Global Aerodynamic Modeling Using Automated Local Model Networks in Real Time

Rose Weinstein*
NASA Langley Research Center, Hampton, Virginia, 23681

James E. Hubbard, Jr., NAE†
Texas A&M University, College Station, TX, 77843

A novel method is presented for automated real-time global aerodynamic modeling using local model networks, known as Smoothed Partitioning with Localized Trees in Real Time (SPLITR), as part of NASA's Learn-to-Fly technology development initiative. The global nonlinear aerodynamics are partitioned into several local regions known as cells, with the dimension, location, and timing of each partition automatically selected based on a residual characterization procedure, under the constraints of real-time operation. Regression trees represent the successive partitioning of the global flight envelope and describe the evolution of the cell structure. Recursive equation-error least-squares parameter estimation in the time domain is used to estimate a model that represents the local aerodynamics in each region, so that it can be updated independently with non-contiguous data in the range of each cell over time. A weighted superposition of these piecewise local models across the flight envelope forms a global nonlinear model that also accurately captures the local aerodynamics. The SPLITR approach is demonstrated using both simulation and flight data, and the results are analyzed in terms of model predictive capabilities as well as interpretability. The results show that SPLITR can be used to automatically partition complex nonlinear aerodynamic behavior, produce an accurate model, and provide valuable physical insight into the local and global aerodynamics.

I. Nomenclature

a_x, a_y, a_z	=	body-axis translational acceleration, g
b	=	wing span, ft
$\bar{c}$	=	wing mean aerodynamic chord, ft
C_X, C_Y, C_Z	=	body-axis nondimensional force coefficients
C_L, C_D	=	wind-axis nondimensional force coefficients
C_l, C_m, C_n	=	body-axis nondimensional moment coefficients
d	=	propeller diameter, ft
I_x, I_y, I_z, I_{xz}	=	inertia tensor elements, slug-ft^2
J	=	advance ratio
m	=	mass, slug
N	=	number of data points
p, q, r	=	body-axis roll, pitch, and yaw rates, rad/s
$\bar{q}$	=	dynamic pressure, lbf/ft^2
S	=	wing reference area, ft^2
T	=	thrust, lbf
V	=	true airspeed, ft/s
w	=	validity function
α	=	angle of attack, rad or deg
β	=	sideslip angle, deg
δ_e	=	elevator deflection, rad
μ	=	mean
ν	=	residual
σ	=	standard deviation
Σ	=	covariance matrix
ϕ	=	partitioning variable
Ω	=	propeller speed, rad/s

Superscripts

^{T}	=	transpose
$\dot{}$	=	time derivative
$\hat{}$	=	estimate
$\bar{}$	=	mean
$^{-1}$	=	matrix inverse

Subscripts

$_0$	=	reference or bias value

*Pathways Student, Flight Dynamics Branch, MS 308; Graduate Research Assistant, Department of Aerospace Engineering, University of Maryland, AIAA Student Member.

†TEES Eminent Professor, Hagler Institute Fellow, Department of Mechanical Engineering, AIAA Fellow.

II. Introduction

The conventional paradigm for the development and flight testing of new or modified aircraft is an iterative, time-consuming process that typically involves numerous test techniques to generate an aircraft model and design a control system. The NASA Learn-to-Fly (L2F) initiative [1, 2], under the Transformational Tools and Technologies (TTT) Project, aims to improve this process by replacing most of the ground-based testing and analysis with automated, onboard, real-time tools that provide in-flight aircraft modeling [3, 4], guidance [5], and control [6–9], with no reliance on prior knowledge of the particular aircraft dynamics, as depicted in Fig. 1. This enabling technology can provide real-time self-awareness for autonomous systems and assist health monitoring, fault detection, and self-healing capabilities. In particular, as Urban Air Mobility (UAM) becomes a widespread priority, and novel and unconventional vehicles are emerging, rapid and efficient tools for developing, testing, and ensuring the safety of these systems are critical. Furthermore, traditional understanding of conventional aircraft aerodynamic properties cannot necessarily be fully relied upon for these unprecedented configurations, so the tools must be adaptive, customizable, and informative as these systems are explored.

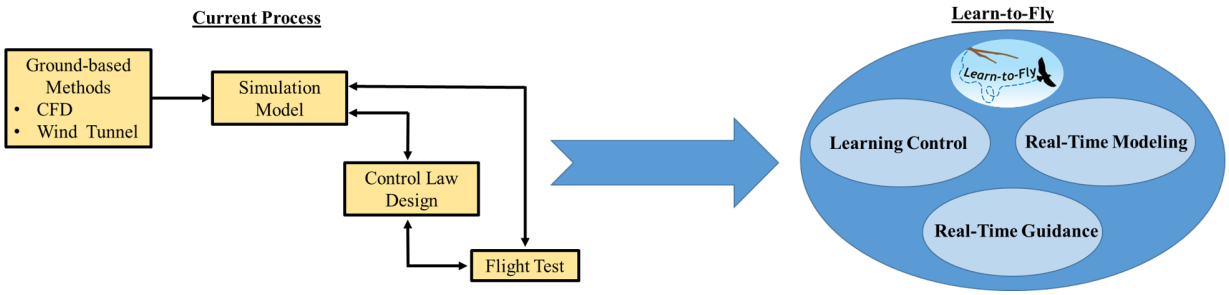


Fig. 1 Conventional aircraft development process vs. Learn-to-Fly concept.

Many of the L2F capabilities rely on an identified model, and this work focuses on building the global nonlinear aerodynamic model of aircraft in real time. System identification of aircraft aerodynamics from flight data often involves small-amplitude perturbation maneuvers about a trimmed reference flight condition, for which a linear model form is assumed, and which results in only a locally valid model. For global modeling, the aerodynamics are known to be more generally nonlinear, and so the modeling process must account for the additional complexities. Early work has combined such perturbation maneuvers across multiple flights by interpolating between various flight conditions [10, 11].

Past work that has considered the model as a black box system employed a wide variety of techniques, including machine learning and neural networks [12–14]. These approaches take advantage of powerful learning tools; however, they often require a considerable amount of training data, do not operate in real time, and lack physical insight and interpretability while prioritizing the input-output relationships.

More commonly, aerodynamic force and moment coefficients are expressed parametrically as polynomial expansions using explanatory variables that are measured in flight, which is consistent with viewing the estimated parameters as stability and control derivatives from which physical insight can be extracted. The challenge lies in establishing a pool of candidate linear and nonlinear model terms, known as regressors, and then selecting those that retain the strongest modeling capability. Past offline work has used stepwise regression, spline functions, and multivariate simplex splines in this context [15–17]. Recent L2F work has shown that multivariate orthogonal function (MOF) modeling can be used to efficiently model the nonlinear aerodynamics in this manner, and has been successfully tested in flight in real time [4, 18, 19]. Notably, this orthogonalization procedure has been used to build the generic global aerodynamic (GGA) model structure [20]. While this method generates a single nonlinear polynomial model across the global aerodynamics, it does not have the capability to characterize significant localized variations. The inclusion of candidate spline functions in the regressor pool can provide improved localization, but without knowledge of where within the explanatory variables these terms should be placed, a general expansive pool covering all practical possibilities is prohibitive for real-time operation.

Central to the work discussed in this paper is an approach known as data partitioning, through which the global aerodynamics are partitioned into several regions where local polynomial models are estimated. Whereas approaches like stepwise regression and MOF characterize nonlinearities by incorporating higher-order and multivariate modeling terms, the data partitioning method instead divides the flight data into regions where simple models can be used. Local model networks (LMNs) extend this idea by expressing the global nonlinear model as a smoothly weighted combination

of the local models. Furthermore, regression trees can be utilized to describe the evolution of the LMN cell structure and parameter estimates. These approaches retain the easily interpretable polynomial model form, while providing a nonlinear model that is globally valid and that still accurately captures the local aerodynamics.

The method introduced in this paper, called Smoothed Partitioning with Localized Trees in Real Time (SPLITR), extends LMNs for real-time parameter estimation and automated determination of cell structure and location, with an emphasis on retaining interpretability of the resulting model. The goal of this research is to develop and test a modeling approach that can be used in real time onboard an aircraft with no reliance on prior knowledge of the particular aerodynamics, that is physically meaningful and easily interpretable, and that provides excellent predictive capabilities.

Section III discusses the background that this work is built upon, including data partitioning, local model networks, and regression trees, as well as associated literature. Section IV details the SPLITR approach, which is composed of experiment design, parameter estimation, cell structure determination, and global weighting. Section V discusses a simplified example of applying SPLITR to F-16 simulation data, and Section VI presents the SPLITR modeling results using E1 flight data. Finally, Section VII discusses conclusions drawn from this work.

III. Background

A. Data Partitioning

When flight data cover a wide range of explanatory variables and cannot be adequately captured by a single linear model, several local models can be used to describe the large amplitude or nonlinear behavior through data partitioning [21]. This approach converts a complex nonlinear modeling problem into several simpler local, and possibly linear, models by dividing the flight data into multiple regions, and estimating a separate model using the data in each cell. The global nonlinearities are often characterized by particular explanatory variables, such as nominal angle of attack, which can be divided into local regions in which the effects of the nonlinearities are mitigated or removed. If the local model can be approximated as linear, then extensive linear systems theory can be employed to investigate and understand the dynamics. With certain parameter estimation methods, such as equation-error least-squares in the time domain, the data associated with the model in each cell do not need to be contiguous in time, so flight test data across the global flight envelope can be incorporated into the local cell to which they belong. Data partitioning for aerodynamic modeling was first employed in Refs. [22, 23] for large-amplitude maneuvers; however, a fine prescribed resolution of cells was specified through analysis of data density, and so the cell locations were not customized based on particular nonlinear variations in the data, which resulted in a complex cell structure.

B. Local Model Networks

A local model network is an architecture that is used to approximate a nonlinear function as a weighted combination of several piecewise local models, as shown in Fig. 2 for a network consisting of $k = 1, \ldots, M$ cells [24].

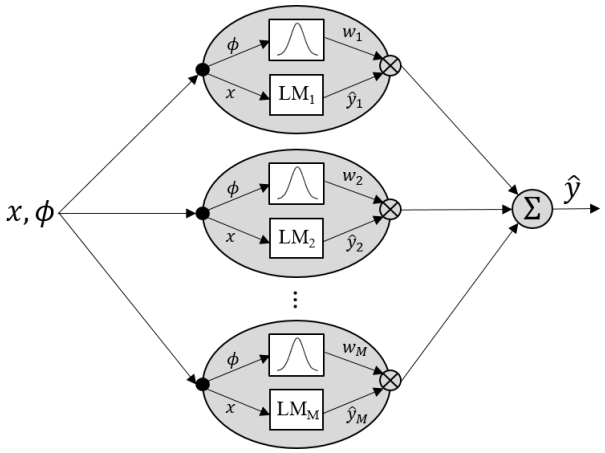


Fig. 2 Local model network architecture (adapted from Ref. [25]).

The explanatory variables, $\boldsymbol{x} = [\boldsymbol{x}_1\ \boldsymbol{x}_2\ \ldots\ \boldsymbol{x}_{n_x}]$, are used as regressors for local function approximation, and the

partitioning variables (PVs), $\boldsymbol{\phi} = [\boldsymbol{\phi}_1\ \boldsymbol{\phi}_2\ \ldots\ \boldsymbol{\phi}_{n_\phi}]$, are the terms that are specified as possible sources of nonlinearity with which to partition the model. The variables $\boldsymbol{x}$ and $\boldsymbol{\phi}$ can be selected independently and may be specified to include all, some, or none of the same terms. The normalized validity function for the k^{th} local model, $\boldsymbol{w}_k$, which is a nonlinear function of $\boldsymbol{\phi}$, weights each local model and represents the strength of influence of the local model across the PVs. The validity functions can be chosen from a variety of different shapes, including Gaussian, ramp, and boxcar functions. The global nonlinear output $\hat{\boldsymbol{y}}$ is the superposition of the estimated local model outputs $\hat{\boldsymbol{y}}_k$ that are weighted with the normalized validity functions, as will be discussed in more detail in Section IV.D. Note that in Fig. 2, the validity function normalization is not explicitly shown. Unlike traditional global modeling techniques in which the optimization is performed globally, the LMN architecture emphasizes the relevance of locality by performing several local optimizations. And unlike black box modeling techniques which sacrifice internal model transparency, the cell partitioning provides interpretability and physical insight within the global model construct.

C. Regression Trees

Regression trees can be used to represent a logical, hierarchical partitioning scheme that successively divides up the PVs for LMNs and develops the cell structure. These decision trees also provide a useful visualization of the evolution of the cell structure and the successive decision making that generates the partitions. The representation of the spatial localization of the nonlinearities in the data by partitioning the explanatory variable space can manifest in various ways. An axis-orthogonal split partitions a single dimension, or PV, at once and maintains the benefit of simplicity and interpretability of the resulting cells. Axis-oblique partitions allow splits that extend through two or more dimensions at once, and while they permit a more flexible cell structure, multi-dimensional oblique splits can add complexity and sacrifice model transparency. Within axis-orthogonal partitioning schemes, binary split locations are commonly employed, which can be easily visualized in a decision tree such as in Fig. 3, where a new cell is added at each split level in a case with two PVs. The cell that is divided is referred to as the parent, and the subsequent partitioned regions are called the child cells.

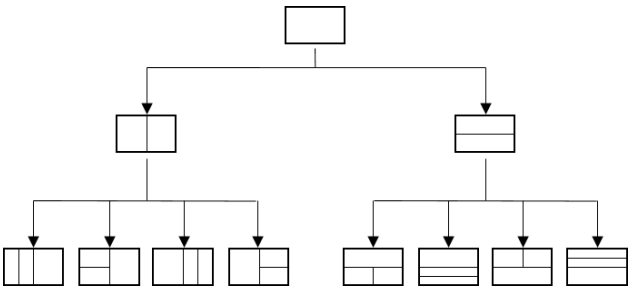


Fig. 3 Binary axis-orthogonal decision tree (adapted from Ref. [26]).

