Piecewise Polynomial Model Identification using Constrained Least Squares for UAS Stall

Vincent Guibert, Jean-Philippe Condomines, Mathieu Brunot, Murat Bronz

Piecewise Polynomial Model Identification using Constrained Least Squares for UAS Stall

Vincent Guibert [*,**] **Jean-Philippe Condomines** [*]
Mathieu Brunot [**] **Murat Bronz** [*]

[*] *ENAC, Université de Toulouse, 31055 Toulouse, France*
(e-mails: name.surname@enac.fr)
[**] *ONERA – The French Aerospace Lab, 31055 Toulouse, France*
(e-mails: vincent.guibert-ext@onera.fr, mathieu.brunot@onera.fr)

Abstract: In this paper, we propose a new piecewise polynomial model (PwPM) for the modelling of an Unmanned Aircraft System (UAS) aerodynamic coefficients over a wide flight envelope, where they are characterized by nonlinear and hysteresis phenomena. An associated identification method using Constrained Least Squares (CLS) is then presented and successfully applied to experimental data obtained from wind tunnel measurements. The resulting fitting is finally compared with the similar piecewise models PWPFIT and AERODAS, showing a greater precision in its modelling, while maintaining their relative simplicity and computation performance.

Keywords: Flight dynamics identification, Piecewise polynomial, Constrained least squares, Unmanned aircraft system, Stall modelling

1. INTRODUCTION

Accurate Unmanned Aircraft System (UAS) dynamics modelling is rarely straightforward and requires extensive experiments to ensure a sufficient representation of every aspect of the operational envelope of the aircraft. (Hoffer et al., 2014; Uhlig and Selig, 2017; Cunis et al., 2019)

Based on an identification phase using wind tunnel campaigns, Computational Fluid Dynamics (CFD) calculations or flight tests, a parametric model needs to be elaborated which is as representative as possible of the physical reality. This task is extremely challenging due to nonlinear and unsteady aerodynamic phenomena beyond the nominal flight envelope, as is the case during an upset situation (Saderla et al., 2018). For instance, at the stall angle of attack the flow around the wings detaches and goes from laminar to turbulent. This phenomenon creates a disparity between aerodynamic forces when flying in the so-called pre-stall (below the stall angle of attack) and post-stall (above this same angle of attack) domains. In addition, a stalled aircraft can still reach angles of attack below the previous value without re-attaching the air flow and only do so later at a new and lower angle of attack. This second nonlinear phenomenon is called aerodynamic hysteresis, and is of practical importance since it affects the behaviour of the aircraft and its possible recovery from stall and/or spin flight conditions.

Furthermore, with the ongoing certification and integration into civil airspace of UASs found in EU regulations (EU, May 2019), there is a clear need for reliable full-envelope models of UAS flight dynamics, and in particular during upset situations. Such models can be used for the certification of an aircraft or a control law as well as for safety and automation operations, including flight-envelope protection and restoration (Ancel et al., 2017; Bertrand et al., 2017). Having a robust upset recovery controller leveraging accurate predictions of the vehicle dynamics can indeed provide an increased safety for a drone in critical situations where it experiences an in-flight anomaly, as well as for other users of the airspace, ground infrastructures and people.

Modelling the aerodynamic behaviour of an aircraft can be done in several ways. For instance, one can solve numerically the Navier-Stokes equations as in Keller et al. (2019), use semi-empirical models as in Leishman and Beddoes (1989) or the AERODAS model (Spera, 2008), or use fully data-driven models such as locally weighted projection regression (Farcy et al., 2020) or splines (Morelli et al., 2013). However, while splines present today a powerful yet complex tool for accurate and smooth interpolation (Tol et al., 2016), they are unsuitable for functional analysis of trim conditions and stable sets, as in Kwatny et al. (2013). Moreover, high-fidelity models often require hard to obtain data, such as the flow separation point, which is usually not available. More recently, research works (Cunis et al., 2019) have been conducted on piecewise polynomial models in order to provide a constructive method fit for system analysis due to their continuous and differentiable nature. However, those results do not represent completely a real-life upset situation, due to the authors' choice to focus on deep stall control and as such to consider neither the hysteresis phenomenon nor a higher count of pieces.

Motivated by all these reasons, this paper focuses on a new formulation for a piecewise polynomial model and

its identification in order to improve the accuracy of aerodynamic forces modelling for UASs, and in particular around the aerodynamic hysteresis. The contributions include: (i) the definition in Section 2 of a new Piecewise Polynomial Model (PwPM) for the lift coefficient ; (ii) the presentation of the identification problem, where the Least Square (LS) cost metric is introduced Section 3 ; (iii) a quick and efficient algebraic solution to this problem using Lagrange multipliers in Section 4, followed by a proposal for hyper-parameters optimisation. Finally, experimental results based on wind tunnel data are presented Section 5, with a comparison of PwPM's performance to that of the PWPFIT and AERODAS methods.

