta}} J(\vec{\theta}, x) \quad \text{with } J(\vec{\theta}, x) = \frac{1}{n}\left(\hat{C}_L(\vec{\theta}, x) - C_L\right)^T \left(\hat{C}_L(\vec{\theta}, x) - C_L\right) \tag{7}$$

In this equation, x are the nonlinear regressors (α and $\dot{\alpha}$) and C_L is the lift coefficient, all calculated from the flight path reconstruction. The total number of data points is denoted by n. The parameter vector $\vec{\theta}$ is defined as:

$$\vec{\theta} = \begin{bmatrix} a_1 & \alpha^* & \tau_1 & \tau_2 & C_{L_0} & C_{L_\alpha} & C_{L_{\alpha^2}} \end{bmatrix}^T \tag{8}$$

Lastly, $\hat{C}_L(\vec{\theta}, x)$ is the model output for the lift coefficient, calculated with the parameter estimates currently obtained from the optimization routine. It is calculated by numerically solving Eq. (2) with the current parameter estimates, resulting in X. This is then used in Eq. (3) to calculate $\hat{C}_L$.

The nonlinear optimization routine of `fmincon` makes use of the gradient $\frac{\partial J(\vec{\theta},x)}{\partial \vec{\theta}}$ of the cost function surface $J(\vec{\theta},x)$. Calculating this is not trivial, as discussed in [19]. This calculation is related to that of the Fisher information matrix, as discussed in subsubsection II.F.3, and here performed numerically.

The parameter estimation is performed for 500 different initial conditions, with each initial condition consisting of seven parameter values randomly generated from a uniform distribution with upper and lower bounds as given in Table 1. The global optimum is regarded as the median of all optima found that come within 5 % of the lowest found final cost function value. The stall parameters following from the nonlinear optimization are set constant for the following linear estimation. The linear estimation is a straightforward ordinary least squares estimation with $\left[\left(\frac{1+\sqrt{X}}{2}\right)^2 \alpha\right]$ regarded as an extra regressor. It is calculated by numerically solving Eq. (2) with the found nonlinear parameter estimates. From the linear estimation the final estimates of C_{L_0}, C_{L_α} and $C_{L_{\alpha^2}}$ are obtained.

2. Parameter Estimate Behaviour: Optima Distribution Metrics

The optima distribution metrics can be calculated for any individual parameter, or they can be compared by their normalized metrics. The individual parameter behaviour of any set of realizations can be visualized in boxplots. This shows the median $\tilde{\hat{\theta}}$, the inter-quartile range $Q_{2,3}$ (consisting of quartiles 2 and 3) and the full range of optima excluding outliers Q_{1-4} (consisting of all quartiles 1 to 4). Outliers are defined as being more than 1.5 times the inter-quartile range away from the inter-quartile range. These three metrics are calculated for every partition of the three slicing types, such that a boxplot can be drawn at every slice number n_{slice}. This individual behaviour can be analyzed for both the simulation data and the test flight data.

In order to compare the behaviour of the parameter estimates to each other, these metrics can also be normalized with respect to the actual parameter value. This can only be done for the simulated flight data where the actual value is known. The normalised bias of the parameter estimates can be calculated as:

$$\frac{\tilde{\hat{\theta}}_i - \theta_i}{\theta_i} \times 100\% \tag{9}$$

In this equation, $\tilde{\hat{\theta}}_i$ is the median of the sample of all estimated parameters of the 30 realizations and θ_i is the actual value setting in the simulation from Table 1. Also the behaviour of the mean squared error MSE can be plotted, in the same manner as for the parameters. The median of the MSE of all realizations is denoted by $M\tilde{S}E$. The same quartile ranges $Q_{2,3}$ and Q_{1-4} exist for the MSE.

3. Parameter Estimate Behaviour: Fisher Information

Information content in a signal can be quantified via a description of the sensitivity of a model's output y_i to changes in a parameter θ_j. This is given by the (partial) derivative $\frac{\partial y_i}{\partial \theta_j}$. For multiple-input-multiple-output models with $j \in \{1, 2, \ldots, N_p\}$ number of parameters gathered in a parameter vector $\vec{\theta}$, and $i \in \{1, 2, \ldots, N_o\}$ the number of system outputs, the information content is given by the $N_p \times N_p$ Fisher information matrix M, defined as [24]:

$$M = \sum_{k=1}^{N} S(k)^T R^{-1} S(k) \tag{10}$$

In Eq. (10), $k = 1, 2, \ldots, N$ indicates the discrete sample number of the data signal. R is a $N_o \times N_o$ diagonal matrix of which the elements introduce a scaling to the output sensitivities according to the measurement noise related to that output, i.e. the noise covariance. The output sensitivities themselves are captured in the $N_o \times N_p$ output sensitivity matrix $S(k)$, which is given by [24] as:

$$S(k) = \begin{bmatrix} \frac{\partial y_1(k)}{\partial \theta_1} & \frac{\partial y_1(k)}{\partial \theta_2} & \cdots & \frac{\partial y_1(k)}{\partial \theta_{N_p}} \\ \frac{\partial y_2(k)}{\partial \theta_1} & \frac{\partial y_2(k)}{\partial \theta_2} & \cdots & \frac{\partial y_2(k)}{\partial \theta_{N_p}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial y_{N_o}(k)}{\partial \theta_1} & \frac{\partial y_{N_o}(k)}{\partial \theta_2} & \cdots & \frac{\partial y_{N_o}(k)}{\partial \theta_{N_p}} \end{bmatrix} \tag{11}$$

The sensitivity matrix $S(k)$ can usually be calculated analytically, derived from the dynamic equations of the to-be-estimated model [24]. For linear systems with only a single output, the matrix $S(k)$ is equal to the kth row of the regression matrix of this system. The inverse of the Fisher information M is the dispersion matrix D. The diagonal entries of the matrix D are the theoretical lower limit for the estimated covariances of the parameters, i.e., the Cramér-Rao Lower Bound:

$$D = M^{-1} \leq \mathrm{Cov}[\vec{\theta}] \tag{12}$$

Hence, the Cramér-Rao Lower Bound for the parameter standard deviations σ_{θ_j} is obtained as the square-root of the diagonal elements of D, i.e.:

$$\sigma_{\theta_j} = \sqrt{D_{j,j}} \quad \text{with } j = 1, 2, \ldots, N_p \tag{13}$$

An advantage of the use of the Fisher Information when analyzing signals is that it is related to the *theoretical* lower limit of the parameter covariances. Therefore, it is unrelated to the optimizer used for estimating the parameters. It purely describes the added value of the signal to estimate a certain parameter.

More interesting than the total Fisher information in a signal is the derivative of the Fisher information in time. This can say something about the value of a certain part of a data signal for estimation of a certain parameter. For example, for a data slice n_{slice} between data point indices $k_{n_{slice_{begin}}}$ and $k_{n_{slice_{end}}}$, the Fisher information in one specific output y_i can be calculated for one specific parameter θ_j via:

$$\frac{\Delta M_{y_i \theta_j}}{\Delta n_{slice}} = \sum_{k=k_{n_{slice_{begin}}}}^{k_{n_{slice_{end}}}} S_{i_{y_i}, j_{\theta_j}}(k) \frac{1}{R_{i_{y_i}, i_{y_i}}} S_{i_{y_i}, j_{\theta_j}}(k) \quad \text{with } i_{y_i} \in \{1, 2, \ldots, N_o\} \text{ and } j_{\theta_j} \in \{1, 2, \ldots, N_p\} \tag{14}$$

In the usual case R is a diagonal matrix [24], making this a scalar problem. As the total Fisher information is a sum via Eq. (10), the Fisher information of one slice $\frac{\Delta M_{y_i \theta_j}}{\Delta n_{slice}}$ is in fact the derivative per slice. In this equation, i_{y_i} and j_{θ_j} are the row and column number, respectively, for output y_i and parameter θ_j in the matrix $S(k)$.

The Fisher information is parametric as it is dependent on the partial derivatives $\frac{\partial y_i}{\partial \theta_j}$ of the model. This means that it differs for different given model structures. The derivation of the analytical definition of the Fisher information for the model given by Eq. (3) is described below. This definition is used to calculate the information content of the C_L signal from flight path reconstruction.

Recall the C_L model as given by Eq. (3) and the corresponding parameter vector as given by Eq. (8):

$$C_L = C_{L_0} + C_{L_\alpha} \left(\frac{1 + \sqrt{X}}{2} \right)^2 \alpha + C_{L_{\alpha^2}} (\alpha - 6^\circ)_+^2 \quad \text{and} \quad \vec{\theta} = \begin{bmatrix} a_1 & \alpha^* & \tau_1 & \tau_2 & C_{L_0} & C_{L_\alpha} & C_{L_{\alpha^2}} \end{bmatrix}^T$$

In the considered parameter estimation problem, there is only a single output, the measurement y_{C_L}. The sensitivity matrix $S(k)$ then contains all derivatives of y_{C_L} relative to the parameters in $\vec{\theta}$. The resulting matrix $S(k)$ is the result of applying the chain rule via:

$$\begin{aligned} S(k) &= \begin{bmatrix} \frac{\partial y_{C_L}(k)}{\partial a_1} & \frac{\partial y_{C_L}(k)}{\partial \alpha^*} & \frac{\partial y_{C_L}(k)}{\partial \tau_1} & \frac{\partial y_{C_L}(k)}{\partial \tau_2} & \frac{\partial y_{C_L}(k)}{\partial C_{L_0}} & \frac{\partial y_{C_L}(k)}{\partial C_{L_\alpha}} & \frac{\partial y_{C_L}(k)}{\partial C_{L_{\alpha^2}}} \end{bmatrix} = \\ &= \begin{bmatrix} \frac{\partial y_{C_L}(k)}{\partial X} \frac{\partial X(k)}{\partial a_1} & \frac{\partial y_{C_L}(k)}{\partial X} \frac{\partial X(k)}{\partial \alpha^*} & \frac{\partial y_{C_L}(k)}{\partial X} \frac{\partial X(k)}{\partial \tau_1} & \frac{\partial y_{C_L}(k)}{\partial X} \frac{\partial X(k)}{\partial \tau_2} & \frac{\partial y_{C_L}(k)}{\partial C_{L_0}} & \frac{\partial y_{C_L}(k)}{\partial C_{L_\alpha}} & \frac{\partial y_{C_L}(k)}{\partial C_{L_{\alpha^2}}} \end{bmatrix} \end{aligned} \tag{15}$$

