2017 International Conference on
Unmanned Aircraft Systems (ICUAS)
June 13-16, 2017, Miami, FL, USA

# Flight Test Instrumentation System for Small UAS System Identification

Han-Hsun Lu, Joshua Harris, Vinicius Guimaraes Goecks, Ezekiel Bowden, John Valasek
{hanhsun.lu,joshua.a.harris,vinicius.goecks,p47thunderbolt,valasek}@tamu.edu
Vehicle Systems & Control Laboratory
Aerospace Engineering Department
Texas A&M University
College Station, TX, 77843–3141

***Abstract*—Common open-source flight controllers have the ability to log state data, but they often are unable to log at the rates required to obtain good identification of model characteristics. They are also often unable to log important flow parameters such as angle-of-attack and sideslip angle. This paper presents a custom flight test instrumentation system that is capable of providing accurate full-state and control surface deflection measurements for Small Unmanned Air Systems. The system consists of an embedded single-board computer integrated with a collection of sensors including commercial off-the-shelf air data and inertial navigation systems, in addition to direct measurement of control deflections. Developmental Flight Test Instrumentation data logging software runs onboard the computer. This software is released as open-source. Performance of the system is verified by flight test on a quarter scale Piper PA-18 Super Cub airframe. The Observer/Kalman Identification algorithm is used in post-processing to generate linear state-space models, which are then simulated with other input sets and compared to flight data. Results presented in the paper demonstrate that the integrated flight test instrumentation system is able to provide sufficiently accurate data at a high enough sample rate for the generation of quality linear state-space models.**

## I. Introduction

Many aerospace systems can be modeled as a linear state-space system describing perturbed motion around a trim point (i.e. an equilibrium condition of the full nonlinear system) as shown in Eq. (1):

$$\dot{\boldsymbol{x}}(t) = A(t)\boldsymbol{x}(t) + B(t)\boldsymbol{u}(t) \tag{1a}$$

$$\boldsymbol{y}(t) = C(t)\boldsymbol{x}(t) + D(t)\boldsymbol{u}(t) \tag{1b}$$

In Eq. (1), $\boldsymbol{x} \in \mathbb{R}^n$ is the perturbed state vector, $\boldsymbol{u} \in \mathbb{R}^m$ is a perturbed control vector, and $\boldsymbol{y} \in \mathbb{R}^p$ is the perturbed output vector. The matrix $A \in R^{n\times n}$ is the state matrix, the matrix $B \in \mathbb{R}^{n\times m}$ is the control distribution matrix, the matrix $C \in \mathbb{R}^{p\times n}$ is the output matrix, and the matrix $D \in \mathbb{R}^{p\times m}$ is the carry-through or feedforward matrix. If the 4-tuple $(A, B, C, D)$ is independent of time, the system is called a Linear Time Invariant (LTI) system. Three broad classes of problems result from the linear state-space formulation of Eq. (1):

1) *control*: given $\boldsymbol{x}$ and/or $\boldsymbol{y}$, with $(A, B, C, D)$ known, determine $\boldsymbol{u}$ to meet a control objective;
2) *estimation*: given $\boldsymbol{y}$ and $\boldsymbol{u}$, with $(A, B, C, D)$ known, determine the state $\boldsymbol{x}$; and
3) *identification*: given $\boldsymbol{y}$, $\boldsymbol{x}$, and $\boldsymbol{u}$, determine the model $(A, B, C, D)$.

This paper considers this last problem in the context of increasingly ubiquitous Small Unmanned Air Systems (SUAS). Common SUAS flight control systems such as the open-source ArduPilot and PX4 projects traditionally use Proportional-Integral-Derivative (PID) control structures as these methods can be tuned ad-hoc without requiring a system model [1], [2]. This is a consequence of the scarcity of high-quality flight models for SUAS, since control algorithms that are model-based cannot be used.

Flight models are traditionally generated by several methods of varying complexity and fidelity. At the low-fidelity end and for preliminary design [3], linear parametric models of an aircraft can be obtained using linear aerodynamics and empirical techniques [4] with errors in the 10-20% range for the most critical parameters. Aerodynamic prediction codes are then generally used to populate aerodynamics databases over the range of the flight envelope [5], [6]. Wind tunnel tests are then used for verification and validation of the computational models. Finally, models can be obtained from experimental flight data. This is generally the most accurate and, for manned aircraft, the most expensive approach.

Several classes of algorithms exist for generating models of systems from experimental data. These algorithms can be partitioned into *parameter identification* algorithms, which determine parametric models of a system, and *system identification* algorithms, which determine non-parametric models. Widely used parameter identification algorithms for aircraft include the Modified Maximum Likelihood Estimation (MMLE) [7], [8], System Identification Programs for Aircraft (SIDPAC) [9], and artificial neural network based approaches [10], [11]. Common system identification algorithms include the Eigensystem Realization Algorithm (ERA) [12], Observer/Kalman Identification (OKID) [13], the Comprehensive Identification from Frequency Responses (CIFER®) algorithm [14], Free Re-

sponse Functions [15], and Observer/Controller Identification (OCID) [16]. Reference [17] provides a historical overview of system identification approaches for flight vehicles.

System identification can be an expensive endeavor for manned aircraft. SUAS have the advantage of being relatively inexpensive to test in general, and in many cases it is not only less expensive but faster to obtain models of SUAS through system identification. Doing so requires the ability to record state and control time histories at sufficiently fast rates to prevent aliasing and thereby accurately capture dynamic responses.

This paper introduces a new flight test instrumentation system for SUAS parameter and system identification flight, superseding an earlier system used in previous work by the authors [18]–[20]. The new instrumentation system requires no external or proprietary software for data logging. It offers modularity and extra digital and analog input/output ports for additional sensors. Since the system comprises commercial-off-the-self (COTS) components and system on a chip (SoC) hardware, it meets the low size, weight, power and cost (SWaP-C) requirements common in aerospace applications. Its high-frequency sampling capability, at a maximum of 100 Hz, is also ideal for the system identification and modeling applications.

The paper is organized as follows. Section II concisely reviews aircraft equations of motion. Section III outlines the OKID algorithm for identification of discrete linear state-space systems. The instrumentation system and its specifications, including details of the airframe and avionics, are presented in Section IV. Flight results and identified models are presented in Section V and conclusions are presented in Section VI.

## II. Aircraft State-Space Modeling

Aircraft can be modeled as rigid bodies using classical mechanics as a set of first-order nonlinear ordinary differential equations Eq. (2):

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(t, \boldsymbol{x}, \boldsymbol{u}) \tag{2}$$

In Eq. (2), $\boldsymbol{x} \in \mathbb{R}^n$ is the state vector, $\boldsymbol{u} \in \mathbb{R}^m$ is the control vector, and $\boldsymbol{f} : \mathbb{R}^+ \times \mathbb{R}^n \times \mathbb{R}^m \mapsto \mathbb{R}^n$ is a nonlinear function of the state and control. Interested readers can consult Refs. [3], [21], [22] for an in-depth discussion of flight mechanics and aircraft stability and control.

### A. Aircraft Nonlinear Equations of Motion

The aircraft body-fixed frame $\mathcal{B} : \{\hat{\boldsymbol{x}}_\mathrm{b}, \hat{\boldsymbol{y}}_\mathrm{b}, \hat{\boldsymbol{z}}_\mathrm{b}\}$ is defined such that $\hat{\boldsymbol{x}}_\mathrm{b}$ points out the nose, $\hat{\boldsymbol{y}}_\mathrm{b}$ points out the right wing, and $\hat{\boldsymbol{z}}_\mathrm{b}$ points down completing a right-handed coordinate system, as seen in Figure 1 through the definition of the total velocity vector $V_\mathrm{T}$ and its components.