2. PIECEWISE POLYNOMIAL MODEL (PWPM)

As previously stated, this study was motivated by the desire to model the aerodynamic coefficients of a UAS during stall. Since stall behaviour is mainly represented by the lift coefficient, it is this particular example that will be presented here. Based on *a priori* knowledge in Leishman and Beddoes (1989) and the experimental data presented Section 5, the model should respect the five following items:

(1) There should be two established domains respectively representing the pre-stall and post-stall dynamics.
(2) Those domains are to be linked by two transitions for going back and forth between them.
(3) No polynomial depends directly on the pitch rate $\dot{\alpha}$, although it can still be used to determine in which domain the drone is flying.
(4) The model is to be continuous and continuously differentiable on the whole of its domain.
(5) Each polynomial should be of the lowest degree possible to avoid over-fitting and maintain simplicity and computation performance.

As such, the following four-pieced polynomial model is proposed, where each polynomial is cubic in α and does not depend on $\dot{\alpha}$, as presented Fig. 1:

- A first polynomial, defining the pre-stall dynamics.
- A second polynomial, making the transition from the pre-stall domain at the stall angle of attack α_0 to the post-stall domain at α_1.
- A third polynomial, modelling the stalled dynamics.
- A final (fourth) polynomial, linking back from the post-stall to the pre-stall dynamics between the separation angles of attack α_2 and α_3.

In addition, in order to accommodate requirement 4 of the previous list, each two neighbouring polynomials (1 and 2 ; 1 and 4 ; 3 and 2 ; 3 and 4) are to respect same-value and tangency constraints at their separation. This proposed model leaves us with 16 polynomial coefficients to be identified under a set of 8 constraints.

3. MODEL IDENTIFICATION

Identifying the model is akin to the fitting problem below.

3.1 Problem description

Let there be a set of n measurements composed of q abscissae values $\boldsymbol{x_1}, \boldsymbol{x_2}, \cdots, \boldsymbol{x_q} \in \mathbb{R}^n$ and a single ordinate

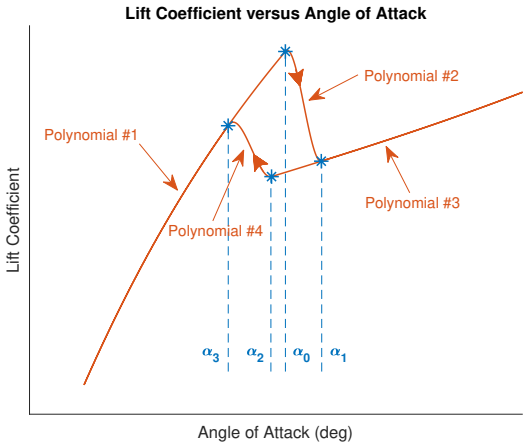

Fig. 1. The proposed model for the lift coefficient

value $\boldsymbol{z} \in \mathbb{R}^n$ defining a function $\mathfrak{f} : \mathbb{R} \times \mathbb{R} \times \cdots \times \mathbb{R} \to \mathbb{R}$ to be identified.

$$\forall j \in [\![1, n]\!], \quad \boldsymbol{z}_j = \mathfrak{f}\left(\boldsymbol{x_1}_j, \boldsymbol{x_2}_j, \cdots, \boldsymbol{x_q}_j\right) \tag{1}$$

Let there be a model, defined beforehand as in Section 2, defining p polynomials, their respective degree with each of the abscissa parameters d_j^i (where this notation denotes the degree of the i^{th} polynomial with respect to the j^{th} parameter) and their associated domains in the abscissa space $\mathcal{P}_i \subset \mathbb{R}^q$. Those domains should form a partition of the abscissa space such that they are disjoint two by two and cover it completely.

$$\begin{cases} \mathcal{P}_i \cap \mathcal{P}_j = \varnothing & \forall\ i, j \in [\![1, p]\!],\ i \neq j \\ \bigcup_{i=1}^{p} \mathcal{P}_i = \mathbb{R}^q \end{cases} \tag{2}$$

In order to better represent the underlying data, the model also defines a set of m constraint functions c_j depending linearly in the polynomial coefficients to be respected by the fitted polynomials. Those constraints will be considered respected if they reach a desired value d_j.

$$\text{"The } j^{th} \text{ constraint is respected"} \Leftrightarrow c_j(\cdot) = d_j \tag{3}$$

This linearity condition was chosen to simplify the algebraic solution below. It can seem reducing but it allows for the definition of most common constraints, including all derivatives continuity (including regular continuity) and specific value at a given abscissa.