Finding the derivatives when $\theta_j \in \{C_{L_0}, C_{L_\alpha}, C_{L_{\alpha^2}}\}$ is straightforward. These are:

$$\frac{\partial y_{C_L}(k)}{\partial \theta_i} = \begin{cases} 1 & \text{when } \theta_j = C_{L_0} \\ \left(\frac{1 + \sqrt{X(k)}}{2} \right)^2 \alpha(k) & \text{when } \theta_j = C_{L_\alpha} \\ \max(0, \alpha(k) - 6^\circ)^2 & \text{when } \theta_j = C_{L_{\alpha^2}} \end{cases} \tag{16}$$

The value of $X(k)$ over the entire interval $k = 1, 2, \ldots, N$ can be found by numerically integrating Eq. (2). The derivatives when $\theta_j \in \{a_1, \alpha^*, \tau_1, \tau_2\}$ are more difficult to find, but can be obtained via the procedure as explained for the nonlinear cost function derivatives associated with X in [19]. These are:

$$\frac{\partial y_{C_L}(k)}{\partial X} = \frac{1}{4} C_{L_\alpha} \alpha(k) \left(1 + \frac{1}{\sqrt{X(k)}}\right) \tag{17}$$

$$\frac{\partial G(k)}{\partial X} = -\frac{1}{\tau_1} \tag{18}$$

$$\frac{\partial G(k)}{\partial \theta_j} = \begin{cases} -\frac{\frac{1}{2}\{1-\tanh^2[a_1(\alpha(k)-\tau_2\dot{\alpha}(k)-\alpha^*)]\}\{\alpha(k)-\tau_2\dot{\alpha}(k)-\alpha^*\}}{\tau_1} & \text{when } \theta_j = a_1 \\ -\frac{\frac{1}{2}\{1-\tanh^2[a_1(\alpha(k)-\tau_2\dot{\alpha}(k)-\alpha^*)]\}\{-a_1\}}{\tau_1} & \text{when } \theta_j = \alpha^* \\ -\frac{\frac{1}{2}\{1-\tanh[a_1(\alpha(k)-\tau_2\dot{\alpha}(k)-\alpha^*)]\}-X(k)}{\tau_1^2} & \text{when } \theta_j = \tau_1 \\ -\frac{\frac{1}{2}\{1-\tanh^2[a_1(\alpha(k)-\tau_2\dot{\alpha}(k)-\alpha^*)]\}\{-a_1\dot{\alpha}(k)\}}{\tau_1} & \text{when } \theta_j = \tau_2 \end{cases} \tag{19}$$

The result for $\frac{\partial X(k)}{\partial \theta_j} = S$ (do not confuse S with $S(k)$!) can be found by numerically integrating the ordinary differential equation $\frac{dS}{dt} = \frac{\partial G(k)}{\partial X} S + \frac{\partial G(k)}{\partial \theta_j}$ over the interval $k = 1, 2, \ldots, N$. Note that in this case it is thus not only important that the model structure is known, but also that an *a priori* estimate of the parameters in $\vec{\theta}$ is needed as these are explicitly present in the derivative equations.

Finally, it is needed to find the Fisher information present in every 1 s slice of simulation data. In the current research, matrix R is set to 1 as there is only a single output. Also, there is only one output y_{C_L}, thus $i_{y_i} = 1$. For the seven remaining parameters, for every slice, Eq. (14) simplifies to:

$$\frac{\Delta M_{\theta_j}}{\Delta n_{slice}} = \sum_{k=k_{n_{slice_{begin}}}}^{k_{n_{slice_{end}}}} S_{1,j_{\theta_j}}(k)^2 \quad \text{with } j_{\theta_j} \in \{1, 2, \ldots, 7\} \tag{20}$$

This can then be compared directly to the parameter estimates' median $\tilde{\hat{\theta}}$ and distribution shape $Q_{2,3}$ and Q_{1-4}, in order to find the optimal window for each parameter to use in the slice-based parameter estimation.

G. Slice-based Parameter Estimation

The final goal of this paper is to prove that using different time windows of stall flight test data for estimation of a certain parameter can increase model accuracy. This concept is applied in the slice-based parameter estimation step.

The input to the slice-based parameter estimation is an optimal time window, found by analyzing the parameter estimate behaviour in the simulation data as explained above. The start and end of this time window are identified by the optimal pre-stall and post-stall slice number $n_{slice_{opt}}$, for every individual parameter, for every of the three stall input types. This $n_{slice_{opt}}$ is found heuristically and is the point in time where the median of the parameter optima of the 30 realizations is equal to the actual parameter value used in the simulation. It is checked if this time window also coincides with increased Fisher information in the data. How this works in practice is explained in section III.

The time window found is used in the actual parameter estimation with the real flight data. The realizations of the flight test data are randomly divided into fifteen training realizations and seven validation realizations. This is from a ratio of roughly 2-to-1 within each stall input type (i.e. 2-to-1 for no input, 9-to-4 for 3-2-1-1, 4-to-2 for wiggle). First, the nonlinear estimation is performed on the optimal window for the first parameter, a_1. This value is then saved. Then, the optimal window for α^* is chosen and the estimation is performed again. During this, the parameter a_1 can be varied again to give full freedom to the nonlinear optimization algorithm. This is then also performed for τ_1, τ_2 and the stability and control derivatives. At this point, X is calculated for the full data such that it can be used as a regressor in the linear estimation, in which the final values for C_{L_0}, C_{L_α} and $C_{L_{\alpha^2}}$ are calculated. Aside from using the optimal windows, the optimization routine is the same as explained for the parameter estimate behaviour analysis in subsubsection II.F.1.

To compare the increase in accuracy of the slice-based modeling method to the normal modeling method, both procedures as depicted in Figure 1 and Figure 2 are applied to the flight test training data. This results in two different final parameter estimate vectors $\hat{\vec{\theta}}$. For both, the model outputs $\hat{y}_{C_L}$ are calculated based on the reconstructed states from the validation flight test data and these are compared to the measured y_{C_L} by means of the MSE and $RRMS$ metrics.

III. Results

A. Parameter Estimate Behaviour Analysis

The parameter estimate behaviour analysis is performed using the three introduced metrics. The first are the medians $\tilde{\theta}$ and $\tilde{MSE}$ of the parameter optima and MSE, respectively. Second is the associated distribution of these optima and the MSE, which can be described by visualizing the inter-quartile range $Q_{2,3}$ (quartiles 2 and 3) and the full range of optima excluding outliers Q_{1-4} (all quartiles 1 to 4). Last is the derivative of the Fisher information per slice $\frac{\Delta M}{\Delta n_{slice}}$. These metrics are visualized both in a comparative and an individual manner. These types of graphs were constructed for all stall input types, slicing types and different parameters, resulting in a total of 138 figures. For conciseness, only a selected number of these graphs are discussed in this paper. They represent parameter estimate behavioural characteristics that are consistent across all different obtained results.

Important to note is that figures showing α, X and $\frac{\Delta M}{\Delta n_{slice}}$ for the simulated data are those of realization 1. However, after the flight path reconstruction step, the time traces of the different simulation realizations only show very minor differences. For the real flight test data, α for all different realizations is shown and for X and $\frac{\Delta M}{\Delta n_{slice}}$ the average of all realizations is depicted. Note that for flight test data this is purely indicative as the actual X and $\frac{\Delta M}{\Delta n_{slice}}$ are not known, due to the actual parameter values being unknown; they are calculated with the *a priori* values listed in Table 1.

For slicing type 1, the steady, straight, symmetric flight part before and after the stall was up to 120 s long. No significant differences were observed when this first and last 60 s was included or not. To decrease computational time for slicing type 2 and 3, the first and last 60 s of data were omitted from the analysis. This means that for these slicing types the graphs are shorter than for slicing type 1.

1. Behaviour of the Mean Squared Error and Interaction between Parameter Estimates

In Figure 6 the behaviour of the MSE is shown for flight test data with a wiggle input. Throughout the stall, the MSE increases and thereafter it decreases. This behaviour exists in all stall types, both in simulation data and in flight test data. The only difference between the simulated data and the flight test data is that in the simulated data the distribution of MSEs is very narrow, i.e., hardly any spread.

