Flush Air Data Sensing Based on Dimensionless Input and Output Neural Networks With Less Data

YANG LIU
Chinese Academy of Sciences, Beijing, China
University of Chinese Academy of Sciences, Beijing, China

CHEN-AN ZHANG
Chinese Academy of Sciences, Beijing, China

XUNSHI YAN
Tsinghua University, Beijing, China

WEN LIU
Chinese Academy of Sciences, Beijing, China

Neural networks have the ability to deal with the flush air data sensing (FADS) system of various vehicles. However, the demand for large quantities of training data limits its application. To overcome the problem, this article develops a FADS algorithm called dimensionless input and output neural networks FADS (DIO-NNFADS) to estimate air data states. The DIO-NNFADS is utilized to approximate the aerodynamic model defined by dimensional analysis, which decouples the freestream static pressure. Thus, trained by less data from a single flight profile, the DIO-NNFADS can achieve good accuracy in the entire flight envelope, effectively reducing the training data for neural networks. The Mach number, angle of attack, angle of sideslip, and the pressure coefficients are directly output by the DIO-NNFADS. And the static pressure and dynamic pressure are solved by the equations composed of the measured pressures and pressure coefficients. The proposed FADS algorithm is verified on a simplified supersonic model through numerical simulation. Results show that the algorithm can estimate the Mach number within the relative error of 2.9%, static pressure and dynamic pressure within the relative error of 6.2%, and the angle of incidence within the absolute error of 0.4° in the entire flight envelope. Besides, the optimal size of the training data set for the DIO-NNFADS is discussed. Furthermore, the influence of port layout and selection is analyzed, and the algorithm also shows good performance for a port layout without stagnation point.

I. INTRODUCTION

Air data states, such as the Mach number, angle of attack, angle of sideslip, static pressure, and dynamic pressure, are always critical to the flight control, guidance, and postflight analysis of most atmospheric flight vehicles [1], [2]. The systems used to sense, calculate, and output air data states are called the air data sensor (ADS) [3].

Traditional ADS systems for air data sensing involve intrusive booms that extend beyond the local boundary layer [4], [5]. It provides a stable measurement and is suitable for many practical applications. However, these intrusive booms can cause unwanted lateral instabilities for some vehicles with high angles of attack and increase radar cross-sectional area for surface vessels [6], [7]. Another serious problem is that these booms cannot withstand the tremendous heat load under hypersonic flight [8].

To deal with the above problems, a flush air data sensing (FADS) system was developed by NASA in the early 1960s [9], [10]. Unlike intrusive booms that penetrate the flow away from the influence of the vehicle, the FADS system uses pressure distribution from multiple pressure ports on the forebody of the vehicle to predict air data states that are complete nonintrusive air data sensing systems [8]. The FADS system can avoid the hypersonic heat load caused by the flow-sensing booms and extend the effective range of the ADS from the low-subsonic to the hypersonic flow regime. Since the FADS system uses the natural contours of the forebody and does not probe the flowfield, instabilities of the vehicle are not affected, and the radar cross-sectional area does not increase either. Due to these advantages, the FADS system has been successfully applied to high-performance vehicles in various speed domains, such as X-15 [11], X-31 [12], X-33 [8], and X-43A [13] in the United States, HYFLFX [14] in Japan, SHEFEX II [15] in Germany, etc.

Traditional FADS algorithms such as the triples algorithm are based on a semiempirical pressure model, which is established by fusing potential flow theories (at low-speed flow) and the modified Newtonian theory (at supersonic flow) [16]. However, these algorithms are only suitable for vehicles with blunt noses [17], [18], [19], and an additional calibration is required to improve the accuracy and stability of the algorithm [20]. Besides, the lookup table algorithm is also used to estimate air data states from measured surface

Manuscript received 21 August 2021; revised 20 June 2022; accepted 16 August 2022. Date of publication 26 August 2022; date of current version 12 April 2023.

DOI. No. 10.1109/TAES.2022.3201813

Refereeing of this contribution was handled by G. Fasano.

The work was supported in part by the Strategic Priority Research Program of Chinese Academy of Sciences under Grant XDA17030100, in part by the National Science and Technology Major Project of China under Grant ZX069, and in part by the National Natural Science Foundation of China under Grant 11902324.

Authors' addresses: Yang Liu is with the State Key Laboratory of High Temperature Gas Dynamics, Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Engineering Science, University of Chinese Academy of Sciences, Beijing 100049, China, E-mail: (liuyang2@imech.ac.cn). Chen-an Zhang and Wen Liu are with the State Key Laboratory of High Temperature Gas Dynamics, Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China, E-mail: (zhch a@imech.ac.cn; lw@imech.ac.cn). Xunshi Yan is with the Institute of Nuclear and New Energy Technology, Tsinghua University, Beijing 100084, China, E-mail: (yanxs@tsinghua.edu.cn). *(Corresponding author: Chen-an Zhang.)*

pressure distribution [21]. However, it needs a particular pressure port selection, and the algorithm requires an elaborate design to generate lookup tables. The accuracy and stability of the lookup table algorithm are at the cost of algorithm complexity.

With strong fitting capabilities, neural networks can approximate complex nonlinear relationships between the input and output variables of the system without requiring explicit prior knowledge and manual design that are very well suited to the FADS system [22], [23], [24]. Compared with the triples algorithm and the lookup table, neural networks have the advantage of being easier to develop while providing a higher level of detail in the mapping between the two vector spaces and are feasible for the sharp-nosed vehicles. Rohloff first proved that it is feasible to use the trained neural networks to represent the FADS estimation mapping [25], and discussed the fault tolerance and extrapolation stability of the neural-networks-based FADS system [26]. However, the training data was only sampled from a single flight profile, which limits the applicability of the trained neural networks to similar flight conditions. Then, Rohloff extracted a wide range of training data across the entire flight envelope from multiple flight tests, and built a neural-networks-based FADS system to estimate freestream static pressure and dynamic pressure [20]. Nevertheless, it is expensive and time-consuming that massive wind tunnel tests or flight tests are used to prepare vast bulk of training data for neural networks. Thus, most neural-network-based FADS systems are only applied to small unmanned aerial vehicles (UAVs) [27], [28], [29], [30] in recent years. These UAVs fly at low speed, and only the angle of incidence and velocity need to be estimated. Therefore, to expand the application of the neural-network-based FADS, it is necessary to explore new techniques to reduce the amount of training data for neural networks.

A feasible idea is to decouple the freestream static pressure, which is inspired by the traditional semiempirical FADS algorithm. Specifically, although the measured pressure values at different altitudes are different, the pressure distribution is similar between the flight conditions with the same Mach number and angle of incidence but different static pressure. Making full use of this similarity can effectively reduce the size of the training data set for the neural-networks-based FADS system. For example, the triples algorithm ingeniously defines a pressure transformation decoupling freestream static pressure [16]. However, this transformation is not feasible for sharp-nosed vehicles. Besides, the semiempirical pressure model is independent of the Mach number that makes Mach number, static pressure, and dynamic pressure need to be solved iteratively. Therefore, it is difficult to directly apply this transformation to neural networks.

In the basic theory of aerodynamics, choosing a reference to make the physical quantity dimensionless can profoundly reveal the essential relationship between various variables [31]. For example, freestream static pressure or dynamic pressure is often chosen as the reference to make the pressure quantity dimensionless, but they are the parameters to be solved in the FADS system. In Rohloff's research, static pressure and dynamic pressure can be calculated from the measured pressures by using neural networks [20]. It can be considered that the measured pressures can indirectly represent the static pressure and dynamic pressure. Therefore, an alternative solution is to use the measured pressure as the reference pressure.

To make full use of the collected information, this article uses each measured pressure as a reference to make the remaining pressures dimensionless, and an aerodynamic model is defined by dimensional analysis, which can decouple the freestream static pressure. A dimensionless input and output neural networks FADS (DIO-NNFADS) is developed to approximate the aerodynamic model, effectively reducing the training data scale for neural networks. The DIO-NNFADS is trained by less data, which is sampled from a single flight profile. Mach number, angle of attack, angle of sideslip, and pressure coefficient at each port are directly output by the DIO-NNFADS. Static pressure and dynamic pressure are solved by the equations composed of the measured pressures and pressure coefficients. Furthermore, this article explores the optimal size of the training data set for neural networks and discusses the applicability of this algorithm to various pressure port layouts.

The rest of this article is organized as follows. Section II proposes an implicit FADS aerodynamic model, and Section III designs a FADS algorithm to estimating air data states from the measured pressures. Then, Section IV describes the simplified supersonic model that is used to verify the designed FADS algorithm. Section V introduces the methods of data acquisition. Finally, Section VI concludes this article.

II. FLUSH AIR DATA SENSING AERODYNAMIC MODEL

The FADS system relies on the aerodynamic model that relates measured pressure distribution to the air data states. The semiempirical pressure model described by Whitmore has been validated for a high-performance fighter aircraft forebody and several simple shapes (for example, sphere and cylinder) [16]. However, the model suitable for a wide variety of vehicle shapes has not been established. Dimensional analysis is an effective tool to reveal the essential relationship between various variables that can help us analyze and establish the aerodynamic model and it is utilized to find a general aerodynamic model suitable for various vehicle shapes.

A. Derivation of Aerodynamic Model

With reference to the freestream static pressure p_∞ and dynamic pressure q_∞, the pressure of a port i on the surface of the vehicle can be expressed as

$$p_i = q_\infty C_{p_i} + p_\infty \tag{1}$$

where C_{p_i} is the pressure coefficient of the ith port.

The semiempirical pressure model described by Whitmore fuses potential flow and modified Newtonian flow to describe the pressure coefficient in terms of the angle of incidence [8]. Different from the semiempirical pressure model, C_{p_i} is the function of the spatial position of the ith port and Mach number M_∞, angle of attack α, and angle of sideslip β:

$$C_{p_i} = Cp(\boldsymbol{r_i}, M_\infty, \alpha, \beta) \tag{2}$$

where $\boldsymbol{r_i} = [x_i/l, y_i/l, z_i/l]$ is the spatial position of ith port, and l is the reference length of the vehicle. However, static pressure and dynamic pressure are the parameters to be solved and cannot be chosen as the reference pressure. Based on the measured pressures, Rohloff used two neural networks to estimate static pressure and dynamic pressure, respectively [20]. It can be expressed as

$$p_\infty = \boldsymbol{\Pi}_{\mathbf{p}_\infty}(p_1, p_2, \ldots, p_n) \tag{3a}$$

$$q_\infty = \boldsymbol{\Pi}_{\mathbf{q}_\infty}(p_1, p_2, \ldots, p_n) \tag{3b}$$

where $\boldsymbol{\Pi}_{\mathbf{p}_\infty}$ and $\boldsymbol{\Pi}_{\mathbf{q}_\infty}$ represent the neural networks used to estimate static pressure and dynamic pressure, respectively.

