Complete Online Algorithm for Air Data System Calibration

Juan D. Jurado*
Air Force Institute of Technology, Wright–Patterson Air Force Base, Ohio 45433
and
Clark C. McGehee†
U.S. Air Force Flight Test Center, Edwards Air Force Base, California 93524

DOI: 10.2514/1.C034964

Air data systems require costly calibration of their static pressure sensors to characterize errors caused by the act of flying. Altitude-based methods for measuring these so-called static position errors, such as the tower flyby, produce accurate results but require an elaborate flyby site, multiple experiments to capture the relationship between error and airspeed, and are limited to subsonic airspeeds due to inherent hazards to land-based and aircraft structures from low-altitude supersonic flight. Airspeed-based methods using the Global Positioning System (GPS) are generally easier to execute, but they tend to yield less precise results and still require multiple experiments. Additionally, they require temperature probe calibration from external sources. This paper proposes a self-contained online method for complete air data calibration. The proposed method uses a Kalman smoother to fuse GPS altitude and airspeed measurements, aircraft attitude, and air data to produce the full static position error curve as a function of Mach number in a single experiment, with no need for external temperature calibration and with no supersonic limitations. The proposed method is validated using T-38C flight data, and it is shown to reduce cost by 88% while modeling a 42% larger domain when compared to current methods.

Nomenclature

a	=	test-temperature speed of sound, ft/s
a_{SL}	=	sea-level speed of sound, ft/s
b_j	=	jth temperature optimization coefficient
C_b^n	=	body-frame to navigation-frame rotation matrix
C_w^b	=	wind-frame to body-frame rotation matrix
H_c	=	calibrated altitude, ft
H_{ic}	=	instrument-corrected altitude, ft
h_g	=	geometric altitude, ft
$\bar{h}_g$	=	mean geometric altitude, ft
K_t	=	temperature recovery factor
M_{ic}	=	instrument-corrected Mach number
M_{pc}	=	position-corrected Mach number, M
P	=	total number of smoothing spline knots
P_a	=	ambient pressure, psi
P_s	=	measured static pressure, psi
P_T	=	total pressure, psi
P_0	=	ambient pressure linearization point, psi
S	=	number of data samples
T_a	=	ambient temperature, K
T_{ic}	=	total temperature, K
T_{SL}	=	sea-level standard temperature, K
T_{std}	=	test-altitude standard temperature, K
$\boldsymbol{v}_g$	=	ground velocity vector, ft/s
v_j	=	jth ground speed component (north, east, down), ft/s
v_T	=	true airspeed, kt
$\boldsymbol{v}_W$	=	wind velocity vector, ft/s
v_{W_j}	=	jth wind speed component (north, east down), ft/s
$\boldsymbol{v}_T^n$	=	true velocity vector in the navigation frame, ft/s
$\boldsymbol{v}_T^w$	=	true velocity vector in the wind frame, ft/s
α_c	=	corrected angle of attack, rad
α_i	=	indicated angle of attack, rad
β_c	=	corrected angle of sideslip, rad
β_i	=	indicated angle of sideslip, rad
$\hat{\beta}_j$	=	jth model regression coefficient
γ	=	flight-path angle, rad
γ_v	=	true airspeed scale factor
ΔH_{pc}	=	altitude position error correction, ft
ΔP_p	=	static position error, psi
ΔV_T	=	true airspeed error, ft/s
$\Delta\alpha$	=	angle-of-attack correction, rad
δ	=	pressure ratio
Θ	=	pitch angle, rad
θ	=	temperature ratio
Φ	=	roll angle, rad
ϕ	=	ground track, rad
Ψ	=	yaw angle or true heading, rad

I. Introduction

PRODUCTION aircraft are typically equipped with a pitot-static sensor system, which is sometimes referred to as an air data system (ADS). The ADS is composed of a pitot tube, which measures the total air pressure; a static port, which measures static air pressure; and an air data computer (ADC), which combines the sensor readings into various airspeed and altitude instrument readings. The ADC uses pitot-static relationships to convert differences between the total and static pressures into airspeed readings, as well as static pressure measurements into altitude readings. Because airspeed and altitude are directly derived from pressure, they are intrinsically linked to lift and drag that, in turn, are linked to key performance parameters such as the climb rate, climb angle, specific range, and endurance. Unfortunately, the act of flying through an air mass inherently corrupts the static port's ability to measure ambient pressure, or the true static pressure in the undisturbed atmosphere, and creates an error called the static position error (SPE) [1].

The SPE, or ΔP_p, is defined as the difference between the static pressure P_s and the ambient pressure P_a; and it is often normalized by P_s when comparing readings from various flight conditions using

$$\frac{\Delta P_p}{P_s} = \frac{P_s - P_a}{P_s} \tag{1}$$

Because it affects static pressure readings, the SPE is responsible for errors in both airspeed and altitude. Such errors are not only unique for each type of aircraft but also tend to change as a function of

Received 10 February 2018; revision received 27 July 2018; accepted for publication 28 July 2018; published online 29 October 2018.

*Ph.D. Student, Department of Electrical and Computer Engineering, 2950 Hobson Way, Bldg. 640.

†Flight Sciences Test Operations Manager, 461st Flight Test Squadron, 225 North Flight Line Road, Bldg. 1820.

the Mach number and angle of attack (AOA). Many offline algorithms have been used to estimate the SPE via altitude or airspeed measurements [1–7]. However, even the most advanced techniques tend to either require a large logistical footprint, result in biased estimates, or use assumptions that only apply to a small subset of airframes. This paper proposes a novel algorithm for determining the SPE that is significantly more accurate than state-of-the-art methods, and it can be executed in an online fashion for any aircraft without the need for multiple controlled experiments.

II. Background

A considerable amount of research has been devoted to solving the problem of the SPE. Most notably, the flight-test community has developed numerous experiments designed to characterize the SPE for each type of aircraft across its entire Mach number domain. In general, three types of techniques have been found in the literature: altitude methods, airspeed methods, and pressure methods. Because the SPE affects both altitude and airspeed, such techniques are aimed at determining the airspeed and altitude error, respectively, as a function of airspeed, using external truth sources. Meanwhile, pressure techniques directly measure static pressure errors using ambient pressure readings from weather balloons.

A. Altitude Methods

The most widely used altitude method for SPE calibration is called the tower flyby (TFB) [1,2]. A general TFB diagram is shown in Fig. 1. The TFB technique is easy to execute from a flying perspective and produces data that are easy to process. The TFB aims to determine an altitude error correction ΔH_{pc} by comparing the indicated altitude in the aircraft's altimeter H_{ic} to an externally measured reference altitude H_c, which is derived from a theodolite measurement at a ground-based observation tower. The aircraft flies at a constant altitude and airspeed as it passes by the observation tower, where the "truth" altitude H_c is recorded. At the same time, the aircraft records its altitude H_{ic}. The error correction relationship is then given by

$$\Delta H_{\text{pc}} = H_c - H_{\text{ic}} \tag{2}$$

The error correction given by Eq. (2) is then used to sample the SPE across the entire Mach number domain for a given aircraft by repeating the TFB at various Mach number conditions. The resulting altitude error can be converted to a corresponding pressure error using Eq. (1) with

$$P_s = P_{\text{SL}}(1 - 6.87559 \times 10^{-6} H_{\text{ic}})^{5.2559} \tag{3}$$

$$P_a = P_{\text{SL}}(1 - 6.87559 \times 10^{-6} H_c)^{5.2559} \tag{4}$$

where P_{SL} is the atmospheric pressure at sea level on a standard day [8]. The computed pressure error can then be used to infer airspeed errors at similar conditions. Even though it is simple and accurate, the TFB method is limited, in that it requires multiple (time-consuming) flybys to sample the underlying $\Delta H_{\text{pc}}(M)$ curve, a team of individuals at the tower site to perform manual theodolite readings and, most importantly, an established TFB site with known geometric conditions. Additionally, obtaining TFB data for transonic and supersonic conditions proves to be problematic due to sonic boom concerns for nearby structures and personnel, as well as structural concerns for the test aircraft due to the high dynamic pressure experienced at supersonic speeds and low altitude.

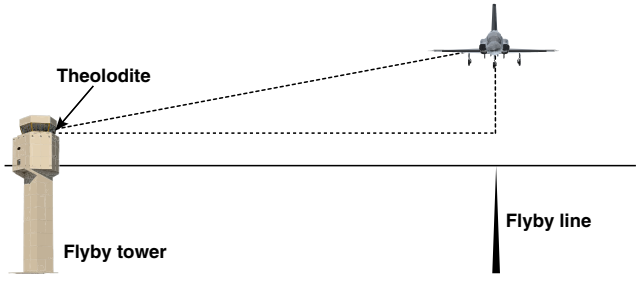

Fig. 1 Illustration of the tower flyby.

B. Airspeed Methods

Airspeed methods for characterizing the SPE have seen the most innovation in recent literature due to the emergence and proliferation of the Differential Global Positioning System (DGPS) in military and commercial aircraft. Several state-of-the art airspeed techniques [3,4] [5–7] rely on a simplified two-dimensional transformation from the body-frame (b-frame) to the navigation frame (n-frame), which is usually referred to as the "wind triangle" and shown in Fig. 2.