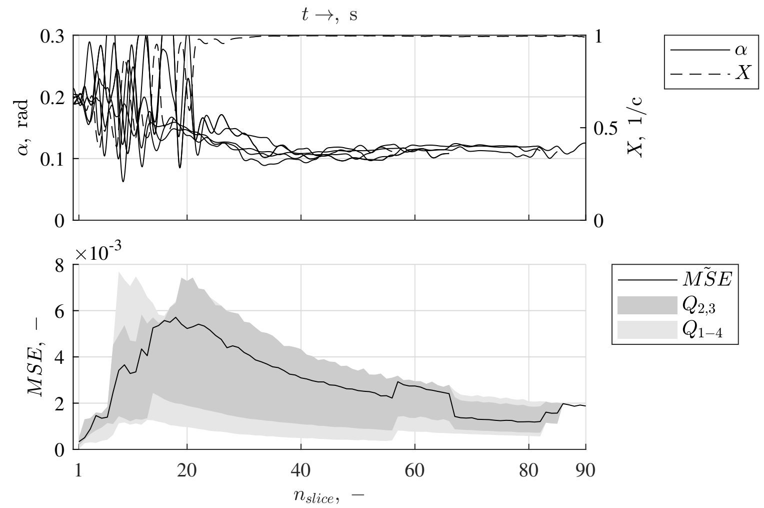


Fig. 6 Behaviour of the MSE. Flight test data, wiggle input, post-stall. The upper plot shows the angle of attack α and average flow separation point X. In the lower plot, $\tilde{MSE}$ is the median of all six realizations, $Q_{2,3}$ is the inter-quartile range, Q_{1-4} is the full range excluding outliers. As the plots show the post-stall, the time runs same to the slicing numbers, i.e. from left to right.

The normalized biases of the parameter estimates in Figure 7 are a clear example of the suspected cause of the increase and then decrease in MSE with increasing n_{slice}. The interaction between the X-parameter estimates (i.e., a_1, α^*, τ_1, and τ_2) and the normal stability and control derivatives can be observed. This interaction is present in all different stall types. Changes in the X-parameter estimates are present only immediately before and after the stall. All

X-parameters have converged to their final estimate before $n_{slice} = 10$. This is different for the stability and control derivatives. They differ significantly from their actual values right before and after the stall, with the estimates of C_{L_0} and $C_{L_{\alpha^2}}$ even being more than 100% lower than their actual values. Erratic changes in all stability and control derivative estimates can be observed as changes occur in the X-parameters close to the stall. Only after $n_{slice} = 10$ do the stability and control derivatives start to move steadily to their final values, which are all within 50% of the actual values.

The described behaviour of the parameter estimates can explain the changes in MSE in Figure 6. In the stall, unsteady flight conditions exist and the X parameter estimates are adjusted by the optimization routine to keep the rise in MSE as small as possible. The changes in X parameter estimates are accommodated by changes in the stability and control derivatives, evident by the large differences C_{L_0} and $C_{L_{\alpha^2}}$ attain from their actual value before $n_{slice} = 10$. After the stall this effect occurs too, but with the parameters switched. Here, changes in the X-parameters accommodate variations in the stability and control parameter estimates. As the stability and control derivatives get close to their actual values, the model becomes better at describing the steady flight conditions that become a larger part of each partition. This causes the decrease in MSE after the stall. An adverse effect of this is that the model becomes worse at modeling the aircraft in the stall. This is evident by the large differences of the estimates of τ_1 and τ_2 to their actual value in Figure 7 after $n_{slice} = 5$. These differences become more than 200%.

It is for this reason that many of the X parameter optimal slice numbers in Table 3 contain no more than 5 s of data before or after the stall. On the contrary, for the stability and control derivatives it is observed that the more data is used the better their estimates become.

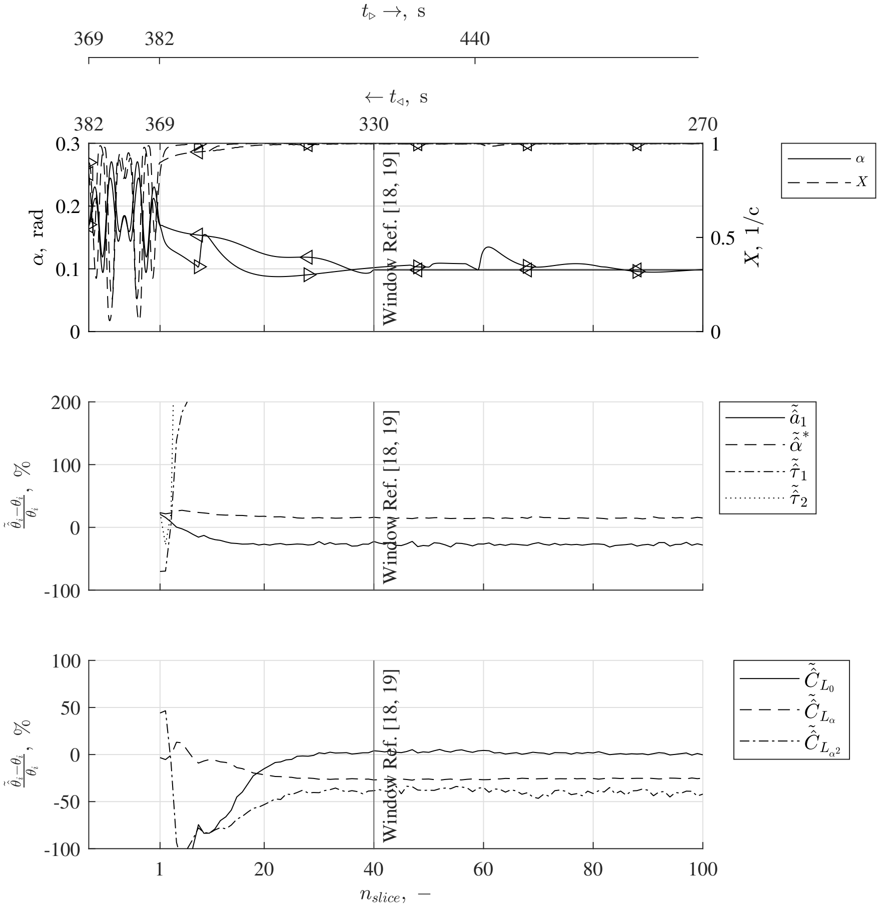


Fig. 7 Compared behaviour of all parameters. Simulation data, wiggle input, both pre-stall and post-stall (slicing type 3). The upper plot shows the angle of attack α and the flow separation point X. In the two lower plots, each line is the normalized bias $\frac{\tilde{\hat{\theta}}_i-\theta_i}{\theta_i}$ for each parameter. The left triangle ◂ indicates the pre-stall data, where times runs from right to left and the right triangle ▸ indicates the post-stall data where times runs from left to right. Each slice n_{slice} contains 1 s of the pre-stall and 1 s of the post-stall data.

2. *Behaviour of α^* in Different Stall Input Types*

The behaviour of a parameter estimate is strongly influenced by the type of input given during the stall. In Figure 8 an example of this is shown, in this case for the parameter α^*. The subfigures show the parameter estimate behaviour for simulation data with a no-input stall and a stall with 3-2-1-1 inputs.

Two main observations can be made from Figure 8. First, the 3-2-1-1 inputs induce a constant error in the estimate of the α^* parameter, where this error is near zero for the no-input stall. Second, the Fisher information related to α^* is higher for the no-input stall. These observations are related: the 3-2-1-1 input causes a very sudden stall entry and subsequent periodic motion in the angle of attack, resulting in four separate peaks in the Fisher information. The no-input stall is more gradual. Because of this, the total Fisher information is higher for the gradual stall, i.e., the area under the shown Fisher derivative is larger. This means that a no-input stall generates more information for the α^* parameter, which can in part explain the smaller error.

Why the Fisher information is higher for the no-input stall can physically be explained as follows. For this stall type, the angle of attack is slowly increased through the point where $X = 0.5$. This is also the point where the derivative of the Fisher information is highest, as the parameter α^* dictates this point and can hence be estimated more accurately. For a stall where inputs are given resulting in a rapidly changing angle of attack, such as the 3-2-1-1 or wiggle, the angle of attack where $X = 0.5$ cannot be estimated accurately due to significant transient effects being present. Also, the amount of time in which the separation point is in the vicinity of 0.5 is limited. For this reason, it was chosen that only the no-input stall is used for estimation of the α^* parameter, as indicated in Table 3.

3. *Behaviour of τ_2 in Different Stall Input Types*

Next to the α^* parameter, the τ_2 parameter is also affected by the different considered input types. In Figure 9, the parameter estimate behaviour of τ_2 is shown for the simulation data of both the no-input stall and the stall with wiggle inputs.

For the no-input stall, a strong tendency of the τ_2 estimate exists to go to the lower bound and only move away from the bound after $n_{slice} = 80$. This tendency is also observed for the stall with the wiggle input, however, the τ_2 estimate does not remain at the lower bound as long as for the no-input stall and attains its final value of roughly 0.3 s between $n_{slice} = 15$ and $n_{slice} = 25$. This 'lower-bound tendency' is also present in the flight test data in varying severity, but in some cases also is not present at all. This means that the parameter estimate behaviour for the simulation data shows resemblance with the flight test data, although sometimes they can also differ.

