AFIT/GE/ENG/93D-11

AD-A274 181

DTIC
ELECTE
DEC 23 1993
S A

ENHANCED TRACKING OF AIRBORNE TARGETS USING FORWARD LOOKING INFRARED AND LASER RETURN MEASUREMENTS

THESIS

Patrick J. Grondin, Captain, USAF

AFIT/GE/ENG/93D-11

Approved for public release; distribution unlimited

93-31027

93 12 22 140

AFIT/GE/ENG/93D-11

ENHANCED TRACKING OF AIRBORNE TARGETS USING FORWARD LOOKING INFRARED AND LASER RETURN MEASUREMENTS

THESIS

Presented to the Faculty of the Graduate School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Electrical Engineering

DTIC QUALITY INSPECTED 3

Accesion For
NTIS
DTIC
Unannounced
Justification
By
Distribution
Availability
Dist
A-1

Patrick J. Grondin

Captain, USAF

December 1993

Approved for public release; distribution unlimited

Acknowledgements

This thesis continues the development of Multiple Model Adaptive Estimation (MMAE) in conjunction with the ballistic missile tracking problem. This latest MMAE effort incorporated the plume pogo position and velocity and the offset equilibrium point into the elemental filter structure. This problem scenario offered many diverse and interesting challenges.

I would like to thank my thesis advisor, Dr. Peter Maybeck, for the great help and inspiration he provided throughout the project. Working with Dr. Maybeck was truly the best part of "academic" AFIT.

I also applaud all of the AFIT students who have previously worked on this missile tracking problem. I feel honored to continue their lineage.

My friends and colleagues deserve a great deal of praise and thanks; especially, Tom Kobalarz, Dave Lane, Neil Hansen, Vince Renya, Chip Mosle, Odel Reynolds, Mark Keating, Mike Logan, and Rob Reid.

Lastly, I owe a debt to AFIT for the opportunity to meet my lovely wife, Janet. She helped me in several areas; from her wise words of wisdom to her words of comfort. She was always there!

Patrick J. Grondin

Table of Contents

Acknowledgements ii

List of Figures iv

List of Tables xxiv

Abstract xxvi

I. Introduction 1-1

1.1 Background 1-6

1.2 Summary of Previous Research 1-8

1.3 Thesis Objectives 1-33

1.4 Thesis Overview 1-35

II. Kalman Filters, MMAE and Simulation Coordinate Frames 2-1

2.1 Introduction 2-1

2.2 Kalman Filter Theory 2-1

2.2.1 Introduction 2-1

2.2.2 Linear Kalman Filter 2-3

2.3 Multiple Model Adaptive Estimation 2-9

2.4 Simulation Space 2-15

2.4.1 Coordinate Frames 2-15

2.4.1.1 Inertial Reference Frame 2-16

2.4.1.2 Target Plume Reference Frame 2-17

2.4.1.3 α-β-r Reference Frame 2-17

2.4.1.4 α-β (FLIR Image) Plane 2-18

2.4.1.5 Absolute α-β-r Reference Frame 2-19

2.4.1.6 Trans-FLIR Plane 2-19

2.4.1.7 ALT/ACT Plane 2-19

2.4.2 FLIR Image Plane 2-19

2.4.2.1 FLIR Field-of-View 2-21

2.4.2.2 Target Models on FLIR Plane 2-22

2.4.2.3 Target Plume Model/FLIR Plane 2-22

2.4.2.4 Target Plume Projection onto the FLIR Plane 2-25

2.4.2.5 Target Plume Velocity Projection onto the FLIR Plane 2-28

2.4.3 FLIR Sensor Pointing Controller 2-31

2.5 Summary 2-32

III. Truth Model 3-1

3.1 Introduction 3-1

3.2 Dynamics Model 3-4

3.2.1 Target Dynamics State Description 3-9

3.2.2 Atmospheric Jitter Model 3-13

3.2.3 Bending/Vibration Model 3-17

3.2.4 Plume Pogo Model 3-21

3.3 Measurement Models 3-25

3.3.1 FLIR Model 3-26

3.3.2 Low-Energy Laser Speckle Reflection Model 3-30

3.3.2.1 The Hardbody Reflectivity Model 3-31

3.3.2.2 Plume Reflectance Model 3-40

3.3.3 The Doppler Measurement Model 3-43

3.3.3.1 The Doopler Effect 3-44

3.3.3.2 Hardbody Doppler Return 3-48

3.3.3.3 Plume Doppler Return 3-50

3.3.3.4 Doppler Measurement Noises 3-53

3.4 Truth Model Parameters 3-58

3.4.1 Target Trajectory Initial Conditions 3-58

3.4.2 Target Model, Dimensions, and Orientation 3-59

3.4.3 Intensity Functions 3-60

3.4.4 Atmospheric Jitter 3-60

3.4.5 Bending/Vibration 3-60

3.4.6 Plume Pogo Characteristics 3-60

3.4.7 Spatially Correlated Background Noise 3-61

3.4.8 Low-Energy Laser Speckle Return Measurement Dimensions 3-61

3.4.9 Plume Reflectance Model 3-61

3.4.10 Low-Energy Doppler Return Measurement Dimensions 3-61

3.4.11 Hardbody Reflectivity Measurement Model 3-62

3.5 Summary 3-63

IV. Filter Models 4-1

4.1 Introduction 4-1

4.2 Dynamics Models 4-1

4.2.1 Target Dynamics Model 4-5

4.2.2 Atmospheric Disturbance Model 4-6

4.2.3 Pogo Dynamics Model 4-8

4.2.4 Centroid Equilibrium Point/Center-of- Mass Offset Model 4-9

4.3 Measurement Models 4-11

4.3.1 FLIR Measurement Model 4-11

4.3.1.1 Enhanced Correlator Algorithm 4-12

4.3.1.2 Template Generation 4-15

4.3.1.3 "Pseudo-Measurements" 4-18

4.3.2 Doppler Measurement Model 4-23

4.3.3 Filter Parameters 4-26

4.3.3.1 Initial Conditions 4-27

4.3.3.2 Tuning Values 4-28

4.4 Summary 4-30

V. Elemental Filter and MMAE Results/Analysis 5-1

5.1 Introduction 5-1

5.2 Elemental Kalman Filter Simulation Results/Analysis 5-2

5.2.1 Introduction 5-2

5.2.2 Single-Filter Data Runs 5-4

5.2.2.1 Truth Parameter Values: Low Amplitude, Low Frequency 5-6

5.2.2.2 Truth Parameter Values: Low Amplitude, High Frequency 5-14

5.2.2.3 Truth Parameter Values: Large Amplitude, Low Frequency 5-17

5.2.2.4 Truth Parameter Values: Large Amplitude, High Frequency 5-21

5.2.3 Selection of Candidate MMAE Configurations ... 5-23

5.3 MMAE Simulation Results and Analysis 5-25

5.3.1 Introduction 5-25

5.3.2 MMAE Configuration #1 5-27

5.3.2.1 Configuration #1 Run with Truth Parameter Values = $0.112/20\pi$ 5-29

5.3.2.2 Configuration #1 Run with Truth Parameter Values = $1.12/2\pi$ 5-31

5.3.2.3 Configuration #1 Run with Truth Parameter Values = $1.12/20\pi$ 5-33

5.3.2.4 Discretization of Parameter Space For Configuration #1 5-34

5.3.3 MMAE Configuration #2 5-36

5.3.4 MMAE Configuration #3 5-39

5.3.5 MMAE Configuration #4 5-42

5.3.6 MMAE Configuration #5 5-44

5.4 Summary 5-46

VI. Conclusions and Recommendations 6-1

6.1 Introduction 6-1

6.2 Conclusions 6-1

6.2.1 Single-Filter Data Runs 6-1

6.2.2 MMAE Data Runs 6-3

6.2.3 Final Performance Issues 6-5

6.3 Recommendations 6-7

6.3.1 Further Research with Current MMAE Scheme ... 6-7

6.3.2 Implementable Algorithms 6-9

Appendix A. Data Processing Statistics Method A-1

Appendix B. Examples of Performance Plots B-1

Appendix C. Filter Plots with Truth Parameter Values: $0.112/2\pi$ C-1

Appendix D. Filter Plots with Truth Parameter Values: $0.112/20\pi$ ... D-1

Appendix E. Filter Plots with Truth Parameter Values: $1.12/2\pi$ E-1

Appendix F. Filter Plots with Truth Parameter Values: $1.12/20\pi$ F-1

Appendix G. MMAE Configuration #1 Output Plots G-1

Appendix H. MMAE Configuration #2 Output Plots H-1

Appendix I. MMAE Configuration #3 Output Plots I-1

Appendix J. MMAE Configuration #4 Output Plots J-1

Appendix K. MMAE Configuration #5 Output Plots K-1

Bibliography BIB-1

Vita

List of Figures

Figure 1.1 Missile Tracking Scenario 1-3

Figure 1.2 Linear Kalman Filter/Enhanced Correlator Algorithm 1-7

Figure 1.3 Multiple Model Adaptive Filter 1-12

Figure 1.4 Data Processing Scheme Using FFT and IFFT 1-14

Figure 1.5 Discrete Implementation of Cross-Sectional Reflectivity Function 1-28

Figure 2.1 Three Primary Coordinate Frames in Simulation Space .. 2-16

Figure 2.2 FLIR Plane, Trans-FLIR Plane, and ALT-ACT Plane 2-20

Figure 2.3 Target Plume Image in 8 x 8 FLIR Field-of-View 2-21

Figure 2.4 Spatial Relationship of Target Plume Gaussian Intensity Functions 2-24

Figure 2.5 Target Plume Intensity Centroid Projection Geometry ... 2-26

Figure 2.6 Intensity Centroid Dispersion Axes in FLIR Plane 2-27

Figure 2.7 Inertial Velocity FLIR Plane Projection Geometry 2-30

Figure 3.1 Plume Intensity Function Position on FLIR Image Plane .. 3-3

Figure 3.2 Target Centroid Image on α-β Plane with "Pseudo" Angles 3-10

Figure 3.3 Composite Plume Intensity Function on FLIR Plane 3-26

Figure 3.4 3-d Hardbody Reflectivity Model Relative to FLIR Image Plane 3-32

Figure 3.5 Discrete Implementation of Cross-Sectional Reflectivity Function 3-33

Figure 3.6 Projection Geometry Onto FLIR Image Plane 3-36

Figure 3.7 Ideal Low-Energy Laser Scan 3-38

Figure 3.8 Sweep Techniques of Laser Scan 3-39

Figure 3.9 Biased Offset Measurement Caused by Plume Reflectance 3-42

Figure 3.10 Spectra of Received Signals [43] 3-45

Figure 3.11 Pulsed Signal Frequency Spectrum 3-47

Figure 3.12 Spectra of Coherent and Non-coherent Pulsed Signals [44] 3-49

Figure 3.13 Spectra of Plume and Hardbody-Induced Doppler Returns 3-51

Figure 3.14 Detection Characteristics 3-57

Figure 4.1 Linear Kalman Filter/Enhanced Correlator Algorithm . . . 4-12

Figure 4.2 Filter Estimate of Offset Distance (Ideal Conditions) 4-24

Figure 5.1 Single Kalman Filter Block Diagram Implementation 5-2

Figure 5.2 Graphical View of Parameter Space 5-5

Figure 5.3 Configuration #1 Parameter Space 5-28

Figure 5.4 Configuration #2 Parameter Space 5-37

Figure 5.5 Configuration #3 Parameter Space 5-39

Figure 5.6 Discretization of Parameter Space Configuration #3 5-41

Figure 5.7 Configuration #4 Parameter Space 5-43

Figure 5.8 Configuration #5 Parameter Space 5-45

Figure C.1 X/Y Position (Filter States 1 and 2) Error Statistics (Truth = 0.112/20π, Filter = 0.112/20π) C-2

Figure C.2 X/Y Velocity (Filter States 3 and 4) Error Statistics (Truth = 0.112/20π, Filter = 0.112/20π) C-3

Figure C.3 X Atmospheric Jitter (Filter State 5)
(Truth = 0.112/20π, Filter = 0.112/20π) C-4

Figure C.4 Y Atmospheric Jitter (Filter State 6)
(Truth = 0.112/20π, Filter = 0.112/20π) C-5

Figure C.5 Pogo Position Offset (Filter State 7)
(Truth = 0.112/20π, Filter = 0.112/20π) C-6

Figure C.6 Pogo Velocity (Filter State 8)
(Truth = 0.112/20π, Filter = 0.112/20π) C-7

Figure C.7 Center-of-Mass to Equilibrium Offset (Filter State 9)
(Truth = 0.112/20π, Filter = 0.112/2π0) C-8

Figure C.8 Plume Centroid Error Statistics (Truth = 0.112/20π, Filter =
0.112/20π) C-9

Figure C.9 Filter Residual Quantities (Truth = 0.112/20π,
Filter = 0.112/20π) C-10

Figure C.10 Pogo Position (Filter State 7)
(Truth = 0.112/20π, Filter = 0.112/2π) C-11

Figure C.11 Pogo Velocity (Filter State 8)
(Truth = 0.112/20π, Filter = 0.112/2π) C-12

Figure C.12 Filter Residual Quantities (Truth = 0.112/20π,
Filter = 0.112/2π) C-13

Figure C.13 Pogo Position (Filter State 7)
(Truth = 0.112/20π, Filter = 1.12/20π) C-14

Figure C.14 Pogo Velocity (Filter State 8)
(Truth = 0.112/20π, Filter = 1.12/20π) C-15

Figure C.15 Filter Residual Quantities (Truth = 0.112/20π,
Filter = 1.12/20π) C-16

Figure C.16 Pogo Position (Filter State 7) (Truth = 0.112/20π,
Filter = 1.12/2π) C-17

Figure C.17 Pogo Velocity (Filter State 8) (Truth = 0.112/20π, Filter = 1.12/2π) C-18

Figure C.18 Filter Residual Quantities (Truth = 0.112/20π, Filter = 1.12/2π) C-19

Figure D.1 X/Y Position (Filter States 1 and 2) Error Statistics (Truth = 0.112/20π, Filter = 0.112/20π) D-2

Figure D.2 X/Y Velocity (Filter States 3 and 4) Error Statistics (Truth = 0.112/20π, Filter = 0.112/20π) D-3

Figure D.3 X Atmospheric Jitter (Filter State 5) (Truth = 0.112/20π, Filter = 0.112/20π) D-4

Figure D.4 Y Atmospheric Jitter (Filter State 6) (Truth = 0.112/20π, Filter = 0.11 /20π) D-5

Figure D.5 Pogo Position Offset (Filter State 7) (Truth = 0.112/20π, Filter = 0.112/20π) D-6

Figure D.6 Pogo Velocity (Filter State 8) (Truth = 0.112/20π, Filter = 0.112/20π) D-7

Figure D.7 Center-of-Mass to Equilibrium Offset (Filter State 9) (Truth = 0.112/20π, Filter = 0.112/20π) D-8

Figure D.8 Plume Centroid Error Statistics (Truth = 0.112/20π, Filter = 0.112/20π) D-9

Figure D.9 Filter Residual Quantities (Truth = 0.112/20π, Filter = 0.112/20π) D-10

Figure D.10 Pogo Position (Filter State 7) (Truth = 0.112/20π, Filter = 0.112/2π) D-11

Figure D.11 Pogo Velocity (Filter State 8) (Truth = 0.112/20π, Filter = 0.112/2π) D-12

Figure D.12 Filter Residual Quantities (Truth = 0.112/20π, Filter = 0.112/2π) D-13

Figure D.13 Pogo Position (Filter State 7)
(Truth = 0.112/20π, Filter = 1.12/20π) D-14

Figure D.14 Pogo Velocity (Filter State 8)
(Truth = 0.112/20π, Filter = 1.12/20π) D-15

Figure D.15 Filter Residual Quantities
(Truth = 0.112/20π, filter = 1.12/20π) D-16

Figure D.16 Pogo Position (Filter State 7)
(Truth = 0.112/20π, Filter = 1.12/2π) D-17

Figure D.17 Pogo Velocity (Filter State 8)
(Truth = 0.112/20π, Filter = 1.12/2π) D-18

Figure D.18 Filter Residual Quantities
(Truth = 0.112/20π, Filter = 1.12/2π) D-19

Figure E.1 X/Y Position (Filter States 1 and 2) Error Statistics
(Truth = 1.12/2π, Filter = 1.12/2π) E-2

Figure E.2 X/Y Velocity (Filter States 3 and 4) Error Statistics
(Truth = 1.12/2π, Filter = 1.12/2π) E-3

Figure E.3 X Atmospheric Jitter (Filter State 5)
(Truth = 1.12/2π, Filter = 1.12/2π) E-4

Figure E.4 Y Atmospheric Jitter (Filter State 6)
(Truth = 1.12/2π, Filter = 1.12/2π) E-5

Figure E.5 Pogo Position Offset (Filter State 7)
(Truth = 1.12/2π, Filter = 1.12/2π) E-6

Figure E.6 Pogo Velocity (Filter State 8)
(Truth = 1.12/2π, Filter = 1.12/2π) E-7

Figure E.7 Center-of-Mass to Equilibrium Offset (Filter State 9)
(Truth = 1.12/2π, Filter = 1.12/2π) E-8

Figure E.8 Plume Centroid Error Statistics
(Truth = 1.12/2π, Filter = 1.12/2π) E-9

Figure E.9 Filter Residual Quantities
(Truth = 1.12/2π, Filter = 1.12/2π) E-10

Figure E.10 Pogo Position (Filter State 7)
(Truth = 1.12/2π, Filter = 1.12/20π) E-11

Figure E.11 Pogo Velocity (Filter State 8)
(Truth = 1.12/2π, Filter = 1.12/20π) E-12

Figure E.12 Filter Residual Quantities
(Truth = 1.12/2π, Filter = 1.12/20π) E-13

Figure E.13 Pogo Position (Filter State 7)
(Truth = 1.12/2π, Filter = 0.112/2π) E-14

Figure E.14 Pogo Velocity (Filter State 8)
(Truth = 1.12/2π, Filter = 0.112/2π) E-15

Figure E.15 Filter Residual Quantities
(Truth = 1.12/2π, Filter = 0.112/2π) E-16

Figure E.16 Pogo Position (Filter State 7)
(Truth = 1.12/2π, Filter = 0.112/20π) E-17

Figure E.17 Pogo Velocity (Filter State 8)
(Truth = 1.12/2π, Filter = 0.112/20π) E-18

Figure E.18 Filter Residual Quantities
(Truth = 1. 12/2π, Filter = 0.112/20π) E-19

Figure F.1 X/Y Position (Filter States 1 and 2) Error Statistics
(Truth = 1.12/20π, Filter = 1.12/20π) F-2

Figure F.2 X/Y Velocity (Filter States 3 and 4) Error Statistics
(Truth = 1.12/20π, Filter = 1.12/20π) F-3

Figure F.3 X Atmospheric Jitter (Filter State 5)
(Truth = 1.12/20π, Filter = 1.12/20π) F-4

Figure F.4 Y Atmospheric Jitter (Filter State 6)
(Truth = 1.12/20π, Filter = 1.12/20π) F-5

Figure F.5 Pogo Position Offset (Filter State 7)
(Truth = 1.12/20π, Filter = 1.12/20π) F-6

Figure F.6 Pogo Velocity (Filter State 8)
(Truth = 1.12/20π, Filter = 1.12/20π) F-7

Figure F.7 Center-of-Mass to Equilibrium Offset (Filter State 9)
(Truth = 1.12/20π, Filter = 1.12/20π) F-8

Figure F.8 Plume Centroid Error Statistics
(Truth = 1.12/20π, Filter = 1.12/20π) F-9

Figure F.9 Filter Residual Quantities
(Truth = 1.12/20π, Filter = 1.12/20π) F-10

Figure F.10 Pogo Position (Filter State 7)
(Truth = 1.12/20π, Filter = 1.12/2π) F-11

Figure F.11 Pogo Velocity (Filter State 8)
(Truth = 1.12/20π, Filter = 1.12/2π) F-12

Figure F.12 Filter Residual Quantities
(Truth = 1.12/20π, Filter = 1.12/2π) F-13

Figure F.13 Pogo Position (Filter State 7)
(Truth = 1.12/20π, Filter = 0.112/20π) F-14

Figure F.14 Pogo Velocity (Filter State 8)
(Truth = 1.12/20π, Filter = 0.112/20π) F-15

Figure F.15 Filter Residual Quantities
(Truth = 1.12/20π, Filter = 0.112/20π) F-16

Figure F.16 Pogo Position (Filter State 7)
(Truth = 1.12/20π, Filter = 0.112/2π) F-17

Figure F.17 Pogo Velocity (Filter State 8)
(Truth = 1.12/20π, Filter = 0.112/2π) F-18

Figure F.18 Filter Residual Quantities
(Truth = 1.12/20π, Filter = 0.112/2π) F-19

Figure G.1 X/Y Position (Filter States 1 and 2) Error Statistics (Truth = 0.112/20π) G-2

Figure G.2 X/Y Velocity (Filter States 3 and 4) Error Statistics (Truth = 0.112/20π) G-3

Figure G.3 X Atmospheric Jitter (Filter State 5) (Truth = 0.112/20π) G-4

Figure G.4 Y Atmospheric Jitter (Filter State 6) (Truth = 0.112/20π) G-5

Figure G.5 Pogo Position Offset (Filter State 7) (Truth = 0.112/20π) G-6

Figure G.6 Pogo Velocity (Filter State 8) (Truth = 1.12/20π) G-7

Figure G.7 Center-of-Mass to Equilibrium Offset (Filter State 9) (Truth = 0.112/20π) G-8

Figure G.8 Plume Centroid Error Statistics (Truth = 0.112/20π) G-9

Figure G.9 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 0.112/20π) G-10

Figure G.10 Filter Residual Quantities; Computed *with* leading coefficient and $A_k(t_i)$ (Truth = 0.112/20π) G-11

Figure G.11 Filter Residual Quantities; Computed *without* leading coefficient (Truth = 0.112/20π) G-12

Figure G.12 X/Y Position (Filter States 1 and 2) Error Statistics (Truth = 1.12/2π) G-13

Figure G.13 X/Y Velocity (Filter States 3 and 4) Error Statistics (Truth = 1.12/2π G-14

Figure G.14 X Atmospheric Jitter (Filter State 5) (Truth = 1.12/2π) G-15

Figure G.15 Y Atmospheric Jitter (Filter State 6) (Truth = 1.12/2π) G-16

Figure G.16 Pogo Position Offset (Filter State 7)
(Truth = 1.12/2π) G-17

Figure G.17 Pogo Velocity (Filter State 8)
(Truth = 1.12/2π) G-18

Figure G.18 Center-of-Mass to Equilibrium Offset (Filter State 9)
(Truth = 1.12/2π) G-19

Figure G.19 Plume Centroid Error Statistics (Truth = 1.12/2π) G-20

Figure G.20 Filter Residual Quantities; Computed *without* leading
coefficient or $A_k(t_i)$ (Truth = 1.12/2π) G-21

Figure G.21 Filter Residual Quantities; Computed *with* leading coefficient
and $A_k(t_i)$ (Truth = 1.12/2π) G-22

Figure G.22 Filter Residual Quantities; Computed *without*
leading coefficient (Truth = 1.12/2π) G-23

Figure G.23 X/Y Position (Filter States 1 and 2) Error Statistics
(Truth = 1.12/20π) G-24

Figure G.24 X/Y Velocity (Filter States 3 and 4) Error Statistics
(Truth = 1.12/20π) G-25

Figure G.25 X Atmospheric Jitter (Filter State 5)
(Truth = 1.12/20π) G-26

Figure G.26 Y Atmospheric Jitter (Filter State 6)
(Truth = 1.12/20π) G-27

Figure G.27 Pogo Position Offset (Filter State 7)
(Truth = 1.12/20π) G-28

Figure G.28 Pogo Velocity (Filter State 8)
(Truth = 1.12/20π) G-29

Figure G.29 Center-of-Mass to Equilibrium Offset (Filter State 9)
(Truth = 1.12/20π) G-30

Figure G.30 Plume Centroid Error Statistics (Truth = 1.12/20π) G-31

Figure G.31 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1.12/20π) G-32

Figure G.32 Filter Residual Quantities; Computed *with* leading coefficient and $A_k(t_i)$ (Truth = 1.12/20π) G-33

Figure G.33 Filter Residual Quantities; Computed *without* leading coefficient (Truth = 1.12/20π) G-34

Figure G.34 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1. 12/4π) G-35

Figure G.35 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1.12/8π) G-36

Figure G.36 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1.12/9.5π) G-37

Figure G.37 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1.12/11π) G-38

Figure G.38 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1. 12/14π) G-39

Figure G.39 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = .784/20π) G-40

Figure G.40 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 0.448/20π) G-41

Figure G.41 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 0.112/2π) G-42

Figure H.1 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1.12/2π) H-2

Figure H.2 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1.12/8π) H-3

Figure H.3 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1.12/10π) H-4

Figure H.4 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1.12/10.5π) H-5

Figure H.5 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1.12/11π) H-6

Figure H.6 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1.12/12π) H-7

Figure H.7 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1.12/14π) H-8

Figure H.8 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1.12/20π) H-9

Figure H.9 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 0.448/2π) H-10

Figure H.10 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 0.112/2π) H-11

Figure H.11 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 0.05/2π) H-12

Figure H.12 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 0.112/20π) H-13

Figure I.1 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/2π) I-2

Figure I.2 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1.12/8π) I-3

Figure I.3 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/9.5π) I-4

Figure I.4 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/11π) I-5

Figure I.5 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/12.5π) I-6

Figure I.6 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/14π) I-7

Figure I.7 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/20π) I-8

Figure I.8 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.784/2π) I-9

Figure I.9 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.784/9.5π) I-10

Figure I.10 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.784/10.5π) I-11

Figure I.11 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.784/11π) I-12

Figure I.12 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.784/11.5π) I-13

Figure I.13 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.784/12.5π) I-14

Figure I.14 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.784/20π) I-15

Figure I.15 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.448/2π) I-16

Figure I.16 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.448/9.5π) I-17

Figure I.17 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.448/10.5π) I-18

Figure I.18 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.448/11π) I-19

Figure I.19 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.448/12.5π) I-20

Figure I.20 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.448/13π) I-21

Figure I.21 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.448/20π) I-22

Figure I.22 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.112/2π) I-23

Figure I.23 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.112/11π) I-24

Figure I.24 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.112/20π) I-25

Figure I.25 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 2.24/2π) I-26

Figure I.26 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 2.24/9.5π) I-27

Figure I.27 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 2.24/11π) I-28

Figure I.28 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 2.24/12.5π) I-29

Figure I.29 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 2.24/20π) I-30

Figure I.30 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.280/11π) I-41

Figure I.31 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.28/20π) I-42

Figure J.1 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/2π) J-2

Figure J.2 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/8π) J-3

Figure J.3 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/11π) J-4

Figure J.4 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/14π) J-5

Figure J.5 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/20π) J-6

Figure J.6 Filte Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.112/2π) J-7

Figure J.7 Filter Residual Quantities, Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.112/20π) J-8

Figure J.8 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.784/20π) J-9

Figure J.9 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.448/205π) J-10

Figure J.10 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.448/2π) J-11

Figure K.1a Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/2π) K-2

Figure K.1b Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/2π) K-3

Figure K.2a Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 2.24/2π) K-4

Figure K.2b Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 2.24/2π) K-5

Figure K.3a Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/20π) K-6

Figure K.3b Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/20π) K-7

Figure K.4a Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 2.24/20π) K-8

Figure K.4b Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 2.24/20π) K-9

Figure K.5a Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.448/11π) K-10

Figure K.5b Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.448/11π) K-11

List of Tables

Table 1.1 KF and Correlation Tracker Statistics Comparison 1-10

Table 5.1 Single Kalman Filter Truth and Filter Parameter Values ... 5-3

Table 5.2 Time-Averaged Error Statistics for Elemental Filter (Truth = $0.112/2\pi$, Filter = $0.112/2\pi$) 5-7

Table 5.3 Time-Averaged Error Statistics for Elemental Filter (Truth = $0.112/2\pi$, Filter = $0.112/20\pi$) 5-12

Table 5.4 Summary of the Pogo Position and Pogo Velocity Errors for Truth Condition of $0.112/2\pi$ with Various Filter Parameter Values 5-13

Table 5.5 Time-Averaged Error Statistics for Elemental Filter (Truth = $0.112/20\pi$, Filter = $0.112/20\pi$) 5-15

Table 5.6 Summary of Pogo Position and Pogo Velocity Errors for Truth Condition of $0.112/20\pi$ with Various Filter Parameter Values 5-16

Table 5.7 Time-Averaged Error Statistics for Elemental Filter (Truth = $1.12/2\pi$, Filter = $1.12/2\pi$) 5-19

Table 5.8 Summary of the Pogo Position and Velocity Errors for Truth Conditions of $1.12/2\pi$ with Various Filter Parameter Values 5-20

Table 5.9 Time-Averaged Error Statistics for Elemental Filter (Truth = $1.12/20\pi$, Filter = $1.12/20\pi$) 5-21

Table 5.10 Summary of the Pogo Position and Velocity Errors for Truth Conditions of $1.12/20\pi$ with Various Filter Parameter Values5-22

Table 5.11 Test MMAE Configuration 5-26

Table 5.12 Time-Averaged Error Statistics for MMAE Configuration #1 (Truth = $0.112/20\pi$) 5-30

Table 5.13 Time-Averaged Error Statistics for MMAE Configuration #1 (Truth = $1.12/2\pi$) 5-32

Table 5.14 Time-Averaged Error Statistics for MMAE Configuration #1 (Truth = $1.12/20\pi$) 5-34

Table 5.15 Configuration #1 at Various Truth Conditions 5-35

Table 5.16 Configuration #2 at Various Truth Conditions 5-38

Table 5.17 Configuration #3 at Various Truth Conditions 5-40

Table 5.18 Configuration #4 at Various Truth Conditions 5-44

Table 5.19 Configuration #5 at Various Truth Conditions 5-45

Table 6.1 Center-of-Mass Error Comparison; Truth = $0.112/2\pi$ 6-6

Table 6.2 Center-of-Mass Error Comparison; Truth = $0.112/20\pi$ 6-6

Table 6.3 Center-of-Mass Error Comparison; Truth = $1.12/2\pi$ 6-6

Table 6.4 Center-of-Mass Error Comparison; Truth = $1.12/20\pi$ 6-7

Abstract

The Air Force Institute of Technology has been involved in developing Kalman filter based trackers of ballistic missiles for 15 years. The goal of this thesis is to develop a Multiple Model Adaptive Estimator (MMAE) that tracks the missile plume (using a forward looking infrared sensor) and the missile hardbody center-of-mass (additionally using low energy laser returns) for the purpose of directing a high power laser to incapacitate the missile.

The missile plume "pogos" about an offset equilibrium point (relative to the hardbody center-of-mass) with an amplitude and frequency of oscillation that are not precisely known *a priori*. The MMAE algorithm estimates these parameters to improve performance in tracking the hardbody center-of-mass.

To accomplish this MMAE structure, single Kalman filters were developed and tested at the different parameter values. A Kalman filter residual analysis was used on these working single filters to define the MMAE structure that provided the most effective adaptation and most accurate target tracking.

A three-filter MMAE structure gave the lowest hardbody center-of-mass tracking errors. The two-dimensional parameter space, pogo amplitude and frequency, was successfully partitioned according to the frequency of oscillation. When the plume pogo amplitude is large, the MMAE structure substantially reduces the tracking errors of the hardbody center-of-mass, compared to a tracker without adaptive pogo estimation.

Enhanced Tracking of Airborne Targets Using Forward Looking Infrared and Laser Return Measurements

I. Introduction

The United States of America has been under direct threat from nuclear tipped Intercontinental Ballistic Missiles (ICBM's) for over thirty years. This threat, even with the breakup of the Soviet Union, is alive and well, as is seen by the several countries around the world that have acquired or are actively engaged in acquiring this offensive nuclear capability. A defensive weapon that could be used to combat an ICBM in its boost phase of flight is a High Energy Laser (HEL). This high powered weapon has the unique ability to concentrate energy onto a small area of the target to render the guidance system of the vehicle useless or to destroy the target vehicle completely. A key to this HEL defensive weapon is the accuracy that needs to be attained, as there is a finite amount of time that the laser beam must be focused onto the target vehicle to accomplish the task.

This HEL is part of a system of components that includes a Forward Looking Infrared (FLIR) sensor, a Low Energy Laser (LEL), and a pointing/tracking guidance computer program. The FLIR sensor will provide position update information to the computer program. The LEL is used to pinpoint the missile hardbody. The returns

from this LEL device are used as inputs to the tracking algorithm along with the FLIR measurements for further updates to the computer program. The pointing/tracking computer program is the heart of this research.

This program will utilize the specific Kalman filter work that has been progressing for over thirty years in the area of prediction and filtering. A Kalman filter (KF) is used to estimate various conditions about a target vehicle. These conditions are summarized in terms of a "state vector," a collection of variables that describe certain dynamic characteristics of that vehicle. This vector ($\hat{x}$) is estimated at various times throughout this digital simulation. The most important times are at the instant just prior to a sampled-data measurement update from the FLIR sensor and the instant just after this update has occurred. The KF estimates the state vector, $\hat{x}$, at time t_i^+ from the updates provided by the FLIR and LEL sensors. This KF algorithm then uses an internal mathematical dynamics model made up of a collection of differential equations to make the best possible prediction of $\hat{x}$ at t_{i+1}^-, one sample period later [1].

The scenario that is simulated for this research is to have an ICBM being tracked in its boost phase of flight from a very large distance. The FLIR Line Of Sight (LOS) vector is pointing at the target vehicle with optics common to the FLIR, the LEL, and the HEL. This system must accurately track the target vehicle despite the variables that are introduced by nature (atmospheric jitter), vehicle dynamics

(physical laws of motion), bending/vibration of the optical platform, and the vehicle plume (plume pogo effect). Radiation from the target plume is projected onto the two-dimensional FLIR image plane with the position and velocity of the centroid of this plume estimated by the KF. The LEL is then swept along the filter-predicted velocity vector to identify the center of mass of the missile hardbody. This scenario is shown pictorially in Figure 1.1.

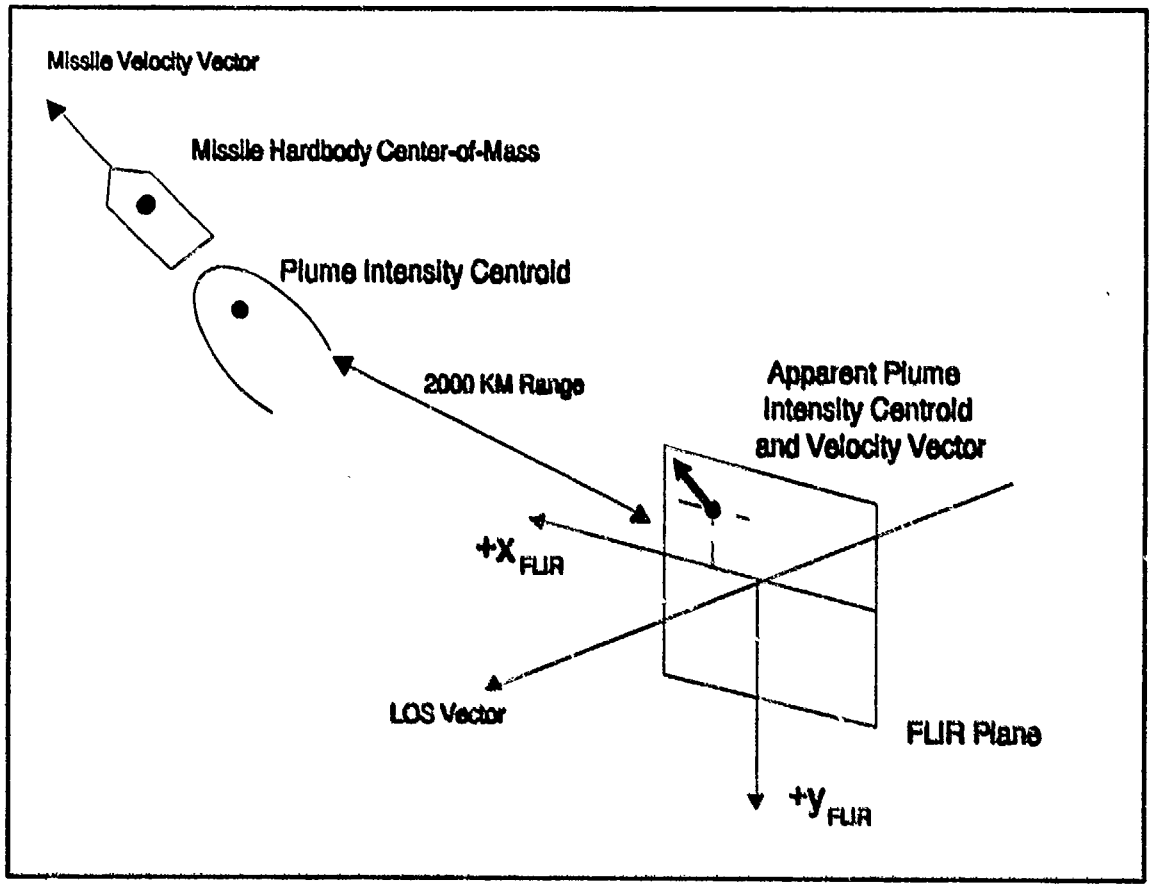


Figure 1.1 Missile Tracking Scenario

The physical anomalies of atmospheric jitter and plume pogo effect must be taken into account for accurate tracking. Atmospheric jitter is the distortion of light as it traverses the atmosphere. Light arrives at the FLIR plane in waves with a phase that is independent of the previous light wave. Distorted phase fronts of emanated waves result in translational shifts on the FLIR image plane, called jitter. An example of the problems caused by this phenomenon would be to try to aim a pistol at a far-off target that is sitting on a hot, black surface (a target on a road). The target would be enveloped in a "shimmer" of light. It would be very difficult to focus on that target because of this atmospheric jitter effect. Furthermore, whatever platform the optics is mounted upon will have an associated bending/vibration mode that is activated whenever the platform is slewed for tracking purposes. Finally, the plume pogo effect is a characteristic of a rocket plume such that the plume oscillates about an equilibrium point relative to the missile hardbody as the rocket travels through the air. This pogo effect has an associated frequency and amplitude of oscillation, neither of which are known perfectly *a priori*.

These physical anomalies are represented in the Kalman filter mathematical model as either "states" or "parameters." The differences between states and parameters is that parameters vary more slowly than states and therefore do not have to be represented in the model as full dynamic states. The addition of a state adds a higher dimension and the associated mathematical complications to the model.

Compensating for parameter variation will be accomplished in a different manner.

In this research, the amplitude and frequency of oscillation of the plume pogo effect are to be represented as parameters, while the pogo offset position and velocity, atmospheric jitter and the bending/vibration uncertainties will be full-blown states. There will be several KF's that will be used simultaneously, each producing an output state estimate. Each of these KF's will be programmed with a different value for the plume pogo effect parameters of amplitude and frequency of oscillation. If the real world missile happens to be traveling along with the particular value of frequency and amplitude as is programmed into KF #3, then KF #3 will produce the best state estimate. The output of the bank of these KF's will be blended in a Bayesian fashion that will take into account the outputs of all of the KFs no matter how bad each of the particular filters is performing, weighing them according to how good their assumed parameter value seems to be.

Besides producing an output state estimate, each of the filters produces a "residual;" the difference between the incoming measurements and that filter's best estimate of the incoming measurements. This residual quantity is a good measure of how close that particular KF is to the actual real world. The residual will be used to calculate the probability weighting factor that will be multiplied with the state estimate of that particular filter to come up with that particular filter's contribution to the state estimate output for the entire computer algorithm. This computer

algorithm scheme is called a "Multiple Model Adaptive Estimator" (MMAE) and is further described in Section 1.2.

1.1 BACKGROUND

The Phillips Laboratory (formerly the Air Force Weapons Laboratory), at Kirtland Air Force Base, New Mexico, has sponsored research for 15 years at the Air Force Institute of Technology (AFIT) in the use of directed energy weapons to render certain airborne targets (ICBM's) useless. The research started when AFIT demonstrated the ability of a Kalman Filter (KF) based algorithm to outperform the standard correlation tracker that the Air Force Weapons Laboratory had been using.

Central to the AFIT tracking scheme is the two-dimensional FLIR sensor. The FLIR is a 300-by-500 picture element (pixel) array of individual radiation collecting surfaces. Each pixel, 15 microradians per two-dimensional side, passively collects radiation from the plume and the laser returns from the LEL. Within the entire 300-by-500 pixel FLIR image plane is an 8-by-8 tracking window. This smaller window provides a position update of the plume centroid.

The block diagram that is being implemented is shown in Figure 1.2. This algorithm has been used for 5 Masters theses in a row starting with Rizzo [34]. The

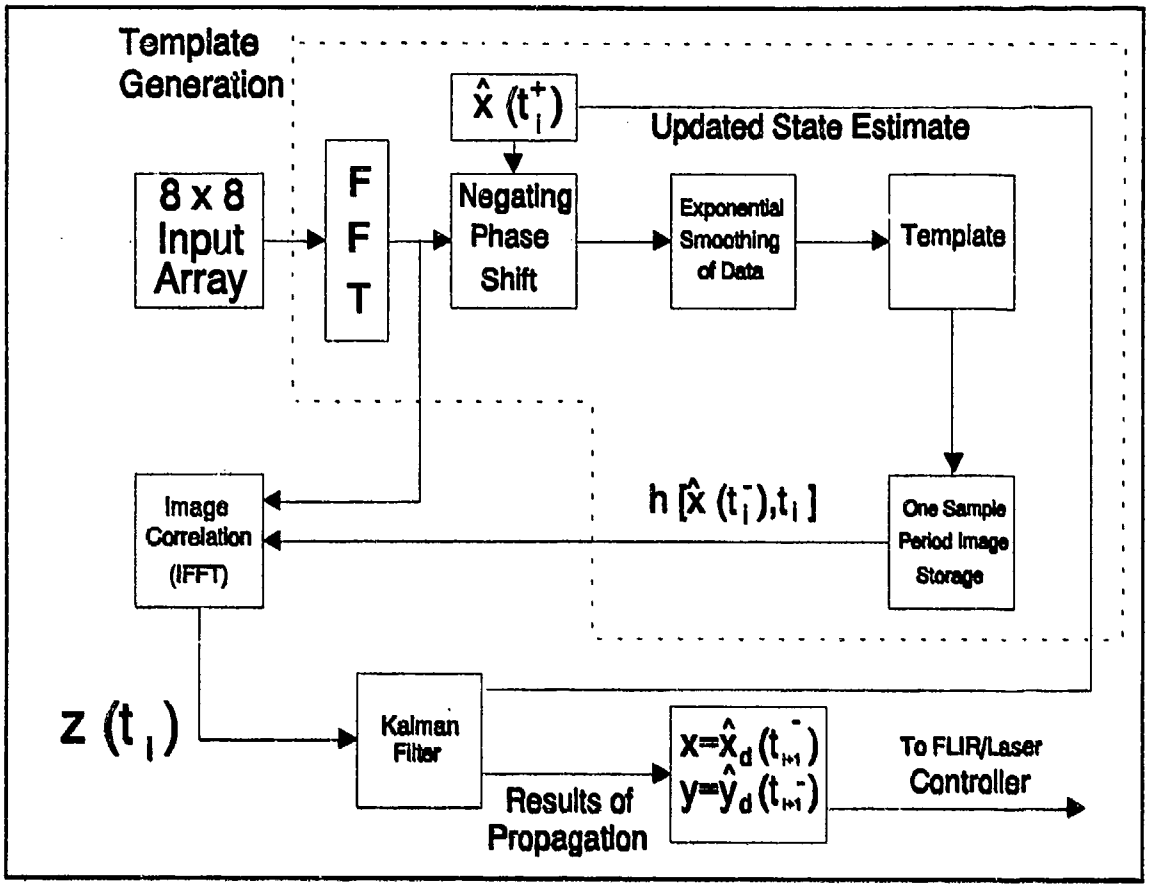


Figure 1.2 Linear Kalman Filter/Enhanced Correlator Algorithm

FLIR output is provided to the KF/Correlation Tracker algorithm. This algorithm compares the current FLIR data frame to an online-constructed template of the target/plume intensity pattern. The correlator then determines the optimal offsets in two dimensions that yield maximum correlation with the data generated by the template. These offsets are pseudo-measurements for the Linear KF which estimates the position and velocity of the plume centroid and performs a propagation cycle to form the best estimate of the states and measurements at the next sample period.

The FLIR is also updated to have the FLIR Line-Of-Sight (LOS) vector point at the best estimate of the target vehicle location at the time of the next sample period.

1.2 SUMMARY OF PREVIOUS RESEARCH

Sixteen theses and a number of other documents report the ongoing developments of the ballistic missile tracking problem. Each thesis contains a synopsis of previous work. This section provides a general overview of the previous ballistic missile tracker research completed at AFIT.

Research in this area was initiated by Mercier [26] in 1978, who compared Extended Kalman Filter (EKF) performance to that of the AFWL correlation tracker under identical conditions. An eight-state truth model was developed for simulation purposes, consisting of two target position states and six atmospheric jitter states. The position states defined the target location in each of two FLIR plane coordinate directions (azimuth and elevation), by accurately portraying target trajectories in three-dimensional space and projecting onto the FLIR plane. These two position variables, generated by trajectory generation "external" to the truth model's state equations, were treated as states simply for convenience in simulation, as for ease of generating errors for plots. The atmospheric jitter was modeled by a third order shaping filter driven by white noise for each FLIR plane axis, as provided by The Analytic Sciences Corporation (TASC) [16]; three states defined the atmospheric

distortion in each of the two FLIR plane coordinate directions. The Kalman Filter dynamics model consisted of four states: two states representing target position, and two representing the atmospheric jitter (based on reduced order models, versus the six states of the truth model). In both the truth model and filter dynamics model, the target position states and atmospheric jitter states were defined in each of the two FLIR plane coordinate directions. In the filter, the target position and jitter states were each modeled as a first-order, zero-mean, Gauss-Markov process. The FLIR provided sampled data measurements to the filter at a 30 Hertz (Hz) rate. The FLIR measurement noises corrupting each pixel output due to background clutter effects and internal FLIR noises were modeled in the filter as both temporally and spatially uncorrelated. The target was considered as a point source of light (i.e. a long-range target) having benign dynamics. The corresponding Airy disc on the FLIR image plane was modeled as a bivariate Gaussian distribution with circular equal intensity contours. The conventional correlation tracker and the extended Kalman Filter were compared across three different signal-to-noise ratios (SNR), using a ten-run Monte Carlo analysis to obtain the tracker error statistics. The results of the comparison (means and standard deviations of tracking error) are shown in Table 1.1 for a Gaussian intensity function dispersion, σ_g, equal to one pixel. (For a Gaussian intensity function dispersion equal to one pixel, most of the useful information is contained in an area of about five pixels square).

Signal-to Noise Ratio	Correlation Tracker		Extended Kalman Filter	
	Mean Error	1 σ	Mean Error	1 σ
20	7.0	8.0	0.0	0.2
10	8.0	10.0	0.0	0.2
1	15.0	30.0	0.0	0.8

Table 1.1 KF and Correlation Tracker Statistics Comparison

While the correlation tracker showed dramatic performance degradation as the SNR was decreased, the Kalman filter showed only a minor change in its performance at the lowest SNR tested. The extended Kalman filter was shown to be superior to the correlation tracker by an order of magnitude in the root mean square (rms) tracking error, provided the models incorporated into the filter were a valid depiction of the tracking scenario. This success motivated a follow-on thesis to improve filter modeling and thereby to enhance the performance.

The research accomplished by Harnly and Jensen [8,20] investigated modeling improvements in the filter and tested more dynamic target simulations. A comparison was made between a new six-state filter and a new eight-state filter. The six-state filter dynamics target model included the four previous states as well as two target velocity states in the FLIR plane coordinates (azimuth and elevation); the

dynamics model of the eight-state filter included two acceleration states in the FLIR coordinates as well. The acceleration was modeled as Brownian motion (BM) ($\dot{a} = w$, where w is a zero-mean white Gaussian noise). The filter was also designed to perform residual monitoring, which allowed the filter to react adaptively, and maintain track, by quickly increasing the values in the filter-computed state covariance matrix $\boldsymbol{P}$, which in turn increased the filter gain $\boldsymbol{K}$, when a target maneuver was detected. A recommendation was also made to examine increasing the Field-Of-View (FOV) during target jinking maneuvers to avoid losing lock. The constant-intensity contours of the target were modeled as elliptical patterns as opposed to the earlier circular equal-intensity contours in order to simulate closer range targets, such as air-to-air missiles. The major axis of the target FLIR image was aligned with the estimated velocity vector. A number of different target trajectories were tested against the six-state and eight-state filters, and while the six-state filter performed well during moderate jinking maneuvers, the eight-state filter performed substantially better while tracking high-g target maneuvers.

Other approaches to modeling the dynamics of the target in the filter were considered by Flynn [6]. He compared a Brownian motion (BM) acceleration target dynamics model [8] and a constant turn-rate (CTR) dynamics model. The CTR model portrayed the target behavior by modeling the acceleration as that associated with CTR dynamics. Concatenating such constant turn-rate segments together provides an accurate portrayal of manned target evasive maneuver trajectories. Additionally,

a Bayesian multiple model adaptive filter (MMAF) was developed using the BM acceleration dynamics model. A MMAF (Figure 1.3) consists of a bank of K independent Kalman filters, each of which is tuned to a specified target dynamics characteristic or parameter (a_1, a_2, ... a_K in Figure 1.3). The time histories of the residuals ($r_k(t_i)$ for k = 1,2,...,K in Figure 1.3) of these K Kalman filters are processed to compute the conditional probabilities ($p_k(t_i)$ in Figure 1.3) that each discrete parameter value is "correct." The residuals of the Kalman filter, based upon the "correct" model, are expected to be consistently smaller (relative to the filter's

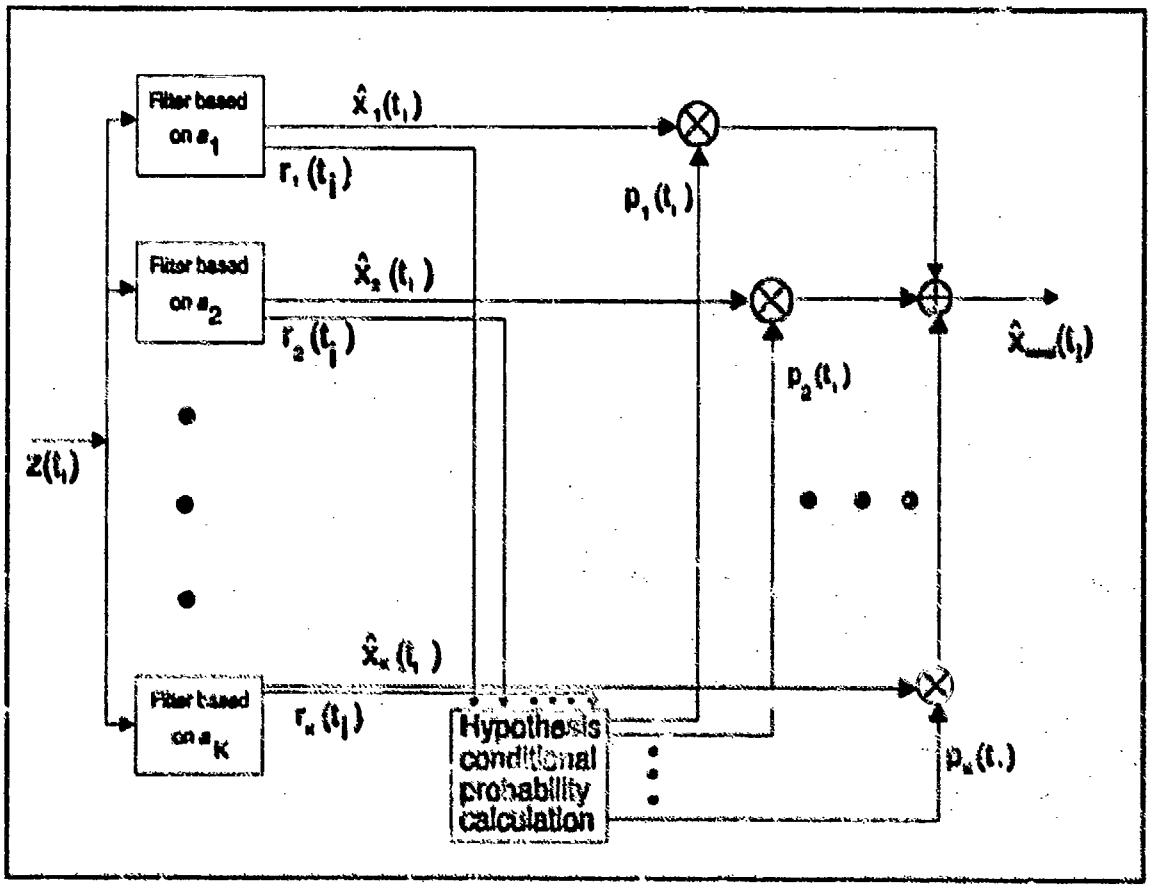


Figure 1.3 Multiple Model Adaptive Filter

internally computed residual rms values) than the residuals of the other mismatched filters (i.e. based upon "incorrect" models) [6]. If that is true, then the MMAF algorithm appropriately weights that particular Kalman filter more heavily than the other Kalman filters. These values are used as weighing coefficients to produce a probability-weighted average of the elemental filter outputs [6]. Therefore, the state estimate ($\hat{x}_{mmaf}(t_i)$ in Figure 1.3) is actually the probabilistically weighted average of the state estimates generated by each of the K separate Kalman filters ($\hat{x}_k(t_i)$ in Figure 1.3). In this research K was 3 and testing of the three filter models was conducted for three different flight trajectories which included 2-g, 10-g, and 20-g pull-up maneuvers. Unfortunately, the residuals of the 3 Kalman filters did not differ from each other enough to perform the weighing function properly, and the MMAF did not track well. The BM and CTR filters both performed equally well at 2-g's. The CTR filter was found to be substantially better than the BM filter for 10-g and 20-g pull-up maneuvers.

The research to that point had assumed that the filter had *a priori* knowledge of the target shape and intensity profile, and that is was a single-hot-spot target. Singletery [37] improved the realism in the target model by developing a model in the FLIR plane which included multiple hot spots. However, he returned to the case of very benign targets, as treated by Mercier. The filter did not assume *a priori* knowledge of the target size, shape, or number or location of hot spots. A new data processing scheme (Figure 1.4) was developed which included the use of the Fast

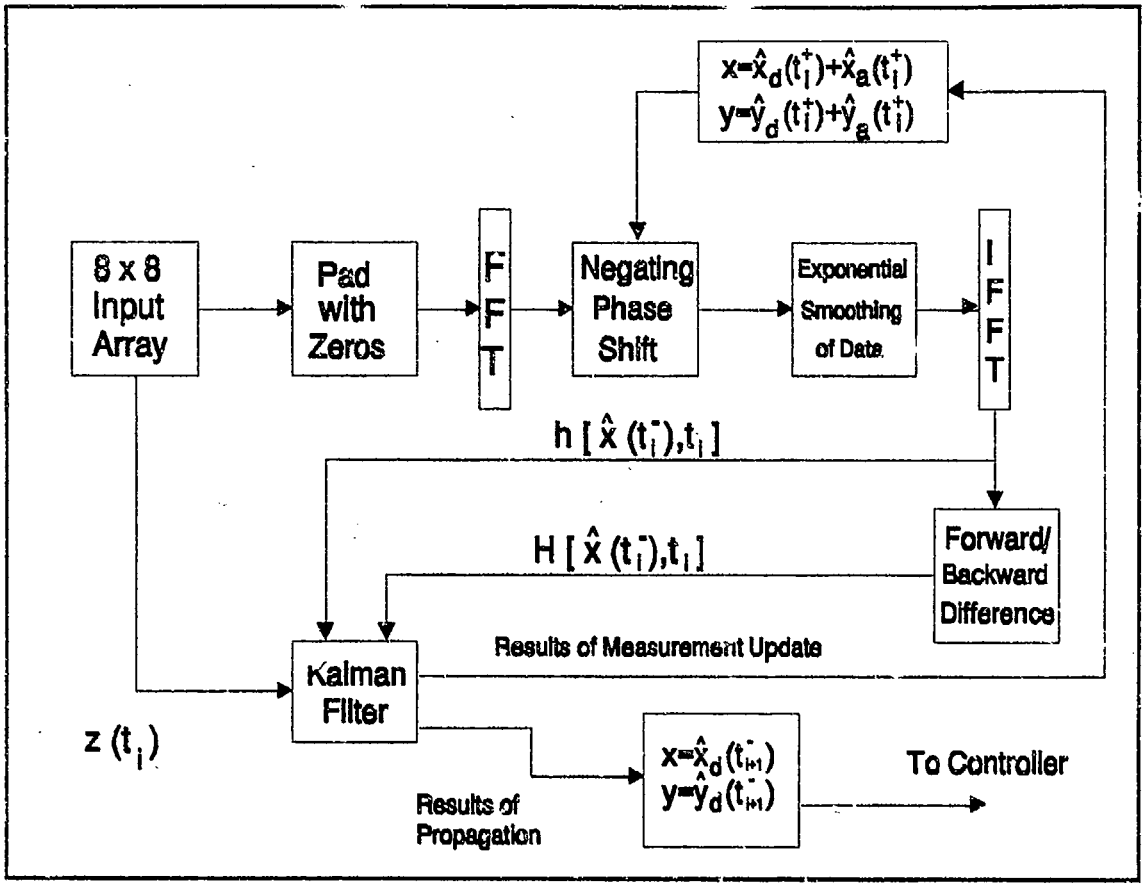


Figure 1.4 Data Processing Scheme Using FFT and IFFT

Fourier Transform (FFT) and the Inverse Fast Fourier Transform (IFFT), each of which can be produced with a lens if optical processing is used. The plan included two data paths for processing the intensity measurements $z(t_i)$. On the first path (the lower path in Figure 1.4), the 8 x 8 array of intensity measurements from the FLIR is arranged into a 64-dimensional measurement vector. This measurement vector is applied to the extended Kalman filter (as in prior work). The purpose of the second path is to provide an estimate of the target's intensity shape function. Centered target shape functions are time-averaged with previous centered shape functions in

order to generate the estimated target template (**h** in Figure 1.4) and partial derivatives of it with respect to the states (**H** in Figure 1.4), as needed by the extended Kalman filter. The first block in that path of Figure 1.4 has the 8 x 8 FLIR data padded with rows and columns of zeros to make up a 24 x 24 pixel array. The purpose of this padding is to ensure that the periodicity of the Discrete Fourier Transform will not degrade effects within the original data array [23]. This 24 x 24 pixel image is then centered using the shifting theorem of Fourier transforms. The shift theorem states that a translation of an image in the spatial domain results in a linear phase shift in the spatial frequency domain. To negate the translational effects of an uncentered target image in the spatial domain, the Fourier transform of the translated image is multiplied by the complex conjugate of the desired linear phase shift [37]. The extended Kalman filter model, in path one, which was developed by Mercier [26], was used to provide the optimal estimate of the required linear translation. The filter state estimates are used to develop the complex conjugate of the linear phase shift and provide the centered measurement functions. Before the IFFT is taken, the resulting pattern is exponentially smoothed to yield an approximation to averaging the result with N previously centered frames of data, to accentuate the actual target intensity image function and to attenuate the effect of measurement noise. The result is a centered pattern with noise effects substantially reduced. Following the application of the IFFT to form the nonlinear function of intensity measurements (**h** of Figure 1.4), the spatial derivative (approximated by a differencing operation) is used to determine the linearized function of intensity

measurements (H of Figure 1.4). These are both used by the Kalman filter in processing the next sampled measurement [37]. Two sets of position estimates are provided by the KF block; first, the platform controller is furnished the propagated dynamics position estimate, $\hat{x}_d(t_{i+1}^-)$ and $\hat{y}_d(t_{i+1}^-)$, as this is the best estimate of the plume intensity centroid at the time of the next measurement. Note that this estimate only has dynamics information concerning the plume centroid, no atmospheric jitter information is included, since it is desired to point at the *true* location of the target, not the jittered version of it. The other position estimate, $\hat{x}_d(t_i^+) + \hat{x}_a(t_i^+)$ and $\hat{y}_d(t_i^+) + \hat{y}_a(t_i^+)$, is used for centering the target image in the incoming raw FLIR data. Note that this estimate does include dynamics *and* atmospheric position information since the real world target image is translated by atmospheric jitter along with real world physical dynamics. The results of this data processing scheme were inconclusive due to filter divergence problems. Despite the problems encountered with the filter, the concept was considered to have performance potential.

Rogers [35] continued the work of developing a Kalman filter tracker which could handle multiple hotspots with no *a priori* information as to the size, shape, intensity, or location of the target hot spots. Moreover, he continued the application to benign target motion, as Singletery [37] had done before, in order to concentrate on the feasibility of adaptively identifying the target shape. Using digital signal processing on the FLIR data (as described above) to identify the target shape, the

filter uses the information to estimate target offset from the center of the FOV, which in turn drives a controller to center the image in the FLIR plane. Algorithm improvements included replacing the Forward-Backward Difference block of Figure 1.4 with a partial differentiation operation accomplished as a simple multiplication in the Fourier domain before the IFFT block.

Rogers also considered an alternative design that used the target image **h** as a template for an enhanced correlator, as was shown in Figure 1.2. The position offsets produced as outputs from the correlator were then used as "pseudo-measurement" inputs to a linear Kalman filter. The enhanced correlation algorithm of Figure 1.2 compares the FLIR image to an estimated template instead of the previous image, as is done in the standard correlator. This tracking concept is thus a hybrid of correlation tracking and Kalman filtering [35]. Its performance was compared to the results of earlier extended Kalman filters that used the raw FLIR data as measurements [8]. The extended Kalman filter performed well without *a priori* knowledge of the shape and location of the intensity centroid, the enhanced correlator used with the linear Kalman filter showed comparable performance with the extended Kalman filter while providing reduced computational loading.

Millner [28] and Kozemchak [11] tested an extended Kalman filter and a linear Kalman filter/enhanced correlation algorithm against close range, highly maneuverable targets. The linear four-state filter used in the previous research was

replaced by an eight-state filter consisting of position, velocity, acceleration, and atmospheric jitter states in each of the two coordinates of the FLIR plane (azimuth and elevation). Two target dynamics models were also investigated. The target was first modeled as a first-order Gauss-Markov acceleration process, and secondly with a constant turn-rate mode. Both filters performed well without *a priori* knowledge of the target size, shape, and hot spot location, using the FFT data processing method for identifying the target shape function [35,37]. However, at target maneuvers approaching 5 g's, the filter performance degraded considerably. It was noted that the tracking was substantially better when the Kalman filter dynamics model closely matched the target trajectory, but that a single nonadaptive filter was inadequate.

The Bayesian MMAF technique [6] was reinvestigated by Suizu [40] based on the recommendations of the previous work. The MMAF (Figure 1.3) consisted of two elemental Kalman filters. One elemental filter was tuned for benign target maneuvers and obtained sampled measurement information from an 8 x 8 pixel FOV in the FLIR plane. A second filter was tuned for highly dynamic maneuvers and obtained sampled measurement information from a 24 x 24 pixel FOV in the FLIR plane. The technique allowed the MMAF to maintain track on target trajectories up to 20 g's at a distance of 20 kilometers. The MMAF was configured for both the linear Kalman filter/enhanced correlation algorithm [35] and the extended Kalman filter. Both filtering schemes exhibited comparable rms tracking performance results, with the correlator/linear Kalman filter having smaller mean errors and larger

standard deviations than the extended Kalman filter, as seen in earlier work of Rogers [35]. The state rms tracking error was on the order of 0.2 to 0.4 pixels (one pixel being equivalent to 20 microradians on a side).

The potential of the MMAF technique with the FFT processing method was continued by Loving [14]. A third filter was added to the bank of filters, tuned for intermediate target maneuvers and obtaining sampled measurement information from the 8 x 8 FOV in the FLIR plane. This MMAF showed significant performance advantages over all the previous filters. Additionally, a Maximum A Posteriori (MAP) algorithm was developed and compared with the Bayesian MMAF. The MAP algorithm differs from the Bayesian MMAF of Figure 1.3 in that the MAP algorithm uses the residuals of the separate filters to select the one filter with the highest probabilistic validity, while the Bayesian MMAF uses a probability-weighted average of all filters in the bank. The Bayesian and the MAP techniques produced similar results and both delivered performance that surpassed previous filters.

Netzer [31] expanded the study of the MMAF algorithm. He investigated a steady-state bias error that was seen when tracking a target exhibiting a high-g constant-turn-rate maneuver. A major cause of this bias is the MMAF being tuned to anticipate equivalent accelerations in all FLIR plane directions. This causes mistuning in the x-direction (azimuth) while maintaining lock on the highly dynamic y-direction (elevation) transient for a trajectory starting horizontally and then pulling

up with a high-g maneuver. This motivates the concept of individual x- and y-channel target-motion models (and tuning parameters) in the elemental filters in the MMAF, which would allow adaptive filtering for maneuvers in the x- and y-channels independently [31]. The size of the FOV was also investigated. When a target came to within five kilometers of the FLIR platform, the 8 x 8 FOV was saturated with the intensity centroid image, resulting in a loss of track. This analysis motivates a changing FOV to maintain lock for targets and also warrants the possibility of adding another Kalman filter which is tuned for extremely harsh maneuvers at close ranges. A study of the aspect ratio (AR) associated with target's intensity centroid was also accomplished to identify filter tracking characteristics for various target image functions [31]. This study used "greyscale plots" to support the analysis. A greyscale plot is a pictorial display of an image in which shading of the image is used to indicate similar parameters. In this case, the plot indicated regions of varying levels of the intensity of the filter-reconstructed target image in a 24 x 24 pixel FOV. Four different AR values of 0.2, 0.5, 5.0, and 10.0 were compared to the nominal AR of 1. The results showed that tracking was slightly impaired for images with AR as high as 5. The reduced performance was primarily along the semi-major axis of an elliptically modeled intensity centroid. Additionally, a target-decoy experiment was conducted in which a high density decoy was also located in the FOV with the target. Since the decoy was modeled with different dynamics not given to the filter, it was hoped that the filter would reject the decoy. This was not the case; the filter locked onto the hotter decoy image. This indicates that the inability of the current filter

algorithm to reject this type of bright hotspot requires isolating the target image in a small FOV or some other concept to ensure tracking of the desired target.

The previous research efforts [14,31,40] used Gauss-Markov acceleration models in the development of the MMAF. Tobin [41] implemented the CTR dynamics model in another MMAF. His results showed that the Gauss-Markov MMAF exhibited smaller bias errors while the CTR MMAF gave smaller steady state standard deviation errors; both filters had comparable rms errors. Motivated by earlier research [31], he also developed an 8 x 24 pixel FOV for both the x- and y-directions of the FLIR image plane to be used with filters designed to anticipate harsh target accelerations in a specific direction (along which the longer side of the FOV would be oriented). This resulted in five elemental filters: the original three as in the research by Loving and Netzer, plus the additional two just described. The results showed that the filter maintained lock on a target during a highly dynamic maneuver in the y-direction while maintaining substantially better steady state bias performance in the benign x-direction. However, this preliminary study only tuned the filters for high acceleration in the horizontal or vertical single directions (as well as filters in which no specific direction was preferred).

Leeney [12] expanded the previously used Gauss-Markov truth model by incorporating bending vibrational states. The elemental filters in the MMAF were not modeled with this information through explicit state variables, but performed well

up to a 10-g maneuver. A performance investigation was also conducted as to the effects of increasing the measurement update rate from the previously used 30 Hz to 50 Hz. The sampling rate of 50 Hz showed a minor performance improvement, but also increased the computational loading because of the higher rate. A preliminary study was also done on replacing the 8 x 24 pixel FOV in the x- and y-directions [41] on the FLIR plane with a single rotatable 8 x 24 pixel FOV, which is also known as the rotating rectangular-field-of-view (RRFOV). The idea was to align the long side of the rectangular FOV with an estimate of the acceleration vector. The higher precision velocity estimate was actually used instead of the noisier acceleration estimate, and it was assumed that the acceleration direction would be essentially orthogonal to the current velocity vector direction for aircraft type targets and maneuvering missiles. Additionally, the five elemental Kalman filters in the MMAF bank would be reduced to four by using this FOV rotation scheme. The results were not conclusive, but the insight provided motivation to continue the study.

The RRFOV research was continued by Norton [32]. He discovered that the appropriate choice of the filter dynamics driving noise strength Q dictated the filter's response to a high-g jinking maneuver, and that the size of the FOV could be reduced to an 8 x 8 pixel rotating FOV, also known as the rotating square field of view (RSFOV). His investigation showed that a non-rotating square FOV could provide good performance, but that the dynamics noise strength Q matrix value must be large in the elements corresponding to the direction of the acceleration vector. A

mathematical matrix transformation was developed which "rotated" the Q matrix to keep the larger values aligned with the acceleration vector. A study of both the rotating FOV and rotating the Q matrix provided advantages and disadvantages for each method. Both methods are affected by the tuning parameters used to represent the rms level of acceleration of the target. The rotating FOV improves the x-direction (azimuth) estimation for dominant y-direction (elevation) dynamics compared to previous MMAF algorithms (on pull-up maneuvers), but does not improve y-direction estimation for dominant y-direction dynamics. Rotating the Q matrix adaptively improves estimation of both x- and y-directions and improves the jink maneuver onset error transients, but is dependent on the orthogonality of the velocity and acceleration vectors and proper initial tuning parameters. The conclusion was that both methods employed together provide the ability to adjust filter characteristics to differentiate between harsh and benign dynamics in any orientation of target acceleration (rotating Q) while at the same time maintaining appropriate view resolution in the directions of both benign and harsh dynamics (rotating FOV). Therefore, the combination allows for tracking highly maneuvering targets without sacrificing the resolution provided by the smaller RSFOV [32].

The research up to this point was primarily directed towards tracking aircraft and missiles from a ground-based FLIR plane. Rizzo [34] initiated research on a space-based platform which could track targets using the same filtering techniques. Since the linear Kalman filter/enhanced correlator algorithm had proven to be

computationally more efficient than the extended Kalman filter, it was chosen as the system filter for study. The plume "pogo" (oscillation) phenomenon of a missile in the boost phase of flight was modeled in the truth model and in one of two filters used for the analysis. The pogo was modeled as a second-order Gauss-Markov process, and applied in the direction of the missile velocity vector. The plan was to estimate the pogo states adaptively using the MMAF algorithm, treating the pogo amplitude and oscillation frequency as uncertain parameters. Although the elemental filters were developed, no MMAF performance was accomplished, due to elemental filter performance difficulties.

Three rotation schemes were also developed and tested. The first scheme, referred to as the rotating field-of-view (RFOV), involved using the 8 x 8 FOV filter and aligning a single axis of the FLIR plane with the estimated velocity vector of the target; therefore one of the coordinate axes of the FOV would stay aligned with the oscillation of the plume. Note that, for this class of targets, harsh maneuvering accelerations orthogonal to the velocity vector were not anticipated. The second scheme, referred to as the diagonal rotating field-of-view (DRFOV), used the 8 x 8 FOV with the diagonal aligned with the oscillation of the plume. The motivation behind this scheme is that the 8 x 8 FOV, oriented in such a fashion, will be able to "see" more of the target's intensity image, thus enabling the sensor to obtain more measurement information [34]. The third tracking scheme was the rotating rectangular field-of-view (RRFOV) algorithm developed from previous research

[12,41]. The RFOV, DRFOV, and the RRFOV algorithms [31] were tested along with the non-rotating field-of-view (NRFOV) filter. The NRFOV is the standard tracker sed in previous studies [12,31,41]. The DRFOV scheme was shown to be superior to the other three tested for providing enhanced tracking of a missile hardbody whose plume is undergoing a pogo phenomenon.

The eight-state filter (without pogo states; two target position states, two target velocity states, two target acceleration states, and two atmospheric jitter states) and the ten-state filter (with pogo states) revealed a problem that may have gone unnoticed in previous work. Following tuning of the filters with a ten-state truth model (2 target dynamics states, 6 atmospheric states, and 2 pogo states; note the bending/vibration states were removed), it was discovered that the eight-state (without pogo states) filter outperformed the ten-state (with pogo states) filter. An investigation into the cause of the irregularity revealed that there was a serious observability problem in both filters. The affected states were target velocity and acceleration. A recommendation was made to remove the acceleration states in the ten-state filter, and to model the velocity states in this new eight-state filter as a first-order Gauss-Markov process.

Eden [3] resumed the research of the space-based FLIR platform. The scope of the tracking problem was expanded by requiring the filter to track the hardbody of the missile rather than just the intensity centroid from the plume on the FLIR.

Since the FLIR could not supply the needed information about the hardbody location relative to the image center-of-intensity to the Kalman filter, another measurement source was developed. Under the advisement of the Phillips Laboratory, the new measurement source was identified as a low-energy laser. The laser actively acquires measurement data while the FLIR obtains its measurement information passively. This scheme calls for a six-state Kalman filter (consisting of two position states, two velocity states, and two atmospheric jitter states) to provide both a position vector and a velocity vector estimate of the target plume. The low-energy laser is scanned along this estimated velocity vector from the target plume image intensity center to intercept the hardbody. The hardbody is modeled as a rectangle with binary reflectivity. When the low-energy laser (modeled with a beam width of 2.75 meters at the target) illuminates the hardbody, the reflection is received by a low-energy laser sensor on the platform. This speckle information is provided to a single-state Kalman filter which estimates the distance between the center-of-mass of the missile hardbody and the center-of-intensity (centroid) of the plume along the velocity vector direction. The center-of-mass of the hardbody is defined as the midpoint of the scan across the hardbody if the centerline of the laser beam crosses the aft end of the missile and the top (nose) of the hardbody, or if the laser beam crosses the aft end and one of the sides of the hardbody. The results of the laser scan show that the interception of the laser with the hardbody occurred only 10-20% of the time. This low ratio of hitting the target was tentatively attributed to the six-state filter being tuned for estimating only the intensity centroid location on the FLIR plane and not for

precise velocity estimation. Since the velocity vector must be accurately estimated for active illumination of the target to be a viable concept, it was recommended that the filter also be tuned for accurate velocity estimates.

Tracking the center-of-mass of a missile hardbody using FLIR measurements and low-energy laser illumination was further investigated by Evans [4]. He surmised that the tracking error could provide more insight if it were separated into the x- and y- (azimuth and elevation) components, or into along-track and across-track (2-d perpendicular axes of the hardbody) components (Eden [3] had only evaluated the one along-track component). Evans proposed the latter method would provide better information relative to the principle axes directions of the physical error phenomenon. An eight-state filter was developed by augmenting Eden's six-state filter [3] with two additional bias states used to estimate the x- and y-components of offset between the plume centroid and the hardbody center-of-mass [4]. A comparison between the eight-state filter and Eden's one-state filter used in conjunction with the six-state FLIR filter, resulted in negligible difference in performance. Evans' analysis of the eight-state filter's error statistics showed that the tracking error is much greater in the along-track direction than in the across-track direction, and thus the separate one-state filter and six-state FLIR filter performed as well as the all-inclusive eight-state filter.

Aside from investigating the tracking error statistics, Evans enhanced Eden's hardbody model, which treated reflectivity as a binary on/off function, with a hardbody reflectivity model that accounted for the cylindrical nature of a missile and its typical nonorthogonality to the line-of-sight, to provide increased realism in the simulation. Two reflectivity functions, cross-sectional and longitudinal, were defined base upon empirical data obtained from a radar return off of a 20 x 249 inch cylinder with hemispherical endcaps, rotated longitudinally in the plane of the radar source [5]. As shown in Figure 1.5, the cross-sectional and longitudinal reflectivity functions

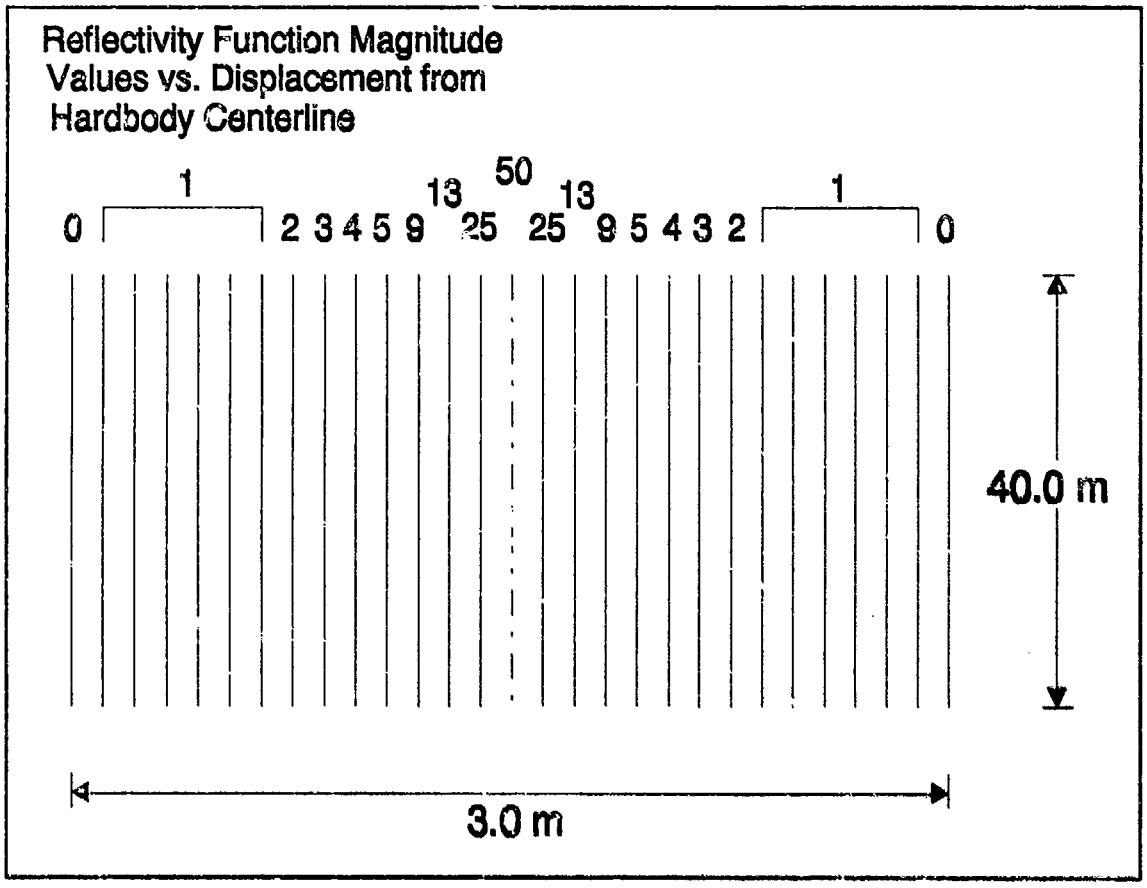


Figure 1.5 Discrete Implementation of Cross-Sectional Reflectivity Function

were incorporated into Eden's rectangular hardbody model as 29 discrete weighted line segments along the longitudinal axis of the hardbody.

