9-2005

Nonlinear Regression Methods for Estimation

Eric B. Nelson

Nonlinear Regression Methods for Estimation

DISSERTATION

Eric B. Nelson, Captain, USAF

AFIT/DS/ENG/05-05

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/DS/ENG/05-05

Nonlinear Regression Methods for Estimation

DISSERTATION

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy in Electrical Engineering

Eric B. Nelson, B.S.Mech.Engr., M.S.Aero.Engr., M.B.A.

Captain, USAF

September 2005

AFIT/DS/ENG/05-05

Nonlinear Regression Methods for Estimation

Eric B. Nelson, B.S.Mech.Engr., M.S.Aero.Engr., M.B.A.

Captain, USAF

Approved:

Dr. Meir Pachter Chairman	Date
Dr. Peter S. Maybeck Member	Date
Dr. Mark E. Oxley Member	Date
Dr. Andrew G. Sparks Member	Date
Dr. Anthony N. Palazotto Dean's Representative	Date

Accepted:

Robert A. Calico, Jr. Dean, Graduate School of Engineering and Management	Date

AFIT/DS/ENG/05-05

Abstract

One's choice of estimation technique is highly dependent upon the application. In this dissertation, regression techniques are developed for batch estimation and applied to three specific areas, namely, ballistic trajectory launch point estimation, adaptive flight control, and radio-frequency target triangulation. Specifically, linear regression with an intercept is considered in detail. First, an augmentation formulation is developed. Then, linear regression similar to UD factorization in Kalman filtering is considered, in which an intercept is determined from the average of measurements and is subtracted from these measurements. Extensions of theory are applied to nonlinear regression applications as well. Examples of linear and nonlinear regression with an intercept are used to consider passive parameter estimation applications. Further examples use the intercept parameter estimate to identify the effects of trim change that are associated with the occurrence of a control surface failure. Then, these estimates are used to adjust the inner loop control gains via a feedforward command, hence providing an automatic reconfigurable retrim of an aircraft. Finally, the author uses the developed linear regression algorithms to consider reduced information applications, such as initial position target determination from bearings-only measurement data. Problem reformulation is considered such that estimation errors due to modelling error assumptions are minimized. Additionally, linearization-induced-truncation errors are avoided altogether; hence, the method eliminates the need for an intercept parameter in this particular application. In total, this dissertation develops algorithms for batch processes that broaden the envelope of successful estimation within the three aforementioned application areas. Additionally, the developed batch algorithms do not adversely impact the estimation ability in cases that are already estimated successfully by conventional approaches.

Acknowledgments

I appreciate the collaboration with Mr. Stanton Musick from AFRL/SNAT, Wright Patterson AFB, Ohio. Thanks for securing the funding dollars that support my area of research.

I appreciate having worked with Mr. Corey Schumacher from AFRL/VACA, Wright Patterson AFB, Ohio. Thanks for the insights he provided in the collaborative estimation effort using only Angle of Arrival Measurements.

I appreciate having worked with Mr. Philip Chandler from AFRL/VACA, Wright Patterson AFB, Ohio. Thanks for providing my initial education in the collaborative estimation research area and for providing feedback in the nonlinear estimation and missile trajectory tracking application.

I would like to thank Prof. Peter Maybeck, Prof. Mark Oxley, and Dr. Andrew Sparks for their continued insightful advice on issues pertaining to this research effort. For instance, their insights on the potential need for intercept generalization were invaluable for this dissertation. Sometimes hearing "these results do not show me anything, you can come up with a better example within this application" is exactly what I needed to hear. Thanks again for your efforts as part of my PhD committee.

I would like to thank Prof. Meir Pachter for his continued support, encouragement, and guidance over the past three years. He is a fantastic research advisor and committee chair. I enjoyed working with you.

I would like to thank my wife, Amy, and daughter, Hanna, for their encouragement in the day-to-day, their prayers of support, and for putting up with the life style that accompanies a graduate studies program. Thanks to God for his fellowship over the past three years. I look forward to the next chapter in my life.

Eric B. Nelson

Table of Contents

Page

List of Figures

List of Tables

List of Theorem, Lemmas, and Corollary

List of Symbols

Symbol		Page
x_o	Parameter element; position coordinate	5
y_o	Parameter element; position coordinate	5
V_{x_o}	Parameter element; velocity component	5
V_{y_o}	Parameter element; velocity component	5
$\hat{\theta}$	Parameter estimate	11
$\tilde{V}$	Equation error; $W + V$	18
$\tilde{\alpha}$	Input matrix	20
X	Total state variable	28
U	Total control variable	28
u	Control perturbation	28
$\bar{X}$	Trim state	28
$\bar{U}$	Trim control	28
x_i	Observer's known x-position	30
y_i	Observer's known y-position	30
x	Emitter's x-position	31
y	Emitter's y-position	31
$\mathbf{H}$	Augmented regressor	34
$\boldsymbol{\theta}$	Augmented parameter	34
$\hat{c}$	Estimate of intercept	38
$\bar{z}$	Measurement column mean; $\frac{1}{N}e^T Z$	40
$\bar{h}_i$	Regressor's ith column mean; $\frac{1}{N}e^T h_i e$	40
$\bar{Z}$	Vector comprising of measurement column averages	40
$\bar{H}$	Matrix comprising of regressor column averages	40
$\tilde{Z}$	Measurement perturbation variable	41
$\tilde{H}$	Perturbation regressor	41
M	Transformation matrix	41
I	Identity matrix	41
T	Transformation matrix	43

Symbol		Page
D	Diagonal matrix	43
d_i	Components of diagonal matrix D	43
K	Transformation matrix	43
θ_1	Parameter of reduced full rank system of $N-1$ equations	43
$x(t)$	Truth signal	45
ω	Angular frequency	45
A, B	Coefficients in harmonic signal example	45
c'	DC component of signal	45
$\mathcal{A}$	Signal amplitude	45
ϕ	Phase	46
$x_m(t)$	Measured signal	46
$v_x(t)$	Measurement noise for position	46
$v_{\ddot{x}}(t)$	Measurement noise for acceleration	46
ΔT	Sampling interval	48
w	Measurement error	61
W	Equation error	62
$q^{(k)}$	Tuning parameter	62
ξ	Equation error constraint	66
z_o	Parameter element; position coordinate	69
V_{z_o}	Parameter element; velocity component	69
c_b	Ballistic coefficient	69
g	Acceleration of gravity	69
$R(t)$	Range	70
θ	Elevation	70
T	Duration of measurement interval	71
V_o	Muzzle velocity	71
Γ	Input matrix	76
$\tilde{\alpha}$	Input matrix	86

Symbol		Page
$r(\theta)$	Scalar function of the parameter θ	121
Z_α	Stability derivative for angle of attack; component of the change in angle of attack equation	124
Z_q	Stability derivative for pitch rate; component of the change in angle of attack equation	124
Z_δ	Control derivative; component of the change in angle of attack equation	124
$A(s)$	Actuator transfer function	124
$r(\delta_c)$	Elevator command	125
τ	Time lag constant	125
K_P	Proportional gain	125
K_I	Integral gain	125
K_α	Linear gain applied to the angle of attack	125
z	"Integrator charge"	125
r	Pilot input	127
F_{tail}	Aerodynamic force exerted on the tail	129
l_t	Distance between the aircraft's center of gravity and the tail	129
m	Mass of the aircraft	129
Δx_{CG}	Distance between the aircraft's aerodynamic center and center of mass	129
$\bar{q}$	Dynamic pressure	129
η	Aerodynamic efficiency coefficient	129
S_t	Tail's surface area	129
δ_{trim}	Change in trim	130
δ_d	Step elevator setting disturbance	130
d	Disturbance	131
δ_t	Feed-forward command	135
σ	Measurement error standard deviation	168
v_i	Zero-mean Gaussian-distributed measurement noise	170

Symbol		Page
w_i	Zero-mean Gaussian-distributed measurement noise	170
$\tilde{R}$	Scaled equation error covariance matrix	172
ρ	Radius of circle around emitter	172

List of Subscripts and Diacritical Markings

List of Abbreviations

Abbreviation		Page
UD	U – Upper triangular and unitary matrix; D-Diagonal matrix	iv
RF	Radio frequency	6
ILS	Iterative Least Squares	7
MAMBA	Mobile Artillery Monitoring Battlefield Radar	18
(	Iterative Generalized Least Squares	22
NASA	National Air and Space Administration	22
AFWAL	Air Force Wright Aeronautical Laboratory	23
SRFCS	Self Repairing Flight Control System	23
CRCA	Control Reconfigurable Combat Aircraft	23
FDI	Fault Detection and Isolation	23
SDC	Self Designing Controller	24
VISTA	Variable In-Flight Simulator Test Aircraft	24
AFRL	Air Force Research Laboratory	24
RESTORE	Reconfigurable Flight Control of Tailless Fighter Aircraft	24
UAV	unmanned air vehicle	25
FCS	flight control system	25
AOA	Angle of Arrival	29
DSP	Digital Signal Processing	29
GPS	Global Positioning System	31
PI	Proportional Integral, pertaining to a control law	126

Nonlinear Regression Methods for Estimation

I. Introduction

Batch estimation hinges on the solution to a regression problem. In particular, a linear regression with intercept c [28] is considered:

$$Z = H \cdot \theta + e \cdot c + V \quad , \quad V \in \mathcal{N}(0, R) \tag{1.1}$$
$$Z, V \in \mathbb{R}^N, \quad R \in \mathbb{R}^{N \times N}, \quad \theta \in \mathbb{R}^n, \quad c \in \mathbb{R}^1$$

in which the regressor H is an $N \times n$ matrix, Z is the measurement vector, θ is the parameter vector of interest, and e is an N-vector of ones

$$e = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}_{N \times 1} .$$

The equation error V is zero-mean, Gaussian-distributed with equation error covariance R. The equation error covariance R is an $N \times N$ real, symmetric positive definite matrix. A closed-form solution for the linear regression with intercept is developed in Section 3.1 and summarized in Theorem 3.

One should consider two major areas of application for the batch estimation paradigm described herein. First, the presence of a bias c could signal a failure, resulting in an offset of the nominal dynamics equation. In the context of aircraft flight control, control surface failure will induce a change in trim. Specifically, in Chapter V, the pitch control channel of an aircraft is considered. The pertinent flight control equation is [9], [30]

$$\dot{q} = M_\alpha \cdot \alpha + M_q \cdot q + M_\delta \cdot \delta, \quad 0 \leq t \leq T \tag{1.2}$$

The measured variables are: $\delta(t)$, $q(t)$, $\alpha(t)$, $\dot{q}(t)$, in which δ represents an elevator deflection, q represents pitch rate, α denotes angle of attack, and $\dot{q}$ is the pitch acceleration. The parameter is defined: $\theta = (M_\alpha,\ M_q,\ M_\delta)^T \in \mathbb{R}^3$, in which M_α and M_q are stability derivatives and M_δ is the control derivative of the pitch plane. The possible change in trim is quantified as:

$$c = M_\delta \cdot \bar{\delta}$$

in which $\delta_e \triangleq \delta - \bar{\delta}$. The dynamics equation becomes

$$\dot{q} = M_\alpha \alpha + M_q q + M_\delta \delta_e + c \quad .$$

Measurements are obtained for delta deflection δ_e, pitch rate q, angle of attack α, and pitch acceleration $\dot{q}$. Measurement noise is assumed as zero-mean, Gaussian-distributed. Hence, measurements, denoted by the subscript m, are

$$\begin{aligned}
\dot{q}_m &= \dot{q} + v_{\dot{q}_m} \\
q_m &= q + v_{q_m} \\
\alpha_m &= \alpha + v_{\alpha_m} \\
\delta_m &= \delta_e + v_{\delta_m}
\end{aligned}$$

One can rearrange these measurement equations and insert them into the pitch plane flight control equation to obtain

$$\dot{q}_m = [\alpha_m \quad q_m \quad \delta_m] \cdot \theta + e \cdot c + v_{\dot{q}_m} - M_\alpha \cdot v_\alpha - M_q \cdot v_q - M_\delta \cdot v_\delta \tag{1.3}$$

in which the parameter $\theta \triangleq \begin{bmatrix} M_\alpha \\ M_q \\ M_\delta \end{bmatrix}$ and e is an N-vector of ones. One should note that N indicates the window size or, equivalently, the number of measurements within the batch. One can now relate (1.3) to (1.1). The measurement vector is

$$Z = \begin{bmatrix} \dot{q}_{m_1} \\ \vdots \\ \dot{q}_{m_N} \end{bmatrix}$$

the regressor matrix H is

$$H = \begin{bmatrix} \alpha_{m_1} & q_{m_1} & \delta_{m_1} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_{m_N} & q_{m_N} & \delta_{m_N} & 1 \end{bmatrix}$$

and the equation error V is

$$V = \begin{bmatrix} v_{\dot{q}_{m_1}} - M_\alpha \cdot v_{\alpha_{m_1}} - M_q \cdot v_{q_{m_1}} - M_\delta \cdot v_{\delta_{m_1}} \\ \vdots \\ v_{\dot{q}_{m_N}} - M_\alpha \cdot v_{\alpha_{m_N}} - M_q \cdot v_{q_{m_N}} - M_\delta \cdot v_{\delta_{m_N}} \end{bmatrix} \quad .$$

The determination of the equation error covariance R is saved for later chapters. In Chapters III and V, novel batch estimation algorithms are developed to address applications that require the modelling of a bias. Examples are presented that highlight the effectiveness of the estimation methodology herein developed. Estimating $\hat{c}$ affords automatic retrimming using feed-forward control, as opposed to relying solely on feedback control and integral action.

Second, the dissertation research gives attention to nonlinear regression, which is the primary focus of Chapter IV and is also considered in Chapter VI. The nonlinear regression is:

$$Z = h(\theta) + V \quad , \quad V \in \mathcal{N}(0, R) \tag{1.4}$$

in which Z is a $\mathbb{R}^N$-valued measurement random vector, $\theta \in \mathbb{R}^n$ is the parameter, $N \geq n$, and h is a nonlinear function of the parameter θ. The equation error V is a $\mathbb{R}^N$-valued random vector that is zero-mean, Gaussian-distributed with covariance R, which is an $N \times N$ real, symmetric positive definite matrix. In the conventional approach, an Iterative Least Squares algorithm is used to obtain the parameter estimate [15]. An iterative parameter estimation algorithm is developed. Thus, suppose that a preliminary parameter estimate $\hat{\theta}^{(k)}$ is available. One then expands the nonlinear differentiable function $h(\theta)$ about the current parameter estimate $\hat{\theta}^{(k)}$,

$$\begin{aligned} h(\theta) &= h(\hat{\theta}^{(k)} + \theta - \hat{\theta}^{(k)}) \\ &= h(\hat{\theta}^{(k)}) + \frac{\partial h}{\partial \theta}(\hat{\theta}^{(k)}) = \hat{\theta}^{(k)} \cdot (\theta - \hat{\theta}^{(k)}) + r \quad . \end{aligned}$$

One can insert the results of this expansion into (1.4) in order to yield

$$Z + H_k \cdot \hat{\theta}^{(k)} - h(\hat{\theta}^{(k)}) = H_k \cdot \theta + r + V \quad . \tag{1.5}$$

The N-vector residual r is the truncation error, which is explicitly included in the derivation, and the $N \times n$ regressor matrix is the Jacobian

$$H_k \triangleq \frac{\partial h}{\partial \theta}(\hat{\theta}^{(k)}) \quad .$$

Notice that this derivation pursues a regression in terms of total variables; hence, the left-hand side of (1.5) (i.e., the measurement) assumes the expanded form of $Z + H_k \cdot \hat{\theta}^{(k)} - h(\hat{\theta}^{(k)})$. Using an intercept in nonlinear regression is meant to account for the truncation error r caused by linearization. In the case of a strong nonlinearity, estimates benefit by accounting for the nonlinearity presence by using an intercept $\hat{c}$ [22].

In order to put this into context, one can consider the "clean measurement" relation for azimuth angle ϕ at a time $t = k\Delta T$, as discussed in Chapter 4.3. By clean, the author asks the reader momentarily to assume that the bearing measurement ϕ is free of noise. This assumption will enable the author to relate geometric relation to the

assumed truth kinematics first, before reinserting the noise corrupted measurements. For ballistic trajectory determination from battlefield radar measurements [2]

$$\tan\left(\phi(k\cdot\Delta T)\right)=\frac{y_o+V_{y_o}k\Delta T}{x_o+V_{x_o}k\Delta T}$$

in which $k=1,\ldots,N$. The parameter in this equation is the launch information $\theta=(x_o,\ y_o,\ V_{x_o},\ V_{y_o})^T\in\mathbb{R}^4$, in which (x_o,y_o) are the launch point coordinates and (V_{x_o},V_{y_o}) are the velocity coordinates. The measurement record is

$$\phi_m(k\cdot\Delta T)=\phi(k\cdot\Delta T)+v_\phi(k)$$

in which the subscript m denotes measured variables and v_ϕ is additive zero-mean, Gaussian-distributed noise quantified by the radar's measurement uncertainty standard deviation σ_ϕ. The radar measurement equation for azimuth is rearranged and inserted into the clean measurement equation to obtain

$$\tan\left(\phi_m(k\cdot\Delta T)-v_\phi(k)\right)=\frac{y_o+V_{y_o}k\Delta T}{x_o+V_{x_o}k\Delta T} \tag{1.6}$$

in which $k=1,\ldots,N$. The reader should note that the left-hand side of (1.6) is a nonphysical mathematical construction that provides a convenient form to pursue the components of (1.5) that pertain to the observation relation. For instance, one attains the vector $h^{(\phi)}(\theta)$ from the nonlinear observation function h defined by

$$h^{(\phi)}(\theta)\ \triangleq\ \begin{pmatrix}\frac{y_o}{x_o}\\ \frac{y_o+V_{y_o}\Delta T}{x_o+V_{x_o}\Delta T}\\ \vdots\\ \frac{y_o+V_{y_o}\Delta T(N-1)}{x_o+V_{x_o}\Delta T(N-1)}\end{pmatrix}\quad .$$

The observation equation $h(\theta)$ is linearized about the current parameter estimate $\hat{\theta}^{(k)}$ at step k. Section 4.4.4 determines the components of the linearized regressor matrix in detail.

The strength of the nonlinearity r in (1.5) that is neglected depends on the relative position between radar and projectile launch point, the sampling rate, and the data window size. The presence of a strong nonlinearity in the regressor corresponds to rapid movement in the azimuth angle relative to the characteristic time scale. The reader should consider a case with large azimuth angle excursions within the data window, such as that portrayed in Figure 4.14(a) between $\sim$0.45 to 0.65 nondimensional time units. The plot shows the radar rotating over 100^o in this relatively short time interval. Because the standard ILS parameter estimation algorithm neglects the higher order residual r, parameter estimation is unobtainable when enough measurements from this time interval are included in the batch data window, as shown in Figures 4.19(a) and 4.20(a).

In this dissertation, one attempts to mitigate the source of this linearization-induced truncation error by estimating the residual r that was previously neglected in (1.5) by estimating an intercept c. The theory for including the residual using an intercept parameter is developed in Chapter IV and applied in Chapters IV and VI. The result is to achieve valuable parameter estimates for problem geometry scenarios that were formerly unobtainable, as in the aforementioned example culminating in Figures 4.19 and 4.20.

The dissertation is organized as follows. Chapter II contains a literature review of material concerned with batch estimation, and then addresses material specific to ballistic trajectory launch point estimation, adaptive aircraft flight control, and radio-frequency (RF) target triangulation. These specific applications are important, as they correspond to the application areas considered later in Chapters IV-VI. In Chapter III, the augmentation of the parameter space with an intercept is motivated and a novel linear regression method for parameter estimation is analyzed. The thesis

then considers linear regression with an intercept in a non-augmented form. The batch estimation, linear regression, and two new methods that incorporate an intercept to account for unmodelled bias, are used in the context of estimating the frequency of a sinusoidal waveform. When a DC offset is present, parameter estimates converge to the correct true parameter values when the two new estimation methods are used, while estimates do not converge while using standard Iterative Least Squares (ILS) [[23], [24], and [28]]. It is also important to note that the augmented ILS algorithm with an constrained intercept does not adversely impact estimate quality in the nearly linear case when compared to standard ILS, provided that sufficient data is available.

Chapter IV addresses specific applications in which using nonlinear regression for parameter estimation and augmenting the parameter with an intercept is beneficial. The need for balancing the linearization-induced truncation error with the measurement-noise-induced equation error using a Ridge Regression argument from statistics [[13], [14]] is discussed. Constraining the intercept is discussed in detail. The herein developed parameter identification process is demonstrated in the context of a ballistic trajectory determination using radar measurements. Consistently improved estimation performance by the novel parameter estimation algorithm over conventional Iterative Least Squares estimation is achieved. The simulation results demonstrate that the estimation error covariance is reduced, the predicted estimation error is commensurate with the experimentally obtained covariance, and the envelope of measurement geometries in which good estimation is possible, despite a strong nonlinearity, is expanded. Additionally, estimation results do not degrade from standard ILS performance in cases in which the nonlinearity strength is weak.

Chapter V highlights the main features of the novel aircraft parameter estimation methodology and makes the case for using feed-forward control derived from the on-line intercept estimate, as opposed to relying on feed-back control and integral action for disturbance rejection under control effector failure. An F-16 class aircraft model is presented, narrowing the scope of the modelling effort to the inner loop control system for the pitch channel short period dynamics. An elevator failure is

considered that reduces the control surface effectiveness by 50%. Section 5.3 develops the augmented linear regression specific to parameter estimation of the F-16 critical stability and control derivatives using the pitch channel's $\dot{q}$ equation. In Section 5.4, simulation results are presented and discussed. The simulations demonstrate that the real-time parameter estimation methodology developed herein improves on the parameter estimate quality when the intercept parameter is included to account for a possible trim change during and after failure. After the failure, an automatic re-trimming method which uses system identification for feed-forward control produces superior aircraft response over the conventional approach of exclusively relying on feedback and integral action for retrimming. In addition, the simulation results verify that the proposed reconfigurable flight controller does not adversely impact aircraft handling and response under unfailed conditions.

Chapter VI is concerned with obtaining an RF emitter's position estimate from bearings-only measurements using triangulation. Standard approaches assume azimuth angle measurements, and an ILS algorithm is used to calculate the emitter's position. The modified triangulation approach developed in this chapter is based on the insight that, in practice, the bearing measurement's phasor ($\cos\phi$, $\sin\phi$) is available directly. Hence, incorporating the phasor into the estimation algorithm represents the measurement hardware more accurately when compared to the conventional azimuth angle measurement assumption. This approach also allows for the use of an efficient linear regression algorithm and avoids equation error due to linearization. As a result, the modified formulation for emitter geo-location using bearings-only measurements produces superior parameter estimates, that is, improved emitter geo-location, compared to the standard formulation and ILS algorithm. Concluding remarks follow in Chapter VII.

II. Literature Review

Most of the estimation work currently performed entails recursive algorithms that handle a wide variety of applications. However, this dissertation is concerned with *batch* estimation that successfully addresses the applications discussed in Chapters III through VI, namely, ballistic trajectory launch point estimation, adaptive aircraft flight control, and RF target triangulation. Batch estimation may well provide the methodology of choice for cases in which a closed-form solution to the dynamics equations is available. Novel batch estimation algorithms are developed and employed for the aforementioned applications. These methods may prove useful in additional applications, such as those in which a bias caused by failure might appear and applications featuring a strong nonlinear observation equation and linear dynamics. Thus, related research on batch estimation using linear regression is reviewed.

This chapter is organized as follows. First, estimation methods in current use are outlined. Then, a brief overview of batch estimation is provided. Next, literature and overview material related to each application discussed in this dissertation is considered. Specifically, Chapter III develops two methods of augmenting the linear regression with an intercept. A vibration's frequency is estimated while the single-tone signal is offset with a DC component. Chapter IV discusses projectile launch point estimation. Chapter V considers using linear regression in order to provide fault detection for an aircraft flight control system. Moreover, one can use the intercept that is identified on-line in order to retrim the aircraft using feed-forward control rather than relying on integral action solely. The magnitude of the estimated change in trim is used to retrim the aircraft using feed-forward control. In Chapter VI, it is shown how one could use bearings-only/angle-of-arrival measurements in order to triangulate a radar's position efficiently. Finally, a summary section is provided.

2.1 Estimation Techniques

Deterministic approaches are typically appropriate for applications in which the dynamics are well known and the operating environment is predictable. For example, one could consider satellite orbit determination. The satellite's orbit is governed by a nearly deterministic process, that is, orbital mechanics. As a result, a satellite's position and velocity states are obtainable to a high fidelity with deterministic estimation techniques [[35], [36]]. One approach entails a batch least squares algorithm [[18], [36]]. Alternatively, a recursive approach typically used for deterministic estimation is Bayes filtering. Wiesel summarizes this recursive process in [36]. This is similar to inverse covariance Kalman filtering [19], except the presence of process noise in the dynamics model of the inverse covariance Kalman filter is ignored.

A second class of estimation paradigms acknowledges the stochastic nature of dynamic processes. One is then inclined to use a recursive estimation approach. Numerous researchers have explored the benefits of recursive algorithms such as Kalman filtering. In stochastic systems, Kalman filtering is fully equivalent to inverse covariance Kalman filtering. This methodology is suited nicely to incorporate uncertainty into the system's assumed dynamics, such as in a system identification application on-board a fighter aircraft. The standard and extended Kalman filter are derived and aptly demonstrated in Maybeck [[18], [19], [20]].

The aforementioned methods are well suited for some classes of applications. Numerous factors influence the estimation technique one may chose to use, including the desired precision of the estimate, the desired quality of the estimate, computational resource availability, knowledge about the problem/modelling issues, such as how close the assumed dynamics and assumed measurement relations match reality, and the environment's uncertainty. The desired precision of the estimate refers to the degree of repeatability of estimation convergence to the truth parameter value. This is measured by the standard deviation of the experimentally determined estimate. One is also interested in the filter-predicted estimation error covariance. It is very

important that the filter-predicted parameter estimation error covariance reflects the actual parameter estimation error covariance. When considering the specific classes of problems discussed in Chapter I, one is motivated to reconsider the conventional recursive estimation methods and batch estimation.

2.2 Standard Linear Regression

First, one should consider the standard linear regression framework from statistics:

$$Z = H\theta + V \quad , \quad V \in \mathcal{N}(0, R) \tag{2.1}$$

in which Z is the measurement vector, H is the regressor, θ represents the parameters, and V is the Gaussian-distributed equation error. The least squares method yields the parameter estimate $\hat{\theta}$ that minimizes the norm of the residual vector r, in which $Z - H\hat{\theta} = r$. As a result, the parameter estimate is $\hat{\theta} = (H^T H)^{-1} H^T Z$ for an identity weighting matrix. The minimum variance estimate of the standard linear regression is derived in [36]. The minimum variance parameter estimate for (2.1) is

$$\hat{\theta} = (H^T R^{-1} H)^{-1} H^T R^{-1} Z$$

for R^{-1} as weighting matrix in quadratic cost, in which

$$R = \mathbf{E}(V \cdot V^T)$$

The equation error covariance matrix reflects the difference between is the error between the parameter estimate and the true parameter values.

$$P_{\hat{\theta}} = \mathbf{E}((\hat{\theta} - \theta)(\hat{\theta} - \theta)^T)$$

so that [36]

$$P_{\hat{\theta}} = (H^T R^{-1} H)^{-1} \quad .$$

This is also the maximum likelihood parameter estimate.

Now one should reconsider the two primary problem types that are addressed in this dissertation. First, Chapters III and V consider the impact of bias, or trim change,

that often times, is neglected. The standard estimation methods are oblivious about the occurrence of a bias, and therefore, produce incorrect estimates. For example, it is demonstrated in Chapter III that the estimate of a vibration frequency in the presence of a neglected DC signal is wrong, as shown in Figure 3.4. The true harmonic frequency is $\omega = 0.5\frac{rad}{s}$, hence $\hat{\theta} = \hat{\omega}^2$ should approach $0.25\frac{rad^2}{s^2}$. However, the parameter estimate using standard ILS approaches zero while the single tone vibration is impacted by the DC offset. Hence, one must plan for the possibility of a DC offset.

In Chapter V, the impact of using the standard linear regression in adaptive flight control is discussed. By adaptive, the author reveals that when a trim change occurs at the onset of failure, the estimation process must accommodate the change in order to still produce satisfactory stability and control derivative estimates. When a control surface failure occurs, the standard linear regression is not able to accommodate a trim change that will likely accompany the onset of failure, as shown in Figures 5.13 through 5.15. For instance, Figure 5.13(a) shows an erratic, rapidly varying parameter estimate $\hat{M}_\alpha$ after the onset of failure at $t = 5$ seconds. The impact of poor estimates is significant in reconfigurable control. The trim change is captured by the intercept c that is introduced in the novel estimation algorithm. Moreover, in the reconfigurable flight control work, the estimate of the intercept c is used to calculate a feed-forward command, such that no change in trim is experienced. In turn, the feed-forward control further improves the estimates of the stability and control derivatives. Feed-forward control is clearly superior to traditional, nonadaptive flight control that relies solely on integral action for retrimming the aircraft, as illustrated in Figure 5.34. The figure shows that the reconfigurable flight control system produces superior pitch rate tracking after failure at 5.0 seconds when compared to the traditional flight control system.

Second, one should consider the linearization-induced-truncation error caused by neglecting the residual r in (1.5). The standard Iterative Least Squares estimation method is introduced in Chapter I. The implications for the ballistic launch point determination problem are discussed. The conclusion is that standard ILS restricts

the operational envelope of successful estimation, particularly in those cases in which the post-linearization-neglected residual is large, that is, the nonlinearity is strong.

Chapter VI discusses the application of obtaining an emitter's position estimate from bearings-only measurements using triangulation. The scenario considered entails receivers on aircraft, pinged by an emitter at an unknown location. The data is accumulated from the various platforms; hence, the process is cooperative and takes advantage of measurements of opportunity. In this case, the failure of the standard ILS approach to capture the effect of linearization is compounded by the measurement modelling error. Figure 6.8(b) shows the inability of standard ILS to obtain reasonable estimates when typical noisy measurement data is used. The ten Monte Carlo experiments reveal an average parameter estimate varying between 500 to 1000 meters from the adjusted true emitter position. On the other hand, Figure 6.8(a) shows good position estimates for the emitter position that approach the true emitter position as the window size increases. This provides additional motivation to pursue augmented linear regression that handles the demands of these applications.

2.3 Application Areas

This section presents reference material pertinent to the development for the applications discussed in Chapters III to VI. In order to encourage the reader, the author reveals significant intermediate results pertinent to the reference material. In addition, the author provides figures of significant results to indicate the overall objectives of this research.

2.3.1 Previous Work Using Linear Regression with Intercept for Identification. Current research [[7], [9], [11], [25]] considers the benefits and potential applications of static system identification and batch parameter estimation, using modified linear regression techniques instead of conventional batch methods [[15], [36]]. This literature motivates efforts to formalize the batch process estimation theory in Chapter III, which culminates in deriving the explicit solution to the augmented linear regression

with intercept. Batch parameter estimation is successfully used in [7] for the identification of distributed aircraft control derivatives in real-time. A regularized minimum variance estimate of the stability and control derivatives is obtained at each time step. In [25], a stochastic framework is developed to derive a rigorous centralized Differential Global Positioning System estimation algorithm for a formation of air vehicles. The users' positions and velocities are estimated using kinematic GPS and batch nonlinear regression.

Several efforts address additional applications that allow for the presence of bias and use batch linear regression. Concerning the inclusion of an intercept in the estimation algorithm [11], an adaptive control problem is considered that uses data generated by a linear regression model in which intercept and slope coefficients are unknown. An assumed certainty equivalence based control rule using estimates of the intercept and slope coefficients is proposed. In [9], a control surface failure is considered in the context of a reconfigurable flight control system. This particular application is now described in detail in order to show how the system identification does in fact reduce to a linear regression problem. The estimates and filter-predicted equation error covariance are obtained without iteration [9]. The research presented in this dissertation will use this simplification concept in order to produce the estimates from the augmented linear regression techniques developed in Chapter III. The rigorous solution to Theorem 3 enables an effective reconfigurable flight control system that counters the change in trim condition that may occur with control surface failure.

Specifically, linear regression is used in the system identification/parameter estimation algorithm. Thus, the linear regression is

$$Z = H\theta + V \quad , \quad V \in \mathcal{N}(0, R(\theta))$$

in which the regressor H is in augmented form, i.e., an intercept is incorporated and the equation error V is Gaussian with parameter dependent equation error covariance

[9]

$$R(\theta) = r(\theta)I \quad .$$

The reader should recall the pertinent flight control equation (1.2) for the pitch plane mentioned in Chapter I. Measurements are obtained for delta deflection δ, pitch rate q, angle of attack α, and pitch acceleration $\dot{q}$. Measurement noise is assumed as zero-mean, Gaussian-distributed. Hence:

$$\begin{aligned} \dot{q}_m &= \dot{q} + v_{\dot{q}_m} \\ q_m &= q + v_{q_m} \\ \alpha_m &= \alpha + v_{\alpha_m} \\ \delta_m &= \delta + v_{\delta_m} \end{aligned} \tag{2.2}$$

One can rearrange these measurement equations and insert them into the pitch plane flight control equation to obtain

$$\dot{q}_m = [\alpha_m \quad q_m \quad \delta_m] \cdot \theta + v_{\dot{q}_m} - M_\alpha \cdot v_\alpha - M_q \cdot v_q - M_\delta \cdot v_\delta \tag{2.3}$$

in which the parameter is $\theta \triangleq \begin{bmatrix} M_\alpha \\ M_q \\ M_\delta \end{bmatrix}$. As one of the key developments in [9], the equation error covariance matrix is then calculated as

$$\begin{aligned} R(\theta) &= \mathbf{E}[V \cdot V^T] \\ &= \mathbf{E}\left(\begin{bmatrix} v_{\dot{q}_1}^2 - M_\alpha^2 v_{\alpha_1}^2 + M_q^2 v_{q_1}^2 + M_\delta^2 v_{\delta_1}^2 \\ \vdots \\ v_{\dot{q}_N}^2 - M_\alpha^2 v_{\alpha_N}^2 + M_q^2 v_{q_N}^2 + M_\delta^2 v_{\delta_N}^2 \end{bmatrix} \cdot \begin{bmatrix} v_{\dot{q}_1}^2 - M_\alpha^2 v_{\alpha_1}^2 + M_q^2 v_{q_1}^2 + M_\delta^2 v_{\delta_1}^2 \\ \vdots \\ v_{\dot{q}_N}^2 - M_\alpha^2 v_{\alpha_N}^2 + M_q^2 v_{q_N}^2 + M_\delta^2 v_{\delta_N}^2 \end{bmatrix}^T \right) \\ &= \left(\sigma_{\dot{q}}^2 + M_\alpha^2 \sigma_\alpha^2 + M_q^2 \sigma_q^2 + M_\delta^2 \sigma_\delta^2\right) \cdot I_{N \times N} \end{aligned}$$

in which M_α and M_q are the aircraft's stability derivatives in the pitch channel, M_δ is the control derivative, and σ_α, σ_q, σ_δ, and $\sigma_{\dot{q}}$ are the corresponding standard deviations associated with each measurement, and N represents the number of measurements in the batch. Hence,

$$R(\theta) = r(\theta) \cdot I_{N \times N}$$

is homoscedastic (a scaled identity matrix), in which the scalar $r(\theta)$ is defined as

$$r(\theta) = \sigma_{\dot{q}}^2 + M_\alpha^2 \sigma_\alpha^2 + M_q^2 \sigma_q^2 + M_\delta^2 \sigma_\delta^2 \quad .$$

The ability to express the equation error covariance $R(\theta)$ as an identity matrix multiplied by a scalar enables the minimum variance estimate determination to reduce from

$$\hat{\theta} = (H^T R^{-1} H)^{-1} H^T R^{-1} Z$$

to the simplified least squares form

$$\hat{\theta} = (H^T H)^{-1} H^T Z \quad . \tag{2.4}$$

Similarly, the filter-predicted estimation error covariance now reduces to

$$\begin{aligned} P &= (H^T R^{-1} H)^{-1} \\ &= r(\theta)(H^T H)^{-1} \quad . \end{aligned} \tag{2.5}$$

Since the regressor H is not parameter-dependent, both the parameter estimate θ and the predicted equation error covariance P are obtained without iteration. This important result is used in the dissertation work presented in Chapter V.

2.3.2 Previous Work of Theory for Nonlinear Applications. Chapter IV advances the novel idea of including an intercept parameter in the nonlinear regression equations in order to account for linearization-induced truncation errors. Chapter IV extends the linear estimation theory developed in Chapter III into nonlinear regression. The research presented in this thesis provides an attack on nonlinearity by including an intercept parameter in nonlinear regression that accounts for linearization-induced truncation error. It is shown that the estimation performance is enhanced, particulary when the nonlinearity is strong. Specifically, the envelope of successful estimation for batch processes is expanded when the modified ILS algorithms incorporate an intercept c to account for the higher order residual r previously neglected in (1.5). Also, when the nonlinearity is weak, the results do not degrade from standard ILS.

This section now discusses observations from [24] about the parameter estimation processes described earlier in Chapter I. The modified iterative least squares algorithm recognizes the presence of an intercept multiplied by an N-vector of ones in order to represent the residual r shown in (1.4), in which N represents the number of measurements within the batch window. Namely in Section 4.1, the nonlinear regression of the following form

$$Z + H_k \cdot \hat{\theta}^{(k)} - h(\hat{\theta}^{(k)}) = H_k \cdot \theta \cdot + e \cdot c + W + V \tag{2.6}$$

is considered in which V and W are independent. Rearranging (2.6) and using the vector of ones notation e, one obtains the reformulated linear regression at the kth iteration step

$$\begin{aligned} Z + H_k \cdot \hat{\theta}^{(k)} - h(\hat{\theta}^{(k)}) &= (\, H_k \vdots e\,) \begin{pmatrix} \theta \\ \dots \\ c \end{pmatrix} + \tilde{V} \\ &= \boldsymbol{H}_k \cdot \boldsymbol{\theta} + \tilde{V} \end{aligned} \tag{2.7}$$

in which the equation error becomes $\tilde{V} \triangleq W + V$ and the equation error covariance matrix $R^{(k)} = \mathbf{E}(\tilde{V}\tilde{V}^T) = R + q^{(k)} \cdot I_N$. The solution of the linear regression (2.7) augmented with the intercept is

$$\hat{\boldsymbol{\theta}}^{(k+1)} = (\mathbf{H}_k^T {R^{(k)}}^{-1} \mathbf{H}_k)^{-1} \mathbf{H}_k^T {R^{(k)}}^{-1} [Z + H_k \hat{\theta}^{(k)} - h(\hat{\theta}^{(k)})]$$

in which $k = 0, 1, \cdots, N$. This formulation is now referred to as augmented iterative least squares with unconstrained intercepts.

The algorithm that implements the augmented ILS with unconstrained intercepts is demonstrated in the context of ballistic trajectory determination from radar measurements, the results of which are shown in Section 4.5.2. This application is relevant on today's battlefield, as demonstrated by the British armed forces' recent acquisition of the Mobile Artillery Monitoring Battlefield Radar (MAMBA) [2] to track enemy projectiles and geo-locate the enemy artillery position. The developed, nonlinear regression methodology is also applicable to a variety of other applications, including satellite trajectory determination.

Clearly, the Monte Carlo simulation experiments in [24] show that, by modelling the linearization/truncation error of the ILS procedure as a constant c, improved estimation results were obtained in two ways. The impact of the bias caused by the neglected nonlinearity during linearization is reduced and the percentage of successful cases that converge to the correct parameter estimate increases, even in the presence of a strong nonlinearity. Figure 4.8 shows an example of one case in which successful estimation convergence case for the augmented ILS algorithm with unconstrained intercepts is obtained while the standard ILS algorithm does not converge for all batch sizes. In all examples in which both estimation algorithms converge, standard ILS converges with a shorter data record when compared to augmented ILS with unconstrained intercept. In addition, when the nonlinearity is weak as in Figure 4.12, the augmented ILS with unconstrained intercept yields noticeable increases in the variance of the parameter estimate when compared to standard ILS, particularly if

too short of a measurement data record is used. In this particular example, the dash-dot and solid lines, corresponding to the estimates produced by the augmented ILS with constrained intercept methods, do not approach the true position value until the window size $N >\sim 85$ measurements. On the contrary, the standard ILS algorithm, as shown by the dashed line, produces a good estimate of the position when $N < 20$. One could conclude that this increase in variance is a direct result from augmentation with the linearization error/intercept parameter, as the intercept would not have much "bias" to absorb in the near-linear case. Again, Monte Carlo simulation experiments summarized within the Section 4.5.2 support this conjecture.

The proposed extension addresses the aforementioned deficiencies with augmented ILS with unconstrained intercepts directly. A concept from statistics, Ridge Regression [13], [14], is adapted to the solution of nonlinear estimation problems. In [14], a new signal processing technique is proposed for improving navigation system accuracy when the geometric dilution of precision (GDOP) causes collinearity, which in turn brings about poor position estimates. The main goal is smaller Mean Square Error (MSE) rather than minimizing the residual. This objective is accomplished by finding an estimator with a parameter estimation variance of $[H^T H + \kappa I]^{-1}$.

This dissertation presents a successful application of Ridge Regression constraints to the linear regression in order to overcome the predicted parameter estimation error variance increase as shown when the augmented ILS algorithm with unconstrained intercept is employed. Additionally, the proposed application of Ridge Regression addresses the data record size issue for achieving rapid convergence: an issue that plagues augmented ILS with unconstrained intercepts. Ridge Regression is used to acknowledge the smallness of a intercept parameter c by enforcing a constraint in the form of prior information. The Ridge Regression constraints balance the linearization-induced truncation error and the measurement-noise-induced equation error that occurs during the linearization of the nonlinear observation function $\|c\| \cong \sigma$; Specifically, in the projectile launch point estimation application previously mentioned in Chapter I, recall that there are three measurement equations in a three-

dimensional example, one for range, azimuth, and elevation. Ridge Regression-type constraints are introduced to enforce the noise balancing requirement; hence, the nonlinear regression is augmented with the three linear equations:

$$
\begin{aligned}
0 &= c_R + \xi_R \quad , \quad \xi_R \in \mathcal{N}(0, \sigma_R^2) \\
0 &= c_\phi + \xi_\phi \quad , \quad \xi_\phi \in \mathcal{N}(0, \tilde{\alpha}^2 \sigma_\phi^2) \\
0 &= c_\theta + \xi_\theta \quad , \quad \xi_\theta \in \mathcal{N}(0, \sigma_E^2)
\end{aligned}
$$

Thus, while acknowledging the presence of nonlinearity, these equations constrain the intercept magnitude in the nonlinear regression and are conducive to good parameter estimates. In addition, the application of Ridge Regression goes beyond addressing matrix noninvertibility due to the singularity of an ill-conditioned system [32].

This enhanced method improves the estimates greatly, as illustrated for a strongly nonlinear case by comparing the results in Figures 4.8 and 4.19. The use of Ridge Regression expands the envelope of measurement geometries in which good estimation performance is achieved despite a strong nonlinearity. Parameter estimates converge more rapidly than in standard Iterative Least Squares when the nonlinearity is strong. Even with a short measurement data record, the experimentally determined parameter estimation error variance is reduced compared to the predicted equation error variance for augmented ILS with constrained intercepts, as shown in Figure 4.19(b). Figure 4.19(a) also shows that the exact opposite is true for the standard ILS estimation method when the nonlinearity is strong. For the strong nonlinear example, neglecting the residual r, originally introduced in (1.5) represents a significant modelling error. Additionally, when given sufficient data in the batch regression process, the constrained augmented ILS methodology does not adversely impact estimate quality in the near-linear case compared to standard ILS. This is important because the modified ILS method proposed in this thesis does not lose the ability to provide good estimates in test cases in which standard ILS already succeeds, as validated by extensive simulation trials, the highlights of which are summarized in Section 4.5.

The research conducted in Chapter IV highlights a few other points that are important to implementation of the estimation methods. First, nondimensionalization improves the conditioning of the estimation process in this example. Because the inversion of $H^T R^{-1} H$ is required, proper scaling is important, particularly when parameter numerical values could range as much as 10^6 in dimensional units. *Matlab* computational precision then becomes an issue.

Second, the application presented in Chapter IV deals with an estimate of an initial state. Traditional recursive approaches are formulated such that a smoother is required to obtain such estimates. In this case, a benefit of batch data processing is that one can formulate the estimation problem to obtain the initial state estimates directly, obviating the need for the mechanics required to smooth using recursive techniques. This is even more significant when the desired parameter estimates (i.e., those values corresponding with the launch time) proceed the measurements included in the data window (i.e., the first measurement occurs when the launch is detected, where $t_{detection} > t_{launch}$) [24]. The initial estimates of $\hat{z}$ and $\hat{V}_z$ are used to determine the launch time estimate from

$$\hat{z} = z_o + (\hat{V}_z + c_b \cdot g \cdot \hat{t}) \cdot \hat{t} - \frac{1}{2} \cdot c_b \cdot g \cdot \hat{t}^2 \quad .$$

The estimate of time is now available to propagate forward the initial state estimates determined by the various ILS methods. Section 4.5 explores this technique to determine the projectile launch point coordinates.

Finally, the need for higher sampling rates might force the division of the measurement window into smaller data records [24]. Such is the case in projectile launch point determination when the azimuth and/or elevation angles change too rapidly, caused by a stressful problem geometry, such as in Figure 4.22. The sampling rate necessary to capture the nonlinearity demonstrated in this example would produce a large data window, making batch processing infeasible for the computational resources available for this thesis research. In addition, rapid sampling is not necessary

for most of the projectile trajectory. Chapter IV uses Monte Carlo simulations to investigate a sub-batch (Figure 4.23) under the condition that the nonlinearity is relatively weak. For this initial state application, batch nonlinear regression is easily performed, regardless of the overall size of the data window, as Iterative Generalized Least Squares (IGLS) results from sub-batches are merged using the proposed recombination algorithms.

In the proposed launch point estimation example, an appropriate batch process combines all estimation data from the sub-batches [[24], [28]]. In the future, the novel batch algorithms developed in this dissertation are available to enhance estimation in which a batch algorithm works in conjunction with a recursive algorithm. For example, [31] describes conventional algorithms that combine batch and on-line (i.e., recursive) estimation processes in order to exploit the complementary nature of image and inertial sensing. The results show that one can use this combined estimation approach to recover accurate motion estimates, even in cases in which the optimal batch estimation from image measurements alone is not sufficient to recover accurate motion estimates. In related estimation applications in which bias and/or linearization induced truncation error are prevalent, the novel algorithms derived in this dissertation may further enhance overall estimation results, including those techniques that combine batch and recursive methods.

2.3.3 Adaptive Control Using Static System Identification with Intercept. Since the 1960s, research efforts have pursued the somewhat elusive goal of adaptive and reconfigurable flight control. A summary of on-line failure detection work includes a variety of methods, including those specific to a NASA F-8 testbed [37]. In Willsky's literature survey, Chien's formulation is discussed about appending the "monitoring system" directly to the feedback control system. The augmented controller does not disturb the system until the monitor detects the onset of bias within the system. There are drawbacks to using a fixed bias within a recursive formulation of this problem which induces a steady-state effect with failure onset. Chien proposes [37]

$$z(t) = x(t) + m \cdot \xi(t) + v(t)$$

in which

$$\xi(t) = \begin{cases} 0 & \text{for} \quad t < T \\ 1 & \text{for} \quad t \geq T \end{cases},$$

T is the time of failure, and m is an unknown constant. One should note that the measurement equation formulation proposed by Chen is quite similar to the augmentation of an intercept within the linear regression formulation used in this dissertation, as discussed in Chapters III and V. However, this thesis focuses on batch estimation processes instead of recursive approaches.

In the 1980s, the Air Force Wright Aeronautical Laboratory (AFWAL), in conjunction with McDonnel Douglas, General Electric, and Alphatech, undertook the Self Repairing Flight Control System (SRFCS) program, which included flight tests on a NASA F-15 [33]. After Failure Detection Indication (FDI) of control surface damage, an adaptive control algorithm calculated new control effectiveness parameters using a Kalman filter. The control mixer redistributed the forces and moments among the remaining healthy surfaces using a pseudo-inverse computation. This is significant to the research proposed in this thesis in that new control gains are computed using adaptive control algorithms. The control gain updates are reliant on the best information available for the system provided by a system identification algorithm. In addition, the AFWAL Control Reconfigurable Combat Aircraft (CRCA) program pursued model-based estimation for event driven Fault Detection and Isolation (FDI) by Grumman, Lear Astronics, and Charles River Analytics [8]. This model-based framework is incorporated into this dissertation research, as shown by the inner loop flight control system for the pitch channel in Figure 5.3.

As previously mentioned in Section 2.3.1, more recent research in the 1990s considers the benefits and potential applications of static system identification and batch parameter estimation, using linear regression techniques from statistics. Static system identification and batch parameter estimation are successfully used in [7] for the real-time identification of an aircraft's distributed control derivatives. A regularized minimum variance estimate of the stability and control derivatives is obtained at each $\triangle$T (i.e., time step) using a sliding data window. The parameter estimate information is used to adjust the flight control system's gains. When this work transitioned to industry in the mid-1990s, Barron Associates and Lockheed Martin flight tested an adaptive flight control system–the Self Designing Controller (SDC)–on the F-16 Variable Stability In-Flight Simulator Test Aircraft (VISTA) [1]. One such approach [34] considers a receding horizon optimal flight controller using an indirect, self-tuning regulator, in which an adaptive control law computes gains using parameters provided by real-time system identification. In [34], a modified sequential least squares algorithm with additional constraints is used in order to produce the parameter estimates. This research effort also pursues an indirect, self-tuning architecture in order to update the inner loop control gains of the flight control system. The research in this dissertation relies on a modified batch estimation algorithm to produce a parameter estimate that also allow for removing the trim change caused by failure.

In the late 1990s/2000 time frame, Lockheed Martin and Boeing pursued reconfigurable flight control work on fighter aircraft via the Air Force Research Laboratory's (AFRL) Reconfigurable Flight Control of Tailless Fighter Aircraft (RESTORE) program. The Lockheed Martin effort culminated in a real-time demonstration of the complete adaptive and reconfigurable flight control system [12]. This RESTORE effort also includes a modified sequential least squares (MSLS) algorithm for system identification that modifies the cost function for minimization and adds additional constraints on the permissible values of the parameters. Interestingly, this approach's ability to converge on the true parameters is dependent on good constraint selection. Thus, [12] states that this is the best that one could do using the aforementioned

approach, given the high levels of collinearity in this complex aircraft model. Also, the X-36 unmanned air vehicle (UAV) flight test of Boeing's reconfigurable flight control system (FCS) demonstrated the successful accommodation of a control effector failure [5]. This effort employs an on-line neural network to regulate the error in the plant inversion adaptively in order to overcome modelling uncertainties, failures, or damage. As in both of the RESTORE efforts, the research presented in this dissertation uses indirect adaptive control in order to reconfigure the flight control system adaptively. The RESTORE approaches differ from the methods proposed in this dissertation by the method chosen to identify the magnitude and impact of the control surface failure.

In this dissertation, indirect adaptive control for reconfigurable flight control is advanced by using a modified ILS algorithm for static system identification. This approach requires the measurement of the pitch acceleration $\dot{q}_m$, as shown in (2.2). Static system identification renders the parameter estimation problem (2.3) linear and then yields the estimate (2.4) and predicted estimation error covariance (2.5), as briefly mentioned in Chapter I. The physical aircraft's stability and control derivatives are directly estimated.

For reconfigurable flight control that accounts for control surface failure and trim change, this dissertation advocates moving away from recursive system identification and instead using a moving window/batch estimation and linear regression. The modified algorithm derived within Section 3.1 and summarized in Theorem 3 is required in order to gauge the magnitude of the change in trim caused by a control surface failure. The rigorous, closed-form solution to the parameter estimation algorithm gives an "unbiased" estimate and reduces to

$$\hat{\theta} = (H^T H)^{-1} H^T Z + \frac{1}{e^T [I - H(H^T H)^{-1} H^T] e} (H^T H)^{-1} H^T e e^T \left(\left[H (H^T H)^{-1} \right] H^T - I \right) Z$$

for the reconfigurable flight control problem discussed in Chapter V. The reader should recall that $\hat{\theta}$ is the parameter estimate, H is the regressor matrix, Z is the

measurement vector, I is an identity matrix, and e is an N-vector of ones. Also recall from Section 2.3.1 that this reduced form of Theorem 3 emerges because the equation error covariance R is homoscedastic. Hence,

$$R(\theta) = r(\theta)I$$

in which $r(\theta)$ is a parameter dependent scalar. As a result, the equation error covariance term cancels from the equation. Likewise, the parameter estimation error covariance reduces to

$$P_\theta = r(\hat{\theta}) \left[(H^T H)^{-1} + \frac{(H^T H)^{-1} H^T e e^T H (H^T H)^{-1}}{e^T \left[I - H(H^T H)^{-1} H^T \right] e} \right]$$

for the example in Chapter V. These equations effectively augment the linear regression with an intercept parameter.

In the flight control application, this dissertation advocates minimizing the use of integral action in a Type-1 inner loop control system and instead feed-forward a control signal derived from the intercept estimate in order to retrim the aircraft automatically. The reader should analyze the adaptive and reconfigurable flight controller block of the flight control system in Figure 5.9. The intent of using the static system identification of the stability and control derivatives, M_α and M_δ, respectively, is to produce the control gain adjustment that updates the inner loop control gains online. The adaptive and reconfigurable flight control system that relies on the modified linear regression technique during and after failure. In order to show the reader the benefits of the proposed flight control system compared to the conventional nonadaptive flight control system, consider the scenario presented in Chapter chap:5 in which a significant failure occurs to an F-16's horizonal stabilitor. The results are confined to the pitch axis and are shown for convenience. The proposed method improves pitch rate tracking significantly over the conventional, nonadaptive flight control system, as shown after $t = 5$ seconds in Figure 2.1. Moreover, the inclusion of an intercept

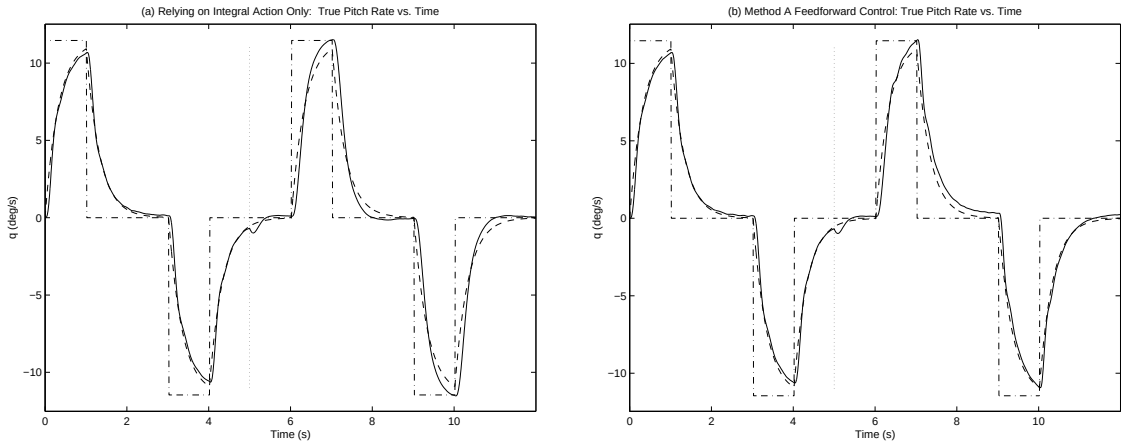


Figure 2.1: Pitch rate time history. Dash-dot lines (desired pitch rate command) Dashed lines (Prefiltered pitch rate command) Solid lines (Actual pitch rate) (a) Relies on conventional Type-1 inner loop controller for disturbance rejection (Same as Figure 5.26(a)) (b) Reconfigurable flight control that updates both K_P and K_α. (Same as Figure 5.33).

enhances the quality of the aircraft stability and control derivatives' estimates, as shown in Figure 5.34.

The research presented in this dissertation uses static system identification because it allows one to use the modified linear regression algorithm for parameter estimation. Moreover, the aircraft's physical parameters, that is, the stability and control derivatives, are directly obtained. This dissertation advocates using a *moving* window batch estimation process [9] instead of recursive estimation. A batch process using a moving window ensures that fresh data is used, so that changes in parameters are tracked and adaptive and reconfigurable control action is realized.

Furthermore, the inclusion of an intercept allows the flight control law to use "total" variables, rather than perturbation variables–"real world" flight control systems utilize total sensor measurements. While in state space form, the conventional perturbation dynamics are

$$\dot{x} = Ax + Bu \quad .$$

In practice, the total variables $X = \bar{X} + x$ and $U = \bar{U} + u$, are used. Here $\bar{X}$ is the trim state and $\bar{U}$ is the trim control. In terms of total variables, the trim equation is

$$A\bar{X} + B\bar{U} + C = 0$$

in which C is the change in trim. This equation yields the dynamics with an intercept

$$\begin{aligned} \dot{X} = \dot{x} &= Ax + Bu \\ &= A(X - \bar{X}) + B(U - \bar{U}) \\ &= AX + BU + C \quad . \end{aligned}$$

The inclusion of an intercept allows one to write the flight control laws using total variables, rather than perturbation variables. Since sensors measure total variables, it is convenient to use this sensor information directly in the inner loop flight control work.

It is also important to note that, under unfailed conditions, the inclusion of an intercept is not harmful to the parameter estimates [28]. In Section 5.4.3, the case in which no trim change occurs due to failure is considered. This occurs at flight conditions in which the aircraft's aerodynamic center is very close to the aircraft's center of gravity. Figure 5.19 shows that the modified linear regression algorithm estimate of the stability derivative M_α is not adversely impacted by the inclusion of the intercept c when compared to the estimate produced by the standard linear regression algorithm.

2.3.4 Bearings-Only Measurements. Chapter VI considers estimation techniques pertaining to limited measurement information. Oftentimes in RF emitter geo-location work and also in radar tracking, angle measurements are used in the following form

$$\phi \in \mathcal{N}\left(\phi_m, \sigma_\phi^2\right) \tag{2.8}$$

in which ϕ represents the azimuth angle or Angle of Arrival (AOA) and the subscript m indicates a measured quantity. This notation describes the AOA ϕ with a measured azimuth angle mean ϕ_m and variance σ_ϕ^2. AOA is defined as the angle measure between the direction from which the signal arrives at the receiver and some reference line [4].

A source of error [4] in AOA systems is the orientation of the antenna array relative to the angular reference. The accuracy of the calibration data is an additional source of angular error. According to [4], the error related to signal-to-noise ratio is the most significant error source that relates to the scenario described herein. As the signal strength decreases, the reduced signal-to-noise ratio causes variations in the measured AOA. Since Root Mean Square (RMS) error reduces the impact of a few large errors when most errors are small, it is usually accepted as the effective error of an emitter-location system [3]. By convention, the width of the error area is determined [3]

$$W = 2D \tan \sigma$$

in which W is the distance from the true angle vector to the RMS error vector, D is the distance from the site to the emitter, and σ is the RMS angle error. As typified in [3], the conventional approach views each AOA measurement as the angular error about the true azimuth. Multiple measurements are used to triangulate upon the measurements in order to determine an elliptical error probable about the emitter.

Oftentimes in practice, one measures the phasor, $\sin \phi$ and $\cos \phi$ directly. This assumption is the case if 1) syncros/resolvers are used in electro-mechanical control systems, or 2) if a phase angle is electronically measured and one has access to the I and Q signals, as is the case in RF receivers. In the latter case, and in Digital Signal Processing (DSP), one then refers to complex sine waveforms to represent the phasor pair.

In this dissertation, it is stipulated that the *sine* and *cosine* measurements of the bearing angle, or AOA, are directly available, which dramatically changes the nature of the estimation problem. The measurement of bearing is illustrated in Figure 2.2. This

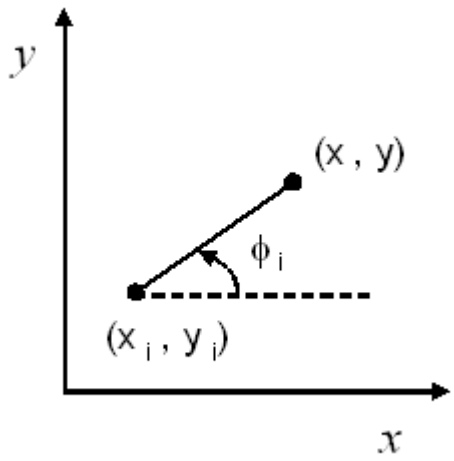


Figure 2.2: Bearing Measurement i. The emitter's position is (x, y). The ith receivers' positions is (x_i, y_i) and is known.

research assumes that the observer's GPS position is known $(x_i,\ y_i)$. The objective is to investigate the impact of AOA measurement modelling. Even though there is measurement error on the GPS coordinate position, this error is very small compared to the measurement error on the azimuth angle [4]. For example, Figure 6.3 reveals that, when considering a 5^o measurement error on ϕ and perfect observer position, all estimation methods discussed herein produce estimates with greater than 300 meter standard deviation error, as determined from the predicted equation error covariance for a batch size that includes 12 measurements. When the AOA measurement error is increased to 20^o and perfect observer position is assumed, the position estimates shown in Figure 6.8 include greater than 1000 meter standard deviation error, as determined from the predicted equation error covariance for the data window that includes 12 measurements. Hence, the impact of observer position GPS measurement error is negligible. Instead, this research focuses on the issues pertaining to AOA measurement.