Let there finally be a cost function $J \in \mathbb{R}$ to be minimised, quantifying the quality of the fitting. In this paper, a weighted Least Square (LS) formulation will be used such that J can be written

$$J = \sum_{j=1}^{n} \frac{1}{2} \gamma_j \left[\boldsymbol{z}_j - \mathfrak{F}(\boldsymbol{x_1}_j, \boldsymbol{x_2}_j, \cdots, \boldsymbol{x_q}_j)\right]^2 \tag{4}$$

where $\mathfrak{F} : \mathbb{R} \times \mathbb{R} \times \cdots \times \mathbb{R} \longrightarrow \mathbb{R}$ is the fitted model prediction and the coefficients $\gamma_j \neq 0$ are non-zero weighting coefficients on each measurement. It is noted that this choice of a cost function could reduce the quality of the model in the case where the abscissae (angles of attack) are too noisy. The difference in precision with other forms such as the Total Least Squares (TLS) is however only marginal, making this simpler LS formulation more attractive.

3.2 Matrical formulation

Based on the piecewise polynomial shape of the model $\mathfrak{F}$, one can introduce a simplified expression for each of the polynomials through a vector of monomial products and that of the polynomial coefficients. The vector of monomial products for the i^{th} polynomial, $\mu_i : \mathbb{R} \times \mathbb{R} \times \cdots \times \mathbb{R} \longrightarrow \mathbb{R}^{\delta_i}$, returns all the possible combinations of monomials products according to the maximal degrees defined by the model,

$$\mu_i(x_1, x_2, \cdots, x_q) = \begin{bmatrix} x_1^{d_1^i} \times x_2^{d_2^i} \times \cdots \times x_{q-1}^{d_{q-1}^i} \times x_q^{d_q^i} \\ x_1^{d_1^i} \times x_2^{d_2^i} \times \cdots \times x_{q-1}^{d_{q-1}^i} \times x_q^{d_q^i - 1} \\ \vdots \\ x_1^{d_1^i} \times x_2^{d_2^i} \times \cdots \times x_{q-1}^{d_{q-1}^i} \times x_q^{0} \\ x_1^{d_1^i} \times x_2^{d_2^i} \times \cdots \times x_{q-1}^{d_{q-1}^i - 1} \times x_q^{d_q^i} \\ \vdots \\ x_1^{0} \times x_2^{0} \times \cdots \times x_{q-1}^{0} \times x_q^{0} \end{bmatrix} \tag{5}$$

where $\delta_i = \prod_{j=1}^{q} \left(d_j^i + 1\right)$ is the dimension of this vector. It is noted here that the order in which the monomial products appear is arbitrary and does not matter as long as it remains the same throughout the problem. In addition, it is not required that all monomial products are used as long as the coefficients vector is amended in the same way.

By noting $\boldsymbol{\theta_i} \in \mathbb{R}^{\delta_i}$ the vector of coefficients for that same polynomial, one can then write

$$\begin{aligned} &\forall (x_1, x_2, \cdots, x_q) \in \mathcal{P}_i, \\ &\quad \mathfrak{F}(x_1, x_2, \cdots, x_q) = \langle\ \mu_i(x_1, x_2, \cdots, x_q)\ ,\ \boldsymbol{\theta_i}\ \rangle \\ &\qquad\qquad = \mu_i(x_1, x_2, \cdots, x_q)^\intercal \boldsymbol{\theta_i} \end{aligned} \tag{6}$$

where the notation $\langle\ \cdot\ ,\ \cdot\ \rangle$ denotes the inner product (here in the $\mathbb{R}^{\delta_i}$ space).

Using this simplified notation, one can then rewrite the cost function

$$\begin{aligned} J &= \sum_{i=1}^{p} J_i, \\ J_i &= \sum_{j=1}^{n_i} \frac{1}{2} \boldsymbol{\gamma_{i}}_j \left[\boldsymbol{z_{i}}_j - \mu_i(\boldsymbol{x_{1,i}}_j, \boldsymbol{x_{2,i}}_j, \cdots, \boldsymbol{x_{q,i}}_j)^\intercal \boldsymbol{\theta_i} \right]^2 \end{aligned} \tag{7}$$

where J_i is the contribution of the i^{th} polynomial to the total cost, the vectors $\boldsymbol{z_i}, \boldsymbol{x_{1,i}}, \cdots, \boldsymbol{x_{q,i}} \in \mathbb{R}^{n_i}$ are the restriction of their respective vectors to only the measurements whose abscissae lie in the domain $\mathcal{P}_i$, the vector $\boldsymbol{\gamma_i} \in \mathbb{R}^{n_i}$ is the vector of weights for the measurements in this same domain and n_i is the number of those measurements. This can be further simplified using the matrical formulation

$$J_i = \frac{1}{2} \left(\boldsymbol{z_i} - \boldsymbol{\Phi_i \theta_i}\right)^\intercal \boldsymbol{\Gamma_i} \left(\boldsymbol{z_i} - \boldsymbol{\Phi_i \theta_i}\right) \tag{8}$$

where $\boldsymbol{\Gamma_i} = \mathrm{diag}(\boldsymbol{\gamma_i}) \in \mathbb{R}^{n_i \times n_i}$ is the weight matrix and $\boldsymbol{\Phi_i} \in \mathbb{R}^{n_i \times \delta_i}$ is the matrix of monomial products for the i^{th} polynomial:

$$\boldsymbol{\Phi_i} = \begin{bmatrix} \mu_i(\boldsymbol{x_{1,i}}_1, \cdots, \boldsymbol{x_{q,i}}_1)^\intercal \\ \mu_i(\boldsymbol{x_{1,i}}_2, \cdots, \boldsymbol{x_{q,i}}_2)^\intercal \\ \vdots \\ \mu_i(\boldsymbol{x_{1,i}}_{n_i}, \cdots, \boldsymbol{x_{q,i}}_{n_i})^\intercal \end{bmatrix} \tag{9}$$

