Aerodynamic sensing for hypersonics via scientific machine learning

Julie Pham*
The University of Texas at Austin, Austin, TX 78712, USA

Bryan J. Morreale†
Texas A&M University, College Station, TX 77843, USA

Noel Clemens‡, Karen Willcox§
The University of Texas at Austin, Austin, TX 78712, USA

This work develops a scientific machine learning (SciML) approach to infer aerodynamic parameters from measurements of the structural strain field induced by aerodynamic pressure loads. The SciML approach provides the computational underpinning for a novel sensing strategy for hypersonic vehicles to estimate distributed surface pressure loads, providing valuable information for guidance, navigation, and control. The feasibility of the idea is first explored by formulating the inference problem as a deterministic optimization problem. Numerical studies show the effects of sensor sparsity and measurement noise on the accuracy of inferred aerodynamic parameters. Optimal decision trees are then proposed as an interpretable, rapid inverse mapping from the strain measurements to the aerodynamic parameters. Two types of decision trees, optimal classification trees and optimal regression trees, are assessed. The performance of each optimal decision tree is evaluated for the estimation of the aerodynamic parameters and reconstruction of the distributed surface pressure loads. Using synthetic strain data with realistic levels of noise, the pressure loads can be predicted to within an average of 2% relative error.

I. Introduction

Key research areas in hypersonic vehicle design have recently emphasized the development of highly maneuverable, self-aware, and autonomous systems. To meet these requirements, reliable in-flight sensing capabilities are necessary in order to provide sufficient real-time information for guidance, navigation, and control. Conventional sensing technologies include inertial measurement units, GPS satellites, and optical lidar sensors. However, these devices may be dependent on several external factors such as satellite signal, visual obstructions, or adversarial action in denied airspace environments. Additionally, direct measurements of pressure or heat flux require sensors mounted external to the vehicle, making them susceptible to the extreme aerothermal conditions experienced during hypersonic flight. Therefore, new paradigms in sensing strategies must be explored to collect in-flight data that are rich enough to inform a fully autonomous, self-aware hypersonic vehicle. In this work, we seek to develop a novel vehicle-as-a-sensor strategy that can infer aerodynamic quantities of interest from measurements of structural deformation of the vehicle airframe.

Estimating unknown quantities of interest from noisy, sparse, and indirect measurements is a task broadly explored in the field of inverse problems. The formulation and solution of an inverse problem involves the integration of data and predictive simulation models. A deterministic formulation of an inverse problem leads to an optimization problem that seeks to find the parameters that best match the observed data [1]. Alternatively, a Bayesian formulation of an inverse problem brings in the critical element of uncertainty quantification. A Bayesian inverse problem seeks to find the posterior distribution of parameters that is informed by both observed data and prior information [2]. For systems governed by partial differential equations, as in our target hypersonic application, the solution of an inverse problem is typically computationally expensive because it involves repeated simulations of a physics-based model in the process of solving the optimization problem (in the deterministic case) or sampling the posterior distribution (in the statistical

*Graduate Research Assistant, Department of Aerospace Engineering and Engineering Mechanics, AIAA Student Member.

†Graduate Research Assistant, Department of Aerospace Engineering, AIAA Student Member.

‡Professor, Department of Aerospace Engineering and Engineering Mechanics, AIAA Fellow.

§Director, Oden Institute for Computational Engineering and Sciences, AIAA Fellow.

case). This computational cost has been addressed through surrogate and reduced-order models [1], but even so the direct solution of an inverse problem is unlikely to be feasible in real-time onboard a hypersonic vehicle.

This challenge motivates our use of scientific machine learning (SciML), which combines physics-based models and machine learning, in order to achieve drastic reductions in the computational cost of solving an inverse problem. In an offline phase, data produced by physics-based models are used to train a map that relates measured quantities (i.e., structural strain) to the target inference parameters (i.e., aerodynamic parameters). Once this inverse map is learned, it is encoded in a machine learning model that can be queried rapidly, allowing for real-time estimates of the target parameters. Machine learning has already demonstrated several successes in making estimates of state parameters for aerospace applications. For example, the pitch angle of a jet wing was successfully predicted from fiber Bragg strain measurements using a neural network [3], resulting in more accurate predictions than conventional calculations. As another example, the damage state of a wing for an unmanned aerial vehicle (UAV) was predicted in [4] using optimal classification trees [5, 6]. Given real-time sensor measurements, the UAV was able to estimate its level of structural damage and dynamically alter its flight path based on the gathered information. We seek to employ similar SciML techniques to enable the hypersonic vehicle-as-a-sensor concept.

Our proposed approach relies on the availability of high-fidelity physics-based computational models, as well as the measurement data from which to infer the quantities of interest. Hypersonic simulation has seen many developments, from high-fidelity simulations [7, 8] to reduced-order models [9, 10] in aerodynamics and aerothermoelasticity. Hypersonic measurement technologies also continue to improve. Strain measurements using fiber-optic sensors can provide information about shape changes for aerospace vehicles [11, 12]. In the laboratory, digital image correlation and fast-response pressure-sensitive paint can measure complex aerodynamic loads and structural response of test articles [13]. A key contribution of our work is to leverage the complementary strengths of predictive simulation and measured data, which we achieve by exploiting the rapid estimation capability delivered by machine learning classifiers.

Inverse problems in hypersonics are increasingly important for control and design purposes, especially for quantities of interest such as aerothermal loads and heat flux [14–17]. In this paper, the quantity of interest to be inferred is the full-field aerodynamic pressure distribution over the surface of the vehicle. Since the pressure loads induce a structural response in the vehicle, information about the loads are encoded in elastic response quantities, such as the strain over the vehicle. In this case, we consider the entire vehicle as a sensor. The benefit of this approach is that we can reconstruct pressure field information over the entire surface of the vehicle, as opposed to pointwise measurements of pressure at sparsely distributed locations. Having distributed pressure load information is highly informative for control and predictive purposes. An additional benefit of the approach is that the deformation of the airframe can be measured internal to the vehicle, avoiding the harsh aerothermal conditions of hypersonic flight. In this paper, we provide the computational underpinnings of the vehicle-as-a-sensor strategy through feasibility studies with simulation data. The SciML methodology and studies support a larger effort to develop the sensing concept. This larger effort includes extensive experimental validation in ongoing and future work.

The subsequent sections of the paper present the following topics. Section II states the problem formulation and methodology for both the deterministic inverse problem and SciML approaches. Section III describes the conceptual hypersonic vehicle testbed problem, which we employ as a case study for numerical experiments. Section IV presents results from each inverse approach for the testbed vehicle. Finally, Section V concludes the paper.

II. Aerodynamic Parameter Inference: Problem Formulation and Approaches

This section describes the problem formulation and two main approaches. Section II.A presents the problem formulation, and then Section II.B outlines the deterministic inverse approach, where the inverse problem is solved via optimization. Section II.C then describes SciML approach, which employs optimal decision trees as an inverse map.

A. Problem Formulation

1. Definitions

We consider a hypersonic vehicle at an instant in time, at operating conditions of Mach number M and angle of attack α. The aerodynamic state of the vehicle, $\mathbf{x}_{\text{aero}}$, comprises the density, velocity, and pressure of the flow field around the vehicle. The structural state of the vehicle, $\mathbf{x}_{\text{struct}}$, is the displacement of the deformed vehicle. We have the observed strain at sensor locations on the vehicle, contained in a d-dimensional observed data vector $\tilde{\mathbf{y}} \in \mathcal{Y}$, where $\mathcal{Y}$ is the space of possible sensor measurements. We define the quantities of interest, $\mathbf{q}$, which are the surface pressures over the vehicle. The goal is to infer $\mathbf{q}$ from the observed data $\tilde{\mathbf{y}}$. To approach this task, we define the parameter vector

$\mathbf{p} = [M, \alpha]^\top \in \mathcal{P}$, where the Mach number M and angle of attack α are intermediate quantities of interest that can be used to predict $\mathbf{q}$, and $\mathcal{P}$ is the space of possible operating conditions. We then approach the inference problem in two steps: (1) solve the inverse problem from $\tilde{\mathbf{y}}$ to $\mathbf{p}$, and (2) use the inferred $\mathbf{p}$ to predict $\mathbf{q}$.

Each of these quantities are related via simulation-based forward models, with the parameters $\mathbf{p}$ as the input operating conditions. We denote the forward aerodynamic model by $F_{\text{aero}}(\mathbf{x}_{\text{aero}}; \mathbf{p}) = 0$; this represents, for example, a computational fluid dynamics (CFD) simulation. Solving the forward aerodynamic model yields the aerodynamic state $\mathbf{x}_{\text{aero}}$, from which we can then compute the pressure quantities of interest $\mathbf{q} = G_{\text{aero}}(\mathbf{x}_{\text{aero}})$. We denote the structural forward model as $F_{\text{struct}}(\mathbf{x}_{\text{struct}}; \mathbf{q}(\mathbf{p})) = 0$, which given the pressure loads $\mathbf{q}$ as inputs, is solved to yield the structural state, $\mathbf{x}_{\text{struct}}$. We can then compute the model strain data at the sensor locations as $\mathbf{y} = G_{\text{struct}}(\mathbf{x}_{\text{struct}})$. We denote the combined aerostructural forward model as $F(\mathbf{x}_{\text{aero}}, \mathbf{x}_{\text{struct}}; \mathbf{p}) = 0$. Note that the model strain data $\mathbf{y}$ differs from the observed strain data $\tilde{\mathbf{y}}$; we model the discrepancy with additive noise, $\mathbf{v} \in \mathcal{V}$, where $\mathcal{V}$ is the space of possible noise realizations. The noise can represent sensor noise, sensor bias, or model error. Here, we model the noise as a Gaussian vector with independent, identically distributed elements with mean μ and standard deviation σ.