Following from these definitions, the *stability axis* system is constructed from the body-axis system by rotating along the angle-of-attack $\alpha$ such that the new $\hat{\boldsymbol{x}}_\mathrm{s}$ axis is aligned with the projection of the total velocity vector $V_\mathrm{T}$ along the $xz$ plane. In this section, the aircraft equations of motion will be developed in the body axis system, with later linear models

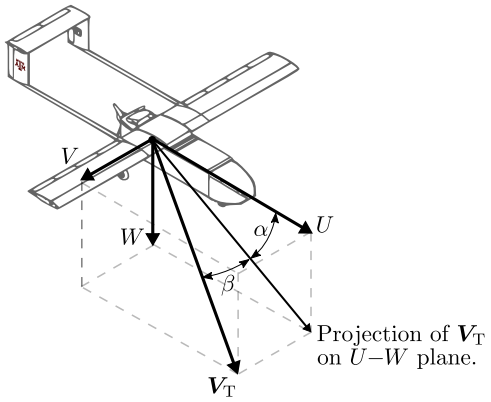


Fig. 1: Definition of body-axis velocity components and flow angles.

expressed in stability axes [23]. The dynamics of an aircraft can be written compactly in vector matrix form as Eq. (3),

$$\begin{aligned} m(\dot{\boldsymbol{V}}_\mathrm{T} + \boldsymbol{\omega} \times \boldsymbol{V}_\mathrm{T}) &= m\boldsymbol{g} + \boldsymbol{F}_\mathrm{A} + \boldsymbol{F}_\mathrm{T} \\ I\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times I\boldsymbol{\omega} &= \boldsymbol{\ell}_\mathrm{A} + \boldsymbol{\ell}_\mathrm{T} \end{aligned} \tag{3}$$

Here, $\boldsymbol{V}_\mathrm{T} = U\hat{\boldsymbol{x}}_\mathrm{b} + V\hat{\boldsymbol{y}}_\mathrm{b} + W\hat{\boldsymbol{z}}_\mathrm{b}$ is the translational velocity vector, $\boldsymbol{\omega} = P\hat{\boldsymbol{x}}_\mathrm{b} + Q\hat{\boldsymbol{y}}_\mathrm{b} + R\hat{\boldsymbol{z}}_\mathrm{b}$ is the angular velocity vector, $\boldsymbol{g}$ is the gravity vector, $\boldsymbol{F}_\mathrm{A}$ and $\boldsymbol{F}_\mathrm{T}$ are the applied forces due to aerodynamic and thrust effects, $I \in \mathbb{R}^{3\times3}$ is the inertia tensor, and $\boldsymbol{\ell}_\mathrm{A}$ and $\boldsymbol{\ell}_\mathrm{T}$ are the applied moments due to aerodynamic and thrust effects. The components of the moment vectors are denoted $\{L, M, N\}$ for rolling, pitching, and yawing moments respectively [24].

To fully specify the aircraft response a set of six kinematic equations (three translational and three rotational) are required. These equations are dependent on the choice of position-level coordinates. An inertial position vector in a North-East-Down (NED) frame and the 3-2-1 Euler angle set $\{\psi, \theta, \phi\}$ are common choices. The kinematic differential equations for the 3-2-1 Euler angles can be found in any textbook on flight mechanics such as Ref. [22], and are reproduced below:

$$\begin{aligned} \dot{\Phi} &= P + Q\sin\Phi\tan\Theta + R\cos\Phi\tan\Theta \\ \dot{\Theta} &= Q\cos\Phi - R\sin\Phi \\ \dot{\Psi} &= (Q\sin\Phi + R\cos\Phi)\sec\Theta \end{aligned} \tag{4}$$

From Eqs. (3–4), dropping position-level coordinates for the translational motion, the aircraft state vector is Eq. (5),

$$\boldsymbol{X} = [U, V, W, P, Q, R, \Psi, \Theta, \Phi]^\mathsf{T} \tag{5}$$

### B. Decoupling the Equations of Motion

For a conventional aircraft at a steady, level flight condition at near-zero bank angle, the equations of motion can be decoupled into two sets: longitudinal (i.e. pitch axis) and lateral/directional (lat/d, i.e. roll and yaw axes). The longitudinal state vector consists of the variables $U$, $W$, $Q$, and $\Theta$ in body-axes, and the lateral/directional state vector consists of the variables $V$, $P$, $R$, $\Phi$, and $\Psi$ in body-axes. For conventional aircraft the longitudinal controls are throttle $\delta_\mathrm{T}$ and elevator

$\delta_{\mathrm{E}}$, and the lat/d controls are aileron $\delta_{\mathrm{A}}$ and rudder $\delta_{\mathrm{R}}$ [3]. For non-conventional aircraft the available control surfaces can be used directly or ganged together to form pseudo-control effectors [25]. For the latter approach a control allocation algorithm is usually required to obtain the actual surface deflections needed to obtain a pseudo-control command [26].

### C. Linear Aircraft Models

The aircraft dynamics can be trimmed at a flight condition with states $\boldsymbol{X}_1$ and controls $\boldsymbol{U}_1$ such that the dynamics reduce to $\mathbf{0} = \boldsymbol{f}(\boldsymbol{X}_1, \boldsymbol{U}_1)$. An appropriate linearization technique can be applied to Eq. (2) to generate an LTI model

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\boldsymbol{u} \tag{6}$$

where $\boldsymbol{x}$ and $\boldsymbol{u}$ are perturbations on the state and control respectively. The full nonlinear state is then $\boldsymbol{X} = \boldsymbol{X}_1 + \boldsymbol{x}$, and the nonlinear control is $\boldsymbol{U} = \boldsymbol{U}_1 + \boldsymbol{u}$; from these relations it is straightforward to convert between full states and perturbed states for control law implementation, plotting, and other uses [3]. In the stability axis system the parametric longitudinal linear state-space equations are:

$$\begin{Bmatrix} \dot{u} \\ \dot{\alpha} \\ \dot{q} \\ \dot{\theta} \end{Bmatrix} = \begin{bmatrix} X'_{\mathrm{u}} & X'_{\alpha} & X'_{\mathrm{q}} & -g\cos\Theta_1 \\ Z'_{\mathrm{u}} & Z'_{\alpha} & Z'_{\mathrm{q}} & -g\sin\Theta_1 \\ M'_{\mathrm{u}} & M'_{\alpha} & M'_{\mathrm{q}} & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{Bmatrix} u \\ \alpha \\ q \\ \theta \end{Bmatrix} + \begin{bmatrix} X'_{\delta_{\mathrm{E}}} & X'_{\delta_{\mathrm{T}}} \\ Z'_{\delta_{\mathrm{E}}} & Z'_{\delta_{\mathrm{T}}} \\ M'_{\delta_{\mathrm{E}}} & M'_{\delta_{\mathrm{T}}} \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \delta_{\mathrm{E}} \\ \delta_{\mathrm{T}} \end{Bmatrix} \tag{7}$$

The primed quantities result from decoupling the $\dot{\alpha}$ and $\dot{q}$ equations. The angle-of-attack $\alpha$ is synthesized from the velocity component $w$ and the steady-state velocity $U_1$ as $\alpha \approx w/U_1$. This relation uses the small-angle approximation and is valid for flight conditions that are steady and level. It is therefore valid for linear models generated for these flight conditions.

The longitudinal dynamics are a fourth-order system that nominally exhibit two standard second-order modes: a high frequency, highly-damped mode exhibited mainly in body-axis pitch rate $q$ and angle-of-attack $\alpha$ (*short period*) and a low frequency, lightly-damped mode exhibited in airspeed $u$ and pitch attitude angle $\theta$ (*phugoid*). For aircraft with relaxed static stability it is common for the system to exhibit two first-order modes and a non-standard third-order mode referred to as the *third-oscillatory* mode [3].