Finally, one can express the cost function in pure matrical formulation

$$J = \frac{1}{2} \left(\boldsymbol{z} - \boldsymbol{\Phi\theta}\right)^\intercal \boldsymbol{\Gamma} \left(\boldsymbol{z} - \boldsymbol{\Phi\theta}\right) \tag{10}$$

with $\boldsymbol{\Phi} \in \mathbb{R}^{n \times \delta}$ the matrix of monomial products for all measurements, $\boldsymbol{\theta} \in \mathbb{R}^{\delta}$ the total vector of polynomial coefficients and $\boldsymbol{\Gamma} \in \mathbb{R}^{n \times n}$ the total weight matrix. The total polynomial coefficients vector has dimension $\delta = \sum_{i=1}^{p} \delta_i$.

$$\boldsymbol{\Phi} = \mathrm{diag}(\boldsymbol{\Phi_1}, \boldsymbol{\Phi_2}, \cdots, \boldsymbol{\Phi_p}) \tag{11a}$$

$$\boldsymbol{\theta} = [\boldsymbol{\theta_1}^\intercal\ \boldsymbol{\theta_2}^\intercal\ \cdots\ \boldsymbol{\theta_p}^\intercal]^\intercal \tag{11b}$$

$$\boldsymbol{\Gamma} = \mathrm{diag}(\boldsymbol{\Gamma_1}, \boldsymbol{\Gamma_2}, \cdots, \boldsymbol{\Gamma_p}) \tag{11c}$$

It can be noted that the row-wise order in which the measurements appear does not matter and there is therefore no need to sort the data prior to the fitting.

Moreover, based on the linearity with respect to $\boldsymbol{\theta}$ of the constraint functions, one can write that

$$\forall j \in [\![1, m]\!], \quad \exists\ \boldsymbol{c_j} \in \mathbb{R}^{\delta} \mid c_j(\cdot) = \langle\ \boldsymbol{c_j}\ ,\ \boldsymbol{\theta}\ \rangle = \boldsymbol{c_j}^\intercal \boldsymbol{\theta} \tag{12}$$

such that all the constraints can too be rewritten under the matrical formulation

$$\text{"All the constraints are respected"} \Leftrightarrow \boldsymbol{C\theta} = \boldsymbol{d} \tag{13}$$

where $\boldsymbol{C} \in \mathbb{R}^{m \times \delta}$ is a matrix deriving from those constraint vectors and $\boldsymbol{d} = [d_1\ d_2\ \cdots\ d_j]^\intercal \in \mathbb{R}^m$ is the vector of the constraints desired values.

$$\boldsymbol{C} = [\boldsymbol{c_1}\ \boldsymbol{c_2}\ \cdots\ \boldsymbol{c_m}]^\intercal \tag{14}$$

The problem at hand can then be entirely expressed under the simplified matrical form of the Constrained Least Squares (CLS) problem:

Problem 1. (Constrained Least Squares).

$$\begin{aligned} \text{Find } \boldsymbol{\theta^*} = \arg\min_{\boldsymbol{\theta}} &\frac{1}{2} \left(\boldsymbol{z} - \boldsymbol{\Phi\theta}\right)^\intercal \boldsymbol{\Gamma} \left(\boldsymbol{z} - \boldsymbol{\Phi\theta}\right) \\ s.t. \quad &\boldsymbol{C\theta} = \boldsymbol{d} \end{aligned}$$

where $\boldsymbol{\theta^*}$ is the constrained solution.

4. PROBLEM SOLUTION

4.1 Common solving methods

Several work have already been conducted on the CLS problem. Three methods in particular come to mind:

A first solution makes use of the simplification:

$$\boldsymbol{\theta^*} = \arg\min_{\boldsymbol{\theta}} \frac{1}{2} \left(\boldsymbol{z} - \boldsymbol{\Phi\theta}\right)^\intercal \boldsymbol{\Gamma} \left(\boldsymbol{z} - \boldsymbol{\Phi\theta}\right) + \left\|\boldsymbol{d} - \boldsymbol{C\theta}\right\|_2^2 \tag{15}$$

This however suffers from a trade-off between a low value for the cost J and respecting the constraints. Depending on the data at hand, the obtained results can be unusable because the constraints are not respected.