Evans also found that the sensitivity level of the low-energy sensor is a factor in determining the reflectivity received at the sensor [4]. The sensitivity level represents a threshold below which the reflected return is indistinguishable from sensor noise. A sensitivity factor is incorporated in the simulation to define the appropriate sensitivity level required to detect a hardbody's return as well as represent the physical limitations of the sensor.

Performance data collection from the eight-state filter and one-state/six-state filter combination hinged upon the successful illumination of the hardbody by the low-energy laser. Evans was faced with a low target intercept rate (10%-20%), which inhibited any useful error analysis of the center-of-mass offset filters. Realizing this, Evans generated an ad hoc technique of offsetting the low-energy laser scan relative to the FLIR estimated velocity vector and "sweeping" the scan across the hardbody, thus providing guaranteed hardbody illumination information. However, the "sweep" is not an optimal tool and should only be used to test the center-of-mass offset filters in the simulation [4]. Both the new reflectivity hardbody model and laser sweep were employed to evaluate the performance of the eight-state filter and one-state/six-state filter combination center-of-mass estimators.

Herrera [9] continued to investigate the use of laser returns to determine the offset between the hardbody and the plume intensity centroid. However, he used the information contained in the Doppler spectra of the returns as opposed to the speckle reflectance magnitude. Experiments had shown that the laser speckle return of a solid-propellant rocket motor is of the same magnitude as that of the hardbody as a result of the metallic particles present in the propellant [1]. Thus, part of the plume is misinterpreted as the missile hardbody. The returns from the plume can cause a non-negligible bias in the intensity centroid to center-of-mass offset estimate of 25 to 30 meters, up to 90% of the times a laser scan is successful. This tendency was not incorporated into the simulations completed by Eden and Evans [3,4]. Herrera first showed that, as suspected, a bias existed in the offset estimates using the one-state/six-state filter combination based on plume speckle return measurements utilized by Eden and Evans.

Herrera proposed that the two types of information in the Doppler frequency spectra, magnitude of frequency shift and spread of the return spectrum, could be used to obtain a finer discernment of the plume/hardbody interface. This proposal was based on the fact that the spectral content of the hardbody and plume returns exhibit very different Doppler characteristics that should be readily distinguishable. His approach to using this information was not to simulate the Doppler phenomenon itself, but to simulate the quality of the returns provided as measurements to a single-state linear Kalman filter that estimates the offset between plume intensity

centroid and missile hardbody center-of-mass. Herrera simulated the quality of the low energy laser return as a function of laser wavelength and signal-to-noise ratio, and simulated a specified probability of no Doppler information at a given sample time due to either the plume and hardbody spectra being indistinguishable or the low-power laser beam missing the target body [9].

To prove the utility of using Doppler spectra, Herrera developed a one-state offset filter based on Doppler measurements to replace the one-state offset estimator used by Eden and Evans [3,4]. He maintained the same independent filter structure as used before, which utilized a six-state filter (four target dynamics and two atmospheric jitter states) in conjunction with the offset filter. He also developed a two-elemental-filter Modified MAP MMAF that incorporated both the speckle return and the Doppler return measurements. Both configurations successfully showed that the Doppler return information allowed more accurate determination of the hardbody/plume interface. The one-state filter based only on Doppler measurement data delivered unbiased estimates of the offset, and in the case of the two-state adaptive filter based on both Doppler and speckle information, use of Doppler spectra permitted accurate calculation of the bias in the offset measurement from the speckle return [9].

Ching [1] performed tests to determine the cause of the observability problems in the filter. He accomplished an observability Gramian analysis for six different

filter models, looking at the size of the eigenvalues relative to each other. A stochastic observability test was performed on a 12-state (6 jitter states and 6 target dynamic states; 2 each for position, velocity, and acceleration) and an 8-state (same target dynamics states with only 2 jitter states) filter model. This was then repeated without including the acceleration states in either model. The purpose of this part of Ching's thesis was to investigate whether there was some interaction between the atmospheric jitter and other states, as well as to evaluate the observability of the acceleration states. A different 8-state filter (2 position and 2 velocity states, 2 atmospheric jitter states and 2 plume pogo states) was also tested. Results from this part of the research show large variations (orders of magnitude) of the observability Gramian matrix. Positive and negative eigenvalues were seen. This was due to either true unobservable states or to a numeric precision problems associated with nearly unobservable states. The results were not completely conclusive. Several filter configurations were run for a sensitivity analysis in determining the estimatability and interaction between the jitter and pogo phenomena. Ching found that estimation of jitter is important to the accuracy of the filter; pogo estimation is easier to estimate, but is not as major a factor as jitter in the estimation accuracy. Several Monte Carlo runs were also accomplished using the AFIT software with the MMAE elemental filter. He researched one of the elemental linear KFs that will be used in the MMAE algorithm to provide adaptivity to amplitude and frequency of the plume pogo phenomenon, assuming particular values for those two parameters. Software errors introduced problems that directly affected the results from these

Monte Carlo runs. The single filter that Ching researched forms the basis from which this thesis is built upon.

1.3 THESIS OBJECTIVES

This thesis is the culmination of over fifteen years of research into the problem of accurately tracking missiles using a FLIR sensor. The focus of recent theses has been based on the data algorithm of Figure 1.2. Rizzo used this block diagram, replacing the KF block with a Multiple Model Adaptive Estimator composed of linear Kalman filters running in parallel; the intent was to adapt to amplitude and frequency of the "plume pogo" effect, but difficulties arose because (1) target acceleration states were included in the missile dynamics model and (2) only FLIR data was used, with no additional measurement to help distinguish between plume center-of-intensity location and the hardbody center-of-mass. Eden, Evans, and Herrera incorporated a Low Energy Laser (LEL) measurement update to help estimate the missile hardbody center-of-mass; disregarding "plume pogo." Ching researched one of the elemental filters that will make up the KF bank in the MMAE algorithm of this research.

This research uses a FORTRAN computer simulation to implement the block diagram of Figure 1.2. The Kalman filter block will be replaced by an MMAE incorporating a bank of parallel-running, linear KF's that are intended to propagate

and update the state estimate based on a specific assumed value of pogo amplitude and frequency. The multiple model adaptation process of the MMAE will estimate the uncertain pogo parameters of amplitude and frequency of oscillation of the vehicle plume. Although accuracy of parameter estimation is important and will be fully investigated, the primary performance criterion of interest is the accuracy with which the overall tracking algorithm follows the true target center-of-mass. The state estimate from the KF bank will be a Bayesian mixture of the outputs from each elemental KF, as shown in Figure 1.3. A certain probability weighting will be assigned to the output of each particular filter according to how well that filter is estimating the real world. The individual probabilities multiplied by that particular filter's state estimate add to form the best available state estimate as a probability-weighted average. The uncertain parameters of amplitude and frequency of oscillation will be estimated in a similar manner [17].

The KF's in the MMAE scheme will be programmed and tuned for a specific real world condition of the plume pogo amplitude and frequency of oscillation. The Bayesian blend of these KF's (Figure 1.3) follows the slowly varying parameters of amplitude and frequency of oscillation of the plume pogo observed in the real world. The elemental KF's will be individually tuned to provide optimal performance while adequately discretizing the parameter space. The proper discretization of the parameter space is one of the focuses of this research as this is critical in identifying the pogo parameters.

1.4 THESIS OVERVIEW

This chapter has introduced the problem statement and given a historical perspective on this work with the description of all of the past work in this research strain. Chapter 2 will describe the basic theory of Kalman filters and the development of Multiple Model Adaptive Estimation (MMAE) theory. The simulation space for the FLIR and target vehicle and their reference frames will also be described in this chapter. Chapter 3 details the dynamics and measurement models from the real world truth model with Chapter 4 explaining the Kalman filter dynamics and measurement models. Chapter 5 will discuss the obtained results and Chapter 6 will contain the conclusions and recommendations for further study.

II. Kalman Filters, MMAE and Simulation Coordinate Frames

2.1 Introduction.

This chapter presents the basics of Kalman Filters (KF's), Multiple Model Adaptive Estimation (MMAE) theory, and a description of the coordinate frames used in this simulation. The theoretical basis and derivation of KF's has existed for a long time and thus will not be rigorously developed. Much of this chapter's material is taken from the excellent work from previous theses [1,3,4,9,34]. The elemental filters within the MMAE algorithm are based on the theory presented in the KF section. The MMAE section will essentially be taken from Herrera [9] and Maybeck, Volume II, Section 10-8 [17]; the reader is asked to consult these references for further clarification of any ideas that are not made completely understandable.

2.2 Kalman Filter Theory

2.2.1 Introduction. The KF is a recursive optimal data processing algorithm. It is recursive in that the same algorithm is used over and over using the entire time history of data. This algorithm is optimal if the system can be represented by a linear system model driven by white Gaussian noise. The KF

estimate is constructed from a mathematical model that propagates an estimate over a measurement sample period. The model is started from some known initial condition mean value with a known initial covariance. The initial statistics are represented by the following equations:

$$E\{\, x(t_0)\} = \hat{x}_0 \tag{2.1}$$

$$E\{\, [x(t_0) - \hat{x}_0][x(t_0) - \hat{x}_0]^T \,\} = P_0 \tag{2.2}$$

where the notation (^) indicates an estimated value and E{ } is the expectation or ensemble average of possible outcomes. A variable is displayed in boldface lower case characters when representing a vector quantity; a matrix is denoted by boldface capital letters. At each sample time an update is performed by the filter using the measurement, the predicted state estimate, and a computed gain. The filter then propagates this estimate to just prior to the next sample time, when the next update is performed. Statistics (expected value and noise strength or covariance) of the driving white Gaussian noise and the measurement noise must also be known *a priori*.

The KF receives measurements at a certain sample rate and propagates the state conditioned upon the measurement time history $\mathbf{Z}(t_j)$, which is given as: where $\mathbf{z}(t_j)$ is the measurement data available at sample time t_j. The conditional mean and covariance of the state variables are given by:

$$\mathbf{Z}(t_i) = \begin{bmatrix} z(t_1) \\ z(t_2) \\ . \\ . \\ z(t_i) \end{bmatrix} \tag{2.3}$$

where $z(t_j)$ is the measurement data available at sample time t_j. The conditional mean and covariance of the state variables are given by:

$$\hat{x}(t_i^+) = E\{ x(t_i) | \mathbf{Z}(t_i) = \mathbf{Z}_i \} \tag{2.4}$$

$$P(t_i^+) = E\{ [x(t_i) - \hat{x}(t_i^+)][x(t_i) - \hat{x}(t_i^+)]^T | \mathbf{Z}(t_i) = \mathbf{Z}_i \} \tag{2.5}$$

where $\mathbf{Z}_i$ is a realization or observed set of values of the measurement history $\mathbf{Z}(t_i)$.

An important feature of the KF is that it provides a measure of how well it is doing the job of estimating the real world. This is through its residual: the difference between the actual measurement and the filter's best prediction of that measurement before it arrives. If this residual value is large, the filter is not estimating real world conditions very well. This property of the filter is exploited in the MMAE scheme, as will be seen in Section 2.3.

2.2.2 Linear Kalman Filter. The mathematical model that is used for the basis of a linear Kalman Filter (LKF) is of the form:

$$\dot{x}(t) = F(t)x(t) + B(t)u(t) + G(t)w(t) \tag{2.6}$$

where:

$F(t)$	=	Homogeneous system state dynamics matrix
$x(t)$	=	State vector
$B(t)$	=	Deterministic input matrix
$u(t)$	=	Deterministic control input vector
$G(t)$	=	Driving noise input matrix
$w(t)$	=	White Gaussian driving noise vector

The mean and strength of the white Gaussian dynamics driving noise is given by:

$$E\{w(t)\} = 0 \tag{2.7}$$

$$E\{w(t)w(t+\tau)^T\} = Q(t)\delta(\tau) \tag{2.8}$$

The linear system described by Equations (2.1) - (2.3) is a continuous-time system that represents the real world. This real world must be described by a discretized system since the simulation and filter implementation will be done using a digital computer. This discrete-time system is described by:

$$x(t_{i+1}) = \Phi(t_{i+1},t_i)x(t_i) + B_d(t_i)u(t_i) + G_d(t_i)w_d(t_i) \tag{2.9}$$

where:

$\Phi(t_{i+1},t_i)$ = The system state transition matrix which is defined as an n-by-n matrix that satisfies the differential equation and initial condition given by:

$$\frac{d\Phi(t,t_i)}{dt} = F(t)\Phi(t,t_i) \tag{2.10}$$

$$\Phi(t_i,t_i) = I \tag{2.11}$$

and:

$x(t_i)$ = Discrete-time vector of states of interest

$B_d(t_i)$ = Discrete-time control input matrix

$u(t_i)$ = Discrete-time deterministic control input vector

$w_d(t_i)$ = Discrete-time independent, white Gaussian noise process, independent of $x(t_o)$, with mean and covariance statistics defined as:

$$E\{w_d(t_i)\} = 0 \tag{2.12}$$

$$E\{w_d(t_i)w_d(t_j)^T\} = \begin{cases} Q_d(t_i) & t_i = t_j \\ 0 & t_i \neq t_j \end{cases} \tag{2.13}$$

with:

$$Q_d(t_i)=\int_{t_i}^{t_{i+1}} \Phi(t_{i+1},\tau)G(\tau)Q(\tau)G(\tau)^T\ \Phi(t_{i+1},\tau)^T\ d\tau \tag{2.14}$$

The KF incorporates measurements from external measuring devices at each sample time to improve the state estimate. The discrete-time linear measurement model is of the form:

$$z(t_i) = H(t_i)x(t_i) + v(t_i) \tag{2.15}$$

where:

$z(t_i)$	=	m-dimensional measurement vector at sample time t_i
$H(t_i)$	=	State observation matrix
$x(t_i)$	=	State vector
$v(t_i)$	=	White Gaussian measurement noise

The discrete-time white Gaussian measurement noise v is independent of both $x(t_0)$ and w for all times, and has a mean and covariance, R, given by:

$$E\{\ v(t_i)\} = 0 \tag{2.16}$$

$$E\{\ v(t_i)v(t_j)^T\ \} = \begin{cases} R(t_i) & t_i = t_j \\ 0 & t_i \neq t_j \end{cases} \tag{2.17}$$

The KF propagates the state conditional mean and its covariance from the instant in time immediately following the most recent measurement update, t_i^+, to the

instant in time immediately preceding the next measurement update, t_{i+1}^-, by integration (accomplished numerically, or analytically by means of state transition matrices) of the following equations:

$$\dot{\hat{x}}(t/t_i) = F(t)\hat{x}(t/t_i) \tag{2.18}$$

$$\dot{P}(t/t_i) = F(t)P(t/t_i) + P(t/t_i)F(t)^T + G(t)Q(t)G(t)^T \tag{2.19}$$

where the notation $\hat{x}(t/t_i)$ denotes optimal estimates of x at time t, conditioned on measurements through time t_i, and with initial conditions:

$$\hat{x}(t_i/t_i) = \hat{x}(t_i^+) \tag{2.20}$$

$$P(t_i/t_i) = P(t_i^+) \tag{2.21}$$

where $\hat{x}(t_i^+)$ and $P(t_i^+)$ are the results of the previous measurement update cycle. At time t_0, $\hat{x}_0$ and P_0 from Equations (2.1) and (2.2) are used to initialize the first propagation.

That update cycle when a measurement becomes available at time t_i is based on the following equations:

$$K(t_i) = P(t_i^-)H(t_i)^T[H(t_i)P(t_i^-)H(t_i)^T + R(t_i)]^{-1} \tag{2.22}$$

$$\hat{x}(t_i^+) = \hat{x}(t_i^-) + K(t_i)[z(t_i) - H(t_i)\hat{x}(t_i^-)] \tag{2.23}$$

$$P(t_i^+) = P(t_i^-) - K(t_i)H(t_i)P(t_i^-) \tag{2.24}$$

where $\hat{x}(t_i^-)$ and $P(t_i^-)$ are the result of the propagation integration up to time t_i, $\mathbf{K}(t_i)$ is the time-varying Kalman filter gain matrix that assigns "weights" to the new information (consisting of the difference between the actual measurement and the filter's estimate of the measurement, $H(t_i)\hat{x}(t_i^-)$, as seen in Equation (2.23)), based on known measurement noise statistics and filter-computed covariances.

In some instances (as with the case when pogo is included in the elemental filter), the discrete-time measurement update is a known <u>nonlinear</u> function of the state vector. In such cases, the following nonlinear extended KF update model is used in place of Equation (2.15). The measurements are modeled as:

$$z(t_i) = h[x(t_i),t_i] + v(t_i) \tag{2.25}$$

where $h[x(t_i),t_i]$ is a known vector of functions of state and time, and $v(t_i)$ is the same discrete-time white Gaussian measurement noise as defined before. When a nonlinear measurement is available, Equation (2.22) is still used to determine the KF gain matrix, but the matrix H is defined by:

The updated state vector becomes a function of the nonlinear residual, $[z(t_i)-h[\hat{x}(t_i^-),t_i]]$ and Equation (2.23) is modified to become:

$$\hat{x}(t_i^+) = \hat{x}(t_i^-) + K(t_i)\{ z(t_i) - h[\hat{x}(t_i^-),t_i]\} \tag{2.27}$$

2.3 *Multiple Model Adaptive Estimation*

The optimality of the state estimator is dependent upon complete knowledge of the parameters that define the best model for system dynamics, output relations, and statistical description of uncertainties [17]. For Kalman filter tracking applications, maximum performance is achieved when the parameters of the filter dynamics model match the parameters of the target being tracked. Often, the parameters are known only with some uncertainty and may exhibit time-varying characteristics (such as in the case of maneuvering targets with changing acceleration levels). Thus, there is a need to devise a method that produces optimum state estimates despite the incomplete *a priori* knowledge of parameter values, and provides the estimates in an adaptive, on-line fashion. The MMAE satisfies these requirements [17].

To implement the MMAE algorithm, it becomes necessary to discretize the parameter space by the judicious choice of discrete values that are representatively dispersed throughout the continuous range of possible values. For the tracking

To implement the MMAE algorithm, it becomes necessary to discretize the parameter space by the judicious choice of discrete values that are representatively dispersed throughout the continuous range of possible values. For the tracking problem at hand, a target is assumed to be able to display K different discrete sets of pogo conditions corresponding to one of K discrete combinations of pogo oscillation frequency and amplitude. As previously shown in Figure 1.3, a Kalman filter is then designed for each choice of parameter value, resulting in a bank of K separate elemental filters.

Let $\boldsymbol{a}$ denote the vector of uncertain parameters in a given linear time-invariant state model for a dynamic system. A system model would be represented by the following first-order, stochastic differential equation:

$$\dot{x}(t) = F(a)x(t) + B(a)u(t) + G(a)w(t) \tag{2.28}$$

with noise corrupted, discrete-time measurements given by:

$$z(t_i) = H(a)x(t_i) + v(t_i) \tag{2.29}$$

where:

$x(t)$	=	n-dimensional system state vector
$u(t)$	=	r-dimensional deterministic control vector
$w(t)$	=	s-dimensional white, Gaussian, zero-mean noise vector process of strength $Q(a)$
$z(t_i)$	=	m-dimensional measurement vector

$v(t_i)$	=	m-dimensional discrete-time white, Gaussian, zero-mean noise vector process of covariance $\boldsymbol{R}(\boldsymbol{a})$
$\boldsymbol{F}(\boldsymbol{a})$	=	n x n system plant matrix
$\boldsymbol{B}(\boldsymbol{a})$	=	n x r input distribution matrix
$\boldsymbol{G}(\boldsymbol{a})$	=	n x s noise distribution matrix
$\boldsymbol{H}(\boldsymbol{a})$	=	m x n matrix relating measurement to states

Note that $\boldsymbol{F}$, $\boldsymbol{B}$, $\boldsymbol{G}$, $\boldsymbol{H}$, $\boldsymbol{Q}$, and $\boldsymbol{R}$ are assumed to be functions (potentially) of the parameter $\boldsymbol{a}$, but <u>not</u> of time t.

The parameter vector, $\boldsymbol{a}$, is discretized into a set of K finite vector values, $\boldsymbol{a}_1$, $\boldsymbol{a}_2$, ... , $\boldsymbol{a}_K$, and associated with each $\boldsymbol{a}_k$ is a different system model of the form given by Equations (2.28) and (2.29). Each elemental Kalman filter, tuned for a specific $\mathbf{a}_k$, produces a state estimate which is weighed appropriately using the hypothesis conditional probability $p_k(t_i)$ to produce the state estimate $\hat{\boldsymbol{x}}_{mmae}(t_i)$ as a probability weighted average, as shown in Figure 1.3, where:

$$p_k(t_i) = \frac{f_{x(t_i)/a,Z(t_{i-1})}(z_i \;/a_k, Z_{i-1})p_k(t_{i-1})}{\sum_{j=1}^{K} f_{x(t_i)/a,Z(t_{i-1})}(z_i \;/a_j, Z_{i-1})p_j(t_{i-1})} \tag{2.30}$$

$$f_{x(t_i)/a,Z(t_{i-1})}(z_i \;/a_k, Z_{i-1}) = \frac{\exp\{\cdot\}}{(2\pi)^{m/2}\left|A_k(t_i)\right|^{1/2}} \tag{2.31}$$

where:

$$\{\cdot\} = \{-\frac{1}{2} r_k(t_i)^T A_k(t_i)^{-1} r_k(t_i)\}$$

with:

$A_k(t_i)$	=	kth filter's computed residual covariance
	=	$H_k,(t_i)P_k(t_i^-)H_k(t_i)^T + R_k(t_i)$
$r_k(t_i)$	=	kth filter's residual
	=	$[z(t_i) - H_k(t_i)\hat{x}_k(t_i^-)]$
a_k	=	Parameter value assumed in the kth filter
$P_k(t_i^-)$	=	kth filter's computed state error covariance before incorporating the measurement at time t_i
$z(t_{i-1})$	=	Measurement history up to time t_{i-1}

The residual of the kth elemental Kalman filter, that best matches the current pogo conditions associated with the parameter value a_k, is expected to be smaller than the residuals of the other mismatched filters, so that the exponential term in Equation (2.31) is smallest for the kth elemental filter. Therefore, the hypothesis conditional probability given by Equation (2.30) with index corresponding to the "correct" filter will then be largest among the conditional probabilities, thus assigning the most weight to the "correct" state estimate. This algorithm performs well if each elemental filter is optimally tuned for best performance for a specific pogo condition, causing its residual to be distinguishable from those of the mismatched filters. It is also important not to add excessive amounts of pseudonoise to compensate for model

magnitude, then Equation (2.30) will result in the growth of the p_k associated with the filter with the smallest value of $|A_k|$. The values of $|A_k|$ are independent not only of the residuals, but also of the "correctness" of the K models, and so the result would be totally erroneous [17]. Therefore, the scalar denominator in the right hand side of Equation (2.31) might be removed in the final implementation of the algorithm.

The output of the MMAE algorithm is the probability-weighted average of the elemental filters' estimates, given by:

$$\hat{x}_{mmae}(t_i^+) = \sum_{k=1}^{K} p_k(t_i)\hat{x}_k(t_i^+) \tag{2.32}$$

The conditional covariance matrix for the MMAE is computed as:

$$P_{mmae}(t_i^+) = \sum_{k=1}^{K} p_k(t_i)[P_k(t_i^+) + \hat{y}_k(t_i^+)\hat{y}_k(t_i^+)^T] \tag{2.33}$$

where:

$\hat{y}_k(t_i^+)$ = $\hat{x}_k(t_i^+) - \hat{x}_{mmae}(t_i^+)$

$p_k(t_i^+)$ = kth filter's conditional hypothesis probability

$P_k(t_i^+)$ = kth filter's state error covariance matrix after incorporating the measurement at time t_i

$P_k(t_i^+)$ = kth filter's state error covariance matrix after incorporating the measurement at time t_i

Since the values of $p_k(t_i)$ and $\hat{x}_{\text{mmae}}(t_i^+)$ depend upon the discrete measurements taken through time t_i, $P_{\text{mmae}}(t_i^+)$ cannot be precomputed as in the case for the elemental filters. However, Equation (2.30) need not be computed for the on-line filter algorithm.

The calculated probabilities of Equation (2.30) should be modified by an artificial lower bound [12,17,31]. This lower bound will prevent a mismatched filter's hypothesis conditional probability from converging to (essentially) zero. If a filter's p_k should reach zero, it will remain zero for all time, as can be seen from the iterative nature of Equation (2.30). This effectively removes that filter from the bank and degrades the responsiveness of the MMAE to future changes of the parameter values. If some future pogo condition matched the model for which the p_k was locked onto zero, that elemental filter's estimate would not be appropriately weighted and the MMAE estimate would be in error. In previous work, Tobin [41] established a lower bound of 0.001 for $p_k(t_i)$. After the lower bound is imposed, the p_k's are rescaled so they add to one.

2.4 Simulation Space

Simulation of the tracking scenario, which encompasses the target trajectory, the FLIR sensor operation, and the low-energy laser illumination of the missile hardbody and the generation of the speckle return and Doppler measurements, is performed on a digital computer. A 3-dimensional "simulation space" is generated wherein a target body and plume are propagated along a realistic trajectory. Several coordinate frames in the simulation space provide the means of mathematically projecting the target plume's infrared image and velocity vector onto the two-dimensional FLIR image plane [5,6,27]. In addition, these frames are utilized to project a representation of the hardbody center-of-mass, as well as to define the start and orientation of the low-energy laser scan [1] for generating speckle and Doppler measurements for the filter algorithm. This chapter describes the different coordinate frames of the simulation space and covers the process of pointing the FLIR sensor at the target during tracking. This section is taken largely from Herrera's work [9].

2.4.1 Coordinate Frames. As shown in Figure 2.1, three primary coordinate frames are defined in the simulation space: a system inertial reference frame, a target reference frame, and an α-β-r reference frame. Each of these reference frames is described in the following paragraphs.

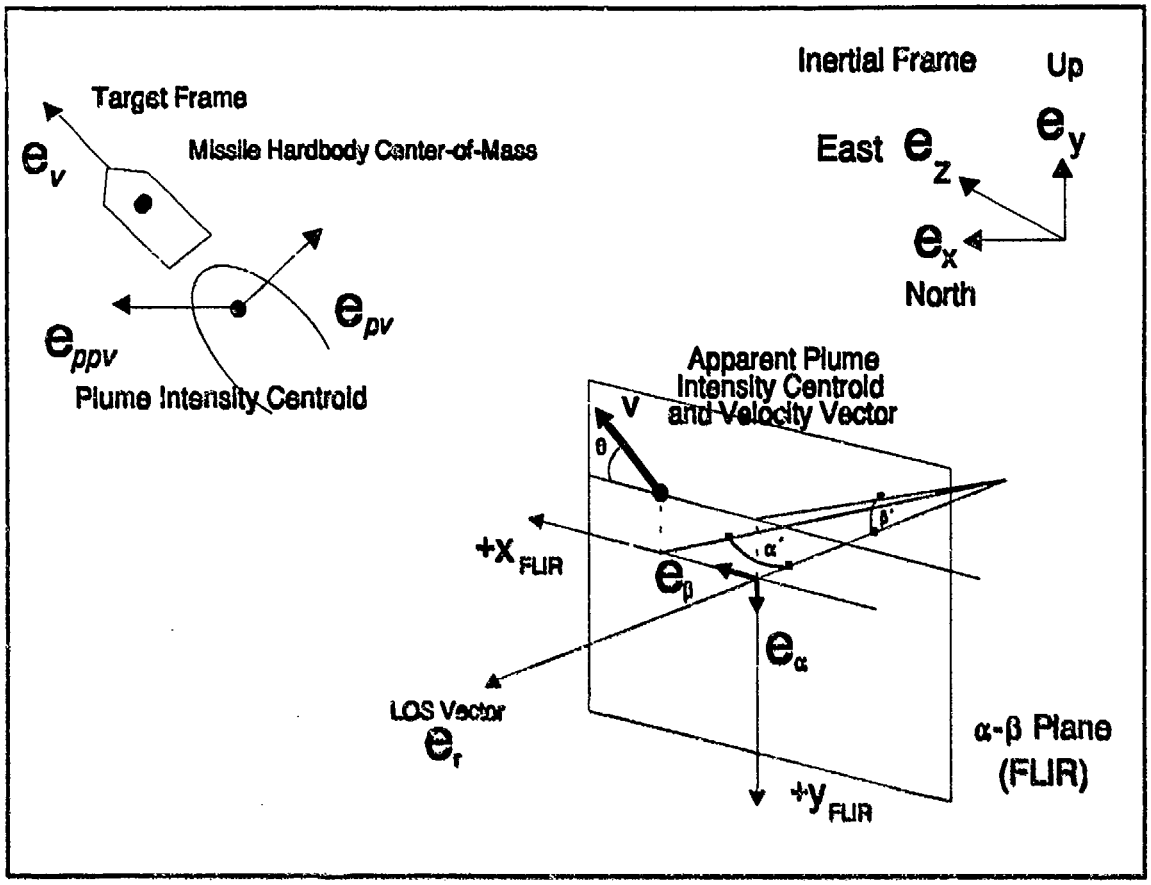


Figure 2.1 Three Primary Coordinate Frames in Simulation Space

2.4.1.1 Inertial Reference Frame. The inertial reference frame is a North-Up-East (NUE) frame wherein the target flight trajectory occurs.

Origin: location of the FLIR sensor

Axes: e_x - due north, tangent to the earth's surface, defines zero azimuth

e_y - inertial "up"

e_z - vector completing right-hand coordinate set, defines 90° azimuth

Note: The azimuth angle (α) is measured eastward from $\mathbf{e}_x$. The elevation angle (β) is measured "up" from the horizontal plane defined by $\mathbf{e}_x$ and $\mathbf{e}_z$.

2.4.1.2 Target Plume Reference Frame. This frame is located at the target plume with one of its unit vectors co-linear with the target's true velocity vector.

Origin: plume intensity centroid

Axes: $\mathbf{e}_v$ - along the true velocity vector

$\mathbf{e}_{pv}$ - out the right side of the target, orthogonal to both $\mathbf{e}_v$ and the LOS vector ('*pv*' means "perpendicular to the velocity vector"; note that e_v and e_{pv} form a plane parallel to the FLIR image plane, to be discussed in Section *2.4.1.4*)

$\mathbf{e}_{pvv}$ - vector completing the right-hand coordinate set

2.4.1.3 α-β-r Reference Frame. The α-β-*r* reference frame is defined by the azimuth angle α' and the elevation angle β' measured with respect to the FLIR line-of-sight (LOS) vector $\mathbf{e}_r$. The true azimuth α and the true elevation β are referenced from the true north and the horizon. This frame is used to project the target's position and velocity onto the FLIR plane.

Origin: plume intensity centroid

Axes: $\mathbf{e}_r$, coincident with the true sensor-to-target LOS vector; $\mathbf{e}_\alpha$ and $\mathbf{e}_\beta$ define a plane perpendicular to $\mathbf{e}_r$, rotated from inertial $\mathbf{e}_x$ and $\mathbf{e}_y$ by the azimuth angle (α) and elevation angle (β)

There are three special coordinate frames associated with the α-β-r reference frame: the α-β (FLIR) plane, the absolute α-β-r reference frame, and the trans-FLIR plane.

2.4.1.4 α-β (FLIR Image) Plane. The FLIR plane is used to obtain the measurements of the target plume position and is the reference frame for the geometrically derived velocity vector components of the target's intensity centroid. The FLIR plane is defined by the $\mathbf{e}_\alpha$ and $\mathbf{e}_\beta$ unit vectors, with the LOS vector (orthogonal to the FLIR plane) representing the pointing orientation of the FLIR sensor, and the high and low-energy lasers. Note the orientation of the $+y_{FLIR}$ axis in Figure 2.1, which allows the LOS vector to be positive towards the target, where it is considered the third member of a right-handed set of coordinates as defined by the unit vectors $\mathbf{e}_\beta$, $\mathbf{e}_\alpha$, and $\mathbf{e}_r$.

Due to the large distance to the target (approximately 2,000 kilometers), small angle approximations are invoked, allowing the "pseudo" azimuth and elevation angles, α' and β', to be linearly proportional to the x and y cartesian coordinates in the FLIR plane. The x and y coordinates are measured in pixels (a pixel of linear length corresponds to 15 microradians of arc) and will provide a means of evaluating the performance of the Kalman filter associated with tracking the intensity centroid of the target.

2.4.1.5 Absolute α-β-r Reference Frame. The absolute α-β-r reference frame is fixed in inertial space at the *initial* α-β-r coordinates of the target. This coordinate system defines the initial pointing direction of the FLIR LOS vector $\mathbf{e}_r$, and is also used to define the true and filter-estimated target positions and velocity components on the FLIR plane.

2.4.1.6 Trans-FLIR Plane. This plane is defined as the result of translating the center of the FLIR FOV to the true center-of-mass of the missile hardbody. The frame is used to determine the x_{FLIR} and y_{FLIR} coordinate errors of the hardbody center-of-mass filter's estimates, for performance analysis purposes.

2.4.1.7 ALT/ACT (Along-Track/Across-Track) Plane. This plane, shown in Figure 2.2, is a rotation of the trans-FLIR plane by the true orientation angle θ_t, formed by the target trajectory with respect to the FLIR coordinate plane. It is used to determine the along-track and across-track components of the tracking error, i.e., the mean and covariance of the hardbody center-of-mass estimates [4].

2.4.2 FLIR Image Plane. All dynamic events associated with the target plume intensity "pattern" or "function," and with the active illumination of the missile hardbody in 3-dimensional inertial space, are projected onto the 2-dimensional FLIR image plane. The measurements generated as a result of IR detection by the FLIR sensor are provided to the enhanced correlator algorithm, which produces "pseudo-

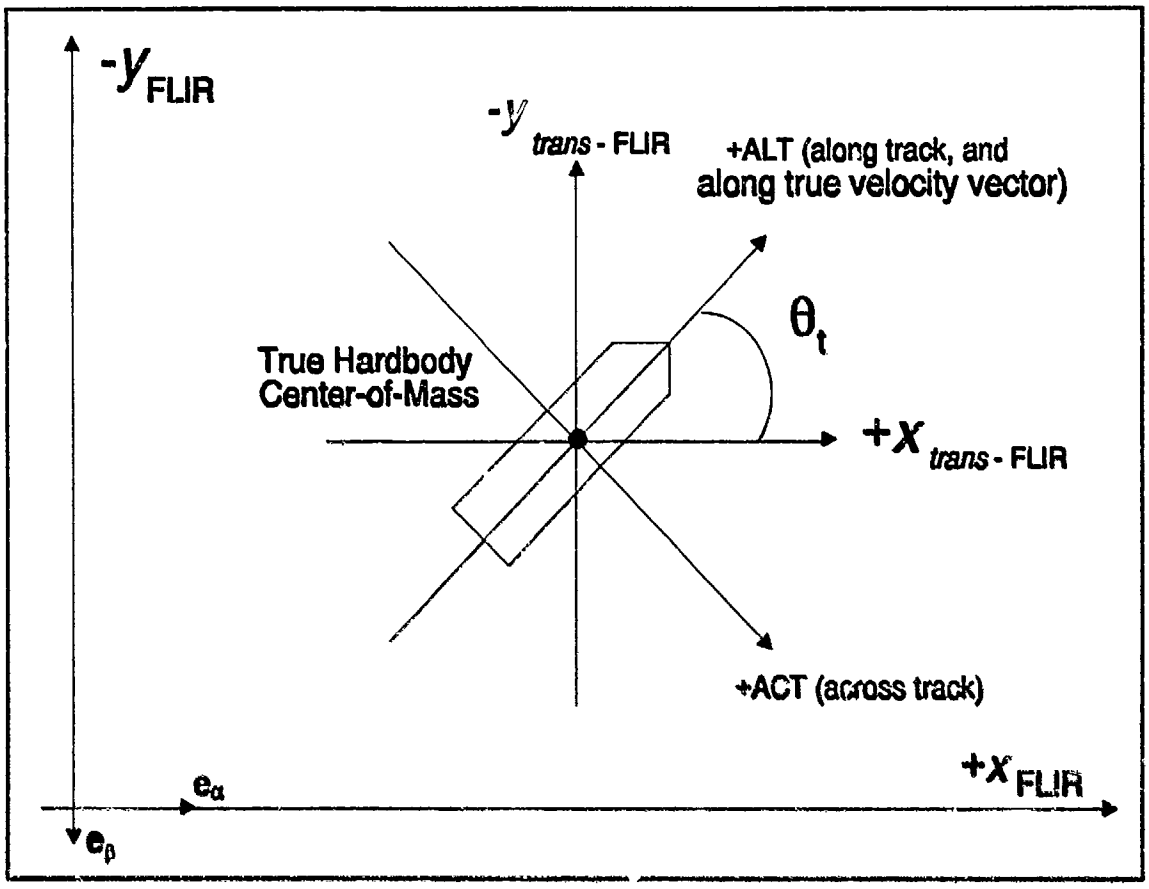


Figure 2.2 FLIR Plane, Trans-FLIR Plane, and ALT-ACT Plane

measurements" to the FLIR Kalman filter to update its state estimates. For the missile hardbody, LEL-generated measurements of the offset distance between the hardbody center-of-mass and the plume intensity centroid are geometrically projected onto the FLIR image plane. Thus, the FLIR image plane is the realm in which the performance of the MMAE filter is evaluated. Also note that it is a natural plane for such evaluation of a laser weapon, since pointing angle errors are critical and range is not. This section introduces the FLIR Field-of-View (FOV) "tracking window," and discusses the construction and projection of the target models.

2.4.2.1 FLIR Field-of-View. The FLIR FOV, shown in Figure 2.3, consists of an 8 x 8 pixel sub-array (in the FLIR sensor 300 x 500 pixel array) which provides sensed information as a function of the varying intensity of the plume IR image and the background and internal FLIR noise. Based upon this information, the position estimates from each six-state FLIR Kalman filter within the MMAE algorithm serve to center the centroid of the plume IR image in the FOV. Since the low-energy laser is boresighted with the FOV, the FLIR filter position and velocity estimates of the intensity centroid define the origin and orientation of the laser scan to "paint" the

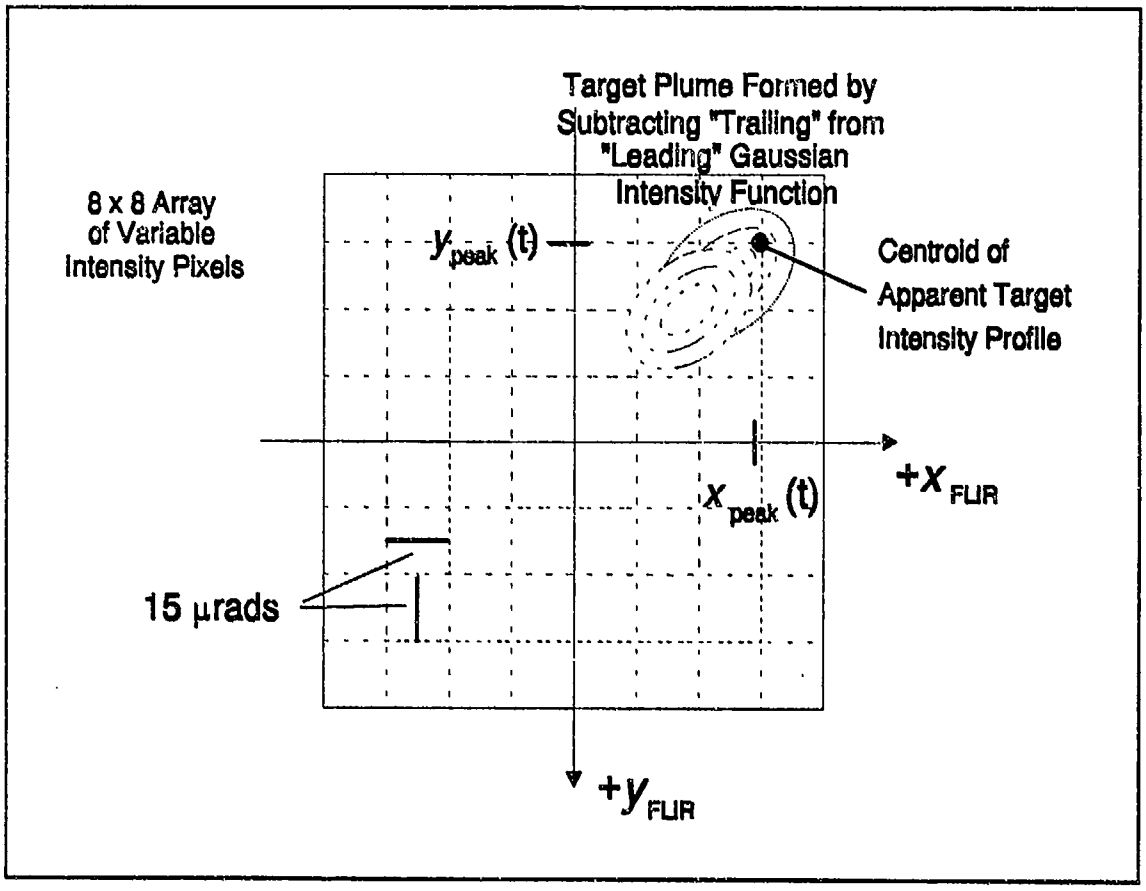


Figure 2.3 Target Plume Image in 8 x 8 FLIR Field-of-View

hardbody. The errors of the FLIR filter's estimate of the centroid position and velocity, and the hardbody center-of-mass filter's estimate of offset, are expressed in units of "pixels." These errors become meaningful through a pixel proportionality constant, k_p, equal to 15 microradians per pixel [19]. With this constant, 1 pixel corresponds to approximately 30 meters for a range of 2,000 kilometers.

2.4.2.2 Target Models on the FLIR Plane. The difference of two Gaussian intensity functions creates a planform that models the hotspot of the plume target on the FLIR plane [19], as shown in Figure 2.3. The "trailing" function is subtracted from the "leading" function to construct a suitable approximation of empirically observed plume intensity profiles. The missile hardbody is not sensed by the FLIR sensor. However, it is geometrically projected onto the FLIR plane as a rectangle, located an offset distance from the plume centroid along the target's velocity vector. Since the FLIR sensor can only detect the IR intensity shape function of the plume, the remainder of this discussion emphasizes the intensity centroid model. More about the hardbody model will be presented in Chapter III.

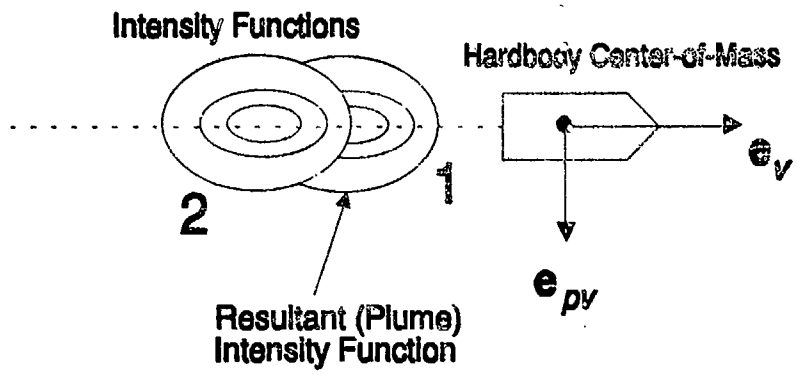
2.4.2.3 Target Plume Model on the FLIR Plane. The radiated energy intensity from each intensity function is represented as a bivariate Gaussian distribution with elliptical constant intensity contours. Each of the two bivariate Gaussian intensity functions has its semi-major axis aligned with the target velocity vector as seen in the FLIR image plane, and is given by [19]:

$$I[x,y,x_{peak}(t),y_{peak}(t)] = I_{\max}\exp[-0.5(\Delta x\Delta y)\boldsymbol{P}^{-1}(\Delta x\Delta y)^T] \tag{2.34}$$

where:

Δx	=	$(x - x_{\text{peak}})\cos\theta_t + (y - y_{\text{peak}})\sin\theta_t$, measured along the ALT axis of Figure 2.2
Δy	=	$(y - y_{\text{peak}})\cos\theta_t - (x - x_{\text{peak}})\sin\theta_t$, measured along the ACT axis of Figure 2.2
θ_t	=	True target orientation angle between the projection of the velocity vector and the x-axis in the FLIR plane; see Figure 2.2
x, y	=	Coordinate axes on the α-β plane
$x_{\text{peak}}, y_{\text{peak}}$	=	Peak intensity coordinates of each single Gaussian intensity function
$I_{\max}$	=	Maximum intensity function
$\boldsymbol{P}$	=	2 x 2 target dispersion matrix whose eigenvalues (σ_v^2 and σ_{pv}^2) define the dispersion of the elliptical constant intensity contours

Figure 2.4 illustrates the spatial relationship between the two intensity functions along the target $\mathbf{e}_v$ axis. The difference between the intensity functions is the equilibrium displacement if there were no pogo effect; the "pogo" causes oscillations about this equilibrium point. The displacement values are based on the assumption



Gaussian Intensity Function	Displacement of Centroid along e_v	Displacement of Centroid along e_{pv}	Displacement of Centroid along e_{ppv}
1	- 65 meters	0 meters	0 meters
2	- 110 meters	0 meters	0 meters

Figure 2.4 Spatial Relationship of Target Plume Gaussian Intensity Functions

that the dispersion of the exhaust plume in the e_{pv} direction (normal to both e_v and the LOS vector) is approximately 20 *times* the diameter of the missile [19]. With the dimensions of the hardbody chosen as 40 meters long and 3 meters in diameter, the centroid of the first intensity function is located 65 meters behind the hardbody center-of-mass. The placement of the first centroid simulates the composite centroid of the exhaust plume being close to the missile exhaust nozzle, whereas the position of the second centroid enables one to simulate different plume shapes. The second, "ng" centroid is arbitrarily located 110 meters from the center-of-mass, and the

defined spatial relationship remains fixed in the target frame during the simulation (should the difference between the two Gaussian intensity functions become negative, the simulation clips the difference to zero). Any external forces acting on the missile other than thrust and gravity are assumed negligible, which thus yields an assumed zero sideslip angle as well as zero angle of attack. These assumptions allow the semi-major axes of the elliptical constant-intensity contours to be aligned with the projection of the target's velocity vector onto the FLIR image plane, and provides a simplified simulation geometry while retaining the essential features of the trajectory simulation.

2.4.2.4 Target Plume Projection onto the FLIR Plane. As the target plume is propagated through inertial space, the output of the FLIR pixels is simulated by projecting the two intensity functions onto the FLIR plane. The geometry of the projection is shown in Figure 2.5. The "reference target image" is oriented on the FLIR plane to correspond to the largest apparent planform (i.e., with its velocity vector orthogonal to the LOS vector) at a given initial reference range, r_0. As seen in Figure 2.6, the target intensity image is defined by the dispersion along the principle axes of the two Gaussian intensity functions, given by:

$$\sigma_{pv} = \sigma_{pvo}\left(\frac{r_o}{r}\right) \tag{2.35}$$

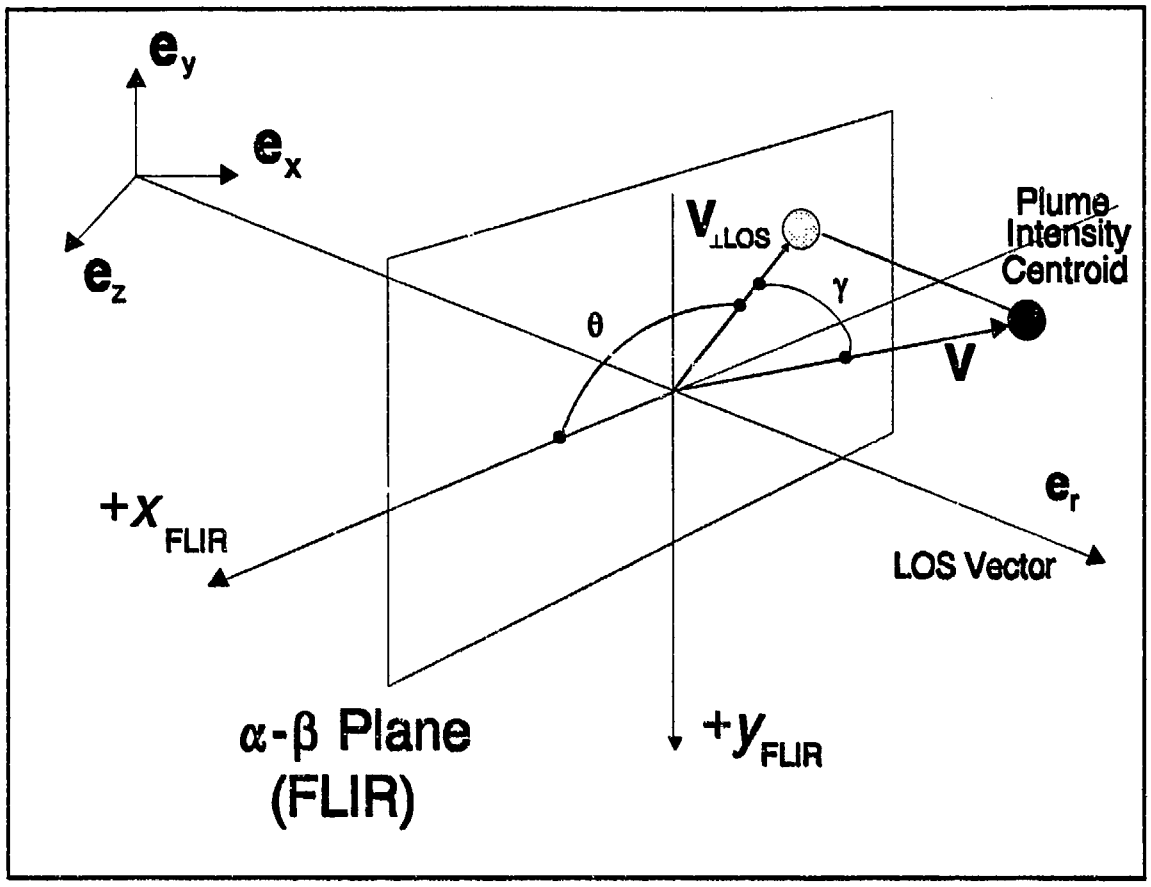


Figure 2.5 Target Plume Intensity Centroid Projection Geometry

$$\sigma_v = \left(\frac{r_o}{r}\right)[\sigma_{pvo} + (\sigma_{vo} - \sigma_{pvo})\cos\gamma)]$$
$$= \sigma_{pv}\left[1 + \frac{v_{\perp LOS}}{v}(AR - 1)\right] \tag{2.36}$$

where:

$\sigma_{vo}, \sigma_{pvo}$ = Initial dispersions of the target intensity functions along e_v and e_{pv} in the target frame of the reference image

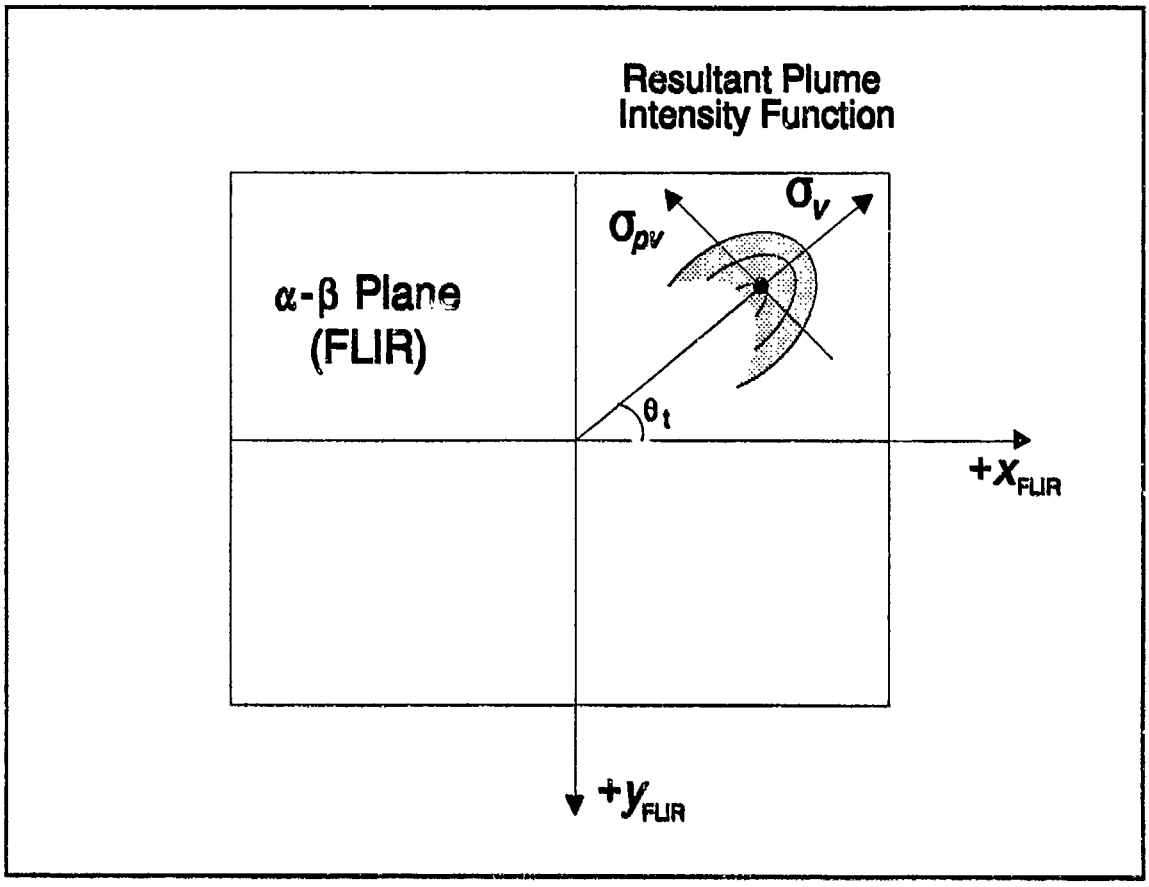


Figure 2.6 Intensity Centroid Dispersion Axes in FLIR Plane

σ_v, σ_{pv}	=	Current dispersions of the target image
r_o	=	Initial sensor-to-target range of the reference image
r	=	Current sensor-to-target range
$\mathbf{v}$	=	Initial velocity vector of the target
v	=	Magnitude of $\mathbf{v}$
$\mathbf{v}_{\perp LOS}$	=	Projection of $\mathbf{v}$ onto the α-β plane (FLIR); i.e., the component of $\mathbf{v}$ perpendicular to the LOS vector

$v_{\perp LOS}$ = Magnitude of $\mathbf{v}_{\perp LOS}$:

$$v_{\perp LOS} = [\dot{\alpha}^2 + \dot{\beta}^2]^{1/2} \tag{2.37}$$

γ = Target aspect angle between $\mathbf{v}$ and the α-β (FLIR) plane

θ = Angle between $\mathbf{v}_{\perp LOS}$ and $+x_{\text{FLIR}}$

AR = σ_{vo}/σ_{pvo}: aspect ratio of the reference image

Referring back to Figure 2.4, the location of each intensity function, or "hotspot," is initialized as a displacement from the hardbody center-of-mass. The intensity functions are oriented in the FLIR plane via the true target orientation angle θ_t. The relative positions of the two intensity functions in the FLIR plane vary in response to the change in target aspect angle γ (Figure 2.5) while the spatial relationship of the hotspots remains the same in the three-dimensional target frame. If the plume pogo forcing input is applied, the hotspots do not remain fixed in the target frame, causing the composite image centroid to oscillate along the velocity vector and produce additional perturbations to the hotspot image in the FLIR plane [19].

2.4.2.5 Target Plume Velocity Projection onto the FLIR Plane. The general discrete-time equation that models the target dynamics is given by:

$$\boldsymbol{x}(t_{i+1}) = \Phi(t_{i+1},t_i)\boldsymbol{x}(t_i) + \boldsymbol{B}_d(t_i)\boldsymbol{u}(t_i) + \boldsymbol{G}_d(t_i)\boldsymbol{w}_d(t_i) \tag{2.38}$$

where:

$\Phi(t_{i+1},t_i)$	=	System state transition matrix
$x(t_i)$	=	Discrete-time vector of states of interest
$B_d(t_i)$	=	Discrete-time control input matrix
$u(t_i)$	=	Discrete-time deterministic control input vector
$G_d(t_i)$	=	Discrete-time driving noise input matrix
$w_d(t_i)$	=	Discrete-time, zero-mean, white Gaussian noise process with independent components and covariance Q_d

Based on the geometry shown previously in Figure 2.5, the projection of the target's inertial velocity vector onto the FLIR image plane is the deterministic input vector given by [5]:

$$u(t_i) = [\dot{\alpha}'(t_i)\ \dot{\beta}'(t_i)]^T \tag{2.39}$$

where:

$u(t_i)$	=	True target deterministic input vector
$\dot{\alpha}'(t_i)$	=	Target azimuth rate in the FLIR plane
$\dot{\beta}'(t_i)$	=	Target elevation rate in the FLIR plane

As seen in the inertial frame diagrams of Figure 2.7, the azimuth can be defined as:

$$\alpha(t) = \arctan\left[\frac{z(t)}{x(t)}\right] \tag{2.40}$$

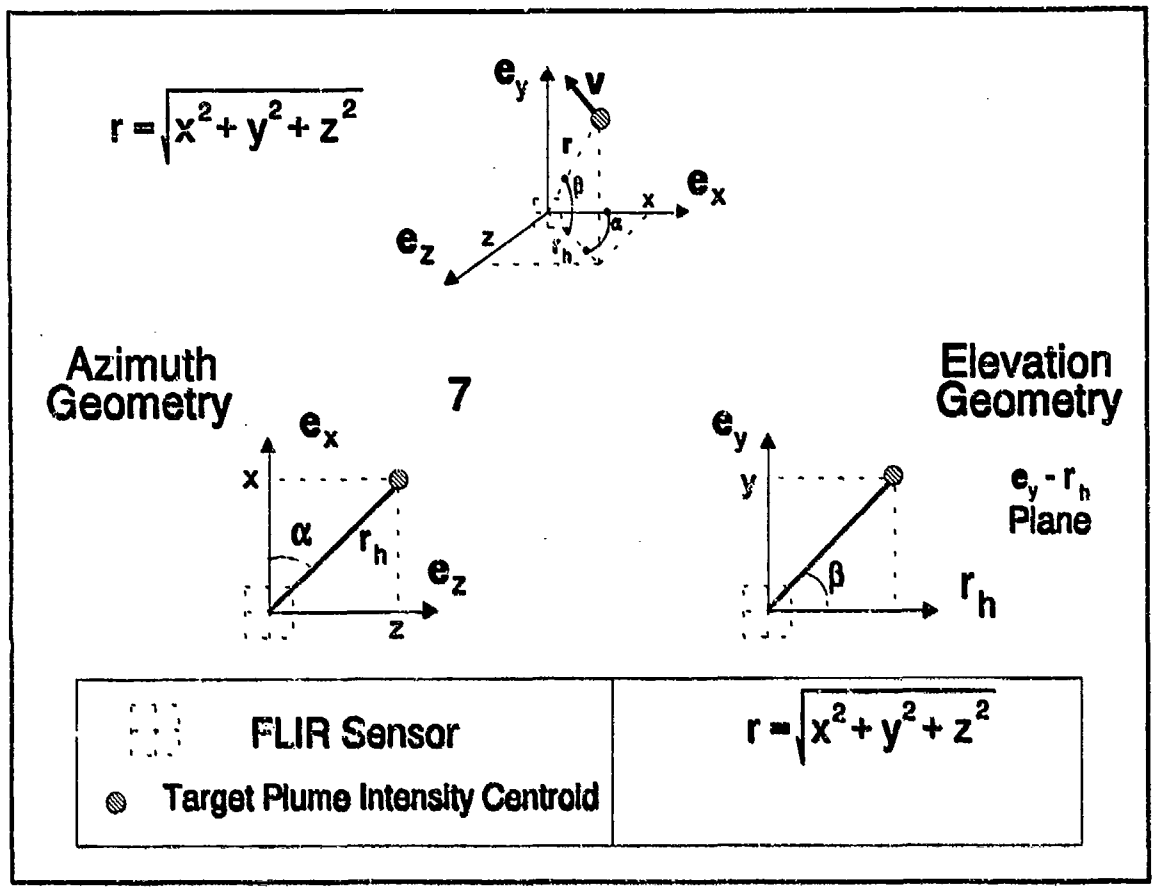


Figure 2.7 Inertial Velocity FLIR Plane Projection Geometry

Taking the time derivative of Equation (2.40) and noting that the sensor-to-target range is large so that $\dot{\alpha}'(t_i) \cong \dot{\alpha}(t_i)$, the azimuth velocity in the FLIR plane is given by:

$$\dot{\alpha}'(t) = \dot{\alpha}(t) = \frac{x(t)v_z(t) - z(t)v_x(t)}{x^2(t) + z^2(t)} \tag{2.41}$$

where:

v_x, v_z = components of the target's inertial velocity in the e_x and e_z directions

Similarly, the elevation velocity in the FLIR plane is given by:

$$\dot{\beta}'(t) = \dot{\beta}(t) - \frac{r_h(t)v_y(t) - y(t)\dot{r}_h(t)}{r^2(t)} \tag{2.42}$$

where:

v_y = Component of the target's inertial velocity in the $\mathbf{e}_y$ direction

r_h = Horizontal projection of the sensor-to-target range, with its time derivative as expressed as:

$$\dot{r}_h(t) = \frac{x(t)v_z(t) + z(t)v_x(t)}{r_h(t)} \tag{2.43}$$

2.4.3 FLIR Sensor Pointing Controller. The filter's propagated estimates of the intensity centroid's position dictate the necessary change in azimuth and elevation that the FLIR sensor should undergo over the next sample period to center the hotspot on the FLIR FOV plane at the next measurement sample time. Ideally, these positional estimates are fed as commands to a pointing controller that physically implements the directional changes within one sample period (1/60 second). The original sample period used in the recent past for the benign trajectory scenarios has been 1/30 second. The newer sample period matches the current hardware and software being developed at the Phillips Laboratory [1].

2.5 *Summary*

This chapter presented the mathematical models of the linear Kalman filter, the MMAE algorithms, and a description of the simulation space. The linear Kalman filter is an optimal estimator and constitutes an elemental filter in the MMAE structure used for the AFIT adaptive tracking system. The MMAE is an adaptive algorithm that optimally combines the estimates of individual Kalman filters that are tuned for a specific parameter value.

III. Truth Model

3.1 Introduction

A "truth model" represents the designer's best mathematical interpretation of the real world dynamics as applicable to the system of interest. Such a model is the product of extensive data analysis, shaping filter design and validation in order to be confident that it adequately represents the real world, since the performance evaluation and systematic design procedure is totally dependent upon this assumption [12]. This truth model description should be as detailed as possible, given the simulation tools available to the designer. A full-scale "truth model" that gives a complete system description of the real world would require an infinite dimensional state model. This would be computationally impossible on a digital computer; therefore, the dominant characteristics of the real world system to be modeled must be captured using a finite number of states. The truth model will be the benchmark that will be used as the real world in the design of the elemental Kalman filters. These elemental Kalman filters are designed by systematically reducing the truth model to form the filter design model, with this resulting filter evaluated against the full-state truth model to ensure that performance specifications are satisfied.

The dynamics of the target intensity centroid's image on the FLIR detector plane are a result of true target motion, atmospheric jitter due to distorted infrared wavefronts, bending/vibration of the optical hardware, and pogo effects of the plume's oscillations. The truth model is composed of the following fourteen states [1,7,22]:

2 target dynamic states (Plume center-of-intensity centroid states)

6 atmospheric states

4 mechanical bending states

2 pogo oscillation states

These dynamics are represented as changes of the image intensity centroid in the FLIR plane, with the centroid components x_c and y_c being measured in pixels from the center of the FOV in the x and y FLIR plane directions. Referring to Figure 3.1, the position of the target image centroid at any one time is given by:

$$x_c = x_t + x_a + x_b + x_p \cos\theta_t \tag{3.1}$$

$$y_c = y_t + y_a + y_b - x_p \sin\theta_t \tag{3.2}$$

where:

x_c, y_c	=	Target image intensity centroid coordinates
x_t, y_t	=	Coordinate deviation due to target dynamics
x_a, y_a	=	Coordinate deviation due to atmospheric jitter

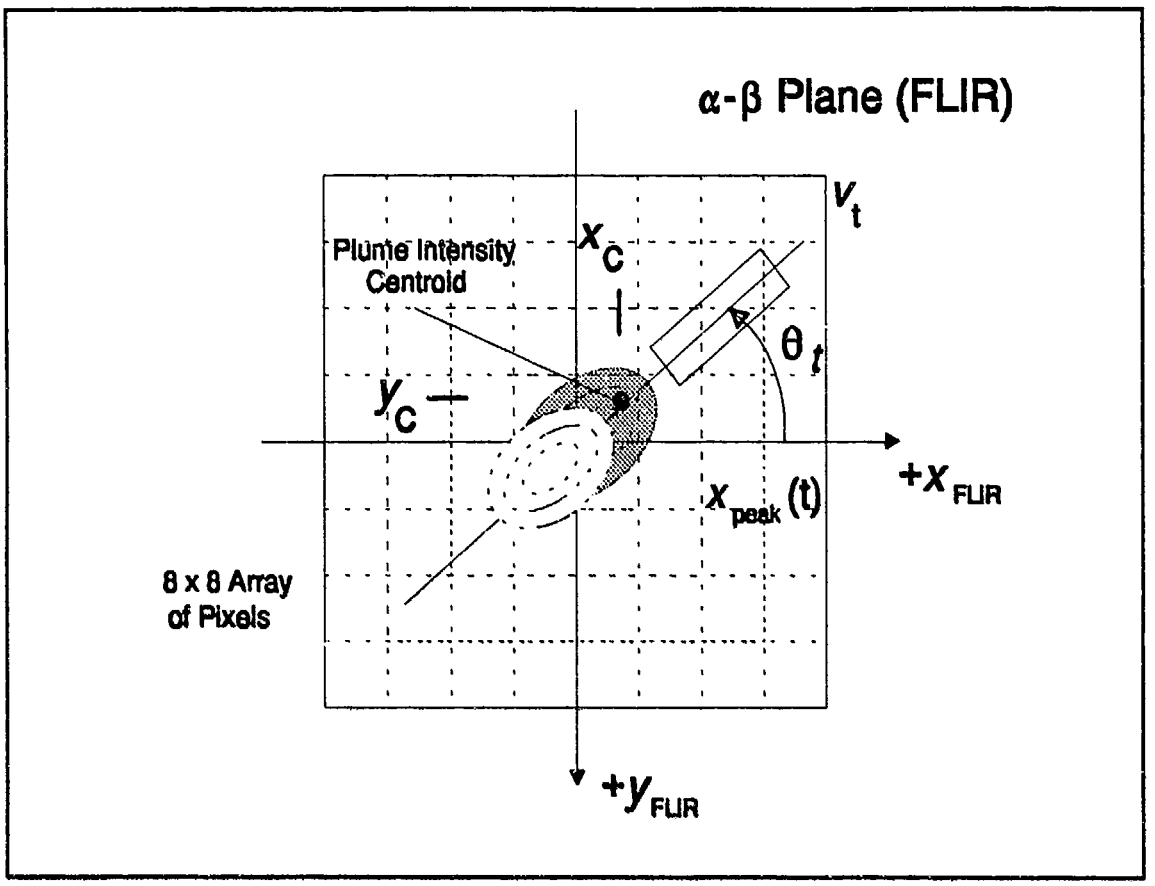


Figure 3.1 Plume Intensity Function Position on FLIR Image Plane

x_b, y_b	=	Coordinate deviation due to bending/vibration of optical hardware
x_p	=	Coordinate deviation due to pogo oscillations along the velocity vector direction
θ_t	=	True target orientation angle

Note the minus sign before the resolved pogo component in Equation (3.2) due to the coordinate definition of the FLIR coordinate frame. The states x_a, x_b, x_t, x_p, y_a, y_b, and y_t comprise the output position states which are extracted from an overall state model

in the form of fourteen coupled scalar stochastic differential equations. The states, x_t and y_t, are each modeled by means of first-order differential equations; x_b, y_b, and x_p are each modeled with second-order differential equations; x_a and y_a are modeled with third-order differential equations. These differential equations, when in state-space format, comprise the dynamics portion of the FLIR tracker truth model.

3.2 *Dynamics Model*

The fourteen-state model state vector is described by a first-order, stochastic differential equation given by:

$$\dot{\boldsymbol{x}}_T(t) = \boldsymbol{F}_T\, \boldsymbol{x}_T(t) + \boldsymbol{B}_T\, \boldsymbol{u}_T(t) + \boldsymbol{G}_T \boldsymbol{w}_T(t) \tag{3.3}$$

where:

$\boldsymbol{F}_T$	=	14 x 14 time-invariant truth model plant matrix
$\boldsymbol{x}_T(t)$	=	14-dimensional truth model state vector
$\boldsymbol{B}_T$	=	14 x 2 time-invariant truth model control distribution matrix
$\boldsymbol{u}_T(t)$	=	2-dimensional input vector
$\boldsymbol{G}_T$	=	14 x 14 noise distribution matrix ($\boldsymbol{G}_T = \mathbf{I}$)
$\boldsymbol{w}_T(t)$	=	14-dimensional, white Gaussian noise process with mean and covariance kernel statistics:

$$E\{w_T(t)\} = 0$$
$$E\{w_T(t)w_T(t+\tau)^T\} = Q_T\ \delta(t) \tag{3.4}$$

To simulate the target dynamics model on a digital computer, the "equivalent discrete-time" solution to Equation (3.3) is given by [16]:

$$x_T(t_{i+1}) = \Phi_T(t_{i+1},t_i)\,x_T(t_i) + B_{Td}\,u_{Td}(t_i) + G_{Td}\,w_{Td}(t_i) \tag{3.5}$$

where the state transition matrix $\Phi_T(t,t_i)$ is the solution to the differential equation:

$$\frac{d\Phi_T(t,t_i)}{dt} = F_T\,\Phi_T(t,t_i) \tag{3.6}$$

with the initial condition: $\Phi_T(t_i,t_i) = \mathbf{I}$, (note that, for constant $\mathbf{F}_T$, $\Phi_T(t,t_i)$ can be expressed as $\Phi_T(t-t_i)$) and:

$x_T(t_i)$	=	14-dimensional discrete-time truth model state vector
B_{Td}	=	14 x 2 discrete-time truth model control distribution matrix
$u_{Td}(t_i)$	=	2-dimensional discrete-time input matrix
G_{Td}	=	14 x 14 noise discrete-time noise distribution matrix, (G_T=I)
$w_{Td}(t)$	=	14-dimensional discrete-time, white Gaussian noise process with mean and covariance statistics:

$$E\{w_{Td}(t_i)\} = 0 \tag{3.7}$$

$$E\{w_{Td}(t_i)w_{Td}^T(t_i)\} = Q_{Td}$$

$$= \int_{t_i}^{t_{i+1}} \Phi_T(t_{i+1} - \tau)G_T Q_T G_T^T \Phi_T^T(t_{i+1} - \tau)d\tau \tag{3.8}$$

where Q_T is defined in Equation (3.4). The discrete-time input distribution matrix B_{Td} is defined as:

$$B_{Td} = \int_{t_i}^{t_{i+1}} \Phi_T(t_{i+1} - \tau)B_T \, d\tau \tag{3.9}$$

Note that this computation assumes $u_T(t)$ is constant over each sample period: $u_T(t)=u_{Td}(t_i)$ for all $t \in [t_i, t_{i+1})$. This input simulates a true constant inertial velocity trajectory for the missile.

The fourteen states of the discrete-time truth model are defined in the x and y coordinate axes of the FLIR plane as:

x_{FLIR}	=	1 target state (Plume intensity centroid), 3 atmospheric states, 2 bending/vibration states
y_{FLIR}	=	1 target state (Plume intensity centroid), 3 atmospheric states, 2 bending/vibration states
v_t^*	=	Two plume pogo states (position and velocity)

*In the FLIR plane except in the direction of the missile velocity vector.

These fourteen states are augmented together into the truth model state vector:

$$x_T = \begin{bmatrix} x_t \\ \cdots \\ x_a \\ \cdots \\ x_b \\ \cdots \\ x_p \end{bmatrix} \tag{3.10}$$

where:

x_t	=	2-dimensional target dynamics state vector
x_a	=	6-dimensional atmospheric state vector
x_b	=	4-dimensional bending/vibration state vector
x_p	=	2-dimensional plume pogo state vector

The 14 x 14 discrete-time truth model state transition matrix, Φ_T, is given by:

$$\Phi_T = \begin{bmatrix} \Phi_t & \vdots & 0 & \vdots & 0 & \vdots & 0 \\ \cdots & & \cdots & & \cdots & & \cdots \\ 0 & \vdots & \Phi_a & \vdots & 0 & \vdots & 0 \\ \cdots & & \cdots & & \cdots & & \cdots \\ 0 & \vdots & 0 & \vdots & \Phi_b & \vdots & 0 \\ \cdots & & \cdots & & \cdots & & \cdots \\ 0 & \vdots & 0 & \vdots & 0 & \vdots & \Phi_p \end{bmatrix} \tag{3.11}$$

where partitions correspond to the dimensionality of the states defined above. The 14 x 2 discrete-time truth model distribution matrix $\boldsymbol{B}_{Td}$ is given by:

$$\boldsymbol{B}_{Td} = \begin{bmatrix} \boldsymbol{B}_{td} \\ \cdots \\ \mathbf{0} \end{bmatrix} \tag{3.12}$$

where $\boldsymbol{B}_{td}$ is a 2 x 2 discrete-time control distribution matrix. The 14-dimensional discrete-time truth model white Gaussian noise process $\boldsymbol{w}_{Td}$ is given by:

$$\mathbf{w}_{Td} = \begin{bmatrix} \mathbf{0} \\ \cdots \\ \mathbf{w}_{ad} \\ \cdots \\ \mathbf{w}_{bd} \\ \cdots \\ \mathbf{w}_{pd} \end{bmatrix} \tag{3.13}$$

where:

$\boldsymbol{w}_{ad}(t_i)$ = 6-dimensional discrete-time, white Gaussian noise related to atmospheric jitter states

$\boldsymbol{w}_{bd}(t_i)$ = 4-dimensional discrete-time, white Gaussian noise related to bending states

$w_{pd}(t_i)$ = 2-dimensional discrete-time, white Gaussian noise related to plume pogo states

The block diagonal form of Equation (3.5), as seen in Equations (3.10) - (3.13), allows the models for target dynamics, atmospheric jitter, bending/vibration, and plume pogo to be presented separately. The following sections discuss each of the discrete state models which form the stochastic discrete-time truth model.

3.2.1 Target Dynamics State Description. As depicted in Figure 3.2, the α-β plane (FLIR image plane) is coincident with the FLIR sensor FOV, and perpendicular to the LOS vector $\mathbf{e}_r$. In the simulation, the 3-dimensional target dynamics are projected onto the FLIR image plane, and the position components of the target's intensity centroid are obtained from the azimuth and elevation displacement angles (α' and β'). Since the target distance is simulated as 2,000 kilometers, small angle approximations are used for measuring the angle displacements in the cartesian coordinate system of the FLIR image plane. These "pseudo" angles, α' and β', are Euler angles referenced from the current LOS vector and measured in microradians. The order of calculation using Euler angles would normally be critical, but large distance, hence small angle approximation ($\sin\alpha' \approx \alpha'$ and $\cos\alpha' \approx 1$) overrides this constraint. Note that the unusual orientation of the $+y_{FLIR}$ axis in Figure 3.2 allows the positive z axis to be in the positive $\mathbf{e}_r$ direction (by the right-hand rule).

