



HAL
open science

Pitch-Axis Identification for a Guided Projectile Using a Wind-Tunnel-Based Experimental Setup

Guillaume Strub, Simona Dobre, Vincent Gassmann, Spilios Theodoulis,
Michel Basset

► **To cite this version:**

Guillaume Strub, Simona Dobre, Vincent Gassmann, Spilios Theodoulis, Michel Basset. Pitch-Axis Identification for a Guided Projectile Using a Wind-Tunnel-Based Experimental Setup. IEEE/ASME Transactions on Mechatronics, 2016, 21 (3), pp.1357-1365. 10.1109/TMECH.2016.2525719 . hal-01510186

HAL Id: hal-01510186

<https://hal.science/hal-01510186>

Submitted on 19 Apr 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Pitch Axis Identification for a Guided Projectile using a Wind Tunnel-based Experimental Setup

Guillaume Strub^{1,2}, Simona Dobre¹, Vincent Gassmann¹, Spilios Theodoulis¹, Michel Basset²

Abstract—This article details the identification of a pitch axis model for an 80mm fin-stabilized, canard-guided projectile through a Hardware-in-the-Loop experimental setup. This setup is based on an autonomous functional projectile prototype installed in a subsonic wind tunnel by the means of a 3 Degrees of Freedom (DoF) gimbal mount. A nonlinear dynamical model is first derived from flight mechanics principles, then a linearized model is obtained through Taylor series expansion. The *a priori* and *a posteriori* identifiability of the proposed linear model are assessed and the associated experimental input signals are accordingly designed. The model parameters are then estimated using a numerical optimization procedure and the associated uncertainty is obtained through a bootstrapping method. The results and their implication on the projectile flight control design are finally discussed.

Index Terms—Guided Projectiles, Flight Mechanics, Systems Identification, Uncertain Systems

I. INTRODUCTION

THE interest in guided projectile concepts has grown steadily over the last decades. Indeed, they can vastly improve precision and range over traditional ballistic ammunition [1], [2]. The latter typically exhibit poor performance due to their sensitivity to initial launch conditions and their inability to reject in-flight disturbances. However, they remain significantly less expensive than missile systems, which employ high-performance actuators and sensors as well as sophisticated guidance, navigation and control (GNC) algorithms. The key idea in guided projectile design is to develop missile-like GNC functionality using low-cost, g-hardened actuators, sensors and embedded processors.

The projectile flight control laws are at the core of the GNC algorithms and deal with platform stabilization, disturbance rejection and reference tracking. The design of these control laws is mostly done through model-based control techniques [3], [4], hence the need for an accurate projectile dynamic model. Projectiles and missiles obey the principles of flight mechanics, which describe the rigid body dynamic and kinematic behavior [5]. This behavior is characterized by the aerodynamic forces and moments, which cannot be analytically computed. Several methods are able to quantify the aerodynamic coefficients of projectiles. In that respect, basic

information is generally gathered through Computational Fluid Dynamics (CFD) analysis [6], empirical or semi-empirical codes like PRODAS (PROjectile Design/Analysis System) [7] or Missile DATCOM [8], wind tunnel tests [9] and/or free-flight experiments [10].

In this work we present an additional technique based on a functional autonomous projectile prototype installed in a wind tunnel, by means of a support allowing rotation around all axes (Roll, Pitch and Yaw). The model parameters, describing the behavior of the projectile in flight, can be estimated by the means of system identification and wind-tunnel experimental data. As the input signals can be fully specified, this technique leads to better excitation and conditioning than free-flight identification, in which the input can be difficult to modify in flight. The proposed setup is also applicable to control studies and especially controller validation through real-time testing.

The proposed approach is akin to Hardware-In-the-Loop simulation, which is used in various fields such as robotics, automotive or aerospace design and consists in including hardware components in a simulation loop [11]. One typical HIL use case is the simulation of a part of a system which may not be available or can be difficult to implement in an experimental setup. For example in Verma *et al.* [12], the dynamics of a high-mobility multipurpose wheeled vehicle are simulated using a scaled-down model in a HIL simulation, where the ultimate goal is to build a collision avoidance algorithm testbed. In our case, the projectile free-flight behavior is difficult to exploit, therefore it is emulated using the controlled environment provided by the wind tunnel and support structure.

To the authors' best knowledge, there are few similar setups for identification and control investigation on guided projectiles. In Hann *et al.* the roll dynamics of a sounding rocket in a vertical wind tunnel are analyzed through a minimal modeling approach and integral-based parameter identification. Fresconi *et al.* [13] propose a projectile prototype using a similar low-cost maneuver system, where a roll controller using linear quadratic optimal control and a PRODAS-derived model is designed and assessed in terms of performance. In both cases, only the roll axis is considered, whereas the proposed setup enables simultaneous rotation on the pitch, roll and yaw axes.

The present article focuses on the identification of a control-oriented pitch axis model. Using flight mechanics principles, a nonlinear dynamic model is constructed and linearized for operation around equilibrium points. A complete identification study is then conducted. First, the identifiability of the proposed linear model parameters is assessed. An optimal input signal is then built using this knowledge, and the projectile is

*This work is in part financed by the French Defense Procurement Agency (DGA, *Direction Générale de l'Armement*)

¹G. Strub, S. Dobre, V. Gassmann and S. Theodoulis are with the French-German research Institute of Saint-Louis (ISL), 5 rue du Général Cassagnou, 68301 Saint-Louis, France. E-mail: guillaume.strub@isl.eu

²G. Strub and M. Basset are with the Modeling, Intelligence, Process and Systems (MIPS) Laboratory EA2332, 12 rue des Frères Lumière, 68093 Mulhouse, France

excited around several equilibrium points. Finally, the linear model parameters and associated uncertainties are estimated from the collected experimental data.

This paper is structured as follows: Section II describes the ACHILES experimental setup. In Section III, the projectile pitch axis model is derived from flight mechanics principles. Section IV deals with the identifiability and estimation of the proposed model. The results and their implication on flight control design are discussed in Section V.

II. EXPERIMENTAL SETUP

Our implementation of the proposed experimental setup, named ACHILES (*Automatic Control Hardware-In-the-Loop Experimental Setup*), is based on an 80mm caliber projectile with four front-located guidance canards and four aft-located stabilization fins, illustrated in Fig. 1. This configuration offers several advantages for identification and control studies: the airframe stability only depends on the wings geometry and the guidance fins offer good control authority. The projectile is installed in a continuous subsonic wind tunnel using a 3-DoF gimbal-like structure, enabling rotation along the roll, pitch, and yaw axes while restraining all linear motion.