In the stability axis system the lat/d linear state-space model is:

$$\begin{Bmatrix} \dot{\beta} \\ \dot{p} \\ \dot{r} \\ \dot{\phi} \end{Bmatrix} = \begin{bmatrix} \frac{Y_\beta}{U_1} & \frac{Y_{\mathrm{p}}}{U_1} & 1+\frac{Y_{\mathrm{r}}}{U_1} & \frac{g\cos\Theta_1}{U_1} \\ L'_\beta & L'_{\mathrm{p}} & L'_{\mathrm{r}} & 0 \\ N'_\beta & N'_{\mathrm{p}} & N'_{\mathrm{r}} & 0 \\ 0 & 1 & \tan\Theta_1 & 0 \end{bmatrix} \begin{Bmatrix} \beta \\ p \\ r \\ \phi \end{Bmatrix} + \begin{bmatrix} \frac{Y_{\delta_{\mathrm{A}}}}{U_1} & \frac{Y_{\delta_{\mathrm{R}}}}{U_1} \\ L'_{\delta_{\mathrm{A}}} & L'_{\delta_{\mathrm{R}}} \\ N'_{\delta_{\mathrm{A}}} & N'_{\delta_{\mathrm{R}}} \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \delta_{\mathrm{A}} \\ \delta_{\mathrm{R}} \end{Bmatrix} \tag{8}$$

The primed terms result from decoupling the $\dot{p}$ and $\dot{r}$ equations, and the sideslip angle $\beta$ is obtained from the approximation $\beta \approx v/U_1$. It is valid for linear models generated for these flight conditions for the same reason stated previously. The linearized heading angle kinematics $\dot{\psi} = r$ from Eq. (4) are not shown in Eq. (8) because from an identification perspective, the relationship of $\dot{\psi}$ is known exactly and therefore should not be identified. The equations for $\dot{\theta}$ and $\dot{\phi}$ are included in Eqs. (7–8), respectively, as the other states are not independent of $\theta$ and $\phi$.

The lat/d state-space model Eq. (8) forms a fourth-order system with two standard first-order modes and a standard second-order mode for most conventional aircraft configurations. The *roll* mode is a first-order mode, which, as the name implies, is primarily composed of the aircraft body-axis roll rate $p$. The other first-order mode is referred to as *spiral* and is typically a very slow mode primarily composed of roll ($\phi$) and heading angle ($\psi$). The second-order mode is known as the *Dutch roll* and is an oscillatory motion exhibited mostly in body-axis yaw rate ($r$), sideslip angle ($\beta$), and body-axis roll rate. It is a mode that often needs improved damping for acceptable flying qualities [22].

Equations (7–8) are the basic models that are desired to be identified from the techniques presented in this work, and subsequently used to determine modal characteristics. The output equation Eq. (1b) is not identified, as 1) full state measurements are available and 2) models are assumed to be strictly proper (i.e. $D = 0$) so the system input-output relationship is fully encoded in the state equation. The user can choose appropriate $C$ and $D$ matrices after identification to represent the measurements obtained from the system after the flight test instrumentation is removed.

## III. Observer/Kalman Filter Identification

During the 1980s many system identification methods were developed to identify linear state-space models for spacecraft and aircraft with flexible structural characteristics. The majority of these methods are based on Fast Fourier Transform (FFT), Maximum Likelihood Estimation (MLE), and least squares [8]. A drawback of the FFT and MLE methods is that a somewhat rich input is required to prevent ill-conditioned computation. The Observer/Kalman Filter Identification (OKID) method was developed in the 1990s by Juang [15]. It is a direct

Kalman filter gain approach that is formulated in the time-domain and is capable of handling general response data. This is especially valuable for aircraft modeling since pure impulse excitations are difficult to apply and the noise/signal ratio of sensing data are usually high. It also has the benefit of allowing for nonzero initial conditions and does not require the response to reach steady-state before collection. The present work is an extension of the concept that OKID can be successfully used to identify state-space models of flight vehicles [27]. OKID has the benefit of only requiring input/output time histories to perform system identification. This reduces the amount of a priori system specific information required to perform system identification and increases the process of vehicle modeling.

The basic formulation of the OKID algorithm begins with the linearized, discrete-time, state-space equations augmented with an observer gain:

$$\begin{aligned} \boldsymbol{x}(k+1) &= \bar{A}\boldsymbol{x}(k) + \bar{B}\boldsymbol{v}(k) \\ \boldsymbol{y}(k) &= C\boldsymbol{x}(k) + D\boldsymbol{u}(k) \end{aligned} \tag{9}$$

where $\boldsymbol{x}(k) \in \mathbb{R}^n$, $\boldsymbol{y}(k) \in \mathbb{R}^m$, $\boldsymbol{u}(k) \in \mathbb{R}^r$, are state, output and control inputs with

$$\begin{aligned} \bar{A} &= A + GC \\ \bar{B} &= [B + GD, -G] \\ \boldsymbol{v}(k) &= \begin{bmatrix} \boldsymbol{u}(k) \\ \boldsymbol{y}(k) \end{bmatrix} \end{aligned} \tag{10}$$

and $G \in \mathbb{R}^{n\times m}$ is an arbitrary matrix chosen to make the matrix $\bar{A}$ stable. Assuming zero initial conditions and integer $p$ satisfying $CA^kB \approx 0$ for $k \geqslant p$, substituting and iterating through each time step using Equation (9), the Observer Markov Parameters (OMP) comprised of a input-output relationship becomes

$$\bar{\boldsymbol{y}} = C\bar{A}^p\boldsymbol{x} + \bar{Y}\bar{V} \tag{11}$$

where

$$\begin{aligned} \bar{y} &= \begin{bmatrix} \boldsymbol{y}(p) & \boldsymbol{y}(p+1) & \cdots & \boldsymbol{y}(l-1) \end{bmatrix} \\ \bar{Y} &= \begin{bmatrix} D & C\bar{B} & C\bar{A}\bar{B} & \cdots & C\bar{A}^{(p-1)}\bar{B} \end{bmatrix} \\ \bar{V} &= \begin{bmatrix} \boldsymbol{u}(p) & \boldsymbol{u}(p+1) & \cdots & \boldsymbol{u}(l-1) \\ \boldsymbol{v}(p-1) & \boldsymbol{v}(p) & \cdots & \boldsymbol{v}(l-2) \\ \boldsymbol{v}(p-2) & \boldsymbol{v}(p-1) & \cdots & \boldsymbol{v}(l-3) \\ \vdots & \ddots & \ldots & \vdots \\ \boldsymbol{v}(0) & \boldsymbol{v}(1) & \cdots & \boldsymbol{v}(l-p-1) \end{bmatrix} \end{aligned} \tag{12}$$

The matrix $\bar{Y}$ is partitioned with the system Markov parameters such that

$$\bar{Y} = \begin{bmatrix} D & C\bar{B} & C\bar{A}\bar{B} & \cdots & C\bar{A}^{(p-1)}\bar{B} \end{bmatrix} = \begin{bmatrix} Y_0 & Y_1 & Y_2 & \cdots & Y_p \end{bmatrix} \tag{13}$$

from which the OMP are obtained.

$$\begin{aligned} \bar{Y}_0 &= D \\ \bar{Y}_k &= C\bar{A}^{(k-1)}\bar{B} \\ &= \begin{bmatrix} C(A+GC)^{(k-1)}(B+GD) & -C(A+GC)^{(k-1)}G \end{bmatrix} \\ &= \begin{bmatrix} \bar{Y}_k^{(1)} & -\bar{Y}_k^{(2)} \end{bmatrix} \quad k = 1, 2, 3, \ldots \end{aligned} \tag{14}$$