In [5], the true airspeed error ΔV_T, which is caused by the SPE, is estimated using the DGPS by assuming a constant and unknown wind vector. The aircraft flies a 360 deg turn at a constant indicated altitude and indicated airspeed, which allow the unknown parameters ($\boldsymbol{v}_W$ and $\boldsymbol{v}_{T_i}$) to become observable to a linear model. The model is developed using two-dimensional vector geometry from the wind triangle via

$$\boldsymbol{v}_T + \boldsymbol{v}_W = \boldsymbol{v}_G \tag{5}$$

$$\Rightarrow (\boldsymbol{v}_{T_i} + \Delta \boldsymbol{V}_T) + \boldsymbol{v}_W = \boldsymbol{v}_G \tag{6}$$

$$\Rightarrow \Delta \boldsymbol{V}_T + \boldsymbol{v}_W = \boldsymbol{v}_G - \boldsymbol{v}_{T_i} \tag{7}$$

$$\Rightarrow \Delta V_T \cos(\psi) + v_{W_N} = v_G \cos(\phi) - v_{T_i} \cos(\psi) \tag{8}$$

$$\Delta V_T \sin(\psi) + v_{W_E} = v_G \sin(\phi) - v_{T_i} \sin(\psi) \tag{9}$$

$$\Rightarrow \underbrace{\begin{bmatrix} v_{G_1}\cos(\phi_1) - v_{T_{i_1}}\cos(\psi_1) \\ \vdots \\ v_{G_M}\cos(\phi_M) - v_{T_{i_M}}\cos(\psi_M) \\ v_{G_1}\sin(\phi_1) - v_{T_{i_1}}\sin(\psi_1) \\ \vdots \\ v_{G_M}\sin(\phi_M) - v_{T_{i_M}}\sin(\psi_M) \end{bmatrix}}_{2S\times 1} = \underbrace{\begin{bmatrix} \cos(\psi_1) & 1 & 0 \\ \vdots & \vdots & \vdots \\ \cos(\psi_M) & 1 & 0 \\ \sin(\psi_1) & 0 & 1 \\ \vdots & \vdots & \vdots \\ \sin(\psi_M) & 0 & 1 \end{bmatrix}}_{2S\times 3} \begin{bmatrix} \Delta V_T \\ v_{W_N} \\ v_{W_E} \end{bmatrix} \tag{10}$$

where S is the number of data samples collected during the turn; v_{T_i} is the aircraft's measured (or SPE corrupted) true airspeed (TAS); ψ is the true heading; v_G and ϕ are the ground speed and ground track, respectively, as measured by the DGPS; ΔV_T is the unknown TAS

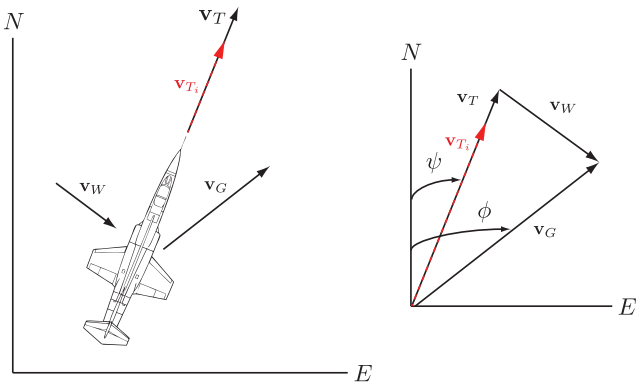

Fig. 2 Illustration of the wind triangle.

error; and v_{W_N} and v_{W_E} are the unknown constant wind vector components. In contrast to similar methods such as the cloverleaf [3] (in which the aircraft is flown at three distinct headings instead of around a full circle), this method produces a statistical model for its estimated variables that takes advantage of modern-day in-flight data recording systems. However, it is limited in the fact that, much like the TFB, it relies on multiple experiments to collect the necessary point samples of the underlying $\Delta V_{\text{pc}}(M)$ function, and some aircraft may not be able to sustain a constant-speed turn at supersonic conditions. Additionally, v_{T_i} is difficult to measure because it must be derived from the indicated airspeed V_{ic} and ambient temperature T_a, which is given by

$$T_a = \frac{T_{\text{ic}}}{1 + 0.2K_t M_{\text{pc}}^2} \tag{11}$$

where T_{ic} is the measured temperature (known as the total temperature), M_{pc} is the SPE-corrected Mach number, and K_t is a temperature calibration parameter that must be derived from other experiments such as the TFB or external sources such as weather balloons. As shown, Eq. (11) requires knowledge of the SPE to correct T_{ic}; however, determining the SPE requires K_t. Therefore, the problem is usually solved by determining K_t before the SPE, approximating M_{pc} with the indicated Mach number M_{ic} when deriving v_{T_i}, and iterating until convergence is achieved. Having obtained ΔV_T, the subsonic airspeed error can be converted to a pressure error using

$$\frac{\Delta P_p}{P_s} = \left(\frac{1}{(q_{c_{\text{ic}}}/P_s) + 1} - \frac{1}{(q_c/P_a) + 1}\right)\frac{P_T}{P_s} \tag{12}$$

where

$$\frac{q_c}{P_a} = \left(1 + 0.2\left(\frac{V_{T_i} + \Delta V_T}{a}\right)^2\right)^{7/2} - 1 \tag{13}$$

$$\frac{q_{c_{\text{ic}}}}{P_s} = \left(1 + 0.2\left(\frac{V_{T_i}}{a}\right)^2\right)^{7/2} - 1 \tag{14}$$

$$a = a_{\text{SL}}\sqrt{\frac{T_a}{T_{\text{SL}}}} \tag{15}$$

where a_{SL} is the speed of sound at sea level on a standard day [8], and T_{SL} is the standard temperature at sea level. It is important to note that Eqs. (13) and (14) take different forms for supersonic v_T and v_{T_i}, which can be found in [2].

Another class of calibration methods [7,9–11] uses recursive estimation techniques such as the Kalman filter (KF) [12] in order to converge onto calibration parameters of interest. In [9], a so-called scale factor γ_v, which was assumed to be constant for the entire airspeed (or Mach number) domain such that

$$V_T = \gamma_v V_{T_i} \tag{16}$$

was estimated using DGPS measurements, as well as often estimating the AOA and angle of sideslip (AOS) simultaneously. These recursive methods along with the angle of sideslip estimation simulations in [13] provided the baseline foundation for the proposed solution due to their use of the KF and nonlinear regression [14,15]. However, they were limited in the fact that the constant scale factor assumption was only valid for small aircraft (namely, unmanned vehicles) with a limited Mach number domain, as shown in Sec. IV. Additionally, their use of a the KF was limited to static airspeed conditions similar to the methods in [3,5,7], requiring once again the need to repeat the experiment at multiple Mach number conditions in order to sample the underlying function.

C. External Reference Methods

One of the most accurate methods for estimating the SPE is the pressure survey method [1]. In this technique, a weather balloon capable of measuring T_a, H_c, and P_a is launched into the local air mass. The balloon measurements of P_a can then directly be used to compute the SPE using Eq. (1). Obviously, this method provides the most accurate results because it directly measures the desired error. However, it is rarely used unless experimental budgets are amenable due to its cost and associated logistical footprint. Besides the financial and logistical complications, the survey method is also limited by the assumed constant atmospheric properties between the balloon launch site and the area where the experimental aircraft collects its data. This assumption also limits the ability to perform this technique in an online fashion because the truth data needed for calibration are only available and/or valid for a limited time and geographical region. Similarly, in the pacer method [2], an aircraft that has been previously calibrated can also be used as an external reference when flown alongside the uncalibrated aircraft. The benefits of such a method include the ability to compare both altitude and airspeed simultaneously, model a large portion of the Mach number domain in a single experiment (if a level acceleration is performed), and model supersonic airspeeds at safe altitudes. However, much like the pressure survey method, the pacer method suffers similar logistical footprint issues because it may be difficult to schedule a calibrated aircraft with a similar performance envelope as the aircraft to be calibrated. Additionally, any errors incurred during the calibration of the pacer aircraft will be directly transferred into the calibration of the aircraft in question.

D. Contributions

Having explored the underlying characteristics of the SPE problem and the state-of-the-art solutions, we now turn to the proposed algorithm, henceforth referred to as the Jurado–McGehee online self-survey (JMOSS), and its specific contributions. The JMOSS algorithm provides a drastic improvement over all other methods in that it 1) uses a hybrid pressure–airspeed–altitude algorithm inside a backward-smoothing extended Kalman filter (BSEKF) framework to estimate ΔP_p and K_t in an online fashion, without the need for multiple experiments or external truth sources; 2) develops an autonomous information-theory-based spline smoothing process, referred to as the Akaike spline model (ASM), which balances model complexity with error reduction and captures transonic and supersonic effects with no prior knowledge of the $\Delta P_p(M_{\text{ic}})$ functional form; and 3) enables full Mach number domain characterization, including transonic and supersonic effects using a single experiment, without the need to sustain supersonic speeds.

E. Outline

The remainder of this paper is organized into three additional sections. Section III develops the flying and data-processing algorithms that enable the research advancements proposed herein. Section IV presents results from a T-38C flight-test program, comparing the proposed algorithm against state-of-the-art airspeed, altitude, and pressure methods; and using weather balloon pressure survey data as the reference truth. Finally, Sec. V summarizes the research effort, and it presents conclusions and future work.