An overview of the ACHILES architecture is presented in Fig. 2. It includes the hardware components such as actuators, inertial measurement unit and embedded computer as well as the software components, namely the development environment and the target application.

A. Projectile Hardware Description

The projectile prototype was designed as a fully autonomous system, which is a common approach in the field of mobile robotics (e.g. terrestrial robots [14], surface vehicles [15] or aerial robots [16]), where it may be complex to offload computation on a separate computer. Indeed, physical tethers may be impractical or unfeasible and wireless links introduce latency, which has negative impact on control loop performance and stability. Therefore, all algorithm computation is processed on an embedded computer, whereas an external computer is added

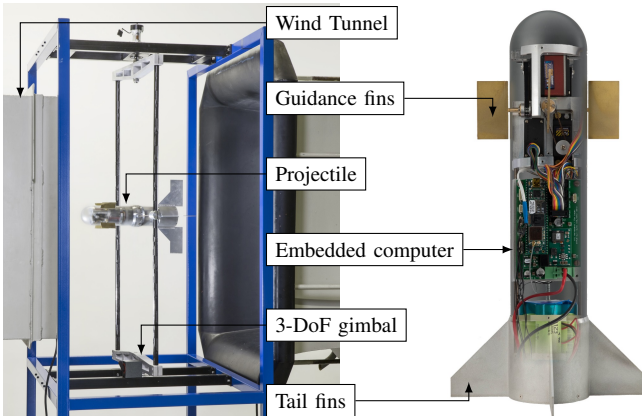


Fig. 1. Main components of the ACHILES experimental setup.

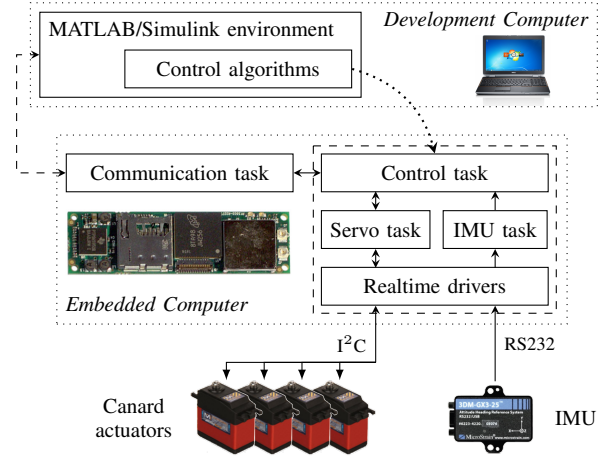


Fig. 2. Overview of the ACHILES hardware and software architecture.

for supervision and datalogging. The main hardware components (Actuators, Internal Measurement Unit, Embedded Computer) are detailed in the following paragraphs.

1) *Actuators*: These components are responsible for driving the guidance fins in accordance to the control signals issued by the GNC algorithms. Since they are included in the control loop, their dynamic behavior has an impact on the overall stability and performance of the closed-loop system. Detailed knowledge of the actuator behavior is therefore necessary in order to design projectile control laws, and it is desirable that they operate linearly and are saturation-free for a certain actuation range. The base platform for the actuators are off-the-shelf R/C servomotors, which are widely available, compact and of a low cost. However, the included control electronics usually do not fulfill the above requirements and are therefore replaced with a custom control board based on a dsPIC microcontroller, which allows custom control law implementation. The Hobbyking HK47010DMG metal-gear servo was used for this modification, as it includes an absolute magnetic position sensor, which is more precise and durable than a potentiometer. The servo controller is implemented as a cascade of a proportional position controller and a proportional-integral velocity controller, which have been tuned using nonsmooth \mathcal{H}_∞ synthesis [17] for a response time of about 100 ms with no overshoot. The normalized step response of the actuators, for different step amplitudes, is presented in Fig. 3a and the associated control signals are illustrated in Fig. 3b. One can note that the response is linear up to a 60° input step, and values above this threshold drive the actuator into saturation.

2) *Inertial Measurement Unit*: A Microstrain 3DM-GX3 provides the GNC algorithms with an estimate of the projectile's angular attitude and velocities. This device contains three accelerometers, three gyrometers and three magnetometers as well as an embedded inertial data fusion algorithm. The IMU communicates with the embedded computer through an RS232 link at 921600 baud using a message-based protocol. In normal operation, the IMU periodically sends an update message containing the latest inertial estimates at a rate of 100 Hz,

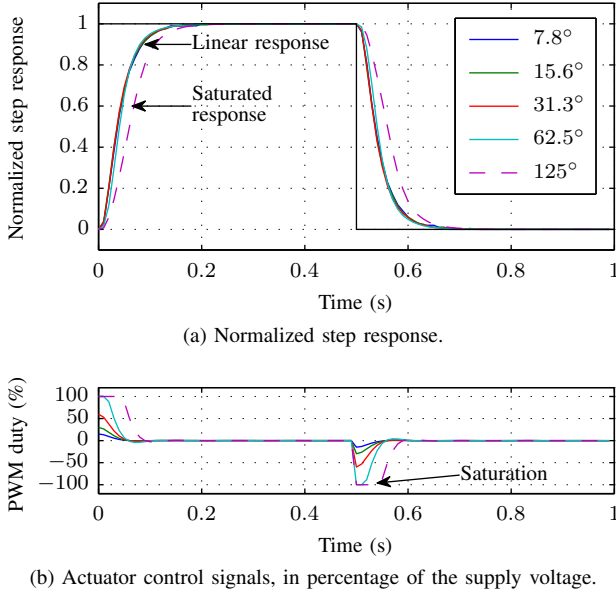


Fig. 3. Normalized actuator step response and corresponding control signals, for a set of reference angles.

and serves as the main timebase for the embedded algorithms.

3) *Embedded Computer*: The GNC algorithm computation task and the communication task with the development computer are performed on an onboard Gumstix Overo FireSTORM computer on-module. This module is based on a Texas Instruments DM3730 Cortex-A8 ARM processor running at 700 MHz with 512 MB RAM, a microSD card for program and data storage, an embedded WiFi chipset and provides I²C, UART and SPI extension busses. A Pinto-TH extension board interfaces the Gumstix with the projectile's main electronic board, which provides the level-conversion and power supplies for interfacing with the actuators and sensors.

B. Software Environment

As ACHILES is dedicated to research investigations on identification and control design, the key design objective was to build a complete rapid prototyping environment that allows to focus on the algorithm design and testing tasks. These tasks are performed using the Simulink environment and then the developed algorithms are deployed on the projectile prototype using a wireless network link. The software environment is divided in two parts: a development environment based on MATLAB/Simulink, which runs on a standard desktop PC, and the target software, which runs on the embedded computer.