2. *Mappings and synthetic data*

To solve the inverse problem via scientific machine learning, we require training data that consists of many observations $\tilde{\mathbf{y}}$ over a range of operating conditions. We produce synthetic strain measurements $\tilde{\mathbf{y}}$ via a forward map, also called the *parameter-to-observable* map, denoted by

$$\tilde{F} : \mathcal{P} \times \mathcal{V} \to \mathcal{Y} \tag{1}$$

This mapping encapsulates the forward models, governing equations, and noise model. To outline the steps of producing synthetic measurements using this forward mapping, we first produce solution pairs, or snapshots, of the forward aerostructural model, $(\mathbf{y}_j, \mathbf{p}_j)$ along with the corresponding distributed pressure load $\mathbf{q}_j$, for $j = 1, 2, \ldots, n$, where j denotes the j^{th} snapshot index, and n is the total number of snapshots. We then generate synthetic realizations of noise by sampling from the Gaussian noise model to obtain a noise vector $\mathbf{v}_j^s$ for $s = 1, 2, \ldots, n_s$, where n_s is the number of random noise samples, and each element in $\mathbf{v}_j^s$ is an independent sample from the normal distribution $\mathcal{N}(\mu, \sigma^2)$. A synthetic strain measurement is then given by $\tilde{\mathbf{y}}_j^s = \mathbf{y}_j + \mathbf{v}_j^s$. Algorithm 1 outlines the generation of training data, as well as the formation of data matrices used for training in SciML.

The inference problem is stated as follows: for a given $\tilde{\mathbf{y}}$, we seek to infer the parameters $\mathbf{p}$, or the operating conditions, from which we can predict $\mathbf{q}$. We define the inverse map, T, from the strain data $\tilde{\mathbf{y}}$ to the parameters $\mathbf{p}$ as

$$T : \mathcal{Y} \to \mathcal{P} \tag{2}$$

In Section II.B, we employ a deterministic optimization approach as the inverse map. Then, in Section II.C, we use the training data combined with a SciML approach to produce optimal decision trees as a rapid inverse map. In both approaches, once the parameters $\mathbf{p}$ have been estimated, we then predict the quantities of interest $\mathbf{q}$.

Algorithm 1 Training Data Generation

```
Inputs: $\{\mathbf{p}_j\}_{j=1}^n$
for $j = 1$ to $n$ do
    Solve $F(\mathbf{x}_{\text{aero}}, \mathbf{x}_{\text{struct}}; \mathbf{p}_j) = 0$    ▹ solve aerostructural forward model
    $\mathbf{q}_j \leftarrow G_{\text{aero}}(\mathbf{x}_{\text{aero}})$    ▹ store surface pressures for parameters $\mathbf{p}_j$
    $\mathbf{y}_j \leftarrow G_{\text{struct}}(\mathbf{x}_{\text{struct}})$    ▹ store model strain data for parameters $\mathbf{p}_j$
    for $s = 1$ to $n_s$ do
        Generate random noise sample $\mathbf{v}_j^s$    ▹ sample each element of $\mathbf{v}_j^s$ from $\mathcal{N}(\mu, \sigma^2)$
        $\tilde{\mathbf{y}}_j^s \leftarrow \mathbf{y}_j + \mathbf{v}_j^s$    ▹ store synthetic strain measurement
    end for
    $\mathbf{Y}_j \leftarrow \begin{bmatrix} \tilde{\mathbf{y}}_j^1 & \tilde{\mathbf{y}}_j^2 & \ldots & \tilde{\mathbf{y}}_j^{n_s} \end{bmatrix}$    ▹ collect synthetic strain measurements for each snapshot
end for
$\mathbf{Y} \leftarrow \begin{bmatrix} \mathbf{Y}_1 & \mathbf{Y}_2 & \ldots & \mathbf{Y}_n \end{bmatrix}$    ▹ collect all synthetic measurements
return $\{\mathbf{q}_j\}_{j=1}^n, \mathbf{Y}$
```

3. *Surface pressure parameterization*

Since the surface pressure $\mathbf{q}$ is high-dimensional, we parameterize the pressure field using the proper orthogonal decomposition (POD) of the pressure snapshot matrix $\mathbf{Q}$, which is assembled by concatenating each $\mathbf{q}_j$ column-wise into a matrix $\mathbf{Q}$. We center the snapshot matrix using the mean $\bar{\mathbf{q}}$ of the snapshots. We then compute the (thin) singular value decomposition of the centered snapshot matrix, $\mathbf{Q}' = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$, where the columns of $\mathbf{U}$ are the left singular vectors, $\mathbf{\Sigma}$ is the diagonal matrix containing the singular values, and the columns of $\mathbf{V}$ are the right singular vectors. We can then obtain a reduced order reconstruction of a pressure field by

$$\mathbf{q} \approx \bar{\mathbf{q}} + \sum_{k=1}^{r} c_k \mathbf{u}_k \tag{3}$$

where r is the number of POD modes to be retained, $\mathbf{u}_k$ is the k^{th} left singular vector, and c_k are the POD coefficients, or modal coordinates. Letting $\mathbf{c} = [c_1, c_2, \ldots, c_r]^\top$, we compute $\mathbf{c}$ by the orthogonal projection of $\mathbf{q}$ onto the subspace spanned by the first r left singular vectors, $\mathbf{c} = \mathbf{U}_r^\top \mathbf{q}$, where $\mathbf{U}_r$ retains the first r columns of $\mathbf{U}$. The POD allows for the least-squares optimal rank-r representation of $\mathbf{Q}$, and is employed in the next sections as a low-dimensional representation of the pressure field.

B. Deterministic Optimization Approach

In a deterministic optimization approach, for a given (synthetic) observed data vector $\tilde{\mathbf{y}}$, we search the space of the aerodynamic parameters $\mathcal{P}$ to find the optimal parameter vector $\mathbf{p}$ which minimizes the data misfit objective function defined by

$$\min_{\mathbf{p}} \sum_{i=1}^{d} \left(\frac{y_i(\mathbf{p}) - \tilde{y}_i}{\gamma_i} \right)^2 \qquad \text{where } F(\mathbf{x}_{\text{aero}}, \mathbf{x}_{\text{struct}}; \mathbf{p}) = 0,\ \mathbf{y} = G_{\text{struct}}(\mathbf{x}_{\text{struct}}) \tag{4}$$

where the subscript i denotes the i^{th} element in $\mathbf{y}$, γ_i is a normalization factor which is chosen to be $\max\{3\sigma, \tilde{y}_i\}$, and $F(\mathbf{x}_{\text{aero}}, \mathbf{x}_{\text{struct}}; \mathbf{p}) = 0,\ \mathbf{y} = G_{\text{struct}}(\mathbf{x}_{\text{struct}})$ is the forward aerostructural problem solve. This objective function represents a least squares minimization problem, and the addition of the normalization factor provides a regularizing effect to account for noisy measurements of small strain responses. Each evaluation of the objective function involves one solve of the forward model for the parameters $\mathbf{p}$ at the current iteration to obtain $\mathbf{y}$. In this paper, the limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm is used to solve the minimization problem in Equation 4. In our implementation, we simplify the forward model by computing the surface pressure for each iteration via linear interpolation of POD coefficients of the pre-computed pressure snapshots $\mathbf{Q}$, instead of solving the CFD problem at each evaluation of the objective function. Through this, we obtain interpolated POD weights, which we use to reconstruct the pressure field using Equation 3. Consequently, we also constrain the parameters to lie within the range of the parameters contained in the snapshots. Once the optimal parameters are identified, the CFD model (or POD reconstruction) can be re-evaluated at the final parameters to compute the surface pressure loads. Using the deterministic optimization approach enables us to explore the tractability of the inverse problem, including potential ill-posedness or non-uniqueness, and quantify expected performance of an inverse map based on the level of sensor noise.