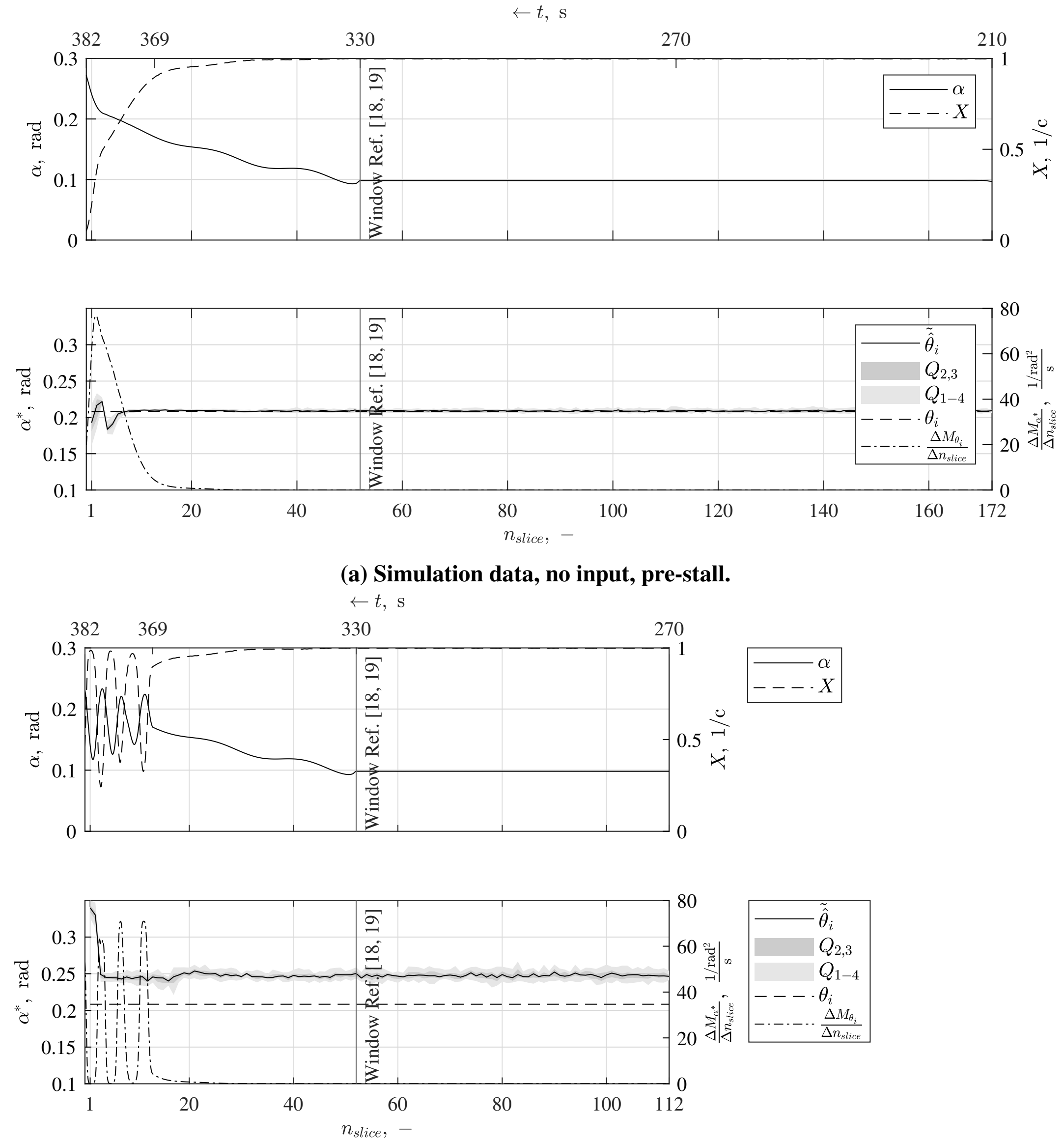
Analogous to the case of parameter α^*, the Fisher information related to the parameter τ_2 can also explain the behaviour of the estimate of τ_2. For the no-input case, the Fisher information is zero everywhere, except for the stall recovery. This is also when the lower-bound tendency begins. All the information that is available for estimation of parameter τ_2 is thus concentrated in only a narrow window of roughly $12 \leq n_{slice} \leq 14$. For the wiggle inputs, the Fisher information contains multiple peaks distributed over the duration of the entire stall, summing to a higher total information content in the signal. The τ_2 parameter describes the hysteresis effect, which only occurs if large and quick changes in the angle of attack, and thus movement of the flow separation point, are present. In the no-input stall this is only the case during stall recovery, but the wiggle inputs induce more flow separations and re-attachments throughout the stall.

Nevertheless, in both cases it is clear that the τ_2 parameter estimate benefits from data that is only very close to the stall. Otherwise, it may be estimated to the lower bound as in the no-input case or it may be wrongly estimated as for the wiggle inputs after $n_{slice} = 15$. This confirms the importance of the research in this paper: simply slicing in the manner of [18, 19] would in both cases result in different, biased parameter estimates than the actual parameter value. This is indicated by the vertical line in the graph, which is approximately the cutoff point for data in the research of [18, 19].

4. *Behaviour of τ_1 in Simulation Data versus Flight Test Data*

In the slice-based modeling method, parameter estimate behaviour that results from simulation data is used as a substitute for the expected, but unknown, parameter estimate behaviour resulting from flight test data. Therefore, the similarity of the parameter behaviour resulting from both data types must be compared. In Figure 10, the behaviour of the estimate τ_1 is shown, following from a stall with 3-2-1-1 inputs in both simulation data and flight test data.

From the simulation data it becomes clear that the τ_1 estimate also shows the lower-bound tendency that was observed for τ_2, although to a lesser extent. As soon as the Fisher information derivative shows has the first peak, the estimate moves to the lower bound. The lower-bound tendency in this case is interesting, as the actual parameter value of 0.2547 s is not near the lower bound. This may be the reason that the estimate of τ_1 moves away from this bound



(a) Simulation data, no input, pre-stall.

(b) Simulation data, 3-2-1-1 input, pre-stall.

Fig. 8 Individual behaviour of parameter α^*. The upper plot of each subfigure shows the angle of attack α and flow separation point X. In the lower plots, $\tilde{\hat{\theta}}_i$ is the median of the parameter optima of all 30 realizations, $Q_{2,3}$ is the inter-quartile range, Q_{1-4} is the full range excluding outliers. θ_i is the actual parameter value in the simulation and $\frac{\Delta M_{\theta_i}}{\Delta n_{slice}}$ is the Fisher information derivative related to the parameter. As the plots show the pre-stall, the time runs opposite to the slicing numbers, i.e. from right to left.

very suddenly around $n_{slice} = 18$, contrary to τ_2 in Figure 9b, where this movement is more gradual; i.e., in the range $15 \leq n_{slice} \leq 25$.

In the flight test data, the lower-bound tendency can also be observed. Until $n_{slice} = 8$, the estimate for τ_1 moves to the lower bound, before very suddenly moving to the upper bound after this. Thus, it seems that analyzing the parameter estimate behaviour of simulation data is indeed representative of the parameter estimate behaviour that follows from the flight test data. However, for the test flight data, the distribution of the optima is significantly broader. The same is generally observed for all input and slicing types. Thus, parameter estimate behaviour is less predictable and consistent for test flight data than for simulation data, even though significant trends in the behaviour in flight test data can generally be predicted by use of simulation data.

From the parameter estimate behaviour in the simulation data, another observation can be made. A moment in time where the distribution of optima is narrow, i.e. where the optima seem to be more certain, does not necessarily represent a timespan where that estimate is correct. In Figure 10a, the reliability is highest around $n_{slice} = 35$, but this does not deliver a correct estimate. However, in other cases this similarity does seem present, for example in Figure 9b in the range $8 \leq n_{slice} \leq 15$. The same contrary evidence can be found for multiple input and slicing types and also other parameters than τ_1 and τ_2.

Finally, Figure 10a again shows the important finding also observed earlier for τ_2: making use of the full available data, as in the research of [18, 19], may result in biased estimates for certain parameters, in this case for τ_1. Using the window of [18, 19] would result in an estimate value of roughly 0.8 s, instead of the correct 0.2547 s.