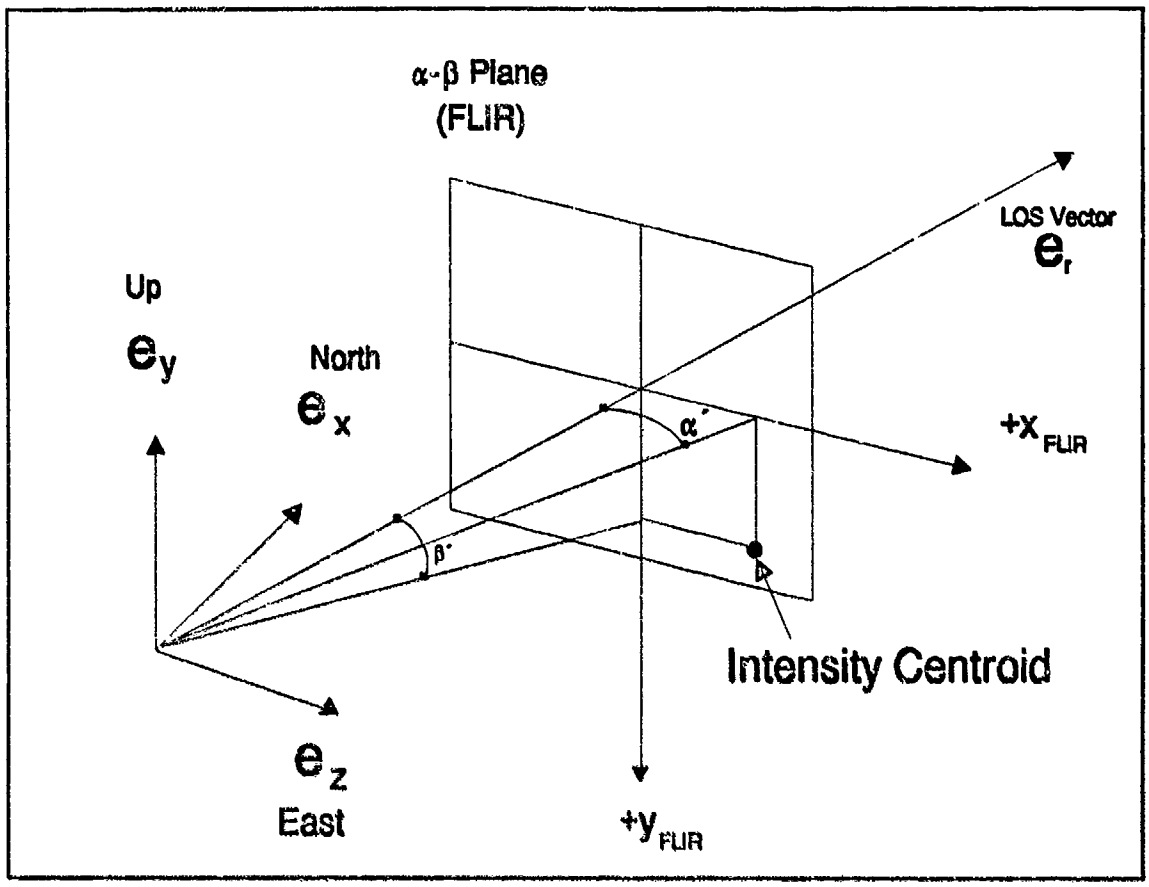


Figure 3.2 Target Centroid Image on α-β Plane with "Pseudo" Angles

The linear translational coordinates, x_c and y_c of Equations (3.1) and (3.2), locate the target intensity function on the FLIR plane and are measured in pixels of displacement from the center of the FLIR FOV. The angular and linear measurements are related by the pixel proportionality constant k_p, which is the angular FOV of a single pixel. Presently, the value of k_p is approximately 15 microradians per pixel for long range targets [3,34].

The derivation of the state space model of the target dynamics assumes that the azimuth and elevation rates ($\dot{\alpha}'$ and $\dot{\beta}'$, respectively) remain essentially constant over each sample period Δt. Then the discrete-time target dynamics model is:

$$x_t(t_{i+1}) = x_t(t_i) + \frac{(\dot{\alpha}')(\Delta t)}{k_p} \tag{3.14}$$

$$y_t(t_{i+1}) = y_t(t_i) - \frac{(\dot{\beta}')(\Delta t)}{k_p} \tag{3.15}$$

Arranging these equations in state space form yields:

$$x_t(t_{i+1}) = \Phi_t(t_{i+1}, t_i)\, x_t(t_i) + B_t\, u_t(t_i) \tag{3.16}$$

$$\begin{bmatrix} x_t(t_{i+1}) \\ \\ y_t(t_{i+1}) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_t(t_i) \\ \\ y_t(t_i) \end{bmatrix} + \begin{bmatrix} \frac{\Delta t}{k_p} & 0 \\ \\ 0 & -\frac{\Delta t}{k_p} \end{bmatrix} \begin{bmatrix} \dot{\alpha}'(t_i) \\ \\ \dot{\beta}'(t_i) \end{bmatrix} \tag{3.17}$$

where:

$\dot{\alpha}'(t_i)$ = $d\alpha'/dt$, measured in microradians/second and constant over the time interval Δt

$\dot{\beta}'(t_i)$ = $d\beta'/dt$, measured in microradians/second and constant over the time interval Δt

Δt = Sample time interval, $t_{i+1} - t_i$ (1/60 second)

k_p = Pixel proportionality constant (15 microradians/pixel)

Using these relationships in block form of the overall truth model, by inspection of Equation (3.11), the upper left block is:

$$\Phi_t = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{3.18}$$

and the upper block of Equation (3.12) is:

$$B_{td} = \begin{bmatrix} \frac{\Delta t}{k_p} & 0 \\ 0 & -\frac{\Delta t}{k_p} \end{bmatrix} \tag{3.19}$$

with the input vector in Equation (3.5) given by:

$$u_{td} = \begin{bmatrix} \dot{\alpha}'(t_i) \\ \dot{\beta}'(t_i) \end{bmatrix} \tag{3.20}$$

The minus sign of the lower right term in Equation (3.19) is due to the difference in the y axis orientation between the inertial coordinate frame and the FLIR coordinate plane.

The two target dynamics states of Equations (3.10) are used to propagate the missile along its trajectory. The input angular velocity values of $\dot{\alpha}'$ and $\dot{\beta}'$ are computed and included in the solution to Equation (3.17) so as to cause the simulation of inertial constant velocity target trajectory, as projected onto the FLIR

image plane. The formulation of these truth model target dynamics states in deterministic state space form has two advantages. First, Equation (3.17) can be substituted back into Equation (3.5) to form a single augmented vector difference equation that defines the truth model. Second, the state space form allows the addition of white (or time-correlated) noise to Equation (3.17), if a stochastic, rather than a deterministic dynamics model, is desired.

3.2.2 Atmospheric Jitter Model. The model for the translational displacement of the intensity function due to atmospheric disturbances is based on a study by The Analytic Sciences Corporation [26]. Physically, atmospheric disturbances cause infrared radiation phase front distortions, which, when brought through the optical system, result in apparent translational shifts of the target. Using power spectral density characteristics, the atmospheric jitter phenomenon in each FLIR plane axis direction can be modeled as the output of a third-order shaping filter driven by white Gaussian noise [26]. The Laplace domain representation of the shaping filter transfer function is given by:

$$\frac{x_a(s)}{w_a(s)} = \frac{K_a \omega_1 \omega_2^2}{(s \quad \omega_1)(s + \omega_2)^2} \tag{3.21}$$

where:

x_a = Output of shaping filter (x_{FLIR} direction)

w_a = Zero-mean, scalar, unit-strength white Gaussian noise

K_a = Gain, adjusted for desired atmospheric jitter rms value

ω_1 = Break frequency, 14.14 radians/second

ω_2 = Double-pole break frequency, 669.5 radians/second

The atmospheric jitter effects can be modeled similarly in the y_{FLIR} direction, where y_a would be the output of an identical shaping filter to that defined in Equation (3.21). The two shaping filters are assumed to be independent of each other and can thus be augmented to form a six-state model. The linear stochastic differential equation that describes the atmospheric jitter is given by:

$$\dot{x}_a(t) = F_a x_a(t) + G_a w_a(t) \tag{3.22}$$

where:

F_a = 6 x 6 time-invariant atmospheric jitter plant matrix

$x_a(t)$ = 6-dimensional atmospheric jitter state vector

G_a = 6 x 2 noise distribution matrix

$w_a(t)$ = 2-dimensional, independent, zero-mean white Gaussian noise with unit strength and independent components described as:

$$E\{w_a(t)\} = 0$$
$$E\{w_a(t)w_a(t+\tau)^T\} = Q_a\,\delta(\tau) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\delta(\tau) \tag{3.23}$$

The six atmospheric states in the state vector correspond to the low frequency pole and the higher frequency double pole in each of the x_{FLIR} and the y_{FLIR} directions. The atmospheric jitter plant matrix is defined in Jordan canonical form as:

$$F_a = \begin{bmatrix} -\omega_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\omega_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & -\omega_2 & 0 & 0 & \\ 0 & 0 & 0 & -\omega_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\omega_2 & 1 \\ 0 & 0 & 0 & 0 & 0 & -\omega_2 \end{bmatrix} \tag{3.24}$$

The noise distribution matrix G_a is :

$$G_a = \begin{bmatrix} \dfrac{K_a\omega_1\omega_2^2}{(\omega_1 - \omega_2)^2} & 0 \\ -\dfrac{K_a\omega_1\omega_2^2}{(\omega_1 - \omega_2)^2} & 0 \\ \dfrac{K_a\omega_1\omega_2^2}{(\omega_1 - \omega_2)} & 0 \\ 0 & \dfrac{K_a\omega_1\omega_2^2}{(\omega_1 - \omega_2)^2} \\ 0 & -\dfrac{K_a\omega_1\omega_2^2}{(\omega_1 - \omega_2)^2} \\ 0 & \dfrac{K_a\omega_1\omega_2^2}{(\omega_1 - \omega_2)} \end{bmatrix} \tag{3.25}$$

The equivalent discrete-time model for Equation (3.22) is of the form:

$$x_a(t_{i+1}) = \Phi_a(t_{i+1},t_i)\,x_a(t_i) + w_{ad}(t_i) \tag{3.26}$$

The augmented six-state state transition matrix derived from the time-invariant plant matrix of Equation (3.24) is [26]:

$$\Phi_a(\Delta t) = \begin{bmatrix} 0 & \Phi_{a22} & \Phi_{a23} & 0 & 0 & 0 \\ 0 & 0 & \Phi_{a33} & 0 & 0 & 0 \\ 0 & 0 & 0 & \Phi_{a44} & 0 & 0 \\ 0 & 0 & 0 & 0 & \Phi_{a55} & \Phi_{a56} \\ 0 & 0 & 0 & 0 & 0 & \Phi_{a66} \end{bmatrix} \tag{3.27}$$

where:

$\Phi_{a11} = \Phi_{a44} = \exp(-\omega_1 \Delta t)$

$\Phi_{a22} = \Phi_{a55} = \exp(-\omega_2 \Delta t)$

$\Phi_{a23} = \Phi_{a56} = \Delta t \exp(-\omega_2 \Delta t)$

$\Phi_{a33} = \Phi_{a66} = \exp(-\omega_2 \Delta t)$

Δt = sample time interval, $(t_{i+1} - t_i)$

The 6-dimensional, zero-mean, discrete-time, white, Gaussian noise $w_{ad}(t_i)$ has statistics defined as:

$$E\{ w_{ad}(t_i)\} = 0$$

$$E\{ w_{ad}(t_i) w_{ad}(t_i)^T \} = Q_{ad}$$

$$= \int_{t_i}^{t_{i+1}} \Phi_a(t_{i+1} - \tau) G_a Q_a G_a^T \Phi_a^T(t_{i+1} - \tau) d\tau \tag{3.28}$$

The individual components of Q_{ad} are not included here due to their length and complexity. The reader may refer to the software for a full description [25].

For the approximated two-state atmospheric jitter model for use in the filter development, only a single-pole shaping filter is used in each direction to produce the approximated Power Spectral Density (PSD). The state space equations are truncated from six to two states with only the first break frequency, ω_1, used in each direction. The plant matrix in Equation (3.24) becomes a 2 x 2 with $-\omega_1$ as the diagonal terms and Equation (3.25) also becomes 2 x 2 with $K_0\omega_1$ on the diagonal.

3.2.3 Bending/Vibration Model. The mechanical bending states were added to the truth model to account for the vibrational effects in the FLIR data that occur when the sensor is mounted on a moving, non-rigid optical platform [12]. Based on tests at the AFWL (now Phillips Laboratory), it was concluded in previous research

[12] that bending effects in both the x_{FLIR} and y_{FLIR} direction can be represented by a second order shaping filter, driven by white Gaussian noise. The Laplace domain transfer function for each of these bending models is:

$$\frac{x_b(s)}{w_b(s)} = \frac{K_b \omega_{nb}^2}{s^2 + 2\zeta_b \omega_{nb} s + \omega_{nb}^2} \tag{3.29}$$

where:

x_b	=	FLIR plane positional offset (x_{FLIR} direction) due to mechanical bending disturbance
w_b	=	Zero-mean, unit strength, white Gaussian noise
K_b	=	Gain adjustment to obtain desired rms bending output; $K_b^2 = 5 \times 10^{-13}$ (Note: K_b^2 is given here because the strength of the bending white noise is expressed in terms of this parameter, rather than K_b)
ζ_b	=	Damping coefficient, equal to 0.15
ω_{nb}	=	Undamped natural frequency for bending, ($\omega=\pi$ rad/sec)

The FLIR plane positional offset in the y_{FLIR} direction, y_b, is identically modeled with the shaping filter defined in Equation (3.29). The two shaping filters are assumed to be independent of each other and can thus be augmented to form a four-state mode. The linear stochastic differential equation that describes the bending/vibration is given by:

$$\dot{x}_b(t) = F_b x_b(t) + G_b w_b(t) \tag{3.30}$$

where:

F_b	=	4 x 4 time-invariant bending plant matrix
$x_b(t)$	=	4-dimensional bending state vector
G_b	=	4 x 2 noise distribution matrix
$w_b(t)$	=	2-dimensional, white Gaussian noise process with unit strength components that are independent of each other:

$$E\{w_b(t)\} = 0$$
$$E\{w_b(t)w_b(t+\tau)^T\} = Q_b\,\delta(\tau) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \delta(\tau) \tag{3.31}$$

The bending/vibration plant matrix is defined as:

$$F_b = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\omega_{nb}^2 & -2\zeta_b\omega_{nb} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\omega_{nb}^2 & -2\zeta_b\omega_{nb} \end{bmatrix} \tag{3.32}$$

The noise distribution matrix G_b is:

$$G_b = \begin{bmatrix} 0 & 0 \\ \omega_{nb}^2 K_b & 0 \\ 0 & 0 \\ 0 & \omega_{nb}^2 K_b \end{bmatrix} \tag{3.33}$$

The equivalent discrete-time model for Equation (3.30) is of the form:

$$x_b(t_{i+1}) = \Phi_b(t_{i+1},t_i)x_b(t_i) + w_{bd}(t_i) \tag{3.34}$$

where:

$$\Phi_b(\Delta t) = \begin{bmatrix} \Phi_{b11} & \Phi_{b12} & 0 & 0 \\ \Phi_{b21} & \Phi_{b22} & 0 & 0 \\ 0 & 0 & \Phi_{b33} & \Phi_{b34} \\ 0 & 0 & \Phi_{b43} & \Phi_{b44} \end{bmatrix} \tag{3.35}$$

and:

$\Phi_{b11} = \Phi_{b33} = \exp(-\sigma_b\Delta t)[\cos(\omega_b\Delta t + (\sigma_b/\omega_b)\sin(\omega_b\Delta t)]$

$\Phi_{b12} = \Phi_{b34} = \exp(-\sigma_b\Delta t)[(1/\omega_b)\sin(\omega_b\Delta t)]$

$\Phi_{b21} = \Phi_{b43} = -\exp(-\sigma_b\Delta t)[(\omega_b)\sin(\omega_b\Delta t)]$

$\Phi_{b22} = \Phi_{b44} = \exp(-\sigma_b\Delta t)[\cos(\omega_b\Delta t) - (\sigma_b/\omega_b)\sin(\omega_b\Delta t)]$

Δt = Sample time interval, $(t_{i+1}-t_i)$

σ_b = Real part of the root of the characteristic equation in Equation (3.29), (σ_b = 0.47124 second^{-1})

ω_b = Imaginary part of the root of the characteristic equation in Equation (3.29), (ω_b = 3.10605 radians/second)

The 4-dimensional, discrete-time, white Gaussian noise process vector $w_{bd}(t_i)$ has mean and covariance statistics:

$$E\{ w_{bd}(t_i)\} = 0$$

$$E\{ w_{bd}(t_i) w_{bd}(t_i)^T \} = Q_{bd}$$

$$= \int_{t_i}^{t_{i+1}} \Phi_b(t_{i+1} - \tau)\, G_b Q_b G_b^T\, \Phi_b^T\, (t_{i+1} - \tau) d\tau \tag{3.36}$$

3.2.4 Plume Pogo Model. To account for the oscillatory nature of a typical missile plume in the boost phase, a plume pogo model was developed [34]. A second-order Gauss-Markov model was generated using physical insight, and visual observation of the pogo phenomenon. The model allows for the study of the amplitude and frequency characteristics of the oscillatory nature of the plume, and of the effect upon tracking a missile using a Kalman filter.

The transfer function of the plume pogo model is described in the Laplace domain as:

$$\frac{x_p(s)}{w_p(s)} = \frac{K_p \omega_{np}^2}{s^2 + 2\zeta_p \omega_{np} s + \omega_{np}^2} \tag{3.37}$$

where:

x_p = Plume pogo shaping filter output along the direction of the velocity vector

w_p = Zero-mean, unit strength, white Gaussian noise

ζ_p = Assumed damping coefficient, ($\zeta = 0.05$)

ω_{np} = Nominal undamped natural frequency for pogo; assumed range is 0.1-10 Hertz, with a nominal value of 1.0 Hertz

K_p = Gain adjustment to obtain desired rms pogo amplitude determined by [40]:

$$K_p = 2\sigma_p \sqrt{\frac{\zeta_p}{\omega_{np}}} \tag{3.38}$$

where:

σ_p = Desired rms pogo along the velocity vector

The linear stochastic differential equation that describes the plume pogo is given in state space form as:

$$\dot{x}_p(t) = \begin{bmatrix} 0 & 1 \\ -\omega_{np}^2 & -2\zeta_p\omega_{np} \end{bmatrix} x_p(t) + \begin{bmatrix} 0 \\ K_p\omega_{np}^2 \end{bmatrix} w_p(t) \tag{3.39}$$

where:

$x_p(t)$ = 2-dimensional pogo state vector composed of pogo position and velocity states

$w_p(t)$ = 1-dimensional zero-mean, white Gaussian noise with statistics:

$$\begin{aligned} E\{w_p(t)\} &= 0 \\ E\{w_p(t)w_p(t+\tau)\} &= Q_p\delta(t-\tau); \quad Q_p = 1 \end{aligned} \tag{3.40}$$

The equivalent discrete-time model for Equation (3.39) is of the form:

$$x_p(t_{i+1}) = \Phi_p(t_{i+1},t_i)\,x_p(t_i) + w_{pd}(t_i) \tag{3.41}$$

$$x_p(t_{i+1}) = \begin{bmatrix} \Phi_{p11}(\Delta t) & \Phi_{p12}(\Delta t) \\ \Phi_{p21}(\Delta t) & \Phi_{p22}(\Delta t) \end{bmatrix} x_p(t_i) + w_{pd}(t_i) \tag{3.42}$$

where:

$$\begin{aligned}
\Phi_{p11}(\Delta t) &= \frac{1}{\sqrt{1-\zeta_p^2}}\exp(-\zeta_p\omega_{np}\Delta t)\sin\left[\omega_{np}\sqrt{1-\zeta_p^2}\Delta t + \arctan\left(\frac{\sqrt{1-\zeta_p^2}}{\zeta_p}\right)\right] \\
\Phi_{p12}(\Delta t) &= \frac{1}{\omega_{np}\sqrt{1-\zeta_p^2}}\exp(-\zeta_p\omega_{np}\Delta t)\sin\left(\omega_{np}\sqrt{1-\zeta_p^2}\Delta t\right) \\
\Phi_{p21}(\Delta t) &= \frac{-\omega_{np}}{\sqrt{1-\zeta_p^2}}\exp(-\zeta_p\omega_{np}\Delta t)\sin\left(\omega_{np}\sqrt{1-\zeta_p^2}\Delta t\right) \\
\Phi_{p22}(\Delta t) &= \frac{1}{\sqrt{1-\zeta_p^2}}\exp(-\zeta_p\omega_{np}\Delta t)\sin\left[\omega_{np}\sqrt{1-\zeta_p^2}\Delta t + \arctan\left(\frac{\sqrt{1-\zeta_p^2}}{-\zeta_p}\right) + \pi\right]
\end{aligned} \tag{3.43}$$

The 2-dimensional, discrete-time, white Gaussian noise process $w_{pd}(t_i)$ has mean and covariance statistics:

$$E\{w_{pd}(t_i)\} = 0$$

$$E\{w_{pd}(t_i)w_{pd}(t_i)^T\} = Q_{pd}$$

$$= \int_{t_i}^{t_{i+1}} \Phi_p(t_{i+1} - \tau)G_p Q_p G_p^T \Phi_p^T (t_{i+1} - \tau)d\tau \tag{3.44}$$

The 2-dimensional pogo state vector defines the position of the plume image intensity centroid relative to the equilibrium point of oscillation, and its velocity component due to the pogo phenomenon along the longitudinal axis of the missile. For the simulation, it is assumed that the velocity vector lies coincident with the longitudinal axis of the hardbody. As shown in Figure 1.6, the plume oscillates about an equilibrium point also located on the longitudinal axis. This equilibrium point is defined by the initial positions of the two intensity functions in the target coordinate frame (to be discussed Section *3.3.1*), and remains at a constant distance from the hardbody center-of-mass throughout the simulation (the spatial relationship of the intensity functions can be seen in Figure 3.3 in Section *3.3.1*). The crescent-shaped plume represents one of many equal-intensity contour lines of the actual plume. The angle of attack and sideslip angle of the missile are also assumed negligible, and have zero values for the simulation [34].

3.3 *Measurement Models*

It is *crucial* for a simulation tool to have accurate models concerning the measurement updates to be used. These models must be products of experience in using the actual physical devices, tempered with the reality in dealing with computer simulations similar to this research thesis. There are two different measurement models that are used in this research. The first model is the measurement update from the FLIR 8 x 8 tracking window receiving radiation from the plume intensity centroid. This update comes from comparing (in an enhanced correlation algorithm) the truth-model-generated measurement with the template that is generated from the state estimate of the previous sample period. This correlator then produces a "measurement" that goes to the Kalman filter bank as a 2-dimensional vector, x_c and y_c, as the offset necessary to provide the highest correlation between the data and the template in the α-β (FLIR) plane coordinates.

The second measurement is generated with returns from the Low-Energy Laser (LEL) that uses the updated estimates from the FLIR measurement to "paint" the hardbody target using the filter generated plume centroid coordinates as a starting point. The LEL illuminates along the filter-estimated velocity vector to locate the two endpoints of the missile hardbody. These endpoint coordinates are then used to form a noise corrupted "center-of-mass" one-dimensional LEL measurement update. Section *3.3.1* describes the FLIR update while Sections *3.3.2* and *3.3.3* detail the LEL

update along with the Doppler effect in the returned signal that is used to determine where the endpoints of the hardbody are located.

3.3.1 Flir Model. The FLIR sensor model is composed of an 8 x 8 pixel array "tracking window" extracted from the total array of 300 x 500 pixels. The missile plume is projected onto the FLIR focal plane, with its characteristic crescent-shaped intensity function formed as the difference of two bivariate Gaussian intensity functions (the difference is clipped to zero if negative), as shown in Figure 3.3. This

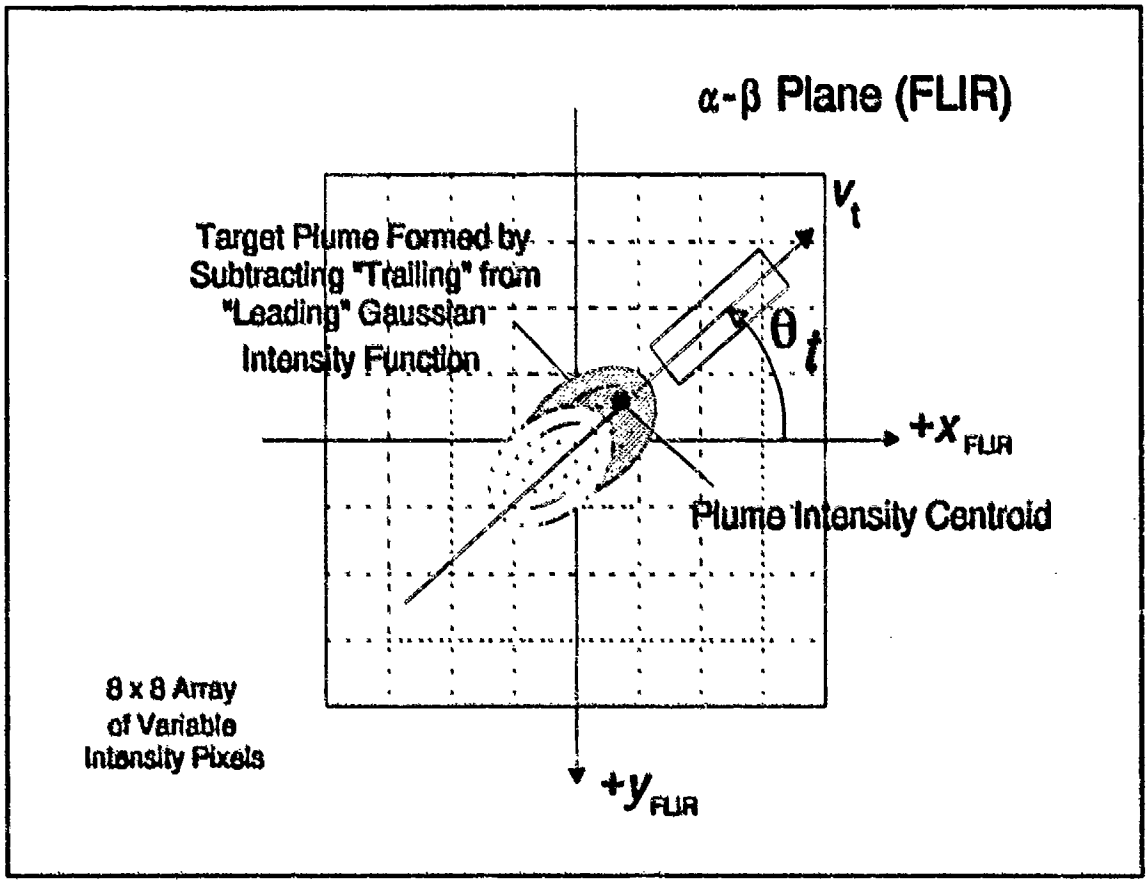


Figure 3.3 Composite Plume Intensity Function on FLIR Plane

model depends upon knowledge of several parameters: the size of the major and minor axes of the elliptical contours of each bivariate Gaussian function, and the orientation of the principal axes in the FLIR image plane (the major axis of each ellipse points along the velocity vector in the FLIR plane). The target intensity function so obtained is corrupted by spatially correlated and temporally uncorrelated background noise and spatially and temporally uncorrelated internal FLIR noise, according to models of actual data taken from a FLIR sensor looking at various backgrounds [32].

For each pixel in the FLIR FOV (the 8 x 8 array "tracking window"), the target's intensity function, correlated background noise, and FLIR internal noise are added together to produce an intensity measurement. For the 8 rows and 8 columns of the FOV, the intensity measurement corresponding to the pixel in the jth row and k^{th} column at sampling time t_i is given by:

$$z_{jk}(t_i) = \frac{1}{A_p} \int_{pixel_{jk}} \{ I_1 [x, y, x_{peak1}(t_i), y_{peak1}(t_i)] - I_2 [x, y, x_{peak2}(t_i), y_{peak2}(t_i)] \} \, dx dy + n_{jk}(t_i) + b_{jk}(t_i) \tag{3.45}$$

where:

$z_{jk}(t_i)$ = Output of pixel in the jth row and k^{th} column

A_p = Area of one pixel

I_1, I_2	=	Intensity function of first and second Gaussian intensity function, respectively, of Figure 3.3
x, y	=	Coordinates of any point within pixel jk
x_{peak1}, y_{peak1}	=	Coordinates of maximum point of first Gaussian intensity function
x_{peak2}, y_{peak2}	=	Coordinates of maximum point of second Gaussian intensity function
$n_{jk}(t_i)$	=	Effect of internal FLIR sensor noise on jk^{th} pixel
$b_{jk}(t_i)$	=	Effect on spatially correlated background noise on jk^{th} pixel

The sensor error, $n_{jk}(t_i)$, is the result of thermal noise and dark current in the IR detectors (pixels). This error is assumed to be both temporally and spatially uncorrelated [34].

The background noise, $b_{jk}(t)$, was observed in the FLIR data by AFWL personnel during a tracking operation [8]. It is represented as a spatially correlated noise with radial symmetry, with a correlation that decays exponentially. Harnly and Jensen [8] concluded that spatial correlation can be depicted as a correlation distance of approximately two pixels in the FLIR plane, and simulated by maintaining non-zero correlation coefficients between each pixel and its two closest neighbors symmetrically in all directions. In that two-pixel distance, the correlation decays exponentially to one-tenth of its peak value.

The generation of spatially correlated white Gaussian noises is accomplished by allowing non-zero cross correlations between the measurement noises, $b_{jk}(t_i)$, associated with each of the 64 pixels from the 8 x 8 pixel FLIR FOV. The correlated measurement noise in Equation (3.45) is given as:

$\boldsymbol{b}(t_i)$ = 64-dimensional vector of spatially correlated noise with statistics:

$$E\{\boldsymbol{b}(t_i)\} = \boldsymbol{0}$$
$$E\{\boldsymbol{b}(t_i)\boldsymbol{b}(t_j)^T\} = \boldsymbol{R}\,\delta_{ij} \tag{3.46}$$

where $\boldsymbol{R}$ is a 64 x 64 measurement noise covariance matrix and δ_{ij} is the Kronecker delta, defined to assume the value of one if $i = j$ and zero otherwise. This matrix describes the spatial correlation between pixels, and is given by [13]:

$$\boldsymbol{R} = \sigma_R^2 \begin{bmatrix} 1 & r_{1,2} & r_{1,3} & \cdots & r_{1,64} \\ r_{2,1} & 1 & r_{2,3} & \cdots & r_{2,64} \\ r_{3,1} & r_{3,2} & 1 & \cdots & r_{3,64} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ r_{64,1} & r_{64,2} & r_{64,3} & \cdots & 1 \end{bmatrix} \tag{3.47}$$

where σ_R^2 is the variance of each scalar noise and the correlation coefficients $r_{j,k}$ are evaluated to reflect the radially symmetric, exponentially decaying pattern. The spatially correlated background noise $\boldsymbol{b}(t_i)$ is simulated as:

$$\boldsymbol{b}(t_i) = \sqrt[c]{\boldsymbol{R}}\,\boldsymbol{b}'(t_i) \tag{3.48}$$

where:

$\sqrt[c]{}$ = Cholesky square root

$b'(t_i)$ = 64-dimensional vector of readily simulated discrete, independent white Gaussian noise with statistics:

$$E\{b'(t_i)\} = 0$$
$$E\{b'(t_i)b'^T(t_j)\} = I\,\delta_{ij} \tag{3.49}$$

3.3.2 Low-Energy Laser Speckle Reflection Model. The low-energy speckle reflection model evolved through the work accomplished by Eden and Evans [3,4]. The model makes no attempt to simulate the detailed physical phenomena associated with the speckle return of the reflected laser from the plume or hardbody. Rather, the model simulates the reflectivity information from the hardbody speckle return which would be derived by speckle detection circuitry.

The low-energy laser speckle reflection model simulates a linear measurement to the Kalman filter for estimating the offset distance from the plume intensity centroid to the hardbody center-of-mass along the vehicle's FLIR image plane velocity vector. The first attempt to model the laser speckle return consisted of the hardbody represented as a rectangle with a binary-valued reflectivity function, which provided a binary indication of the hardbody whenever successful interception by the laser beam occurred [3]. With this model, speckle reflection information was equally obtained over the entire vehicle. This was followed by an enhanced, 3-dimensional

reflectivity model which accounted for the realistic distribution of the laser speckle return according to the curvature and aspect angle of the hardbody [4]. The 3-dimensional model is employed for this research since the Doppler return is also a function of reflectivity [9,38,39,44]. The following subsections discuss the development of Evans' 3-dimensional hardbody reflectivity model and introduce the plume reflectance model.

3.3.2.1 The Hardbody Reflectivity Model. The 3-dimensional reflectivity model was developed by Evans [4] based upon his analysis of empirical data obtained from the 6585th Test Group, Holloman AFB, New Mexico [5]. The data illustrates the return power (expressed in decibels-square meters) as a function of radar cross section (RCS) from a 20 x 249 inch cylinder with hemispherical endcaps as it was rotated longitudinally in the plane of the radar source. RCS is defined as the projected area of a metal sphere which would return the same echo signal as the target, had the sphere been substituted for the target [38]. The data showed peak values at 90° and 270°, where the cylinder was orthogonal to the line of sight, and sharp dropoffs in reflection as the angle deviated from the orthogonal condition. The reflectivity model, shown relative to the FLIR image plane in Figure 3.4, modifies the previous rectangular model to include 29 discrete weighted line segments in the true velocity vector direction along the length of the model. Two functions define the hardbody reflectivity model: the cross sectional function and the longitudinal function.

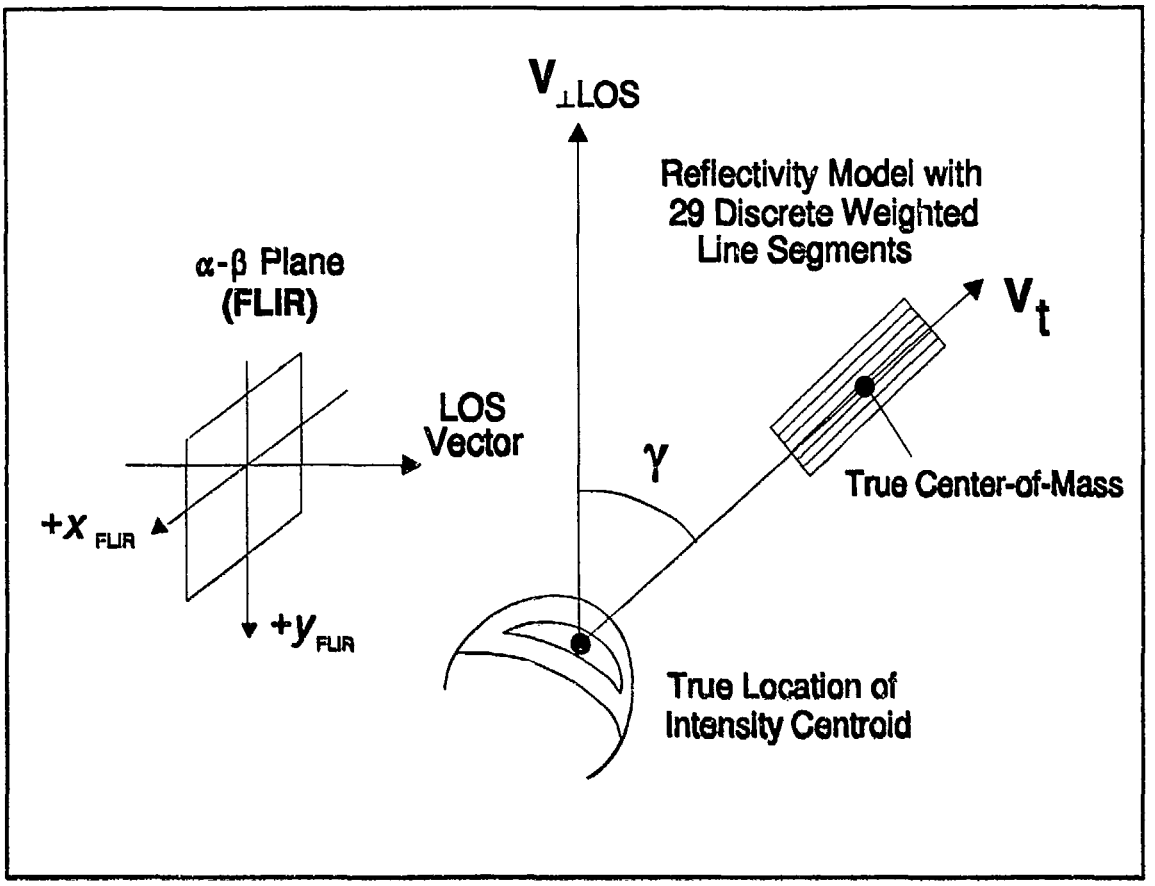


Figure 3.4 3-d Hardbody Reflectivity Model Relative to FLIR Image Plane

Each discrete weighted line represents a cross-sectional reflectivity function which duplicates the empirical data from Holloman. The reflectivity function models the curvature by defining the strength of the reflected signal at each discrete line, where the amplitude of the reflected signal is highest along the missile centerline and discretely tapers towards the hardbody sides in 0.1 meter increments. The discrete implementation of the cross-sectional reflectivity function for the simulation is shown in Figure 3.5. Note the peak reflection of the cross-sectional reflectivity function's center is represented by an arbitrary value of 50 units of reflection magnitude [4].

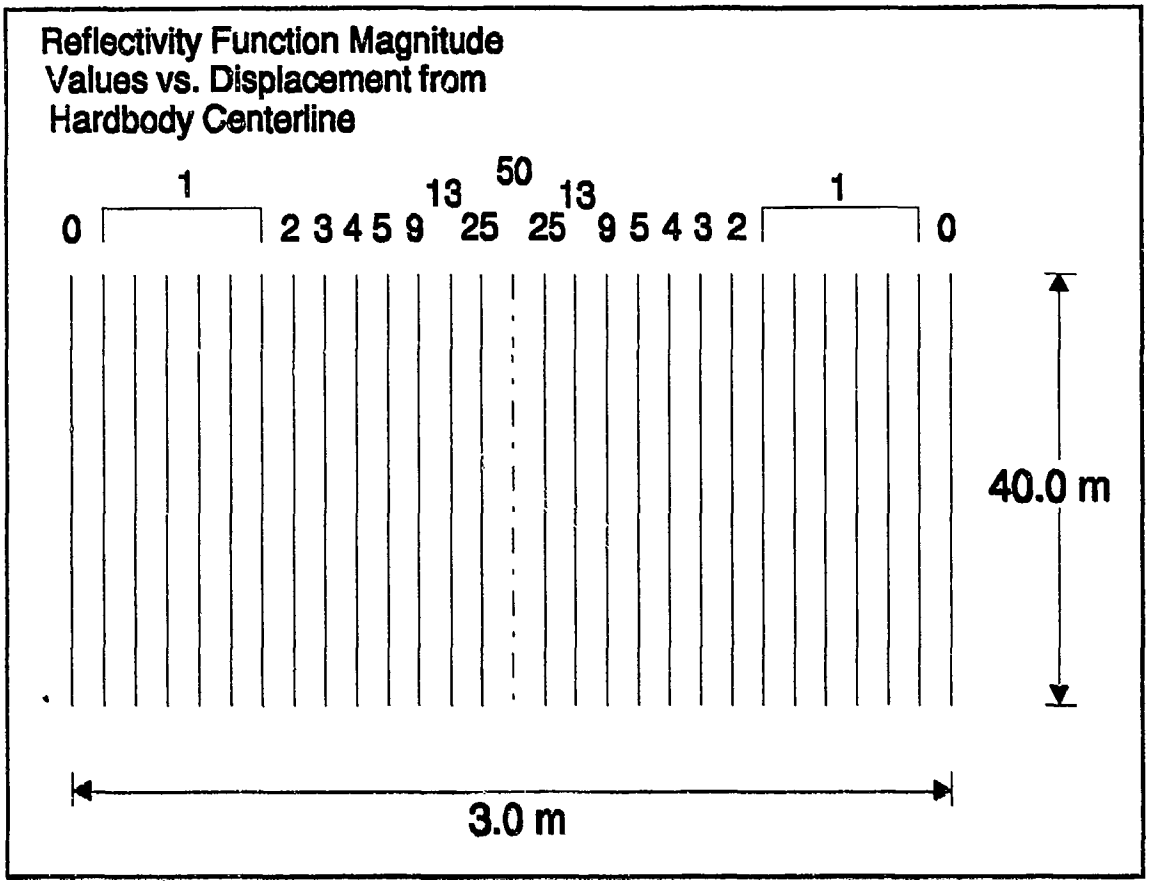


Figure 3.5 Discrete Implementation of Cross-Sectional Reflectivity Function [9]

The remaining line segments are scaled accordingly to match the empirical data. The reflectivity function also yields zero or significantly reduced reflection for those portions of the original rectangle far from the missile centerline, so the effective area of the hardbody is less than that of the binary model.

Note that, in Figure 3.4, v_t is the true velocity, not necessarily in the FLIR image plane, and $v_{\perp LOS}$ is its component in the FLIR image plane. The angle γ, defined as the angle between the inertial velocity vector and the FLIR plane, is

utilized by the longitudinal reflectivity function to provide a scaling factor of the total reflection function if the missile centerline is oriented other than normal to the FLIR plane. Similar to the cross-sectional reflectivity function, the longitudinal function assigns a scaling factor to the reflected signal based upon the angular aspect of the target velocity.

Another factor in determining the received speckle reflection is the sensitivity level of the low-energy laser sensor. This sensitivity is represented in the simulation as a threshold limit below which the low-energy laser sensor cannot detect the reflection return. To illustrate the function of the sensor sensitivity factor, consider the hardbody at an aspect angle γ relative to the FLIR image plane. In this orientation, the maximum amount of reflection is obtained in the simulation by multiplying the peak reflection value (50 units of magnitude) by an appropriate scaling factor [4]. The sensitivity threshold function $\mu(\cdot)$ is defined as a function of a threshold reflection magnitude m_r. If a reflection magnitude is less than m_r, the reflective output is clipped to zero (see defining equation for $\mu(\cdot)$ in the next paragraph). Therefore, $\mu(\cdot)$ represents the sensor's ability to discern a target's return signal [4].

The total reflectivity function is given by [4]:

$$R_T = \sum_{i=1}^{n} \mu \left[A_i F(\gamma)\right] \tag{3.50}$$

where:

R_T = Total reflectivity received by the low energy sensor

n = Number of line segments crossed by laser scan

$\mu(\cdot)$ = Sensitivity threshold function of low-energy sensor:

$$\mu(x) = \begin{cases} x & if\, x \geq m_r \\ 0 & if\, x < m_r \end{cases}$$

A_i = Cross-sectional reflectivity function's reflection amplitude of the ith discrete line segment

$F(\gamma)$ = Longitudinal reflectivity function, where γ is the angle between target $\mathbf{v}_t$ and the α-β plane

As the hardbody traverses along its trajectory in 3-dimensional inertial space, the projection of its motion onto the 2-dimensional FLIR image plane generates the corresponding propagation of the first two states in the truth model. Similarly, to simulate the center-of-mass measurements in terms of FLIR plane variables, the hardbody models are also projected onto the 2-dimensional FLIR plane. Referring to Figure 3.6, the geometry for projection is described by:

$$ML_{FLIR} = ML_{Actual} \cos\gamma \tag{3.51}$$

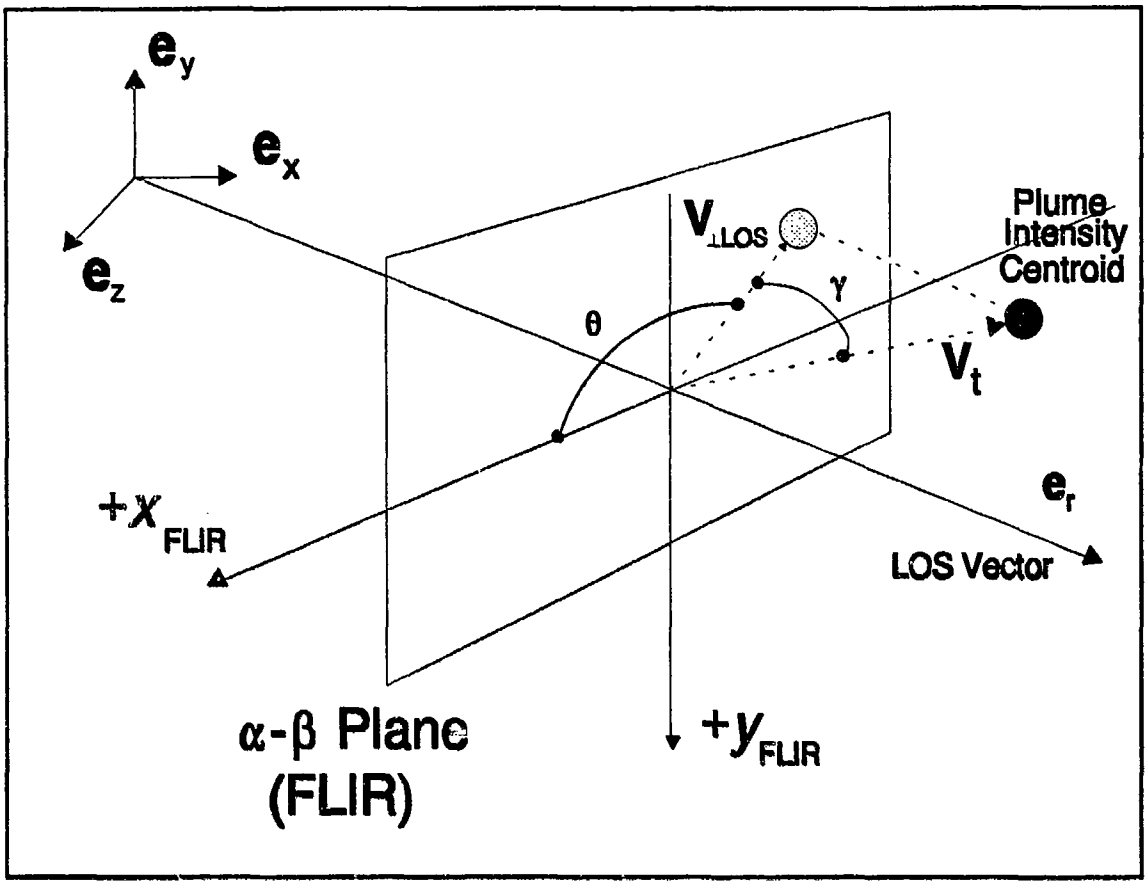


Figure 3.6 Projection Geometry onto FLIR Image Plane

where:

ML_{FLIR} = FLIR plane projection of missile length

ML_{Actual} = True missile length in pixels

γ = Angle between $\mathbf{v}_t$ (velocity vector of the target) and the FLIR plane

Similarly, since the hardbody longitudinal axis is assumed to be aligned with the velocity vector (along which the offset is aligned), the offset between the hardbody

and the plume is scaled by the same factor when projected onto the FLIR plane. Once the projection is accomplished, the hardbody is located on the FLIR plane by offsetting the hardbody's center (midway between the projected endpoints) from the truth model intensity centroid along the truth model velocity vector, $\mathbf{V}_{\perp LOS}$, by [(*Offset distance*$_{Actual}$)cosγ].

The subtended arc of the low power laser beam is simulated as a rectangle with the smaller side represented as the finite width of a dithered laser beam after it has traveled 2,000 kilometers. Shown in Figure 3.7 are the ideal conditions for the laser scan. (Generally, the filter estimates of the intensity centroid position, the orientation angle, and the velocity vector are not equal to the truth model values.) One end of the long centerline of laser scan rectangle is located at the estimated intensity centroid, positioned at the center of the FLIR FOV. The other end of the laser scan rectangle is taken as three times the truth model offset distance between the intensity centroid and the hardbody center-of-mass (3 x 87.5 = 262.5 meters or 8.75 pixels) to ensure the laser scan is long enough to intercept the hardbody, despite the effects of "pogo." The second endpoint of the laser rectangle along its centerline is given as:

$$\begin{aligned} x_p &= x_c + L\cos\theta_f \\ y_p &= y_c - L\sin\theta_f \end{aligned} \tag{3.52}$$

where:

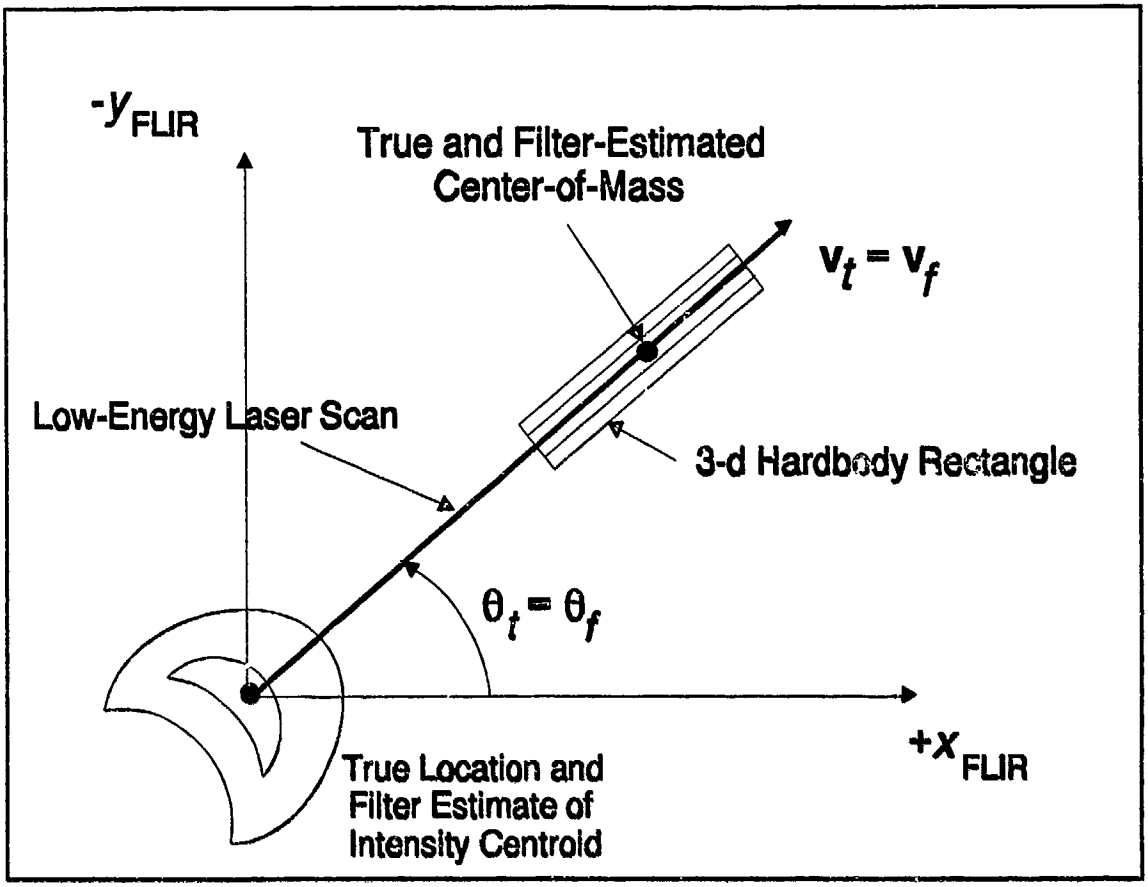


Figure 3.7 Ideal Low-Energy Laser Scan

x_p, y_p	=	FLIR plane coordinates of the second end of the centerline of the laser rectangle
x_c, y_c	=	FLIR plane intensity centroid coordinates
L	=	Length of the laser rectangle
θ_f	=	Six-state (FLIR) filter estimate of velocity orientation angle

As mentioned earlier, the FLIR filter's imprecise centering of the intensity centroid caused inadequate hardbody illumination rates by the laser scan in the

original research by Eden [3]. (The estimated velocity vector, and thus the estimated orientation angle, θ_f, were estimated precisely, however.) As a result, an *ad hoc* sweep routine was developed, shown in Figure 3.8, that offsets the initial laser scan clockwise from the estimated velocity vector. The laser scans are swept counterclockwise in order to assure illumination of the entire body. Evans found that, without pogo, a 30° offset was required, and 35° with pogo applied [4].

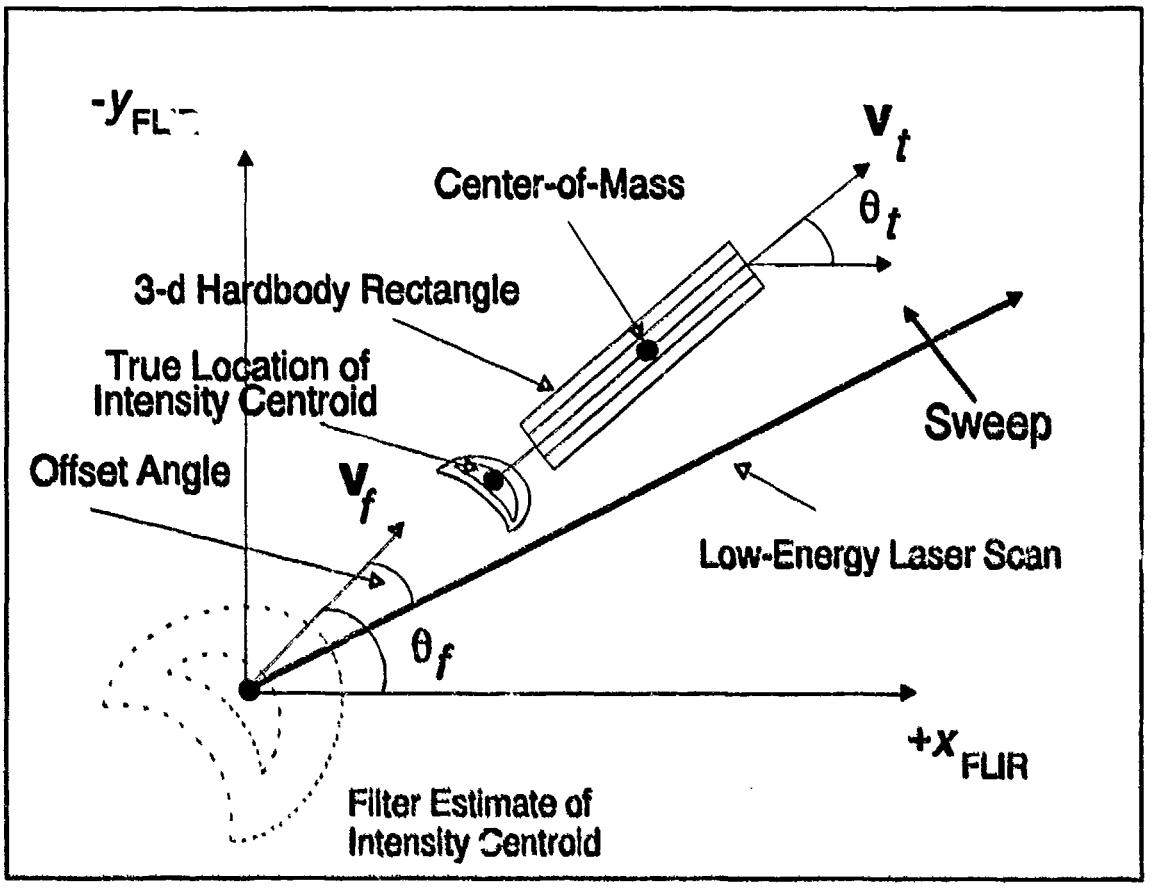


Figure 3.8 Sweep Techniques of Laser Scan

3.3.2.2 Plume Reflectance Model. Prior to Herrera's research [9], the concept of illuminating the missile hardbody with a low-energy laser and analyzing the speckle return (also called backscatter radiation) was predicated upon the assumption that the missile plume would not possess a speckle return similar to the hardbody's, when illuminated by a low-energy laser. The laser scan travels along the intensity centroid's velocity vector until a speckle return is received, signifying the start of the metallic hardbody. The scan continues along the hardbody until no backscatter exists, signaling the end of the hardbody, and thus information is provided to calculate the center-of-mass. However, experimental data confirmed the presence of reflectance from solid-propellant rocket motors [29] which significantly alters the previous conception.

Experimental programs at the Arnold Engineering and Development Center (AEDC), in Tennessee, have observed and measured laser backscatter radiation from the exhaust plume of a solid-propellant rocket motor [29]. The measurements of the plume's backscatter radiation were found to be on the same order of magnitude and comparable to that of a hardbody [1], due to aluminum particles and other substances in the plume. During the STARLAB flight experiment, which collected plume data under actual flight conditions, a rocket booster and its exhaust plume were "painted" by a low energy laser. Video recordings of the flight experiment showed the randomized appearance and low-frequency oscillation of the plume's reflectance [1]. The existence of plume reflectance creates an ambiguity that impedes the precision

tracking necessary to define the plume/hardbody interface. The plume reflectance causes a bias in the estimated hardbody location, biased longitudinally toward the plume.

Since for this thesis, the Doppler measurement model was utilized instead of speckle, the offset measurements from the LEL were assumed to be unbiased (see next section). However, this section has been included in the thesis description for continuity and, since the bias effect is still implemented in the software if the speckle return model is used, it is retained as reference for future researchers.

The purpose of the plume reflectance model is to simulate the presence of plume backscatter radiation and its effect upon the offset measurement. Figure 3.9 depicts the reflectance from both the plume and hardbody, as observed in the STARLAB flight experiment. From the viewpoint of the speckle return sensor, the plume reflectance has the effect of elongating the apparent missile hardbody in the direction of the plume. The plume reflectance model simulates the hardbody elongation by applying a bias to the offset measurement in the direction of the elongation, defined as in the opposite direction of the estimated velocity vector. In the simulation, the model first receives the offset measurement as determined by the low-energy speckle reflection model. The biased measurement, $x_{offset_{bias}}$, is formed by converting the bias into pixels, projecting it onto the FLIR plane, and subtracting it from the original offset measurement. The biased offset measurement is then

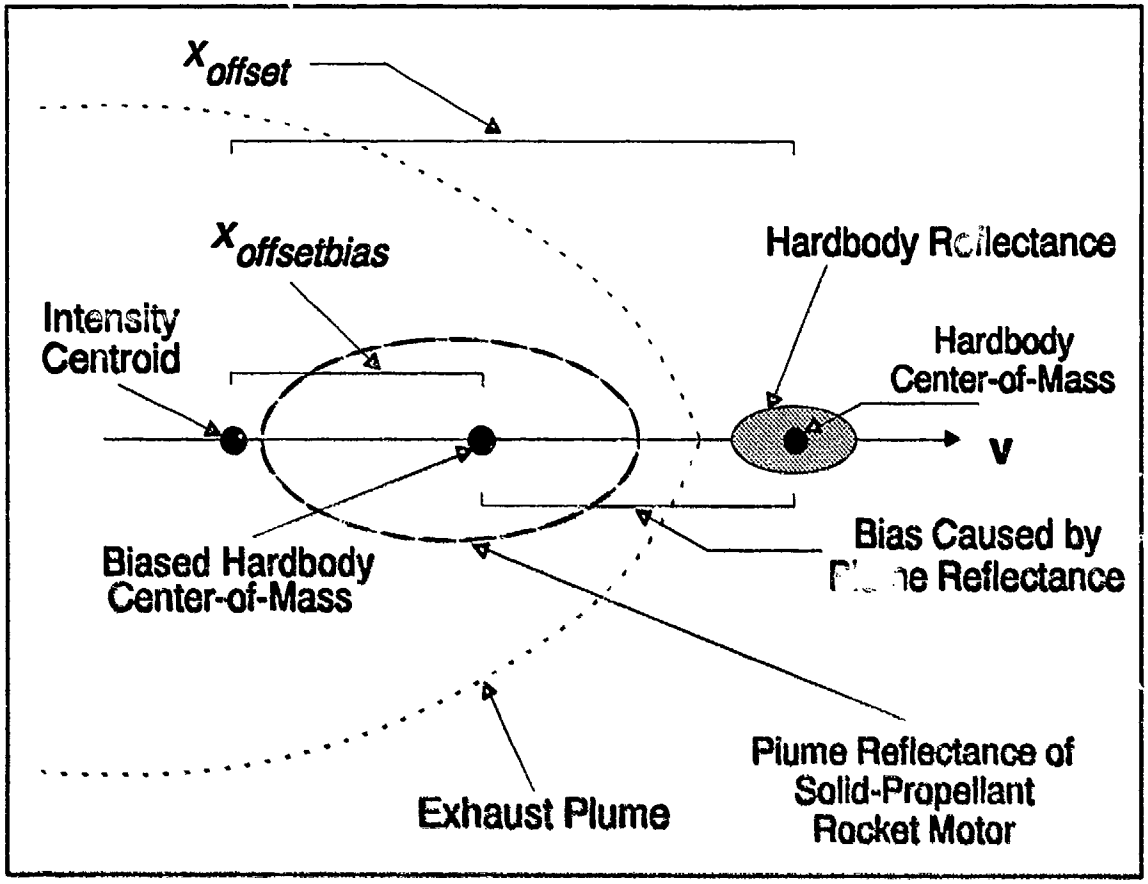


Figure 3.9 Biased Offset Measurement Caused by Plume Reflectance

provided to the filter for its update. The plume reflectance model is given by:

$$x_{offsetbias} = x_{offset} - \left(\frac{b}{Rk_p}\right)\cos\gamma \tag{3.53}$$

where:

$x_{offsetbias}$ = Biased offset measurement due to plume speckle reflectance

x_{offset} = Offset measurement from the low-energy reflectivity model, without plume speckle reflectance effect

b	=	Bias value
R	=	Range
k_p	=	Pixel proportionality constant (15 microradians/pixel)
γ	=	Angle between 3-dimensional inertial space velocity vector and the FLIR image plane

The randomized nature of the plume's reflectance is modeled as a percentage of time that appearance of the bias occurs. A random number generator, of uniformly distributed random variable output, provides the logic to turn the bias "on and off" according to the percentage selected. In correspondence with Phillips Laboratory personnel, it was found that a bias of approximately 25-30 meters with an appearance percentage of 90 - 95% was observed during the STARLAB flight experiment [1].

3.3.3 The Doppler Measurement Model. The Doppler measurement model simulates the offset measurements that are obtained by exploiting the differences between hardbody and plume-induced Doppler returns. As with the laser speckle return research of Eden and Evans, the modeling of the actual physical properties of the Doppler phenomenon will not be attempted. Instead, modeling efforts will entail simulating the information that would be available from Doppler detection circuits as measurement data for the Kalman filter. The following subsections briefly introduce and describe the basic concepts of the Doppler phenomenon, as applicable to the properties of the hardbody-induced and plume-induced Doppler returns. The

treatment of the Doppler phenomenon is not intended to be rigorous and reflects the level of understanding necessary to appreciate the manner with which the Doppler returns are employed to generate an offset measurement relative to the intensity centroid. For a rigorous development of the Doppler phenomenon, refer to *Principles and Practice of Laser-Doppler Anemometry* by F. Durst, A. Melling, and J. H. Whitelaw [2], and *The Doppler Effect* by T. P. Gill [7].

3.3.3.1 The Doppler Effect. Many define the Doppler effect as a shift in the frequency of a wave radiated, reflected, or received by an object in motion [38,39]. From a radar, Doppler shifts are produced by the relative motion between the radar and the target. The radar may use a pulsed, coherent laser beam that propagates the electromagnetic energy to "paint" the target of interest. If the target is in motion and illuminated by a low-energy laser, the returned signal (or backscatter) is represented as a time-delayed, Doppler-shifted version of the transmitted signal, wherein the amount of Doppler shift is proportional to the reflecting target's range rate relative to the laser transmitter [38,39]. A continuous transmitted signal is given as:

$$E_t = E_o \cos(2\pi f_o t) \tag{3.54}$$

For this transmitted signal, the echo signal from a moving target will be [38]:

$$E_r = k\, E_o \cos\left[2\pi(f_o \pm f_d)t + \phi\right] \tag{3.55}$$

where:

E_o = Amplitude of transmitted signal

f_o	=	Transmitted frequency
E_r	=	Reflected signal
k	=	An attenuation constant that represents losses incurred during propagation
f_d	=	Doppler frequency shift
ϕ	=	A phase shift, dependent upon the range of detection

Figure 3.10 shows the frequency spectrum of the return signal, shifted from the transmitted frequency, f_o, by the Doppler shift, f_d, given by [38]:

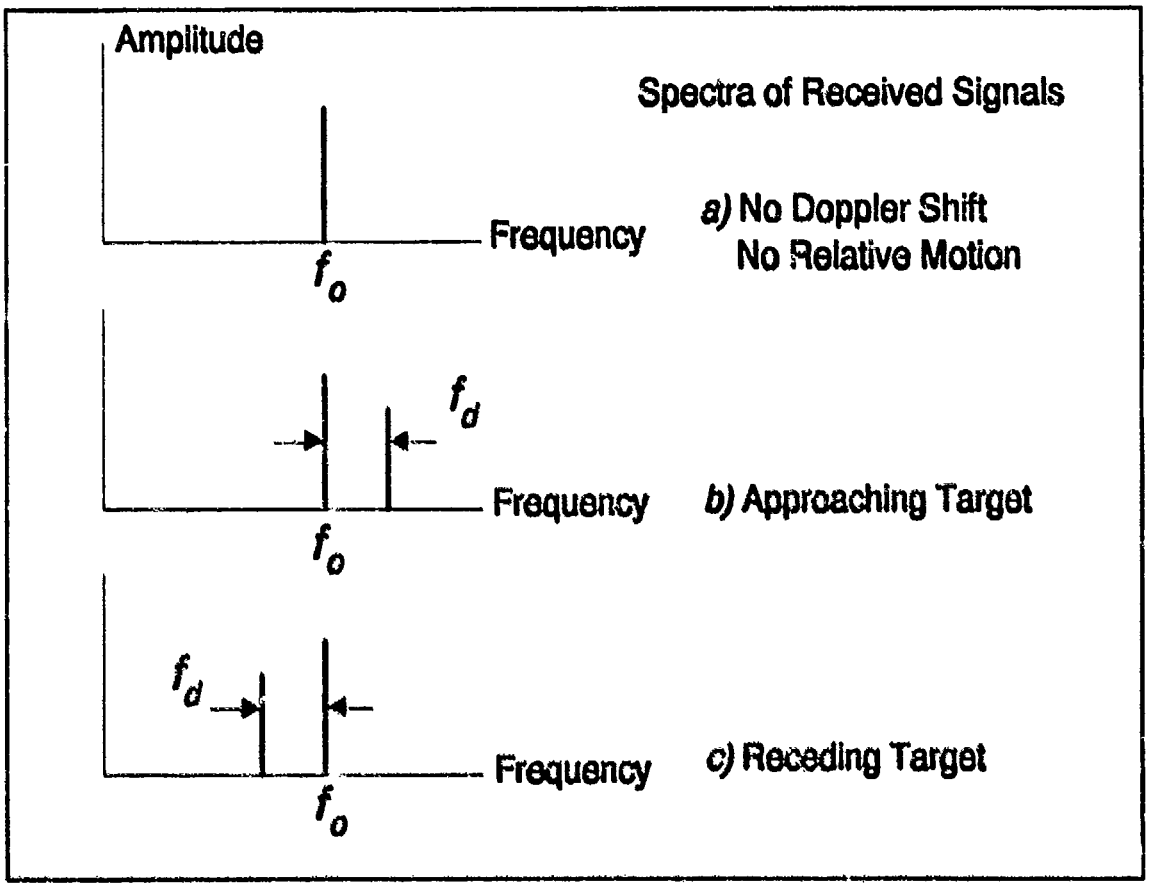


Figure 3.10 Spectra of Received Signals [43]

$$\pm f_d = \pm\frac{2v_r}{\lambda} = \pm\frac{2v f_o}{c} \tag{3.56}$$

where:

v_r	=	Relative velocity of target with respect to transmitter
λ	=	Transmitted wavelength
c	=	Velocity of signal propagation (3×10^8 meters/second)

The relative velocity, v_r, is expressed as:

$$v_r = v \sin\gamma \tag{3.57}$$

where:

v	=	Target velocity in 3-dimensional inertial space
γ	=	angle between the target trajectory and plane perpendicular to the laser LOS (FLIR plane; see Figure 3.6)

The plus sign associated with the Doppler frequency shift applies if the distance between target and transmitter is decreasing (approaching target), and conversely, the minus sign applies if the distance is increasing (receding target).

As shown in Figure 3.10, the frequency spectrum of a continuous reflected sinusoidal signal appears as a straight vertical line. The scenario proposed by the Phillips Laboratory calls for a pulsed and coherent laser beam to illuminate a ballistic boosting target [1]. Both these laser properties have an impact upon the nature of the returned spectrum.

For illustration purposes, Figure 3.11 shows a train of independent pulses having a pulse width (PW) of 0.001 seconds and a constant pulse repetition frequency (PRF), along with its associated frequency spectrum. Because the pulses are "on" a fraction of the time, the amplitude of the frequency spectrum decreases but is still centered at f_o. The total power is in fact distributed over a band of frequencies extending from 1000 Hz below f_o to 1000 Hz above it, for a null-to-null bandwidth of 2 KHz. The bandwidth (i.e. spectrum spread), is inversely proportional to the pulse width and is given by [39]:

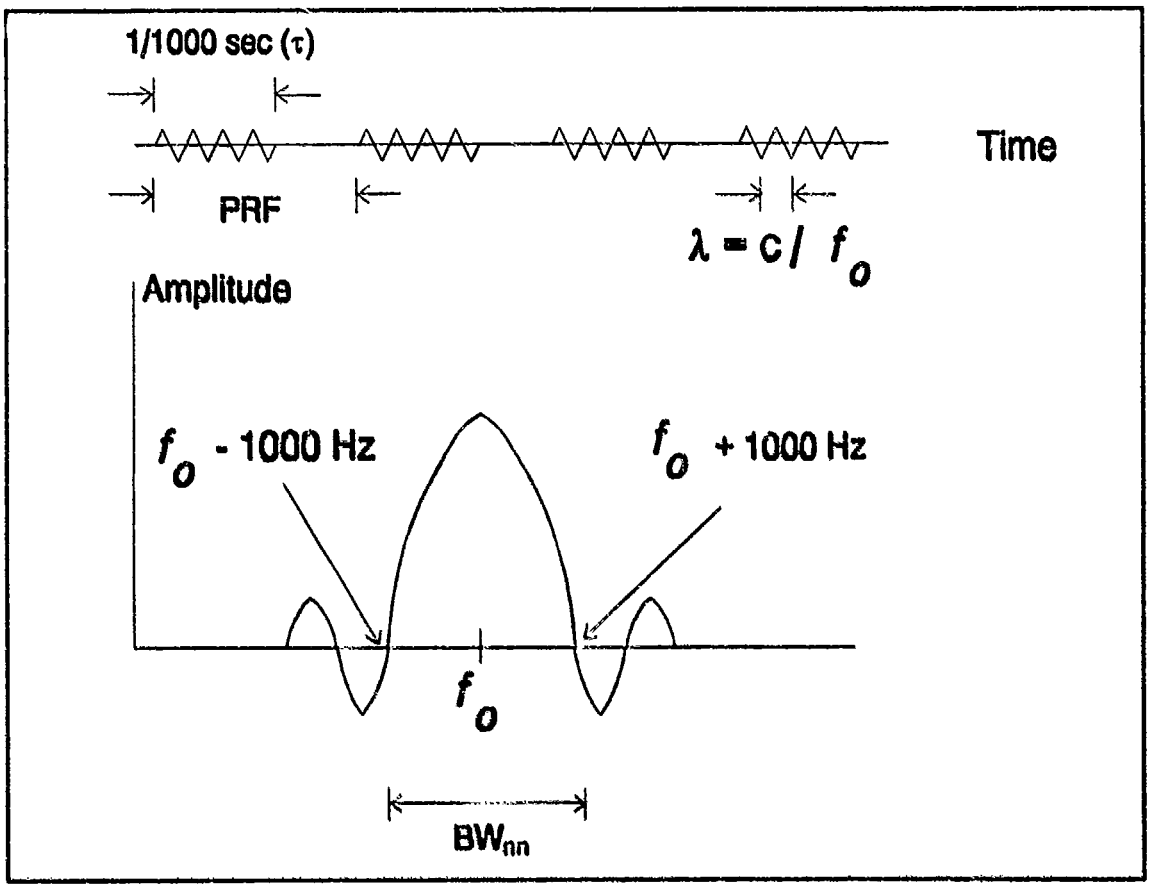


Figure 3.11 Pulsed Signal Frequency Spectrum

$$BW_{nn} = \frac{2}{\tau} \tag{3.58}$$

where:

BW_{nn} = Null-to-null bandwidth

τ = Pulse width (seconds)

By coherence is meant a consistency, or continuity, in the phase of a signal from one pulse to the next [39]. The term ϕ in Equation (3.55) represents the phase shift, which is a function of the range during detection. Figure 3.12 illustrates the difference between the frequency spectrum of a coherent signal and a non-coherent signal. With non-coherent transmission, the signal's central spectral lobe is spread over a band of frequencies. In contrast, the spectrum associated with coherent transmission shows the signal appearing at many points. Its spectrum, in fact, consists of a series of evenly spaced lines, wherein the interval between the spectral lines equals (1/PRF) [39]. Further comparison reveals that the coherent frequency spectrum is stronger (having a higher amplitude) than the non-coherent signal because the energy has been concentrated into a few narrow lines. In addition, the envelope within which these lines fit has the same shape, [sin(x)/x], and the same null-to-null BW, $2/\tau$, as the spectrum of the non-coherent signal.

3.3.3.2 Hardbody Doppler Return. At a range of 2,000 kilometers, the missile hardbody can be defined as a smooth, dense single point target. It is assumed that

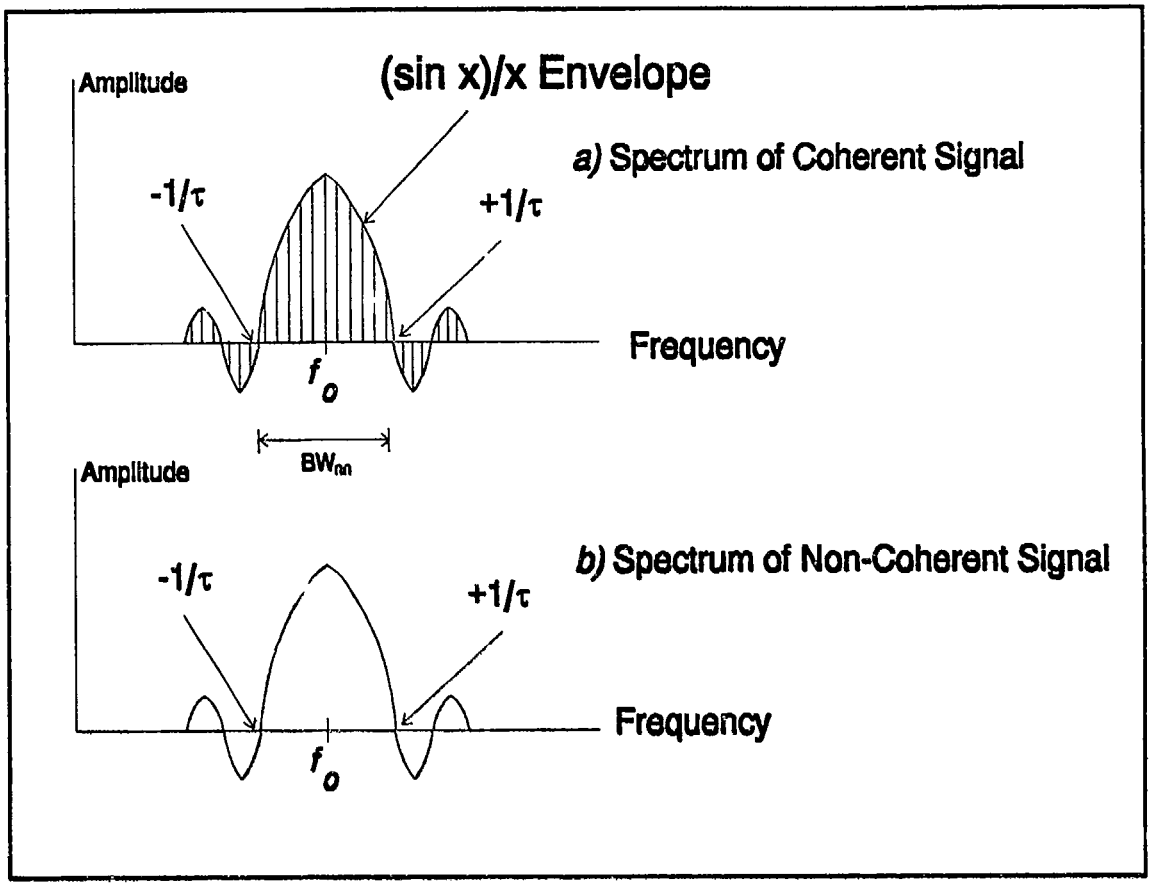


Figure 3.12 Spectra of Coherent and Non-coherent Pulsed Signals [44]

the velocity of any point of the hardbody due to the hardbody's rotational motion is much less than the hardbody's linear velocity and is considered negligible. It is also assumed that the target hardbody's velocity remains constant over the duration of a transmitted pulse. With such a target, the spectrum of the return will have a bandwidth that closely approximates ($2/\tau$), and centered about the Doppler-shifted frequency corresponding to the relative rate.