The measured pressures can be used to indirectly represent static pressure and dynamic pressure. Thus, an alternative solution is choosing a measured pressure as the reference pressure. For the ith and jth port, define the dimensionless pressure ratio ω_{ij}

$$\omega_{ij} = \frac{p_i}{p_j} = \frac{\frac{q_\infty}{p_\infty}Cp(\boldsymbol{r_i}, M_\infty, \alpha, \beta) + 1}{\frac{q_\infty}{p_\infty}Cp(\boldsymbol{r_j}, M_\infty, \alpha, \beta) + 1}. \tag{4}$$

According to the literature [8], $\frac{q_\infty}{p_\infty}$ is the function of M_∞

$$G(M_\infty) = \frac{q_\infty}{p_\infty} = \begin{cases} \left[1 + 0.2M_\infty^2\right]^{3.5} - 1 & (M_\infty > 1) \\ \dfrac{166.92M_\infty^7}{\left[7M_\infty^2 - 1\right]^{2.5}} - 1 & (M_\infty \le 1) \end{cases}. \tag{5}$$

Substitute (5) into (4), obtain

$$\omega_{ij} = \frac{G(M_\infty)Cp(\boldsymbol{r_i}, M_\infty, \alpha, \beta) + 1}{G(M_\infty)Cp(\boldsymbol{r_j}, M_\infty, \alpha, \beta) + 1}. \tag{6}$$

The above equation can be written as a general form

$$\omega_{ij} = f(\boldsymbol{r_i}, \boldsymbol{r_j}, M_\infty, \alpha, \beta). \tag{7}$$

This is an important result that ω_{ij}, the pressure ratio of two ports, is a function of $\boldsymbol{r_i}$, $\boldsymbol{r_j}$, M_∞, α, and β. For the measured pressures of n ports, at least $n-1$ independent pressure ratios can be defined. However, a question is which pressure port should be selected as a reference. To make full use of the collected information, we select each measured pressure in turn as a reference to make the remaining pressures dimensionless; $n(n-1)$ pressure ratios can be obtained as follows:

$$\begin{cases} \omega_{12} & = f(\boldsymbol{r_1}, \boldsymbol{r_2}, M_\infty, \alpha, \beta) \\ \omega_{13} & = f(\boldsymbol{r_1}, \boldsymbol{r_3}, M_\infty, \alpha, \beta) \\ & \vdots \\ \omega_{ij} & = f(\boldsymbol{r_i}, \boldsymbol{r_j}, M_\infty, \alpha, \beta) \\ & \vdots \\ \omega_{n(n-1)} & = f(\boldsymbol{r_n}, \boldsymbol{r_{n-1}}, M_\infty, \alpha, \beta) \end{cases} \tag{8}$$

where $i \le n$, $j \le n$, and $i \ne j$, $\boldsymbol{r_i}$ is the spatial position of the ith port. Once the aerodynamic configuration of the vehicle and the pressure port layout are determined, $\boldsymbol{r_i}$ is the known parameter. Define the pressure ratio array $\Omega = [\omega_{12}, \omega_{13}, \ldots, \omega_{ij}, \ldots, \omega_{n(n-1)}]$, which is only the function of M_∞, α, and β. Inversely, the dimensionless air data states, M_∞, α, and β, can be framed as an inverse mapping problem of (8). Assume $\boldsymbol{z_1} = [M_\infty, \alpha, \beta]$, the inverse form of the (8) can be expressed as follows:

$$\boldsymbol{z_1} = \boldsymbol{\Pi_1}(\omega_{12}, \omega_{13}, \ldots, \omega_{ij}, \ldots, \omega_{n(n-1)}) \tag{9}$$

$\boldsymbol{\Pi_1}$ is the implicit function that relates dimensionless air data states (M_∞, α, and β) and the pressure ratio array.

Once the parameters of M_∞, α, and β are determined, the pressure coefficient at each port can also be expressed as the function of the pressure ratio array. Combine (2) and (9), and define $\boldsymbol{z_2} = [C_{p_1}, C_{p_2}, \ldots, C_{p_i}, \ldots, C_{p_n}]$, to obtain

$$\boldsymbol{z_2} = \boldsymbol{\Pi_2}(\omega_{12}, \omega_{13}, \ldots, \omega_{ij}, \ldots, \omega_{n(n-1)}) \tag{10}$$

$\boldsymbol{\Pi_2}$ is the implicit function that relates the pressure coefficient at each port (C_{p_i}, $i = 1, 2, \ldots, n$) and the pressure ratio array.

According to (9) and (10), dimensionless air data states (M_∞, α, and β) and the pressure coefficient at each port (C_{p_i}, $i = 1, 2, \ldots, n$) can be defined as the function of the pressure ratio array. It can be considered as a general aerodynamic model with an implicit mathematical expression, which can be solved by neural networks based on observation data.

B. Discussion

By dimensional analysis, an implicit aerodynamic model is defined that can be used to estimate air data states. There are several important differences between the current aerodynamic model and the semiempirical pressure model. First, the pressure coefficient of the current aerodynamic model is a function of Mach number, angle of attack, and angle of sideslip, which is more consistent with the physics of flow. Mach number, angle of attack, and angle of sideslip can be inversely expressed as a function of the pressure ratio array so that they can be directly and accurately solved without complex calibration algorithms. However, the pressure coefficient of the previous semiempirical pressure model is only the function of angle of attack, angle of sideslip. As a result, Mach number needs to be solved iteratively. Second, referring to the measured pressure, the dimensionless parameter of pressure ratio has a simple form in mathematics. It can be regarded as a general form, which is suitable for various pressure port layouts, such as the port layout on the sharp-nosed vehicle. However, the transformation defined in the triples algorithm is exclusively for blunt-nosed vehicles. Finally, each measured pressure can be used as a reference to make the remaining pressures dimensionless, and $n(n-1)$ pressure ratios can be obtained to solve air data states. It can be considered as a pressure transformation that can make full use of the measured information.

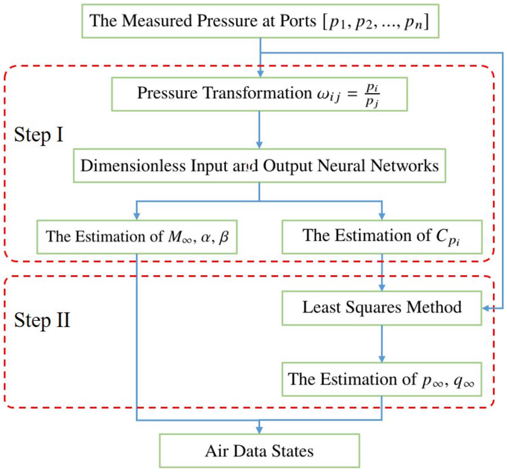

The Measured Pressure at Ports $[p_1, p_2, ..., p_n]$
Step I
Pressure Transformation $\omega_{ij} = \frac{p_i}{p_j}$
Dimensionless Input and Output Neural Networks
The Estimation of M_∞, α, β
The Estimation of C_{p_i}
Step II
Least Squares Method
The Estimation of p_∞, q_∞
Air Data States

Fig. 1. Flowchart of the proposed FADS estimation algorithm.

III. ALGORITHM

A. General Process for Estimating Air Data States

This section presents the general process of the proposed FADS algorithm, which can estimate air data states, such as the Mach number, angle of attack, angle of sideslip, static pressure, and dynamic pressure based on measured pressures on the vehicle. The flowchart is shown in Fig. 1, including two steps: 1) the estimations of dimensionless variables, and 2) the estimations of static pressure and dynamic pressure. For step 1), dimensionless air data states (M_∞, α, and β) and the pressure coefficient at each port (C_{p_i}, $i = 1, 2, \ldots, n$) are directly output by the DIO-NNFADS. For step 2), static pressure and dynamic pressure are solved by the equations composed of the measured pressures and pressure coefficients. With the above two steps, all air data states can be calculated.

B. Estimation of Dimensionless Air Data States

Mathematically, the FADS estimation can be regarded as a problem of system parameter identification, that is, the processing of using observations to estimate system parameters. Neural networks show dramatic advantages in system parameter identification, which is very well applied to the FADS system.

1) *Principle of Neural Networks:* As shown in Fig. 2, the structure of neural networks usually consists of three parts: The input layer, the hidden layers, and the output layer. The data accepted from the input layer are weighted and activated in each hidden layer. For the lth hidden layer, the result of the jth neuron is expressed as follows:

$$h_{l,j} = f\left(\sum_{i=1}^{I} h_{l-1,i} w_{ij}^{l} + b_j^l\right) \qquad (11)$$

$h_{l-1,i}$ is the data of the ith neuron of the $(l-1)$th layer, I is the number of nodes in the $(l-1)$th layer, w_{ij}^l is the weight connecting two adjacent layers, b_j^l is the bias, and $f(\cdot)$ is the LeakyReLU activation function [32].

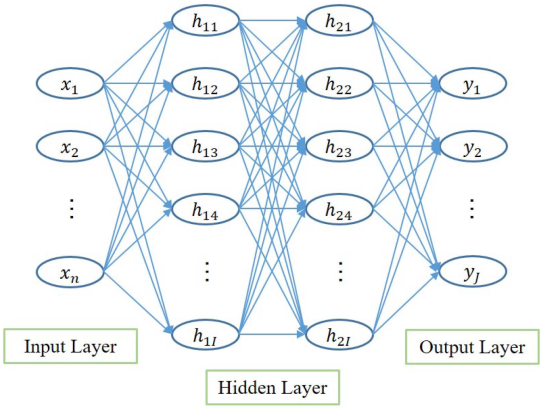


Fig. 2. Structure of artificial neural networks.