1) *Development Environment*: Algorithm prototyping for ACHILES is done using the MATLAB/Simulink environment, especially the automatic C code generation functionality. A custom Code Generation target has been developed, providing high-level Simulink blocks for interacting with the projectile sensors and actuators. An exemple use case is presented in Fig. 4.

The code generation process consists in converting the Simulink model in C code that will be referenced by the target application. The model and the application are then cross-compiled for the ARM target, and the compiled application

is uploaded to the embedded computer. A HIL simulation can then be run, using the existing monitoring tools provided by Simulink in order to visualize the system output signals (angular attitude and velocities) as well as internal variables.

2) *Embedded Target Software*: Implementing a HIL simulation or a control law on real hardware implies that the algorithms will be executed in real-time. Indeed, the controller hardware and software must be able to guarantee a response before a fixed time deadline, usually a multiple of the sampling period. A real-time operating system is therefore necessary, as traditional desktop operating systems are unsuitable for these tasks. A Linux-based solution was chosen to reduce development time, as it includes built-in support for the Gumstix processor and peripherals as well as multiple communication protocols. Since the *vanilla* Linux kernel is not real-time capable, the Xenomai patch [18] was applied to provide soft- and hard-real-time capabilities. It consists in a high-priority real-time microkernel which coexists with the Linux kernel thanks to a low-level abstraction layer.

The target application serves as an execution framework for the Simulink-generated model code and provides services for interfacing with actuators and sensors and communication with the development computer. The high-level interface with the actuators and sensors is implemented respectively in the Servo and IMU real-time tasks. In order to access the low-level serial communication busses (I²C, UART), specific real-time drivers had to be written as those provided by Linux are not suitable for this use. The Simulink-generated model code is periodically called by the Control real-time task, which is synchronized with the IMU task so that the 100 Hz execution pace is provided by the Inertial Measurement Unit. Finally, the communication channel with the Simulink environment is built atop the wireless network connection. This task runs in the Linux domain as it is not time-critical, so the existing network drivers are used.

III. PITCH AXIS MODELING

The dynamic behavior of a flying vehicle is governed by the 6-DoF equations of motion, which express the rigid body translational and rotational kinematics and dynamics [5]. In the

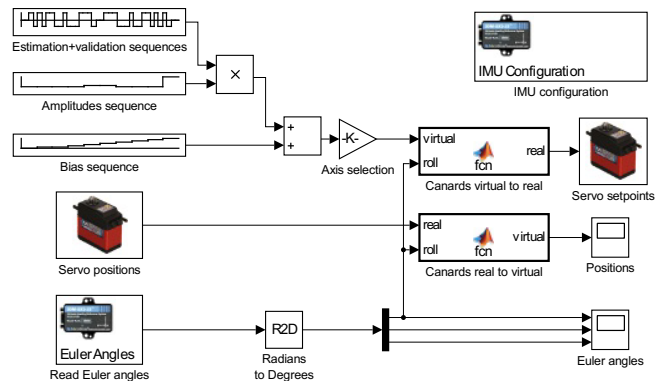


Fig. 4. Simulink diagram built with custom blocks for interfacing ACHILES' actuators and sensors and used for data collection in system identification experiments

ACHILES setup, the projectile is held at its center of gravity by the 3-DoF support structure, thereby preventing all linear motion. This paper focuses on the pitch axis, and it is assumed that no motion will occur on the roll and yaw axes, which can be physically locked.

Figure 5 represents the projectile pitching motion and the associated parameters, which are the angle of attack α , the body pitch rate q and the fin deflection angle δ_m . The angle of attack is defined as the angle between the wind direction 1^W and the projectile longitudinal axis 1^B , which is also equal to the pitch angle θ in the proposed configuration. The fin deflection angle is relative to the body, with the sign taken so that a positive increment in δ_m induces a positive pitching control moment on the projectile body. This control signal is equally distributed over the four canards $\delta_1 \cdots \delta_4$ such that they act as a single virtual pitch canard plane.

A. Nonlinear Model Derivation

The projectile pitch attitude equations are expressed using the flight mechanics framework [5]. Knowing that herein the objective is to focus on the pitch axis identification, it will be assumed in the following part that the roll and yaw axes are locked. Thus, the only motion left is about the 2^B axis and the pitch dynamic and kinematic differential equations correspond to the application of Euler's law to a planar body rotating about its center of gravity:

$$\begin{aligned} \dot{q} &= I_2^{-1} \bar{q} S d C_m(\alpha, \mathcal{M}) \\ \dot{\alpha} &= q \end{aligned} \quad (1)$$

where I_2 is the projectile inertia along the pitch axis 2^B and $\bar{q} S d C_m(\alpha, \mathcal{M})$ corresponds to the pitching moment, where C_m is the total pitch moment aerodynamic coefficient. Other quantities are the projectile reference area S , the caliber d and the dynamic pressure $\bar{q} = \frac{1}{2} \rho(h) V^2$, where $\rho(h)$ is the altitude-dependent air density. Here, h is constant (ground level) and V is the airspeed in the wind tunnel. The total pitch aerodynamic coefficient can be decomposed as [5]:

$$C_m(\alpha, \mathcal{M}) = C_{m\alpha}(\alpha, \mathcal{M})\alpha + \left(\frac{d}{2V}\right) C_{mq}(\mathcal{M})q + C_{m\delta}(\alpha, \mathcal{M})\delta_m \quad (2)$$

where the aerodynamic derivatives $C_{m\alpha}$, C_{mq} and $C_{m\delta}$ are unknown nonlinear functions of the Mach number \mathcal{M} and the angle of attack α and correspond to the partial derivatives of C_m with respect to α , q and δ_m . The Mach number is defined as $\mathcal{M} = \frac{V}{a(h)}$ where $a(h)$ is the speed of sound at altitude h . At sea level, $a = 343$ m/s for a temperature of 20°C and a pressure of 1.013 bar.

B. Linearization and Trimming

Equations (1) and (2) form a nonlinear, parameter-dependent (NLPD) state-space model of the form:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{f}_{\mathbf{x}}[\mathbf{x}(t), u(t), \boldsymbol{\sigma}(t)] \\ \mathbf{y}(t) &= \mathbf{f}_{\mathbf{y}}[\mathbf{x}(t), u(t), \boldsymbol{\sigma}(t)] \end{aligned} \quad (3)$$

with $\mathbf{x} = [\alpha \ q]^T$ being the state vector, $u = \delta_m$ the control input and $\boldsymbol{\sigma} = [V \ h]^T$ an external parameter vector. The measured output y of this system is the angle of attack α .