3.3.3.3 Plume Doppler Return. The case of the exhaust plume can be represented as the situation in which numerous point targets are imaged together. The plume can be described as a randomly distributed array of point targets which are dispersed in range and velocity. The plume particulates are small (submicron in size), nonspherical and nonhomogeneous, and their size and spatial distribution vary strongly with the radial distance from the plume axis [9,45]. Typically, larger particles are concentrated near the plume's symmetry axis, and in contrast to the hardbody, the numerous exhaust plume particles exhibit numerous velocity orientations over the duration of a laser pulse.

When the laser beam illuminates such a large number of point targets, the superposition of each particle's backscatter radiation within the laser beamwidth will form the resultant return [44,45]. Thus, the Doppler frequency spectrum will be quite broad, due to the numerous Doppler shifts of the numerous plume particulate velocities [1,9]. This Doppler spreading of spectral lines arises from the fact that backscatter from a particulate will be shifted in frequency in a manner depending on the approach or recession of the particulate as seen from the tracker location. The plume experimental programs at AEDC have observed and measured plume Doppler reflectance frequency spectrums with null-to-null BWs of 2 - 5 GHz [28]. This sharply contrasts the hardbody-induced return, for which the spectrum null-to-null BW equals $2/\tau$, with an order of magnitude in MHz. However, one other significant difference exists between the hardbody and plume-induced Doppler returns.

Generally, the velocity of the plume will be oriented 180° from the hardbody's velocity [1,9]. This is shown in Figure 3.13(a), where the respective Doppler frequency shifts will be opposite in sign. A majority of the observed plume particles would have a relative radial velocity towards the tracker and the resultant return would have a negative Doppler frequency shift. Conversely, the hardbody as shown is receding from the tracker and will thus exhibit a positive Doppler frequency shift. Hence, by exploiting the two differences in plume and hardbody-induced Doppler

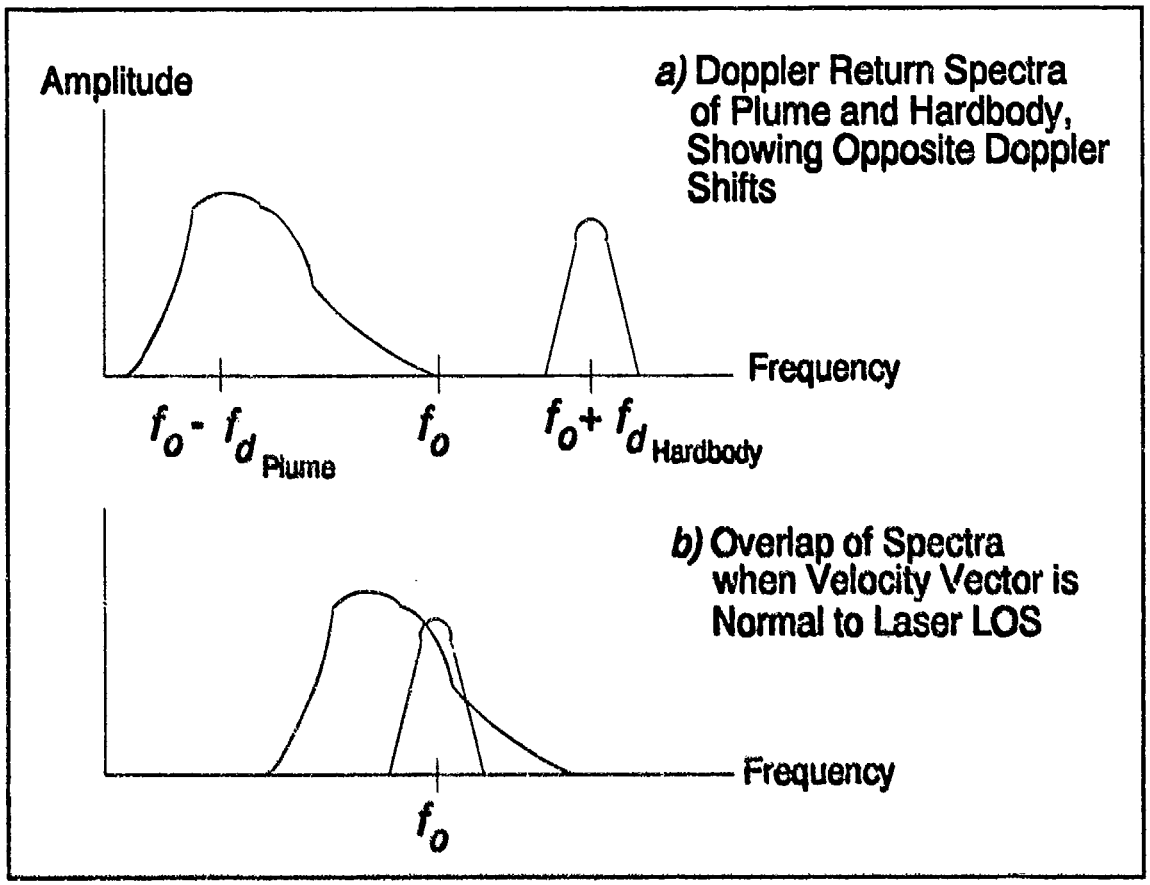


Figure 3.13 Spectra of Plume and Hardbody-Induced Doppler Returns

returns, namely center frequency and breadth of the two corresponding spectra, precise tracking and definition of the plume/hardbody interface can be realized.

However, the angle γ, of which the relative velocity v, is a function, has an impact upon the discernability between the plume and the hardbody-induced Doppler shifts. Referring to Equation (3.57), as γ approaches 0°, where the plume and hardbody velocity vectors become orthogonal to the LOS vector, the radial velocity of the target relative to the tracker approaches nil and no Doppler shift is produced. Figure 3.13(b) shows that, under these circumstances, the return spectra of the plume and hardbody converge towards the transmitted frequency and eventually overlap, obscuring most of the hardbody-induced Doppler return. This imperfect ability to detect the hardbody spectrum, as distinct from the plume spectrum, will be addressed in the next section, which develops the Doppler measurement model.

The measurement modeling approach taken by this thesis is to consider the usual circumstance of the Doppler return of the hardbody being significantly distinctive from that of the plume. The Doppler detector must be designed to filter out the broader plume return and only pass the hardbody return, a function achievable with a Doppler matched filter design [9,38]. This vital concept signifies that the Doppler truth measurement model can neglect the plume's Doppler return and solely simulate the hardbody-induced Doppler return. Although there may be instances of no apparent distinction between the plume and hardbody spectra, these

occurrences will be embodied in a probability-of-miss parameter (P_m), to be discussed later.

Since Doppler information is obtainable from backscatter radiation, which includes the speckle return [38,39], a 3-d hardbody reflectivity model, detailed in Section *3.3.2*, is utilized in this modeling approach. However, in contrast to the laser speckle return measurement model, the biasing effect caused by the plume's reflectance is no longer applicable and is not incorporated into the Doppler measurement model. As a result, the center-of-mass measurement and offset measurement generated by the Doppler measurement model will simulate a zero-mean error-corrupted version of the true offset measurement, x_{offset}, for the filter.

3.3.3.4 Doppler Measurement Noises. The Phillips Laboratory sponsored a study in which Dr. Paul McManamon investigated feasible and implementable wavelengths to illuminate the plume and hardbody, while meeting the space tracking scenario requirements [9]. His choice of wavelengths, based upon ranges, power requirements, hardbody temperatures, and tracking accuracies, range from 0.53 to 15 μm. For this study, the shortest wavelength 0.53 μm (which provided the greatest precision in the measured value) was selected for use in a sensitivity analysis. The tracking inaccuracies associated with this wavelength are adopted in the Doppler measurement model to corrupt the offset measurement realistically.

The tracking accuracy for a laser beam is a function of the amount of power, or amplitude, of the return signal. The return signal, in turn, is dependent upon several variables, among which are the target's radar cross section (RCS) and the location of the target in the laser beam [9,33,39]. A target ideally located in the center of the laser beam reflects the maximum return signal (i.e., optimum signal-to-noise ratio, SNR). If the target falls off to the side of the beam's center, then less energy hits the target. The degree of tracking accuracy then becomes a question of, how far off to the side can a target be to reflect the signal at an acceptable level?

Dr. McManamon addressed this issue [9] by first defining the acceptable beam diffraction limit as the angle within the 3 db power points of the laser beam. He defines the diffraction limit as:

$$\theta_{3db} = 1.08\frac{\lambda}{d} \tag{3.59}$$

where:

θ_{3db}	=	Half angle defined from beam center to half-power points, in radians
λ	=	Wavelength, in meters
d	=	Radar aperture, in meters

One then determines the acceptable level of signal loss within the θ_{3db} limits. In Dr. McManamon's assessment, a 10% loss can be tolerated, and he determined that this

loss is reflected by decreasing the diffraction limit by a factor of 2.667 [9]. Equation (3.59) yields:

$$\theta_B = \frac{\theta_{3db}}{2.667} \tag{3.60}$$

where:

θ_B = Allowed diffraction limit for 10% signal loss

The measurement noise for the Doppler measurement model thus consists of the tracking angle errors, in pixels, as a function of the diffraction limited beam and acceptable signal-to-noise ratio (SNR). Herrera's study [9] included the following values of SNR for the sensitivity analysis: 10, 8, 6, and 4. The relationship is given as [9]:

$$\theta_T = \frac{\theta_B}{3\sqrt{SNR}} \frac{1}{k_p} \tag{3.61}$$

where:

θ_T = rms tracking angle errors in pixels

θ_B = Beam diffraction limit

SNR = Signal-to-noise ratio

k_p = Pixel proportionality constant, 15 μrads/pixel

In addition to providing the offset measurement, the Doppler measurement model also simulates a return signal probability-of-miss, P_m. The probability-of-miss

encompasses two cases. First, the probability-of-miss takes into account the situation in which the hardbody is illuminated by the low-energy laser, but the return is not detected due to attenuation of the returning signal as it propagates the 2,000 kilometer range, beam-bending as a result of atmospheric distortions (the intended location of the laser scan should have illuminated the target, but bending of the beam resulted in no intersection with the target); or due to signal losses (i.e., high sensor sensitivity threshold; refer to Section *3.3.2.1*) within the receiving equipment. In this case, a loss of speckle information would also result. Secondly, in Equation (3.51), it was shown that the relative velocity is a function of γ, such that no Doppler shift occurs if the target's velocity is normal to the transmitter's LOS. Hence, as shown in Figure 3.13, as γ approaches 0°, both the broadened plume-induced Doppler spectrum and hardbody-induced spectrum will converge and overlap. The two spectra will become more indistinguishable, perhaps rendering detection of the hardbody's Doppler return impossible. In this second case, there would not be a simultaneous loss of speckle information.

The simulation of the probability-of-miss is similar to the technique employed by the plume reflectance model. A random number generator, with a uniformly distributed output, also provides the logic to turn the hardbody laser backscatter "on and off." Figure 3.14 shows the detection characteristic for a known signal. The graph presents a set of parametric curves that give the probability-of-detection, P_d, values as a function of peak signal-to-noise ratio (SNR) for various values of

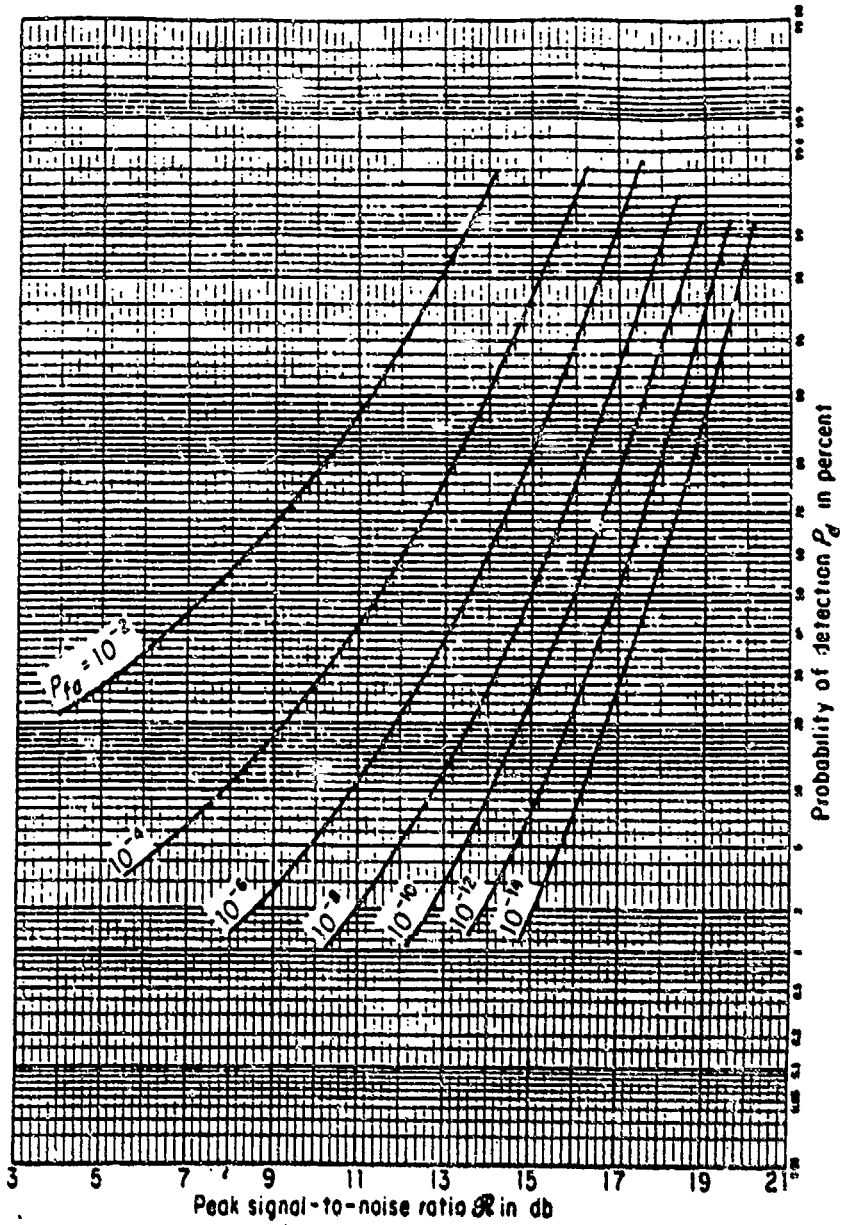


Figure 3.14 Detection Characteristics [43]

probability-of-false alarm, P_{fa}. P_{fa} is defined as falsely indicating the presence of a return signal when none exists [38]. Both P_d and P_{fa} are specified by the system requirements; the radar designer computes the probability-of-false alarm and, from Figure 3.14, determines the minimum detectable signal. A range of 70 - 99 percent probability-of-detection is representative of current Doppler detection equipment capabilities with the tracking scenario [9].

3.4 *Truth Model Parameters*

The discussions in the previous sections introduced some of the truth model parameters used in the simulation. The purpose of this section is to provide a consolidated listing of the parameters and initial conditions of the truth model.

3.4.1 Target Trajectory Initial Conditions. The initial conditions of the target inertial position, velocity, and velocity vector orientation angle, θ, are as follows:

$$e_x = 27{,}000 \text{ meters}$$

$$e_y = 100{,}000 \text{ meters}$$

$$e_z = 2{,}000{,}000 \text{ meters}$$

$$v_x = -2{,}500 \text{ meters/second}$$

$$v_y = 4{,}330 \text{ meters/second}$$

$$v_z = 0 \text{ meters/second}$$

$$\theta = 60°$$

These initial conditions represent data from the Atlas ICBM; the physical meaning is tied to the reference frame description of Section *2.4.1.1* where e_x is along the LOS vector (thus the 2,000 kilometer distance). The angle, θ, is the true angle that is programmed as a constant in the software.

3.4.2 Target Model, Dimensions, and Orientation. The target plume consists of a crescent-shaped intensity function formed from the difference of two bivariate Gaussian intensity functions. Each Gaussian function is modeled with elliptical constant-intensity loci with an aspect ratio of 1.5, and a semi-minor axis of one. For this thesis, Evans' 3-dimensional reflectivity model is used to model the hardbody. The hardbody length is 40 meters (1.33 pixels) and 3 meters (0.1 pixels) wide. The offset distance of the hardbody center-of-mass from the intensity centroid (actually measured to the pogo equilibrium zero value) is 87.5 meters (2.92 pixels), a carryover from the previous thesis. For the simulation, the intensity centroid and the hardbody longitudinal axis are aligned with velocity vector, and the hardbody has zero sideslip and zero angle-of-attack.

3.4.3 Intensity Functions. The two Gaussian bivariate intensity functions, shown in Figure 3.3, are centered at 65 and 110 meters behind the missile. Each intensity function has a maximum intensity value of 20 intensity units.

3.4.4 Atmospheric Jitter. The variance and mean squared value for the atmospheric jitter in both FLIR directions are 0.2 pixels^2.

3.4.5 Bending/Vibration. From Equation (3.29), the values for the second-order bending/vibration model are as follows:

λ_b^2 = 5×10^{-13}

ζ_b = 0.15

ω_{nb} = π radians/seconds

3.4.6 Plume Pogo Characteristics. The size of the plume is on the order of 30 times the diameter of the missile at the altitudes of interest. The values below represent values of pogo oscillation as determined in previous research [34].

pogo oscillation = 0.1 - 10 Hz (nominal is 1 Hz)

pogo rms = 0.0112 - 1.12 pixels (nominal is 0.112 pixels, which is equal to 3.36 meters at the target; one pixel at 15 μradians per side at the range of 2,000 kilometers equals 30 meters)

3.4.7 Spatially Correlated Background Noise. The rms value of v_{jk}, the summed effect of the spatially correlated background noise b_{jk} and the FLIR sensor noise n_{jk}, of Equation (3.45), equals one. This produces a SNR of 20.

3.4.8 Low-Energy Laser Speckle Return Measurement Dimensions. The low-energy scan is represented as a rectangle at the hardbody target. The scan length is 262.5 meters (8.75 pixels), which is three times the true model center-of-mass offset distance, and the scan width is 0.1 meters. The measurement noise associated with the speckle return is obtained by taking 1% of the hardbody's length, and converting to pixels, giving a variance of 0.000178 pixels^2 [4].

3.4.9 Plume Reflectance Model. The bias utilized by the plume reflectance model is approximately 25 - 30 meters and appears 90 - 95% of the time while the plume is illuminated during the boost phase [1]. For the simulation, nominal values for the bias and rate of appearance are set at 25 meters and 90%, respectively.

3.4.10 Low-Energy Doppler Return Measurement Dimensions. The Doppler measurement noise rms tracking errors are functions of wavelength, radar aperture, and SNR. The previous thesis studied filter performance dependent upon the wavelength values of 0.53 μm, 1.06 μm, 2.01 μm, 4.00 μm, 6.00 μm, 8.00 μm, and 10.60 μm, with SNR values of 10, 8, 6, and 4, and probability-of-miss P_m values of 0.00, 0.01, 0.02, 0.03, 0.04, 0.05, 0.10, 0.20, and 0.30. Since the purpose of this thesis

was to incorporate pogo into the filter, the filter was given the benefit of the best possible Doppler measurements throughout the simulation. Therefore the conditions were set for a wavelength of 0.53 μm, a SNR of 10 and a P_m value of 0.01. The radar aperture d of Equation (3.59) of 0.5 meters was carried over from the previous thesis.

3.4.11 Hardbody Reflectivity Measurement Model. The function $\mu(\cdot)$, in Equation (3.50), represents the sensitivity threshold of the low-energy laser return sensor. The magnitude of reflection must be greater than the threshold, m_r, in order to detect the return from the hardbody. In the simulation, the value of the threshold is set to 0.00. This was to allow reception of measurements if any backscatter was received at all (perfect LEL receiving equipment).

3.5 Summary

This chapter presented the mathematical description of the truth model. The truth model consists of 14 states: 2 deterministic target trajectory states, 6 stochastic atmospheric jitter states, 4 stochastic bending/vibration states, and 2 stochastic plume pogo states. The infrared target plume model is formed from the difference of 2 bivariate Gaussian functions. The FLIR measurements are corrupted by spatially and temporally uncorrelated FLIR sensor noise, and spatially correlated and temporally uncorrelated background noise. The low-energy laser measurement models, which provide an offset measurement from the intensity centroid to the hardbody center-of-mass, consist of the plume reflectance model, the 3-dimensional hardbody reflectivity model, and the Doppler measurement model. The plume reflectance model simulates the elongation of the apparent hardbody in the speckle measurement data due to the simultaneous hardbody and plume speckle return. The 3-dimensional hardbody reflectivity model provides realistic backscatter that is a function of the hardbody's curvature and aspect angle. The Doppler measurement model also utilizes the backscatter information from the 3-dimensional reflectivity model and corrupts that information with noise having rms angle tracking errors associated with a particular wavelength, radar aperture, and SNR.

IV. Filter Models

4.1 Introduction

This chapter describes the linear Kalman filter (KF) dynamics and measurement models that make up the basic structure for one of the elemental filters of the Multiple Model Adaptive Estimator shown in Figure 1.3. This elemental KF model has changed many times over the course of this research strain to be finally consolidated in Ching's [1] thesis. These KF structures are reduced-order, simplified versions of the truth model. Section *4.2* will describe the defining equations of the elemental filter dynamics model. Section *4.3* details the filter measurement model structure.

4.2 Dynamics Models

The elemental AFIT filter used in this work is a single nine-state filter combining models that have been developed by past AFIT students from this research line [1,3,8,9,26,28,34,35]. The filter consists of two hardbody center-of-mass position states (note that these two states are different from the first two states of the truth model, which are position of plume intensity centroid), two hardbody center-of-mass

velocity states, two atmospheric jitter position states (affecting the plume centroid in the FLIR plane), two pogo oscillation states (affecting centroid location relative to hardbody center-of-mass), and a plume intensity centroid/hardbody center-of-mass offset state. The state vector for this elemental filter is:

$$x(t) = [\, x_t \; y_t \; v_x \; v_y \; x_a \; y_a \; x_p \; v_p \; x_o \,]^T \tag{4.1}$$

where:

x_t	=	x component of target (center-of-mass of hardbody) position (azimuth) relative to center of the FOV
y_t	=	y component of target (center-of-mass of hardbody) position (elevation) relative to center of FOV
v_x	=	x component of target (center-of-mass of hardbody) velocity
v_y	=	y component of target (center-of-mass of hardbody) velocity
x_a	=	x component of atmospheric jitter
y_a	=	y component of atmospheric jitter
x_p	=	plume pogo offset distance (along velocity vector) of plume centroid from the equilibrium point
v_p	=	velocity of the pogo oscillation (along velocity vector)
x_o	=	offset distance between the plume centroid equilibrium point and the hardbody center-of-mass

Each state in Equation (4.1) is coordinatized in the α-β (FLIR) plane. A comparison between the filter model and truth model show that some state reduction

has occurred. Only two atmospheric states are represented in the filter, compared with six in the truth model. The high frequency poles have been eliminated (four states, two for each double pole in each coordinate direction) due to their negligible impact [32]. The bending/vibration states have also been eliminated for this reason [12]. The pogo states in the filter are identical to the pogo states in the truth model. The total offset distance from the filter-predicted plume centroid to the filter predicted hardbody center-of-mass is composed of a linear combination of filter states, including x_a, y_a, x_p, and x_o.

The filter model is described by the following time-invariant, linear stochastic differential equation [21]:

$$\dot{x}(t) = F\,x(t) + G\,w(t) \tag{4.2}$$

where:

F	=	Time-invariant system (plant) matrix
$x(t)$	=	9-dimensional filter state vector
G	=	9 x 6 time-invariant noise distribution matrix
$w(t)$	=	6-dimensional, white Gaussian noise process with independent components, and mean and covariance kernel statistics:

$$E\{w(t)\} = 0$$
$$E\{w(t)w(t+\tau)^T\} = Q\,\delta(t) \tag{4.3}$$

The filter state estimate and error covariance matrix are propagated forward to the next measurement update using the following discrete-time filter propagation equations [16]:

$$\hat{x}(t_{i+1}^-) = \Phi(\Delta t)\hat{x}(t_i^+) \tag{4.4}$$

$$P(t_{i+1}^-) = \Phi(\Delta t)P(t_i^+)\Phi^T(\Delta t) + Q_d \tag{4.5}$$

where:

$\hat{x}(t_i)$	=	Filter estimate of the 9-dimensional state vector
$\Phi(\Delta t)$	=	9 x 9 time-invariant state transition matrix for propagation over the sample period: $\Delta t = t_{i+1} - t_i$
$P(t_i)$	=	9 x 9 filter covariance matrix
(t_i^-)	=	Time instant before FLIR measurement is incorporated into the estimate at time t_i
(t_i^+)	=	Time instant after FLIR measurement is incorporated into the estimate at time t_i

Q_d = 9 x 9 filter dynamics discrete noise covariance given by:

$$Q_d = \int_{t_i}^{t_{i+1}} \Phi(t_{i+1} - \tau) G \, Q \, G^T \, \Phi^T(t_{i+1} - \tau) d\tau \tag{4.6}$$

The block diagonal components of Equations (4.2), (4.4), and (4.5) associated with each state are described in the next four sections.

4.2.1 Target Dynamics Model. The elemental filter uses four states to describe the target dynamics. The velocity states are represented as random constants plus noise with the time-invariant continuous-time dynamics system (plant) matrix, F_t, given by:

$$F_t = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.7}$$

The noise distribution matrix, G_t, is:

$$G_t = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{4.8}$$

The strength of the white Gaussian noise w_t, is given by Q_t, is:

$$Q_t = \begin{bmatrix} Q_x & 0 \\ 0 & Q_y \end{bmatrix} \tag{4.9}$$

where Q_x and Q_y are the noise strength values in the x and y directions. The time-invariant target dynamics state transition matrix, $\Phi_t(\Delta t)$, is given by:

$$\Phi_t(\Delta t) = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.10}$$

The solution to Equation (4.6) using Equations (4.8), (4.9), and (4.10) yields the filter dynamics discrete noise covariance, Q_{dt}, given by:

$$Q_{dt} = \begin{bmatrix} \frac{1}{3}Q_x\Delta t^3 & 0 & \frac{1}{2}Q_x\Delta t^2 & 0 \\ 0 & \frac{1}{3}Q_y\Delta t^3 & 0 & \frac{1}{2}Q_y\Delta t^2 \\ \frac{1}{2}Q_x\Delta t^2 & 0 & Q_x\Delta t & 0 \\ 0 & \frac{1}{2}Q_y\Delta t^2 & 0 & Q_y\Delta t \end{bmatrix} \tag{4.11}$$

4.2.2 Atmospheric Disturbance Model. The atmospheric jitter model describes the motion of the plume image in the FLIR plane due to atmospheric disturbances (refraction variations from moisture, thermal variations, etc.). The six-state filter has

been reduced to two states, one in each the x and y direction. The time-invariant system matrix, $\boldsymbol{F}_a$, of the continuous-time dynamics model of Equation (4.2) is:

$$\boldsymbol{F}_a = \begin{bmatrix} -\omega_a & 0 \\ 0 & -\omega_a \end{bmatrix} \tag{4.12}$$

where:

ω_a = Atmospheric jitter break frequency, 14.14 rad/sec

The noise distribution matrix, $\boldsymbol{G}_a$, is:

$$\boldsymbol{G}_a = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{4.13}$$

The strength of the white Gaussian noise, $\boldsymbol{Q}_a$, is:

$$\boldsymbol{Q}_a = \begin{bmatrix} \dfrac{2\sigma_a^2}{\tau_a} & 0 \\ 0 & \dfrac{2\sigma_a^2}{\tau_a} \end{bmatrix} \tag{4.14}$$

where:

σ_a^2 = Variance and mean-squared value for the atmospheric jitter process

τ_a = Correlation time constant for atmospheric jitter process ($\tau_a = 1/\omega$)

The time-invariant target dynamics state transition matrix, $\Phi_a(\Delta t)$, is given by:

$$\Phi_a(\Delta t) = \begin{bmatrix} \exp(-\omega_a \Delta t) & 0 \\ 0 & \exp(-\omega_a \Delta t) \end{bmatrix} \tag{4.15}$$

The part of the solution to Equation (4.6) due to atmospheric jitter, using Equations (4.13), (4.14), and (4.15) yield the filter dynamics noise covariance, Q_{da}, is:

$$Q_{da} = \begin{bmatrix} q_{da11} & 0 \\ 0 & q_{da22} \end{bmatrix} \tag{4.16}$$

where:

$$q_{da11} = q_{da22} = \sigma_a^2 \left[1 - \exp\left(-\frac{2(\Delta t)}{\tau_a} \right) \right] \tag{4.17}$$

4.2.3 Pogo Dynamics Model. The implemented filter plume pogo model is identical to the truth model described in Section *3.2.4*. The time-invariant system matrix, F_p, is given by:

$$F_p = \begin{bmatrix} 0 & 1 \\ -\omega_{pf}^2 & -2\zeta_{pf}\omega_{pf} \end{bmatrix} \tag{4.18}$$

where:

ω_{pf} = Undamped natural pogo frequency (0.1 - 10 Hz)

ζ_{pf} = Filter damping coefficient chosen to be 0.05 [34]

The noise distribution matrix, G_p, is:

$$G_p = \begin{bmatrix} 0 \\ K_{pf}\ \omega_{pf}^2 \end{bmatrix} \tag{4.19}$$

and the white Gaussian noise, w_p is of unit strength:

$$Q_p = 1 \tag{4.20}$$

The time-invariant target dynamics state transition matrix, $\Phi_p(\Delta t)$, is given by:

$$\Phi_p(\Delta t) = \begin{bmatrix} \Phi_{p11}(\Delta t) & \Phi_{p12}(\Delta t) \\ \Phi_{p21}(\Delta t) & \Phi_{p22}(\Delta t) \end{bmatrix} \tag{4.21}$$

with the individual elements of the $\Phi_p(\Delta t)$ given in Equation (3.43). The filter dynamics noise covariance matrix, Q_{da} is not included here due to its length and complexity. The complete matrix description is found in the AFIT software [25].

4.2.4 Centroid Equilibrium Point / Center-of-Mass Offset Model. In previous theses [8,9,14], the measurement determined from the LEL (by either speckle return or Doppler spectra of the plume and hardbody) was processed in an independent center-of-mass offset filter. The estimate from that filter was then added to a FLIR filter estimate of the position of the centroid in order to obtain center-of-mass position. Since pogo was not included in the filter models, the offset between the center-of-mass and the intensity centroid was modeled as a constant. For this thesis, the same dynamics model is used, but the offset state is augmented to the previous

models to form a single nine-state elemental filter. The centroid offset state, modeled as a constant bias, describes the distance between the center-of-mass of the missile hardbody and the equilibrium point about which the plume pogos. Equations (4.2), (4.4), and (4.5) are still applicable but are expressed in scalar form since there is only a single state. The bias is modeled as the output of a simple integrator, with driving pseudo-noise for filter tuning purposes.

The elements of the linear, time-invariant stochastic differential equation are:

F_o = 0

G_o = Time-invariant noise distribution matrix, equal to unity

$w_o(t)$ = White Gaussian noise process, independent of the noises driving the target dynamics and atmospheric jitter models, with mean and covariance kernel statistics:

$$E\{w_o(t)\} = 0$$
$$E\{w_o(t)w_o(t+\tau)\} = Q_o\,\delta(t) \tag{4.22}$$

and $Q_o = 1$.

The elements of the equivalent discrete-time filter propagation Equations (4.4) and (4.5) are given by:

$\Phi_o(\Delta t)$ = Time-invariant state transition matrix, equal to unity

Q_{do} = Filter dynamics noise variance equal to $Q_o\Delta t$

4.3 Measurement Models

This section discusses the two different measurement models t re used in this simulation. The thesis students in this research strain only had one measurement source, the FLIR, until Eden [8], who added the second source, a low-energy laser. The FLIR measurement model is detailed in the first subsection, where it is explained how the raw data is processed through the enhanced correlation algorithm. This raw data, compared with data templates, form the "pseudo-measurements" for the linear Kalman filter. The linear and non-linear update functions used for the Kalman filters are also presented. The last subsection describes the measurement model used for the low-energy laser measurements.

4.3.1 FLIR Measurement Model. Measurements of the plume intensity centroid's position are generated by an enhanced correlator algorithm, shown in Figure 4.1, developed by Rogers [22,35]. This enhanced correlator algorithm compares the incoming FLIR data frame to a template that represents an estimate of the target plume's intensity function. Previous correlators compared the current data frame with the previous data frame. The "pseudo-measurements" of the centroid's position offsets produced by the enhanced correlator are a nonharsh nonlinear function of the states being estimated, and thus a nearly linear Kalman filter is used [9].

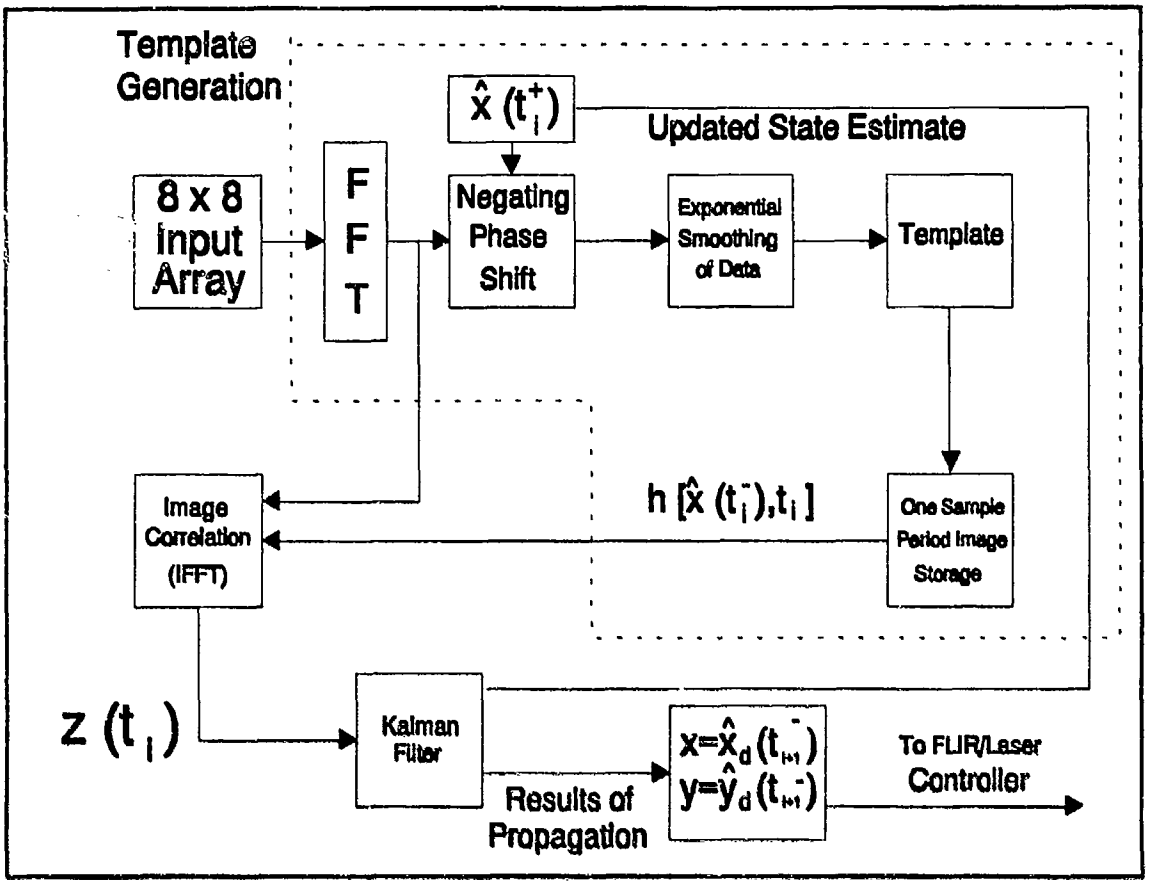


Figure 4.1 Linear Kalman Filter/Enhanced Correlator Algorithm

4.3.1.1 Enhanced Correlator Algorithm. The algorithm presented here was developed as an alternative to an earlier 64-dimensional, nonlinear measurement model. Previously, an extended Kalman filter processed raw FLIR measurement data from a standard FLIR sensor, with no correlation algorithm utilized [26]. With the enhanced correlator algorithm, a nearly linear Kalman filter is employed since the output measurements from the correlation algorithm are 2-dimensional position "measurements" that are nonharsh nonlinear functions of the states to be estimated. As will be seen later in the section, the nonlinearity is a sinusoidal function caused

by the introduction of the plume pogo effect into the model. The sinusoidal function is a mild nonlinearity in comparison to the extended Kalman filter measurement model required to process raw FLIR data. This enhanced correlator/linear Kalman filter configuration performs as well as the extended Kalman filter with respect to rms tracking errors and further provided a reduction in computational loading [1]. The "enhancement" occurs in the following manner [35]:

1. The most current FLIR data is correlated with a template (which is an estimate of the target's intensity function), instead of with the previous FLIR data frame.

2. Instead of outputting the location of the peak of the correlation function, a technique known as "thresholding" is used along with a simple center-of-mass computation. The enhanced correlator outputs the center-of-mass of the portion of the correlation function that is greater than some predetermined lower bound. Consequently, the enhanced correlator has no difficulty distinguishing global peaks from local peaks, as do many conventional "peak-finding" correlation algorithms.

3. The FLIR/laser pointing commands are generated via the one-sample-period-ahead predictions from the Kalman filter propagation cycle instead of by the "raw measurement" output of a standard correlation algorithm.

4. The Kalman filter estimate, $\hat{x}(t_i^+)$, is used to center the template, so that the offsets seen in the enhanced correlator algorithm should be smaller than those visible in the conventional correlator. This increases the amount of "overlap" between the actual FLIR data and the stored template, and thus improves performance.

Referring back to Figure 4.1, the enhanced correlation algorithm uses the 8x8 array of target intensities obtained by the FLIR measurement, to establish a 64-element shape function from the target plume intensity profile. The current FLIR data is correlated against the template of the previously stored shape function that has been centered on the FLIR image plane. The outputs of the algorithm are two linear offsets, x_c and y_c in Equations (3.1) and (3.2), that yield the highest correlation of the current data with the template. These "pseudo-measurements" are then fed to the linear Kalman filter for its update cycle. The filter provides the updated estimate, $\hat{x}(t_i^+)$, used to center the FLIR intensity profile to be included in the template generation for the next measurement.

For forming the next template, the current intensity function image is centered on the FLIR plane by translational shifts using centroid offset estimates from $\hat{x}(t_i^+)$, using the "shifting property" of the Fourier Transform, where negating phase shifts are applied in the spatial frequency domain to accomplish a translational shift in the original domain. Rather than perform the difficult correlation in the time domain, the Fourier domain allows one to apply multiplication to implement the "translational

shift" of the intensity functions and eventual correlation with the template. Exponential smoothing is then used to average the result with previously centered images to yield an updated template.

4.3.1.2 Template Generation. The template reconstructs the shape, size, and location of the intensity centroid using the raw noise-corrupted FLIR measurements. The template generation begins with an input of a FLIR frame of data to the enhanced correlator algorithm of Figure 4.1. Using the "shifting" property of the fast Fourier transform (FFT), which states that a translational shift in the spatial domain is equivalent to a linear phase shift in the frequency domain, the required phase shift is computed by:

$$F\{g(x - x_{shift}, y - y_{shift})\} = G(f_x, f_y)\exp\{-j2\pi(f_x x_{shift} + f_y y_{shift})\} \tag{4.23}$$

where:

$F\{\cdot\}$	=	Fourier transform operator
$g(x,y)$	=	2-dimensional spatial data array
$G(f_x,f_y)$	=	$F\{g(x,y)\}$
f_x,f_y	=	Spatial frequencies

The Fourier transform is implemented in the simulation software using the Cooley-Tukey algorithm [35]. The target plume intensity shape function is "centered on the FLIR plane" by phase shifting the transformed function an amount equal to:

$$x_{shift}(t_i) = \hat{x}_t(t_i^+) + \hat{x}_a(t_i^+) + (\hat{x}_p(t_i^+) - \hat{x}_o(t_i^+))\cos\hat{\theta}_f$$
$$y_{shift}(t_i) = \hat{y}_t(t_i^+) + \hat{y}_a(t_i^+) - (\hat{x}_p(t_i^+) - \hat{x}_o(t_i^+))\sin\hat{\theta}_f \tag{4.24}$$

where $\hat{\theta}_f$ is the filter's estimate of the velocity vector angle in the FLIR plane (recall Figure 3.7), such that:

$$\cos\hat{\theta}_f = \frac{\hat{v}_x}{\sqrt{\hat{v}_x^2 + \hat{v}_y^2}}$$
$$\sin\hat{\theta}_f = -\frac{\hat{v}_y}{\sqrt{\hat{v}_x^2 + \hat{v}_y^2}} \tag{4.24a}$$

where $\hat{x}_t$, $\hat{y}_t$, $\hat{v}_x$, $\hat{v}_y$, $\hat{x}_a$, $\hat{y}_a$, and $\hat{x}_p$ are the state estimates defined in Equation (4.1). Once the data is centered on the FLIR plane, it is incorporated into an updated template for the next sample period. In the simulation, the Kalman filter's first update cycle is bypassed to form the initial template.

The template is generated by averaging the N most recent centered intensity functions observed by the FLIR sensor. The averaging process tends to accentuate the target intensity function and attenuate the corrupting background and FLIR noises. The memory size N is chosen according to how rapidly the shape functions change, i.e., highly dynamic intensity functions require small values of N, while slowly varying functions use large N values. Typically, a true finite memory averager would require a large computer memory [17]. However, the enhanced correlator

algorithm circumvents the memory storage issue by incorporating an "exponential smoothing" technique to approximate the averaging. This technique has properties similar to finite memory averaging, but with the advantage of requiring only the storage of a single FLIR frame of data. The template is maintained by the exponential smoothing algorithm given by:

$$\hat{I}(t_i) = \gamma I(t_i) + (1 - \gamma)\hat{I}(t_{i-1}) \tag{4.25}$$

where:

$\hat{I}(t_i)$	=	"Smoothed estimate" (template) of the target's intensity function
$I(t_i)$	=	"Raw" intensity function from the current FLIR data frame
γ	=	Smoothing constant: $0 < \gamma \leq 1$

The smoothing constant, γ, is comparable to the value selected for N. From Equation (4.25), it can be seen that large values of γ emphasize the current data frame and correspond to small values of N. Bases on previous studies [14,40], a smoothing constant of $\gamma = 0.1$ is used for this thesis.

A reinitialization algorithm is used after the first ten sample periods (although it could be called periodically in actual implementation). Once the template is computed, its centroid is calculated and shifted to the center of the field-of-view for the template, thus eliminating any initial pointing biases. It is this template which

is now stored and correlated with the next FLIR data to produce the "pseudo-measurements" [43].

4.3.1.3 "Pseudo-Measurements." The template serves as the best estimate of the shape of the target plume intensity function prior to receiving a new FLIR data frame. The cross-correlation of the incoming FLIR data with the template provides the position offsets from the center of the FOV to the centroid of the target intensity image. The cross-correlation is computed by taking the inverse fast Fourier transform (IFFT) of the equation [35]:

$$F\{g(x,y) * l(x,y)\} = G(f_x,f_y)L^*(f_x,f_y) \tag{4.26}$$

where:

$F\{\cdot\}$	=	Fourier transform operator
$g(x,y)$	=	Measured target intensity function of the current FLIR data frame
$l(x,y)$	=	Expected target plume intensity function (i.e., template)
$g(x,y) * l(x,y)$	=	Cross-correlation of $g(x,y)$ and $l(x,y)$
$G(f_x,f_y)$	=	$F\{g(x,y)\}$
$L^*(f_x,f_y)$	=	Complex conjugate of $F\{l(x,y)\}$

After the IFFT is accomplished, the values of the correlation function, $g(x,y) * l(x,y)$, are modified such that any value less than 30% of the function's maximum value is set to zero [14,31]. This "thresholding" technique is used to eliminate false peaks in

the correlation function that occur due to noise and other effects. As shown earlier in Figure 4.1, the output of the image correlation is the offset of the "thresholded" FLIR intensity centroid from the center of the FLIR FOV. This offset is assumed to be the result of the summed effects of target dynamics, atmospheric jitter, the pogo effect, and measurement noise.

For the FLIR measurement update, the x- and y- components of the offsets are the pseudo-measurements provided to the Kalman filter. These offsets are expressed as:

$$\begin{aligned} x_{offset} &= x_t + x_a + (x_p - x_o)\cos\theta_f + v_{f1} \\ y_{offset} &= y_t + y_a - (x_p - x_o)\sin\theta_f + v_{f2} \end{aligned} \tag{4.27}$$

where:

$$\begin{aligned} \cos\theta_f &= \frac{v_x}{\sqrt{v_x^2 + v_y^2}} \\ \sin\theta_f &= \frac{-v_y}{\sqrt{v_x^2 + v_y^2}} \end{aligned} \tag{4.28}$$

and θ_f is the angle between the velocity vector and the x-axis of the FLIR plane (recall Figure 3.7). These two measurements can be represented in state space form as:

$$z(t_i) = h_f[\, x_f(t_i), t_i] + v_f(t_i) \tag{4.29}$$

where:

$z(t_i)$	=	$[\, x_{offset}(t_i), y_{offset}(t_i)\,]^T$; 2-dimensional vector measured in pixels
$h_f[x_f(t_i), t_i]$	=	Nonlinear measurement function vector given by Equation (4.27)
$x_f(t_i)$	=	9 x 1 state vector from Equation (4.1)
$v_f(t_i)$	=	2-dimensional, discrete-time, white Gaussian measurement noise (in pixels) with statistics:

$$E\{\, v(t_i)\,\} = 0$$

$$E\{\, v(t_i) v(t_j)^T \,\} = \begin{cases} R & t_i = t_j \\ 0 & t_i \neq t_j \end{cases} \tag{4.30}$$

Note that because of the pogo states and offset state, x_o, being defined along the velocity vector and being included in the output equations, this measurement model is nonlinear in the filter states, and the extended Kalman filter update cycle described in Chapter II (Equations (2.26) and (2.27)) must be applied. These update equations are:

$$\begin{aligned} K(t_i) &= P_f(t_i^-) H_f^T [H_f P_f(t_i^-) H_f^T + R_f]^{-1} \\ \hat{x}_f(t_i^+) &= \hat{x}_f(t_i^-) + K(t_i)\{\, z(t_i) - h_f[\hat{x}_f(t_i^-), t_i]\,\} \\ P_f(t_i^+) &= P_f(t_i^-) - K(t_i) H_f P_f(t_i^-) \end{aligned} \tag{4.31}$$

where:

$K(t_i)$	=	9 x 2 filter gain matrix
$P_f(t_i)$	=	9 x 9 filter covariance matrix
$h_f[x_f(t_i),t_i]$	=	2-dimensional nonlinear measurement function; Equation (4.29)
H_f	=	Linearized measurement matrix; Equation (2.26)
R_f	=	2 x 2 measurement noise covariance matrix; Equation (4.34)
$\hat{x}_f(t_i)$	=	9-dimensional estimated state vector; Equation (4.1)
$z(t_i)$	=	2-dimensional measurement vector; Equation (4.29)
(t_i^-)	=	Time instant just prior to measurements being incorporated at time t_i
(t_i^+)	=	Time instant just after measurements are incorporated at time t_i

The linearized FLIR measurement matrix H_f is given by:

$$H_f = \begin{bmatrix} 1 & 0 & H_{13} & H_{14} & 1 & 0 & H_{17} & 0 & H_{19} \\ 0 & 1 & H_{23} & H_{24} & 0 & 1 & H_{27} & 0 & H_{29} \end{bmatrix} \tag{4.32}$$

where:

$$H_{13} = \frac{\partial h_1[x,t_i]}{\partial x_3} = \left. \frac{x_4^2\,(x_7 - x_9)}{[x_3^2 + x_4^2]^{\frac{3}{2}}} \right|_{x = \hat{x}_f(t_i^-)} \tag{4.33a}$$

$$H_{14} = \frac{\partial h_1[x,t_i]}{\partial x_4} = \left. \frac{-x_3\,x_4\,(x_7 - x_9)}{[x_3^2 + x_4^2]^{\frac{3}{2}}} \right|_{x = \hat{x}_f(t_i^-)} \tag{4.33b}$$

$$H_{17} = \frac{\partial h_1[x,t_i]}{\partial x_7} = \frac{x_3}{[x_3^2 + x_4^2]}\Bigg|_{x = \hat{x}_f(t_i^-)} \tag{4.33c}$$

$$H_{19} = \frac{\partial h_1[x,t_i]}{\partial x_9} = \frac{-x_3}{[x_3^2 + x_4^2]}\Bigg|_{x = \hat{x}_f(t_i^-)} \tag{4.33d}$$

$$H_{23} = \frac{\partial h_2[x,t_i]}{\partial x_3} = \frac{-x_3\, x_4\, (x_7 - x_9)}{[x_3^2 + x_4^2]^{\frac{3}{2}}}\Bigg|_{x = \hat{x}_f(t_i^-)} \tag{4.33e}$$

$$H_{24} = \frac{\partial h_2[x,t_i]}{\partial x_4} = \frac{x_3^2\, (x_7 - x_9)}{[x_3^2 + x_4^2]^{\frac{3}{2}}}\Bigg|_{x = \hat{x}_f(t_i^-)} \tag{4.33f}$$

$$H_{27} = \frac{\partial h_2[x,t_i]}{\partial x_7} = \frac{x_4}{[x_3^2 + x_4^2]}\Bigg|_{x = \hat{x}_f(t_i^-)} \tag{4.33g}$$

$$H_{29} = \frac{\partial h_2[x,t_i]}{\partial x_9} = \frac{-x_4}{[x_3^2 + x_4^2]}\Bigg|_{x = \hat{x}_f(t_i^-)} \tag{4.33h}$$

The measurement noise $v_f(t_i)$, represents the combined corrupting effects of the spatially correlated background noise, the FLIR sensor noise, and the errors due to the FFT/IFFT processes. The covariance matrix, R_f, associated with this error is given by [8,28,35]:

$$R_f = \begin{bmatrix} 0.00436 & 0 \\ 0 & 0.00436 \end{bmatrix} \; pixels^2 \tag{4.34}$$

4.3.2 Doppler Measurement Model. The primary purpose of this research is the precise tracking of the missile hardbody and determination of its center-of-mass location in the presence of plume pogo. The basic premise underlying the dynamics modeling efforts is that the center-of-mass is located at a constant offset distance relative to an equilibrium point about which the intensity centroid oscillates due to the pogo effect. The offset distance is oriented angularly using the filter-estimated velocity in the FLIR image plane [3]. Figure 4.2 illustrates the geometry of estimating the offset distance and the dependence of the scan and offset computation upon the filter's estimates of the position and velocity of the intensity centroid immediately after the FLIR update. Note that Figure 4.2 depicts the ideal situation; in general, the filter estimates of the centroid position, velocity, and the orientation angle are not equal to the truth model values. A low-energy laser is scanned along the filters' estimate of the velocity vector, starting at the filters' centroid estimate. The missile hardbody reflections determine the center-of-mass as the midpoint of the line segment joining the two endpoints. The offset measurement delivered by the scan is a function of the constant offset plus translation of the centroid from its equilibrium point due to the pogo phenomenon.

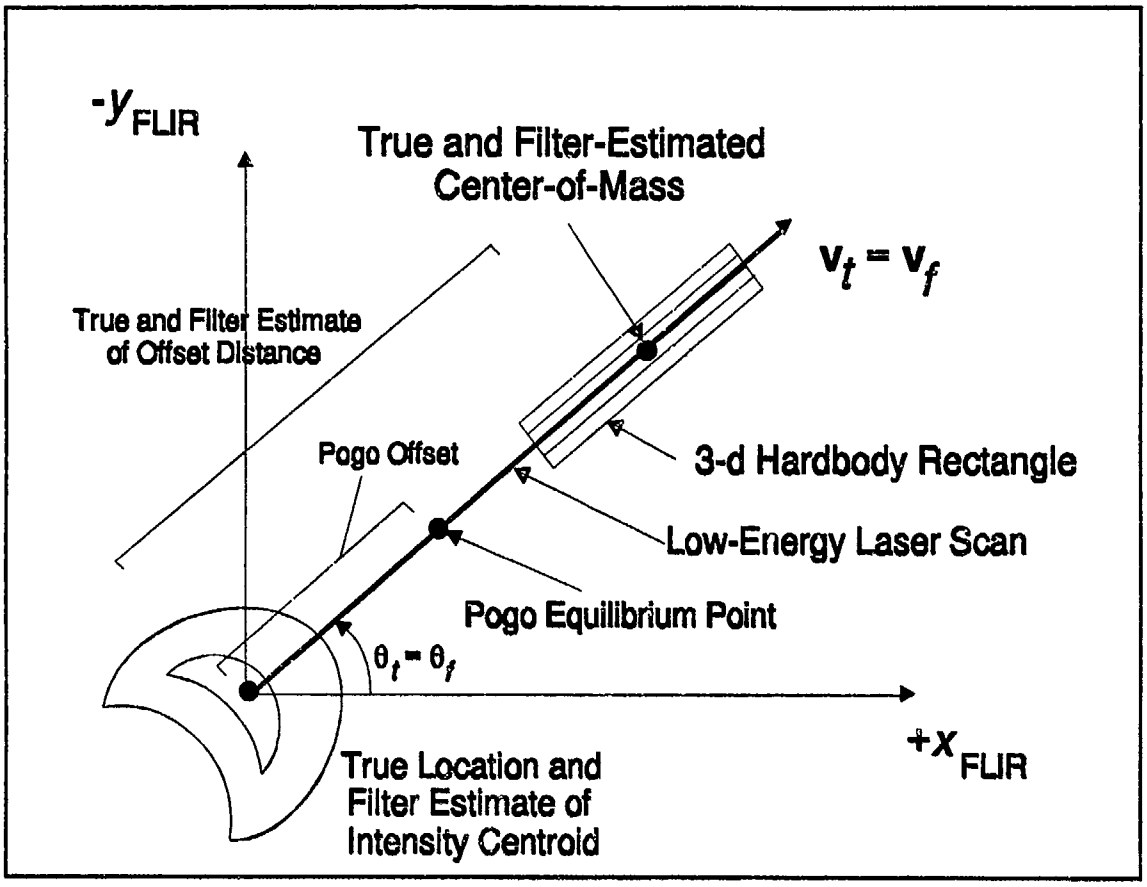


Figure 4.2 Filter Estimate of Offset Distance (Ideal Conditions)

Previous theses [3,4,9] utilized a two-filter approach in which the FLIR filter and center-of-mass filter functioned autonomously; the "FLIR" filter had no knowledge of the existence of the "center-of-mass" filter. Both Eden [3] and Evans [4] utilized the low-energy-laser speckle return of the hardbody/plume interface to generate measurements for the "center-of-mass" filter. Herrera [9] utilized the Doppler spectra in the laser return to derive successfully a more accurate, unbiased, offset measurement, but still maintained the same basic independent-filter structure. Ching [1] combined the "FLIR" and "center-of-mass" filters into the current nine-state

filter model. The Doppler measurement model used for this research was developed in previous these [1,9], and the model description is largely taken from these research reports.

The Doppler measurement model provides a measurement based upon the low-energy laser Doppler return of the hardbody. The significant dissimilarities between the plume and hardbody-induced Doppler returns can be exploited to discern the plume/hardbody interface (Section *3.3.3*) precisely, and provide information regarding the location of the hardbody. The low-energy laser measurement is provided to the Kalman filter whenever the laser intercepts the hardbody, and the hardbody-induced (and plume-induced) Doppler return is received by Doppler return sensor equipment. The resulting measurement to be provided to the filter is a noise-corrupted measurement of offset distance, which is a linear function of the filter's offset and pogo estimates. The measurement is given by:

$$x_{offset} = x_9 - x_7 \tag{4.35}$$

The discrete-time scalar measurement model is given by:

$$z(t_i) = H_o\, x(t_i) + v_f(t_i) \tag{4.36}$$

where:

$z(t_i)$	=	Measurement of the offset distance
H_o	=	1 x 9 measurement matrix
$x(t_i)$	=	9 x 1 state vector of the filter

$v_f(t_i)$ = Discrete-time, white Gaussian measurement noise with statistics:

$$E\{ v(t_i) \} = 0$$

$$E\{ v(t_i)v(t_j)^T \} = \begin{cases} R_f & t_i = t_j \\ 0 & t_i \neq t_j \end{cases} \tag{4.37}$$

where $R_f = R_t$ (true Doppler measurement variance), a function of low-energy laser wavelength, radar aperture, and signal-to-noise ratio (Section *3.3.3*). The measurement matrix, $\boldsymbol{H}_o$, is defined in accordance with Equation (4.35) as:

$$\boldsymbol{H}_o = [0\ 0\ 0\ 0\ 0\ 0\ -1\ 0\ 1] \tag{4.38}$$

Since the measurement is linear, the linear Kalman filter update cycle described in Section *2.2.2* (Equations (2.22) - (2.24)) is used.

In some instances, the low-energy laser sweep may be unsuccessful in generating a measurement, due to missing the hardbody because of poor estimation of the centroid location and velocity vector orientation, or due to poor conditions for discerning the differences in Doppler frequency spectra between the hardbody and plume. If a measurement is not generated by the LEL sweep, the LEL update is bypassed.

4.3.3 Filter Parameters. This section provides a consolidated reference of the parameters used for this simulation. Presented below are definitions of the modeling parameters, initial conditions, and tuning parameters for the nine-state filter used in this research.

4.3.3.1 Initial Conditions. The filter initial state estimate, $\hat{x}_0$, is artificially initialized to zero error for the position, velocity, and pogo states of Equation (4.1). The position states, x_1 and x_2, are initialized to the true center-of-mass with the target intensity centroid centered in the FLIR FOV. The velocity states, x_3 and x_4, are initialized in accordance with the target's initial trajectory conditions as defined in Section *3.4.1*. Both atmospheric states, x_5 and x_6, are initialized to zero. The offset of the plume from its equilibrium point (pogo position) and the velocity of this oscillation (plume pogo velocity) are initialized to zero since both are naturally zero-mean processes. The constant distance between the equilibrium point and the center-of-mass are also initialized to true conditions (87.5 meters, or 2.92 pixels).

The initial state covariance matrix, $P(t_0)$, is:

$$P(t_0) = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & .2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & .2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 500 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 250 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & .2 \end{bmatrix} \tag{4.39}$$

where the units of the covariance associated with the hardbody center-of-mass position states, x_1 and x_2, the atmospheric states, x_5 and x_6, the pogo position state, x_7, and the offset state, x_9, are pixels^2, and those of the center-of-mass velocity states, x_3

and x_4,and the pogo velocity state, x_8, are expressed in pixels^2/second^2 [4]. The new values in equation (4.39), 500 and 250, gave the best results during the simulation data runs.

4.3.3.2 Tuning Values. The measurement covariance matrix for the FLIR, R_{FLIR} was established empirically in past research [22,35]. R_{FLIR} (with units of pixels^2) is given by:

$$R_{FLIR} = \begin{bmatrix} 0.00363 & 0 \\ 0 & 0.00598 \end{bmatrix} \tag{4.40}$$

The measurement variance for the Doppler measurement, $R_{Doppler}$, is equal to the true measurement variance and is a function of the low-energy laser wavelength, SNR, and aperture diameter of the transmitter. The filter measurement variance is carried over from Herrera's research and is given by [9]:

$$R_{Doppler} = R_t = \left[\frac{\theta_B}{3k_p\sqrt{SNR}} \right]^2 \tag{4.41}$$

where:

$R_{Doppler}$	=	Filter measurement noise variance
R_t	=	True measurement noise variance
θ_B	=	Beam diffraction limit
k_p	=	Pixel proportionality constant; 15 μrads/pixel
SNR	=	Signal-to-noise ratio; 10

Both atmospheric variances, σ_{ax}^2 and σ_{ay}^2, are equal to 0.2 pixels^2, in accordance with the truth model (Section *3.2.2*) [4]. The process noise strength for the pogo state is dependent upon the desired rms amplitude of the pogo oscillation, σ_p^2, [34]. The filter pogo gain constant, K_{pf}, is initially set equal to the truth pogo gain constant, K_p (Section *3.2.4*, Equation (3.38)) and then adjusted if necessary while leaving the truth noise strength constant. Nominal rms pogo amplitude for this research was carried over from Rizzo's research and set to $\sigma_p^2 = 0.112$ pixels^2 at a frequency of $\omega_p = 1$ Hz. The offset state dynamics noise variance, Q_{do} from Section *4.2.4*, is equal to 0.9 pixels^2, based upon Evans' research [4]. The probability of miss for the Doppler measurement model was set at 0.01 (Section *3.3.3.4*)

4.4 Summary

A nine-state extended Kalman filter is used in this research to investigate its performance in tracking the hardbody center-of-mass from a large distance, in the presence of vehicle dynamics, atmospheric jitter, and plume pogoing effects. This chapter has described this extended Kalman filter's linear internal dynamics model dealing with the propagation of the state estimate, $\hat{x}$. The two different measurement models, FLIR and low-energy laser Doppler model, were discussed in the context of their role in the update process. These two distinct updates are accomplished in a method in which the FLIR 2 x 1 update is processed and then the scalar laser update is performed if there is a laser "hit" on the hardbody center-of-mass. The last section provided a summary of the initial conditions used for this extended Kalman filter algorithm.

V. *Elemental Filter and MMAE Results/Analysis*

5.1 Introduction.

The software simulation of the real world, measurement generation, enhanced correlator algorithm and various Kalman filters have been developed at AFIT over the last 15 years [31]. This thesis is the continuation of the overall research effort into this missile tracking problem. The linear Kalman filter used in conjunction with an enhanced correlator algorithm provides a viable tracking algorithm that is used in different configurations that further our knowledge in this MMAE tracking area.

Figure 5.1 shows the block diagram used to explore the parameter space of amplitude and frequency of oscillation of the plume pogo phenomenon. Section *5.2* details the single-filter implementation and performance analysis of this block diagram, with the follow-on thought processes used for MMAE configuration choices. The insight gained from the analysis of Section *5.2* is used to form several MMAE configurations. Results and performance analysis of the computer runs from these various configurations is accomplished in Section *5.3*.

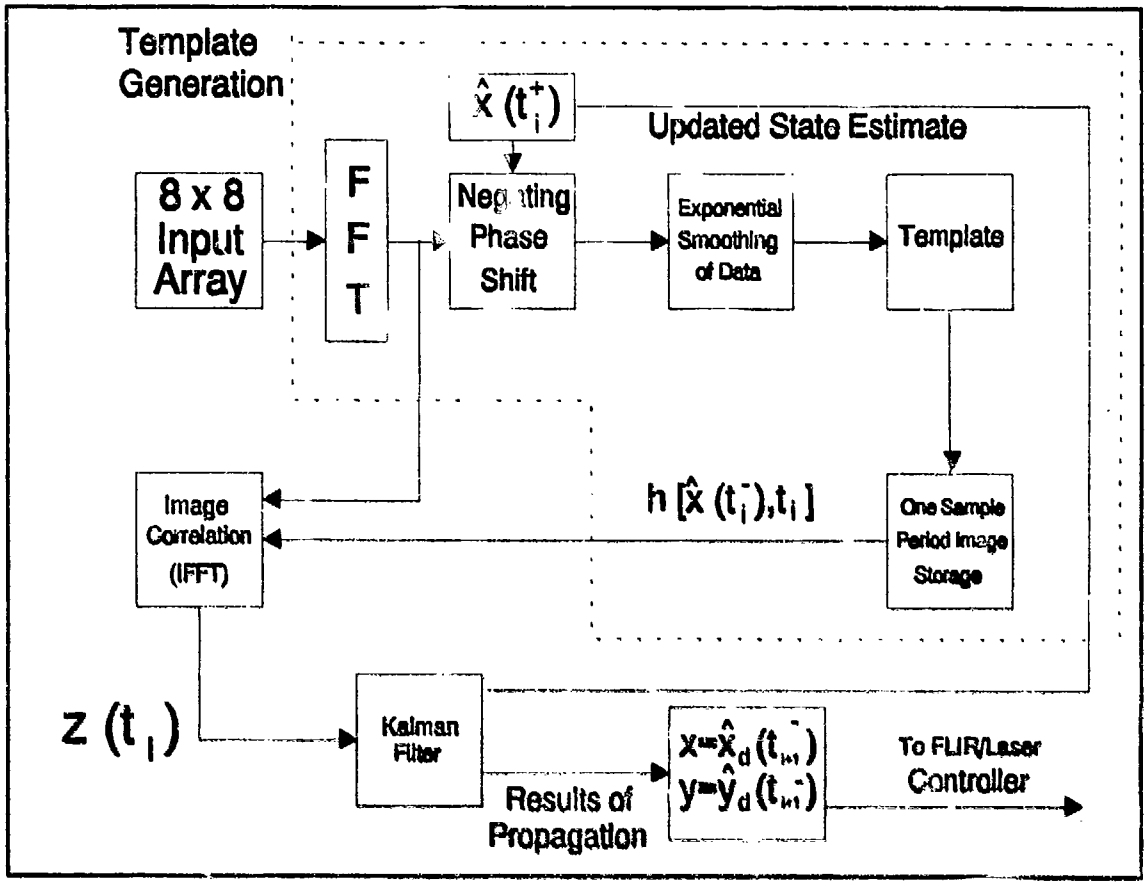


Figure 5.1 Single Kalman Filter Block Diagram Implementation

5.2 *Elemental Kalman Filter Simulation Results/Analysis*

5.2.1 Introduction. The purpose in accomplishing the single-filter data runs is to ensure that a working Kalman filter (KF) model is being used for the foundation of the elemental filters that make up the MMAE structure. This working model should be tested at all of the parameter conditions to ensure that divergence of the filter is not seen. Analysis of these computer runs will help in evaluating data performance and will give insight into the choice of the various MMAE

configurations. Comparison of the single filter residuals will also help in the prediction of success or failure of the different MMAE configurations.

Table 5.1 introduces the parameter values for the different elemental filter data simulations and the Appendix locations for the output plots. All of these data runs are made up of 5 *Monte Carlo* (MC) simulations. These plots show the

Truth Condition (Amplitude/Frequency)	Filter Condition (Amplitude/Frequency)	Appendix Location for Output Plots
0.112/2π	0.112/2π	C.1 - C.9
	0.112/20π	C.10 - C.12
	1.12/2π	C.13 - C.15
	1.12/20π	C.16 - C.18
0.112/20π	0.112/20π	D.1 - D.9
	0.112/2π	D.10 - D.12
	1.12/20π	D.13 - D.15
	1.12/2π	D.16 - D.18
1.12/2π	1.12/2π	E.1 - E.9
	1.12/20π	E.10 - E.12
	0.112/2π	E.13 - E.15
	0.112/20π	E.16 - E.18
1.12/20π	1.12/20π	F.1 - F.9
	1.12/2π	F.10 - F.12
	0.112/20π	F.13 - F.15
	0.112/2π	F.16 - F.18

Table 5.1 Single Kalman Filter Truth and Filter Parameter Values

error in each filter state plotted in pixels versus time (60 samples per second). Statistics from the MC runs, mean error and standard deviation of the mean error, are compiled at times t_i^- and t_i^+, just prior to and just after the update measurement at each sample period. The chosen pogo amplitude values of 0.112 and 1.12 pixels in Table 5.1 (approximately 3 and 30 meters at the range of 2,000 kilometers) make physical sense, as do the pogo frequencies of 2π and 20π (1 and 10 Hertz). These tabular results are contained throughout this chapter.

5.2.2 Single-Filter Data Runs. The single-KF data simulations were performed with the fact that a KF that is artificially informed of the correct truth condition will provide the best results of any possible scenario. The parameter space defined in Figure 5.2 is used to describe the combinations of data runs. The point in Figure 5.2 that is not shown in Table 5.1, $0.112/11\pi$, was added after insight was gained from performance of the single-filter data runs. Several 5-run *Monte Carlo* (MC) simulations are made for the 5 single-filter configurations defined in Table 5.1. On the first set of MC runs, the KF is artificially informed of the correct truth condition for the purpose of developing a "best scenario" baseline set of data. The MC simulations in which the filter has the "incorrect" parameter values are performed for three reasons. First, it is necessary to have non-divergent elemental filters running in this MMAE scheme. There is no built-in adaptation process for a divergent filter. The second reason is for parameter identification purposes. The filters that show the best residuals should be those that are programmed with the

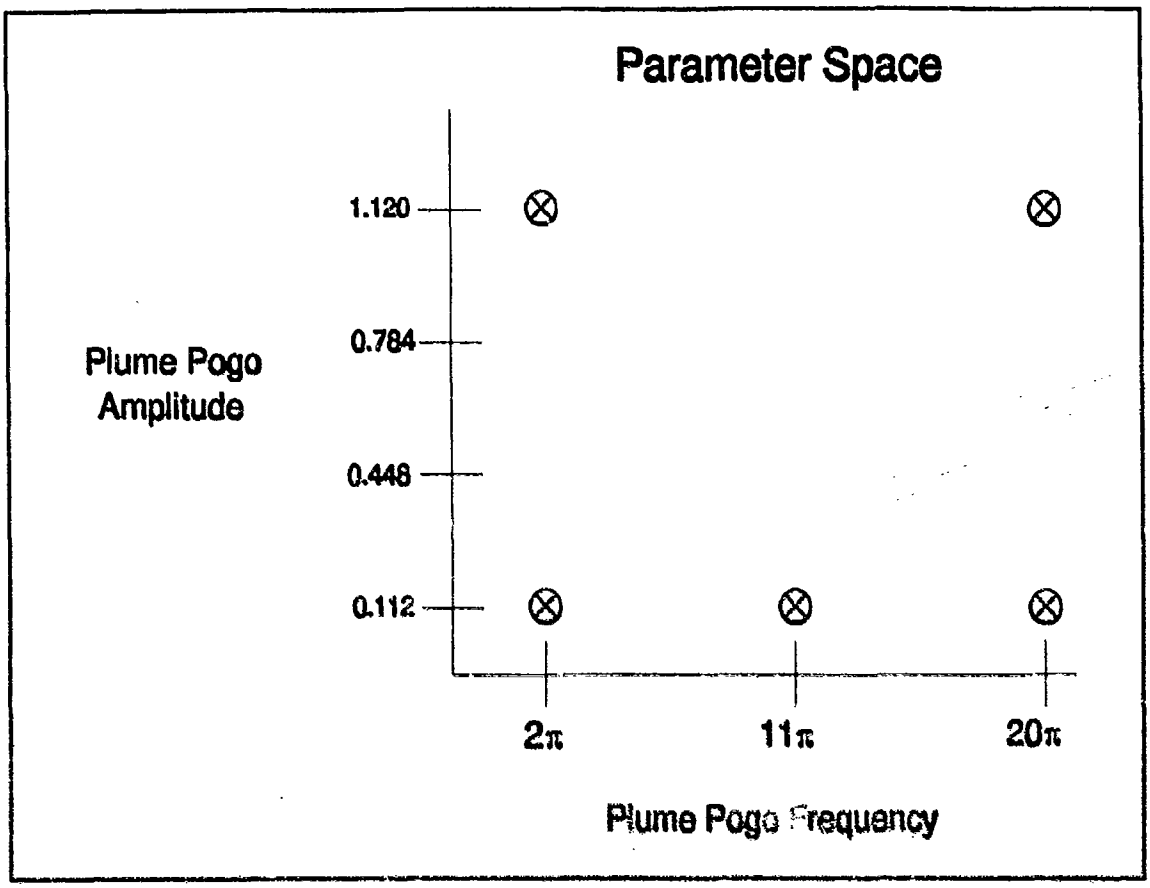


Figure 5.2 Graphical View of Parameter Space

parameter values that best match the real world conditions that the filter is tracking. Comparison of the output residuals from these single runs will be used to choose the MMAE configuration with the best chance of parameter identification success. The third reason is to have a "best case" set of filter performance data which can be compared later to the different MMAE configurations.

The tuning of these Kalman filters is a critical issue, as there is a balance that must be met over the range of these potential elemental filters. On the one hand, the

low dynamic filters (low amplitude and low frequency pogo parameter values) cannot be tuned so tightly as to cause divergence when used against a higher dynamic truth model. Also, the high dynamic filters cannot be so conservatively tuned as to cause the low-pogo-dynamics filters to attribute errors wrongly from one state to another, thus giving false outputs. To be as effective as possible, tight tuning should be used on all of the states for all of the filters.

The next five sections, which provide an indepth analysis of the plots of Appendices C through F, show that the five chosen elemental filters work well. They behave as they should, given the input filter parameter values, and indeed provide a working elemental filter model for use in a Multiple Model Adaptive Estimator.

5.2.2.1 Truth Parameter Values: Low Amplitude, Low Frequency. This section describes the cases in which the truth parameter values are set to an amplitude of 0.112 pixels and a frequency of 2π radians per second, with various filter parameter settings. The filter parameter values are $0.112/2\pi$ (filter is "artificially" informed of the correct truth condition), $0.112/20\pi$, $1.12/2\pi$, $1.12/20\pi$, and $0.112/11\pi$.

The first set of *Monte Carlo* (MC) data runs has the truth parameter values equaling the filter parameter values, set to $0.112/2\pi$, corresponding to the lower left point of Figure 5.2. The output error statistics are detailed in Table 5.2. These error

State	Mean Error (t_i^-)	Mean Error (t_i^+)	STD. DEV. $\sigma(t_i^-)$	STD. DEV. $\sigma(t_i^+)$
X Position*	0.060	0.061	0.427	0.401
Y Position*	0.072	0.074	0.341	0.313
X Velocity	0.010	0.006	2.791	2.789
Y Velocity	0.058	0.048	2.593	2.586
X Jitter	0.024	0.026	0.428	0.399
Y Jitter	0.023	0.020	0.473	0.458
Pogo Position#	0.004	0.004	1.756	0.176
Pogo Velocity#	0.032	0.031	1.100	1.100
Offset Distance#	0.119	0.120	0.514	0.515
X Centroid	0.004	0.003	0.458	0.444
Y Centroid	0.173	0.176	0.576	0.565

Table 5.2 Time-Averaged Error Statistics for Elemental Filter
(Truth = $0.112/2\pi$, Filter = $0.112/2\pi$)

*X and Y Hardbody Center-of-Mass Position
#Measured in the Direction of the Velocity Vector

statistics are compiled as steady-state statistics from $t = 2$ to 6.5 seconds (60 samples per second). The residuals from this data run are plotted as three separate measurements (2 FLIR measurements, one each in the x- and y- directions, and one low-energy laser measurement).

The filter error plots of Appendix C, Plots C.1 through C.8, show that there is no divergence in any of the states. Plot C.1a (Hardbody center-of-mass position state) shows that the filter- predicted plus/minus one standard deviation is very close to the actual plus/minus one standard deviation. The filter-versus-actual error plot is close

to being zero-mean. The more *Monte Carlo* (MC) runs that are made the "flatter" this plot would be. The error in filter state 2 (y- position of the hardbody center-of-mass), plot C.1b, is similar to C.1a in that the filter-versus-actual error is zero-mean and the filter-predicted plus/minus one sigma is close to the filter-versus-actual plus/minus one sigma. Plots C.2a and C.2b show the error in filter states 3 and 4 (velocity in x- and y- direction of hardbody center-of-mass). This error again is zero-mean, but the plus/minus one sigma of the filter-versus-actual error is well within the filter-predicted plus/minus one sigma; peculiar to these two filter states. When the tuning of the filter was so tight as to drive the filter-predicted plus/minus one sigma to meet the filter-versus-actual error plus/minus one sigma, divergence was seen on the position states.

Filter states 5 and 6 (x- and y- atmospheric states) of plots C.3a and C.4a show the mean values of the actual filter and truth data plotted. The filter does a good job in following the actual atmospherics, as is seen by the close proximity of the filter and truth plots. The filter-versus-actual error for this data, plotted in C.3b and C.4b, show that the error is very close to being zero-mean, considering the relatively small number of MC runs. Also, the filter-predicted plus/minus one sigma is close to the filter versus-actual-error plus/minus one sigma. The tuning is tight for these states, as is seen in the plus one sigma of the filter is actually inside the plus one sigma for the filter versus actual error. The actual rms jitter error is 0.2 pixels, so these errors are substantial compared to the size of the jitter. Note, the filter-predicted one sigma

seen in all of the plots that show the filter-versus-actual error is a standard saw-toothed pattern as anticipated for a propagate/update cycling of a filter, but it is sampled (plotted) less frequently than the 60 Hertz sample rate and so has a strange appearance in the plots.

The filter states 7 and 8 (Pogo position and velocity in the direction of the velocity vector) are shown with the filter state data plotted against the truth model in C.5a and C.6a. Since the pogo amplitude is small the filter has a hard time of precise tracking for this phenomenon. In plots C.5b and C.6b, the filter-versus-actual error is plotted against the filter-predicted plus/minus one sigma. For both of these cases, the filter-predicted plus/minus one sigma is outside of the filter-versus-actual error plus/minus one sigma.