The general relationship between the actual system Markov parameters and the OMP can be shown to be

$$\begin{aligned} D &= Y_0 = \bar{Y}_0 \\ Y_k &= Y_k^{(1)} - \sum_{i=1}^{k} \bar{Y}_i^{(2)} Y_{(k-i)} \quad \text{for } k = 1, \ldots, p \\ Y_k &= -\sum_{i=1}^{p} \bar{Y}_i^{(2)} Y_{(k-i)} \quad \text{for } k = p+1, \ldots, \infty \end{aligned} \tag{15}$$

The next step is to use a singular value decomposition (SVD) on the Hankel matrix:

$$\begin{aligned} H(k-1) &= \begin{bmatrix} Y_k & Y_{k+1} & \cdots & Y_{k+\beta-1} \\ Y_{k+1} & Y_{k+2} & \cdots & Y_{k+\beta} \\ \vdots & \vdots & \ddots & \vdots \\ Y_{k+\alpha-1} & Y_{k+\alpha} & \cdots & Y_{k+\alpha+\beta-2} \end{bmatrix} \\ H(0) &= P_n \Sigma Q_n^{\mathsf{T}} \end{aligned} \tag{16}$$

The ERA is then used to solve the Hankel matrix for the desired state-space realization $(A, B, C, D)$:

$$\begin{aligned} \hat{A} &= \Sigma_n^{-1/2} P_n^{\mathsf{T}} H(1) Q_n \Sigma_n^{-1/2} \\ \hat{B} &= \Sigma_n^{1/2} Q_n^{\mathsf{T}} \\ \hat{C} &= P_n \Sigma_n^{1/2} \\ \hat{D} &= Y_0 \end{aligned} \tag{17}$$

Note that $\hat{A}$, $\hat{B}$, and $\hat{C}$ are the estimated system matrices determined using OKID. The $(\hat{A}, \hat{B}, \hat{C}, \hat{D})$ represent the identified discrete linear state-space system:

$$\begin{aligned} \boldsymbol{x}(k+1) &= \hat{A}\boldsymbol{x}(k) + \hat{B}\boldsymbol{u}(k) \\ \boldsymbol{y}(k) &= \hat{C}\boldsymbol{x}(k) + \hat{D}\boldsymbol{u}(k) \end{aligned} \tag{18}$$

## IV. Flight Instrumentation System

### A. Airframe Description

The 1/4 scale Hangar-9 PA-18 Super Cub SUAS is the base air vehicle. The $700 commercial off-the-shelf (COTS) Super Cub has a wingspan of 2.7 m, empty weight of 7.5 kg, and endurance of 30-45 minutes with extended batteries. It has a 295 kV E-Flite Power 110 electric brushless motor, 85 A HV brushless Electronic Speed Controller (ESC), and an APC 19×10E propeller. Figure 2 shows the Super Cub airframe used for the work detailed in this paper.

Fig. 2: Hangar-9 1/4-Scale PA-18 Super Cub

### B. Instrumentation

The instrumentation consists of an embedded computer and a selection of sensors to record system states and control effector positions. Several requirements are identified:

1) The instrumentation system should be independent of the flight control system.
2) The instrumentation system must be capable of logging aircraft states and control effector positions at rates no less than 100 Hz.
3) The instrumentation system should use COTS components as much as possible.
4) The instrumentation system should have a modular software architecture and be highly extensible.
5) The instrumentation system should be easily moved between different vehicles.

It is desired to keep the instrumentation system separate from the flight control system so that faults in the instrumentation do not affect anything in a flight-critical path for safety-of-flight reasons. A related benefit of this separation is the ability to upgrade the flight control system separately from the instrumentation. The previous system used by the authors integrated additional sensors into the flight software on a Pixhawk autopilot, which required a custom firmware to be flashed prior to system identification flights. This precluded easy firmware updates. The requirement for 100 Hz logging results from the need successfully identify rigid-body fixed-wing aircraft modes based on engineering judgment. COTS components are preferred to reduce development time and cost and to ensure easy access to replacement components.

A modular software architecture is desired to ensure that the system is able to support multiple sensors and airframes without requiring significant modification of the code. It is also desired to allow interfacing with other systems for applications such as onboard, real-time model identification and prognostics and health monitoring.

Finally, it is desired that the system be easily transplanted between different vehicles. The Vehicle Systems & Control Laboratory operates several types of fixed-wing aircraft and desires the ability to generate high quality linear models for each type.

Detailed descriptions of the system components follow.

*1) Instrumentation Computer:* A Texas Instruments BeagleBone Black Rev. C embedded ARM SoC is used as the base computing platform. Table I lists hardware specifications for the BeagleBone Black. The BeagleBone requires relatively low power and is inexpensive at a list price $45. The BeagleBone's small size and low weight allow for it to be easily mounted on a variety of fixed-wing platforms. A Micro SD card reader allows for large amounts of sensor data to be logged. The BeagleBone Black is attached to the bottom of the Super Cub fuselage.

TABLE I: BeagleBone Black Specifications [28]

| | |
|---|---|
| Processor | TI AM335x Sitara 1 GHz ARM® Cortex A-8 |
| RAM | 512 MB DDR3 |
| Microcontrollers | 2 × PRU 32-bit |
| I/O | USB, Fast Ethernet, HDMI, 2 × 46 GPIO, 4 usable 3.3 V TTL UART |

*a) Software:* The BeagleBone Black runs a stripped-down version of Debian "Jessie" Linux® from the onboard eMMC. A custom software package called the "Developmental Flight Test Instrumentation" (DFTI) handles reading, parsing, and logging sensor data. Sensor data is logged as comma-separated values (CSV) files to allow for easy programmatic access in languages such as MATLAB® or Python. The DFTI software uses a multithreaded design to allow for each sensor to be read as quickly as possible. Due to this design choice and since DFTI is designed primarily for offline batch identification algorithms, each sensor is logged in a separate CSV file to avoid repeated or missing data resulting from concurrency and parallelism issues. The system additionally has a timer set to periodically flush the output stream buffers to ensure data is written to disk. Each sensor log contains a Unix timestamp for each line in order to facilitate data reconstruction in post-processing. Data logged by the instrumentation system and the flight control system can be matched and compared using the GPS time. A MATLAB package for post-processing the data is included as part of the DFTI software.

DFTI was implemented as a Linux command line application in C++11 using the Qt5 framework to allow for easy concurrency and threading with the signal/slot paradigm. While this introduces some overhead due to the Qt5 event loop, testing has demonstrated that the system is able to log sensor data at the desired 100 Hz although the actual rate at which data is written to the output files is slightly lower.

The software uses an object-oriented design to promote code reuse and maximize extensibility. Serial-based sensors for example are all derived from a base class that implements most common functionality for serial ports and the signal/slot messaging; this also ensures a consistent interface between sensors. Additionally, the code is split into modules that are each compiled into shared (i.e. dynamically-linked) libraries to allow for reuse of compiled code between different programs. Currently, there are two programs as part of the DFTI suite: the main logging program and a small test program for troubleshooting individual sensors.

Due to the use of Qt5 libraries for interfacing with the system serial ports and for handling I/O, it is likely that the DFTI software could be compiled for macOS or Windows systems as well as Linux with minor modification. This has not been tested and support for other operating systems is not planned at this point. DFTI is available as an open-source project under the liberal BSD 2-clause license on GitHub, and includes Doxygen documentation of all classes and methods[29].

DFTI is designed to be configured via ubiquitous INI files, and allows for both system-wide and user-specific configuration files. Configuration parameters include serial ports and baud rates for sensors, the ability to enable/disable sensors, and other sensor-specific configurations. To facilitate deployment, the DFTI software is packaged as a Debian package to allow installation through the system package manager. The package

contains the main logging program (`dfti`) as well as the simplified test interface (`dftitest`). The full assembled data acquisition system running DFTI is shown in Figure 3.