This system can be linearized using a *first-order* Taylor series expansion [19] around any equilibrium flight condition $(\bar{\mathbf{x}}, \bar{u}, \bar{\boldsymbol{\sigma}})$ called trim point [20]. At the trim condition, the state derivative is by definition:

$$\dot{\bar{\mathbf{x}}} \triangleq 0 = \mathbf{f}_{\mathbf{x}}(\bar{\mathbf{x}}, \bar{u}, \bar{\boldsymbol{\sigma}}) \quad (4)$$

The projectile's equilibrium manifold, or *trim map*, is determined experimentally by measuring the steady-state angle of attack for different fin deflection angles for fixed airspeed. The obtained trim map is represented in Fig. 6 and exhibits a linear flight domain for $\bar{\delta}_m \in [-12^\circ, +12^\circ]$. Outside of this range, the influence of the canards becomes negligible and the projectile starts to have a stalling behavior.

When the state equations (3) are linearized around a family of equilibrium points, the result is a q -LPV model of the form:

$$\begin{aligned} \dot{\mathbf{x}}_{\delta}(t) &= \mathbf{A}(\boldsymbol{\rho})\mathbf{x}_{\delta}(t) + \mathbf{B}(\boldsymbol{\rho})u_{\delta}(t) \\ \mathbf{y}_{\delta}(t) &= \mathbf{C}(\boldsymbol{\rho})\mathbf{x}_{\delta}(t) + \mathbf{D}(\boldsymbol{\rho})u_{\delta}(t) \end{aligned} \quad (5)$$

where $\boldsymbol{\rho} = [\bar{\boldsymbol{\sigma}} \ \bar{u}]^T$ is the trim vector and $\mathbf{x}_{\delta} = \mathbf{x} - \bar{\mathbf{x}}(\boldsymbol{\rho})$, $u_{\delta} = u - \bar{u}(\boldsymbol{\rho})$, and $\mathbf{y}_{\delta} = \mathbf{y} - \bar{\mathbf{y}}(\boldsymbol{\rho})$ are deviations from equilibrium.

Elements of the \mathbf{A} , \mathbf{B} , \mathbf{C} and \mathbf{D} matrices are gradients of $\mathbf{f}_{\mathbf{x}}$ evaluated at the trim point $(\bar{V}, \bar{h}, \bar{\alpha}, \bar{\delta}_m)$:

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} 0 & 1 \\ M_{q\alpha}(\boldsymbol{\rho}) & M_{qq}(\boldsymbol{\rho}) \end{bmatrix} & \mathbf{B} &= \begin{bmatrix} 0 \\ M_{q\delta}(\boldsymbol{\rho}) \end{bmatrix} \\ \mathbf{C} &= [1 \ 0] & \mathbf{D} &= 0 \end{aligned} \quad (6)$$

where $M_{q\alpha} = (\frac{\bar{q} S d}{I_2}) C_{m\alpha}$, $M_{qq} = (\frac{\bar{q} S d}{I_2})(\frac{d}{2V}) C_{mq}$, $M_{q\delta} = (\frac{\bar{q} S d}{I_2}) C_{m\delta}$ and $C_{m\alpha}$, C_{mq} , $C_{m\delta}$ are the resulting values of the aerodynamic derivatives at trim.

IV. IDENTIFICATION PROCEDURE

In system identification, there are two inverse problems to be solved, namely the choice of a model structure and the estimation of the parameters for the chosen model. In the present case, the model structure is imposed using the previously defined *quasi*-LPV model obtained from flight mechanics principles, resulting in a grey-box model parameter estimation problem. Due to the inverse nature of this problem, it is necessary to ensure that it is well-posed (in the Hadamard sense) and well-conditioned [21]. To this end, the system identification procedure presented in [22] is followed and detailed in the following paragraphs.

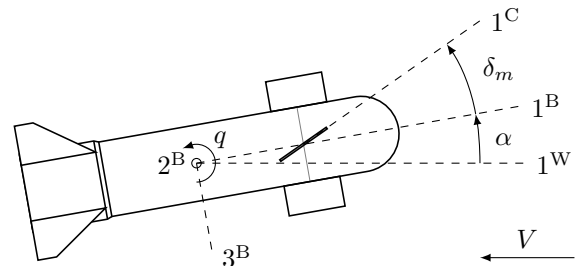


Fig. 5. Projectile side view and parameters

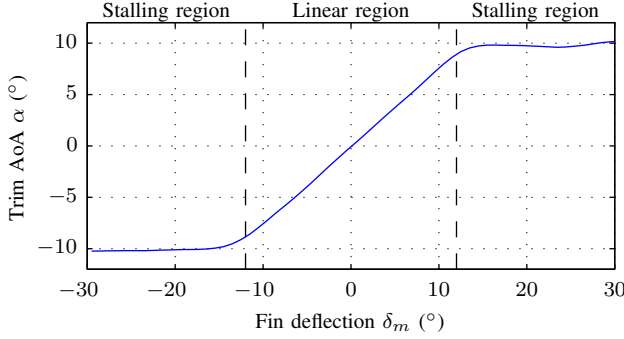


Fig. 6. Projectile pitch trim map at $V = 25$ m/s

A. Model Postulation

The q -LPV model (5-6) describes the ideal, noise-free pitch behavior of the projectile on the complete flight envelope. The system behavior in the neighborhood of a trim point is characterized by this structure with a constant trim vector ρ , and the actual measurements are subject to noise, and high-order dynamics may have been neglected. To take these effects into account, the model structure (5-6) is supplemented with an error signal $e(t)$ and an associated 2×1 state error matrix $\mathbf{K} = [k_1 \ k_2]^\top$, forming the following model structure [23]¹:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}(\boldsymbol{\theta})\mathbf{x}(t) + \mathbf{B}(\boldsymbol{\theta})u(t) + \mathbf{K}(\boldsymbol{\theta})e(t) \\ y(t) &= \mathbf{C}(\boldsymbol{\theta})\mathbf{x}(t) + \mathbf{D}(\boldsymbol{\theta})u(t) + e(t) \end{aligned} \quad (7)$$

where the state vector \mathbf{x} , input u and output y correspond to the same variables as in (5). In this structure, matrices \mathbf{A} , \mathbf{B} , \mathbf{C} and \mathbf{D} , which were previously defined in Eq. (6), are functions of the parameter vector $\boldsymbol{\theta} = [M_{a\alpha} \ M_{qq} \ M_{q\delta} \ k_1 \ k_2]^\top$ to be estimated in the sequel.