Filter state 9 (Offset distance between the plume centroid equilibrium point and the hardbody center-of-mass) is plotted versus the truth data in C.7a and the filter-versus-actual error plotted in C.7b. These plots show that the filter does a pretty good job of estimating this state (error plot is approximately zero-mean). The filter-predicted plus/minus one sigma is close to the filter-versus-actual error plus/minus one sigma.

Plots C.8a and C.8b are the filter versus actual error of the x- and y- centroid position (not an actual filter state but a linear combination of filter states). Plot C.8a

shows zero-mean but the plot of C.8b shows a slight bias for the y- centroid position. This is not really an important error, as the filters' main purpose is to track the hardbody center-of-mass, not just the plume intensity centroid.

The plots labeled C.9 in Appendix C show the three different residuals with their plus/minus one standard deviation. These plots show the residuals to be zero-mean, given the 5-MC run simulation. The scaling of these residual plots is large so as to allow easy comparison with later plots.

For the sake of brevity, only the plots from the different 5-run MC simulations that have a direct bearing on this analysis will be shown from this point on. Also the error statistics of Tables 5.2 and 5.3 that are compiled for each of the simulations would unduly expand this document; thus, only the error statistics that show trends will be shown. Tables 5.3, 5.5, 5.7, and 5.9 show a comparison of the Pogo Position and Pogo Velocity errors for all the simulations of each of the Sections *5.2.2.1-4*.

The filter parameter values are changed to $0.112/20\pi$ (Far right point along the constant 0.112 line in Figure 5.2) to produce plots C.10 - C.12. The pogo position state 7 plots (C.10 for filter $0.112/20\pi$; C.5 for filter $0.112/2\pi$) reveal the differences between the two different data runs. The filter bounces back and forth over the low amplitude, but really doesn't track the "real world" low frequency pogo at all. The scaling of plot C.10b is such that the filter is looking for a high frequency signal and

the transients have not completely dissipated before the plotting routine starts (this plotting routine starts at the fifth sample point). The plot of filter state 8 (pogo velocity), C.11a and C.11b (note the y-axis scales), shows a large difference in the pogo velocity during this data run as compared to the data run of Plot C.6a and C.6b. When the pogo frequency is changed in the filter, such that there is a mismatch between truth and model pogo frequency and filter-assumed frequency, the filter loses its ability to track this state. The plots of filter states 1, 2, 3, 4, 5, 6, and 9 are "close" to what is seen in Plots C.1,2,3,4,7, and 8 and are not shown.

Error statistics analysis of Tables 5.2 and 5.3 show that the two data runs are close for the center-of-mass position and velocity states, the atmospheric states, the offset state, and the centroid statistics. The pogo position of the high frequency filter is better than the artificially informed filter, even though the tracking is obviously off (Plot C.10). The most volatile state for this simulation is the pogo velocity state, which has a much greater error in the high frequency filter, due to the erroneous filter-assumed pogo frequency (and note that the velocity is not updated directly with a measurement, unlike the pogo position). The rms error is almost five times as great for this high frequency filter (Pogo position and velocity data are summarized in Table 5.4).

State	Mean Error (t_i^-)	Mean Error (t_i^+)	STD. DEV. $\sigma(t_i^-)$	STD. DEV. $\sigma(t_i^+)$
X Position*	0.061	0.062	0.429	0.403
Y Position*	0.071	0.072	0.335	0.309
X Velocity	0.032	0.027	0.276	2.763
Y Velocity	0.037	0.025	2.478	2.472
X Jitter	0.021	0.022	0.427	0.396
Y Jitter	0.025	0.023	0.469	0.452
Pogo Position#	0.005	0.005	0.133	0.137
Pogo Velocity#	0.064	0.056	4.531	4.558
Offset Distance#	0.040	0.043	0.541	0.541
X Centroid	0.039	0.038	0.446	0.434
Y Centroid	0.110	0.115	0.595	0.596

Table 5.3 Time-Averaged Error Statistics for Elemental Filter
(Truth = $0.112/2\pi$, Filter = $0.112/20\pi$)

*X and Y Hardbody Center-of-Mass
#Measured in the Direction of the Velocity Vector

The filter parameter values are then changed to $1.12/2\pi$ (Point at the upper left corner in Figure 5.2) to produce the data runs shown in Plots C.13-16. Again, the pogo position and velocity states are different in the "high" amplitude filter. In plots C.13b and C.14b, the errors don't "settle" as much as the errors in Plots C.5b and C.6b. This result is analytically seen from the comparison of the pogo position and velocity states of Table 5.4. The errors from this new filter are approximately 35% greater in the pogo position state and approximately 75% greater in the pogo velocity state. Comparison of the residual output graphs of C.15 (new filter erroneously based

Filter Parameter Values	Pogo Position Error $\sigma(t_i^+)$*	Pogo Velocity Error $\sigma(t_i^+)$*
0.112/2π (Filter = Truth)	0.176	1.100
0.112/20π	0.137	4.558
1.12/2π	0.236	1.943
1.12/20π	0.285	16.471

Table 5.4 Summary of the Pogo Position and Pogo Velocity Errors for Truth Condition of 0.112/2π with Various Filter Parameter Values

*Errors at time, t_i^-, were very similar to the error at time, t_i^+, for all simulations.

on higher amplitude pogo assumption) and C.9 (correct filter/truth model agreement on pogo amplitude) show almost no difference. If anything, the residuals of the "incorrect" filter (C.15) are better than the "correct" filter residuals (C.9).

Again, the output graphs of filter states 1, 2, 3, 4, 5, 6, and 9 and the x- and y centroid error are similar enough not to be shown. It should be noted that, for this simulation, the "incorrect" filter has about a 10% smaller error on the y- direction center-of-mass state. This possibly accounts for the laser residual seeming to be slightly better for the "incorrect" filter versus the "correct" filter.

The filter parameter values are then changed to 1.12/20π (Upper right point in Figure 5.2) to produce the output plots of C.16 - C.18. The pogo position filter state (C.16a) and the pogo velocity filter state (C.17a) have been misinformed of both the truth condition amplitude and frequency. Comparison of statistical error values

of these states in Table 5.4 bear this out. The velocity state error is particularly obtuse as the error for the "incorrect" filter is approximately 15 times the error for the correct filter. Note the pogo velocity errors from Table 5.4; incorrect frequency seems much more critical than the incorrect amplitude assumption. Close analysis of the residual plots (C.18 for "incorrect" filter, C.9 for the "correct" filter) show that the "incorrect" filter residuals are slightly worse. The difference is almost imperceptible.

5.2.2.2 Truth Parameter Values: Low Amplitude, High Frequency. The filter is artificially informed of the correct truth conditions, an amplitude of 0.112 pixels and a frequency of 20π radians per second, to produce the output filter state plots, D.1 through D.9, of Appendix D. These filter state error plots will not be as rigorously analyzed as was done in Section *5.2.2.1*; just the important trends will be explored. The error plots of the hardbody center-of-mass position and velocity states (Filter states 1-4 plotted in Figures D.1a, D.1b, D.2a, and D.2b) and the error plots for the two atmospheric states (Filter states 5 and 6 plotted in Figures D.3b and D.4b) are zero-mean with well-behaved plus/minus one sigma characteristics (filter-predicted one-sigma is close to the filter-versus-actual error plus/minus one sigma). The pogo position and velocity states (Filter states 7 and 8 plotted in Figures D.5 and D.6) show that the filter tracks the true pogo position and velocity pretty well. Plots D.5a and D.6a show this result clearly, as the filter mean values match the true conditions. Filter state 9, the offset distance between the plume centroid equilibrium

point and the hardbody center-on-mass, is plotted in Figure D.7, where the filter oscillates over the true condition of 2.92 pixels. The error plot of D.7b is zero-mean and non-divergent, with a plus one-sigma that is outside of the filter-predicted one-sigma. This is again due to the tight filter tuning. The linear combination centroid error plots, D.8a and D.8b, are both zero-mean. Table 5.5 shows the explicit error statistics for all of the filter states.

The filter parameter values are changed to 0.112/2π to produce a full set of error plots for all of the filter states. For brevity, only the plots that emphasize

State	Mean Error (t_i)	Mean Error (t_i^*)	STD. DEV. $\sigma(t_i)$	STD. DEV. $\sigma(t_i^*)$
X Position*	0.031	0.031	0.444	0.418
Y Position*	0.056	0.058	0.327	0.304
X Velocity	0.034	0.028	2.795	2.792
Y Velocity	0.038	0.025	2.499	2.494
X Jitter	0.022	0.023	0.428	0.398
Y Jitter	0.025	0.023	0.468	0.454
Pogo Position#	0.002	0.002	0.095	0.096
Pogo Velocity#	0.063	0.055	6.262	6.208
Offset Distance#	0.021	0.023	0.514	0.514
X Centroid	0.021	0.019	0.447	0.435
Y Centroid	0.076	0.081	0.578	0.579

Table 5.5 Time-Averaged Error Statistics for Elemental Filter
(Truth = 0.112/20π, Filter = 0.112/20π)

*X and Y Hardbody Center-of-Mass
#Measured in the Direction of the Velocity Vector

an important trend will be shown. The new "incorrect" (the filter assumes a low frequency for the pogo) low frequency plots of filter states 7 and 8 (Pogo position and velocity states) are plotted in Figures D.10 and D.11. The low amplitude/low frequency filter does not do a good job of precise tracking of the position of this low amplitude/high frequency truth condition (Plot D.11a). The pogo position error is almost twice the position error of the "correct" filter. This is also seen from comparison of the error statistics of Table 5.6. The pogo position and velocity state error for the "artificially" informed filter and the various other filter parameter values of this section are summarized in Table 5.6. The pogo velocity state error, which is the usual problem, is within reasonable tolerance. There is little difference between the residual output plots of D.9 ("Correct" filter) and D.11 ("Incorrect" filter).

Filter Parameter Values	Pogo Position Error $\sigma(t_i^+)$*	Pogo Velocity Error $\sigma(t_i^+)$*
0.112/20π (Filter = Truth)	0.096	6.208
0.112/2π	0.165	6.472
1.12/20π	0.269	16.560
1.12/2π	0.250	6.800

Table 5.6 Summary of the Pogo Position and Pogo Velocity Errors for Truth Condition of 0.112/20π with Various Filter Parameter Values

*Errors at time, t_i, were very similar to the error at time, t_i^+, for all simulations.

The filter values are changed to $1.12/20\pi$ to produce a full set of plots, but just the pogo position and velocity filter state plots will be shown, as the others are similar to the plots from the "correct" filter. The pogo position and pogo velocity (Figures D.12a and D.13a) plots show that the filter knows the frequency but is missing the pogo amplitude (as expected from the programmed filter parameter values). The comparison of the error statistics of Table 5.6 also show this, as both states have more than double the error standard deviations. There is a slight, barely perceptible difference in the FLIR (2) residual of D.9 ("Correct" filter) and D.14 ("Incorrect" filter).

The filter values are changed to $0.112/2\pi$ (Both values are different from the truth values) to produce Appendices D.15 - D.17. The " incorrect" filter is off in both frequency and amplitude for both the pogo position and velocity states. The errors of Table 5.6 are not as different from the "correct" filter statistical errors as might be expected. The position state errors are still off by a factor of 2.5, but the velocity state error is close. The residual plots are again very close with no perceptible difference.

5.2.2.3 Truth Parameter Values: Large Amplitude, Low Frequency. The plots for the filter parameter values that equal the truth conditions are located in Plots E.1 through E.9 of Appendix E. Filter states 1 and 2 (Figure E.1) show some excursions from the filter-predicted plus/minus one sigma as the greater dynamic conditions

come into play, but no divergence is seen. The FLIR x- and y- velocity states are contained within the filter-predicted plus/minus one sigma (Figure E.2). The atmospheric states of E.3 and E.4 are very similar to what is seen throughout this chapter. The pogo position and velocity states are much more interesting with the higher amplitude. The filter pogo follows the truth pogo very well in both the position and velocity states (E.5a and E.6a), with both of their error plots being zero-mean with the filter-predicted one-sigma near the filter-versus-actual error plus/minus one sigma. The filter does a nice job of predicting the offset state 9 (Figure E.7), with this error also being zero-mean. The centroid prediction plots are also zero-mean, with some excursions from zero due to the higher dynamics. The statistical errors for all of the filter states of this "artificially" informed filter are summarized in Table 5.7.

The filter is then set to $1.12/20\pi$ to produce a full set of charts, with only the interesting plots shown. The pogo position plot of E.10 and pogo velocity plot of E.11 show that, as expected, the filter cannot track the slow amplitude truth condition. The filter residual outputs of E.9 ("Correct" filter) and E.12 ("Incorrect" filter) show a now visual difference in the residuals for these two simulations.

State	Mean Error (t_i^-)	Mean Error (t_i^+)	STD. DEV. $\sigma(t_i^-)$	STD. DEV. $\sigma(t_i^+)$
X Position*	0.099	0.099	0.763	0.756
Y Position*	0.061	0.062	1.207	1.186
X Velocity	0.042	0.040	2.851	2.843
Y Velocity	0.018	0.008	3.408	3.405
X Jitter	0.019	0.020	0.435	0.410
Y Jitter	0.023	0.021	0.475	0.452
Pogo Position#	0.045	0.045	0.500	0.491
Pogo Velocity#	0.011	0.019	3.893	3.815
Offset Distance#	0.066	0.069	0.674	0.675
X Centroid	0.046	0.045	0.756	0.748
Y Centroid	0.160	0.163	1.186	1.162

Table 5.7 Time-Averaged Error Statistics for Elemental Filter
(Truth = $1.12/2\pi$, Filter = $1.12/2\pi$)

*X and Y Hardbody Center-of-Mass
#Measured in the Direction of the Velocity Vector

The higher dynamics has induced larger errors when the filter is using an incorrect parameter value. Filter states 1-3 have errors of 0.956, 1.353, and 3.431 pixels (compared to the values of 0.756, 1.186, and 2.843 in Table 5.7), which are approximately 20% higher than the errors for the simulation in which the truth and filter parameter values are equal. The largest errors are still seen in the pogo position and velocity states, as shown in Table 5.8. Table 5.8 summarizes the pogo position and velocity errors for all of the various filter parameter values for the truth condition of $1.12/2\pi$. Again, note the large pogo velocity errors for the filters which assume a wrong pogo frequency.

Filter Parameter Values	Pogo Position Error $\sigma(t_i^+)$*	Pogo Velocity Error $\sigma(t_i^+)$*
$1.12/2\pi$ (Filter = Truth)	0.491	3.815
$1.12/20\pi$	1.120	24.967
$0.112/2\pi$	1.130	6.957
$0.112/20\pi$	1.157	11.465

Table 5.8 Summary of the Pogo Position and Pogo Velocity Errors for Truth Condition of $1.12/2\pi$ with Various Filter Parameter Values

*Errors at time, t_i^-, were very similar to the error at time, t_i^+, for all simulations.

The new filter parameter values are set at $0.112/2\pi$. Again the most interesting output plots are the pogo position and velocity states, which are located in E.13 - E.15. The small amplitude filter does not keep up with the truth condition in either the position or velocity graphs, as expected. There is a very slight difference in output residual plots of E.9 ("Correct" filter) and E.15 ("Incorrect" filter). The error statistics, summarized in Table 5.8, show twice the error on the pogo position state and slightly below twice the error on the pogo velocity state.

The next filter parameter values are $0.112/20\pi$ (both values different from the truth conditions), which yield plots that are shown in Figures E.16 - E.18. This small amplitude, high frequency filter is unable to track the true pogo position and velocity states (E.16a and E.17a). The error statistic comparison of Tables 5.8 also shows this error. The FLIR residual plots of E.9 ("Correct" filter) and D.18 ("Incorrect" filter) show a slightly perceptible difference in the FLIR residuals, with the laser residual having a visible difference.

5.2.2.4 Truth Parameter Values: Large Amplitude, High Frequency. The full set of filter plots for the simulation in which the filter conditions match the truth conditions are shown in Figures F.1 through F.9 of Appendix F. The filter states 1 and 2 plots of F.1 show the very tight tuning and yet no divergence. The pogo position and velocity plots show the filter keeping up with the truth condition in amplitude and phase (F.5a and F.6a). Even in this extremely highly dynamic environment, the filter has a good estimate of the offset distance (F.7). The error statistics are shown in Table 5.9 for comparison purposes.

State	Mean Error (t_i)	Mean Error (t_i^*)	STD. DEV. $\sigma(t_i)$	STD. DEV. $\sigma(t_i^*)$
X Position*	0.054	0.053	0.691	0.679
Y Position*	0.052	0.050	0.990	1.003
X Velocity	0.116	0.108	2.51	2.951
Y Velocity	0.084	0.073	2.448	2.446
X Jitter	0.012	0.010	0.422	0.388
Y Jitter	0.033	0.033	.0471	0.452
Pogo Position#	0.017	0.022	0.491	0.393
Pogo Velocity#	0.562	0.248	33.753	29.242
Offset Distance#	0.110	0.111	0.433	0.434
X Centroid	0.012	0.016	0.646	0.653
Y Centroid	0.058	0.065	0.941	0.994

Table 5.9 Time-Averaged Error Statistics for Elemental Filter
(Truth = 1.12/20π, Filter = 1.12/20π)

*X and Y Hardbody Center-of-Mass
#Measured in the Direction of the Velocity Vector

The filter values are set at the different conditions of: $1.12/2\pi$, $0.112/20\pi$, and $0.112/2\pi$, with the pogo position and velocity states plotted in Figures F.10 - F.12 (Filter = $1.12/2\pi$), F.13 - F.15 (Filter = $0.112/20\pi$), and F.16 - F.18 (Filter = $0.112/2\pi$). The same trends that were seen in the preceding sections where the pogo position and velocity states inadequately track the truth condition are seen in the graphical plots of Figures F.10 - F.18 and in the compiled error statistics of Table 5.10 for this extreme, highly dynamic case. Comparison of the filter residual of F.9 ("Correct" filter) versus the FLIR residuals of F.12, F.15, and F.18 ("Incorrect" filters) yields a major point of this research, as there is a discernible difference in residuals of the filter with the correct versus incorrect parameter values. This information is used for the choices of the MMAE configurations.

Filter Parameter Values	Pogo Position Error $\sigma(t_i^+)$*	Pogo Velocity Error $\sigma(t_i^+)$*
$1.12/20\pi$ (Filter = Truth)	0.393	29.242
$1.12/2\pi$	1.026	65.211
$0.112/20\pi$	0.697	45.810
$0.112/2\pi$	1.044	64.284

Table 5.10 Summary of the Pogo Position and Pogo Velocity Errors for Truth Condition of $1.12/20\pi$ with Various Filter Parameter Values

*Errors at time, t_i^-, were very similar to the error at time, t_i^+, for all simulations.

5.2.3 Selection of Candidate MMAE Configurations. The criterion for selection of the different test MMAE configurations is that the appropriate discretization of the parameter space is based upon the residuals of the respective single-filter simulations. The residuals of two filters (one filter's parameter values being equal to the truth parameter values, the other having parameter values equal to one of the other proposed elemental filters) are compared at a certain truth condition to see if there is a difference in their residuals. These residuals will be used in the probability calculations for each of the respective filters within the MMAE algorithm.

The residuals from the filters on the constant 1.12 amplitude line (Frequency = 2π, 20π) are visibly different at each of the respective truth conditions ($1.12/2\pi$, $1.12/20\pi$) and thus are candidates for elemental filters in the MMAE configuration. At the lower amplitude truth conditions, none of the residuals are different enough from each other to include or discard either of the two low amplitude elemental filters automatically. This leads to three different configurations: two different 3-filter models and a 4-filter model. The 3-filter MMAE models should definitely include the elemental filters on the constant 1.12 amplitude line and perhaps one of the low amplitude filters per MMAE model, thus making two candidate MMAE configurations. The other configuration is a 4-filter model that would include both elemental filters on the constant 1.12 amplitude line and both elemental filters on the constant 0.112 amplitude line.

Another different 3-filter configuration stems from the comparative results of Tables 5.4, 5.8, and 5.10. The greater pogo velocity errors are those for cases in which the "incorrect" filter frequency is different from the "correct" filter frequency. This would lead to having three different frequencies for each of the chosen elemental filters in the MMAE configuration (This error result is not seen for the case of Table 5.6). A third low amplitude MMAE configuration is suggested to include the two high amplitude filters as already analyzed, and a new low amplitude filter with a frequency of 11π (bisecting the 2π-20π constant 0.112 amplitude line). Thus, the two large amplitude filters would be readily distinguishable from each other due to assumed frequency for the pogo, and a single filter would handle the lower amplitude pogo oscillations with an intermediate frequency to be representative of the physical range of possibilities.

Another possible MMAE configuration stems from earlier work in this research line, when the tracking of tactical missiles was investigated. A 5-filter MMAE model would include even higher amplitude filters (with an assumed pogo amplitude of 2.24 pixels) at frequencies of 2π and 20π, along with the filters of $1.12/2\pi$, $1.12/20\pi$, and $0.112/11\pi$. This new high amplitude is not physically motivated (2.24 pixels = 60 meters at the range of 2,000 kilometers; too large a realistic pogo effect) for this particular problem, but scaling of parameters and adaptation schemes using a larger tracking window (24 x 24 versus 8 x 8 tracking window for this problem) could make this MMAE filter configuration applicable.

5.3 MMAE Simulation Results and Analysis

5.3.1 Introduction. The basic purpose for this research is to generate a working MMAE model that is robust in parameter identification and has performance capabilities that are close to the "best" that can be done at a particular real world truth condition. The parameter identification will be enhanced through appropriate discretization of the parameter space, as introduced in Figure 5.2. The proper discretization is carried out by investigating the probability calculations of each of the elemental filters in the MMAE bank: the filter with the highest calculated probability should be the filter with the parameter values which are closest to the real world true parameter values.

The actual implementation of this MMAE model is shown by the block diagram of Figure 5.1 except the Kalman filter block is replaced with the MMAE algorithm composed of several Kalman filters running in parallel. The different configurations of this MMAE algorithm that were introduced in Section *5.2.3* are listed in Table 5.11. Sections *5.3.2-6* detail the results and performance analysis from the computer simulations of the different configurations of Table 5.11

As Figure 5.2 showed, five points were considered for the elemental filters of an MMAE algorithm. The two upper points are essential to performance because

	Number of Filters	Filter Parameter Values - Amplitude-Frequency of Oscillation				
Config #1	3	0.112-20π	1.12-2π	1.12-20π		
Config #2	3	0.112-2π	1.12-2π	1.12-20π		
Config #3	3	0.112-11π	1.12-2π	1.12-20π		
Config #4	4	0.112-2π	0.112-20π	1.12-2π	1.12-20π	
Config #5	5	0.112-11π	1.12-2π	1.12-20π	2.24-2π	2.24-20π

Table 5.11 Test MMAE Configuration

at the higher amplitude, it pays to estimate pogo, as the two points yield very distinguishable residuals. Chronologically, the four corner points were considered, then one or the other low amplitude points was removed, and finally, three points were considered, but with the low amplitude filter assuming the intermediate frequency at 1'π.

A side issue is explored concerning the probability density function calculation of Equations (2.30) and (2.31). These equations contain the covariance of the residuals, $A_k(t_j)$, in the leading coefficient and the exponent of the density function. In previous work [36], the results improved when this covariance has been stripped out of this calculation. In other words, the probability calculation could be carried out without using the leading coefficient, or with $A_k(t_j)$ removed from the exponential term of Equation (2.31) as well. This issue will be explored in Section *5.3.2*, with Sections *5.3.3-6* using the method that yields the best parameter identification

results. The conclusion that is reached from this exercise is that the best parameter identification results are seen when the leading coefficient is removed *and* the covariance of the residuals, $A_k(t_j)$, is removed from the exponent in the calculation of Equations (2.30) and (2.31). This conclusion is thought to be based on the possibility that the $A_k(t_j)$ matrix is ill-conditioned.

5.3.2 MMAE Configuration #1 (Filter Parameter Values: 0.112/20π, 1.12/2π, 1.12/20π). This MMAE filter configuration is tested at various truth conditions with 5-run *Monte Carlo* (MC) simulations and is shown pictorially in Figure 5.3. The tested truth conditions will first be the three parameter values of the included elemental filters, to analyze the effectiveness of the configuration against known conditions. The following data runs will be with truth conditions that are chosen to explore the discretization of the parameter space. Each of the simulations in this section will be run three different times to investigate the effect of the covariance of the residuals on the probability calculations. The first run will have the leading coefficient and the covariance of the residuals, $A_k(t_j)$, intact in the calculation of Equation (2.31). The second run will have the leading coefficient stripped from Equation (2.31). The third run will have the coefficient and $A_k(t_j)$ eliminated completely from Equation (2.31). These simulations will be analyzed for performance of the filter and for proper discretization of the parameter space. Performance of the filter will also be compared to the single-filter runs from Section 5.2.

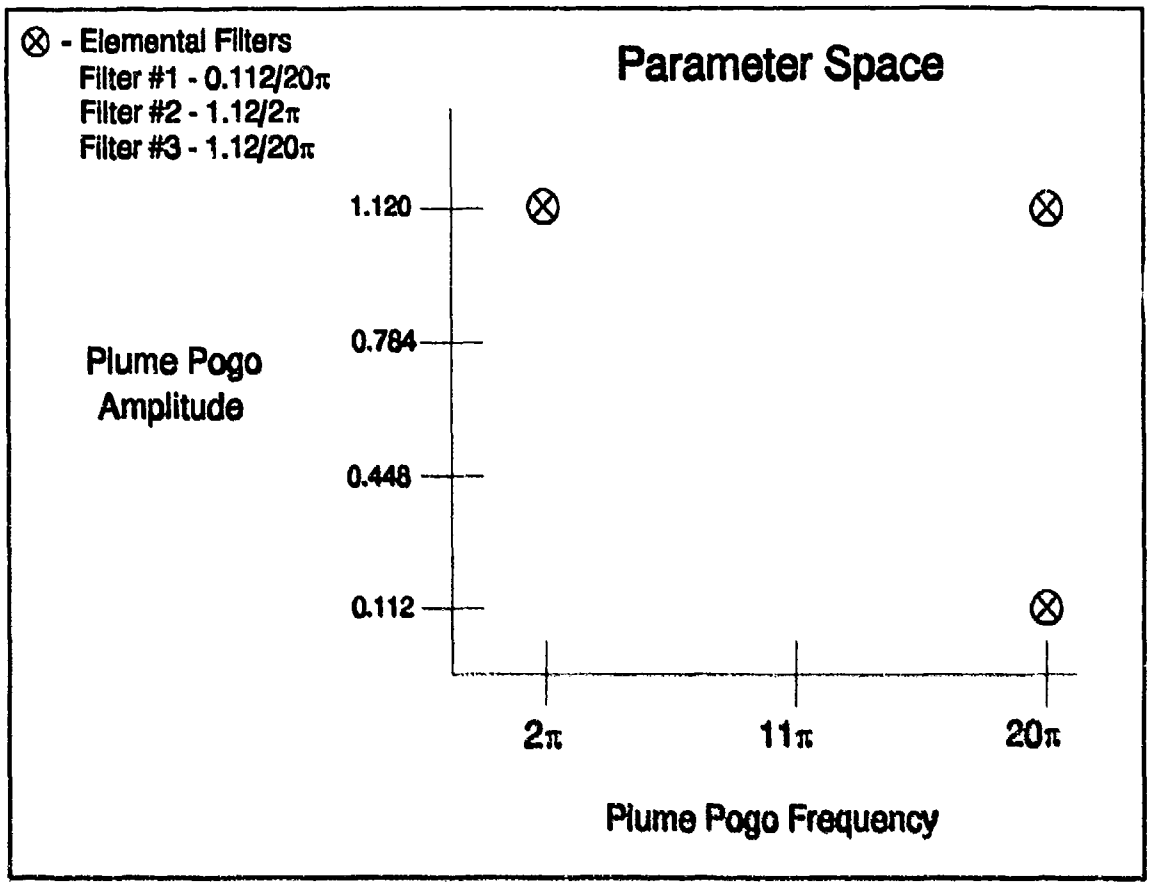


Figure 5.3 Configuration #1 Parameter Space

The output graphical results for this section are presented in Appendix G. Due to the large number of output plots, only the plots that support important trends will be presented. The important graphs from this section are the full set of plots that display each of the Bayesian blended states for comparison to the previous sections' plots at each truth condition. This full set of plots will be the simulation in which the coefficient is removed and the covariance of the residuals is stripped from the exponential calculation of Equation (2.31). This method provided the most consistent and true probability calculation for the discretization of the parameter space, which

is shown by the "averaged probability" output plots from each elemental filter of the MMAE configuration (see, for instance, Figures G.9 through G.11). The performance of this filter configuration is compared to that of the single filter with the "artificially" informed parameter values. The same compilation of the error statistics from the simulations in which there is no leading coefficient and no $A_k(t_i)$ is shown for comparison to the data runs from Section *5.2*.

5.3.2.1 Configuration #1 Run with Truth Parameter Values = 0.112/20π. The parameter space and the MMAE Configuration #1 are shown in Figure 5.3. This is the space in which this configuration #1 will attempt to operate.

Figures G.1 - G.8 show the output of MMAE filter states 1-9 and the linear combination centroid plots. These plots should be compared to plots of the "artificially" informed filter run at this truth condition (Figures D.1 - D.8, in Appendix D). The plots from the MMAE configuration compare very well against all of the single filter plots from the "artificially" informed filter. The noticeable difference is in the pogo position and velocity states (G.5a and G.6a versus D.5a and D.6a), in which the filter estimates in the MMAE plots appear to be overshooting in amplitude the truth data although the filter-predicted plus/minus one sigma contains the filter-versus-actual error in plots G.5b and G.6b.

The statistical errors compiled in Table 5.12 also show that this MMAE configuration does a good job of tracking on all states when compared to the error statistics of the "artificially" informed filter (Table 5.5). The differences in the pogo position and velocity states are seen in the error statistics as the MMAE produces almost twice the position error and about a 20% higher velocity error.

The "averaged probability" plots compare the output probabilities of the three elemental filters, defined in Figure 5.3, when the simulation uses the three different

State	Mean Error (t_i)	Mean Error (t_i')	STD. DEV. $\sigma(t_i)$	STD. DEV. $\sigma(t_i')$
X Position*	0.016	0.016	0.442	0.416
Y Position*	0.045	0.047	0.299	0.278
X Velocity	0.019	0.013	2.734	2.731
Y Velocity	0.071	0.058	2.404	2.401
X Jitter	0.020	0.021	0.427	0.394
Y Jitter	0.025	0.022	0.462	0.441
Pogo Position#	0.001	0.000	0.170	0.171
Pogo Velocity#	0.131	0.066	1.853	8.603
Offset Distance#	0.028	0.030	0.409	0.409
X Centroid	0.004	0.002	0.449	0.432
Y Centroid	0.069	0.074	0.528	0.518

Table 5.12 Time-Averaged Error Statistics for MMAE Configuration #1
(Truth = 0.112/20a)

*X and Y Hardbody Center-of-Mass
#Measured in the Direction of the Velocity Vector

methods of the probability density function calculation. Figure G.9 shows the averaged probabilities (over the 5 MC runs) of the case where the leading coefficient is stripped off and the residual covariance, $A_k(t_j)$, is not used in the exponent calculation. All three of the filters take a share of the probabilities, no one filter dominates and certainly not filter #1, for which this truth condition matches. Plot G.10 shows the case where the coefficient is used and $A_k(t_j)$ is used in the exponent. Filter #2 looks to win out over the other two filters although the other two filters do take some of the probability. Figure G.11 shows the case where the coefficient is stripped from the calculation of Equation (2.31). These plots show the wrong Filter #3 taking all of the probability almost immediately. Despite this difficulty in correctly identifying the "correct" discrete parameter point, the state estimation does not suffer terribly due to small amplitude pogo (seen from comparison of Tables 5.5 and 5.12).

5.3.2.2 Configuration #1 Run with Truth Parameter Values= 1.12/2π. Figures G.12 - G.19 show the output graphs of all of the Bayesian blended filter states. These plots should be compared against Figures E.1 - E.8 from Appendix E, the outputs from the single filter run in which the filter conditions equal the true condition. The plots from the MMAE compare favorably with the single filter plots (despite an unexplained glitch in the data just after the six second point). On the pogo position and pogo velocity plots (G.16, G.17, E.5, and E.6), the MMAE filter estimate is not

quite as good as the single-filter pogo position estimate. This is reflected in the error compilations of Table 5.13 and Table 5.8.

The averaged probability plots of G.20 - G.22 show that when the coefficient is not used and $A_k(t_i)$ is not used in the exponent (G.20), the correct filter (Elemental filter #2) takes over the majority of the probability with a steady gain over the course of the simulation. The probability stays above the 50% mark at approximately one and a half seconds (90 sample periods). The plot where the coefficient and $A_k(t_i)$ are

State	Mean Error (t_i^-)	Mean Error (t_i^+)	STD. DEV. $\sigma(t_i^-)$	STD. DEV. $\sigma(t_i^+)$
X Position*	0.147	0.148	0.881	0.866
Y Position*	0.107	0.105	1.378	1.357
X Velocity	0.027	0.028	3.097	3.089
Y Velocity	0.026	0.023	3.490	3.494
X Jitter	0.019	0.020	0.432	0.407
Y Jitter	0.028	0.027	0.467	0.443
Pogo Position#	0.024	0.023	0.579	0.569
Pogo Velocity#	0.015	0.177	4.213	4.619
Offset Distance#	0.177	0.177	0.897	0.899
X Centroid	0.051	0.051	0.759	0.752
Y Centroid	0.070	0.071	1.211	1.190

Table 5.13 Time-Averaged Error Statistics for MMAE Configuration #1
(Truth = $1.12/2\pi$)

*X and Y Hardbody Center-of-Mass
#Measured in the Direction of the Velocity Vector

used (G.21) show the correct filter #2 taking the majority of the time. The filter #2 probability curve is not as smooth or consistent as was seen in G.20. Figure G.22 depicts the case where the coefficient is not used in Equation (2.31). The wrong filter #3 has most of the probability until the end of the simulation, when filter #2 takes some probability.

5.3.2.3 Configuration #1 Run with Truth Parameter Values = 1.12/20π. Figures G.23 - G.30 depict the output filter states that should be compared to the single-filter results in Figures F.1 - F.8 of Appendix F. The robustness of the MMAE is shown in the very favorable comparison between these sets of output plots. The error statistical comparison between the data of Tables 5.14 and 5.9 show that the errors in all of the filter states are extremely close (except the MMAE has a smaller error on the offset filter state 9).

The averaged probability plots of G.31 - G.33 show that all cases have the correct elemental filter taking almost all of the probability in a very short time. The plot corresponding to the case in which the coefficient and $A_k(t_i)$ are not used (G.31) is less noisy than the plot where Equation (2.31) is intact (G.32) and the plot where just the coefficient is stripped off (G.33).

State	Mean Error (t_i^-)	Mean Error (t_i^+)	STD. DEV. $\sigma(t_i^-)$	STD. DEV. $\sigma(t_i^+)$
X Position*	0.076	0.074	0.681	0.669
Y Position*	0.007	0.009	0.984	0.999
X Velocity	0.143	0.135	2.977	2.975
Y Velocity	0.064	0.053	2.491	2.487
X Jitter	0.008	0.005	0.420	0.387
Y Jitter	0.035	0.035	0.472	0.456
Pogo Position#	0.019	0.023	0.491	0.402
Pogo Velocity#	0.592	0.172	34.032	29.645
Offset Distance#	0.107	0.108	0.370	0.371
X Centroid	0.010	0.005	0.646	0.653
Y Centroid	0.115	0.122	0.936	0.981

Table 5.14 Time-Averaged Error Statistics for MMAE Configuration #1
(Truth = $1.12/20\pi$)

*X and Y Hardbody Center-of-Mass
#Measured in the Direction of the Velocity Vector

5.3.2.4 Discretization of Parameter Space for Configuration #1. Discretization of the parameter space of Figure 5.3 is continued by selecting several sets of truth parameter values to run through the MMAE configuration #1. The averaged probabilities are plotted, with the sum total of these output graphs defining the analysis of this discretized parameter space. The plots of the last three sections reveal that filter #1 did not take the total probability; therefore, concentration will be initially place on the constant 1.12 amplitude line. Only the plots of the probabilities calculated using the method in which the leading coefficient and $A_k(t_i)$

of Equation (2.31) are *not* used have been included. This method provided consistently better parameter identification results. The chosen truth conditions and the Appendix location of the output probability plots are summarized in Table 5.15.

As the constant amplitude 1.12 line is traversed, it is seen that, indeed the parameter space is divided between the two high amplitude filters. At the frequencies of 4π, 8π, and 9.5π, filter #2 takes the majority of the probability. Filter #3 starts to take over at a frequency between 9.5π and 11π (the point seems to be closer to 11π from the shape of the graphs). As the amplitude is decreased along the constant 20π line, it is seen from Plots G.39 - G.40 that Filter #1 is never able to take

Truth Condition	Filter with Majority of Probability	Appendix Location
$1.12/4\pi$	2	G.34
$1.12/8\pi$	2	G.35
$1.12/9.5\pi$	2	G.36
$1.12/11\pi$	3	G.37
$1.12/14\pi$	3	G.38
$0.784/20\pi$	3	G.39
$0.448/20\pi$	3	G.40
$0.112/20\pi$	2	G.41

Table 5.15 Configuration #1 at Various Truth Conditions

the majority of the probability. The output plots from the final point of Table 5.15 look very similar to the output plots of G.20 (Truth = $1.12/2\pi$), despite the strange result of Filter #2 attracting the majority of the probability for this point..

5.3.3 MMAE Configuration #2 (Filter Parameter Values: $0.112/2\pi$, $1.12/2\pi$, $1.12/20\pi$). This new 3-filter configuration is shown in the parameter space of Figure 5.4. Note that the two large-amplitude discrete points are retained from the previous configuration, but that the small-amplitude discrete point is switched from high to low frequency. Having *both* large-amplitude points is critical to performance, but it seems that only one low-amplitude point is needed, and this configuration addresses sensitivity to *placement* of that small-amplitude point. The analysis of this section will focus on the discretization of this parameter space, especially as it compares to that of the previous configuration. The filter state output plots from this 3-filter configuration (at the three truth conditions corresponding to the conditions assumed by each of the three elemental filters) compare to the output plots from the "correct" single-filter run (Truth conditions = Filter conditions) just as favorably as was seen in Section 5.3.2 and thus will not be shown. The error statistics for this MMAE and the single-filter run also matched as well as was seen in the last section and again will not be presented.

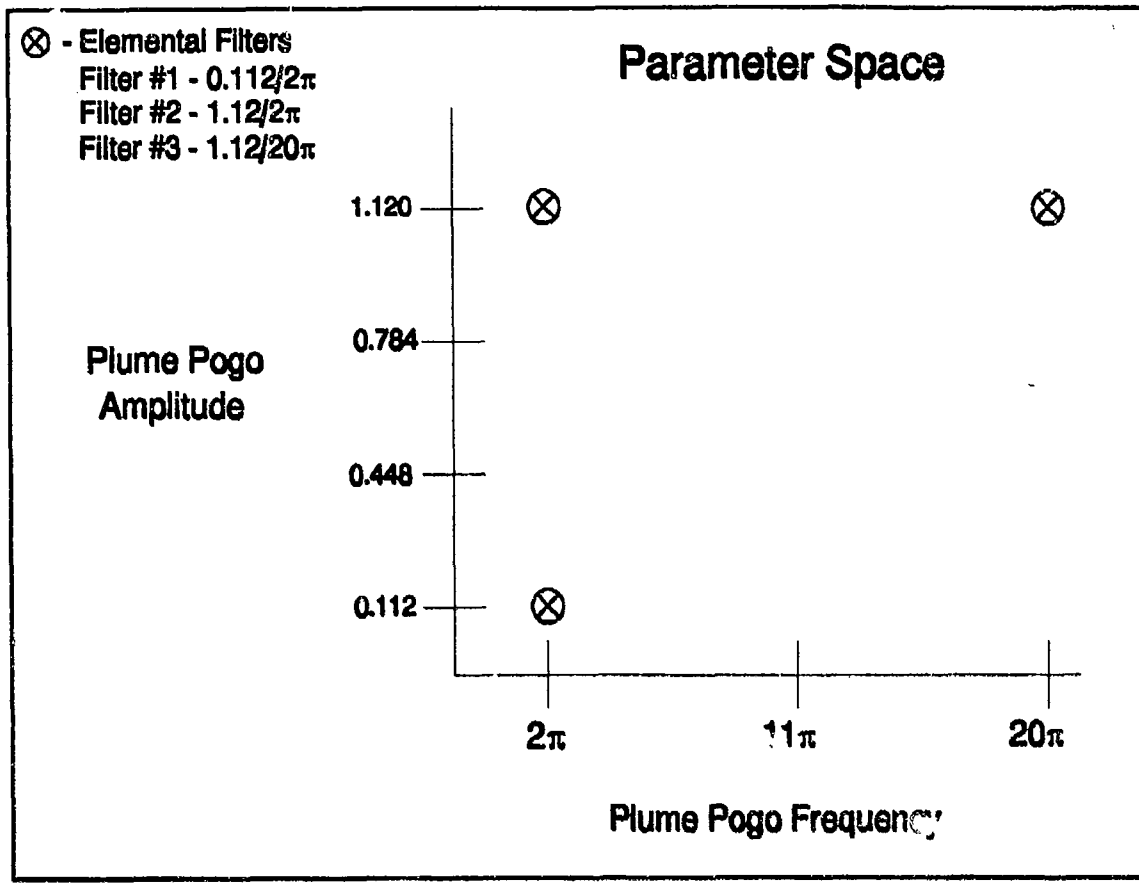


Figure 5.4 Configuration #2 Parameter Space

The output probability plots from these MMAE runs are the graphs of Appendix H. The actual truth conditions against which configuration #2 was run are summarized in Table 5.16 with the Appendix location of these plots.

The results from analyzing the plots of Appendix H show that the parameter space can be partially divided into sections in which one or the other large-amplitude

Truth Condition	Filter with Majority of Probability	Appendix Location
$1.12/2\pi$	2	H.1
$1.12/8\pi$	2	H.2
$1.12/10\pi$	2	H.3
$1.12/10.5\pi$	2 and 3	H.4
$1.12/11\pi$	3	H.5
$1.12/12\pi$	3	H.6
$1.12/14\pi$	3	H.7
$1.12/20\pi$	3	H.8
$0.448/2\pi$	2	H.9
$0.112/2\pi$	2	H.10
$0.05/2\pi$	2	H.11
$0.112/20\pi$	2 and 3	H.12

Table 5.16 MMAE Configuration #2 at Various Truth Conditions

filter dominates the probability calculation. There is a clear division where this occurs along the constant 1.12 amplitude line. Filter #2 takes the majority of the probability at a frequency of 10π while filter #3 gets the majority at 11π. The area of 10π-11π is where this hardover shift occurs. Again, the filter #1 of this configuration couldn't be forced to take over the probability even when the amplitude went to 0.05 (H.11). The data run of the last entry of Table 5.16 shows a kind of "nebulous" region where the probability is spread around among the filters.

5.3.4 MMAE Configuration #3 (Filter Parameter Values: 0.112 / 11π, 1.12 / 2π, 1.12 / 20π). The next logical configuration choice for a low amplitude filter was tried in 3-filter MMAE configuration show in Figure 5.5. This configuration brought the greatest success in the discretization of the total parameter space. The output plots are located in Appendix I, with Table 5.17 summarizing all of the truth conditions against which this configuration was run.

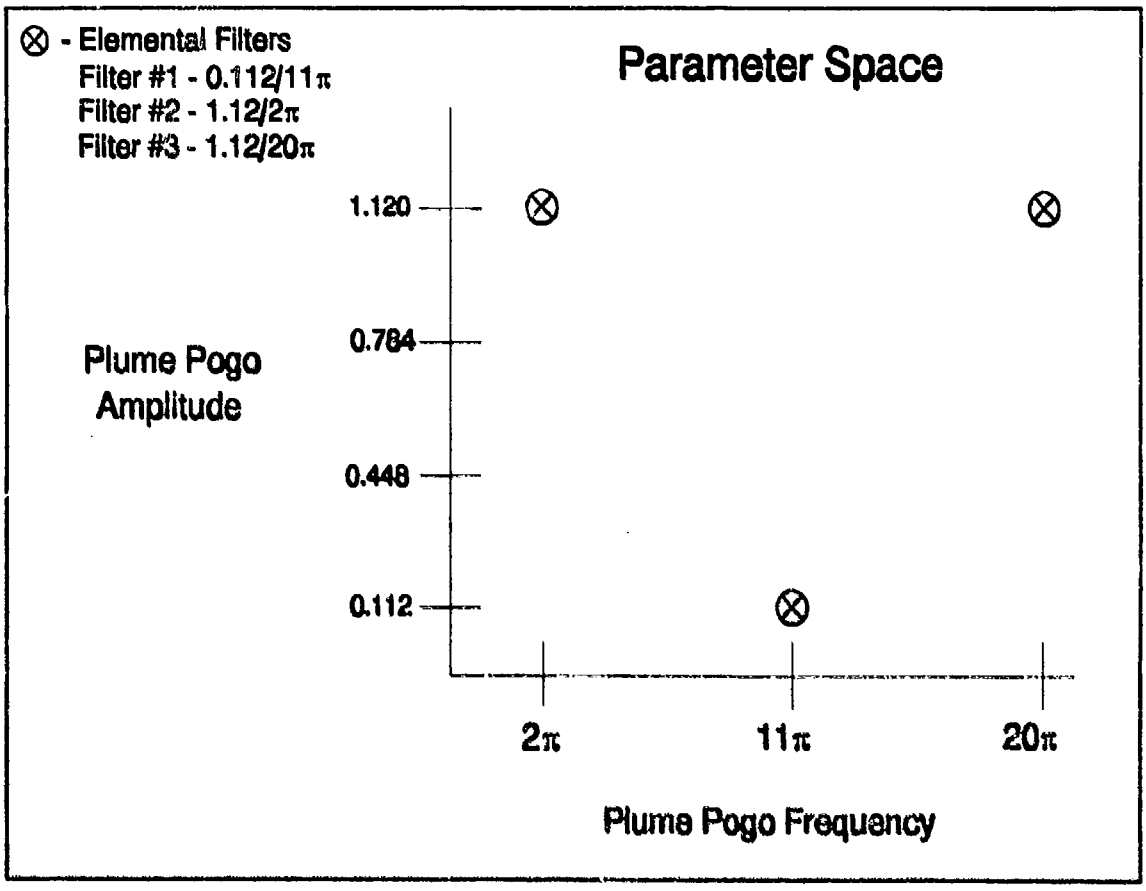


Figure 5.5 Configuration #3 Parameter Space

Truth Condition	Filter with Majority of Probability	Appendix Location
1.12/2π	2	I.1
1.12/8π	2	I.2
1.12/9.5π	2	I.3
1.12/11π	1	I.4
1.12/12.5π	1 and 3	I.5
1.12/14π	3	I.6
1.12/20π	3	I.7
0.784/2π	2	I.8
0.784/9.5π	2	I.9
0.784/10.5π	1	I.10
0.784/11π	1	I.11
0.784/11.5π	1	I.12
0.784/12.5π	1 and 3	I.13
0.784/20π	3	I.14
0.448/2π	2	I.15
0.448/9.5π	1 and 2	I.16
0.448/10.5π	1, 2, and 3	I.17
0.448/11π	1	I.18
0.448/12.5π	1 and 3	I.19
0.448/13π	3	I.20
0.448/20π	3	I.21
0.112/2π	1 and 2	I.22
0.112/11π	1 and 2	I.23
0.112/20π	1, 2, and 3	I.24
2.24/2π	2	I.25
2.24/9.5π	2	I.26
2.24/11π	1	I.27
2.24/12.5π	3	I.28
2.24/20π	3	I.29
0.28/11π	1 and 2	I.30
0.28/20π	3	I.31

Table 5.17 MMAE Configuration #3 at Various Truth Conditions

The partitioning of this parameter space turns out to be dictated by the frequency parameter, as is shown in Figure 5.6. In scanning the plots of Appendix I along the constant amplitude lines of 1.12, 0.784, and 0.448, the low-frequency filter takes the most probability at 9.5π for the three different amplitudes. An observation to note is the startling similarities of the constant-frequency plots at 11π (I.4, I.11, and I.18).

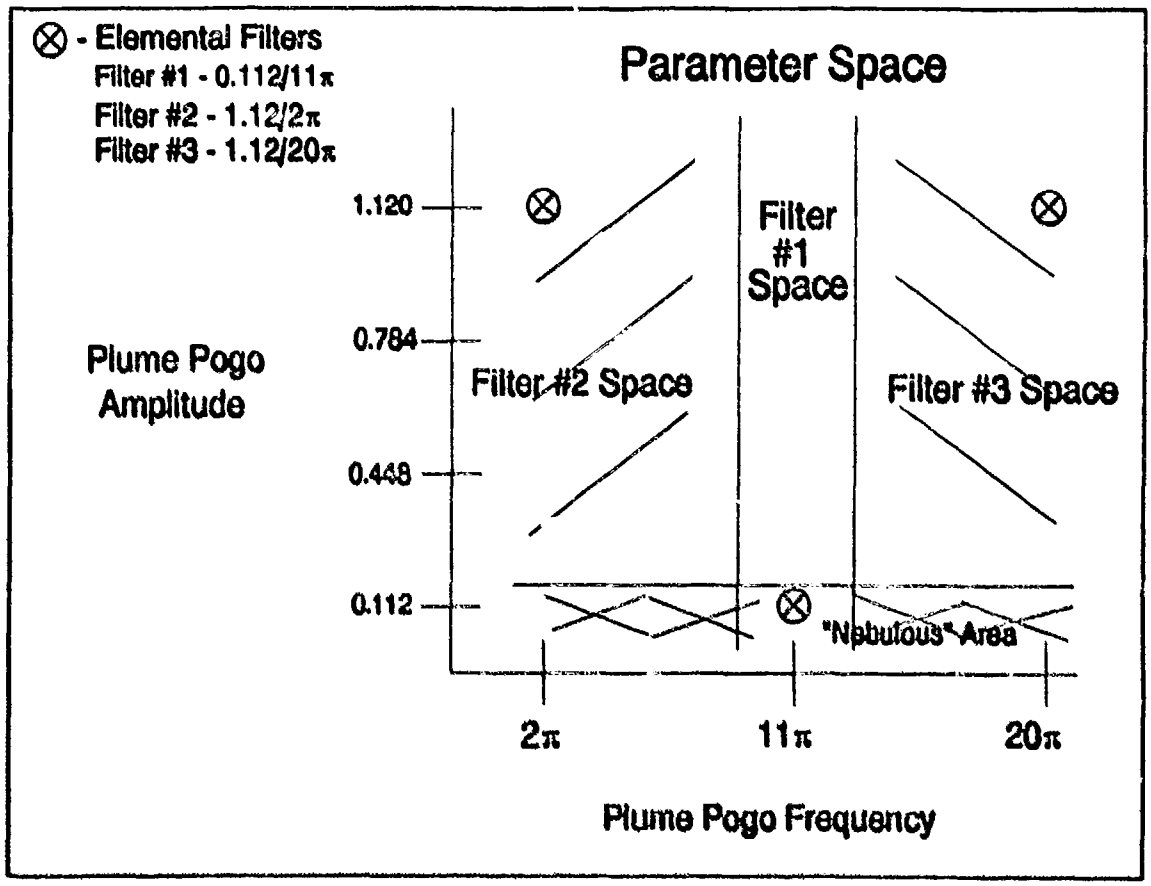


Figure 5.6 Discretization of Parameter Space Configuration #3

Another region that has "similar probability plots is along the constant 0.112 line where filter #2 takes over at low frequencies with filter #1 taking some probability at 2π, 11π, and 20π. This region was called a "nebulous" region because there is no dominant filter. Plots I.30 and I.31 show an extension of this area. This "frequency" discretization extends higher in amplitude as seen from scanning the constant 2.24 amplitude plots of I.26 - I.29. Also, plot I.31 shows filter #3's dominance even at the low amplitude of 0.28 (chosen to be halfway between the amplitudes of 0.112 and 0.448 pixels).

5.3.5 MMAE Configuration #4 (Filter Parameter Values: $0.112/2\pi$, $0.112/20\pi$, $1.12/2\pi$, and $1.12/20\pi$). This 4-filter MMAE configuration was partially explored for the discretization of parameter space of Figure 5.7. The truth parameter values and Appendix J location of the output plots are summarized in Table 5.18.

The results from this configuration show a definite break in the parameter space along the constant 1.12 amplitude line between filter #3 and #4. The low amplitude filters #1 and #2 never really took the probability, even when they were (supposed to be) based upon the best parameter values. This strange partitioning of the parameter space is now expected, following the results from Section *5.3.3*. It is worthy of note how similar the plots are when traversing the 2π constant frequency line (J.2, J.6, and J.10). Also interesting are the charts along the constant 20π

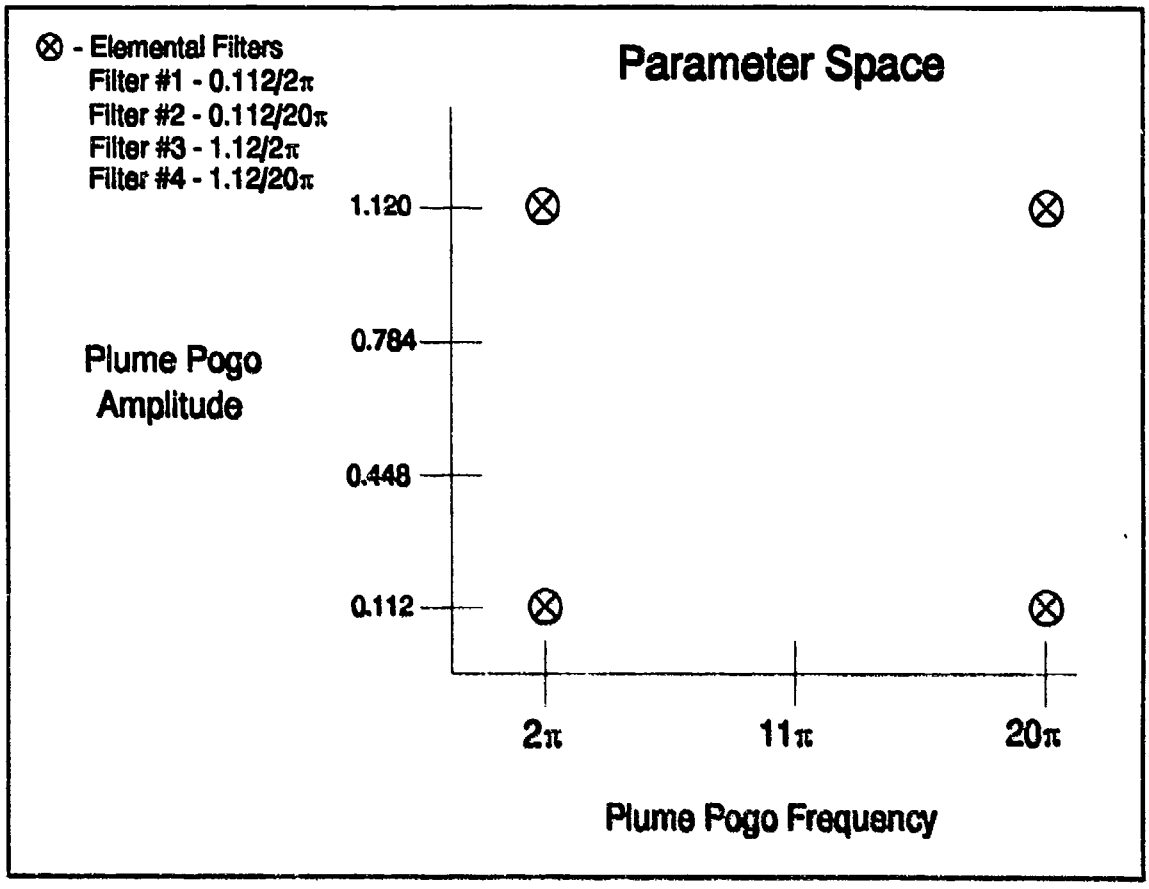


Figure 5.7 Configuration #4 Parameter Space

frequency line. Starting from the highest amplitude and going to lesser amplitudes along this line (1.12 - J.5, 0.784 - J.8, 0.448 - J.9, and 0.112 - I.7), the plots act as they should, with the averaged probabilities of filters #2 and #4 pulling against one another for the probability (#4 "wins" in J.5 and J.8, with #2 "winning" in I.9). Then the curious happens when the amplitude hits 0.112, where the "nebulous" effect described in the preceding section takes over.

Truth Condition	Filter with Majority of Probability	Appendix Location
1.12/2π	3	J.1
1.12/8π	3	J.2
1.12/11π	4	J.3
1.12/14π	4	J.4
1.12/20π	4	J.5
0.112/2π	3	J.6
0.112/20π	3	J.7
0.784/20π	4	J.8
0.448/20π	2	J.9
0.448/2π	3	J.10

Table 5.18 Configuration #4 with Various Truth Conditions

5.3.6 MMAE Configuration #5 (Filter Parameter Values: 0.112/11π, 1.12/2π, 1.12/20π, 2.24/2π, 2.24/20π). This last MMAE configuration was tried to see if the filter could distinguish the truth conditions between the higher amplitudes of 1.12 versus 2.24 at the constant frequencies of 2π and 20π, and further, if there would be any substantial benefit of including the two additional filters upon *state* estimation precision. The parameter space is shown in Figure 5.8, and Table 5.19 summarizes the truth conditions that were used.

Truth Condition	Filter with Majority of Probability	Appendix Location
1.12/2π	2	K.1
2.24/2π	2	K.2
1.12/20π	3	K.3
2.24/20π	3	K.4
0.448/11π	1	K.5

Table 5.19 Configuration #5 with Various Truth Conditions

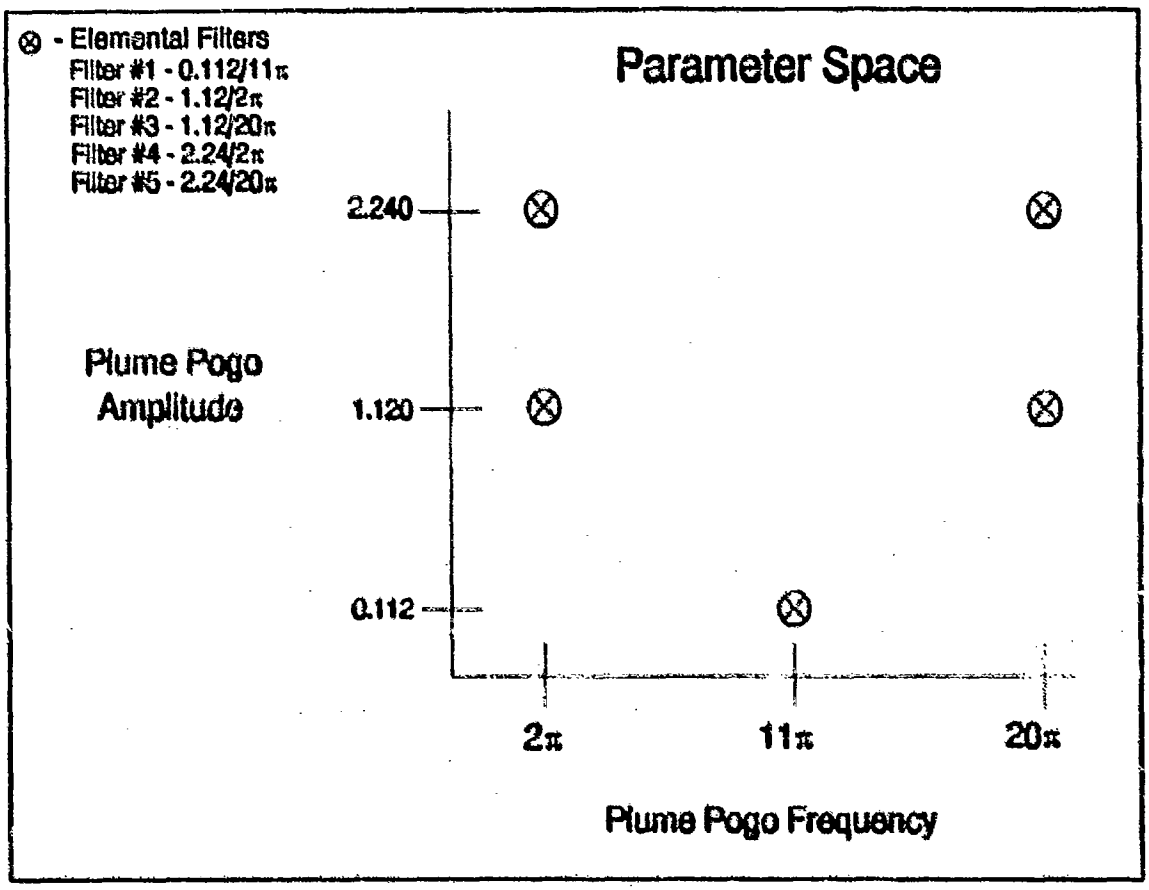


Figure 5.8 Configuration #5 Parameter Space

THIS

PAGE

IS

MISSING

IN

ORIGINAL

DOCUMENT

5-46

effect to be seen in other state estimates, including the estimate of hardbody location. This chapter has shown that a working elemental filter exists that was successfully used in several MMAE algorithm structures for the purpose of discretizing the two dimensional parameter space of amplitude and frequency of oscillation of the plume pogo phenomenon.

VI. *Conclusions and Recommendations*

6.1 Introduction

This chapter summarizes the final conclusions of this thesis and suggests areas for further study. Section *6.2* draws conclusions based on the results obtained in Chapter V. Section *6.3* contains suggestions for continued research in applying this FLIR/Low-Energy-Laser algorithm to the ballistic missile tracking problem.

6.2 Conclusions

Numerous conclusions have been made throughout this research. These conclusions are presented in the following subsections.

6.2.1 Single-Filter Data Runs. The initial portion of this work dealt with becoming familiar with the FORTRAN computer program that implemented this tracking scenario. The main purpose, in this beginning stage, was to test the four (and an eventual fifth) proposed elemental filters to ensure that a working Multiple Model Adaptive Estimation algorithm could be pursued. The task at hand was to get the filter working properly. There were four seemingly minor errors found in the

software that, when fixed, provided this working single-filter algorithm. The interested reader is referred to the thesis advisor for the actual corrections to the code.

The main and most important conclusion drawn from the first half of this research is that the single-filter model works at the proposed filter parameter values for amplitude and frequency of oscillation of the pogo phenomenon. This conclusion is verified by the plots of Appendices C through F. The "wrongly" programmed filters provide output according to these "wrong" input pogo parameter values, with the higher errors seen on all of the states and especially the pogo position and velocity states. The "correctly" programmed filters follow the truth model state if the amplitude of the pogo is large enough (the "correct" filter is not really able to track a small amplitude pogo value precisely, but it is not essential to be able to do so at such small amplitudes).

An obvious conclusion from these single-filter data runs is that residual-monitoring of the Kalman filters is a powerful tool which is applied to this specific tracking problem. The residual outputs from these single-filter data runs were analyzed and then used to pick the "best" MMAE structure for this particular problem. The lack of success with the 4-filter and 5-filter structures of this thesis was predicted from the filter residual analysis. This is also the case for the partial success of MMAE configurations #1 and #2 (two of the elemental filters were

programmed with the same frequency in these configurations). Residual monitoring was a key ingredient to this research.

A final conclusion from these single-filter data runs comes from the necessity of accomplishment of these baseline data runs. The single-filter error data from the "artificially" informed Kalman filters was used as a benchmark for comparison: a "best that can be done." Also, the plots of the individual states were comparison tools for the completed MMAE. Going through this initial work was essential to the success of this MMAE algorithm.

6.2.2 MMAE Data Runs. The success that is seen in the final portion of this research is directly attributable to the residual-monitoring that has been described throughout. MMAE configuration #3 was the most successful in partitioning the parameter space. The clear boundaries of transition between the filters' probability dominance is a little surprising because of the small difference in the residual quantities. This configuration holds the most promise for an implementable configuration.

An idea coming from the analysis of this parameter space points to an "attraction of probability" that is seen or not seen in the MMAE configuration outputs. There seems to be more parameter space probability attractability as the pogo frequency and amplitude increase: as one goes higher and to the right in the

defined parameter space. This "obvious" idea is readily seen in the constant 0.112 and 1.12 amplitude lines. The differences between the residuals of the filters at $1.12/2\pi$ and $1.12/20\pi$ are very noticeable, while virtually no difference exists in the residuals of the lower amplitude filters of $0.112/2\pi$ and $0.112/20\pi$. Also, the constant frequency lines of 2π and 20π partially show this idea, as there is a partial partitioning that forms between the constant frequency filters of $1.12/20\pi$ and $0.112/20\pi$ in Configuration #1. This partial partitioning is not seen in the lower frequency filters of $0.112/2\pi$ and $1.12/2\pi$ in Configuration #2. Finally, the transition area along the constant 1.12 amplitude line for configurations #1 through #3 occurs closer to the smaller frequency filter of $1.12/2\pi$. The larger frequency filter of $1.12/20\pi$ is more dominant. These ideas point to a larger "attraction of probability" as amplitude and frequency are increased in this parameter space.

Throughout this MMAE section, the dominant parameter value is frequency. The parameter space is almost entirely partitioned according to this parameter. This fact should be a consideration when this research line moves forward. This frequency dominance is not surprising, and it even makes sense. It is essential in the tracking of a signal to be in correct synchronization and phase with an incoming signal. The tracker has no chance at all, if out of synchronization. If the amplitude is off, the errors have larger deviations from the zero-error line but, the error signal cycles through this zero-error line.

The side issue that was explored in Section *5.2* showed a marked difference in the probability results that were obtained from the three different probability calculation methods. The method used throughout the main portion of the MMAE runs is called Maximum Entropy with Identity Covariance (ME/I) [3]. The covariance of the residuals is removed completely from the probability calculations by setting this variable to the identity matrix. The covariance matrix is thought to be possibly misrepresented due to model uncertainties, and thus to weight all residuals equally in the probability calculation, the covariance is set to identity.

6.2.3 Final Performance Issues. The MMAE algorithm Configuration #3 had very comparable tracking statistics of filter states #1 and #2, the hardbody center-of-mass, even when correct parameter identification broke down in the "nebulous," low pogo amplitude region. These error statistics are shown in Tables 6.1 through 6.4. When the truth parameter values resided in the nebulous region, all of the MMAE individual states had error statistics very comparable to the "best" single-filter except the pogo position state. The MMAE saw 33% higher errors than the "best" single-filter for this state. It is thought that the two higher amplitude/frequency filters were causing this larger pogo error by artificially amplifying their own particular pogo position error. When the truth conditions were set to higher pogo amplitude values, the pogo position errors revealed good agreement between the MMAE and the single-

Truth = 0.112/2π	Center-of-Mass Errors	
Filter =	Filter State #1 - x error	Filter State #2 - y error
0.112/2π	0.402	0.313
1.12/2π	0.395	0.280
0.112/20π	0.403	0.309
1.12/20π	0.417	0.291
MMAE Configuration #3	0.409	0.283

Table 6.1 Center-of-Mass Error Comparison; Truth = 0.112/2π

Truth = 0.112/20π	Center-of-Mass Errors	
Filter =	Filter State #1 - x error	Filter State #2 - y error
0.112/20π	0.418	0.304
0.112/2π	0.408	0.381
1.12/20π	0.418	0.274
1.12/2π	0.421	0.286
MMAE Configuration #3	0.416	0.278

Table 6.2 Center-of-Mass Error Comparison; Truth = 0.112/20π

Truth = 1.12/2π	Center-of-Mass Errors	
Filter =	Filter State #1 - x error	Filter State #2 - y error
1.12/2π	0.756	1.186
1.12/20π	0.956	1.353
0.112/2π	0.878	1.437
0.112/20π	0.900	1.427
MMAE Configuration #3	0.804	1.244

Table 6.3 Center-of-Mass Error Comparison; Truth = 1.12/2π

Truth = 1.12/20π	Center-of-Mass Errors	
Filter =	Filter State #1 - x error	Filter State #2 - y error
1.12/20π	0.679	1.008
1.12/2π	0.788	1.091
0.112/20π	0769	1.174
0.112/2π	0.771	1.289
MMAE Configuration #3	0.671	0.991

Table 6.4 Center-of-Mass Error Comparison; Truth = 1.12/20π

filter data runs. Again, the accuracy of estimating this pogo position state is important, but not as critical as the tracking of the hardbody center-of-mass (filter states #1 and #2).

6.3 *Recommendations*

This section concludes this thesis with a number of suggestions for future research topics. The suggestions are divided into two major parts that are described in the following subsections.

6.3.1 Further Research With Current MMAE Scheme. The most obvious recommendation would be a direct follow-on thesis that further investigates the

partitioning of the parameter space using several different MMAE configurations. The higher probability attracting areas could be exploited with the positioning of more elemental filters to take advantage of the residual differences, to try to minimize the tracking errors caused from the incorrect assumptions of the frequency and amplitude parameter values. A possible suggestion for this different configuration would be to have no low-amplitude filters, but to have all of the filters programmed in a "one-dimensional" parameter space across the constant 1.12 amplitude line. A variation of this scheme would have one low-amplitude filter programmed with an amplitude between the 1.12 and 0.112 values (0.448 or 0.784) with a frequency that is different from 20π (approximately 14π, for the purpose of exploiting the frequency partitioning phenomenon), along with the two high amplitude filters of $1.12/2\pi$ and $1.12/20\pi$. This configuration could take advantage of the partial partitioning that was seen along the constant 20π line in Configuration #1, while also taking advantage of the different frequency parameter values. This new MMAE configuration's tracking errors would then be compared to the "best" single-filter and the MMAE configuration #3 of this research.

The expansion of the parameter space higher in amplitude should also be investigated if the tracking scenario should change. Divergence was seen when amplitude values of five times the current amplitude values were programmed into the filter. Although the higher amplitude pogo values were not physically motivating

for this particular scenario, should the tracking range be changed to a value less than the current 2,000 kilometers, such high pogo amplitudes could become a major factor.

6.3.2 Implementable Algorithms. A look to the future at this point might bring further suggestions for research topics. The current MMAE scheme could be used to estimate the pogo amplitude and frequency parameters. A single Kalman filter would be used as a state estimator that would have the pogo parameters periodically updated by parameter estimates from the MMAE algorithm.

Different tracking scenarios could also be accommodated with this particular MMAE algorithm. Individual pixels on the 300 x 500 FLIR could be combined in groups, then to be used as single pixels in some of the elemental filters of the MMAE algorithm of this thesis. There would still be an 8 x 8 tracking window, but each pixel of this 8 x 8 window would be made up of a cluster of individual pixels. The number of pixels per cluster could be scaled as a function of the range to a target. This adaptation scheme would use an MMAE algorithm similar to what is presented in this thesis.

Appendix A

Data Processing Statistics Method

This Appendix explains how the statistics listed in the thesis and plotted in the succeeding appendices were determined. The equations for the statistics and the data used to plot them are discussed.

The performance of the Kalman filters used in this thesis is evaluated using multiple Monte Carlo runs. A Monte Carlo analysis involves collecting statistical information generated from simulating samples of stochastic processes [21]. Ten Monte Carlo runs are generally considered sufficient to converge to the actual statistics that would result from an infinite number of runs [9,24]. After collecting N samples of truth model and filter model data for each of N Monte Carlo runs, the true error statistics can be approximated by computing the sample mean error and error variance for the N runs. The sample mean error and error variance are computed by:

$$\bar{E}(t_i) = \frac{1}{N} \sum_{n=1}^{N} \left[x_{truth_n}(t_i) - x_{filter_n}(t_i) \right] \tag{A.1}$$

$$\sigma^2(t_i) = \frac{1}{N-1} \sum_{n=1}^{N} \left[x_{truth_n}(t_i) - x_{filter_n}(t_i) \right]^2 - \frac{N}{N-1} \bar{E}(t_i) \tag{A.2}$$

where:

$\bar{E}(t_i)$	=	sample mean of the error of interest at time t_i
$\sigma^2(t_i)$	=	sample error variance at time t_i
$x_{truth\,n}(t_i)$	=	truth model value of the variable of interest at time t_i during simulation n
$\hat{x}_{filter\,n}(t_i)$	=	filter estimate of the variable of interest at time t_i during simulation n
N	=	number of Monte Carlo runs

The statistics are calculated before the measurement update at (t_i^-) and after the update at (t_i^+). In the performance plots displayed in Appendices B through E, the statistics at each instant in time are plotted together; that is, the statistics before and after the measurement update are plotted on the same time axis. They are reduced further to obtain average scalar values over the time of the run, by temporally averaging the mean error and standard deviation (σ) time histories from two seconds into the simulation until the end. The first two seconds are not used to ensure that the data reflects only steady state performance [9]. The errors are measured in units of pixels, where a pixel is 15 microradians on a side (approximately 30 meters at a distance of 2,000 kilometers).

Appendix B

Examples of Performance Plots

This Appendix displays an example of the performance plots referenced throughout the thesis. An explanation of the plot components and their meaning is also given.

Two different types of data plots are presented in Appendices C through K to assess the performance of the center-of-mass filters employed in this thesis. The first type of plot, the state comparison plot shown in Figure B.1(a), provides a direct comparison of the filter estimated and true value of the state. For these plots, the ensemble average (over N Monte Carlo runs) of the true value of the state is shown as a solid line. The ensemble average value of the filter estimate at any instant in time is shown as a dashed line.

The second type of plot, the error statistics plot shown in figure B.1(b), provides a measure of the tracking performance. The plot shows the mean filter error, averaged over the N Monte Carlo runs at each instant in time, for a state or variable of interest. In addition, this type of plot displays the actual 1σ (standard deviation) centered on the mean, or mean $\pm$ 1σ curves. They are the two dotted lines that surround the mean curves. All the filters for this thesis were designed to assume zero mean errors in all states, so the filter computed estimate of standard deviation is plotted relative to the abscissa. The legend for the symbology in the error statistics plots is shown here.