C. Scientific Machine Learning Approach

1. *Optimal Classification Trees*

The SciML approach has two main phases: (1) the offline phase, where simulation data is used to train the inverse map, and (2) the online phase, where the inverse map is evaluated in real-time to give in-flight estimates of the aerodynamic parameters of the vehicle. In this paper, we focus on the training process in the offline phase. To develop an inverse map T using a machine learning approach, we propose the use of an optimal classification tree (OCT), which offers a rapid method of classifying the input feature vector $\tilde{\mathbf{y}}$ to a corresponding discrete output which encapsulates the parameters of interest, $\mathbf{p}$. In this case, the inverse map T is the OCT itself. The training of the optimal classification tree is formulated as mixed-integer optimization problem, which has been demonstrated to be efficiently solvable by Bertsimas and Dunn [5, 6]. Learning the inverse map T can be formally stated as a minimization problem given by

$$\min_{T} R(T) + \lambda |T| \tag{5}$$

where $R(T)$ is the misclassification error of the classification tree T given the training data, and the parameter λ is the complexity parameter which controls the architecture of the decision tree, with constraints described in [5]. For classification, we must assign class labels to the snapshot solution pairs $(\mathbf{y}_j, \mathbf{p}_j)$ for $j = 1, 2, \ldots, n$, denoted as C_ℓ for $\ell = 1, 2, \ldots, n_\ell$. For sufficiently small parameter ranges, we can select $n_\ell = n$, meaning that we define a distinct class for each snapshot parameter case. This is the approach we have taken in this work. However, with increasing n, we can choose $n_\ell < n$ to limit the number of classes and thus the complexity requirement of the OCT. In this general case, we define an indexing vector $\mathbf{m}$ with dimension n to track the index of the class assigned to each snapshot. Once the classes have been defined, our learning problem is to classify new, noisy realizations of strain measurements into the discrete classes, C_ℓ.

To develop the optimal classification trees, we employ the Interpretable AI software implementation [18] for training and evaluating the models. To generate the training data required for the learning process, we must use the forward model and noise model to produce many synthetic measurements. The choice of n_s, which is the number of noise realizations, must be selected based on the expected noise level, the number of snapshots, and the number of classes n_ℓ. Algorithm 2 outlines the class assignment and training process.

Algorithm 2 OCT Training

```
Inputs: $\mathbf{Y} \subset \mathbb{R}^{d\times(n\times n_s)}$, $\{C_\ell\}_{\ell=1}^{n_\ell}$, $\mathbf{m}$
for $j = 1$ to $n$ do
    $\ell \leftarrow \mathbf{m}(j)$                                   ▷ obtain index for class corresponding to snapshot $j$
    $\mathbf{C}_j = \left[C_\ell \text{ repeated for 1 to } n_s\right]$   ▷ repeat class label assignment for all noise realizations for snapshot $j$
end for
$\mathbf{C} \leftarrow \left[\mathbf{C}_1 \quad \mathbf{C}_2 \quad \ldots \quad \mathbf{C}_n\right]$   ▷ collect labels for all data points in $\mathbf{Y}$
$\mathbf{Y}_{\text{train}}, \mathbf{C}_{\text{train}} \leftarrow \text{split}(\mathbf{Y}, \mathbf{C})$   ▷ split data into testing and training sets
$T \leftarrow \text{train}(\mathbf{Y}_{\text{train}}, \mathbf{C}_{\text{train}})$   ▷ train OCT using IAI package [18]
return $T$
```

Once the inverse map T is trained, it can then be invoked in the online phase to make a rapid prediction, as demonstrated in Algorithm 3. This would enable the inverse mapping to be used as a real-time sensing strategy.

Algorithm 3 OCT Online

```
Inputs: $\tilde{\mathbf{y}}, T, \{\mathbf{q}_j\}_{j=1}^{n}, \{C_\ell\}_{\ell=1}^{n_\ell}$
$\hat{C} \leftarrow T.\text{predict}(\tilde{\mathbf{y}})$   ▷ predict class label using trained tree $T$ for incoming observations $\tilde{\mathbf{y}}$
$\hat{\mathbf{p}}, \hat{\mathbf{q}} \leftarrow \mathbf{p}_j, \mathbf{q}_j$ corresponding to $C_\ell = \hat{C}$   ▷ obtain $\hat{\mathbf{p}}, \hat{\mathbf{q}}$ from snapshot library corresponding to the predicted class
return $\hat{\mathbf{p}}, \hat{\mathbf{q}}$
```

2. *Optimal Regression Trees*

Another type of optimal decision tree which allows for continuous output predictions is the optimal regression tree with linear predictions (ORT-L) [19]. Instead of discrete output cases, an optimal regression tree contains a linear model in each of the terminal leaves of the tree, determined via least-squares regression on the training data input and output features. This allows for continuous output predictions, while also potentially reducing the complexity of the tree structure. Consequently, we can reduce the output prediction dimension by reconstructing the pressure from a small subset of the POD coefficients. To make predictions for multiple POD coefficients using optimal trees, we progress to a chain of optimal regression trees where we sequentially identify each POD coefficient, using the previous coefficient predictions to inform the next coefficient prediction. The coefficients are identified in the order of the leading POD modes; that is, $\hat{c}_1$ corresponding to the leading POD mode $\mathbf{u}_1$, then $\hat{c}_2$, up to $\hat{c}_k$ where $k = r$. This introduces a conditional dependency on the prediction of higher modes, which is more informative than the independent prediction of each POD coefficient independently. In essence, the optimal regression tree discretizes the input space into regions in which the relation between the POD coefficient and the input features can be approximated by a linear model. The linear model in each of the output leaves is of the form

$$\hat{c}_k = \sum_{i=1}^{d} a_i \tilde{y}_i + \sum_{i=1}^{k-1} b_i \hat{c}_i \tag{6}$$

where $\tilde{y}_i$ is the input feature (sensor measurement), $\hat{c}_i$ are the previously predicted modal coefficients corresponding to higher modes, and a_i, b_i are the corresponding weights computed via least squares regression. The performance metric $R(T)$ in this case is chosen to be the mean squared error of the ORT-L prediction compared to the true POD coefficient. Algorithm 4 demonstrates how we sequentially train r trees, $T_1, T_2, \ldots, T_k$, for prediction of each POD coefficient.

Algorithm 4 Sequential ORT-L Training

```
**Inputs:** $\mathbf{Y} \subset \mathbb{R}^{d\times(n\times n_s)}$, $\{\mathbf{q}_j\}_{j=1}^{n}$, $\mathbf{V}_r$
**for** $j = 1$ to $n$ **do**
    $\mathbf{c}_j \leftarrow \mathbf{V}_r^\top \mathbf{q}_j$    ▷ compute POD coefficients
    $\mathbf{C}_j = \begin{bmatrix}\mathbf{c}_j \text{ repeated for 1 to } n_s\end{bmatrix}$    ▷ collect POD coefficients for each point corresponding to snapshot $j$
**end for**
$\mathbf{C} \leftarrow \begin{bmatrix}\mathbf{C}_1 & \mathbf{C}_2 & \ldots & \mathbf{C}_n\end{bmatrix}$    ▷ collect POD coefficients for all data points in $\mathbf{Y}$
$\mathbf{Y}_{\text{train}}, \mathbf{C}_{\text{train}} \leftarrow \text{split}(\mathbf{Y}, \mathbf{C})$    ▷ split data into testing and training sets
**for** $k = 1$ to $r$ **do**
    $\mathbf{c}_{\text{train}} \leftarrow \mathbf{C}_{\text{train}}(k, :)$    ▷ isolate single POD coefficient for training
    $T_k \leftarrow \text{train}(\mathbf{Y}_{\text{train}}, \mathbf{c}_{\text{train}})$    ▷ train ORT-L using IAI package [18]
    $\hat{\mathbf{c}} \leftarrow T_k.\text{predict}(\mathbf{Y}_{\text{train}})$    ▷ predict POD coefficient for training data using newly trained ORT-L
    $\mathbf{Y}_{\text{train}} \leftarrow \text{concatenate}(\mathbf{Y}_{\text{train}}, \hat{\mathbf{c}})$    ▷ augment training data matrix with predicted POD coefficient
**end for**
**return** $\{T_k\}_{k=1}^{r}$
```

Similar to the OCTs, Algorithm 5 demonstrates how the sequential ORT-L would predict the surface pressure in the online phase given a measurement $\tilde{\mathbf{y}}$.

Algorithm 5 ORT-L Online

```
**Inputs:** $\tilde{\mathbf{y}}, \bar{\mathbf{q}}, \{\mathbf{u}_k\}_{k=1}^{r}, \{T_k\}_{k=1}^{r}$
**Initialize:** $\mathbf{z} = \tilde{\mathbf{y}}$    ▷ initialize feature vector to be sequentially updated
**for** $k = 1$ to $r$ **do**
    $\hat{c}_k \leftarrow T_k.\text{predict}(\mathbf{z})$    ▷ predict POD coefficient using trained tree
    $\mathbf{z} \leftarrow \text{append}(\mathbf{z}, \hat{c}_k)$    ▷ add predicted coefficient to feature vector
**end for**
$\hat{\mathbf{q}} \leftarrow \bar{\mathbf{q}} + \sum_{k=1}^{r} \hat{c}_k \mathbf{u}_k$    ▷ reconstruct pressure field using Eq. 3
**return** $\hat{\mathbf{q}}$
```

III. Adapted IC3X Testbed Problem

An adapted version of the Initial Concept 3.X (IC3X) hypersonic vehicle is considered as the testbed problem for proof-of-concept numerical studies in this work. The IC3X was initially proposed by Pasiliao et al. [20], and a finite element model for the vehicle was developed by Witeof et al. [21]. The IC3X is a representative vehicle that is air-launched, and propelled via a rocket-scramjet cycle for a three-phase nominal trajectory for a standard mission. The geometry of the vehicle is shown in Figure 1. In this paper, we consider the outer-mold-line of the IC3X to construct an aluminum thin-walled structural model.