The emitter's position is $(x,\ y)$. The standard RF emitter geo-location framework assumes using N receivers whose known coordinates, e.g., Global Positioning System (GPS) coordinates (x_i, y_i), that are recorded when the bearing measurements ϕ_{i_m} are taken, in which $i = 1, \ldots, N$. In the conventional approach, the angle measurement model shown in (2.8) is used.

As previously stated, it is postulated that a syncro's pair of AOA measurements, $(\sin\phi_i)_m$ and $(\cos\phi_i)_m$, are available. The two separate syncro measurements have Gaussian-distributed errors. This dissertation maintains that the proposed formulation better reflects the physics of the measurement situation. In turn, this formulation allows one to use efficient linear regression for parameter estimation; as a result, the need for linearization is obviated and the linearization-induced truncation error is eliminated. Improved emitter geo-location is obtained.

In order to provide a preview of the achieved reduction in lineraization-induced truncation error explored in Section 6.2, the measurement equation is written in a form used to represent a straight line in projective geometry

$$x\sin\phi_i + y\cos\phi_i = x_i\sin\phi_i + y_i\cos\phi_i \qquad , \qquad i = 1, \ldots, N \quad . \tag{2.9}$$

The approach that is advocated in Chapter VI exploits the access to the phasor measurements and reflects the true phasor plus a zero-mean, Gaussian-distributed measurement noise:

$$\begin{aligned} (\sin\phi_i)_m &= \sin\phi_i + v_i \quad , \quad v_i \in \mathcal{N}(0, \sigma^2) \\ (\cos\phi_i)_m &= \cos\phi_i + w_i \quad , \quad w_i \in \mathcal{N}(0, \sigma^2) \end{aligned} \tag{2.10}$$

By inserting (2.10) into (2.9), one can rearrange the measurement equations directly into the form of a linear regression, thus obviating the need for linearization:

$$(\sin\phi_i)_m\ x - (\cos\phi_i)_m\ y = x_i(\sin\phi_i)_m - y_i(\cos\phi_i)_m + (x - x_i)v_i - (y - y_i)w_i \tag{2.11}$$

in which $i = 1, \ldots, N$. Now one should compare the aforementioned development (2.11) to the conventional approach developed later in Section 6.6 which assumes that the measured azimuth angle is Gaussian-distributed so that

$$\phi_i = \phi_{m_i} + v_i, \qquad v_i \in \mathcal{N}(0, \sigma_\phi^2) \quad . \tag{2.12}$$

By inserting (2.12) into (2.9), one rearranges the measurement equation to form

$$\cos(\phi_{m_i} + v_i)y - \sin(\phi_{m_i} + v_i)x = y_i \cos(\phi_{m_i} + v_i) - x_i \sin(\phi_{m_i} + v_i) \quad .$$

Since $|v_i| \ll 1$, as shown in Table 6.1 in Section 6.7 and according to the ILS paradigm [36], one then linearizes each of the N equations about ϕ_{m_i}, neglecting higher order terms. Specifically,

$$\begin{aligned} \cos(\phi_{m_i})y - \sin(\phi_{m_i})x = y_i \cos(\phi_{m_i}) - x_i \sin(\phi_{m_i}) + sin(\phi_{m_i})y \cdot v_i + \\ \cos(\phi_{m_i})x \cdot v_i - y_i \sin(\phi_{m_i}) \cdot v_i - x_i \cos(\phi_{m_i}) \cdot v_i \quad . \end{aligned}$$

This equation is rearranged to fit the form of a linear regression $Z = H \cdot \theta + V$, that is,

$$x_i \sin(\phi_{m_i}) - y_i \cos(\phi_{m_i}) = \sin(\phi_{m_i})x - \cos(\phi_{m_i})y + [(x - x_i)\cos(\phi_{m_i}) + (y - y_i)\sin(\phi_{m_i})]\, v_i \; . \tag{2.13}$$

Hence, linearization-induced truncation error is reduced with the recommended approach (2.11) which acknowledges access to the measurement phasor instead of the conventional approach (2.13) that assumes a measured AOA.

2.4 Conclusions of Literature Review

This chapter first reviews some of the most common estimation techniques in use today. Given the nature of the applications considered in this dissertation, this research pursues an often overlooked estimation formulation using batch processes. In

the overview of batch estimation, it is emphasized that standard linear regression is inadequate to handle a change in trim and/or to overcome the effects of linearization-induced truncation error when the nonlinearity is strong. Novel approaches are required to address applications in projectile launch point determination, adaptive flight control, and emitter geo-location. Previous literature and overview material related to each of the application areas in this dissertation is discussed.

III. Linear Regression Technique

Two approaches are developed in this chapter that incorporate an intercept to the iterative least squares approach to estimation. Simulation results compare the estimation capability achieved by the two approaches to that produced by standard ILS.

3.1 Augmented Algorithm

In flight control applications, the presence of an intercept c in a linear or nonlinear regression could signal a failure or a change in trim. Estimating the parameter c enables automatic retrimming using feed-forward control, as opposed to feedback and integral action [28]. Moreover, a jump in the parameter estimate $\hat{c}$ signals the failure event. Hence, one should consider the generalized linear regression with an intercept c in (1.1). The linear regression in augmented form is called affine and is written as follows:

$$Z = \left[\begin{array}{c:c} H & e \end{array} \right] \left[\begin{array}{c} \theta \\ \cdots \\ c \end{array} \right] + V$$

with V of covariance R. This equation is rewritten as

$$Z = \mathbf{H}\boldsymbol{\theta} + V$$

in which the augmented regressor $\mathbf{H} = \left[\begin{array}{c:c} H & e \end{array} \right]$ is an $N \times (n+1)$ matrix and the augmented parameter $\boldsymbol{\theta} \triangleq (\theta, c)^T$ is $\boldsymbol{\theta} \in \mathbb{R}^{n+1}$. The minimum variance parameter estimate [36] is

$$\hat{\boldsymbol{\theta}} = (\mathbf{H}^T R^{-1} \mathbf{H})^{-1} \mathbf{H}^T R^{-1} Z \tag{3.1}$$

in which R is assumed invertible and the augmented filter-predicted estimation error covariance is the $(N+1) \times (N+1)$ matrix

$$P = (\mathbf{H}^T R^{-1} \mathbf{H})^{-1} \quad . \tag{3.2}$$

A closed-form solution of the linear regression with intercept is now developed.

First, the aforementioned linear regression is expanded and (3.1) and (3.2) yield

$$\begin{bmatrix} \hat{\theta} \\ \cdots \\ \hat{c} \end{bmatrix} = \left[\begin{bmatrix} H^T \\ \cdots \\ e^T \end{bmatrix} R^{-1} \begin{bmatrix} H & \vdots & e \end{bmatrix} \right]^{-1} \begin{bmatrix} H^T \\ \cdots \\ e^T \end{bmatrix} R^{-1} Z \tag{3.3}$$

and

$$P_{\theta,c} = \left[\begin{bmatrix} H^T \\ \cdots \\ e^T \end{bmatrix} R^{-1} \begin{bmatrix} H & \vdots & e \end{bmatrix} \right]^{-1} \quad . \tag{3.4}$$

It is determined that

$$\left[\begin{bmatrix} H^T \\ \cdots \\ e^T \end{bmatrix} R^{-1} \begin{bmatrix} H & \vdots & e \end{bmatrix} \right] = \begin{bmatrix} H^T R^{-1} H & H^T R^{-1} e \\ e^T R^{-1} H & e^T R^{-1} e \end{bmatrix} \quad .$$

The development requires an inverse of this matrix, as now shown.

Lemma 1 .

Let M_1, M_2, M_3, and M_4 be matrices such that $M_1 \in \mathbb{R}^{n \times n}$, $M_2 \in \mathbb{R}^{n \times m}$, $M_3 \in \mathbb{R}^{m \times n}$, $M_4 \in \mathbb{R}^{m \times m}$ and M_1, M_4 are invertible. One should consider the block matrix

$$M = \begin{bmatrix} M_1 & M_2 \\ M_3 & M_4 \end{bmatrix}$$

and assume that the relevant matrices are invertible. Then

$$M^{-1} = \begin{bmatrix} (M_1 - M_2 \cdot M_4^{-1} \cdot M_3)^{-1} & -M_1^{-1} \cdot M_2 (M_4 - M_3 \cdot M_1^{-1} \cdot M_2)^{-1} \\ -M_4^{-1} \cdot M_3 (M_1 - M_2 \cdot M_4^{-1} \cdot M_3)^{-1} & (M_4 - M_3 \cdot M_1^{-1} \cdot M_2)^{-1} \end{bmatrix} . \quad (3.5)$$

See Appendix A for the proof.

Hence, let

$$\left[\begin{bmatrix} H^T \\ \cdots \\ e^T \end{bmatrix} R^{-1} \begin{bmatrix} H & \vdots & e \end{bmatrix} \right]^{-1} = \begin{bmatrix} H^T R^{-1} H & H^T R^{-1} e \\ e^T R^{-1} H & e^T R^{-1} e \end{bmatrix}^{-1} \triangleq \begin{bmatrix} X & x \\ x^T & a \end{bmatrix} . \quad (3.6)$$

Using Lemma 1 and the definitions in (3.6):

$$X = (H^T R^{-1} H - H^T R^{-1} e \frac{1}{e^T R^{-1} e} e^T R^{-1} H)^{-1} \quad (3.7)$$

$$x = -a(H^T R^{-1} H)^{-1} H^T R^{-1} e \quad (3.8)$$

$$a = \frac{1}{e^T R^{-1} e - e^T R^{-1} H (H^T R^{-1} H)^{-1} H^T R^{-1} e} \quad . \quad (3.9)$$

The derivation also requires the Matrix Inversion Lemma.

Lemma 2 .

Let A, B, C, and D be matrices such that $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{m \times n}$, $D \in \mathbb{R}^{m \times m}$ and A, D are invertible. If $(A - BD^{-1}C)^{-1}$ exists, then

$$(A - BD^{-1}C)^{-1} = A^{-1} + A^{-1}B(D - CA^{-1}B)^{-1}CA^{-1} \quad . \quad (3.10)$$

The proof is given in Appendix B.

One can apply Lemma 2 to manipulate X in (3.7):

$$X=(H^TR^{-1}H)^{-1}+(H^TR^{-1}H)^{-1}H^TR^{-1}e\cdot$$
$$[e^TR^{-1}e-e^TR^{-1}H(H^TR^{-1}H)^{-1}H^TR^{-1}e]^{-1}\cdot e^TR^{-1}H(H^TR^{-1}H)^{-1} \quad .$$

One should note that the bracketed term in the preceding equation is previously defined as a in (3.9). Also, one should note that a is a scalar. This implies that

$$X \;=\; (H^TR^{-1}H)^{-1}+a(H^TR^{-1}H)^{-1}H^TR^{-1}ee^TR^{-1}H(H^TR^{-1}H)^{-1} \quad .$$

Finally, the definition for x is inserted into (3.8) to obtain

$$X=(H^TR^{-1}H)^{-1}+\frac{1}{a}xx^T \quad . \tag{3.11}$$

At this point, the explicit formulae for the parameter estimate $\hat{\theta}$ and the intercept estimate $\hat{c}$ is derived. Returning to (3.3), the results are combined with (3.6) to obtain the parameter estimate

$$\hat{\theta}=(XH^T+xe^T)R^{-1}Z \quad .$$

Equation (3.11) is incorporated into the above expression, which gives

$$\hat{\theta}=(H^TR^{-1}H)^{-1}H^TR^{-1}Z+x(\frac{1}{a}x^TH^T+e^T)R^{-1}Z \quad . \tag{3.12}$$

Equation (3.12) contains of the parameter estimate $\left(H^TR^{-1}H\right)^{-1}H^TR^{-1}Z$ that is obtained when the presence of the intercept is not acknowledged, plus a correction term due to the inclusion of the intercept, $x\left(\frac{1}{a}x^TH^T+e^T\right)R^{-1}Z$. The correction term is expanded in (3.12) by reinserting the scalar a (3.9) and the vector x (3.8) expressions, and the explicit result is obtained

$$\hat{\theta} \;=\; (H^TR^{-1}H)^{-1}H^TR^{-1}Z+\frac{1}{e^TR^{-1}\left[R-H(H^TR^{-1}H)^{-1}H^T\right]R^{-1}e}\cdot$$
$$(H^TR^{-1}H)^{-1}H^TR^{-1}ee^T\left(\left[R^{-1}H(H^TR^{-1}H)^{-1}\right]H^T-I\right)R^{-1}Z \quad .$$

The filter-predicted estimation error covariance is resolved from the explicit expansion in (3.3) and the definition in (3.6). Hence

$$\begin{aligned} P_\theta &= X \\ &= (H^T R^{-1} H)^{-1} + \frac{(H^T R^{-1} H)^{-1} H^T R^{-1} e e^T R^{-1} H (H^T R^{-1} H)^{-1}}{e^T R^{-1} \left[R - H(H^T R^{-1} H)^{-1} H^T\right] R^{-1} e} \quad . \end{aligned}$$

Similarly, (3.3) and (3.6) give the intercept estimate of

$$\hat{c} = (x^T H^T + a e^T) R^{-1} Z \quad . \tag{3.13}$$

By inserting (3.8) into the expression for $\hat{c}$ in (3.13),

$$\hat{c} = a \left[e^T - e^T R^{-1} H (H^T R^{-1} H)^{-1} H^T\right] R^{-1} Z$$

is calculated. This equation is rearranged further

$$\hat{c} = a e^T \left[I - R^{-1} H (H^T R^{-1} H)^{-1} H^T\right] R^{-1} Z$$

that is,

$$\hat{c} = a e^T R^{-1} \left[R - H (H^T R^{-1} H)^{-1} H^T\right] R^{-1} Z \quad .$$

Finally, the value of the scalar a from (3.9) is substituted, yielding the explicit intercept formula

$$\hat{c} = \frac{e^T R^{-1} \left[R - H (H^T R^{-1} H)^{-1} H^T\right] R^{-1} Z}{e^T R^{-1} \left[R - H (H^T R^{-1} H)^{-1} H^T\right] R^{-1} e} \quad .$$

The filter-predicted intercept estimation error standard deviation is also resolved from (3.4) and (3.6) such that

$$\sigma_c = a = \frac{1}{\sqrt{e^T R^{-1} \left[R - H (H^T R^{-1} H)^{-1} H^T\right] R^{-1} e}} \quad .$$

The derivation is summarized in Theorem 3.

Theorem 3 .

Assume the hypothesis R and H are matrices such that $R \in \mathbb{R}^{n\times n}$ and $H \in \mathbb{R}^{n\times m}$ Assume that Z and e are vectors such that $Z \in \mathbb{R}^{m\times 1}$, $e \in \mathbb{R}^{m\times 1}$ and R are invertible. If $\left(H^T R^{-1} H\right)^{-1}$ exists. The explicit closed-form solution to the linear regression with intercept (1.1) is as follows. The minimum variance parameter estimate is

$$\begin{aligned}\hat{\theta} &= (H^T R^{-1} H)^{-1} H^T R^{-1} Z + \frac{1}{e^T R^{-1}\left[R - H(H^T R^{-1} H)^{-1} H^T\right] R^{-1} e} \cdot \\ & \quad (H^T R^{-1} H)^{-1} H^T R^{-1} e e^T \left(\left[R^{-1} H (H^T R^{-1} H)^{-1}\right] H^T - I\right) R^{-1} Z\end{aligned}$$

with the filter-predicted estimation error covariance

$$P_\theta = (H^T R^{-1} H)^{-1} + \frac{(H^T R^{-1} H)^{-1} H^T R^{-1} e e^T R^{-1} H (H^T R^{-1} H)^{-1}}{e^T R^{-1}\left[R - H(H^T R^{-1} H)^{-1} H^T\right] R^{-1} e}$$

and the intercept estimate is

$$\hat{c} = \frac{e^T R^{-1}\left[R - H(H^T R^{-1} H)^{-1} H^T\right] R^{-1} Z}{e^T R^{-1}\left[R - H(H^T R^{-1} H)^{-1} H^T\right] R^{-1} e}$$

with the filter-predicted intercept estimation error standard deviation

$$\sigma_c = \frac{1}{\sqrt{e^T R^{-1}\left[R - H(H^T R^{-1} H)^{-1} H^T\right] R^{-1} e}}$$

The vector e is the N-vector of ones.

Corollary 4 .

The minimum variance intercept estimate

$$\hat{c} = 0$$

if and only if

$$e^T R^{-1}(Z - H\hat{\theta}) = 0$$

in which $(Z - H\hat{\theta})$ is the return difference.

One can consider the special case in which the equation error covariance is a scalar:

$$\begin{aligned} R &= \theta^2, \quad \text{a scalar} \\ a &= \frac{\theta^2}{N - e^T H(H^T H)^{-1} H^T e} \qquad (3.14) \\ x &= -\frac{\theta^2}{N - e^T H(H^T H)^{-1} H^T e}(H^T H)^{-1} H^T e \quad . \qquad (3.15) \end{aligned}$$

The parameter estimate becomes

$$\hat{\theta} = (H^T H)^{-1} H^T \{I - \tfrac{1}{N - e^T H(H^T H)^{-1} H^T e} \cdot e e^T [I - H(H^T H)^{-1} H^T]\} z \qquad (3.16)$$

and the intercept estimate reduces to

$$\hat{c} = \tfrac{1}{N - e^T H(H^T H)^{-1} H^T e} e^T [I - H(H^T H)^{-1} H^T]\} z \quad . \qquad (3.17)$$

These equations are derived in Appendix C.

3.2 Algorithm II: Removing the Averages

The linear regression with intercept (1.1) is now considered. One can calculate the averages $\bar{z}$ and $\bar{h}_i$ defined by

$$\begin{aligned} \bar{z} &\triangleq \frac{1}{N} e^T Z \\ \bar{h}_i &\triangleq \frac{1}{N} e^T h_i e \end{aligned}$$

in which

$$[h_i, \ldots, h_n] = H, \qquad h_i \in \mathbb{R}^N \quad .$$

The average measurement vector $\bar{Z}$ and average regressor matrix $\bar{H}$ are defined as

$$
\begin{aligned}
\bar{Z} &\triangleq e\bar{z} \\
&= \frac{1}{N} e e^T Z \\
\bar{H} &\triangleq [e\bar{h}_1, \ldots, e\bar{h}_n] \\
&= [e\frac{1}{N} e^T h_1, \ldots, e\frac{1}{N} e^T h_n] \\
&= \frac{1}{N} e e^T H \quad .
\end{aligned}
$$

One could assume that the data Z and linear transformation matrix H come from an experiment under the condition that a dynamic system is excited away from equilibrium. The data in Z and H represent state and control setting measurements. It is then plausible to assume the relationship

$$\bar{z} = (\bar{h}_1, \ldots, \bar{h}_n) \cdot \theta + c \quad . \tag{3.18}$$

This equation represents the dynamical system's trim condition (i.e., the dynamical system's equilibrium point). Hence, it is argued that confining one's attention to state and control perturbations away from trim will remove the presence of the intercept c from the measurement in (1.1).

Thus, the perturbation variables $\tilde{Z}$ and $\tilde{H}$ are defined as

$$
\begin{aligned}
\tilde{Z} &\triangleq Z - \bar{Z} \\
\tilde{H} &\triangleq H - \bar{H}
\end{aligned}
$$

The following relations are calculated:

$$
\begin{aligned}
\tilde{Z} &= M \cdot Z \\
\tilde{H} &= M \cdot H
\end{aligned}
$$

in which $M = I - \frac{1}{N} e e^T$ is an $N \times N$ matrix. One should note that M is symmetric and is positive semi-definite. Indeed,

$$\begin{aligned} x^T M x &= \| x \|_2^2 - \frac{1}{N}(x^T e) \cdot (e^T x) \\ &= \| x \|_2^2 - \frac{1}{N}(e^T x)^2 \end{aligned}$$

Now, by Schwarz's inequality [21]:

$$\begin{aligned} \left| e^T x \right| &\leq \| e \|_2 \cdot \| x \|_2 \\ &= \sqrt{N} \cdot \| x \|_2 \quad . \end{aligned}$$

This implies that

$$(e^T x)^2 \leq N \cdot \| x \|_2^2 \quad .$$

Equality holds if and only if $x = \alpha e$ for some $\alpha \in \mathbb{R}^1$. Hence,

$$x^T M x \geq 0$$

and, in fact

$$x^T M x > 0 \quad \forall x \neq 0, \quad \forall x \neq \alpha e \quad .$$

Therefore, the rank of M is $N - 1$. Hence, it is easy to see that the null space of M is one-dimensional and is spanned by the vector $v = e$. Indeed, one has

$$\begin{aligned} Me &= 0 \qquad \Leftrightarrow \\ (I - \frac{1}{N} e e^T) e &= 0 \qquad \Leftrightarrow \\ (e e^T) e &= N e \quad . \end{aligned}$$

Hence, in the sequel, one should consider the reduced linear regression

$$\tilde{Z} = \tilde{H} \cdot \theta + V \qquad , \qquad V \in \mathcal{N}(0, R) \quad . \tag{3.19}$$

The regressor $\tilde{H}$ is rank deficient. The linear regression (3.19) is obtained by subtracting (1.1) from (3.18).

Concerning the rank deficiency of $\tilde{H}$, one can diagonalize M

$$M = T^{-1}DT$$

in which D is the diagonal matrix

$$D = \begin{bmatrix} d_1 & 0 & \dots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & d_{N-1} & \vdots \\ 0 & \cdots & \cdots & 0 \end{bmatrix} \quad .$$

One should note that $d_i > 0$, for $i = 1, \dots, (N-1)$. Also, T is a N $\times$ N, non-singular transformation matrix.

Now, M_1 and M_2 are defined

$$M_1 \triangleq T^{-1} \cdot \begin{bmatrix} \sqrt{d_1} & 0 & \dots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & \sqrt{d_{N-1}} & \vdots \\ 0 & \cdots & \cdots & 0 \end{bmatrix}_{N \times (N-1)}$$

in which the rank of M_1 is $N-1$,

$$M_2 \triangleq \begin{bmatrix} \sqrt{d_1} & 0 & \dots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & \sqrt{d_{N-1}} & \vdots \\ 0 & \cdots & \cdots & 0 \end{bmatrix}_{(N-1) \times N} \cdot T$$

in which the rank of M_2 is $N-1$, and

$$\begin{aligned} K &\triangleq M_2 \cdot H \\ \theta_1 &\triangleq K \cdot \theta, \quad \theta_1 \in \mathbb{R}^{N-1} \quad . \end{aligned}$$

Now,

$$\tilde{Z} = M_1 M_2 H\theta + V \quad .$$

Next, one can solve the reduced full rank linear regression

$$\tilde{Z} = M_1 \cdot \theta_1 + V \quad .$$

This implies that θ_1 has mean $\hat{\theta}_1$ and filter-predicted equation error covariance P_{θ_1}, in which

$$\begin{aligned} \hat{\theta}_1 &= (M_1^T R^{-1} M_1)^{-1} M_1^T R^{-1} \tilde{Z} \\ P_{\theta_1} &= (M_1^T R^{-1} M_1)^{-1} \quad . \end{aligned}$$

Finally, the reduced full rank linear system of N-1 equations is solved as

$$\theta_1 = K\theta$$

which implies that

$$\begin{aligned} \theta &= (K^T K)^{-1} K^T \theta_1 \\ &= (H^T M_2^T M_2 H)^{-1} H^T M_2^T \theta_1 \quad . \end{aligned}$$

This equation implies that the parameter θ has mean $(K^T K)^{-1} K^T \theta_1$ and covariance $(K^T K)^{-1} K^T P_{\theta_1} K (K^T K)^{-1}$. Therefore, one can write the expanded form of the estimate's mean and covariance. One obtains the parameter that is uninfluenced by c. Hence, the minimum variance estimate is

$$\hat{\theta}_1 = (K^T K)^{-1} K^T (M_1^T R^{-1} M_1)^{-1} M_1^T R^{-1} \tilde{Z} \quad .$$

That is,

$$\hat{\theta}_1 = (H^T M_2^T M_2 H)^{-1} H^T M_2^T (M_1^T R^{-1} M_1)^{-1} M_1^T R^{-1} \tilde{Z}$$

and the filter-predicted estimation error covariance

$$P_\theta = (H^T M_2^T M_2 H)^{-1} H^T M_2^T \cdot (M_1^T R^{-1} M_1)^{-1} M_2 H (H^T M_2^T M_2 H)^{-1} \quad .$$

In summary, by removing the intercept c, the derivation loses one equation. At the same time, the remaining components of the parameter are estimated.

3.3 Comparing the Two Approaches

In this section, the two approaches for linear regression that handle an intercept, as developed in Sections 3.1 and 3.2, are compared. In order to demonstrate these ideas, the following example is discussed. A signal consisting of a single tone vibration and a DC component is measured. Noisy acceleration measurements are also available. The objective in this example is to estimate the vibration's frequency, amplitude, phase, and the DC component.

3.3.1 Analysis. One can consider the "clean measurement" relation for the single tone vibration with DC component. By clean, the author asks the reader momentarily to assume that the measurement is free of noise. This assumption will enable the author to relate geometric relation to the assumed truth kinematics first, before reinserting the noise corrupted measurements. Hence, the clean signal is

$$x(t) = A\cos(\omega t) + B\sin(\omega t) + c' \tag{3.20}$$

in which the signal's amplitude is

$$\mathcal{A} = \sqrt{A^2 + B^2},$$

its phase is

$$\phi = \begin{cases} \tan^{-1}(\frac{A}{B}) & \text{if} \quad B \neq 0 \\ \frac{\pi}{2} & \text{if} \quad B = 0, A \geq 0 \\ -\frac{\pi}{2} & \text{if} \quad B = 0, A < 0 \end{cases}$$

and the DC component is c'. Prony's method [17] is employed; thus, it is readily verifiable that $x(t)$ in (3.20) satisfies the differential equation

$$\ddot{x} + \omega^2 x = \omega^2 c' \quad .$$

The dynamics for the single tone vibration are

$$\ddot{x} + \omega^2 x = c \quad . \tag{3.21}$$

Assuming that $\dot{x}(0) = 0$, then $B = 0$. The solution to this ordinary differential equation is:

$$x(t) = \frac{c}{\omega^2} + A\cos(\omega t) \tag{3.22}$$

and

$$\ddot{x}(t) = -A\omega^2 \cos(\omega t) \quad . \tag{3.23}$$

(The differential equation solution details are shown in Appendix D.)

3.3.2 Stochastic Modelling. Equation (3.21) is rearranged as follows

$$\ddot{x} = -x\omega^2 + c \tag{3.24}$$

in which $c \triangleq \omega^2 c'$. Now, the parameter is defined $\theta \triangleq \omega^2$. Thus,

$$\ddot{x} = -x\theta + c \quad . \tag{3.25}$$

The measured signal and measured acceleration, as indicated by the subscript m, are

$$x_m(t) = x(t) + v_x(t)$$
$$\ddot{x}_m(t) = \ddot{x}(t) + v_{\ddot{x}}(t)$$

in which the measurement noise $v_x(t) \in \mathcal{N}(0, \sigma_x^2)$, $v_{\ddot{x}}(t) \in \mathcal{N}(0, \sigma_{\ddot{x}}^2)$, $\mathbf{E}[v_x(t)v_{\ddot{x}}(t)] = 0$, $\mathbf{E}[v_x(t)v_x(t + \Delta T)] = 0$, and $\mathbf{E}[v_{\ddot{x}}(t)v_{\ddot{x}}(t + \Delta T)] = 0$. By inserting (3.24) into (3.25), one obtains

$$\ddot{x}_m(t) = [-x_m(t)]\theta + c + v_{\ddot{x}} + v_x\omega^2 \quad .$$

By accumulating a data record over time, one can form the linear regression

$$Z = H\theta + ec + V$$

in which the measurement vector is

$$Z = \begin{pmatrix} \ddot{x}_m(0) \\ \vdots \\ \ddot{x}_m((N-1)\Delta T) \end{pmatrix}$$

the regressor is

$$H = - \begin{pmatrix} x_m(0) \\ \vdots \\ x_m((N-1)\Delta t) \end{pmatrix}$$

and the equation error is

$$V = \begin{pmatrix} v_{\ddot{x}}(0) + \omega^2 v_x(0) \\ \vdots \\ v_{\ddot{x}}((N-1)\Delta T) + \omega^2 v_x((N-1)\Delta T) \end{pmatrix} \quad .$$

Next, the equation error covariance matrix is calculated as

$$\begin{aligned} R &= \mathbf{E}[VV^T] \\ &= \mathbf{E}[(v_{\ddot{x}} + \theta v_x)(v_{\ddot{x}} + \theta v_x)^T] \quad . \end{aligned}$$

The noise of the acceleration and position measurements are uncorrelated. The measurement sets at each time increment are independent. Hence, the expectation for the entire data batch reduces to

$$\begin{aligned} R &= E[v_{\ddot{x}} v_{\ddot{x}}^T + \hat{\theta}^2 v_x v_x^T] \\ &= (\sigma_{\ddot{x}}^2 + \hat{\theta}^2 \sigma_x^2) \cdot I \quad . \end{aligned}$$

Thus, the equation error covariance matrix R is a scaled identity matrix. Because R is determined by an unknown parameter, it would appear that the parameter estimation requires iteration. Fortunately, this is not the case. The parameter dependence in the following equation is cancelled out because R is a diagonal matrix. Hence, the parameter estimate is

$$\hat{\boldsymbol{\theta}} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T Z$$

in which $\hat{\boldsymbol{\theta}} \triangleq \begin{pmatrix} \hat{\theta} \\ \cdots \\ \hat{c} \end{pmatrix}$ and $\mathbf{H} = \begin{bmatrix} H & \vdots & e \end{bmatrix}$. The augmented parameter estimate is determined without iterations. Finally, the filtered-predicted estimation error covariance is obtained directly as

$$P = (\mathbf{H}^T\mathbf{H})^{-1} \cdot (\sigma_{v_{\ddot{x}}} + \sigma_x\hat{\theta}^2) \cdot I$$

without iteration, in which $\hat{\theta^2} = \hat{\omega^4}$.

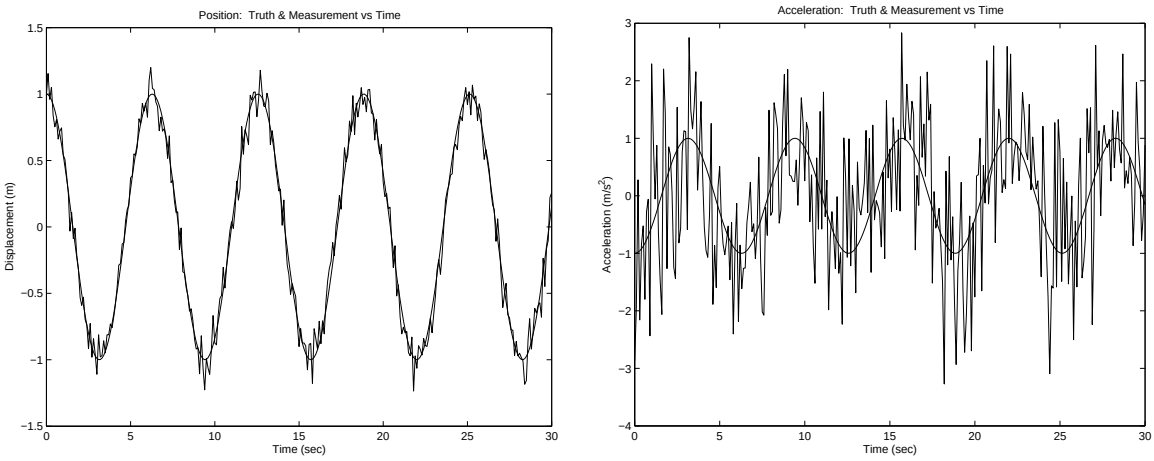
3.3.3 Simulation Experiment. In order to simplify the example so that the results clearly show the impact of the DC offset, this dissertation now considers the special case in which $B = 0$, so that $\mathcal{A} = A$ and $\phi = \frac{\pi}{2}$. The following variables are selected: $A = 1$, $c' = 1$, and $\omega = 1$ rad/s. With the same rationale as discussed in Section 3.3.1, a clean signal is used

$$x(t) = \cos(t) + 1 \quad .$$

The synthetic measurement vectors are now generated:

$$\begin{aligned} x_m(k\Delta T) &= \cos(k\Delta T) + v_x(k) \\ \ddot{x}_m(k\Delta T) &= -\cos(k\Delta T) + v_{\ddot{x}(k)}, \qquad k = 1, \ldots, N-1 \end{aligned} \tag{3.26}$$

under the condition that $v_x(k)$ and $v_{\ddot{x}}(k)$ are two separate Gaussian-distributed random sequences with variances σ_x^2 and $\sigma_{\ddot{x}}^2$, respectively, in which $\sigma_{\ddot{x}} \gg \sigma_x$. The sequences in (3.26) are used to form the Z and H vectors. Finally, having obtained c and ω, the



(a) Harmonic Oscillation. (b) Harmonic Acceleration Measurement.

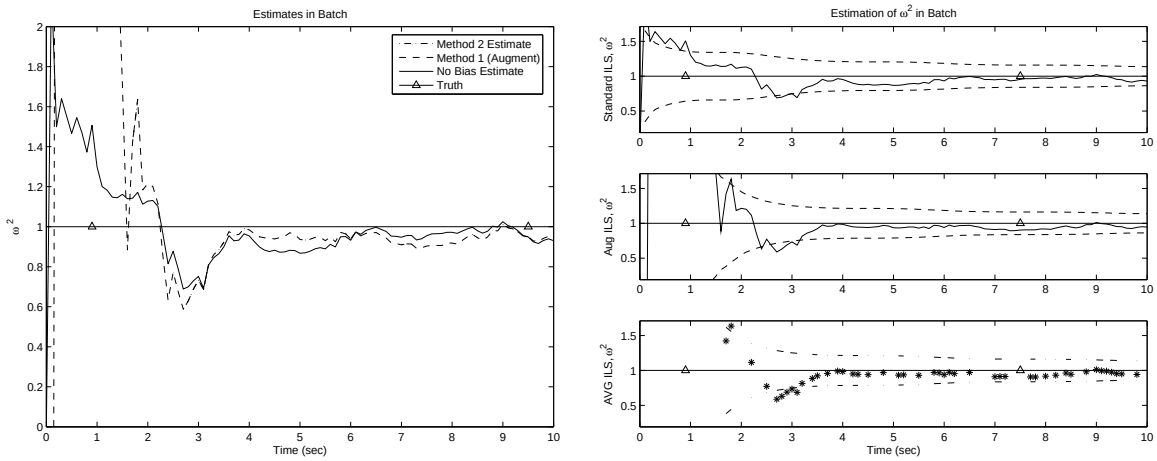
Figure 3.1: The Measurement Noise and Noise on Regressor are Depicted on Single Tone Vibration. No Bias Case. The true harmonic frequency is 1 rad/s. In (a), the noise on the regressor measurement is zero-mean, Gaussian-distributed with an error of 0.1 m. In (b), the measurement noise is zero-mean, Gaussian-distributed with an error of 1 m/sec^2.

DC component $\hat{c}' = \frac{\hat{c}}{\hat{\omega}^2}$ and the amplitude A are calculated by solving a least squares problem:

$$x_m(k\Delta T) = [\cos(k\Delta T)]A + v_x(k\Delta T), \quad k = 1, \ldots, N \quad .$$

The two linear regression methods are now compared. First, a single Monte Carlo baseline case is depicted in which no DC offset is included. The measurement and regressor are each sampled every tenth of a second, as shown in Figure 3.1. The sample noise is zero-mean, Gaussian-distributed. Identical noise scripts are used in each of three batch estimation experiments: standard linear regression and the linear regression with intercept methods described in Sections 3.1 and 3.2, respectively.

The results of the three batch estimation algorithms for a single Monte Carlo experiment are depicted in Figure 3.2. There are negligible differences between the harmonic frequency squared parameter estimates $\hat{\theta}$ after about 2 seconds (or 20 samples). In fact, the regression algorithm augmented with an intercept and the averaging regression algorithm produce estimates that overlay each other in Figure 3.2(a). All

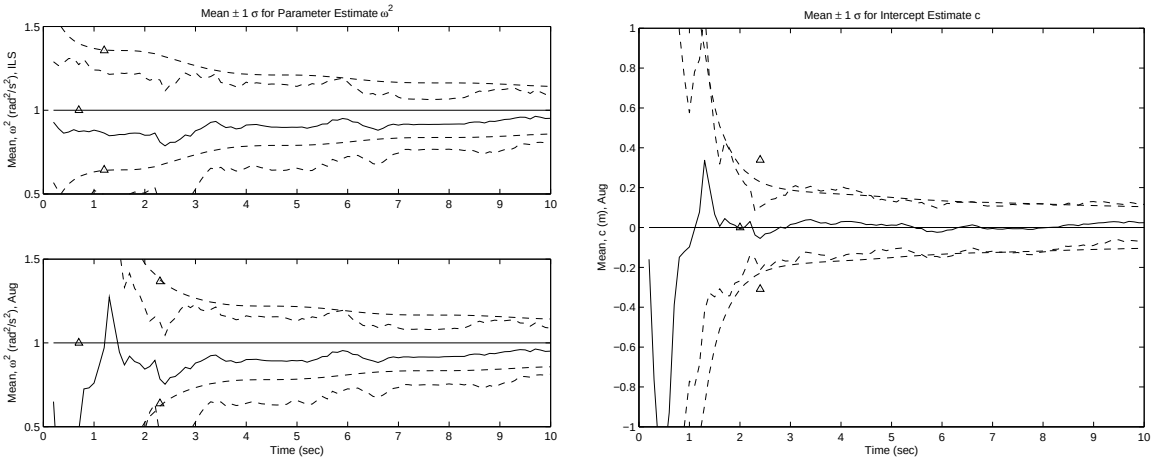


(a) (b)

Figure 3.2: No Bias Case. The parameter estimate is for the harmonic frequency squared $\hat{\omega^2}$. The measurement noise and the noise on the regressor are depicted in Figure 3.1. In (a), the true squared harmonic frequency is shown at 1 rad^2/sec^2. In (b), the dashed lines show ± 1 standard deviation from the truth parameter, as calculated via the filter-predicted equation error covariance matrices.

three techniques converge on the appropriate parameter value ω^2. The linear regression with augmented intercept method also identifies that the bias is 0. The averaging regression methodology encounters matrix singularity problems that are repressed for some batch sizes. This algorithm only converges at some window sizes, as depicted by the $*$s in the bottom plot of Figure 3.2(b). For a given batch size N available at some time $t = N\Delta t$, the lack of an $*$ indicates the inability of this estimation methodology to produce an estimate.

Additionally, Figure 3.3 shows estimation results for 10 Monte Carlo experiments using the standard linear regression and the linear regression augmented with an intercept. The averaging regression algorithm is omitted because of the aforementioned singularity problems. Again, the figure shows negligible differences between techniques after about 2 seconds (or 20 samples). This figure also shows that the standard deviation from the experimentally determined variance between the parameter estimates converges to the standard deviation from the predicted equation error co-



(a) (b)

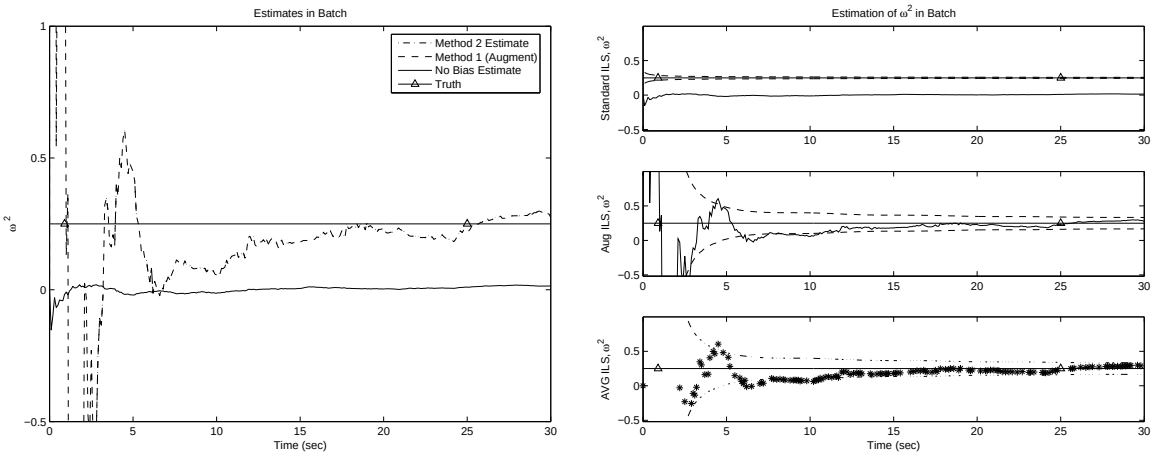
Figure 3.3: 10 Monte Carlo Simulations for No Bias Case. In (a), the parameter estimate is for the harmonic frequency squared $\hat{\omega^2}$ (solid lines) $\pm 1\sigma$ from the experimentally determined standard deviation of the estimates (dashed lines). The true squared harmonic frequency is shown at 1 rad^2/sec^2 (solid lines with $\triangle$s)$\pm 1\sigma$ from the filter-predicted equation error covariance (dashed lines with $\triangle$s). In (b), the intercept estimate $\hat{c}$ for the regression augmented with intercept frequency squared $\hat{\omega}^2$ (solid lines) $\pm 1\sigma$ from the experimentally determined standard deviation of the estimates (dashed lines). The true bias on the acceleration measurements is 0 (solid lines with $\triangle$s)$\pm 1\sigma$ from the filter-predicted equation error covariance (dashed lines with $\triangle$s).

variance. For completeness, Figure 3.3(b) shows that the intercept estimate $\hat{c}$ behaves as expected for the linear regression algorithm augmented with an intercept.

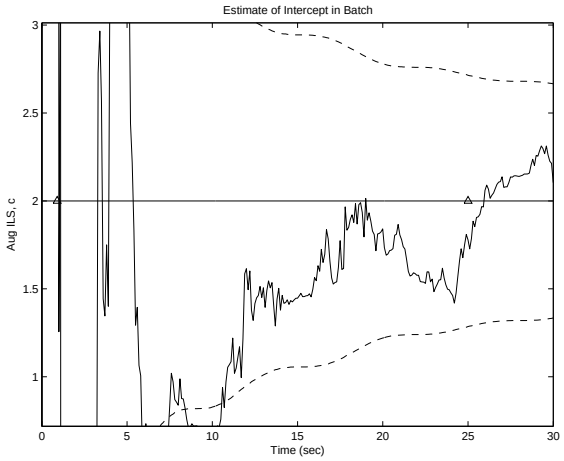
Next, a single tone vibration with significant DC offset is considered in a single Monte Carlo experiment. The harmonic frequency ω is reduced to 0.5 rad/s. A, as defined in (3.20), equals 1. Finally, the true bias c equals 2. The results of all three batch estimation algorithms for a single Monte Carlo experiment are depicted in Figure 3.4. First, the results show that the standard linear regression is oblivious to the bias. Hence, the parameter estimate is produced based on a faulty dynamics equation assumption. This estimate is not usable. The regression algorithm augmented with an intercept and averaging regression algorithm identify the parameter correctly after about 10 seconds (100 samples), in which $\omega^2 = 0.25\,\text{rad}^2/\,\text{sec}^2$. The averaging regression technique again experiences matrix singularity problems at certain batch sizes. Again, this algorithm only converges at some window sizes, as depicted by the $*$s in the bottom plot of Figure 3.4(b). AS before, the lack of an $*$ indicates the inability of this estimation method to produce an estimate for a given batch size N available at some time $t = N\Delta t$. Additionally, the large matrices involved in this batch process make this method computationally intensive. Finally, the augmented regression methodology correctly estimates the bias state at approximately 2 meters (Figure 3.4(c)).

Figure 3.5 shows estimation results for 10 Monte Carlo experiments using the standard linear regression and the linear regression augmented with an intercept. One should note that the true harmonic frequency $\omega^2 = 1$ and DC offset $c = 4$ depicted in Figure 3.5 are not the same as in the single Monte Carlo single tone vibration with DC offset depicted in Figure 3.4. Figure 3.5 is intended for comparisons with results in Section 3.4.

Clearly, Figure 3.5(a) shows that the estimates produced by the linear regression algorithm augmented with an intercept behave as expected even with a DC offset present. Again, the standard deviation from the experimentally determined variance

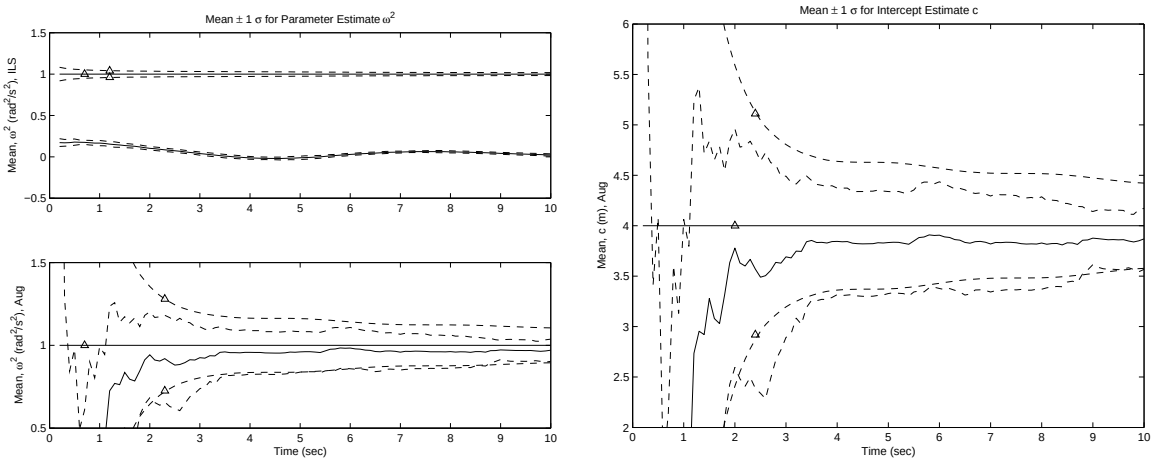


(a) (b)



(c) Estimate of Bias

Figure 3.4: Presence of Bias. The parameter estimate is for the harmonic frequency squared ω^2. In (a), the frequency squared is 0.25 rad^2/sec^2. The parameter estimates form the standard linear regression and the two linear regression methods with intercept are plotted for comparison. Clearly, the standard linear regression method fails to identify the parameter, while the derived algorithms that account for an intercept identify the harmonic frequency squared value. In (b), the same parameter estimates are plotted individually versus the truth harmonic frequency squared value. In addition, the dashed lines show ± 1 standard deviation from the truth parameter, as calculated via the filter-predicted equation error covariance. In (c), the bias in the truth model bias 2. The bias estimate is determined by regression augmented with an intercept (Method I, augmented algorithm).



(a) (b)

Figure 3.5: 10 Monte Carlo Simulations for single tone Vibration with DC offset. In (a), the parameter estimate is for the harmonic frequency squared $\hat{\omega}^2$ (solid lines) $\pm 1\sigma$ from the experimentally determined standard deviation of the estimates (dashed lines). The true squared harmonic frequency is shown at 1 rad^2/sec^2 (solid lines with $\triangle$s)$\pm 1\sigma$ from the filter-predicted equation error covariance (dashed lines with $\triangle$s). In (b), the intercept estimate $\hat{c}$ for the regression augmented with intercept frequency squared $\hat{\omega}^2$ (solid lines) $\pm 1\sigma$ from the experimentally determined standard deviation of the estimates (dashed lines). The true bias on the acceleration measurements is 0 (solid lines with $\triangle$s)$\pm 1\sigma$ from the filter-predicted equation error covariance (dashed lines with $\triangle$s).

between the parameter estimates converges to the standard deviation from the predicted equation error covariance. The standard linear regression technique fails to identify the parameter correctly in the presence of the DC offset. For completeness, Figure 3.5(b) shows that the intercept estimate $\hat{c}$ behaves as expected for the linear regression algorithm augmented with an intercept.

In summary, this subsection demonstrates the regression algorithms in the context of a single tone vibration example with DC offset. The linear regression algorithm augmented with an intercept and the algorithm that removes the averages both yield the correct solution to a linear regression with intercept. However, only the linear regression algorithm augmented with an intercept is useable in all cases. As one would suspect, the standard linear regression is oblivious to the presence of an intercept and a biased estimate is obtained.

3.4 Linear Regression with Only Linear Acceleration Measurements

One can consider the special case in which access to both acceleration and position measurements are available; however, the regression is formed using only the linear acceleration measurements. One should obtain superior estimates by incorporating all available measurement data at each time increment, especially when the position measurement is less noisy than acceleration. As discussed in Section 3.3, the key to using position measurements is to recognize from (3.22) that the position is susceptible to bias. Equation (3.23) shows that acceleration is inherently unbiased in this example.

One can form the linear regression

$$Z = H \cdot \theta + e \cdot c + V$$

in which the measurement vector is

$$Z = \begin{pmatrix} \ddot{x}_m(0) \\ \ddot{x}_m(1) \\ \vdots \\ \ddot{x}_m((N-1)\Delta T) \end{pmatrix}$$

The regressor is

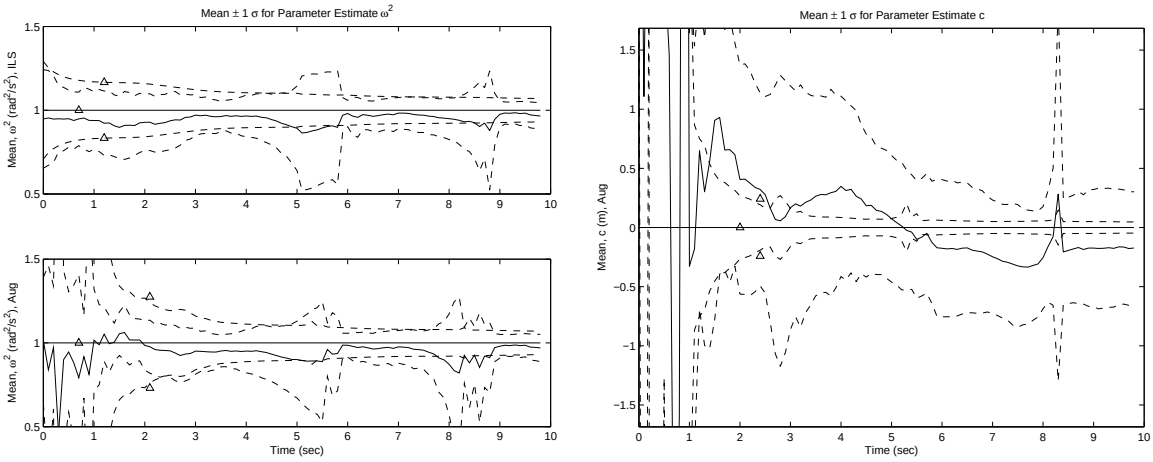
$$H = - \begin{pmatrix} \frac{c}{\theta_i} + A \cdot \cos(\sqrt{\theta_i} \cdot 0) \\ \frac{c}{\theta_i} + A \cdot \cos(\sqrt{\theta_i} \cdot 1) \\ \vdots \\ \frac{c}{\theta_i} + A \cdot \cos(\sqrt{\theta_i} \cdot (N-1)) \end{pmatrix}$$

and the equation error is

$$V = \begin{pmatrix} v_{\ddot{x}}(0) \\ v_{\ddot{x}}(1) \\ \vdots \\ v_{\ddot{x}}((N-1)\Delta T) \end{pmatrix} \quad .$$

The linear regression produces a non-parameter-dependent estimate covariance matrix R. This approach only incorporates one measurement for each time step, namely for the typical measurement vector Z. As a result, the parameter estimate produced by this methodology is not susceptible to intercept bias introduced in the position measurement. This method requires an initialization routine. One should assume momentarily that a "perfect" measurement vector is obtained. In other words, for the purpose of initialization, one can assume that $z_m = z_o$ and use fsolve.m in *Matlab* to solve for the least squares solution of

$$Z_m = -A \cdot \omega_t^2 \cos(\omega_t \cdot t) \quad .$$



(a) (b)

Figure 3.6: 10 Monte Carlo Simulations For Single Tone Vibration With DC Offset. Linear regression only incorporates the acceleration measurements and neglects biased position measurements. In (a), the parameter estimate is for the harmonic frequency squared $\hat{\omega}^2$ (solid lines) $\pm 1\sigma$ from the experimentally determined standard deviation of the estimates (dashed lines). The true squared harmonic frequency is shown at 1 rad^2/sec^2 (solid lines with $\triangle$s)$\pm 1\sigma$ from the filter-predicted equation error covariance (dashed lines with $\triangle$s). In (b), the intercept estimate $\hat{c}$ for the regression augmented with intercept frequency squared $\hat{\omega}^2$ (solid lines) $\pm 1\sigma$ from the experimentally determined standard deviation of the estimates (dashed lines). The true bias on the acceleration measurements is 0 (solid lines with $\triangle$s)$\pm 1\sigma$ from the filter-predicted equation error covariance (dashed lines with $\triangle$s).

This method always results in forming an unbiased estimate, but a trade-off is involved. First, one is not using all of the measurement information available. The estimation algorithm employed in this section is not using the additional position measurement information each time step. One should note that the neglected position measurement is the "cleaner" measured signal, as the measured variance for position σ_x^2 is much less than that for the measured variance for acceleration $\sigma_{\ddot{x}}^2$. Figure 3.6 clearly shows the degradation of the parameter estimates when relying only on the noisy acceleration measurements, when compared to the estimates produced by the linear regression algorithm augmented with an intercept shown in Figure 3.5. The algorithms used to produce Figures 3.5 and 3.6 are derived from the same 10 Monte Carlo experiments containing identical noise scripts. In fact, to make this point about

degradation perfectly clear, the acceleration measurements are only five times worse than the position measurements (i.e. $\sigma_x = 0.1$rad, $\sigma_{\ddot{x}} = 0.5\frac{rad}{s^2}$). This assumption is much more favorable than the assumptions about the acceleration measurement error in all previous Monte Carlo experiments. Hence, it is important to use all available measurement-types, especially the less noisy position estimates.

Second, this approach requires an iteration process, whereas one was not required before in Section 3.3. In the simple harmonic frequency example, this may seem relatively insignificant. However, this factor becomes much more significant as the magnitude of the problem increases, such as in estimation for an aircraft flight control system later explored in Chapter V.

3.5 Conclusion

Two estimation algorithms are developed in this chapter that acknowledge the presence of a change in trim to produce superior estimates compared to the estimate produced by standard linear regression. The regression algorithms are successfully demonstrated in a single tone vibration example in which a DC offset is included. Standard linear regression is oblivious to the presence of an intercept, and a biased estimate is obtained. The linear regression algorithm augmented with an intercept and the algorithm that removes the averages both yield the correct solution to a linear regression with intercept. The parameter is correctly identified. The algorithm that removes the averages can induce poor conditioning. Additionally, this method is computationally intensive. The linear regression method augmented with intercept also correctly estimates the magnitude of the DC offset. The algorithm that is developed for solving linear regressions with an intercept also performs well in those cases in which a DC offset is not present. Finally, simulation is used to demonstrate the value of using available position measurements in the regressor H in addition to acceleration measurements in the measurement vector Z.

IV. Parameter Space Augmentation for Nonlinear System Identification

Parameter identification using batch estimation is pursued using nonlinear regression augmented with an intercept in order to address nonlinearity effects. The intercept parameter is constrained to balance the linearization-induced truncation error with measurement-noise-induced equation error. When compared to conventional Iterative Least Squares (ILS), the novel algorithm reduces the estimation error covariance and expands the envelope of measurement geometries in which good estimation is possible despite strong nonlinearity. Additionally, estimation performance does not degrade from standard ILS when nonlinearity strength is weak.

4.1 Nonlinear Regression

The nonlinear regression is considered

$$Z = h(\theta) + V \quad , \quad V \in \mathcal{N}(0, R) \tag{4.1}$$

in which $Z \in \mathbb{R}^N$ is the measurement vector, $\theta \in \mathbb{R}^n$ is the parameter, $N \geq n$, and h is the nonlinear function of the parameter θ. The equation error $V \in \mathbb{R}^N$ is zero-mean, Gaussian-distributed equation error with covariance R, which is an $N \times N$ real, symmetric positive definite matrix. An iterative parameter estimation process is employed. One can suppose that a preliminary parameter estimate $\hat{\theta}^{(k)}$ is available. The function $h(\theta)$ is assumed continously differentiable and expanded about the current parameter estimate $\hat{\theta}^{(k)}$:

$$\begin{aligned} h(\theta) &= h(\hat{\theta}^{(k)} + \theta - \hat{\theta}^{(k)}) \\ &= h(\hat{\theta}^{(k)}) + \frac{\partial h}{\partial \theta}(\hat{\theta}^{(k)}) \cdot (\theta - \hat{\theta}^{(k)}) + r \end{aligned}$$

in which $\hat{\theta}^{(i)}$ is the current parameter estimate and the N-vector residual r is the truncation error, which is included explicitly in the derivation. Hence,

$$h(\theta) = h(\hat{\theta}^{(k)}) + \frac{\partial h}{\partial \theta}(\hat{\theta}^{(k)}) \cdot \theta - \frac{\partial h}{\partial \theta}(\hat{\theta}^{(k)}) \cdot \hat{\theta}^{(k)} + r \quad .$$

Inserting this expansion into (4.1) yields

$$Z = h(\hat{\theta}^{(k)}) + \frac{\partial h}{\partial \theta}(\hat{\theta}^{(k)}) \cdot \theta - \frac{\partial h}{\partial \theta}(\hat{\theta}^{(k)})\hat{\theta}^{(k)} + r + V \quad .$$

This equation implies

$$Z + H_k \cdot \hat{\theta}^{(k)} - h(\hat{\theta}^{(k)}) = H_k \cdot \theta + r + V \tag{4.2}$$

in which the $N \times n$ regressor matrix is

$$H_k \triangleq \frac{\partial h}{\partial \theta}(\hat{\theta}^{(k)}) \quad .$$

Using an intercept in nonlinear regression is meant to account for the truncation error r caused by linearization. It is claimed that the estimates benefit by accounting for the presence of a strong nonlinearity using an intercept c [22]. This claim is now examined in detail.

In the "conventional" approach, an Iterative Least Squares algorithm is used to obtain the parameter estimate. The presence of truncation error, r, is ignored, so Equation (4.2) simplifies to

$$Z + H_k \cdot \hat{\theta}^{(k)} - h(\hat{\theta}^{(k)}) = H_k \cdot \theta + V$$

and iterates as in [36]:

$$\hat{\theta}^{(k+1)} = (H_k^T R^{-1} H_k)^{-1} H_k^T R^{-1} [Z + H_k \hat{\theta}^{(k)} - h(\hat{\theta}^{(k)})] \tag{4.3}$$

Also, one can recall that at step k, H_k is $\hat{\theta}^{(k)}$-dependent. At the instant of convergence, in which $\hat{\theta}^{(k)} \rightarrow \hat{\theta}$ as $i \rightarrow \infty$, one finally calculates the predicted parameter estimation error covariance

$$P = (H_{\infty}^{T} R^{-1} H_{\infty})^{-1}$$

in which H_{∞} describes the limit of the regressor.

In fact, the preceding iteration (4.3) is akin to a Gauss-Newton algorithm. In actual standard ILS [15], one "sets" $H_k \cdot \hat{\theta}^{(k)} = h(\hat{\theta}^{(k)})$; that is, one uses the simplified iteration

$$\hat{\theta}^{(k+1)} = \left(H_k^T R^{-1} H_k\right)^{-1} H_k^T R^{-1} Z \quad .$$

It is fair to say that standard, or conventional, ILS perform very well, for example, in the GPS algorithm, in which the geometry is very good.