Mean Error	————————
Mean Error $\pm$ 1σ	.....
Zero $\pm$ Filter Computed 1σ	----------

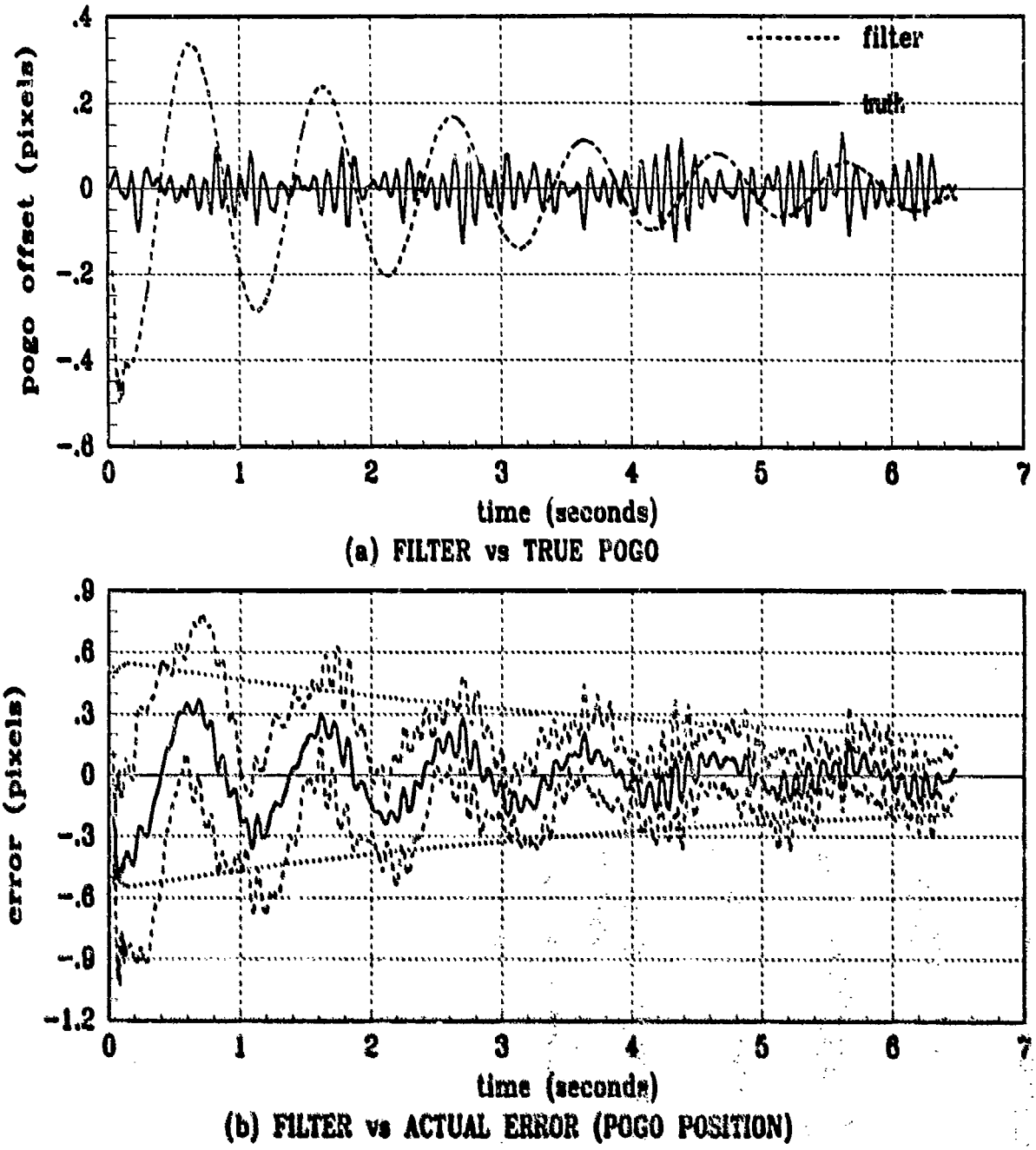


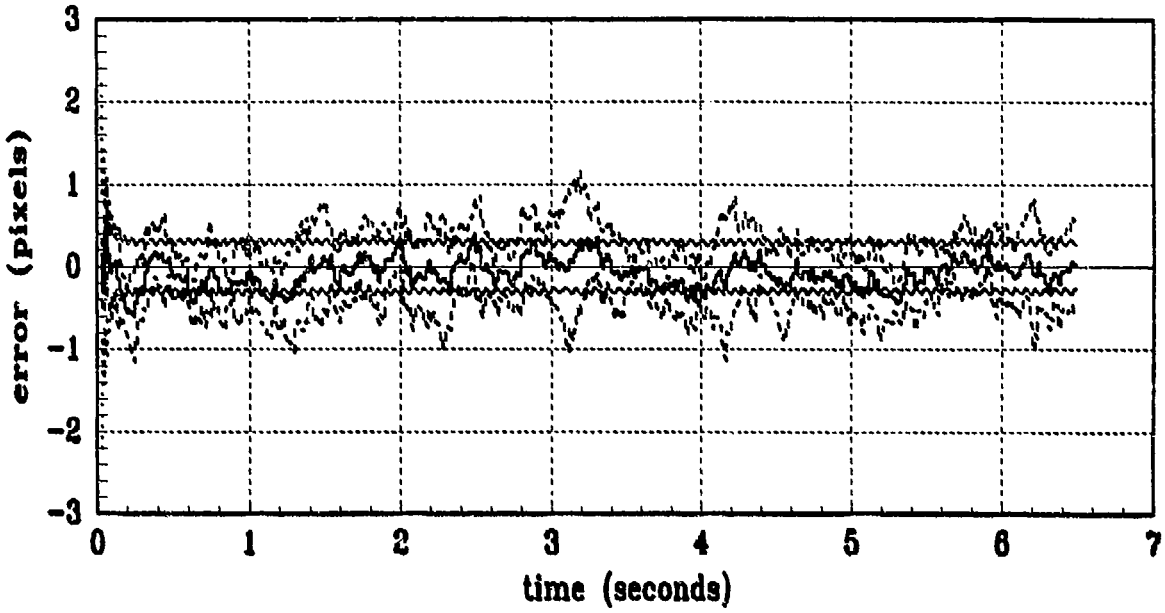
Figure B.1 Example of State Plot and Error Statistics Plot

Appendix C

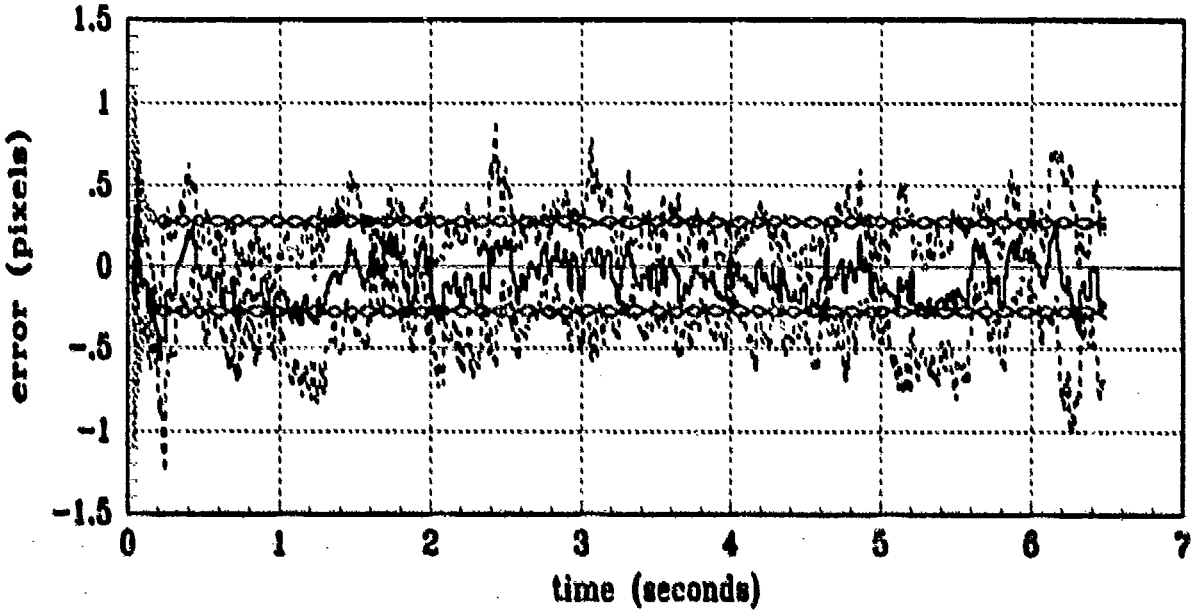
Filter Plots with Truth Parameter Values:

Amplitude = 0.112, Frequency = 2π

This appendix contains the state and error statistics plots of the nine-state elemental filters. The data depicted in the two types of plots in this appendix are explained in *Appendix A*. The state comparison plots show the sample mean truth state over the 5 *Monte Carlo* runs compared to the same statistic for the filter estimate. The error statistics plots represent the error mean $\pm$ standard deviation values in pixels (or pixel/second for velocity and pogo velocity), of the errors between the filter estimate and true state; true mean $\pm$ 1 true standard deviation are plotted, along with zero $\pm$ 1 filter-computed standard deviation.

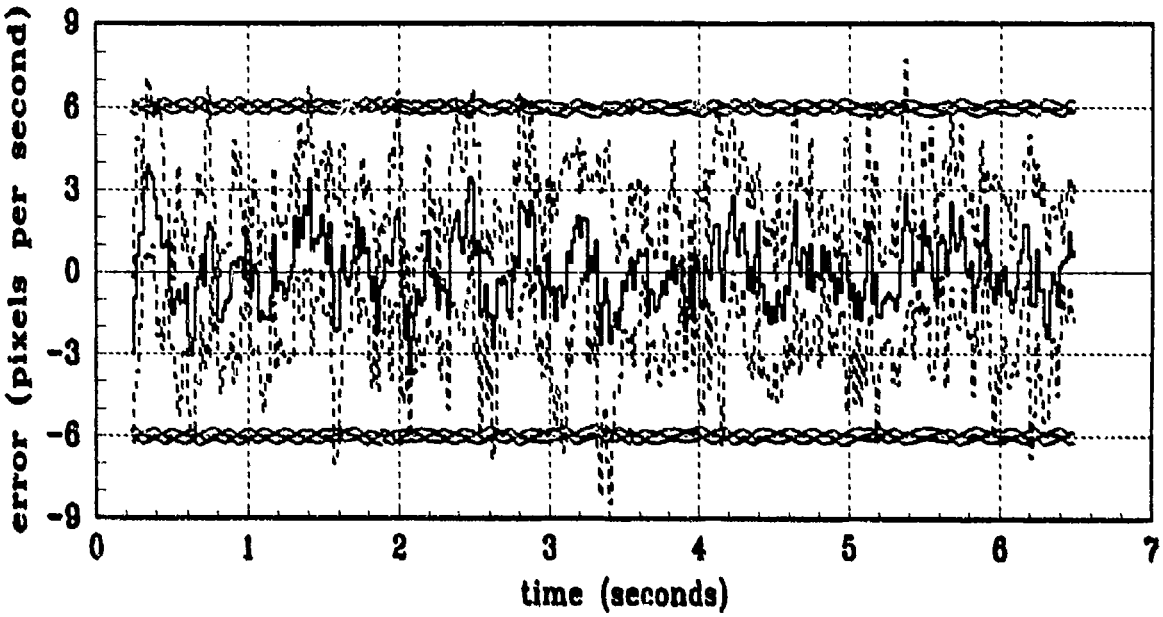


(a) FILTER vs ACTUAL ERROR (X-POSITION)

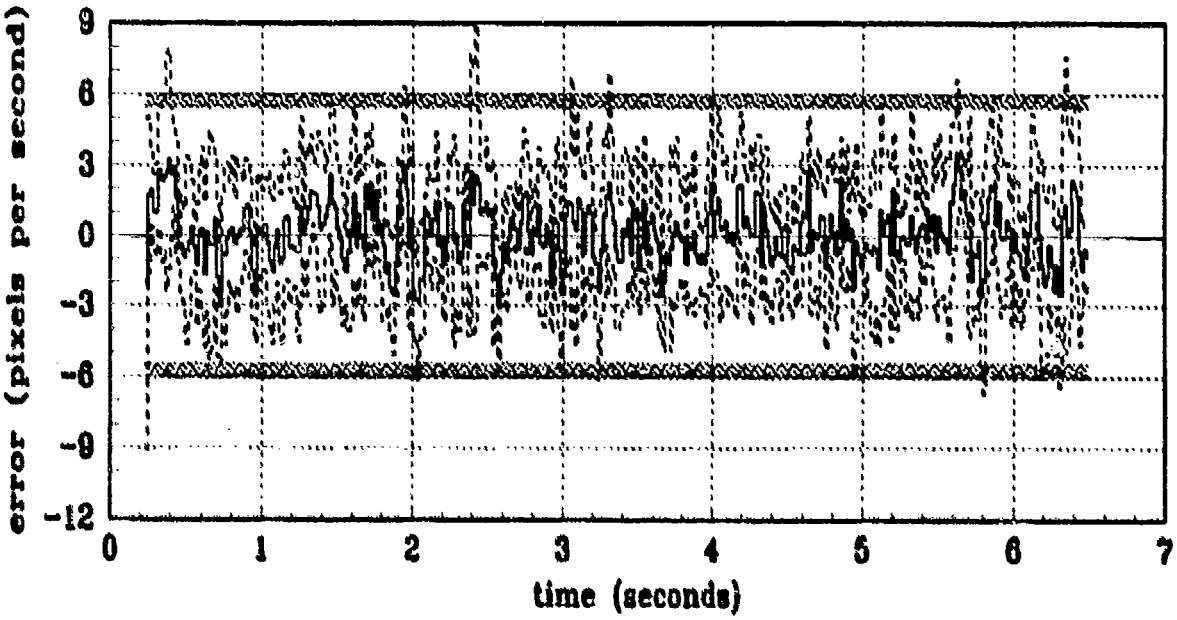


(b) FILTER vs ACTUAL ERROR (Y-POSITION)

Figure C.1 X/Y Position (Filter States 1 and 2) Error Statistics
(Truth = $0.112/2\pi$, Filter = $0.112/2\pi$)



(a) FILTER vs ACTUAL ERROR (X-VELOCITY)



(b) FILTER vs ACTUAL ERROR (Y-VELOCITY)

Figure C.2 X/Y Velocity (Filter States 3 and 4) Error Statistics
(Truth = $0.112/2\pi$, Filter = $0.112/2\pi$)

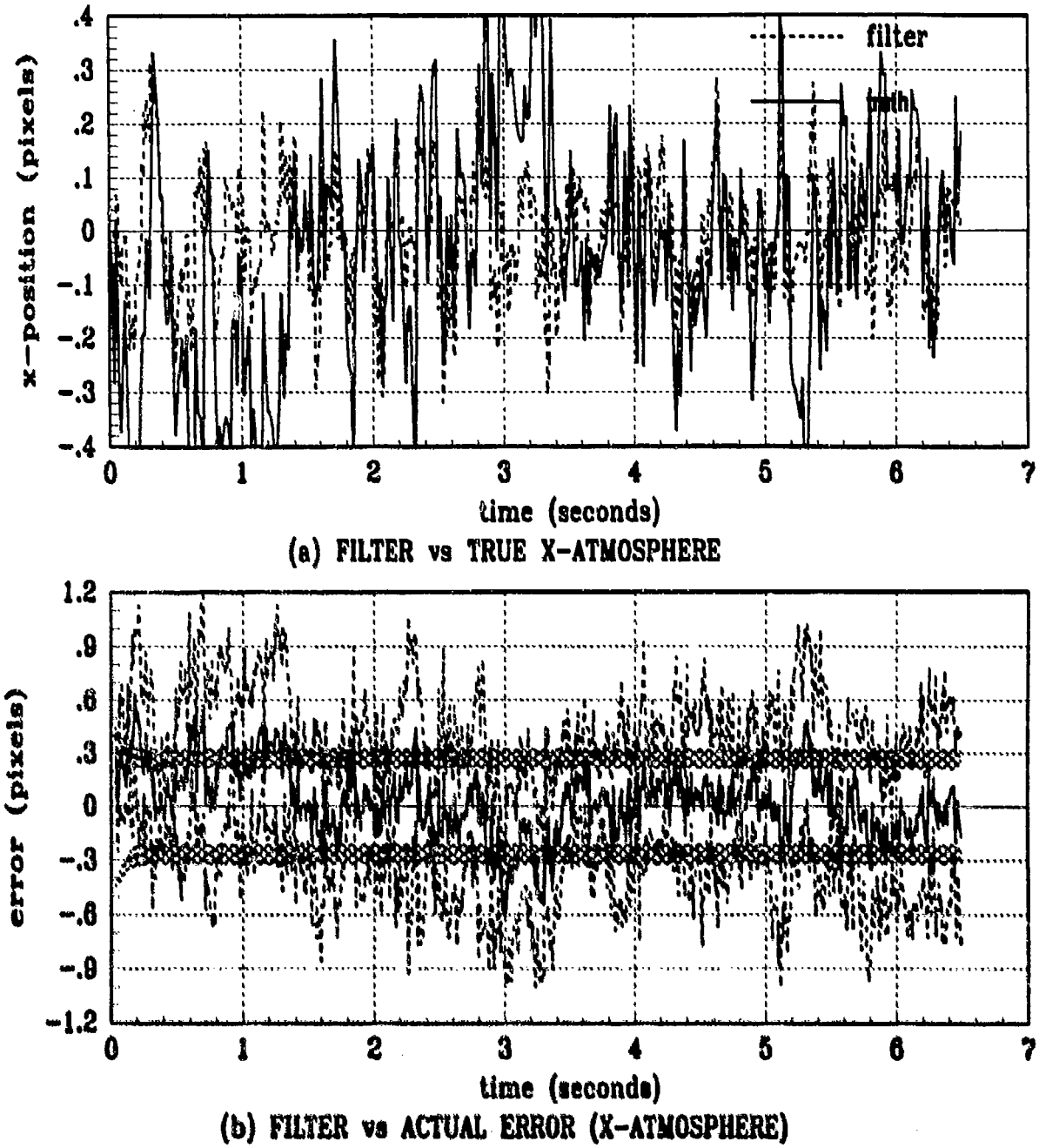
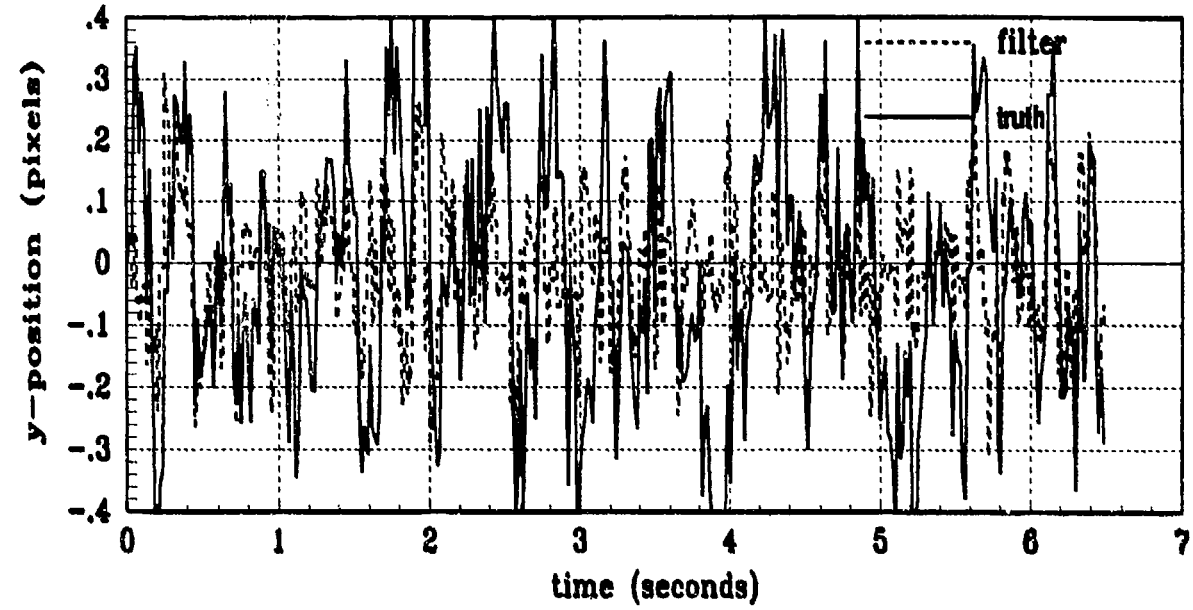
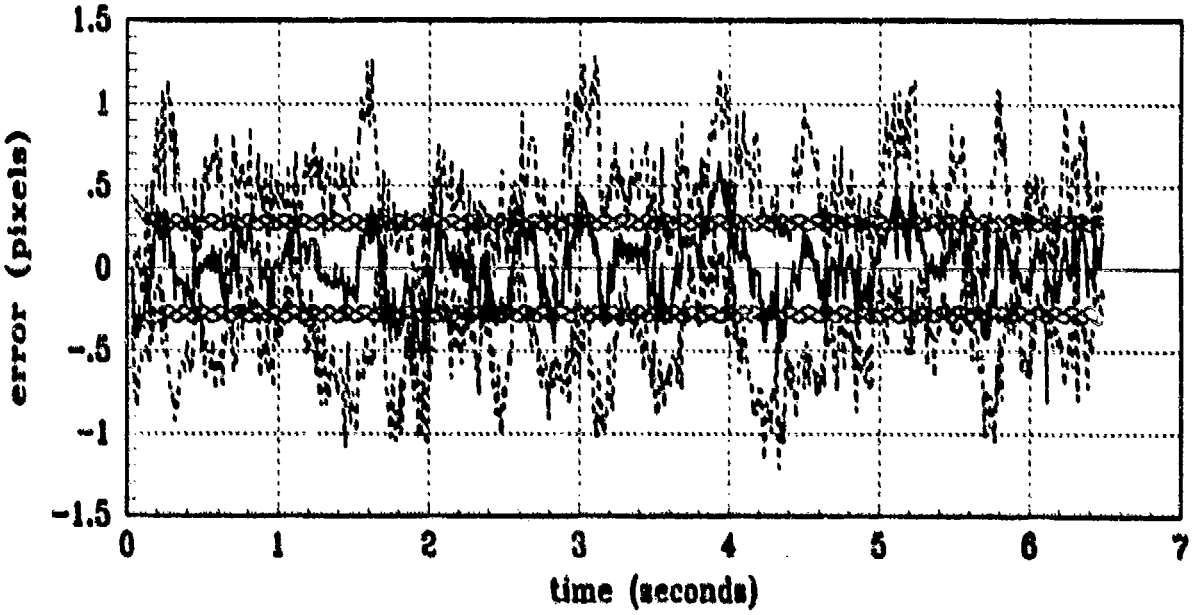


Figure C.3 X Atmospheric Jitter (Filter State 5)
(Truth = $0.112/2\pi$, Filter = $0.112/2\pi$)



(a) FILTER vs TRUE Y-ATMOSPHERE -



(b) FILTER vs ACTUAL ERROR (Y-ATMOSPHERE)

Figure C.4 Y Atmospheric Jitter (Filter State 6)
(Truth = 0.112/2π, Filter = 0.112/2π)

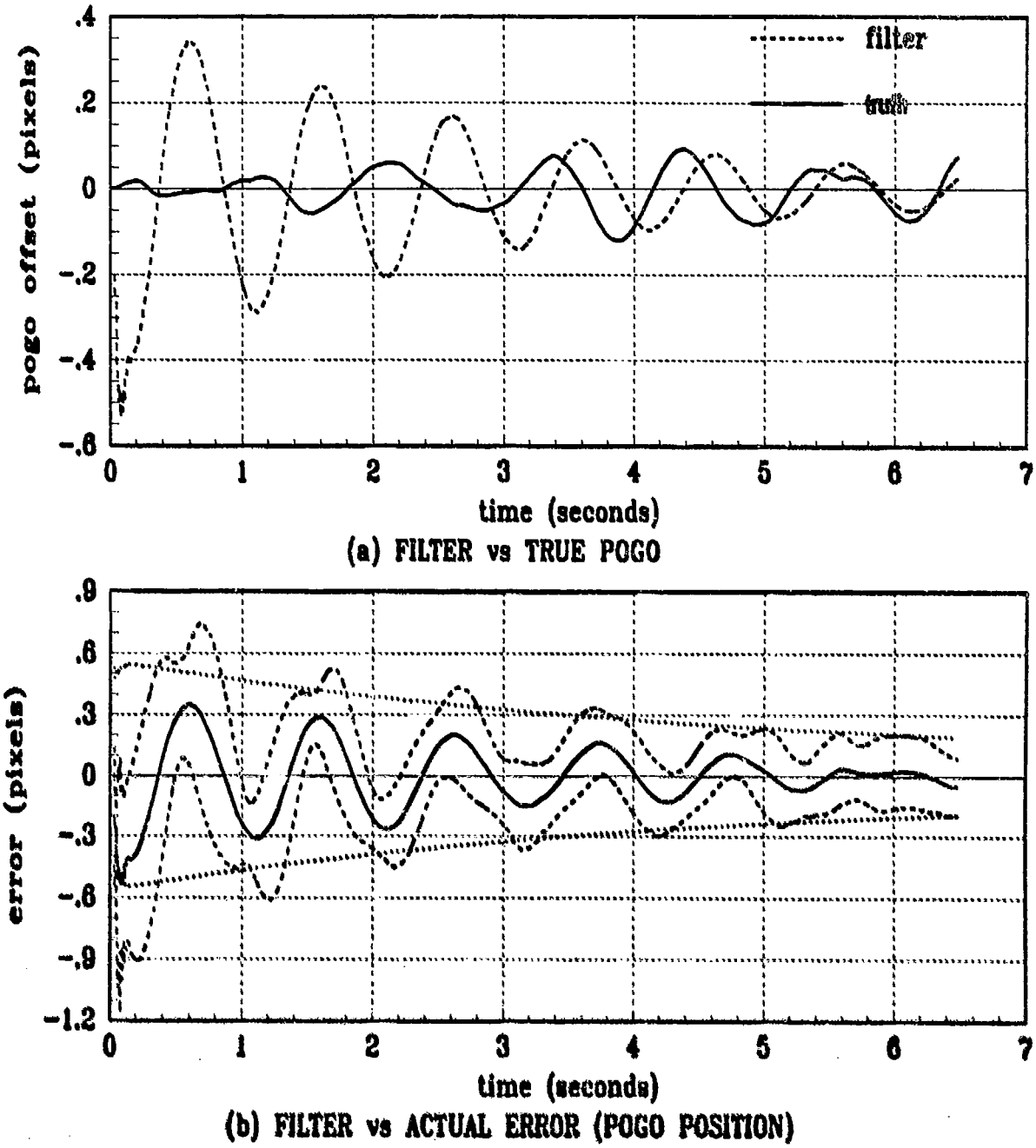
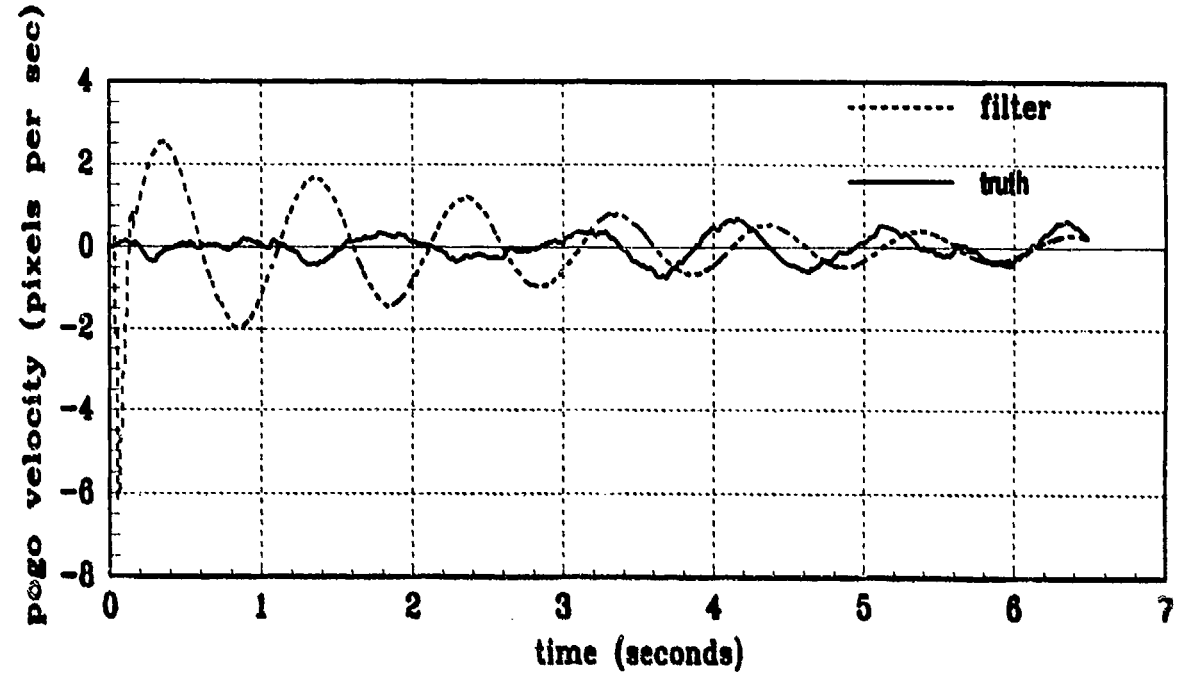
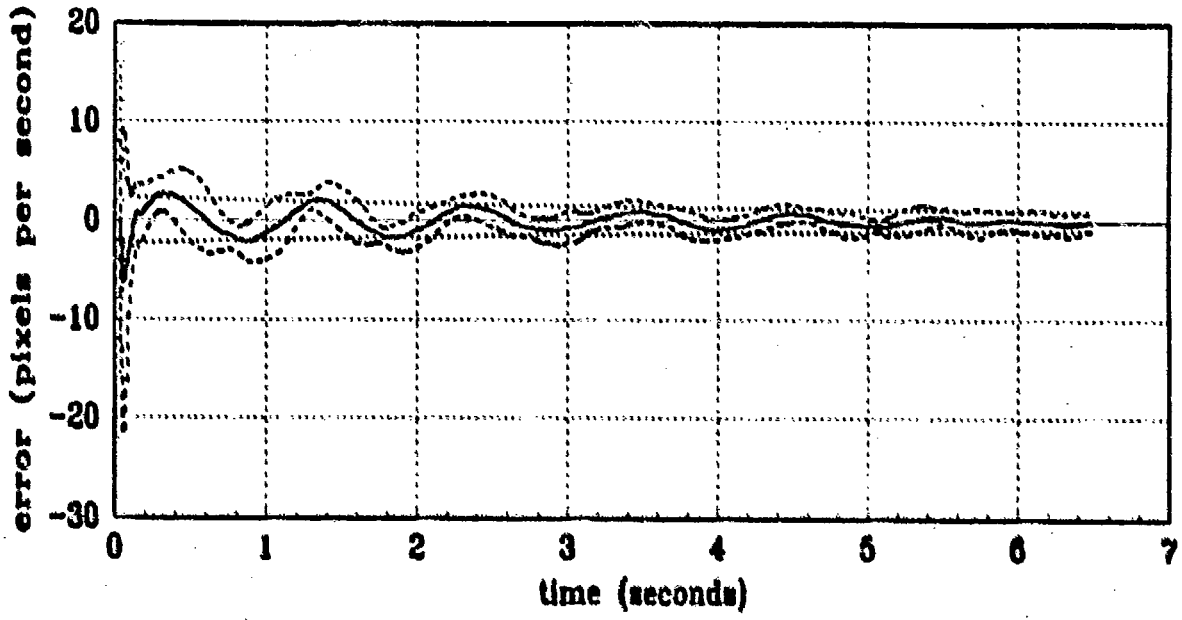


Figure C.5 Pogo Position Offset (Filter State 7)
(Truth = $0.112/2\pi$, Filter = $0.112/2\pi$)

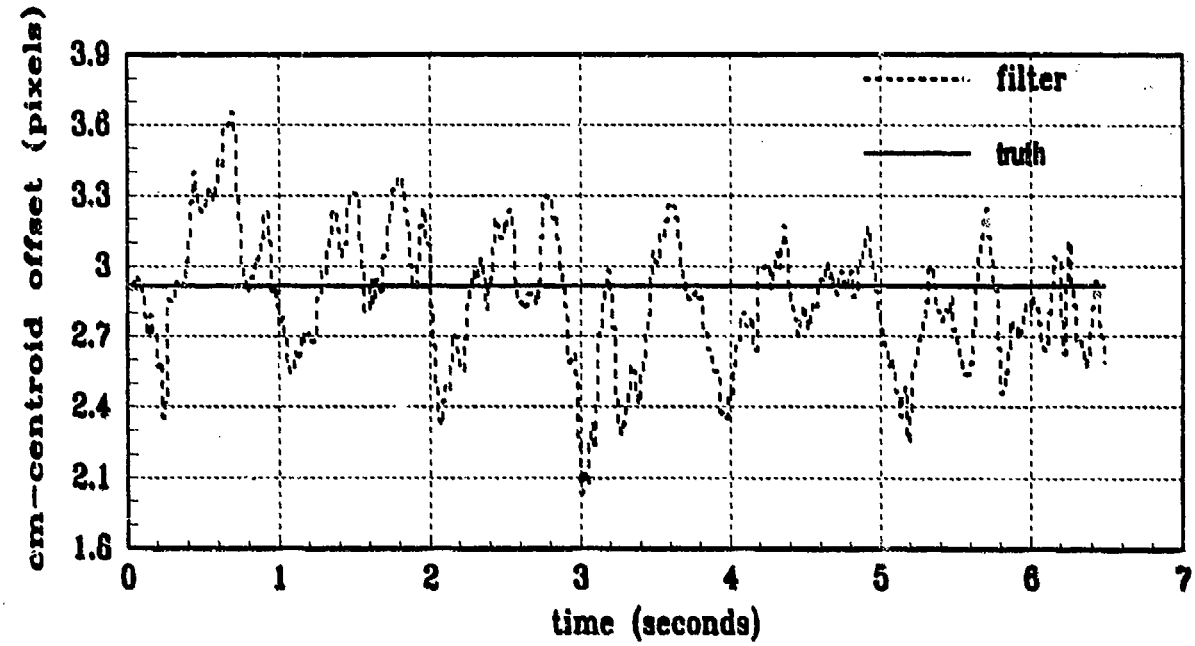


(a) FILTER vs TRUE POGO VELOCITY

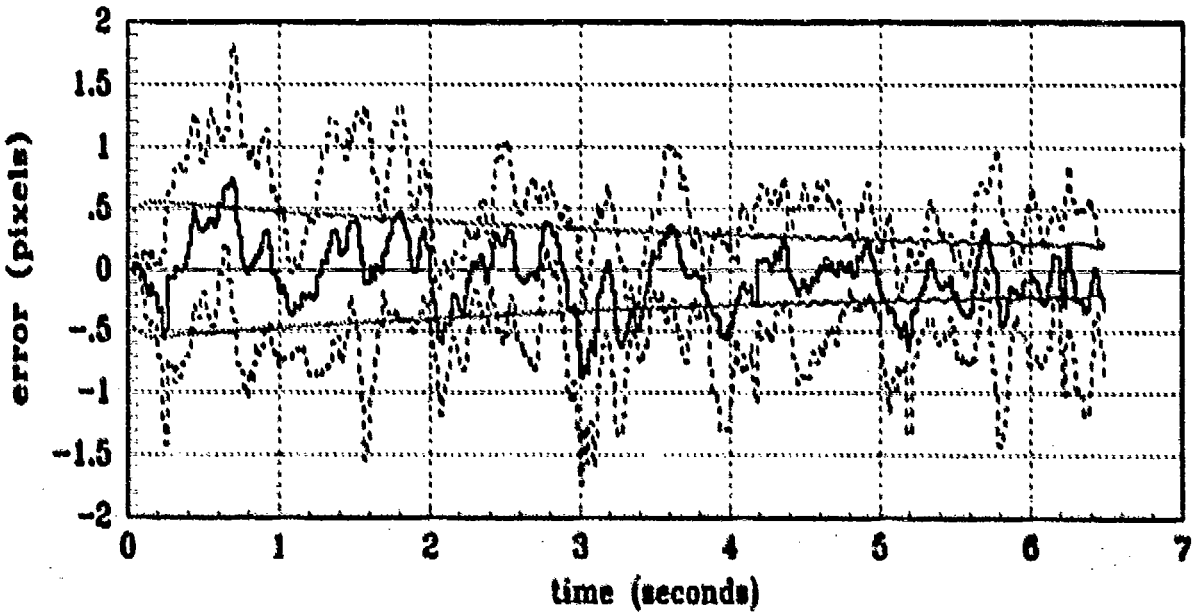


(b) FILTER vs ACTUAL ERROR (POGO VELOCITY)

Figure C.6 Pogo Velocity (Filter State 8)
(Truth = $0.112/2\pi$, Filter = $0.112/2\pi$)

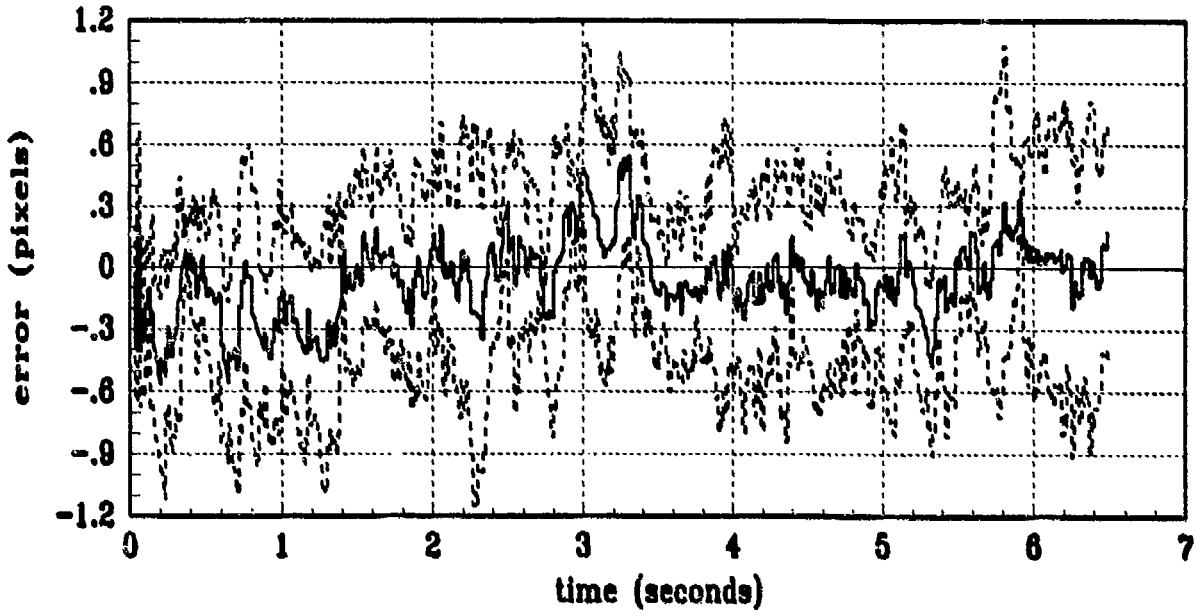


(a) FILTER vs TRUE COM-EQUILIBRIUM POINT OFFSET

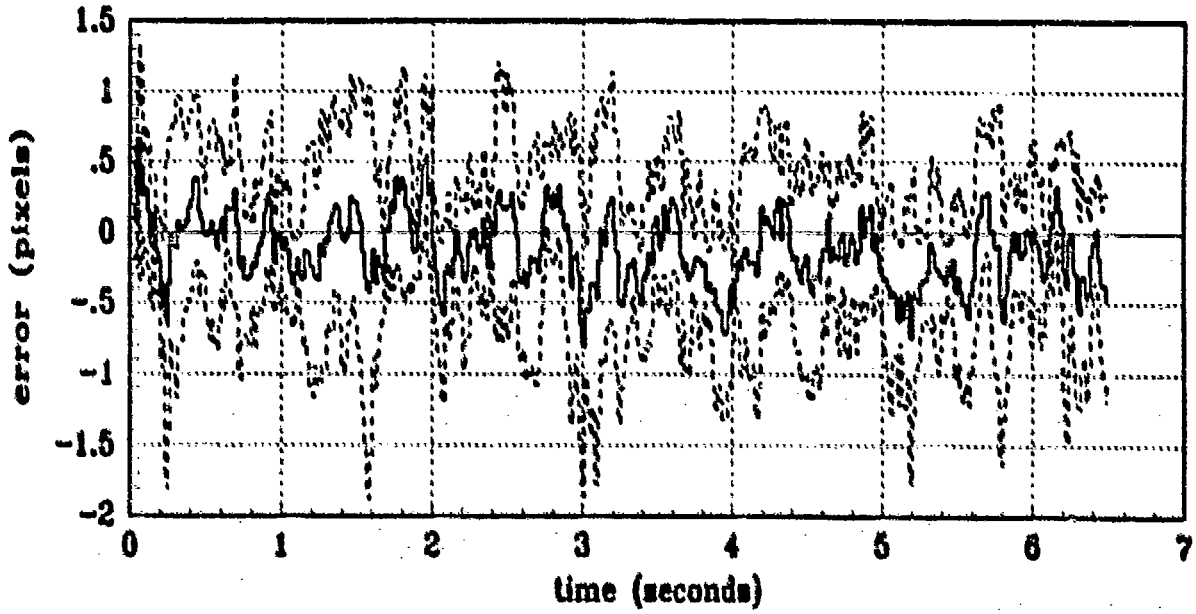


(b) FILTER vs ACTUAL ERROR (COM-EQUILIBRIUM POINT OFFSET)

Figure C.7 Center-of-Mass to Equilibrium Offset (Filter State 9)
(Truth = $0.112/2\pi$, Filter = $0.112/2\pi$)



(a) FILTER vs ACTUAL ERROR (X-CENTROID POSITION)



(b) FILTER vs ACTUAL ERROR (Y-CENTROID POSITION)

Figure C.8 Plume Centroid Error Statistics
(Truth = $0.112/2\pi$, Filter = $0.112/2\pi$)

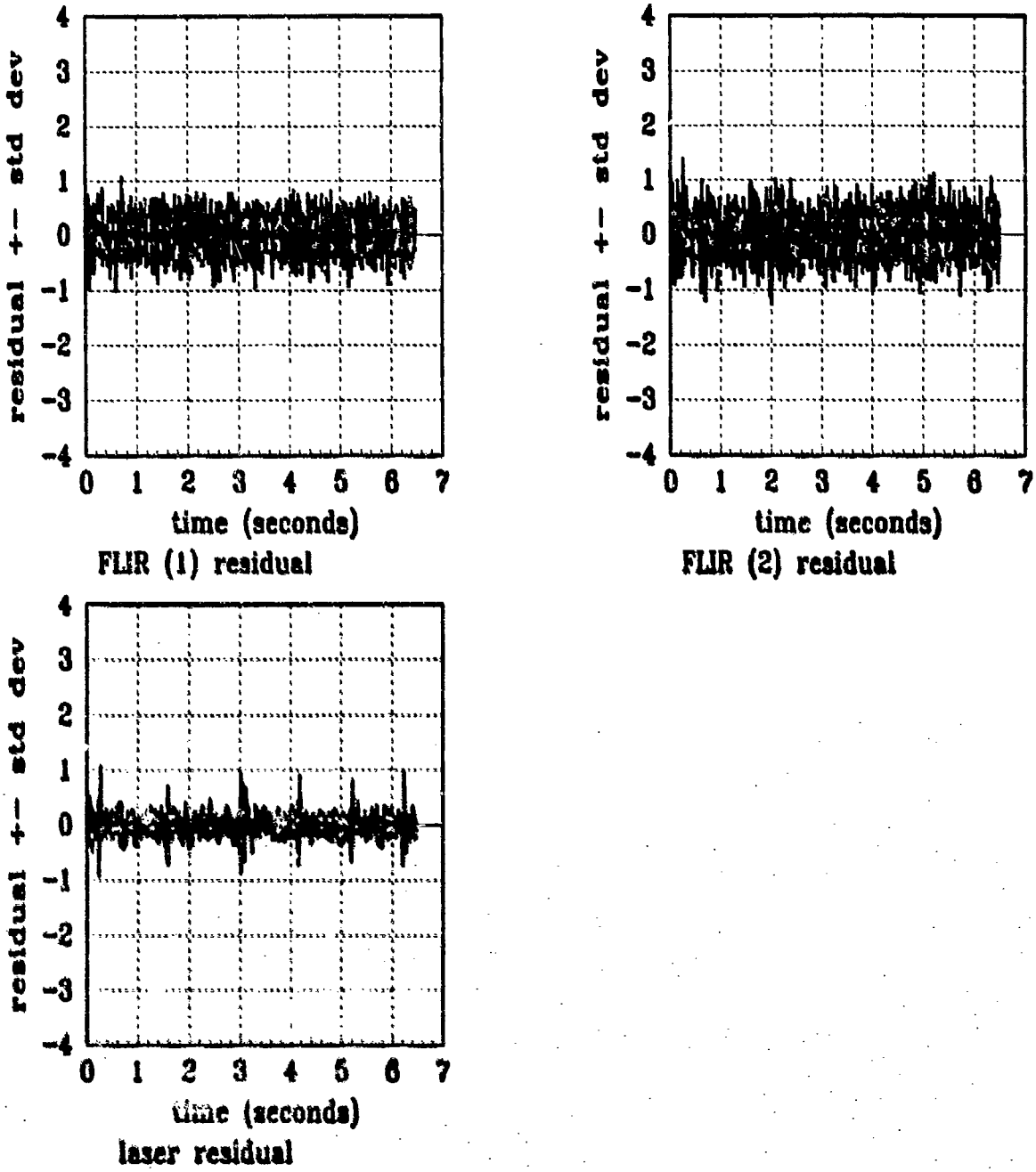
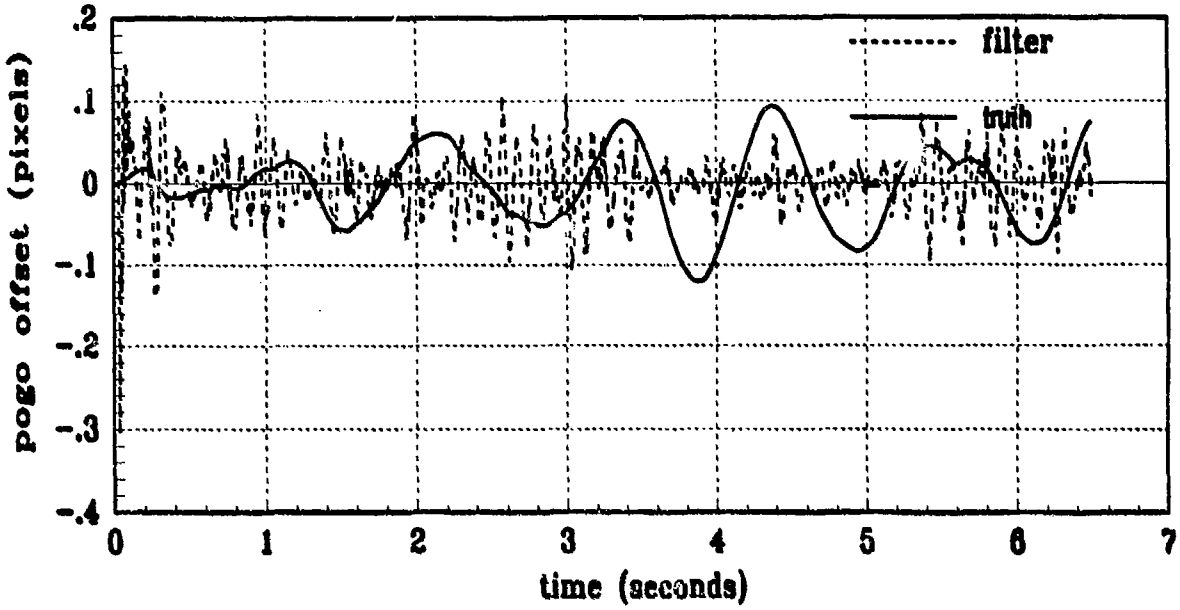
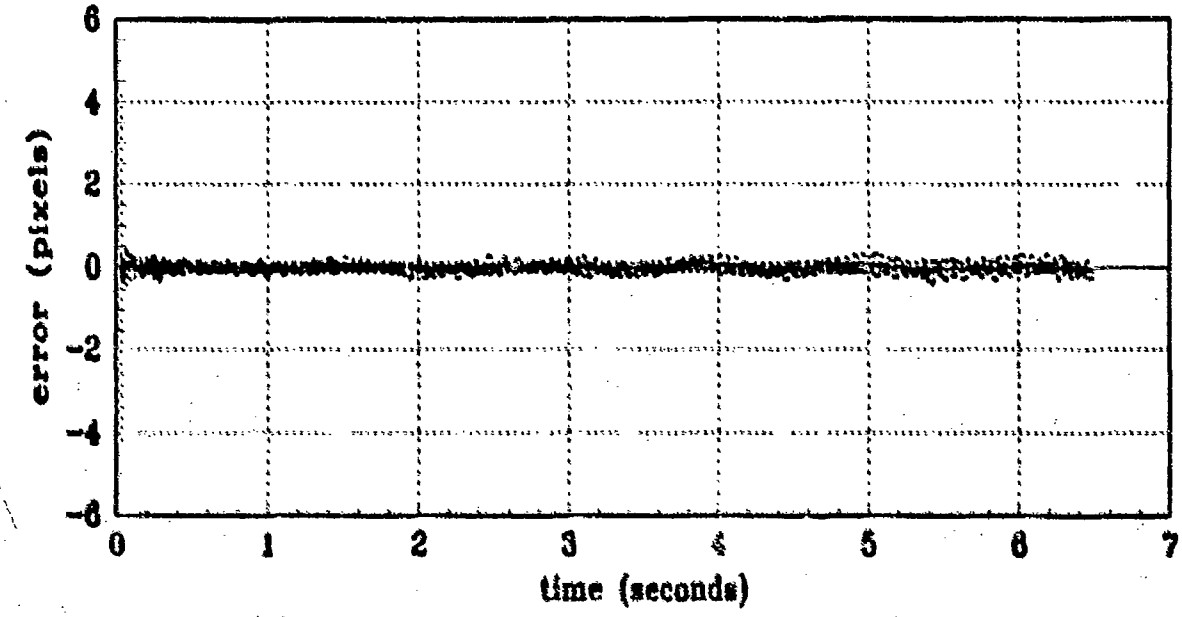


Figure C.9 Filter Residual Quantities
(Truth = 0.112/2π, Filter = 0.112/2π)



(a) FILTER vs TRUE POGO



(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure C.10 Pogo Position (Filter State 7)
(Truth = $0.112/2\pi$, Filter = $0.112/20\pi$)

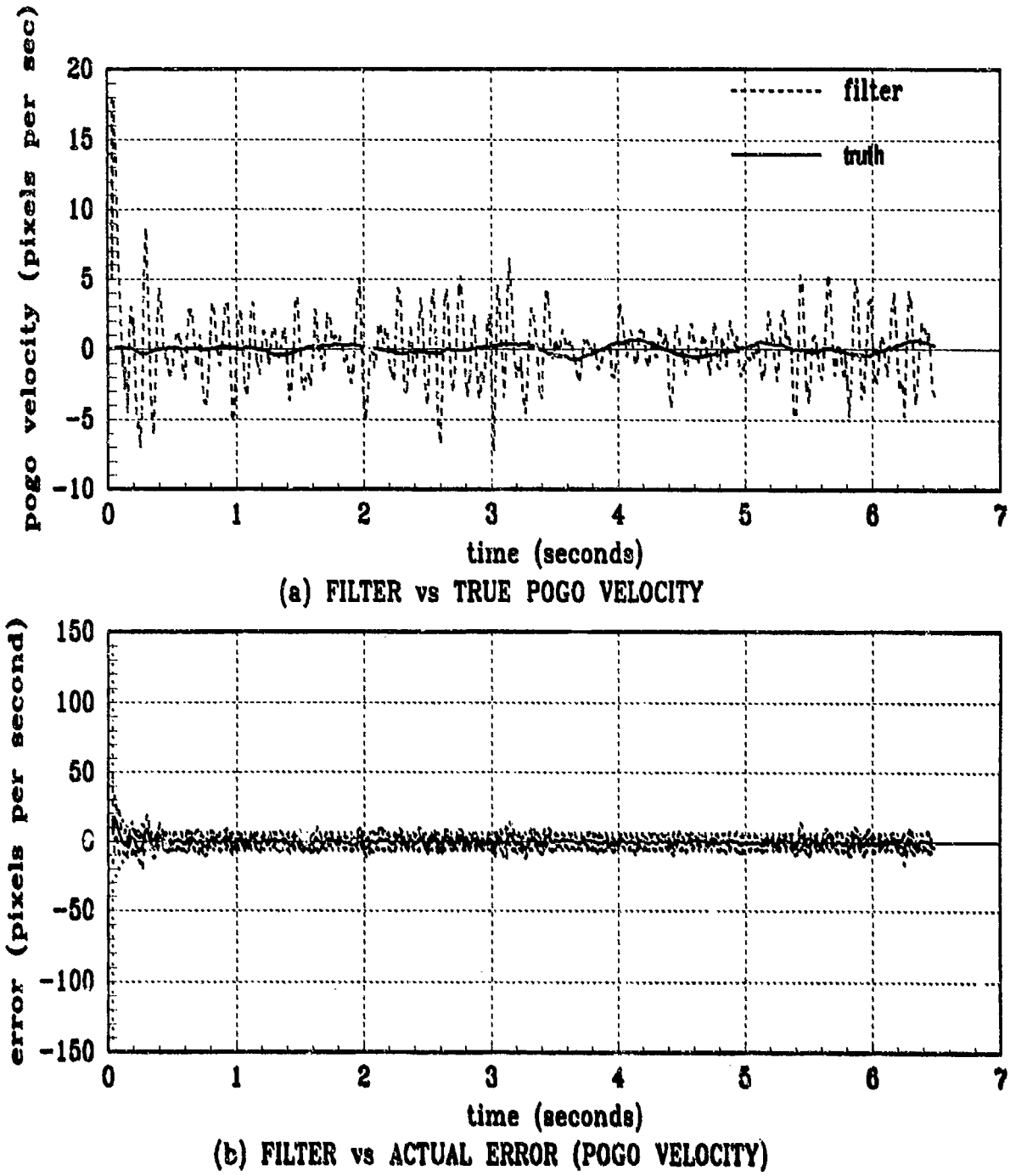
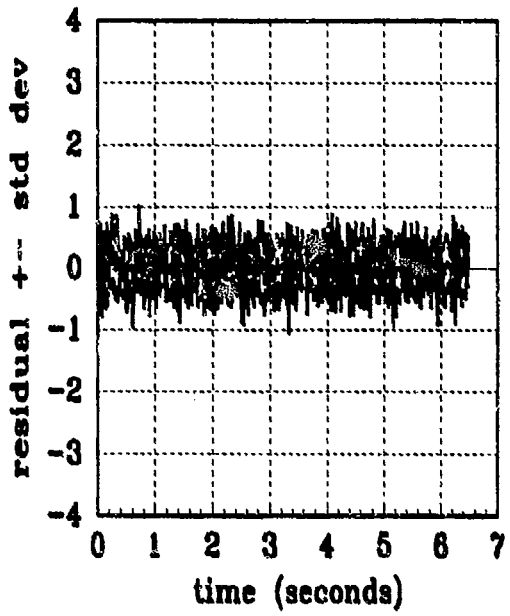
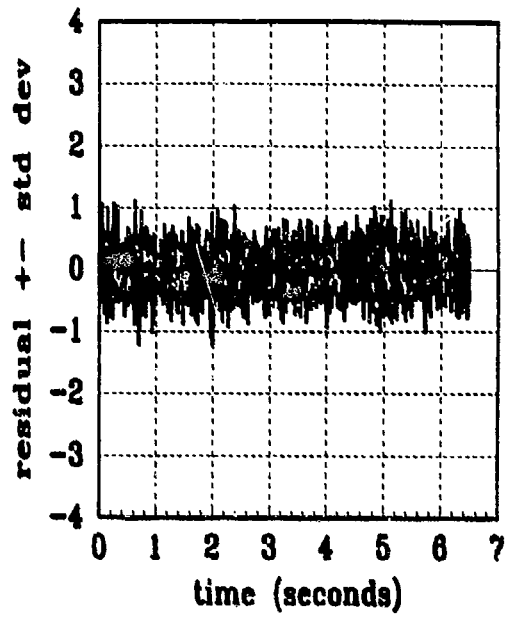


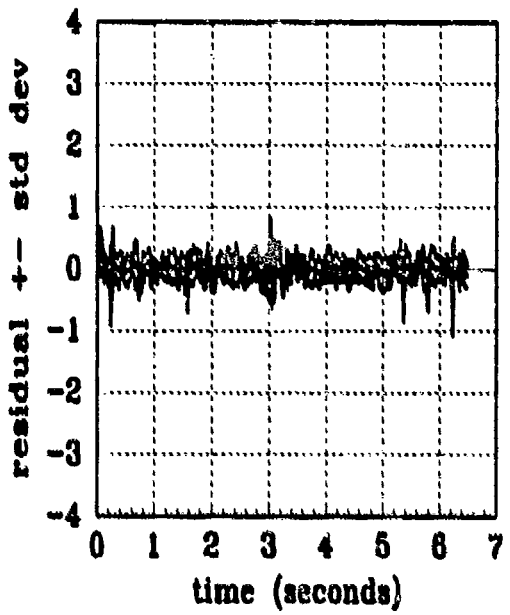
Figure C.11 Pogo Velocity (Filter State 8)
(Truth = $0.112/2\pi$, Filter = $0.112/20\pi$)



FLIR (1) residual



FLIR (2) residual



laser residual

Figure C.12 Filter Residual Quantities
(Truth = $0.112/2\pi$, Filter = $0.112/20\pi$)

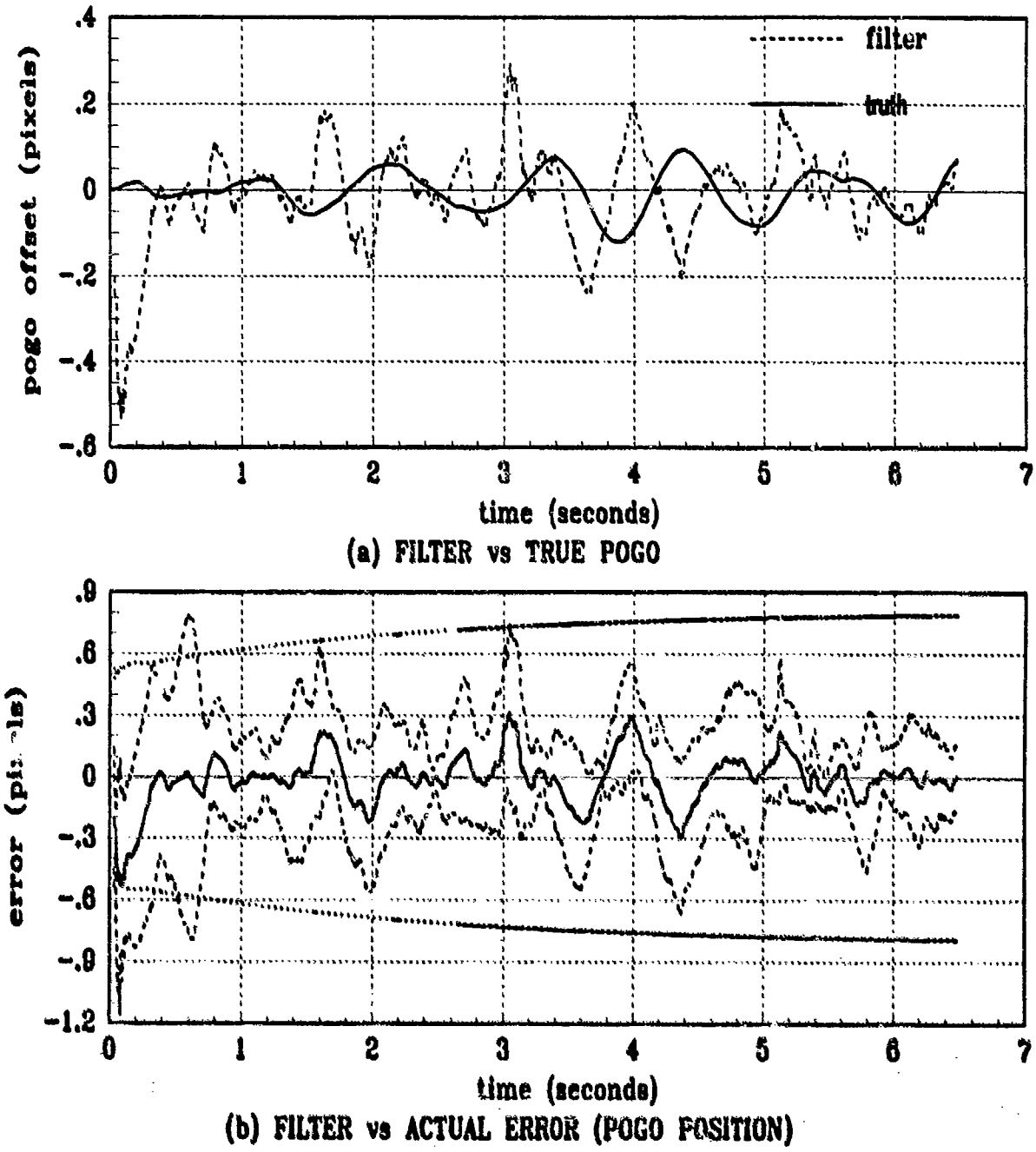
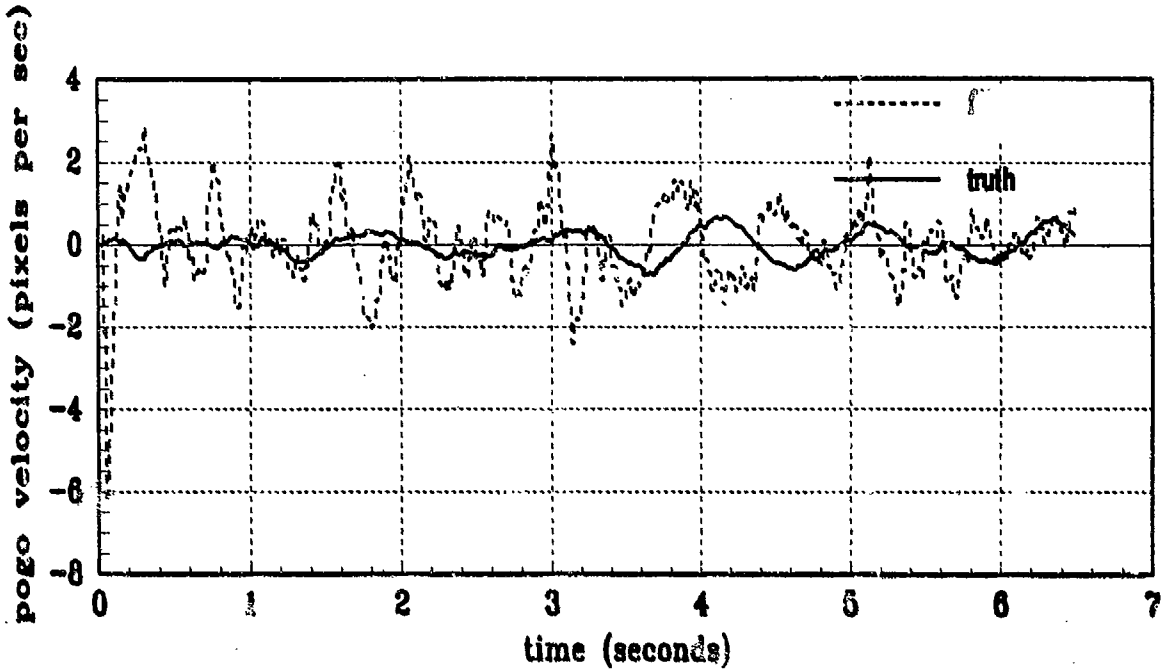
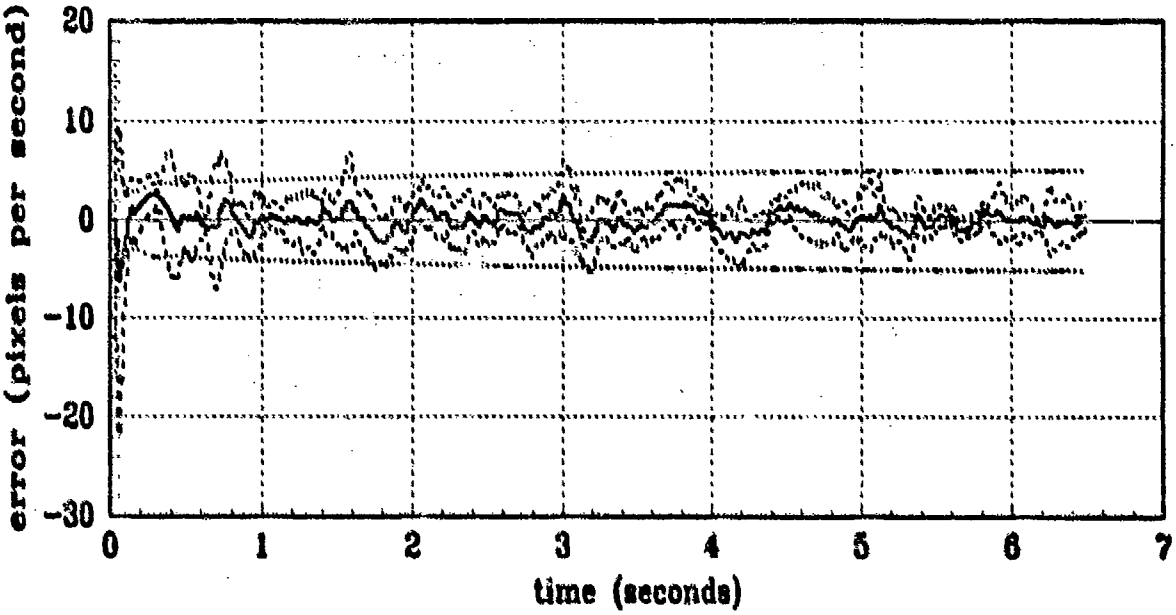


Figure C.13 Pogo Position (Filter State 7)
(Truth = $0.112/2\pi$, Filter = $1.12/2\pi$)



(a) FILTER vs TRUE POGO VELOCITY



(b) FILTER vs ACTUAL ERROR (POGO VELOCITY)

Figure C.14 Pogo Velocity (Filter State 8)
(Truth = $0.112/2\pi$, Filter = $1.12/2\pi$)

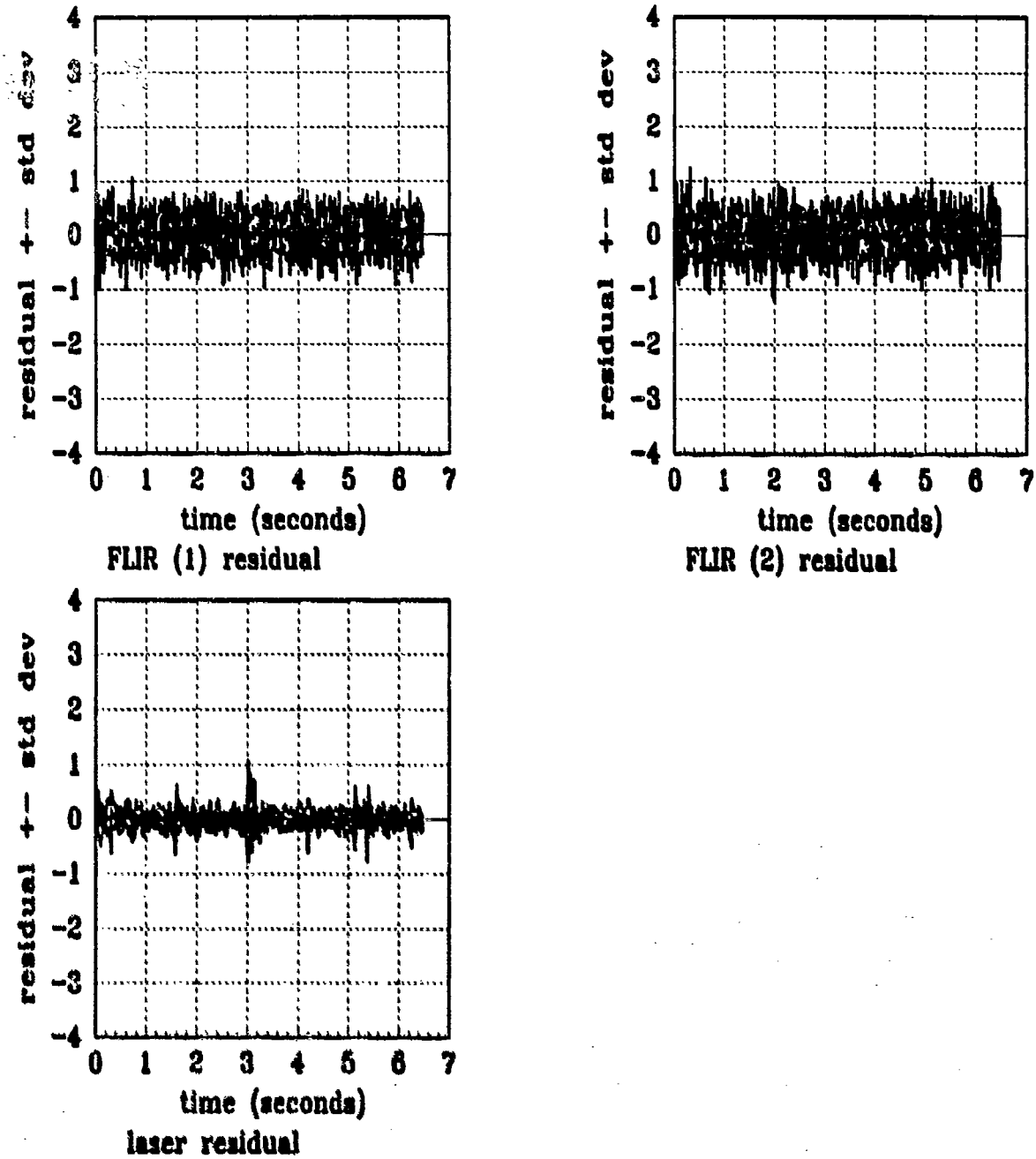
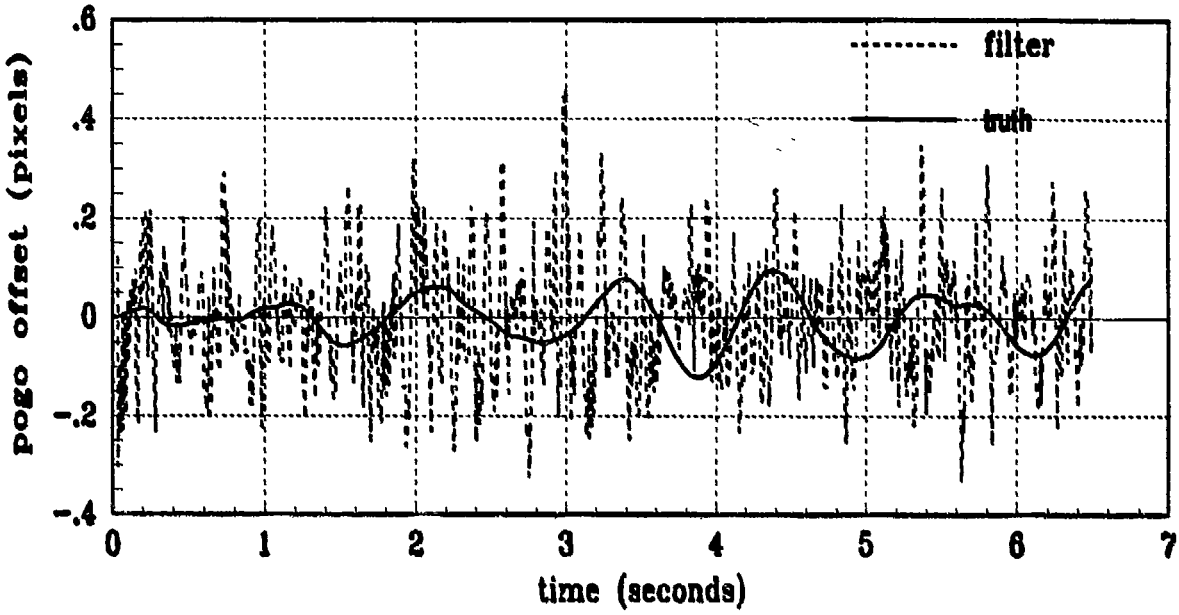
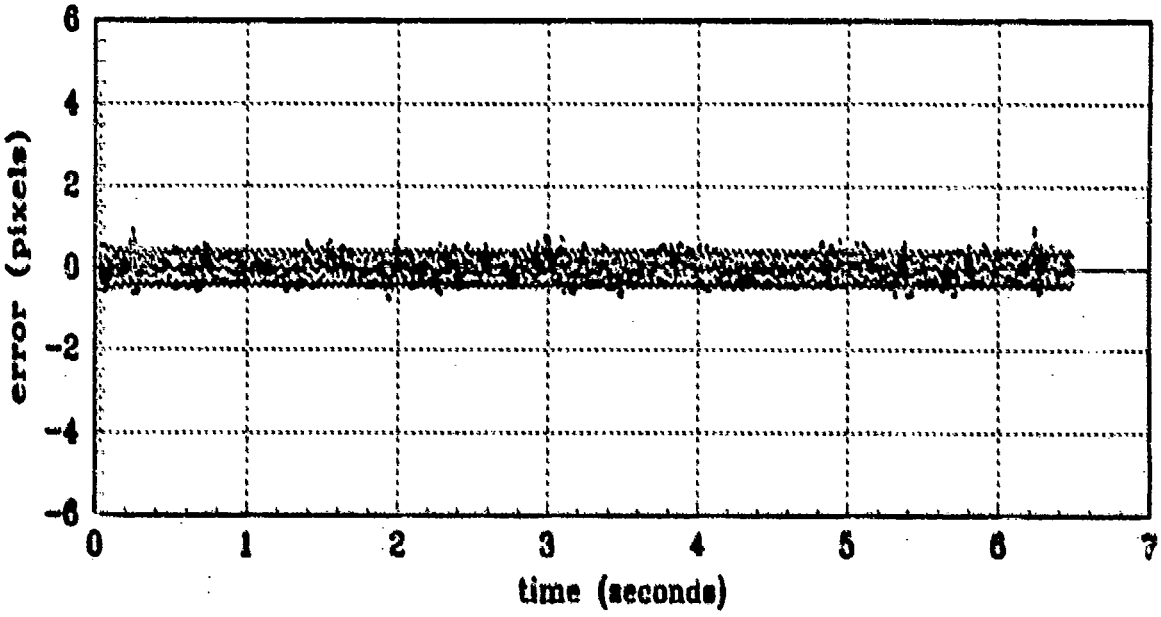


Figure C.15 Filter Residual Quantities
(Truth = $0.112/2\pi$, filter = $1.12/2\pi$)



(a) FILTER vs TRUE POGO



(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure C.16 Pogo Position (Filter State 7)
(Truth = $0.112/2\pi$, Filter = $1.12/20\pi$)

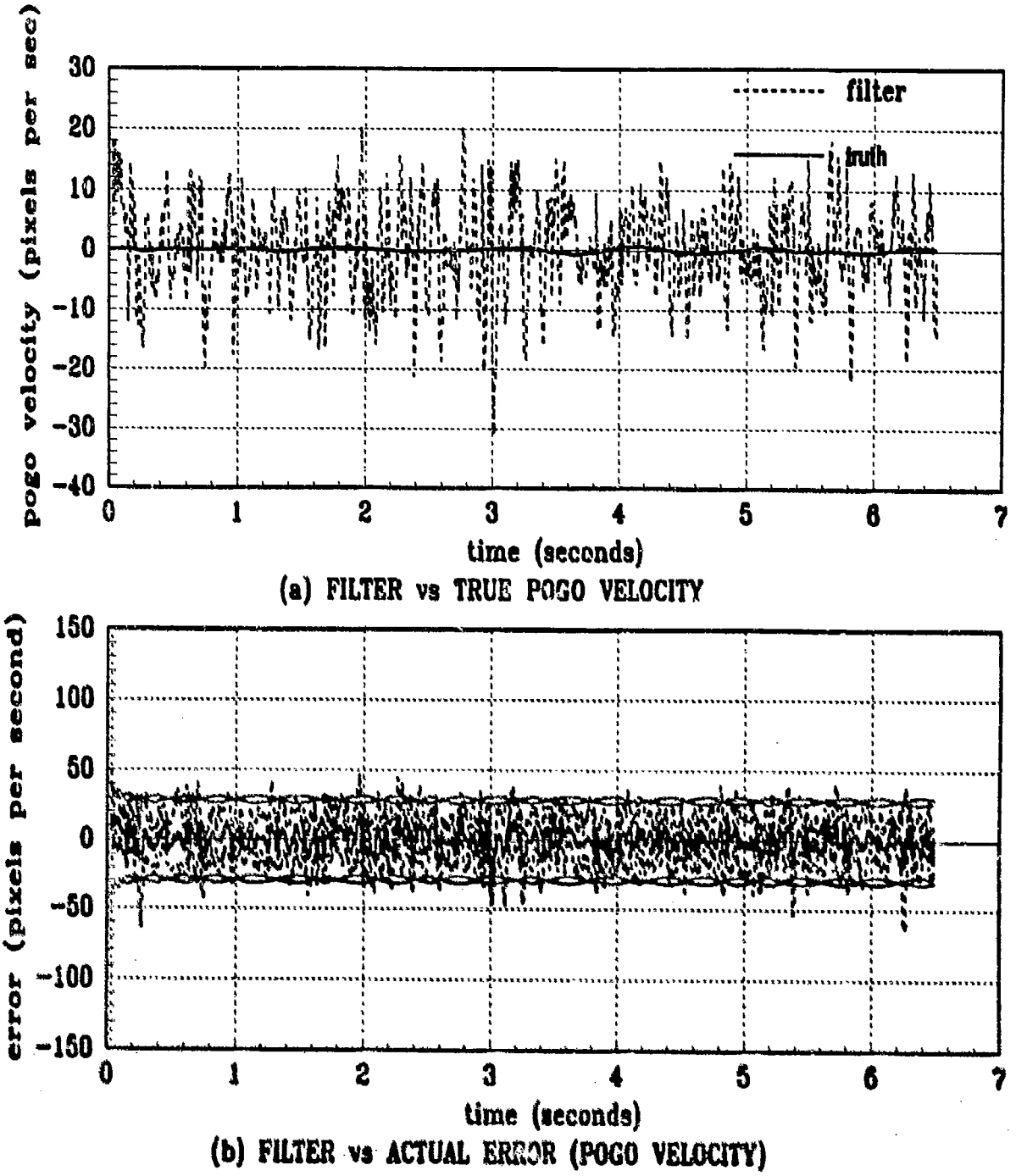
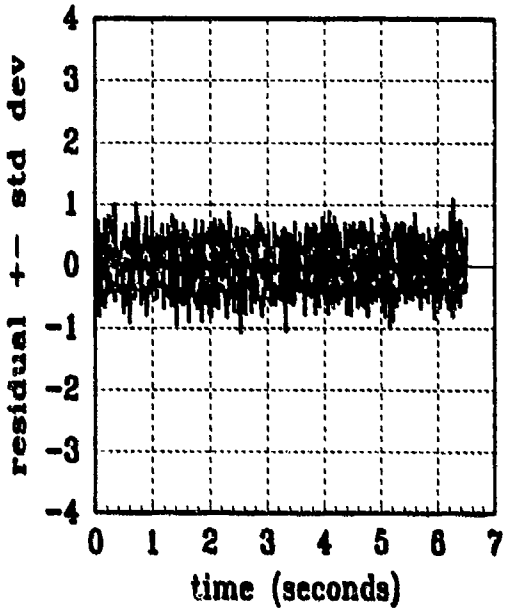
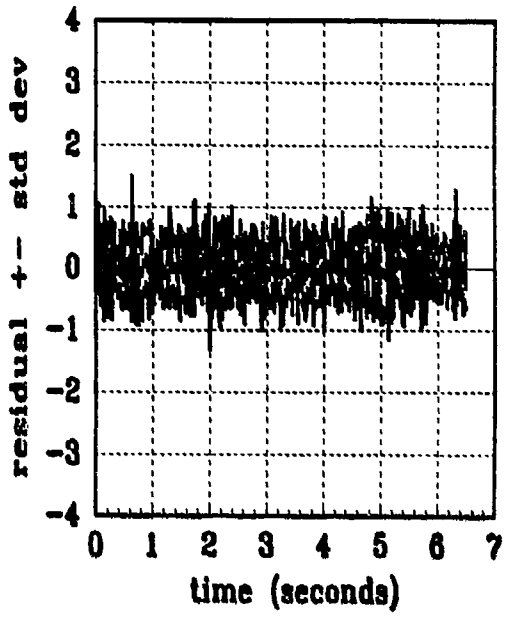


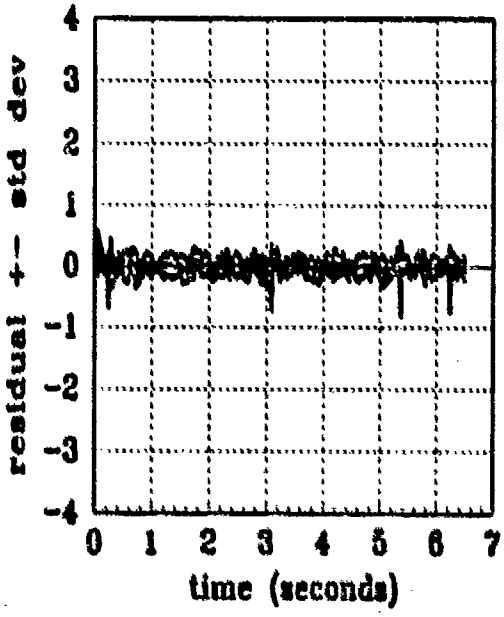
Figure C.17 Pogo Velocity (Filter State 8)
(Truth = $0.112/2\pi$, Filter = $1.12/20\pi$)



FLIR (1) residual



FLIR (2) residual



laser residual

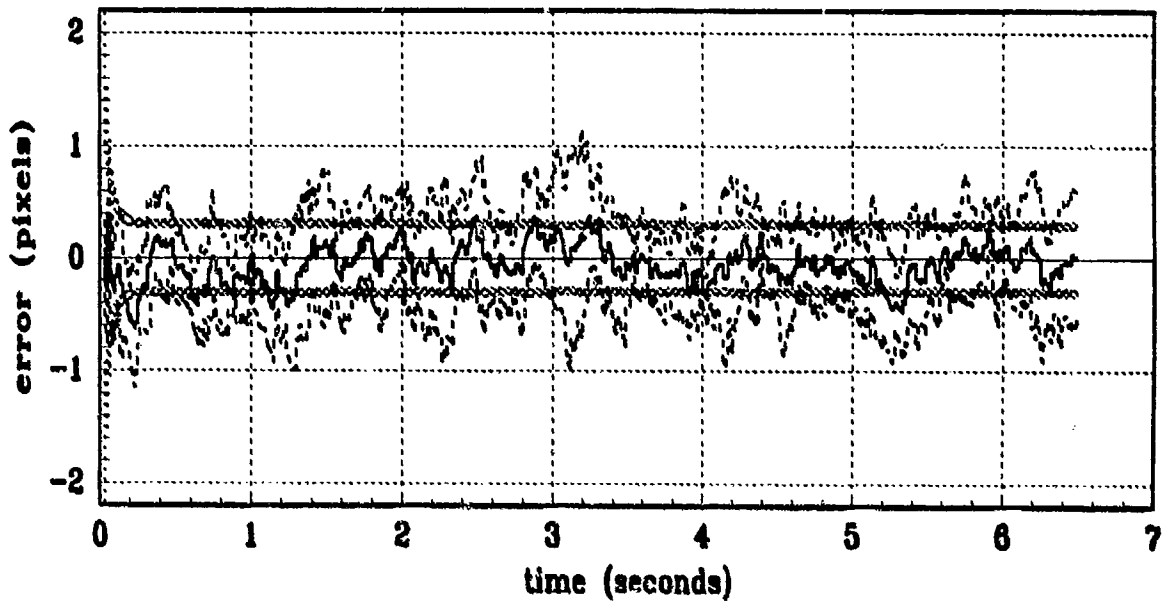
Figure C.18 Filter Residual Quantities
(Truth = $0.112/2\pi$, Filter = $1.12/20\pi$)

Appendix D

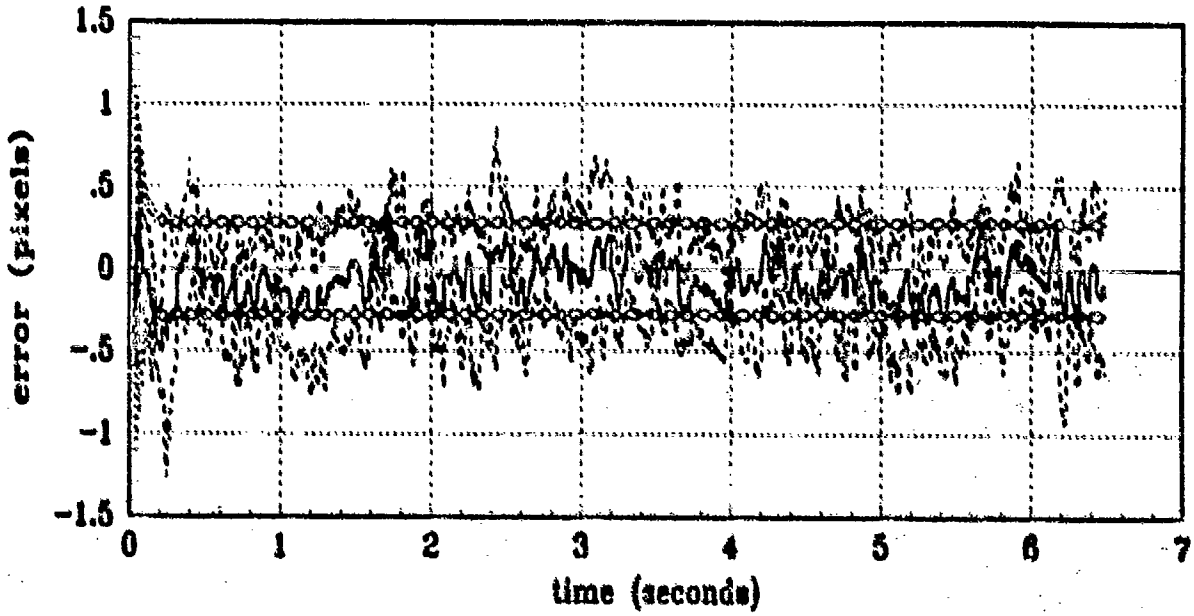
Filter Plots with Truth Parameter Values:

Amplitude = 0.112, Frequency = 20π

This appendix contains the state and error statistics plots of the nine-state elemental filters. The data depicted in the two types of plots in this appendix are explained in ***Appendix A***. The state comparison plots show the sample mean truth state over the 5 ***Monte Carlo*** runs compared to the same statistic for the filter estimate. The error statistics plots represent the error mean $\pm$ standard deviation values in pixels (or pixel/second for velocity and pogo velocity), of the errors between the filter estimate and true state; true mean $\pm$ 1 true standard deviation are plotted, along with zero $\pm$ 1 filter-computed standard deviation..