Iterative, binary axis-orthogonal regression tree algorithms such as LOLIMOT have been successfully used in a variety of nonlinear static and dynamic modeling applications [25, 27, 28], and many of these tools have been incorporated into a Regression Tree object and associated functions in MATLAB® [29]. In the context of aerodynamics, these techniques were used to model the aerodynamic coefficients for nonlinear X-31 flight data in Ref. [26], and similar iterative partitioning involving fuzzy logic in Ref. [18] was used to estimate aerodynamic models for an Aermacchi MB-326M Impala jet aircraft while the parameters were updated in real time, with further applications of this fuzzy modeling approach explored in Ref. [9].

IV. Approach

This section describes the methodology behind the SPLITR approach, which includes experiment design, parameter estimation, cell structure determination, and global weighting. During practical real-time onboard applications, computational power is limited, so the tools must be recursive and efficient, and data must be processed immediately and cannot be stored for later use, such as in iterative schemes. These constraints of feasible real-time operation strongly influenced the approach at each step. The priority of preserving model interpretability also played a large role in the SPLITR development. Throughout this section, intermediate results from modeling $C_L = f(\alpha)$ using simulated F-16 data will be shown to illustrate certain concepts, with the aircraft properties and final modeling results discussed in

Section V. Software for input design, recursive least-squares parameter estimation, and data filtering throughout this work used parts of the software toolbox called System IDentification Programs for AirCraft, or SIDPAC [21].

A. Experiment Design

The quality of the estimated model is predicated on the data used in the system identification, so the experiment design is an essential step toward ensuring rich flight data to inform the modeling process. First, sufficient dynamic excitation is required to obtain data with high information content. Traditionally, single-axis excitation using doublets or frequency sweeps has been used, but prior research has shown that Programmable Test Inputs (PTIs) that apply automated orthogonal optimized multi-sine perturbation inputs on multiple control surfaces simultaneously produce informative multi-axis data in a more efficient manner. The PTIs are designed within a bandwidth that includes the expected range of aircraft modal frequencies, and they are phase-optimized for minimum relative peak factor to provide perturbations that do not cause the aircraft to deviate far from the nominal flight condition [21, 30, 31].

In this work, a model is considered global if it is valid over the entire flight envelope for which modeling data have been obtained, and the modeling data include maneuvers at multiple flight conditions. For sufficient nonlinear global aerodynamic modeling, wide data coverage is needed across each of the relevant explanatory variables, with low correlation between them. For the flight data used in this work, the nominal flight condition was slowly varied throughout several maneuvers including figure eight patterns and approaches to stall, while the PTIs were overlaid on the pilot commands to excite the longitudinal and lateral-directional dynamics across a wide range of explanatory variables [3]. In particular, data partitioning requires data coverage not only globally, but also locally to approximate the aerodynamics in each cell. As will be discussed in Section IV.C, these maneuvers across the flight envelope need to be repeated as the model is being developed.

B. Parameter Estimation

Within each cell, a local aerodynamic model is estimated for each force and moment coefficient. These response variables are not measured directly in flight, but are calculated based on flight data using the following equations, which are defined in the aircraft body axes and retain the full nonlinear aircraft dynamics [21].

$$C_X = \frac{ma_x - T}{\bar{q}S} \tag{1}$$

$$C_Y = \frac{ma_y}{\bar{q}S} \tag{2}$$

$$C_Z = \frac{ma_z}{\bar{q}S} \tag{3}$$

$$C_l = \frac{I_x}{\bar{q}Sb}\left[\dot{p} - \frac{I_{xz}}{I_x}(pq + \dot{r}) + \frac{(I_z - I_y)}{I_x}qr\right] \tag{4}$$

$$C_m = \frac{I_y}{\bar{q}S\bar{c}}\left[\dot{q} + \frac{(I_x - I_z)}{I_y}pr + \frac{I_{xz}}{I_y}(p^2 - r^2)\right] \tag{5}$$

$$C_n = \frac{I_z}{\bar{q}Sb}\left[\dot{r} - \frac{I_{xz}}{I_z}(\dot{p} - qr) + \frac{(I_y - I_x)}{I_z}pq\right] \tag{6}$$

The force equations for C_X and C_Z can alternatively be expressed in the wind axes as

$$C_L = -C_Z \cos\alpha + C_X \sin\alpha \tag{7}$$

$$C_D = -C_X \cos\alpha - C_Z \sin\alpha \tag{8}$$

The estimated model is a polynomial expansion in terms of explanatory variables measured in flight, including air flow angles, body-axis angular rates, and control surface deflections. For example, the pitching moment coefficient can be expressed using linear longitudinal regressors as

$$C_m = C_{m_0} + C_{m_\alpha}\alpha + C_{m_q}\frac{q\bar{c}}{2V_0} + C_{m_{\delta_e}}\delta_e \tag{9}$$

With the emphasis on interpretability in this work, the estimated parameters can be considered as stability and control derivatives that describe the local aerodynamics in each cell, and if each cell has the same model structure, these influence coefficients can be compared across the global flight envelope.

The parameters for each model are estimated by minimizing the equation error in the least-squares sense, with this process described in detail in Ref. [21]. The regression equation can be expressed as

$$\boldsymbol{z} = \boldsymbol{X}\boldsymbol{\theta} + \boldsymbol{\nu} \tag{10}$$

where $\boldsymbol{z} \in \mathbb{R}^N$ is the response variable data to be modeled over N measurements, $\boldsymbol{X}$ is the $N \times n_x$ matrix of explanatory variables, $\boldsymbol{\theta} \in \mathbb{R}^{n_x}$ is the vector of model parameters, and $\boldsymbol{\nu} \in \mathbb{R}^N$ is the modeling error.

The cost function to solve for the best estimator $\hat{\boldsymbol{\theta}}$ that minimizes the squared differences between the measured response variable and the model output is

$$J(\boldsymbol{\theta}) = \frac{1}{2}(\boldsymbol{z} - \boldsymbol{X}\boldsymbol{\theta})^T(\boldsymbol{z} - \boldsymbol{X}\boldsymbol{\theta}) \tag{11}$$

The solution that provides the optimal result is

$$\hat{\boldsymbol{\theta}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{z} = \boldsymbol{D}\boldsymbol{X}^T\boldsymbol{z} = \boldsymbol{M}^{-1}\boldsymbol{X}^T\boldsymbol{z} \tag{12}$$

where $\boldsymbol{M} = \boldsymbol{X}^T\boldsymbol{X}$ is the Fisher information matrix which contains the information content in the regressor data, and $\boldsymbol{D} = \boldsymbol{M}^{-1}$ is the dispersion matrix.

The parameter covariance matrix is a measure of uncertainty in the parameter estimates, and if the residuals are assumed to be white, it is defined as

$$\boldsymbol{\Sigma}(\hat{\boldsymbol{\theta}}) = E\left[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T\right] = \hat{\sigma}^2(\boldsymbol{X}^T\boldsymbol{X})^{-1} \tag{13}$$

with the model fit error variance approximated as

$$\hat{\sigma}^2 = \frac{(\boldsymbol{z} - \hat{\boldsymbol{y}})^T(\boldsymbol{z} - \hat{\boldsymbol{y}})}{N - n_x} \tag{14}$$

While the formulation defined above can be used to estimate a model with batch data, $\hat{\boldsymbol{\theta}}$ can also be updated recursively for real-time parameter estimation. If the model is defined at the $(i-1)^{th}$ data point, then as each new measurement $z(i)$ and $\boldsymbol{x}(i)$ is received, the parameter estimates can be updated through Eqs. (15 –17) [21].

$$K(i) = \frac{\boldsymbol{D}(i-1)\boldsymbol{x}(i)}{1 + \boldsymbol{x}^T(i)\boldsymbol{D}(i-1)\boldsymbol{x}(i)} \tag{15}$$

$$\boldsymbol{D}(i) = [\mathbb{I} - \boldsymbol{K}(i)\boldsymbol{x}^T(i)]\boldsymbol{D}(i-1) \tag{16}$$

$$\hat{\boldsymbol{\theta}}(i) = \hat{\boldsymbol{\theta}}(i-1) + \boldsymbol{K}(i)[\boldsymbol{z}(i) - \boldsymbol{x}^T(i)\hat{\boldsymbol{\theta}}(i-1)] \tag{17}$$

The parameter covariance is updated with a recursive estimate of the fit error variance using Eqs. (18 –19).

$$\hat{\sigma}^2(i) = \left(\frac{i-1}{i}\right)\hat{\sigma}^2(i-1) + \frac{1}{i}\nu^2(i) \tag{18}$$

$$\boldsymbol{\Sigma}[\hat{\boldsymbol{\theta}}(i)] = \hat{\sigma}^2(i)\boldsymbol{D}(i) \tag{19}$$

After the modeling process has been completed, the effectiveness of the resulting model can be described by several modeling metrics. In particular, the coefficient of determination(R^2), defined in Eq. (20), is a model fit quality measure that varies from 0 to 1 and describes how much of the variation in the data about the mean value is captured by the model.

$$R^2 = 1 - \frac{\sum_{i=1}^{N}[z(i) - \hat{y}(i)]^2}{\sum_{i=1}^{N}[z(i) - \bar{z}]^2} \tag{20}$$

The equation-error optimization enables recursive parameter updates as data are obtained in real time, and it is performed in the time domain to allow points that are not contiguous in time to be incorporated into the associated models in each cell. For example, Fig. 4 shows F-16 simulation data that are divided into three cells across the complete range of α, where the data associated with the α range in each cell are incorporated into the respective local models of low-, mid-, and high-α aerodynamics.

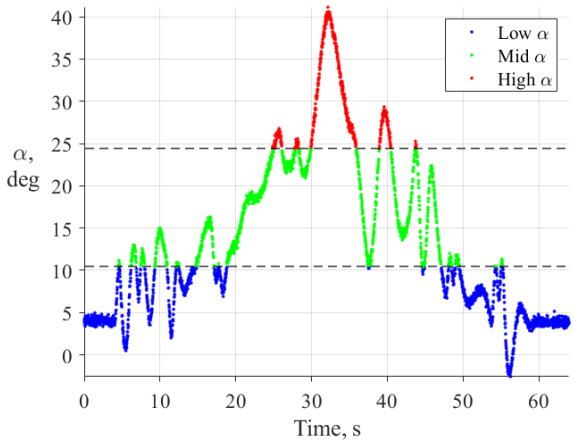


Fig. 4 F-16 simulation time history data with α partitioning.

The parameter estimation can be performed using either a global or local optimization process. Prior work employs global learning where the parameters across all of the cells are optimized simultaneously in a single global weighted regression problem [18]. This centralized approach can lead to an ill-conditioned regression problem and interdependent parameter estimates with high uncertainties that cannot be interpreted individually [24]. Other work utilizes local learning by posing a separate weighted regression problem for each local model [26]. However, the individual cell weights are a normalized function of the weights in all of the cells, as will be discussed in Section IV.D. So when a new measurement or a new cell is added, the estimates from all other cells would change, meaning the parameter estimates from all cells are still interdependent. In this work, the local parameter estimation problem is completely decoupled from the global modeling by first estimating the parameters in each cell, and then overlaying the validity functions afterwards. This approach retains the interpretability of the model in each cell, and allows new measurements to simply be incorporated into the model to which they belong, irrespective of the other cells.

C. Cell Structure Determination

The cell structure is developed through a decision-making process that is used to successively partition the nonlinear flight data into relevant local regions, with the objective of improving global model accuracy by accounting for nonlinearities, while maintaining a parsimonious model with as few cells as possible.

i. Problem Statement

Section III.C discussed several works, and offline, iterative, binary axis-orthogonal algorithms, that successively partition a system's aerodynamics through specified PVs. This research similarly constrains the splits to be axis-orthogonal to increase interpretability of the cell structure. Binary splits, however, can be highly restricting, in particular if the nonlinearities are biased toward one side of the PV input space as is typical for aerodynamics, and which can result in redundant cells and overly complex models. The SPLITR approach more generally allows variable breakpoints that are informed by residual analysis, and forgoes the advantages of these iterative methods for real-time compatibility. Accordingly, the cell structure determination problem can be summarized by posing the following questions.

Is a split needed? Given the current cell structure, the model needs to be evaluated to determine if it is sufficient, or if the data in one or more cells are not characterized well by the local model.

When to split? How much data or statistical evidence is needed to split the cell? If new cells are added too rapidly, it can result in an overly complex model with redundant cells, whereas if splits are performed too cautiously, then the model is degraded in real time, and a real-time model-based control law may suffer as a result.

Which cell to split? Which local model is performing poorly? Is it due to noisy data, or is it due to a perceived nonlinearity that this local model is not sufficiently capturing? The approach needs to keep track of estimates of the prevalence of noise in the data, as well as model fit statistics.

Which dimension to split? Once a particular cell has been identified as having a poor model due to nonlinear or unmodeled dynamics, then allowable axis-orthogonal split dimensions include those identified as PVs. If the relevant sources of nonlinearity are known in advance, this pool can be more carefully specified.

Where to split along a dimension? Herein lies the challenge of determining where within an individual cell the next breakpoint should be placed.

ii. Preliminary Aspects

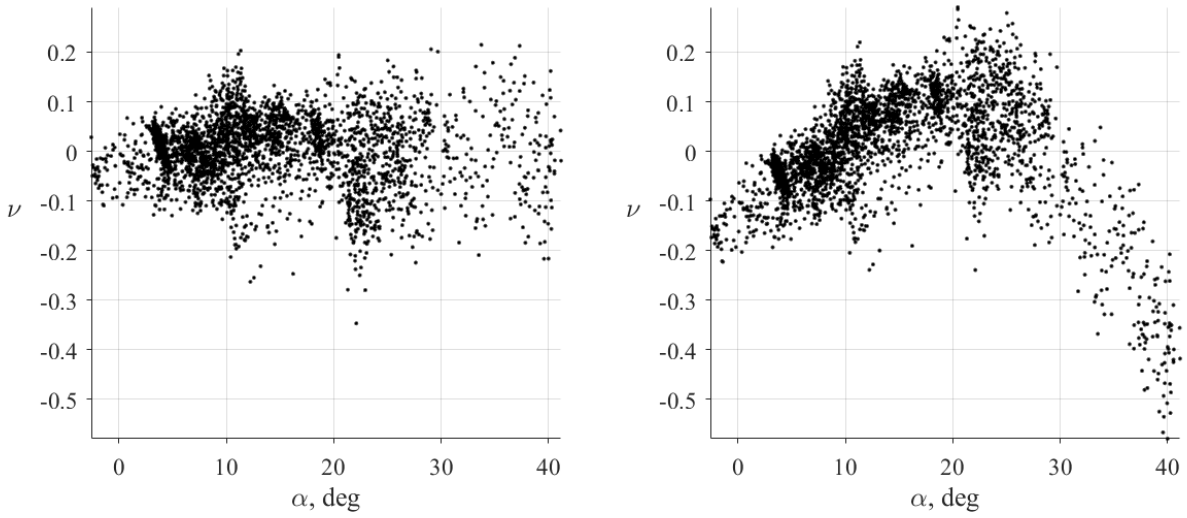
Consistent among past work is that the iterative, offline cell structure determination process is driven by improving a global model fit performance metric. During each iteration, a binary axis-orthogonal split is typically proposed in all existing cells with all possible dimensions, and the entire global model is recomputed for each case, resulting in a series of proposed models that all have one more cell than the previous iteration. The proposed cell structure that improves the global performance metric the most is chosen for the next model iteration [25–29]. With the advantage of being offline, this iterative approach has access to all of the modeling data at once, and estimates a new set of parameters for each proposed model. This is not the case for a real-time approach, where all past data cannot be practically stored and used to recompute parameter estimates after a new cell is added. Therefore, a new real-time decision-making procedure for cell structure determination is necessary.