III. Methodology

This section describes the flight and data-processing algorithms developed during this research, which enable the specific contributions previously outlined. Figure 3 illustrates the information flow from required input data to BSEKF output and subsequent ASM estimation. The specific methods used are described in the following sections.

A. Flight Technique

The flight technique needed to meet observability requirements was based on [5]. However, it was found that SPE estimates tend to become noisy during nonlevel flight, most likely due to dynamic changes in the

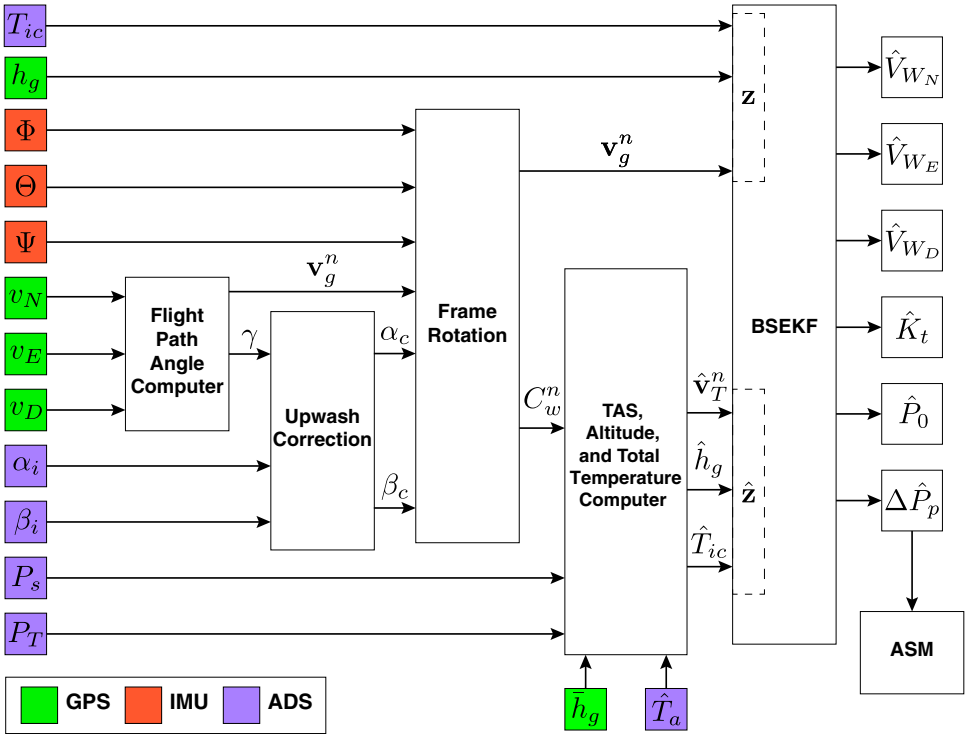

Fig. 3 Data-processing flow for JMOSS algorithm.

AOA and AOS during turns. As such, the flight technique for the proposed algorithm was modified to meet the observability requirements for wind estimation and Mach number dependency by separating them into three distinct phases: 1) a constant-altitude deceleration from $M_{\max}$ to M_{turn}; 2) a constant-altitude constant-airspeed 360 deg turn at M_{turn}; and 3) a constant-altitude deceleration from M_{turn} to $M_{\min}$, where $M_{\max}$ is the aircraft's maximum Mach number, $M_{\min}$ is the minimum Mach number, and M_{turn} is an arbitrary constant turning speed within that domain. The three-phase design of the flight technique provided three main efficiencies in the context of data collection. First, it enabled collection of flight-test data across the entire Mach number domain in a single experiment, without the need for sustained supersonic conditions. Next, the constant-Mach-number turn phase allowed for wind observability. Finally, the decoupling of turning and deceleration allowed for the collection of noncorrupted SPE data while still meeting wind observability requirements. Several experiments were conducted in order to establish repeatability and algorithm stability. Table 1 summarizes the actual values for the aforementioned conditions that were used for experimental data collection. Additionally, the experiments summarized in Table 2 were performed in order to collect data for later comparison against the proposed method.

Table 1 Summary of flight conditions for JMOSS experiments

	Conditions				
Experiment	H_{ic}, ft PA	$M_{\max}$	M_{turn}	$M_{\min}$	Duration, min
1	18,270	1.06	0.54	0.52	7.40
2	19,950	1.05	0.64	0.54	6.81
3	18,820	1.06	0.73	0.53	8.95
4	21,230	1.05	0.92	0.54	8.32

Table 2 Summary of flight conditions for comparison methods

Method	Points	Domain, Mach number	Duration, min
Level turn	10	0.53–0.92	53.57
Cloverleaf	9	0.52–0.94	56.69
Tower flyby	10	0.54–0.90	42.00

B. Required Data

One of the key enabling technologies provided in this paper is the development of a hybrid pressure–airspeed–altitude method using a BSEKF. Table 3 summarizes all the required data and their sources for the JMOSS algorithm. As such, pressure readings (P_T and P_s) from the ADS are used directly, instead of indirectly via H_{ic} or V_{ic}. As previously discussed, T_{ic} is required in order to estimate T_a, which is done inside the algorithm, eliminating the need for an external temperature calibration. Next, the AOA and AOS from the ADS, along with aircraft body angles from the inertial measurement unit (IMU), are required to compute the necessary direction cosine matrices (DCMs) to transform vectors from the wind frame (w-frame) to the n-frame. Finally, the Global Positioning System (GPS) velocity and altitude measurements are required to compute the flight-path angle (a parameter needed in AOA and AOS correction) and provide measurement updates to the BSEKF.

C. AOA and AOS Corrections

Begin by correcting the indicated AOA α_i for upwash errors, which are a function of Mach number [16], using

$$\alpha_c = \alpha_i + \Delta\alpha(M_{\text{ic}}) \tag{17}$$

$$\Delta\alpha(M_{\text{ic}}) = \hat{\beta}_0 + \hat{\beta}_1 M_{\text{ic}} + \hat{\beta}_2 M_{\text{ic}}^2 \tag{18}$$

Table 3 Required data parameters for JMOSS algorithm

Name	Symbol	Source
Static pressure	P_s	ADS
Total pressure	P_T	ADS
Total temperature	T_{ic}	ADS
Indicated AOA	α_i	ADS
Indicated AOS	β_i	ADS
Roll angle	Φ	IMU
Pitch angle	Θ	IMU
Yaw angle	Ψ	IMU
North ground speed	v_N	GPS
East ground speed	v_E	GPS
Down ground speed	v_D	GPS
Geometric altitude	h_g	GPS

where M_{ic} is derived from P_s and P_T using standard pitot-static equations [2], α_c is the corrected AOA, α_i is the indicated AOA, and the function $\Delta\alpha(M_{\text{ic}})$ is given by the second-order polynomial model

$$\begin{bmatrix} \Theta_1 - \gamma_1 - \alpha_{i_1} \\ \vdots \\ \Theta_S - \gamma_S - \alpha_{i_S} \end{bmatrix} = \begin{bmatrix} 1 & M_{\text{ic}_1} & M_{\text{ic}_1}^2 \\ \vdots & \vdots & \vdots \\ 1 & M_{\text{ic}_S} & M_{\text{ic}_S}^2 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} \tag{19}$$

where Θ is the pitch angle, S is the number of measurements, and γ is the flight-path angle given by

$$\gamma = \arcsin\left(\frac{-v_D}{v_T}\right) \tag{20}$$

which can be approximated using the ground velocity vector $\boldsymbol{v}_g$ by

$$\gamma \approx \arcsin\left(\frac{-v_D}{\|\boldsymbol{v}_g\|}\right) \tag{21}$$

assuming V_T is much larger than the wind speed. Finally, the AOS is corrected by projecting β_i onto the corrected w-frame using

$$\beta_c = \arctan(\cos(\alpha_c)\tan(\beta_i)) \tag{22}$$

D. Ambient Temperature Optimization

Before processing the data using the BSEKF, an estimate of ambient temperature is obtained by minimizing the least-squares [14] cost function given by

$$\min_{b_1,b_2,b_3} C(b_1,b_2,b_3) = \sum_{s=1}^{S}\left[T_{\text{ic}_s} - \hat{T}_{\text{ic}_s}\right]^2 \tag{23}$$

where

$$\hat{T}_{\text{ic}_s} = \hat{T}_{a_s}\left(1 + 0.2\hat{K}_{t_s}M_{\text{ic}_s}^2\right) \tag{24}$$

$$\hat{T}_{a_s} = T_{\text{SL}} f_{\theta_H}(h_{g_s}) + b_1 \tag{25}$$

$$\hat{K}_{t_s} = b_2 + b_3 M_{\text{ic}_s}^2 \tag{26}$$

and the function f_{θ_H} was given in [2]. Essentially, minimizing Eq. (23) leads to optimal coefficients b_1, b_2, and b_3 that best approximate the actual T_{ic} measurements while constraining $\hat{T}_a$ to follow the standard temperature profile given by h_g, plus a constant bias, and K_t to depend on M_{ic}^2. Once converged, the resulting optimal estimates of $\hat{T}_a$ are used as control inputs in the BSEKF, for which a better estimate of K_t, based on M_{pc}, is produced alongside the other variables of interest.