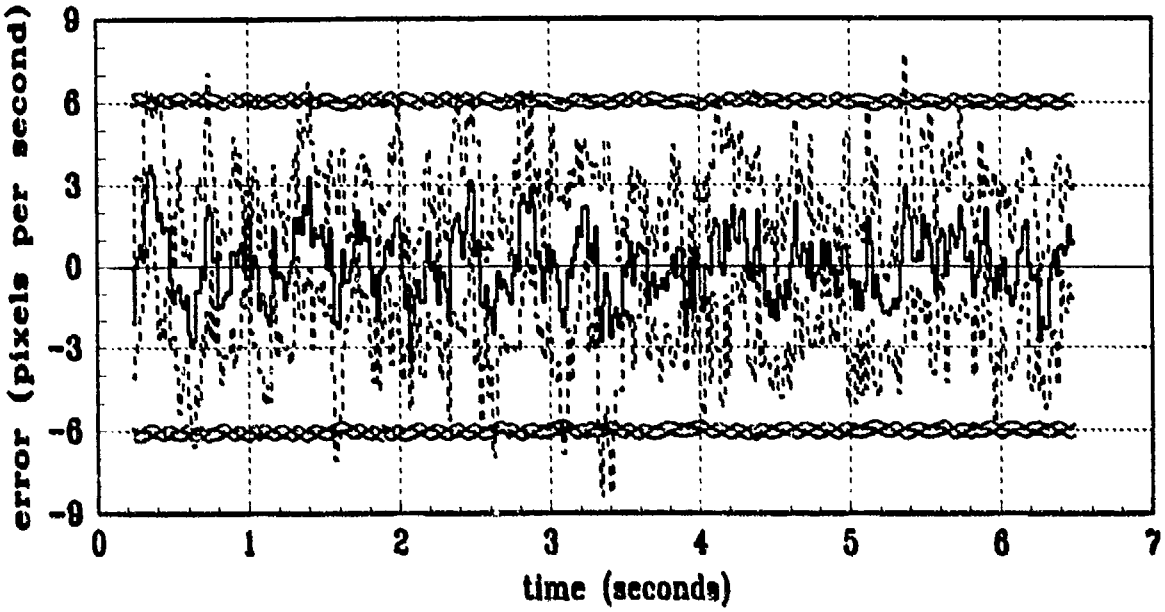


(a) FILTER vs ACTUAL ERROR (X-POSITION)

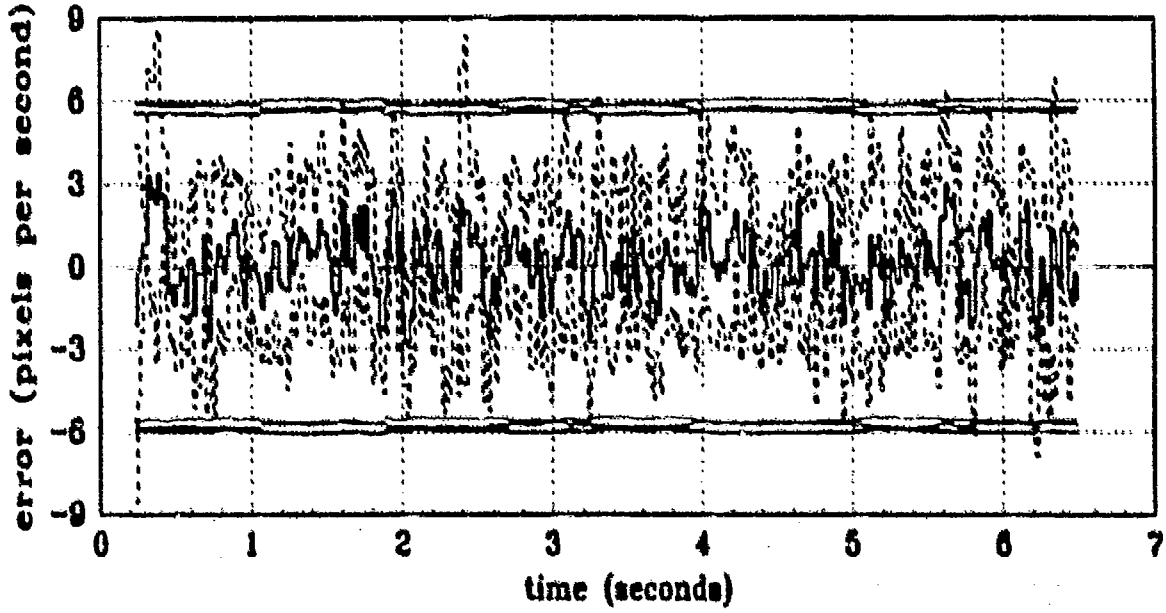


(b) FILTER vs ACTUAL ERROR (Y-POSITION)

Figure D.1 X/Y Position (Filter States 1 and 2) Error Statistics
(Truth = 0.112/20π, Filter = 0.112/20π)



(a) FILTER vs ACTUAL ERROR (X-VELOCITY)



(b) FILTER vs ACTUAL ERROR (Y-VELOCITY)

Figure D.2 X/Y Velocity (Filter States 3 and 4) Error Statistics
(Truth = 0.112/20π, Filter = 0.112/20π)

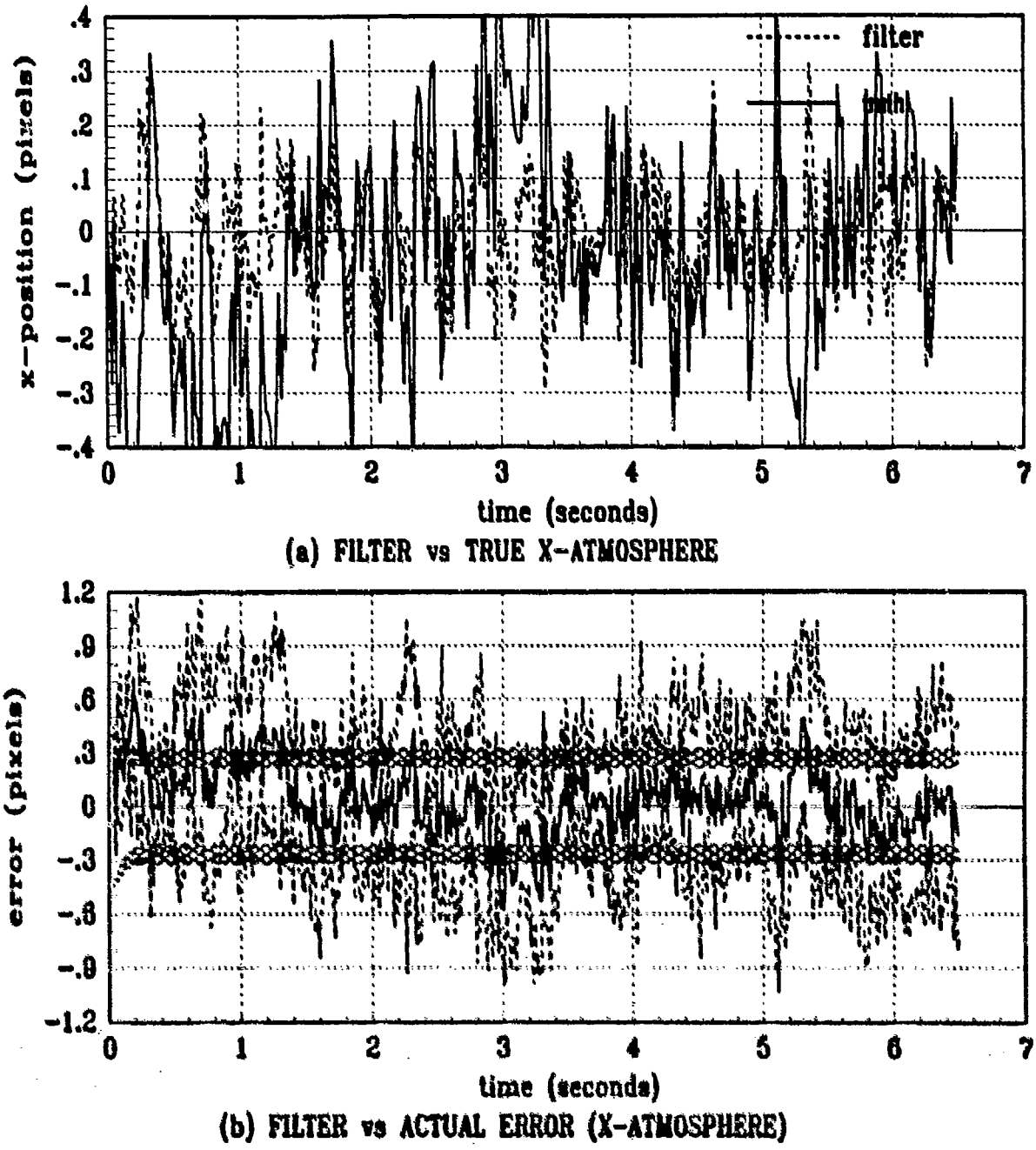
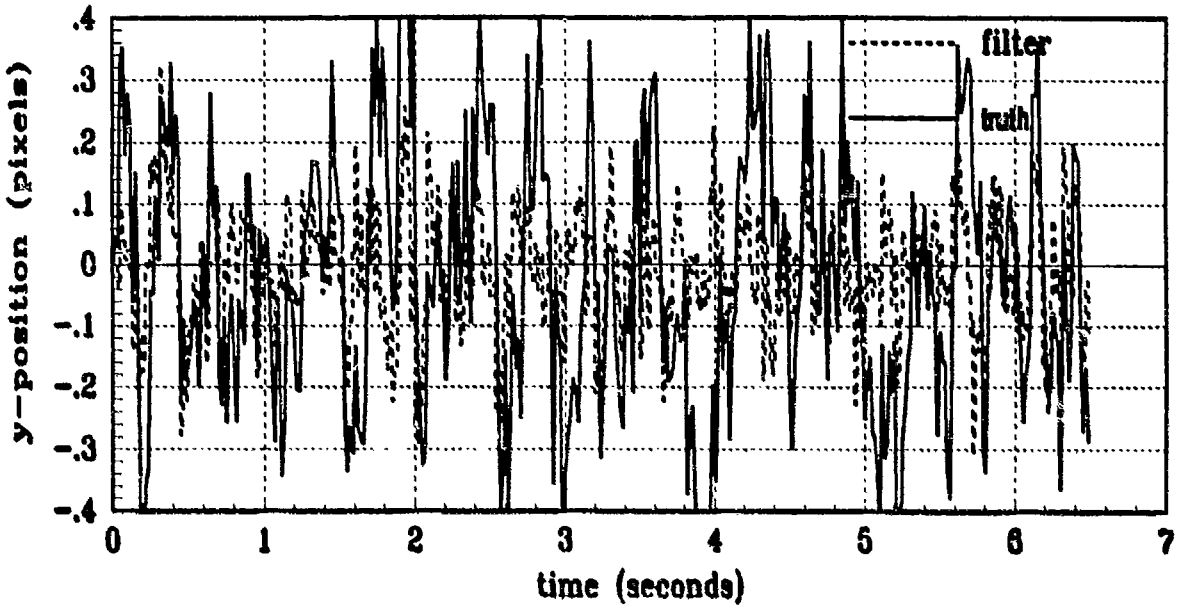
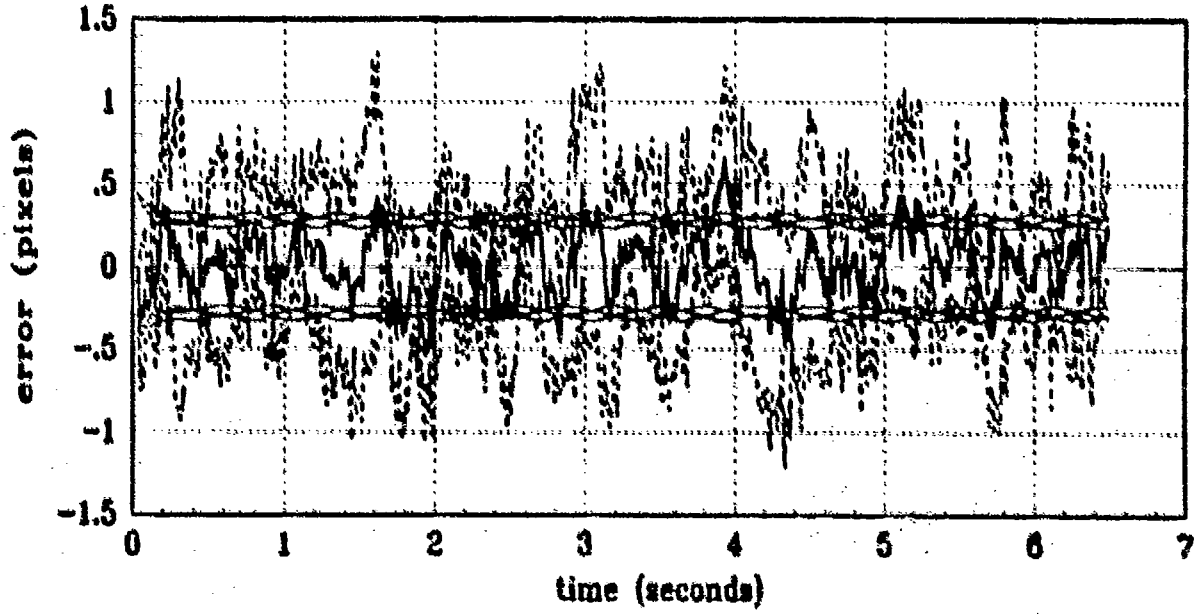


Figure D.3 X Atmospheric Jitter (Filter State 5)
(Truth = 0.112/20π, Filter = 0.112/20π)

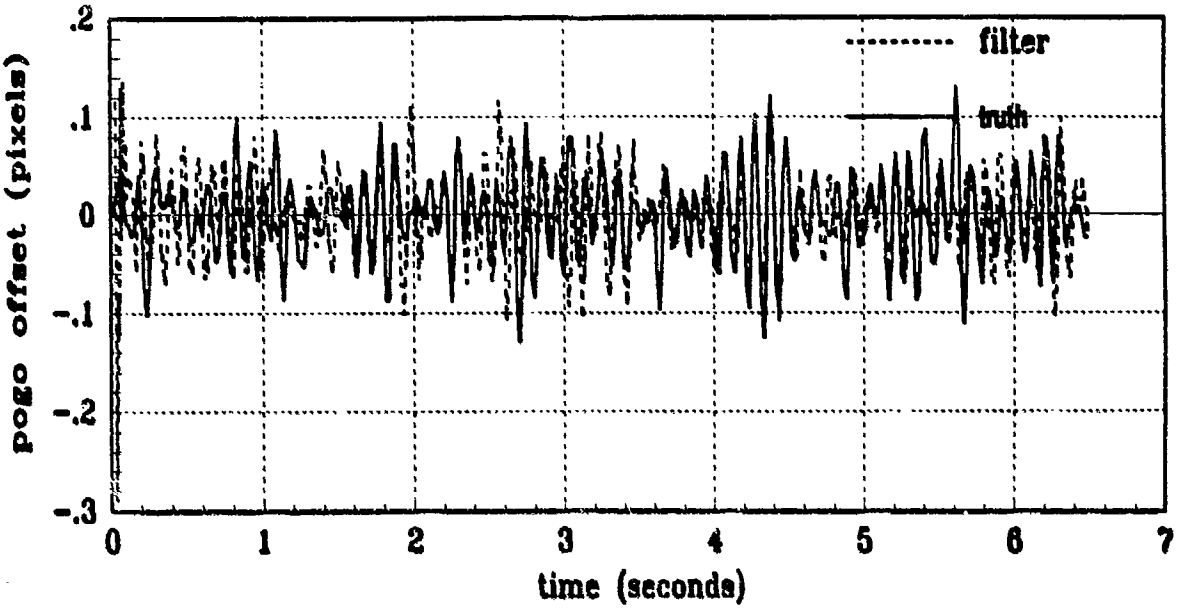


(a) FILTER vs TRUE Y-ATMOSPHERE -

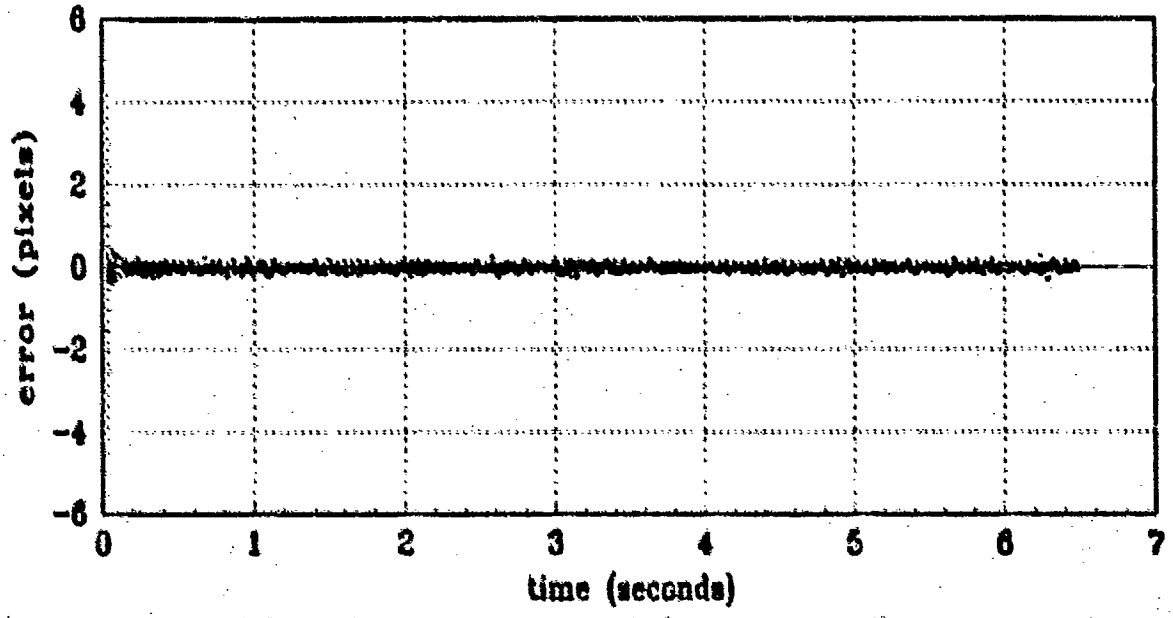


(b) FILTER vs ACTUAL ERROR (Y-ATMOSPHERE)

Figure D.4 Y Atmospheric Jitter (Filter State 6)
(Truth = 0.112/20π, Filter = 0.112/20π)



(a) FILTER vs TRUE POGO



(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure D.5 Pogo Position Offset (Filter State 7)
(Truth = 0.112/20x, Filter = 0.112/20π)

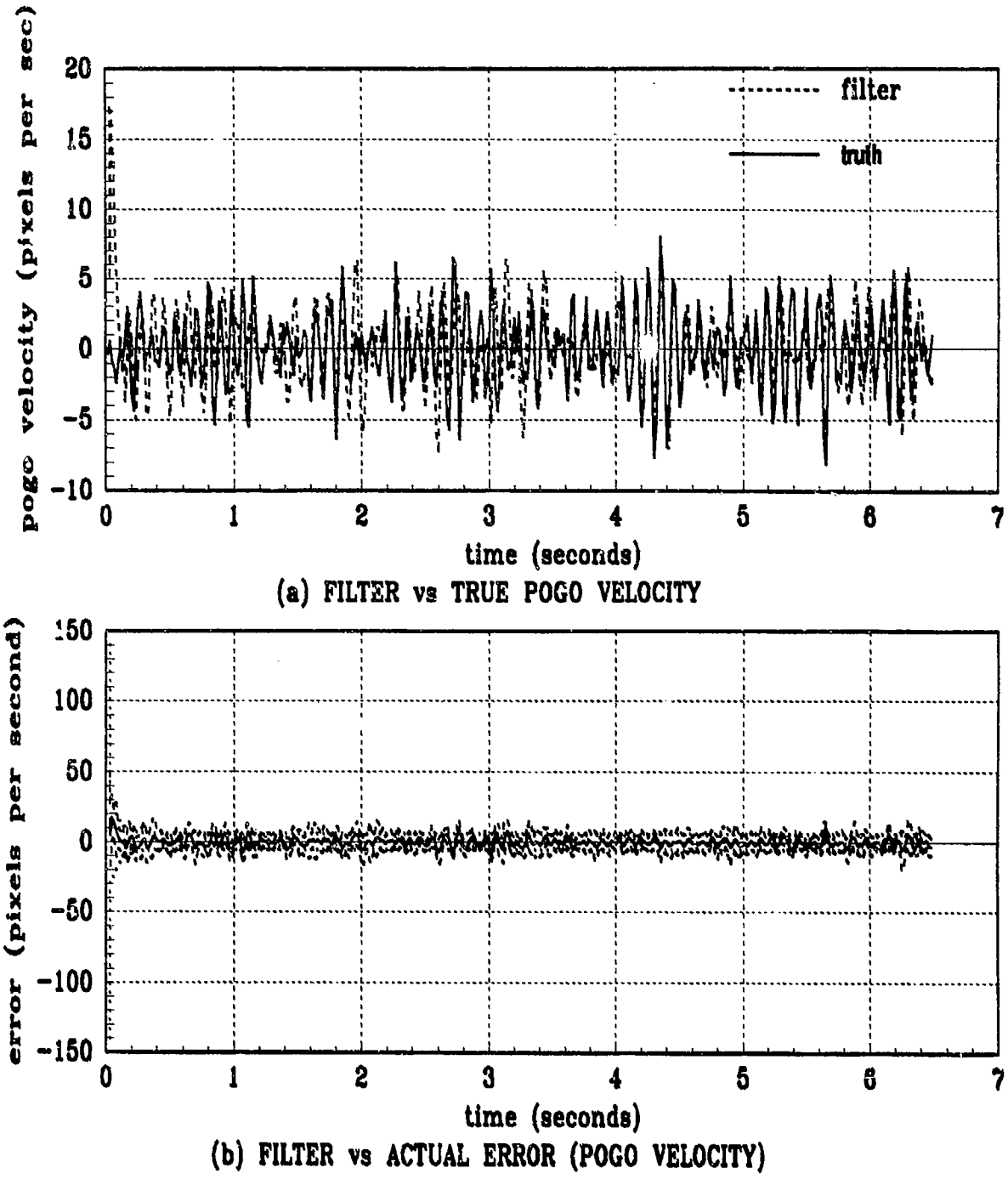


Figure D.6 Pogo Velocity (Filter State 8)
(Truth = $0.112/20\pi$, Filter = $0.112/20\pi$)

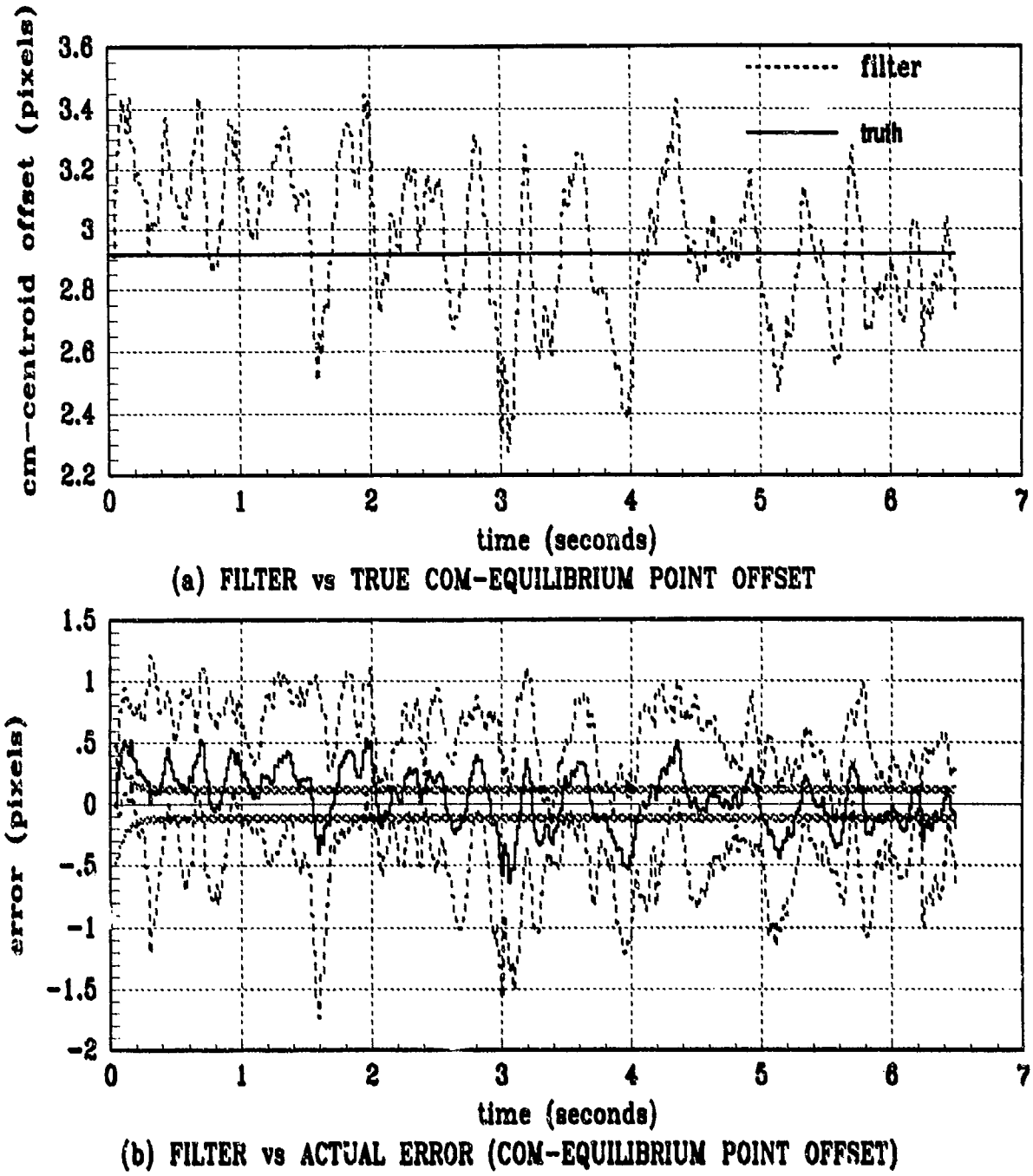
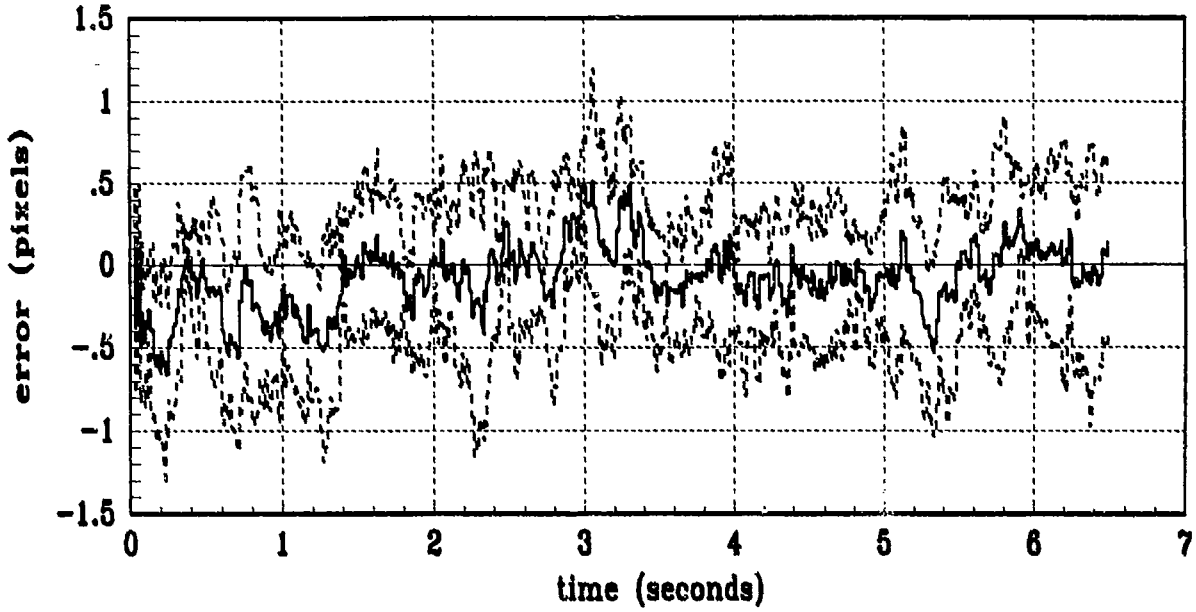
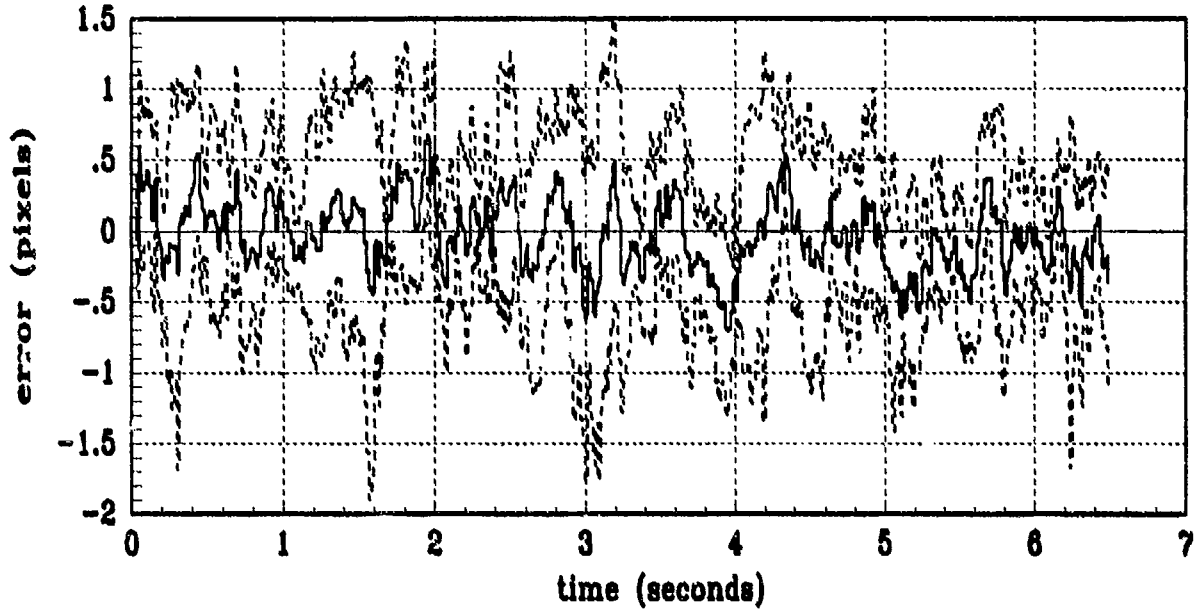


Figure D.7 Center-of-Mass to Equilibrium Offset (Filter State 9)
(Truth = $0.112/20\pi$, Filter = $0.112/2\pi 0$)

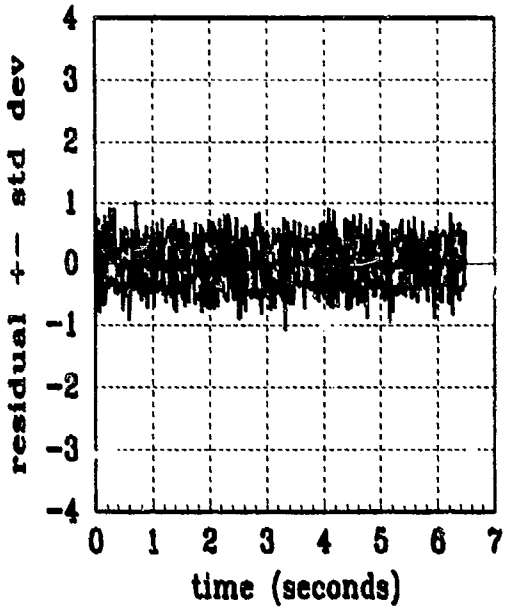


(a) FILTER vs ACTUAL ERROR (X-CENTROID POSITION)

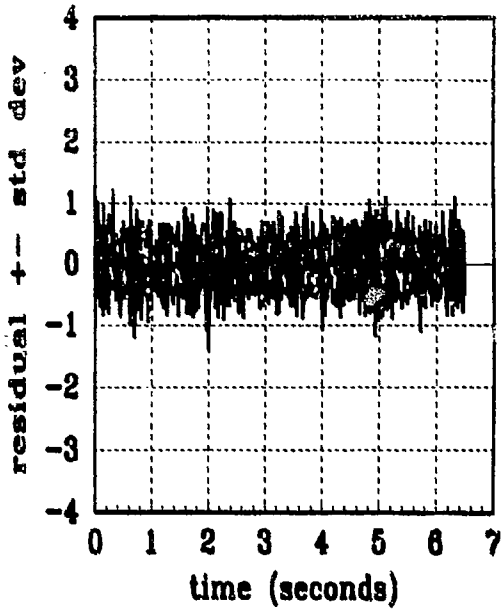


(b) FILTER vs ACTUAL ERROR (Y-CENTROID POSITION)

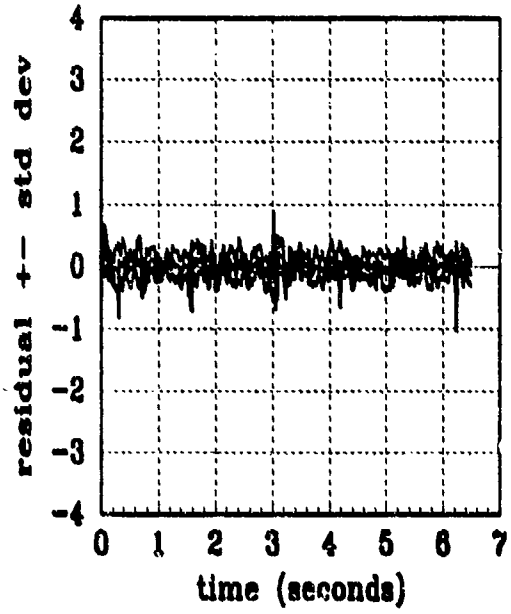
Figure D.8 Plume Centroid Error Statistics
(Truth = $0.112/20\pi$, Filter = $0.112/20\pi$)



FLIR (1) residual



FLIR (2) residual



laser residual

Figure D.9 Filter Residual Quantities
(Truth = $0.112/20\pi$, Filter = $0.112/20\pi$)

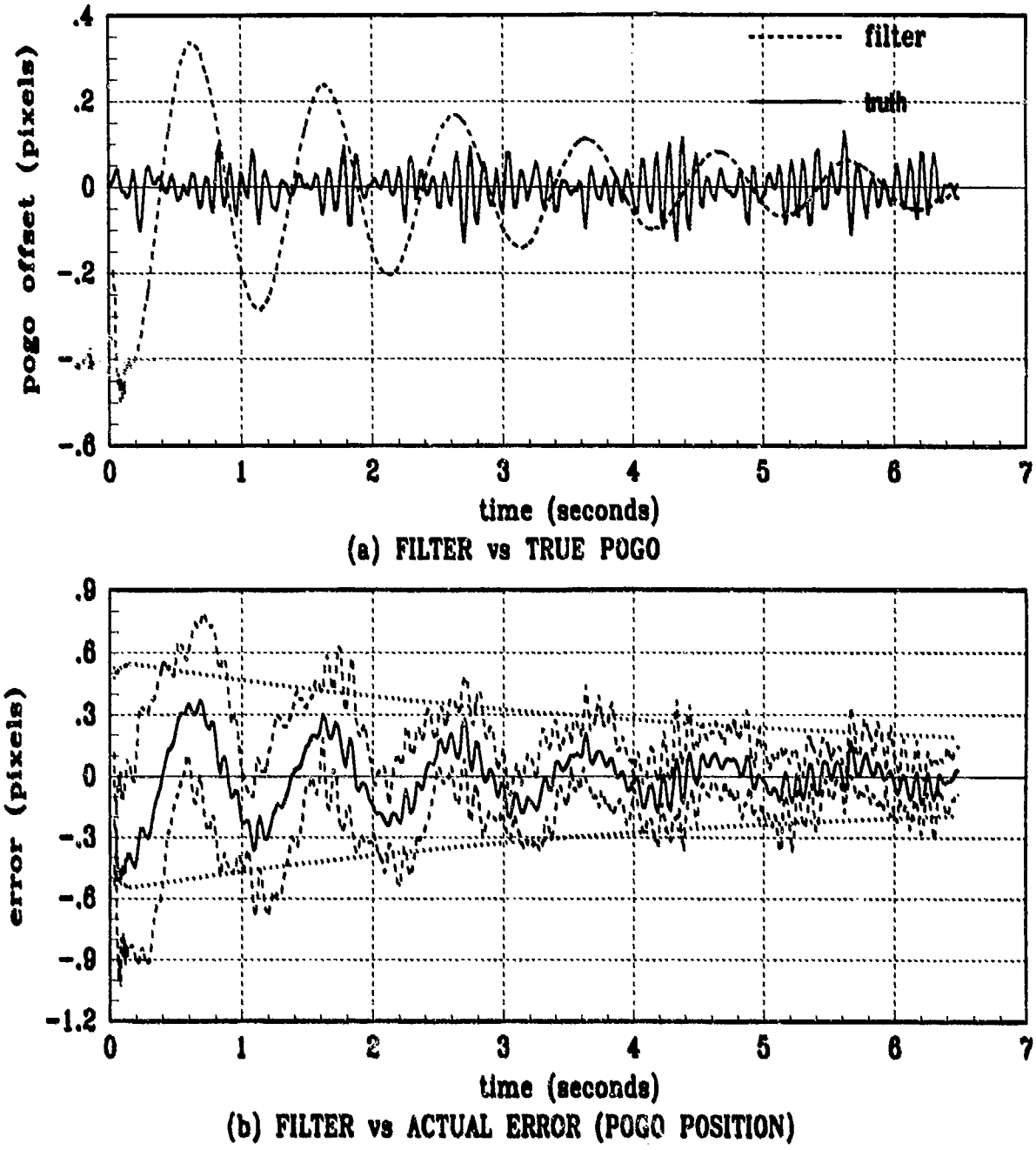


Figure D.10 Pogo Position (Filter State 7)
(Truth = $0.112/20\pi$, Filter = $0.112/2\pi$)

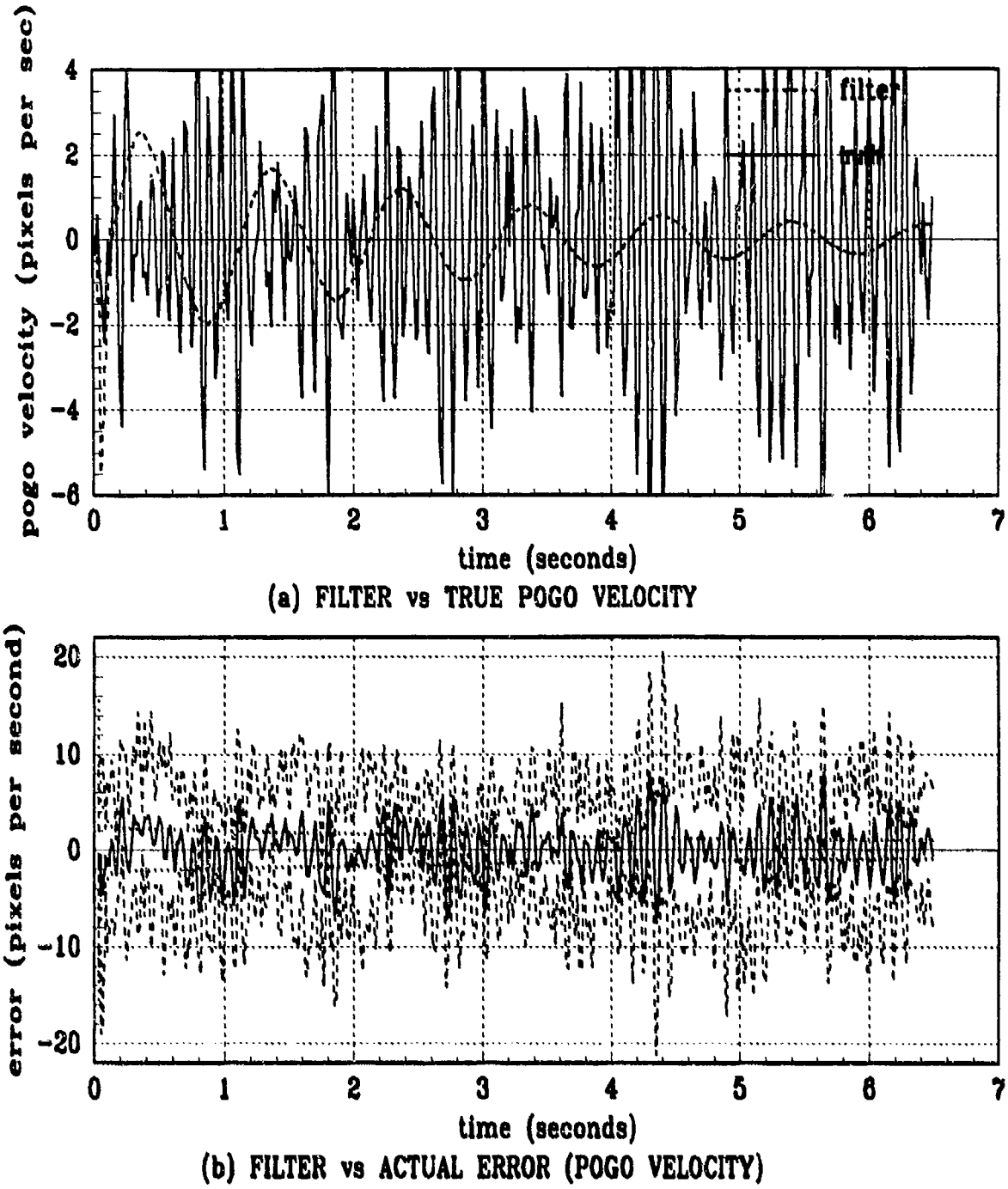
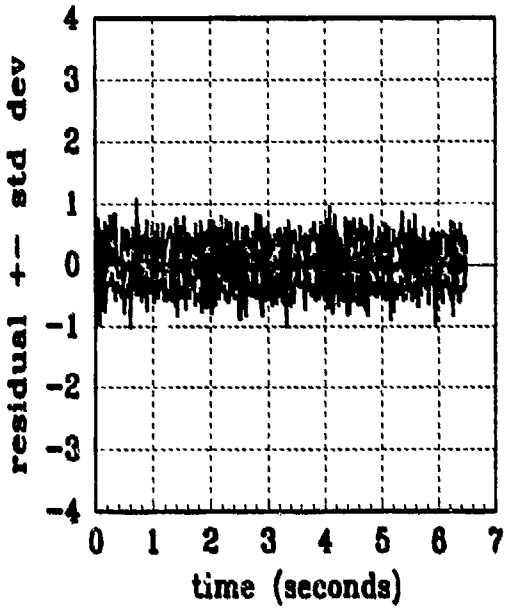
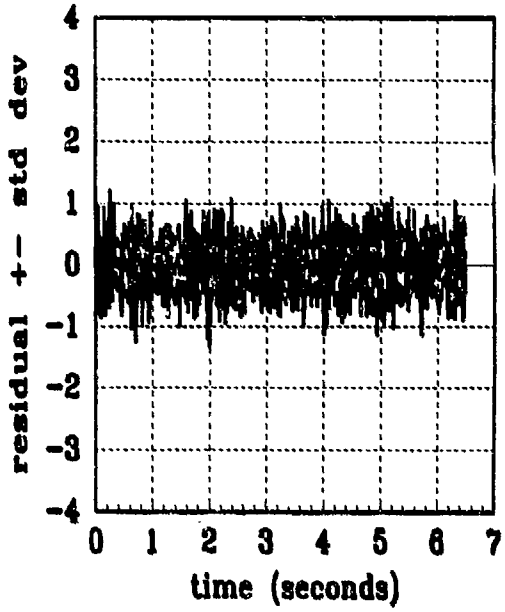


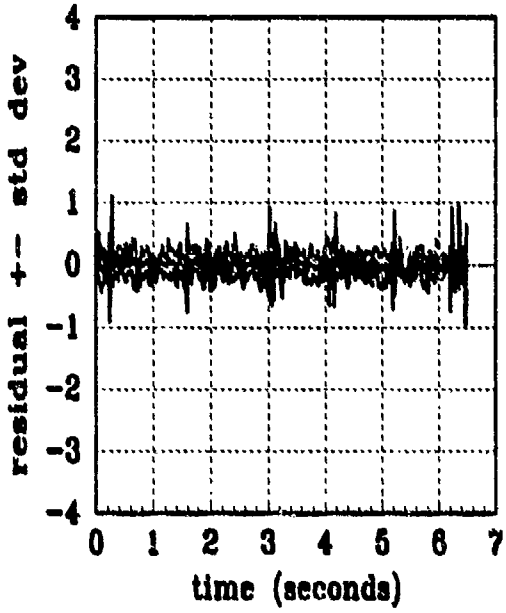
Figure D.11 Pogo Velocity (Filter State 8)
(Truth = $0.112/20\pi$, Filter = $0.112/2\pi$)



FLIR (1) residual

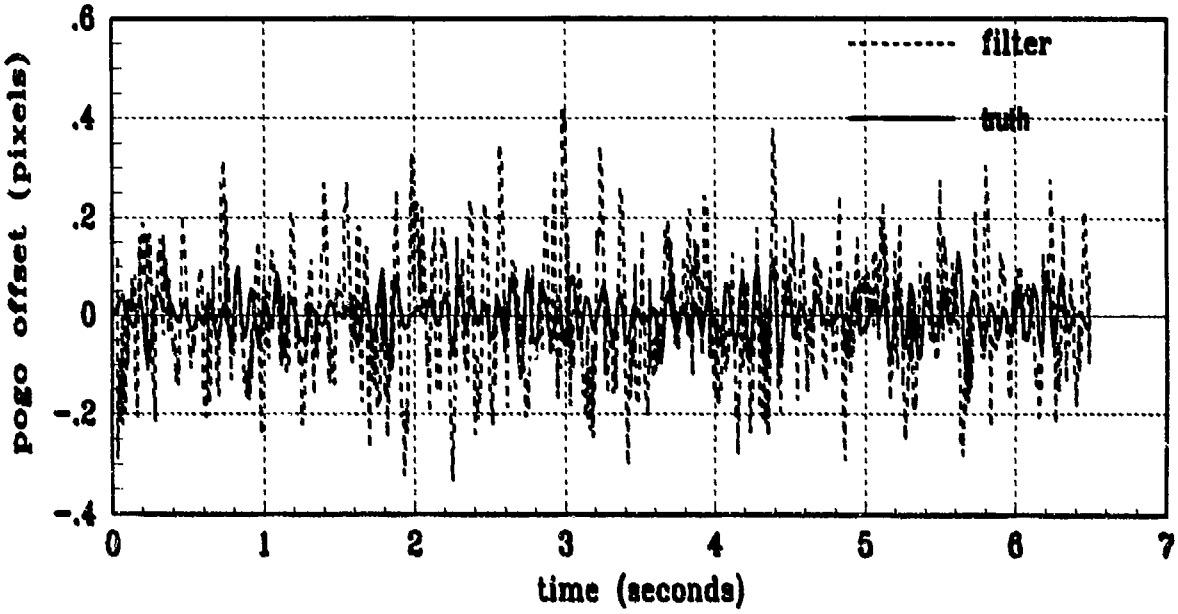


FLIR (2) residual

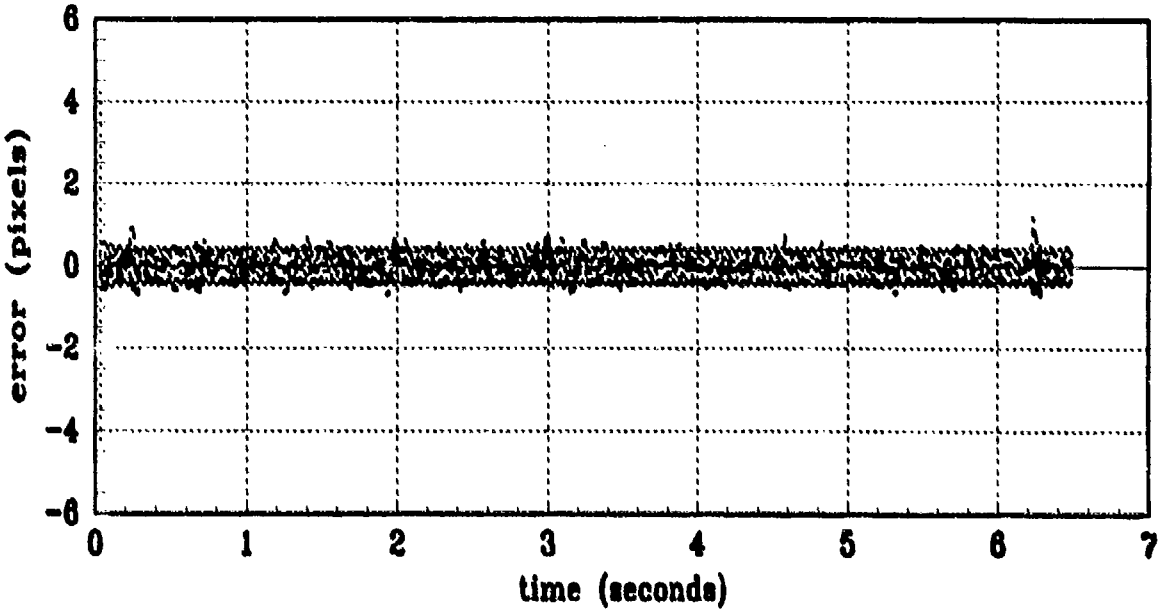


laser residual

Figure D.12 Filter Residual Quantities
(Truth = $0.112/20\pi$, Filter = $0.112/2\pi$)



(a) FILTER vs TRUE POGO



(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure D.13 Pogo Position (Filter State 7)
(Truth = $0.112/20\pi$, Filter = $1.12/20\pi$)

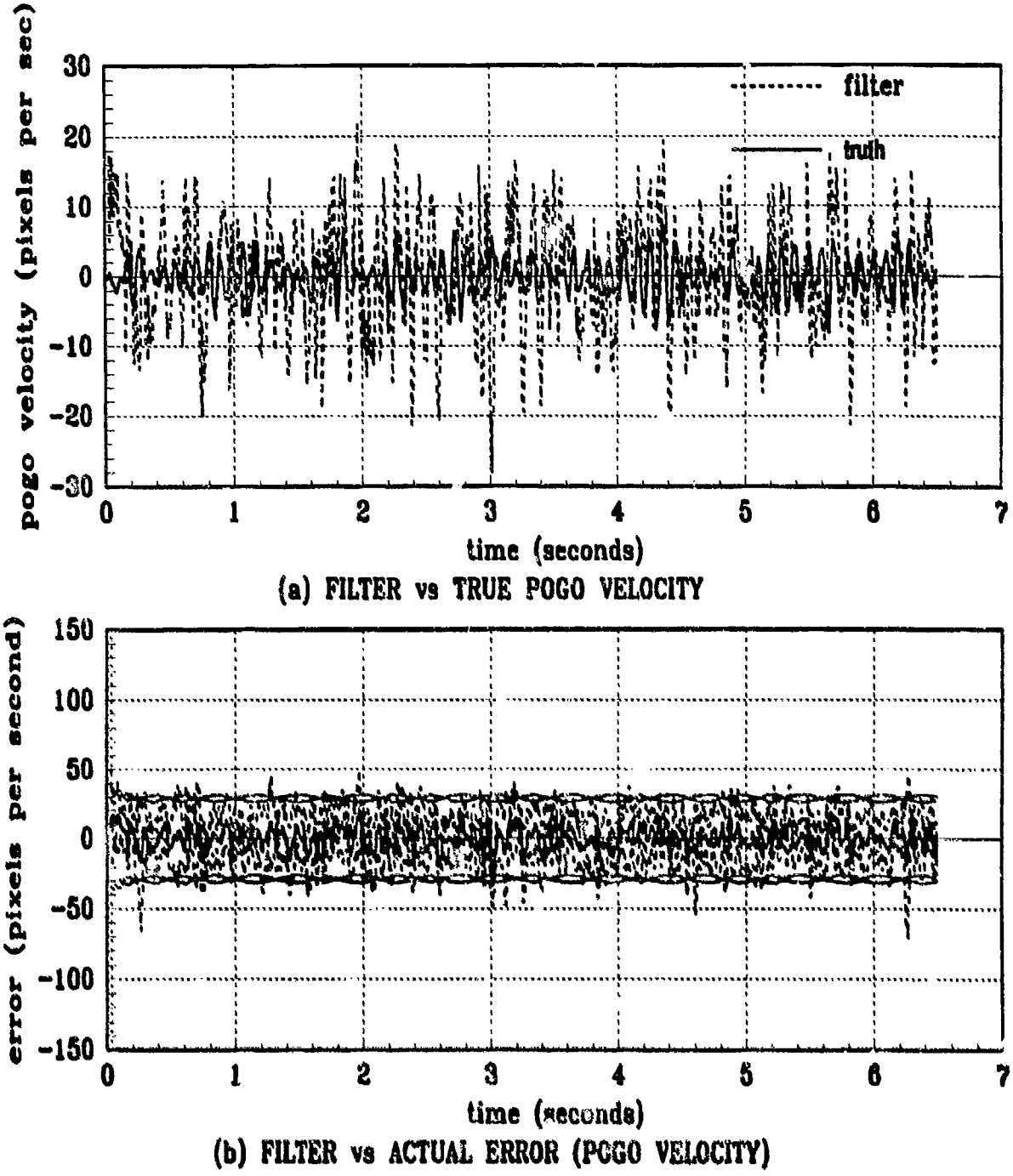


Figure D.14 Pogo Velocity (Filter State 8)
(Truth = $0.112/20\pi$, Filter = $1.12/20\pi$)

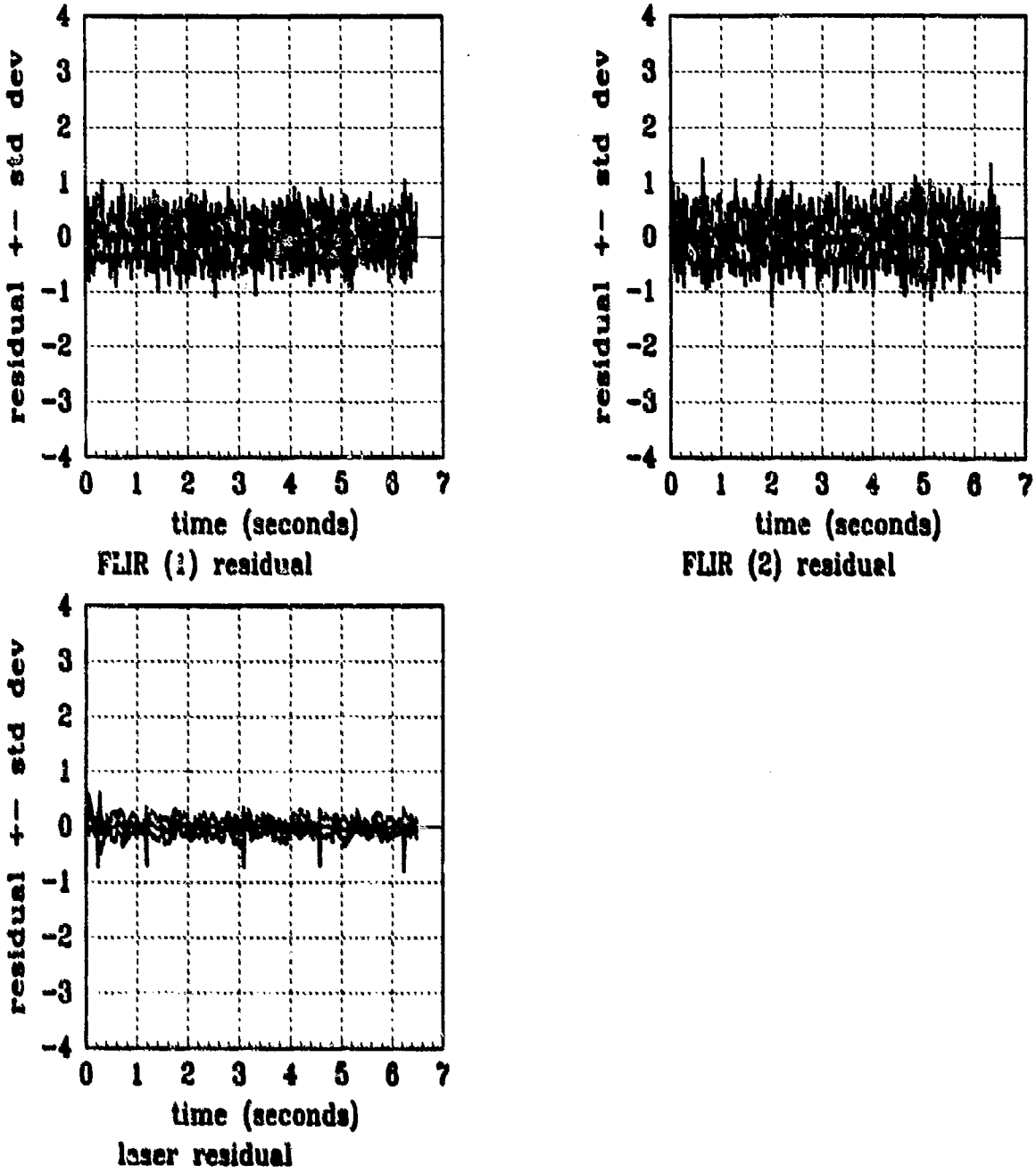
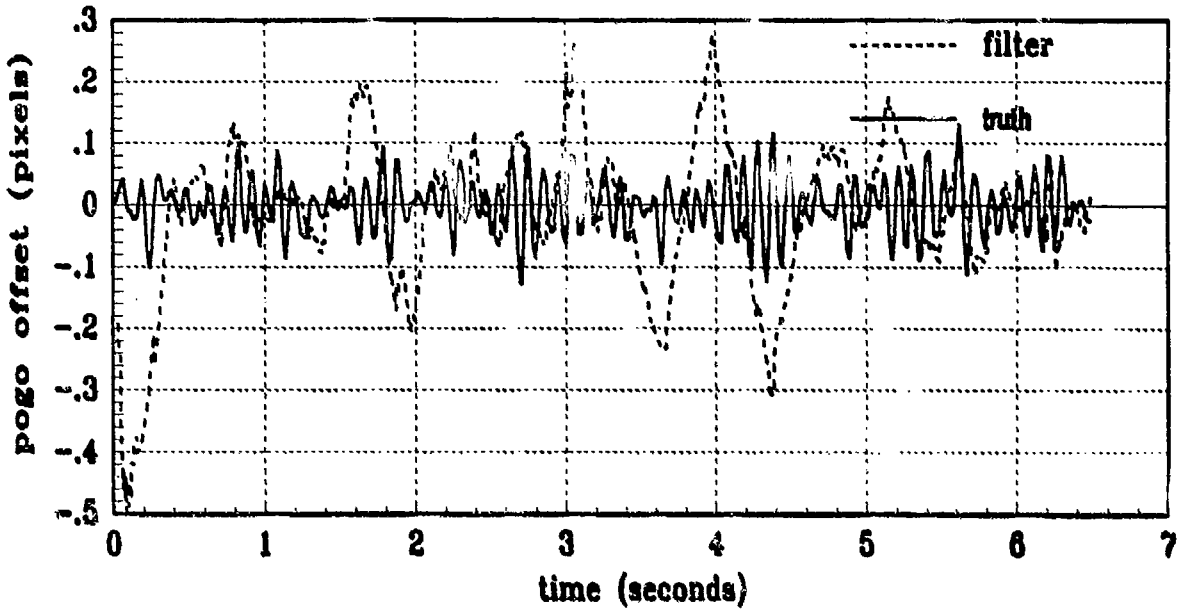
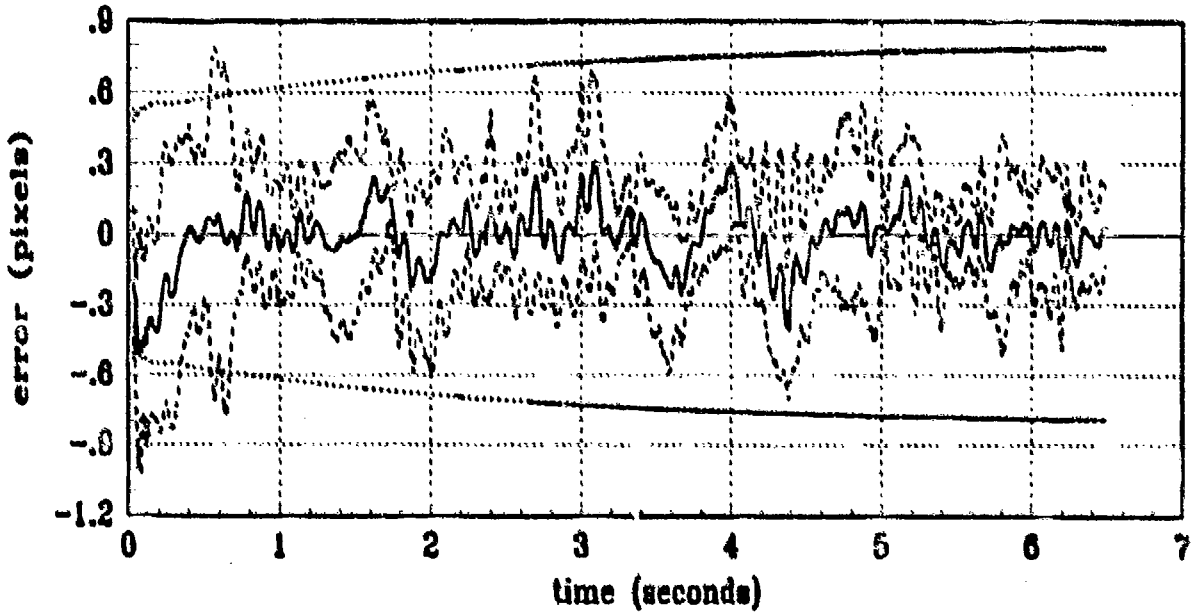


Figure D.15 Filter Residual Quantities
(Truth = $0.112/20\pi$, filter = $1.12/20\pi$)

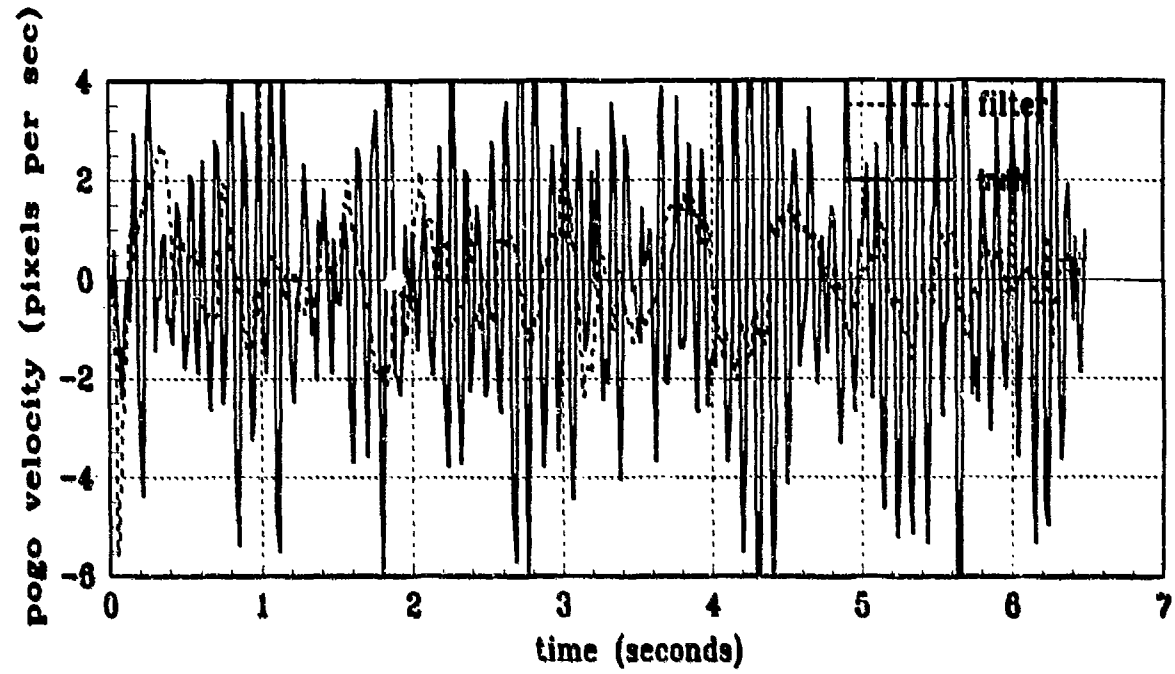


(a) FILTER vs TRUE POGO

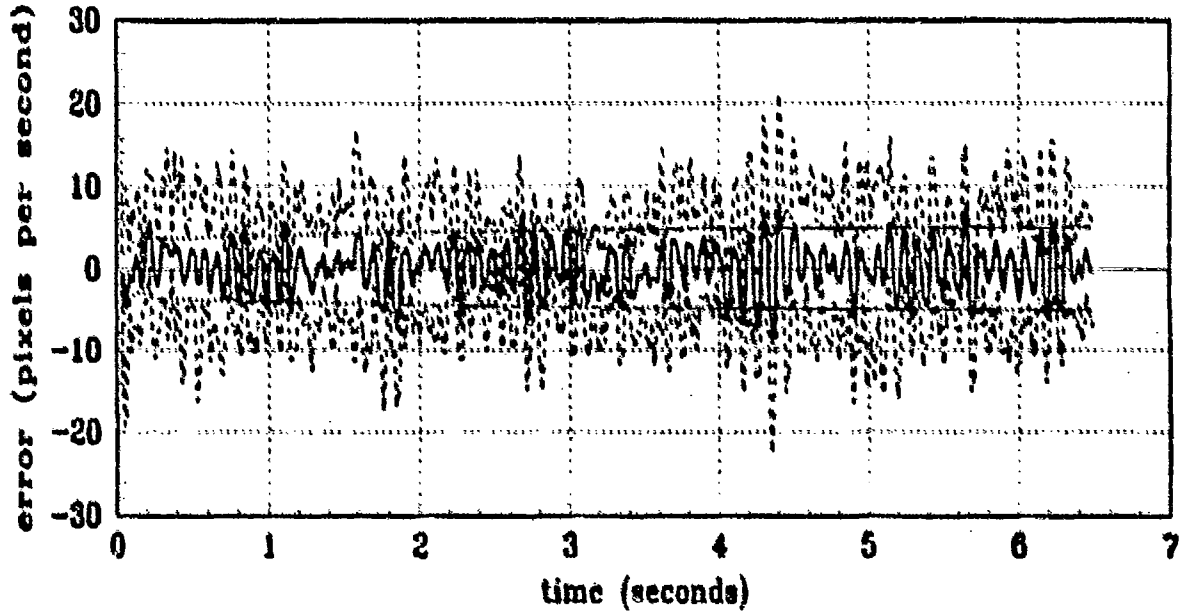


(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure D.16 Pogo Position (Filter State 7)
(Truth = $0.112/20\pi$, Filter = $1.12/2\pi$)



(a) FILTER vs TRUE POGO VELOCITY



(b) FILTER vs ACTUAL ERROR (POGO VELOCITY)

Figure D.17 Pogo Velocity (Filter State 8)
(Truth = $0.112/20\pi$, Filter = $1.12/2\pi$)

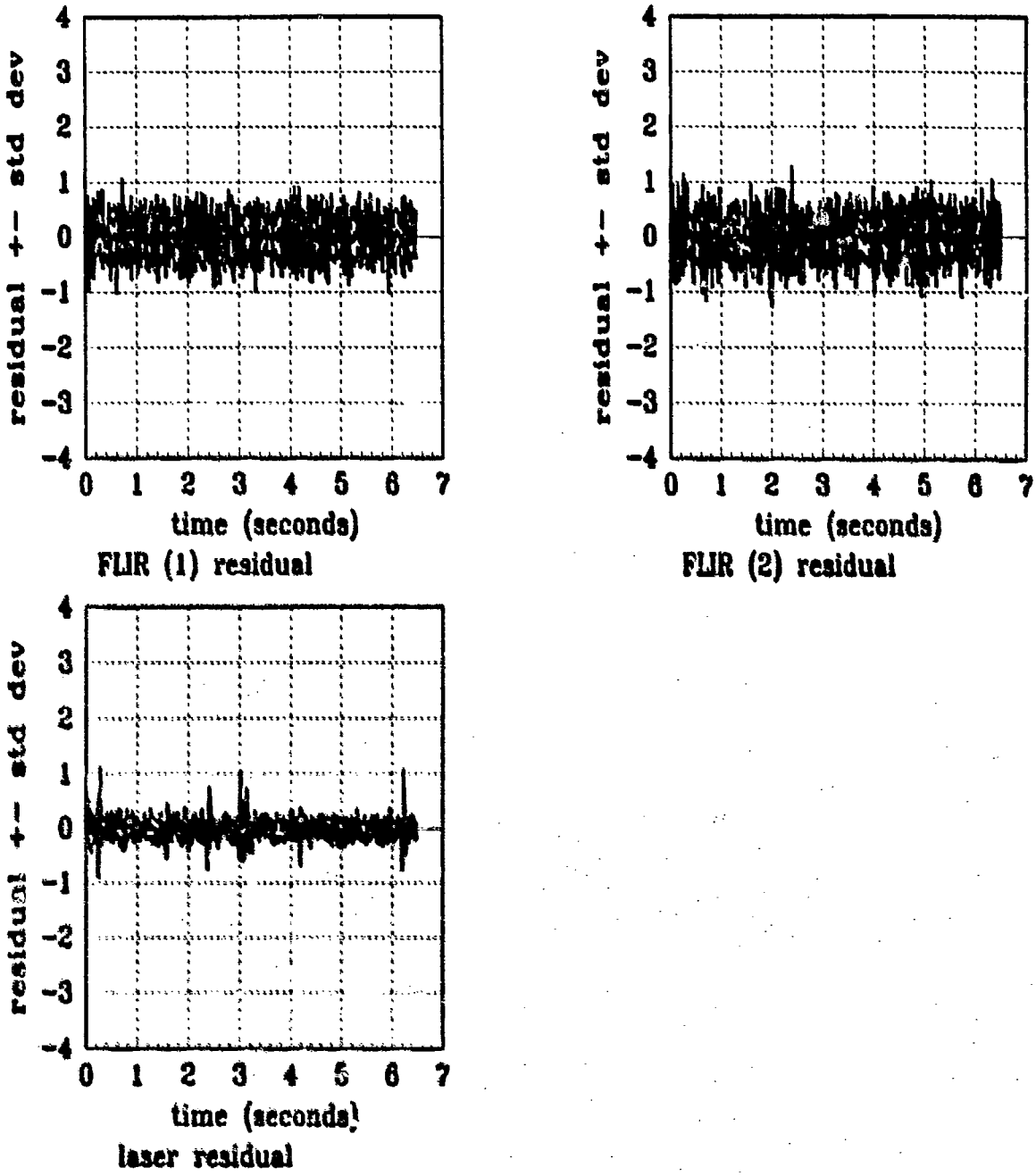


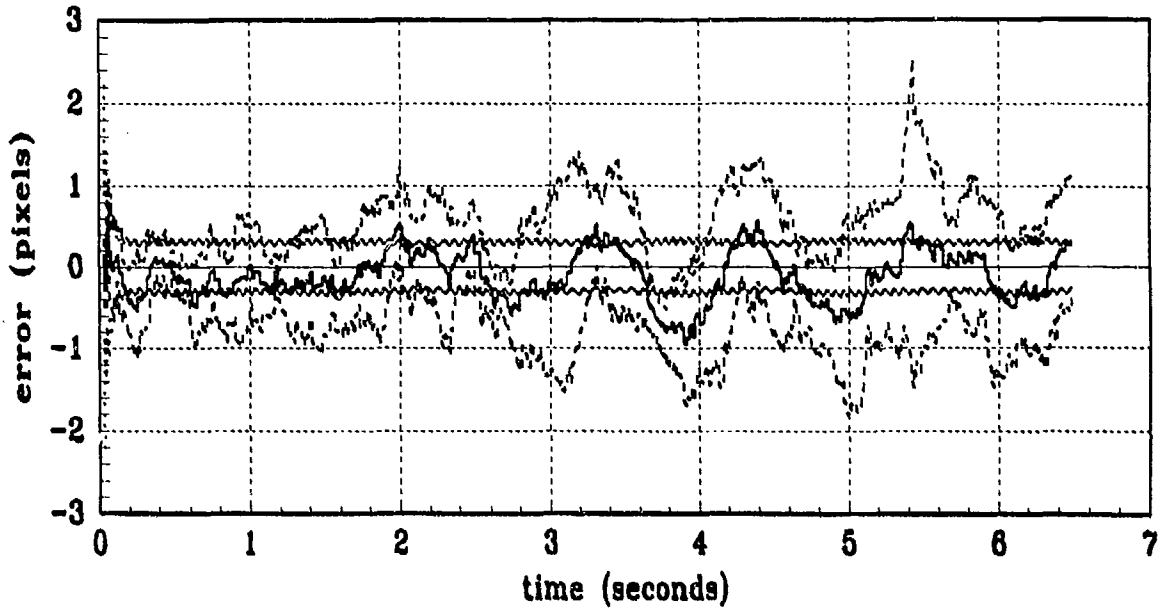
Figure D.18 Filter k lual Quantities
(Truth = 0.112/20π, Filter = 1.12/2π)

Appendix E

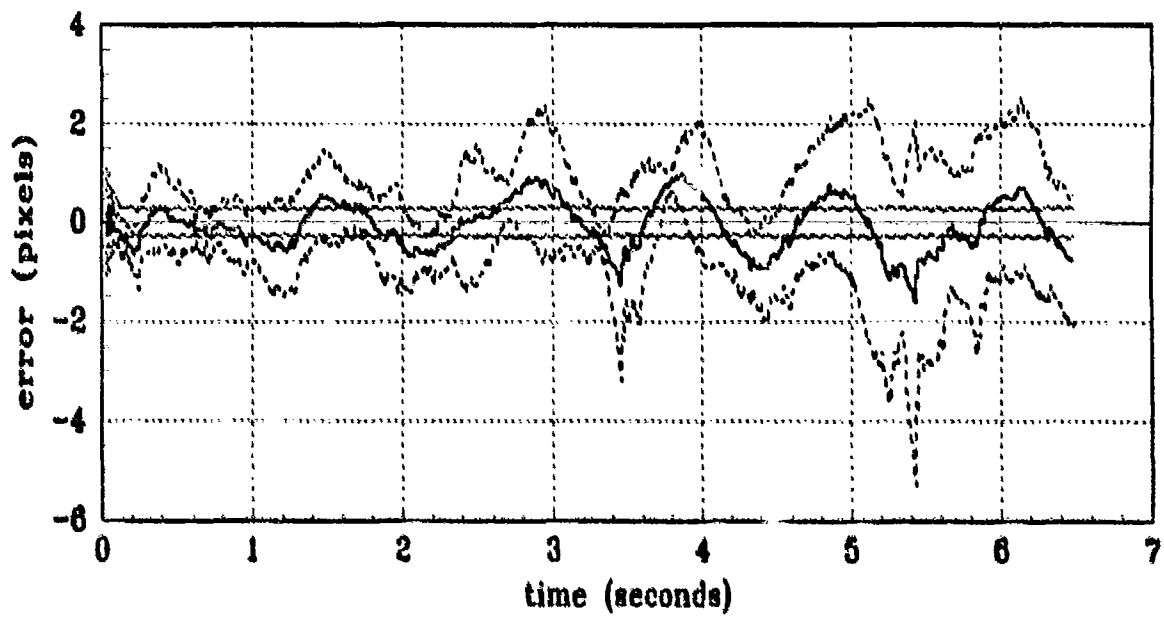
Filter Plots with Truth Parameter Values:

Amplitude = 1.12, Frequency = 2π

This appendix contains the state and error statistics plots of the nine-state elemental filters. The data depicted in the two types of plots in this appendix are explained in *Appendix A*. The state comparison plots show the sample mean truth state over the 5 *Monte Carlo* runs compared to the same statistic for the filter estimate. The error statistics plots represent the error mean $\pm$ standard deviation values in pixels (or pixel/second for velocity and pogo velocity), of the errors between the filter estimate and true state; true mean $\pm$ 1 true standard deviation are plotted, along with zero $\pm$ 1 filter-computed standard deviation.