A challenge with the offline approaches, and one that is apparent in any system identification problem, is choosing an ideal metric that adequately characterizes the model fit quality, and serves as a reliable stopping criterion to preclude excessive splits and ensure a parsimonious model. Common model fit statistics such as the coefficient of determination (R^2), Akaike information criterion (AIC), etc., or a combination thereof, are used to characterize the model fit during the model development process for these approaches. While they can be useful tools for model assessment, they are global measures of the fit and can provide misleading information about the model fit quality if not interpreted correctly. Furthermore, specifying a satisfactory threshold can be arbitrary across different data sets with varied data quality and noise levels.

The SPLITR approach to cell structure determination, similar to that of parameter estimation, examines the performance of the local models instead of the global model fit. This localized approach infers that if the local models perform well, then the weighted combination that comprises the global model will provide a good fit as well. Instead of considering model fit statistics, SPLITR examines the essence of the information contained in these metrics, which is the residuals, i.e. the differences between the measured data and the estimated model output, as shown in Eq. (21) for the i^{th} measurement.

$$\nu(i) = z(i) - \hat{y}(i) \tag{21}$$

The residuals represent samples of the modeling error, and they can be used to assess model adequacy and reveal model deficiencies through a residual characterization procedure, which will be described in the data processing sequence next. Furthermore, the individual residuals are discrete samples that describe the errors locally, thus providing information regarding the magnitude of the model error as well as the location within the ranges of measurements. Ideally, the residuals should be unbiased, have constant variance, and be mutually uncorrelated. In flight data, the residual variance is often correlated with certain flight variables, but should still not display deterministic character. Figure 5a shows the residuals as a function of α from the SPLITR model using the F-16 data set. Notice that the variance tends to increase at higher α, which is consistent with the larger presence of noise in that region. Figure 5b, on the other hand, shows residuals with a deterministic character that indicates poorly modeled aerodynamics, and which results from a model with only a single linear cell for the same F-16 data. Ideally, the whiteness of the residuals in each cell would be evaluated through the autocorrelation or cross-correlation functions, but since the data are not necessarily contiguous in time, this method is not feasible, and another approach must be formulated.



(a) Random content (SPLITR model) **(b) Deterministic content (linear model)**

Fig. 5 Residuals for F-16 models of $C_L = f(\alpha)$.

The SPLITR process can operate under two separate update rates, with the process overview shown in Fig. 6, and a detailed depiction of the data processing and cell splitting procedures shown in Fig. 7. At a faster rate, each new measurement is processed immediately so that all useful information it contains is extracted, and the associated cell's model parameters are recursively updated. Since the cell structure isn't expected to change rapidly, the cell splitting procedure operates at a slower rate, which is chosen to be outside of the expected range of rigid-body dynamics for the aircraft. These update rates can be modified for other aircraft and data rates.

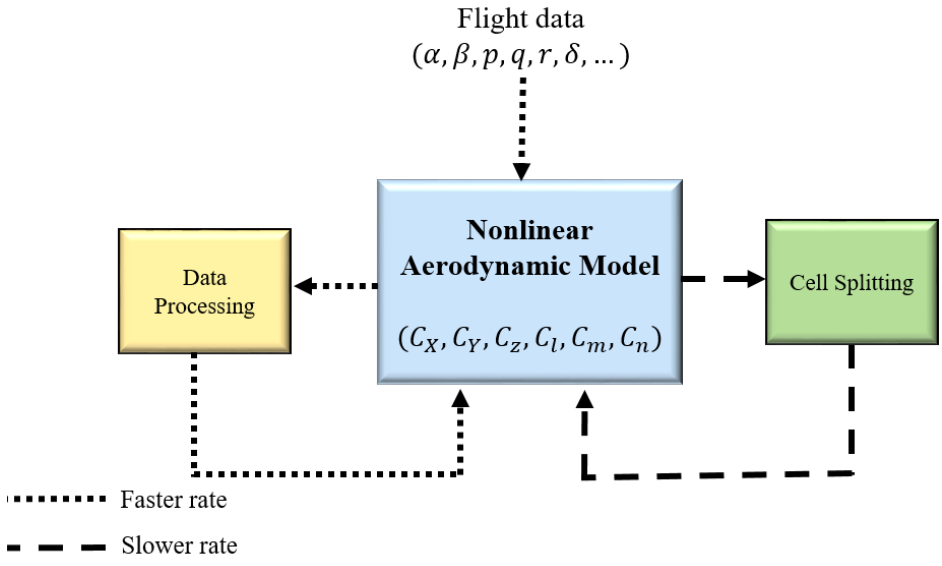


Fig. 6 SPLITR process overview.

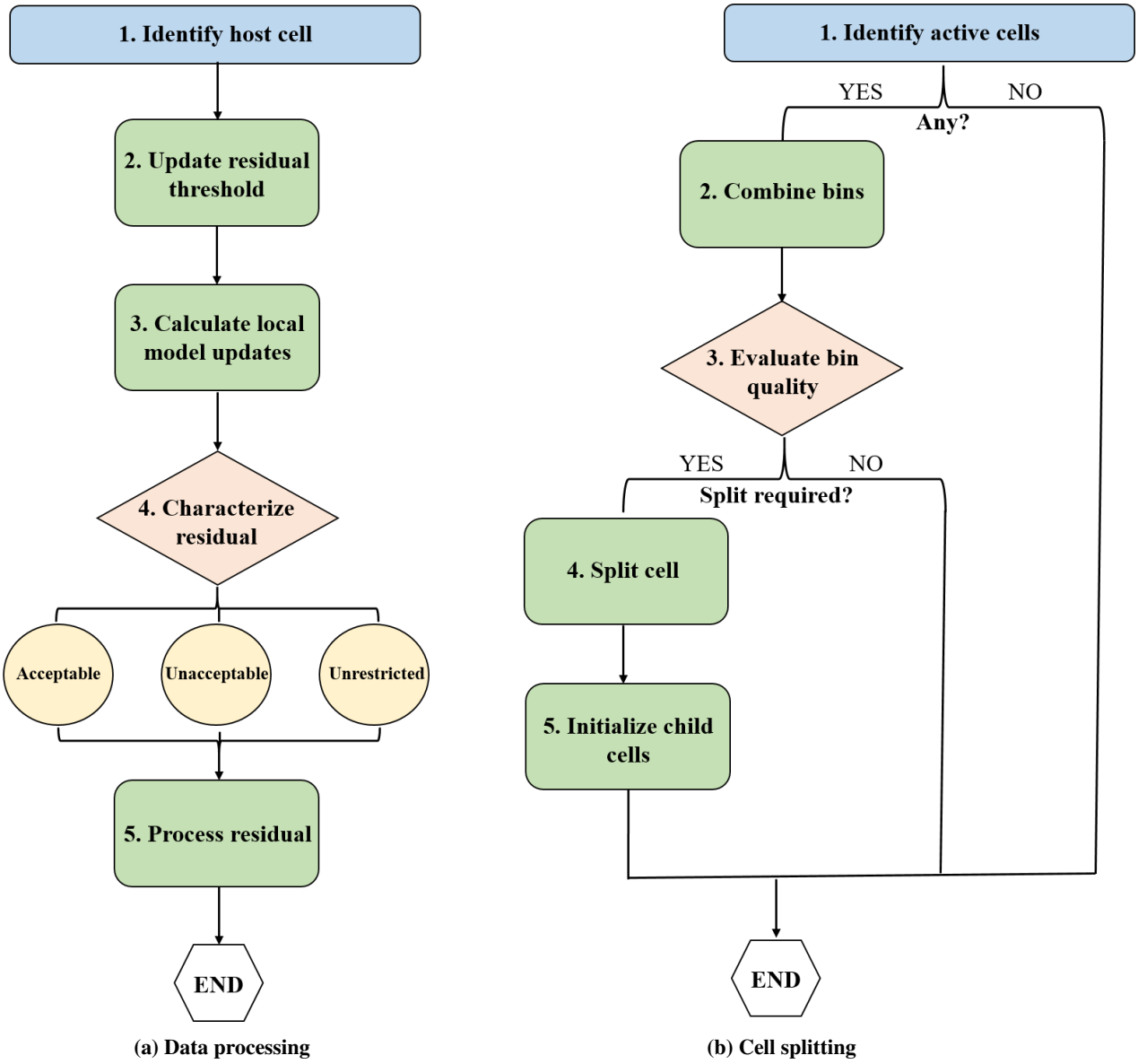


Fig. 7 SPLITR model update procedures.

iii. Data Processing Procedure

The data processing sequence is shown in Fig. 7a, and each step will be summarized below. The i^{th} measurement supplied to the model consists of the response variable $z(i)$, which is the force or moment coefficient, the associated explanatory variables $\boldsymbol{x}(i)$, and the specified partitioning variables $\boldsymbol{\phi}(i)$.

1) *Identify host cell:* For a given cell structure, each response variable measurement is parametrized by its associated PVs and belongs to a single cell. Note that if there are n_ϕ possible split dimensions, then the coordinates of the cell boundaries are $\in \mathbb{R}^{n_\phi \times n_\phi}$.

2) *Update residual threshold:* Residual characterization is based upon distinguishing between random and deterministic properties in the residuals, the latter of which would be attributed to unmodeled dynamics, or nonlinearities in the data that are not captured by the local model. It is important not to mistake a poor model fit as a justification for splitting if it is due to noisy modeling data. Therefore, an acceptable threshold is needed to characterize the residuals such that those that lie within the bound represent reasonable model error that includes

noise influences, and those that lie outside represent deterministic model deficiency. To compute the threshold, each force and moment coefficient to be modeled is passed through a real-time discrete high-pass (HP) filter to obtain an estimate of the noise. A running root-mean-square (RMS) calculation of the noise estimate is computed for the response variable data in each cell, and it is multiplied by a factor λ_ν to account for statistical variations and to describe an acceptable residual threshold that is outside the noise level. These noise estimates are binned across the range of each PV, as will be detailed later in this section, so different cells across the range of PVs would be expected to have varied thresholds based on the noise levels in the data they contain.

3) *Calculate local model updates:* The parameter estimation scheme discussed in Section IV.B is used to recursively calculate the updated local cell parameters with the information contained in each new measurement. The local model output for the i^{th} measurement incorporated into the k^{th} cell is then given as

$$\hat{y}_k(i) = \boldsymbol{x}(i)\hat{\boldsymbol{\theta}}_k(i) \tag{22}$$

and the associated local residual for that point is calculated as

$$\nu_k(i) = z(i) - \hat{y}_k(i) \tag{23}$$

4) *Characterize residual:* Using the calculated residual threshold, the absolute value of each residual is characterized in one of three ways. An unrestricted residual is from a measurement that is unconditionally allowed into the model. This can be at the beginning of the modeling process as it waits to receive enough information for a reliable initial model, or, as will be discussed later in this section, immediately following a split so the child cell models can adjust. A measurement associated with an unrestricted residual is consequently not regarded for splitting purposes. An acceptable residual is one that is not considered an unrestricted residual, that is below the calculated residual threshold for that cell, and is regarded as supported by the local model and/or influenced by noise. An unacceptable residual is one that is also not characterized as unrestricted, that is outside the residual threshold, and is therefore considered an indication of a poor model fit in the associated cell.

5) *Process residual:* To provide discretized error information within each cell that will inform a possible split location, the absolute values of the residuals are binned across the range of each PV based on a prescribed resolution that is the minimum acceptable width of a cell. The bins are designated at the beginning of the modeling process, and as the cell structure is modified, the cell to which they belong changes accordingly with the cell boundaries. Determining the split location is thereby reduced from a continuous problem across the range of a cell to a discrete choice along one of the bin boundaries.

In real-time operation, the residuals in each bin must be processed in a way that their influence toward a possible split can be retained and recursively updated. The properties of the absolute values of the residuals in each bin are therefore represented by the mean and variance, which are given for N data points as

$$\mu_N = \frac{|\nu_1| + |\nu_2| + \ldots + |\nu_N|}{N} = \frac{\sum_{i=1}^{N} |\nu_i|}{N}, \quad \sigma_N^2 = \frac{\sum_{i=1}^{N} (|\nu_i| - \mu_N)^2}{N} \tag{24}$$

As new measurements are received, the mean and variance calculations of the associated bins are recursively updated as

$$\mu_{N+1} = \frac{\mu_N N + |\nu_{N+1}|}{N+1}, \quad \sigma_{N+1}^2 = \frac{\sigma_N^2 N}{N+1} + \frac{(|\nu_{N+1}| - \mu_{N+1})^2}{N} \tag{25}$$

Each bin contains a count of the acceptable and unacceptable residuals, and a set of four recursively updated associated statistics: the mean and standard deviation of the acceptable residuals, μ^A, σ^A, and those of both the acceptable and unacceptable residuals, μ^B, σ^B. Note that the unrestricted residuals are not incorporated into the bin statistics.

Since the parameter estimates in each cell are changing over time as new data are obtained, the real-time residuals that are calculated and processed here are obsolete as soon as the parameters are updated. Theoretically, if all of the past data were saved then the residuals could be recomputed each time the parameters are updated. However, this is considered a tolerable artifact of the constraints of real-time operation.