The discrete form of this model structure is called a *directly parametrized innovations form* [23], where $e(t)$ is the prediction error. This form naturally leads to the estimation of a *predictor* model, where the input and previous outputs are used to estimate the future outputs.

In the present case, the estimated model will be used for control law synthesis, therefore the model will *simulate* the projectile dynamics, without using the actual outputs.

This difference is of great importance when considering the identification procedure. Indeed, in the *predictor* focus, all parameters have to be estimated simultaneously, whereas in the *simulation* focus, the error model is estimated separately.

B. A priori Identifiability

The *a priori* or theoretical identifiability property of a model structure indicates whether the input-output behavior is described by a unique or finite set of parameters. It is evaluated in an idealized framework, where 1) the process and the model have an identical structure, 2) the data are noise-free and 3) the input u and measurement times can be chosen at will [24]. Let $\boldsymbol{\theta}$ denote the parameters of the model and $\boldsymbol{\theta}^*$ the parameters of the process. The parameter θ_i will be *globally identifiable*

¹For ease of notation the subscript “ δ ” is dropped

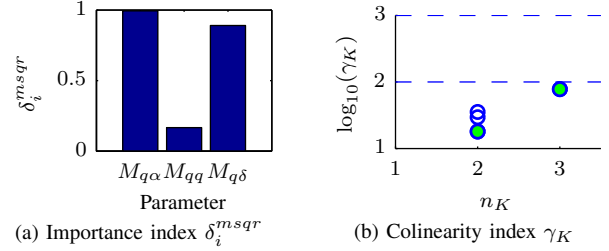


Fig. 7. Assessment of parameter identifiability using (a) the parameter importance index δ_i^{msqr} (b) the γ_K colinearity index of model parameter subsets, where n_K denotes the size of the subset

if for almost any admissible parameter vector $\boldsymbol{\theta}^*$:

$$\exists u \in \mathbb{R}^{\mathbb{R}^+}, y(t, \boldsymbol{\theta}, u(t)) = y(t, \boldsymbol{\theta}^*, u(t)) \ \forall t \in \mathbb{R}^+ \Rightarrow \theta_i = \theta_i^* \quad (8)$$

A similar definition exists for *local identifiability*, where a neighborhood of $\boldsymbol{\theta}^*$ is considered. In the present case, the global *a priori* identifiability of the model structure (7) is assessed using a Laplace transform approach.

C. A posteriori Identifiability

A posteriori identifiability differs from *a priori* identifiability in that the input signal and measurement times are imposed. Even if a parameter is *a priori* identifiable, it may not be identifiable in practice because of a lack of information in the experimental data. For a given model structure, input signal u and measurement times t_k , the parameter θ_i is *a posteriori* identifiable if for almost any admissible parameter vector $\boldsymbol{\theta}^*$:

$$y(t_k, \boldsymbol{\theta}, u_k) = y(t_k, \boldsymbol{\theta}^*, u_k) \ \forall k \in \llbracket 1, N \rrbracket \Rightarrow \theta_i = \theta_i^* \quad (9)$$

where N is the number of samples.

As with *a priori* identifiability, there is a *local* definition where a neighborhood of $\boldsymbol{\theta}^*$ is considered. This condition can be expressed as [22]:

$$\forall k \in \llbracket 1, N \rrbracket, \sum_{i=1}^{n_p} \underbrace{\frac{\partial y(t_k, \boldsymbol{\theta}, u_k)}{\partial \theta_i}}_{s_{y/\theta_i}(t_k, \boldsymbol{\theta}^*, u)} \Big|_{\boldsymbol{\theta}^*} \cdot d\theta_i = 0 \Rightarrow d\boldsymbol{\theta} = 0 \quad (10)$$

where n_p is the number of parameters. In the above equation, $s_{y/\theta_i}(t_k, \boldsymbol{\theta}^*, u_k)$ is the sensitivity of the output with respect to the parameter θ_i , for an input signal u and measurement time t_k . This condition can also be written as:

$$\mathbf{S}_y(\boldsymbol{\theta}^*, u) \cdot d\boldsymbol{\theta} = 0 \Rightarrow d\boldsymbol{\theta} = 0 \quad (11)$$

where $\mathbf{S}_y(\boldsymbol{\theta}^*, u)$ is the sensitivity matrix, defined as:

$$\mathbf{S}_y(\boldsymbol{\theta}^*, u) = \begin{bmatrix} s_{y/\theta_1}(t_1, \boldsymbol{\theta}^*, u_k) & \cdots & s_{y/\theta_{n_p}}(t_1, \boldsymbol{\theta}^*, u_k) \\ \vdots & \ddots & \vdots \\ s_{y/\theta_1}(t_N, \boldsymbol{\theta}^*, u_k) & \cdots & s_{y/\theta_{n_p}}(t_N, \boldsymbol{\theta}^*, u_k) \end{bmatrix} \quad (12)$$

The parameters are then locally *a posteriori* identifiable iff the columns of $\mathbf{S}_y(\boldsymbol{\theta}^*, u)$ are linearly independent. In our application, the sensitivity functions were derived from the output equation of the considered model (7), leaving out the error model as it is estimated separately. In practice, the above condition is not sufficient as the sensitivity matrix may be

linearly independent but ill-conditioned, due to the presence of noise in the measurements.

A further step consists in the selection of practically identifiable parameters, using importance and colinearity indices [25]. Initial observations showed that for a fixed airspeed, the projectile's pitching behavior does not change significantly when the trim angle of attack is varied, as long as the projectile operates in the linear trim region. This suggests that the sensitivity study can be done on a single operating point.

Parameter importance indices quantify the sensitivity of the model output to individual parameter changes. A suitable ranking criterion is the the mean-square distance δ_i^{msqr} , defined as

$$\delta_i^{msqr} = \sqrt{\frac{1}{N} \sum_{k=1}^N \hat{s}_{y/\theta_i}(t_k, \boldsymbol{\theta}^*, u_k)^2} \quad (13)$$

where $\hat{s}_{y/\theta_i}(t_k, \boldsymbol{\theta}^*, u_k) = s_{y/\theta_i}(t_k, \boldsymbol{\theta}^*, u_k)\theta_i$ is the normalized sensitivity function relative to θ_i . The relative importance of the model parameters $M_{a\alpha}$, M_{qq} , $M_{q\delta}$ is represented in Fig. 7a. The least influential parameter is M_{qq} , which defines the system damping, and is usually difficult to estimate using existing experimental or numerical techniques.