After multiple weighted and activated, the prediction $\hat{y}^{(j)}$ is given at the output layer. To avoid overfitting, the mean square error (MSE) loss function with the $L2$ regular penalty is defined

$$\text{Loss} = \frac{1}{2J}\sum_{j=1}^{J}\left(y^{(j)} - \hat{y}^{(j)}\right)^2 + \lambda \sum \|w\|_2^2 \qquad (12)$$

where J is the dimension of the output layer, $y^{(j)}$ is the true value of the samples, $\lambda \sum \|w\|_2^2$ is an $L2$-regularization term, and λ is a non-negative hyperparameter that controls the magnitude of the penalty.

2) *Dimensionless Input and Output Neural Networks:* An aerodynamic model defined by dimensional analysis is proposed in Section II-A. The model indicates that dimensionless air data states and pressure coefficient at each port are the function of the pressure ratio array transformed by the measured pressures. However, it is difficult to solve the explicit expression of the aerodynamic model. Thus, neural networks are used to infinitely approximate the aerodynamic model for estimating dimensionless air data states and pressure coefficient at each port.

According to the implicit expression of the (9) and (10), the input and output of neural networks can be determined. The input of neural networks is the pressure ratio array, and the output contains two parts: 1) Dimensionless air data states (M_∞, α, and β), 2) pressure coefficient at each port (C_{p_i}, $i = 1, 2, \ldots, n$). Since both the input and output of neural networks are dimensionless variables, the proposed estimation algorithm is called dimensionless input and output neural networks FADS (DIO-NNFADS).

Fig. 3 gives the details of DIO-NNFADS. First, n pressures are measured by the sensor array with n ports. The measured pressures are transformed into a dimensionless pressure ratio array as the input of neural networks. To fully dig out the information contained in the measured pressures, each measured pressure can be chosen as the reference, in turn, to make the remaining pressures dimensionless. A total of $n(n-1)$ pressure ratios can be obtained for the input of neural networks. Second, the pressure ratio array is transferred from the input layer to the hidden layers. Through the linear weighted and nonlinear activated in two

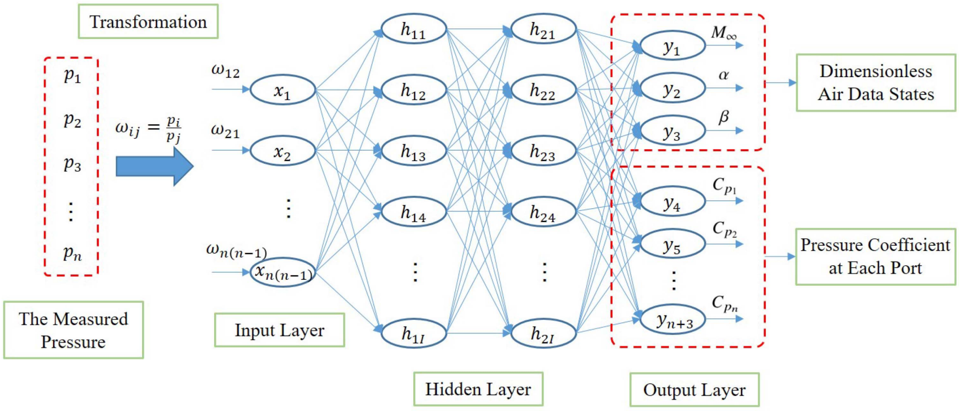
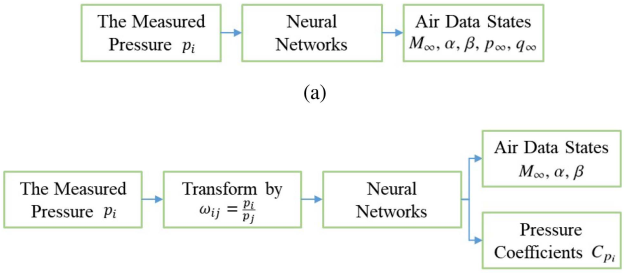


Fig. 3. Structure of dimensionless input and output neural networks.

TABLE I
Hyperparameters of Dimensionless Input and Output Neural Networks

Layer	Dimension
The measured pressures	n
Pressure ratio array	$n(n-1)$
Input layer	$n(n-1)$
Hidden layer 1	256
Hidden layer 2	256
Output layer	$n+3$


The Measured Pressure p_i
Neural Networks
Air Data States $M_\infty, \alpha, \beta, p_\infty, q_\infty$
The Measured Pressure p_i
Transform by $\omega_{ij} = \frac{p_i}{p_j}$
Neural Networks
Air Data States M_∞, α, β
Pressure Coefficients C_{p_i}

(a)

(b)

Fig. 4. Comparison between traditional neural-network-based FADS and the DIO-NNFADS. (a) Traditional neural-network-based FADS. (b) Dimensionless input and output neural networks FADS.

hidden layers with 256 nodes, a set of features are extracted. Finally, these extracted features are linear weighted to output dimensionless air data states and pressure coefficient at each port. A total of $n+3$ dimensionless variables are output. Table I gives the details of the DIO-NNFADS.

Fig. 4 gives the comparison of traditional neural-network-based FADS and the DIO-NNFADS. Traditional neural-network-based FADS directly establishes the mapping between the measured pressures and air data states, ignoring the similarity between the flight conditions with the same Mach number and the angle of incidence but different static pressure. Thus, large amounts of data across the entire flight envelop are required to train neural networks, which is costly in terms of extensive wind tunnel tests and flight tests. Different from traditional neural-network-based FADS, the input of DIO-NNFADS is the pressure ratio array transformed by the measured pressures, and the output contains dimensionless air data states and pressure coefficient at each port, as shown in Fig. 4(b). The essence of DIO-NNFADS is to approximate the aerodynamic model in Section II-A based on the observation data. The DIO-NNFADS has the capability of decoupling static pressure. Thus, it can be trained with less data from a single flight profile and is also suitable for the entire flight envelope.

C. Estimation of Static Pressure and Dynamic Pressure

Pressure coefficient at each port can be directly calculated by the DIO-NNFADS. Together with the measured pressures, only static pressure and dynamic pressure are still unknown parameters in (1). For the pressure sensor array with n ports, the resulting n overdetermined system (more equations than the unknown variables) of equations is

$$\begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix} = \begin{bmatrix} C_{p_1} & 1 \\ C_{p_2} & 1 \\ \vdots & \vdots \\ C_{p_n} & 1 \end{bmatrix} \begin{bmatrix} q_\infty \\ p_\infty \end{bmatrix}. \tag{13}$$

Define the matrix

$$M = \begin{bmatrix} C_{p_1} & 1 \\ C_{p_2} & 1 \\ \vdots & \vdots \\ C_{p_n} & 1 \end{bmatrix}. \tag{14}$$

Because the number of equations is always greater than that of unknown variables to be derived, p_∞ and q_∞ can be expressed by least square estimation. The result is

$$\begin{bmatrix} q_\infty \\ p_\infty \end{bmatrix} = \left[M^T M\right]^{-1} M^T \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix}. \tag{15}$$

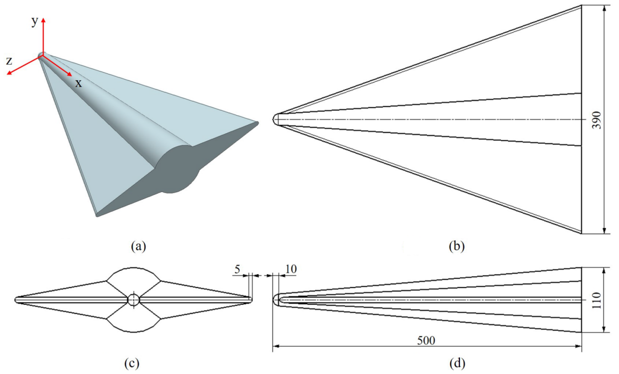


Fig. 5. Three-dimensional view of the simplified supersonic model (unit: mm).

The result can be further simplified as

$$p_\infty = \frac{-\left(\sum_{i=1}^{n} C_{p_i}\right)\left(\sum_{i=1}^{n} p_i C_{p_i}\right) - \left(\sum_{i=1}^{n} C_{p_i}^2\right)\left(\sum_{i=1}^{n} p_i\right)}{\left(\sum_{i=1}^{n} C_{p_i}^2\right) - \left(\sum_{i=1}^{n} C_{p_i}\right)^2} \tag{16a}$$

$$q_\infty = \frac{\sum_{i=1}^{n} p_i C_{p_i} - \left(\sum_{i=1}^{n} C_{p_i}\right)\left(\sum_{i=1}^{n} p_i\right)}{\left(\sum_{i=1}^{n} C_{p_i}^2\right) - \left(\sum_{i=1}^{n} C_{p_i}\right)^2}. \tag{16b}$$

Equations (16a) and (16b) are the working forms of the solver equations used in the FADS system.

D. Summary

With the above two steps, all the air data states can be estimated. Mach number, angle of attack, and angle of sideslip are directly output by DIO-NNFADS. Static pressure and dynamic pressure can be calculated by the equations composed of the measured pressures and pressure coefficients. Compared with the previous researches, the DIO-NNFADS has the following advantages.

1) The pressure ratio array with a simple transformation is applicable for various pressure port layouts. All measured pressures are used as a reference in turn, which can make full use of the collected pressure information and ensure the robustness of the DIO-NNFADS.
2) Compared with traditional neural-networks-based FADS, the DIO-NNFADS can decouple the freestream static pressure, greatly reducing the training data for neural networks.

IV. TEST VEHICLE GEOMETRY AND PORT LAYOUT

A. Vehicle Geometry

The proposed FADS algorithm is verified on a simplified supersonic model. As shown in Fig. 5, the nose of the vehicle is a hemisphere with a radius of 10 mm, followed by a conical fuselage. To meet the volume requirements, the wing is a delta wing with a certain thickness and has a leading-edge radius of 5 mm. The vehicle geometry is symmetric about the vertical meridian and lateral meridian, respectively. The vehicle is designed to fly in the range of altitude and Mach number being 10–25 km and 1.5–5. Table II describes the possible flight conditions. The range of corresponding static pressure is 2.5–26.5 kPa.