Now, one should consider a modified ILS algorithm–in fact, a Gauss Newton algorithm approach to (4.1) that now acknowledges the presence of truncation error, r. The intercept c is defined as

$$c \triangleq \frac{1}{N} \sum_{i=1}^{N} r_i$$

as are fluctuations w_i in the residual r_i

$$w_i \triangleq r_i - c, \qquad i = 1, \ \ldots, \ N \quad .$$

Then

$$h(\theta) = h(\hat{\theta}^{(k)}) + \frac{\partial h}{\partial \theta}|_{\theta=\theta^{(k)}} \cdot (\theta - \hat{\theta}^{(k)}) + e \cdot c + W$$

in which e is a vector of ones and

$$W \triangleq \begin{pmatrix} w_1 \\ \vdots \\ w_N \end{pmatrix} \quad .$$

Assuming that the fluctuations w_i in the residual r about their average c are random, and making a 1st order ergodic assumption, W is considered a zero-mean random vector in this dissertation. The 1st order ergodic assumption is to justify the complete description of the fluctuation w_i using the first two moments, which is probably not accurate. However, this assumption enables one to eventually use Theorem 3, as is discussed shortly.

The tuning parameter $q^{(k)}$ are scalars that quantify the strength of the fluctuations in nonlinearity at the kth iteration step. Thus, the tuning parameter $q^{(k)}$ sequence is defined as

$$q^{(k)} > 0 \tag{4.4}$$

such that

$$\lim_{k \to \infty} q^{(k)} = 0$$

for $k = 1, 2, \ldots$. In addition, the derivation stipulates that $W \in \mathcal{N}(0, q^{(k)} \cdot I_N)$. . One should note that the equation error has increased to $W + V \triangleq \tilde{V}$, such that W and V are independent. Hence, the equation error covariance matrix is augmented as follows:

$$R^{(k)} = R + q^{(k)} \cdot I_N \quad . \tag{4.5}$$

The magnitude of $q^{(k)}$ is related to the distance of the current parameter estimate $\hat{\theta}^{(k)}$ from true θ.

In summary of the modified ILS algorithm, the presence of the nonlinearity is acknowledged. The strength of nonlinearity manifests itself through the remainder's average value c, and the magnitude of its fluctuations about c, which is captured by the tuning parameter $q^{(k)}$. As one approaches convergence, the magnitude of the step away from the point of linearization decreases. Therefore, one should notice that the sequence $q^{(k)} \to 0$ as $k \to \infty$ and, as discussed in Section 4.2, c is made commensurate with the measurement-noise-induced equation error (a.k.a. augmented ILS with constrained intercepts).

Finally, (4.1) is rewritten as

$$Z + H_k \cdot \hat{\theta}^{(k)} - h(\hat{\theta}^{(k)}) = H_k \cdot \theta \cdot + e \cdot c + W + V \tag{4.6}$$

in which V, W are independent. Rearranging (4.6) and using the vector of ones notation e, the reformulated linear regression is obtained at the kth iteration step

$$\begin{aligned} Z + H_k \cdot \hat{\theta}^{(k)} - h(\hat{\theta}^{(k)}) &= (\, H_k \vdots e\,) \begin{pmatrix} \theta \\ \dots \\ c \end{pmatrix} + \tilde{V} \\ &= \boldsymbol{H}_k \cdot \boldsymbol{\theta} + \tilde{V} \end{aligned} \tag{4.7}$$

in which the equation error becomes $\tilde{V} \triangleq W + V$. The solution of the linear regression (4.7) augmented with the intercept is

$$\hat{\boldsymbol{\theta}}^{(k+1)} = (\mathbf{H}_k^T R^{-1} \mathbf{H}_k)^{-1} \mathbf{H}_k^T R^{-1} [Z + H_k \hat{\theta}^{(k)} - h(\hat{\theta}^{(k)})] \tag{4.8}$$

in which $k = 0, 1, \cdots, N$. (See Section 3.1.) The expansion of (4.8) is given in Appendix E. As k increases and $\|\hat{\boldsymbol{\theta}}^{(k+1)} - \hat{\boldsymbol{\theta}}^{(k)}\|$ decreases, the tuning parameter $q^{(k)}$ is reduced. After convergence is achieved (i.e., when $\hat{\boldsymbol{\theta}}^{(k)} \to \hat{\boldsymbol{\theta}}$, $\mathbf{H}_k \to \mathbf{H}_\infty$, and $R_k \to R_\infty$ as $k \to \infty$), one can verify that the intercept estimate $\hat{c}$ is small. Even in the absence of an intercept, the new augmented ILS algorithm proposed in this section is not the conventional ILS algorithm

$$\hat{\theta}^{(k+1)} = \left(H_k^T R^{-1} H_k\right)^{-1} H_k^T R^{-1} Z \quad .$$

Explicitly stated, the measurement vector shown in (4.8) includes the correction terms in the measurement vector, namely $H_k\hat{\theta}^{(k)}$ and $h(\hat{\theta}^{(k)})$. For large values of k, the parameter $\hat{\theta}$ does not change much. This observation is in stark contrast to the initial iterations. Thus, initially, the effects of the nonlinearity are pronounced (i.e., the truncation error $c \neq 0$). Hence, in order to assist convergence for the nonlinear regression with intercept case, the block diagonal terms of the covariance R in (4.8) were doubled for early iterations. As the iteration continues for 12 iterations, the extra uncertainty weighting is gradually reduced to 1. Although this implementation procedure is ad hoc, it helps to facilitate convergence of the iterative least squares algorithm while addressing strong nonlinearities. After convergence is achieved (i.e., when $\boldsymbol{\theta}^{(k)} \rightarrow \boldsymbol{\theta}$ as $k \rightarrow \infty$), one obtains the filter-predicted estimation error covariance

$$P = (\mathbf{H}_\infty^T R_\infty^{-1} \mathbf{H}_\infty)^{-1} \quad . \tag{4.9}$$

The significant factors that influence the quality of the parameter estimate are measurement arrangement geometry and the duration of the measurement interval, which is reflected with the condition number of the regressor H.

One should consider the development of the intercept c in (4.7) more closely. If one composes a nonlinear regression based on measurements over time using one nonlinear measurement equation only, then the intercept c is a scalar and e consists of an N vector of ones. Hence, when setting up a nonlinear regression formed from scalar measurements recorded over time, one can think of an average truncation error via the intercept c, and some additional random error:

$$\begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_N \end{bmatrix}_{N\times 1} = c \cdot e + W \tag{4.10}$$

as in (4.6). When m nonlinear measurement equations are used, in which $m > 1$, one can include up to m separate intercepts within the parameter $c \in \mathbb{R}^m$. This is significant because one can account for each residual r_i introduced by nonlinearity within a each measurement equation.

4.2 Balancing the Linearization

When an intercept parameter c is introduced into a nonlinear regression to address the truncation effects of linearization, this research uses the following adaptation of the Ridge Regression [[13], [14]] concept from statistics. A general observation from working with nonlinear applications; the unconstrained intercepts within the iterative least squares algorithm from converging to the correct solution as problem complexity increased. The Ridge Regression constraints enable the intercepts to obtain an appropriate magnitude. During the augmented ILS iterations, the derivation requires that, at the point of convergence, the residual truncation error due to the nonlinearity is commensurate with the equation-error-induced measurement noise. Hence, the derivation balances the linearization-induced truncation error and the measurement-noise-induced equation error by requiring

$$|c| \cong \sigma \tag{4.11}$$

in which σ quantifies the intensity of the measurement noise. The latter is reflected in the equation error covariance $R = \sigma^2 \tilde{R}$, in which $\tilde{R}$ is the nondimensional equation error covariance. The requirement expressed in (4.11) is enforced by including prior information on the intercept, that is, the derivation stipulates that

$$c \in \mathcal{N}(0, \sigma^2) \quad . \tag{4.12}$$

Thus, for every intercept component, an additional fictitious measurement equation is introduced as follows

$$0 = (0, \ldots, 0, 1)\boldsymbol{\theta} + \xi \tag{4.13}$$

in which $\xi \in \mathcal{N}(0, \sigma^2)$, $\tilde{V}$ is the equation error in the regression (4.7), and $\mathrm{E}(\tilde{V}\xi)$= 0. Thus, the parameter space is augmented with additional constraints equal to the number of measurements at each time step. One can recall that the parameter is now augmented as

$$\boldsymbol{\theta} = \begin{pmatrix} \theta \\ \ldots \\ c \end{pmatrix} \quad .$$

Two implementation approaches to the previously introduced constraint are now considered. The first method entails augmenting the constraints directly to the regression equation (4.7). The measurement vector Z and regressor matrix H formerly used in (4.8) are augmented according to the tuning parameter discussed in (4.4)

$$\begin{aligned} \boldsymbol{Z_R} &= \begin{pmatrix} Z + H_k\hat{\theta}^{(k)} - h(\hat{\theta}^{(k)}) \\ 0_{(M\times 1)} \end{pmatrix}_{(M\cdot N+M)\times 1} \\ \boldsymbol{H_R} &= \begin{pmatrix} \boldsymbol{H}_{(M\cdot N\times(n+M))} & & \\ \ldots & \ldots & \ldots \\ 0_{(M\times n)} & \vdots & I_{(M\times M)} \end{pmatrix} \end{aligned} \tag{4.14}$$

and are used in the batch estimation iteration. One should note that M is the number of nonlinear measurement equations potentially requiring an intercept, N is the batch size, and n indicates the number of parameters, excluding the intercepts. In order to determine the new equation error covariance matrix $\mathbf{R_R}$, one uses (4.5) to calculate the expectation

$$\mathbf{R_R} = \mathbf{E}\left(\begin{pmatrix} \tilde{V} \\ \xi \end{pmatrix} \cdot \begin{pmatrix} \tilde{V} & \xi \end{pmatrix} \right) \tag{4.15}$$
$$= \begin{bmatrix} R_{M\cdot N \times M\cdot N} & & 0_{M\cdot N \times M} \\ \cdots & \cdots & \cdots \\ 0_{M\times M\cdot N} & & R_{\text{ridge}_{(M\times M)}} \end{bmatrix}$$

in which R_{ridge} is determined by the measurement noises associated with the specifics of the problem. This framework lacks the flexibility of enabling the user to apply the intercept constraint at the time of his choosing within the iteration process. From an implementation standpoint, a framework that permits this flexibility is desirable.

An alternative method of augmenting the intercept constraints into the nonlinear regression is now described. First, the two-step algorithm determines the estimate $\hat{\boldsymbol{\theta}}$ and its corresponding predicted estimation error covariance matrix $\mathbf{P} = \begin{bmatrix} P_\theta & P_{\theta\, c} \\ P_{c\,\theta} & P_c \end{bmatrix}$, by assuming an unconstrained intercept using (4.8) with the estimation error covariance (4.9). One should note that the P_θ portion of $\mathbf{P}$ corresponds to the parameter portion of the filter-predicted equation error covariance matrix and is of interest here. Next, a linear regression formed using the aforementioned estimate (4.8), covariance matrix (4.9), and intercept constraint (4.12) is considered:

$$\begin{bmatrix} \hat{\boldsymbol{\theta}} \\ 0 \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \hat{\theta} \\ \hat{c} \end{bmatrix} \\ 0 \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \theta \\ c \end{bmatrix} \\ c \end{bmatrix} + \begin{bmatrix} \tilde{V} \\ \\ \xi \end{bmatrix} \tag{4.16}$$
$$\boldsymbol{Z_R} = \qquad\qquad \boldsymbol{H_R} \;+\; \boldsymbol{V_R} \qquad , \qquad \boldsymbol{V_R} \in \mathcal{N}(0, \boldsymbol{R_R})\,.$$

This final linear regression is then solved using the measurement vector $\boldsymbol{Z_R}$, regressor matrix $\boldsymbol{H_R}$, and equation error covariance $\boldsymbol{R_R}$:

$$\boldsymbol{Z_R} = \begin{pmatrix} \hat{\theta} \\ \hat{c} \\ 0 \end{pmatrix}, \qquad \boldsymbol{H_R} = \begin{bmatrix} I_{(n+M),(n+M)} \\ 0_{(M\times n)} \vdots I_{(M\times M)} \end{bmatrix}$$

$$\boldsymbol{R_R} = \begin{bmatrix} P_{\theta(n\times n)} & 0_{(n\times M)} \\ 0_{(M\times n)} & \text{Diag}\left(\begin{bmatrix} \sigma_1^2 \\ \vdots \\ \sigma_M^2 \end{bmatrix}\right) \end{bmatrix}$$

Thus, the final parameter estimate and predicted parameter estimation error covariance are obtained. The intercept augmented linear regression using Ridge Regression constraints is solvable in one-step, but the aforementioned two-step process provides additional flexibility during implementation to help ensure convergence in a strongly nonlinear problem, such as the first example presented in Section 4.5. For a given batch size, the user determines an appropriate number of iterations that perform the linear regression calculations shown in (4.8) and (4.9). Then, one applies the Ridge Regression constraints appended in (4.16) after an appropriate number of iterations. This method is preferable to a one-step Ridge Regression implementation, since experience shows that it increases the percentage of instances of convergence to the correct parameter estimate in strongly nonlinear cases [23]. For cases in which the nonlinearity's impact is less significant, as that presented in the second example in Section 4.5, Ridge Regression yields the correct parameter estimate regardless of whether a one-step or two-step implementation is used.

One can now choose to implement the intercept constraints embedded in (4.14) and (4.15) at any point during the iteration process. As previously stated, the two-step method provides the flexibility of enabling the user to apply the intercept constraint at the time of his choosing within the iteration process. This flexibility of implementation expands the envelope of successful estimation for these batch processes. In addition,

individual estimates approach the true parameter values closer than those estimates achieved by the one-step method. If one wants to match the result obtained by the one-step method, one can incorporate the constraints immediately following the first iteration of (4.8) in the two-step method, and continue to apply the intercept constraint throughout the iteration process for a given batch size. The cons of using a two-step method versus the one-step method are insignificant. The difference in computational requirements for implementation is negligible.

4.3 Ballistic Trajectory Tracking

The novel nonlinear parameter estimation process is demonstrated in the context of a ballistic trajectory determination from battlefield radar measurements [2]. The objective is to track enemy projectiles and geo-locate the enemy firing position. The developed methodology is also directly applicable to satellite trajectory determination [36].

The ballistic trajectory is modelled by using a kinematic model [24] typically used in fire control systems:

$$\begin{aligned} x(t) &= x_o + V_{x_o} t \\ y(t) &= y_o + V_{y_o} t \\ z(t) &= z_o + V_{z_o} t - \frac{1}{2} c_b g t^2 \end{aligned} \tag{4.17}$$

in which x_o, y_o, and z_o are the unknown launch point coordinates, V_{x_o}, V_{y_o}, and V_{z_o} are the projectile's muzzle velocity components in the corresponding Cartesian direction, c_b is the known ballistic coefficient of the projectile, and t is time. Since the example assumes to know the type of projectile that the enemy is firing, c_b is assumed known in this development. One should have interest in the parameter vector

$$\theta = (x_o, y_o, z_o, V_{x_o}, V_{y_o}, V_{z_o})^T \in \mathbb{R}^6 \quad . \tag{4.18}$$

The measurement situation at hand is carefully modelled and nonlinear regression is used to estimate the parameter θ in order to obtain the launch position.

Radar measurements are comprised of range R, azimuth ϕ, and elevation θ. Without loss of generality, one can assume that the radar is located at the origin, as shown in Figure 4.1. Thus, perfect noise-free data satisfies the nonlinear measurement

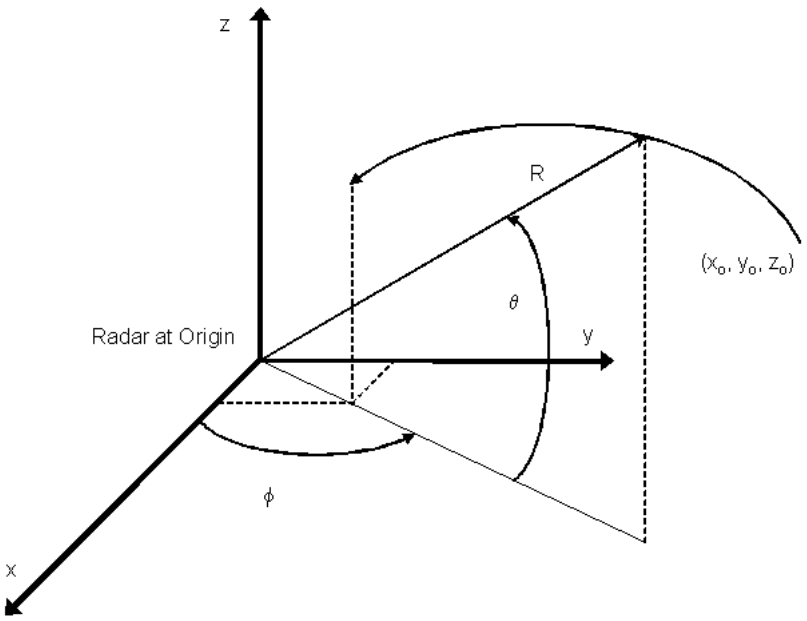


Figure 4.1: Ballistic Tracking Commences at (x_o, y_o, z_o).

equations for $t \geq 0$:

$$\begin{aligned} R(t) &= \sqrt{x^2(t) + y^2(t) + z^2(t)} \\ \tan(\phi(t)) &= \frac{y(t)}{x(t)} \qquad \text{for } \; x(t) \neq 0 \\ R(t)\sin(\theta(t)) &= z(t) \end{aligned} \tag{4.19}$$

The projectile's launch point is not known; hence, this is the estimation objective of the nonlinear regression. During the iteration and upon convergence, the strength of the nonlinearity in h is reduced using this formulation.

Table 4.1: Summary of Nondimensional Variables.

$x_o \rightarrow \frac{x_o}{V_o T}$	$y_o \rightarrow \frac{y_o}{V_o T}$	$z_o \rightarrow \frac{z_o}{V_o T}$	$R \rightarrow \frac{R}{V_o T}$
$v_R \rightarrow \frac{v_R}{V_o T}$	$\sigma_R \rightarrow \frac{\sigma_R}{V_o T}$	$V_{x_o} \rightarrow \frac{V_{x_o}}{V_o}$	$V_{y_o} \rightarrow \frac{V_{y_o}}{V_o}$
$V_{z_o} \rightarrow \frac{V_{z_o}}{V_o}$	$\Delta T \rightarrow \frac{\Delta T}{T}$	$t \rightarrow \frac{t}{T}$	$c_b \rightarrow c_b \frac{gT}{V_o}$

The trajectory is sampled. At this point in this chapter, all variables are nondimensionalized for this example with combinations of the following scaling factors, as appropriate: T is the duration of the measurement interval and V_o is the muzzle velocity of the projectile. Since the simulation assumes to know the type of projectile fired at the friendly target, one can use the nominal V_o for nondimensionalization. Nondimensionalization in estimation is important, as it allows one to gauge the conditioning of the regression objectively. Reasons for using nondimensionalization for this projectile launch point application are discussed in Section 2.3.2. Because *Matlab* is selected to perform the simulations, computational precision becomes an issue when parameter numerical values range more than 10^6 in dimensional units. Table 4.1 shows the nondimensional variables.

Considering (4.17) and (4.19), this implies that at time $t = k\Delta T$:

$$
\begin{aligned}
&R(k \cdot \Delta T) = \sqrt{(x_o + V_{x_o} k\Delta T)^2 + (y_o + V_{y_o} k\Delta T)^2 + (z_o + V_{z_o} k\Delta T - \tfrac{1}{2} c_b k^2 \Delta T^2)^2} \\
&\tan\left(\phi(k \cdot \Delta T)\right) = \frac{y_o + V_{y_o} k\Delta T}{x_o + V_{x_o} k\Delta T} \\
&R(k\Delta T) \cdot \sin\left(\theta(k \cdot \Delta T)\right) = z_o + V_{z_o} k\Delta T - \tfrac{1}{2} c_b k^2 \Delta T^2
\end{aligned}
\tag{4.20}
$$

in which $k = 0, 1, \ldots, N-1$. The measurement record is

$$
\begin{aligned}
R_m(k \cdot \Delta T) &= R(k \cdot \Delta T) + v_R(k) \\
\phi_m(k \cdot \Delta T) &= \phi(k \cdot \Delta T) + v_\phi(k) \\
\theta_m(k \cdot \Delta T) &= \theta(k \cdot \Delta T) + v_\theta(k)
\end{aligned}
\tag{4.21}
$$

in which the subscript m denotes the measured variables and v_R, v_ϕ, and v_θ are additive Gaussian white noise quantified by the radar's measurement uncertainties' standard deviations σ_R, σ_ϕ, and σ_θ respectively. The derivation carefully models the measurement situation at hand and uses nonlinear regression to estimate the parameter, θ, and to obtain the projectile launch position. Finally, in order to acquire a fix on projectile launch position, knowledge of launch point elevation is assumed known via contour map in the example. An iterative "outer loop" process that accounts for the possible elevations could enable the algorithms to obtain a reasonable estimate of the launch point coordinates, with some uncertainty bound. The character of the terrain might indicate how large of a bomb one might choose to respond, as flat terrain would not impact the uncertainty of the launch point coordinates much whereas hilly terrain would increase the level of uncertainty.

The radar measurement equations (4.21) are rearranged and inserted into (4.20) in order to obtain

$$
\begin{aligned}
&R_m(k\cdot\Delta T) - v_R(k) = \sqrt{(x_o + V_{x_o}k\Delta T)^2 + (y_o + V_{y_o}k\Delta T)^2 + (z_o + V_{z_o}k\Delta T - \tfrac{1}{2}c_b k^2\Delta T^2)^2} \qquad (4.22)\\
&\tan\left(\phi_m(k\cdot\Delta T) - v_\phi(k)\right) = \frac{y_o + V_{y_o}k\Delta T}{x_o + V_{x_o}k\Delta T}\\
&\left(R_m(k\cdot\Delta T) - v_R(k)\right)\cdot\sin\left(\theta_m(k\cdot\Delta T) - v_\theta(k)\right) = z_o + V_{z_o}k\Delta T - \frac{1}{2}c_b k^2\Delta T^2
\end{aligned}
$$

in which, $k = 1, \ldots, N$. The reader should note that the left-hand side of (4.22) is a nonphysical mathematical construction that provides a convenient form to pursue the components of (4.2) pertaining to the observation relation. The equations are left in this form, as they are now ready for linearization.

4.4 Nonlinear Regression

In this section, the nonlinear regression (4.1) is composed in which the solution renders the projectile's launch point coordinates. The generalized ILS algorithms developed in Sections 4.1 and 4.2 are exercised.

4.4.1 Measurements. In this ballistic trajectory problem, the measurement vector is composed as follows

$$Z_{N\times 1}^{(R)} \triangleq \begin{pmatrix} R_m(0) \\ R_m(\Delta T) \\ \vdots \\ R_m((N-1)\Delta T) \end{pmatrix}$$

$$Z_{N\times 1}^{(\phi)} \triangleq \begin{pmatrix} \tan\left(\phi_m(0)\right) \\ \tan\left(\phi_m(\Delta T)\right) \\ \vdots \\ \tan\left(\phi_m((N-1)\Delta T)\right) \end{pmatrix}$$

$$Z_{N\times 1}^{(R,\theta)} \triangleq \begin{pmatrix} R_m(0)\cdot\sin\left(\theta_m(0)\right) \\ R_m(\Delta T)\cdot\sin\left(\theta_m(\Delta T)\right) \\ \vdots \\ R_m((N-1)\Delta T)\cdot\sin\left(\theta_m((N-1)(\Delta T))\right) \end{pmatrix}$$

and finally

$$Z_{(3N)\times 1} \triangleq \begin{pmatrix} Z^{(R)} \\ Z^{(\phi)} \\ Z^{(R,\theta)} \end{pmatrix} \quad .$$

One can note that the third component of the measurement vector $Z^{(R,\theta)}$ reflects regression formulation described in (4.19) and(4.20).

4.4.2 Equation Error. Concerning the equation error associated with the nonlinear measurement equations (4.20), let

$$V_{N\times 1}^{(R)} \triangleq \begin{pmatrix} v_R(0) \\ v_R(\Delta T) \\ \vdots \\ v_R((N-1)\Delta T) \end{pmatrix} \tag{4.23}$$

$$V_{N\times 1}^{(\phi)} \triangleq \begin{pmatrix} \frac{1}{1+\phi_m^2(0)} v_\phi(0) \\ \frac{1}{1+\phi_m^2(\Delta T)} v_\phi(\Delta T) \\ \vdots \\ \frac{1}{1+\phi_m^2((N-1)\Delta T)} v_\phi((N-1)\Delta T) \end{pmatrix}$$

$$V_{N\times 1}^{(R,\theta)} \triangleq \begin{pmatrix} \sin(\theta_m(0)) \cdot v_r(0) + R_m(0)\cos(\theta_m(0)) \cdot v_\theta(0) \\ \sin(\theta_m(\Delta T)) \cdot v_r(\Delta T) + R_m(\Delta T)\cos(\theta_m(\Delta T)) \cdot v_\theta(\Delta T) \\ \vdots \\ \sin(\theta_m(N-1)\Delta T) \cdot v_r((N-1)\Delta T) + R_m((N-1)\Delta T)\cos(\theta_m(N-1)(\Delta T)) \cdot v_\theta((N-1)\Delta T) \end{pmatrix}$$

in which $v_R(k\Delta T)$ is the range measurement error at time $k\Delta T$, $v_\phi(k\Delta T)$ is the azimuth measurement error at time $k\Delta T$, and $v_\theta(k\Delta T)$ is the elevation measurement error at time $k\Delta T$. These components yield the equation error vector

$$V_{(3N)\times 1} \triangleq \begin{pmatrix} V^{(R)}(\theta) \\ V^{(\phi)}(\theta) \\ V^{(R,\theta)}(\theta) \end{pmatrix} \quad .$$

Again, one can note that the third component of the measurement equation error vector $V^{(R,\theta)}$ reflects the impact of the regression formulation described in (4.19) and(4.20). The explicit nonlinear regression is obtained in terms of the parameter stated in (4.18), which consists of $3N$ equations for (4.7).

Next, the measurement error vector $v_{(3N\times 1)}$ is now considered. One should note that $v_{R_{N\times 1}}$ is already specified in (4.23), as $V^{(R)}_{N\times 1} = v_{R_{N\times 1}}$. In addition,

$$v_{\phi_{N\times 1}} \triangleq \begin{pmatrix} v_\phi(0) \\ v_\phi(\Delta T) \\ \vdots \\ v_\phi((N-1)\Delta T) \end{pmatrix}$$

and

$$v_{\theta_{N\times 1}} \triangleq \begin{pmatrix} v_\theta(0) \\ v_\theta(\Delta T) \\ \vdots \\ v_\theta((N-1)\Delta T) \end{pmatrix} \quad .$$

Therefore, the measurement error vector is

$$v_{(3N)\times 1} \triangleq \begin{pmatrix} v_{RN\times 1} \\ v_{\phi N\times 1} \\ v_{\theta N\times 1} \end{pmatrix} \quad .$$

Finally, the $N \times N$ diagonal matrices are defined:

$$\begin{aligned} D_N^{(\phi)} &\triangleq \mathrm{Diag}\left(\frac{1}{1+\phi_m^2(k\Delta T)}\right)_{k=0}^{(N-1)} \\ D_N^{(E)} &\triangleq \mathrm{Diag}\left(\sin\left(\theta_m k\Delta T\right)\right)_{k=0}^{(N-1)} \\ D_N^{(R,E)} &\triangleq \mathrm{Diag}\left(R_m(k\Delta T)\cdot\cos\left(\theta_m k\Delta T\right)\right)_{k=0}^{(N-1)} \end{aligned} \tag{4.24}$$

in which the subscript m denotes measured variables. Next, the $(3N)\times(3N)$ matrix is composed

$$\Gamma_{(3N)\times(3N)} \triangleq \begin{bmatrix} I_N & 0 & 0 \\ 0 & D_N^{(\phi)} & 0 \\ D_N^{(E)} & 0 & D_N^{(R,E)} \end{bmatrix}$$

and one can write the reformulated linear regression (4.1) in the form

$$Z = h(\theta) + \Gamma v$$

in which the equation error $V = \Gamma v$.

In this reformulated linear regression, the equation error covariance is determined

$$R = \mathbf{E}(V \cdot V^T) = \mathbf{E}(\Gamma v \cdot (\Gamma v)^T) \quad .$$

Hence, one can calculate

$$\begin{aligned} R = \mathbf{E}(& \begin{pmatrix} v_R \\ D_N^{(\phi)} \cdot v_\phi \\ D_N^{(E)} \cdot v_R + D_N^{(R,E)} \cdot v_\theta \end{pmatrix} \cdot \begin{pmatrix} v_R & D_N^{(\phi)} \cdot v_\phi & D_N^{(E)} \cdot v_R + D_N^{(R,E)} \cdot v_\theta \end{pmatrix}) \\ = & \begin{bmatrix} R(1,1) & R(1,2) & R(1,3) \\ R(2,1) & R(2,2) & R(2,3) \\ R(3,1) & R(3,2) & R(3,3) \end{bmatrix} \quad . \end{aligned}$$

The equation error covariance R is explicitly given by a nondiagonal matrix:

$$
\begin{aligned}
R(1,1) &= \mathbf{E}[v_R v_R^T] \\
R(1,2) &= D_N^{\phi}\mathbf{E}[v_R v_\phi^T] \\
R(1,3) &= D_N^{E}\mathbf{E}[v_R v_R^T] + D_N^{(R,E)}\mathbf{E}[v_R v_\theta^T] \\
R(2,1) &= D_N^{(\phi)}\mathbf{E}[v_\phi v_R^T] \\
R(2,2) &= {D_N^{(\phi)}}^2\mathbf{E}[v_\phi v_\phi^T] \\
R(2,3) &= D_N^{(\phi)}D_N^{(E)}\mathbf{E}[v_\phi v_R^T] + D_N^{(\phi)}D_N^{(R,E)}\mathbf{E}[v_\phi v_\theta^T] \\
R(3,1) &= D_N^{(E)}\mathbf{E}[v_R v_R^T] + D_N^{(R,E)}\mathbf{E}[v_\theta v_R^T] \\
R(3,2) &= D_N^{(E)}D_N^{(\phi)}\mathbf{E}[v_R v_\phi^T] + D_N^{(R,E)}D_N^{(\phi)}\mathbf{E}[v_\theta v_\phi^T]
\end{aligned} \tag{4.25}
$$

$$
\begin{aligned}
R(3,3) = \quad & {D_N^{(E)}}^2\mathbf{E}[v_R v_R^T] + D_N^{(R,E)}D_N^{(E)}\mathbf{E}[v_\theta v_R^T] \\
& + D_N^{(E)}D_N^{(R,E)}\mathbf{E}[v_R v_\theta^T] + {D_N^{(R,E)}}^2\mathbf{E}[v_\theta v_\theta^T]
\end{aligned}
$$

One should note that

$$
\begin{aligned}
\mathbf{E}[v_R v_R^T] &= \sigma_R^2 \cdot I_N \\
\mathbf{E}[v_\phi v_\phi^T] &= \sigma_\phi^2 \cdot I_N \\
\mathbf{E}[v_\theta v_\theta^T] &= \sigma_\theta^2 \cdot I_N \quad .
\end{aligned}
$$

Assuming that range, azimuth, and elevation measurement noises are independent, all other expectations shown in (4.25) are equal to zero. Hence, the equation error covariance R reduces to

$$
R = \begin{bmatrix}
\sigma_R^2 \cdot I_N & 0 & \sigma_R^2 \cdot D_N^{(E)} \\
0 & \sigma_\phi^2 \cdot D^{(\phi)^2} & 0 \\
\sigma_R^2 \cdot D_N^{(E)} & 0 & \sigma_R^2 \cdot D_N^{(E)^2} + \sigma_\theta^2 \cdot D_N^{(R,E)^2}
\end{bmatrix} \quad .
$$

Since R is symmetric, one only needs to calculate $R(1,1)$, $R(1,2)$, $R(1,3)$, $R(2,2)$, $R(2,3)$, and $R(3,3)$.

Identical noise realizations are used to compare the performance of the standard ILS algorithm with the augmented nonlinear regression with intercept. As mentioned near the end of Section 4.1, near the point of convergence, the parameter $\hat{\theta}$ does not change much. This is in stark contrast to the initial iterations. Thus, initially, the effects of the nonlinearity are pronounced (i.e., the truncation error $c \neq 0$). Figure 4.2 yields insights into the iteration process used to obtain the parameter estimate, $\hat{x}_o$. The asterisks $*$s show the final iteration values at selected window sizes for a standard

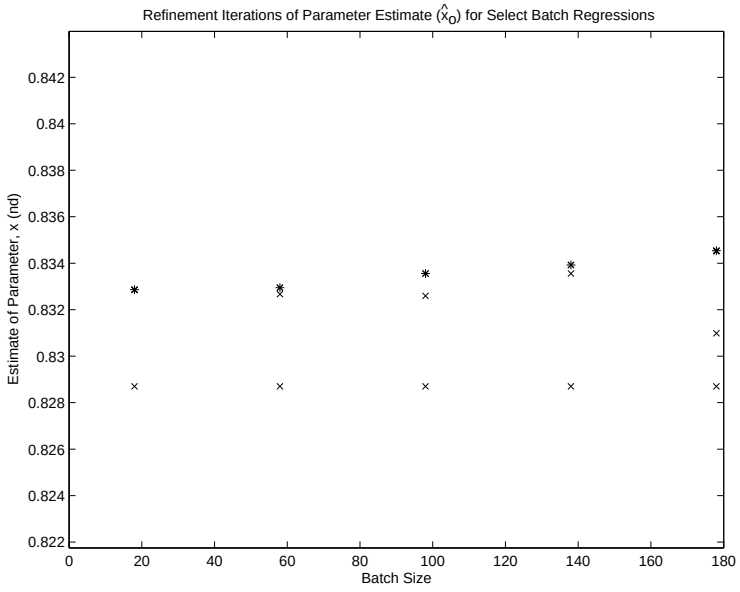


Figure 4.2: Refinement Iteration Process of Parameter Estimate for Selected Batch Estimation Size. The asterisks $*$s denote the final iteration and the $\times$s denote are intermediate iteration estimates. One should note that the standard ILS algorithm typically converges after 3 iterations in this Monte Carlo run.

ILS algorithm Monte Carlo run. As the figure shows, the standard ILS algorithm typically achieves solution convergence after four iterations in the example. However, experience with these algorithms shows that the modified ILS algorithm requires more iterations in order to converge upon an estimate. In order to assist the convergence of the nonlinear regression with intercept, the presence of the nonlinearity/fluctuating intercept is acknowledged. As previously mentioned at the end of Section 4.1, the

block diagonal terms of the covariance R in (4.26) are doubled for early iterations. As the iteration proceeds, the extra uncertainty weighting is gradually reduced to one after 12 iterations. As stated in Section 4.1, this ad hoc implementation procedure helps to facilitate convergence of the iterative least squares algorithm while addressing strong nonlinearities.

4.4.3 Nonlinear Observation Function. Next, the nonlinear observation function h is formed. The vectors are defined

$$h^{(R)}(\theta) \triangleq \begin{pmatrix} \sqrt{x_o^2 + y_o^2 + z_o^2} \\ \sqrt{(x_o + V_{x_o}\Delta T)^2 + (y_o + V_{y_o}\Delta T)^2 + (z_o + V_{z_o}\Delta T - \frac{1}{2}c_b\Delta T^2)^2} \\ \vdots \\ \sqrt{(x_o + V_{x_o}(N-1)\Delta T)^2 + (y_o + V_{y_o}(N-1)\Delta T)^2 + (z_o + V_{z_o}(N-1)\Delta T - \frac{1}{2}c_b\Delta T^2(N-1)^2)^2} \end{pmatrix}$$

$$h^{(\phi)}(\theta) \triangleq \begin{pmatrix} \frac{y_o}{x_o} \\ \frac{y_o + V_{y_o}\Delta T}{x_o + V_{x_o}\Delta T} \\ \vdots \\ \frac{y_o + V_{y_o}\Delta T(N-1)}{x_o + V_{x_o}\Delta T(N-1)} \end{pmatrix}$$

$$h^{(R,\theta)}(\theta) \triangleq \begin{pmatrix} z_o \\ z_o + V_{z_o}\Delta T - \frac{1}{2}c_b\Delta T^2 \\ \vdots \\ z_o + V_{z_o}\Delta T(N-1) - \frac{1}{2}c_b\Delta T^2(N-1)^2 \end{pmatrix}$$

and

$$h(\theta)_{(3N)\times 1} \triangleq \begin{pmatrix} h^{(R)}(\theta) \\ h^{(\phi)}(\theta) \\ h^{(R,\theta)}(\theta) \end{pmatrix} \quad .$$

Again, one can note that the third component of the nonlinear observation relation $h^{(R,\theta)}$ reflects the impact of the regression formulation described in (4.19) and(4.20).

4.4.4 Linearization. The observation equation $h(\theta)$ is now linearized in this sub-section. One can suppose that $\hat{\theta}^{(k)}$ is the current parameter estimate at step k. The observation relation is linearized about $\hat{\theta}^{(k)}$. Let

$$H_k \triangleq \frac{\partial h}{\partial \theta}|_{\theta=\hat{\theta}^{(k)}} \quad .$$

In view of (4.8), the regressor matrix H_k is composed as follows:

$$H_k = \begin{bmatrix} \frac{\partial h^{(R)}(\theta)}{\partial \theta}|_{\theta=\hat{\theta}(k)} \\ \frac{\partial h^{(\phi)}(\theta)}{\partial \theta}|_{\theta=\hat{\theta}(k)} \\ \frac{\partial h^{(R,\theta)}(\theta)}{\partial \theta}|_{\theta=\hat{\theta}(k)} \end{bmatrix}$$

in which the parameter vector is specified in (4.18).

First, one should consider the components that make up $\frac{\partial h^{(R)}(\theta)}{\partial \theta}|_{\theta=\hat{\theta}(k)}$:

$$H(1,1) = \frac{\partial h^{(R)}}{\partial x_o} = \frac{x_o}{\sqrt{(x_o^2 + y_o^2 + z_o^2)}}$$
$$H(1,2) = \frac{\partial h^{(R)}}{\partial y_o} = \frac{y_o}{\sqrt{(x_o^2 + y_o^2 + z_o^2)}}$$
$$H(1,3) = \frac{\partial h^{(R)}}{\partial z_o} = \frac{z_o}{\sqrt{(x_o^2 + y_o^2 + z_o^2)}}$$

in which $H(1,4) = \frac{\partial h^{(R)}}{\partial V_{x_o}} = 0$, $H(1,5) = \frac{\partial h^{(R)}}{\partial V_{y_o}} = 0$, and $H(1,6) = \frac{\partial h^{(R)}}{\partial V_{z_o}} = 0$. For all additional range measurements after the initial measurement, the following equations explicitly yield the linearized regressor:

$$
\begin{aligned}
H(2:N,1) &= \frac{\partial h^{(R)}}{\partial x_o} = \frac{x_o + V_{x_o}(k-1)\Delta T}{\text{Den}} \\
H(2:N,2) &= \frac{\partial h^{(R)}}{\partial y_o} = \frac{y_o + V_{y_o}(k-1)\Delta T}{\text{Den}} \\
H(2:N,3) &= \frac{\partial h^{(R)}}{\partial z_o} = \frac{z_o + V_{z_o}(k-1)\Delta T - \frac{1}{2}c_b(k-1)^2\Delta T^2}{\text{Den}} \\
H(2:N,4) &= \frac{\partial h^{(R)}}{\partial V_{x_o}} = \frac{x_o + V_{x_o}(k-1)\Delta T(k-1)\Delta T}{\text{Den}} \\
H(2:N,5) &= \frac{\partial h^{(R)}}{\partial V_{y_o}} = \frac{y_o + V_{y_o}(i-1)\Delta T(i-1)\Delta T}{\text{Den}} \\
H(2:N,6) &= \frac{\partial h^{(R)}}{\partial V_{z_o}} = \frac{z_o + V_{z_o}(k-1)\Delta T(k-1)\Delta T - \frac{1}{2}c_b(k-1)^2\Delta T^2}{\text{Den}}
\end{aligned}
$$

in which

$$
\text{Den} = \sqrt{(x_o + V_{x_o}(k-1)\Delta T)_o^2 + (y_o + V_{y_o}(k-1)\Delta T)_o^2 + (z_o + V_{z_o}(k-1)\Delta T - \frac{1}{2}c_b(k-1)^2\Delta T^2)^2}\,.
$$

Next, one should consider the components that make up $\frac{\partial h^{(\phi)}(\theta)}{\partial \theta}|_{\theta=\hat{\theta}(k)}$:

$$
\begin{aligned}
H(N+1,1) &= \frac{\partial h^{(\phi)}}{\partial x_o} = -\frac{y_o}{x_o^2} \\
H(N+1,2) &= \frac{\partial h^{(\phi)}}{\partial y_o} = \frac{1}{x_o}
\end{aligned}
$$

in which $H(N+1,3) = \frac{\partial h^{(\phi)}}{\partial z_o} = 0$, $H(N+1,4) = \frac{\partial h^{(\phi)}}{\partial V_{x_o}} = 0$, $H(N+1,5) = \frac{\partial h^{(\phi)}}{\partial V_{y_o}} = 0$, and $H(N+1,6) = \frac{\partial h^{(\phi)}}{\partial V_{z_o}} = 0$. For all additional azimuth measurements after the initial measurement, the linearized regressor entries are:

$$H(N+2:2N,1) = \frac{\partial h^{(\phi)}}{\partial x_o} = -\frac{y_o+V_{y_o}(k-1)\Delta T}{(x_o+V_{x_o}(k-1)\Delta T)^2}$$
$$H(N+2:2N,2) = \frac{\partial h^{(\phi)}}{\partial y_o} = \frac{1}{x_o+V_{x_o}(k-1)\Delta T}$$
$$H(N+2:2N,3) = \frac{\partial h^{(\Phi)}}{\partial z_o} = 0$$
$$H(N+2:2N,4) = \frac{\partial h^{(\phi)}}{\partial V_{x_o}} = -\frac{(y_o+V_{y_o}(k-1)\Delta T)(k-1)\Delta T}{(x_o+V_{x_o}(k-1)\Delta T)^2}$$
$$H(N+2:2N,5) = \frac{\partial h^{(\phi)}}{\partial V_{y_o}} = \frac{(k-1)\Delta T}{x_o+V_{x_o}(k-1)\Delta T}$$
$$H(N+2:2N,6) = \frac{\partial h^{(\phi)}}{\partial V_{z_o}} = 0$$

Finally, the components that make up $\frac{\partial h^{(R,\theta)}(\theta)}{\partial \theta}|_{\theta=\hat{\theta}(k)}$ are considered. One should note that $H(2N+1,3) = \frac{\partial h^{(R,\theta)}}{\partial z_o} = 1$, while the remaining components are:

$$H(2N+1,1) = \frac{\partial h^{(R,\theta)}}{\partial x_o} = 0$$
$$H(2N+1,2) = \frac{\partial h^{(R,\theta)}}{\partial y_o} = 0$$
$$H(2N+1,4) = \frac{\partial h^{(R,\theta)}}{\partial V_{x_o}} = 0$$
$$H(2N+1,5) = \frac{\partial h^{(R,\theta)}}{\partial V_{y_o}} = 0$$
$$H(2N+1,6) = \frac{\partial h^{(R,\theta)}}{\partial V_{z_o}} = 0$$

For all additional measurements after the initial increment, the following equations yield the linearized regressor entries:

$$
\begin{aligned}
H(2N+2:3N,1) &= \frac{\partial h^{(R,\theta)}}{\partial x_o} = 0 \\
H(2N+2:3N,2) &= \frac{\partial h^{(R,\theta)}}{\partial y_o} = 0 \\
H(2N+2:3N,3) &= \frac{\partial h^{(R,\theta)}}{\partial z_o} = 1 \\
H(2N+2:3N,4) &= \frac{\partial h^{(R,\theta)}}{\partial V_{x_o}} = 0 \\
H(2N+2:3N,5) &= \frac{\partial h^{(R,\theta)}}{\partial V_{y_o}} = 0 \\
H(2N+2:3N,6) &= \frac{\partial h^{(R,\theta)}}{\partial V_{z_o}} = (k-1)\Delta T
\end{aligned}
$$

4.4.5 Intercept. At this point, one can recall the generalized intercept equation shown in (4.10). One can incorporate a separate intercept for each measurement equation. In the case of projectile launch point determination considered here, a separate intercept could apply for the range, azimuth, and elevation nonlinear measurement equations in the respective observation relations, namely the multivariable intercept parameter

$$
c = \begin{bmatrix} c_R \\ c_\phi \\ c_\theta \end{bmatrix}_{3\times 1}
$$

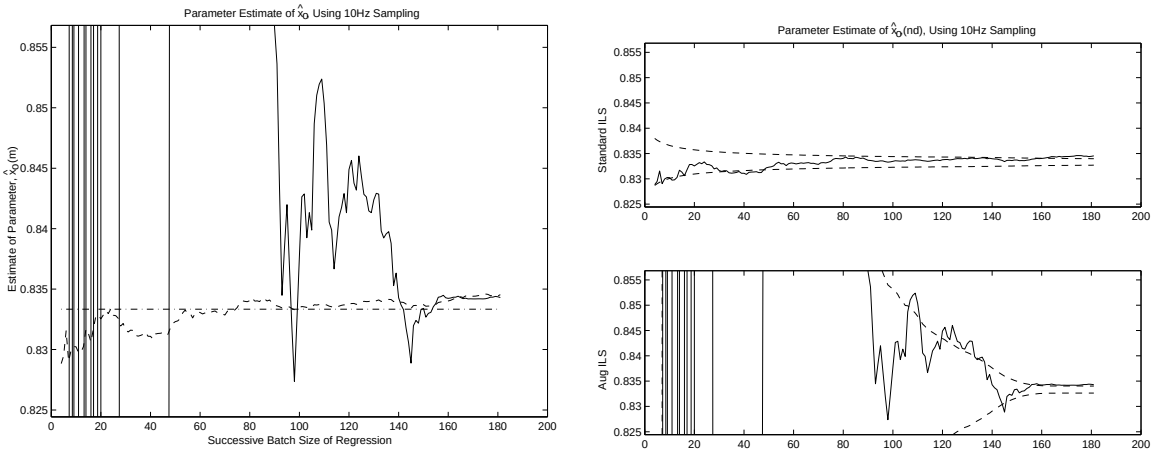
and

$$
e = \begin{bmatrix} e_R & 0 & 0 \\ 0 & e_\phi & 0 \\ 0 & 0 & e_\theta \end{bmatrix}_{3N\times 3} .
$$

As recognized in [22], there is an inherent tradeoff in estimation between the number of parameters in the regression and estimate quality. Hence, the author originally proposed that a single intercept parameter c is multiplied with a vector of ones.

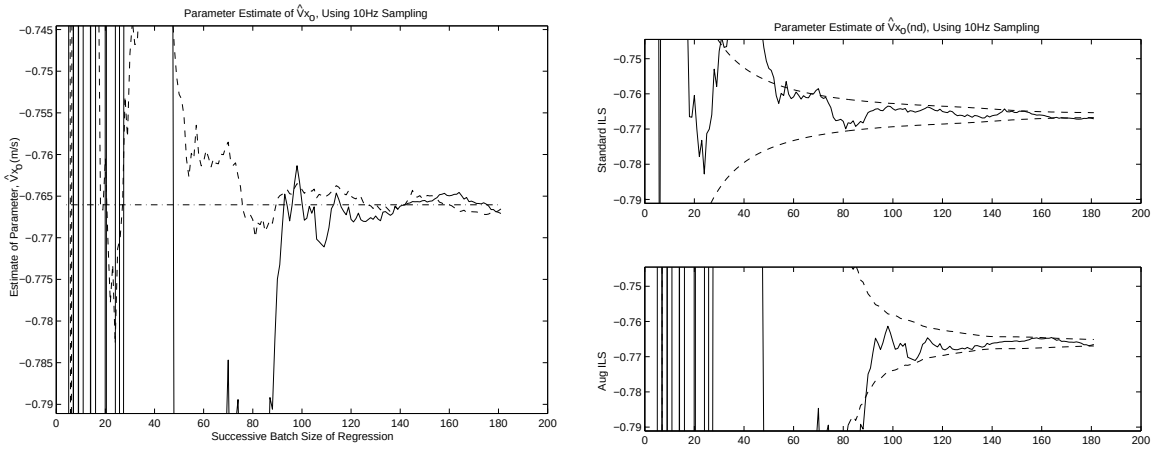
$$e = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}_{3N \times 1} . \tag{4.26}$$

The length of e corresponds to the number of physical measurements; namely N range measurements, N azimuth measurements, and N elevation measurements. This approach assumes that the resulting equation error caused from the nonlinear approximation is negligibly small compared to existing equation error $\tilde{V}$, as previously developed in Sections 4.3 and 4.4. There is good reason to assume this. Previous work with linear dynamic simulations reinforce this assumption about the intercept [22]. After considering new Monte Carlo simulations, the linear regression algorithm augmented with and intercept performs poorly under these conditions. Figures 4.3 and 4.4 show widely varying estimates for this approach, until the batch size is $\sim$160 data points. This implies that the intercept from the range, azimuth, and elevation measurements itself is strongly nonlinear. Likewise, disappointing estimation performance also result while using multiple intercepts, one for range, one for azimuth, and one for elevation. However, when one considers a generalized intercept that addresses elevation error and neglects error contributions from range and azimuth, the estimation process for the nonlinear regression augmented with an unconstrained intercept greatly improves as shown by simulation. This implementation result reflects the fact that the elevation measurement equation is more nonlinear than the range or azimuth measurement equations. In order to avoid interference between three unconstrained intercepts which results in convergence on an incorrect estimate, this approach focuses on addressing the strong nonlinearity in the elevation measurement equation. In this



(a) (b)

Figure 4.3: Augmented ILS vs. Standard ILS Position Estimate. Strongly nonlinear launch point estimation example. Typical Monte Carlo run. In (a), the solid line is the estimate from the augmented ILS with single intercept (vector of ones shown in (4.26)) algorithm $\hat{x}_o$. Dashed line from standard ILS. Dash-dot line–Truth value. In (b), the solid line represents the parameter estimate, while the dashed lines represent the estimated $\pm 1\sigma$.



(a) (b)

Figure 4.4: Augmented ILS vs. Standard ILS Velocity Estimate. Strongly nonlinear launch point estimation example. Typical Monte Carlo run. In a), the solid line is the estimate from the augmented ILS with single intercept (vector of ones shown in (4.26)) algorithm $\hat{V}_{x_o}$. Dashed line from standard ILS. Dash-dot line–Truth value. In (b), the solid line represents the parameter estimate, while the dashed lines represent the estimated $\pm 1\sigma$.

case, the estimation process contains the vector e is

$$e = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \dots \\ 1 \\ \vdots \\ 1 \end{bmatrix} \begin{matrix} \\ \\ 2N \times 1 \\ \\ \\ \\ N \times 1 \end{matrix}$$

In effect, this generalized intercept approach includes an intercept in order to impact the nonlinear elevation measurement equation only. Results from the augmented ILS with an unconstrained intercept approach are presented later in Section 4.5.2.

4.4.6 Including Intercept Constraints. The Ridge Regression type constraint introduced in (4.13) enforces the requirement that, at the point of convergence, the residual truncation error due to the nonlinearity in the measurement equations is of the same magnitude as the equation error caused by measurement noise. As previously stated in Section 4.2, implementation of the augmented ILS algorithm with unconstrained intercepts is hindered from converging to the correct solution as problem complexity increases. The Ridge constraints enable the intercepts to obtain an appropriate magnitude. Specifically, the nonlinear regression that incorporates intercepts for range, azimuth, and elevation is augmented with the three linear equations:

$$\begin{aligned} 0 &= c_R + \xi_R \quad , \quad \xi_R \in \mathcal{N}(0, \sigma_R^2) \\ 0 &= c_\phi + \xi_\phi \quad , \quad \xi_\phi \in \mathcal{N}(0, \tilde{\alpha}^2 \sigma_\phi^2) \\ 0 &= c_\theta + \xi_\theta \quad , \quad \xi_\theta \in \mathcal{N}N(0, \sigma_E^2) \end{aligned} \tag{4.27}$$

Conforming with the 1st order ergodic assumption concerning the temporal behavior of truncation error introduced by linearization, define

$$\tilde{\alpha} \triangleq \frac{1}{N}\sqrt{\text{Trace}(D_N^{(\phi)^2})}$$
$$\sigma_E \triangleq \frac{1}{N}\sqrt{\sigma_R^2\text{Trace}(D_N^{(E)^2}) + \sigma_\theta^2\text{Trace}(D_N^{(R,E)^2})}$$

in which N indicates the batch size, and the matrices $D_N^{(\phi)}$, $D_N^{(E)}$, and $D_N^{(R,E)}$ are previously introduced in (4.24). σ_R, σ_ϕ, and σ_θ represent measurement uncertainties' standard deviations. The 1st order ergodic assumption is used here to justify a complete description of the equation error noise using the first two moments, which is probably not accurate. However, this provides a constraint that is conducive to implementation.

One should consider the specifics for the two approaches used to implement the Ridge Regression for this application, as previously described in Section 4.2. The one-step method entails augmenting the constraints directly to the regression equation in (4.7). It is assumed that the user wishes to implement three intercepts for the given regression, namely for range (ρ), azimuth (ϕ), and elevation (θ). The vector and matrix components formerly used to solve for $\hat{\boldsymbol{\theta}}$ in (4.8) are replaced with the following and used in the iteration for the batch estimation, namely,

$$\boldsymbol{Z_R} = \begin{pmatrix} Z + H_i\hat{\theta}^{(i)} - h(\hat{\theta}^{(i)})_{(3N\times 1)} \\ \cdots \\ 0 \\ 0 \\ 0 \end{pmatrix}_{(3N+3\times 1)} \tag{4.28}$$

and

$$\boldsymbol{H_R} = \begin{pmatrix} & \boldsymbol{H}_{3N\times 9} & \\ \cdots & \cdots & \cdots \\ 0_{3\times 6} & \vdots & I_{3\times 3} \end{pmatrix} \quad .$$

In order to determine the new equation error covariance matrix $\mathbf{R_R}$, one must consider the following expectation expression

$$\begin{aligned} \boldsymbol{R_R} &= \mathbf{E}\Big(\begin{pmatrix} \tilde{V} \\ w_R \\ w_\phi \\ w_\theta \end{pmatrix} \cdot \begin{pmatrix} \tilde{V} & w_R & w_\phi & w_\theta \end{pmatrix} \Big) \\ &= \begin{bmatrix} R_{3N\times 3N} & & 0_{3N\times 3} \\ \cdots & \cdots & \cdots \\ 0_{3\times 3N} & & R_{\text{ridge}} \end{bmatrix} \end{aligned}$$

in which

$$R_{\text{ridge}} = \begin{bmatrix} \sigma_R^2 & 0 & 0 \\ 0 & \tilde{\alpha}^2 \cdot \sigma_\phi^2 & 0 \\ 0 & 0 & \sigma_{(R,E)}^2 \end{bmatrix} \quad . \tag{4.29}$$

As previously stated in Section 4.2, there is no correlation between $\tilde{V}$ and the noise components of the Ridge constraints. Hence, the off-axis terms of R_R are zeros.

As previously stated in Section 4.2, this one-step framework lacks the flexibility of enabling the user to apply the intercept constraint at the time of his choosing within the iteration process. Applying a constraint too early may hinder the convergence process. A framework that permits flexibility in applying the constraint is desirable.

The two-step Ridge Regression approach for this missile launch point determination is now described. First, the two-step algorithm determines the estimate $\hat{\boldsymbol{\theta}}$ and

its corresponding predicted estimation error covariance matrix $\mathbf{P} = \begin{bmatrix} P_\theta & P_{\theta\ c} \\ P_{c\ \theta} & P_c \end{bmatrix}$, by assuming an unconstrained intercept using (4.8) with the estimation error covariance (4.9). One can recall from Section 4.2 that the P_θ portion of $\mathbf{P}$ corresponds to the parameter portion of the filter-predicted equation error covariance matrix and is of interest here. The parameter estimate corresponds to that given in (4.8), with the covariance given in (4.9).

Next, one can perform a linear regression using the aforementioned estimate (4.8), covariance matrix (4.9), and Ridge Regression intercept constraints presented in (4.27). Explicitly, one can write

$$\begin{bmatrix} \hat{\boldsymbol{\theta}} \\ 0 \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \hat{\theta} \\ \hat{c} \end{bmatrix} \\ 0 \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \theta \\ c \end{bmatrix} \\ c \end{bmatrix} + \begin{bmatrix} V \\ \\ w \end{bmatrix} \quad .$$

One should note that c is defined as any combination of the intercepts mentioned in (4.27). The two-step algorithm then performs the linear regression using the "measurement"

$$\boldsymbol{Z_R} = \begin{pmatrix} \hat{\theta} \\ \hat{c} \\ 0 \end{pmatrix}$$

the regressor

$$\boldsymbol{H_R} = \begin{bmatrix} I_{(6+3)\times(6+3)} \\ 0_{(3\times 6)} \ \vdots \ I_{(3\times 3)} \end{bmatrix}$$

Table 4.2: Battlefield Radar Measurement Error Standard Deviations

σ_θ	σ_ϕ	σ_R
0.1^o	0.1^o	10m

and the equation error covariance matrix

$$\boldsymbol{R_R} = \begin{bmatrix} P_{\theta(6\times6)} & 0_{(6\times3)} \\ 0_{(3\times6)} & \text{Diag}\left(\begin{bmatrix}\sigma_1\\ \sigma_2\\ \sigma_3\end{bmatrix}\right) \end{bmatrix} \quad .$$

Now, one can choose to implement the Ridge Regression intercept constraints at any point during the iteration process . If someone wants to match the previous result determined by (4.28) and (4.29), one can incorporate the constraints immediately following the first iteration of (4.8), and continue to use the intercept constraint throughout the iteration process for a given batch size.

Finally, in order to enhance the ability of the augmented ILS algorithm with intercept to converge upon the parameter θ, an ad hoc implementation strategy is used. The diagonal of the R matrix is doubled during early iterations as stated previously in Section 4.4.2. Then, the additional uncertainty that was added to the R matrix is gradually reduced.

4.5 Simulation

This chapter's scenario entails an enemy that is launching artillery shells from an unknown location onto the friendly base. The truth trajectory and measurements data vector are generated. Noise is added to the measurement data vector. The noise on the measurements is zero-mean, Gaussian-distributed, according to the statistics given in Table 4.2. The measurement error standard deviations used in the simulation experiments reflect current near state-of-the-art hardware specifications for battlefield

radars. Angle and range measurements are taken at some fixed rate. An expanding data window of length $t = N \cdot \Delta T$ is used.

4.5.1 Initial Parameter Estimate. An initial guess of the parameter θ starts the iteration process for each successive batch. Poor initialization can lead to problems in the iteration process. A parameter initialization concept adapted from [[26], [29]] is used for both standard ILS and augmented ILS. By setting $R = I_N$, one readily obtains an initial guess for the parameter

$$\hat{\theta}^{(0)} = (H^T \cdot H)^{-1} H^T Z \quad . \tag{4.30}$$

The regressor H in (4.30) is constructed using all measurement data available within the given batch to determine the initializing parameter $\hat{\theta}^{(0)}$. H is derived as

$$H \triangleq \frac{\partial h}{\partial \theta}|_{x_m, y_m, z_m} \quad .$$

The linearization of the observation relation $h(\theta)$ is determined exclusively by the values of the first three components of θ, namely x_o, y_o, and z_o. For the purpose of initialization, set

$$x_o \triangleq x_m, \quad y_o \triangleq y_m, \quad z_o \triangleq z_m$$

in which x_m, y_m, and z_m are obtained using (4.19). In other words, for the purpose of initialization, the effects of measurement noise are neglected and used directly in the least squares initialization equation presented in (4.30).