Fig. 3: DFTI system side view.

*2) Air Data System:* An Aeroprobe Corporation Micro Air Data Computer (μADC) is used to record the true airspeed ($V_T$), flow angles (angle-of-attack $\alpha$ and sideslip angle $\beta$), altitude ($h$), and total and static pressures ($p_{tot}$ and $p_{stat}$, respectively). The μADC is connected to a five-hole probe (5HP) and calculates the air data from the five total pressure ports and static ports on the 5HP at 100 Hz. Table II presents the specifications of the μADC, including limits on the available measurements. The μADC outputs data over an RS-232

TABLE II: Air Data System Specifications [30]

| | |
|---|---|
| Max Airspeed Error at Cruise | $\pm 2.5\,\mathrm{m/s}$ |
| Flow Angle Range | $\pm 20°$ |
| Flow Angle Resolution | $0.1°$ |
| Max Flow Angle Error | $\pm 1.0°$ |
| Max Calibrated Airspeed | $64\,\mathrm{m/s}$ |
| Minimum Reported Airspeed | $7.9\,\mathrm{m/s}$ |
| Airspeed Resolution | $0.36\,\mathrm{m/s}$ |
| Weight (μADC) | $135\,\mathrm{g}$ |
| Size (μADC) | $6.6\,\mathrm{cm} \times 7.87\,\mathrm{cm} \times 3.3\,\mathrm{cm}$ |
| Current Draw | $<390\,\mathrm{mA}$ @ $12\,\mathrm{V}$ DC |

serial connection in an ASCII data format. An RS-232 to 3.3 V TTL converter is used to shift the voltage levels of the serial connection for interfacing with the instrumentation computer. Initially an RS-232 to USB converter was used, but testing demonstrated that the fast sampling of the μADC along with other sensors was sufficient to saturate the USB bus on the computer. This caused the watchdog timer to count down and trigger an interrupt.

The μADC is mounted inside the fuselage of the Super Cub while the 5HP is mounted under the right wing and is connected by pressure tubing though the half-span to the μADC. This is a compromise between accuracy and convenience as long tubing lengths reduce the sensor accuracy. A future airframe modification will place the μADC close to the 5HP in the wing and run data and power lines to the fuselage. Figure 4 shows the mounting of the 5HP.

Fig. 4: Detail view of 5-Hole Probe mount.

*3) Inertial Navigation System:* A VectorNav VN-200 Inertial Navigation System (INS) is used to log the majority of aircraft states. The VN-200 is a small, high-quality, ruggedized INS ideally suited for SUAS applications. Table III lists selected specifications for the VN-200. The VN-200 supports

TABLE III: VN-200 INS Specifications [31]

| | |
|---|---|
| Accuracy (Pitch/Roll) | $0.5\,°\mathrm{RMS}$ |
| Accuracy (Heading) | $0.5\,°\mathrm{RMS}$ |
| Angular Resolution | $<0.05°$ |
| Gyro Noise Density | $0.0035\,°\,\sqrt{\mathrm{Hz}}/\mathrm{s}$ |
| Gyro Alignment Error | $\pm 0.05°$ |
| Gyro Resolution | $<0.02\,°/\mathrm{s}$ |

both RS-232 and 3.3 V TTL UART serial ports as well as SPI communication. For the flight test instrumentation UART was chosen as a simple communications protocol. The VN-200 supports both ASCII and binary protocols for encoding the sensor data. The protocol choice and sensor outputs to communicate can be configured in the VectorNav Sensor Explorer software. Binary encoding of the data was used to maximize the data rates and the signals selected for logging include:

- GPS time (ns)
- attitude quaternion
- body-axis angular rates ($\mathrm{rad/s}$)
- latitude/longitude/altitude ($\mathrm{deg/deg/m}$)
- North-East-Down velocity ($\mathrm{m/s}$)
- body-axis accelerations ($\mathrm{m/s^2}$)

The GPS time signal from the INS is used to match times with other systems and can also be used to set the system clock on the BeagleBone Black to the correct date and time when network time is unavailable.

The VN-200 IMU unit is mounted as close to the vehicle center of gravity as possible while the GPS antenna is placed near the top of the fuselage to minimize interference. Figure 5 is shows the VN-200 INS and GPS antenna.

*4) Control Surface Deflection Measurements:* Previous iterations of VSCL instrumentation recorded commanded surface positions as pulse width modulated signals. This has several drawbacks, including identifying the effect of the actuator dynamics as part of the aircraft dynamics and the inability to detect failures in the actuator. Direct measurement of the

Fig. 5: VectorNav VN-200 Inertial Navigation System.

control surface deflections is desired to address these issues. In combination with the actuator commands logged on the flight control system, direct surface position measurements can also be used to identify models of the actuator dynamics. Additionally, control surface deflection measurements can provide additional feedback signals for techniques such as Control Rate Weighting [32] to prevent phenomena such as pilot-induced oscillation, as well as reducing wear on actuators.

The Super Cub model has the following aerodynamic control surfaces: left aileron $\delta_{\mathrm{A_L}}$, right aileron $\delta_{\mathrm{A_R}}$, left trailing edge flap (TEF) $\delta_{\mathrm{TEF_L}}$, right TEF $\delta_{\mathrm{TEF_R}}$, left elevator $\delta_{\mathrm{E_L}}$, right elevator $\delta_{\mathrm{E_R}}$, and rudder $\delta_{\mathrm{R}}$. The aileron, elevator, and TEF surfaces are ganged so there are the three standard aerodynamic controls (aileron, elevator, and rudder) and the TEFs are deflected symmetrically. Using the convention found in Klein and Morelli [9], Eq. (19) is one such combination for aileron and elevator. Positive surface deflections generate negative aerodynamic moments under this convention. Individual control effectors follow a right-hand rule for determining positive deflections.

$$\delta_{\mathrm{A}} = \frac{1}{2}(\delta_{\mathrm{A_R}} - \delta_{\mathrm{A_L}}) \qquad \delta_{\mathrm{E}} = \frac{1}{2}(\delta_{\mathrm{E_L}} + \delta_{\mathrm{E_R}}) \qquad (19)$$

Ganging is accomplished through RC servomechanism Y-splitters and servo orientation. However, since each surface is controlled by its own servomechanism the airframe could be modified to allow independent actuation of each control surface by a control law and control allocation algorithm. Independent actuation would also allow differential deflection of roll control effectors to reduce adverse yaw effects. As a result each surface has a potentiometer mounted for position feedback in order to allow for identifying models with each surface as an input.

Seven BI Technologies 6127V1A360L.5FS linear potentiometers are installed on the each surface to measure the deflection angles. The potentiometers are rigidly mounted to the airframe and are connected to the control surfaces by standard RC control horns and rods. As a result of this installation scheme the control surface deflection measurements are the most invasive part of the instrumentation. An Arduino Uno reads the analog output of each potentiometer, which is then converted to a digital signal by the 10-bit onboard analog-to-digital converter (ADC). The custom Arduino firmware then converts the voltage signals to deflection angles in degrees and logs the data at approximately 250 Hz. The data is communicated to the main instrumentation computer over 3.3 V TTL UART. Figure 6 shows the setup of the linear potentiometers prior to the installation in the aircraft.

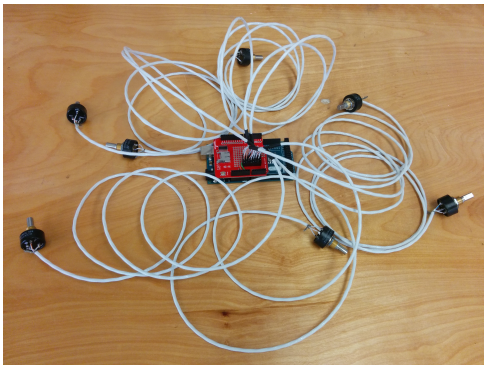

Fig. 6: Data acquisition setup for measurement of control surface deflection angles.