If the residual is characterized as unrestricted or acceptable, the parameter updates computed in Step 3 above are accepted, and the local model is updated accordingly. If the residual is considered unacceptable, then the local model is returned to its prior state, and the associated measurement is stored. Note that although some data is subsequently stored and later accessed, this amount of data is minimal compared to the total amount of flight data, and is reasonable for onboard computation. Alternatively, this data storage step could be skipped, which would effectively discard the measurements associated with the unacceptable residuals and rely solely on future data, as discussed of Step 5 in the cell splitting sequence below.

iv. Cell Splitting Procedure

The cell splitting procedure is shown in Fig. 7b, and is described next. During this sequence, the residuals are queried to determine if the cell structure is adequate, or if one or more cells should be split.

1) *Identify active cells:* Since the cell structure determination is directly associated with residuals characterized as unacceptable, only those cells that have received measurements whose residuals are characterized accordingly since the last cell structure check need to be examined. Additionally, only the range of active bins within the cell, or the range of bins that actually contains data, is analyzed.

2) *Combine bins:* If a user-defined minimum cell width is relatively small across the range of a particular PV, this resolution may not be necessary if data are obtained quickly across a wide range, and instead larger cell widths should be considered first. Furthermore, with such a fine initial resolution, it might take a while for each bin to collect enough information to produce useful, informative bin statistics. Therefore, a maximum number of bins across each cell is also pre-specified, so that early on in the cell division process only larger splits will occur, and later on, the finer resolution can be considered. Accordingly, the bin resolution cannot be changed in real time since the associated past residuals are not accessible in order to compute new bin statistics. Therefore, from the beginning the statistics are computed in the minimum resolution bins, and then those statistics from adjacent bins are combined during the cell structure checks up to the maximum number of bins. Equation (26) shows how the means and variances are combined for two bins using their individual values as well as the number of data points in each, designated as N_1 and N_2.

$$\mu_{N_1+N_2} = \frac{\mu_{N_1}N_1 + \mu_{N_2}N_2}{N_1+N_2}, \quad \sigma^2_{N_1+N_2} = \frac{\sigma^2_{N_1}N_1 + \sigma^2_{N_2}N_2}{N_1+N_2} + \frac{\mu^2_{N_1}N_1 + \mu^2_{N_2}N_2 - (N_1+N_2)\mu^2_{N_1+N_2}}{N_1+N_2} \tag{26}$$

3) *Evaluate bin quality:* A pass/fail status of each (combined) bin is then determined to describe if the bin contains an acceptable amount of error, or if the indication of deterministic error is strong. If the mean of both the acceptable and unacceptable residuals is drawn beyond a specified number λ_σ of standard deviations from the mean of the acceptable residuals, as shown in Eq. (27), then the bin is considered failed.

$$\mu^B > \mu^A + \lambda_\sigma \sigma^A \tag{27}$$

A bin may contain both acceptable and unacceptable residuals, so Eq. (27) ensures not only that there are unacceptable residuals in the bin, but that their magnitude sufficiently outweighs the acceptable residuals found there. Furthermore, for a bin to fail the status check, it must contain a minimum amount of residuals to ensure that sufficient information is obtained. If a bin fails the status check, the normalized severity of the unacceptable residuals is computed as in Eq. (28) to represent how many standard deviations the mean of both types of residuals is drawn away from that of the acceptable residuals, and is also saturated at 1.

$$binseverity = \frac{\mu^B - \mu^A}{\sigma^A \lambda_{\sigma,norm}} \tag{28}$$

The bin severities from adjacent failed bins are summed to obtain a group severity, which is then compared to a prescribed maximum total severity. If it surpasses the limit, then that bin group is a candidate to inform the next cell split. Within a single cell, the group of bins with the largest total severity beyond the limit is chosen to inform the split. The above analysis can be performed in parallel across multiple PVs to choose both the split dimension and location along that dimension in a single cell through the same decision process. However, note that the discretized binning and calculations will need to be performed n_ϕ times, so it is advisable to limit the number of partitioning options within the understanding of the aerodynamics and possible sources of nonlinearity.

4) *Split cell:* If a split has been justified, then using the selected set of bins, two split locations are considered, i.e. to split at the right boundary of the right-most bin or at the left boundary of the left-most bin. The chosen split location is that which partitions the failed bins toward the side with the fewest additional bins, i.e. with the smallest distance to the edge of the bin range. The parent cell is then divided into two child cells, with one on each side of the specified split location. The bin counts and statistics discussed in Step 5 of the data processing sequence are zeroed for the child cells.

5) *Initialize child cells:* When a parent cell is divided into two child cells, the presumption is that the parent model was inadequate at least across a certain range of the data it included. However, its parameters were estimated using data across the ranges of both child cells, so when a split is performed, the parameter estimates associated with the parent model no longer equally represent both sides of the split. Four possible approaches to initializing the child cell models immediately following a split are discussed.

   First, ideally if all past modeling data are stored and accessible, then when a cell is split, the parameter estimates for the child cells can be recomputed in batch with all of the past data that belong in each cell. This approach, however, is not practical in real time.

   Second, all possible configurations of cell structures can be posed a priori, and as the data are received in real time, all possible cell models will be recursively updated, while only the models in the current cell structure will be activated. Note that in this approach, each measurement can belong to more than one cell. Then when a split is performed, the respective latent models will be activated. While this approach would also provide a model for the exact data in each cell, this would only be practical with a very coarse resolution of possible splits and few split dimensions. If there are multiple split dimensions and a fine minimum resolution, the curse of dimensionality implies not only a large number of minimum resolution cells, but also an astronomical number of intermediate cell structures as the model is successively partitioned. If there is prior knowledge of the aircraft aerodynamics that can inform a reasonable number of possible cell structures, then this approach may be more practical.

   Third, an approach that would be feasible in real time is to zero the model in each of the child cells and begin from scratch to estimate a new model with the future data obtained in the cell. Again, this would provide a model with the correct data; however, it is inexpedient to effectively discard all of the past data collected in the region of the parent cell, particularly because much of it could still be relevant to the child models.

   Finally, a practical approach for real-time model initialization accepts the fact that the parent model was built on data contained in both child cells, and despite that, each child cell's model parameters are initialized with those of the parent cell. Since the information matrix cannot be recomputed in real time with past data, the child cell is initialized with a percentage of the parent cell's information matrix, $\boldsymbol{M} = \boldsymbol{X}^T \boldsymbol{X}$, that is proportional to the number of past measurements in the parent cell that are within its range. These measurement counts are binned similar to the residuals and are summed across the bins that belong to each child cell. Note that the inherent assumption behind this approach is that each measurement contributes equally to the information matrix in terms of information content, which is not necessarily true, but which provides a good starting guess within the real-time constraints. The dispersion matrix, $\boldsymbol{D} = \boldsymbol{M}^{-1}$, is then used to continue the recursive parameter updates for the child cell, as in Eqs. (15 –17). Lastly, the stored measurements associated with the unacceptable residuals from each side of the split are incorporated into the respective child models so that they are adjusted instantaneously to include the data that were deemed not fit well by the parent model. Following the child cell initialization, a specified number of new unrestricted measurements are also automatically allowed into the model to allow it to adjust to the new region it is defined over, before residuals are further characterized. The child cell's residual threshold can be initialized as well based on the RMS of the noise estimates of the response variable data recorded in the bins belonging to the cell.

D. Global Model

Thus far, the SPLITR parameter estimation and cell structure determination processes within the LMN architecture have been decoupled from the model weighting process, and instead have focused on the local cells and preserving their accuracy and interpretability. The global nonlinear model, which is the output of the LMN, is then a weighted combination of the piecewise local models superimposed with the validity functions.

Validity functions are used across a variety of disciplines under different names such as basis functions in basis function networks, activation functions in neural networks, and membership functions in fuzzy logic. The common operational utility is to provide a smooth transition between piecewise components in a global architecture, and to characterize the regions of validity of each local model by weighting their contribution to the global output across the

ranges of relevant PVs. This work utilizes Gaussian validity functions which consist of two parameters: the center, which is located where the associated model has the strongest validity; and the standard deviation, which characterizes the influence of each local cell across the range of PVs.

The validity function for the k^{th} local model is given as

$$\hat{w}_k(\boldsymbol{\phi}) = exp\left[-\frac{1}{2}\sum_{j=1}^{n_\phi}\frac{(\phi_j - c_{k,j})^2}{\sigma_{k,j}^2}\right] \tag{29}$$

where c_k is the center of the Gaussian, and σ_k is the standard deviation. In this work, the Gaussian functions were placed in the center of the cells, and the standard deviation was a function of cell width such that $\sigma_{k,j} = \frac{2}{5}\lambda_s(b_{k,j} - a_{k,j})$, where λ_s is the smoothness factor that describes the influence of each cell, and $[a_{k,j}, b_{k,j}]$ define the range of the k^{th} local cell across the j^{th} PV. The scaling factor of $\frac{2}{5}$ is an empirical result from Ref. [26], such that a nominal smoothness factor of $\lambda_s = 1$ can be used.

The validity functions across all cells are then normalized to ensure a partition of unity, such that their sum across the PV input space is equal to 1.

$$w_k = \frac{\hat{w}_k(\boldsymbol{\phi})}{\sum_{p=1}^{M}\hat{w}_p(\boldsymbol{\phi})} \tag{30}$$

Finally, the global model output over N measurements and M cells for the i^{th} point is given as

$$\hat{y}(i) = \sum_{k=1}^{M}\hat{y}_k(i)w_k(i) = \sum_{k=1}^{M}\boldsymbol{x}(i)\hat{\boldsymbol{\theta}}_k(i)w_k(i) \tag{31}$$

Note that since the validity function is not directly incorporated into the regression and is simply overlaid on the local models, the resulting weighted nonlinear model is no longer optimal in the least-squares sense. Regardless, the global model fit results were insensitive to small variations in the Gaussian standard deviations. The role of the validity functions is primarily to ensure smooth transitions over the cell boundaries without impacting the parameter estimates in the individual cells.

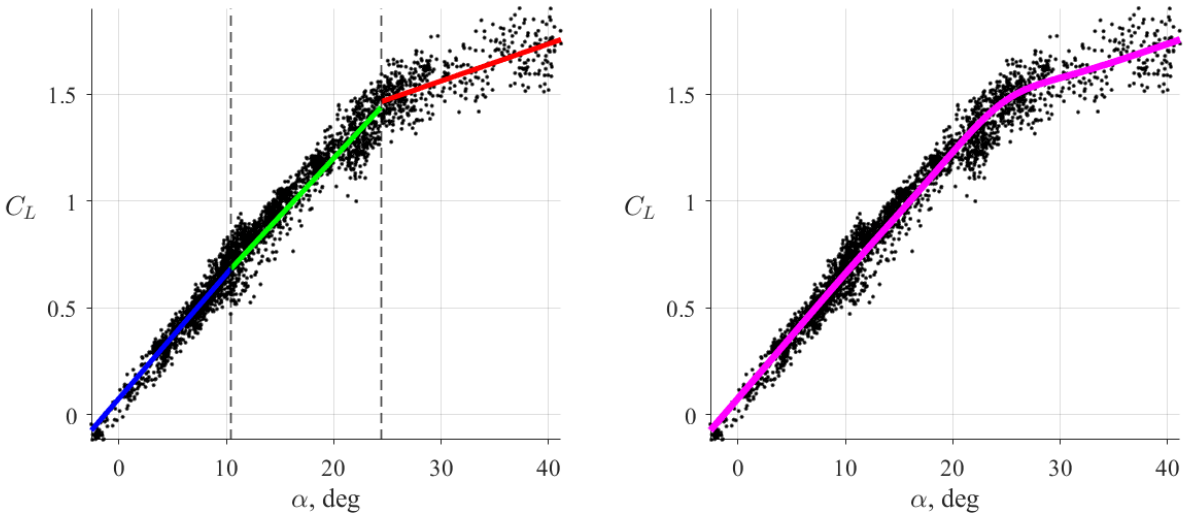
V. F-16 Simulation Results

This section demonstrates the SPLITR modeling approach through a simplified example using data from an F-16 simulation [21]. Simulated flight data were acquired from within SIDPAC, and the nominal geometry and mass properties of the F-16 aircraft used in the nonlinear simulation are summarized in Table 1. The flight data consisted of a maneuver that began at a trimmed angle of attack of 4 deg at 25,000 ft. The nominal elevator input was slowly varied along with simultaneous manual perturbations intended to excite the longitudinal dynamics across a wide range of α, as shown in Fig. 4. Two-percent Gaussian white noise was added to the data such that the noise content increases with α, in a way that is similar to what would be seen in flight.

Using this simulation data, a single linear model was estimated for $C_L = f(\alpha)$, and the resulting residuals in Fig. 5b clearly show that deterministic modeling information is not captured, which is the nonlinear behavior of C_L at high α. When the SPLITR approach was used to estimate a model, α was divided into three cells, as shown in Fig. 4, and the corresponding residuals in Fig. 5a are improved. The associated linear piecewise models shown in Fig. 8a represent the modeling data in each cell well, and the weighted global nonlinear model is shown in Fig. 8b, with $R^2 = 0.98$. This simplified one-dimensional example demonstrates how a nonlinear function can be successfully modeled with several piecewise linear functions using the SPLITR approach, and that the resulting model captures the local and global aerodynamics well. However, aerodynamic forces and moments are generally functions of multiple explanatory variables, and they are more difficult to estimate from flight data that contain unknown errors, including noise and other disturbances. This F-16 example provides a useful visualization of the model fits, while the next section will apply the SPLITR approach more generally to flight data obtained from the E1 aircraft.

Table 1 F-16 geometry and mass properties.

Symbol	Value	Unit
$\bar{c}$	11.32	ft
b	30	ft
S	300	ft^2
m	647.2	slug
I_x	9,496	slug-ft^2
I_y	55,814	slug-ft^2
I_z	63,100	slug-ft^2
I_{xz}	982	slug-ft^2



(a) Local models **(b) Global nonlinear model**

Fig. 8 SPLITR model fits of $C_L = f(\alpha)$ for F-16 data.

VI. E1 Flight Test Results

First, the E1 test aircraft will be described, followed by a discussion on the practical aspects of applying the SPLITR method to E1 flight data. Then the SPLITR results will be shown through an in-depth case study that describes the model development and results for C_m, followed by a further discussion on other findings.