E. BSEKF Implementation

Using the notations described in [17], the main estimation engine of the algorithm is driven by a six-state BSEKF [12,17–19] with system dynamics defined by

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{G}(t)\boldsymbol{w}(t) \tag{27}$$

$$\boldsymbol{x} = \begin{bmatrix} \Delta P_p & v_{W_N} & v_{W_E} & v_{W_D} & K_t & P_0 \end{bmatrix}^T \tag{28}$$

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}^T \tag{29}$$

where ΔP_p is the SPE; v_{W_N}, v_{W_E}, and v_{W_D} are the wind components; K_t is the temperature recovery factor; P_0 is an ambient pressure about which the relationship between the geometric altitude and pressure altitude is linearized; and $\boldsymbol{w}(t)$ is a bivariate Gaussian white-noise process with

$$E[\boldsymbol{w}(t)] = [0 \quad 0]^T \tag{30}$$

$$E[\boldsymbol{w}(t)^T\boldsymbol{w}(t+\tau)] = 0.1\begin{bmatrix} \delta(\tau) & 0 \\ 0 & \delta(\tau) \end{bmatrix} \tag{31}$$

The BSEKF discrete measurement model at time k is defined by

$$\boldsymbol{z}_k = \boldsymbol{h}[\boldsymbol{x}_k, \boldsymbol{u}_k] + \boldsymbol{v}_k \tag{32}$$

$$\boldsymbol{u} = \begin{bmatrix} P_s & P_T & \alpha_c & \beta_c & \Phi & \Theta & \Psi & \bar{h}_g & \hat{T}_a \end{bmatrix}^T \tag{33}$$

where $\bar{h}_g$ is the mean geometric altitude, and the vector $\boldsymbol{v}_k$ is composed of five independent Gaussian white-noise processes with

$$E\left[\boldsymbol{v}_k\right] = \underset{5\times1}{\mathbf{0}} \tag{34}$$

$$E\left[\boldsymbol{v}_k^T\boldsymbol{v}_l\right] = \delta_{kl}\underset{5\times5}{\boldsymbol{I}} \tag{35}$$

The nonlinear measurement function $\boldsymbol{h}$ in Eq. (32) estimates incoming GPS groundspeed and altitude measurements, as well as total temperature measurements, to form the vector

$$\hat{\boldsymbol{z}}_k = \begin{bmatrix} \hat{\boldsymbol{v}}_T^n + \hat{\boldsymbol{v}}_W & \hat{h}_g & \hat{T}_{\text{ic}} \end{bmatrix}^T \tag{36}$$

$$= \begin{bmatrix} \hat{v}_N & \hat{v}_E & \hat{v}_D & \hat{h}_g & \hat{T}_{\text{ic}} \end{bmatrix}^T \tag{37}$$

and is constructed from standard pitot-static equations [2] using

$$\hat{P}_a = P_s - \hat{\Delta}P_p \tag{38}$$

$$\hat{M}_{\text{pc}} = f\left(\hat{P}_a, P_T\right) \tag{39}$$

$$\hat{T}_{\text{ic}} = \hat{T}_a\left(1 + 0.2\hat{K}_t\hat{M}_{\text{pc}}^2\right) \tag{40}$$

$$\hat{a} = a_{\text{SL}}\sqrt{\frac{\hat{T}_a}{T_{\text{SL}}}} \tag{41}$$

$$\hat{v}_T = \hat{M}_{\text{pc}}\hat{a} \tag{42}$$

$$\hat{\boldsymbol{v}}_T^w = \begin{bmatrix} \hat{v}_T & 0 & 0 \end{bmatrix}^T \tag{43}$$

$$\hat{\boldsymbol{v}}_T^n = C_b^n C_w^b \hat{\boldsymbol{v}}_T^w \tag{44}$$

where the function in Eq. (39) was given in [2], the DCMs C_b^n and C_w^b are created using the frame transformations in [20] given by

$$C_w^b = \begin{bmatrix} \cos(\alpha_c)\cos(\beta_c) & -\cos(\alpha_c)\sin(\beta_c) & -\sin(\alpha_c) \\ \sin(\beta_c) & \cos(\beta_c) & 0 \\ \sin(\alpha_c)\cos(\beta_c) & -\sin(\alpha_c)\sin(\beta_c) & \cos(\alpha_c) \end{bmatrix} \tag{45}$$

$$C_b^n = \begin{bmatrix} \cos(\Theta)\cos(\Psi) & \cos(\Psi)\sin(\Theta)\sin(\Phi) - \cos(\Phi)\sin(\Psi) & \sin(\Phi)\sin(\Psi) + \cos(\Phi)\cos(\Psi)\sin(\Theta) \\ \cos(\Theta)\sin(\Psi) & \cos(\Phi)\cos(\Psi) + \sin(\Theta)\sin(\Phi)\sin(\Psi) & \cos(\Phi)\sin(\Theta)\sin(\Psi) - \cos(\Psi)\sin(\Phi) \\ -\sin(\Theta) & \cos(\Theta)\sin(\Phi) & \cos(\Theta)\cos(\Phi) \end{bmatrix} \tag{46}$$

the estimated wind vector $\hat{\boldsymbol{v}}_W$ is given by

$$\hat{\boldsymbol{v}}_W = \begin{bmatrix} \hat{v}_{W_N} & \hat{v}_{W_E} & \hat{v}_{W_D} \end{bmatrix}^T \tag{47}$$

the estimated geometric altitude measurement $\hat{h}_g$ is given by

$$\hat{\delta} = \frac{\hat{P}_a}{P_{\text{SL}}}, \qquad \hat{\delta}_0 = \frac{\hat{P}_0}{P_{\text{SL}}} \tag{48}$$

$$\hat{H}_c = f_{H_\delta}(\hat{\delta}), \qquad \hat{H}_{c_0} = f_{H_\delta}(\hat{\delta}_0) \tag{49}$$

$$\hat{\theta} = f_{\theta_H}(\hat{H}_c) \tag{50}$$

$$T_{\text{std}} = T_{\text{SL}}\hat{\theta} \tag{51}$$

$$\hat{h}_g = \bar{h}_g + \frac{\hat{T}_a}{T_{\text{std}}}\left(\hat{H}_c - \hat{H}_{c_0}\right) \tag{52}$$

P_{SL} is the sea-level standard pressure, and the functions f_{H_δ} and f_{θ_H} were given in [2].

To enable online estimation, the BSEKF is initialized with no prior knowledge of the system states. As such, all initial estimates are set to zero, with the exception of $\hat{K}_t$, which is set to one. Additionally, the initial state estimation covariance matrix is set to a 6×6 identity matrix. Because the turn data ($M_{\text{ic}} = M_{\text{turn}}$) are not necessarily collected at $t_k = 0$, the BSEKF is processed "forward" from $t_k = 0$ to $t_k = (M-1)\Delta t$, where Δt is the sampling period, in order to converge onto accurate estimates of the wind states and corresponding $\hat{\Delta}P_p$, K_t, and P_0. Next, the resulting final state estimates from the forward run are used as initial estimates for the BSEKF smoothing run from $t_k = (M-1)\Delta t$ to $t_k = 0$ in order to smooth any biased $\hat{\Delta}P_p$, K_t, and δP_0 estimates that occurred on the forward run before $M_{\text{ic}} = M_{\text{turn}}$.

F. Akaike Spline Model

Having obtained the estimates from the BSEKF, one may choose to fit a model to the SPE observations with respect to Mach number in a number of ways. In this research, the resulting BSEKF estimates of the SPE are modeled as a function of M_{ic} using a novel linear smoothing spline model referred to as the ASM. The ASM algorithm is crucial in smoothing BSEKF output with no prior knowledge of the functional relationship between $\Delta P_p/P_s$ and M_{ic} for the ADS being calibrated. Additionally, it allows for the accurate modeling of unknown changes to the functional form in the transonic and supersonic regions, as shown in Sec. IV.

Algorithm 1 illustrates a pseudocode implementation of the ASM process. ASM smoothing begins with a simple second-order model of the form

$$\underbrace{\begin{bmatrix} \frac{\hat{\Delta}P_{p_1}}{P_{s_1}} \\ \vdots \\ \frac{\hat{\Delta}P_{p_S}}{P_{s_S}} \end{bmatrix}}_{\boldsymbol{y}} = \underbrace{\begin{bmatrix} 1 & M_{\text{ic}_1} & M_{\text{ic}_1}^2 \\ \vdots & \vdots & \vdots \\ 1 & M_{\text{ic}_S} & M_{\text{ic}_S}^2 \end{bmatrix}}_{\mathbf{X}} \underbrace{\begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix}}_{\boldsymbol{\beta}} \tag{53}$$

where S is the number of measurements in the experiment. Next, a simple optimization routine is executed to sequentially add smoothing spline knots using

Algorithm 1 ASM = fitASM($\boldsymbol{m}_{\text{ic}}$,$\hat{\boldsymbol{\Delta}}\boldsymbol{P}_p/\boldsymbol{P}_s$)