To balance this issue, a weight constant $\omega \in \mathbb{R}^+$ has been proposed to put more weight on the constraint while retaining the simplicity of the previous formulation under the new form

$$\boldsymbol{\theta^*} = \arg\min_{\boldsymbol{\theta}} \frac{1}{2} \left(\boldsymbol{z} - \boldsymbol{\Phi\theta}\right)^\intercal \boldsymbol{\Gamma} \left(\boldsymbol{z} - \boldsymbol{\Phi\theta}\right) + \omega \left\|\boldsymbol{d} - \boldsymbol{C\theta}\right\|_2^2 \tag{16}$$

such that when ω approaches $+\infty$, the constraint is perfectly respected (Golub and Van Loan, 2013).

Finally, the most common solution is to make use of iterative algorithms such as quadratic programming, as is used by MATLAB's "lsqlin" method (Coleman and Li, 1996). This method is more general, as it also authorizes inequality constraints, but it can have quite a long execution time and require high computation power and memory space.

4.2 Algebraic solution

An exact solution to Problem 1 can be found using the Lagrange multiplier method under the assumption that $\boldsymbol{\Delta} = \boldsymbol{\Phi}^\intercal\boldsymbol{\Gamma}\boldsymbol{\Phi} \in \mathbb{R}^{\delta\times\delta}$ is invertible and so is $\boldsymbol{C}\boldsymbol{\Delta}^{-1}\boldsymbol{C}^\intercal \in \mathbb{R}^{m\times m}$. It arises from the KKT matrix equality

$$\begin{bmatrix}\boldsymbol{\Delta} & \boldsymbol{C}^\intercal \\ \boldsymbol{C} & \boldsymbol{0}\end{bmatrix}\begin{bmatrix}\boldsymbol{\theta}^* \\ \boldsymbol{\lambda}^*\end{bmatrix} = \begin{bmatrix}\boldsymbol{\Phi}^\intercal\boldsymbol{\Gamma}\boldsymbol{z} \\ \boldsymbol{d}\end{bmatrix} \tag{17}$$

where the solution for $\boldsymbol{\theta}^*$ is then given by the 2×2 block matrix inversion formula:

$$\boldsymbol{\theta}^* = \boldsymbol{\theta}^\circ + \boldsymbol{P}\left(\boldsymbol{d} - \boldsymbol{C}\boldsymbol{\theta}^\circ\right) \tag{18}$$

where $\boldsymbol{\theta}^\circ = \boldsymbol{\Delta}^{-1}\boldsymbol{\Phi}^\intercal\boldsymbol{\Gamma}\boldsymbol{z} \in \mathbb{R}^\delta$ is the unconstrained solution and $\boldsymbol{P} = \boldsymbol{\Delta}^{-1}\boldsymbol{C}^\intercal(\boldsymbol{C}\boldsymbol{\Delta}^{-1}\boldsymbol{C}^\intercal)^{-1} \in \mathbb{R}^{\delta\times m}$ is a correction matrix. Several properties can be shown from this solution: 1) in the case where the unconstrained solution respects the constraints, the constrained solution is also the unconstrained one ($\boldsymbol{C}\boldsymbol{\theta}^\circ = \boldsymbol{d} \implies \boldsymbol{\theta}^* = \boldsymbol{\theta}^\circ$), and 2) in any case the above result respects the constraints (by noting that $\boldsymbol{C}\boldsymbol{P} = \boldsymbol{I_m}$).

It can also be noted that by construction the non-singularity conditions on $\boldsymbol{\Delta}$ and $\boldsymbol{C}\boldsymbol{\Delta}^{-1}\boldsymbol{C}^\intercal$ can be simplified using the fact that $\boldsymbol{\Gamma}$ is diagonal with non-zero coefficients to the form

$$\text{"A solution exists"} \Leftrightarrow \begin{cases}\boldsymbol{\Phi} \text{ has full column rank} \\ \boldsymbol{C} \text{ has full row rank}\end{cases} \tag{19}$$

from which an additional condition can be derived: there must be at least as many coefficients than there are constraints ($m \leq \delta$). It might be worth noting that in the case $m = \delta$, the solution becomes $\boldsymbol{\theta}^* = \boldsymbol{C}^{-1}\boldsymbol{d}$ which is the only value respecting the constraint.

4.3 Hyper-parameters optimisation

Although the solution from (18) always yields the optimal solution to the CLS problem, it requires the knowledge of both matrices $\boldsymbol{\Phi}$ and $\boldsymbol{C}$ to compute. In the case where either the distribution of measurements in the various domains $\mathcal{P}_i$ or the constraints c_j depend on a set of additional parameters $\boldsymbol{s} \in \mathbb{R}^r$ another optimisation process must be devised.

Due to the impracticality of conducting a grid search over a high-dimensional grid, the solution we retained for this is the use of an iterative optimisation algorithm over the optimal cost

$$J^*(\boldsymbol{s}) = J(\boldsymbol{\theta}^*)\big|_{\boldsymbol{s}} \tag{20}$$

where $\boldsymbol{s}$ is treated as a vector of hyper-parameters. This solution has the advantage of reducing the complexity of the iterative algorithm by only considering the r hyper-parameters for optimisation, ensuring higher chances of convergence and speed.