A. E1 Aircraft

The test vehicle used in this work was a 40% scale Extra 330SC radio-controlled battery-powered aircraft, designated as E1 and shown in Fig. 9. The cruise speed for E1 was approximately 50 kts, and the nominal altitude throughout the flights used in these results was 600 ft. The control surfaces include ailerons, flaps, elevator, and rudder. The geometry and mass properties for the flight vehicle are summarized in Table 2, and were determined through ground-based tests. The aircraft was instrumented with an inertial navigation system (INS) that provided 3-axis translational accelerations, angular rates, Euler angles, GPS position, and velocity. An air flow angle vane mounted on a boom on the right wingtip measured the angle of attack and sideslip angle, a pitot tube measured static and dynamic pressures, encoders measured each of the control surface deflections, and a Hall-effect sensor measured the propeller speed. These measurements were used to calculate the aerodynamic forces and moments given in Eqs. (1 –6), as well as the explanatory variables used for modeling. In particular, the calculation for advance ratio is given as

$$J = \frac{V}{\Omega d} \tag{32}$$

where V is the airspeed, Ω is the engine speed, and d is the propeller diameter. The flight computer recorded data at 50 Hz, and during the modeling portion of the flights, was used to automatically inject the PTIs onto each of the control surfaces by summing these automated inputs with the pilot commands. Additional information about the E1 test vehicle can be found in Refs. [1, 32].



Fig. 9 E1 test aircraft.

Table 2 E1 geometry and mass properties.

Symbol	Value	Unit
$\bar{c}$	1.97	ft
b	10.17	ft
S	19.26	ft^2
m	1.910	slug
I_x	2.964	slug-ft^2
I_y	8.776	slug-ft^2
I_z	11.716	slug-ft^2
I_{xz}	0.750	slug-ft^2

B. Practical Aspects

The SPLITR algorithm is intended to be an automated modeling method that can be applied to various aircraft, so this section will summarize the current algorithm parameters and other specifications that still must be user-prescribed. Table 3 summarizes the relevant tuning parameters, all of which were discussed in detail in Section IV, and lists the values used for these results. In general, the modeling process was found to be insensitive to small variations in all of these parameters across different sets of flight data. Many terms, such as the smoothness factor, were chosen empirically and do not strongly impact the results. Others, such as the maximum number of allowable cells and minimum cell resolution, are user-specified to limit the model complexity. In these results, the maximum number of cells was not limited to demonstrate the effective model complexity management. The most impactful parameter was found to be the residual threshold factor λ_ν which is related to the cutoff between acceptable and unacceptable residuals, and influences the split determination. A value too large will prevent any splits from occurring, and a value too small will result in an overly complex model. In general, a factor of 5-10 has been shown as a conservative statistical approach to describe an acceptable range of statistical data variations [4]. Recall that λ_ν is used to multiply the noise estimate of the response variable data, so a factor of 5 effectively places the residual threshold at 5σ. Since the aircraft rigid-body dynamics for this scaled vehicle were estimated to be less than 2 Hz, a fourth-order HP Butterworth filter was used with a cutoff frequency of 3 Hz to estimate the noise [4].

Table 3 SPLITR algorithm parameters.

Model Complexity	
Maximum # of cell bins	10
Minimum cell resolution (α, β)	0.5 deg
Maximum # of allowable cells	Inf
Residual Threshold	
HP filter order	4
HP filter cutoff frequency	3 Hz
Residual threshold factor, λ_ν	5
Bin Status	
Standard deviation factor, λ_σ	0.75
# of standard deviations to normalize, $\lambda_{\sigma,norm}$	1
Total bin severity threshold	2
# of minimum points in bin for failure	20
Unrestricted Residuals	
# of unrestricted points before initial model	250
# of unrestricted points following split	150
Global Model	
Smoothness factor, λ_s	1
Update Rate	
Cell splitting update rate	5 Hz

Other information that must be defined in advance includes the aircraft mass, geometry, and inertia properties to normalize the forces and moments. For each model, the regressors, available PVs, and expected ranges of PVs must also be specified.

In this work, complex nonlinear behavior in the response variables is addressed by partitioning the flight data to capture the simpler dynamics locally. However, this approach can be viewed more generally such that any deterministic behavior in the residuals as a function of the PVs outside of the residual threshold will be treated the same by triggering a split. The regressors are specified in advance, separate from the cell structure determination. So if an inadequate regressor pool is chosen, either one that contains irrelevant or strongly correlated regressors, or one that is missing important terms, then the algorithm may try to compensate for the insufficient modeling information by splitting unnecessarily. These successive splits may therefore not actually improve the model fit.

To simplify the problem in these results, only linear regressors were chosen based on the flight data, which will result in a LMN consisting of local, linear models. For example, the pitching moment coefficient was prescribed the model structure of

$$C_m = C_{m_0} + C_{m_\alpha}\alpha + C_{m_q}\frac{q\bar{c}}{2V_0} + C_{m_{\delta_e}}\delta_e + C_{m_J}J \tag{33}$$

Higher-order and multivariate terms can be included as well in each cell, which would yield a network of local nonlinear models. This would likely reduce the model complexity by resulting in fewer cells, but would also pose identifiability problems for each cell, unless local model structure determination techniques were applied.

As a further simplification for demonstration and visualization, the PVs were specified as α and β for the longitudinal and lateral-directional coefficients, respectively, since the main nonlinear dependencies of the forces and moments are on these variables. Since the residual binning is performed in parallel across the range of each PV, it is computationally more efficient to minimize the number of PVs. Also, fewer split dimensions will yield a more transparent model.

In these results, the modeling process was not performed onboard the aircraft, but rather using data obtained from the E1 flights. Since the E1 flight computer records data at 50 Hz, the data processing sequence operated at 50 Hz as well. The cell splitting procedure ran at 5 Hz, which was outside of the expected range of rigid body dynamics for this test aircraft. The flight data used in these results were obtained from several flights during which the pilot performed figure eight and stall maneuvers, which are discussed in Section IV.A. The pilot inputs were overlaid with PTIs, and the goal was to obtain global data across a wide range of flight variables, with Fig. 10 showing the resulting crossplot of α and β. The throttle position was also varied with random step inputs throughout the flights, and in particular, power was

added during some of the stall maneuvers.

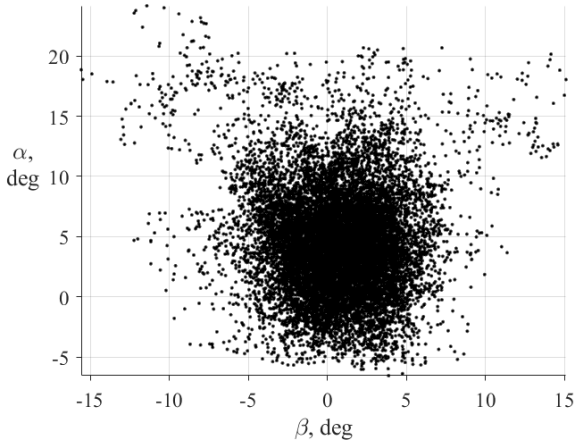


Fig. 10 Crossplot of α and β modeling data.

C. Case study of C_m model

Figure 11 shows the time history of the response variable C_m and the corresponding PV α over a period of flight test data used for modeling. The time at which each split occurs is indicated by a vertical dashed line in each plot. Note that the nonlinear dependency of C_m on α is evident as the splits occur during the high-α maneuvers as the flight envelope is expanded and new local models are required. In flight test data, particularly when the PTIs are active across all axes simultaneously, there are variations in all of the flight variables at once. Visually discerning where the nonlinear breakpoints may be placed as a function of the known PVs can therefore be unclear, as shown by the plot of C_m vs. α in Fig. 12. This is in contrast to the simplified example using the F-16 data, and implies that the algorithm logic must be relied upon to determine the appropriate breakpoints.

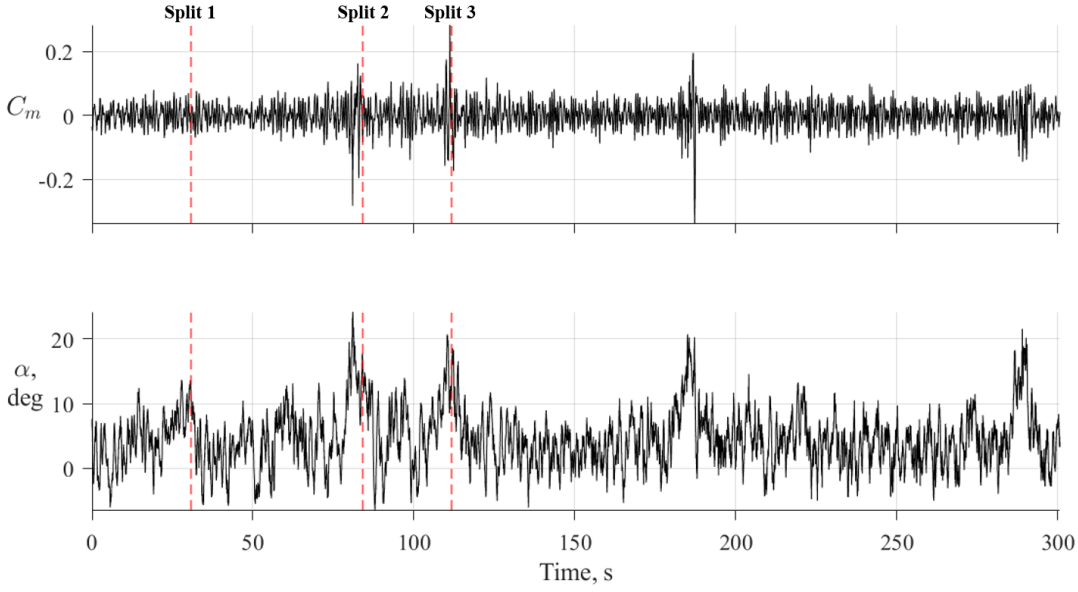


Fig. 11 Time histories of response variable and PV modeling data.

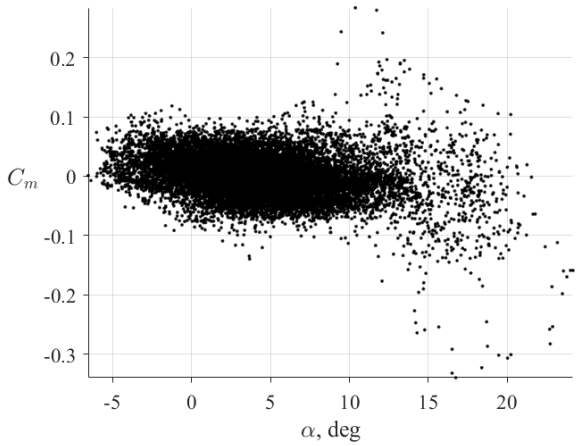


Fig. 12 C_m vs. α modeling data.

The cell structure evolution for C_m parametrized across the range of α is shown in the hierarchical depiction in Fig. 13, which also describes the cell numbering scheme. The first split partitions the high-α from the low-α aerodynamics, while the latter two splits concentrate on the nonlinear behavior around where stall might be expected to occur. The final cell structure shown with four cells is consistent with a physical interpretation of a single linear model across a wide range of low-α data, another model in the post-stall region, and a clustering around stall.

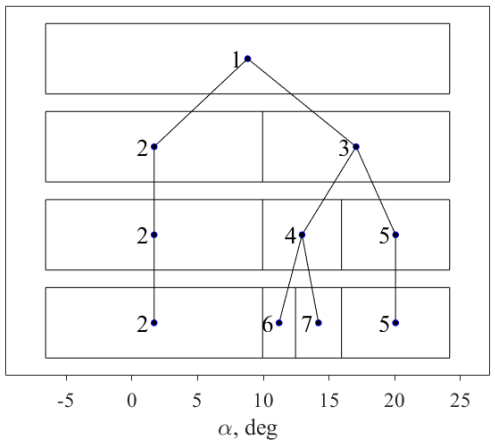


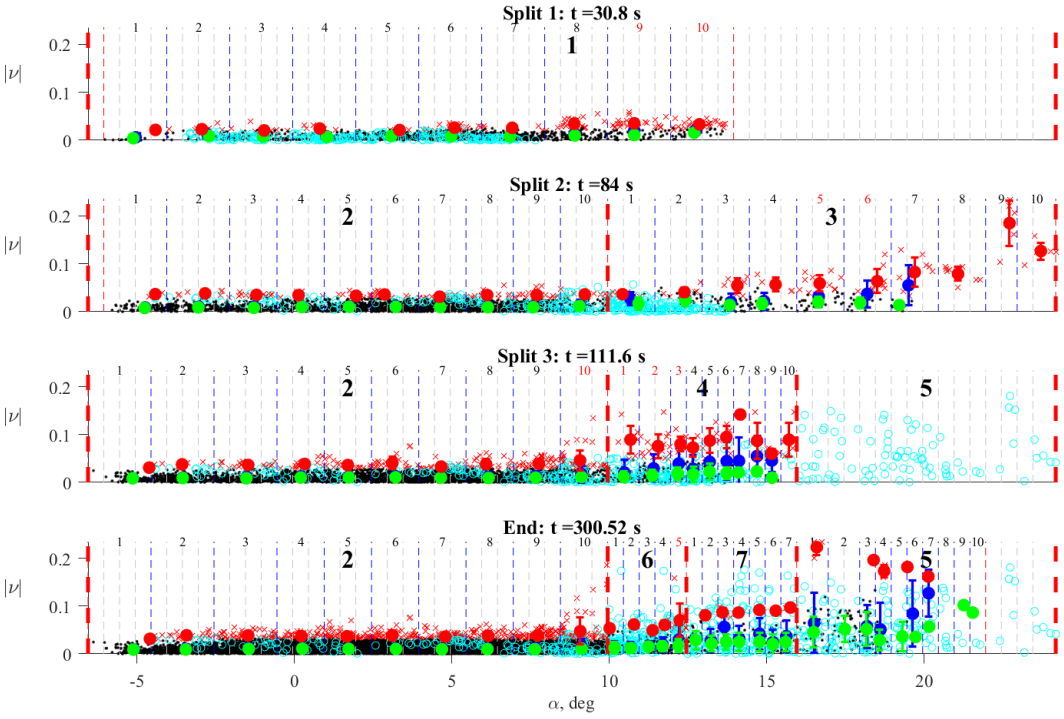
Fig. 13 Cell structure evolution for C_m.