The regressor matrix H for initialization is composed as follows:

$$H = \begin{bmatrix} \frac{\partial h^{(R)}(\theta)}{\partial \theta}|_{x_m, y_m, z_m} \\ \frac{\partial h^{(\phi)}(\theta)}{\partial \theta}|_{x_m, y_m, z_m} \\ \frac{\partial h^{(\theta)}(\theta)}{\partial \theta}|_{x_m, y_m, z_m} \end{bmatrix} \quad .$$

Specifically, the components that make up $\frac{\partial h^{(R)}(\theta)}{\partial \theta}|_{x_m,y_m,z_m}$ are

$$
\begin{aligned}
H(1:N,1) &= \frac{\partial h^{(R)}}{\partial x_o} = \frac{x_m(k)}{\sqrt{x_m(k)^2+y_m(k)^2+z_m(k)^2}} \\
H(1:N,2) &= \frac{\partial h^{(R)}}{\partial y_o} = \frac{y_m(k)}{\sqrt{x_m(k)^2+y_m(k)^2+z_m(k)^2}} \\
H(1:N,3) &= \frac{\partial h^{(R)}}{\partial z_o} = \frac{z_m(k)}{\sqrt{x_m(k)^2+y_m(k)^2+z_m(k)^2}} \\
H(1:N,4) &= \frac{\partial h^{(R)}}{\partial V_{x_o}} = \frac{x_m(k)(k-1)\Delta T}{\sqrt{x_m(k)^2+y_m(k)^2+z_m(k)^2}} \\
H(1:N,5) &= \frac{\partial h^{(R)}}{\partial V_{y_o}} = \frac{y_m(k)(k-1)\Delta T}{\sqrt{x_m(k)^2+y_m(k)^2+z_m(k)^2}} \\
H(1:N,6) &= \frac{\partial h^{(R)}}{\partial V_{z_o}} = \frac{z_m(k)(k-1)\Delta T}{\sqrt{x_m(k)^2+y_m(k)^2+z_m(k)^2}}
\end{aligned}
$$

in which $k = 1, \ldots, N$. Similarly, the components that make up $\frac{\partial h^{(\phi)}(\theta)}{\partial \theta}|_{x_m,y_m,z_m}$ are provided now. The linearized regressor entries for azimuth are:

$$
\begin{aligned}
H(N+1:2N,1) &= \frac{\partial h^{(\phi)}}{\partial x_o} = -\frac{y_m(k)}{x_m(k)^2} \\
H(N+1:2N,2) &= \frac{\partial h^{(\phi)}}{\partial y_o} = \frac{1}{x_m(k)} \\
H(N+1:2N,3) &= \frac{\partial h^{(\Phi)}}{\partial z_o} = 0 \\
H(N+1:2N,4) &= \frac{\partial h^{(\phi)}}{\partial V_{x_o}} = -\frac{y_m(k)(k-1)\Delta T}{x_m(k)^2} \\
H(N+1:2N,2) &= \frac{\partial h^{(\phi)}}{\partial y_o} = \frac{(k-1)\Delta T}{x_m(k)} \\
H(N+1:2N,6) &= \frac{\partial h^{(\phi)}}{\partial V_{z_o}} = 0
\end{aligned}
$$

in which $k = 1, \ldots, N$. Finally, for $\frac{\partial h^{(R,\theta)}(\theta)}{\partial \theta}|_{x_m,y_m,z_m}$, the following equations yield the linearized regressor entries:

$$
\begin{aligned}
H(2N+1:3N,1) &= \frac{\partial h^{(R,\theta)}}{\partial x_o} = 0 \\
H(2N+1:3N,2) &= \frac{\partial h^{(R,\theta)}}{\partial y_o} = 0 \\
H(2N+1:3N,3) &= \frac{\partial h^{(R,\theta)}}{\partial z_o} = 1 \\
H(2N+1:3N,4) &= \frac{\partial h^{(R,\theta)}}{\partial V_{x_o}} = 0 \\
H(2N+1:3N,5) &= \frac{\partial h^{(R,\theta)}}{\partial V_{y_o}} = 0 \\
H(2N+1:3N,6) &= \frac{\partial h^{(R,\theta)}}{\partial V_{z_o}} = (k-1)\Delta T
\end{aligned}
$$

in which $k = 1, \ldots, N$. A significant benefit of using this initialization process is that it is not computationally burdensome, as large matrix inversions are not required.

For comparison purposes, one should consider an initial guess of the parameter θ used to start the iteration process for each successive batch estimate that relies entirely on the first two measurement observations.

$$
\hat{\theta} = \begin{bmatrix}
x_m(1) \\
y_m(1) \\
z_m(1) \\
\frac{x_m(2)-x_m(1)}{\Delta T} \\
\frac{y_m(2)-y_m(1)}{\Delta T} \\
\frac{z_m(2)-z_m(1)+\frac{1}{2}g\cdot\Delta T^2}{\Delta T}
\end{bmatrix} .
$$

This approach appears simple and adequate for initialization, but this is not necessarily the case in nonlinear applications. An unsuccessful augmented ILS algorithm with unconstrained intercept iteration using an initialization method that relies on the first two measurements only is shown in Figure 4.5. Figure 4.6 shows the parameter estimation results using the same measurements and modified ILS algorithm

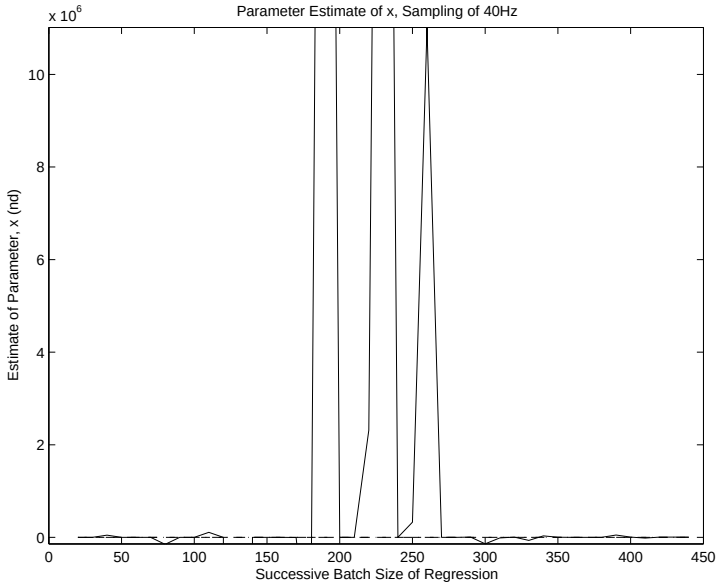


Figure 4.5: Using the first two measurements to find the initial parameter estimate. The figure shows the $\hat{x}_t$ parameter using constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS produced result is also plotted (dashed line). The truth model's value for x at $t = 0.180$ non-dimensional time units is the straight line at 3.267 non-dimensional position units (dash-dot line).

augmented with an unconstrained intercept when the initialization in (4.30) is used. Clearly, the initialization scheme is important. Table 4.3 shows the impact of the initialization schemes for a representative Monte Carlo experiment. At the same time, the calculation time required in order to obtain the initial parameter is negligible.

4.5.2 Standard ILS and Augmented ILS with Unconstrained Intercept. This subsection now reports 2-D scenario results of standard ILS and augmented ILS without intercept constraints, for the nonlinear regression, developed in Sections 4.1 and 4.4.1 through 4.4.5. In order to compare the estimation methods, the simulation experiments are confined to a 2-D scenario–(x, z) plane. Hence, for this subsection, the reduced parameter vector is considered:

$$\theta = (x_o, z_o, V_{x_o}, V_{z_o})^T \in \mathbb{R}^4 \quad .$$

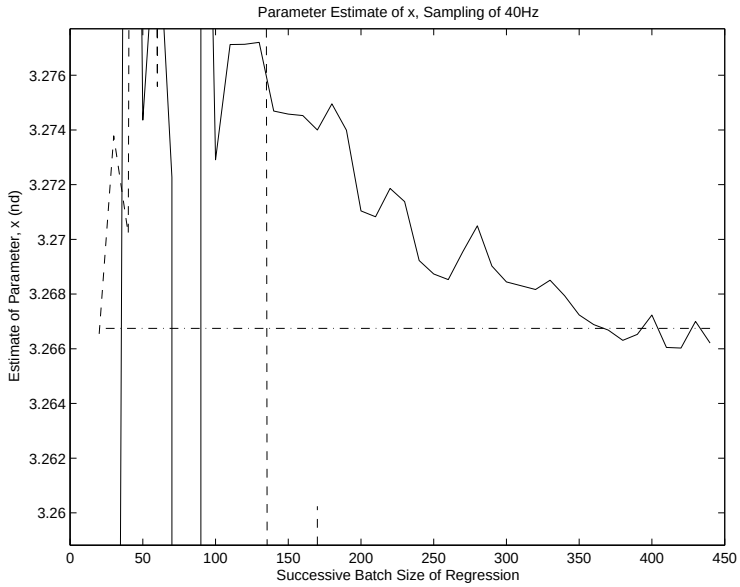


Figure 4.6: Expanding measurement window used in nonlinear regression to find the initial parameter estimate. The figure shows the $\hat{x}_t$ parameter using constrained intercepts for range, azimuth, and elevation (solid line). The action of standard ILS using only the first two measurements is also shown (dashed line). The truth model's value for x at $t = 0.180$ non-dimensional time units is pictured as the straight line at 3.267 non-dimensional position units (dash-dot line).

Table 4.3: Example Monte Carlo Run: Initial Parameter

	$x(0)$	$y(0)$	$z(0)$	$V_x(0)$	$V_y(0)$	$V_z(0)$
Truth	3.267	1.164	0.09816	-6.634	-3.830	4.468
Two Measurements	4.483	0.00180	17.741	621.5	-106.6	-222.1
Window Length N=20	2.910	2.163	0.1046	-4.513	3.499	-0.00394
Window Length N=400	2.902	2.188	0.1830	-5.121	-6.857	-0.8776
Initial Position Parameters (in non − dimensional position units)						
Initial Velocity Parameters (in non − dimensional velocity units)						

The truth model and measured data vector are generated. The noise on the measurements has zero mean and is Gaussian-distributed. The measurement errors reflect current near state-of-the-art hardware accuracy specifications for battlefield radars. Angle and range measurement noises are assumed independent at each sample. These measurements are taken every tenth of a second. At each time increment, the parameter estimate is derived in batch using all previous measurements. Thus, an expanding data window is used. The scenario considers an enemy opposition that is launching unguided artillery shells from an unknown location. Hence, in order to start the iteration process for each successive batch, one derives the initial parameter guess entirely from measurement observations.

4.5.2.1 Example 1: Radar Overflight. This example demonstrates the major benefits of the novel ILS algorithm augmented with an intercept compared to standard ILS. By overflying the radar and landing just beyond its location, the geometry causes the nonlinearity in the incoming projectile's elevation angle measurement equation to strengthen. (See Figure 4.7.) The ballistic coefficient associated with this

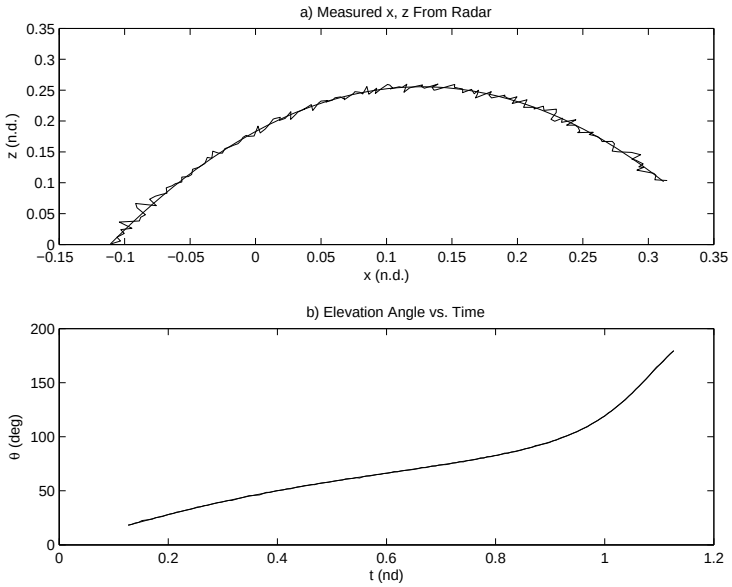


Figure 4.7: Typical Ballistic Trajectory. a) Truth model position and measured position. b) Shows a large excursion in θ, which accentuates the nonlinearity.

projectile is relatively large at $c_b = 1.35$, which further accentuates the nonlinearity

in the θ measurement equation. The increased drag on the projectile exaggerates the curvature of the projectile's trajectory. The projectile is fired from the same elevation as the radar and target; however, there is a 0.127 nondimensional time unit delay in initiating radar measurements. Measurement error standard deviations are $\sigma_\theta = 0.3^0$ and $\sigma_R = 10$ meters, respectively.

As one can see from Figures 4.8 and 4.9, standard ILS diverges when 80 or more data increments are considered, corresponding to the first 0.506 nondimensional time units of the data (from .127 to .633 nondimensional time units of projectile flight). This standard ILS divergence results directly from neglecting the nonlinear-

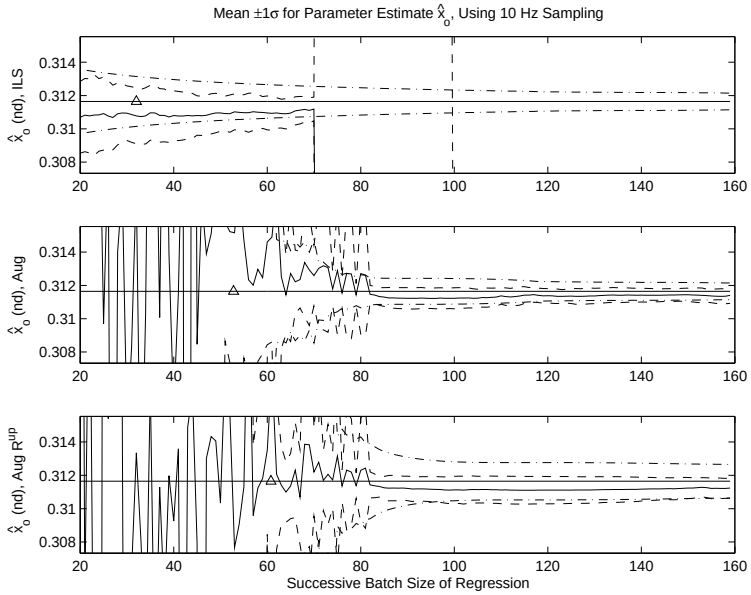


Figure 4.8: Result of 20 Monte Carlo Runs for Strongly Nonlinear Case. The figure shows: (1) the position parameter estimate $\hat{x}_o$ (solid line) (2) $\pm$ the experimentally determined standard deviation 1σ determined from the variation between parameter estimates in the 20 MC runs(dashed line) and (3) $x_{true} = 0.3116$ (solid line with $\triangle$) (4) $\pm 1\sigma$ determined from the filter-predicted estimation error covariance R (dash-dot line).

ity in the elevation angle measurement equation. Interestingly, the augmented ILS case with generalized intercept requires approximately 65 measurements (from .126 to .538 nondimensional time units of projectile flight) for convergence to the true position parameter and true velocity parameter, respectively. Increasing the covariance

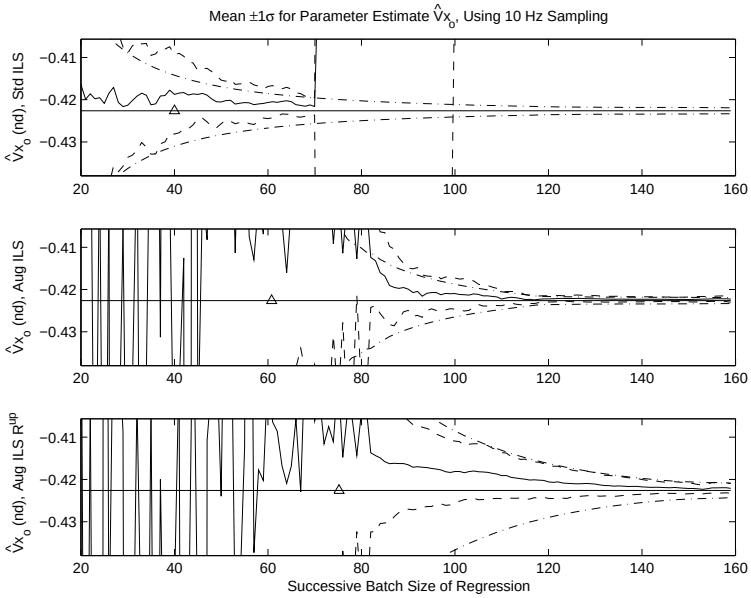


Figure 4.9: Result of 20 Monte Carlo Runs for Strongly Nonlinear Case. The figure shows: (1) the velocity parameter estimate $\hat{V}_{x_o}$ (solid line) (2) $\pm$ the experimentally determined standard deviation 1σ determined from the variation between the estimates in the 20 MC runs (dashed line) and (3) $V_{x_{true}} = -0.4226$ (solid line with $\triangle$) (4) $\pm 1\sigma$ determined from the filter-predicted estimation error covariance R (dash-dot line).

R causes slower convergence to the true parameter values, as expected. One main inference that is introduced here and is developed at the end of Section 4.5.3.1 is that a relation exists between the nonlinearity and the measurement noise level. Specifically, when a strong nonlinearity exists in a particular test case, the measurement variance will dictate the amount of benefit obtained using an augmented ILs algorithm with intercept as opposed to a standard ILS algorithm. Measurement variance that is either extremely good, or quite bad, will diminish the benefit of the augmented ILS algorithm with contrained intercepts compared with the conventional approach. Engagement geometry may dictate using the more sophisticated augmented ILS with intercept algorithm, particularly as the strength of the nonlinearity increases compared to the measurement noise level.

One should note that the projectile launch point estimate is really the primary interest. Up until this point, the estimation algorithm determines parameter estimates corresponding to the beginning of the batch. One of batch regression's main advantages in the launch point determination problem when compared to sequential (i.e. Kalman filtering) type estimation is that smoothing is not required in order to obtain estimates of a parameter set prior to the first measurement within the batch. It is assumed that the launch point elevation is known via contour map in this subsection's example. One can then use the estimate of $\hat{z}$ and $\hat{V}_z$ in order to determine the launch time estimate from

$$\hat{z} = z_o + (\hat{V}_z + c_b \cdot g \cdot \hat{t}) \cdot \hat{t} - \frac{1}{2} \cdot c_b \cdot g \cdot \hat{t}^2 \quad . \tag{4.31}$$

The estimated parameter position $\hat{x}_L$ is now projected back in time to the launch point using (4.31) for 20 Monte Carlo simulations. Figure 4.10 shows the experimentally determined estimate for each successive batch size. As the projectile enters the 4th second of recorded flight (corresponding to the 40th data point), the measurements are increasingly afflicted by the nonlinearity in the elevation relation in this example. As these data points compose an increasingly larger portion of the data window, the standard ILS approach is unable to form reasonable estimates. However,

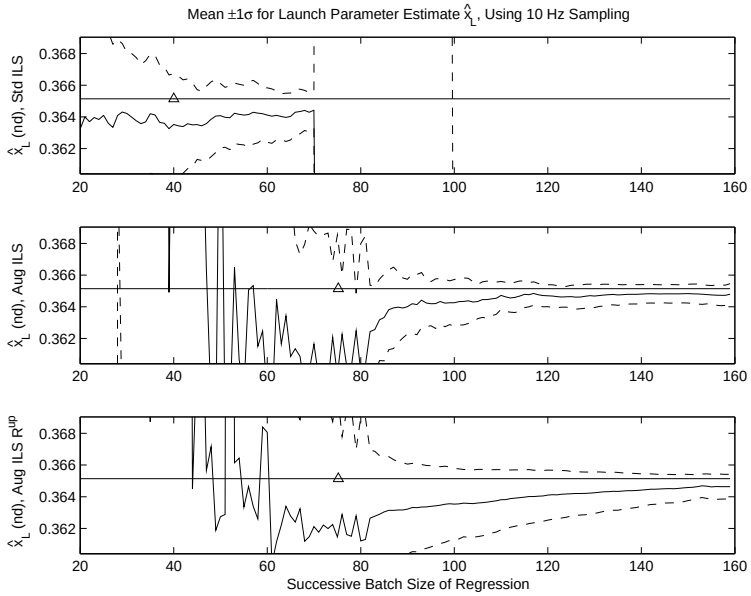


Figure 4.10: Result of 20 Monte Carlo Runs. The figure shows (1) the launch position parameter estimate $\hat{x}_L$ (solid line) (2) $\pm$ the experimentally determined 1σ from the variation betweeen the parameter estimates (dashed line). (3) x_{Ltrue} is also shown (solid line with $\triangle$).

the augmented ILS algorithm with unconstrained intercepts is able to cope with the nonlinearity. The estimates produced by the algorithm converge upon the true parameter value when a large enough data window has accumulated. The plot also shows the experimentally determined standard deviation of the parameter estimate. If the equation error covariance for the launch point estimates is desired, one could increase R to reflect the uncertainty in the time estimate. Additional results associated with this example problem are located in Appendix F.

4.5.2.2 Example 2: Stressful Geometry. In this scenario, the radar and enemy projectile launch location are both on two hilltops, 1500 meters apart. A projectile is fired towards the radar, but falls short into a ravine separating the two hilltops (Figure 4.11). Also, there is little movement in elevation angle for the first 120 measurements.

Figures 4.12 and 4.13 show the results of the estimation process. The estimation problem is not impacted by the nonlinearity, which is weak compared to the

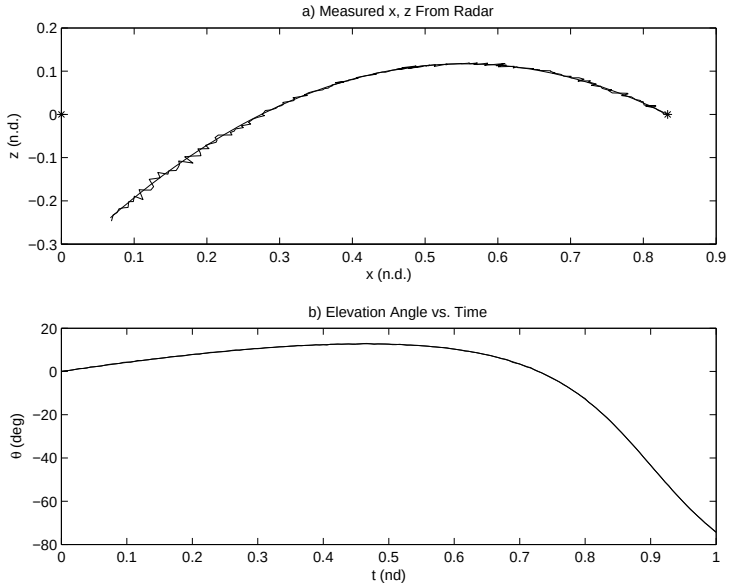


Figure 4.11: Near-linear Example of Typical Monte Carlo Ballistic Trajectory Profile. (a) Shows the truth model position, as well as the position measurements. The stars $*$ indicate the artillery piece position and the radar location. (b) Shows little movement in θ for the first $\sim$75% of the tracked projectile's flight.

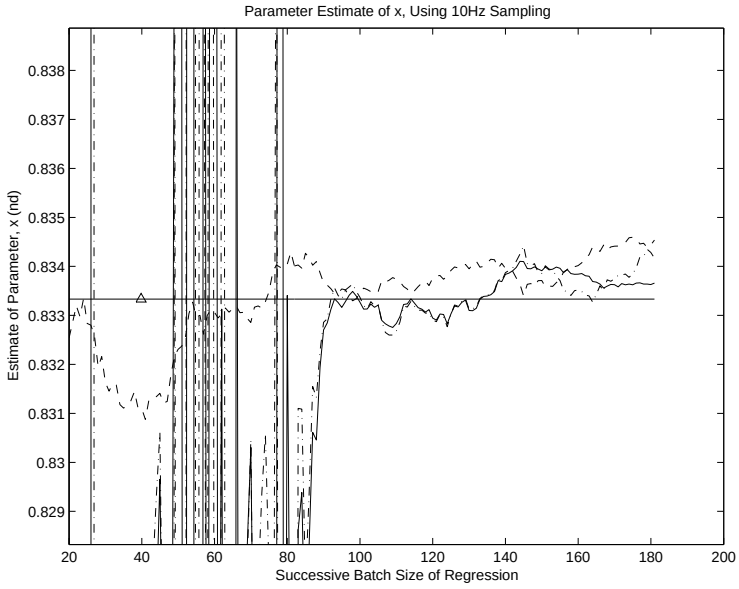


Figure 4.12: Position estimate is provided by a typical Monte Carlo Run with near-linear changes in elevation. Standard ILS quickly renders a good estimate (dashed line). The augmented ILS with intercept (dash-dot line) and the augmented ILS estimate determined with increased uncertainty in R (solid line) also produce good estimates after $\sim$90 measurements. The truth value is denoted by the solid line with a $\triangle$.

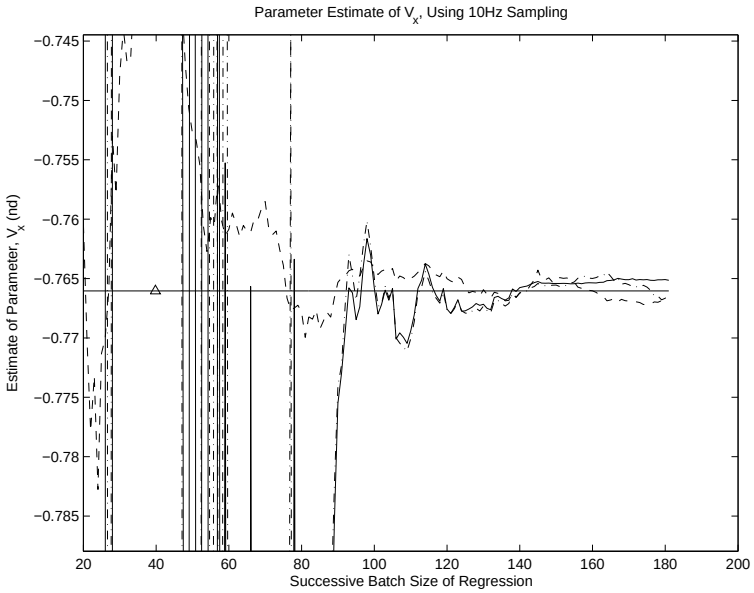


Figure 4.13: Velocity estimate is provided by a typical Monte Carlo Run with near-linear changes in elevation. Line types are the same as in Figure 4.12.

strong measurement noise. The battlefield radar specifications include measurement uncertainties' standard deviations of $\sigma_\theta = 0.1^o$ and $\sigma_R = 10$ meters, respectively, and the ballistic coefficient c_b is 1.0 in this particular example. Standard ILS establishes a good estimate quickly, after incorporating as few as 10 samples. All three estimation methods develop a good estimate by ∼90 data samples, given the 10 Hz sampling rate.

The delay in the augmented ILS algorithm with intercept is caused by the lack of information in the elevation measurement, as the elevation measurement moves little from 0° for the first 0.8 nondimensional time units of projectile flight (Figure 4.11). The intercept is allowed to float in an unconstrained fashion throughout the iteration process. This causes the increased estimate convergence time when compared to Example 1, even though the elevation measurement noise statistics are greatly enhanced in the latter example. It is important to note that, after data is accumulated for 0.444 nondimensional time units, the augmented ILS with generalized intercept does not hurt the estimate.

Finally, the best position estimate $\hat{x}$ is obtained when the block diagonal terms of the covariance matrix R are slightly increased and one uses ∼140 or more measurements in the batch. The estimates are generated with the covariance block diagonal terms at 25% above their typical level. The estimates are quite invariant compared to their counterparts when the batch size contains at least 140 data points. The final ∼40 data points correspond to the last four seconds of the projectile flight. The geometry in Example 2 accentuates two effects in the observations, as the elevation angle in Figure 4.11(b) begins to change rapidly, corresponding to the projectile crashing into the ravine. First, the measurement errors can cause quite erratic "movement" in the ballistic projectile's final decent. The added uncertainty to the diagonal terms deemphasizes this erratic movement in the measurement data. Second, this increase in uncertainty accounts for equation error $\tilde{V}$ previously neglected, as shown in (4.10). Hence, the additional uncertainty reduces the impact of the nonlinear dynamics that are linearized during the estimation process. The reduction in estimate variation that is achieved from increasing the uncertainty in the covariance matrix R in the 2-D case does not extend readily to the 3-D case.

4.5.2.3 Conclusions About Augmented ILS with Unconstrained Intercepts.

The algorithms used to generate the estimates presented in this subsection employ batch data processing and nonlinear regression augmented with an intercept, and demonstrate the novel algorithm in a ballistic trajectory tracking scenario. Augmented ILS with generalized intercept clearly expands the operational envelope of these batch estimation algorithms. It is also important to note that the enhanced augmented estimation method does not adversely impact the estimation ability in the nearly linear estimation cases, if the batch estimation algorithm is given enough measurements.

The estimation performance of standard ILS is improved by incorporating two key features. First, the research recognizes that the intercept is a multi-variable parameter. Each nonlinear measurement equation provides an opportunity to introduce

a scalar intercept. In this case, the best estimation results are obtained when the intercept is only applied to the critical angle measurement equation in which the nonlinearity is strongest. Indeed, the method is extremely effective in cases under the condition that the magnitude of the nonlinearity is large compared with the measurement noise level. In addition, the method does not adversely impact the estimation for nearly linear cases, if given enough measurements within the batch. The results presented so far only show these two extremes. Additional inferences about about the relationship between the strength of the nonlinearity and measurement noise are discussed later in Section 4.5.3.1.

Second, it is recognized that when one linearizes a nonlinear system, there is increased uncertainty above and beyond the equation error caused by measurement noise. Hence, "tuning" by increasing the diagonal terms of the equation error covariance matrix R may assist estimation convergence in some cases, particularly when problem geometry amplifies the deleterious effects of nonlinearity. The benefits on estimation performance accrued by including prior information on the muzzle velocity require investigation. See Appendix G.

4.5.2.4 Reasons for Constraining the Augmented Intercepts. Even though the novel augmented ILS algorithm augmented with unconstrained intercepts realizes an expansion of the envelope of successful estimation over standard ILS, one can observe two motivations to include intercept constraints via Ridge Regression approach. First, in all cases that the estimation iteration converged, standard ILS converged with a shorter data window record when compared to augmented ILS with unconstrained intercept. When using the augmented ILS with unconstrained intercept, estimation results deteriorate–more bias and higher variance–if too little measurement data is present, a direct result from augmentation of the parameter vector θ with an intercept c to account for linearization-induced truncation error. Second, when the nonlinearity is weak, augmented ILS with unconstrained intercept yields a noticeable increase in the variance of the parameter estimate when compared to stan-

dard ILS. The benefits of using augmented ILS with unconstrained intercepts still outweigh the cost when compared to using standard ILS. However, better estimation results are achievable, as discussed in Section 4.5.3. The proposed extension addresses this covariance issue directly while expanding on the two aforementioned benefits.

4.5.3 Standard ILS and Augmented ILS with Constrained Intercept. Now, consider 3-D scenario results of standard ILS and augmented ILS with constrained intercepts. The Ridge Regression constraints are previously developed in Sections 4.2 and 4.4.6. One can recall that, for this 3-D scenario, the parameter vector is as shown in (4.18). The scenario is as described in the prelude of Section 4.5.2.

4.5.3.1 Trajectory with Significant Nonlinearity due to Geometry. A trajectory profile in which measurements of azimuth and elevation (Figure 4.14) vary considerably within the data window, particularly from 0.360 to 0.676 nondimensional time units, is considered. The azimuth plot indicates an artifical 360^o jump,

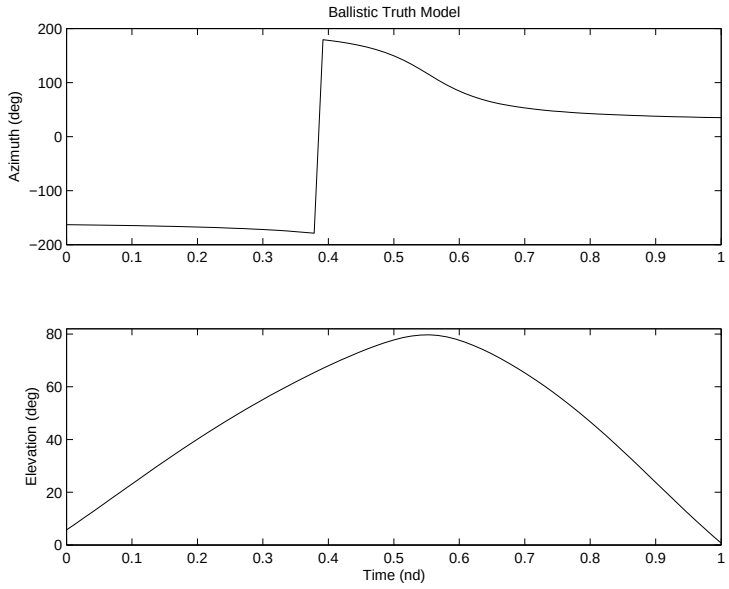


Figure 4.14: Azimuth and Elevation Time History. The augmented ILS algorithm with constrained intercepts is used in an example with strong nonlinearity. The projectile's ballistic coefficient $c_b = 1.2$.

which results from the defined domain. The projectile is launched from $(x_o, y_o) =$

$\sim(-0.287, -0.0878)$ nondimensional position units towards a target located at $(x, y) = \sim (0.200, 0.139)$ nondimensional position units. The measurement data vectors are generated and noise is added to the measurements. The measurement errors used in the simulation experiments reflect current near state-of-the-art hardware specifications for battlefield radars, as reflected in the previous Table 4.2.

First, the results of a single Monte Carlo experiment with data measurements sampled at $3\frac{1}{3}$Hz are considered. This sampling frequency was selected to account for the projectile trajectory and the desired computational capabilities used to process the ILS algorithms. In practice, one would not have this luxury; however, one could use whatever data is available. Figures 4.15 and 4.16 clearly show the performance of ILS augmented with constrained intercepts for range, azimuth, and elevation as superior to that of standard ILS. ILS augmented with constrained intercepts yields

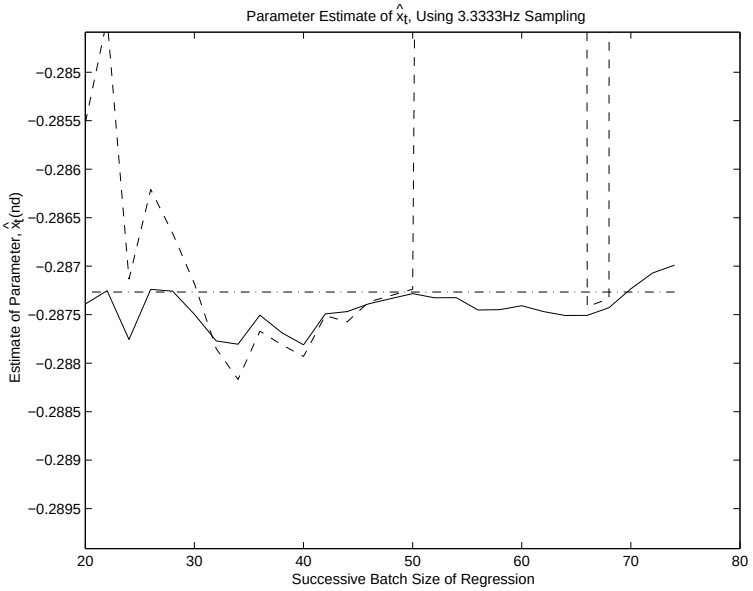


Figure 4.15: Position parameter estimate $\hat{x}_t$ using constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS results are also shown (dashed line). The truth value for x at $t = 0$ is at -0.0282 nondimensional distance units (dash-dot line).

a good position estimate for all batch sizes shown in the figures, particularly when the batch includes measurements that are impacted by the strong nonlinearity in the azimuth and elevation measurement equations. The 45th through 65th data points

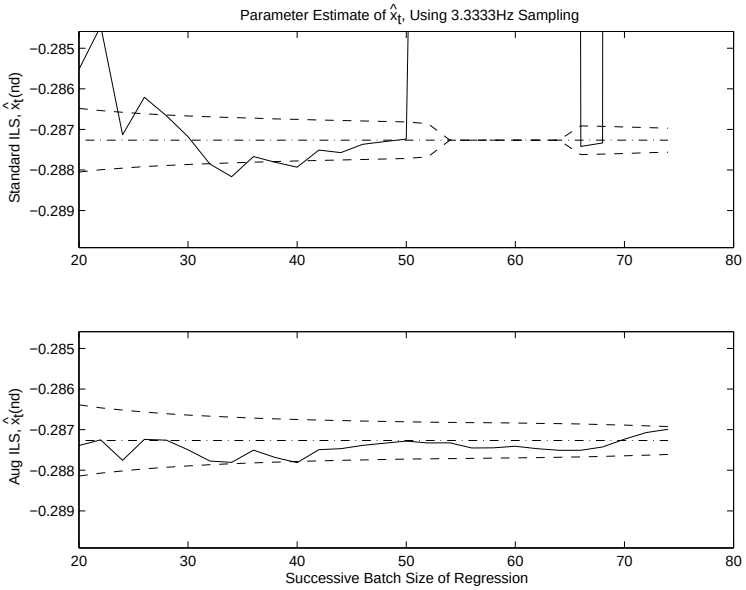


Figure 4.16: Position parameter estimate $\hat{x}_t$ using an constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS is also shown. Dashed lines represent plus minus one standard deviation from the filter-predicted equation error covariance from the true x (dash-dot lines).

correspond to the 0.360 through 0.676 nondimensional time units of projectile flight. The measurements are increasingly afflicted by the nonlinearity in the elevation relation in this example. As these data points compose an increasingly larger portion of the data window, the standard ILS approach is unable to form reasonable estimates. However, the augmented ILS algorithm with constrained intercepts is able to cope with the nonlinearity. Additionally, the Augmented ILS algorithm with constrained intercepts overcomes the two shortcomings described in the Section 4.5.2. Specifically, Figure 4.16 shows that the estimates obtained by the augmented ILS algorithm with constrained intercepts converge with a short data window, superior to that achieved by the standard ILS algorithm. One should note that the early measurements within the data window are quite linear. The same assessment applies to the velocity parameters, such as $\hat{V}_{x_t}$, as shown in Figures 4.17 and 4.18. The standard ILS velocity estimate degrades significantly for the corresponding batches, since the algorithm does not account for nonlinearity in the measurement equation. As these increasingly nonlinear data measurements compose an increasingly larger

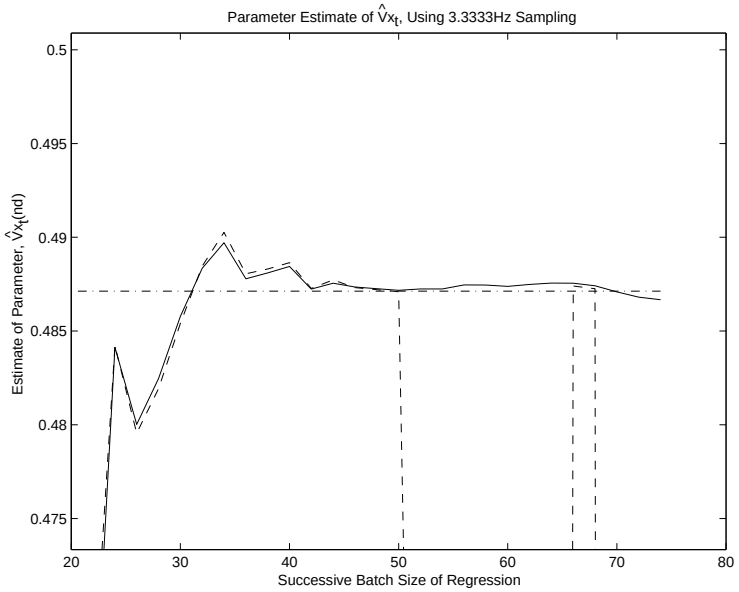


Figure 4.17: Velocity parameter estimate $\hat{V}_{x_t}$ using constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS is also shown (dashed line). The truth model's value for V_x at $t = 0$ is pictured as the straight line at 0.487 nondimensional velocity units.

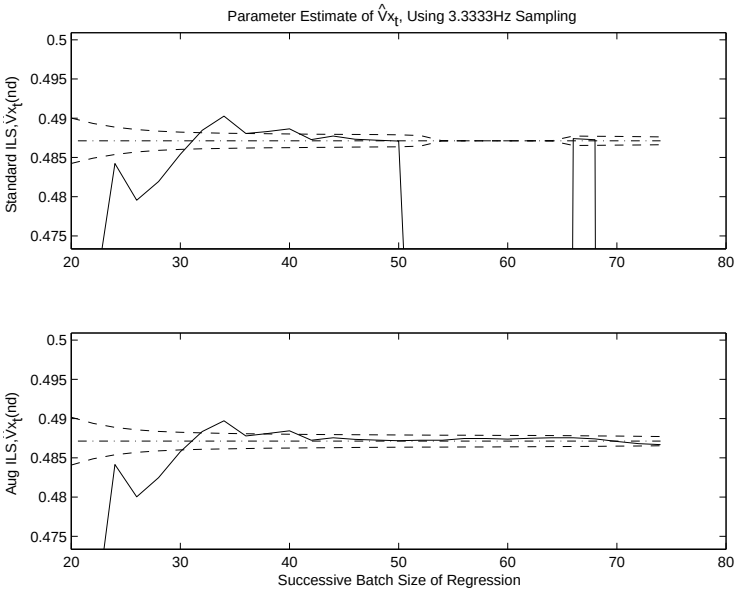


Figure 4.18: Velocity parameter estimate $\hat{V}_{x_t}$ using an constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS is also shown. Dashed lines represent plus minus one standard deviation from the filter-predicted equation error covariance from the true V_x (dash-dot lines).

portion of the data window, the standard ILS approach is unable to form reasonable estimates when $N = 50$ (Figures 4.17 and 4.18). The augmented ILS results are greatly improved when compared to previous simulation results presented to the augmented ILS algorithm with unconstrained intercepts in Section 4.5.2. The augmented ILS algorithm incorporates constraints on the intercepts to limit their value to the noise magnitude for a given batch size, as explained in Section 4.2. Additional results associated with this example are shown in Appendix H.1.

The results from 20 Monte Carlo runs shown in Figure 4.19 clearly demonstrate superior estimation performance achieved by ILS augmented with constrained intercepts for range, azimuth, and elevation, over that of standard ILS. Results for ad-

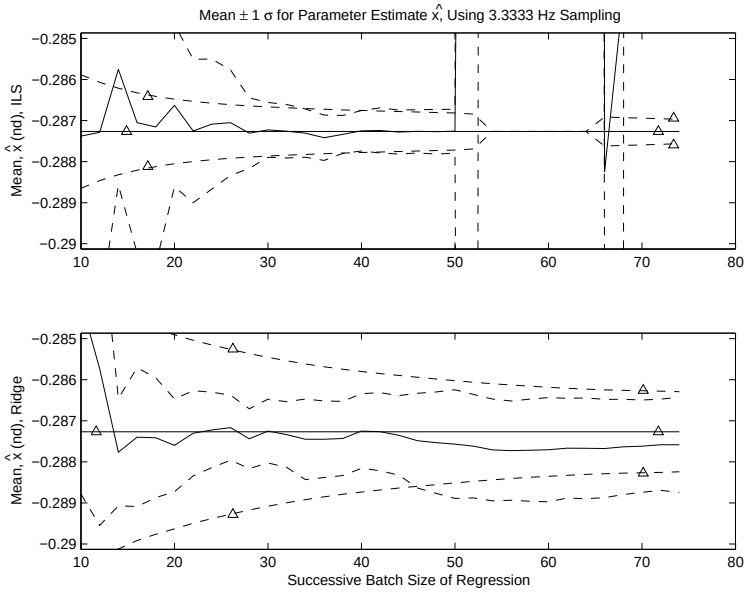


Figure 4.19: 20 Monte Carlo Experiments. Position estimates produced by the augmented ILS algorithm with constrained intercepts are superior to estimates produced by conventional ILS when nonlinearity is strong. The figure shows: (1) $x_{o\text{true}} = -0.287nd \pm 1\sigma$ determined from the filter-predicted covariance estimation error (lines with $\triangle$s). (2) Parameter estimate $\hat{x}_o \pm$ experimentally determined standard deviation 1σ from the variation between the parameter estimates within the 20 Monte Carlo runs. (lines without $\triangle$s).

ditional parameter estimate values are shown in Appendix H.2. As previously shown, as measurements afflicted by strong nonlinearity compose an increasingly larger portion of the data window, the standard ILS approach is unable to form reasonable

estimates for $N \geq 50$. The augmented ILS algorithm with constrained intercepts produced good estimates for all data window lengths. Figure 4.19 also shows that the standard deviations experimentally determined from the variation on the parameter estimates are smaller than the standard deviations of the filter-predicted equation error covariances, demonstrating the augmented ILS method's performance when the nonlinearity is strong relative to the measurement-noise-induced equation error. Constraining the intercepts enables quick convergence to the truth values, with fewer than 20 measurements included in the batch. The figure shows that the opposite is true for the standard ILS algorithm. Additionally, this compares favorably to the performance of the augmented ILS algorithm with unconstrained intercept, which typically required at least 65 measurements in the batch for the convergence. Hence, the augmented ILS with constrained intercepts has clearly overcome the two shortfalls associated with the augmented ILS algorithm with unconstrained intercepts presented in Section 4.5.2. The same assessment applies to the velocity parameter estimates, such as $\hat{V}_{x_t}$, as shown in Figure 4.20.

An extreme case is shown here, in which the geometry causes strong nonlinearity in the measured equation. Numerous scenarios were considered that show the importance of the interplay between nonlinearity strength and measurement noise. The observations about this relationship between nonlinearity strength and measurement noise are summarized in Figure 4.21. In general, estimation for problems with significant nonlinearity benefits when one uses the augmented ILS algorithm with intercepts as opposed to standard ILS. The benefit diminishes when either the measurement error standard deviation is either really low or really high. In the former case, both algorithms benefit from the good measurements with low measurement error, such that the marginal difference between estimation methods diminishes. Likewise, when measurement error degrades significantly, both methods would produce poor results, hence removing any benefit from the choice of estimation approach. The second major generalization is that the derived benefit from using augmented ILS algorithm with constrained intercepts over standard ILS is dependent on the problem geometry. As

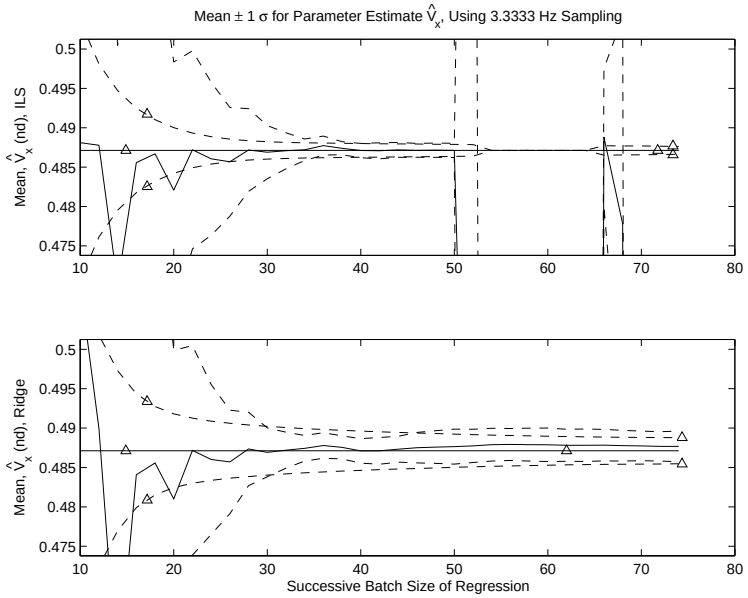


Figure 4.20: 20 Monte Carlo Experiments. Velocity estimates produced by the augmented ILS algorithm with constrained intercepts are superior to estimates produced by conventional ILS when nonlinearity is strong. The figure shows: (1) $V_{xo\text{true}} = -0.4873nd \pm 1\sigma$ determined from the filter-predicted covariance estimation error (lines with $\triangle$s). (2) Parameter estimate $\hat{V}_{x_o} \pm$ experimentally determined 1σ from the variation between the parameter estimates within the 20 Monte Carlo runs (lines without $\triangle$s).

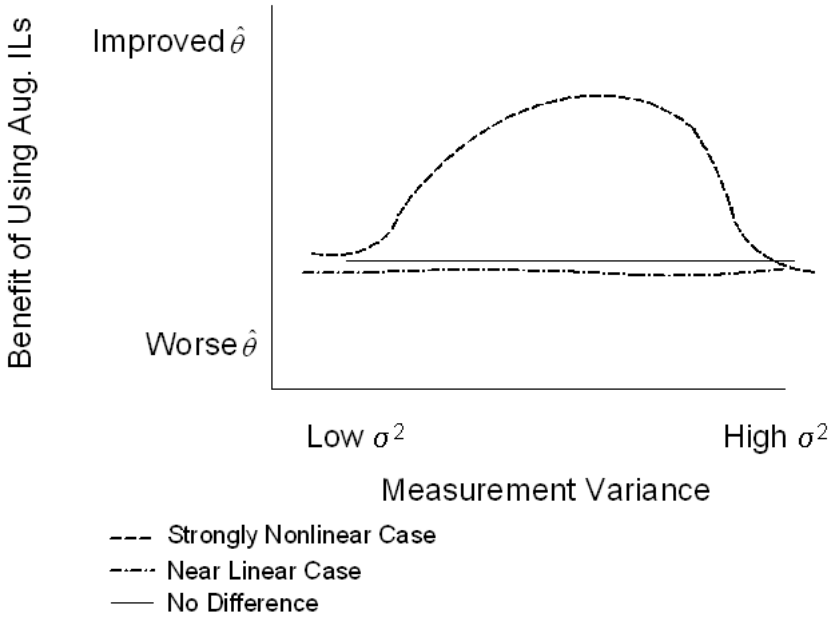


Figure 4.21: Qualitative assessment of improvement to estimate quality. Augmented ILS with constrained intercepts vs. conventional ILS

the nonlinearity increases for the specific geometry case, the estimates increasingly benefit by using the augmented ILS algorithm with constrained intercepts. There is a limit to the envelope in which estimation is possible using either of these approaches. Clearly, the envelope in which estimation is possible is expanded using the augmented ILS algorithm with constrained intercept. Section 4.5.3.2 will address how to deal with such a case, in addition to showing results associated with a near-linear portion of the projectile trajectory.

4.5.3.2 Aggressive Trajectory. The need for higher sampling rates, and consequently long data windows, might require division of the batch into smaller data records. In some batches, the $3\frac{1}{3}$Hz sampling rate is too slow to capture rapid excursions in azimuth and elevation, such as in the aggressive scenario from [27], which is now considered. In the second scenario, the energetic projectile is fired at a relatively steep ascent and very close to the radar. The projectile nearly overflies the radar. As a result, the azimuth angle slews ∼80 degrees in 0.0153 nondimensional time

units and changes ∼130 degrees over a 0.0429 nondimensional time unit period, while the entire projectile flight time lasts 1 nondimensional time unit (Figure 4.22(a)). In

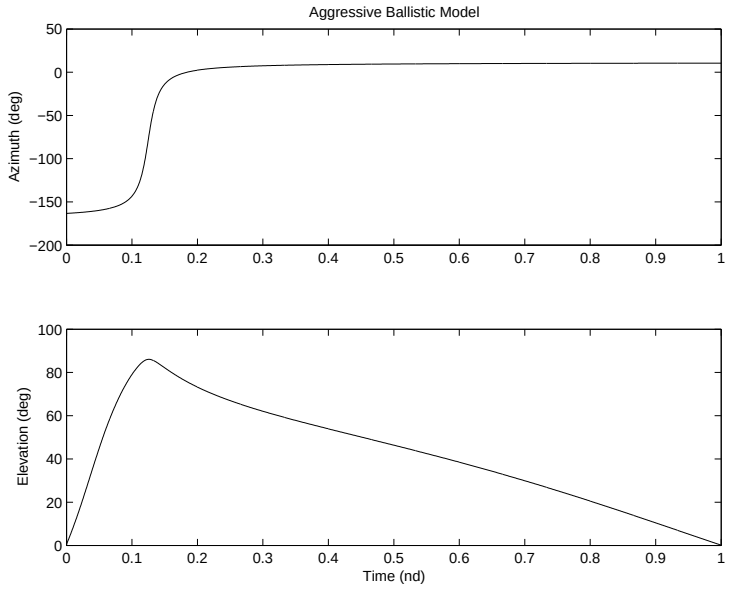


Figure 4.22: Azimuth and Elevation Angles Versus Time for the Aggressive Trajectory.

addition, the elevation angle is also changing rapidly during this time as it climbs from launch (Figure 4.22(b)).

If one samples range, azimuth, and elevation at the frequency required to characterize the dynamic portion of this projectile's flight, the batch size increases significantly. Consequently, the size of the R matrix increases to the point that the computational capabilities are overwhelmed; the solution is to subdivide the batch into sub-batches. The length of each sub-batch is limited to reflect computational capabilities. The author finds that data windows of ∼400 measurements require a reasonable amount of time in order to perform the calculations for the nonlinear regression, given his computational resources available for simulation.

Specifically, estimates and predicted estimation error covariances are obtained for each sub-batch within a single Monte Carlo run. The complete sub-batch information is captured by its initial state estimate and predicted estimation error covariance.

These are used in a final linear regression when the initial launch point is calculated, which is discussed near the end of this subsection.

Figure 4.23 reveals one particular sub-batch of interest, which occurs between 0.184 and 0.208 nondimensional time units and requires a 100 Hz sampling rate. This sub-batch is a near-linear part of the rapidly changing portion of the overall

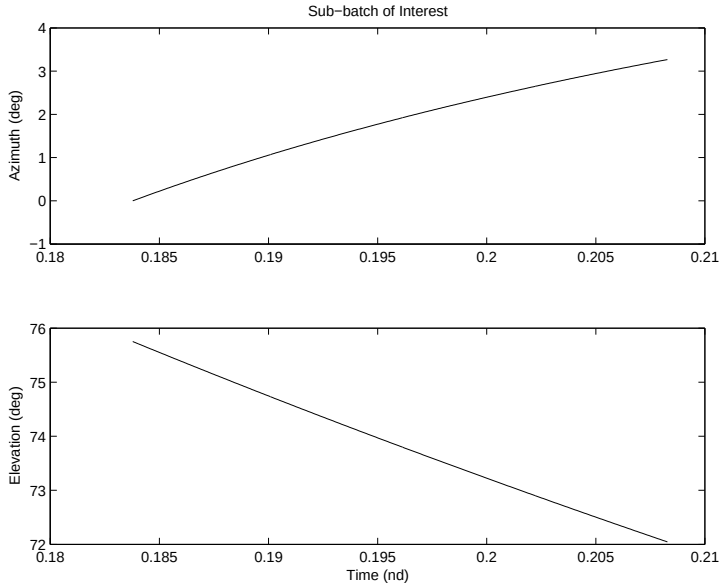


Figure 4.23: Truth Model Angles Versus Time for the Chosen Near-linear Sub-batch Within the Aggressive Trajectory.

trajectory. Figure 4.24 shows the estimated y position parameter for the particular sub-batch of interest. The parameter estimates derived for each sub-batch are then combined. Because the sub-batches do not overlap, the noise corruptions associated with the parameter estimates are uncorrelated batch-to-batch. Therefore, they are easily combined to yield the final parameter estimate, which captures the information in the entire batch. The results presented here are from a typical Monte Carlo run. A significant point here is that augmented ILS with constrained intercept provides parameter estimates that do not degrade when compared to standard ILS, even when the nonlinearities are relatively weak compared to the measurement noise intensity, as presented here in the near overflight case. Additional results for this example are shown in Appendix H.3.

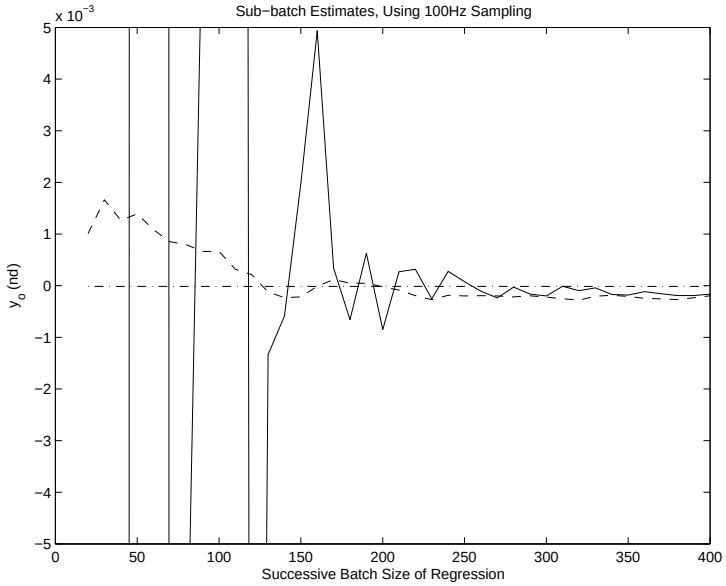


Figure 4.24: Position parameter estimate $\hat{y}_o$ for particular near-linaer sub-batch example using augmented ILS with constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS estimate is also shown (dashed line). The truth model's value for y at $t = 0.178$ non-dimensional time units is pictured as the straight line at approximately 0 non-dimensional position units (dash-dot line).

Estimates and predicted estimation error covariances are obtained for each sub-batch within a single Monte Carlo experiment. However, one is really interested in the projectile launch point estimate. The estimates and covariances from each sub-batch are converted back to project an estimate of the launch point. If one assumes a known launch time, one can project the estimate directly back to the start of the trajectory profile. In this case, one directly uses the calculated parameter estimates and covariances from each sub-batch to estimate the launch point. This assumption is very unreasonable, in that it would require precise knowledge of when the enemy is going to fire the projectile.

Alternatively, the estimate of $\hat{z}$ and $\hat{V}_z$ are used to determine the launch time estimate using knowledge of the terrain via contour map, as previously shown in (4.31). Figure 4.25 shows the results of using (4.31) to obtain the estimate. In effect, the estimates from each sub-batch are projected into a "common launch time" (i.e., the time of launch). This estimated launch time projects directly into determining the

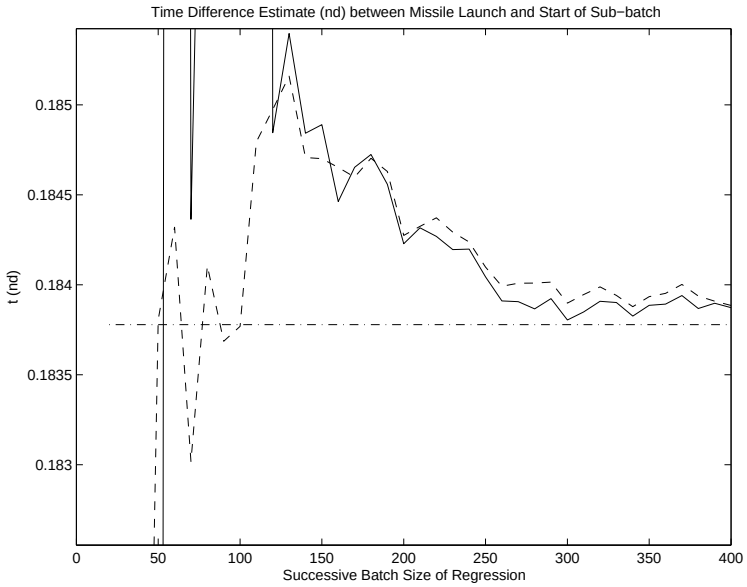


Figure 4.25: Estimate of the launch time prior to the start of the selected near-linaer sub-batch example. The solid line represents the time estimate using the augmented ILS algorithm with constrained intercepts for range, azimuth, and elevation (solid line). The time estimate produced from the standard ILS algorithm is also shown (dashed line). The truth model's value for time at $t = 0.184$ non-dimensional time units (dash-dot line).

launch position. An iterative process can "look up" known elevations relative to the

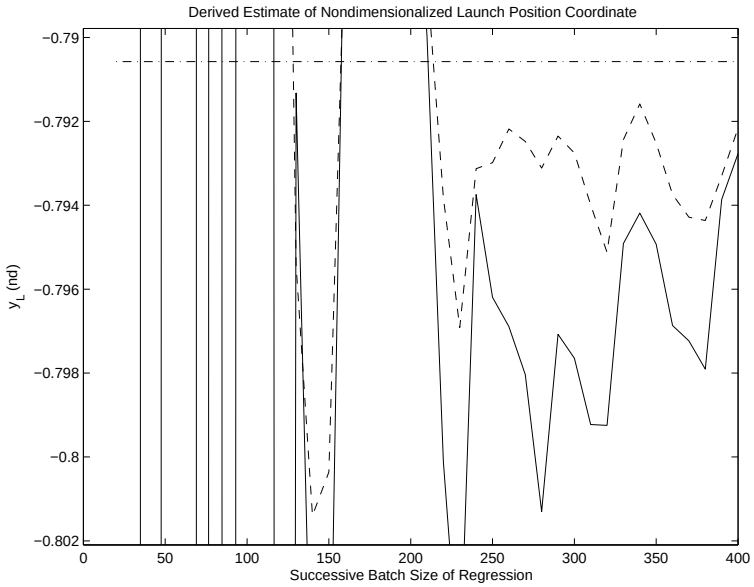


Figure 4.26: Position estimate at launch $\hat{y}_L$ using augmented ILS with constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS estimate is also shown (dashed line). The truth model's value for y_L at launch is pictured as the straight line at -0.0193 non-dimensional position units (dash-dot line).

radar location via contour map, thus enabling solution convergence. After solving for $\hat{t}$, one can determine $\hat{y}_L$ at launch, as shown in Figure 4.26. One should then augment the filter-predicted equation error covariances (i.e., increase diagonal components of P) in order to reflect the uncertainty in the time estimate.