Several optimisation algorithm can be used but it was chosen in our case to use the first order adaptive gradient descent (AdaGrad) method (Duchi et al., 2011). This choice was motivated by the desire to improve on the basic gradient descent for stability and convergence speed while keeping a reduced complexity relative to Newton derivatives. Computing the Hessian is indeed a lengthy process and yielded no significant improvements in our tests on the problem at hand.

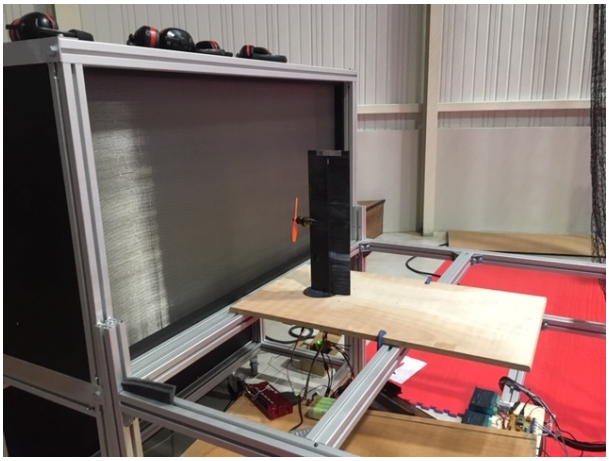

Fig. 2. The experimental setup used to gather data.

It should be noted however that any gradient-based method relies on the continuity of J^*. This is trivially guaranteed while no measurement changes domain due to their variation, assuming that conditions (19) are respected, but it should otherwise be ensured for each two neighbouring domains. In the case of the lift model defined Section 2, the hyper-parameters are the four separation angles of attack α_i, $i \in [\![0, 3]\!]$ described in Fig. 1. Continuity of the optimal cost function is ensured by the continuity constraints between each two neighbouring domains and the fact that the weights are kept constant throughout the study.

5. APPLICATION TO EXPERIMENTAL DATA

Following from the previously-defined model for the lift coefficient and identification process, the polynomial identification method will be applied to experimental data.

5.1 Experimental setup

Experimental data has been obtained as shown Fig. 2 from eight open wind tunnel experiments on a single aileron-less straight wing with NACA profile 0012, chord 0.15m and span 0.5m, placed perpendicular to the wind flow as to get no side slip. The flow straightening grid hides a set of 162 independent fans placed in a 9-by-18 configuration that were tasked with generating a constant and uniform wind velocity for the duration of each experiment, while the angle of attack of the wing was progressively increased from 0° to 35° at a speed of approximately $10°/s$ and back to 0° at $12°/s$. Two wind velocities of 7.5m/s and 10m/s were tested, with four experiments each, and the wind velocity was measured by a Pitot tube located ahead of the wing. Although it can be seen from Fig. 2 that the wing used was fitted with a propeller, it was free-moving during the experiments and only spun due to the incoming wind. The same experiment with a clean, propeller-less

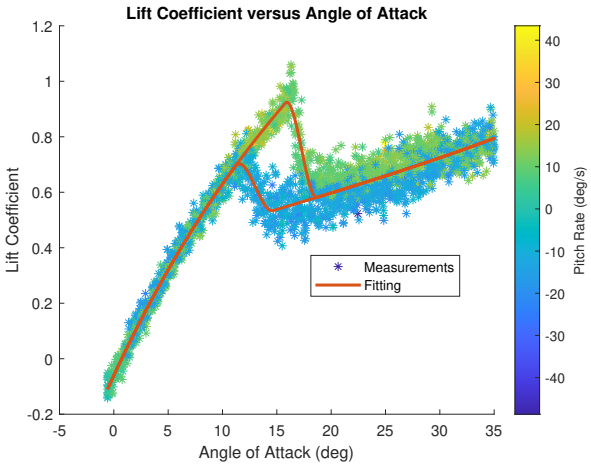

Fig. 3. The fitting results on the experimental data

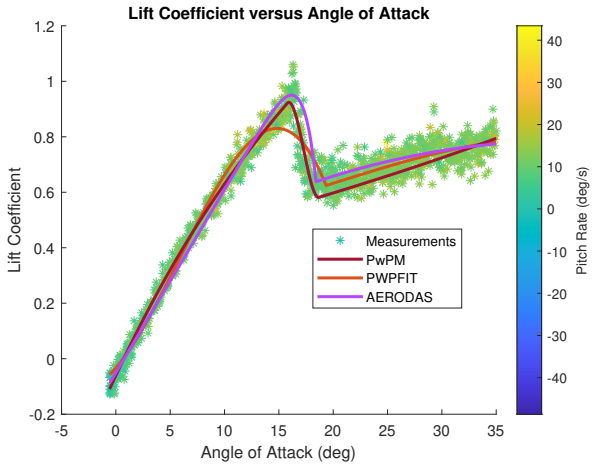

Fig. 4. Comparison of the fitting for the three models

wing yielded no significant change in the collected data and as such it was decided to keep it for this study.