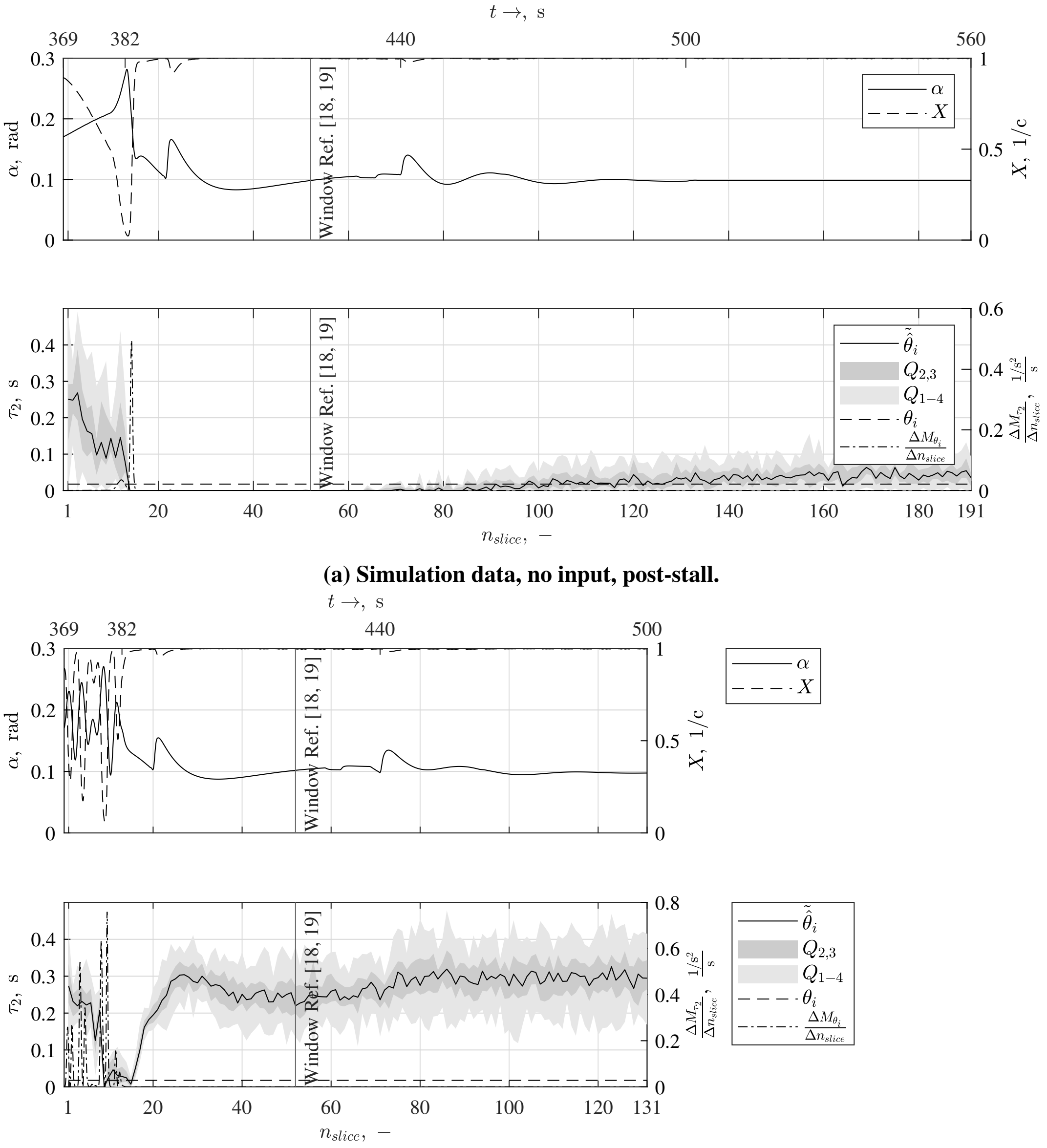
B. Slice-based Estimation Procedure

The slice-based estimation procedure is based on performing the parameter estimation on an optimal time window for each parameter. In the parameter estimate behaviour analysis, the pre-stall and post-stall slice number $n_{slice_{opt}}$ for each optimal time window is chosen manually. This is done by comparing the parameter estimates with the actual parameter value in the simulation. The optimal slice number is chosen as the point where the parameter estimate and the actual parameter value are equal, also taking into account any behavioural features that have been observed as discussed in the foregoing subsection III.A. The selected optimal slice numbers $n_{slice_{opt}}$ are presented in Table 3. An entry with 'n/a' means not applicable, i.e., data from that stall type is not used for estimation of that parameter. For example, the parameter estimate of α^* is only based on training data from the flight test data where no additional inputs were given by the pilots. When an 'all' is indicated for a parameter, it means that for that stall type all available data should be used in the estimation.

Important to note is that the values in Table 3 are the numbers n_{slice} associated with slicing types 1 and 2. However, to select the window of the flight test data used in the slice-based parameter estimation, it is easier to convert the slice numbers to time in seconds before stall entry and after stall recovery, respectively. For example, for the a_1 parameter the no input, pre-stall optimal slice number $n_{slice_{opt}} = 30$. This means that the partition for the training flight test data is begun at $30 - 13 = 17$ s before the stall, as the stall in the simulation data is 13 s long. This time is indicated in parentheses in Table 3.

Table 3 Optimal slicing numbers $n_{slice_{opt}}$ as found for the simulated stall runs. The depicted slice number for the pre-stall is the slice number corresponding to slicing type 1 and the slice number for the post-stall is the slice number corresponding to slicing type 2. The corresponding time before stall entry (in pre-stall columns) and after stall recovery (in post-stall columns) used to slice the training flight test data is indicated in parentheses.

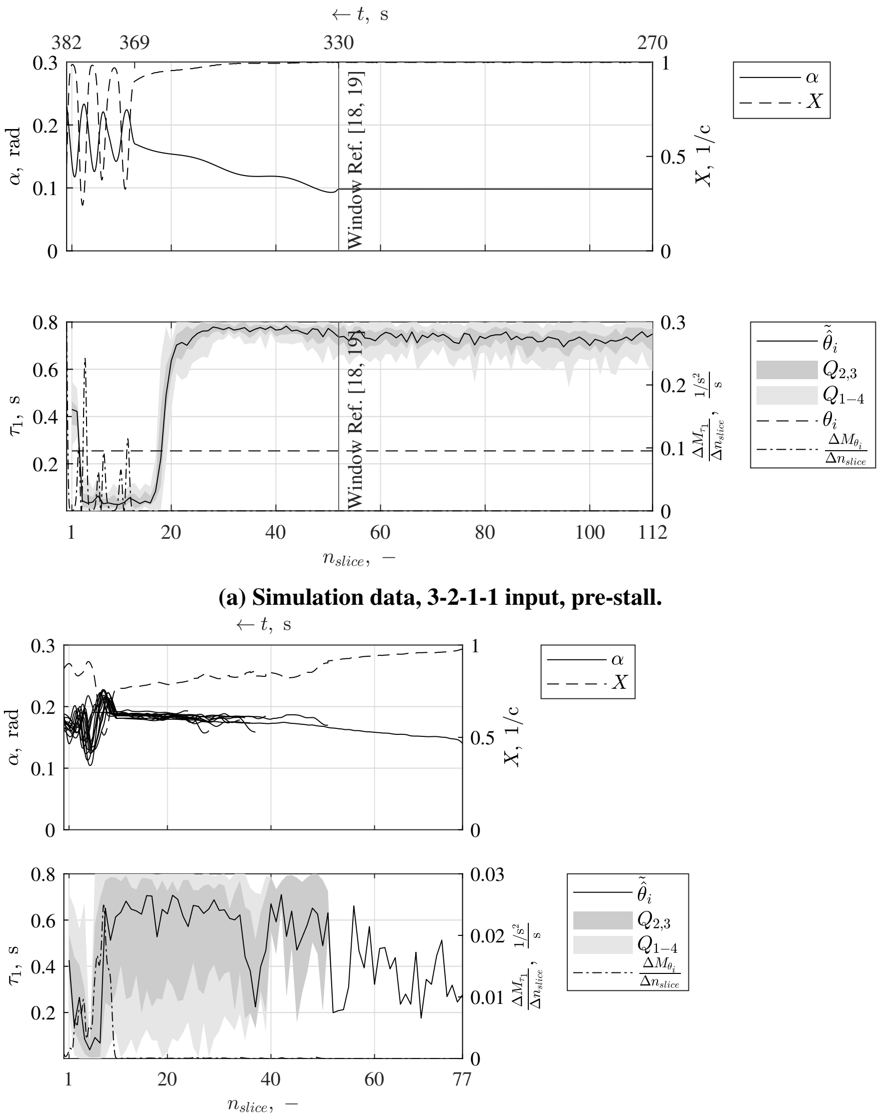
	$n_{slice_{opt}}, -$					
	No input		3-2-1-1		Wiggle	
θ_i	Pre	Post	Pre	Post	Pre	Post
a_1, −	30 (17 s)	all	13 (0 s)	70 (57 s)	19 (6 s)	20 (7 s)
α^*, rad	all	all	n/a	n/a	n/a	n/a
τ_1, s	30 (17 s)	17 (4 s)	18 (5 s)	13 (0 s)	16 (3 s)	16 (3 s)
τ_2, s	30 (17 s)	14 (1 s)	15 (2 s)	14 (1 s)	18 (5 s)	14 (1 s)
$\in \{C_{L_0}, C_{L_\alpha}, C_{L_{\alpha^2}}\}$, −	all	all	all	all	all	all



(a) Simulation data, no input, post-stall.

(b) Simulation data, wiggle input, post-stall.

Fig. 9 Individual behaviour of parameter τ_2. The upper plot of each subfigure shows the angle of attack α and flow separation point X. In the lower plots, $\tilde{\hat{\theta}}_i$ is the median of the parameter optima of all 30 realizations, $Q_{2,3}$ is the inter-quartile range, Q_{1-4} is the full range excluding outliers. θ_i is the actual parameter value in the simulation and $\frac{\Delta M_{\theta_i}}{\Delta n_{slice}}$ is the Fisher information derivative related to the parameter. As the plots show the post-stall, the time runs same to the slicing numbers, i.e. from left to right.



(a) Simulation data, 3-2-1-1 input, pre-stall.

(b) Flight test data, 3-2-1-1 input, pre-stall.

Fig. 10 Individual behaviour of parameter τ_1. The upper plot of each subfigure shows the angle of attack α and flow separation point X (X is the average of all realizations for flight test data). In the lower plots, $\tilde{\hat{\theta}}_i$ is the median of the parameter optima of all realizations, $Q_{2,3}$ is the inter-quartile range, Q_{1-4} is the full range excluding outliers. θ_i is the actual parameter value in the simulation and $\frac{\Delta M_{\theta_i}}{\Delta n_{slice}}$ is the Fisher information derivative related to the parameter ($\frac{\Delta M_{\theta_i}}{\Delta n_{slice}}$ is the average of all realizations for flight test data). As the plots shows the pre-stall, the time runs opposite to the slicing numbers, i.e. from right to left.

C. Accuracy of the Slice-based Modeling Method

To evaluate the increase in accuracy of the slice-based modeling method as depicted in Figure 2 it is compared to the normal modeling method as depicted in Figure 1 (excluding model structure selection). Both methods were applied to the same flight test data set for direct comparison. The normal modeling method is applied to the training flight test data by simply performing the full estimation on the entire realization. The slice-based modeling method uses the selected optimal time window from Table 3 for each parameter. The resulting parameter estimates for both methods are given in Table 4.

Table 4 Parameter estimate comparison of the normal modeling method and slice-based modeling method applied to flight test data.