At this point, one can solve a linear regression to fuse the information effectively into an estimate of the launch position:

$$\begin{bmatrix} \hat{\theta}_1 \\ \vdots \\ \hat{\theta}_k \end{bmatrix} = \begin{bmatrix} I \\ \vdots \\ I \end{bmatrix} \cdot \theta + V \tag{4.32}$$

$$Z = H\theta + V$$

in which

$$R = \begin{bmatrix} P_1 & & -0- \\ & \ddots & \\ -0- & & P_k \end{bmatrix} \quad .$$

Estimates $1, 2, \ldots, k$ correspond to the respective sub-batches as previously described. The best results for the augmented ILS algorithm with intercepts are obtained when the information pertaining to the intercept is stripped from the filter-predicted equation error covariances P_j, in which $j = 1, \ldots, k$. One can recall that the intercept reflects the error size associated with the linearization of the nonlinearity. This information is not needed in the data fusion. The solution of this linear regression is now performed [36] and the finite parameter estimate is calculated

$$\hat{\hat{\theta}} = (H^T R^{-1} H)^{-1} H^T R^{-1} Z \quad .$$

The double hat signifies that $\hat{\hat{\theta}}$ is the data fused parameter estimate of the previously obtained sub-batch estimates.

4.5.3.3 Conclusion of Augmented ILS with Constrained Intercepts. A parameter estimation method is developed using batch data processing and nonlinear regression augmented with a constrained intercept. The inclusion of an intercept is aimed at addressing the significant nonlinearity effects, namely, linearization-induced truncation error. The size of the estimated intercept is made commensurate with the measurement-noise-induced equation error, using a Ridge Regression argument. Also, careful modelling and nondimensionalization are employed, and large batch sized are accommodated as necessary. As a result, and as thoroughly validated by extensive numerical experiments, the estimation performance is improved beyond that achieved by standard ILS in three measures. First, the parameter estimation error variation is reduced, particularly when the nonlinearity is strong. Second, the predicted parameter estimation error standard deviation corresponds to the experimentally recorded

parameter estimation error standard deviation. Third, the probability of the iteration converging to the correct estimate is increased, particularly in formerly difficult measurement geometries and when the nonlinearity is strong; hence, expanding the effective estimation envelope. Additionally, augmented ILS with a constrained intercept does not degrade estimation results when the nonlinearity is relatively weak compared to the measurement noise–the novel parameter estimation algorithm is robust for this important consideration. The method clearly mitigates some vulnerability caused by neglecting nonlinearity, as in standard ILS, while maintaining high quality estimates in geometric situations in which standard ILS also does well. The augmented ILS method is conducive to good parameter estimates in an expanded measurement geometry envelope and without an increase in the estimation error covariance.

4.6 Conclusions for Nonlinear Regression

This research discusses the importance of interplay between strength of nonlinearity and measurement noise level. Augmented ILS with generalized unconstrained intercept clearly expands the envelope of successful estimation for these batch processes. Augmented ILS with unconstrained intercepts does not adversely impact the estimation ability in the nearly linear cases, if the batch estimation algorithm is given a long enough data window. Parameter estimation can benefit by artificially increasing the equation error covariance matrix R diagonal terms in some cases. This acknowledges an elevated level of equation error uncertainty.

Adding the Ridge Regression constraints clearly resolves some vulnerability associated with the augmented ILS algorithm with an unconstrained intercept presented in Section 4.5.2. Specifically, augmented ILS algorithms with a constrained intercept achieves convergence with short data windows. Also, variance of the parameter estimate is greatly reduced. Augmented ILS with constrained intercepts further expands the geometry cases in which obtaining good estimates is possible, compared to standard ILS. Augmented ILS with constrained intercepts is superior when nonlinearity

is dominant. In addition, the methodology maintains the high quality of estimation in "near-linear" geometric cases in which conventional ILS also performs well.

V. Reconfigurable Flight Control

This chapter uses augmented linear regression with unconstrained intercept algorithm developed in Chapter III to improve on system identification performance under trim change when compared to the iterative least squares estimation approach. The adaptive and reconfigurable flight control system then uses the system identification results to adjust the inner loop control gains by actively using a feed-forward control input. This chapter develops the adaptive reconfigurable flight control system. Pitch rate tracking is demonstrated under a sizeable control surface failure. The adaptive reconfigurable flight control system pitch rate tracking performance surpasses the tracking performance produced by the conventional flight control system that solely relies on integral feedback control in terms of ability to counter the failure.

5.1 Introduction to Flight Control

One should consider a batch estimation process with an expanding window and the important special case in which the equation error covariance matrix, while a function of the unknown parameter θ, is a scaled unity matrix [10],

$$R(\theta) = r(\theta)\ I_{N\times N}$$

in which $r(\theta)$ is a scalar. In the flight control application discussed in the sequel, the equation error covariance matrix $R(\theta)$ is a scalar multiple of an identity matrix as shown in Section 2.3.1. From Theorem 3, the parameter estimate reduces to:

$$\hat{\theta} = (H^T H)^{-1} H^T Z + \frac{1}{e^T [I - H(H^T H)^{-1} H^T] e} (H^T H)^{-1} H^T e e^T \left(H(H^T H)^{-1} H^T - I \right) Z \ .$$

The unknown scalar $r(\theta)$ does not impact the parameter estimate and no iterations are required. From Theorem 3, the parameter estimation error covariance reduces to

$$P_\theta = r(\hat{\theta}) \left[(H^T H)^{-1} + \frac{(H^T H)^{-1} H^T e e^T H (H^T H)^{-1}}{e^T \left[I - H(H^T H)^{-1} H^T \right] e} \right]$$

and it is easily obtained once $\hat{\theta}$ is calculated.

The function $r(\theta)$ of the parameter θ is application-specific. The stability and control derivative's estimation process for the pitch channel of an F-16 class aircraft is considered. The focus of this chapter is on the short period dynamics, ignoring the phugoid and elastic modes. Keeping the length of the system identification window sufficiently short validates this assumption. The "$\dot{q}$ equation" is

$$\dot{q} = M_\alpha \cdot \alpha + M_q \cdot q + M_\delta \cdot \delta \tag{5.1}$$

in which α is the angle of attack, q is the pitch rate, δ is the elevator deflection, M_α and M_q are the stability derivatives, while M_δ is the control derivative. The measured variables, denoted by the subscript m, are $\alpha_m = \alpha + v_\alpha$, $q_m = q + v_q$, $\delta_m = \delta + v_\delta$, and the measured pitch acceleration is $\dot{q}_m = \dot{q} + v_{\dot{q}}$. The v quantities denote white, zero-mean Gaussian-distributed noise on the measured variables: $\mathbf{E}[v_\alpha^2] = \sigma_\alpha^2$, $\mathbf{E}[v_q^2] = \sigma_q^2$, $\mathbf{E}[v_\delta^2] = \sigma_\delta^2$, and $\mathbf{E}[v_{\dot{q}}^2] = \sigma_{\dot{q}}^2$. Substituting these measured quantities into (5.1) yields

$$\dot{q}_m = M_\alpha \alpha_m + M_q q_m + M_\delta \delta_m - M_\alpha v_\alpha - (M_q v_q + M_\delta v_\delta - v_{\dot{q}}) \quad . \tag{5.2}$$

The complete linear regression is considered in detail later in Section 5.3. For now, one should consider the equation error vector for the accumulated data record over time, which is pertinent to the crucial calculation of the equation error covariance matrix R. The equation error V is the vector

$$V = \begin{pmatrix} M_\alpha v_\alpha(0) - M_q v_q(0) - M_\delta v_\delta(0) + v_{\dot{q}}(0) \\ M_\alpha v_\alpha(\Delta T) - M_q v_q(\Delta T) - M_\delta v_\delta(\Delta T) + v_{\dot{q}}(\Delta T) \\ \vdots \\ M_\alpha v_\alpha((N-1)\Delta T) - M_q v_q((N-1)\Delta T) - M_\delta v_\delta((N-1)\Delta T) + v_{\dot{q}}((N-1)\Delta T) \end{pmatrix} . \tag{5.3}$$

The equation error covariance matrix assumes independence of individual measurement noises and is evaluated as

$$
\begin{aligned}
R &= \mathbf{E}[VV^T] \\
&= (M_\alpha^2\sigma_\alpha^2 + M_q^2\sigma_q^2 + M_\delta^2\sigma_\delta^2 + \sigma_{\dot{q}}^2) \cdot I_N \\
&= r(\theta) \cdot I_N
\end{aligned}
$$

in which

$$
\begin{aligned}
r(\theta) &= (M_\alpha^2\sigma_\alpha^2 + M_q^2\sigma_q^2 + M_\delta^2\sigma_\delta^2 + \sigma_{\dot{q}}^2) \\
&= \theta^T \begin{bmatrix} \sigma_\alpha^2 & 0 & 0 \\ 0 & \sigma_q^2 & 0 \\ 0 & 0 & \sigma_\delta^2 \end{bmatrix} \theta + \sigma_{\dot{q}}^2 \quad .
\end{aligned}
$$

The direct calculation of the parameter estimate without the need of iteration is a key feature of static system identification. The calculated parameter estimate is then used in the calculation of the predicted parameter estimation error covariance. Furthermore, the standard deviation $\sigma \triangleq \sqrt{r(\theta)}$ is estimated from the measurement record according to [10]

$$
\hat{\sigma} = \sqrt{\left((\tilde{Z}^T\tilde{Z})/(N-n) \right)}
$$

in which $\tilde{Z}$, the return difference, is $\tilde{Z} = Z - H\hat{\theta}$. N is the number of measurements and n is the number of parameters, in which $N > n$. Hence, the data driven estimate of the filter-predicted estimation error covariance matrix for a sliding window batch process is

$$
P = \hat{\sigma}^2 \left(H^T H \right)^{-1} \quad .
$$

One should note that static system identification requires the measurement of the aircraft's pitch acceleration, $\dot{q}$.

Table 5.1: Stability and Control Derivatives. Flight Condition: h = 10,000 ft., M = 0.7. Healthy Aircraft.

$M_\alpha = 3.724$	$M_q = -1.26$	$M_\delta = -19.5$
$Z_\alpha = -1.15$	$Z_q = 0.9937$	$Z_\delta = -0.1770$

5.2 Flight Control System

The parameter estimation process and the design of an adaptive and reconfigurable pitch axis flight control system is demonstrated in the context of a F-16 class aircraft. The focus is on the short period dynamics, ignoring the phugoid and elastic modes. Keeping the length of the system identification window sufficiently short validates this assumption. The dynamics are

$$\begin{aligned} \dot{\alpha} &= Z_\alpha \cdot \alpha + Z_q \cdot q + Z_\delta \cdot \delta \\ \dot{q} &= M_\alpha \cdot \alpha + M_q \cdot q + M_\delta \cdot \delta \end{aligned} \tag{5.4}$$

as stated before, and α is the angle of attack, q is the pitch rate, δ is the elevator deflection, M_α, M_q, Z_α, and Z_q are the stability derivatives, and M_δ and Z_δ are the control derivatives. The truth model stability and control derivatives for a healthy F-16 aircraft and the flight condition $h = 10000$ ft and $M = 0.7$ [10] are given in Table 5.1. One should note that the stability derivative M_α is positive. Hence, the bare aircraft pitch channel is not stable and one depends on feedback control for stabilization.

The actuator dynamics are modelled as a first order lag with a 20 $\frac{rad}{sec}$ corner frequency. That is, the actuator transfer function is

$$A(s) \ = \frac{20}{s+20} \quad .$$

The actuator dynamics are incorporated into the state space truth model by augmenting the dynamics (5.4) with the differential equation

$$\dot{\delta_e} = -\frac{1}{\tau}\delta + \frac{1}{\tau}\delta_c \tag{5.5}$$

in which $\tau = \frac{1}{20}$ seconds and the subscript c denotes the commanded elevator deflection.

The latter is determined by the control law

$$\delta_c = K_P \cdot (q_c - q_m) + K_I \frac{1}{s} \cdot (q_c - q_m) + K_\alpha \cdot \alpha_m \tag{5.6}$$

in which q_c is the commanded pitch rate, q_m is the measured pitch rate, K_P is the proportional gain, K_I is the integral gain, and K_α is a linear gain applied to the angle of attack for the purpose of damping the pitch dynamics. The subscript m denotes measured quantities, that is

$$\alpha_m \triangleq \alpha + v_\alpha \tag{5.7}$$

$$q_m \triangleq q + v_q \tag{5.8}$$

in which v_α and v_q are zero-mean, white Gaussian-distributed measurement noise. The measurement error statistics are $\sigma_\alpha = 0.1$ deg and $\sigma_q = 0.02$ deg/s [16]. One now rewrites the control law (5.6) in terms of the truth variables

$$\delta_c = K_P \cdot (q_c - q - v_q) + K_I \frac{1}{s}(q_c - q - v_q) + K_\alpha \cdot (\alpha + v_\alpha) \quad . \tag{5.9}$$

An additional differential equation is appended to the truth model using the "integrator charge" variable z to capture integral action in state space

$$\dot{z} = q_c - q_m \quad .$$

Again, incorporate (5.8) and rewrite the $\dot{z}$ equation in terms of truth variables

$$\dot{z} = q_c - q - v_q \quad .$$

As demonstrated later, feed-forward control obviates the need for sizable integral action and a reduced K_I gain will suffice, thus precluding integrator windup.

In order to incorporate the controller (5.6) into the state space model of the flight control system, one should incorporate (5.9) directly into (5.5) using the defined variable z

$$\begin{aligned}\dot{\delta} &= -\frac{1}{\tau}\delta + \frac{1}{\tau}\left[K_P \cdot (q_c - q - v_q) + K_I \cdot z + K_\alpha \cdot (\alpha + v_\alpha)\right] \\ &= -\frac{1}{\tau}\delta - \frac{K_P}{\tau}q + \frac{K_I}{\tau}z + \frac{K_\alpha}{\tau}\alpha + \frac{K_P}{\tau}q_c - \frac{K_P}{\tau}v_q + \frac{K_\alpha}{\tau}v_\alpha \quad .\end{aligned}$$

Figure 5.1 shows the inner loop of the pitch control channel. Typically, when α

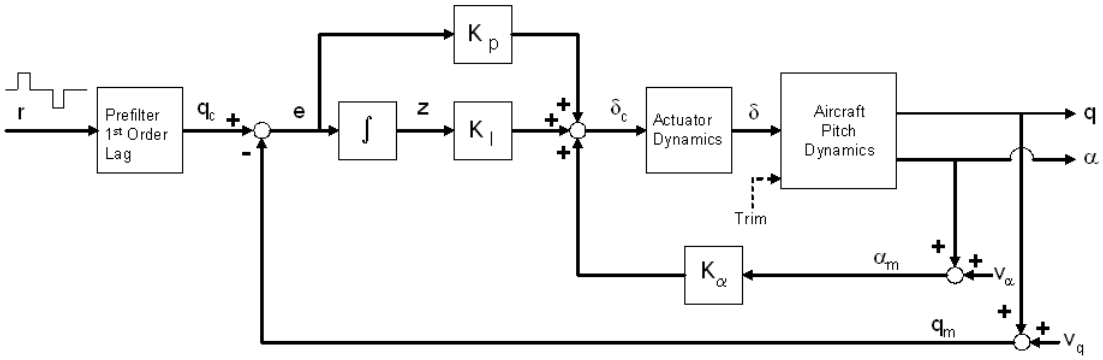


Figure 5.1: Typical F-16 class aircraft inner loop flight control system for pitch plane.

feedback is included in a pitch rate control augmentation system, the measured signal is filtered with a low-pass, low-order filter before application to the PI controller [30]. This filtering is typically used because the measured alpha signal is typically relatively noisy. In this research, a filtered signal is not used. The important thing to know for later system identification is that one must remain consistent in using measured variables. If the α variable is filtered, one must filter the other inputs to the regressor, namely q and δ, and the measurement vector consisting of $\dot{q}$, in the same manner. Failing to do so will introduce lags to some variables and not to others,

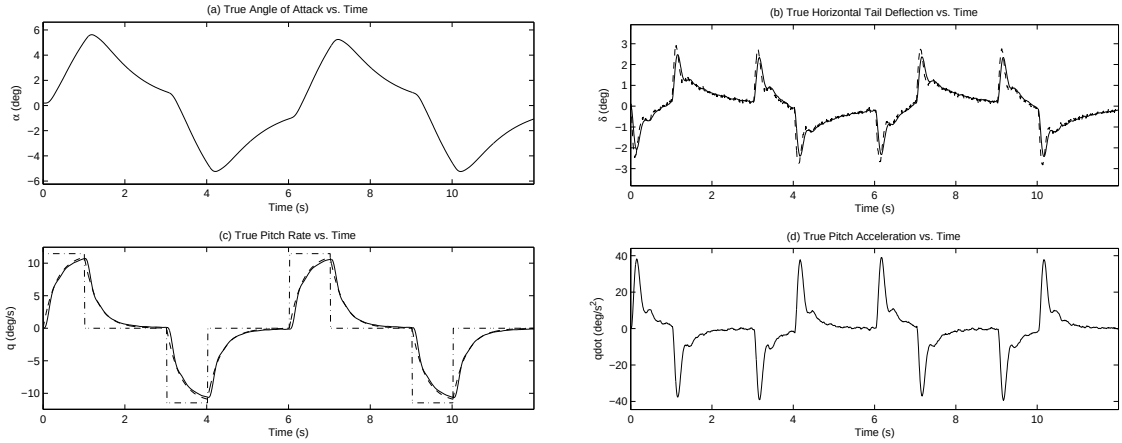


Figure 5.2: Short Period Dynamics for Healthy Aircraft. Solid lines represent the truth model variables, namely α, q, δ, and $\dot{q}$. In (b), the dashed line indicates the commanded elevator deflection. In (c), the dash-dot line indicates the pitch rate command, while the dashed line indicates the prefiltered pitch rate command.

which deleteriously effects the parameter estimate's quality. The method relies on batch size to wash-out the measurement noise effects. In this research, variables are not filtered, whether for flight control or system identification. The reconfigurable flight control system relies on batch size to wash-out the measurement noise effects.

The current flight control system of operational F-16 aircraft uses a PI control law with minimal gain scheduling. As in [16], pilot pitch rate doublet commands are considered that persist as 1.0 second duration pulses with a 0.2 rad/s (11.46 deg/s) amplitude and with input polarities of +, -, +, and - at 0.0, 3.0, 6.0, and 9.0 seconds respectively. This is depicted by the dash-dot line in Figure 5.2(c). This subfigure also shows the pilot input r that is prefiltered with $F(s) = \frac{3}{s+3}$ (solid line). So,

$$\dot{q}_c = -3q_c(t) + 3r(t)$$

in which $r(t)$ is the square wave input from the pilot and q_c is the filtered pilot command input. The sampling rate of the controller is 100Hz.

In summary, the dynamics equations that describe the pitch channel are:

Table 5.2: Controller Gains

K_P	K_I	K_α
-1.1912	-3.45032	0.4

$$
\begin{aligned}
\dot{\alpha} &= Z_\alpha \cdot \alpha + Z_q \cdot q + Z_\delta \cdot \delta \\
\dot{q} &= M_\alpha \cdot \alpha + M_q \cdot q + M_\delta \cdot \delta \\
\dot{\delta} &= -\frac{1}{\tau}\delta + \frac{1}{\tau}\left[K_P \cdot (q_c - q - v_q) + K_I \cdot z + K_\alpha \cdot (\alpha - v_\alpha)\right] \\
\dot{z} &= q_c - q - v_q \\
\dot{q}_c &= -3q_c(t) + 3r(t)
\end{aligned}
\tag{5.10}
$$

Hence, in matrix notation, the (healthy) aircraft flight control system is

$$\dot{x} = Ax + bu + \Gamma w$$

in which $x = \begin{pmatrix} \alpha \\ q \\ \delta \\ z \\ q_c \end{pmatrix}$, $A = \begin{bmatrix} Z_\alpha & Z_q & Z_\delta & 0 & 0 \\ M_\alpha & M_q & M_\delta & 0 & 0 \\ K_\alpha & -K_P & K_I & -\frac{1}{\tau} & \frac{1}{\tau} \\ 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 3 \end{bmatrix}$, $b = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 3 \end{pmatrix}$, $\Gamma = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ K_\alpha & -\frac{K_P}{\tau} \\ 0 & -1 \\ 0 & 0 \end{bmatrix}$,

and $w = \begin{pmatrix} v_\alpha \\ v_q \end{pmatrix}$.

The gains of the inner loop controller were optimized to obtain good tracking performance, that is, to minimize the variation in pitch rate q from the commanded pitch rate q_c. These gains are given in Table 5.2. The variables of interest are shown in Figure 5.2.

Table 5.3: Stability and Control Derivatives. Failed Condition

$M_{\alpha_F} = 7.448$	$M_{q_F} = -0.84$	$M_{\delta_F} = -9.75$
$Z_{\alpha_F} = -2.3$	$Z_{q_F} = 0.99685$	$Z_{\delta_F} = -0.0885$

Failure

At this point, the failure is introduced – a 50% control effector area loss, that is, a 50% horizontal stabilator control surface area loss. The failure is introduced at 5.0 seconds into the flight, causing two significant changes. First, the stability and control derivatives change, as shown in Table 5.3. The loss of a control surface area causes both the stability and control derivatives to change, as opposed to an actuator failure, which would cause only the control derivatives to change. Furthermore, the loss of a horizontal stabilator increases the degree of open loop aircraft instability, as the stability derivative M_α increases from +3.724 to +7.448. Thus, a particularly challenging control problem is addressed here.

Second, a change in trim occurs. One can model the change in trim as follows:

$$F_{tail} \cdot l_t = mg \cdot \Delta x_{CG}$$

and

$$F_{tail} = \bar{q} \cdot \eta \cdot S_t \cdot \delta_{trim} \quad .$$

The variables are defined as follows: F_{tail} is the aerodynamic force exerted on the tail, l_t is the distance between the aircraft's center of gravity and the tail, m is the mass of the aircraft, g is the acceleration of gravity, Δx_{CG} is the distance between the aircraft's aerodynamic center and center of mass, $\bar{q}$ is the dynamic pressure, η is an aerodynamic efficiency coefficient, which captures the fact that the airflow at the tail is less energetic after it has been slowed down by friction while flowing over the wing, and S_t is the tail's surface area. Hence, the disturbance is

$$d \triangleq \delta_{\text{trim}} = \frac{mg \cdot \Delta x_{CG}}{l_t \cdot \bar{q} \cdot \eta \cdot S_t} \quad .$$

For the flight condition considered, the unfailed horizontal tail setting for trim is 2^o. The loss of one horizontal tail is therefore modelled by a 1^o elevator deflection, bearing in mind that, after the failure, the control effectors' surface area has been reduced by 50%. Hence, at the flight condition under consideration, the change in trim resulting from the failure is $\delta_d \simeq -1.0$ degrees. This value is used for the first series of tests that involve a sizeable trim change. Figure 5.3 depicts the revised F-16 aircraft simulation, in which δ_d is a step elevator setting change that models the change in trim caused by the control surface area loss due to the failure, which occurs at 5.0 seconds into the flight.

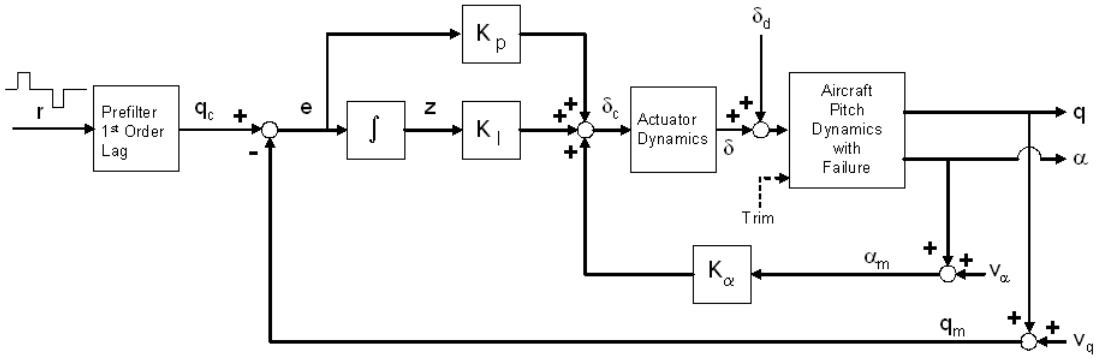


Figure 5.3: Inner loop flight control system for the pitch channel. Trim change due to 50% horizontal stabilator area loss is modelled with the input disturbance $d = \delta_d$.

The differential equations for the failed aircraft model are:

$$
\begin{aligned}
\dot{\alpha} &= Z_{\alpha_F} \cdot \alpha + Z_{q_F} \cdot q + Z_{\delta_F} \cdot \delta + Z_{\delta_F} \cdot \delta_d \\
\dot{q} &= M_{\alpha_F} \cdot \alpha + M_{q_F} \cdot q + M_{\delta_F} \cdot \delta + M_{\delta_F} \cdot \delta_d \\
\dot{\delta} &= -\frac{1}{\tau}\delta + \frac{1}{\tau}\left(K_P \cdot (q_c - q - v_q) + K_I \cdot z + K_\alpha \cdot (\alpha + v_\alpha)\right) \\
\dot{z} &= q_c - q - v_q \\
\dot{q}_c &= -3q_c(t) + 3r(t)
\end{aligned}
$$

The equations are rewritten as follows:

$$\dot{x} = Ax + bu + \Gamma w + \Gamma_d \, d$$

in which $A = \begin{bmatrix} Z_{\alpha_F} & Z_{q_F} & Z_{\delta_F} & 0 & 0 \\ M_{\alpha_F} & M_{q_F} & M_{\delta_F} & 0 & 0 \\ K_\alpha & -K_P & K_I & -\frac{1}{\tau} & \frac{1}{\tau} \\ 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -3 \end{bmatrix}$, $x = \begin{pmatrix} \alpha \\ q \\ \delta \\ z \\ q_c \end{pmatrix}$, $b = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 3 \end{pmatrix}$, $\Gamma = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ K_\alpha & -\frac{K_p}{\tau} \\ 0 & -1 \\ 0 & 0 \end{bmatrix}$,

$w = \begin{pmatrix} v_\alpha \\ v_q \end{pmatrix}$, and $\Gamma_d = \begin{bmatrix} Z_{\delta_F} \\ M_{\delta_F} \\ 0 \\ 0 \\ 0 \end{bmatrix}$.

Figure 5.4 depicts the flight control system dynamics when the failure is encountered at 5 seconds into the flight. The onset of failure triggers the reconfiguration of the controller, such as readjusting the system identification window length and subsequently the inner loop's control gains. A retrimming signal is also generated.

In reality, one does not have access to the truth model; rather, one only has access to the sensor measurements. Two more measurements of interest, elevator deflection δ_e and pitch acceleration $\dot{q}$ are now discussed. Table 5.4 includes the in-

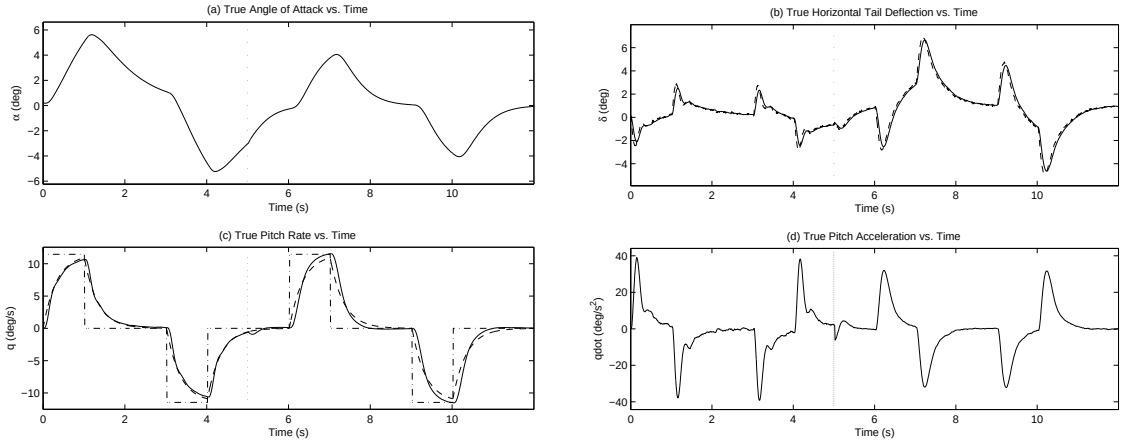


Figure 5.4: Short period dynamics. Failure is induced at $t_f = 5$s (vertical dotted lines). See the Figure 5.2 caption for line type description.

Table 5.4: Noise Statistics

σ_δ	σ_α	σ_q	$\sigma_{\dot{q}}$
0.08deg	0.1deg	0.02deg/s	1.3deg/s^2

formation provided in [16], which was used in (5.7) and (5.8). For completeness, Figures 5.5 through 5.8 depict the truth variables, as shown in Figure 5.4, along with the corresponding measured variables. One of the main points of showing all of these plots is that only the pitch acceleration measurement $\dot{q}_m$ in Figure 5.8(b) clearly signals the failure occurrence among the measured variables [10]. The magnitude of the jump varies depending on the magnitude of the trim change, as reflected in δ_d. Careful simulation analysis shows that the jump in $\dot{q}_m$ takes 0.03 seconds to develop. This realization is important to the Failure Detection Identification (FDI) module, which will eventually trigger the post-failure system identification (described in Sections 5.3, 5.4.1 through 5.4.3) and the feed-forward control algorithm (described in the remainder of Section 5.3) used to cope with the control surface failure. As previously mentioned, the controller sampling rate is 100Hz. Hence, after three time steps, the jump in $\dot{q}$ is evident and is used to readjust the identification window length.

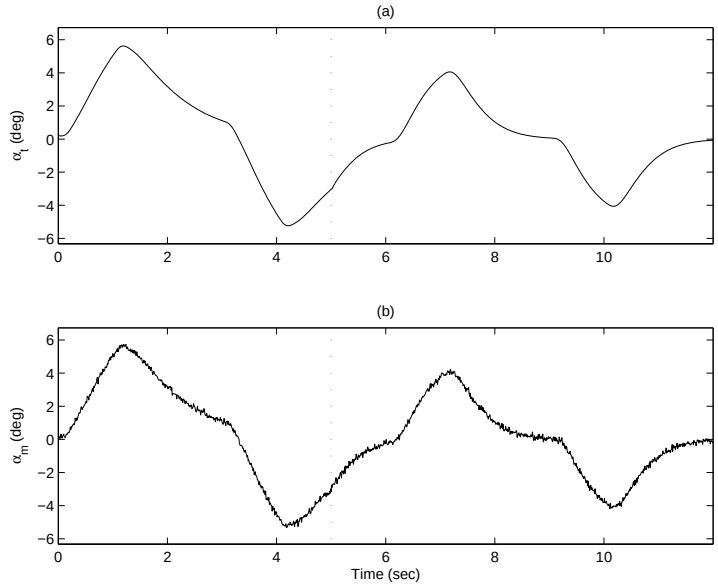


Figure 5.5: (a) Angle of attack response for pitch rate doublet input (b) Measured angle of attack includes error with $\sigma_\alpha = 0.1$ deg. In both subplots, the vertical dotted line indicate the onset of control surface failure.

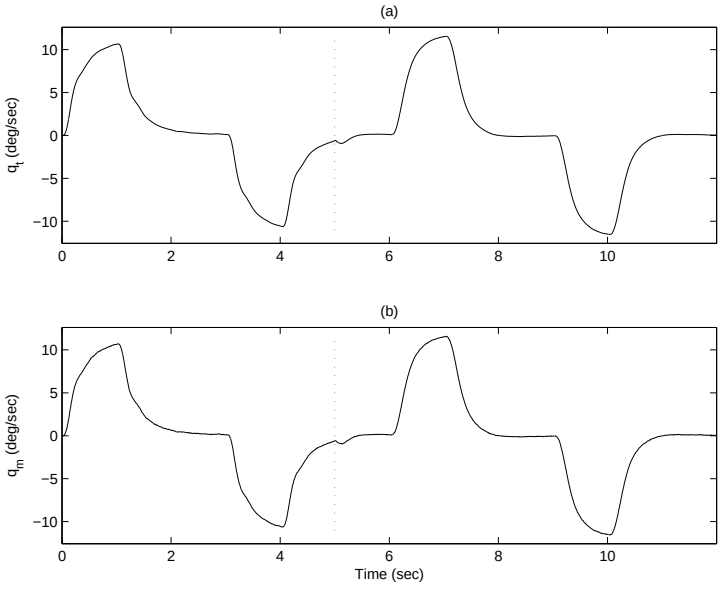


Figure 5.6: (a) Pitch rate response for pitch rate doublet input (b) Measured pitch rate includes error with $\sigma_q = 0.02$ deg/s. In both subplots, the vertical dotted line indicate the onset of control surface failure.

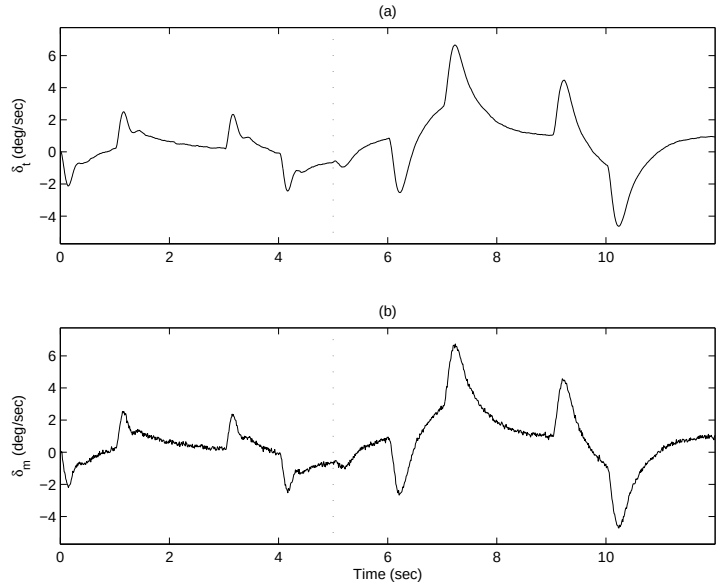


Figure 5.7: (a) Elevator deflection for pitch rate doublet input (b) Measured elevator deflection includes error with $\sigma_\delta = 0.08$ deg. In both subplots, the vertical dotted line indicate the onset of control surface failure.

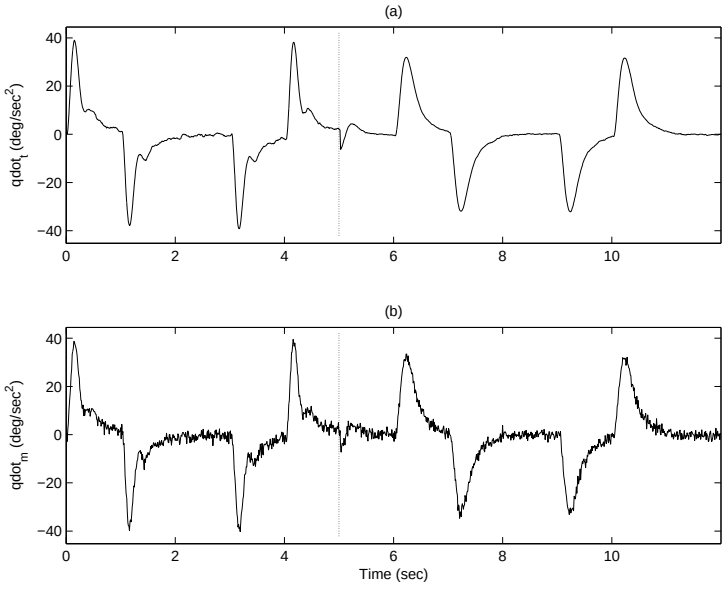


Figure 5.8: (a) Pitch acceleration for pitch rate doublet input (b) Measured pitch acceleration includes error with $\sigma_{\dot{q}} = 1.3$ deg/s^2. In both subplots, the vertical dotted line indicate the onset of control surface failure.

5.3 System Identification

As discussed in Section 5.1, the attention is confined to the pitch axis and the short period dynamics. Equation (5.2) is obtained by substituting the measured quantities into (5.1). The augmented linear regression (1.1) is considered and Theorem 3 is applied. The measurement vector is composed as follows

$$Z_{N\times 1} \triangleq \begin{pmatrix} \dot{q}_m(0) \\ \dot{q}_m(\Delta T) \\ \vdots \\ \dot{q}_m((N-1)\Delta T) \end{pmatrix} \quad .$$

The regressor matrix is

$$H_{N\times 3} \triangleq \begin{bmatrix} \alpha_m(0) & q_m(0) & \delta_m(0) \\ \alpha_m(\Delta T) & q_m(\Delta T) & \delta_m(\Delta T) \\ \vdots & \vdots & \vdots \\ \alpha_m((N-1)\Delta T) & q_m((N-1)\Delta T) & \delta_m((N-1)\Delta T) \end{bmatrix} \quad .$$

The parameter vector is

$$\theta = (M_\alpha, M_q, M_\delta)^T \in \mathbb{R}^3 \quad .$$

The equation error V given by (5.3) was discussed in Section 5.1.

There is interest in estimating the intercept c, which absorbs the unknown trim condition. As a result of the failure, the intercept changes abruptly. The algorithm presented in Theorem 3 is used to estimate the stability and control derivatives and the intercept c. Using the stability and control derivative estimates, the controller's gains are adjusted on-line to preserve the bandwidth of the closed loop control system. The feed-forward command

$$\delta_t = \frac{\Delta \hat{c}}{M'_\delta}$$

is also calculated. The complete adaptive and reconfigurable flight control system is shown in Figure 5.9.

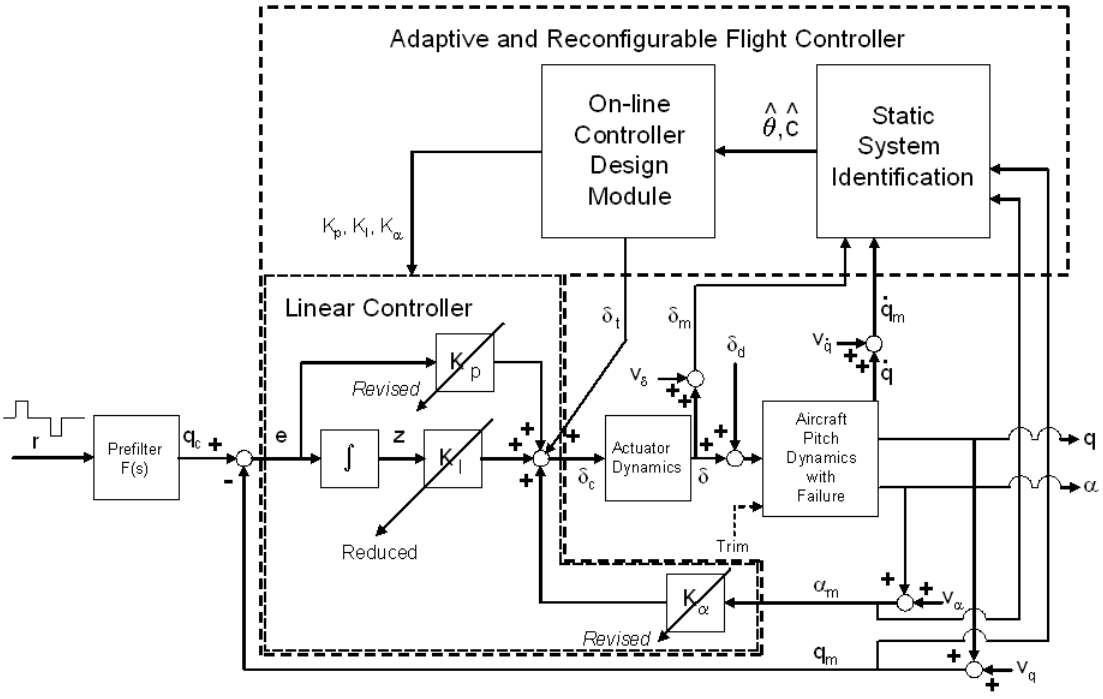


Figure 5.9: Adaptive and reconfigurable flight control system. System identification is used to estimate the stability and control derivatives and the intercept. This information is given to the online controller design module, which recalculates the control gains and a feed-forward control signal δ_t that mitigates a possible change in trim caused by failure.

5.4 Simulation

Pitch doublets over a 12 second time interval are applied. The failure at 5 seconds into the flight entails the loss of one of the horizontal stabilators, thus reducing control surface effectiveness by 50%. This failure also causes a change in the stability and control derivatives and the application of an input disturbance signal. The truth trajectory and clean measurement data vectors are generated. Zero-mean,

Table 5.5: Inner Loop Controller Gains

K_{P_F}	K_{I_F}	K_{α_F}
-1.7868	0	0.2

Gaussian-distributed noise is added to the measurements using the variances shown in Table 5.4. The measurement errors used in the simulation experiments reflect current state-of-the-art hardware specifications. Sensor measurements for α, q, δ, and $\dot{q}$ are assumed independent at each sample, and are taken at a rate of 100 Hz. This, in turn, affords static system identification, i.e., linear regression. At each time increment, the parameter estimate is derived in batch using either an expanding or a sliding data window. The data window is reset when the failure occurs. This is tripped by a jump in the measured $\dot{q}$ and/or a jump in the running intercept estimate $\hat{c}$ provided by the system identification module.

Current aircraft typically rely on inherent flight control system robustness to maintain stability during unanticipated disturbances, achieved primarily from integral action provided by a Type-1 inner loop controller. One can compare the results from the conventional F-16 class flight control system that encounters the major control surface failure with the results from the reconfigurable flight control system discussed in this development. One wants to "reject" the disturbance resulting from a 50% loss of control surface. Hence, one wants to double the effective proportional gain K_P associated with a "tuned" Type-0 flight control system at the onset of failure. So, the inner loop controller optimization development generated from (5.10) and summarized in Table 5.2 is revisited. Table 5.5 shows the appropriate control gains for a tuned Type-0 controller, which implies that $K_I = 0$. Next, one feeds forward the control derivative estimate information after control surface failure in order to adjust the proportional gain K_{P_F} as follows

$$K_{P_F} \triangleq K_{P_{UF}} \cdot \frac{\hat{M}_{\delta_{UF}}}{\hat{M}_{\delta_F}} \tag{5.11}$$

in which the subscript F indicates the value of the proportional gain and control derivative after failure and UF indicates the value of the proportional gain and control derivative before failure.

Throttling the proportional gain K_P according to the system identification module-provided plant information is the most important contributor. In fact, Section 5.4.4 presents simulation results from two adaptive control algorithms that solely use on-line K_P gain adjustment, while the pitch damping gain K_α is set to the non-failed Type-0 control gain after control surface failure and the integral action control gain K_I is reduced to zero. The impact of adaptively setting the damping gain K_α is also explored, in addition to K_P in Section 5.4.5. One wishes to ensure that the aircraft flight control system damping authority is the same before and after the onset of the control surface failure. Hence,

$$M_{\alpha_{UF}} \cdot \alpha + M_{\delta_{UF}} K_{\alpha_{UF}} \cdot \alpha = M_{\alpha_F} \cdot \alpha + M_{\delta_F} K_{\alpha_F} \cdot \alpha \quad .$$

This implies that

$$\begin{aligned} K_{\alpha_F} &= \frac{M_{\alpha_{UF}} + M_{\delta_{UF}} K_{\alpha_{UF}} - M_{\alpha_F}}{M_{\delta_F}} \\ K_{\alpha_F} &= \frac{M_{\alpha_{UF}} - M_{\alpha_F}}{M_{\delta_F}} + \frac{M_{\delta_{UF}}}{M_{\delta_F}} K_{\alpha_{UF}} \end{aligned} \tag{5.12}$$

as previously stated, the subscript F indicates values after control surface failure and UF indicates values before failure. As Section 5.4.5 reports, implementation strategy is important when on-line adjustments (5.11) and (5.12) are applied to K_{P_F} and K_{α_F}, respectively.

5.4.1 System Identification: Unfailed Baseline Case. First, system identification results are shown for the F-16 class aircraft using a pitch doublet input over the 12 second interval, unimpeded by failure. An expanding data window is used in this section to produce parameter estimates for comparison with results presented in Sections 5.4.2 and 5.4.3, in which the failure is modelled. Figures 5.10 through 5.12

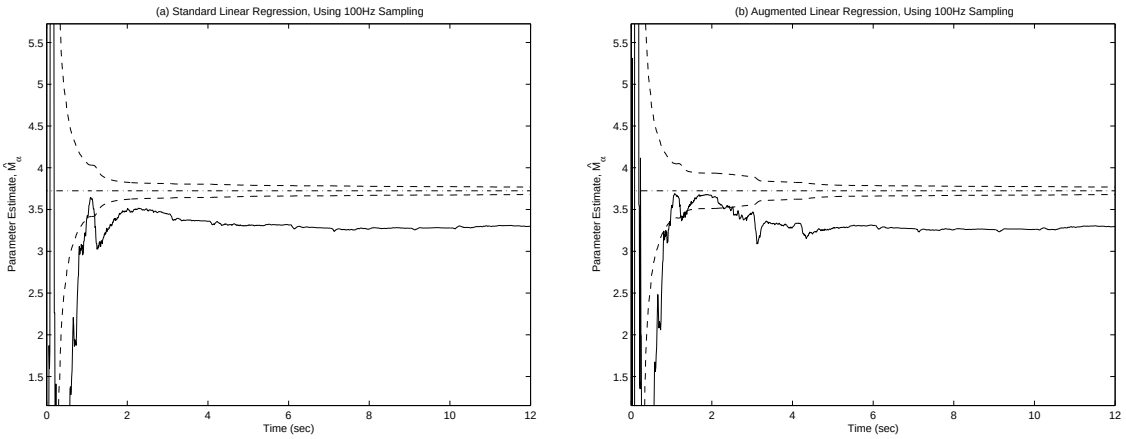


Figure 5.10: System identification of stability derivative parameter $\hat{M}_\alpha$ using expanding window batch process. No failure case. (a) Uses standard linear regression, (b) Uses novel augmented linear regression method derived in Section 3.1. Solid lines–estimates, Dash-dot lines–true parameter value about which the 1-σ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

show the parameter estimates for this case. It is important to note that each of the stability and control estimates are biased. Interestingly, the magnitude of the estimate bias scales proportionally with the magnitude of the stability and control derivative. For example, when M_δ instantaneously adjusts with the onset of the control surface failure, Tables 5.1 and 5.3 reveal a 50% reduction in the control moment. Likewise, the system identification module shows a 50% reduction of the control moment bias. When considering this feed-forward application, one should notice that the magnitude of the proportional control gain K_{P_F} described in (5.11) is not impacted by the bias estimates produced by the system identification. The biased nature of the estimates will have a small impact on the feed-forward damping gain K_α described by (5.12); however, this impact is negligible.

When considering the augmented linear regression compared to the standard linear regression, no significant degradation of estimation performance is observed, even though there is no trim change to accommodate. This is important because in the unfailed case, the augmented estimation process is over-parameterized, which is usually catastrophic in system identification.

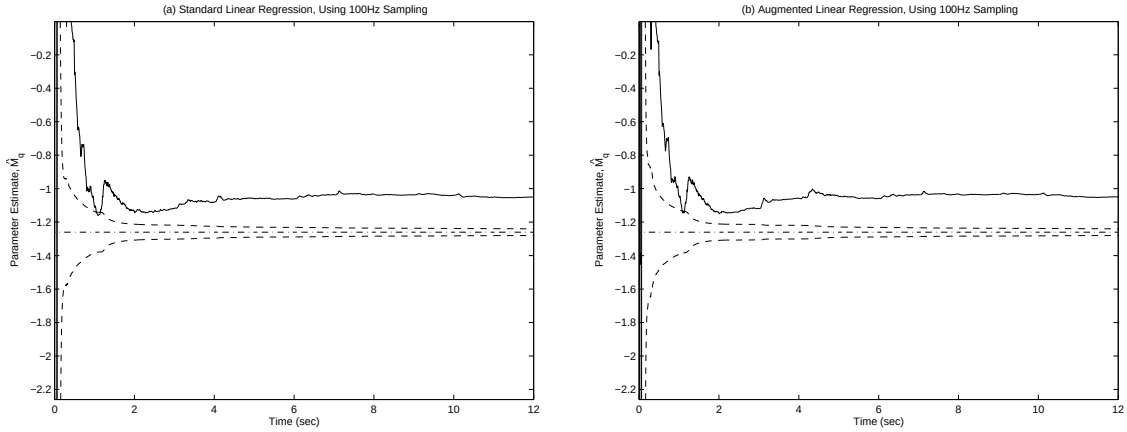


Figure 5.11: System identification of stability derivative parameter $\hat{M}_q$ using expanding window batch process. No failure case. (a) Uses standard linear regression, (b) Uses novel augmented linear regression method. Solid lines–estimates, Dash-dot lines–true parameter value about which the 1-σ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

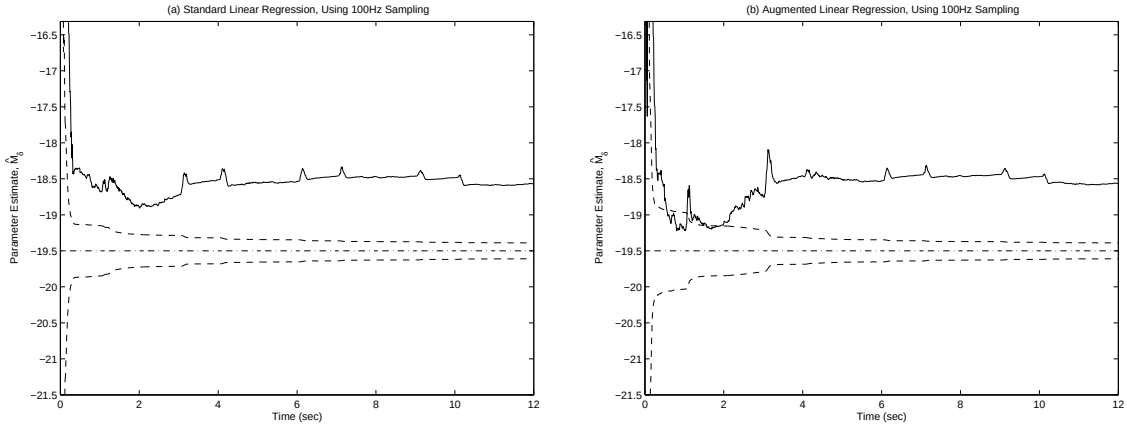


Figure 5.12: System identification of control derivative parameter $\hat{M}_\delta$ using expanding window batch process. No failure case. (a) Uses standard linear regression, (b) Uses novel augmented linear regression method. Solid lines–estimates, Dash-dot lines–true parameter value about which the 1-σ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

5.4.2 System Identification: Failure with Sizeable Trim Change. Next, system identification results are presented for the F-16 class aircraft subject to the same input. The failure occurs at $t = 5.0$ seconds and causes a significant trim change. Blindly using an expanding batch window as in Section 5.4.1 for estimation produces poor post-failure stability and control derivatives and intercept estimates. This would result because the measurements within the batch from before the failure onset would interfere with the system identification of the stabilty and control derivatives after the failure. The first method of system identification is now described. An expanding batch window is initially used for the first 225 measurements (2.25 seconds) and subsequently, a sliding batch window is used until failure detection. One can notice a minor degradation in parameter estimation performance between expanding window (Figures 5.10 and 5.12) and sliding window (Figures 5.13 and 5.15) system identification when comparing the time range of 2.25 to 5 seconds perfect failure trigger which allows one to freeze the estimates at $t = 5.0$s. This is how the simulation was actually conducted in this dissertation research. As previously noted, in practice, three time increments are required to acknowledge the trigger event. Hence, one should delay the parameter freeze by 0.03 seconds. The stability and control derivatives do not significantly vary between t = 5.0-5.03s, so this assumption is of little consequence. From the trigger point on, an expanding window is filled for the next 225 increments. The estimates are unfrozen after 1.4 seconds. This is the experimentally determined time duration required to obtain reliable estimates using the expanding window batch process after the failure is detected. So, in effect, an expanding window is used for estimation between 6.4 to 7.25 seconds. After 7.25 seconds, the sliding window is used for the remainder of the estimation process.

The simulation results are presented in Figures 5.13 through 5.16. When compared to the results presented in Section 5.4.1, one notes some minor degradation resulting from using a sliding window versus an expanding window between 2.25 to 5.0 seconds. When considering the estimation performance of the two methods after control surface failure and removal of the parameter freeze (i.e., after 6.4 seconds), the novel

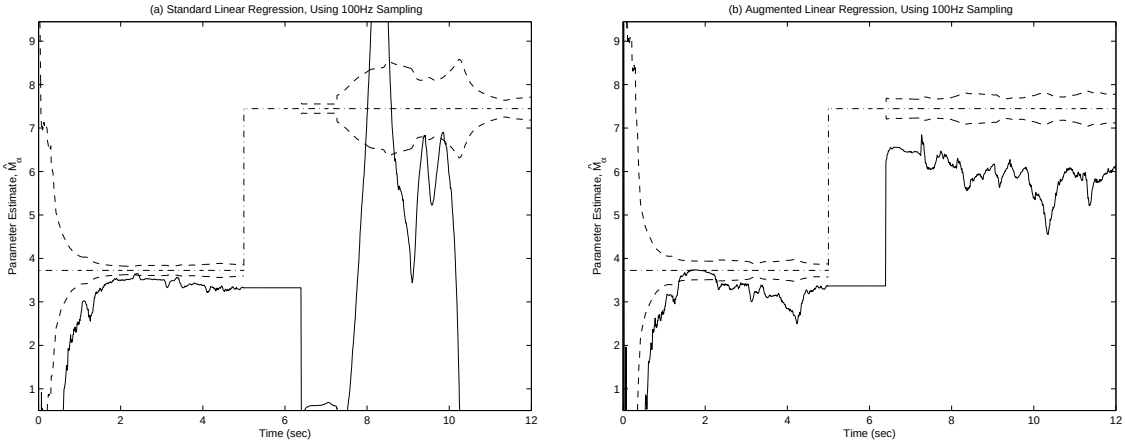


Figure 5.13: Estimate of stability derivative $\hat{M}_\alpha$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s. (a) Uses standard linear regression, (b) Uses novel augmented linear regression algorithm derived in Section 3.1. Solid lines–estimates, Dash-dot lines–true parameter value about which the 1-σ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

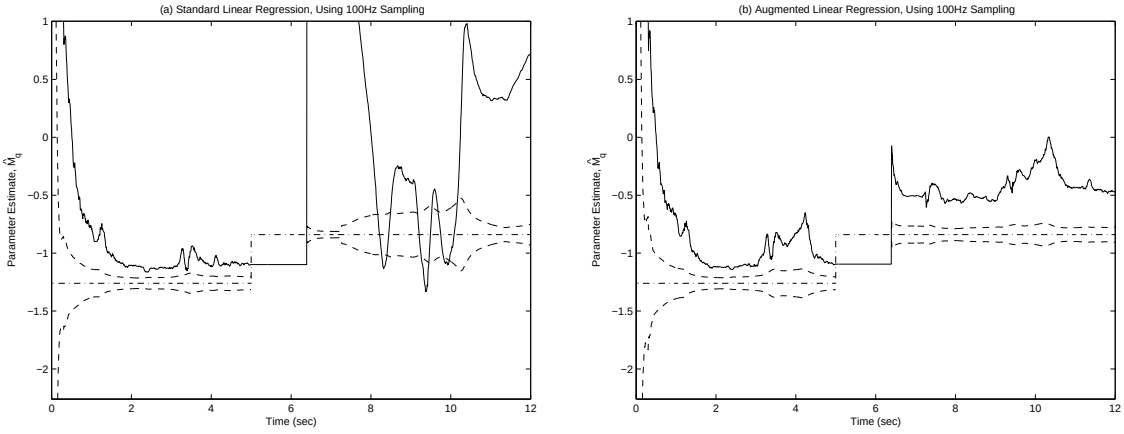


Figure 5.14: Estimate of stability derivative $\hat{M}_q$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s. (a) Uses standard linear regression, (b) Uses novel augmented linear regression algorithm. Solid lines–estimates, Dash-dot lines–true parameter value about which the 1-σ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

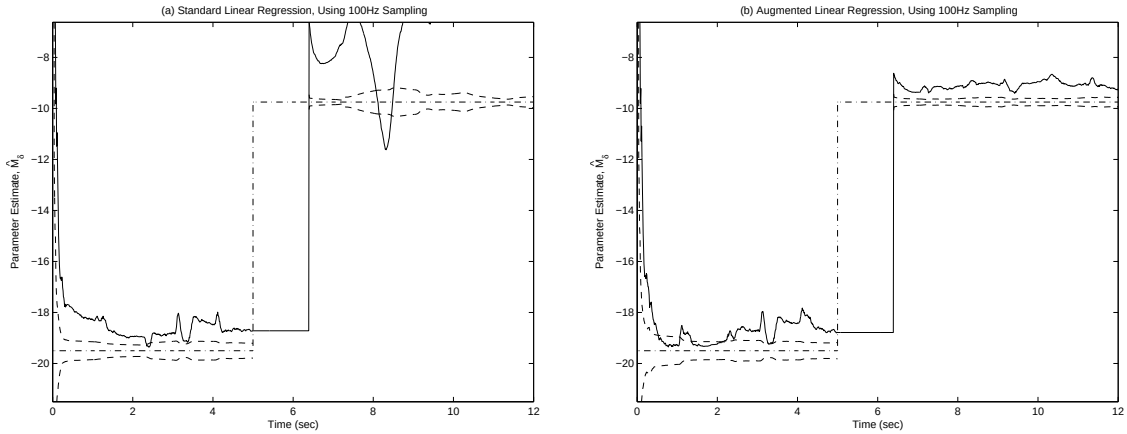


Figure 5.15: Estimate of control derivative $\hat{M}_\delta$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s. (a) Uses standard linear regression, (b) Uses novel augmented linear regression algorithm. Solid lines–estimates, Dash-dot lines–true parameter value about which the 1-σ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

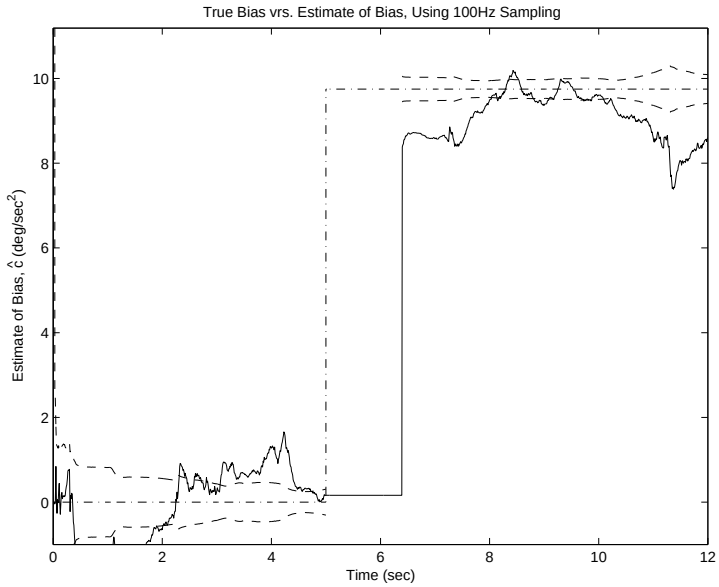


Figure 5.16: Estimate of the intercept $\hat{c}$ $(= \hat{M}'_\alpha * \delta_d)$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s. Solid line–estimate, Dash-dot line–true parameter value about which the 1-σ bounds filter-predicted equation error covariance (dashed lines) are depicted.

augmented linear regression is clearly superior. The intercept-augmented linear regression handles the change in trim of $\delta_d = 1.0^o$, while the standard linear regression is unable to provide reliable estimates of the stability and control derivatives and the intercept, even when a sliding window is used.

It is important to note that each of the stability and control estimates are biased, as recognized in Section 5.4.1. The magnitude of the estimate bias scales proportionally with the magnitude of the stability and control derivative. When considering this feed-forward application, one should notice that the magnitude of the proportional control gain K_{P_F} described in (5.11) is not impacted by the bias estimates produced by the system identification. This is the important control gain when considering aircraft stability and pitch rate tracking. The biased nature of the estimates will have a small impact on the feed-forward damping gain K_α described by (5.12); however, results presented in the later subsections will show that the impact is negligible.

There a reason to continue a sliding batch window for a short period after the trigger (currently $\dot{q}$ [10]) detects a failure within the failure detection identification module. One should consider the estimate of the bias shown in Figure 5.17 which demonstrates this scenario. There is a quick jump in the intercept estimate $\hat{c}$ before significant movement in the other three parameter estimates, one of which is shown in Figure 5.18. Additional stability derivative results are shown in Appendix I.1. In other words, $\hat{c}$ is more sensitive than the other parameter estimates. In fact, a sizeable jump in $\hat{c}$ is triggered at 0.10s after the failure. One can recall from previous explanation that some time must elapse in order to produce reliable estimates using the expanding window batch process after the failure is detected; hence, the parameter freeze is essential for the feed-forward control algorithm. This information is used as a corroborating trigger event, in addition to monitoring $\dot{q}$. One could design the estimation process to freeze the parameter estimates at the initial trigger event in $\dot{q}$.