Input: $\hat{\boldsymbol{\Delta}}\boldsymbol{P}_p/\boldsymbol{P}_s$, $\boldsymbol{m}_{\text{ic}}$ ▶ Inputs are BSEKF output for the SPE, and computed M_{ic}
$\boldsymbol{y} \leftarrow \hat{\boldsymbol{\Delta}}\boldsymbol{P}_P/\boldsymbol{P}_s$ ▶ Create observation vector
if $\max(\boldsymbol{m}_{\text{ic}}) > 1$, **then** ▶ If supersonic data present, add supersonic knots $M_{\text{ic}} \in (0.93, 1)$
  superSonic ← true
end if
$P \leftarrow 0$, go←true ▶ Initialize loop with $P = 0$ kt
while go, **do**
  $\mathbf{X} \leftarrow$ createSplineRegressor(P, $\boldsymbol{m}_{\text{ic}}$, superSonic) ▶ Use Eq. (54) to create $\mathbf{X}$ based on P
  $\hat{\boldsymbol{\beta}}$, $\text{AIC}_c(P+3) \leftarrow (\mathbf{X}^T\mathbf{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$ ▶ Compute $P+3$ total coefficients and corresponding AIC_c value
  if $P = 0$, **then**
    $\boldsymbol{X}_{\text{prev}}, \hat{\boldsymbol{\beta}}_{\text{prev}}, \text{AIC}_{c_{\text{prev}}} \leftarrow \boldsymbol{X}, \hat{\boldsymbol{\beta}}, \text{AIC}_c$ ▶ First time in the loop, add a knot
    $P \leftarrow P + 1$
  else
    if $\text{AIC}_c < 1.01 \times \text{AIC}_{c_{\text{prev}}}$, **then** ▶ If AIC_c is decreased by 1%, try one more knot
      $\boldsymbol{X}_{\text{prev}}, \hat{\boldsymbol{\beta}}_{\text{prev}}, \text{AIC}_{c_{\text{prev}}} \leftarrow \boldsymbol{X}, \hat{\boldsymbol{\beta}}, \text{AIC}_c$
      $P \leftarrow P + 1$
    else
      $\boldsymbol{X}, \hat{\boldsymbol{\beta}} \leftarrow \boldsymbol{X}_{\text{prev}}, \hat{\boldsymbol{\beta}}_{\text{prev}}$ ▶ Otherwise, stop the loop
      go←false
    end if
  end if
end while
Output: ASM
  ASM.Model $\leftarrow \hat{\boldsymbol{\beta}}$
  ASM.deltaPp_Ps $\leftarrow \boldsymbol{X}\hat{\boldsymbol{\beta}}$
  ASM.machIC $\leftarrow \boldsymbol{m}_{\text{ic}}$

$$\underbrace{\begin{bmatrix} \frac{\hat{\Delta}P_{p_1}}{P_{s_1}} \\ \vdots \\ \frac{\hat{\Delta}P_{p_S}}{P_{s_S}} \end{bmatrix}}_{\boldsymbol{y}} = \underbrace{\begin{bmatrix} 1 & M_{\mathrm{ic}_1} & M_{\mathrm{ic}_1}^2 & (M_{\mathrm{ic}_1}-s_1)_+^2 & \cdots & (M_{\mathrm{ic}_1}-s_P)_+^2 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & M_{\mathrm{ic}_S} & M_{\mathrm{ic}_S}^2 & (M_{\mathrm{ic}_S}-s_1)_+^2 & \cdots & (M_{\mathrm{ic}_S}-s_P)_+^2 \end{bmatrix}}_{\mathbf{X}} \underbrace{\begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_{P+3} \end{bmatrix}}_{\boldsymbol{\beta}} \tag{54}$$

where each s_p, $p = 1, \ldots, P$, referred to as a knot, is a preselected inflection point along the Mach number domain; and the operator $()_+$ denotes that negative values of its argument are set to zero, which is equivalent to multiplying by the Heaviside function centered at the knot location. The optimization is based on minimizing the resulting Akaike information criterion $(\mathrm{AIC})_c$ value [21], which balances the error reduction with model complexity. At each increment, a single spline knot is added to the linear model [Eq. (53)] at a location within the M_{ic} domain based on statistical quantiles [22]. Then, the resulting $(\mathrm{AIC})_c$ model criterion is compared to its previous value to verify that at least a 1% decrease in $(\mathrm{AIC})_c$ is achieved by the additional knot. If, at any point, this criterion is not met, the optimization is considered complete and the spline model is finalized. Finally, if the particular experiment contains supersonic data (i.e., $M_{\mathrm{ic}} > 1$), an additional seven knots are automatically added evenly between $M_{\mathrm{ic}} = 0.93$ and $M_{\mathrm{ic}} = 1.00$ in order to capture any potential drastic changes to the functional relation in the transonic and supersonic regions. Once completed, the resulting model inferences, such as the prediction interval (PI) are computed using [23].

IV. Results

Figures 4 and 5 illustrate the state estimation histories of the forward and backward BSEKF passes for a single JMOSS experiment, respectively. As shown in Fig. 4, the BSEKF states are

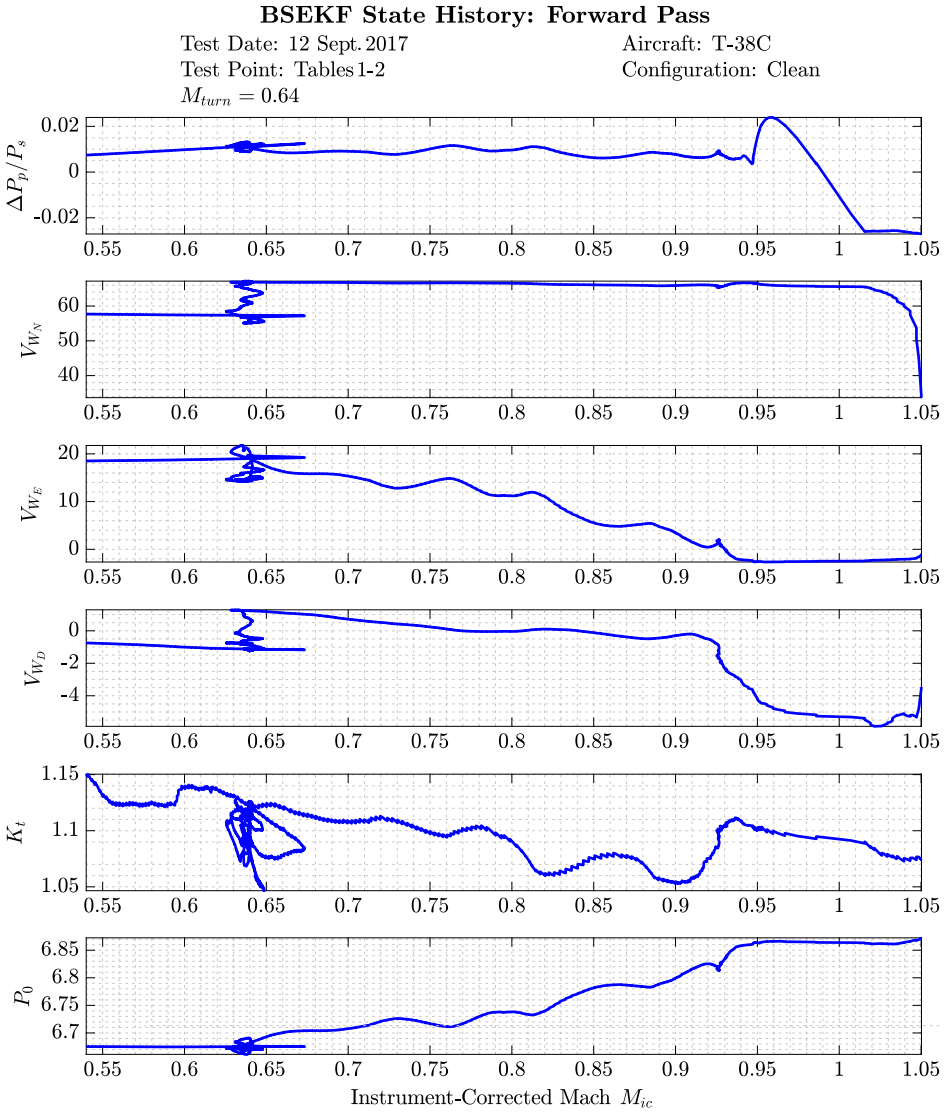

Fig. 4 Illustration of JMOSS BSEKF output on forward pass.