	Normal	Slice-based
θ_i	modeling method	modeling method
a_1, –	31.8630	34.1856
α^*, rad	0.2289	0.2202
τ_1, s	0.3541	0.4595
τ_2, s	0.1297	0.2182
C_{L_0}, –	0.1944	0.2065
C_{L_α}, –	4.5172	4.4192
$C_{L_{\alpha^2}}$, –	5.3935	5.1026

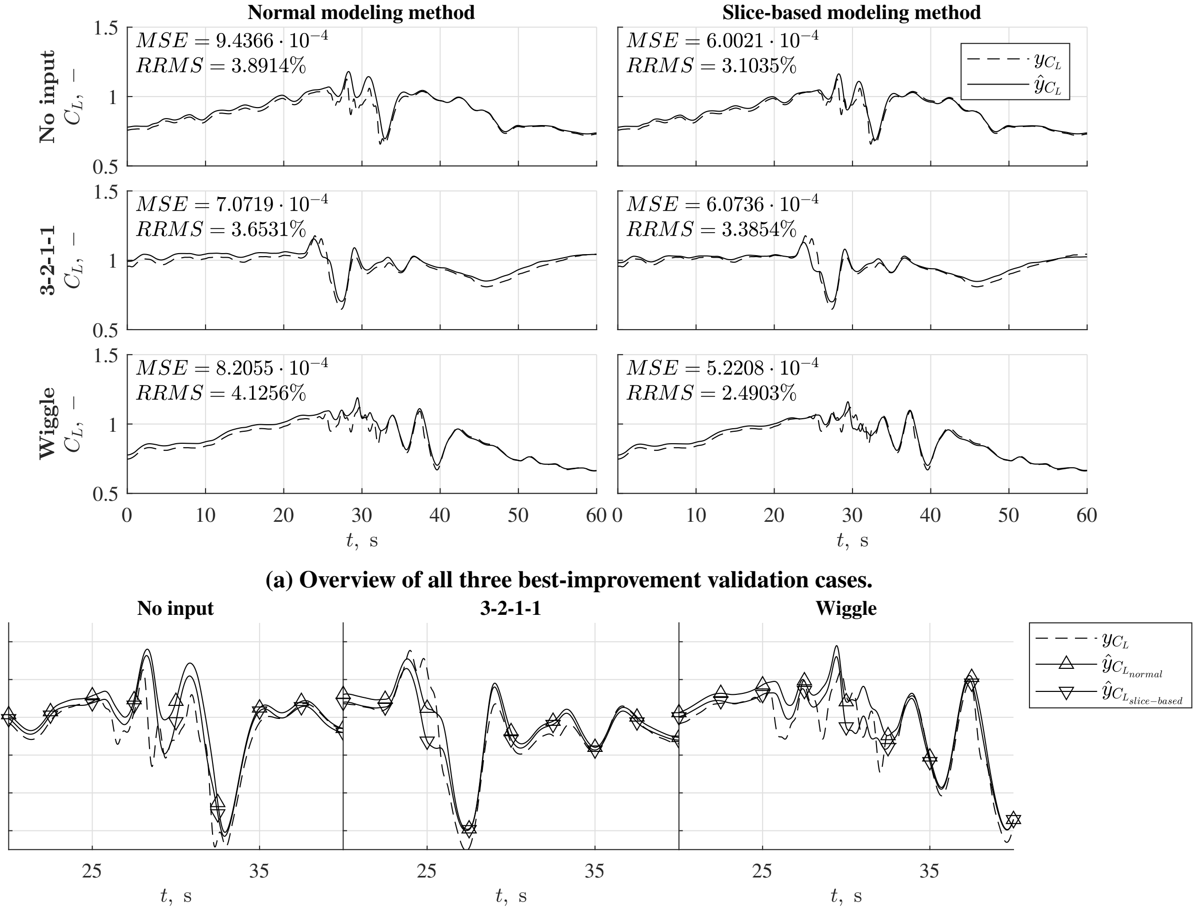
All parameters except τ_1 and τ_2 are roughly equal to each other, which are both significantly higher with the slice-based modeling method than with the normal modeling method. This is expected, as for the parameter estimate behaviour of τ_1 and τ_2 in subsection III.A it was found that specifically these are most inclined to change when using data outside of the stall region. This incline was also found for a_1, however, less severely, also resulting is a slightly different estimate in Table 4. For the other parameters, the parameters only differ slightly: α^* can in both cases be estimated accurately and for C_{L_0}, C_{L_α} and $C_{L_{\alpha^2}}$ the same amount of data is used.

To investigate the effect that these differences in parameter estimates have on model accuracy, the parameter estimates from both methods are applied in Eq. (3) to calculate output y_{C_L} for seven validation data sets. Table 5 shows the MSE and $RRMS$ values for both methods for the seven validation data sets. The MSE and $RRMS$ are in the same order of magnitude for both methods, but differences are present nonetheless. In four of the seven validation cases, the new method achieves better accuracy in the range of roughly $\Delta MSE = -10\%$ to $\Delta MSE = -35\%$. However, in three cases a worse accuracy is achieved. On average the new method shows an increase in performance with $\Delta MSE = -6.24\%$.

Table 5 MSE and RRMS values for the validation datasets.

		Normal modeling method		Slice-based modeling method		
Input	Realization	MSE, –	$RRMS$, %	MSE, –	$RRMS$, %	ΔMSE, %
No input	3	$9.4366 \cdot 10^{-4}$	3.8914	$6.0021 \cdot 10^{-4}$	3.1035	−36.4
3-2-1-1	1	$1.4391 \cdot 10^{-4}$	1.9242	$1.9270 \cdot 10^{-4}$	2.2266	+33.9
3-2-1-1	2	$1.4938 \cdot 10^{-4}$	1.9654	$3.2383 \cdot 10^{-4}$	2.8937	+116
3-2-1-1	7	$3.1524 \cdot 10^{-4}$	3.1063	$2.8571 \cdot 10^{-4}$	2.9572	−9.37
3-2-1-1	10	$7.0719 \cdot 10^{-4}$	3.6531	$6.0736 \cdot 10^{-4}$	3.3854	−14.1
Wiggle	1	$8.2055 \cdot 10^{-4}$	4.1256	$5.2208 \cdot 10^{-4}$	2.4903	−36.4
Wiggle	6	$2.2765 \cdot 10^{-3}$	5.3092	$2.4903 \cdot 10^{-3}$	5.5528	+9.39
Mean		$7.6522 \cdot 10^{-4}$	3.4250	$7.1746 \cdot 10^{-4}$	3.2299	−6.24

The outputs of the models estimated with the normal modeling method and the slice-based modeling method can also be compared visually. In Figure 11a, for each stall input type the validation case with the largest improvement for the slice-based method relative to the normal method is shown. Figure 11a shows both the model outputs $\hat{y}_{C_L}$ and the validation measurement y_{C_L}. In the figure, two main improvements with the slice-based modeling method relative to the normal modeling method can be observed. One is visible before the stall and one is visible within the stall. Before the stall, the normal modeling method shows a constant error in C_L of roughly +0.04. In the new modeling method this error is significantly smaller, between 0 and +0.025. The second improvement exists in the stall, where the slice-based method is better at modeling the nonlinear effects. The normal method shows less sudden changes in C_L during the stall, which are more pronounced with the slice-based method. This is visible in Figure 11b, showing both model outputs $\hat{y}_{C_{L_{normal}}}$ and $\hat{y}_{C_{L_{slice-based}}}$ and the validation measurement y_{C_L} from Figure 11a, but focused on the stall. The above two findings show that the slice-based modeling method can improve model fit not only in the stall, but also outside of the stall.



(a) Overview of all three best-improvement validation cases.

(b) Comparison within the stall.

Fig. 11 Comparison of the normal modeling method and the slice-based modeling method, showing the validation stall runs where the best improvement is achieved (no input realization 3, 3-2-1-1 realization 10, wiggle realization 1). The model outputs are denoted $\hat{y}_{C_L}$ and the validation data measurement is denoted y_{C_L}.

IV. Discussion

A. Interpretation of the Results

This paper has introduced a new slice-based modeling method. It is shown that this new method can improve stall model accuracy through optimal data slicing by analyzing Kirchhoff stall parameter estimate behaviour. A simulation data set representative of available flight test data was created and then sliced and partitioned. By applying a nonlinear and linear estimation to each partition, the parameter estimate behaviour of the Kirchhoff stall model parameters and stability and control derivatives could be analyzed quantitatively. This enabled the selection of an optimal slice number and corresponding optimal window size of data on which to perform the final slice-based parameter estimation. The obtained results have a number of important implications, which are discussed below.

First, applying the new slice-based modeling method results in a stall model that has superior accuracy compared to a model obtained using a 'normal' estimation method. A decrease in MSE of around 10% to 35% can be achieved for the largest share of the validation datasets, as summarized in Table 5. This is significant, as the slice-based estimation method uses the exact same data as the normal estimation method. This answers the primary question of the research discussed by this paper: model accuracy can indeed be improved by choosing specific parts of data to which to apply the estimation of a specific parameter.

The application of this new modeling method is made possible by the novel analysis introduced in this paper, i.e., by explicitly considering parameter estimate behaviour through data slicing and partitioning. Through this method it is for the first time possible to identify directly the effect of certain windows of data on the estimate of a parameter. Formerly, only the Fisher information could be used for this purpose. From the results it becomes clear that the Fisher information is especially useful to quickly identify which type of input given during the stall may result in more useful data for estimation of a specific parameter. An example of this was discussed for the α^* parameter, based on findings in Figure 8. Here, this parameter was estimated with no error relative to the actual parameter value for the simulation that contained the no-input stall. This type of stall contained a higher Fisher information than the stall with the 3-2-1-1 inputs. The latter resulted in an erroneous estimate for α^*. However, the results show that there is not always a direct quantitative link between the Fisher information and the correctness of a certain parameter. The Fisher information can thus not be used as a sole indicator to find the optimal slice number $n_{slice_{opt}}$. A high Fisher information derivative often coincides with large changes in the parameter estimate, as for the movement toward the lower bound before $n_{slice} = 13$ in Figure 9. However, it does not explain all movements, as for example in Figure 10a for the sudden movement at $n_{slice} = 18$. Therefore, it is suggested to always use the Fisher information in combination with the introduced method of parameter estimate behaviour analysis.