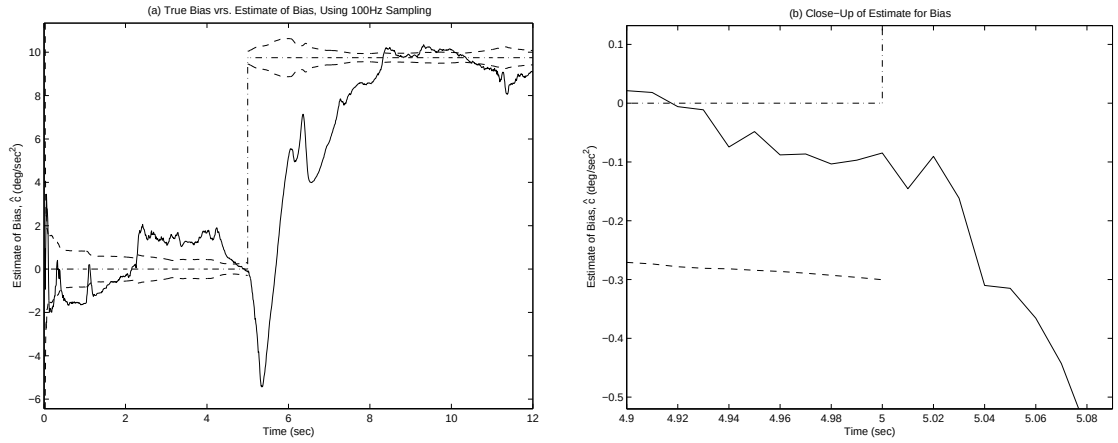


Figure 5.17: Estimate of the intercept $\hat{c}$ $(= \hat{M}'_{\alpha} * \delta_d)$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s. Sliding window is not reset after failure. Solid line–estimate, Dash-dot line–true parameter value about which the 1-σ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

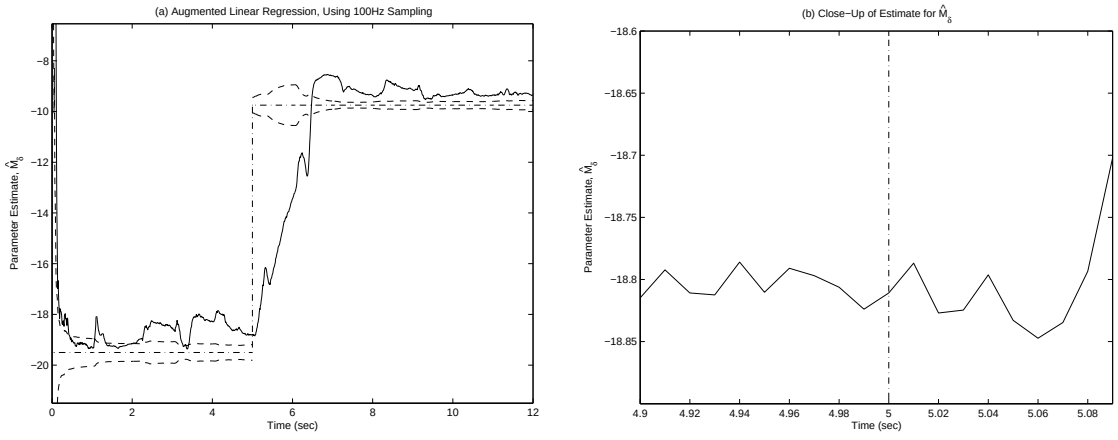


Figure 5.18: Estimate of control derivative $\hat{M}_{\delta}$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s. Sliding window is not reset after failure. Solid line–estimate, Dash-dot line–true parameter value about which the 1-σ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

Calculations continue to use the sliding batch window that monitors an unfrozen $\hat{c}$. In the next section, $\hat{c}$ may trend towards 0, depending on the size of δ_d. The results will show that this event is smoothly handled by standard linear regression and augmented linear regression. The trigger event for $\dot{q}$ is clearly identifiable for this case. However, for some other cases in which $\delta_d \neq 0$ or $\delta_d \neq -1.0$, the trigger event magnitude equals a jump in $\dot{q}$; hence, the failure trigger signature may not appear in the data. So, one could incorporate some decision logic into the failure detection module that monitors trigger events in $\dot{q}$ and $\hat{c}$, initiating the need for a reconfigured system identification window, with the actual delay ranging from 0.03 and 0.10 seconds.

5.4.3 Failure Case with No Trim Change. Next, one should consider the results of system identification using the pitch doublet input in the case in which no trim change occurs due to the failure. This situation happens at flight conditions in which the aircraft's aerodynamic center is very close to the aircraft's center of gravity and therefore the tail is not loaded. In this case, $\delta_d = 0$. The system identification window size adjustment scheme is the same as that previously described in Section 5.4.2. The results are presented in Figures 5.19 through 5.22. Both methods adequately estimate the stability and control derivative parameters. This result is important for the novel augmented linear regression method, in that over-parameterization does not adversely impact the estimation process. Also, there is a negligible impact of using the sliding window versus the expanding window in the unfailed case between 2.25 and 5.0 seconds.

The same argument is made about using $\hat{c}$ as a trigger for failure detection holds here, as it did in Section 5.4.2. This is extremely interesting, since there is no trim change induced during this special case failure. (See Figures 5.23 and 5.24. One should note that additional stability estimates are shown in Appendix I.2.) This methodology provides a valuable second corroborating event to back up the $\dot{q}$ trigger event for setting the data window noted earlier in [10].

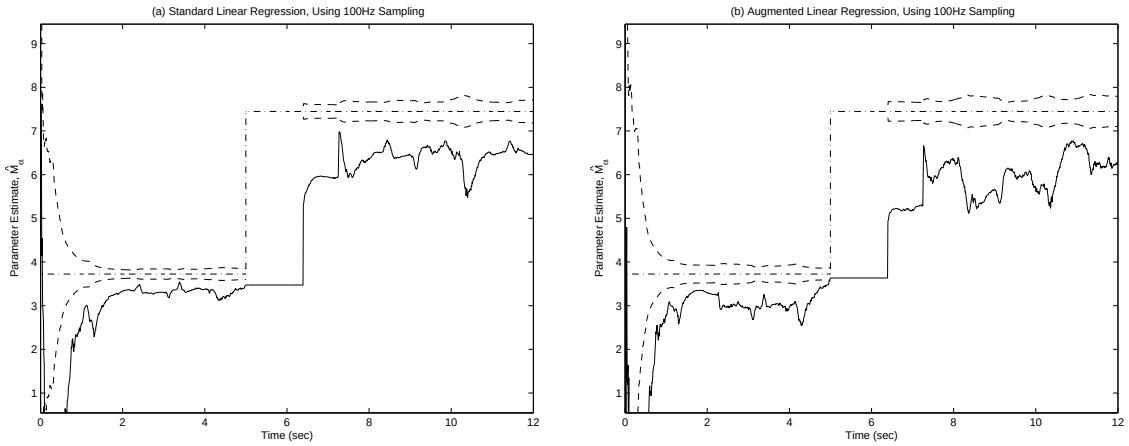


Figure 5.19: Estimate of stability derivative $\hat{M}_\alpha$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s, but there is no trim change. (a) Uses standard linear regression, (b) Uses novel augmented linear regression algorithm derived in Section 3.1. Solid lines–estimates, Dash-dot lines–true parameter value about which the $\pm 1\sigma$ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

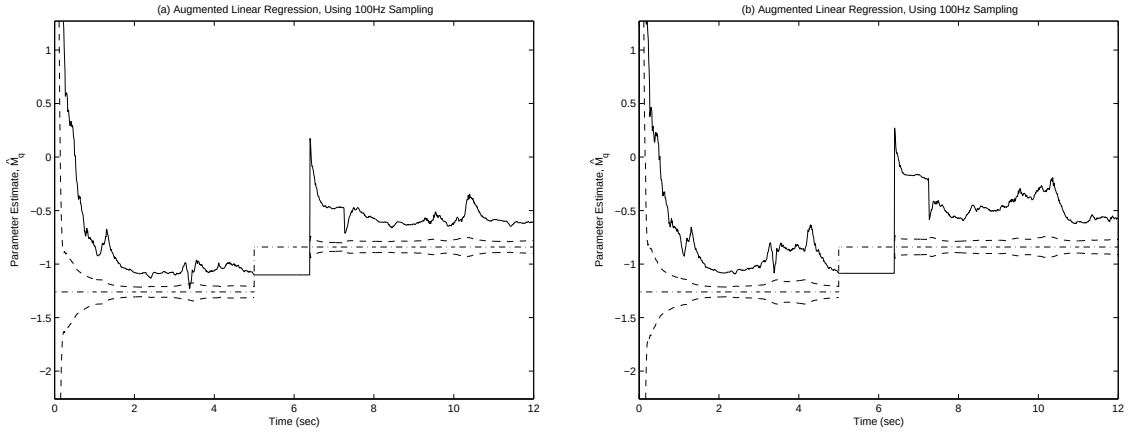


Figure 5.20: Estimate of stability derivative $\hat{M}_q$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s, but there is no trim change. (a) Uses standard linear regression, (b) Uses novel augmented linear regression algorithm. Solid lines–estimates, Dash-dot lines–true parameter value about which the $\pm 1\sigma$ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

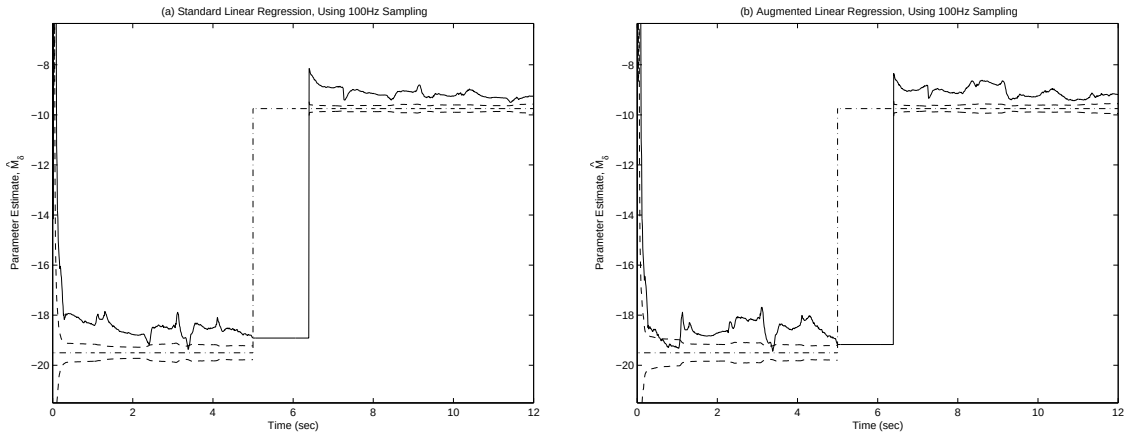


Figure 5.21: Estimate of control derivative $\hat{M}_\delta$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s, but there is no trim change. (a) Uses standard linear regression, (b) Uses novel augmented linear regression algorithm. Solid lines–estimates, Dash-dot lines–true parameter value about which the $\pm 1\sigma$ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

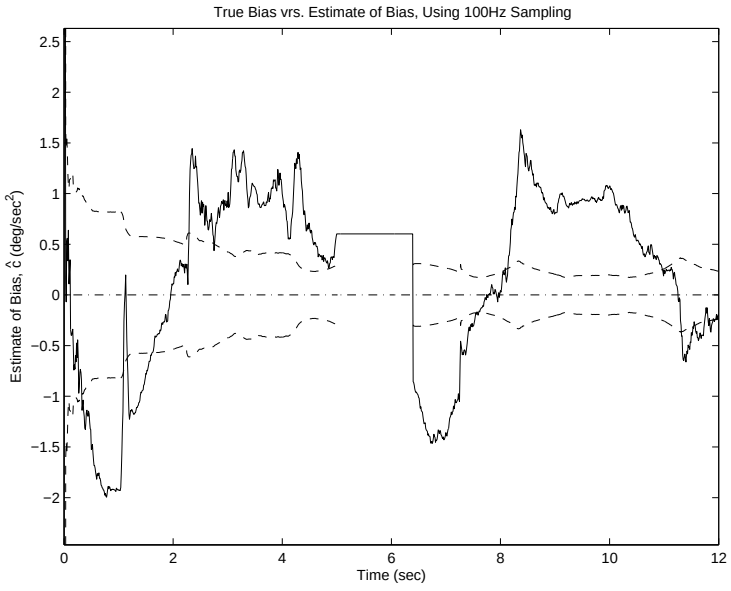


Figure 5.22: System identification of the intercept $\hat{c}$ $(= \hat{M}'_\alpha * \delta_d)$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s, but there is no trim change. Solid line–estimate, Dash-dot line–true parameter value about which the $\pm 1\sigma$ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

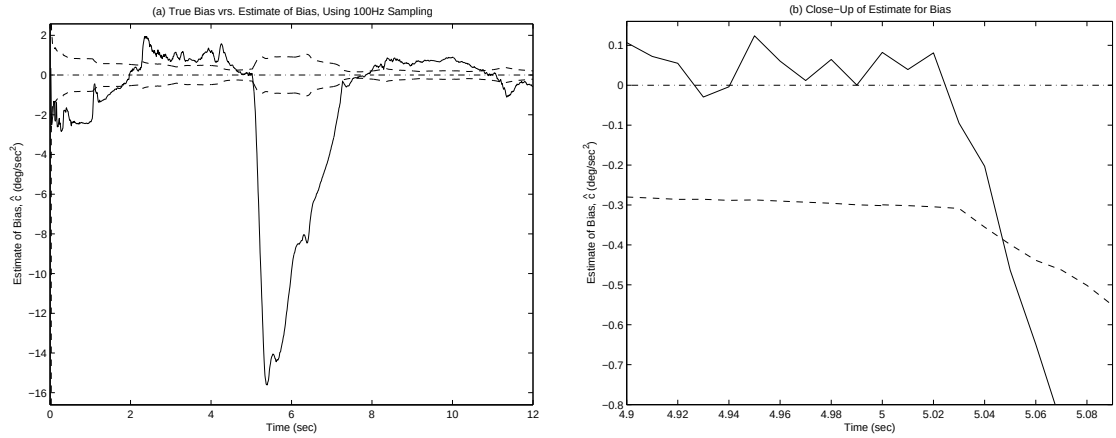


Figure 5.23: Estimate of the intercept $\hat{c}$ $(= \hat{M}'_\alpha * \delta_d)$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s, but there is no trim change. Sliding window is not reset after failure. Solid lines–estimates, Dash-dot lines–true parameter value about which the $\pm 1\sigma$ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

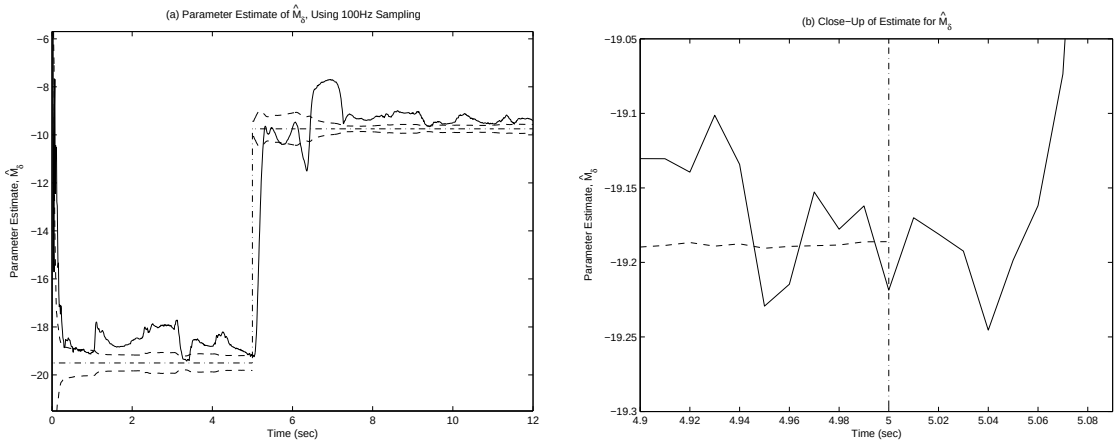


Figure 5.24: Estimate of control derivative parameter $\hat{M}_\delta$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s, but there is not trim change. Sliding window is not reset after failure. Solid lines–estimates, Dash-dot lines–true parameter value about which the predicted $\pm 1\sigma$ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

5.4.4 On-line Adjustment of the Proportional Gain K_P. Two methods of on-line adjustment for K_P are presented. As previously stated, the pitch damping gain K_α is set to the nonfailed Type-0 control gain after control surface failure, while the integral action control gain K_I is set to zero. This configuration is important, in that enables the flight control system to rely on the feed-forward control gain adjustments, rather than relying to integral action when the control surface failure occurs. The reconfigurable control design reflected in Method A relies on the easiest way to implement (5.11). The $\hat{M}_{\delta_{UF}}$ value is allowed to equal the frozen estimate $\hat{M}_\delta$ shown in Figure 5.15 between 5 and 6.4 seconds. One can recall from previous explanation that some time (i.e. 1.4 seconds) must elapse in order to produce reliable estimates using the expanding window batch process after the failure is detected; hence, the parameter freeze is essential for the feed-forward control algorithm. The $\hat{M}_{\delta_F}$ value is assigned to the instantaneous parameter estimate calculated via augmented regression, as developed in Section 5.4.2. Figure 5.25 shows the implemented strategy for the proportional gain. The proportional gain after failure K_{P_F} switches from the original

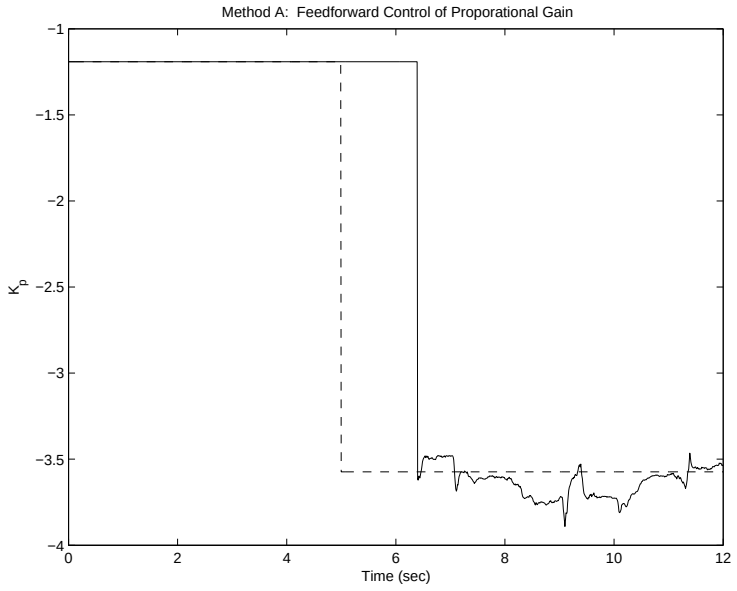


Figure 5.25: Proportional gain K_P during flight simulation experiment, as determined by Method A (solid line). The dashed line indicates the ideal gain switch.

gain K_P when the parameter estimate shown in Figure 5.15 unfreezes at 6.4 seconds.

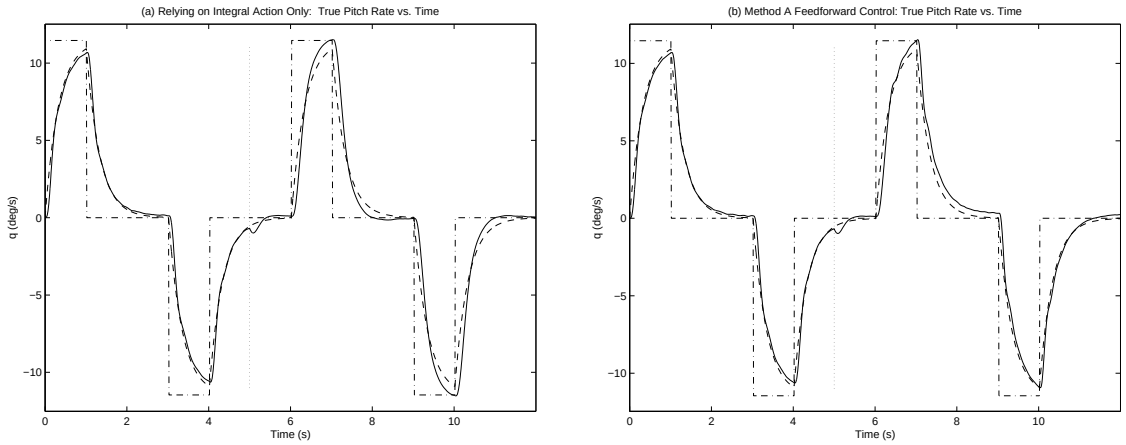


Figure 5.26: Pitch rate time history. Dash-dot lines (desired pitch rate command) Dashed lines (Prefiltered pitch rate command) Solid lines (Actual pitch rate) (a) Relies on conventional Type-1 inner loop controller for disturbance rejection (b) Method A reconfigurable flight control.

The damping gain after failure K_{α_F} is fixed at 0.2 after 6.4 seconds, as prescribed by the Type-0 controller gains summarized in Table 5.5. The reconfigurable control command tracking methodology clearly surpasses the results achieved by the conventional flight control system, which relies entirely on integral action for disturbance rejection, as shown after $t = 6.4$ seconds in Figure 26(a). The impact of using reconfigurable control is immediate. The reconfigurable flight control system encounters a transient at 6.4 seconds when the gain switch occurs, as shown by the pitch rate tracking (solid line) in Figure 26(b). When considering the pitch rate tracking between t= 5-9 seconds, the pilot's aggressive maneuvering prevents near instantaneous matching of the commanded pitch rate by the reconfigurable flight control system. Perhaps if the pilot was less aggressive in his choice of maneuvering during the failure, the pitch rate tracking might better match the pitch rate command. By approximately 9 seconds into the flight, the reconfigurable flight control system compensates very well for the system failure. Also, the overall system identification quality of the stability and control derivatives improves with reconfigurable control–see Figure 5.27 versus Figures 5.13 through 5.15.

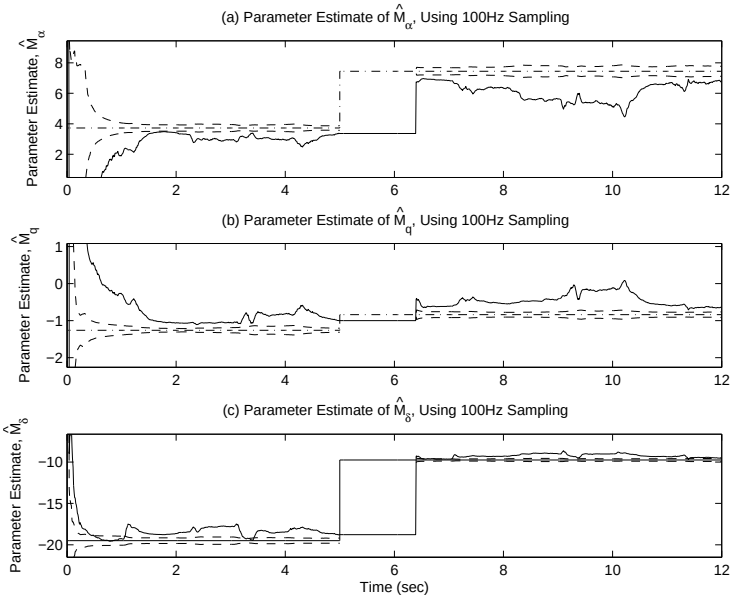


Figure 5.27: Estimate of stability derivative parameter estimates that accompany Method A reconfigurable control design. 50% of the horizontal stabilator surface is lost at $t = 5$ seconds. Solid lines–estimates, Dash-dot lines–true parameter value about which the predicted $\pm 1\sigma$ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

There are two primary drawbacks to using Method A. First, using the final $\hat{M}_\delta$ at failure as a single contributor to the new proportional gain shown in (5.11) introduces more uncertainty than is necessary. This results in an offset in K_P from the desired proportional gain that varies quite a bit from one Monte Carlo run to the next. Likewise, letting $\hat{M}_{\delta_F}$ equal the instantaneous value of $\hat{M}_\delta$ after failure introduces too much variation into K_P, as shown in Figure 5.25. However, this methodology was extremely easy to implement and vastly outperforms the conventional approach that relies solely on integral action.

Several improvements are incorporated into the reconfigurable control approach described as Method B. First, $\hat{M}_{\delta_{UF}}$ as shown in (5.11) now uses the last 100 control derivative estimates prior to the failure at $t = 5$ seconds. This mechanism reduces the amount of variation in $\hat{M}_{\delta_{UF}}$ between Monte Carlo simulations when compared to Method A. With successful trigger event recognition, this is easily recovered from the real-time system identification module.

Second, the reconfigurable control Method A relies solely on integral action between 5 and 6.4 seconds. Simulation experiments show that reconfigurable control is far superior to integral action via feedback, so one should implement the feedforward methodology as soon as a "decent" gain adjustment is possible. As reflected in Figure 5.28, this condition occurs 0.3 seconds after failure, as the linear regression has 30 measurements in an expanding batch window by this time.

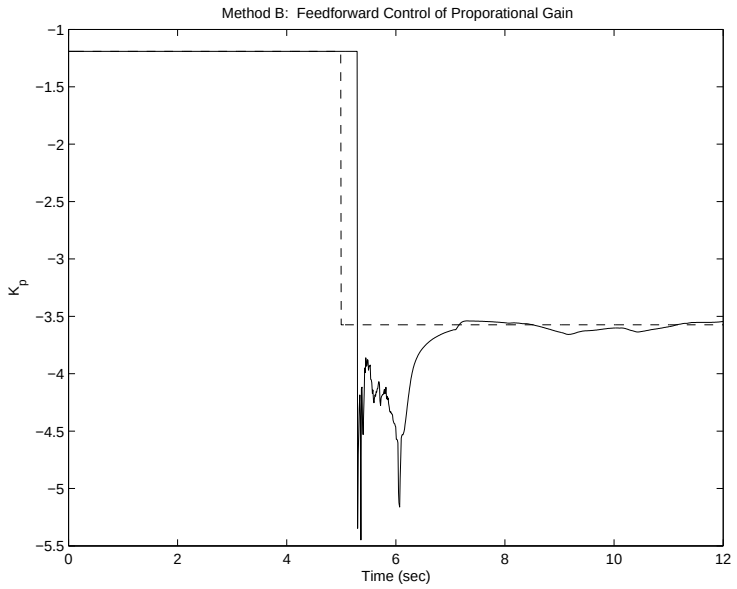


Figure 5.28: Proportional gain K_P during flight simulation experiment, as determined by Method B (solid line). The dashed line indicates the ideal gain switch.

Third, a "trailing average" technique is incorporated to smooth $\hat{M}_{\delta_F}$ used in (5.11) for the reconfigurable control Method B. In order to pursue the competing objectives of (a) revising K_{P_F} quickly and (b) revising K_{P_F} accurately, the following strategy to produce the realtime K_P shown in Figure 5.28 is adopted. From the analysis in Section 5.4.2, one realizes that $\hat{M}_\delta$ provides a good estimate of the control derivatives after the expanding system identification window size achieves 140 measurement. So, K_P is allowed to switch at $t = 5.3$ seconds. The $\hat{M}_{\delta_F}$ value, as given by (5.11), uses the instantaneous control derivative estimate between $t = 5.3$ and 6.4 seconds. Then, $\hat{M}_{\delta_F}$ reflects an average of an expanding window of control derivative estimates M_δ for the next 100 measurements. From $t = 7.4$ seconds onward to the

end of the simulation, $\hat{M}_{\delta_F}$ reflects an average of the trailing 100 control derivative estimates at each time increment. The damping gain after failure K_{α_F} remains fixed at 0.2 as previously described in this section.

Again, improved tracking is achieved using the reconfigurable control system described in Method B, as shown in Figure 5.29. There are improvements visible

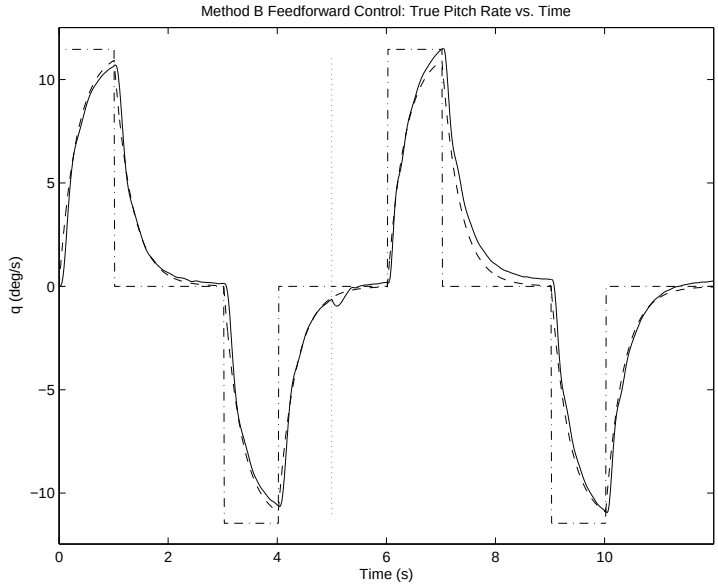


Figure 5.29: Pitch rate time history. Method B reconfigurable flight control used. Solid line (Actual pitch rate) Dash-dot line (desired pitch rate command) Dashed line (Prefiltered pitch rate command)

in pitch rate tracking, particularly between 5.3 and 6.4 seconds, as Method B relies on the "best available" on-line controller design methodology instead of solely relying on controller robustness provided by integral action, as shown in Figure 5.26. Also, Figure 5.28 shows a significant reduction in the reconfigurable proportional gain K_P variation when compared to Figure 5.25, and nearly achieves the "desired" K_P value. Figure 5.30 shows similarly improved overall system identification of the stability and control derivatives with reconfigurable control when compared to Figures 5.13 through 5.15. Finally, the reconfigurable control Method B shows that the fact that M_δ in Figure 5.15 is a biased estimate does not adversely impact the determination of K_P. The bias in M_δ is proportional to the control surface effectiveness.

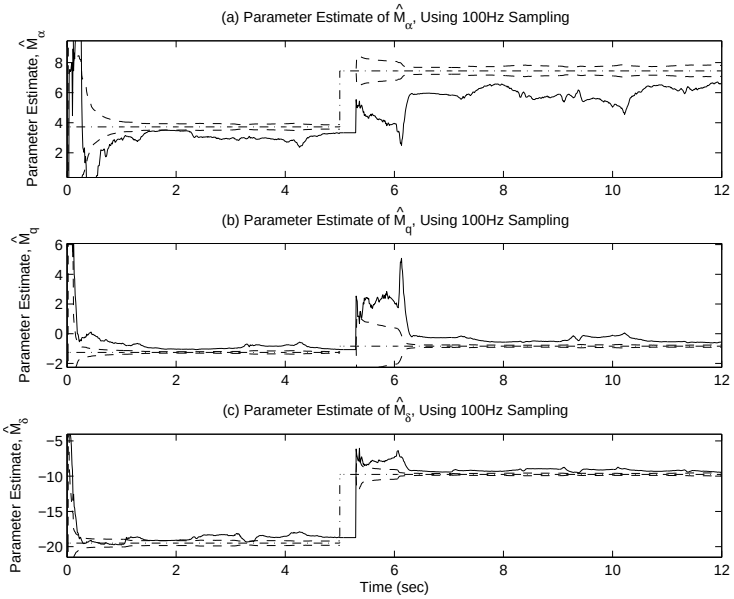


Figure 5.30: Estimate of stability derivative parameter estimates that accompany Method B reconfigurable control design. 50% of the horizontal stabilator surface is lost at $t = 5$ seconds. Solid lines–estimates, Dash-dot lines–true parameter value about which the predicted $\pm 1\sigma$ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

5.4.5 Expanded On-line Controller Design. One can now consider results from simulation experiments when both adaptive control gains K_{P_F} and K_{α_F} are adjusted on-line and thus allowed to vary according to (5.11) and (5.12). Both K_P and K_α are adjusted on-line similar to Method B described in Section 5.4.5. The results of this approach show a neglible impact on K_P (Figure 5.31) while K_α approaches the ideal gain value (Figure 5.32) after failure. Incorporating on-line adjustment methodology for both K_P and K_α produces definite improvements in tracking performance over the previous methodology presented in Section 5.4.4 and the flight control system that relies solely on integral action to reject the control surface failure-induced "disturbance." The resulting improvements to the pitch rate tracking (Figure 5.33) are evident between 5.3 to 7 seconds, when compared to results displayed in Figure 5.29. The increased damping gain resulting from the adjustment of K_α prevents the pitch rate from overshooting the prefiltered pitch rate command. In addition, the overall system identification of the stability and control derivatives (Figure 5.34)

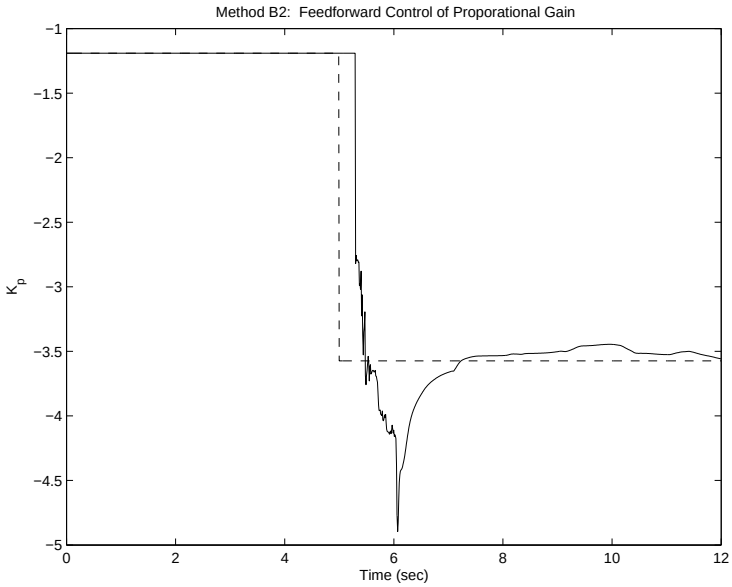


Figure 5.31: Proportional gain K_P during flight simulation experiment, as determined by reconfigurable flight control that updates both K_P and K_α (solid line). The dashed line indicates the ideal gain switch.

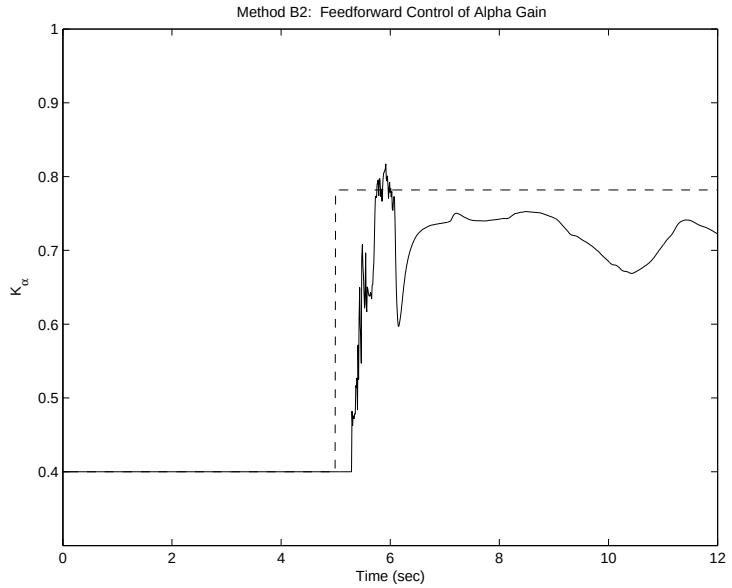


Figure 5.32: Damping gain K_α during flight simulation experiment, as determined by reconfigurable flight control that updates both K_P and K_α (solid line). The dashed line indicates the ideal gain switch.

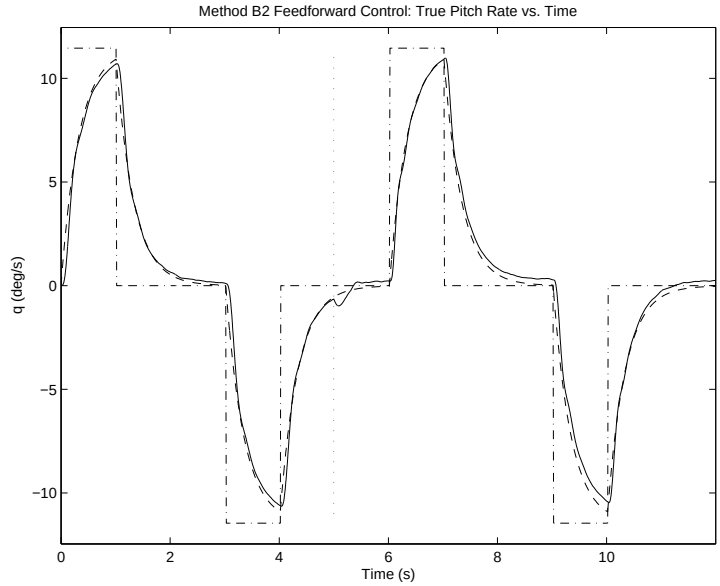


Figure 5.33: Pitch rate time history. Reconfigurable flight control that updates both K_P and K_α is used. Solid line (Actual pitch rate) Dash-dot line (desired pitch rate command) Dashed line (Prefiltered pitch rate command)

with this reconfigurable control law are again improved over results presented in Figure 5.30, particularly between 5.3 and 6.4 seconds. This improved identification, in turn, further helps the on-line controller adjustment process and pitch rate tracking.

In order to differentiate further between pitch rate tracking effectiveness between methodologies, one should compare results from 10 Monte Carlo simulation sets. First, Figure 5.35 shows that the proportional gain K_P is equivalent when comparing the on-line controller design methodologies developed in Sections 5.4.4 and 5.4.5. Second, reconfigurable control can effectively raise the value of the damping gain K_α when appropriate, as shown in Figure 5.36. The stability derivative M_α is much more difficult to estimate correctly than the control derivative M_δ. Part of this estimation sensitivity results from the comparative magnitudes between $\hat{M}_\alpha$ and $\hat{M}_\delta$, where $\hat{M}_\delta > \hat{M}_\alpha$. A more influential contributor to the estimation sensitivity of $\hat{M}_\alpha$ results from the implementation decision that limits the sliding window size to 240 measurements. Figure 5.34(a) shows the choppy nature of $\hat{M}_\alpha$ with this sliding window size arrangement versus the smooth estimates obtains in (c) for $\hat{M}_\delta$. One can recall

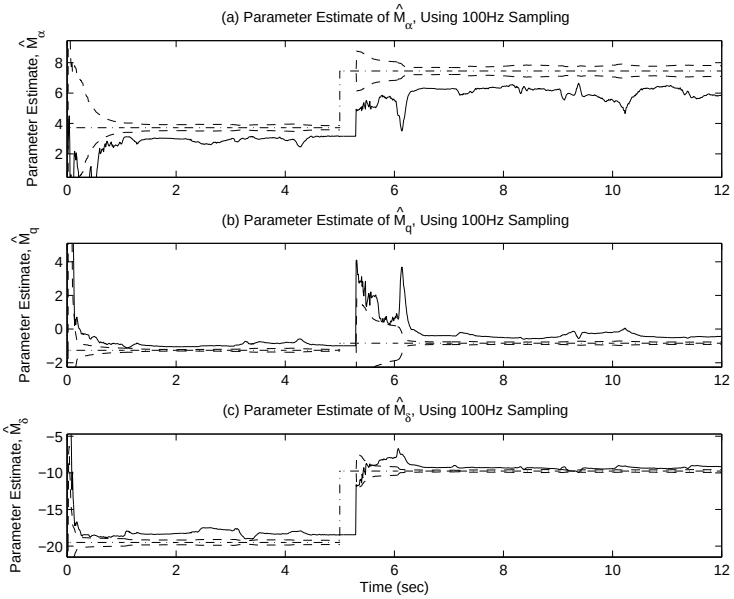


Figure 5.34: Estimate of stability derivative parameter estimates that accompany the reconfigurable flight control that updates both K_P and K_α. 50% of the horizontal stabilator surface is lost at $t = 5$ seconds. Solid lines–estimates, Dash-dot lines–true parameter value about which the $\pm 1\sigma$ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

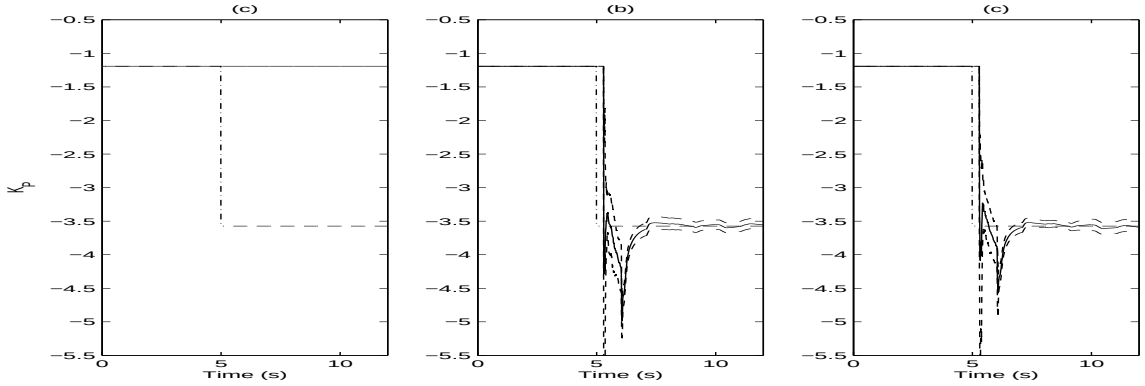


Figure 5.35: Proportional Gain K_P that is determined experimentally from 10 Monte Carlo simulation runs. Comparison of a control system that relies solely on feedback integral action (a) to feed-forward control that adjusts only the K_P gain (Method B) (b) to feed-forward control that adjusts both K_P and the damping gain K_α (c). The experimental mean is represented by the solid lines, while the dashed lines represent $\pm$ one standard deviation from that experimentally determined mean. Dash-dot line indicates the ideal gain K_P.

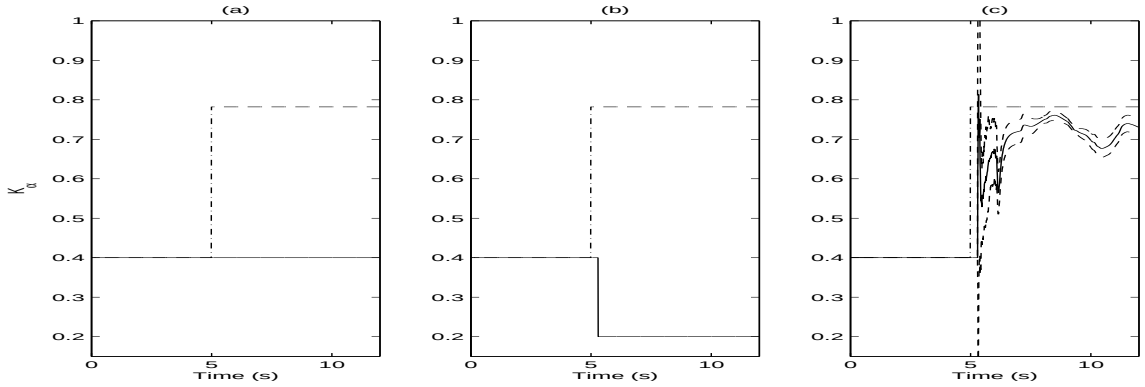


Figure 5.36: Damping Gain K_α that is determined experimentally from 10 Monte Carlo simulation runs. Comparison of a control system that relies solely on feedback integral action (a) to feed-forward control that adjusts only the K_P gain (Method B) (b) to feed-forward control that adjusts both K_P and the damping gain K_α (c). The experimental mean is represented by the solid lines, while the dashed lines represent $\pm$ one standard deviation from that experimentally determined mean. Dash-dot line indicates the ideal gain K_α.

that smooth estimates for $\hat{M}_\alpha$ are achievable with a large window size ($N > 400$), as shown in Figure 5.10 for the unfailed case. However, the feed-forward control design must prioritize the proportional gain K_P first. Hence, K_α, which relies on both M_α and M_δ knowledge, is much less accurate than the proportional gain K_P. On-line controllers that adjust both K_P and K_α clearly minimizes the tracking error (Figure 5.37). The time interval between 6 and 7 seconds is now considered. When the first pitch rate doublet occurs after the onset of control surface failure, the on-line controller design module that adjusts both K_P and K_α enables nice command tracking (Figure 5.37(c)). The adaptive control method clearly removes much of the oscillatory build-up and overshoot in pitch. Clearly, the conventional flight control system that relies solely on integral action (Figure 5.37(a)) overshoots the desired pitch rate command significantly.

Next, one should consider the time interval between 7 and 9 seconds, when the pilot stick is returned to the neutral position. The on-line controller design module adjusts both K_α and K_P and dampens the pitch rate nicely, allowing it to follow the pitch rate command q_c (Figure 5.37(c)). An interesting event occurs from 9 to 10 sec-

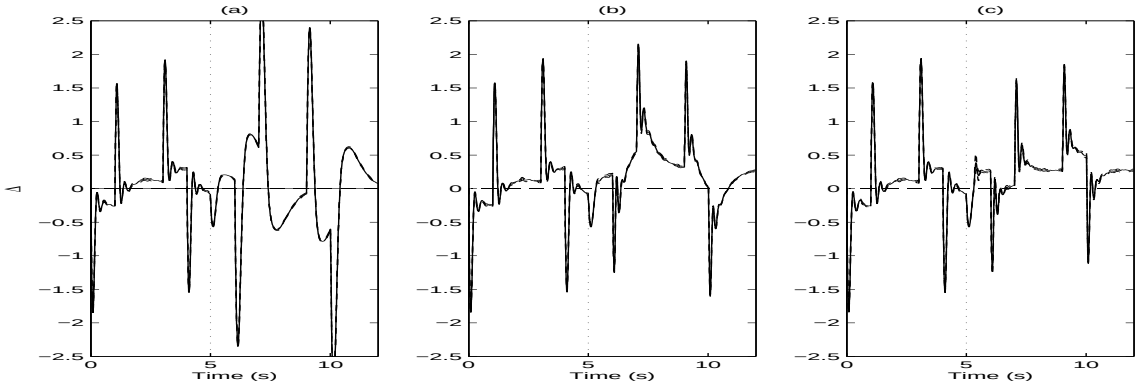


Figure 5.37: Difference between prefiltered tracking command and actual pitch rate. 10 Monte Carlo simulation runs. Comparison of a control system that relies solely on feedback integral action (a) to feed-forward control that adjusts only the K_P gain (Method B) (b) to feed-forward control that adjusts both K_P and the damping gain K_α (c). The experimental mean is represented by the solid lines, while the dashed lines represent $\pm$ one standard deviation from that experimentally determined mean. Dash-dot line indicates the ideal difference.

onds, when the second pitch rate doublet after control surface failure is input. The on-line controller design module that only adjusts K_P (Figure 5.37(b)) yields better command tracking than other options. One can explain this as resulting from the fact that K_α is less than in the other methods presented (e.g. $K_\alpha = 0.2$) Hence, the damping gain K_α provides the least resistance to enable the flight control system to track q_c at the pitch rate doublet's onset (See Figure 5.36 from 9 to 10 seconds.) However, when the doublet is released at 10 seconds, this on-line controller design module (Figure 5.37(b)) yields more oscillatory behavior when compared to the on-line controller design module which adjusts both K_P and K_α (Figure 5.37(c)).

One final important determination pertains to the method of calculating K_α. Three on-line controller design methods are compared that adjust both K_P and the damping gain K_α. All approaches use the last 100 blended derivative estimates to determine $M_{\delta_{UF}}$, M_{δ_F}, and $M_{\alpha_{UF}}$, as shown in (5.12). The number of inputs provided to determine M_{α_F} (used to create K_α in Figure 5.39) are 100 derivative estimates (c), 5 derivative estimates (b), or 1 derivative estimate (a). The experimental mean is

represented by the solid lines, while the dash lines represent ± one standard deviation from that experimentally determined mean.

First, Figure 5.38 shows that the proportional gain K_P is negligibly impacted regardless of the smoothing introduced to M_{α_F}. The damping gain K_α varies signif-

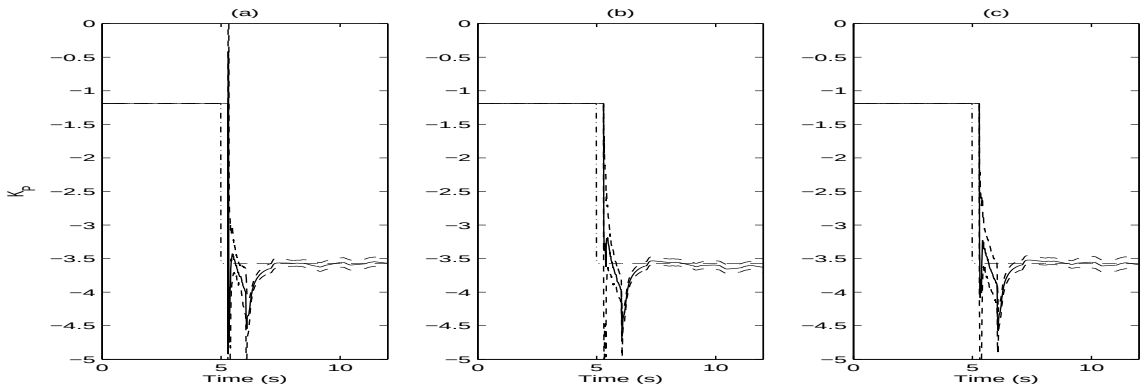


Figure 5.38: Proportional Gain K_P that is determined experimentally from 10 Monte Carlo simulation runs. 10 Monte Carlo simulation runs. Comparison of three feed-forward control methods that adjust both K_P and the damping gain K_α. All approaches use the last 100 blended derivative estimates for $M_{\delta_{UF}}$, M_{δ_F}, and $M_{\alpha_{UF}}$. The number of inputs provided to determine M_{α_F} are 100 derivative estimates (c), 5 derivative estimates (b), or 1 derivative estimate (a). The experimental mean is represented by the solid lines, while the dashed lines represent ± one standard deviation from that experimentally determined mean. The dash-dot lines indicate the ideal gain K_P.

icantly, depending on the smoothing approach used for M_{α_F} (Figure 5.39). If one does not blend the K_α contributions sufficiently, significant variations in the pitch damping gain result near "transition points" of the pitch rate command. Ironically, K_α wants to "increase damping" (i.e., K_α rises) at the onset of a pitch rate command when using on-line adaptive control. This is demonstrated at $t = 9$ seconds in Figure 5.39. On the contrary, the on-line adaptive controller tends to decrease damping (i.e., K_α falls) when the pitch rate command returns to a neutral position, such as at $t = 10$ seconds in Figure 5.39. This is the exact opposite of what one wants for best tracking. Hence, as shown in Figure 5.40, one should blend an appropriate number of contributions (specifically M_{α_F}) to determine K_α in (5.12). Figure 5.40(c) shows the best pitch rate tracking.

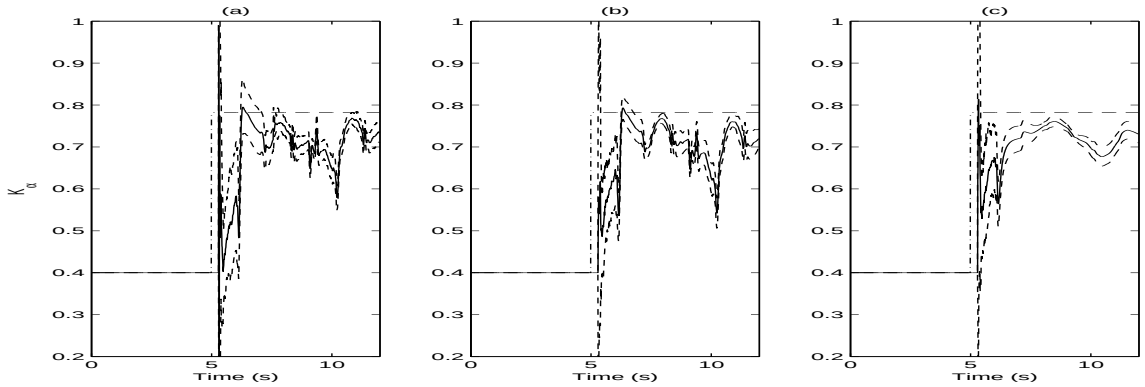


Figure 5.39: Damping Gain K_α that is determined experimentally from 10 Monte Carlo simulation runs. 10 Monte Carlo simulation runs. Comparison of three feed-forward control methods that adjust both K_P and the damping gain K_α. All approaches use the last 100 blended derivative estimates for $M_{\delta_{UF}}$, M_{δ_F}, and $M_{\alpha_{UF}}$. The number of inputs provided to determine M_{α_F} are 100 derivative estimates (c), 5 derivative estimates (b), or 1 derivative estimate (a). The experimental mean is represented by the solid lines, while the dash lines represent $\pm$ one standard deviation from that experimentally determined mean. The dash-dot lines indicate the ideal gain K_α.

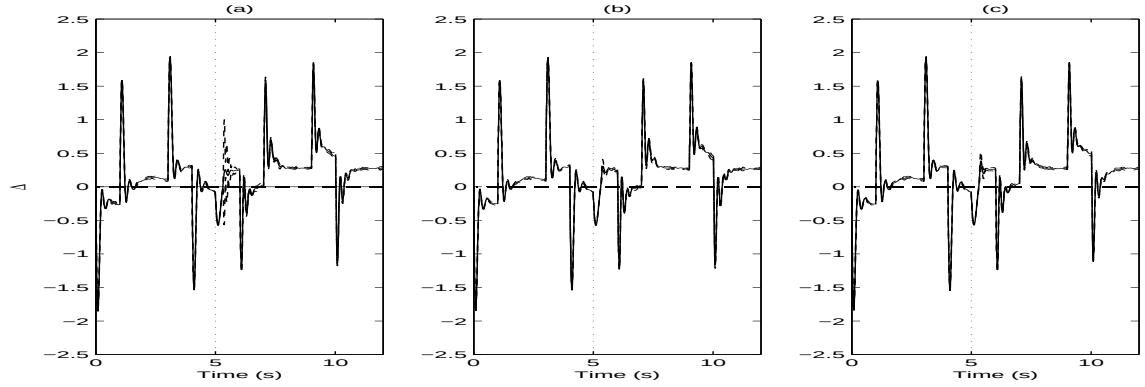


Figure 5.40: Difference between prefiltered tracking command and actual pitch rate. 10 Monte Carlo simulation runs. Comparison of three feed-forward control methods that adjust both K_P and the damping gain K_α. All approaches use the last 100 blended derivative estimates for $M_{\delta_{UF}}$, M_{δ_F}, and $M_{\alpha_{UF}}$. The number of inputs provided to determine M_{α_F} are 100 derivative estimates (c), 5 derivative estimates (b), or 1 derivative estimates (a). The experimental mean is represented by the solid lines, while the dash lines represent $\pm$ one standard deviation from that experimentally determined mean. The dash-dot lines indicate the ideal difference.

5.4.6 Accounting for Control Surface Rate Limits. When employing adaptive and reconfigurable control, one may need to account for actuator rate limiting. The F-16 class aircraft horizontal stabilator rate limit is $|\dot{\delta}_{\max}| = 60$ deg/s. At this point, one should consider Figures 5.41 through 5.44. The methodology that relies solely on integral action for robustness to reject the failure induced disturbance is clearly not impacted by actuator rate limits in this case. The three reconfigurable control methodologies proposed in this paper infringe on the $\dot{\delta}$ limit by $\sim$10% and $\sim$25% respectively.

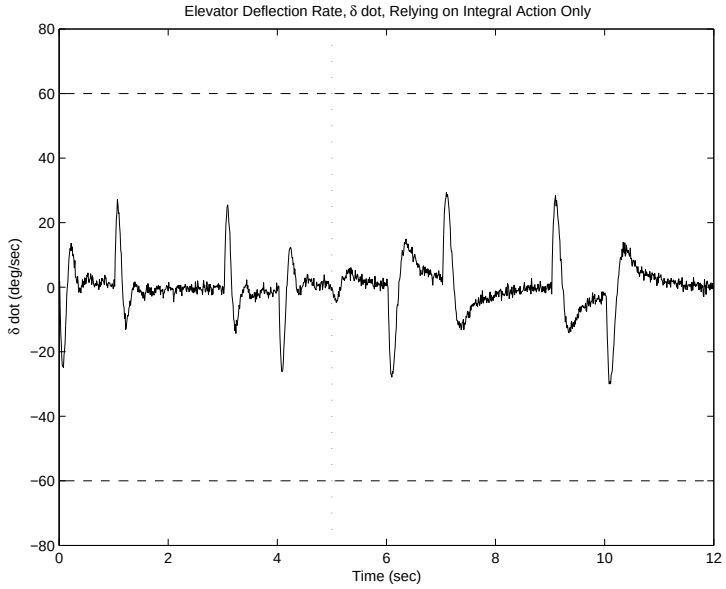


Figure 5.41: Actuator deflection rate, in which only integral control action is used.

First, one should incorporate an actuator hardware upgrade to allow reconfigurable flight control. Hardware already exists with significantly higher actuator rate limits, e.g., the VISTA F-16 aircraft's actuators are rated for $120\frac{deg}{s}$. Second, if hardware upgrades are not desirable, one must limit the commanded pitch rate that reflects the actuator's capability in order to ensure aircraft stability. In other words, one limits the pilot's ability to place commands on q_c, (or r), by further reducing the limits on the already existing command limiter upstream of the prefilter. One is

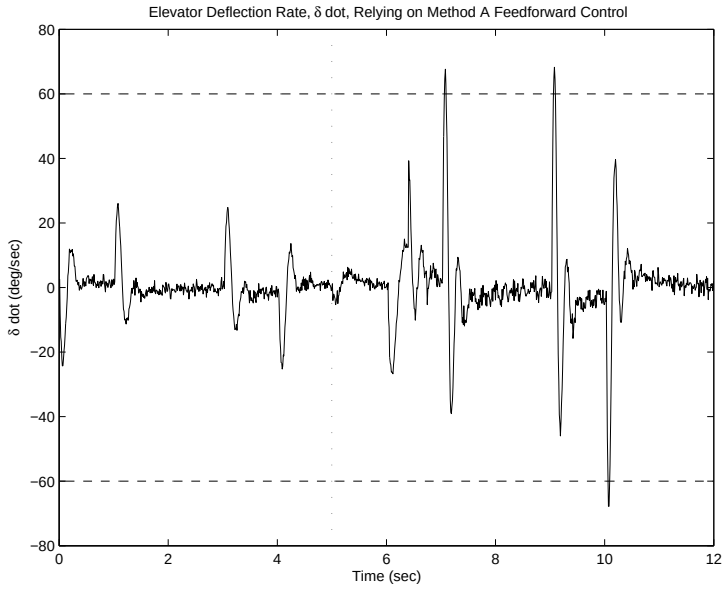


Figure 5.42: Actuator deflection rate, in which reconfigurable control Method A from Section 5.4.4, impacting only the proportional gain K_P, is used.

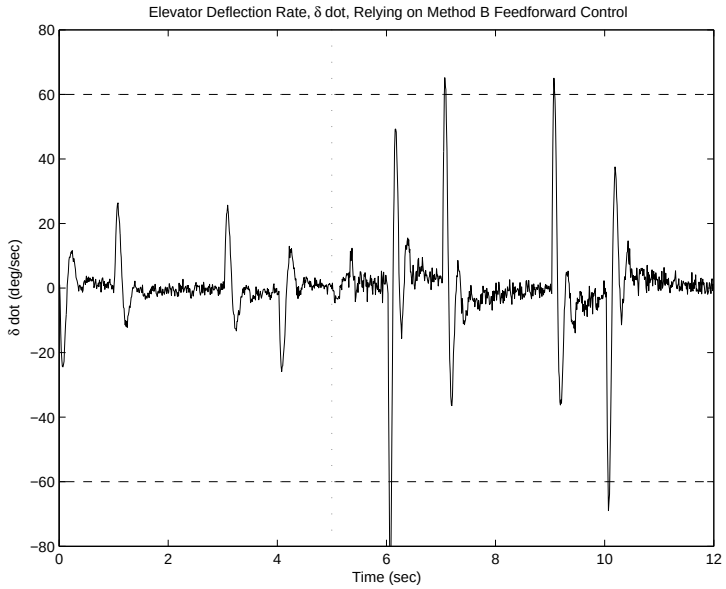


Figure 5.43: Actuator deflection rate, in which reconfigurable control Method B from Section 5.4.4, impacting only the proportional gain K_P, is used.