## V. Flight Test Results

### A. Test Plan

Several flight tests were conducted during early morning in order to have minimum winds aloft and reduce the effect of exogenous inputs on the identified models. To improve data quality all test data was acquired after the aircraft is trimmed.

### B. Input Excitation

Two general approaches are applied in designing inputs for system identification. The first approach requires no *a priori* knowledge of the behavior of the dynamical system. Inputs such as impulses and frequency sweeps fall in this category, where the goal is to excite all of the dynamic modes in a large range of frequency. The second approach requires *a priori* understanding of the system and designs the input with respect to the dynamic modes. Square waves are used to construct this particular kind of excitation. To excite the dynamic modes without flexible structural modes the maneuver time length, control surface magnitude, maneuver sequence, and input correlations are specified [9]. The latter approach is used for SUAS flight testing due to the *a priori* knowledge of the natural frequencies of the structure, and doublets are used for excitation to reduce the excitation error induced by the pilot. Multiple control inputs must be coordinated to maximize data content and to ensure the responses do not exceed the limitations for model structure validity. Figure 7 shows an input excitation sequence from flight test used for lat/d system identification. The excitation frequency is chosen with regard to the expected frequency of the dynamic modes; the input magnitude is selected in order to ensure the vehicle responds in the linear range.

The input excitations are done using two separate test sequences: one for longitudinal modes and the other for lateral/directional modes. To get the most precise data all maneuvers are performed on a 1-2-3 type count. For example, an aileron doublet is performed as a two count (or 2 second)

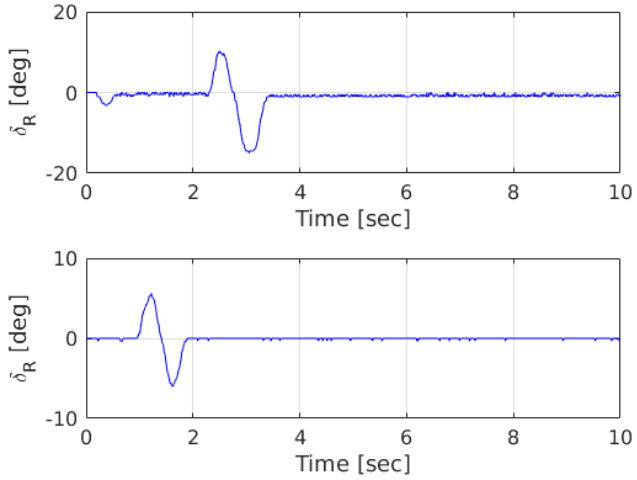


Fig. 7: Coordinated input excitation to perturb lateral/directional modes.

maneuver: one count of left aileron followed by one count of right aileron.

*1) Longitudinal Input Sequences:* For the best response from the aircraft the following sequence is applied: down elevator, up elevator, full throttle, and finally cut throttle. This is performed as a five count maneuver: one count down elevator, one count up elevator, two counts of full throttle, and one count cut throttle. Using this excitation sequence consistently results in three periods of the phugoid mode under stick free conditions.

*2) Lateral/Directional Input Sequences:* The spiral mode is perturbed using a four count maneuver: right rudder, left rudder, right aileron, and left aileron. This sequence allows for at least one complete spiral revolution. The Dutch Roll mode is perturbed using a four count maneuver: right aileron, left aileron, right rudder, and left rudder. The modification of applying the rudder doublet at the end of the input sequence allows for a visible Dutch roll mode and then a complete spiral revolution.

### C. Flight Test Results

The flight was conducted with wind speed around 2 mph gusting to 5 mph prevailing from North Northeast (NNE) of the flight path. Figure 8 shows the results from a lat/d maneuver set. The identified states are shown as dashed red while the actual flight data is solid blue. The identified model tracks all four states and system frequencies very well. Note that the discrepancy in sideslip angle after two seconds is caused by the $\pm 20^\circ$ measuring limit imposed by the µADC.

Mode Singular Values (MSV), Modal Controllability Index (MCI), and Modal Observability Index (MOI) are used for the selection of the identified modes [33]. The indicators are calculated by

$$\begin{aligned} MCI &= 100 \cdot \frac{|B_{\mathrm{m}}|}{\max |B_{\mathrm{m}}|} \\ MOI &= 100 \cdot \frac{|C_{\mathrm{m}}|}{\max |C_{\mathrm{m}}|} \\ MSV &= 100 \cdot \frac{\frac{\sqrt{|B_{\mathrm{m}}| \cdot |C_{\mathrm{m}}|}}{|1-|\zeta||}}{\max \frac{\sqrt{|B_{\mathrm{m}}| \cdot |C_{\mathrm{m}}|}}{|1-|\zeta||}} \end{aligned} \tag{20}$$

where $B_{\mathrm{m}} \in \mathbb{R}^{nm \times r}$ is the modal input matrix, $nm$ is the number of modes, $C_{\mathrm{m}} \in \mathbb{R}^{m \times nm}$ is the modal output matrix, and $\zeta \in \mathbb{R}^{nm}$ is the eigenvalue vector. A filter is added using local regression weighted linear least squares and a second degree polynomial to filter out the measurement noise for the state measurements. Filtered state measurements are then used for system identification calculations. The identified modes and characteristics using OKID are shown in Table IV.

TABLE IV: Super Cub lateral/directional dynamic modes.

| **Mode** | **Spiral** | **Roll** | **Dutch Roll** |
|---|---|---|---|
| **Eigenvalue** | -0.09 | -1.5492 | $-3.6919 \pm$ j3.1821 |
| **Damping Ratio** | — | — | 0.7575 |
| **Natural Frequency (rad/s)** | — | — | 4.88 |
| **MSV (%)** | 100.0 | 66.0 | 56.5 |
| **MCI (%)** | 9.0 | 55.2 | 100 |
| **MOI (%)** | 86.6 | 100.0 | 95.4 |

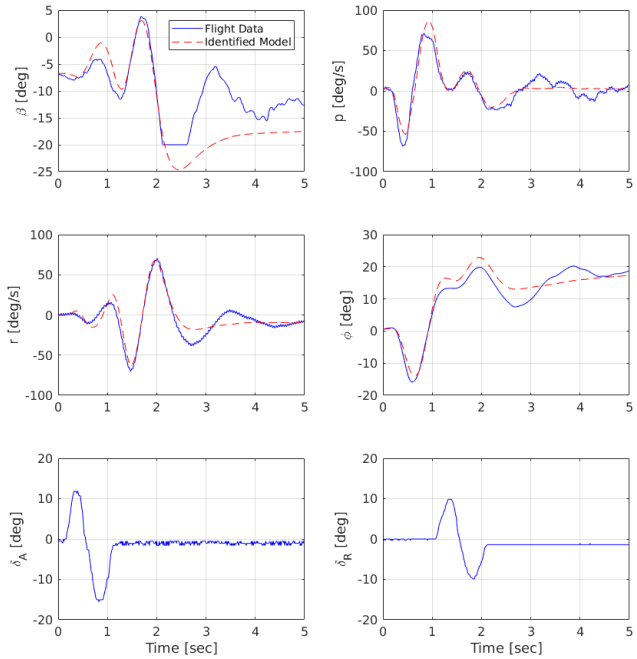


Fig. 8: Comparison between flight data and identified lateral/directional model.