Using these two concepts in conjunction provides a reliable basis to manually select optimal time windows for each parameter, as summarized in Table 3. Mainly, it becomes clear that better model accuracy is achieved when only data close to the stall is used for estimation of the stall parameters, and as much data as is available is used for the stability and control derivatives. Here, 'close to the stall' generally means a window that roughly starts no more than 5 s before stall entry and ends no more than 5 s after stall recovery.

Depending on the type of stall, differences to this general rule can exist. For example, for a_1 the optimal time before and after the stall may vary from 0 s of data before the stall to using all available data after stall recovery. For estimation of α^*, it is recommended to only use data that contains very gradual stall entries without any control inputs and use all data available within these data sets. This is substantiated by analysis of the Fisher information related to α^* in these stall types. The estimates of parameters τ_1 and τ_2 are the most sensitive to the used data window. Sudden changes in parameter estimates occur for slice numbers marginally before or after the stall, which may decrease their power in modeling the behaviour of the aircraft in the stall. The optimization algorithm primarily changes the estimates to lower the MSE in the regions before and after the stall. This is expected, as τ_1 and τ_2 describe nonlinear effects that only take place when dynamic excitations exist around the stall angle of attack. Therefore, the recommended time to include before or after the stall for estimation of these parameters is never more than 17 s and usually even less than 5 s. Next to this, it is found that no-input stalls are not effective for estimating τ_1 and τ_2. These stall types were used in the slice-based parameter estimation in this paper, but it may be an option to not use these stall types at all. Lastly, this paper recommends to use as much data as available for the estimation of the stability and control derivatives C_{L_0}, C_{L_α} and $C_{L_{\alpha^2}}$.

An important feature of the slice-based modeling method is that simulation data is used as a substitute for real flight data. It is assumed that the parameter estimate behaviour for the simulation data is predictive of the parameter estimate behaviour for flight test data. As a means of validation, the parameter estimate behaviour for both types of data were compared. From this, it becomes evident that simulation data does in part represent flight test data, although not fully. This is because for flight test data large differences exist between different realizations of the same stall type. The realizations of the simulation data differ only through an artificially added white noise signal to the simulation output before applying flight path reconstruction and hence provide a much more consistent dataset. Some inputs or conditions that are present in the flight test data are simply too complex to recreate in a simulation. However, some distinct features in parameter estimate behaviour are consistent between both data types. These are, for example, the lower-bound tendencies of the τ_1 and τ_2 parameters, but also the very predictable behaviour of α^*. Both also contain the behaviour of the MSE to increase during the stall and decrease outside of it. Related to this, the general parameter estimate behaviour where in the stall the stability and control derivatives are changed by the optimization routine to accommodate for changes in the stall parameters and vice-versa outside of the stall is present in both simulation data as well as flight test data. Thus, despite some differences between simulation data and flight test data, the roughly comparable behaviour does provide a solid basis for the slice-based modeling method.

Examining the parameter estimate behaviour that results from test flight data can also shed new light on related research, as it may explain the reason why certain previous results were obtained. An example of this is the conclusion drawn in [19] regarding the τ_2 parameter. In [19] it is suggested that the estimate of τ_2 could simply be set to 0, because its estimate of 0.0176 s is very close to 0. Also, only a small difference in MSE compared to the validation data was found when the estimate of τ_2 was actually set to 0. However, the parameter estimate behaviour analysis in this paper may give an explanation for this. First, in this paper it was found that there is a significant interaction between the estimates of the stall parameters and the estimates of the stability and control derivatives. The small change in MSE reported in [19] may be explained by the fact that the stability and control derivatives estimates reduce part of the MSE by compensating for the lack of τ_2. A second explanation is that for the research in [19], only flight test data containing wiggle inputs was used. It was found in this paper that the τ_2 parameter often has a lower-bound tendency when wiggle inputs are applied. This was not found for the 3-2-1-1 inputs in the parameter estimate behaviour analysis. Also including the data with 3-2-1-1 inputs may explain why a significantly higher estimates for τ_2 than that in [19] of 0.0176 s are found, namely 0.1297 s for the normal estimation method and 0.2182 s for the slice-based estimation method. Because also a higher model accuracy is achieved, it is argued that setting τ_2 to 0 should not be done and for every new data set a parameter estimation behaviour analysis is warranted to make a decision on whether such an assumption is valid.

The parameter estimate behaviour analysis is the main contribution of the slice-based modeling method. However, this addition does add significant computational load relative to the normal estimation method. For the full analysis of data, for every slice, in every slicetype, for every realization of every stall input type from both simulation data and flight test data, a full nonlinear estimation of 500 initial conditions had to be performed. This results in a total of over 14.5 million optimizations and an accompanying calculation time of three weeks, for 24 hours per day*. Here, it should be noted that the optimizations are independent, and therefore the problem can be readily partitioned and solved in parallel on large scale parallel computing hardware. For example, in this paper use was made of MATLAB's parallel programming toolbox. In future analyses computational load may also be decreased by varying the resolution of n_{slice} to larger than 1 s at moments in time where less activity in parameter estimate behaviour is expected.

It is demonstrated in this paper that performing a parameter estimate behaviour analysis can deliver new fundamental insights into how parameter estimates change in reaction to specific parts of data that are available for estimation. It is shown that it is possible to increase the accuracy of nonlinear stall models without the need for gathering more flight data by introducing the slice-based modeling method. Additionally, the possibilities of the method extend beyond the realm of stall modeling for fixed wing aircraft. In fact, any system identification procedure that contains a parameter estimation based on measurement data can benefit from the slice-based modeling method. In every parameter estimation problem, certain parts of data are more valuable to some parameters than to other parameters. Tailoring the optimal time window used in the parameter estimation to each individual parameter may increase the accuracy of any model found through system identification.

B. Recommendations

Even though the results show the applicability of the parameter estimate behaviour analysis method and its usefulness to improve model performance with the slice-based modeling method, it leaves room for improvement. Several recommendations can be made.

The current slicing method only makes use of three types of slicing in time, i.e., the pre-stall phase, post-stall phase and both the pre-stall and post-stall phases. These types were defined such that every additional slice adds to the size of a partition. A problem with this is that the resulting parameter estimate behaviour from adding that slice is less pronounced because the partitions contain increasingly more foregoing data. A solution to this could be to create a moving window of some sort, that analyzes each part more specifically. This window size can then also be varied. Another recommendation comparable to this is that the slicing and partitioning can be based not only on time, but on independent variables such as α and $\dot{\alpha}$. However, this would introduce problems with the estimation of the X parameter, as well as all partitions having variable sizes. This would make comparison between the partitions more difficult, unless a solution is found to retain the same amount of data points in each partition.

Also, the current method only adds a white noise signal to the simulation output, causing very similar realizations after the flight path reconstruction. Hence, the simulated data shows less variability than the measured flight test datasets. This applies to the trimmed initial condition, the size and timing of the inputs given, but also to more integral features, such as the actual stall parameter setting in the simulation. For future research a method is recommended that may evaluate the sensitivity of the parameter behaviours to these settings. Also, this creates more differences between the realizations, which can resemble better the differences that exist between the realizations of the real flight data.

*1x PC with 6x Intel Core i7-8700 @3.20GHz (16GB RAM) and 2x PC with 4x Intel Xeon E5-1620 v3 @3.50GHz (16GB RAM)

Related to this is the inclusion of orthogonal function modeling (OFM). As a starting point for this research, the model as found in [19] was used. However, this eliminated the effect of OFM in the estimation procedure. For a full evaluation of the slicing selection method as introduced in this paper, it is recommended that an analysis is performed as to what changes occur in the chosen regressors of the OFM as the slicing type progresses in time. This may add certain knowledge on 'regressor behaviour analysis' to the parameter estimate behaviour analysis performed in this research.

The last recommendation applies to the new modeling method itself. The current method represents a prototype, as the optimal time windows were chosen manually by investigation of the parameter estimate behaviour plots. While this provides a strong argument for using data slicing and selection for stall parameter analysis in the first place, a more thorough optimization method, for example by changing the time windows of each parameter separately and assessing the model performance increase, is needed for large-scale practical application.

V. Conclusion

In this paper it is proven that the accuracy of aerodynamic stall models based on Kirchhoff's theory of flow separation can be improved by use of a novel slice-based modeling method. When applied to validation flight test data the new method can decrease the MSE for a fitted lift coefficient model in more than half of the validation cases by 10% to 35%. On average, a 6% improvement was achieved.

The slice-based modeling method is based on a 'parameter estimate behaviour' analysis. In this analysis, simulation data is generated that represents real flight test data that is available. This simulation data is sliced and combined into different partitions that include different time windows of this data. Applying a parameter estimation procedure to these partitions, shows which parts of the data cause a change in the estimate of a specific parameter. By identifying where the parameter estimates attain the same value as the actual parameter value in the simulation an optimal time window can be identified for every parameter, for every type of stall. This window is then applied in a final parameter estimation on flight test data, resulting in a model with increased accuracy relative to the normally used method. Clear similarities were found between the parameter estimate behaviour in simulation data and the estimate behaviour in flight test data that the simulation data is aimed to represent.

For every parameter in the Kirchhoff based stall model an individual optimal time window was found. For the stall parameters a_1, α^*, τ_1 and τ_2 this window generally starts no more than roughly 5 s before the stall and no more than 5 s after the stall, even though differences exist between the parameters depending on which control inputs are given during the stall. For the stability and control derivatives C_{L_0}, C_{L_α} and $C_{L_{\alpha^2}}$ in the model applies that all available data should be used in the estimation for better model accuracy.