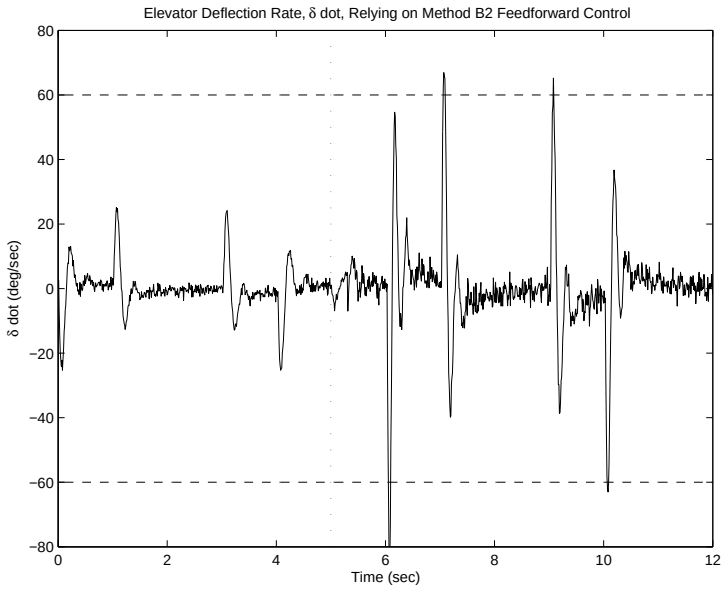


Figure 5.44: Actuator deflection rate, in which reconfigurable control impacting both K_P and K_α, as in Section 5.4.5, is used.

then able to track the reduced pitch rate command, reduced from that shown by the dashed line in Figure 5.4(c).

5.5 Conclusion for Reconfigurable Flight Control

In this chapter, a workable indirect adaptive and reconfigurable control paradigm is developed. Specifically, the focus is on the challenging open-loop unstable pitch plane dynamics of an F-16 class fighter aircraft, considering the possibility of a 50% control surface loss. The mitigation of the deleterious effects caused by a control surface area loss due to failure or battle damage is addressed. The aircraft stability and control derivatives are estimated on-line static system identification. A moving window/batch estimation process is used and the subsequent linear regression is augmented with an intercept. The intercept parameter is included to address the effect of trim change associated with control surface failure, in order to monitor for a failure and to trigger a change that effectively controls the system identification batch size via an expanding and sliding window. Parameter estimate information on the failed plant is then used to revise the inner loop control gains on-line. Moreover, the inter-

cept estimate is used to calculate a control signal. One can feed-forward this signal to redress the control surface loss-induced change in trim and to reduce the need for integral action in the inner loop controller. Simulation results validate the augmented linear regression's estimation accuracy. Good parameter estimation performance is achieved. Since the parameter estimation module is the heart of the adaptive controller, superior reconfigurable control action is demonstrated in simulations. Hence, the reliance on integral action for retrimming is reduced. A workable adaptive and reconfigurable flight control system is achievable.

VI. Bearing-Only Measurements

This chapter is concerned with obtaining an emitter's position estimate from bearings-only measurements using triangulation. Standard approaches assume azimuth angle measurements and use an Iterative Least Squares algorithm to calculate the RF emitter's position. This chapter exploits the insight that, in practice, the bearing measurements phasor $(\cos\phi,\ \sin\phi)$ is available, which more accurately represents the measurement hardware. This allows the use of an efficient linear regression algorithm and avoids equation error due to linearization. As a result, the modified estimation algorithm developed herein to address the problem of bearings-only measurements for emitter geo-location produces superior parameter estimates, that is, improved emitter geo-location, compared to the standard formulation and ILS algorithm.

In Section 6.1, the RF emitter geo-location problem is motivated. The linear homogeneous equation formulation from the projective geometry is introduced in Section 6.2. This equation formulation provides the basis for both the modified estimation approach and the conventional ILS approach. In Section 6.3, the modified triangulation algorithm which considers the impact of available AOA phasor measurements is developed. Since the equation error covariance R is parameter-dependent (i.e., components of the parameter, x and y, are embedded within R), Section 6.4 expands on the Iterative Generalized Least Squares (IGLS) algorithm, previously developed in [29], and is required to produce the position estimates. The explicit formulae to determine the geo-location of the emitter are developed in Section 6.5. For the sake of completeness and comparison purposes, Section 6.6 includes a discussion of the conventional triangulation paradigm. Simulation results and a comparison of the modified triangulation and the standard triangulation paradigms, are discussed in Section 6.7. Section 6.8 provides concluding remarks.

6.1 Emitter Geo-location

The algorithm for cooperative RF emitter geo-location using bearings-only/AOA measurements is developed. The scenario is shown in Figure 6.1. When the emitter

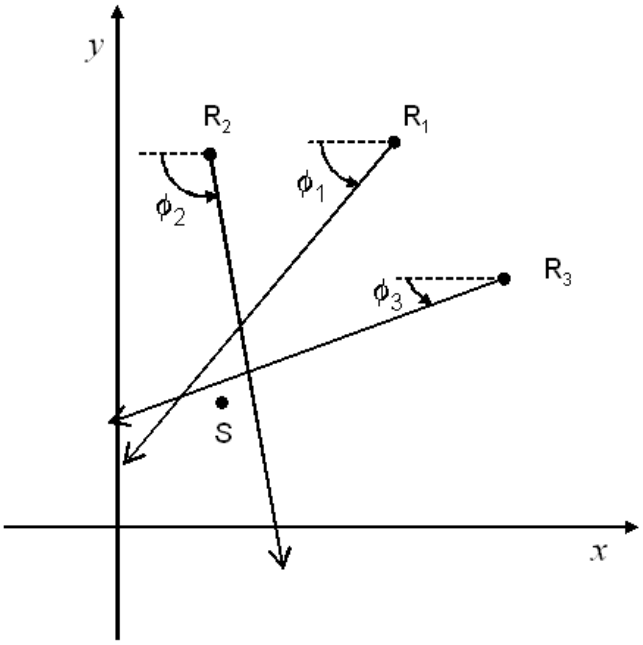


Figure 6.1: The emitter's position is S. The bearings-only measurements are ϕ_i, in which $i = 1, \ldots, N$. The mobile receivers' positions R_i, in which $i = 1, \ldots, N$, are recorded at the time of the ping from the emitter.

at an unknown location $S = (x, y)$, pings a receiver, the AOA ϕ_i is recorded. The uncertainty associated with the AOA measurement is typically high. However, over time, these measurements of opportunity are accumulated and enable the emitter's triangulation as shown in Figure 6.1; hence, the cooperative nature of emitter geo-location. In the planar scenario, two parameters, x and y, the emitter's coordinates, are of interest. Hence, in theory, position estimates obtained with as few as $N = 2$ measurements from an emitter are possible. If an emitter provides more than one measurement opportunity to a mobile receiver at different times during the same sortie, these measurements are treated as separate measurements. It is important to emphasize that in emitter geo-location applications, the measurement error standard deviation σ_ϕ is high. Therefore, many measurements are needed to wash-out the

measurement noise. Furthermore, the measurement arrangement's geometry (GDOP) strongly impacts the achievable emitter geo-location accuracy.

6.2 Geometry

The linear homogeneous equation formulation from projective geometry is applied to obtaining position estimates using bearing-only measurements. With reference to Figure 2.2, a linear equation in homogeneous form is considered

$$x \sin\phi + y \cos\phi + c = 0 \quad . \tag{6.1}$$

Applying (6.1) to the emitter's position and to the receiver's position, the following two relations for each of the N bearings measurements are obtained

$$x \sin\phi_i + y \cos\phi_i + c_i = 0$$
$$x_i \sin\phi_i + y_i \cos\phi_i + c_i = 0$$

in which $i = 1, \ldots, N$. By subtracting the two equations, the ith measurement equation is obtained

$$x \sin\phi_i + y \cos\phi_i = x_i \sin\phi_i + y_i \cos\phi_i \qquad , \qquad i = 1, \ldots, N \quad . \tag{6.2}$$

This relation and the observation that phasor measurements of the AOAs are available is the basis for the modified linear regression-based triangulation method.

6.3 Stochastic Modelling

One can exploit the access to the measurements, $(sin\phi_i)_m$ and $(cos\phi_i)_m$, $i = 1, \ldots, N$. The measurements reflect the true phasor plus a zero-mean, Gaussian-

distributed, measurement noise. Hence

$$\begin{aligned}(\sin\phi_i)_m &= \sin\phi_i + v_i \quad , \quad v_i \in \mathcal{N}(0,\sigma^2) \\ (\cos\phi_i)_m &= \cos\phi_i + w_i \quad , \quad w_i \in \mathcal{N}(0,\sigma^2) \quad .\end{aligned} \tag{6.3}$$

in which v_i and w_i are assumed independent. One can rearrange (6.3) as

$$\begin{aligned}\sin\phi_i &= (\sin\phi_i)_m - v_i \\ \cos\phi_i &= (\cos\phi_i)_m - w_i\end{aligned}$$

and insert this into (6.2):

$$[(\sin\phi_i)_m - v_i]\cdot x - [(\cos\phi_i)_m - w_i]\cdot y = x_i\cdot[(\sin\phi_i)_m - v_i] - y_i\cdot[(\cos\phi_i)_m - w_i]$$

in which $i = 1,\ldots,N$. The equation is rearranged this into the form of a linear regression in x and y,

$$(\sin\phi_i)_m x - (\cos\phi_i)_m y = x_i(\sin\phi_i)_m - y_i(\cos\phi_i)_m + (x - x_i)v_i - (y - y_i)w_i$$

in which $i = 1,\ldots,N$. One should note that in this modified approach, linearization is not required.

By accumulating a data record over time N, one can form the linear regression

$$Z = H\cdot\theta + V \tag{6.4}$$

in which $Z, V \in \mathbb{R}^N$, and the parameter vector $\theta \in \mathbb{R}^2$

$$\theta \triangleq \begin{pmatrix} x \\ y \end{pmatrix} \quad .$$

The measurement vector Z is

$$Z = \begin{pmatrix} x_1(\sin\phi_1)_m - y_1(\cos\phi_1)_m \\ \vdots \\ x_N(\sin\phi_N)_m - y_N(\cos\phi_N)_m \end{pmatrix}_{N\times 1},$$

the regressor H is an $N \times 2$ matrix

$$H = \begin{bmatrix} (\sin\phi_1)_m & -(\cos\phi_1)_m \\ \vdots & \vdots \\ (\sin\phi_N)_m & -(\cos\phi_N)_m \end{bmatrix}_{N\times 2},$$

and the equation error vector is:

$$V = \begin{pmatrix} (y-y_1)w_1 - (x-x_1)v_1 \\ \vdots \\ (y-y_N)w_N - (x-x_N)v_N \end{pmatrix}_{N\times 1}.$$

While the regression (6.4) has the appearance of a standard linear regression, one should notice that the unknown parameter θ appears in the equation error V.

The equation error covariance

$$\begin{aligned} R &= \mathbf{E}[VV^T] \\ &= \begin{pmatrix} (y-y_1)w_1 - (x-x_1)v_1 \\ \vdots \\ (y-y_N)w_N - (x-x_N)v_N \end{pmatrix} \cdot \begin{pmatrix} (y-y_1)w_1 - (x-x_1)v_1 & \dots & (y-y_N)w_N - (x-x_N)v_N \end{pmatrix} \end{aligned}$$

Since v_i and w_i are independent and if $E[v_iv_i] = \sigma$ and $E[w_iw_i] = \sigma$, then

$$R = \begin{bmatrix} (x-x_1)^2\sigma^2 + (y-y_1)^2\sigma^2 & & 0 \\ & \ddots & \\ 0 & & (x-x_N)^2\sigma^2 + (y-y_N)^2\sigma^2 \end{bmatrix} \quad .$$

which reduces to

$$R = \sigma^2 \begin{bmatrix} (x-x_1)^2 + (y-y_1)^2 & & 0 \\ & \ddots & \\ 0 & & (x-x_N)^2 + (y-y_N)^2 \end{bmatrix} \tag{6.5}$$

One should note that

$$R = \sigma^2 \tilde{R}$$

in which

$$\tilde{R} = \tilde{R}(\theta) \quad .$$

The equation error covariance is parameter-dependent, hence, generating the need for an iterative generalized least squares algorithm. This parameter dependence is prevalent, even in the special case in which the observation points (x_i, y_i) are arranged on the circumference of a circle with radius ρ, centered on the target. Then, (6.5) reduces to

$$R = \sigma^2 \rho^2 I_N \quad .$$

One should note that the nature of the parameter estimation problem will not significantly change if the phasor measurement noise is correlated, namely $E[v_i w_i] \neq 0$.

6.4 Iterative Generalized Least Squares Algorithm

The linear regression is considered

$$Z = H\theta + V \quad , \quad V \in \mathcal{N}(0, R(\theta))$$

in which the equation error covariance R is parameter dependent. The classical minimum-variance solution of a linear regression, in which the equation error covariance R is not parameter dependent and is known, [36] is

$$\hat{\theta} = (H^T R^{-1} H)^{-1} H^T R^{-1} Z$$

and the parameter estimation error covariance is

$$P = (H^T R^{-1} H)^{-1} \quad .$$

This parameter dependence motivates one to use the Iterative Generalized Least Squares (IGLS) algorithm [29]

$$\hat{\theta}^{(k+1)} = (H^T R^{-1}(\hat{\theta}^{(k)}) H)^{-1} H^T R^{-1}(\hat{\theta}^{(k)}) Z \quad , \quad k = 0, 1, \ldots \; . \tag{6.6}$$

The initial parameter guess is obtained from the solution of the LS problem, namely

$$\hat{\theta}^{(0)} = (H^T H)^{-1} H^T Z \quad . \tag{6.7}$$

One obtains (6.7) by setting R to a scaled unity matrix, as is the case if the observation points are arranged on the circumference of a circle centered at the target.

The parameter estimate is $\hat{\theta}^{(\infty)}$. Having obtained the parameter estimate, one then calculates the predicted covariance of the parameter estimation error

$$P = (H^T R^{-1}(\hat{\theta}^{(\infty)}) H)^{-1} \quad . \tag{6.8}$$

In this case, (6.6) takes the form

$$\hat{\theta}^{(k+1)} = (H^T \tilde{R}^{-1}(\hat{\theta}^{(k)})H)^{-1} H^T \tilde{R}^{-1}(\hat{\theta}^{(k)})Z \quad , \quad k = 0, 1, \ldots$$

and (6.8) takes the form

$$P = \sigma^2 (H^T \tilde{R}^{-1}(\hat{\theta}^{(\infty)})H)^{-1} \quad .$$

One should note that the inversion of the $N \times N$ $\tilde{R}$ matrix is easy because $\tilde{R}$ is diagonal.

6.5 Explicit Formulae for AOA–Based Geo-location of Emitters

The following notation is introduced

$$r_i^2 \triangleq (x - x_i)^2 + (y - y_i)^2 \quad .$$

Using the notation r_i, the equation error covariance matrix becomes

$$\tilde{R} = \begin{bmatrix} r_1^2 & & 0 \\ & \ddots & \\ 0 & & r_N^2 \end{bmatrix} \quad .$$

One can calculate

$$H^T \tilde{R}^{-1} = \begin{bmatrix} \frac{1}{r_1^2}(\sin\theta_1)_m & \cdots & \frac{1}{r_N^2}(\sin\theta_N)_m \\ \frac{1}{r_1^2}(\cos\theta_1)_m & \cdots & \frac{1}{r_N^2}(\cos\theta_N)_m \end{bmatrix}$$

and then

$$H^T \tilde{R}^{-1} H = \begin{bmatrix} \sum_{i=1}^{N} \frac{1}{r_i^2}(\sin\theta_i)_m^2 & -\sum_{i=1}^{N} \frac{1}{r_i^2}(\sin\theta_i)_m(\cos\theta_i)_m \\ -\sum_{i=1}^{N} \frac{1}{r_i^2}(\sin\theta_i)_m(\cos\theta_i)_m & \sum_{i=1}^{N} \frac{1}{r_i^2}(\cos\theta_i)_m^2 \end{bmatrix} .$$

Inverting this 2×2 matrix, the equation error covariance matrix is obtained

$$\left(H^T \tilde{R}^{-1} H\right)^{-1} = \frac{1}{\triangle} \begin{bmatrix} \sum_{i=1}^{N} \frac{1}{r_i^2} (\cos\theta_i)_m^2 & \sum_{i=1}^{N} \frac{1}{r_i^2} (\sin\theta_i)_m (\cos\theta_i)_m \\ \sum_{i=1}^{N} \frac{1}{r_i^2} (\sin\theta_i)_m (\cos\theta_i)_m & \sum_{i=1}^{N} \frac{1}{r_i^2} (\sin\theta_i)_m^2 \end{bmatrix} \tag{6.9}$$

in which

$$\triangle \triangleq \left(\sum_{i=1}^{N} \frac{1}{r_i^2} (\sin\theta_i)_m^2\right) \left(\sum_{i=1}^{N} \frac{1}{r_i^2} (\cos\theta_i)_m^2\right) - \left(\sum_{i=1}^{N} \frac{1}{r_i^2} (\sin\theta_i)_m (\cos\theta_i)_m\right)^2 \tag{6.10}$$

and, by the Cauchy-Schwartz inequality [21], is always positive. Note that the parameter dependence is in the r_i. Furthermore, one can calculate

$$H^T \tilde{R}^{-1} Z = \begin{bmatrix} \sum_{i=1}^{N} \frac{1}{r_i^2} \left[x_i (\sin\theta_i)_m^2 - y_i (\sin\theta_i)_m (\cos\theta_i)_m\right] \\ \sum_{i=1}^{N} \frac{1}{r_i^2} \left[y_i (\cos\theta_i)_m^2 - x_i (\sin\theta_i)_m (\cos\theta_i)_m\right] \end{bmatrix} .$$

Hence, when N AOA measurements are accumulated, the emitter's position is explicitly calculated as follows:

$$\begin{aligned} \hat{x} = \frac{1}{\triangle} (& \left(\sum_{i=1}^{N} \frac{1}{r_i^2} (\cos\theta_i)_m^2\right) \left(\sum_{i=1}^{N} \frac{1}{r_i^2} \left[x_i (\sin\theta_i)_m^2 - y_i (\sin\theta_i)_m (\cos\theta_i)_m\right]\right) + \\ & \left(\sum_{i=1}^{N} \frac{1}{r_i^2} (\sin\theta_i)_m (\cos\theta_i)_m\right) \left(\sum_{i=1}^{N} \frac{1}{r_i^2} \left[y_i (\cos\theta_i)_m^2 - x_i (\sin\theta_i)_m (\cos\theta_i)_m\right]\right)) \\ \hat{y} = \frac{1}{\triangle} (& \left(\sum_{i=1}^{N} \frac{1}{r_i^2} (\sin\theta_i)_m^2\right) \left(\sum_{i=1}^{N} \frac{1}{r_i^2} \left[y_i (\cos\theta_i)_m^2 - y_i (\sin\theta_i)_m (\cos\theta_i)_m\right]\right) + \\ & \left(\sum_{i=1}^{N} \frac{1}{r_i^2} (\sin\theta_i)_m (\cos\theta_i)_m\right) \left(\sum_{i=1}^{N} \frac{1}{r_i^2} \left[x_i (\sin\theta_i)_m^2 - y_i (\sin\theta_i)_m (\cos\theta_i)_m\right]\right)) \end{aligned}$$

in which the expression for $\triangle$ is given in (6.10). Furthermore, the covariance of the emitter's position estimation error is a scaled version of (6.9), namely

$$P = \sigma^2 \left(H^T \tilde{R}^{-1} H \right)^{-1} = \frac{\sigma^2}{\triangle} \begin{bmatrix} \sum_{i=1}^{N} \frac{1}{r_i^2} (\cos\theta_i)_m^2 & \sum_{i=1}^{N} \frac{1}{r_i^2} (\sin\theta_i)_m (\cos\theta_i)_m \\ \sum_{i=1}^{N} \frac{1}{r_i^2} (\sin\theta_i)_m (\cos\theta_i)_m & \sum_{i=1}^{N} \frac{1}{r_i^2} (\sin\theta_i)_m^2 \end{bmatrix} \quad .$$

The filter-predicted estimation error covariance is evaluated after the parameter $\hat{\theta}$ has converged.

6.6 Conventional Approach

One should now consider the conventional assumption pertaining to azimuth angle measurements. It is assumed that the noise on the measured AOA angle is zero-mean, Gaussian-distributed

$$\phi_i = \phi_{m_i} + v_i, \qquad v_i \in \mathcal{N}(0, \sigma_\phi^2) \quad . \tag{6.11}$$

Also in the conventional approach, it is advantageous to use (6.2). Thus, by inserting (6.11) into (6.2), one obtains

$$\cos(\phi_{m_i} + v_i) y - \sin(\phi_{m_i} + v_i) x = y_i \cos(\phi_{m_i} + v_i) - x_i \sin(\phi_{m_i} + v_i) \quad .$$

Since $|v_i| \ll 1$, as shown in Table 6.1, located in Section 6.7, and according to the iterative least squares paradigm, one linearizes about ϕ_{m_i} each of the N equations according to [36], neglecting higher order terms. Specifically,

$$\begin{aligned} \cos(\phi_{m_i}) y - \sin(\phi_{m_i}) x = {} & y_i \cos(\phi_{m_i}) - x_i \sin(\phi_{m_i}) + \sin(\phi_{m_i})\, y \cdot v_i + \\ & \cos(\phi_{m_i})\, x \cdot v_i - y_i \sin(\phi_{m_i}) \cdot v_i - x_i \cos(\phi_{m_i}) \cdot v_i \quad . \end{aligned}$$

Then, one can rearrange this to fit the form of a linear regression, that is,

$$x_i \sin(\phi_{m_i}) - y_i \cos(\phi_{m_i}) = \sin(\phi_{m_i})\, x - \cos(\phi_{m_i})\, y + [(x - x_i)\cos(\phi_{m_i}) + (y - y_i)\sin(\phi_{m_i})]\, v_i \ .$$

The measurement vector Z and regressor matrix H are formed:

$$Z = \begin{bmatrix} x_1 \sin(\phi_{m_1}) - y_1 \cos(\phi_{m_1}) \\ \vdots \\ x_N \sin(\phi_{m_N}) - y_N \cos(\phi_{m_N}) \end{bmatrix}_{N\times 1}, \qquad H = \begin{bmatrix} \sin(\phi_{m_1}) & \cos(\phi_{m_1}) \\ & \vdots \\ \sin(\phi_{m_N}) & \cos(\phi_{m_N}) \end{bmatrix}_{N\times 2}$$

and $Z = H \cdot \theta + V$. The equation error covariance matrix is calculated as

$$R = \sigma_\phi^2 \cdot \mathrm{DIAG}([(x - x_i)\cos(\phi_{m_i}) - (y - y_i)\sin(\phi_{m_i})]^2)_{N\times N} \quad .$$

One should note that in conventional ILS [15], one linearizes the nonlinear observation relation $h(\theta)$ and sets $h(\hat{\theta}^{(k)}) = H \cdot \hat{\theta}^{(k)}$. Hence, one uses the simplified iteration

$$\hat{\theta}^{(k+1)} = \left(H^T R(\theta^{(k)})^{-1} H\right)^{-1} H^T R(\theta^{(k)})^{-1} Z \quad .$$

A more rigorous ILS algorithm [28] would resort to a Gauss-Newton iteration, as developed here.

The equation error covariance matrix $R(\hat{\theta})$ is dependent on the previous parameter estimate. At the instant of convergence, where $\hat{\theta}^{(k)} \to \hat{\theta}$ as $k \to \infty$, one finally calculates the filter-predicted parameter estimation error covariance matrix

$$P = (H^T R(\theta^{(\infty)})^{-1} H)^{-1} \quad .$$

At this point, it is important to highlight that there are two sources of error in the conventional ILS approach. First, the methodology forces a Gaussian-distributed assumption upon the noise of the measured azimuth angle, instead of reflecting the

physics of the AOA phasor. Hence, a modelling error is introduced. Second, because this example is not dealing with a linear relation suitable for linear regression, one needs to linearize the measurement equation. Hence, linearization-induced truncation error is introduced.

6.6.1 Augmented ILS with Unconstrained Intercept. In order to cope with the two sources of error, an approach is developed [[22] - [24], [28]] that includes an intercept parameter to account for/absorb the linearization-induced truncation errors. In [24], an unconstrained intercept is augmented to the parameter θ in order to capture the effects of linearization-induced truncation error. In the triangulation application discussed here, the unconstrained intercept would capture the effects of both modelling error and truncation error discussed in Section 6.6.

Thus, a linear regression augmented with an intercept c is considered.

$$Z = H \cdot \theta + e \cdot c + V, \qquad V \in \mathcal{N}(0, R)$$

in which

$$Z, V \in \mathbb{R}^N, \;\; \theta \in \mathbb{R}^n, \;\; c \in \mathbb{R}^1 \quad .$$

The regressor H is an N $\times$ 2 matrix, Z is the measurement vector, and $\theta = \begin{pmatrix} x \\ y \end{pmatrix}$ is the parameter vector. The equation error V is zero-mean, Gaussian-distributed with covariance R, which is a $N \times N$ real, symmetric positive definite matrix. Finally, e is the N $\times$ 1 vector of ones

$$e = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}_{N \times 1} \quad .$$

The linear regression with intercept is transformed into a standard linear regression in augmented form

$$Z = \left[H \vdots e \right] \begin{bmatrix} \theta \\ \cdots \\ c \end{bmatrix} + V \quad .$$

This equation is rewritten as

$$Z = \mathbf{H}\boldsymbol{\theta} + V$$

in which the augmented regressor $\mathbf{H} = \left[\begin{array}{ccc} H & \vdots & e \end{array} \right]$ is an $N \times 3$ matrix and the augmented parameter $\hat{\boldsymbol{\theta}} = \begin{pmatrix} x \\ y \\ c \end{pmatrix} \in \mathbb{R}^3$. The minimum variance parameter estimate [36] is

$$\hat{\boldsymbol{\theta}} = (\mathbf{H}^T R^{-1} \mathbf{H})^{-1} \mathbf{H}^T R^{-1} Z \tag{6.12}$$

and the augmented filter-predicted parameter estimation error covariance is the $(N \times 1) \times (N \times 1)$ matrix

$$\mathbf{P} = (\mathbf{H}^T R^{-1} \mathbf{H})^{-1} \quad . \tag{6.13}$$

A closed form explicit solution of a linear regression with intercept is given in Chapter III, Theorem 3 in terms of the original regressor H.

6.6.2 Augmented ILS with Constrained Intercept. As highlighted in [23] and [24], augmented ILS with an unconstrained intercept has a potential problem. Namely, the estimation error covariance increases if there are insufficient measure-

ments included in the data window. In order to address this issue, a constrained intercept c parameter is introduced into a nonlinear regression, as developed in Section 4.2. One can recall that Ridge Regression balances the linearization-induced truncation error and the measurement-induced equation error by requiring

$$\|c\| \cong \sigma$$

in which σ quantifies the intensity of the measurement noise standard deviation in the equation error. So, the augmented ILS with constrained intercept will account for error induced by linearization, but does not purport to address the modelling error caused by assuming that the AOA angle is measured instead of the phasor.

Specifically, in the bearing-only measurement estimation problem, one intercept c_ϕ is used

$$0 = c_\phi + w_\phi \qquad , \qquad w_\phi \in \mathcal{N}(0, \alpha^2 \sigma_\phi^2) \tag{6.14}$$

in which, conforming with the 1st order ergodic assumption concerning the temporal behavior of truncation error caused by linearization, one uses

$$\alpha \triangleq \frac{1}{N} \sqrt{\text{Trace}[(\hat{x} - x_i)\cos(\phi_{m_i}) - (\hat{y} - y_i)\sin(\phi_{m_i})]^2} \quad . \tag{6.15}$$

The 1st order ergodic assumption is used here to justify a complete description of the equation error noise using the first two moments, which is probably not accurate. However, this provides a constraint that is conducive to implementation. One of the problems with this constraint is that it is dependent on the parameter estimate. If the parameter has converged onto a reasonable estimate, the augmented ILS with constrained intercept via Ridge Regression will improve the estimate. However, the method is particularly vulnerable to poor estimates in this specific example. The values x_{truth} and y_{truth} are inserted into (6.15) in place of $\hat{x}$ and $\hat{y}$. For this example, the author wants to use a Ridge Regression constraint that reflects the "upper bound"

(i.e., the best one can do with this methodology). This assumption enables one to determine the impact of the two error sources in conventional ILS, namely process noise and linearization induced error. Hence, for this constraint only, the application algorithm assumes that the truth model position of the target is known. Also, the proposed "naturally linearized" formulation developed in Sections 6.3 through 6.5 produces better estimation results than Ridge Regression's theoretical "upper bound" in this application.

Ridge Regression is implemented using the two-step method developed in 4.2. For the reader's convenience, a brief explanation of the two-step method is now included. First, the estimate ($\hat{\theta}$) and its corresponding covariance matrix P_o are determined by assuming an unconstrained intercept parameter as in [24]. One should recall that the parameter estimate corresponds to that found in (6.12) with the covariance as that found in (6.13).

Next, one performs a linear regression using the estimate (6.12), covariance matrix (6.13), and enforces the Ridge Regression constraints presented in (6.14). Explicitly, one can write

$$\begin{bmatrix} \hat{\boldsymbol{\theta}} \\ 0 \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \hat{\theta} \\ \hat{c} \end{bmatrix} \\ 0 \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \theta \\ c \end{bmatrix} \\ c \end{bmatrix} + \begin{bmatrix} V \\ \\ w \end{bmatrix} \quad .$$

The linear regression components are:

$$Z = \begin{pmatrix} \hat{\theta} \\ \hat{c} \\ 0 \end{pmatrix}$$

$$H = \begin{bmatrix} I_{(2+1),(2+1)} \\ 0_{1\times 2} \ \vdots \ I_{1\times 1} \end{bmatrix}$$

$$R = \begin{bmatrix} P_{\theta(2\times 2)} & 0_{(2\times 1)} \\ 0_{(1\times 2)} & \mathrm{Diag}\left([\sigma_\phi]\right) \end{bmatrix} \quad .$$

The Ridge Regression constraint does not fully address the modelling error issue. One common approach of dealing with this issue when working with Kalman filters is to "tune" the covariance matrix R, that is, increase R in order to cover for modelling error. One can attempt to tune the static identification batch process. Also, it is difficult to tune the equation error covariance matrix R successfully in order to model the ample measurement error associated with the conventional approach to angle measurements; namely, the loss of the Gaussian distribution tails about an angle measurement.

Results from all three methods – conventional ILS, augmented ILS with unconstrained intercepts, and augmented ILS with constrained intercepts – are used for comparison with the geo-location performance of the proposed modified algorithm developed in Sections 6.3 through 6.5.

6.7 Simulation

The scenario considered entails receivers on Unmanned Air Vehicles (UAVs) or aircraft, pinged by an emitter at an unknown location. One accumulates the data from the various platforms; hence, this estimation process is cooperative and takes advantage of measurements of opportunity. It is assumed that the terrain does not

affect the data (i.e., no measurements afflicted by multi-path emissions). In the simulated scenario, the receivers are uniformly distributed within a 10 mile radius around the emitter. Without loss of generality, the emitter is located at the origin, that is, its truth position is $(0, 0)$–see Figure 6.2. Uncertainties associated with the

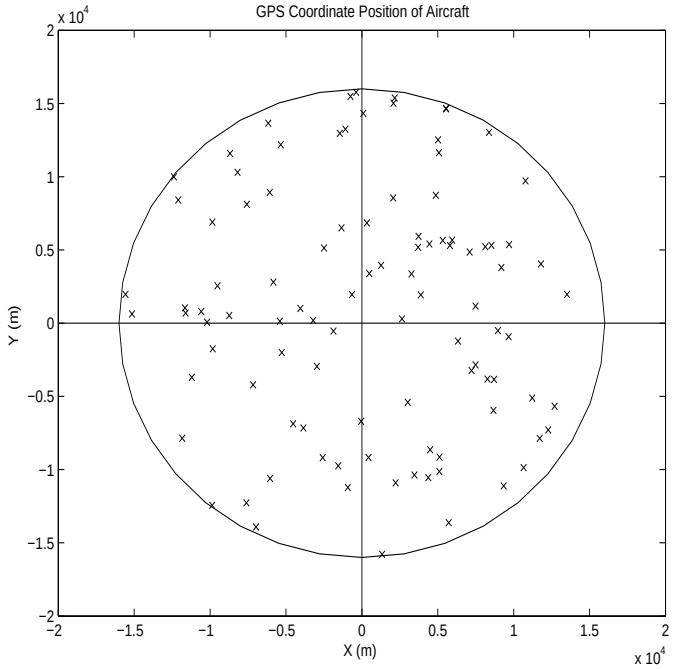


Figure 6.2: Global Positioning System coordinates receivers in typical Monte Carlo run.

measured bearing angle reflect the ranges of measurement accuracy for different types of hardware, given the nature of the pop-up emitter/threat radar in this scenario, the distances involved, and the current state of technology/ELINT hardware in use today – see Table 6.1. In general, the noise is described by $v_i,\ w_i\ \in\ \mathcal{N}(0, 10^{-\frac{SNR}{20}})$, where the Signal-to-Noise Ratio is in db, and $|\sin(\phi)| \leq 1$ and $|\cos(\phi)| \leq 1$.

The results of 10 Monte Carlo experiments with measurement error standard deviation of $\sigma = 5^o$ in azimuth are now considered. The purpose of this first set of experiments is to provide a reference of comparison for later experiments. All approaches produce the same quality of estimation performance when σ is small, but not when σ increases. As Monte Carlo experiments show, the relatively accurate measurements included in this test do not contain enough noise to create any significant

Table 6.1: Summary of Measurement Error

Measurement Error Standard Deviation (deg)	Measurement Error Standard Deviation (rad)	SNR (db)
± 5	± 0.0873	21.179
± 10	± 0.1745	15.162
± 12	± 0.2094	13.580
± 13	± 0.2269	12.883
± 15	± 0.2618	11.640
± 20	± 0.3491	9.141

differences in estimation performance between the novel ILS approach (Figure 6.3(a)) and the conventional ILS estimation techniques. Within a single Monte Carlo experiment, the estimation quality of the augmented ILS methods may produce slightly superior results over standard ILS. This reflects the fact that the error caused by the linearization is minor at worst.

One starts to see the impact of modelling error within individual Monte Carlo experiment estimation results when the measurement error standard deviation is increased to $\sigma = 10^o$. For example, one can consider the single Monte Carlo experiment result shown in Figure 6.4. The improvements in estimation quality in favor of the novel approach (Figure 6.4(a)) are due to one particularly poor measurement (outside 2σ). Most individual Monte Carlo experiments still show negligible estimation differences. The process noise effects highlighted in Figure 6.4 are washed out when the compilation of 10 Monte Carlo runs are viewed, as shown in Figure 6.5.

With $\sigma = 12^0$ in the measured azimuth angle, one can begin to see the impact of modelling error within a 10 Monte Carlo run case. In the conventional approach to AOA measurements modelling, one loses the Gaussian distribution's tails. Thus, in Figure 6.6, the novel triangulation method developed in this research clearly outperforms the standard ILS method. This advantage is accentuated as the measurement

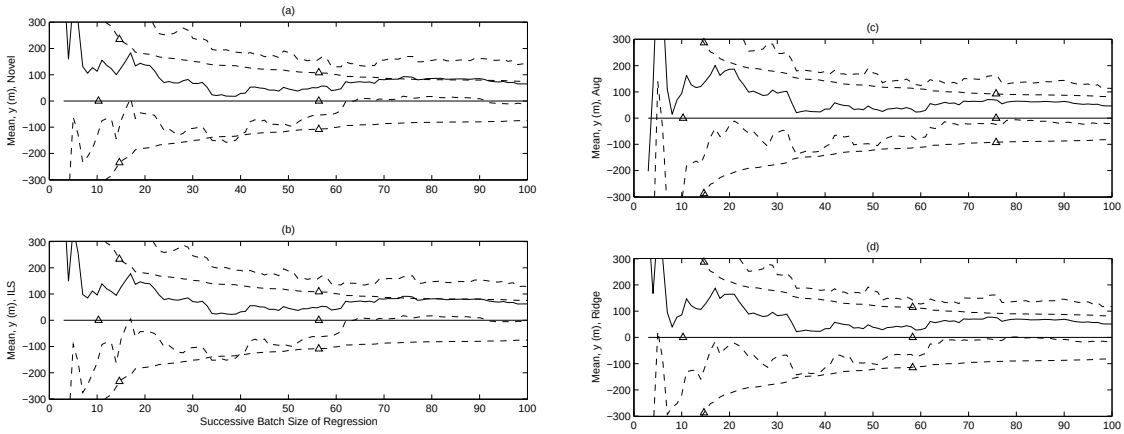


Figure 6.3: Parameter Estimate $\hat{y}$ from 10 Monte Carlo Experiments with $\sigma = 5^o$ Measurement Error Standard Deviation. (a) Novel ILS, (b) Standard ILS, (c) Augmented ILS with unconstrained intercept, (d) Augmented ILS with constrained intercept. Solid lines (no $\triangle$s)–estimates, Dashed lines (no $\triangle$s)–experimentally determined $\pm 1\sigma$ bounds, Solid lines (with $\triangle$s)–adjusted true coordinate, Dashed lines (with $\triangle$s)–$\pm 1\sigma$ bounds from the filter-predicted equation error covariance.

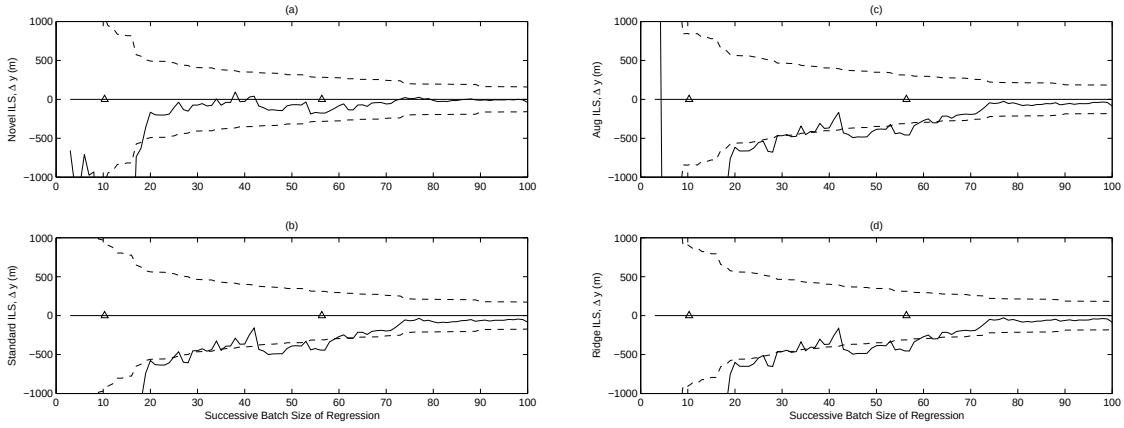


Figure 6.4: Parameter Estimate $\hat{y}$ from Single Monte Carlo Experiments with $\sigma = 10^o$ Measurement Error Standard Deviation. (a) Novel ILS, (b) Standard ILS, (c) Augmented ILS with unconstrained intercept, (d) Augmented ILS with constrained intercept. Solid lines (no $\triangle$s)–estimates, Solid lines (with $\triangle$s)–adjusted true coordinate, Dashed lines–$\pm 1\sigma$ bounds from the filter-predicted equation error covariance.

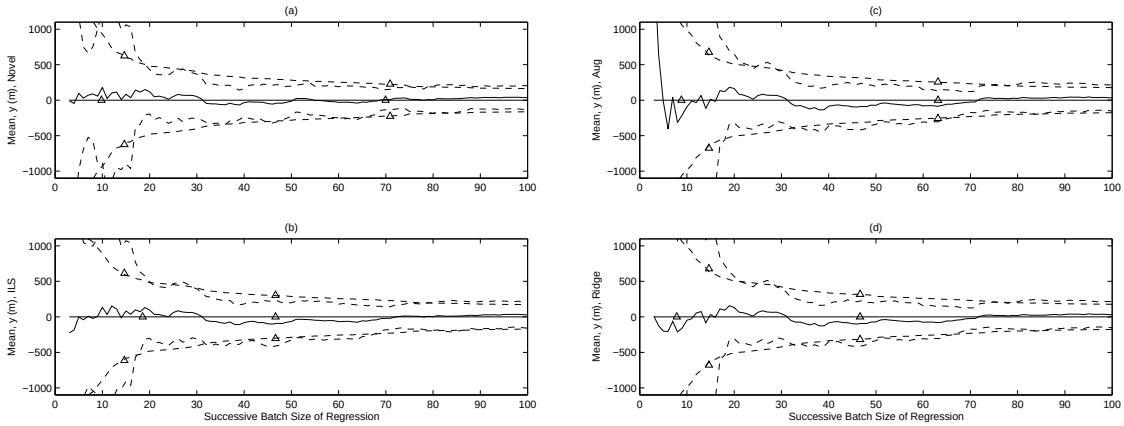


Figure 6.5: Parameter Estimate $\hat{y}$ from 10 Monte Carlo Experiments with $\sigma = 10^o$ Measurement Error Standard Deviation. (a) Novel ILS, (b) Standard ILS, (c) Augmented ILS with unconstrained intercept, (d) Augmented ILS with constrained intercept. Solid lines (no $\triangle$s)–estimates, Dashed lines (no $\triangle$s)–experimentally determined $\pm 1\sigma$ bounds, Solid lines (with $\triangle$s)–adjusted true coordinate, Dashed lines (with $\triangle$s)–$\pm 1\sigma$ bounds from the filter-predicted equation error covariance.

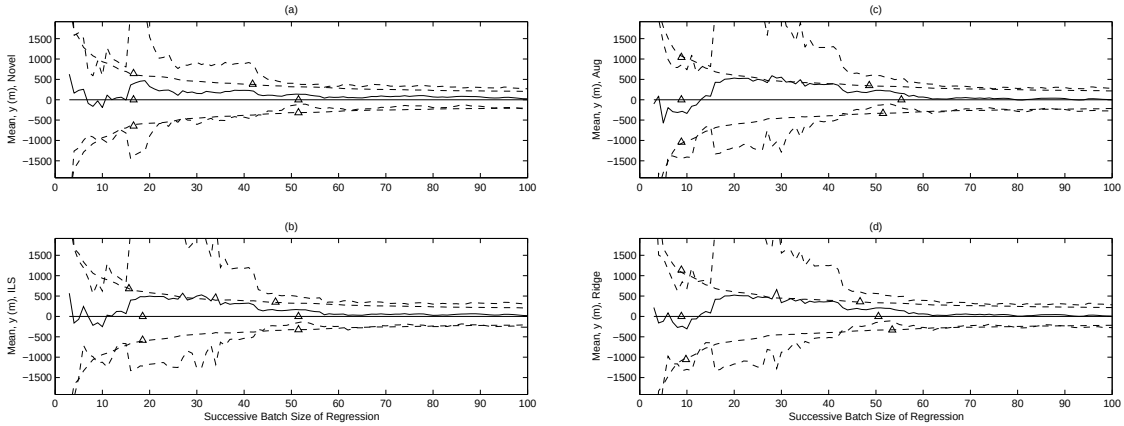


Figure 6.6: Parameter Estimate $\hat{y}$ from 10 Monte Carlo Experiments with $\sigma = 12^o$ Measurement Error Standard Deviation. (a) Novel ILS, (b) Standard ILS, (c) Augmented ILS with unconstrained intercept, (d) Augmented ILS with constrained intercept. Solid lines (no $\triangle$s)–estimates, Dashed lines (no $\triangle$s)–experimentally determined $\pm 1\sigma$ bounds, Solid lines (with $\triangle$s)–adjusted true coordinate, Dashed lines (with $\triangle$s)–$\pm 1\sigma$ bounds from the filter-predicted equation error covariance.

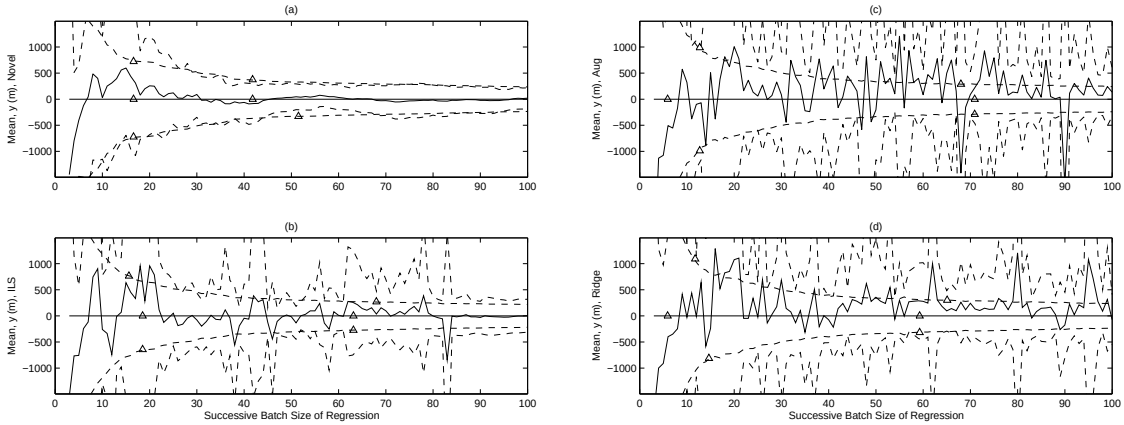


Figure 6.7: Parameter Estimate $\hat{y}$ from 10 Monte Carlo Experiments with $\sigma = 15^o$ Measurement Error Standard Deviation. (a) Novel ILS, (b) Standard ILS, (c) Augmented ILS with unconstrained intercept, (d) Augmented ILS with constrained intercept. Solid lines (no $\triangle$s)–estimates, Dashed lines (no $\triangle$s)–experimentally determined $\pm 1\sigma$ bounds, Solid lines (with $\triangle$s)–adjusted true coordinate, Dashed lines (with $\triangle$s)–$\pm 1\sigma$ bounds from the filter-predicted equation error covariance.

error standard deviation increases, as in Figure 6.7 when $\sigma = 15^0$ and in Figure 6.8 when $\sigma = 20^0$.

The constraint can somewhat mitigate the impact of modelling error such that only linearization is the real culprit, even in the case when the measurement error standard deviation is $\sigma = 20^0$. The measurement modelling error of the conventional estimation approach is accentuated when the AOA ϕ is near 90 or 270^o, in which case the measurement noise pushes the nonlinear trigonometric functions into rapid change regions. The singularity for $\tan\phi$ occurs at these specific angles. Even though $\sin\phi$ and $\cos\phi$ are immune from an outright singularity with the modified linear regression formulation, these regions foster quick changes within the trigonometric relations. So, improper handling of the noise distribution via poor modelling around these important angle regions can produce a significant deleterious effect in estimation.

To prove this point, exclusion zones are incorporated for a given set of batch data (See Figure 6.9). The exclusion zones pertain to the measured azimuth angle, so the GPS coordinates associated with the particular measured azimuth angle may actually indicate the possibility of a ϕ_{true} within the exclusion zones. To prevent a

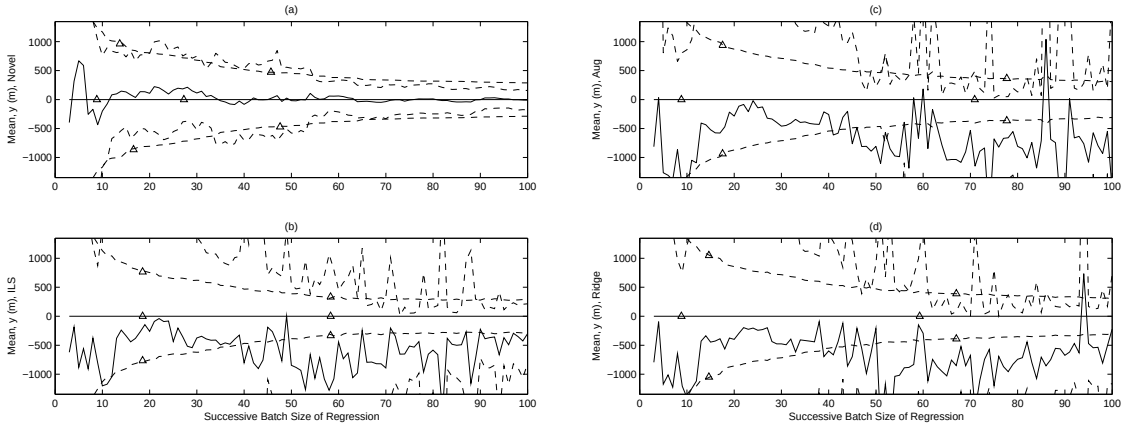


Figure 6.8: Parameter Estimate $\hat{y}$ from 10 Monte Carlo Experiments with $\sigma = 20^o$ Measurement Error Standard Deviation. (a) Novel ILS, (b) Standard ILS, (c) Augmented ILS with unconstrained intercept, (d) Augmented ILS with constrained intercept. Solid lines (no $\triangle$s)–estimates, Dashed lines (no $\triangle$s)–experimentally determined $\pm 1\sigma$ bounds, Solid lines (with $\triangle$s)–adjusted true coordinate, Dashed lines (with $\triangle$s)–$\pm 1\sigma$ bounds from the filter-predicted equation error covariance.

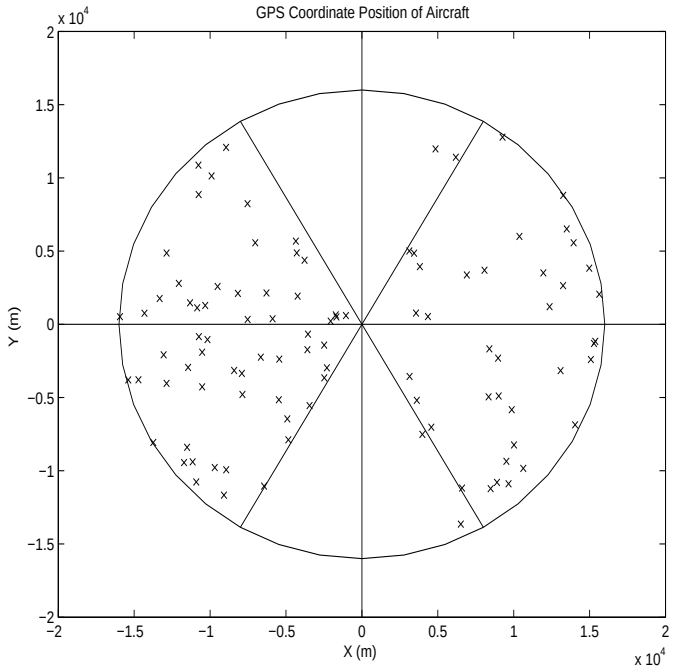


Figure 6.9: Typical Monte Carlo Run. Coordinates of receivers. Exclusion zones are enforced, that is, using measured azimuth angle sectors delimited by 60 to 120^os and 150 to 210^o.

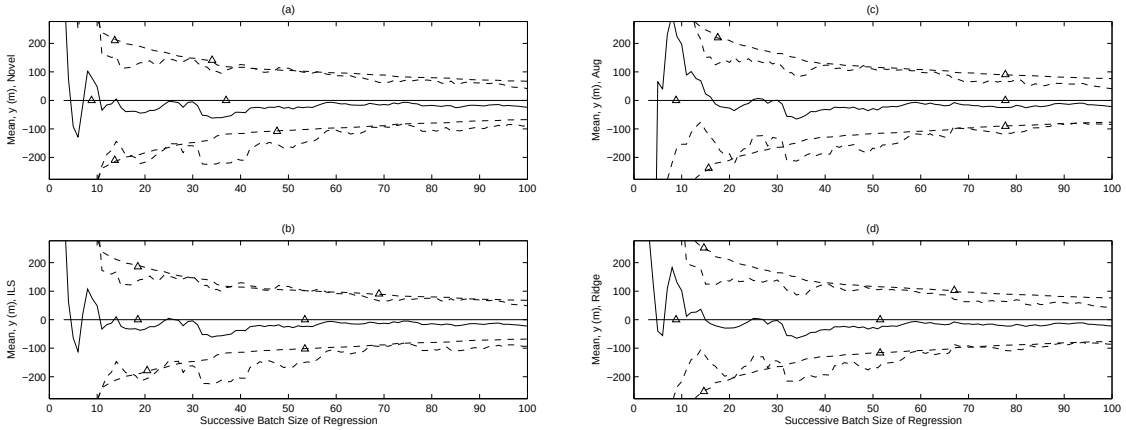


Figure 6.10: Parameter Estimate $\hat{y}$ from 10 Monte Carlo Experiments with $\sigma = 20^o$ Measurement Error Standard Deviation. Exclusion zones are enforced with a measured azimuth angle between 60 to 120^{o}s and 150 to 210^{o}s. (a) Novel ILS, (b) Standard ILS, (c) Augmented ILS with unconstrained intercept, (d) Augmented ILS with constrained intercept. Solid lines (no $\triangle$s)–estimates, Dashed lines (no $\triangle$s)–experimentally determined $\pm 1\sigma$ bounds, Solid lines (with $\triangle$s)–adjusted true coordinate, Dashed lines (with $\triangle$s)–$\pm 1\sigma$ bounds from the filter-predicted equation error covariance.

particular measurement (an extreme outlier) with a high degree of uncertainty ($\sigma = 20^0$) from corrupting the experiment, exclusion zones ($90^o \pm 30^o$ and $270^o \pm 30^o$) are used. This idea of excluding data points based on some arbitrary reference bearing is a poor idea for an estimation methodology, given that one is throwing away valuable measurement data. However, it is used here to demonstrate the vulnerability of conventional estimation approaches to modelling error.

The results for the 10 Monte Carlo experiment case with σ =20^o measurement error standard deviation and the exclusion band of $90^o \pm 30^o$ and $270^o \pm 30^o$ are shown in 6.10. Also, there are small improvements in the $N = 10$ to 20 data window for the Augmented ILS with intercept methodologies over the conventional ILS. This is a reflection of the linerization effects.

6.8 Conclusion

This chapter is concerned with geo-locating an RF emitter using bearings-only/AOA measurements, that is, triangulation. Standard approaches rely on assum-

ing azimuth angle measurements and an Iterative Least Squares estimation algorithm. Revising the conventional model of the measurement situation at hand is advocated. One recognizes that the AOA phasor, $\cos\theta$ and $\sin\theta$, is available that more accurately represents the measurement hardware.

By incorporating the phasor, the modelling error invoked by assuming azimuth angle measurements is avoided. In addition, the modified estimation approach to handling bearings-only measurements yields a linear regression formulation of the emitter geo-location problem without linearization. These assumptions enable the modified estimation algorithm to produce superior parameter estimates using measurements containing significant noise levels compared to conventional geo-location methods and ILS.

VII. Conclusions and Recommendations

This final chapter summarizes the completed work thus far and discusses potential future extensions.

7.1 Summary of Present Work

Specifically, linear regression with an intercept is considered in detail. First, an augmentation formulation is developed. The dissertation presents the closed form solution to the linear regression with intercept (Theorem 3). The research then considers a real world application pertaining to reconfigurable and adaptive flight control. In this application, an intercept is used to capture the trim change caused by a significant control surface failure. This dissertation reveals that the intercept estimate is useable as a corroborating trigger event within the aircraft flight control system's failure detection indication module. Finally, the system identification parameter estimates produced by using Theorem 3's closed form solution are used to generate a feed-forward command that automatically retrims the aircraft. Additionally, the need for integral action is reduced and pitch rate tracking is greatly enhance compared to the conventional flight control system tracking performance.

Next, nonlinear regression is addressed. This dissertation derives a Gauss-Newton-type algorithm for nonlinear regression using the closed form solution for linear regression with intercept developed in Theorem 1. The intercept accounts for linearization-induced truncation error. Ridge Regression constraints are developed in order to balance the linearization-induced truncation error with the measurement-noise-induced equation error. In addition, the dissertation develops a two-step implementation methodology for the Ridge Regression algorithm, thus providing the maximum flexibility to address nonlinear problems. At this point, two applications are considered in detail. First, the nonlinear regression methods are used in projectile launch point determination. Augmented Iterative Least Squares (ILS) with intercept is superior to standard ILS when nonlinearity is strong. Additionally, the augmented ILS algorithm with constrained intercept does not significantly degrade re-

sults in "near-linear" cases. Second, the nonlinear regression method developed herein address Radio Frequency (RF) emitter geo-location using Angle-of-Arrivals (AOAs) measurements. This dissertation recognizes the phasor information that is available in these applications. The "naturally" linearized estimation approach is superior to conventional ILS methods when high noise measurements are available.

7.2 Future Work Proposals

One area of opportunity is to explore extensions to Multiple Model Adaptive Estimation (MMAE) using the linear regression formulation. Much work in the estimation discipline is devoted to working with recursive algorithm. Perhaps one could apply some of the batch techniques developed within this dissertation and apply them to one of the more complex implementation approachs.

Second, much of the work in this dissertation is directed at proposals to handle the need of an intercept, in both linear and nonlinear applications. In the nonlinaer regression, no proofs of concept are provided. There are numerous opportunities to address these issues. One could consider the subclass of problems in which convergence is garuanteed. Opportunities to generalize the use of these methods may also be available.

In terms of the specific applications presented in this dissertation, there are two concepts that are available for pursuit. One could consider the benefits relating to estimation performance afforded by including prior information on the muzzle velocity (Chapter IV). A rough sketch of the applicable constraint is included in Appendix G. Second, as pursued in Chapter VI, angle measurements are not necessarily modelled well by simply including a Gaussian distribution, particularly in a high measurement noise environment. One should consider the novel approach revealed in Chapter VI that uses the phasor pair to avoid lineraization and apply this technique to the multiple measurement type applications, as discussed in the missile launch point determination within Chapter IV. One could explore the interplay between measurement noise

levels and the strength of nonlinearity. One could see if further estimation envelope expansion is possible over standard ILS.

Appendix A. Proof for Lemma 1

This proof is shown in numerous sources, such as [18]. The material is rederived here for completeness.

Let M_1, M_2, M_3, and M_4 be matrices such that $M_1 \in \mathbb{R}^{n\times n}$, $M_2 \in \mathbb{R}^{n\times m}$, $M_3 \in \mathbb{R}^{m\times n}$, $M_4 \in \mathbb{R}^{m\times m}$ and M_1, M_4 are invertible. One should consider the block matrix

$$M = \begin{bmatrix} M_1 & M_2 \\ M_3 & M_4 \end{bmatrix}$$

and assume that the relevant matrices are invertible. Then

$$\begin{bmatrix} M_1M_2 \\ M_3M_4 \end{bmatrix} \cdot \begin{bmatrix} M_1 & M_2 \\ M_3 & M_4 \end{bmatrix}^{-1} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}$$
$$= \begin{bmatrix} M_1 & M_2 \\ M_3 & M_4 \end{bmatrix} \cdot \begin{bmatrix} (M_1 - M_2M_4^{-1}M_3)^{-1} & -M_1^{-1}M_2(M_4 - M_3M_1^{-1}M_2)^{-1} \\ -M_4^{-1}M_3(M_1 - M_2M_4^{-1}M_3)^{-1} & (M_4 - M_3M_1^{-1}M_2)^{-1} \end{bmatrix} .$$

Multiplying these matrices, one finds:

$$= \begin{array}{l} \text{Upper LHS: } M_1(M_1 - M_2M_4^{-1}M_3)^{-1} - M_2M_4^{-1}M_3(M_1 - M_2M_4^{-1}M_3)^{-1} \\ \text{Upper RHS:} -M_1M_1^{-1}M_2(M_4 - M_3M_1^{-1}M_2)^{-1} + M_2(M_4 - M_3M_1^{-1}M_2)^{-1} \\ \text{Lower LHS: } M_3(M_1 - M_2M_4^{-1}M_3)^{-1} - M_4M_4^{-1}M_3(M_1 - M_2M_4^{-1}M_3)^{-1} \\ \text{Lower RHS:} -M_3M_1^{-1}M_2(M_4 - M_3M_1^{-1}M_2)^{-1} + M_4(M_4 - M_3M_1^{-1}M_2)^{-1} \end{array}$$

$$= \begin{array}{ll} \text{Upper LHS:} & (M_1 - M_2M_4^{-1}M_3)(M_1 - M_2M_4^{-1}M_3)^{-1} \\ \text{Upper RHS:} & M_2(M_4 - M_3M_1^{-1}M_2)^{-1} - M_2(M_4 - M_3M_1^{-1}M_2)^{-1} \\ \text{Lower LHS:} & M_3(M_1 - M_2M_4^{-1}M_3)^{-1} - M_3(M_1 - M_2M_4^{-1}M_3)^{-1} \\ \text{Lower RHS:} & (M_4 - M_3M_1^{-1}M_2)(M_4 - M_3M_1^{-1}M_2)^{-1} \end{array}$$

$$= \begin{bmatrix} I0 \\ 0I \end{bmatrix} .$$

Hence,

$$\begin{bmatrix} M_1 M_2 \\ M_3 M_4 \end{bmatrix}^{-1} = \begin{bmatrix} (M_1 - M_2 \cdot M_4^{-1} \cdot M_3)^{-1} & -M_1^{-1} \cdot M_2 (M_4 - M_3 \cdot M_1^{-1} \cdot M_2)^{-1} \\ -M_4^{-1} \cdot M_3 (M_1 - M_2 \cdot M_4^{-1} \cdot M_3)^{-1} & (M_4 - M_3 \cdot M_1^{-1} \cdot M_2)^{-1} \end{bmatrix}$$

as shown in (3.5) $\square$.