TABLE II
Target Flight Condition

Parameters	Range
Mach number	1.5 to 5
Static pressure (kPa)	20.5 to 26.5
Angle of attack (deg)	–6 to 20
Angle of sideslip (deg)	–6 to 6

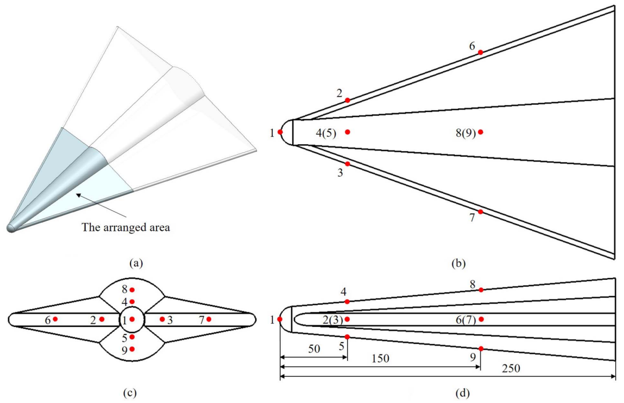


Fig. 6. Details of the pressure port matrix layout.

TABLE III
Location of Ports 1–9

No.	x (mm)	y (mm)	z (mm)
1	0	0	0
2	50	0	–24.7
3	50	0	24.7
4	50	–13.7	0
5	50	13.7	0
6	150	0	–61.9
7	150	0	61.9
8	150	–22.8	0
9	150	22.8	0

B. Port Layout

The ports of the traditional FADS algorithm are usually arranged on the nose of the vehicle. Due to the extremely limited space, it is difficult to arrange all ports on the nose. As shown in Fig. 6, a new port layout is arranged. This layout focuses on using the pressures in other parts of the vehicle to solve the air data states. Only one port is arranged on the model nose to collect the stagnation pressure. Eight ports are symmetrically arranged in $x = 50$ and 150 mm cross sections, respectively. Four ports are on the leading edge, and four ports are on the fuselage of the vehicle. A total of nine pressures can be collected at each flight condition. More details of these ports are listed in Table III.

TABLE IV
Summary of Computational Conditions

Parameters	Training data set		Test data set
	Subset 1	Subset 2	
Mach number	1.5, 2, 3, 4, 5	1.5, 2, 3, 4, 5	1.5 to 5
Static pressure (kPa)	12.1	2.5, 26.5	2.5 to 26.5
Angle of attack (deg)	-6, -4, -2, 0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20	20	–6 to 20
Angle of sideslip (deg)	-6, -4, -2, 0, 2, 4, 6	-6, -4, -2, 0, 2, 4, 6	–6 to 6
Num	490	70	300

V. NUMERICAL SIMULATION

A. Data Set Acquisition

Computational fluid dynamics (CFDs) simulation is used to obtain the pressure distribution at each flight condition. Two data sets are generated: Training data set and test data set. Training data set is used to learn the weights and bias of DIO-NNFADS, and the performance of the learned neural networks is examined by the test data set. As shown in Table IV, training data set contains two subsets: Subset 1 and 2. All flight conditions in subset 1 are calculated with the static pressure of 12.1 kPa and uniformly cover possible conditions in this flight profile. The range of angle of attack is -6° to 20° with an increment of 2°, and the angle of sideslip is -6° to 6° with an increment of 2°. The vehicle is operated at Mach number $M_\infty = [1.5, 2, 3, 4, 5]$. A total of 490 conditions are calculated. The purpose of subset 1 is to enable training data set to uniformly and completely cover a single FADS flight profile. It is known that the pressure coefficients at different altitudes are similar. However, flow separation may occur in some flight conditions at high angles of attack, which causes slight differences in the surface pressure coefficients between similar flight conditions. In addition, the appearance of separation may be different while fly altitudes vary, which may cause different influences on the surface pressure. These differences may reduce the estimation accuracy of the DIO-NNFADS. To further improve the performance of the DIO-NNFADS in such flight conditions, an additional training data set with a high angle of attack, subset 2, is generated. It only includes 70 flight conditions at $\alpha = 20^\circ$, but $p_\infty = 2.5$ and 26.5 kPa, corresponding to the lower and upper limits of the flight altitude. Therefore, a total of 560 flight conditions are calculated and the pressure at each port can be extracted for training data set.

As discussed in Section III-B, the DIO-NNFADS has the capability that the neural networks learned by a single flight profile can be generalized to the entire flight envelope. To evaluate the generalization of the DIO-NNFADS, the test data set, including 300 conditions, is randomly generated in the entire flight envelope.

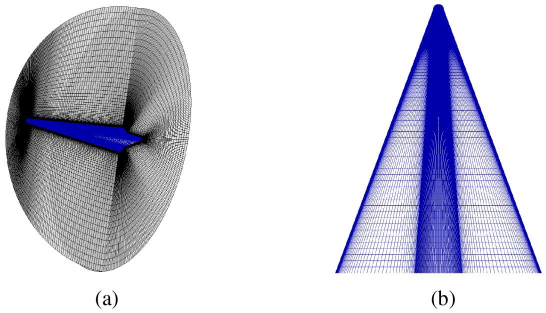


Fig. 7. Grid used in numerical simulation (approximately 5.09 million cells). (a) Spatial mesh. (b) Wall mesh.

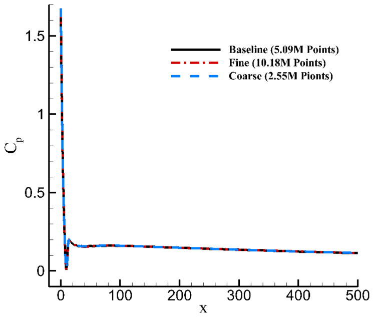


Fig. 8. Results of grid convergence study showing the influence of grid resolution on calculating the wall pressure.

B. Numerical Methods

The governing equations are the Reynolds-averaged Navier–Stokes equations. The turbulence model used in this study is Spalart–Allmaras one-equation model [33]. The AUSM+ spatial discretization scheme is adopted, with an implicit lower–upper symmetric Gauss–Seidel scheme for the temporal integration to accelerate convergence. More details about the CFD solver and its validation can be found in [34], [35].

C. Grid Independence Validation

A baseline grid with approximately 5.09 million cells is used, and the spatial and wall meshes are exhibited in Fig. 7. To ensure that the value of $y+$ is less than 1, the height of the first layer normal to the wall is 0.005 mm. Three grids with different resolutions are generated to demonstrate that the calculation results of CFD are independent of the grid. In addition to the baseline grid, the fine grid has 10.18 million cells, and the coarse has 2.55 million cells. Fig. 8 shows the pressure distribution on the vertical meridian of vehicle calculated with three different grid resolutions under flight condition of M_∞=2.5, $\alpha = 0^\circ$, and $\beta = 0^\circ$. The pressure distribution of the three different resolution grids is quite close and has no significant difference. Thus, the grid convergence with the baseline grid can be confirmed.

TABLE V
Estimation Results on the Training Data Set

Algorithm	Error	M_∞	p_∞	q_∞	α	β
NNFADS	Average error	0.5%	5%	1.1%	0.037°	0.020°
	Maximum error	9.7%	61.5%	12.2%	0.311°	0.286°
DIO-NNFADS	Average error	0.2%	0.4%	0.4%	0.016°	0.005°
	Maximum error	1.8%	3.3%	1.6%	0.255°	0.043°

TABLE VI
Estimation Results on the Test Data Set

Algorithm	Error	M_∞	p_∞	q_∞	α	β
NNFADS	Average error	16.1%	27.5%	4.7%	1.631°	0.368°
	Maximum error	77.2%	159.3%	26.6%	13.825°	2.739°
DIO-NNFADS	Average error	0.6%	1.3%	1.0%	0.078°	0.056°
	Maximum error	2.9%	6.2%	4.2%	0.350°	0.315°

VI. RESULTS AND DISCUSSION

A. Estimation Results

1) *Comparison With Traditional Neural-Network-Based FADS:* To demonstrate the superiority of the DIO-NNFADS, the NNFADS described in literature [25] is also used to analyze the same data set for comparison. The following points should be emphasized.

1) The input of the NNFADS is the measured pressures after normalization to the [0, 1] interval, and the air data states (M_∞, α, β, p_∞, and q_∞) are directly output.
2) The input of the DIO-NNFADS is the pressure ratio array transformed by the measured pressures. The output consists of two types: a) dimensionless air data states (M_∞, α, and β); b) pressure coefficient at each port (C_{p_i}, $i = 1, 2, \ldots, n$). Static and dynamic pressures are computed by the combination of the measured pressures and pressure coefficients.
3) Except for the input and output, the DIO-NNFADS and the NNFADS have the same network structure and the parameters, such as the same number of hidden layers and nodes, and the same learning rate, etc.

Table V gives the estimation results of two algorithms on the training data set. Both algorithms achieve good results for Mach number, angle of attack, and angle of sideslip which illustrates the powerful nonlinear fitting ability of neural networks. However, the DIO-NNFADS performs much better than the NNFADS in estimating static and dynamic pressures.

Table VI presents the estimation results on the test data set. For the estimation results of the DIO-NNFADS, the maximum relative errors for Mach number, static pressure, and dynamic pressure are 2.9%, 6.2%, and 4.2%, respectively, and the maximum error of the angle of incidence is no more than 0.4°. The results present that the DIO-NNFADS achieves high accuracy across the entire flight envelope. For the results of NNFADS, the maximum errors of all air data are beyond the accuracy required by the FADS system, which indicate that it is difficult for NNFADS to effectively estimate air data states based on such less training data.

Fig. 9 provides estimation results of the two algorithms on the test data set in detail. For the DIO-NNFADS, the relative errors of more than 95% calculation conditions are less than 2% in Mach number, 5% in static pressure, and 2.5% in dynamic pressure. The absolute errors of most calculation conditions are no more than 0.2° in the angle of incidence. However, the NNFADS shows good estimation results around the static pressure 12.1 kPa. As shown in Fig. 10, results with large errors are mainly distributed in the region away from this point. Obviously, that is because the training data set is primarily generated with a static pressure 12.1 kPa.

Next, the processing time of NNFADS and DIO-NNFADS is compared. The processing time can be divided into three parts: Preprocessing time, neural network solving time, and postprocessing time. For NNFADS, the preprocessing and postprocessing refer to the normalization of the pressure distribution and the antinormalization of the solved static and dynamic pressures, respectively. For DIO-NNFADS, the preprocessing means transforming the pressure distribution to pressure ratio array, and the postprocessing means solving static and dynamic pressures based on pressure distribution and pressure coefficients.