Next to the parameter estimate behaviour analysis, the Fisher information was found to be a strong indicator of which region of data is beneficial to use in the estimation of a certain parameter. A connection exists between the Fisher information and the parameter estimate behaviour: in parts of the data where the Fisher information is high also changes of the parameter estimate can be observed. However, the Fisher information is not a direct indicator for finding optimal window size, it is very valuable in aiding to make a quick decision on whether a certain stall type or type of input in a flight test is beneficial to a certain parameter or not.

In conclusion, this paper shows that the slice-based modeling method can be used for improving stall models without the need for additional flight tests, but by making smarter use of data that is readily available. This not only makes it possible to improve existing models, but also creates the possibility to make future research flight test data even more efficient and valuable. Furthermore, the slice-based modeling method can be applied to any system identification problem that includes a parameter estimation. This means that the new method can also have a large impact outside the field of stall modeling. Overall, it is clear that the slice-based modeling method can enable researchers to create more representative flight simulation models, resulting in better pilot training, contributing to safer commercial air transport.

References

[1] IATA, "Loss of Control In-Flight Accident Analysis Report," , 2019.

[2] Lambregts, A., Nesemeier, G., Wilborn, J., and Newman, R., "Airplane Upsets: Old Problem, New Issues," *AIAA Modeling and Simulation Technologies Conference and Exhibit*, American Institute of Aeronautics and Astronautics Inc, 2008.

[3] ICAO, "ICAO Doc 10011: MANUAL ON AEROPLANE UPSET PREVENTION AND RECOVERY TRAINING," , 2014.

[4] EASA, "Certification Specifications for Aeroplane Flight Simulation Training Devices 'CS-FSTD(A)'," , 2018.

[5] Gingras, D., and Ralston, J., "Aerodynamics modelling for training on the edge of the flight envelope," *The Aeronautical Journal*, Vol. 116, No. 1175, 2012, pp. 67–86.

[6] Gingras, D. R., Ralston, J. N., Oltman, R., Wilkening, C., Watts, R., and Desrochers, P., "Flight Simulator Augmentation for Stall and Upset Training," *AIAA Modeling and Simulation Technologies Conference*, American Institute of Aeronautics and Astronautics Inc, 2014.

[7] Murphy, P. C., Klein, V., and Frink, N. T., "Nonlinear Unsteady Aerodynamic Modeling Using Wind-Tunnel and Computational Data," *Journal of Aircraft*, Vol. 54, No. 2, 2017, pp. 659–683.

[8] Sri-Jayantha, M., and Strengel, R. F., "Determination of Nonlinear Aerodynamic Coefficients Using the Estimation-Before-Modeling Method," *Journal of Aircraft*, Vol. 25, No. 9, 1988, pp. 796–804.

[9] Grant, P. R., Liu, S., Luo, Z., and Moszczynski, G., "Development of Post-stall Flight Models from Certification Flight Test and Wind-tunnel Data," *AIAA SciTech Forum*, American Institute of Aeronautics and Astronautics Inc, 2017.

[10] Goman, M., and Khrabov, A., "State-Space Representation of Aerodynamic Characteristics of an Aircraft at High Angles of Attack," *Journal of Aircraft*, Vol. 31, No. 5, 1994, pp. 1109–1115.

[11] Fischenberg, D., "Identification of an Unsteady Aerodynamic Stall Model from Flight Test Data," *20th Atmospheric Flight Mechanics Conference*, American Institute of Aeronautics and Astronautics Inc, 1995, pp. 138–146.

[12] Singh, J., and Jategaonkar, R., "Identification of Lateral-Directional Behavior in Stall from Flight Data," *Journal of Aircraft*, Vol. 33, No. 3, 1996, pp. 627–630.

[13] Fischenberg, D., and Jategaonkar, R., "Identification of Aircraft Stall Behavior from Flight Test Data," *RTO SCI Symposium in System Identification for Integrated Aircraft Development and Flight Testing*, North Atlantic Treaty Organization Research and Technology Organization, 1999, pp. 17.1–17.8.

[14] Kumar, R., Ghosh, A., and Misra, A., "Parameter Estimation from Flight Data of Hansa-3 Aircraft Using Quasi-Steady Stall Modeling," *Journal of Aerospace Engineering*, Vol. 26, No. 3, 2013, pp. 544–554.

[15] Dias, J. N., "Unsteady and Post-Stall Model Identification Using Dynamic Stall Maneuvers," *AIAA Atmospheric Flight Mechanics Conference*, American Institute of Aeronautics and Astronautics Inc, 2015.

[16] van der Linden, C., *DASMAT-Delft University Aircraft Simulation Model and Analysis Tool, A Matlab/Simulink Environment for Flight Dynamics and Control Analysis*, Delft University of Technology, 1998.

[17] van den Hoek, M., de Visser, C., and Pool, D., "Identification of a Cessna Citation II Model Based on Flight Test Data," *Fourth CEAS Specialist Conference on Guidance, Navigation and Control*, Springer, 2017, pp. 259–277.

[18] van Horssen, L., de Visser, C., and Pool, D., "Aerodynamic Stall and Buffet Modeling for the Cessna Citation II Based on Flight Test Data," *2018 AIAA Modeling and Simulation Technologies Conference*, American Institute of Aeronautics and Astronautics Inc, 2018.

[19] van Ingen, J. B., de Visser, C. C., and Pool, D. M., "Stall Model Identification of a Cessna Citation II from Flight Test Data Using Orthogonal Model Structure Selection," *AIAA Scitech 2021 Forum*, American Institute of Aeronautics and Astronautics Inc, 2021.

[20] Delfosse, A., Pool, D. M., and de Visser, C., "Asymmetric Stall Modeling of the Cessna Citation II Aircraft," , 2022. Unpublished.

[21] de Visser, C. C., and Pool, D. M., "Stalls and Splines: Current Trends in Flight Testing and Aerodynamic Model Identification," *Journal of Aircraft*, Vol. 60, No. 5, 2023, pp. 1480–1502.

[22] Smets, S., de Visser, C., and Pool, D., "Subjective Noticeability of Variations in Quasi-Steady Aerodynamic Stall Dynamics," *AIAA Scitech 2019 Forum*, American Institute of Aeronautics and Astronautics Inc, 2019.

[23] Imbrechts, A., de Visser, C., and Pool, D., "Just Noticeable Differences for Variations in Quasi-Steady Stall Buffet Model Parameters," *AIAA SciTech 2022 Forum*, American Institute of Aeronautics and Astronautics Inc, 2022.

[24] Klein, V., and Morelli, E. A., *Aircraft System Identification, Theory and Practice*, American Institute of Aeronautics and Astronautics Inc, 2006.

[25] Morelli, E. A., Cunningham, K., and Hill, M. A., “Global Aerodynamic Modeling for Stall/Upset Recovery Training Using Efficient Piloted Flight Test Techniques,” *AIAA Modeling and Simulation Technologies (MST) Conference*, American Institute of Aeronautics and Astronautics Inc, 2013.

[26] Mulder, J., *Design and evaluation of dynamic flight test manoeuvres*, Delft University of Technology, 1986.

[27] Morelli, E. A., “Flight Test of Optimal Inputs and Comparison with Conventional Inputs,” *Journal of Aircraft*, Vol. 36, No. 2, 1999, pp. 389–397.

[28] Seren, C., Bommier, F., Verdier, L., Bucharles, A., and Alazard, D., “Optimal Experiment and Input Design for Flight Test Protocol Optimization,” *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, American Institute of Aeronautics and Astronautics Inc, 2006.

[29] Ly, A., Marsman, M., Verhagen, J., Grasman, R. P., and Wagenmakers, E.-J., “A Tutorial on Fisher information,” *Journal of Mathematical Psychology*, Vol. 80, No. 1, 2017, pp. 40–55.

[30] Batterson, J. G., and Klein, V., “Partitioning of Flight Data for Aerodynamic Modeling of Aircraft at High Angles of Attack,” *Journal of Aircraft*, Vol. 26, No. 4, 1989, pp. 334–339.

[31] Morelli, E. A., “Efficient Global Aerodynamic Modeling from Flight Data,” *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, American Institute of Aeronautics and Astronautics Inc, 2012.

[32] Laban, M., *On-Line Aircraft Aerodynamic Model Identification*, Delft University of Technology, 1994.

[33] Julier, S. J., and Uhlmann, J. K., “A New Extension of the Kalman Filter to Nonlinear Systems,” *Signal Processing, Sensor Fusion, and Target Recognition VI*, SPIE, 1997, pp. 182–193.

[34] Wan, E. A., and van der Merwe, R., “The Unscented Kalman Filter for Nonlinear Estimation,” *IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium*, Institute of Electrical and Electronics Engineers Inc, 2000, pp. 153–158.

[35] Chowdhary, G., and Jategaonkar, R., “Aerodynamic parameter estimation from flight data applying extended and unscented Kalman filter,” *Aerospace Science and Technology*, Vol. 14, No. 2, 2010, pp. 106–117.

[36] Teixeira, B. O., Tôrres, L. A., Iscold, P., and Aguirre, L. A., “Flight path reconstruction - A comparison of nonlinear Kalman filter and smoother algorithms,” *Aerospace Science and Technology*, Vol. 15, No. 1, 2011, pp. 60–71.

[37] Gross, J., Gu, Y., Gururajan, S., Seanor, B., and Napolitano, M. R., “A Comparison of Extended Kalman Filter, Sigma-Point Kalman Filter, and Particle Filter in GPS/INS Sensor Fusion,” *AIAA Guidance, Navigation, and Control Conference*, American Institute of Aeronautics and Astronautics Inc, 2010.