To compute the aerodynamic pressure loads, we use CART3D [22] to solve the compressible Euler equations for inviscid steady fluid dynamics on a Cartesian mesh generated as outlined in [23]. CART3D allows for rapid databasing of large Mach and attitude input spaces as a result of the automated cutcell Cartesian mesh generation with adaptive mesh refinement, which requires only a manifold surface mesh of the geometry. Consequently, this meshing capability allows for complex geometries to be simulated many times over varying input conditions. To define convergence of each

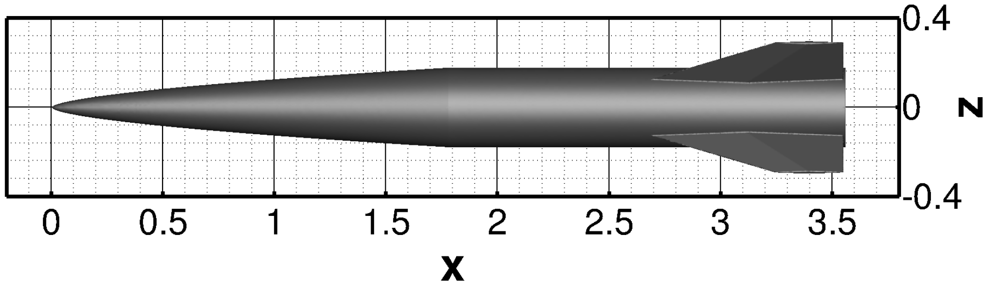

Fig. 1 IC3X geometry, with dimensions in meters.

CART3D solution, an estimated error tolerance of at least 10^{-3} was met for the output functional defined as the sum of the aerodynamic body force coefficients with equal weight [24], indicating that the pressure solutions were sufficiently converged. Figure 2 shows an example of the pressure solution for input parameters ($M = 6$, $\alpha = 2$).

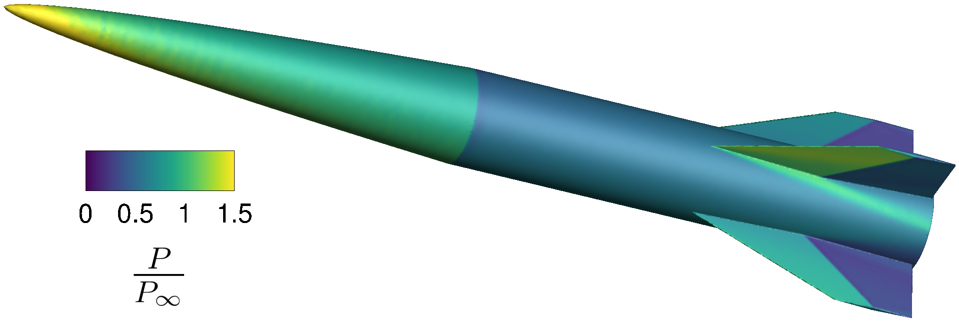

Fig. 2 Pressure (non-dimensionalized) on the surface of IC3X at ($M = 6$, $\alpha = 2$).

Once the CFD solutions are computed, the resultant pressure loads are then applied to a structural finite element model, which is a shell element model of the thin-walled structure used for preliminary computational studies and proof-of-concept analysis. A fixed boundary condition is enforced on the body at $x = 2.6$ m, at the vehicle's approximate center of gravity. The strain response is computed using the ANSYS Mechanical Static Structural solver [25]. We obtain the model strain response at nodal points which we define to be strain sensor locations. For the IC3X, 44 nodal points in the model are defined as the locations of the strain sensors. The sensors are placed in four rows of eleven, spatially distributed in both the axial and azimuthal directions, for a total of 44 sensors. The strain is measured in the streamwise direction. Figure 3 shows the placement of the 44 sensors, as well as a subset of 20 sensors which are considered as an additional limited-sensor case study.

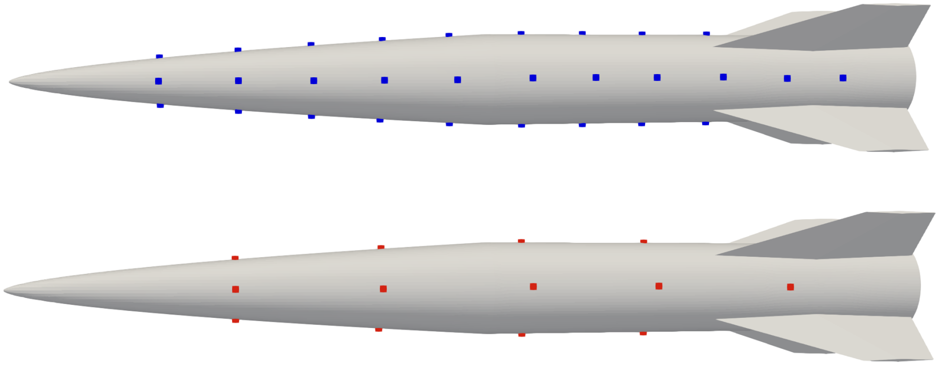

Fig. 3 Strain sensor placement with 44 sensors (top) and 20 sensors (bottom).

IV. Results

In the following sections, we present the results for the IC3X testbed problem for each of the described approaches: the deterministic optimization approach from Section II.B and the SciML approach via optimal classification trees from Section II.C. We consider the discrete parameter set of Mach numbers $\mathcal{M} = \{5, 5.2, 5.4, 5.6, 5.8, 6, 6.2, 6.4, 6.6, 6.8, 7\}$ and angles of attack $\mathcal{A} \in \{0, 1, 2, 3\}$, which are selected based on the expected operating parameters for a nominal cruise trajectory of the IC3X during a standard flight mission. Evaluating the vehicle aerostructural response at all combinations of Mach number and angle of attack results in $n = 44$ total snapshots of the model sensor strain response $\mathbf{y}$ and the corresponding pressure $\mathbf{q}$. To represent many realizations of noisy sensor data for numerical studies, we assume an additive Gaussian noise model as described in Section II.A with zero mean and standard deviation of 1.67% of the maximum sensor strain response over all of the snapshots of model sensor strains.

A. Deterministic Optimization

For demonstrative purposes, we examine the operating state case of ($M = 6$, $\alpha = 2$) as the true aerodynamic parameters. We generate $n_s = 30$ synthetically produced noisy reference measurements, and then solve the minimization problem in Equation 4 to estimate M and α. The resulting output estimates are shown in Figure 4. We observe that 90% of the test sample estimates lie within 2.8% and 6.9% of the true Mach number and angle of attack, respectively. These results show the feasibility of estimating aerodynamic parameters from indirect measurements of strain via a deterministic inverse approach, and that the aerodynamic loads can be reconstructed through the predictions of Mach number and angle of attack. However, each solve of the optimization problem incurs a large computational cost, which inhibits the optimization approach from being informative in real-time. Therefore, we must learn a rapid inverse mapping between the measurements and state parameters to implement the vehicle-as-a-sensor concept as an online sensing strategy.

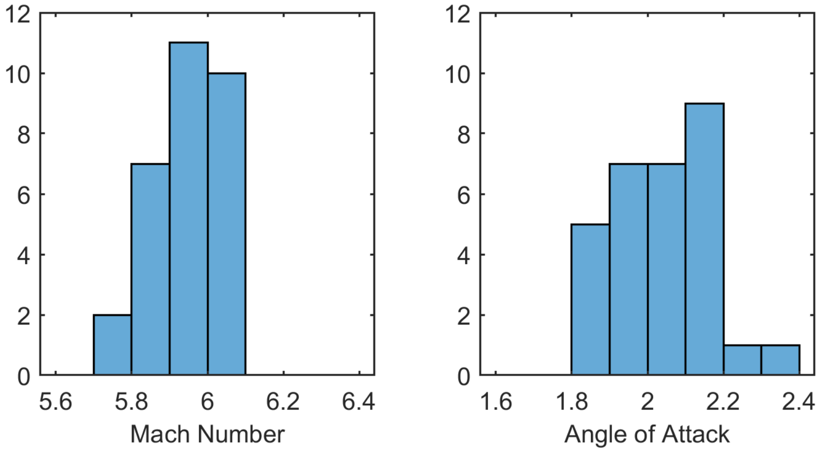

Fig. 4 Parameter estimates for 30 data points via optimization for the IC3X vehicle at ($M = 6$, $\alpha = 2$).

B. Scientific Machine Learning

To produce sufficient training data for the optimal decision trees, $n_s = 200$ noise samples are drawn for each snapshot to produce a total of 8800 data points. We define one class label C_ℓ for each snapshot, so that our 44 snapshots define $n_\ell = 44$ different classes. We train the optimal classification trees using Algorithm 2 with 50% of the dataset for training, and 50% of the dataset for testing.