The colinearity index evaluates the degree of linear dependence among parameter subsets K , that is, if a change in a model parameter in K may be compensated by appropriate changes in the other parameter values in K . This can lead to non-uniqueness of the solution, even if the individual parameters have strong influence on the model output. The colinearity index γ_K is defined as

$$\gamma_K = \text{cond}(\mathbf{F}_K(\boldsymbol{\theta}^*, u)) \quad (14)$$

with \mathbf{F}_K being the empirical Fisher Information Matrix related to the parameter subset K , $n_K < n_p$

$$\mathbf{F}_K = S_y(\boldsymbol{\theta}_K^*, u)^\top S_y(\boldsymbol{\theta}_K^*, u) \quad (15)$$

The index γ_K is computed for each possible subset and is compared against several thresholds given in the literature [22]

$$\begin{cases} \gamma_K < 100 & \text{low colinearity} \\ 100 \leq \gamma_K < 1000 & \text{moderate colinearity} \\ 1000 \leq \gamma_K & \text{strong colinearity} \end{cases}$$

In the present case, the maximum γ_K is less than 100, as illustrated in Fig. 7b. Moreover, as the ratio of the highest to the lowest sensitivity index is less than 8, all parameters are identifiable in practice.

D. Experiment Design

The above defined sensitivity functions depend on the process parameters as well as the input signal and sampling instants. Therefore, the characteristics of the input signal have an important impact on the practical identifiability of the parameters. Several types of signal are commonly used in system identification [23], such as pseudo-random binary sequences (PRBS), multi-sines or filtered white noise.

In the present case, the system is excited around equilibrium using a PRBS, which has white-noise-like properties but is deterministic, periodic and bounded. Such a signal is obtained

using a linear feedback shift register (LFSR) with a maximum-length polynomial. The amplitude of the input signal has to be carefully selected such that the system is sufficiently excited while operating in a neighborhood of the operating point. A reasonable $\pm 1^\circ$ deviation around the trim angle of attack was selected based on the trim map, which shows linear behavior for an angle of attack $\bar{\alpha} \in [-9^\circ, +9^\circ]$. The bit period, which affects the duration of the excitation signal, is chosen so as to maximize the output sensitivity to all parameters, minimize the colinearity of these sensitivity functions, while allowing the system to reach steady-state. A 4-bit LFSR depth leads to a 16-bit long sequence with an excitation order of 15 [23], which is sufficient for the postulated second-order model. The bit duration was set to 5 s, as it is the best compromise between system excitation and experiment length, and leads to an 80 s long excitation sequence.

E. Parameter Estimation

The parameter estimation step consists in finding the parameter vector $\boldsymbol{\theta}$ that minimizes the quadratic cost function

$$J(\boldsymbol{\theta}) = \sum_{k=1}^N (y_m(t_k) - y(t_k, \boldsymbol{\theta}, u_k))^2 \quad (16)$$

where y_m is the vector of experimental outputs. This minimization problem is then solved using numerical optimization techniques, such as Gauss-Newton algorithms.

In practice, parameters of the model structure (7) are estimated in two steps: a first step estimates the parameters of \mathbf{A} and \mathbf{B} , with $\mathbf{K} = 0$, then in a second step the parameters of \mathbf{K} are estimated while \mathbf{A} and \mathbf{B} have the values estimated in the first step.

F. Uncertainty Analysis

As the experimental data are affected by noise and the sensitivity of the output wrt some parameters can be low, there is a need for establishing a degree of confidence on the estimated parameter values. The uncertainty on parameters estimates is also useful for building uncertain models that are used for robustness analysis during control law design. In the present case, a bootstrapping approach is employed [24], which consists in generating a population of fictive data sets using random permutations of the residuals vector e . A population of parameter estimates can then be computed from these data sets, and the relative uncertainty on each parameter θ_i is

$$\Delta\theta_i = \frac{\max(|\theta_i - \theta_{i \max}|, |\theta_i - \theta_{i \min}|)}{|\theta_i|} \quad (17)$$

V. EXPERIMENTAL RESULTS

This section details the parameter estimation results obtained with the ACHILES setup, for different values of the angle of attack and at a fixed airspeed of $V = 25$ m/s. The experimental data were collected according to Fig. 8, using the Simulink implementation of Fig. 4. In the estimation and validation datasets, the output sequences are built upon the measured achieved angle of attack α_m . The PRBS-based

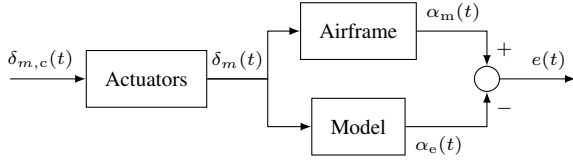
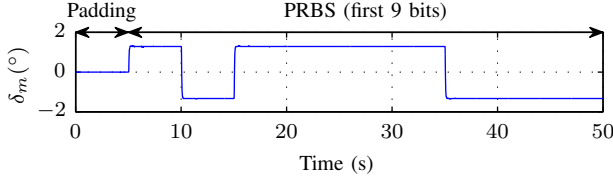
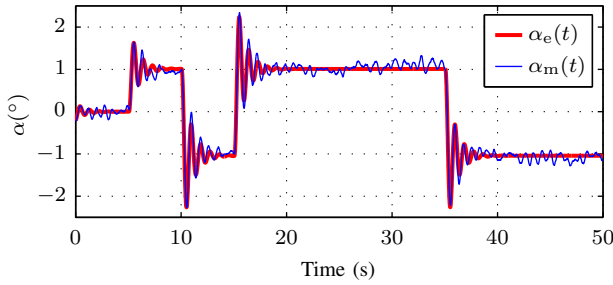


Fig. 8. Outline of the data-collecting and parameter estimation scheme



(a) Deviation from the trim fin angle $\bar{\delta}_m$ as measured on the actuators.



(b) Deviation from the trim angle of attack $\bar{\alpha}$ of the measured and estimated outputs

Fig. 9. Comparison of the model response to measured output at the trim angle of attack $\bar{\alpha} = 5^\circ$. Only part of the signals is shown for clarity.

excitation signal $\delta_{m,c}$ is applied to the control fin actuators, which have a sufficiently large bandwidth to preserve the excitation properties of the signal. The input sequences contain the measured actual fin deflections δ_m , which correspond to the actuator outputs, therefore the actuator dynamics do not need to be included in the model to be estimated. This approach is consistent with the postulated model, which does not take the actuator dynamics into account.