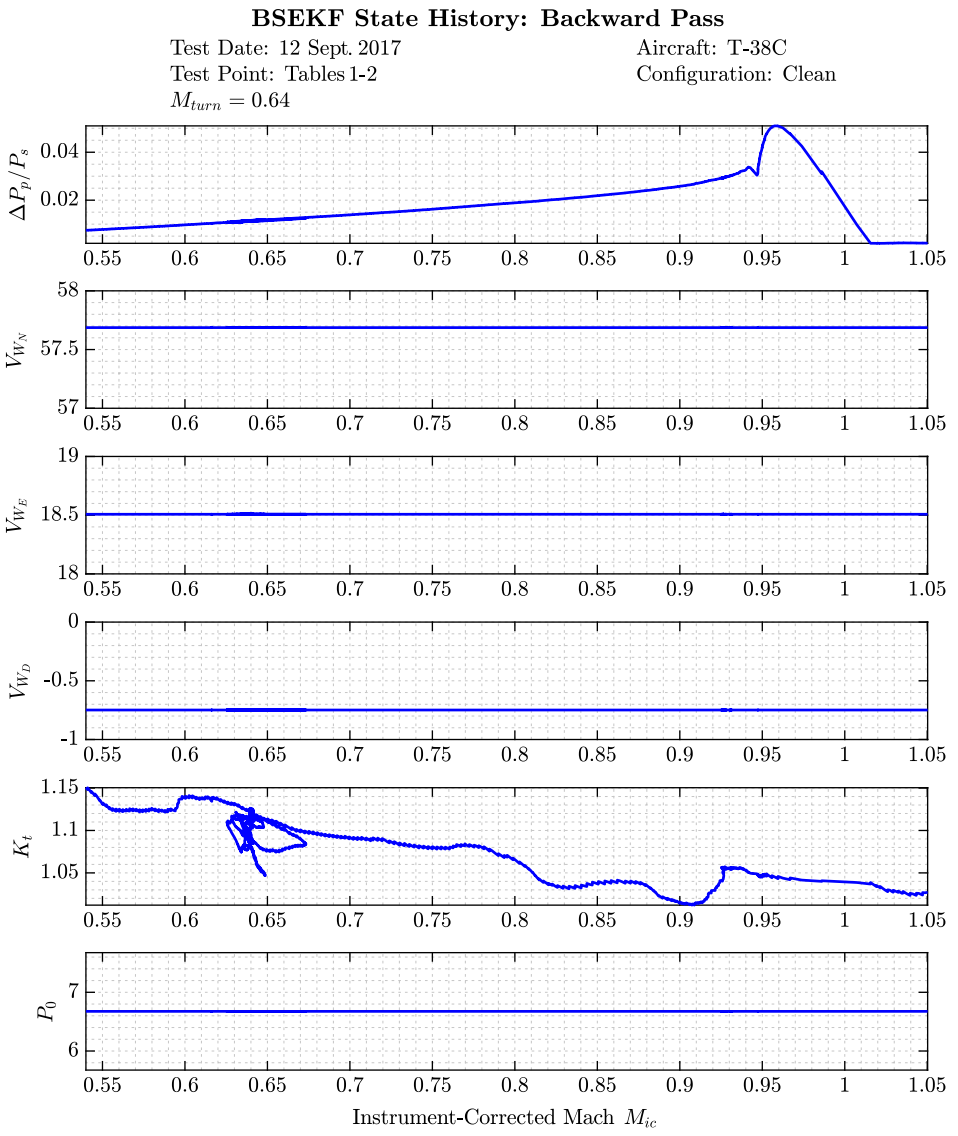

Fig. 5 Illustration of JMOSS BSEKF output on backward pass.

unobservable (and inaccurate) during the forward pass from $M_{\text{ic}} = M_{\text{max}}$ until the turn is executed, and they begin to converge after $M_{\text{ic}} \leq M_{\text{turn}}$ as the measurements are processed from M_{max} to M_{min}. Using the final estimates of the forward pass (i.e., when $M_{\text{ic}} = M_{\text{min}}$) as initial estimates for the backward-smoothing pass produced stable and accurate estimates of all six states, as shown in Fig. 5.

Figures 6 and 7 illustrate the results from a single JMOSS experiment and all JMOSS experiments combined, respectively. As shown in Fig. 6, a single JMOSS experiment yielded accurate results across the entire Mach number domain, with no need for external sources or prior knowledge, while simultaneously calibrating the temperature recovery factor K_t. As shown in Fig. 7, combining the BSEKF results from all four experiments slightly increased the associated model PI due to the variation in BSEKF estimates across experiments, but it increased accuracy when compared to the survey truth data.

Figures 8–11 illustrate the calibration results from level turn, cloverleaf, and TFB techniques. As previously mentioned, these methods required varying levels of logistical footprints, were limited in their Mach number domain, and/or required prior knowledge of K_t. As shown, all methods yielded results that closely followed survey truth data, with varying levels of bias and PI widths. It is important to note the data reduction for the level turn and cloverleaf methods included the enhancements that were developed as part of the JMOSS algorithm (i.e., AOA and AOS corrections, direct computations of airspeeds using pressures, and three-dimensional reference frame rotations), which may have contributed to their accuracy.

Finally, Table 4 compares effort metrics across all methods tested during this research. To highlight the true potential in efficiency from the JMOSS algorithm, only the results from a single experiment were considered. The time figures were computed by summing all flight time dedicated to collecting data for each experiment. The cost figures were directly proportional to a T-38C flight time at a representative flight-test rate of 11,300 (USD)/h. Meanwhile, Δ Mach captured the difference between the minimum and maximum Mach numbers modeled by each experiment. Finally, the mean bias was taken as the average difference between each method's results and survey truth data, contained within the bounds of each method's Mach number domain, and normalized by that width (i.e., divided by Δ Mach). As shown, the JMOSS algorithm was able to produce accurate results with as much as 90% fewer test points, 88% less time/cost, 83% less bias, and 78% less uncertainty, all while modeling 42% more Mach number domain.

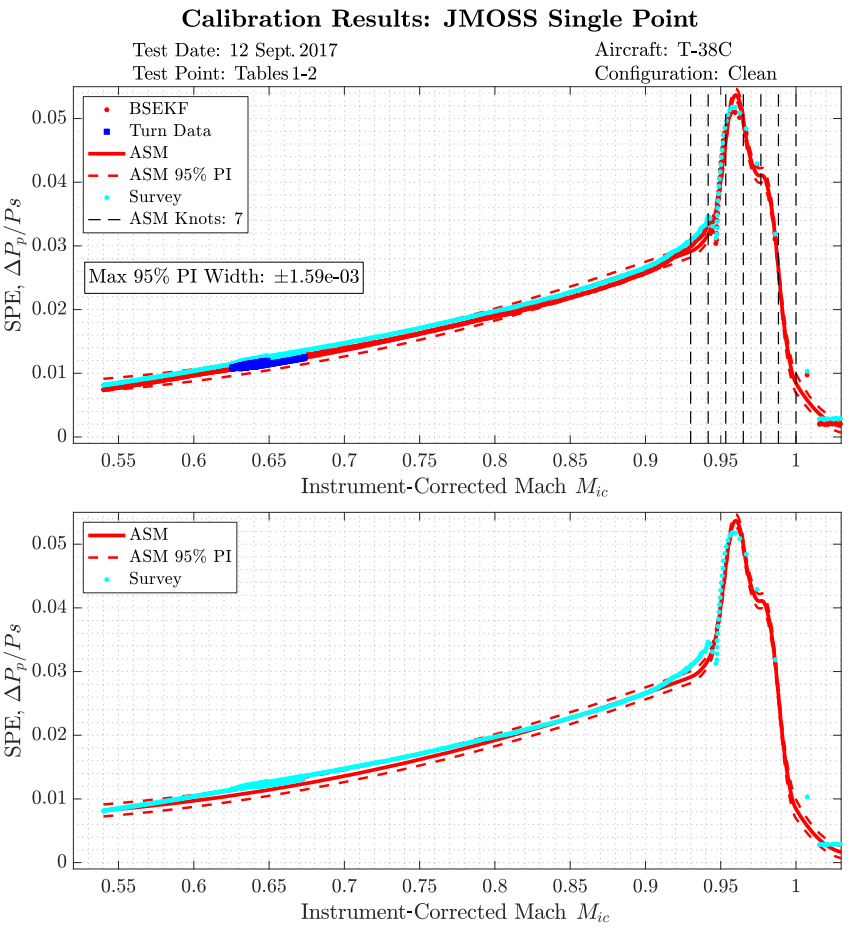

Fig. 6 JMOSS results for a single test point.

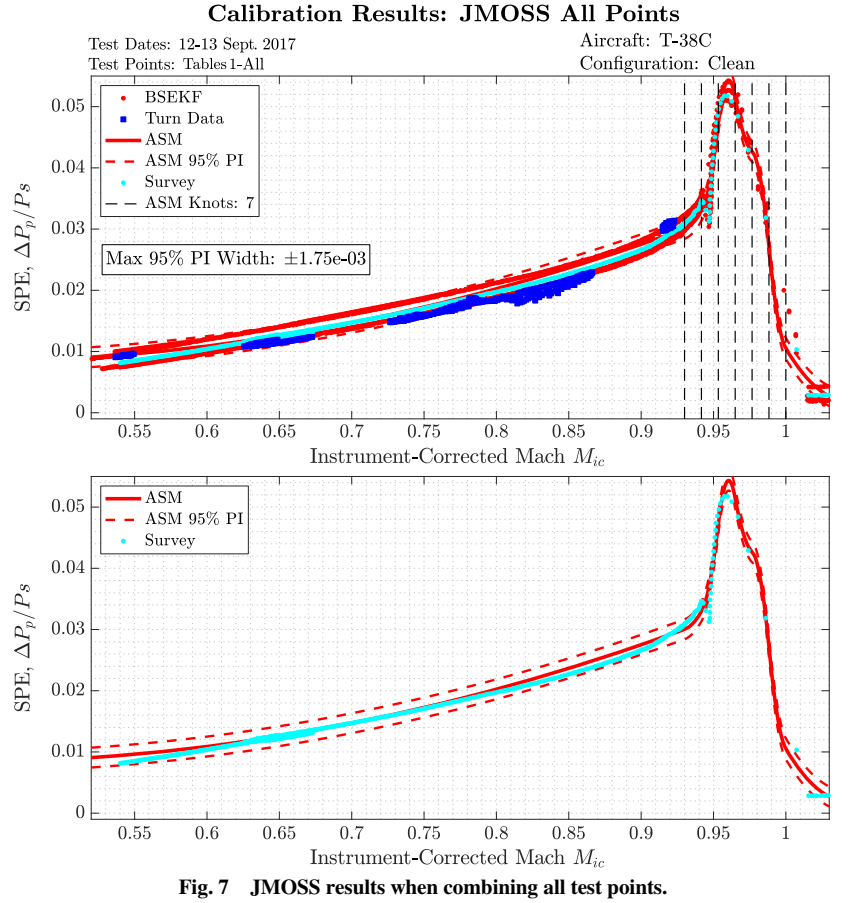

Fig. 7 JMOSS results when combining all test points.