1. Optimal Classification Trees

We begin discussion of the results with illustrative examples that demonstrate the mechanisms of the optimal classification trees. Consider a subset of the parameter cases: $(M = 6, \alpha = 1)$, $(M = 6, \alpha = 2)$, $(M = 6, \alpha = 3)$, and consider a subset of two sensors: Sensor 1 and Sensor 2. We observe in Figure 5 the distribution of observed strain for each case, along with the OCT which classifies the cases using axis-aligned splits — i.e., splits which are dependent on the observed strain from a single sensor. Clearly, each of these cases are distinguishable simply from Sensor 1, indicating that axis-aligned splits are sufficient for this classification problem. Now, consider a different subset of the parameter cases: $(M = 6, \alpha = 1)$, $(M = 7, \alpha = 3)$, as well as a different subset of two sensors. Now, we observe in Figure 6 that the classes may still be distinguishable via a series of axis-aligned splits, but another option is to use a single hyperplane split, which is a linear combination of both input features. Using hyperplane splits decreases the tree complexity, but can also decrease the interpretability of the results in more complex cases. In this work, we employ optimal classification trees with both the standard axis-aligned splits (OCT) and with hyperplane splits (OCT-H) to demonstrate the performance, as well as the interpretability, of the results.

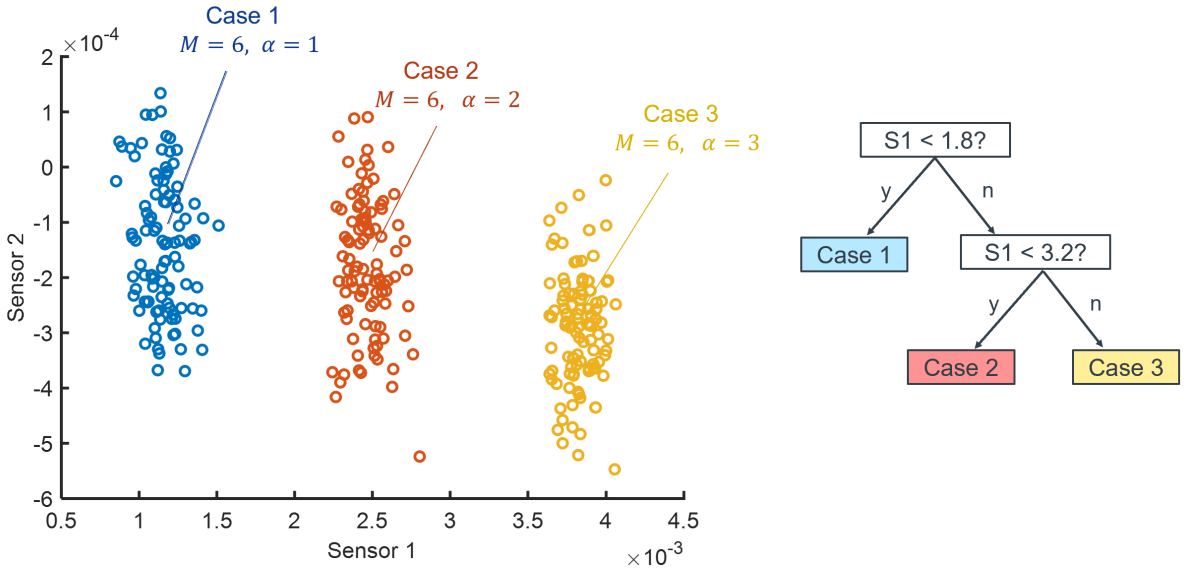

Fig. 5 Strain (synthetic) measurements for two sensors and three output classes, alongside the corresponding optimal classification tree with axis-aligned splits (OCT).

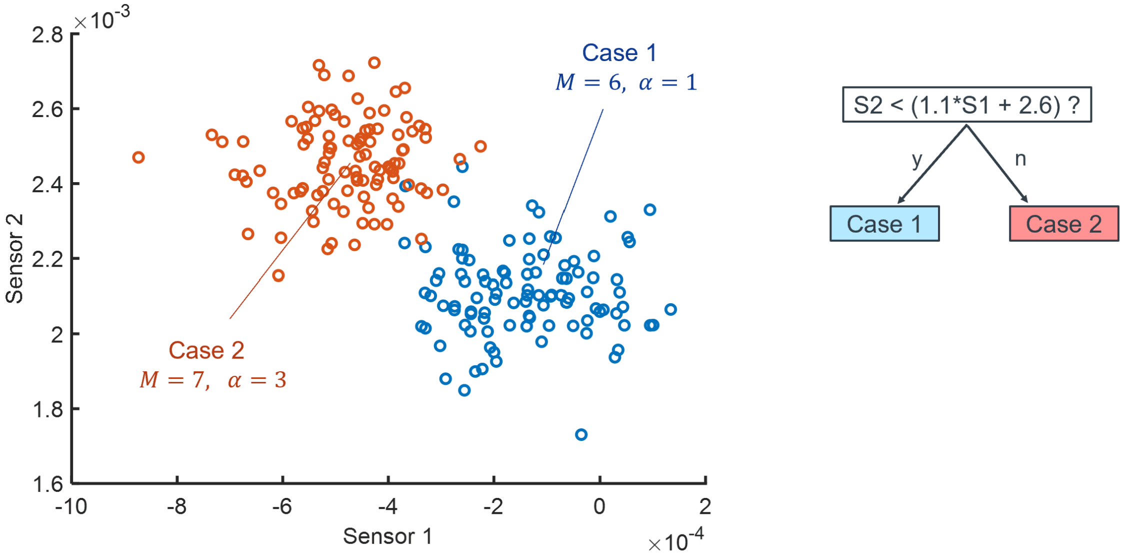

Fig. 6 Strain (synthetic) measurements for two sensors and two output classes, alongside the corresponding optimal classification tree with hyperplane splitting (OCT-H).

Returning to the full parameter set for training, we present a subset of results for operating conditions ($M = 6$, $\alpha = 2$). We consider both the 44 sensor set and the 20 sensor set for comparison. Figure 7 shows the histogram of predictions from the OCT for each sensor case for 100 test data points. We observe that in the 44 sensor case, 79% of the points were correctly classified as Mach 6, and 96% of the points were correctly classified as 2° angle of attack. Meanwhile, in the 20 sensor case, 71% of the points were correctly classfied as Mach 6, and 81% of the points were correctly classified as 2° angle of attack. We see that arbitrarily decreasing the number of sensors detracts from the robustness of the classifier, since the information gain is reduced.

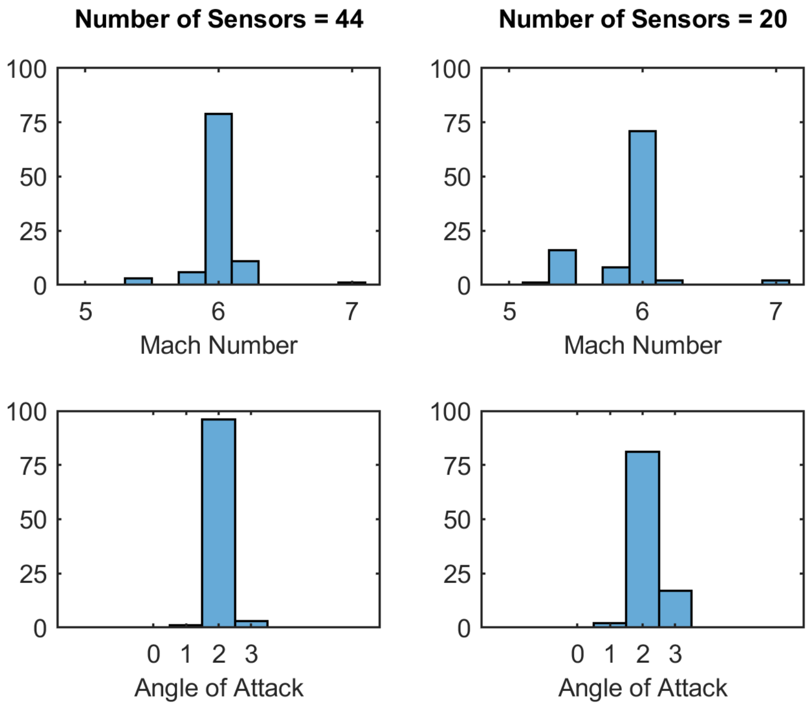

Fig. 7 OCT prediction histogram with 44 sensors (left) and 20 sensors (right) at parameters ($M = 6$, $\alpha = 2$).

This result can be improved by allowing hyperplane splitting using OCT-H. Figure 8 shows the OCT and OCT-H comparison for the 44 sensor case and Figure 9 shows the comparison for the 20 sensor case. The addition of hyperplane splitting in the 44 sensor case increases the correct predictions in Mach number by 14% of the total number of test samples, and enables the correct prediction of all of the test samples for angle of attack. In the 20 sensor case, the hyperplane splits improve the correct Mach number and angle of attack predictions by 15% of the total number of test samples. We also observe that in both cases, the majority of the outlier predictions have been removed, resulting in a more robust classifier in the presence of noise.