A. Parameter Estimates and Uncertainty

Estimation and validation data have been collected for values of the trim angle of attack $\bar{\alpha}$ from 0° to 7° in 1° steps. Figure 9 presents the collected experimental validation data (δ_m, α_m) and the model response α_e for $\bar{\alpha} = 5^\circ$. The signals are truncated to show the first 50 seconds of the experiment, that is, a 5 seconds padding period followed by the first 9 bits of the PRBS. Figure 9b shows good agreement between the validation signal α_m and the model response α_e , and suggests the projectile is underdamped due to the presence of oscillations after input steps. The difference between α_e and α_m is mostly due to unsteady aerodynamic effects.

Table I summarizes the parameter estimates, obtained with the methodology described in Section IV-E, and the associated uncertainties computed by bootstrapping, as in Section IV-F. The model fit percentage is computed as follows:

$$Fit\% = 100 \times \frac{1 - \|y_e - y_m\|_2}{\|y_e - \bar{y}_e\|_2} \quad (18)$$

where \bar{y}_e is the mean of the estimated output signal y_e .

Several observations can be made on the data in Table I. As expected from the sensitivity analysis, the uncertainty on the M_{qq} parameter is higher than the other two parameters by a factor of 2 to 6. The uncertainty on parameter values as well as the fit degrades for low angles of attack. The signal-to-noise ratio is lower at incidences near zero because of aerodynamic coupling effects between the support structure and the tail fins, amplifying the unsteady effects. Another important aspect in this data is that the parameter values do not vary significantly with the trim angle of attack $\bar{\alpha}$. This has interesting implications for control, as discussed in the next subsection.

B. Implications for Control

The moderate dependence of the parameters of the model structure (7) with respect to the trim angle of attack $\bar{\alpha}$ can be usefully exploited for control purposes. As the projectile dynamics do not vary significantly on the considered angle of attack range, a single controller may be designed for operation on the whole range. The differences between the nominal, or synthesis model, and the actual dynamics at each trim point may then be modeled as uncertainty. This representation enables the designer to check whether the synthesized control laws are robustly stable and meet the performance specification for every plant in the uncertainty set [26].

In this article, two uncertainty modeling approaches are investigated: a structured uncertain model that exploits the uncertainty on estimated parameters, and an unstructured uncertain model built around the frequency response of all estimated models.

1) *Structured Uncertainty*: in parametric or structured uncertainty, the model structure is known but some or all of the parameters are uncertain. These parameters are of the form

$$\tilde{\theta}_i = \bar{\theta}_i(1 + r_i\Delta) \quad (19)$$

where θ_i is the mean parameter value, Δ is any real scalar satisfying $|\Delta| \leq 1$ and r_i is the relative uncertainty

$$r_i = \frac{\theta_{i\max} - \theta_{i\min}}{\theta_{i\max} + \theta_{i\min}} \quad (20)$$

2) *Unstructured Uncertainty*: dynamic or unstructured uncertainty represents frequency-dependent uncertainty arising from unmodelled or neglected dynamics. There are several representations of this class of uncertainty, one of the preferred

TABLE I
ESTIMATED PARAMETERS AND ASSOCIATED UNCERTAINTIES FOR VALUES OF $\bar{\alpha}$ FROM 0° TO 7° IN 1° STEPS

$\bar{\alpha}$	$M_{q\alpha}$	$\Delta M_{q\alpha}$	M_{qq}	ΔM_{qq}	$M_{q\delta}$	$\Delta M_{q\delta}$	Fit
0°	-50.3	6%	-2.9	15%	38.8	6%	74%
1°	-53.0	9%	-2.7	20%	34.3	10%	75%
2°	-54.6	7%	-2.8	16%	36.2	7%	82%
3°	-56.1	2%	-2.6	8%	40.8	3%	86%
4°	-52.5	4%	-2.4	12%	40.5	4%	89%
5°	-51.5	5%	-2.2	15%	40.3	5%	89%
6°	-52.7	3%	-2.1	17%	41.5	4%	89%
7°	-54.4	3%	-2.0	8%	38.5	3%	86%

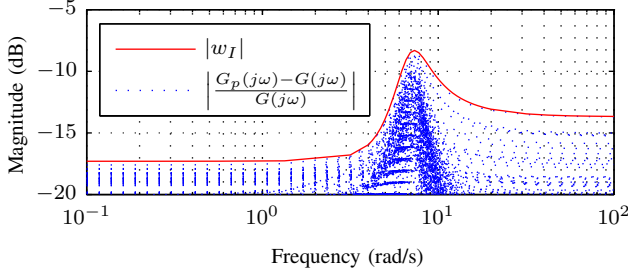


Fig. 10. Relative errors in the unstructured uncertainty case. Solid line: second-order weight $|w_I|$ in (21)

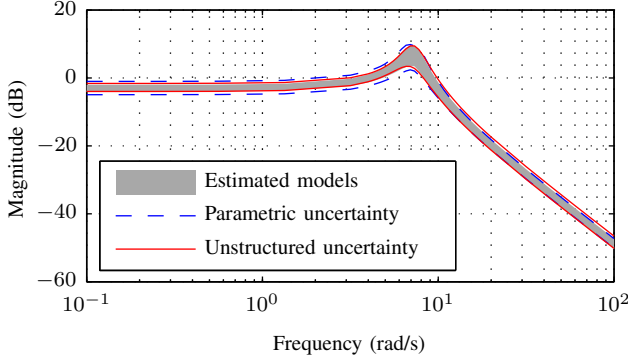


Fig. 11. Envelope of estimated and bootstrapped responses against parametric and unstructured uncertainty envelopes

form being *multiplicative uncertainty* [26], where the set of possible perturbed models Π_I contains models of the form

$$G_p(s) = G(s)(1 + w_I(s)\Delta_I(s)) \quad (21)$$

where $G(s)$ is the nominal plant model, Δ_I is any stable transfer function satisfying $|\Delta_I(j\omega)| \leq 1$, $\forall\omega$ and the multiplicative weight w_I is a stable and minimum-phase transfer function, satisfying $|w_I(j\omega)| \geq l_I(\omega)$, $\forall\omega$. The uncertainty radius $l_I(\omega)$ is defined as:

$$l_I(\omega) = \max_{G_p \in \Pi} \left| \frac{G_p(j\omega) - G(j\omega)}{G(j\omega)} \right|, \quad \forall\omega \quad (22)$$

In the present case, the parameters of the nominal model G are the mean values of the complete set of parameters and Π contains all estimated models. The relative error $|G_p - G|/|G|$ and the bounding multiplicative weight w_I , which here is a second-order filter, are presented in Fig. 10. The uncertainty size is 13.6% and 20.8% at respectively low and high frequencies, and the maximum uncertainty attains 38.5% at the resonance peak, situated in middle frequencies.