Appendix B. Proof for Lemma 2

This proof is shown in numerous sources, such as [18]. The proof is rederived here for completeness.

$$
\begin{aligned}
&(A - BD^{-1}C) \cdot (A - BD^{-1}C)^{-1} = I \\
&\quad = (A - BD^{-1}C)^{-1} \cdot [A^{-1} + A^{-1}B(D - CA^{-1}B)^{-1}CA^{-1}] \\
&\quad = AA^{-1} + AA^{-1}B(D - CA^{-1}B)^{-1}CA^{-1} - BD^{-1}CA^{-1} - BD^{-1}CA^{-1}B(D - CA^{-1}B)^{-1}CA^{-1} \\
&= I + (B - BD^{-1}CA^{-1}B) \cdot (D - CA^{-1}B)^{-1}CA^{-1} - BD^{-1}CA^{-1} \\
&= I + B(I - D^{-1}CA^{-1}B) \cdot (D - CA^{-1}B)^{-1}CA^{-1} - BD^{-1}CA^{-1} \quad .
\end{aligned}
$$

If one assumes that D is a nonzero scalar, then

$$
\begin{aligned}
&= I + BD^{-1}D(I - D^{-1}CA^{-1}B) \cdot (D - CA^{-1}B)^{-1}CA^{-1} - BD^{-1}CA^{-1} \\
&= I + BD^{-1}(D - DD^{-1}CA^{-1}B) \cdot (D - CA^{-1}B)^{-1}CA^{-1} - BD^{-1}CA^{-1} \\
&\quad = I + BD^{-1}ICA^{-1} - BD^{-1}CA^{-1} \\
&\quad = I + 0 \\
&\quad = I \quad .
\end{aligned}
$$

Hence,

$$(A - BD^{-1}C)^{-1} = A^{-1} + A^{-1}B(D - CA^{-1}B)^{-1}CA^{-1}$$

as stated in (3.10).

Appendix C. Special Case of Linear Regression with Intercept: Approach I

One can consider the special case where $R = \theta^2$. Since this is a scalar quantity, one can simplify the resulting parameter estimates $\hat{\theta}$ and $\hat{c}$. The scalar quantity a is shown in (3.9), recalling that

$$a = \frac{1}{e^T R^{-1} e - e^T R^{-1} H (H^T R^{-1} H)^{-1} H^T R^{-1} e} \quad .$$

This special case assumption implies that

$$a = \frac{1}{e^T \frac{1}{\theta^2} e - e^T \frac{1}{\theta^2} H (H^T \frac{1}{\theta^2} H)^{-1} H^T \frac{1}{\theta^2} e} \quad .$$

Since θ is a scalar, the quantity is factored out of the equation. Also, the reader can observe that $e \cdot e^T = N$, where N is the number of observations in the batch estimation.

$$a = \frac{\theta^2}{N - e^T H (H^T H)^{-1} H^T e} \quad .$$

This result was stated in (3.14).

Similarly, one can consider the vector x shown in (3.8), recalling that

$$x = -a (H^T R^{-1} H)^{-1} H^T R^{-1} e \quad .$$

One can insert the scalar θ^2 for the equation error covariance R as before. The simplified result for a as shown in (3.14) is inserted.

$$x = -\frac{\theta^2}{N - e^T H (H^T H)^{-1} H^T e} (H^T \frac{1}{\theta^2} H)^{-1} H^T \frac{1}{\theta^2} e \quad .$$

This equation simplifies to the result shown in (3.15).

$$x = -\frac{\theta^2}{N - e^T H (H^T H)^{-1} H^T e} (H^T H)^{-1} H^T e \quad .$$

Also, since $(H^T H)^{-T} = (H^T H)^{-1}$, x^T is

$$x^T = -\frac{\theta^2}{N - e^T H(H^T H)^{-1} H^T e} e^T H(H^T H)^{-1} \quad . \tag{C.1}$$

Now, the reduced form of the parameter and bias estimates is pursued. Returning to (3.3), one can combine the results from (3.6) to obtain the estimate:

$$\hat{\theta} = (XH^T + xe^T)R^{-1}z \quad .$$

Applying the result from (3.11), one obtains

$$\hat{\theta} = (H^T R^{-1} H)^{-1} H^T R^{-1} z + x(\frac{1}{a} x^T H^T + e^T) R^{-1} z \quad . \tag{C.2}$$

Starting with the result from (C.2), one can insert the scalar for the equation error covariance R and the simplified results for a (3.14), x (3.15), and x^T (C.1).

$$\hat{\theta} = (H^T \tfrac{1}{\theta^2} H)^{-1} H^T \tfrac{1}{\theta^2} z - \tfrac{\theta^2}{N - e^T H(H^T H)^{-1} H^T e} (H^T H)^{-1} H^T e \cdot$$
$$\left[\tfrac{N - e^T H(H^T H)^{-1} H^T e}{\theta^2} \cdot \left[- \tfrac{\theta^2}{N - e^T H(H^T H)^{-1} H^T e} e^T H(H^T H)^{-1} H^T \right] + e^T \right] \frac{1}{\theta^2} z \quad .$$

This equation reduces to

$$\hat{\theta} = (H^T H)^{-1} H^T z - \tfrac{1}{N - e^T H(H^T H)^{-1} H^T e} \cdot (H^T H)^{-1} H^T e \left[e^T H(H^T H)^{-1} H^T - e^T \right] z \ .$$

This equation is reorganized to match the result in (3.16).

$$\hat{\theta} = (H^T H)^{-1} H^T \{ I - \frac{1}{N - e^T H(H^T H)^{-1} H^T e} \cdot e e^T [I - H(H^T H)^{-1} H^T] \} z \quad .$$

Similarly for the intercept $\hat{c}$, one should consider (3.3) and apply the result from (3.6)

$$\hat{c} = (x^T H^T + a e^T) R^{-1} z \quad . \tag{C.3}$$

The bias estimate equation (C.3) is rearranged

$$\hat{c} = a(\frac{1}{a} x^T H^T + e^T) R^{-1} z \quad .$$

By inserting the scalar for the equation error covariance $R = \theta^2$, the simplified result for a (3.14) and the expression for x^T (C.1), the equation becomes

$$\hat{c}=\frac{\theta^2}{N-e^T H(H^T H)^{-1}H^T e}\Big[\frac{N-e^T H(H^T H)^{-1}H^T e}{\theta^2}\cdot \Big(-\frac{\theta^2}{N-e^T H(H^T H)^{-1}H^T e}e^T H(H^T H)^{-1}\Big)H^T + e^T\Big]\frac{1}{\theta^2}z \quad . \tag{C.4}$$

This equation reduces to the result expressed in (3.17).

$$\hat{c} = \frac{1}{N-e^T H(H^T H)^{-1}H^T e}e^T[I - H(H^T H)^{-1}H^T]\}z \quad .$$

Appendix D. Solution to Example Ordinary Differential Equation

The dynamics shown in (3.21) are considered.

$$\ddot{x} + \omega^2 x = c \quad .$$

One can solve the following linear differential equation:

$$\ddot{x} + \omega^2 x = c \quad .$$

In order to find the homogeneous solution:

$$\begin{aligned} \ddot{x} + \omega^2 x &= 0 \\ \lambda_1 &= i\omega \\ \lambda_2 &= -i\omega \quad . \end{aligned}$$

So $x_h = C_1 e^{i\omega t} + C_2 e^{-i\omega t}$. As described in [6], this is equivalent to:

$$\begin{aligned} x_h(t) &= Ae^0 \cos(\omega t) + Be^0 \sin(\omega t) \\ x_h(t) &= A\cos(\omega t) + B\sin(\omega t) \end{aligned}$$

Now, solving for the particular solution, one can guess that $x_p = C_3$, therefore

$$x_p' = 0$$
$$x_p'' = 0$$

so

$$x_p'' + \omega^2 x_p = c$$

becomes

$$0 + \omega^2 C_3 = c$$

which implies that

$$C_3 = \frac{c}{\omega^2} \quad .$$

So, the particular solution is:

$$x_p = \frac{c}{\omega^2} \quad .$$

The general solution is

$$x(t) \simeq A\cos(\omega t) + B\sin(\omega t) + \frac{C}{\omega^2} \quad .$$

If one assumes $\dot{x}(0) = 0$, then $B = 0$ and

$$x(t) \simeq A\cos(\omega t) + \frac{C}{\omega^2}$$

is the solution to the differential equation with the single initial condition. One can also note that

$$x'(t) = -A\omega\sin(\omega t)$$

and

$$x''(t) = -A\omega^2\cos(\omega t) \quad .$$

Hence, the equations match (3.22) and (3.23).

Appendix E. Nonlinear Regression Augmented With an Intercept

The solution of the augmented linear regression shown in (4.8) is considered and restated here

$$\hat{\boldsymbol{\theta}}^{(i+1)} = (\mathbf{H}_i^T R^{-1} \mathbf{H}_i)^{-1} \mathbf{H}_i^T R^{-1} [Z + H_i \hat{\theta}^{(i)} - h(\hat{\theta}^{(i)})]$$

in which the iteration counter is $i = 0, 1, \ldots$. Expanding this equation, one can observe that

$$\begin{pmatrix} \hat{\theta}^{(i+1)} \\ \hat{c}^{(i+1)} \end{pmatrix} = \begin{bmatrix} H_i^T R^{-1} H_i & H_i^T R^{-1} e \\ e_i^T R^{-1} H_i & e_i^T R^{-1} e \end{bmatrix}^{-1} \begin{pmatrix} H_i^T R^{-1} \\ e^T R^{-1} \end{pmatrix} \left[z + H_i \hat{\theta}^{(i)} - h(\hat{\theta}^{(i)}) \right] \quad .$$

The derivation requires

Lemma E.1

Let A be an $N \times N$ real symmetric matrix, b be an $N \times 1$ real vector and $\alpha \in \mathbb{R}^1$. Let $\left(A - \frac{1}{\alpha} b b^T\right)^{-1}$ exist. Then the inverse of the matrix is

$$\begin{pmatrix} A & b \\ b^T & \alpha \end{pmatrix}^{-1} = \begin{pmatrix} X & x \\ x^T & \xi \end{pmatrix}$$

in which

$$\begin{aligned} X &= \left(A - \frac{1}{\alpha} b b^T\right)^{-1} \\ x &= \left(b b^T - \alpha A\right)^{-1} b \\ \xi &= \frac{1}{\alpha} \left[1 - b^T \left(b b^T - \alpha A\right)^{-1} b\right] \quad . \end{aligned}$$

Proof

Multiplying the two matrices yields the identity matrix as

$$\begin{pmatrix} A & b \\ b^T & \alpha \end{pmatrix} \begin{pmatrix} X & x \\ x^T & \xi \end{pmatrix} = \begin{pmatrix} I & 0 \\ 0 & 1 \end{pmatrix}$$

□

which implies that

$$AX + bx^T = I \tag{E.1}$$

$$Ax + \xi b = 0 \tag{E.2}$$

$$b^T X + \alpha x^T = 0$$

$$b^T x + \alpha \xi = 1 \quad . \tag{E.3}$$

Equation (E.3) and $\alpha \neq 0$ implies that

$$\xi = \frac{1 - b^T x}{\alpha} \quad . \tag{E.4}$$

Using (E.2), (E.4), and the assumption that $\left(A - \frac{1}{\alpha} b b^T\right)^{-1}$ exists, yields

$$x = (bb^T - \alpha A)^{-1} b \quad . \tag{E.5}$$

Next, Equations (E.4) and (E.5) are merged to produce

$$\xi = \frac{1}{\alpha} \left[1 - b^T \left(bb^T - \alpha A\right)^{-1} b\right] \quad .$$

Finally, one can insert (E.1) into (E.5) and rearrange the equation

$$
\begin{aligned}
AX &= I - bb^T \left(bb^T - \alpha A^T\right)^{-1} \\
&= I - bb^T \left(bb^T - \alpha A\right)^{-1} \\
&= I - \alpha A \left(bb^T - \alpha A\right)^{-1} + \alpha A \left(bb^T - \alpha A\right)^{-1} - bb^T \left(bb^T - \alpha A\right)^{-1} \\
&= I - \alpha A \left(bb^T - \alpha A\right)^{-1} + \left(\alpha A - bb^T\right) \left(bb^T - \alpha A\right)^{-1} \\
&= I - \alpha A \left(bb^T - \alpha A\right)^{-1} - \left(bb^T - \alpha A\right) \left(bb^T - \alpha A\right)^{-1} \\
&= I - \alpha A \left(bb^T - \alpha A\right)^{-1} - I \\
&= -\alpha A \left(bb^T - \alpha A\right)^{-1} \\
&= A \left(A - \frac{1}{\alpha} bb^T\right)^{-1} \quad .
\end{aligned}
$$

Hence,

$$
X = \left(A - \frac{1}{\alpha} bb^T\right)^{-1} \quad .
$$

Using the substitutions: $A = H_i^T R^{-1} H_i$, $b = H_i^T R^{-1} e$, and $\alpha = e^T R^{-1} e$, **Lemma E.1** yields

$$
\begin{bmatrix} H_i^T R^{-1} H_i & H_i^T R^{-1} e \\ e_i^T R^{-1} H_i & e_i^T R^{-1} e \end{bmatrix}^{-1} = \begin{bmatrix} ULHS & URHS \\ LLHS & LRHS \end{bmatrix} \tag{E.6}
$$

in which

$$
\begin{aligned}
ULHS &= \left[H_i^T R^{-1} \left(R - \tfrac{1}{e^T R^{-1} e} e e^T\right) R^{-1} H_i\right]^{-1} \\
URHS &= -\left[H_i^T R^{-1} \left(\tfrac{R}{e^T R^{-1} e} - e e^T\right) R^{-1} H_i\right]^{-1} H_i^T R^{-1} e \\
LLHS &= -e^T R^{-1} H_i \left[H_i^T R^{-1} \left(\tfrac{R}{e^T R^{-1} e} - e e^T\right) R^{-1} H_i\right]^{-1} \\
LRHS &= \tfrac{1}{e^T R^{-1} e} + \left(\tfrac{1}{e^T R^{-1} e}\right)^2 e^T R^{-1} H_i \cdot \left[H_i^T R^{-1} \left(R - \tfrac{1}{e^T R^{-1} e} - e e^T\right) R^{-1} H_i\right]^{-1} H_i^T R^{-1} e \, .
\end{aligned}
$$

Finally, one can insert (E.6) into (E.1) in order to yield the expanded algorithms for the parameter estimate $\hat{\theta}$ and the intercept estimate $\hat{c}$:

$$
\begin{aligned}
\hat{\theta}^{(i+1)}=&\Big(\left[H_i^T R^{-1} \left(R - \tfrac{1}{e^T R^{-1} e} e e^T\right) R^{-1} H_i\right]^{-1} H_i^T \\
&- \left[H_i^T R^{-1} \left(\tfrac{1}{e^T R^{-1} e} R - e e^T\right) R^{-1} H_i\right]^{-1} \cdot \\
&H_i^T R^{-1} e e^T) R^{-1} \left[Z + H_i \hat{\theta}^{(i)} - h(\hat{\theta}^{(i)})\right] \\
\hat{c}^{(i+1)}=&e^T R^{-1} (I[\tfrac{1}{e^T R^{-1} e} + \left(\tfrac{1}{e^T R^{-1} e}\right)^2 e^T R^{-1} H_i \cdot \\
&\left[H_i^T R^{-1} \left(R - \tfrac{1}{e^T R^{-1} e} e e^T\right) R^{-1} H_i\right]^{-1} H_i^T R^{-1} e] \\
&- H_i \left[H_i^T R^{-1} \left(\tfrac{1}{e^T R^{-1} e} R - e e^T\right) R^{-1} H_i\right]^{-1}) \\
&\cdot \left[Z + H_i \hat{\theta}^{(i)} - h(\hat{\theta}^{(i)})\right]
\end{aligned}
$$

Appendix F. Additional Results: Standard ILS and Augmented ILS with Unconstrained Intercept

The following results correspond to the 20 Monte Carlo Run example discussed in Section 4.5.3.1. This simulation contained significant nonlinearity due the problem geometry. Now, the author can determine the time estimates (Figure F.1) generated by (4.31) for each batch algorithm. Using this estimate for $\hat{t}$, one can propagate

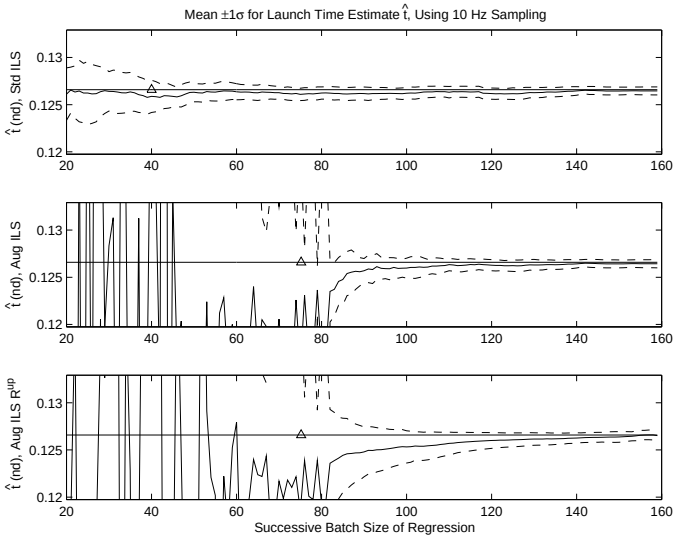


Figure F.1: Launch time estimate $\hat{t}$, as determined using (4.31).

the sub-batch estimates shown in Figures 4.8 and 4.9 back to the time of launch. Figure 4.10 shows the result for the position parameter at launch. Figure F.2 shows the result for the velocity parameter at launch. In summary, this example shows a case in which envelope of successful estimation is given, while the magnitude of the estimated equation error covariance is improved over standard ILS, even when the window size is small. All plots show the inability of the standard ILS to provide reasonable parameter estimates when the nonlinearity becomes significant.

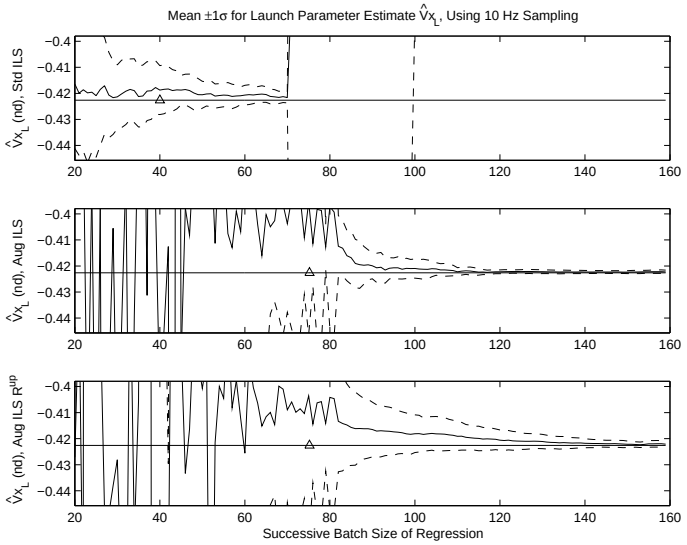


Figure F.2: Result of 20 Monte Carlo Runs. Within each plot, the solid line show the launch velocity parameter estimate, $\hat{V}_{x_L} \pm$ the experimentally determined 1σ from the variation between the parameter estimates (dashed lines). $V_{x_{L_{true}}}$ is also shown (solid line with $\triangle$).

Appendix G. Appending Known Muzzle Velocity Constraint

If the projectile type is known, then it is reasonable to assume that the muzzle velocity V is known. In this case, one can incorporate the following constraint:

$$V = \sqrt{V_{x_o}^2 + V_{y_o}^2 + V_{z_o}^2} \quad .$$

Linearization of the nonlinear constant velocity equation yields

$$V = \frac{1}{\sqrt{\hat{V}_{x_o}^2 + \hat{V}_{y_o}^2 + \hat{V}_{z_o}^2}} \left(\hat{V}_{x_o}, \hat{V}_{y_o}, \hat{V}_{z_o} \right) \begin{pmatrix} V_{x_o} \\ V_{y_o} \\ V_{z_o} \end{pmatrix} \quad .$$

A noise term v is included to reflect the uncertainty in V in order to keep the regression nonsingular. Thus,

$$V = \frac{1}{\sqrt{\hat{V}_{x_o}^2 + \hat{V}_{y_o}^2 + \hat{V}_{z_o}^2}} \left(\hat{V}_{x_o}, \hat{V}_{y_o}, \hat{V}_{z_o} \right) \begin{pmatrix} V_{x_o} \\ V_{y_o} \\ V_{z_o} \end{pmatrix} + v$$

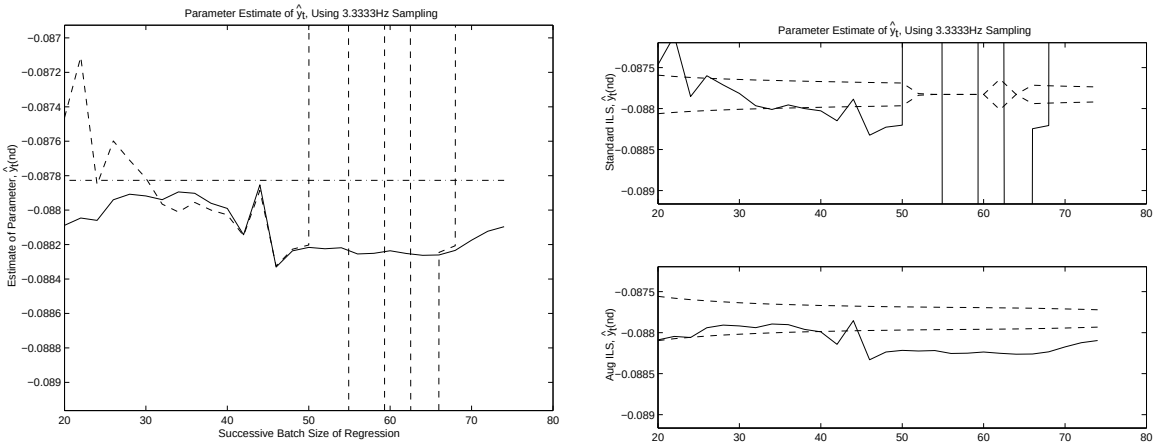
in which

$$v \in \mathcal{N}(0, \sigma^2) \quad .$$

Appendix H. Additional Results: Standard ILS and Augmented ILS with Constrained Intercept

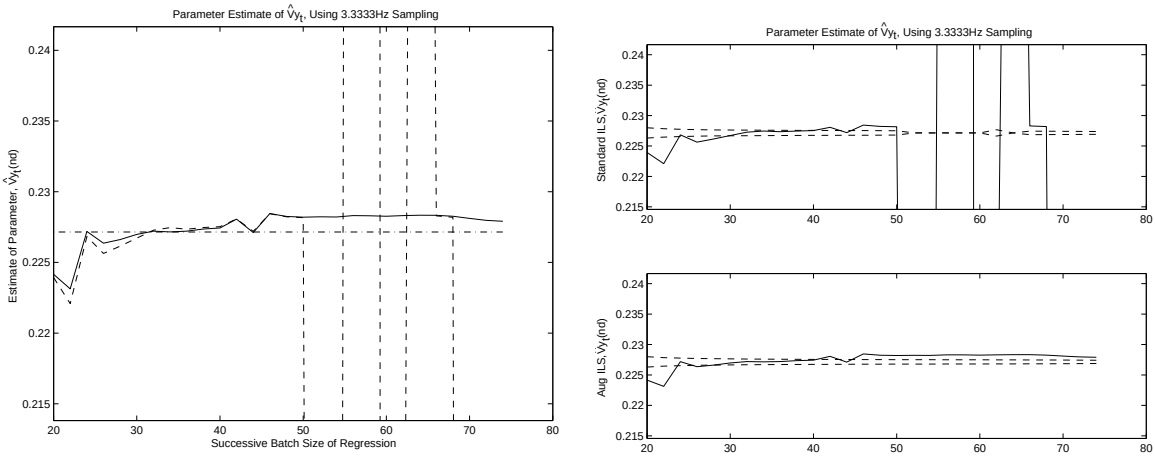
H.1 Single Monte Carlo Run Example: Trajectory with Significant Nonlinearity

The following results correspond to the single Monte Carlo Run example discussed in Section 4.5.3.1. This simulation contains significant nonlinearity due the problem geometry. Results in Figures H.1 through H.4 continue to support the advocacy for using the augmented Iterative Least Squares process with constrained intercepts using Ridge Regression. This example shows a case in which envelope of successful estimation is given, while the magnitude of the estimated equation error covariance is improved over standard ILS, even when the window size is small. All plots show the inability of the standard ILS algorithm to provide a reasonable parameter estimate when the nonlinearity becomes significant.



(a) (b)

Figure H.1: (a) Position parameter estimate $\hat{y}_t$ using constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS is also shown (dashed line). The truth model's value for y at $t = 0$ is pictured at -0.0878 nondimensional distance units (dash-dot line). (b) Position parameter estimate $\hat{y}_t$ using an constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS is also shown. Dashed lines represent plus minus one standard deviation from the filter-predicted equation error covariance from the true y (dash-dot lines).



(a) Harmonic Oscillation. (b) Harmonic Acceleration Measurement.

Figure H.2: (a) Velocity parameter estimate $\hat{V}_{y_t}$ using constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS is also shown (dashed line). The truth model's value for V_y at $t = 0$ is pictured as the straight line at 0.227 nondimensional velocity units. (b) Velocity parameter estimate $\hat{V}_{y_t}$ using an constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS is also shown. Dashed lines represent plus minus one standard deviation from the filter-predicted equation error covariance from the true V_y (dash-dot lines).

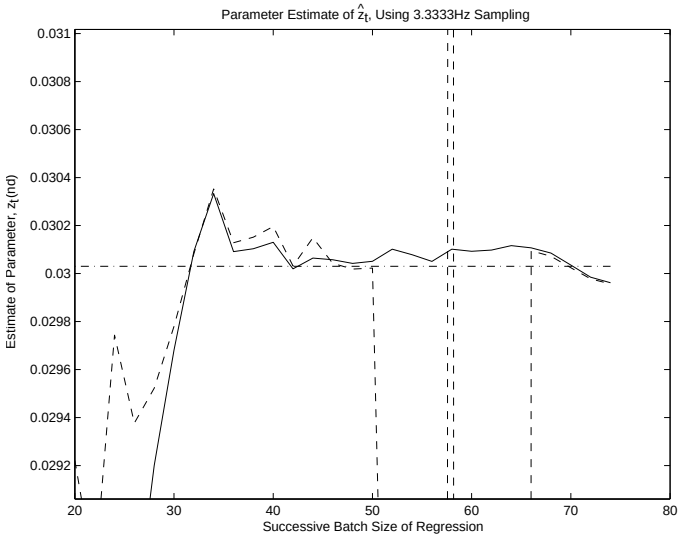


Figure H.3: Position parameter estimate $\hat{z}_t$ using constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS is also shown (dashed line). The truth model's value for z at $t = 0$ is pictured at 0.03002 nondimensional distance units (dash-dot line).

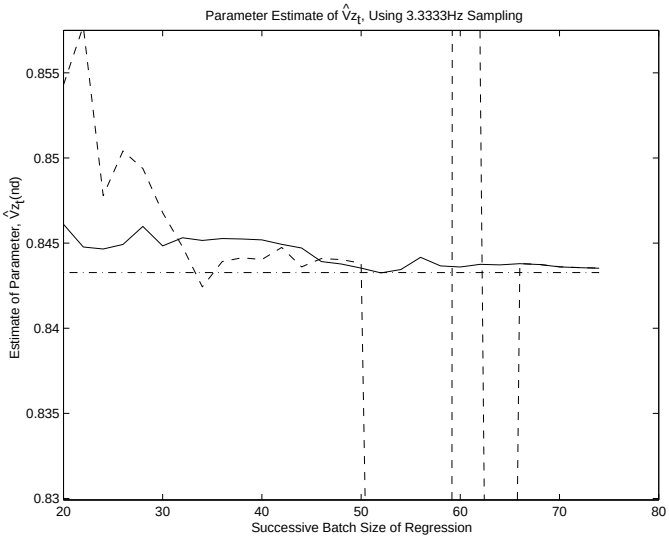


Figure H.4: Velocity parameter estimate $\hat{V}_{z_t}$ using constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS is also shown (dashed line). The truth value for V_z at t=0 is shown at 0.847 nondimensional velocity units (dash-dot line).

H.2 Twenty Monte Carlo Run Examples: Trajectory with Significant Nonlinearity

The results from 20 Monte Carlo runs shown in Figures 4.19 and 4.20 clearly demonstrate superior estimation performance achieved by ILS augmented with constrained intercepts for range, azimuth, and elevation, over that of standard ILS. The remaining two parameter plots of interest are included here in Figures H.5 and H.6.

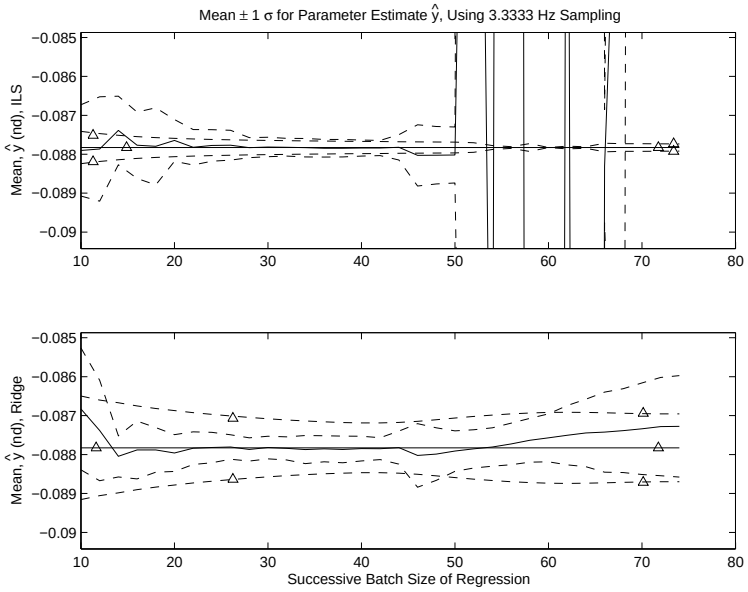


Figure H.5: 20 Monte Carlo Experiments. Position estimates produced by the augmented ILS algorithm with constrained intercepts are superior to estimates produced by conventional ILS when nonlinearity is strong. The figure shows: (1) $y_{o\text{true}} = -0.0878nd \pm 1\sigma$ determined from the filter-predicted covariance estimation error (lines with $\triangle$s). (2) Parameter estimate $\hat{y}_o \pm$ experimentally determined standard deviation 1σ from the variation between the parameter estimates within the 20 Monte Carlo runs. (lines without $\triangle$s).

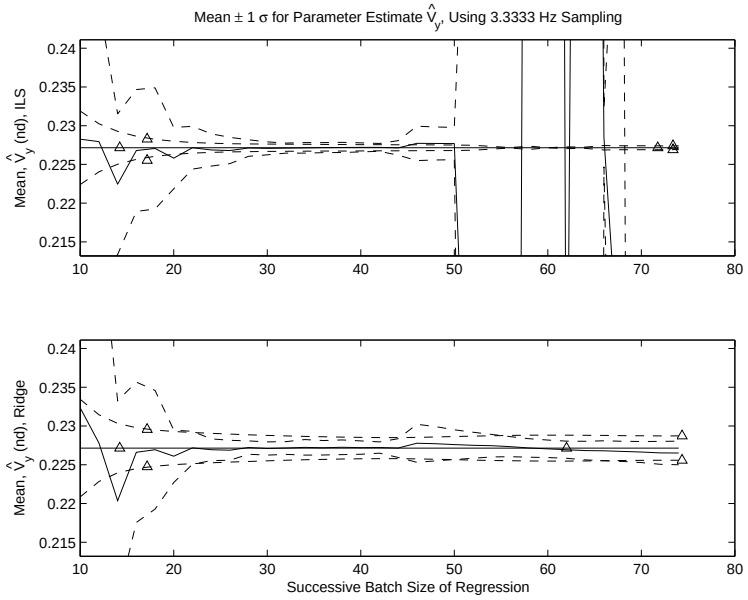


Figure H.6: 20 Monte Carlo Experiments. Velocity estimates produced by the augmented ILS algorithm with constrained intercepts are superior to estimates produced by conventional ILS when nonlinearity is strong. The figure shows: (1) $V_{y_o\text{true}} = 0.227nd \pm 1\sigma$ determined from the filter-predicted covariance estimation error (lines with $\triangle$s). (2) Parameter estimate $\hat{V}_{y_o} \pm$ experimentally determined 1σ from the variation between the parameter estimates within the 20 Monte Carlo runs (lines without $\triangle$s).

H.3 Single Monte Carlo Run Example: Near-Linear Sub-batch of Aggressive Trajectory

The following results correspond to the single Monte Carlo Run example discussed in Section 4.5.3.2. This simulation reveals a sub-batch of interest that is a near-linear part of a rapidly changing portion of the overall trajectory. Figure 4.24 shows the position parameter estimate $\hat{y}$ for the particular sub-batch of interest. Figures H.7 through H.9 show the other parameter estimates, namely $\hat{x}_o$, $\hat{V}_{x_o}$, and $\hat{V}_{y_o}$.

Again, a significant point here is that augmented ILS with constrained intercept

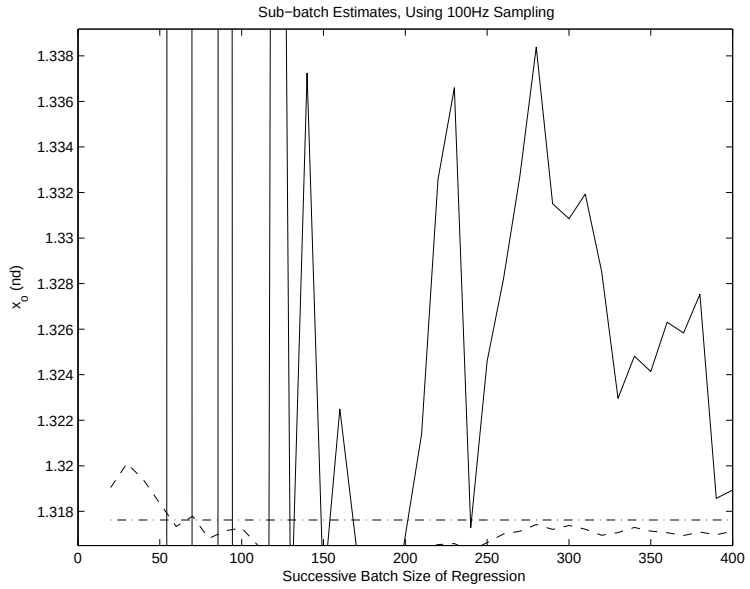


Figure H.7: Position parameter estimate $\hat{x}_o$ for particular sub-batch using augmented ILS with constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS estimate is also shown (dashed line). The truth model's value for x at $t = 0.178$ non-dimensional time units is pictured as the straight line at 1.318 non-dimensional position units (dash-dot line).

provides parameter estimates that do not degrade when compared to standard ILS, even when the nonlinearities are relatively weak compared to the measurement noise intensity, as presented here in the near overflight case.

Next, using the launch time estimates determined in Figure 4.25, one can determine $\hat{x}_L$, $\hat{V}_{x_L}$, and $\hat{V}_{y_L}$ at launch, as shown in Figures H.10 through H.12. One should recall that $\hat{y}_L$ is shown previously in Figure 4.26.

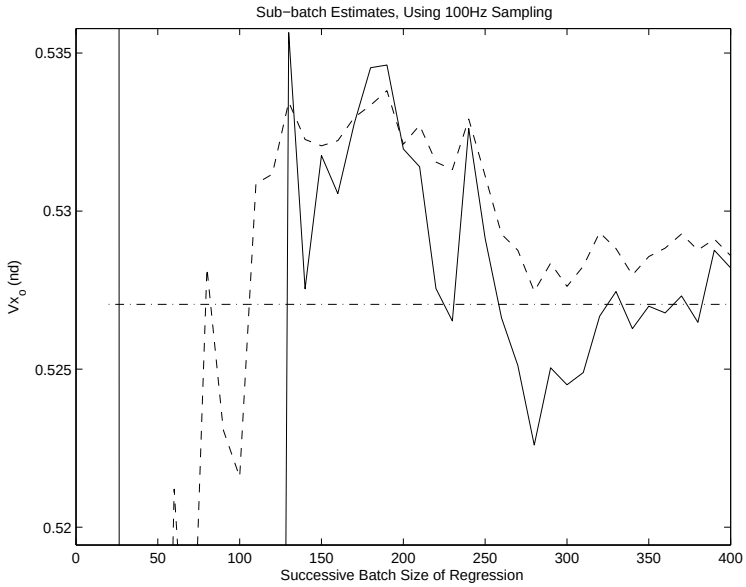


Figure H.8: Velocity parameter estimate $\hat{V}_{x_o}$ for particular sub-batch using augmented ILS with constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS estimate is also shown (dashed line). The truth model's value for V_x at $t = 0.178$ non-dimensional time units is pictured as the straight line at 0.527 non-dimensional velocity units (dash-dot line).

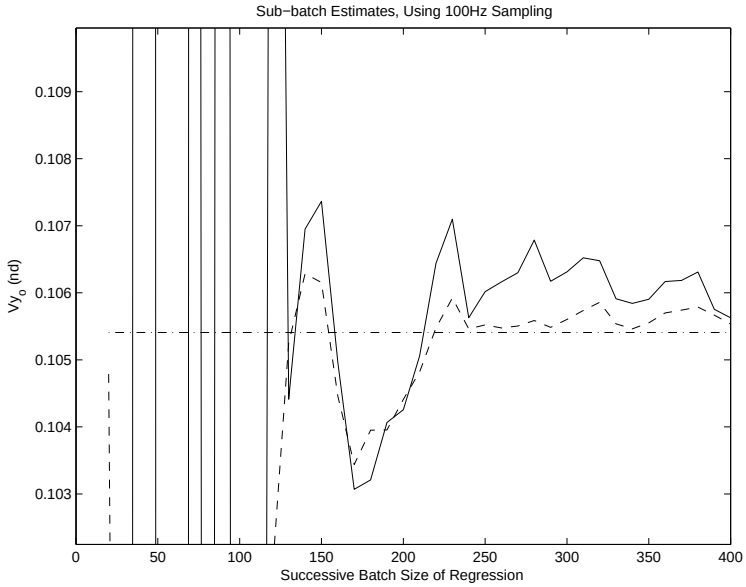


Figure H.9: Velocity parameter estimate $\hat{V}_{y_o}$ for particular sub-batch using augmented ILS with constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS estimate is also shown (dashed line). The truth model's value for V_y at $t = 0.178$ non-dimensional time units is pictured as the straight line at 0.1055 non-dimensional velocity units (dash-dot line).

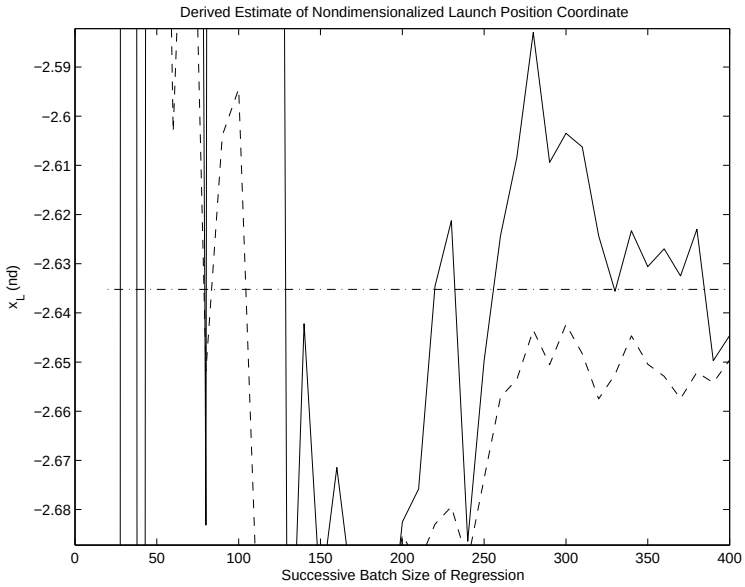


Figure H.10: Position estimate at launch $\hat{x}_L$ using augmented ILS with constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS estimate is also shown (dashed line). The truth model's value for x_L at launch is pictured as the straight line at -2.636 non-dimensional position units (dash-dot line).

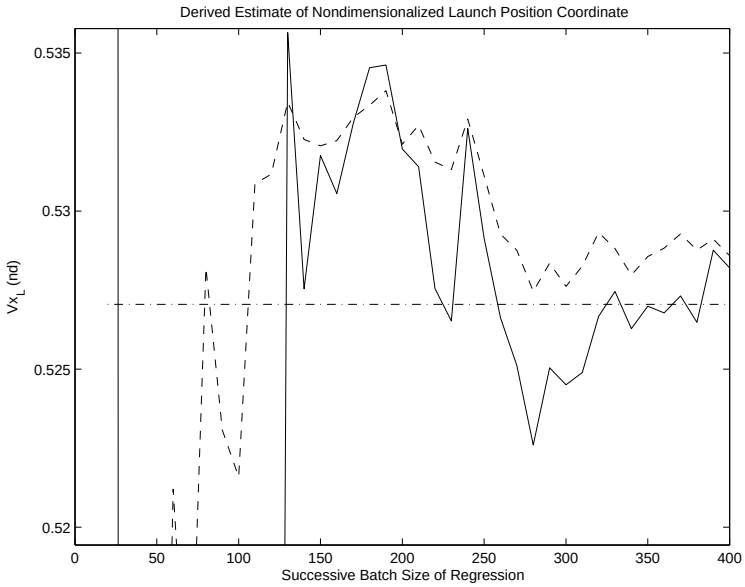


Figure H.11: Velocity estimate at launch $\hat{V}_{x_L}$ using augmented ILS with constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS estimate is also shown (dashed line). The truth model's value for V_{x_L} at launch is pictured as the straight line at 0.527 non-dimensional velocity units (dash-dot line).

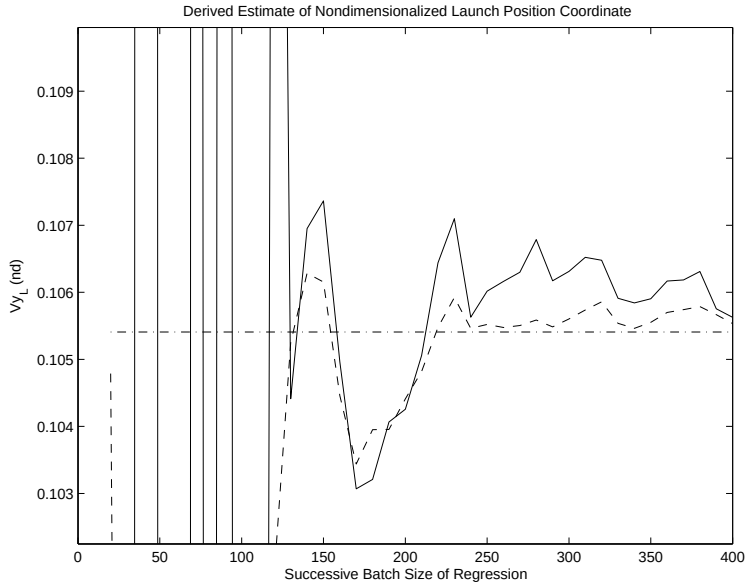


Figure H.12: Velocity estimate at launch $\hat{V}_{y_L}$ using augmented ILS with constrained intercepts for range, azimuth, and elevation (solid line). The standard ILS estimate is also shown (dashed line). The truth model's value for V_{y_L} at launch is pictured as the straight line at 0.1055 non-dimensional velocity units (dash-dot line).

Appendix I. Additional Results: Reconfigurable Flight Control: Corroborating Trigger

I.1 Additional Stability Derivatives: System ID for Failure with Sizeable Trim Change

In Section 5.4.2, a reason to continue the sliding batch window for a short period after the trigger (currently $\dot{q}$ [10]) detects a failure is considered. The author observes that there is a quick jump in the intercept estimate $\hat{c}$ (Figure 5.17) before significant movement in the other three parameter estimates. In other words, $\hat{c}$ is more sensitive than the other parameter estimates. The thesis already shows the control derivative in Figure 5.18. The corresponding stability derivative results are shown in Figures I.1 and I.2.

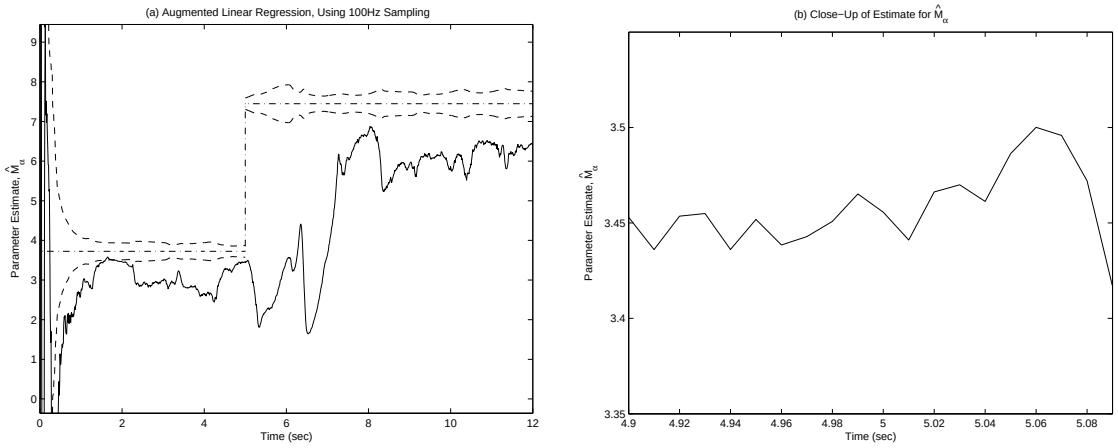


Figure I.1: Estimate of control derivative $\hat{M}_\alpha$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s. Sliding window is not reset after failure. Solid line–estimate, Dash-dot line–true parameter value about which the 1-σ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

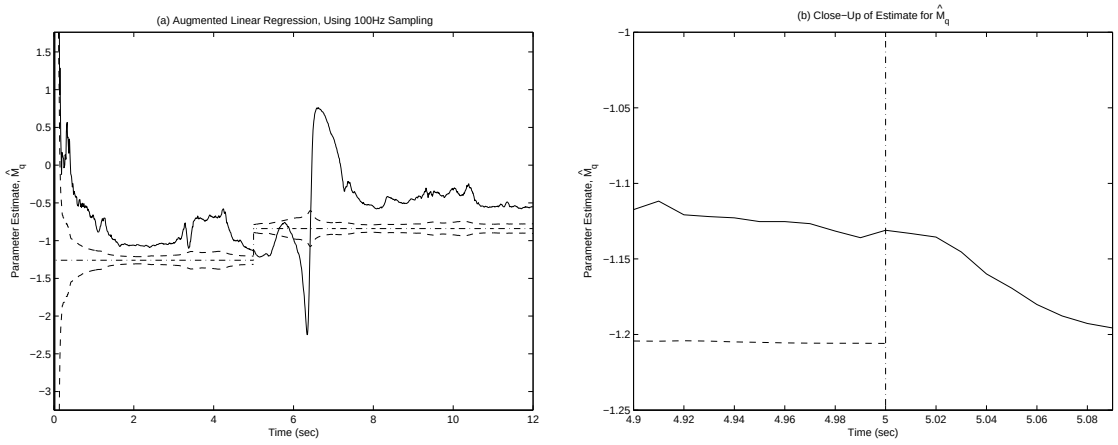


Figure I.2: Estimate of control derivative $\hat{M}_q$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s. Sliding window is not reset after failure. Solid line–estimate, Dash-dot line–true parameter value about which the 1-σ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

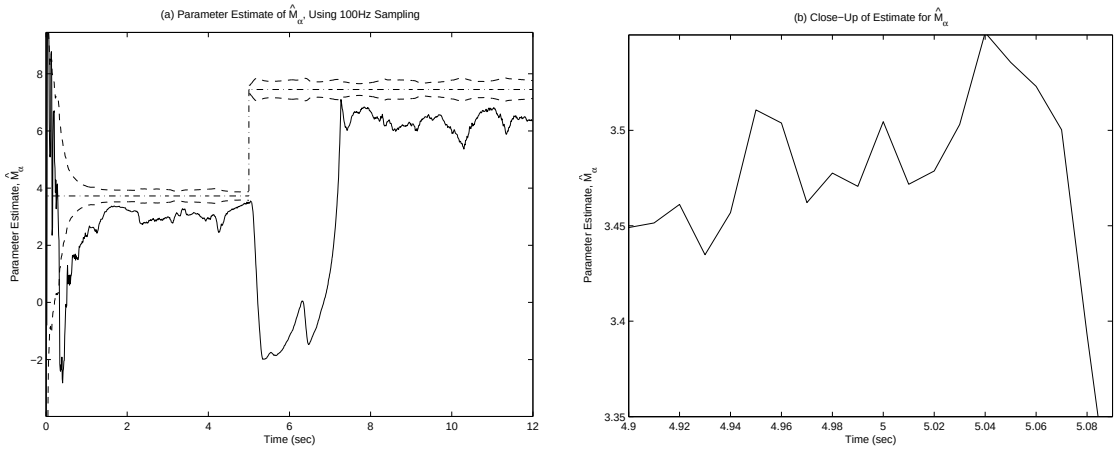


Figure I.3: Estimate of control derivative $\hat{M}_\alpha$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s, but there is not trim change. Sliding window is not reset after failure. Solid lines–estimates, Dash-dot lines–true parameter value about which the 1-σ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

I.2 Additional Stability Derivatives: System ID for Failure with No Trim Change

The same argument made about using $\hat{c}$ as a trigger for failure detection holds here, as discussed in Section 5.4.3. Figure 5.23 showed the bias estimate, as used for trigger identification purposes, while Figure 5.24 showed the control derivative. The additional stability estimates are now shown in Figures I.3 and I.4.

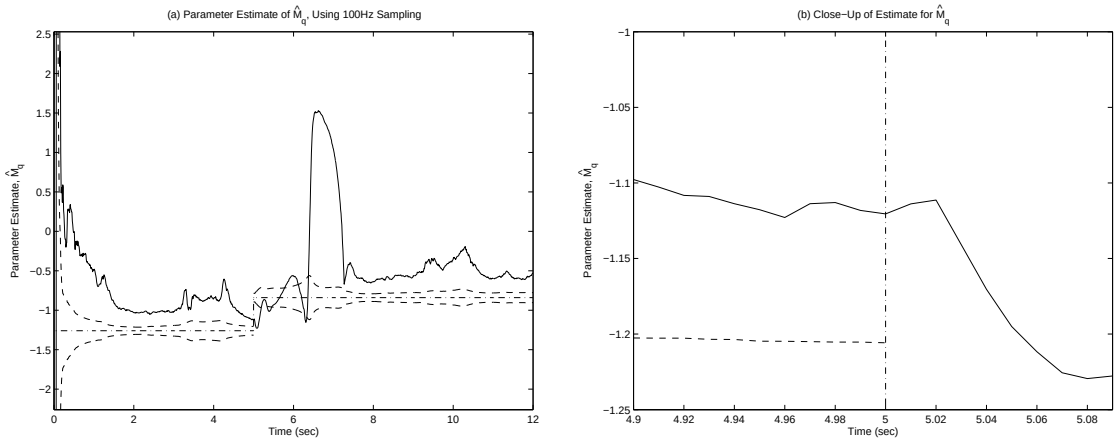


Figure I.4: Estimate of control derivative $\hat{M}_q$ using sliding window batch process. 50% of the horizontal stabilator surface is lost at $t = 5$s, but there is not trim change. Sliding window is not reset after failure. Solid lines–estimates, Dash-dot lines–true parameter value about which the 1-σ bounds from the filter-predicted equation error covariance (dashed lines) are depicted.

Bibliography

1. *"Self-Designing Controller"*. Technical Report WL-TR-97-3095, February 1998.
2. *"U.K. gets Battlefield Radar"*. Technical Report, Defense News, February 2004.
3. Adamy D. "Emitter Location – Conversion of AOA Errors to Location Errors," *Journal of Electronic Defense*, *26*:62–63 (March 2003).
4. Adamy D. "Emitter Location – Error Budget," *Journal of Electronic Defense*, *26*:56–57 (February 2003).
5. Brinker J. S. and Wise K. A. "Flight Testing of a Reconfigurable Flight Control Law on the X-36 Tailless Fighter Aircraft." *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Denver, Colorado*. 14-17 August 2000.
6. Bryson A. E. *Dynamic Optimization*. Addison Wesley Longman, Inc., 1999.
7. Buffington J., Chandler P., and Pachter M. "Integration of On-line System Identification and Optimization-based Control Allocation," (A98-37178, American Institute of Aeronautics and Astronautics (AIAA) Paper):1746–56
8. Caglayan A., Allen S., and Wehmuller K. "Evaluation of a Second Generation Reconfigurable Strategy for Aircraft Flight Control Systems Subjected to Actuator Failure / Surface Damage." *Proceedings of the National Aerospace and Electronics Conference*. 23-27 May 1988.
9. Chandler P., Pachter M., Mears M., and Sheldon S. "Regression Techniques for Aircraft Parameter Identification from Noisy Measurements in Maneuvering Flight." *Proceedings of the 31st Conference on Decision and Control, Tucson, Arizona*. December 1992.
10. Chandler P. R., Pachter M., and Mears M. "System Identification for Adaptive and Reconfigurable Control," *Journal of Guidance, Control, and Dynamics*, *18, No. 3*:516–524 (May-June 1995).
11. Chen K. and Hu I. "On Consistency of Bayes Estimates in a Certainty Equivalence Adaptive System," *IEEE Transactions on Automatic Control*, *43, No. 7*:943–947 (July 1998).
12. Eberhardt R. L. and Ward D. G. "Indirect Adaptive Flight Control of a Tailless Fighter Aircraft." *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Portland, Oregon*. 9-11 August 1999.
13. Kelly R. J. "Additional Results on 'Reducing Geometric Dilution of Precision Using Ridge Regression'," *IEEE Transactions on Aerospace and Electronic Systems*, *26, No. 4*:677–681 (July 1990).

14. Kelly R. J. "Reducing Geometric Dilution of Precision Using Ridge Regression," *IEEE Transactions on Aerospace and Electronic Systems*, *26, No. 1*:154–168 (January 1990).

15. Ljung L. *System Identification: Theory for the User*. Prentice Hall, 2nd ed., 1999.

16. M. Pachter P. C. and Mears M. "Reconfigurable Tracking Control with Saturation," *Journal of Guidance, Control, and Dynamics*, *18, No. 5*:1016–1022 (September-October 1995).

17. Marple S. L. *Digital Spectral Analysis with Applications*. Prentice-Hall, Inc., 1987.

18. Maybeck P. S. *Stochastic Models, Estimation, and Control: Volume 1*. Volume 141-1 in Mathematics in Science and Engineering: Republished by Navtech Book and Software Store, 1994.

19. Maybeck P. S. *Stochastic Models, Estimation, and Control: Volume 2*. Volume 141-2 in Mathematics in Science and Engineering: Republished by Navtech Book and Software Store, 1994.

20. Maybeck P. S. *Stochastic Models, Estimation, and Control: Volume 3*. Volume 141-3 in Mathematics in Science and Engineering: Republished by Navtech Book and Software Store, 1994.

21. Naylor A. W. and Sell G. R. *Linear Operator Theory in Engineering and Science*. Springer, 2000.

22. Nelson E. and Pachter M. "Linear Regression with Intercept – Applications," *AIAA paper number 2004-4757 GNC, Providence, RI* (2004).

23. Nelson E. and Pachter M. "Ballistic Trajectory Tracking Using Constrained Estimation," *Presented at the 13th Mediterranean Conference on Control and Automation (MED05), Cyprus* (June 2005).

24. Nelson E. and Pachter M. "Projectile Launch Point Estimation from Radar Measurements," *Presented at the American Controls Conference (ACC), Portland, OR* (June 2005).

25. Pachter M. "Stochastic Modeling Based DGPS Estimation Algorithm." *Proceedings of the 39th IEEE Conference on Decision and Control, Sydney, Austrailia*. IEEE, December 2000.

26. Pachter M. and Reynolds O. "Identification of a Discrete Time Dynamical System," *IEEE Transactions on Aerospace and Electronic Systems*, *36, No. 1*:212–221 (2000).

27. Pachter M. "Lecture Notes." Aircraft and Missile Controls and Estimation, AFIT, 1992.

28. Pachter M. and Nelson E. "Reconfigurable Flight Control," *To appear in the Journal of Aerospace Engineering* (TBD 2005).

29. Reynolds O. R. and Pachter M. "Phasor Approach to Continuous Time System Identification," *IEEE Transactions on Aerospace and Electronic Systems*, *35, No. 2*:683–699 (1999).

30. Stevens B. L. and Lewis F. L. *Aircraft Control and Simulation*. John Wiley Inc., 2nd ed., 2003.

31. Strelow D. and Singh S. "Online Motion Estimation From Image and Inertial Measurements." *Proceedings of the Workshop on Integration of Vision and Inertial Sensors (INERVIS 2003)*. June 2003.

32. Tikhonov A. N. *Solutions of Ill-posed Problems*. English Translation, Translation editor, John Fritz, Halsted Press, New York, 1977.

33. Urnes J. and et al. . "Flight Demonstration of a Self Repairing Flight Control System on a NASA F-15 Fighter Aircraft." *Proceedings of the AGARD Guidance and Control Panel, 49th Symposium, Toulouse, France*. October 1989.

34. Ward D. and Barron R. "A Self-Designing Receding Horizon Optimal Flight Controller." *Proceedings of the American Control Conference in Seattle, Washington*. June 1995.

35. Wiesel W. E. "Class Notes/Lecture." MECH731 Modern Methods of Orbit Determination, AFIT, Summer 2003.

36. Wiesel W. E. *Modern Methods of Orbit Determination, class lecture material*. AFIT, 8 June, 1998.

37. Willsky A. "A Survey of Design Methods for Failure Detection in Dynamic Systems," *Automatica*, *12*:601–611 (1976).

Index

The index is conceptual and does not designate every occurrence of a keyword. Page numbers in bold represent concept definition or introduction.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
September 2005	Doctoral Dissertation	Sept 2003 — Sept 2005

4. TITLE AND SUBTITLE

Nonlinear Regression Methods for Estimation

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Eric B. Nelson, Capt, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/DS/ENG/05-05

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Stanton Musick, Jeffery Layne
Sensors Directorate, Air Force Research Laboratory
Building 620
Wright-Patterson Air Force Base, OH 45433

10. SPONSOR/MONITOR'S ACRONYM(S)

AFRL/SNAT

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approval for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT Regression techniques are developed for batch estimation and applied to three specific areas, namely, ballistic trajectory launch point estimation, adaptive flight control, and radio-frequency target triangulation. Specifically, linear regression with an intercept is considered in detail. An augmentation formulation is developed. Extensions of theory are applied to nonlinear regression as well. The intercept parameter estimate within the linear regression is used to identify the effects of trim change that are associated with the occurrence of a control surface failure. These estimates are used to adjust the inner loop control gains via a feed-forward command, hence providing an automatic reconfigurable retrim of an aircraft. The regression algorithms are used to consider reduced information applications, such as initial position target determination from bearings-only measurement data. In total, this dissertation develops algorithms for batch processes that broaden the envelope of successful estimation within the three aforementioned application areas. Additionally, the developed batch algorithms do not adversely impact the estimation ability in cases that are already estimated successfully by conventional approaches.

15. SUBJECT TERMS

Nonlinear Regression, Linear Regression, Ridge Regression, Intercept, Batch Estimation, ballistic trajectory launch point estimation, radio-frequency target triangulation, adaptive flight control, reconfigurable flight control, system identification

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Meir Pachter (ENG)
U	U	U	UU	252	19b. TELEPHONE NUMBER *(include area code)* (937)255–6565, X4280; meir.pachter@afit.edu