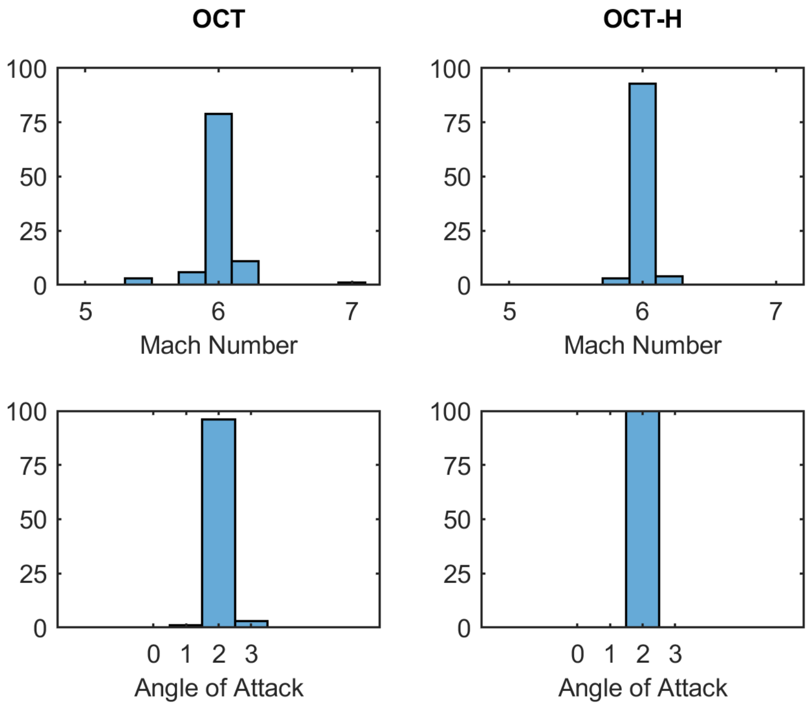

Fig. 8 Prediction histogram for OCT (left) and OCT-H (right) with 44 sensors at parameters ($M = 6$, $\alpha = 2$).

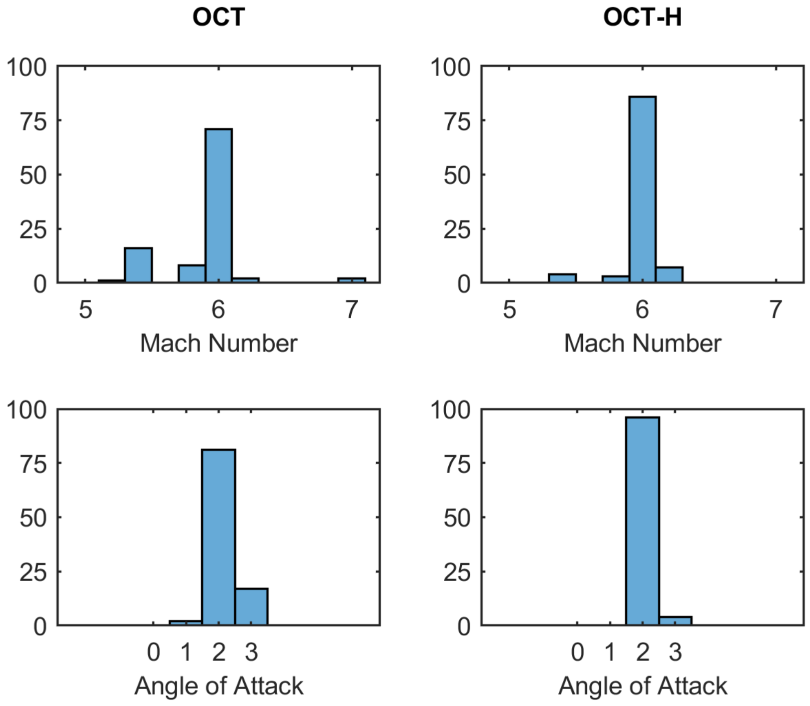

Fig. 9 Prediction histogram for OCT (left) and OCT-H (right) with 20 sensors at parameters ($M = 6$, $\alpha = 2$).

By analyzing the tree splits and features corresponding to the splits in the previous classifiers, we can determine the set of sensors which are the most informative with respect to differentiating between parameter cases. As a result, we choose the optimal 10 sensors to train an additional optimal classification tree with fewer input features. Figure 10 shows the predictions from the 10 sensor OCT and OCT-H classifiers. We see that with a selective choice of 10 sensors, the number of correctly identified test samples were 90% and 93% for the OCT and OCT-H classifiers, respectively. For the angle of attack, all test samples were correctly identified as $\alpha = 2$ in both cases. It is also noteworthy that a well-chosen subset of sensors can be more informative than a larger sensor set. This indicates that noise in uninformative sensors, or sensors with a small strain response, can have substantial effects on the performance results. Consequently, sensor placement is a critical consideration to maximize the performance of the indirect sensing strategy.

Once the Mach number and angle of attack have been estimated, we can easily obtain the distributed pressure loads. For a estimate $\hat{\mathbf{p}}$ of Mach and angle of attack made by the classifier, the predicted pressure distribution is the corresponding pressure snapshot $\mathbf{q}_j$, which can be queried rapidly online from a stored library of snapshots as in Algorithm 3. By our choice of the discrete class definitions C_ℓ, we note that the classes should be chosen with sufficiently small parameter spacing in order to make accurate pressure predictions within a particular tolerance. However, as the state parameter range to be considered increases, a different choice of classes C_ℓ may be better suited to reduce misclassification and tree complexity. This is a topic to be investigated in further studies.

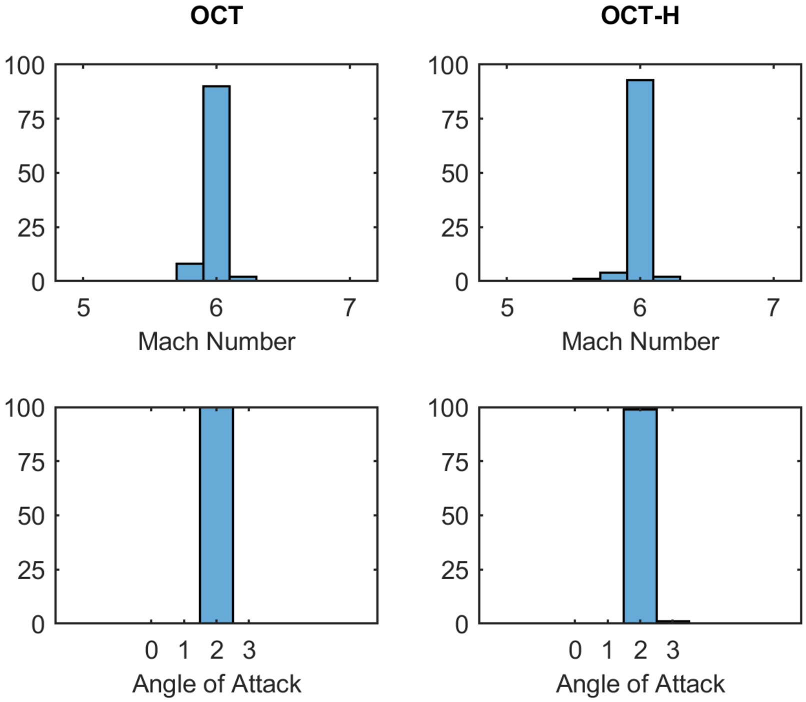

Fig. 10 Prediction histogram for OCT (left) and OCT-H (right) with 10 optimal sensors at parameters ($M = 6$, $\alpha = 2$).

2. *Optimal Regression Trees*

Instead of considering $\mathbf{p} = [M, \alpha]^T$ state parameters for prediction, we seek to predict the leading $r = 2$ modal coefficients for pressure field reconstruction using ORT-L with linear regression models contained in each of the output leaves such that the output prediction space is continuous rather than discrete. In this setting, we first train an independent optimal regression tree to predict the modal coefficient corresponding to the leading POD mode, c_1. We then use the ORT-L to predict $\hat{c}_1$ for each training data point, and use this to augment the input features for training the second ORT-L to identify the second modal coefficient, c_2. We then reconstruct the pressure load using the predictions via Equation 3. To begin, the thin singular value decomposition of the centered snapshot matrix $\mathbf{Q}'$ is computed to obtain the r left singular vectors, which form the columns of $\mathbf{U}_r$. Figure 11 shows the leading five POD modes, $\{\mathbf{u}_k\}_{k=1}^5$, which are the first 5 left singular vectors. The cumulative energy captured in the first two modes is 99.9%.