A closer look through a depiction of the cell structure decision-making process will clarify how the cells evolve over time based on the available data. Figure 14 displays the binned local residuals just prior to each split, i.e. it shows a characterization of the local model fit quality that triggers a split. The last plot is shown at the completion of the flight data for the final cell structure. The residuals are plotted against the complete range of α, so the vertical red lines mark the boundaries of the cells. Only residual data obtained in each cell following a prior split are shown, such that data from the parent cell are excluded. There is a maximum of 10 active combined bins allowed in each cell, and these boundaries are shown in blue with the bins numbered accordingly, and the outer boundaries of the active bin range are shown in red. The numbers for the bins that are considered failed at the time of the split are also colored red. Notice that the bin width decreases as the cells are divided so that the resolution for further splits increases. The minimum bin widths of 0.5 deg used to discretize the possible bin boundaries are shown in light grey.

Beginning with one linear model across the range of α, notice that at low α, although there are numerous residuals that are characterized as unacceptable, their relative magnitude is still small compared to the acceptable residuals, and so the bins pass the status check. At high α, however, the unacceptable residuals outweigh the acceptable ones, and

Bins 9–10 are considered failed, so the first split is performed at the left boundary of Bin 9. Correspondingly, the second split partitions the mid-α data from the highest-α region before Bin 5 of Cell 3. Although Bins 7–10 in Cell 3 contain numerous unacceptable residuals, the total number of residuals is below the specified amount of 20, and so these bins are not considered failed yet. Finally, the third split breaks up the region where stall would be expected. Notice that throughout the splits, the relative magnitude of the residuals is reduced as the model fits are improved. The layer of unacceptable residuals on top in Cell 2 is a result of those measurements lying just beyond the residual threshold, whereas if a larger, more conservative residual threshold factor was chosen, these points would likely be considered acceptable. Notice also, for example, that the magnitude of the acceptable residuals in Cell 5 is much larger than the magnitude for those allowed in Cell 2 because the residual threshold is dependent on the noise levels.

To illustrate this dependency more clearly, Fig. 15 shows the residual threshold over time for all cells. Since each measurement belongs only to a single cell, only the threshold for at most one cell is updated at each point in time, with these points connected with a dashed line. Notice that the noise estimates increase across α as expected. Although no further split was made in the post-stall region in Cell 5, the model there is not necessarily anticipated to be of high quality due to the noise content.



Key			
Residual		**Mean + Std Dev**	
Unrestricted	○	Acceptable	
Acceptable	·	Unacceptable	
Unacceptable	×	Both	

Fig. 14 Binned local residual characterization.

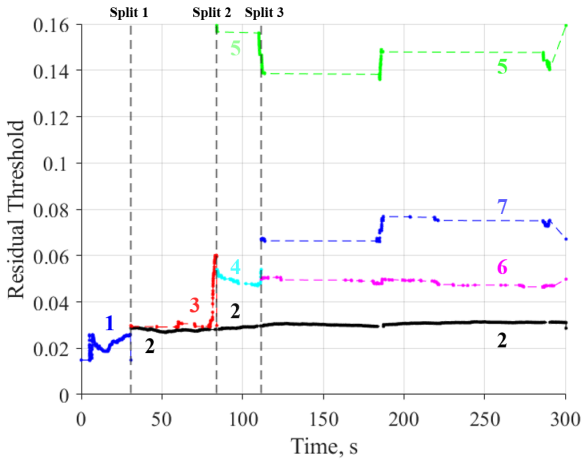


Fig. 15 Residual thresholds across all cells.

Figure 16 shows the final parameter estimates for each cell, along with their associated 95% (2σ) uncertainty bounds, plotted along α at the center of the associated cells. The model fit error variance in Eq. (14) was computed using only the measurements that were actually contained within the range of each cell. Further physical insight into the aerodynamics can be seen here. The static longitudinal stability C_{m_α} is appropriately negative for a statically stable aircraft, and its magnitude increases across the cells, indicating a stronger restoring moment as α is increased. The pitch damping derivative C_{m_q} describes the pitching moment that is caused by pitch rate and is related to the added lift generated on the horizontal tail due to aircraft rotation about the center of mass. Therefore, it is largest in magnitude at low α and decreases across the range of α as the horizontal tail effectiveness is reduced. Accordingly, the elevator effectiveness $C_{m_{\delta_e}}$ is maximized at low α and reduced at higher α. Advance ratio did not have an apparent impact on pitching moment at low α, but the dependency increased at higher α, corresponding to when power was added during the stall maneuvers. Notice that the uncertainty is larger for the estimates in the higher-α cells where the least amount of data are obtained, where noise levels are larger, and where the aerodynamics are more difficult to model, with unsteady and coupled lateral-directional behavior often prevalent.

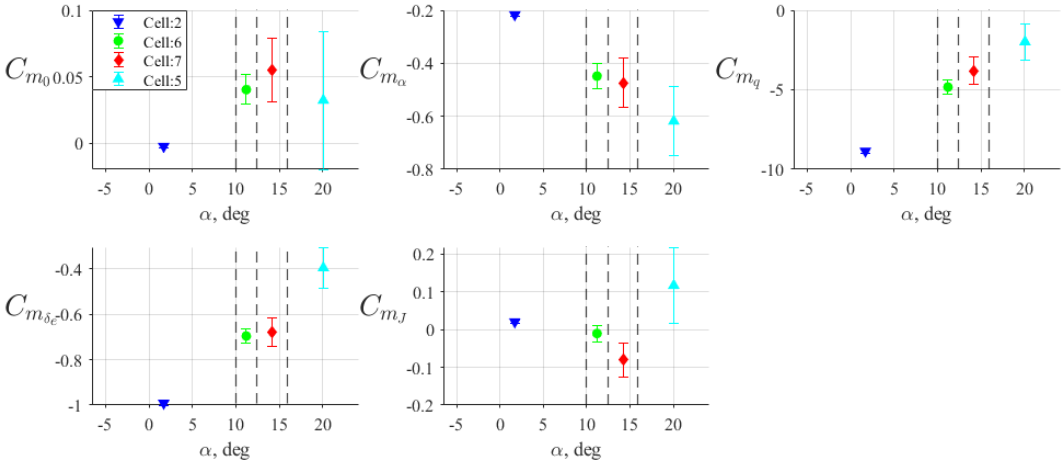


Fig. 16 Local model parameter estimates across all cells with 95% confidence intervals.

Figure 17 shows an example of the time history of the local parameter estimate C_{m_α} through all of the splits. Notice the definitive discontinuities in the estimates following each split – this is when the unacceptable residuals are incorporated into the child cell models instantaneously. The time at which no data is received in a particular cell is also shown with a dashed line here. The parameter estimate for Cell 2 appears to converge due to much data, whereas those

for the other final cells are only updated intermittently during the two stall maneuvers that remain following the last split. Notice the perturbations in the estimates for Cells 5–7 at around 186 s and 290 s, as that is when the latter two stalls occur in the modeling data. Based on the discussion on cell initialization in Section IV.C, SPLITR is highly dependent on repetition of maneuvers across the flight envelope since each child cell's model can only be updated with future data contained in the range of the cell.

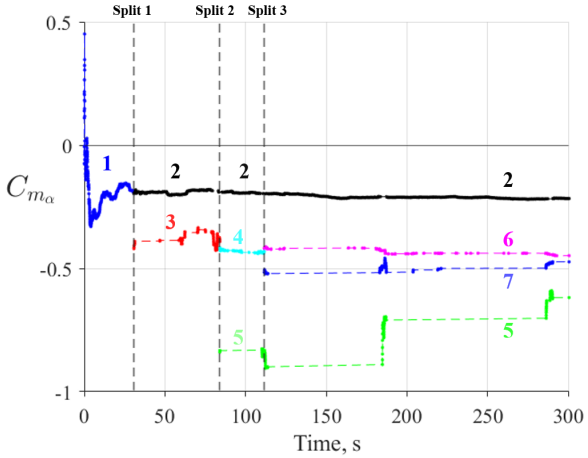
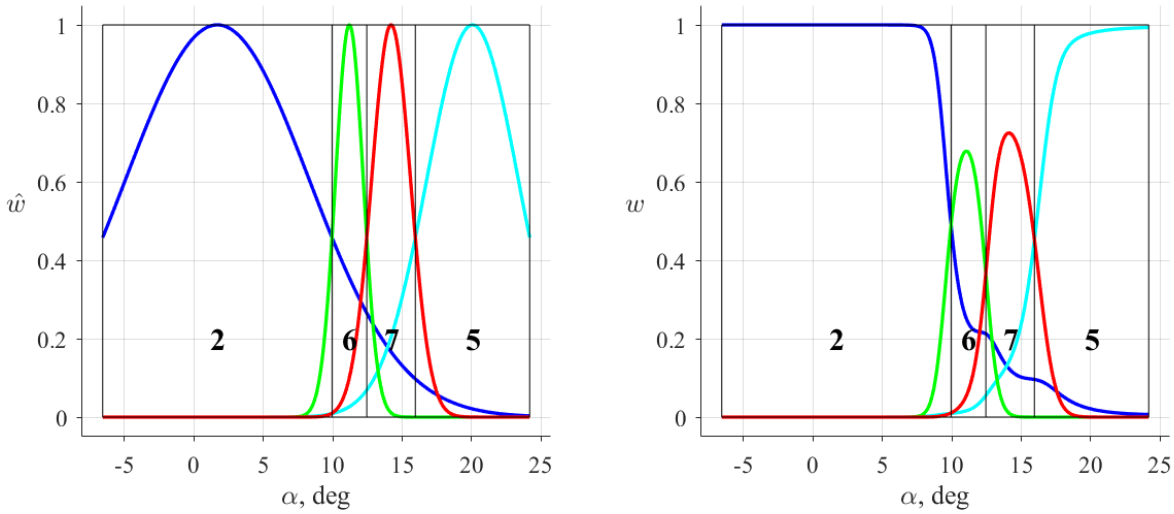


Fig. 17 Time history of C_{m_α} estimates across all cells.

As discussed in Section IV.B, each cell's model is locally approximated irrespective of the adjacent cells or of the global weighting, which retains the physicality and interpretability of the resulting parameter estimates. The global nonlinear model is then calculated using the validity functions shown in Fig. 18 that weight the local models in each cell and describe the influence of each model across the range of the PV. For example, Fig. 19 shows the weighted global parameter estimate for C_{m_α} across all cells, along with the local estimates relevant most strongly at the center of the respective cells. Notice the waves in the global estimate across Cells 6 and 7 and the fact that the parameter estimates that belong to those cells are not weighted by 1 at their cells' centers. This is a result of the normalization of Gaussian validity functions that is required to enforce a partition of unity, and which can potentially cause significant and unintentional deformation of the weighting functions for many cells of varying widths [33].



(a) Validity functions **(b) Normalized validity functions**

Fig. 18 Validity functions for C_m model.

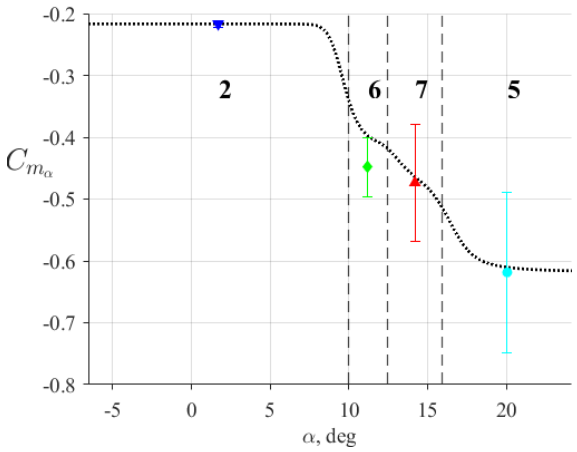


Fig. 19 Local and weighted global parameter estimates for C_{m_α}.

A separate set of flight data that was not part of the modeling process was used for model validation to assess the model's predictive capabilities, and the time histories of C_m and α for this data set are shown in Fig. 20. The validation data included additional figure eight maneuvers and 2 stalls to ensure a wide range of α was represented so that the full extent of the model could be tested.

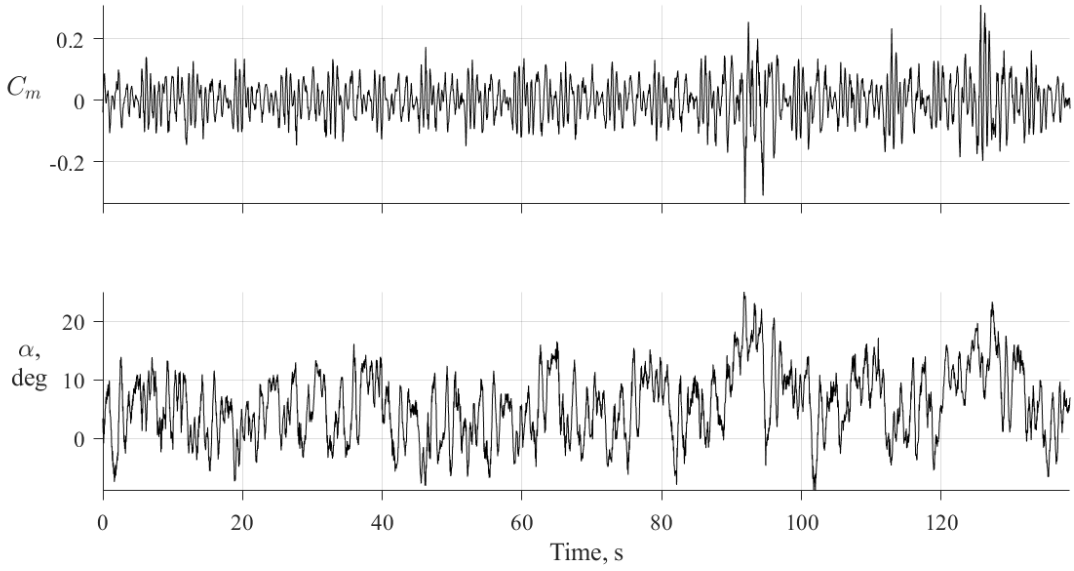


Fig. 20 Time histories of response variable and PV validation data.