To verify the integrity of the identified model the identified model is simulated with different another sets of inputs. These simulation results are displayed in Figure 9. The identified

model shows good consistency with the measured state measurements, and it can be concluded that the identified model is reasonably close to the true vehicle model. The same method

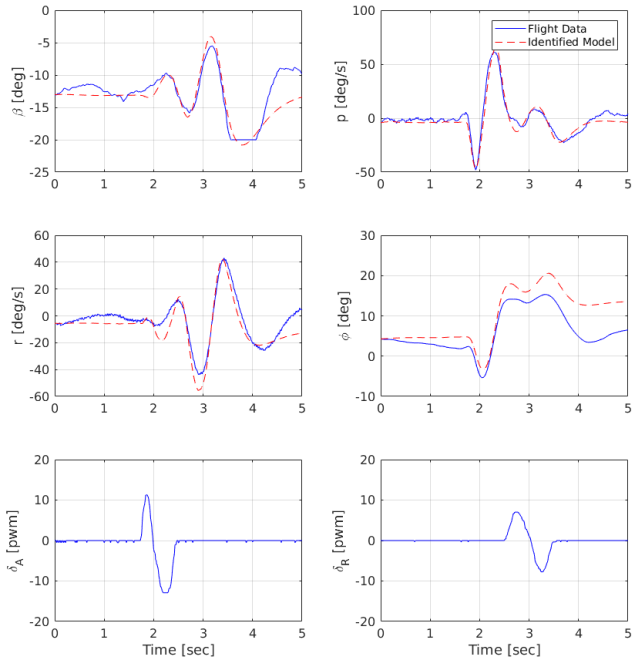


Fig. 9: Model verification with alternate excitation sets.

is implemented on the longitudinal axis using doublet and sine sweep excitations with good results. Figure 10 shows the identified longitudinal model in the dashed red line and flight test data in the solid blue. For SUAS with fairly rigid structures the sine sweep excitation can also be successfully used for system identification using data acquired by the instrumentation system. Table V shows the identified modes and characteristics using OKID.

TABLE V: Super Cub longitudinal dynamic modes.

| **Mode** | **Phugoid** | **Short Period** |
|---|---|---|
| **Eigenvalue** | $-4.6763 \pm \text{j}4.4511$ | $-0.1328 \pm \text{j}0.4627$ |
| **Damping Ratio** | 0.2758 | 0.7243 |
| **Natural Freq. (rad/s)** | 0.4838 | 6.4591 |
| **MSV (%)** | 100.0 | 19.0 |
| **MCI (%)** | 100.0 | 84.0 |
| **MOI (%)** | 67.8 | 100.0 |

## VI. Conclusions

This paper presents the design of a flight test instrumentation system for accurately measuring aircraft state and control time histories to support parameter and system identification. The system features a modular design that supports a range of sensors including air data and inertial navigation systems, and can log all required data at 100 Hz. The Observer/Kalman Identification (OKID) algorithm is applied to flight test data obtained using the new flight test instrumentation system to generate linear state-space models. Results presented in the paper demonstrate that the system produces identified

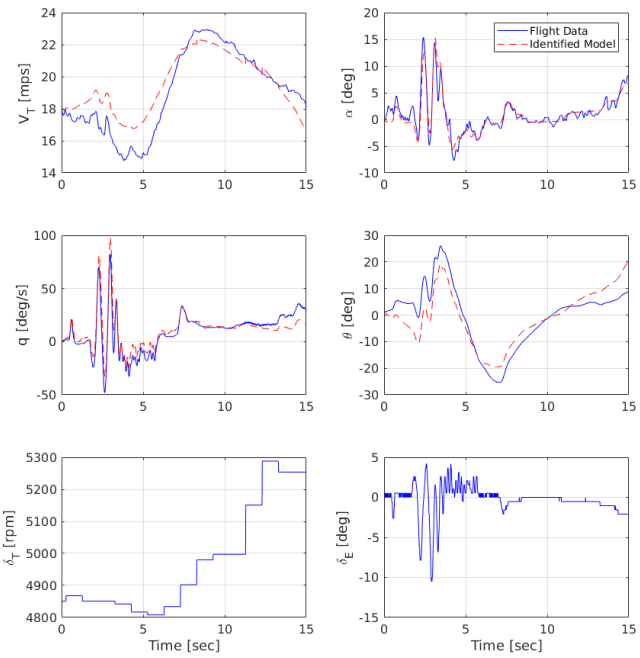


Fig. 10: Longitudinal identification with frequency sweep excitation.

rigid-body linear state-space models of fixed wing Unmanned Air Systems that match recorded flight data reasonably well. Additionally, the identified models are able to match observed flight data when excited with input sets that are different than those used for the original flight testing.

## Appendix
## Identified Models

The identified lateral/directional and longitudinal model for the Hangar-9 1/4 Scale PA-18 Super Cub is Eq. (A.1) and Eq. (A.2). Angular states are in radians, angular rates in radians per second, velocity is in meters per second, and controls in degrees. The lateral/directional model is trimmed at $\beta_1 = -6.75°$, $p_1 = 2.2\,°/\text{s}$, $r_1 = -0.62\,°/\text{s}$, and $\phi_1 = -0.59°$, where a '1' subscript indicates a trim value.

$$\begin{Bmatrix} \dot{\beta} \\ \dot{p} \\ \dot{r} \\ \dot{\phi} \end{Bmatrix} = \begin{bmatrix} 0.07918 & -0.1425 & -0.8387 & -0.414 \\ 4.81 & -7.098 & -3.568 & -2.693 \\ 3.444 & 4.548 & -1.98 & -0.8893 \\ -0.04679 & 0.9998 & -0.03553 & -0.02902 \end{bmatrix} \begin{Bmatrix} \beta \\ p \\ r \\ \phi \end{Bmatrix} + \begin{bmatrix} -0.002815 & 0.01296 \\ -0.666 & -0.2216 \\ 0.2464 & -0.5871 \\ -0.01386 & -0.005222 \end{bmatrix} \begin{Bmatrix} \delta_\text{A} \\ \delta_\text{R} \end{Bmatrix} \tag{A.1}$$

Equation (A.2) is trimmed at $V_{\text{T}_1} = 17.96\,\text{m/s}$, $\alpha_1 = 0.09°$, $q_1 = -0.74\,°/\text{s}$, and $\theta_1 = 1.31°$. Note that in Eq. (A.2) the true airspeed $V_\text{T}$ is substituted for body-axis $x$ velocity $u$ in

Eq. (7).

$$\begin{Bmatrix} \dot{V}_{\mathrm{T}} \\ \dot{\alpha} \\ \dot{q} \\ \dot{\theta} \end{Bmatrix} = \begin{bmatrix} -0.4541 & -2.628 & 1.806 & -7.129 \\ -0.0851 & -2.468 & 1.788 & 0.1256 \\ 0.2701 & -5.163 & -7.527 & 1.255 \\ 0 & -0.2657 & 0.9126 & 0.3046 \end{bmatrix} \begin{Bmatrix} V_{\mathrm{T}} \\ \alpha \\ q \\ \theta \end{Bmatrix} + \begin{bmatrix} 0 & 0.02639 \\ 0 & 0.05085 \\ -0.0001826 & -1.417 \\ 0 & -0.01782 \end{bmatrix} \begin{Bmatrix} \delta_{\mathrm{T}} \\ \delta_{\mathrm{E}} \end{Bmatrix} \tag{A.2}$$

The aircraft mass is $9.3\,\mathrm{kg}$, and the C.G. location is approximately $0.5\,\mathrm{m}$ measured from the tip of the propeller shaft.

## REFERENCES


[1] "Roll, Pitch and Yaw Controller Tuning," ArduPilot Dev Team, 2016, accessed 24 February 2017. [Online]. Available: http://ardupilot.org/plane/docs/roll-pitch-controller-tuning.html

[2] L. Meier *et al.*, "PX4/Firmware: 1.6.0 Release RC1," Jan. 2017. [Online]. Available: https://doi.org/10.5281/zenodo.263784

[3] J. Roskam, *Airplane Flight Dynamics & Automatic Flight Controls: Part I*. DARCorporation, Jan. 2001.