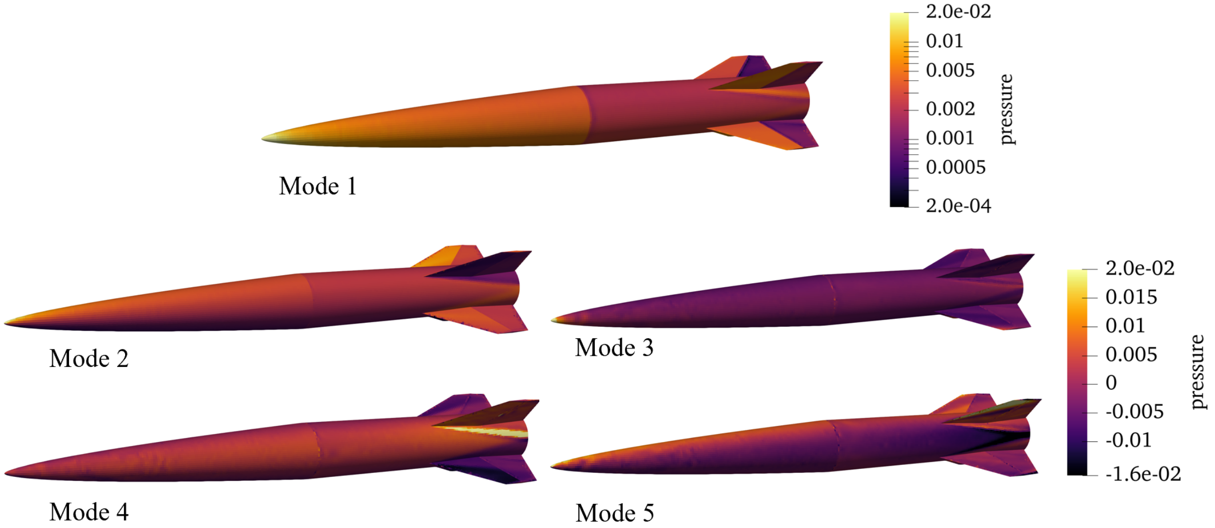

Fig. 11 Depiction of leading five modes of surface pressure snapshot matrix, Q.

Again, we consider the operating state case of ($M = 6$, $\alpha = 2$). For this case, the relative L2 error from the low-rank pressure field approximation with $r = 2$ to the true pressure snapshot is 1.28%. The true reconstruction coefficients are denoted as c_1 and c_2, and the reconstructed pressure field using the true coefficients is considered as the reference pressure field, $\mathbf{q}_{\text{ref}}$. To assess the accuracy of the ORT-L predictions, we define a normalized difference metric for coefficient predictions, given by

$$e_k = \frac{\hat{c}_k - c_k}{\max\{c_k\} - \min\{c_k\}} \tag{7}$$

The difference between the true and predicted coefficients is normalized by the range of the parameters for the given snapshot matrix, to account for the scaling of the non-dimensional coefficients. Figure 12 shows the distribution of the normalized differences e_1 and e_2 for 100 test samples, along with the relative L2 error from the reference pressure field to the pressure reconstructions $\hat{\mathbf{q}}$ using each pair $\hat{c}_1$ and $\hat{c}_2$. The results show that within this range, our coefficient predictions are accurate, resulting a median relative L2 error of 2%, and a maximum relative L2 error of below 9%. To visualize the results, Figure 13 shows the comparison of the reference pressure against the reconstructed pressure field using the predicted coefficients corresponding to the median error, as well as the absolute difference, $|\mathbf{q}_{\text{ref}} - \hat{\mathbf{q}}|$. We see that the reference pressure and predicted pressure fields are not visibly different, since the relative L2 error is small ($< 2\%$). The absolute difference shows the regions of the surface where the difference is largest; in this case, the largest difference in pressure occurs at the nose of the vehicle.

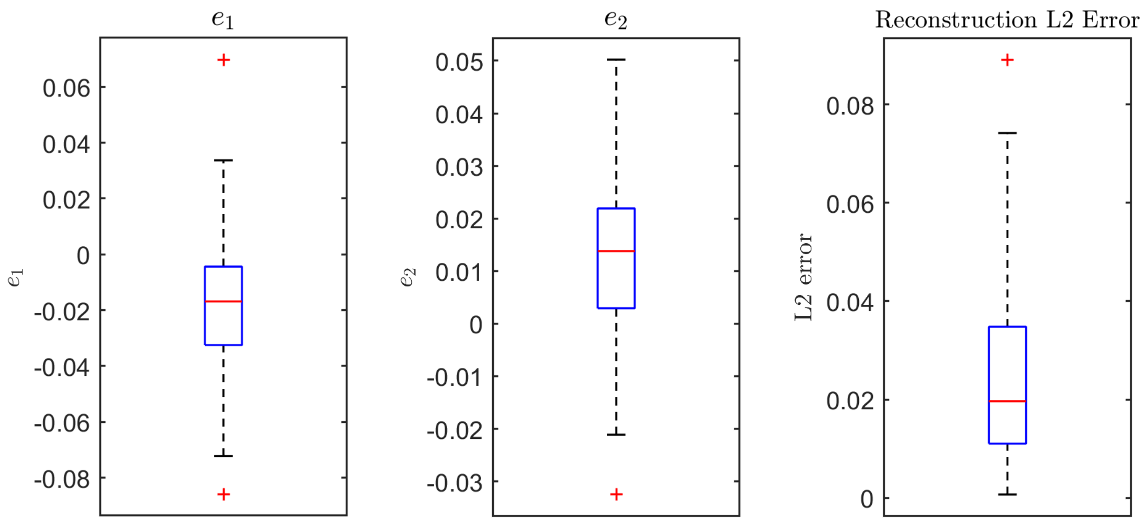

Fig. 12 Distribution of normalized difference metric for ORT-L POD coefficient predictions, along with relative L2 error for pressure reconstructions, for 100 test samples.

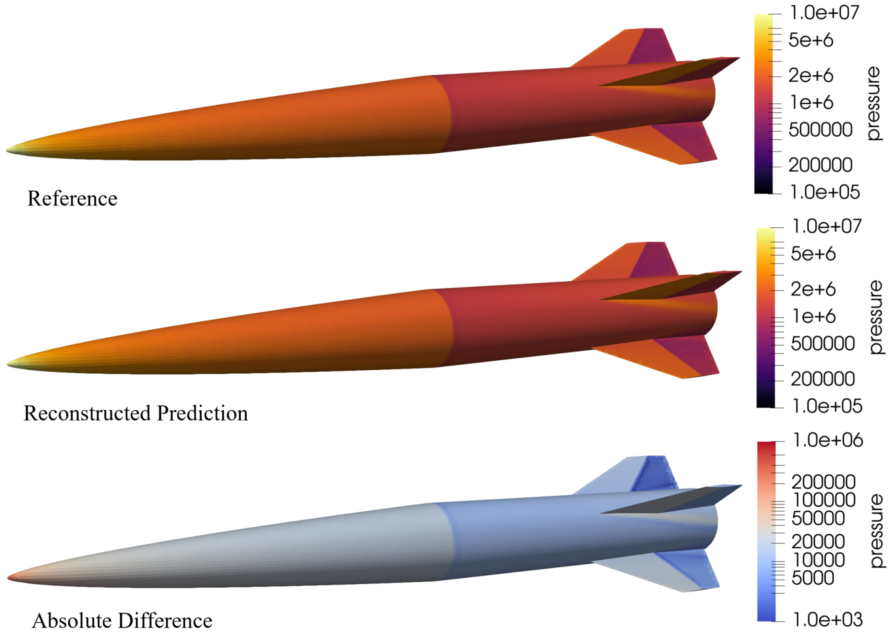

Fig. 13 Pressure visualization of the reference pressure field (top), the reconstructed pressure field using predicted POD coefficients (middle), and the absolute difference (bottom).

We repeated the study for an expanded parameter set to assess the performance with a larger set of angles of attack, $\mathcal{A} \in \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$. For this expanded parameter set, the cumulative energy associated with the leading two pressure POD modes is 99.0%, and the reconstruction error associated with the true coefficients is 2.1%. Figure 14 shows the distribution of the normalized differences e_1 and e_2 for 100 test samples, along with the relative L2 error from the reference pressure field to the pressure reconstructions $\hat{\mathbf{q}}$ using each pair $\hat{c}_1$ and $\hat{c}_2$. We observe that consideration of the larger parameter set results in larger relative L2 reconstruction errors, with a median of 5.6%, and a maximum below 20%. In this case, we may require the identification of more modes in order to capture more features of the pressure field, and consequently produce more accurate pressure reconstructions. Future studies will further investigate an expanded parameter range.