The two algorithms are used to process the same sample by Python 3.6 on a platform (Intel (R) Xeon (R) Gold 5218 Processor, 2.3 GHz), and the results are shown in Table VII. First, the neural network solving time of the two algorithms is very close because their network structures are identical except the inputs. Second, the preprocessing and postprocessing of DIO-NNFADS consumes more time compared to NNFADS. The total processing time of NNFADS and DIO-NNFADS is 0.131 and 0.246 ms, respectively, which are in the same order of magnitude.

2) *Influence of the Sensor Noise and Bias:* First, the influence of sensor noise on the accuracy of the proposed algorithm is considered. In the test data set, three samples with different altitudes of 10 km ($M_\infty = 4$, $\alpha = 11.5°$, and $\beta = -0.7°$), 15 km ($M_\infty = 3.5$, $\alpha = 4.4°$, and $\beta = 0.2°$), and 25 km ($M_\infty = 2.3$, $\alpha = -0.9°$, and $\beta = -3.3°$) are studied. The altitude of 15 km corresponds to the altitude of

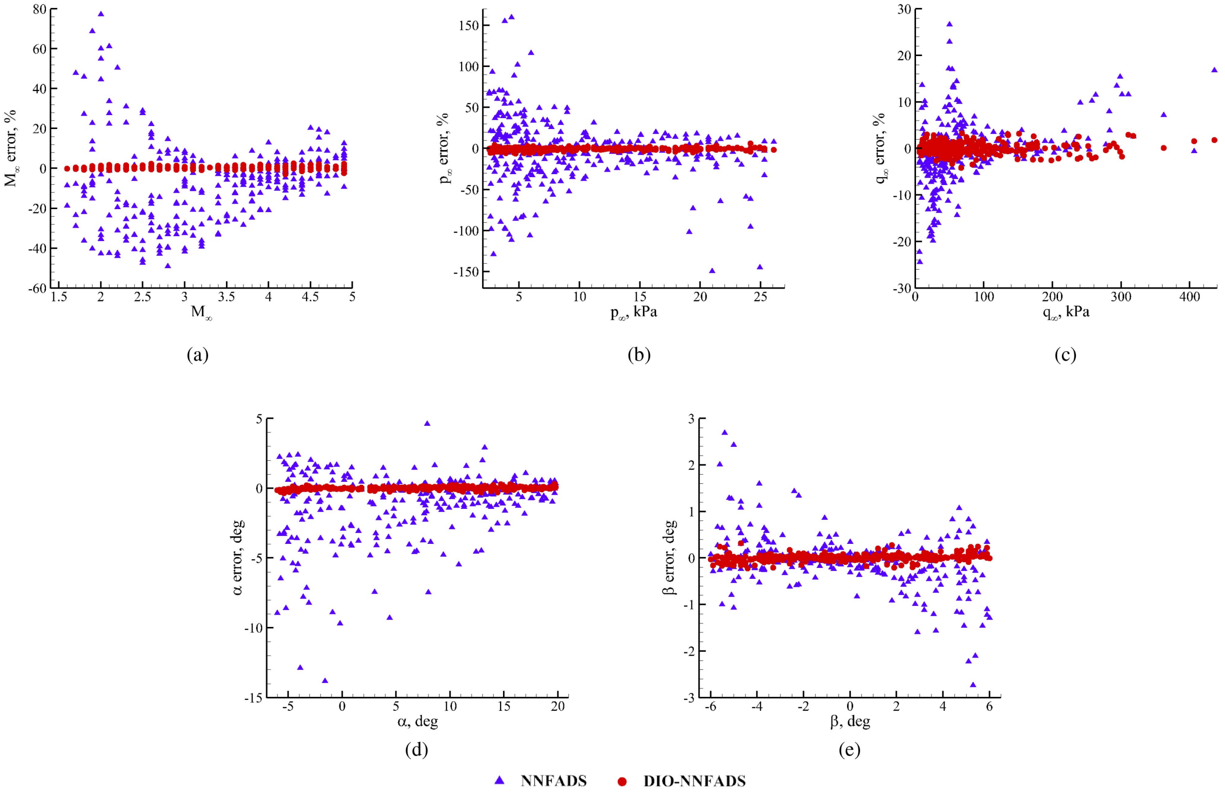


Fig. 9. Comparison of estimation errors between NNFADS and DIO-NNFADS. (a) As a function of Mach number. (b) As a function of static pressure. (c) As a function of dynamic pressure. (d) As a function of angle of attack. (e) As a function of angle of sideslip.

TABLE VII
Processing Time of the NNFADS and DIO-NNFADS

Algorithm	Pre-processing	Neural network solving	Post-processing	Total
NNFADS	0.012 ms	0.099 ms	0.020 ms	0.131 ms
DIO-NNFADS	0.053 ms	0.101 ms	0.092 ms	0.246 ms

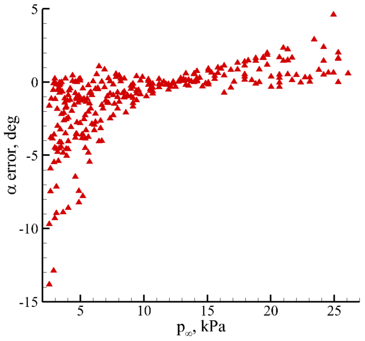
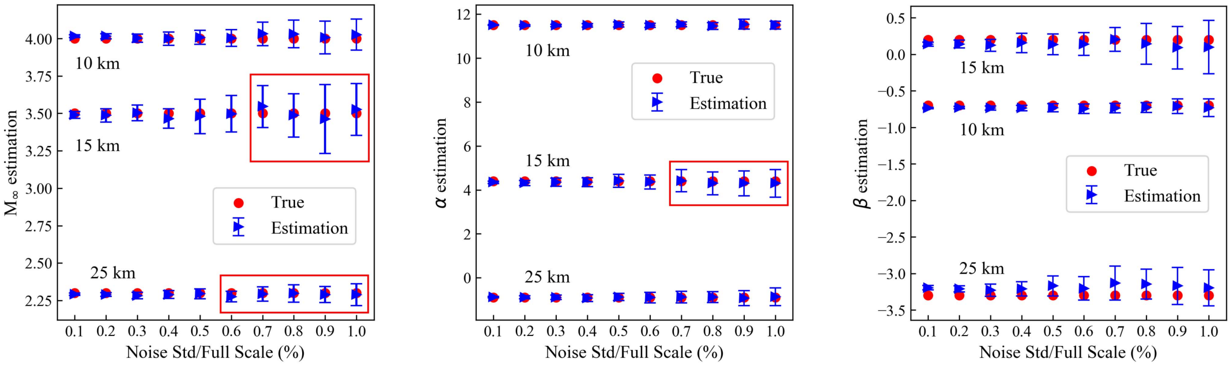


Fig. 10. Error of angle of attack under different static pressure (NNFADS, test data set).

generating training data, and 10 and 25 km correspond to the upper and lower limits of the flight envelope, respectively. Since the measured pressure values cover a range of three orders of magnitude, three types of pressure sensors with different end scale values (1000, 100, and 10 kPa) are equipped to measure pressures at the condition of 10, 15, and 25 km, respectively. Then, ten levels of noise are added to all the pressure signals at each condition, which conform to a normal distribution with the mean value of 0 and standard deviations of 0.1%, 0.2%, 0.3%, 0.4%, 0.5%, 0.6%, 0.7%, 0.8%, 0.9%, and 1% (corresponding to full scale of the sensor), respectively.

Assuming that the pressure signal consists of 30 sampling points, the algorithm outputs under different levels of noise are presented in Fig. 11. It can be seen that the stronger the noise, the larger the uncertainty of the estimation results. At the altitude of 10 km, the estimation results satisfy the requirements according to the accuracy requirements given in literature [8] (±5 percent accuracy for $2.5 \leq M_\infty \leq 5$, ±2.5 percent accuracy for $1.5 < M_\infty \leq 2.5$, $\pm 0.5^\circ$ absolute accuracy for angle of attack and sideslip). For the condition of 15 km, a larger uncertainty is displayed because the static pressure of the free flow is 12.1 kPa, which is at the lower limit of the sensor range (100 kPa). To make the estimations of Mach number and angle of attack meet the requirements, the noise standard deviation needs to be less than 0.7%. For the condition of 25 km, the estimation of Mach number requires the noise standard deviation of less than 0.6%. In addition, all results meet the accuracy requirements for angle of sideslip.



(a) (b) (c)

Fig. 11. Estimation results under different levels of noise (±0.2, ±0.175, and ±0.058 absolute accuracy for the condition of 25, 15, and 10 km, respectively, ±0.5° absolute accuracy for angle of attack and sideslip. Results exceeding the requirements are pointed out by red box). (a) Mach number. (b) Angle of attack. (c) Angle of sideslip.

TABLE VIII
Estimation Errors Under Different Biases

Port	Bias	M_∞ error			α error			β error		
		10km	15km	25km	10km	15km	25km	10km	15km	25km
1	-1.0%	1.7%	**5.8%**	**3.3%**	-0.035°	0.079°	0.082°	0.028°	0.044°	0.056°
	-0.5%	0.6%	3.1%	1.9%	-0.019°	0.071°	0.035°	0.030°	0.055°	-0.023°
	0.0%	-0.4%	0.3%	0.4%	-0.002°	0.061°	-0.010°	0.032°	0.064°	-0.101°
	0.5%	-1.5%	-2.6%	-1.0%	0.014°	0.049°	-0.054°	0.034°	0.069°	-0.181°
	1.0%	-2.5%	**-5.4%**	-2.4%	0.030°	0.033°	-0.095°	0.036°	0.071°	-0.261°
2	-1.0%	-0.7%	-0.6%	0.2%	-0.035°	0.004°	-0.016°	-0.005°	-0.099°	-0.205°
	-0.5%	-0.6%	-0.1%	0.3%	-0.019°	0.033°	-0.013°	0.013°	-0.016°	-0.153°
	0.0%	-0.4%	0.3%	0.4%	-0.002°	0.061°	-0.010°	0.032°	0.064°	-0.101°
	0.5%	-0.3%	0.7%	0.6%	0.014°	0.087°	-0.008°	0.050°	0.142°	-0.050°
	1.0%	-0.1%	1.1%	0.7%	0.029°	0.113°	-0.006°	0.068°	0.219°	0.001°
5	-1.0%	-0.5%	-0.3%	0.3%	0.064°	0.428°	0.201°	0.028°	0.014°	-0.094°
	-0.5%	-0.5%	0.0%	0.4%	0.031°	0.240°	0.093°	0.030°	0.040°	-0.097°
	0.0%	-0.4%	0.3%	0.4%	-0.002°	0.061°	-0.010°	0.032°	0.064°	-0.101°
	0.5%	-0.4%	0.5%	0.5%	-0.035°	-0.112°	-0.114°	0.033°	0.080°	-0.106°
	1.0%	-0.4%	0.6%	0.6%	-0.068°	-0.281°	-0.214°	0.035°	0.093°	-0.112°

[1] $M_\infty = 4$, $\alpha = 11.5°$, and $\beta = -0.7°$ for the condition of 10 km, $M_\infty = 3.5$, $\alpha = 4.4°$, and $\beta = 0.2°$ for 15 km, and $M_\infty = 2.3$, $\alpha = -0.9°$, and $\beta = -3.3°$ for 25 km.