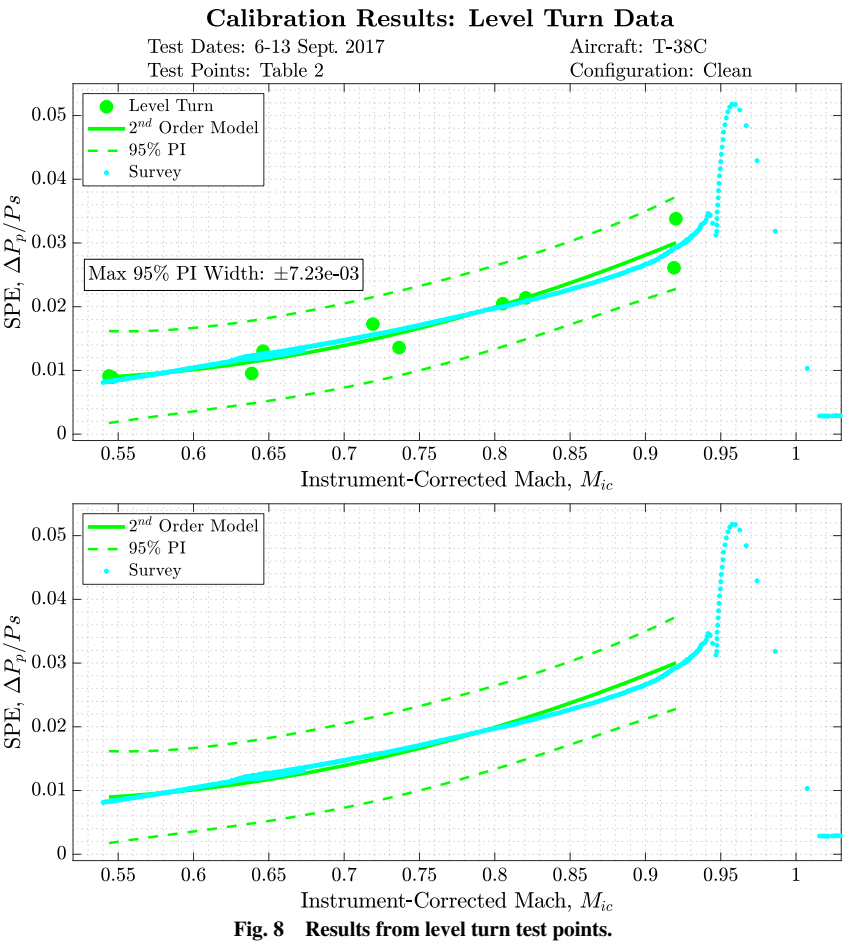

Fig. 8 Results from level turn test points.

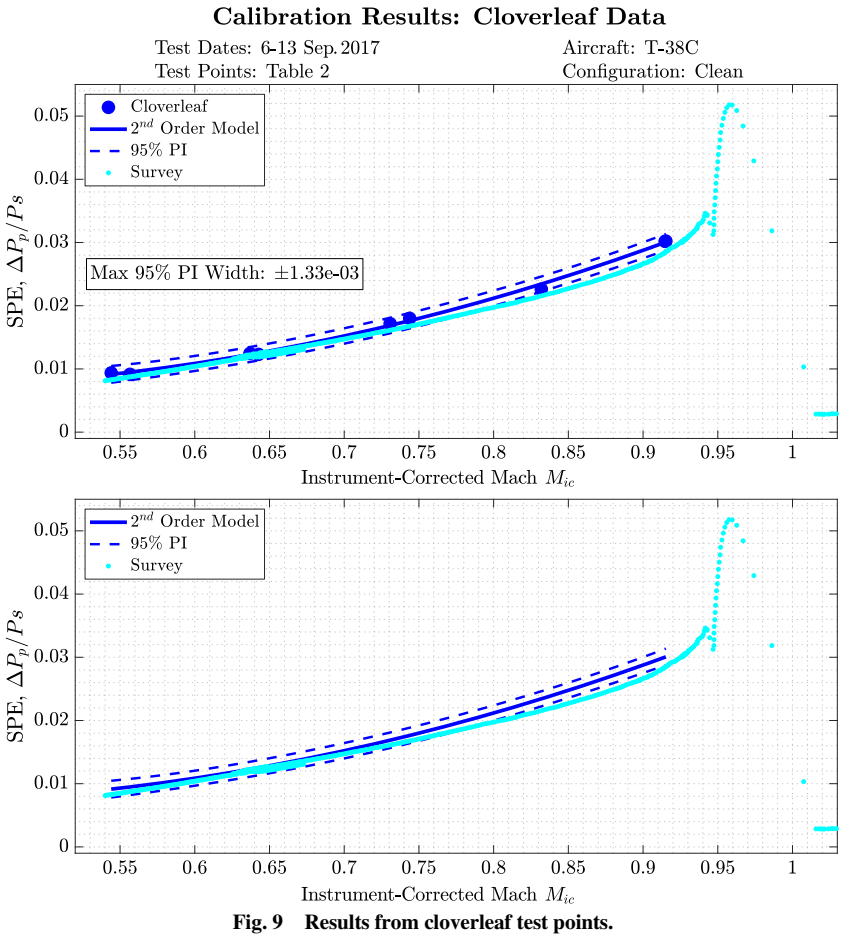

Fig. 9 Results from cloverleaf test points.

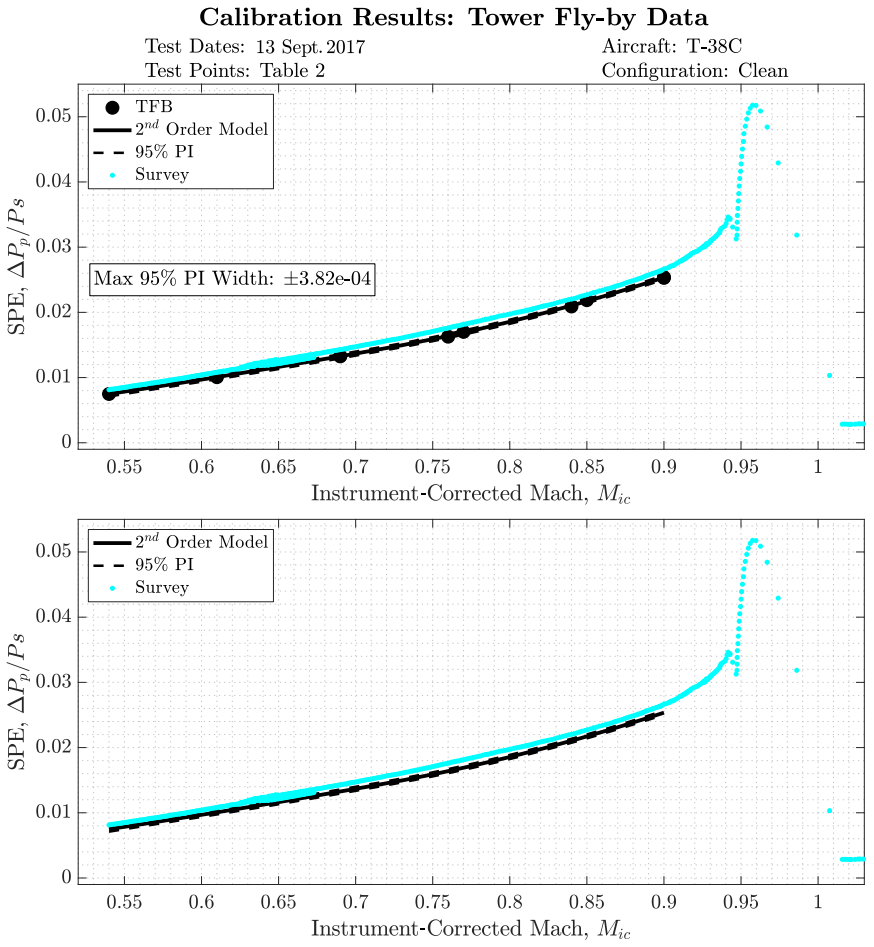

Fig. 10 Results from tower flyby test points.