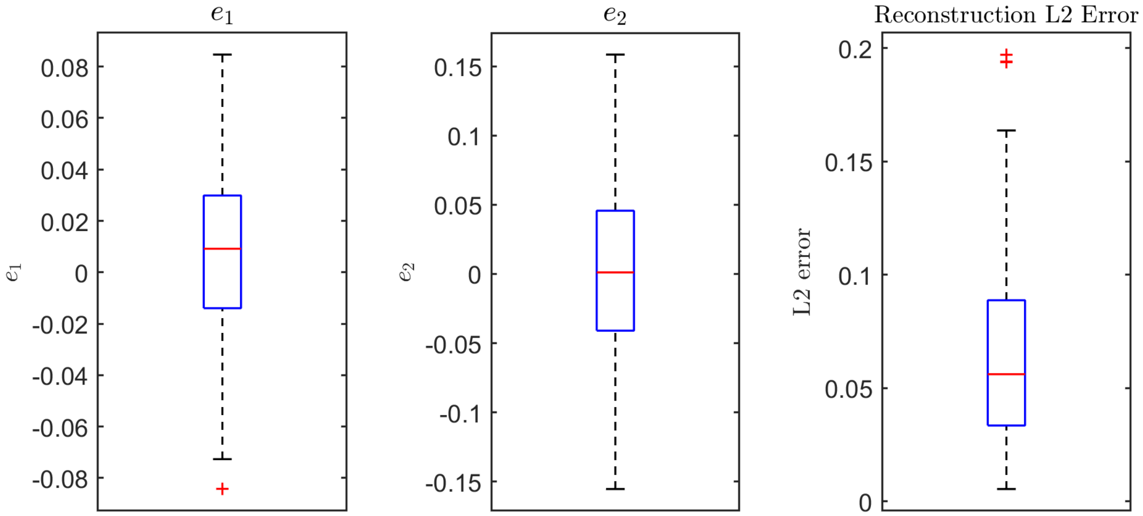

Fig. 14 Distribution of normalized difference metric for ORT-L POD coefficient predictions, along with relative L2 error for pressure reconstructions, for 100 samples using expanded parameter set snapshot modes.

V. Conclusion

This work presents numerical studies that indicate the feasibility of inferring quantities of interest from indirect measurements via SciML for surface pressure estimation in hypersonic vehicles. With synthetically generated noisy data, the optimal decision tree methodology results in a map from the observed strain data to the estimated aerodynamic parameters with reasonable accuracy for the considered parameter range. The study also demonstrates that predictions from the optimal decision trees are influenced by sensor noise and sensor placement, which must be considered in the training process in order to produce improved results. The SciML mapping process results in a large computational speed-up compared to solving a deterministic inverse problem via optimization, which gives the concept the potential to inform a hypersonic control system in real-time. An additional advantage is that the optimal decision trees offer a high level of interpretability in the training process, allowing for simple analysis and diagnostics of the results. This is particularly helpful in engineering applications where an understanding of the physics can guide hyperparameter tuning for training, and performance analysis for predictions.

Future work will consider a larger parameter range for the aerodynamic operating conditions, in addition to thermal effects. Consideration of thermal effects will require distinguishing the strain response due to aerodynamic loads from the strain due to thermal loads. Further considerations regarding sensors, including sensor bias, drift, and malfunction, as well as optimal sensor placement, are also part of future work. Reduced-order models may also provide computational cost reductions in the training process. Finally, demonstration and validation of the approach with experimental data is an important next step to establish the concept as a potential practical sensing strategy.

Acknowledgments

This work was supported by AFOSR grant FA9550-21-1-0089 under the NASA University Leadership Initiative (ULI), with Sarah Popkin as program manager. The authors thank Interpretable AI, LLC for use of their software and members of the FAST ULI team for helpful discussions regarding this work.

References

[1] Ghattas, O., and Willcox, K., "Learning physics-based models from data: perspectives from inverse problems and model reduction," *Acta Numerica*, Vol. 30, 2021, p. 445–554.

[2] Stuart, A. M., "Inverse problems: a Bayesian perspective," *Acta Numerica*, Vol. 19, 2010, pp. 451–559.

[3] Klotz, T., Pothier, R., Walch, D., and Colombo, T., "Prediction of the business jet Global 7500 wing deformed shape using fiber Bragg gratings and neural network," *Results in Engineering*, Vol. 9, 2021, p. 100190.

[4] Kapteyn, M. G., Knezevic, D. J., and Willcox, K., "Toward predictive digital twins via component-based reduced-order models and interpretable machine learning," *AIAA Scitech 2020 Forum*, 2020, AIAA 2020-0418.

[5] Bertsimas, D., and Dunn, J., "Optimal classification trees," *Machine Learning*, Vol. 106, No. 7, 2017, pp. 1039–1082.

[6] Bertsimas, D., and Dunn, J., *Machine learning under a modern optimization lens*, Dynamic Ideas LLC, 2019.

[7] Knutson, A. L., Thome, J. S., and Candler, G. V., "Numerical Simulation of Instabilities in the Boundary-Layer Transition Experiment Flowfield," *Journal of Spacecraft and Rockets*, Vol. 58, No. 1, 2021, pp. 90–99.

[8] McNamara, J. J., and Friedmann, P. P., "Aeroelastic and Aerothermoelastic Analysis in Hypersonic Flow: Past, Present, and Future," *AIAA Journal*, Vol. 49, No. 6, 2011, pp. 1089–1122.

[9] Falkiewicz, N. J., Cesnik, C. E. S., Crowell, A. R., and McNamara, J. J., "Reduced-Order Aerothermoelastic Framework for Hypersonic Vehicle Control Simulation," *AIAA Journal*, Vol. 49, No. 8, 2011, pp. 1625–1646.

[10] Klock, R. J., and Cesnik, C. E. S., "Nonlinear Thermal Reduced-Order Modeling for Hypersonic Vehicles," *AIAA Journal*, Vol. 55, No. 7, 2017, pp. 2358–2368.

[11] Freydin, M., Rattner, M. K., Raveh, D. E., Kressel, I., Davidi, R., and Tur, M., "Fiber-Optics-Based Aeroelastic Shape Sensing," *AIAA Journal*, Vol. 57, No. 12, 2019, pp. 5094–5103.

[12] Pena, F., Martins, B., and Richards, W. L., "Active In-flight Load Redistribution Utilizing Fiber-Optic Shape Sensing and Multiple Control Surfaces," NASA-TM 20180002147, *NASA Technical Memorandum*, 2018.

[13] Musta, M. N., Vanstone, L., Ahn, Y.-J., Eitner, M., Sirohi, J., and Clemens, N., "Investigation of flow-structure coupling for a compliant panel under a shock/boundary-layer interaction using fast-response PSP," *AIAA Aviation 2021 Forum*, 2021, AIAA 2021-2809.

[14] Smith, T., Bowcutt, K., Selmon, J., Miranda, L., Northrop, B., Lau, K., Silvester, T., Mairs, R., Unger, E., Paull, A., Paull, R., Dolvin, D., and Alesi, H., "HIFiRE 4: A low-Cost Aerodynamics, Stability, and Control Hypersonic Flight Experiment," *17th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*, 2011, AIAA 2011-2275.

[15] Bowcutt, K. G., "Physics Drivers of Hypersonic Vehicle Design," *22nd AIAA International Space Planes and Hypersonics Systems and Technologies Conference*, 2018, AIAA 2018-5373.

[16] Kumar, S., and Mahulikar, S. P., "Design of Thermal Protection System for Reusable Hypersonic Vehicle Using Inverse Approach," *Journal of Spacecraft and Rockets*, Vol. 54, No. 2, 2017, pp. 436–446.

[17] Cai, Z., Liu, T., Wang, B., Rubal, J., and Sullivan, J. P., "Numerical Inverse Heat Transfer Analysis for Temperature-Sensitive-Paint Measurements in Hypersonic Tunnels," *Journal of Thermophysics and Heat Transfer*, Vol. 25, No. 1, 2011, pp. 59–67.

[18] Interpretable AI, LLC, "Interpretable AI Documentation," Version 2.2.0, 2021. URL `https://www.interpretable.ai`.

[19] Dunn, J. W., "Optimal trees for prediction and prescription," Ph.D. thesis, Massachusetts Institute of Technology, 2018.

[20] Pasiliao, C. L., Sytsma, M. J., Neergaard, L., Witeof, Z., and Trolier, J. W., "Preliminary Aero-thermal Structural Simulation," *14th AIAA Aviation Technology, Integration, and Operations Conference*, 2014, AIAA 2014-2292.

[21] Witeof, Z., and Neergaard, L., "Initial concept 3.0 finite element model definition," AFRL-RWWV-TN-2014-0013*, Eglin Air Force Base, Air Force Research Laboratory*, 2014.

[22] Aftosmis, M., Berger, M., and Adomavicius, G., "A parallel multilevel method for adaptively refined Cartesian grids with embedded boundaries," *38th Aerospace Sciences Meeting and Exhibit*, 2000, Paper 2000-0808.

[23] Aftosmis, M. J., Berger, M. J., and Melton, J. E., "Robust and Efficient Cartesian Mesh Generation for Component-Based Geometry," *AIAA Journal*, Vol. 36, 1998, pp. 952–960.

[24] Nemec, M., and Aftosmis, M., "Adjoint Error Estimation and Adaptive Refinement for Embedded-Boundary Cartesian Meshes," *AIAA Computational Fluid Dynamics Conference*, 2007, AIAA 2007-4187.

[25] ANSYS, Inc, "ANSYS Academic Research Mechanical," Version 2021 R1, 2021.