[2] Results exceeding the system requirements are bolded.

Next the influence of individual sensor bias is discussed. Three typical sensors located at ports 1, 2, and 5 are studied. Port 1 is located at the stagnation point. Port 2 and 5 are located on the lateral and vertical meridian, respectively. Four biases are investigated: -1%, -0.5%, 0.5%, and 1% (corresponding to full scale of the sensor). As mentioned in the study of noise, the same three samples are tested and the results are shown in Table VIII. For the estimation of the angle of attack and sideslip, all results satisfy the requirements. Port 1 exhibits a large error in Mach number estimation and three results exceed the requirements, which indicates the importance of the stagnation point pressure for Mach number estimation. Port 2 has a larger error on angle of sideslip than angle of attack, illustrating that lateral meridian ports are important for estimating angle of sideslip. Similarly, port 5, located on the vertical meridian, is important for estimating angle of attack.

3) *Influence of Training Subset 2:* The training data set consists of two subsets: 1) subset 1 with the majority of conditions; 2) subset 2 with a small number of samples. Subset 2 is generated to further improve the accuracy of the DIO-NNFADS at high angles of attack. In this section, the DIO-NNFADS is individually trained by subset 1. Compared with the results of the entire training data set (subset 1 and 2), the influence of training data subset 2 can be analyzed.

Table IX presents the estimation results. Only trained by subset 1, the same level of average error is achieved. However, the maximum errors of Mach number, static pressure, dynamic pressure, and angle of attack increase. As shown in Fig. 12, the maximum errors appear at high angles of attack, and the maximum error of angle of attack increased from 0.35° to 0.565°. When the DIO-NNFADS is trained by the subset 1 and 2, the maximum error at high angles of attack is no more than 0.35°. With the above analysis, it can be found that the addition of subset 2 can improve the accuracy of the DIO-NNFADS at high angles of attack.

4) *Visualization:* Generally, as a black-box model, the operating mechanism of neural networks is difficult to understand. However, with the visualization of the parameters, some interesting phenomena can be somewhat explained.

TABLE IX
Estimation Results Under the Different Training Data Set

Traing data set	Error	M_∞	p_∞	q_∞	α	β
Subset 1	Average error	0.6%	1.6%	1.0%	0.085°	0.051°
	Maximum error	4.8%	13.5%	7.5%	0.565°	0.335°
Subset 1+2	Average error	0.6%	1.3%	1.0%	0.078°	0.056°
	Maximum error	2.9%	6.2%	4.2%	0.350°	0.315°

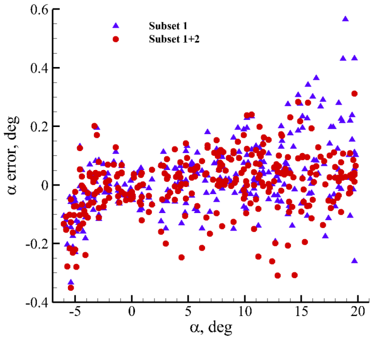
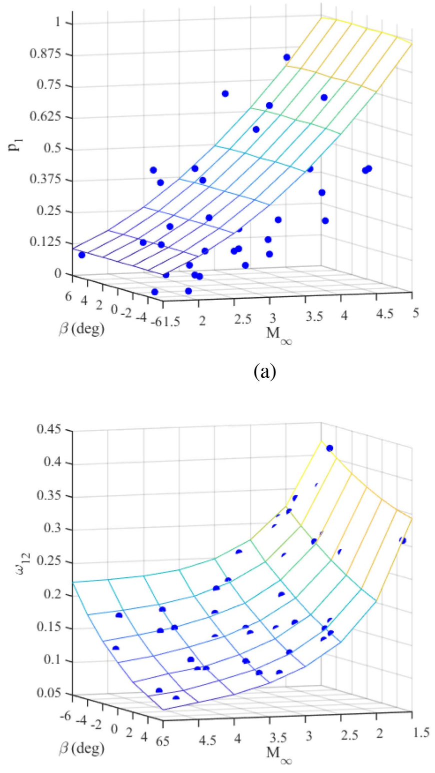


Fig. 12. Estimation error of angle of attack under the different training data set.



(a)

(b)

Fig. 13. Relationship between one input feature of neural networks and the flight conditions (the surface represents the training data at $\alpha = 0°$, and dots represent the test conditions at the small angle of attack). (a) NNFADS. (b) DIO-NNFADS.

An input of NNFADS and DIO-NNFADS at small angles of attack is extracted and then plotted, as shown in Fig. 13. It should be noted that training subset 1 is calculated under the condition with the static pressure of 12.1 kPa, which is presented as a surface in the figure. The test data is randomly generated in the entire flight envelop that is plotted as the blue dots. Fig. 13(a) shows the relationship between the normalization p_1 (one input of NNFADS) and flight conditions (M_∞ and β) at small angles of attack. The blue dots (represent the test data) are distributed over space in three dimensions and are away from the surface (composed of the training data). Fig. 13(b) plots the relationship between the ω_{12} (one input of DIO-NNFADS) and flight conditions. It can be seen that the blue dots are very close to the surface. The difference between the DIO-NNFADS and the NNFADS is vividly shown by the visualization. Although the freestream static pressure is different at different altitudes, the pressure ratio is independent of static pressure. Therefore, based on the less data from a flight profile, the trained DIO-NNFADS shows good generalization across the entire flight envelope. However, when the measured pressure is directly used as the input of the neural network, the training data set should cover the entire flight envelope.

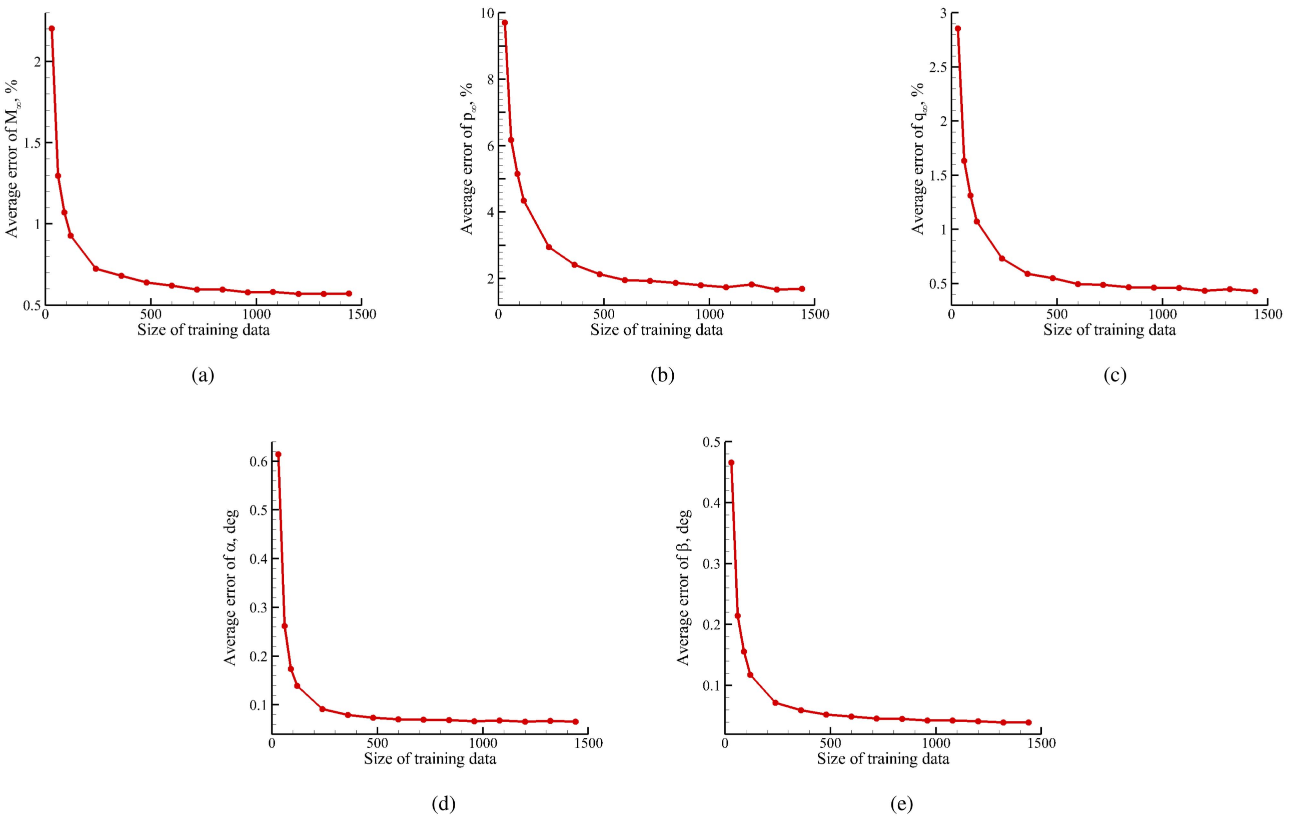
B. Influence of the Size of Training Data Set

The DIO-NNFADS can be trained by less data from one flight profile data, and then generalize to the entire flight envelope. A question needs to be answered is that how much training data need to be prepared for DIO-NNFADS. To answer this question, a denser training data set similar to the training data set in Section V-A but containing more flight conditions is prepared. As shown in Section V-A, the training data set also contains two subsets. Subset 1 includes 1296 calculation conditions under the static pressure of 12.1 kPa. Different from the training data set in Table X, more conditions with different angles of incidence and Mach numbers are calculated. The calculation conditions of angle of attack are the intersection of the increment of 2° and the increment of 3°. The same is true for the angle of sideslip. The range of Mach number is 1.5–5 with an increment of 0.5. Subset 2 contains 144 calculation conditions with high angles of attack. Finally, a total of 1440 conditions are calculated in the denser training data set.