The prediction error for the SPLITR model was compared to that of a batch linear model. The linear comparison model can be thought of as having the initial cell structure shown in Fig. 13, with only 1 cell across the entire range of α. The SPLITR model recognized the inherent nonlinearities in C_m, and this model with four local cells provides a better fit with $R^2 = 0.77$, compared to the linear model, which has $R^2 = 0.66$. The prediction error for these models can also be analyzed more closely than by simply comparing these global statistics.

In contrast to Fig. 14, where the local model residuals were analyzed to inform the cell partitioning, Fig. 21 depicts the global model residuals as a function of α across each cell to show the localizion of the global model fit, and the linear model error is visibly larger throughout most of the α range. Fig. 22 further shows the RMS of the global residuals across 15 bins for each model. The SPLITR model shows a substantial reduction in prediction error at low α and high α compared to the linear model, but little improvement around where stall is expected to occur in Cells 6 and 7. Stall

behavior is a complex and nonlinear phenomenon with coupled and unsteady effects that are not accounted for in these simplified local, linear models. Notice in Fig. 14 that there is scarce information obtained in the region of the right-most bin shown here in Cell 5, and the error is also largest there. Sources of modeling error can also include measurement noise, model structure errors resulting from local nonlinearity or an inadequate regressor pool, in-flight disturbances, and neglected dynamics such as unsteady, structural, or aeroelastic effects.

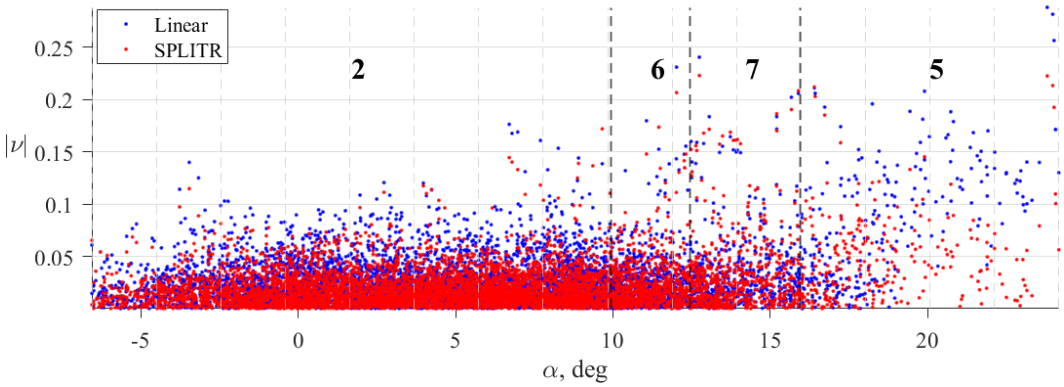


Fig. 21 Global residuals for validation data.

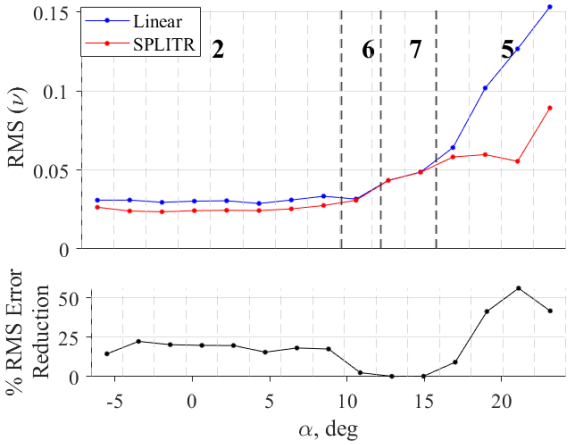
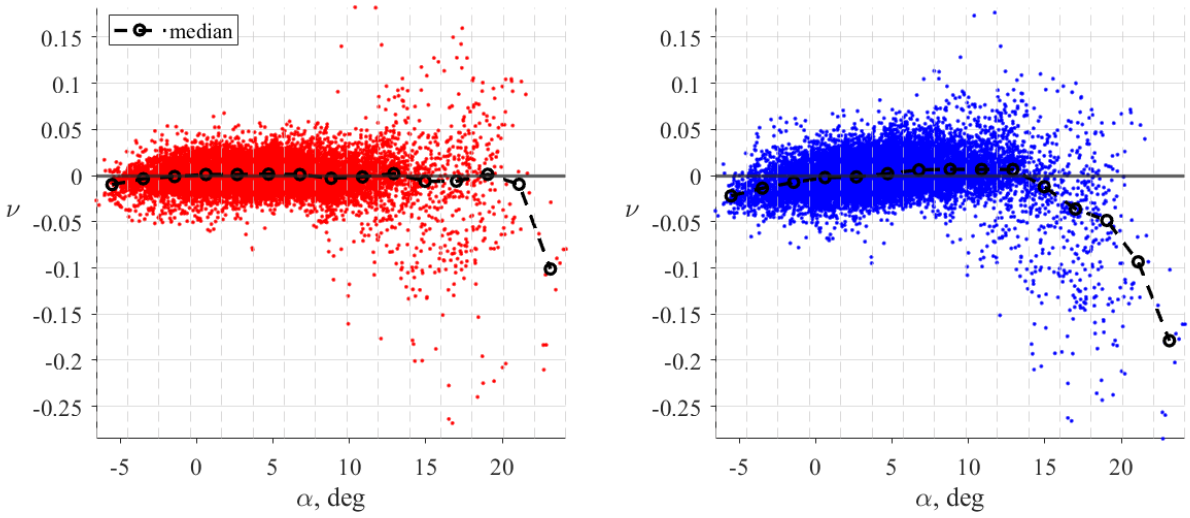


Fig. 22 RMS of binned global residuals for validation data.

The basis behind the cell structure determination process is to decorrelate the residuals as a function of the PVs that are associated with the nonlinearities in the data. For this conventional E1 vehicle, the aerodynamics are fairly linear throughout much of the flight envelope, which is why the linear model also performed well. But the logic behind this method is also applicable for less conventional vehicles. Still, significant improvements were seen in this case, and the SPLITR model succeeded in decorrelating the global residuals as a function of α compared to the linear model, as shown in Fig. 23.



(a) SPLITR model **(b) Linear model**

Fig. 23 Global residuals vs. PV for modeling data.

D. Further Discussion

The response variable modeling data for the other longitudinal force and moment coefficients are shown in Fig. 24, along with the corresponding cell structure evolutions that resulted from the modeling process. C_Z is typically relatively straightforward to model for conventional aircraft, and the data for C_Z are clearly linear across a wide range of α, so the partitions provided better resolution only in the high-α region, similar to C_m. On the other hand, C_X is generally difficult to represent without a separate propulsion model, and since thrust was not measured directly for E1, the C_X calculation in Eq. (1) includes both the thrust and aerodynamic forces. In this case, the cell structure decision making tried to compensate for the missing modeling information by over-splitting in α, which did not improve the model fit. Despite the prior assumption, C_X did not appear to be strongly dependent on α, but rather showed strong correlation with advance ratio. When the residual threshold factor λ_ν was raised to 9, and the PV was selected to be advance ratio instead of α, the model resulted in four cells across the range of advance ratio, and a much improved model fit compared to a single linear model. Notice how the breakpoints are used to partition what appears to be a second-order dependency on advance ratio into several linear regions.

The corresponding depictions for the lateral-directional aerodynamic coefficients, C_Y, C_l, C_n, are shown in Fig. 25. The data were partitioned as functions of β for each model and showed symmetry with larger residuals and resulting splits occurring at high β. Despite different response variable data and noise levels, the split locations for C_l and C_n are very close to each other, indicating a consistent nonlinear dependency on β that was recognized separately across both coefficients. Although engineering judgment might imply that the β partitioning is expected to be symmetric, these results confirm that the algorithm was able to detect the symmetry by showing the cell locations were placed accordingly for C_l and C_n.

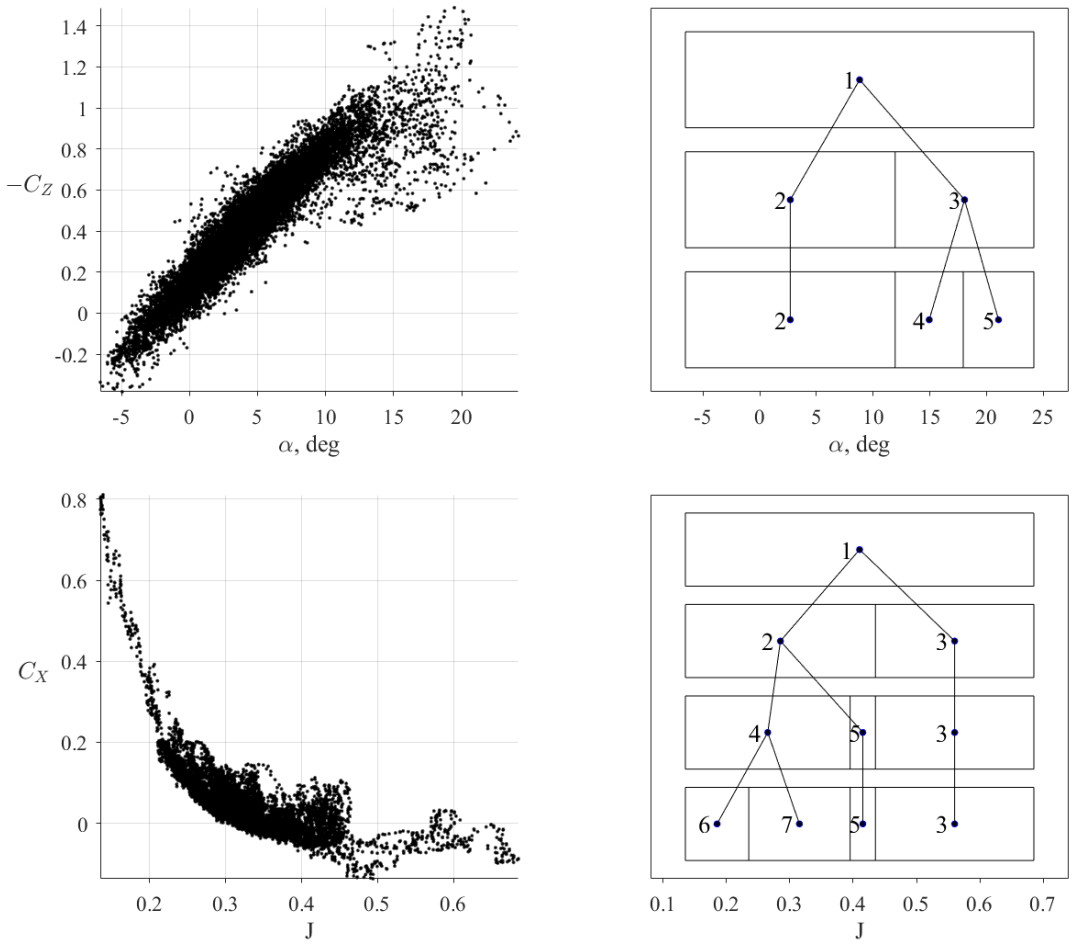


Fig. 24 Other longitudinal response variable modeling data and cell structure evolution.

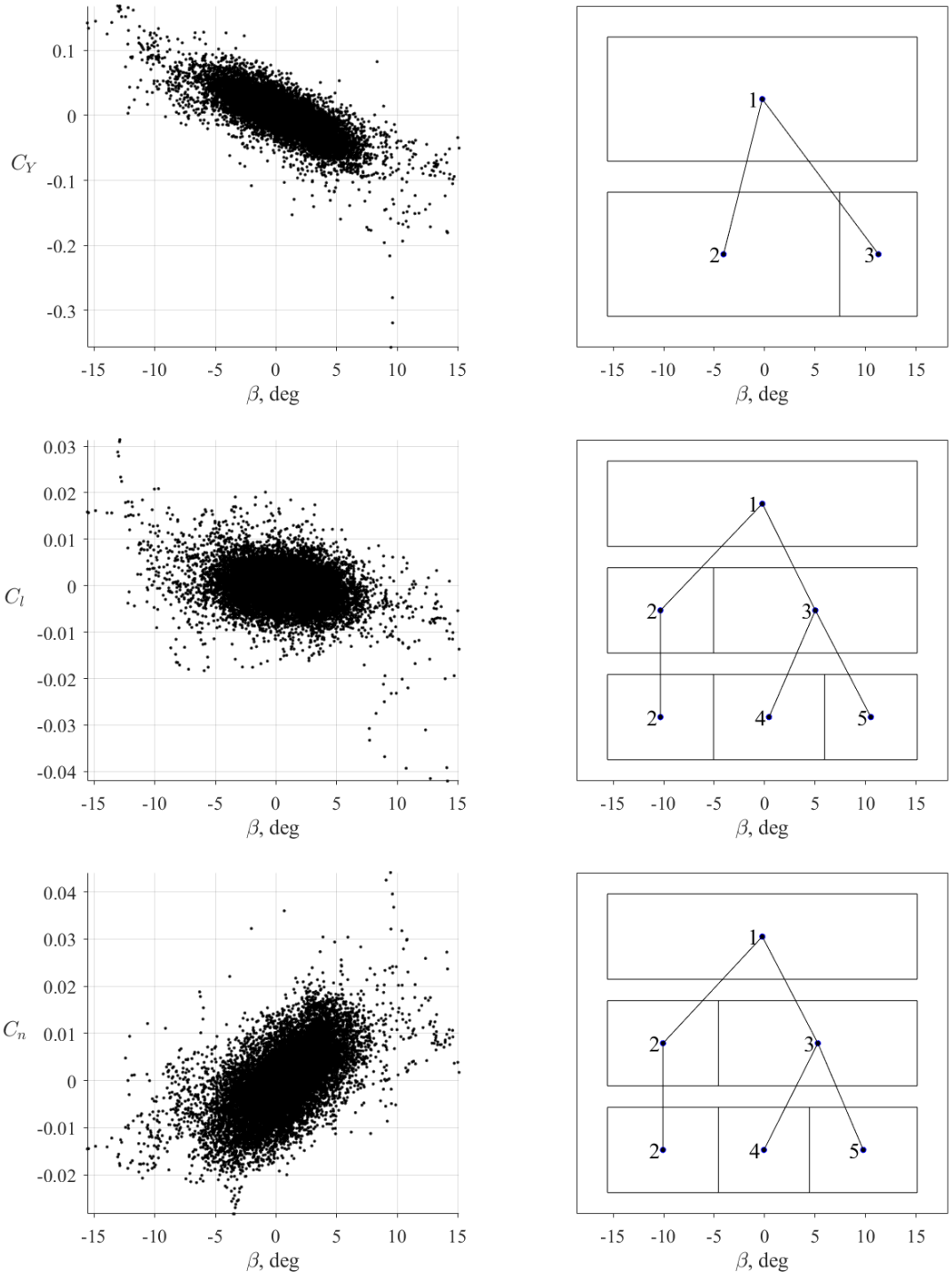


Fig. 25 Lateral-directional response variable modeling data and cell structure evolution.