The uncertainty envelopes for the two considered modeling approaches are compared to the envelope of estimated models in Fig. 11. The parametric uncertain model is more conservative than the unstructured description with a relative uncertainty of 21.9% at low frequencies, and has a similar behavior at medium to high frequencies. For control purposes, the system has a relatively consistent behavior across the considered flight envelope, with a moderately varying damping ratio. Although the uncertainty is not negligible, it can be properly handled with robust control techniques such as \mathcal{H}_∞ control [26].

VI. CONCLUSION

In this paper, the focus was put on the identification of the pitch axis dynamics of the ACHILES setup. The projectile's behavior is expressed as a nonlinear model governed by the laws of flight dynamics. A linearized q -LPV model is obtained around the projectile equilibrium point, and considered for estimation around a fixed operating point with the addition of a noise model. *A priori* and *a posteriori* identifiability studies were conducted and show that the parameters of the proposed model are identifiable. The model parameters have been estimated for different values of the trim angle of attack at a constant airspeed $V = 25$ m/s. The results show good agreement with the validation data and moderate dependence with respect to the trim angle of attack. Uncertain models of the projectile's dynamic behavior have then been built using the estimation results.

As future work, the approaches described herein will be applied to the projectile pitch and yaw axes in coupled motion, and associated control laws will be developed. A second research axis will focus on the airspeed dependence of the developed models and the associated control issues.

REFERENCES

- [1] F. Fresconi, "Guidance and control of a projectile with reduced sensor and actuator requirements," *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 6, pp. 1757–1766, 2011.
- [2] M. Costello, "Extended range of a gun launched smart projectile using controllable canards," *Shock and Vibration*, vol. 8, no. 3, pp. 203–213, Jan. 2001.
- [3] S. Theodoulis, V. Gassmann, P. Wernert, L. Dritsas, I. Kitsios, and A. Tzes, "Guidance and control design for a class of spin-stabilized fin-controlled projectiles," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 2, 2013.
- [4] A. J. Calise, M. Sharma, and J. E. Corban, "Adaptive autopilot design for guided munitions," *Journal of Guidance, Control, and Dynamics*, vol. 23, no. 5, pp. 837–843, Sep. 2000.
- [5] P. H. Zipfel, *Modeling and Simulation of Aerospace Vehicle Dynamics*. American Institute of Aeronautics and Astronautics, 2007.
- [6] P. Weinacht, "Projectile performance, stability, and free-flight motion prediction using computational fluid dynamics," *Journal of Spacecraft and Rockets*, vol. 41, no. 2, pp. 257–263, 2004.
- [7] R. Whyte, "PRODAS - projectile analysis," Arrow Tech Associates, South Burlington, VT, Tech. Rep., 1991.
- [8] W. B. Blake, "Missile DATCOM: User's manual - 1997 FORTRAN 90 revision," DTIC Document, Tech. Rep., 1998.
- [9] J. B. Barlow, W. H. Rae, and A. Pope, *Low-Speed Wind Tunnel Testing, Third Edition*. Wiley New York, 1999.
- [10] M. Albisser, S. Dobre, C. Berner, M. Thomassin, and H. Garnier, "Identifiability investigation of the aerodynamic coefficients from free flight tests," in *AIAA Atmospheric Flight Mechanics Conference*, 2013.
- [11] M. Bacic, "On hardware-in-the-loop simulation," in *IEEE Conference on Decision and Control and European Control Conference*, 2005, pp. 3194–3198.
- [12] R. Verma, D. D. Vecchio, and H. K. Fathy, "Development of a scaled vehicle with longitudinal dynamics of an hmwv for an its testbed," *IEEE/ASME Transactions on Mechatronics*, vol. 13, no. 1, pp. 46–57, Feb. 2008.
- [13] F. Fresconi, I. Celmins, M. Ilg, and J. Maley, "Projectile roll dynamics and control with a low-cost maneuver system," *Journal of Spacecraft and Rockets*, vol. 51, no. 2, pp. 624–627, Mar. 2014.
- [14] J. Ma, H. Kharboutly, A. Benali, F. B. Amar, and M. Bouzit, "Design of omnidirectional mobile platform for balance analysis," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 6, pp. 1872–1881, Dec. 2014.
- [15] J. Laut, E. Henry, O. Nov, and M. Porfiri, "Development of a mechatronics-based citizen science platform for aquatic environmental monitoring," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 5, pp. 1541–1551, Oct. 2014.

- [16] S. Omari, M.-D. Hua, G. Ducard, and T. Hamel, "Hardware and software architecture for nonlinear control of multicopter helicopters," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 6, pp. 1724–1736, Dec. 2013.
- [17] P. Apkarian and D. Noll, "Nonsmooth \mathcal{H}_∞ synthesis," *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 71–86, 2006.
- [18] P. Gerum, "Xenomai - implementing a rtos emulation framework on gnu/linux," *White Paper*: <http://www.xenomai.org>, 2004. [Online]. Available: <http://www.xenomai.org/documentation/branches/v2.3.x/pdf/xenomai.pdf>
- [19] D. Leith and W. Leithead, "Survey of gain-scheduling analysis and design," *International Journal of Control*, vol. 73, no. 11, p. 1001–1025, 2000.
- [20] F. R. Garza and E. A. Morelli, "A collection of nonlinear aircraft simulations in matlab," *NASA Langley Research Center, Technical Report NASA/TM*, vol. 212145, 2003.
- [21] J. Hadamard, "Sur les problèmes aux dérivées partielles et leur signification physique," *Princeton University Bulletin*, vol. 13, no. 49-52, p. 28, 1902.
- [22] S. Dobre, "Global sensitivity and identifiability analyses. application to estimation of photophysical parameters in photodynamic therapy," Ph.D. dissertation, Université Henri Poincaré-Nancy I, 2010.
- [23] L. Ljung, *System Identification - Theory for the User*. Prentice-Hall, 1999.
- [24] E. Walter and L. Pronzato, *Identification of Parametric Models from Experimental Data*. Springer, 1997.
- [25] R. Brun, P. Reichert, and H. R. Künsch, "Practical identifiability analysis of large environmental simulation models," *Water Resources Research*, vol. 37, no. 4, p. 1015–1030, 2001.
- [26] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control: Analysis and Design, Second Edition*. Wiley New York, 2007, vol. 2.