5.2 Results

The resulting fitting is presented Fig. 3, where the hyper-parameters have been found to be $\alpha_0 = 15.77°$, $\alpha_1 = 18.69°$, $\alpha_2 = 14.65°$ and $\alpha_3 = 11.13°$.

Immediately, one can find that the model seems to accurately represent the lift on the whole of the domain considered. The total cost is $J = 1.08 \times 10^{-3}$, which represents in our case half of the Mean Squared Error (MSE) of the fitting with respect to the given data. In particular, the pre-stall model seems to accurately represent the lift of the wing up to the stall angle of 15.77°, with an MSE of only 8.08×10^{-4}. The fitting in the post-stall domain (third polynomial) is also accurately representing the lift coefficient of the wing but suffers from the noisier data given, with a higher MSE of 2.80×10^{-3}. This is due to unstationary effects of the turbulent flow that "pollute" our measurements. Finally, the fitting about the aerodynamic hysteresis interestingly gave good results, with once again an accurate representation of the lift. The MSEs on these domains were 5.42×10^{-3} for the polynomial going from the pre-stall to the post-stall domain and 2.98×10^{-3} going in the opposite direction.

A robustness analysis of the iterative optimisation process was conducted by running the algorithm a thousand times with randomized initial hyper-parameters (taken from the estimated best solution plus or minus 1.5° for the first two separations and ±1.7° for α_2 and α_3). It gave a convergence rate of 100% for a stop condition of $\forall j \in [\![1, r]\!], \|\Delta \boldsymbol{s}_j / \boldsymbol{s}_j\| < 10^{-4}$ (0.01 %). The maximum number of iterations was 518 for an average of 163.6, a median of 146.5 and a standard deviation of 87.18, showing the ability of the algorithm to quickly yield an accurate result.

5.3 Comparison with other methods

The results from Section 5.2 will finally be compared with two piecewise polynomial formulations: the PWPFIT method developed by the authors of Cunis et al. (2019) and the AERODAS model (Spera, 2008). However, since the authors of these papers chose not to consider the hysteresis phenomenon, only the measurements with a positive pitch rate will be used. The results of this comparison are presented Fig. 4.

Without even considering, as stated, the hysteresis, the PwPM method we propose here seems to be better at modelling the lift of the wing, and in particular in the transition from pre-stall to post-stall dynamics, where Fig. 4 shows that PWPFIT fails to account for the brusque variation in lift and AERODAS's simpler shape overshoots the measurements. In addition, both models are not differentiable at the separation between the pre-stall and post-stall curves, an issue that was addressed in part by the authors of Cunis et al. (2020) by adding a sigmoid blending term between the two polynomials. However, doing so loses the polynomial formulation. PwPM in turn seems to underestimate the lift in the post-stall domain, which can be explained by the fact that it takes into account some unrepresented data with negative pitch rates the other twos don't.

The poorer performance of the PWPFIT and AERODAS models can be traced back to some choices by the authors. Indeed, they both consider the lift on the entire flight envelope up to $\alpha = 90°$ of angle of attack to be composed of only two pieces, one for pre-stall flight and one for post-stall, which forces them to sacrifice some precision for the sake of simplicity of the model. As a result, the transition is unsurprisingly where the fitting is the poorest. In addition, the experiment featured is characteristic of the so-called leading edge stall where the air flow detaches abruptly due to the burst of a laminar separation "bubble" near the leading edge and the flow separation point moves backward toward the trailing edge. Such a behaviour is a lot less common in the literature than the trailing edge stall and one can assume that the authors of Cunis et al. (2019) and Spera (2008) developed their models with this second behaviour in mind, making it less effective on the dataset at hand. PwPM is however compatible as is with trailing edge stall behaviour and it is expected that it would still outperform its competitors on such a dataset.

Table 1 proposes a summary of the MSEs for each of the methods on the three domains considered, namely in the pre-stall ($\alpha < \alpha_0$), transition ($\alpha_0 < \alpha < \alpha_1$) and

Table 1. Mean Squared Errors comparison for the methods presented

Domain	PwPM	PWPFIT	AERODAS
pre-stall	8.29×10^{-4}	1.50×10^{-3}	1.47×10^{-3}
transition	5.42×10^{-3}	1.16×10^{-2}	1.69×10^{-2}
post-stall	2.67×10^{-3}	1.72×10^{-3}	1.97×10^{-3}
Whole domain	2.05×10^{-3}	2.42×10^{-3}	2.95×10^{-3}

post-stall ($\alpha > \alpha_1$) domains. Each of the three models considers a different value for α_1 but the value from PwPM was used here for continuity. As noted before, the two methods PWPFIT and AERODAS are better than PwPM on the post-stall domain. However, their overall precision is impacted by their worst fitting on the two other pieces, where their MSEs are significantly higher, and in particular on the transition domain.