The localized regions and associated models identified through SPLITR can provide further insight into where additional modeling data need to be gathered, and can be used to inform an autonomous envelope expansion and model development algorithm. Each local model can be evaluated based on its fit error, the noise levels of the data it contains, its parameter estimate uncertainties, and the number of measurements, among other metrics. The regions with poor local model fits due to lack of data can then be more easily identified, and these regions can be returned to in order to

obtain additional information. It is important to note that a crossplot of the data coverage of certain variables or another form of data density depiction does not provide this information, since the relative number of data points in a certain area of the flight envelope is not a definitive indicator of the model quality in that region. A model fit can be sufficient with few data points if the dynamics are linear and well behaved, and conversely, a poor fit can result even in a highly dense region. Fig. 26 illustrates this with a histogram of the α modeling data that is colored with the binned RMS of the validation prediction error of the SPLITR model, which is shown quantitatively in Fig. 22 for the C_m case. As expected here, the error is in fact inversely correlated with N at high α, but notice also that the model performs well in the lowest-α region where relatively few measurements also lie.

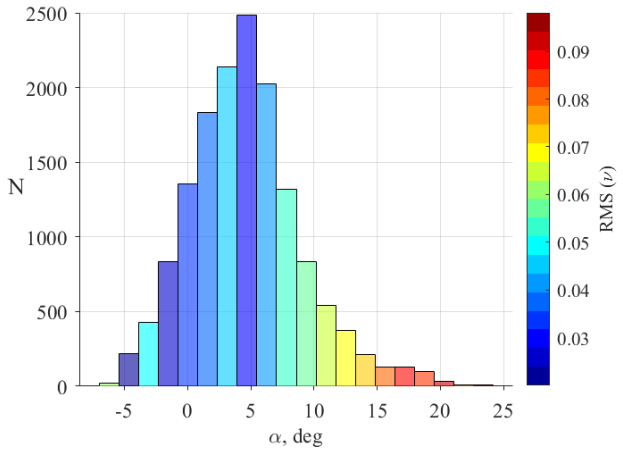


Fig. 26 Histogram of modeling PV data with RMS of prediction error for C_m SPLITR model.

VII. Conclusions

This paper presented a novel approach to automated global nonlinear aerodynamic modeling using local model networks in real time. SPLITR successively partitions complex nonlinear aerodynamic behavior into local regions to develop a weighted global model that also accurately captures the local aerodynamics, and provides valuable physical insight and interpretability. SPLITR provides three sets of informative results: the cell structure, the local models, and the global nonlinear model. The decision making inherent in the cell structure evolution process can be visualized and understood, and the resulting model can also be analyzed and validated, both globally and locally. Furthermore, the model complexity, as characterized by the number of cells, is restricted due to the reliance on noise estimates to prevent overfitting and ensure a parsimonious model.

Despite the significant constraints of real-time operation, which limited the choice of tools and denied access to most past data, SPLITR was developed as a nonlinear modeling technique using methods that are expected to be compatible with real-time use. The algorithm was tested on a laptop computer in this work, but future work can implement it onboard a flight computer to validate the real-time compatibility.

The cell structures were found to be very stable for the results shown, with similar formulations generated by separate flight data sets. Even with only linear regressors and one-dimensional splits, SPLITR produces models with good accuracy and valuable insight. In the E1 results, much of the high-β data actually occurred at high α, so a two-dimensional cell structure that allows partitions through both variables for a single model can perhaps provide a more effective localization for coupled longitudinal and lateral-directional behavior. Although improved transparency is obtained by only allowing a single dimension to be split, strongly coupled nonlinearities can be better resolved by allowing multidimensional axis-orthogonal splits. Ultimately, this modeling approach is designed to be automated, but any *a priori* knowledge of the particular aerodynamics can be useful to specifying the algorithm parameters and regressor and PV pools to lead to more efficient computation.

This real-time modeling approach could potentially be used in the fast-paced context of Urban Air Mobility (UAM) onboard new and unconventional vehicle configurations to obtain an aerodynamic model efficiently and to mitigate the need for many ground-based tests. The local regions within the global aerodynamics can also be used to inform a control law design; in particular, if the models are restricted to be linear, then they can be used with linear control

techniques. Additionally, this method can be incorporated into tools that can provide fault-detection for vehicles that may experience damage in flight by recognizing that the measurements being received are not supported by the existing model, and correcting for the altered dynamics by updating the model in real time. These model updates can also be fed into an adaptive control law, and the control gains can be modified accordingly. If data are received in a new portion of the flight envelope and are not supported by the current model form, a new cell can simply be partitioned to capture the dynamics in this region without altering or degrading the existing model in other regions.

The current limitations of the SPLITR approach described in this paper are summarized as follows. First, the residual characterization procedure identifies deterministic properties in the residuals and responds by splitting cells. If the regressor and PV pools are inadequate and lack important modeling information, multiple splits can occur without model improvement. An automated model structure determination process in each cell can improve the local modeling capabilities, even if the regressor pool is still restricted to include only the linear terms. Second, child cell parameter initialization proved to be a significant challenge in real-time. Without access to past data that can be used to update parameter estimates, the flight envelope needs to be repeatedly covered to obtain new information and ensure each new local model can be updated following a split. Finally, local parameter estimation is treated independently from the global model weighting process, and as a result, the validity functions are simply overlaid on the local models in a non-optimal way. Furthermore, Gaussian normalization for many cells of varying widths can cause undesirable side effects to the shapes of the normalized validity functions. These functions can instead be customized based on the real-time quality assessment of each local model to weight the local models that are considered more reliable with higher validity, based on uncertainty metrics computed from the data.

The effectiveness of the SPLITR approach has been demonstrated using conventional aircraft, for which traditional understanding of aircraft aerodynamics could be used for validation. This real-time modeling capability has the potential to improve the efficiency of the aircraft modeling process and enable novel aircraft configurations to be developed and tested more rapidly and flown more reliably.

Acknowledgments

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE 1840340, and by the NASA Aeronautics Research Mission Directorate (ARMD) Transformational Tools and Technologies (TTT) project. The efforts of the flight test team at NASA Langley Research Center in conducting flight test operations and collecting E1 flight data are gratefully acknowledged. Technical discussions with Dr. Jared Grauer and Dr. Eugene Morelli are also greatly appreciated. We also acknowledge the support of the team from Morpheus Lab at the University of Maryland.

References

[1] Riddick, S. E., "An Overview of NASA's Learn-to-Fly Technology Development," *AIAA Atmospheric Flight Mechanics Conference*, Orlando, FL, January 2020 (to be published).

[2] Heim, E. H., Viken, E. M., Brandon, J. M., and Croom, M. A., "NASA's Learn-to-Fly Project Overview," AIAA 2018-3307, *AIAA Atmospheric Flight Mechanics Conference*, Atlanta, GA, June 2018.

[3] Morelli, E. A., "Autonomous Real-Time Global Aerodynamic Modeling in the Frequency Domain," *AIAA Atmospheric Flight Mechanics Conference*, Orlando, FL, January 2020 (to be published).

[4] Morelli, E. A., "Practical Aspects of Real-Time Modeling for the Learn-to-Fly Concept," AIAA 2018-3309, *AIAA Atmospheric Flight Mechanics Conference*, Atlanta, GA, June 2018.

[5] Foster, J. V., "Autonomous Guidance Algorithms for NASA Learn-to-Fly Technology Development," AIAA 2018-3310, *AIAA Atmospheric Flight Mechanics Conference*, Atlanta, GA, June 2018.

[6] Snyder, S. M., "Autopilot Design with Learn-to-Fly," *AIAA Atmospheric Flight Mechanics Conference*, Orlando, FL, January 2020 (to be published).

[7] Snyder, S. M., Bacon, B. J., and Morelli, E. A., "Online Control Design for Learn-to-Fly," AIAA 2018-3311, *AIAA Atmospheric Flight Mechanics Conference*, Atlanta, GA, June 2018.

[8] Grauer, J. A., "A Learn-to-Fly Approach for Adaptively Tuning Flight Control Systems," AIAA 2018-3312, *AIAA Atmospheric Flight Mechanics Conference*, Atlanta, GA, June 2018.

[9] Weinstein, R., Hubbard, J. E., and Cunningham, M. A., "Fuzzy Modeling and Parallel Distributed Compensation for Aircraft Flight Control from Simulated Flight Data," AIAA 2018-3313, *AIAA Atmospheric Flight Mechanics Conference*, Atlanta, GA, June 2018.

[10] Hui, K., Ricciardi, J., Ellis, K., and Tuomey, D., "Beechjet Flight Test Data Gathering and Level-D Simulator Aerodynamic Mathematical Model Development," AIAA 2001-4012, *AIAA Atmospheric Flight Mechanics Conference*, Montreal, Quebec, Canada, August 2001.

[11] Jategaonkar, R. V., Mönnich, W., Fishenberg, D., and Krag, B., "Identification of C-160 Simulator Data Base from Flight Data," *Proceedings of the 10th IFAC Symposium on System Identification*, Copenhagen, Denmark, Elsevier Sciences Ltd., Oxford, UK, 1994, pp. 1031–1038.

[12] Dupuis, R., Jouhaud, J., and Sagaut, P., "Surrogate Modeling of Aerodynamic Simulations for Multiple Operating Conditions Using Machine Learning," *AIAA Journal*, Vol. 56, No. 9, 2018, pp. 3622–3635.

[13] Larsson, M., De Raedt, P., and Hedlund, M., *Aerodynamic Identification using Neural Networks*, Linköping University Electronic Press, 1997.

[14] Linse, D. J., and Stengel, R. F., "Identification of Aerodynamic Coefficients Using Computational Neural Networks," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 6, November 1993, pp. 1018–1025.

[15] Klein, V., and Batterson, J. G., "Determination of Airplane Model Structure From Flight Data Using Splines and Stepwise Regression," NASA TP-2126, March, 1983.

[16] de Visser, C. C., Mulder, J. A., and Chu, Q. P., "Multidimensional Spline Based Global Nonlinear Aerodynamic Model for the Cessna Citation II," AIAA 2010-7950, *AIAA Atmospheric Flight Mechanics Conference*, Toronto, Ontario, Canada, August 2010.

[17] de Visser, C. C., Mulder, J. A., and Chu, Q. P., "Global Aerodynamic Modeling with Multivariate Splines," AIAA 2008-7500, *AIAA Modeling and Simulation Technologies Conference and Exhibit*, Honolulu, HI, August 2008.

[18] Brandon, J. M., and Morelli, E. A., "Real-Time Onboard Global Nonlinear Aerodynamic Modeling from Flight Data," *Journal of Aircraft*, Vol. 53, No. 5, September 2016, pp. 1261–1297.

[19] Morelli, E. A., "Real-Time Global Nonlinear Aerodynamic Modeling for Learn-to-Fly," AIAA 2016-2010, *AIAA Atmospheric Flight Mechanics Conference*, San Diego, California, January 2016.

[20] Grauer, J. A., and Morelli, E. A., "A Generic Global Aerodynamic Model for Aircraft," *Journal of Aircraft*, Vol. 52, No. 1, January-February 2015, pp. 13–20.

[21] Morelli, E. A., and Klein, V., *Aircraft System Identification – Theory and Practice*, 2nd Edition, Sunflyte Enterprises, Williamsburg, VA, 2016.

[22] Batterson, J. G., "Estimation of Airplane Stability and Control Derivatives From Large Amplitude Longitudinal Maneuvers," NASA TM 83185, 1981.

[23] 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, April 1989, pp. 334–339.

[24] Murray-Smith, R., "A Local Model Network Approach to Nonlinear Modeling," Ph.D. Dissertation, Department of Computer Science, University of Strathclyde, Glasgow, U.K, 1994.

[25] Nelles, O., Fink, A., and Isermann, R., "Local Linear Model Trees (LOLIMOT) Toolbox for Nonlinear System Identification," *12th IFAC Symposium on System Identification*, Vol. 33, No. 15, June 2000, pp. 845–850.

[26] Seher-Weiss, S., "Identification of Nonlinear Aerodynamic Derivatives using Classical and Extended Local Model Networks," *Aerospace Science and Technology*, Vol. 15, 2011, pp. 33–44.

[27] Hartmann, B., Ebert, T., Fischer, T., Belz, J., Kampmann, G., and Nelles, O., "LMNtool - Toolbox zum automatischen Trainieren lokaler Modellnetze," 22. Workshop Computational Intelligence, Dortmund, December 2012.

[28] Nelles, O., and Isermann, R., "Basis function networks for interpolation of local linear models," *Proceedings of 35th IEEE Conference on Decision and Control*, Vol. 1, 1996, pp. 470–475.

[29] `https://www.mathworks.com/help/stats/regression-trees.html`, Accessed: 17-November-2019.

[30] Morelli, E. A., "Flight Test Maneuver Design for Efficient Aerodynamic Modeling," AIAA 2011-6672, *AIAA Atmospheric Flight Mechanics Conference*, Portland, OR, August 2011.

[31] Morelli, E. A., "Efficient Global Aerodynamic Modeling from Flight Data," AIAA 2012-1050, *50th AIAA Aerospace Sciences Meeting*, Nashville, TN, January 2012.

[32] Riddick, S. E., Busan, R. C., Cox, D. E., and Laughter, S. A., "Learn-to-Fly Test Setup and Concept of Operations," AIAA 2018-3308, *AIAA Atmospheric Flight Mechanics Conference*, Atlanta, GA, June 2018.

[33] Shorten, R., and Murray-Smith, R., "Side effects of normalising radial basis function networks," *International Journal of Neural Systems*, Vol. 7, No. 2, 1996, pp. 167–179.