Based on the denser training data set, 15 subsets with different sizes are randomly selected. The number of conditions included in these 15 subsets are 30, 60, 90, 120, 240, 480, 600, 720, 840, 960, 1080, 1200, 1320, and 1440, respectively. DIO-NNFADS is used to analyze these 15 subsets to find the optimal size of training data. Since these subsets are randomly selected, the same experiments are repeated 20 times. To ensure the reliability of the conclusion, the average results of these 20 experiments are used for analysis.

TABLE X
Computational Conditions of the Denser Training Data Set

Parameters	Subset 1	Subset 2
Mach number	1.5, 2, 2.5, 3, 3.5 4, 4.5, 5	1.5, 2, 2.5, 3, 3.5 4, 4.5, 5
Static pressure (kPa)	12.1	2.5, 26.5
Angle of attack (deg)	-6, -4, -3, -2, 0, 2, 3, 4, 6, 8, 9, 10, 12, 14, 15, 16, 18, 20	20
Angle of sideslip (deg)	-6, -4, -3, -2, 0, 2, 3, 4, 6	-6, -4, -3, -2, 0, 2, 3, 4, 6
Num	1296	144



(a)

(b)

(c)

(d)

(e)

Fig. 14. Relationship between the average error and the size of training data. (a) Convergence curve for Mach number. (b) Convergence curve for static pressure. (c) Convergence curve for dynamic pressure. (d) Convergence curve for angle of attack. (e) Convergence curve for angle of sideslip.

Fig. 14 gives the trend of the average error as the size of training data increases. It can be seen that the trends of all air data states are similar. When the size of training data is less than 500, the average error is greatly reduced as the size of training data increases. But when the size of training data is more than approximate 500, the average error no longer decreases. Therefore, the size of the training data set in this study is 560. Note that when the size of the training data set is 300, the average error is less than 0.7% in Mach number, 3% in the static pressure and dynamic pressure, and 0.1° in the angle of incidence, which also shows good estimation results and is deemed sufficient for a FADS application.

C. Influence of Port Layout and Selection

The layouts and position of the pressure ports affect the estimation accuracy. Several different port layouts are built to demonstrate the robustness of this estimation algorithm.

1) *Influence of Port Selection:* As listed in Table XI, in addition to layout A, five port layouts lacking one or more ports are built. Layouts B and C are the minimum number of pressure ports for the FADS system. Layout D contains the ports on the leading edge and the nose, and layout E on the fuselage and nose. Layout F includes all pressure ports except the port on the nose. Through the estimation results, the importance of each component can be demonstrated.

TABLE XI
Pressure Port Selections for Dimensionless Neural Networks

Layouts	Port selection	Description
A	1, 2, 3, 4, 5, 6, 7, 8, 9	Port layout used in this study
B	1, 2, 3, 4, 5	The minimum number of pressure ports
C	1, 6, 7, 8, 9	The minimum number of pressure ports
D	1, 2, 3, 6, 7	Ports located on the lateral meridian
E	1, 4, 5, 8, 9	Ports located on the vertical meridian
F	2, 3, 4, 5, 6, 7, 8, 9	Port layout without stagnation point

Table XII gives the estimation results of these port layouts. Both layouts B and C can estimate all air data states independently, which meet the accuracy requirements of the FADS system. But the accuracy of layout C is higher than layout B. Layouts D and E do not meet the requirements, and by comparing the results of the two layouts, some

TABLE XII
Average Errors Under Different Pressure Port Selections

Layouts	M_∞	p_∞	q_∞	α	β
A	0.6%	1.3%	1.0%	0.078°	0.056°
B	1.4%	2.9%	1.6%	0.108°	0.068°
C	0.6%	1.5%	1.0%	0.091°	0.055°
D	2.5%	4.9%	3.4%	5.117°	0.114°
E	2.0%	4.1%	1.4%	0.117°	2.874°
F	0.8%	1.4%	1.3%	0.068°	0.072°

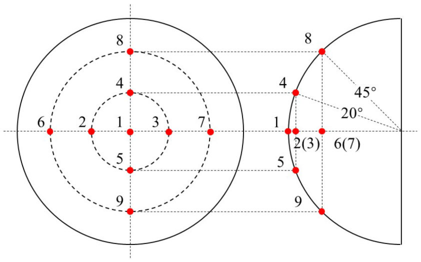


Fig. 15. Port layout G on hemisphere head of the vehicle.

TABLE XIII
Estimation Result With Different Pressure Port Layouts

Layouts	M_∞	p_∞	q_∞	α	β
A	0.6%	1.3%	1.0%	0.078°	0.056°
G	2.8%	5.4%	0.5%	0.061°	0.039°

conclusions can be drawn. Layout E shows good results while layout D shows poor results on the estimation of angle of attack, demonstrating the importance of the vertical meridian ports for estimating angle of attack. Similarly, the lateral meridian ports are very important for estimating angle of sideslip. Such phenomenon is consistent with the results analyzed according to the sensor bias influence.

It is worth emphasizing that layout F shows good estimation results and its accuracy is slightly lower than layout A. It illustrates that the proposed FADS algorithm can accurately estimate all air data states without stagnation pressure. The stagnation point of a hypersonic vehicle often has a tremendous heat load, which puts forward high requirements for the measurement of stagnation point pressure. The nonstagnation port layout can effectively avoid severe heat load and reduce the restrictions on the pressure sensors. It has important implications for the FADS system of hypersonic vehicles.

2) *Comparison With Traditional Port Layout:* Although it is difficult to arrange pressure ports on a hemisphere nose with a radius of 10 mm for an actual hypersonic vehicle, the pressure distribution can be accurately obtained using CFD simulation. Port layout G, a traditional port layout on the hemisphere nose, is also arranged for comparison. As shown in Fig. 15, nine ports are arranged on the vehicle nose, and nine pressures are collected. The DIO-NNFADS is developed to analyze the collected data.

Table XIII presents the results of layout G. The average error of the layout G for Mach number and static pressure is higher than the layout A. But the opposite results are exhibited in the estimation for dynamic pressure. For the traditional port layout G, large errors exist in the region at large Mach numbers, as shown in Fig. 16. It can be explained by Fig. 17(a), which plots the pressure coefficient at port 7 under different Mach numbers. When the Mach number is below 3.5, the pressure coefficient increases rapidly with the increase of the Mach number. But when the Mach number exceeds 3.5, the rate of increase diminishes considerably and the pressure coefficient appears to reach a plateau. It causes neural networks to have a low resolution for high Mach numbers. Small disturbances in the input of neural networks could cause large errors in the estimation of Mach number. As shown in Fig. 17(b), when the port is located on the leading edge, the pressure coefficient increases with the increase of Mach number, and neural networks can clearly distinguish the Mach number. According to literature [36], the blunt-body flows tend to approach Mach number independence at lower Mach number than slender bodies. Hence, arranging more pressure ports on the slender fuselage instead of the blunt nose is helpful to estimate the

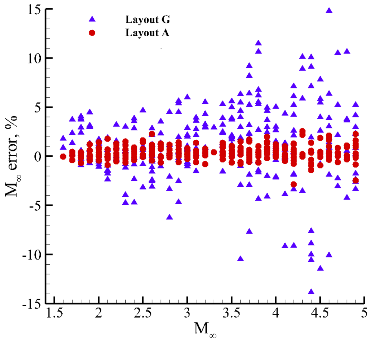
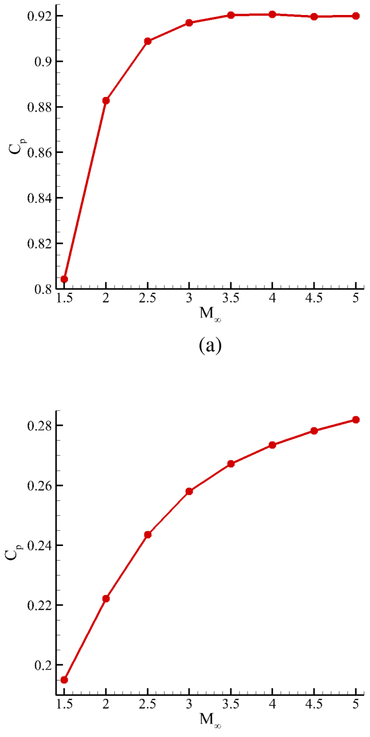


Fig. 16. Mach relative error under different port layouts.



(a)

(b)

Fig. 17. Pressure coefficient of port 7 ($\alpha = 4°$, $\beta = 4°$). (a) Layout G. (b) Layout A.

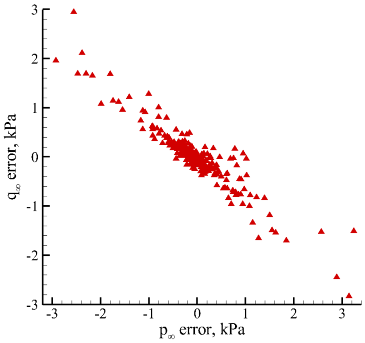


Fig. 18. Relationship between static pressure error and dynamic pressure error (layout G).

Mach number. That is the reason why the Mach number error of layout G is smaller than that of layout A. The same reason can explain the large static pressure error of layout G. However, the errors of dynamic pressure exhibit good estimation results. As shown in Fig. 18, the error characteristics of the static pressure and dynamic pressure are consistent. Due to the high dynamic pressure under high Mach number, the relative error of dynamic pressure is small, which is presented in Table XIII.

VII. CONCLUSION

A new FADS algorithm based on dimensionless input and output neural networks is developed for estimating air data states. Dimensional analysis defines a pressure ratio array that can be regarded as a pressure transformation to solve air data states. Besides, a general aerodynamic model with a simple form is also analyzed and defined. The aerodynamic model can decouple freestream static pressure, and dimensionless input and output neural networks are used to approximate the model. Mach number, angle of attack, angle of sideslip, and pressure coefficients are directly output by dimensionless input and output neural networks. Static pressure and dynamic pressure are obtained by solving the equations composed of the measured pressures and pressure coefficients. Unlike traditional neural-network-based FADS, dimensionless input and output neural networks can decouple the freestream static pressure, effectively reducing the size of training data. The proposed FADS algorithm is verified on a simplified supersonic model. Trained by less data from a single flight profile, the proposed FADS algorithm shows good accuracy and generalization performance across the entire flight envelope. The maximum error of Mach number is less than 2.9%, static and dynamic pressures are less than 6.2%, and the angle of incidence is less than 0.4°. Besides, the optimal size of the training data is also discussed. Furthermore, the proposed algorithm also shows good performance for various pressure port layouts, such as the port layout without stagnation point.