6. CONCLUSION AND PERSPECTIVES

This paper proposed a new piecewise polynomial model (PwPM) for the modelling of the lift coefficients of a UAS as well as a Constrained Least Squares (CLS) method to identify it. The model was chosen specifically for a better modelling of the behaviour of the aircraft at stall when going from laminar to turbulent flow and vice-versa, including the aerodynamic hysteresis. Comparison with the PWPFIT two-pieced polynomial model and the AERODAS one showed that the PwPM approach outperformed both overall and particularly in the transition from pre-stall to post-stall dynamics, fulfilling its requirements. In addition, the proposed method for identification is able to quickly yield accurate results, even with noisy experimental data.

Further work will however be needed on the subject before upset recovery can be tackled. In particular, polynomial models for the other aerodynamic coefficients not presented here will have to be defined and identification on real flight data must be carried on. In addition a recursive adaptation of this algorithm must be defined before attempting flights in the post-stall domains in order to ensure airspace and ground safety.

ACKNOWLEDGEMENTS

This work was supported by the Defense Innovation Agency (AID) of the French Ministry of Defense (research project CONCORDE N° 2019 65 0090004707501).

REFERENCES

Ancel, E., Capristan, F.M., Foster, J.V., and Condotta, R.C. (2017). Real-time risk assessment framework for unmanned aircraft system (uas) traffic management (utm). *17th AIAA Aviation Technology, Integration, and Operations Conference*, 3273, 3273.

Bertrand, S., Raballand, N., Viguier, F., and Muller, F. (2017). Ground risk assessment for long-range inspection missions of railways by uavs. *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, 1343–1351.

Coleman, T.F. and Li, Y. (1996). A Reflective Newton Method for Minimizing a Quadratic Function Subject to Bounds on Some of the Variables. *SIAM Journal on Optimization*, 6(4), 1040–1058.

Cunis, T., Burlion, L., and Condomines, J.P. (2019). Piecewise polynomial modeling for control and analysis of aircraft dynamics beyond stall. *Journal of Guidance, Control, and Dynamics*, 42(4), 949–957.

Cunis, T., Condomines, J.P., Burlion, L., and la Cour-Harbo, A. (2020). Dynamic stability analysis of aircraft flight in deep stall. *Journal of Aircraft*, 57(1), 143–155.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 39.

EU (May 2019). Commission implementing regulation (eu) 2019/947 of 24 may 2019 on the rules and procedures for the operation of unmanned aircraft (text with eea relevance). *Official Journal of the European Union*, 2019.

Farcy, D., Khrabrov, A.N., and Sidoryuk, M.E. (2020). Sensitivity of spin parameters to uncertainties of the aircraft aerodynamic model. *Journal of Aircraft*, 1–16.

Golub, G.H. and Van Loan, C.F. (2013). *Matrix computations. xxi*. The Johns Hopkins University Press Baltimore.

Hoffer, N.V., Coopmans, C., Jensen, A.M., and Chen, Y. (2014). A survey and categorization of small low-cost unmanned aerial vehicle system identification. *Journal of Intelligent & Robotic Systems*, 74(1-2), 129–145.

Keller, D., Farcy, D., and Le Roy, J.F. (2019). Numerical Investigation of the Aerodynamic Behavior of a Generic Light Aircraft. In *AAAF AERO2019*. PARIS, France.

Kwatny, H.G., Dongmo, J.E.T., Chang, B.C., Bajpai, G., Yasar, M., and Belcastro, C. (2013). Nonlinear analysis of aircraft loss of control. *Journal of Guidance, Control, and Dynamics*, 36(1), 149–162.

Leishman, J.G. and Beddoes, T. (1989). A semi-empirical model for dynamic stall. *Journal of the American Helicopter society*, 34(3), 3–17.

Morelli, E.A., Cunningham, K., and Hill, M.A. (2013). Global aerodynamic modeling for stall/upset recovery training using efficient piloted flight test techniques. In *AIAA Modeling and Simulation Technologies (MST) Conference*, 4976.

Saderla, S., Kim, Y., and Ghosh, A. (2018). Online system identification of mini cropped delta uavs using flight test methods. *Aerospace Science and Technology*, 80, 337–353.

Spera, D.A. (2008). Models of Lift and Drag Coefficients of Stalled and Unstalled Airfoils in Wind Turbines and Wind Tunnels. Technical Report NASA/CR-2008-215434, NASA – Jacobs Technology Inc.

Tol, H., De Visser, C., Sun, L., van Kampen, E., and Chu, Q. (2016). Multivariate spline-based adaptive control of high-performance aircraft with aerodynamic uncertainties. *Journal of Guidance, Control, and Dynamics*, 39(4), 781–800.

Uhlig, D.V. and Selig, M.S. (2017). Modeling micro air vehicle aerodynamics in unsteady high angle-of-attack flight. *Journal of Aircraft*, 54(3), 1064–1075.