[4] R. D. Finck *et al.*, "USAF Stability and Control DATCOM," Flight Dynamics Laboratory, Wright Aeronautical Laboratories, Wright-Patterson AFB, OH, Tech. Rep. AFWAL-TR-83-3048, Apr. 1978. [Online]. Available: http://www.dtic.mil/docs/citations/ADB072483

[5] A. E. Albright, C. J. Dixon, and M. C. Hegedus, "Modification and Validation of Conceptual Design Aerodynamic Prediction Method HASC95 With VTXCHN," NASA Langley Research Center, Tech. Rep. NASA Contractor Report 4712, Mar. 1996. [Online]. Available: https://ntrs.nasa.gov/search.jsp?R=19960022946

[6] J. Player and D. R. Gingras, "Rapid Simulation Development for Evaluation of Conceptual Unmanned Aerial Vehicles," in *AIAA Modeling and Simulation Technologies Conference and Exhibit*. Providence, RI: AIAA, Aug. 2004, doi:10.2514/6.2004-5042.

[7] K. W. Iliff and R. E. Maine, "Practical Aspects of Using a Maximum Likelihood Estimation Method to Extract Stability and Control Derivatives from Flight Data," NASA Dryden Flight Research Center, NASA TN D-8209, Apr. 1976.

[8] K. W. Iliff, "Parameter estimation for flight vehicles," *Journal of Guidance, Control, and Dynamics*, vol. 12, no. 5, pp. 609–622, 1989.

[9] E. A. Morelli, "System Identification Program for AirCraft (SIDPAC)," in *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Monterrey, CA, Aug. 2002, doi:10.2514/6.2002-4704.

[10] D. J. Linse and R. F. Stengel, "Identification of Aerodynamic Coefficients Using Computational Neural Networks," *Journal of Guidance, Control, and Dynamics*, vol. 16, no. 6, pp. 1018–1025, 1993. [Online]. Available: http://dx.doi.org/10.2514/3.21122

[11] V. Puttige and S. Anavatti, "Real-Time Neural Network Based Online Identification Technique for a UAV Platform," in *2006 International Conference on Computation Intelligence for Modelling Control and Automation and International Conference on Intelligent Agents Web Technologies and International Commerce (CIMCA'06)*, Nov. 2006, pp. 92–92. [Online]. Available: http://dx.doi.org/10.1109/CIMCA.2006.170

[12] J.-N. Juang and R. S. Pappa, "An eigensystem realization algorithm for modal parameter identification and modal reduction," *Journal of Guidance, Control, and Dynamics*, vol. 8, no. 5, pp. 620–627, 1985. [Online]. Available: http://dx.doi.org/10.2514/3.20031

[13] J.-N. Juang, M. Phan, L. G. Horta, and R. W. Longman, "Identification of observer/Kalman filter Markov parameters," *Journal of Guidance, Control, and Dynamics*, vol. 16, no. 2, pp. 320–329, 1993. [Online]. Available: http://dx.doi.org/10.2514/3.21006

[14] M. B. Tischler and M. G. Cauffman, "Frequency-Response Method for Rotorcraft System Identification: Flight Applications to BO-105 Coupled Rotor/Fuselage Dynamics," *Journal of the American Helicopter Society*, vol. 37, no. 3, pp. 3–17, 1992.

[15] J.-N. Juang, *Applied System Identification*. Upper Saddle River, NJ: Prentice Hall, 1994.

[16] J.-N. Juang and M. Phan, "Identification of System, Observer, and Controller from Closed-loop Experimental Data," *Journal of Guidance, Control, and Dynamics*, vol. 17, no. 1, pp. 91–96, 1994. [Online]. Available: http://dx.doi.org/10.2514/3.21163

[17] P. G. Hamel and R. V. Jategaonkar, "Evolution of Flight Vehicle System Identification," *Journal of Aircraft*, vol. 33, no. 1, pp. 9–28, 1996. [Online]. Available: http://dx.doi.org/10.2514/3.46898

[18] T. Woodbury, J. Valasek, and F. Arthurs, "Flight test results of Observer/Kalman Filter Identification of the Pegasus unmanned vehicle," ser. AIAA Paper 2015-1481. Kissimmee, FL: AIAA Atmospheric Flight Mechanics Conference, Jan. 2015.

[19] F. Arthurs, J. Valasek, and M. D. Zeigler, "Precision Onboard Small Sensor System for Unmanned Air Vehicle Testing and Control," in *AIAA Guidance, Navigation, and Control Conference*, San Diego, CA, 2016, AIAA Paper 2016-1138. [Online]. Available: http://dx.doi.org/10.2514/6.2016-1138

[20] J. Harris, J. Henrickson, F. Arthurs, and J. Valasek, "Aircraft System Identification using Artificial Neural Networks with Flight Test Data," in *2016 International Conference on Unmanned Aircraft Systems (ICUAS'16)*. Arlington, VA: IEEE, Jun. 2016, doi:10.1109/ICUAS.2016.7502624.

[21] D. T. McRuer, D. Graham, and I. Ashkenas, *Aircraft Dynamics and Automatic Control*, ser. Princeton Legacy Library. Princeton, NJ: Princeton University Press, 2014.

[22] D. K. Schmidt, *Modern Flight Dynamics*, 1st ed. New York, NY: McGraw-Hill, 2010.

[23] V. Klein and E. A. Morelli, *Aircraft System Identification: Theory and Practice*, 1st ed., ser. AIAA Education Series. Reston, VA: AIAA, 2006, pp. 306–322.

[24] B. L. Stevens, F. L. Lewis, and E. N. Johnson, *Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems*, 3rd ed. Hoboken, NJ: John Wiley & Sons, Inc., 2016.

[25] D. F. Enns *et al.*, "Application of Multivariable Control Theory to Aircraft Control Laws," Flight Dynamics Directorate, USAF Wright Laboratory, Wright-Patterson AFB, OH, Tech. Rep. WL-TR-96-3099, May 1996. [Online]. Available: http://www.dtic.mil/docs/citations/ADA315259

[26] W. Durham, K. A. Bordignon, and R. Beck, *Aircraft Control Allocation*. Chichester, UK: John Wiley & Sons, Ltd., 2017.

[27] J. Valasek and W. Chen, "Observer/Kalman Filter Identification for Online System Identification of Aircraft," *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 2, pp. 347–353, 2003. [Online]. Available: http://dx.doi.org/10.2514/2.5052

[28] "BeagleBone Black," BeagleBoard.org Foundation, 2016, accessed 24 February 2017. [Online]. Available: http://beagleboard.org/black

[29] J. Harris, V. G. Goecks, H.-H. Lu, and J. Valasek, "VSCL Developmental Flight Test Instrumentation," May 2017. [Online]. Available: https://doi.org/10.5281/zenodo.572272

[30] *Aeroprobe Micro Air Data System V2.0 User Manual*, Aeroprobe Corporation, Revision A, Jan 2015.

[31] *VN-200 Product Brief*, VectorNav Technologies, 12-0003-R4.

[32] G. D. Hanson and R. F. Stengel, "Effects of Displacement and Rate Saturation on the Control of Statically Unstable Aircraft," *Journal of Guidance, Control, and Dynamics*, vol. 7, no. 2, pp. 197–205, Mar.–Apr. 1992, doi:10.2514/3.8567.

[33] R. Longman, M. Bergmann, and J.-N. Juang, "Variance and bias confidence criteria for ERA modal parameter identification," in *Astrodynamics Conference*, 1988, p. 4312.