REFERENCES

[1] B. R. Cobleigh, S. A. Whitmore, E. A. Haering, J. Borrer, and V. E. Roback, “Flush air data sensing (FADS) system calibration procedures and results for blunt forebodies,” NASA TP-1999-209012, Nov. 1999.

[2] A. Wenz and T. A. Johansen, “Moving horizon estimation of air data parameters for UAVs,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 3, pp. 2101–2121, Jun. 2020.

[3] S. A. Whitmore, “Reconstruction of the shuttle reentry air data parameters using a linearized Kalman filter,” American Institute of Aeronautics and Astronautics, AIAA Paper 1983-0118, Aug.1983.

[4] S. A. Whitmore, R. J. Davis, and J. M. Fife, “In-flight demonstration of a real-time flush airdata sensing system,” *J. Aircr.*, vol. 33, no. 5, pp. 970–977, 1996.

[5] M. B. Rhudy, M. L. Fravolini, Y. Gu, M. R. Napolitano, S. Gururajan, and H. Chao, “Aircraft model-independent airspeed estimation without pitot tube measurements,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 51, no. 3, pp. 1980–1995, Jul. 2015.

[6] S. A. Whitmore, T. R. Moes, and T. J. Larson, “Preliminary results from a subsonic high angle-of-attack flush airdata sensing (HI-FADS) system: Design, calibration, and flight test evaluation,” American Institute of Aeronautics and Astronautics, AIAA Paper 1990-0232, 1990.

[7] S. A. Whitmore, T. R. Moes, and T. J. Larson, “High angle-of-attack flush airdata sensing system,” *J. Aircr.*, vol. 29, no. 5, pp. 915–919, 1992.

[8] S. A. Whitmore, B. R. Cobleigh, and E. A. Haering, “Design and calibration of the X-33 flush airdata sensing (FADS) system,” NASA/TM-1998-206540, 1998.

[9] P. M. Siemers, I. Wolf, and P. F. Flanagan, “Shuttle entry air data system concepts applied to space shuttle orbiter flight pressure data to determine air data–STS 1-4,” , American Institute of Aeronautics and Astronautics, AIAA Paper 1983-0118, 1983.

[10] C. D. Pruett, H. Wolf, M. L. Heck, and P. M. Siemers, “An innovative air data system for the space shuttle orbiter: Data analysis techniques,” American Institute of Aeronautics and Astronautics, AIAA Paper 1981-2455, 1981.

[11] J. P. Cary and E. R. Keener, “Flight evaluation of the X-15 ball-nose flow-direction sensor as an air-data system,” NASA-TN-D-2923, 1965.

[12] S. Weiss, “Comparing three algorithms for modeling flush air data systems,” in *Proc. 40th AIAA Aerosp. Sci. Meeting Exhibit*, 2002, Art. no. 535.

[13] E. Baumann, J. W. Pahle, M. C. Davis, and J. T. White, “The X-43A flush airdata sensing system flight-test results,” *J. Spacecr. Rockets*, vol. 47, no. 1, pp. 48–61, 2010.

[14] R. Takaki, M. Takizawa, R. Takaki, and M. Takizawa, “ADS measurement of HYFLEX (hypersonic flight experiment),” in *Proc. 35th Aerosp. Sci. Meeting Exhibit*, 1997, Art. no. 193.

[15] S. Theil, M. Schlotterer, M. Conradt, M. Hallmann, M. Markgraf, and I. Vanschoenbeek, “Hybrid navigation system for the SHEFEX-2 mission,” in *Proc. AIAA Guid., Navigation Control Conf. Exhibit*, 2008, Art. no. 6991.

[16] S. A. Whitmore, T. R. Moes, M. W. Czerniejewski, and D. A. Nichols, “Application of a flush airdata sensing system to a wing leading edge (LE-FADS),” in *Proc. 31st Aerosp. Sci. Meeting*, 1993, Art. no. 634.

[17] M. C. Davis, J. W. Pahle, J. W. White, L. Marshall, M. Mashburn, and R. Franks, “Development of a flush airdata sensing system on a sharp-nosed vehicle for flight at mach 3 to 8,” , American Institute of Aeronautics and Astronautics, AIAA Paper 2000-0504, 2000.

[18] P. Wang, X. Jin, and X. J. Zhao, “Application of the FADS system to hypersonic flight vehicles with sharp wedged fore-bodies,” in *Proc. 18th AIAA/3AF Int. Space Planes Hypersonic Syst. Technol. Conf.*, 2012, Art. no. 5898.

[19] J. C. Ellsworth, “An analytical explanation for the X-43A flush air data sensing system pressure mismatch between flight and theory,” American Institute of Aeronautics and Astronautics, AIAA Paper 2010-4964, 2010.

[20] T. J. Rohloff, S. A. Whitmore, and I. Catton, “Air data sensing from surface pressure measurements using a neural network method,” *AIAA J.*, vol. 36, no. 11, pp. 2094–2094, 1998.

[21] H. Takahashi, M. Kodera, and K. Tani, “Flush air data sensing system for a sharp-nosed hypersonic vehicle with curved-wedge forebody,” *AIAA J.*, vol. 58, no. 11, pp. 4819–4831, 2020.

[22] P. Courrieu, “Three algorithms for estimating the domain of validity of feedforward neural networks,” *Neural Netw.*, vol. 7, no. 1, pp. 169–174, 1994.

[23] A. Calia, E. Denti, R. Galatolo, and F. Schettini, “Air data computation using neural networks,” *J. Aircr.*, vol. 45, no. 6, pp. 2078–2083, 2008.

[24] M. M. Oliveira, G. S. Mayor, J. P. Macedo, and J. H. Bidinotto, “Neural networks to classify atmospheric turbulence from flight test data: An optimization of input parameters for a generic model,” *J. Braz. Soc. Mech. Sci. Eng.*, vol. 44, no. 3, pp. 1–11, 2022, doi: 10.1007/s40430-022-03386-1.

[25] T. J. Rohloff and I. Catton, “Development of a neural network flush air data sensing system,” in *Proc. ASME Int. Mech. Eng. Congr. Expo.*, Fairfield, NJ, USA, 1996, vol. 242, pp. 39–43.

[26] T. J. Rohloff, S. A. Whitmore, and I. Catton, “Fault-tolerant neural network algorithm for flush air data sensing,” *J. Aircr.*, vol. 36, no. 3, pp. 541–549, 1999.

[27] I. Samy, I. Postlethwaite, D.-W. Gu, and J. Green, “Neural-network-based flush air data sensing system demonstrated on a mini air vehicle,” *J. Aircr.*, vol. 47, no. 1, pp. 18–31, 2010.

[28] J. Quindlen and J. Langelaan, “Flush air data sensing for soaring-capable UAVs,” in *Proc. 51st AIAA Aerosp. Sci. Meeting Including New Horiz. Forum Aerosp. Expo.*, 2013, Art. no. 1153.

[29] T. Islam, I. Martin, and D. Moormann, “Design of an active gust load alleviation system for small UAS using a flush airdata sensing system,” in *Proc. Atmospheric Flight Mechanics Conf.*, 2018, Art. no. 2834.

[30] K. T. Borup, T. I. Fossen, and T. A. Johansen, “A machine learning approach for estimating air data parameters of small fixed-wing UAVs using distributed pressure sensors,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 3, pp. 2157–2173, Jun. 2020.

[31] Q. M. Tan, *Dimensional Analysis*. Berlin, Germany:Springer, 2011.

[32] X. J. Jin, C. Y. Xu, J. S. Feng, Y. C. Wei, J. J. Xiong, and S. C. Yan, “Deep learning with S-shaped rectified linear activation units,” in *Proc. AAAI Conf. Artif. Intell.*, 2015, pp. 1737–1743.

[33] P. R. Spalart and S. R. Allmaras, “A one-equation turbulence model for aerodynamic flows,” American Institute of Aeronautics and Astronautics, AIAA Paper 1992-0439, 1992.

[34] W. Liu, C. A. Zhang, H. Q. Han, and F. M. Wang, “Local piston theory with viscous correction and its application,” *AIAA J.*, vol. 55, pp. 942–954, 2017.

[35] W. Liu, C. A. Zhang, F. M. Wang, and Z. Y. Ye, “Design and optimization method for hypersonic quasi-waverider,” *AIAA J.*, vol. 58, pp. 2132–2146, 2020.

[36] J. D. Anderson, *Fundamentals of Aerodynamics*, 5th ed. Maidenhead, U.K.: McGraw-Hill, 2005.

Yang Liu received the B.S. degree in aircraft design and engineering from Northwestern Polytechnical University, Xi’an, China. He is currently working toward the Ph.D. degree in general mechanics and fundamentals of mechanics with the School of Engineering Science, University of Chinese Academy of Sciences and Institute of Mechanics, Chinese Academy of Sciences, Beijing, China.

His research interests include air data sensing system and physics-informed neural networks.

Chen-an Zhang received the Ph.D. degree in fluid–solid interaction and control from Northwestern Polytechnical University, Xi’an, China, in 2010.

He is currently a Professor with the Institute of Mechanics, Chinese Academy of Sciences, Beijing, China. His research interests include aerodynamics, aeroelasticity, and fault diagnosis.

Xunshi Yan received the Ph.D. degree in control science and engineering from Tsinghua University, Beijing, China, in 2013.

He is currently an Associate Research Professor with the Institute of Nuclear and New Energy Technology, Tsinghua University, Beijing, China. His research interests include machine learning, fault diagnosis, and active magnetic bearing.

Wen Liu received the Ph.D. degree in aerodynamics from Northwestern Polytechnical University, Xi’an, China, in 2018.

He is currently an Assistant Professor with the Institute of Mechanics, Chinese Academy of Sciences, Beijing, China. His research interests include hypersonic aerodynamics, flight dynamics, and machine learning.