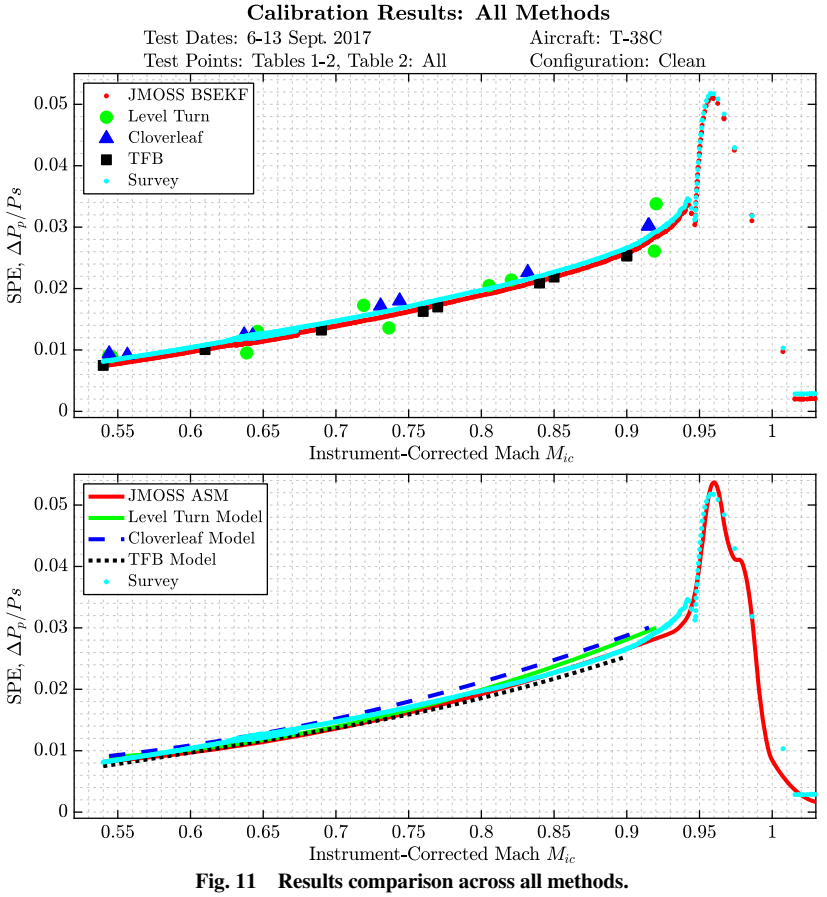

Fig. 11 Results comparison across all methods.

Table 4 Metric comparison for ADS calibration algorithms

Method	JMOSS	Level turn	Cloverleaf	TFB
Points	1	10	9	10
Time, min	6.81	53.57	56.69	42.00
Cost, USD	1,300	10,100	10,700	7,900
Minimum Mach number	0.54	0.53	0.52	0.54
Maximum Mach number	1.05	0.92	0.94	0.90
Δ Mach	0.51	0.39	0.42	0.36
Mean bias	-7.75×10^{-4}	1.89×10^{-3}	4.61×10^{-3}	-8.85×10^{-4}
95% PI	$\pm 1.59 \times 10^{-3}$	$\pm 7.23 \times 10^{-3}$	$\pm 1.33 \times 10^{-3}$	$\pm 3.82 \times 10^{-4}$

V. Conclusions

This paper has introduced a fully self-contained pressure–airspeed–altitude hybrid backward-smoothing extended Kalman filter (BSEKF)-based air data system calibration algorithm with an accompanying autonomous smoothing spline process rooted in information theory. As shown in the previous sections, the proposed algorithm modeled a larger portion of the Mach number domain while drastically reducing the cost, flight time, mean error, and maximum width of the 95% prediction intervals around the resulting model. Additionally, when experimental data from multiple dates and across varying atmosphere conditions were used, the model proved to have stable, repeatable results. The Jurado–McGehee online self-survey algorithm introduced herein provided a fully automated and self-contained means of establishing an accurate static position error correction curve for any aircraft with no prior knowledge and minimal maneuver requirements for observability. The proposed method set the course for an emerging class of online calibration and dynamic performance modeling algorithms that not only took advantage of modern data collection capabilities but also made full use of sensor fusion technology in order to relax the required experimental conditions for such modeling. Future work in this area will includes developing more robust post-BSEKF smoothing techniques beyond the Akaike spline model (e.g., neural networks), identifying potential additional sources of information for sensor fusion, and expanding the concept of sensor-fusion-based online calibration to aircraft performance and flying qualities.

Acknowledgments

The authors would like to thank the guidance and support of the U.S. Air Force Test Center, 412th Test Wing Test Engineering Group; the U.S. Air Force Test Pilot School; and the U.S. Air Force Institute of Technology. All funding was provided by the U.S. Government through the U.S. Department of Defense and the U.S. Air Force. The views expressed in this paper are those of the authors, and they do not reflect the official policy or position of the U.S. Air Force, the U.S. Department of Defense, or the U.S. Government.

References

[1] Haering, E. A., Jr., "Airdata Measurement and Calibration," NASA TM 104316, 1995.

[2] Erb, R. E., "Pitot-Statics Textbook," U.S. Air Force Test Pilot School TR, Edwards AFB, CA, 2015.

[3] Lewis, G. V., "A Flight Test Technique Using GPS for Position Error Correction Testing," *COCKPIT*, Soc. of Experimental Test Pilots Quarterly Publ., Vol. 1997, No. 1, Jan.–March 1997, pp. 20–24.

[4] Lewis, G. V., "Using GPS to Determine Pitot-Static Errors," National Test Pilot School TR, Mojave, CA, 2003.

[5] Jorris, T. R., Ramos, M. M., Erb, R. E., and Woolf, R. K., "Statistical Pitot-Static Calibration Technique Using Turns and Self-Survey Method," *42nd Annual International SFTE Symposium*, Seattle, WA, 2011, Paper 5.

[6] Olson, W. M., "Pitot-Static Calibrations Using a GPS Multi-Track Method," *29th Annual International SFTE Symposium*, 1998, Paper 17.

[7] Niewoehner, R. J., "Refining Satellite Methods for Pitot-Static Calibration," *Journal of Aircraft*, Vol. 43, No. 3, 2006, pp. 846–849. doi:10.2514/1.18976

[8] "U.S. Standard Atmosphere, 1976," National Oceanic and Atmospheric Administration, Washington D.C., 1976, pp. 21–23.

[9] Cho, A., Kang, Y.-s., Park, B.-j., and Yoo, C.-s., "Airflow Angle and Wind Estimation Using GPS/INS Navigation Data and Airspeed," *Control Automation and Systems (ICCAS) 2013 13th International Conference*, IEEE Publ., Piscataway, NJ, 2013, pp. 1321–1324.

[10] Johansen, T. A., Cristofaro, A., Sorensen, K., Hansen, J. M., and Fossen, T. I., "On Estimation of Wind Velocity, Angle-of-Attack and Sideslip Angle of Small UAVs Using Standard Sensors," *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE Publ., Piscataway, NJ, 2015, pp. 510–519. doi:10.1109/icuas.2015.7152330

[11] Haering, E. A., Jr., "Airdata Calibration Techniques for Measuring Atmospheric Wind Profiles," *Journal of Aircraft*, Vol. 29, No. 4, 1992, pp. 632–639. doi:10.2514/3.46212

[12] Kalman, R. E., "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, Vol. 82, No. 1, March 1960, p. 35–45. doi:10.1115/1.3662552

[13] Lando, M., Battipede, M., and Gili, P. A., "Neuro-Fuzzy Techniques for the Air-Data Sensor Calibration," *Journal of Aircraft*, Vol. 44, No. 3, 2007, pp. 945–953. doi:10.2514/1.26030

[14] Marquardt, D. W., "An Algorithm for Least-Squares Estimation of Nonlinear Parameters," *Journal of the Society for Industrial and Applied Mathematics*, Vol. 11, No. 2, 1963, pp. 431–441. doi:10.1137/0111030

[15] Bates, D. M., and Watts, D. G., *Nonlinear Regression Analysis and its Applications*, Wiley Series in Probability and Mathematical Statistics, Applied Probability and Statistics, Wiley, Chichester, NY, 1988.

[16] Yechout, T. R., and Braman, K. B., "Development and Evaluation of a Performance Modeling Flight Test Approach Based on Quasi Steady-State Maneuvers," NASA CR 170414, 1984.

[17] Maybeck, P. S., *Stochastic Models, Estimation, and Control*, Vol. 1, Academic Press, New York, 1979.

[18] Psiaki, M. L., "Backward-Smoothing Extended Kalman Filter," *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 5, 2005, pp. 885–894. doi:10.2514/1.12108

[19] Maybeck, P. S., *Stochastic Models, Estimation, and Control*, Vol. 1, Academic Press, New York, 1982.

[20] Gainer, T. G., and Hoffman, S., "Summary of Transformation Equations and Equations of Motion Used in Free Flight and Wind Tunnel Data Reduction and Analysis," NASA SP-3070, 1972.

[21] Akaike, H., "A New Look at the Statistical Model Identification," *IEEE Transactions on Automatic Control*, Vol. 19, No. 6, 1974, pp. 716–723. doi:10.1109/TAC.1974.1100705

[22] Ruppert, D., "Selecting the Number of Knots for Penalized Splines," *Journal of Computational and Graphical Statistics*, Vol. 11, No. 4, 2002, pp. 735–757. doi:10.1198/106186002853

[23] Kutner, M. H., Nachtsheim, C., and Neter, J., *Applied Linear Regression Models*, Vol. 4, McGraw–Hill/Irwin, New York, 2004.