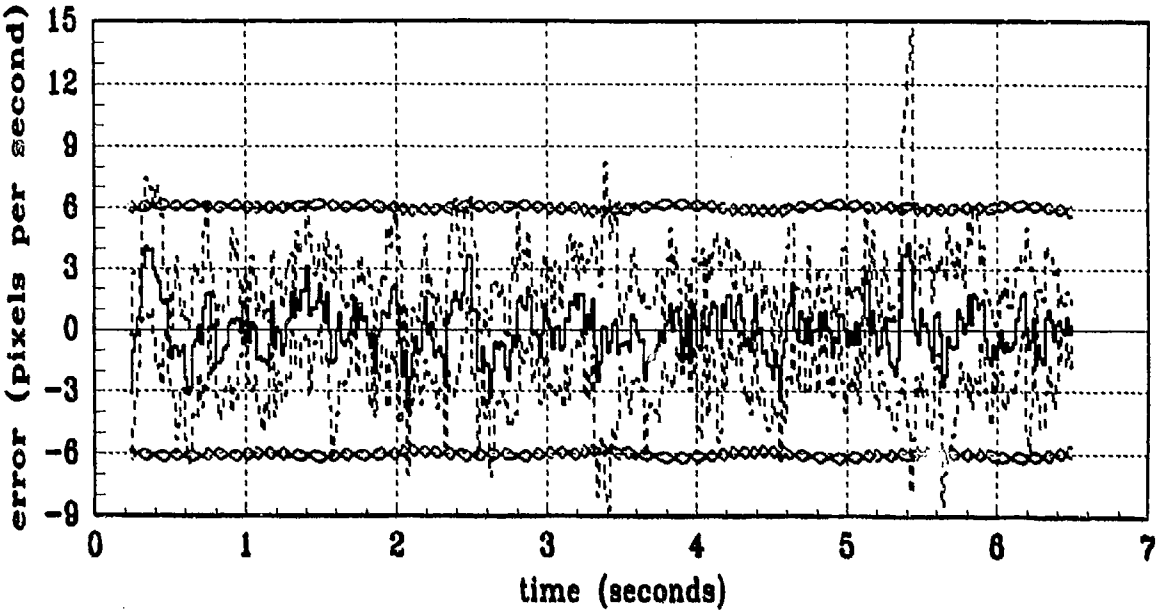


(a) FILTER vs ACTUAL ERROR (X-POSITION)

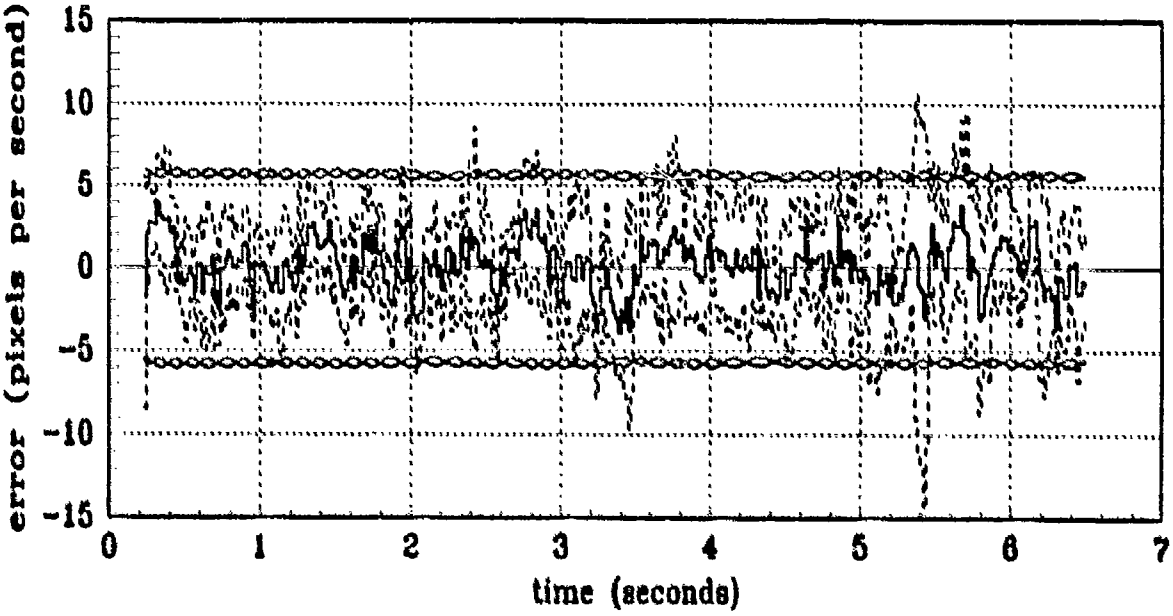


(b) FILTER vs ACTUAL ERROR (Y-POSITION)

Figure E.1 X/Y Position (Filter States 1 and 2) Error Statistics
(Truth = $1.12/2\pi$, Filter = $1.12/2\pi$)



(a) FILTER vs ACTUAL ERROR (X-VELOCITY)



(b) FILTER vs ACTUAL ERROR (Y-VELOCITY)

Figure E.2 X/Y Velocity (Filter States 3 and 4) Error Statistics
(Truth = $1.12/2\pi$, Filter = $1.12/2\pi$)

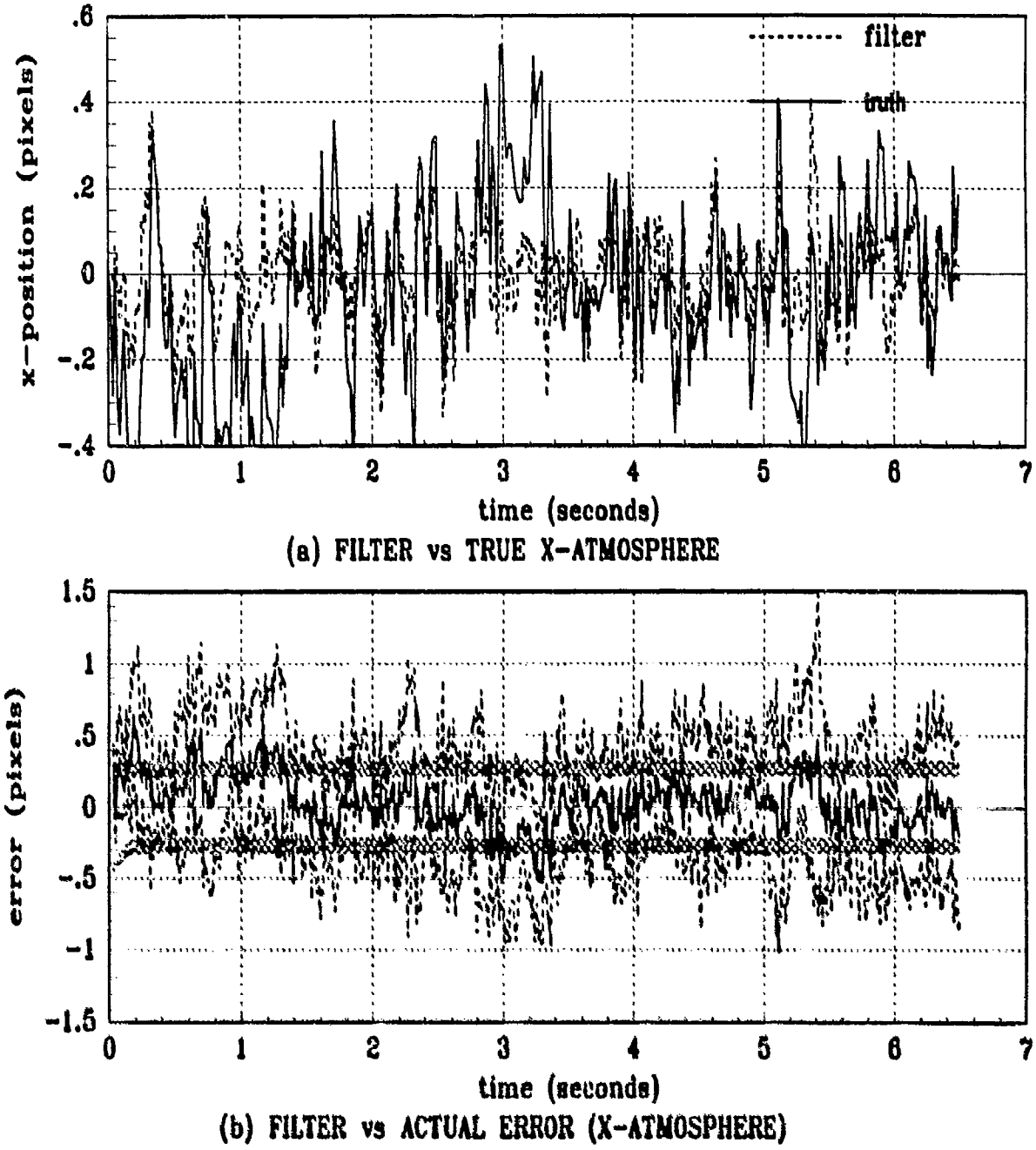


Figure E.3 X Atmospheric Jitter (Filter State 5)
(Truth = $1.12/2\pi$, Filter = $1.12/2\pi$)

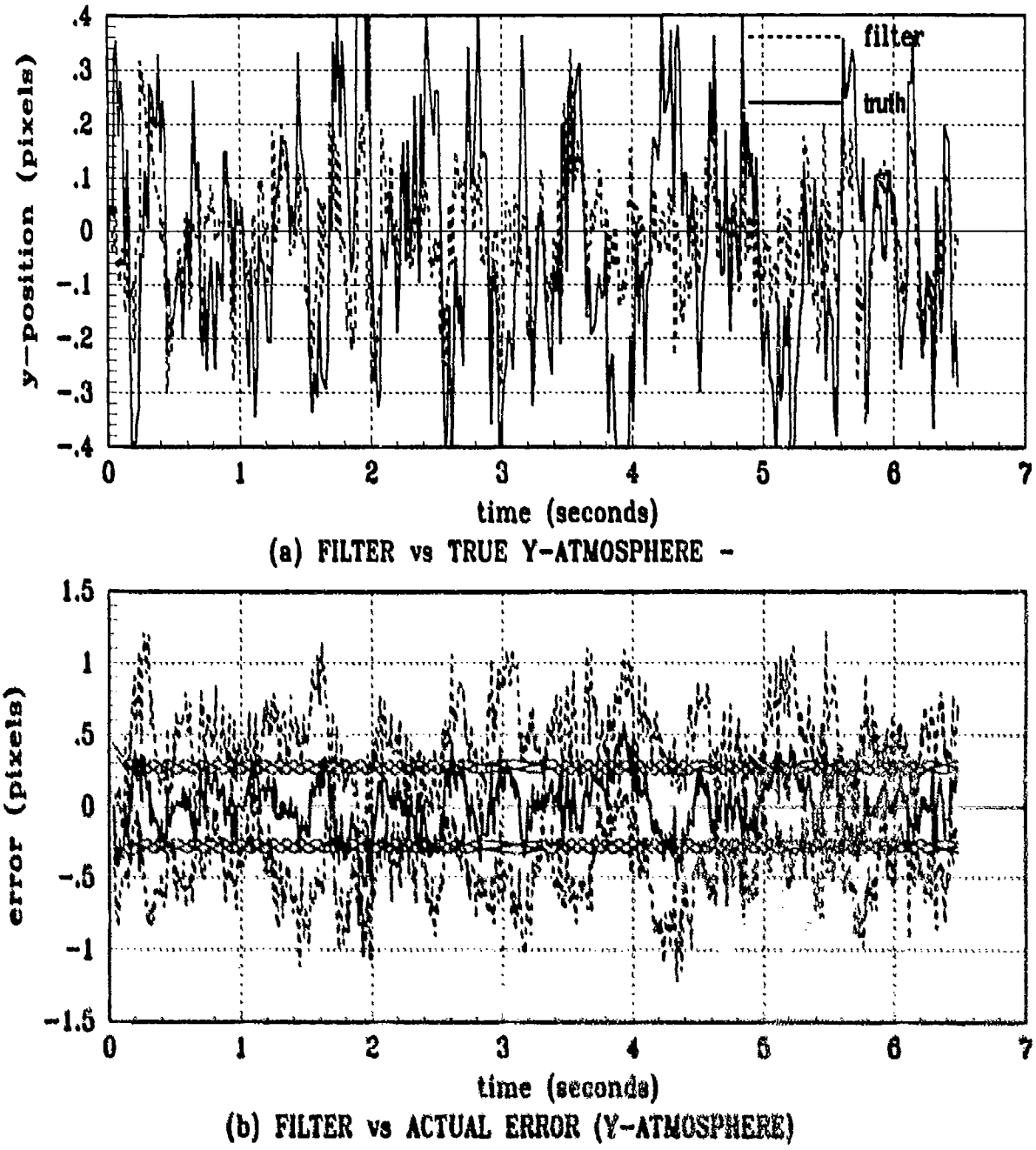
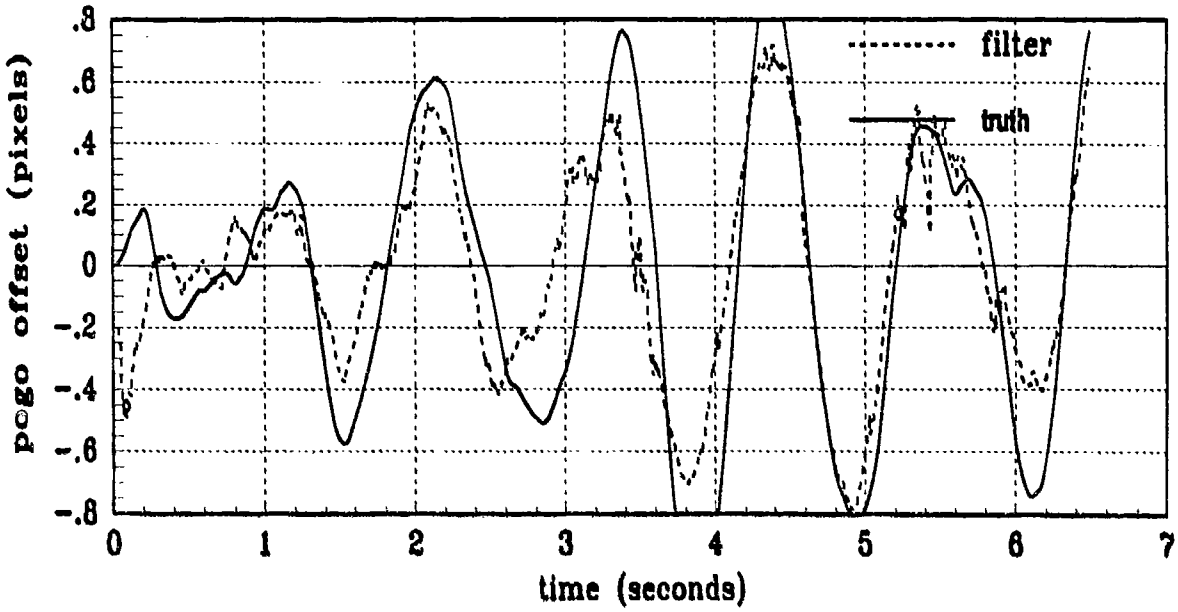
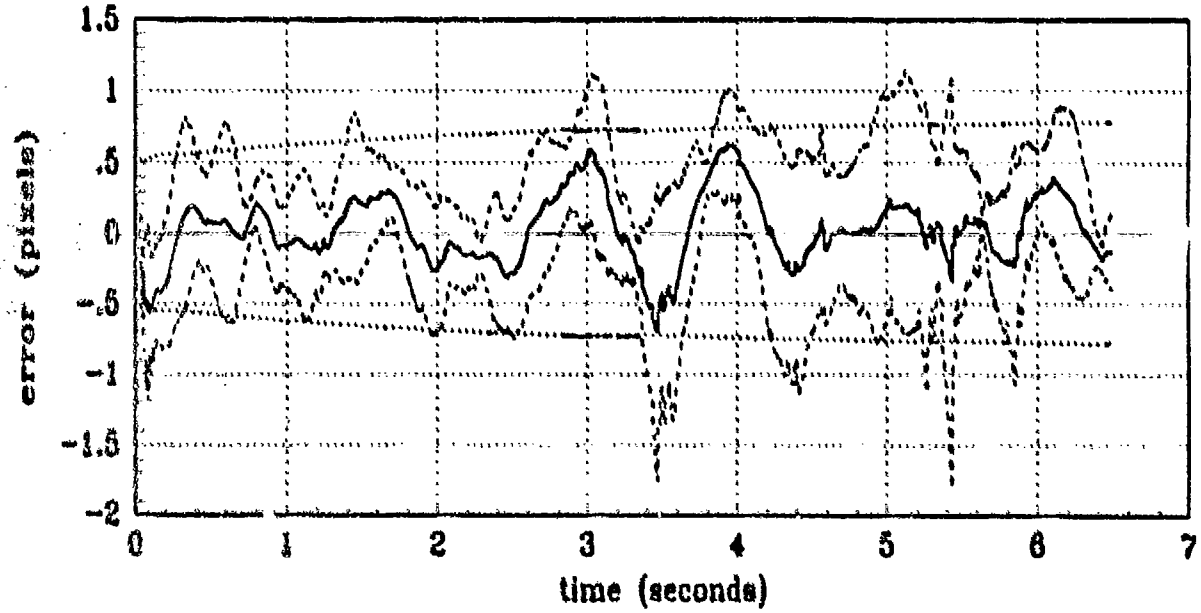


Figure E.4 Y Atmospheric Jitter (Filter State 6)
(Truth = $1.12/2\pi$, Filter = $1.12/2\pi$)



(a) FILTER vs TRUE POGO



(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure E.5 Pogo Position Offset (Filter State 7)
(Truth = $1.12/2\pi$, Filter = $1.12/2\pi$)

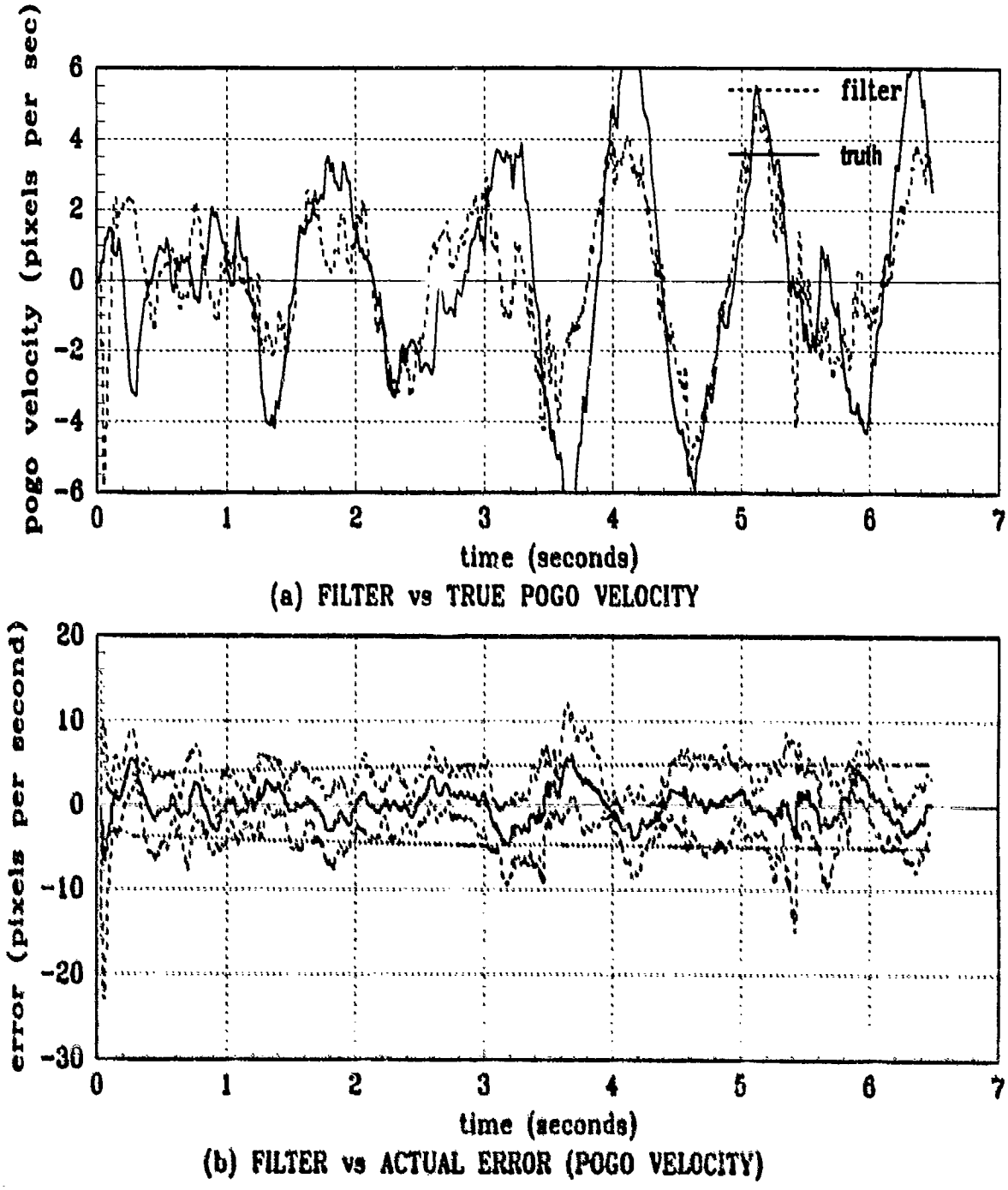
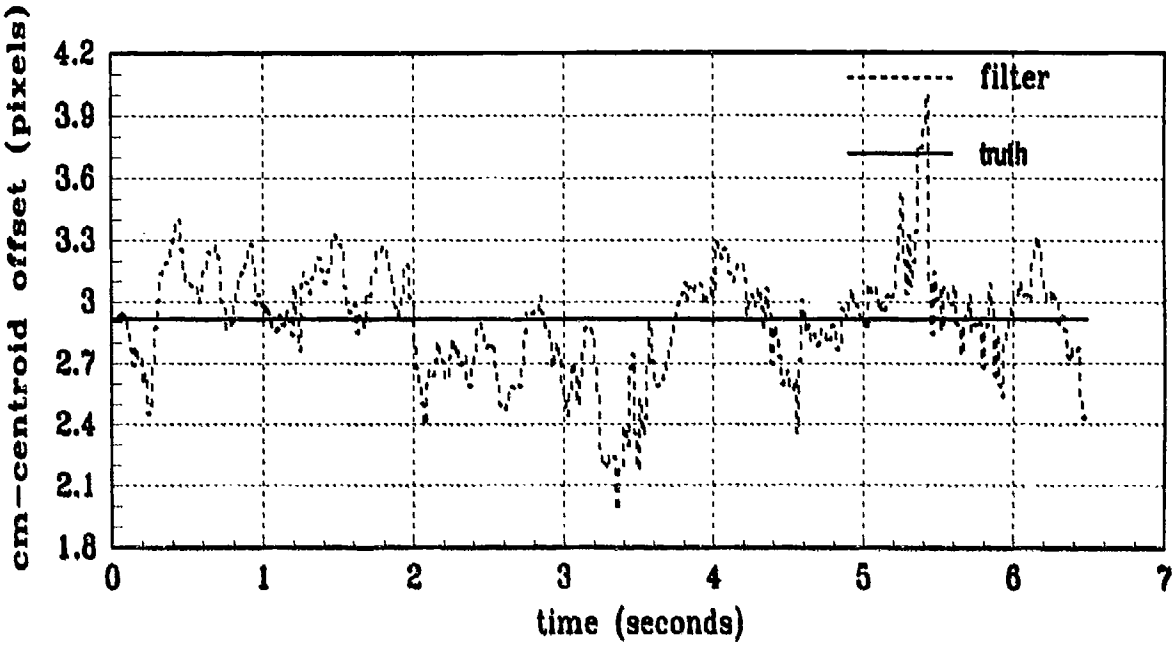
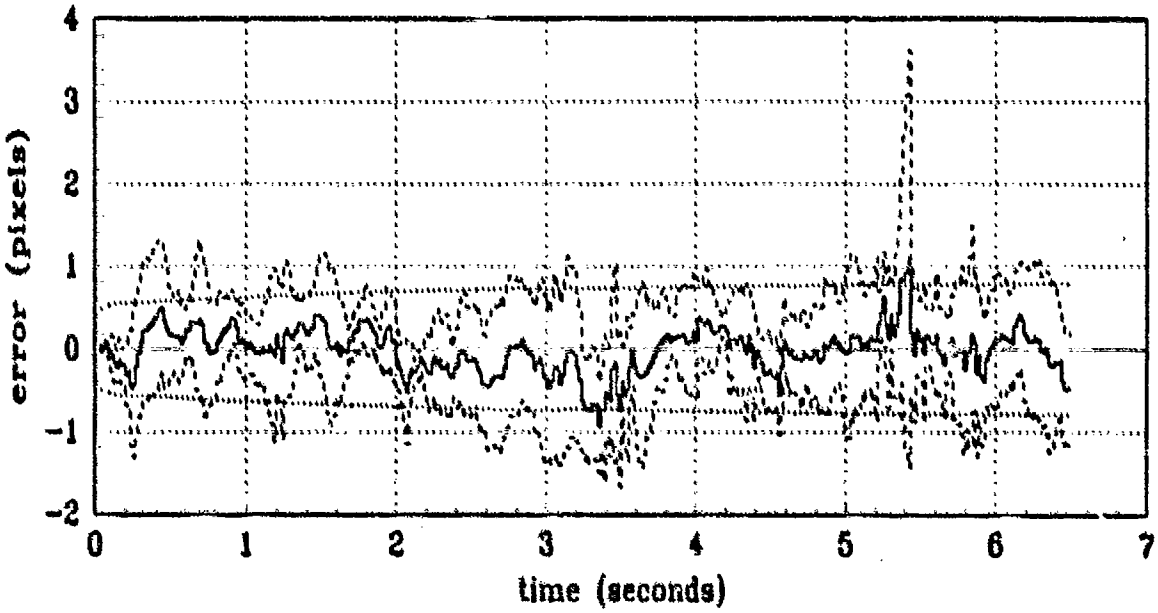


Figure E.6 Pogo Velocity (Filter State 8)
(Truth = $1.12/2\pi$, Filter = $1.12/2\pi$)

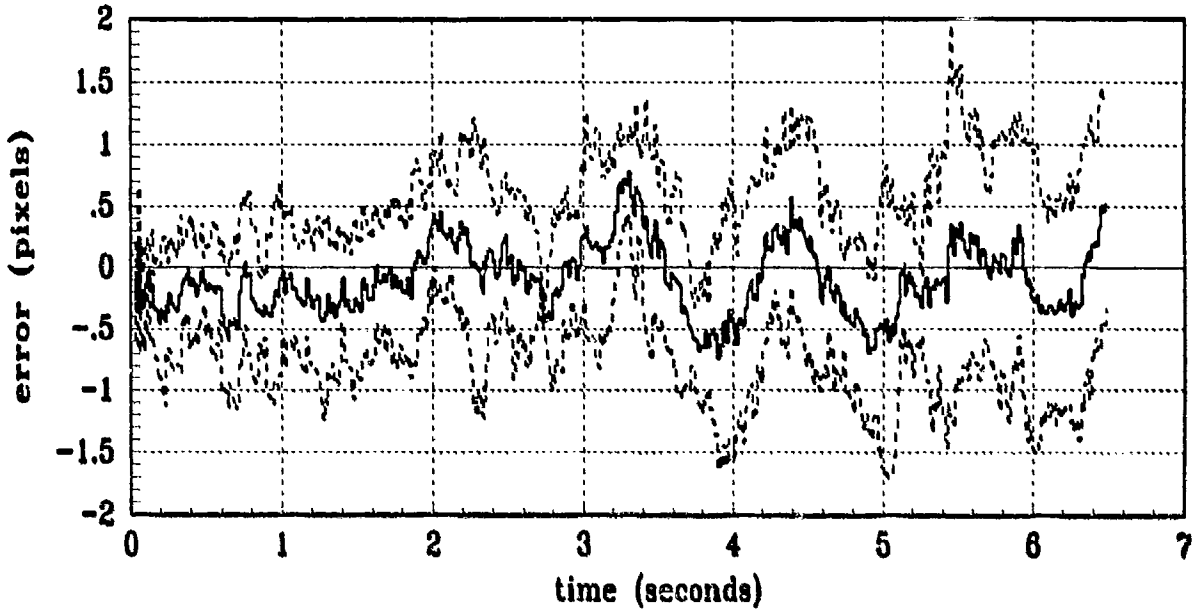


(a) FILTER vs TRUE COM-EQUILIBRIUM POINT OFFSET

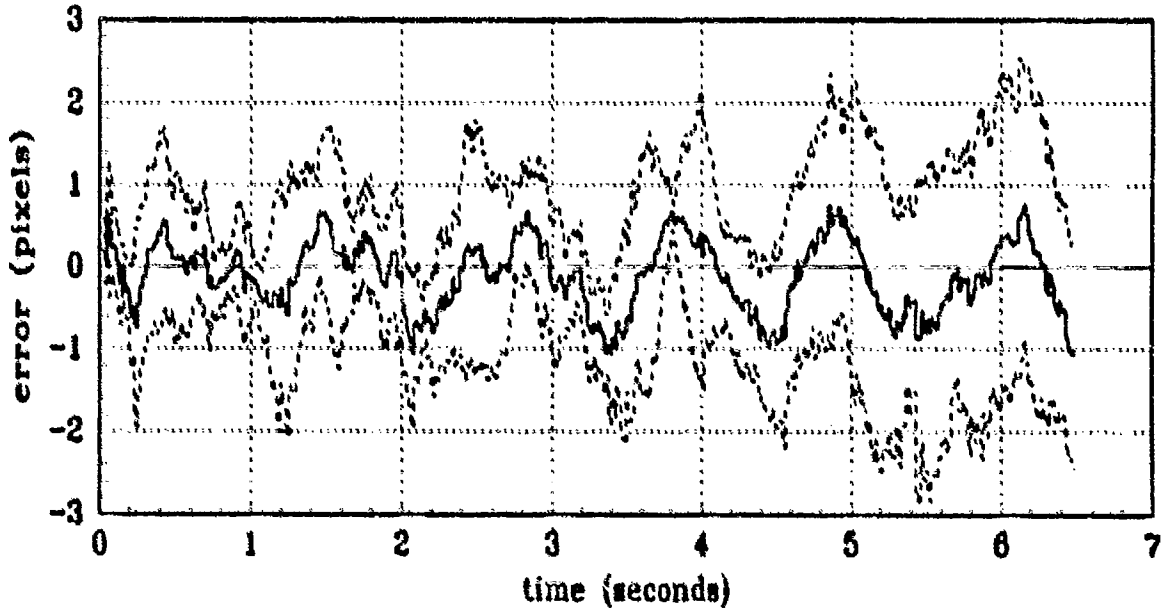


(b) FILTER vs ACTUAL ERROR (COM-EQUILIBRIUM POINT OFFSET)

Figure E.7 Center-of-Mass to Equilibrium Offset (Filter State 9)
(Truth = $1.12/2\pi$, Filter = $1.12/2\pi$)



(a) FILTER vs ACTUAL ERROR (X-CENTROID POSITION)



(b) FILTER vs ACTUAL ERROR (Y-CENTROID POSITION)

Figure E.8 Plume Centroid Error Statistics
(Truth = $1.12/2\pi$, Filter = $1.12/2\pi$)

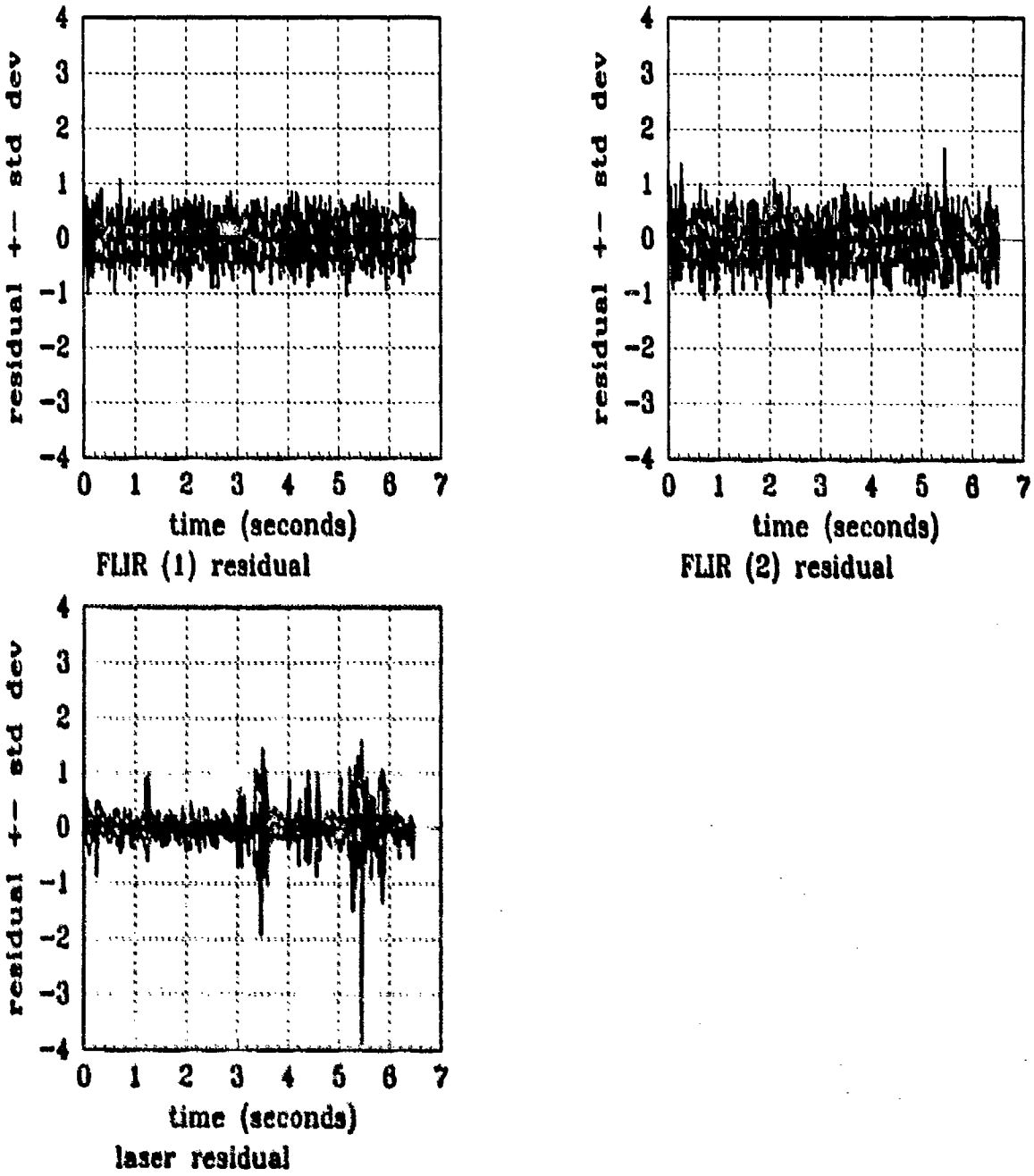
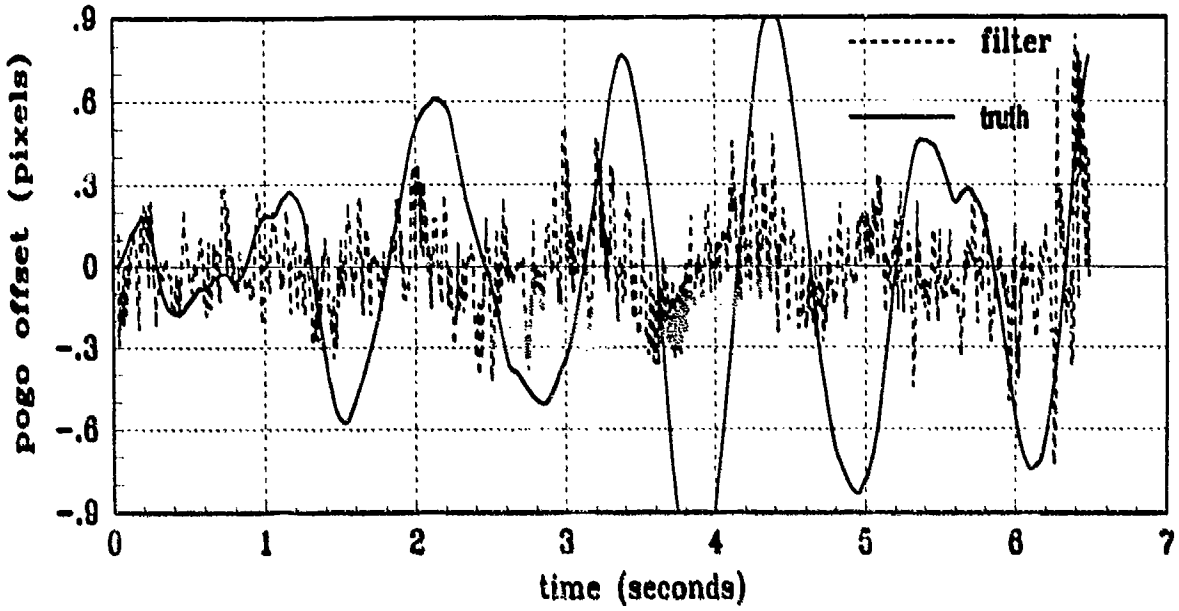
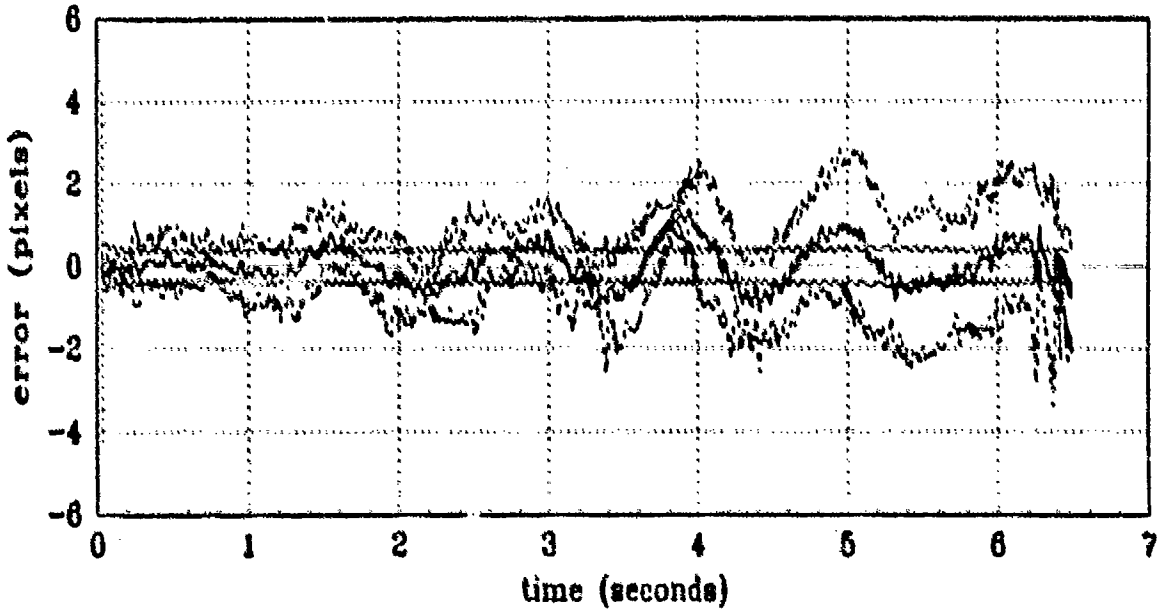


Figure E.9 Filter Residual Quantities
(Truth = $1.12/2\pi$, Filter = $1.12/2\pi$)

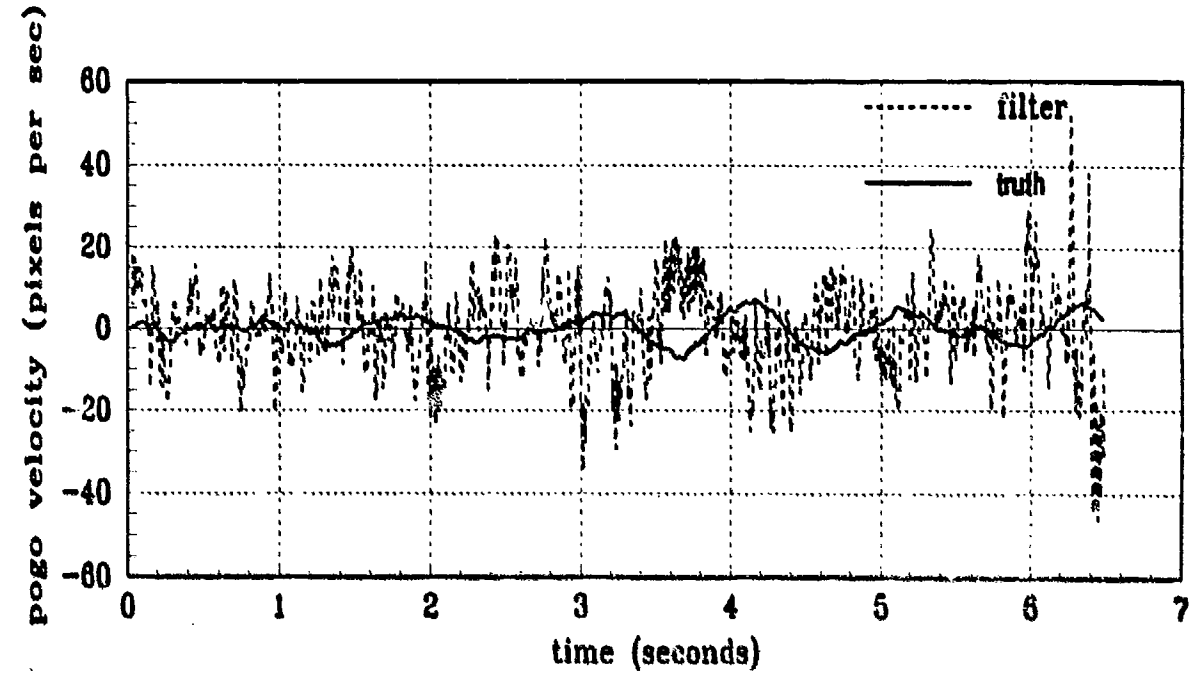


(a) FILTER vs TRUE POGO

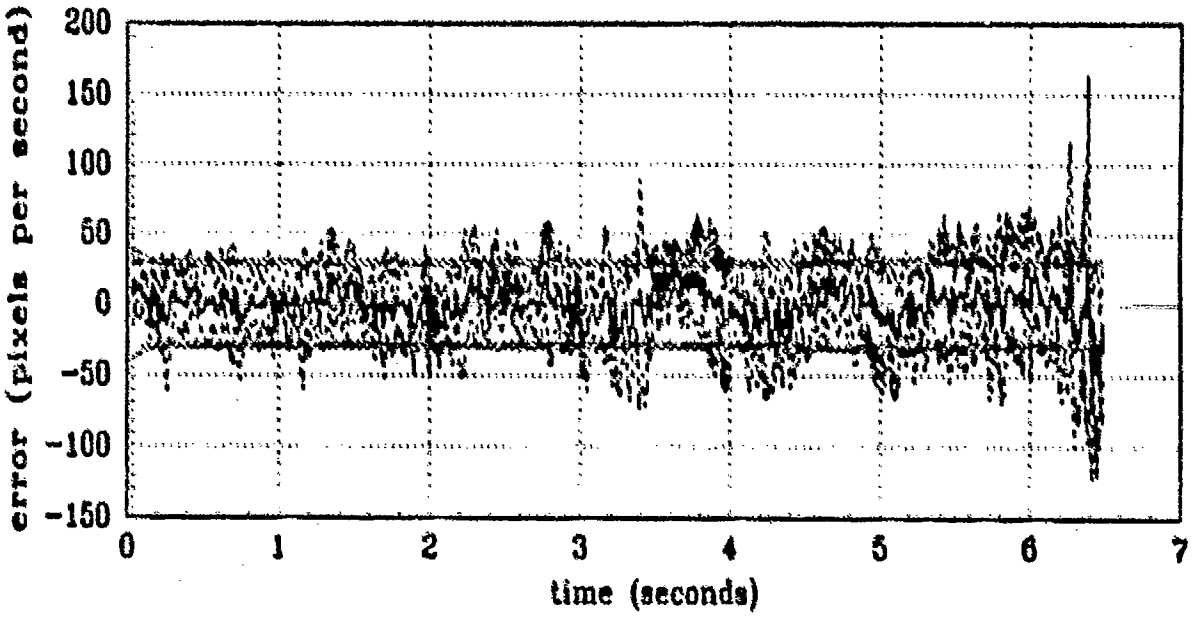


(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure E.10 Pogo Position (Filter State 7)
(Truth = $1.12/2\pi$, Filter = $1.12/20\pi$)



(a) FILTER vs TRUE POGO VELOCITY



(b) FILTER vs ACTUAL ERROR (POGO VELOCITY)

Figure E.11 Pogo Velocity (Filter State 8)
(Truth = $1.12/2\pi$, Filter = $1.12/20\pi$)

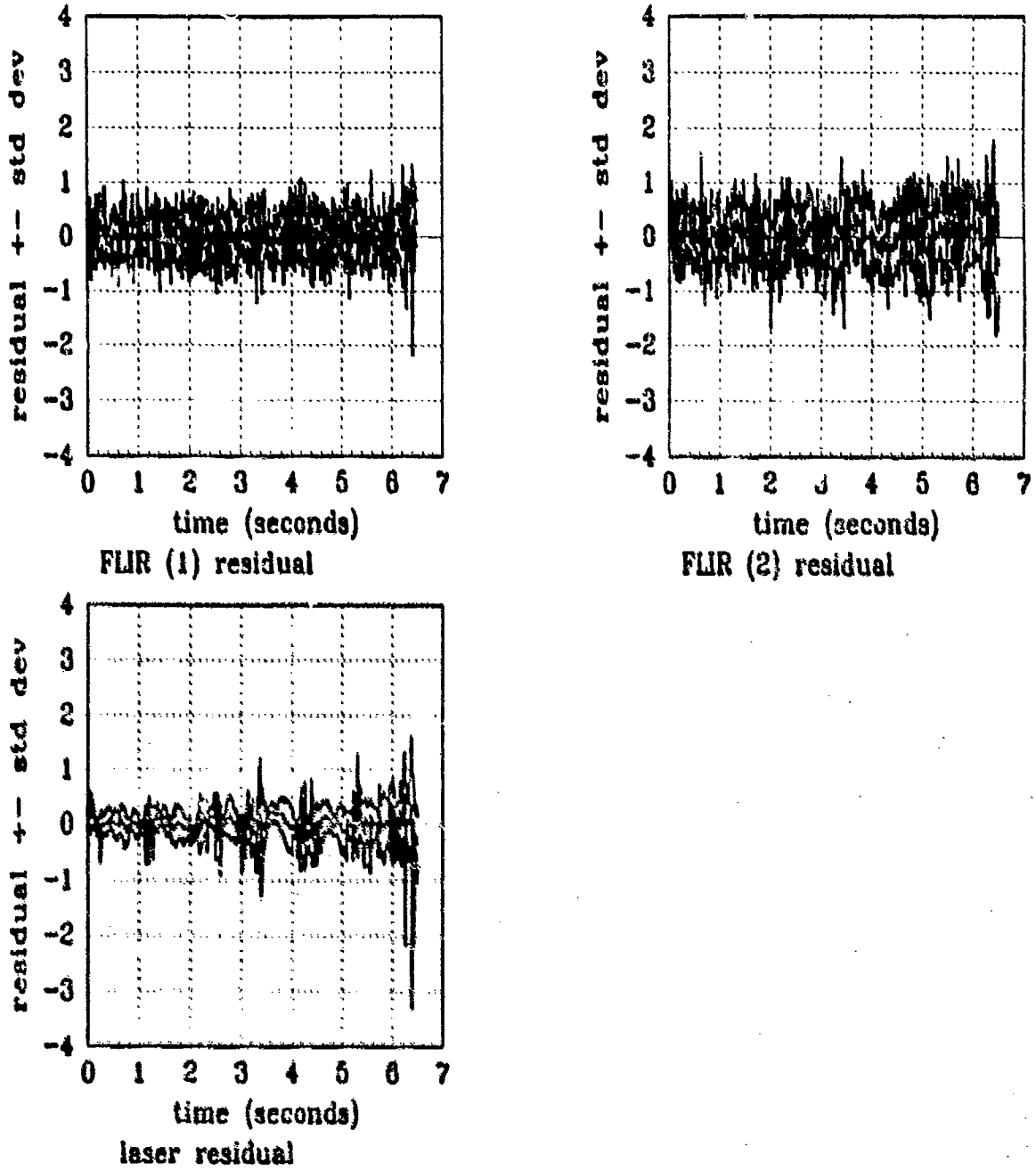
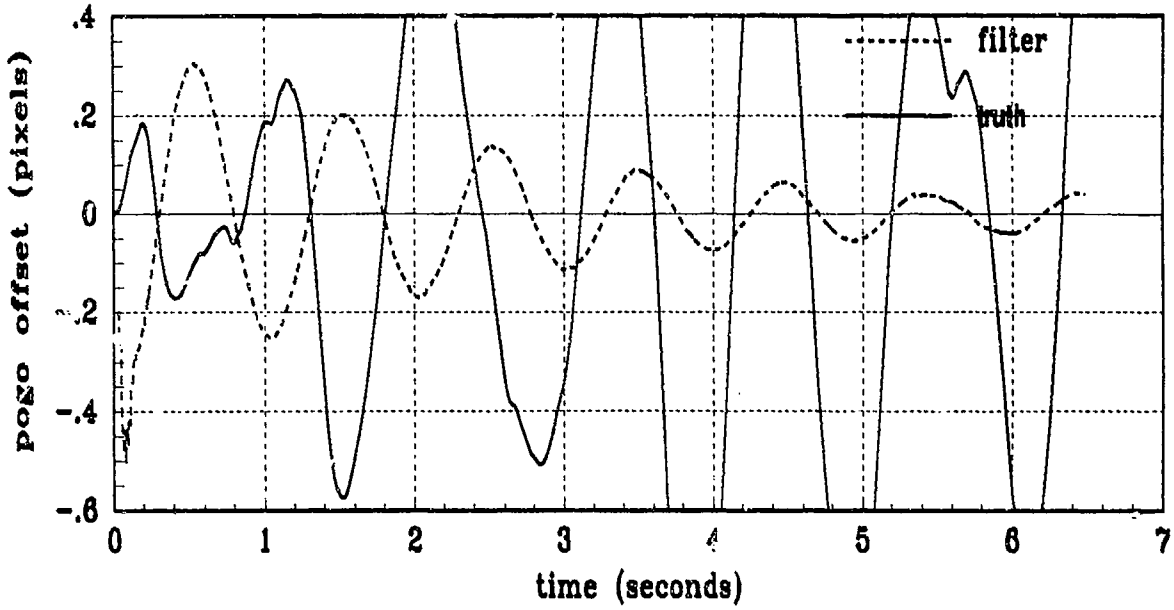
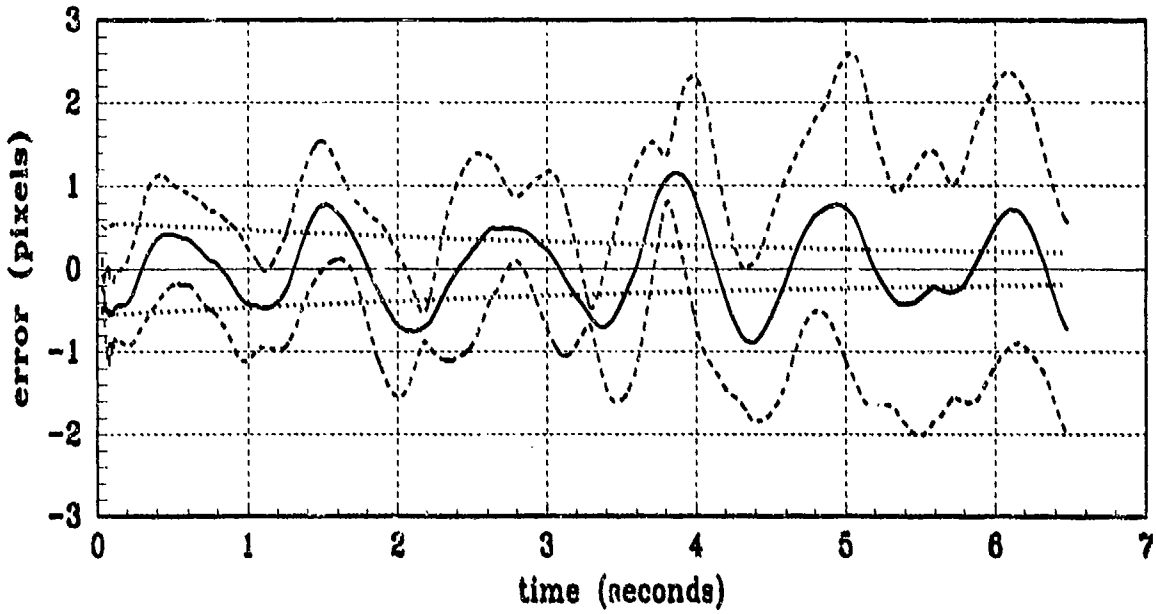


Figure E.12 Filter Residual Quantities
(Truth = 1.12/2π, Filter = 1.12/20π)

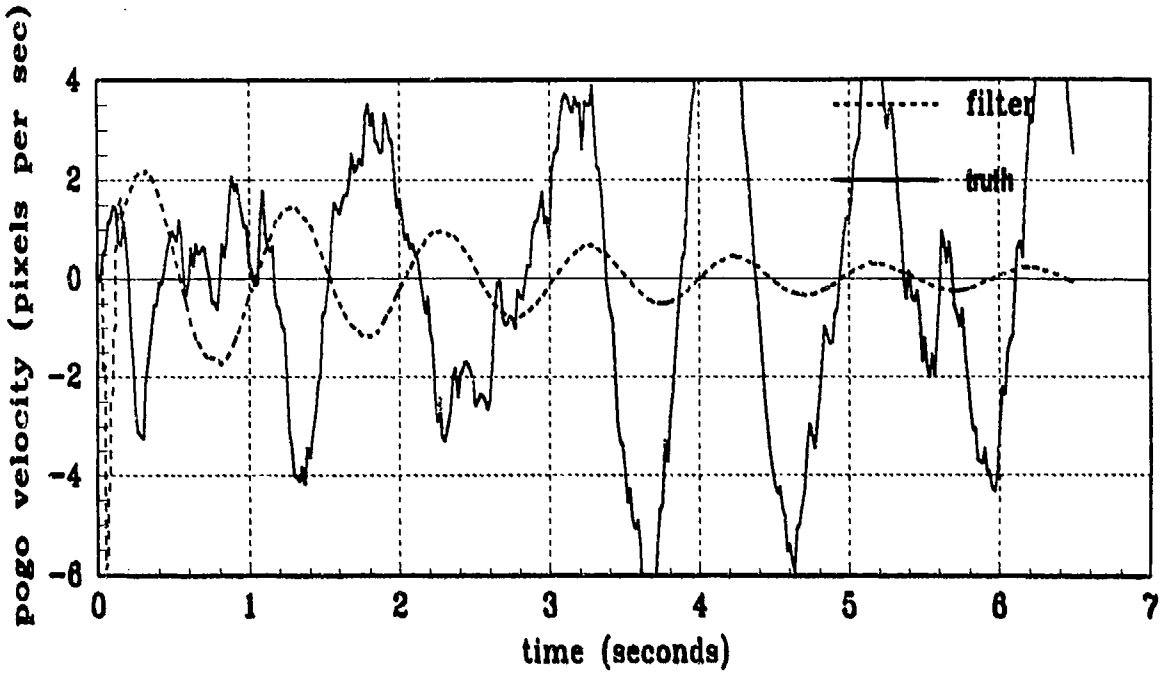


(a) FILTER vs TRUE POGO

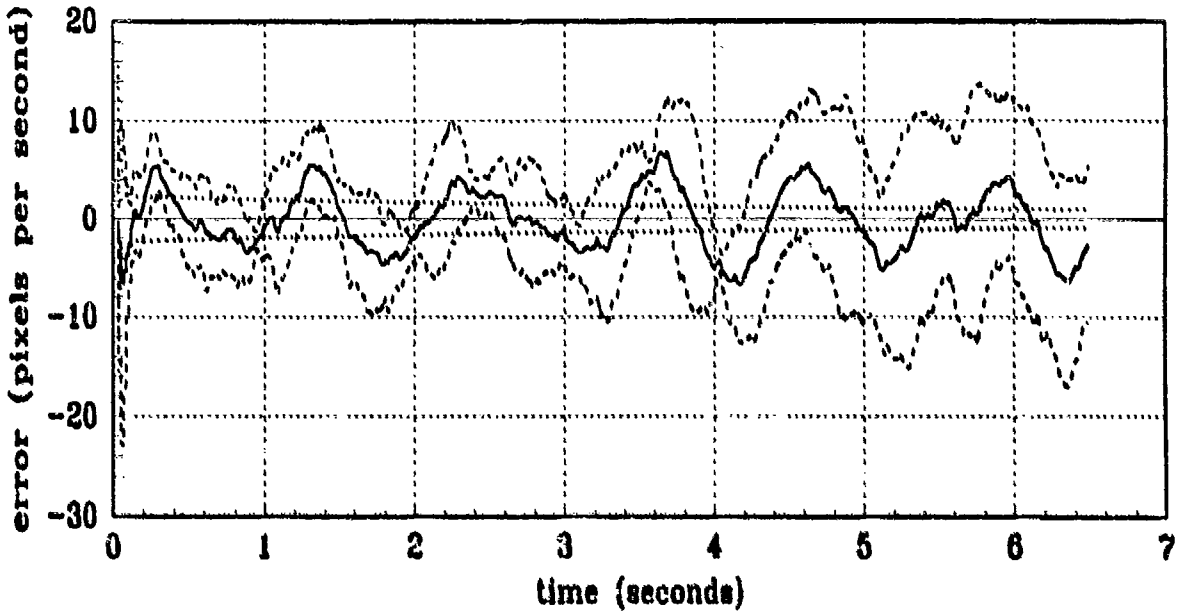


(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure E.13 Pogo Position (Filter State 7)
(Truth = $1.12/2\pi$, Filter = $0.112/2\pi$)



(a) FILTER vs TRUE POGO VELOCITY



(b) FILTER vs ACTUAL ERROR (POGO VELOCITY)

Figure E.14 Pogo Velocity (Filter State 8)
(Truth = $1.12/2\pi$, Filter = $0.112/2\pi$)

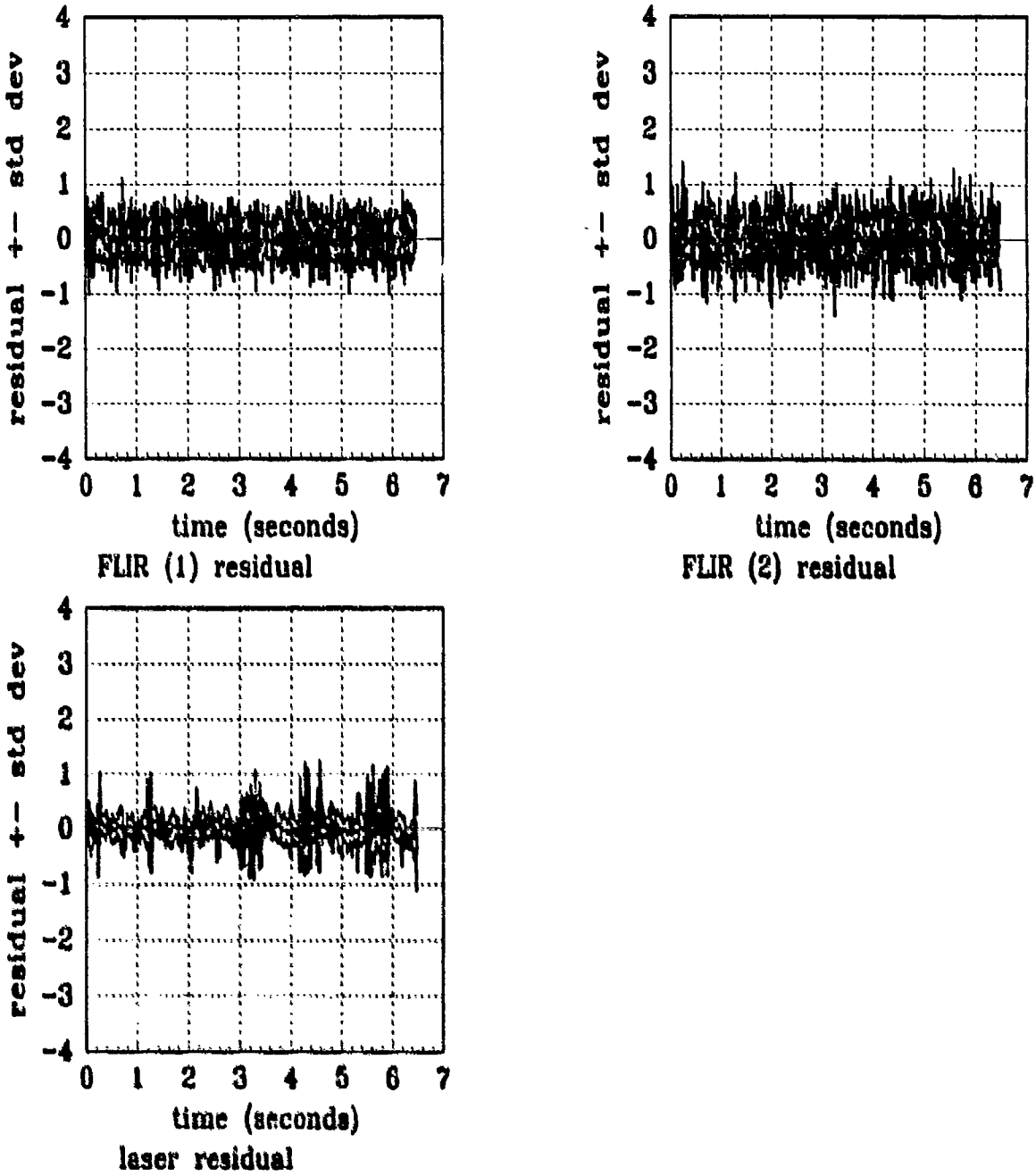
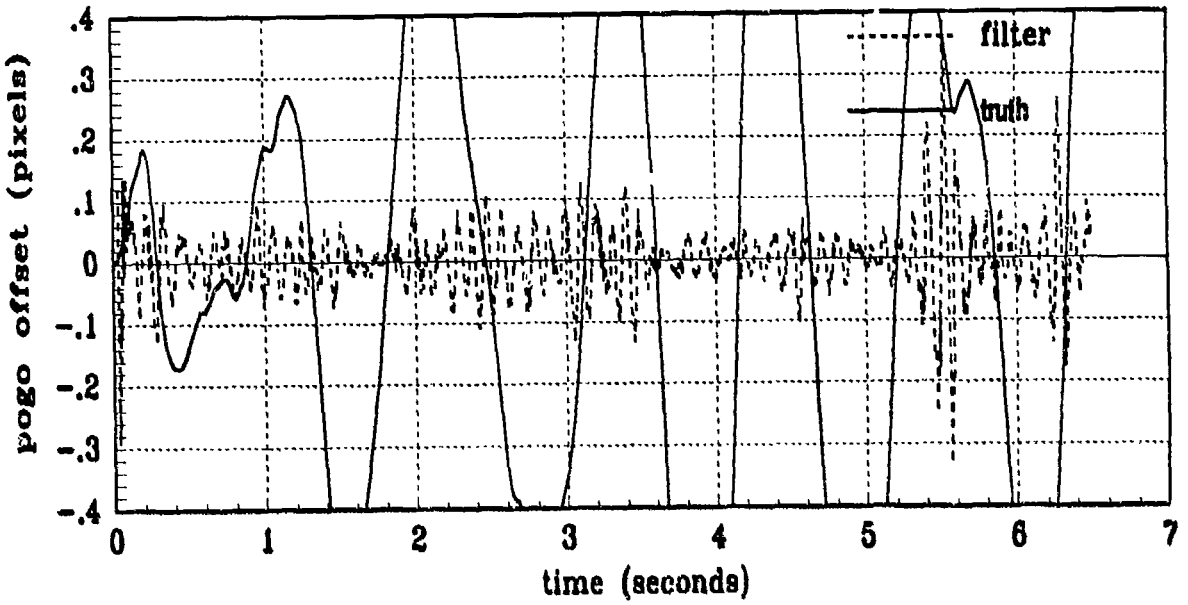
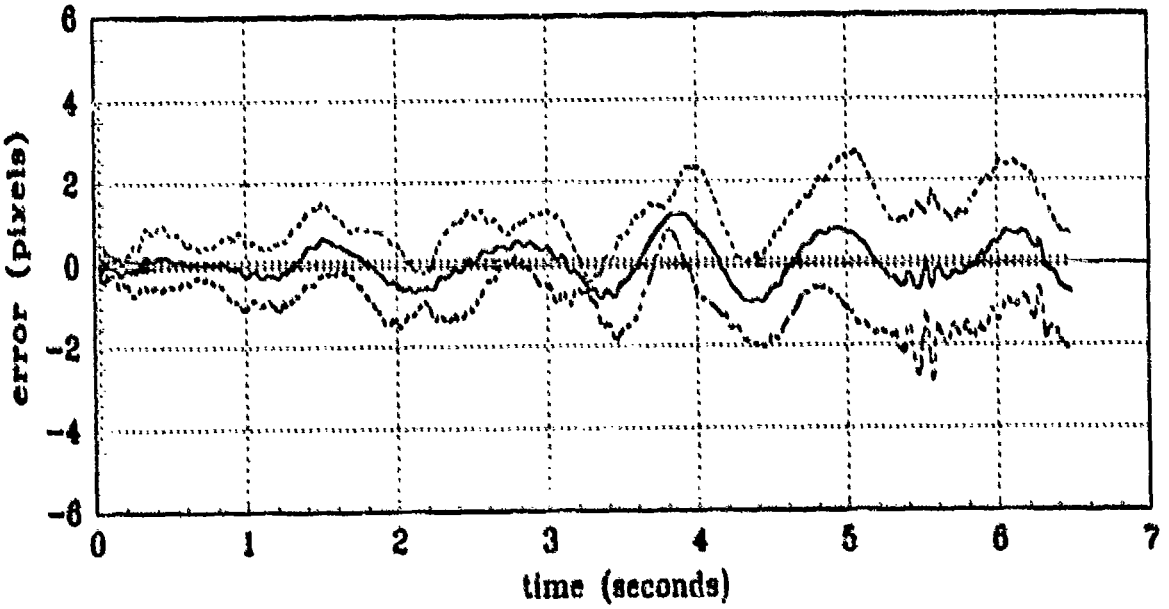


Figure E.15 Filter Residual Quantities
(Truth = $1.12/2\pi$, Filter = $0.112/2\pi$)



(a) FILTER vs TRUE POGO



(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure E.16 Pogo Position (Filter State 7)
(Truth = $1.12/2\pi$, Filter = $0.112/20\pi$)

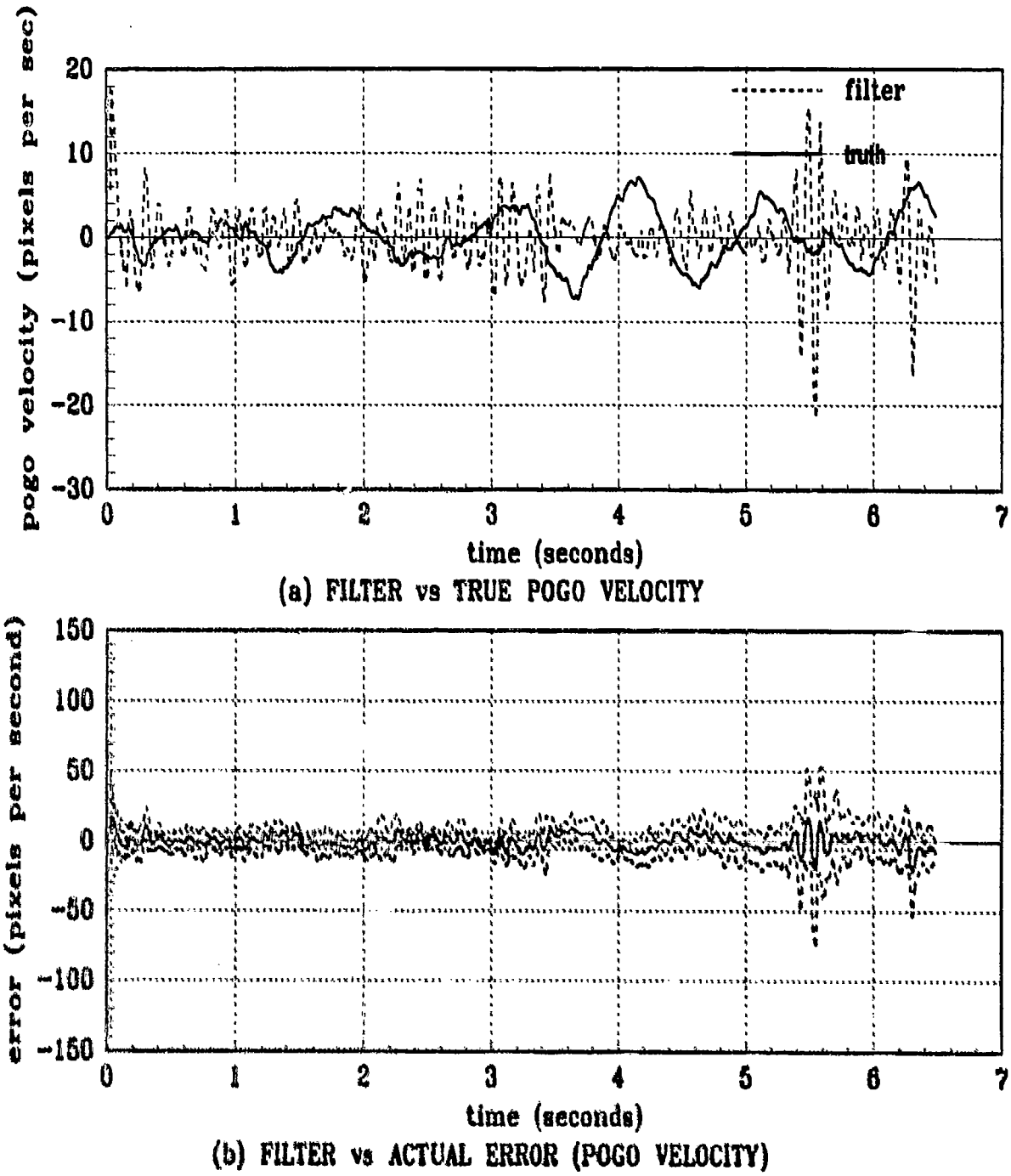


Figure E.17 Pogo Velocity (Filter State 8)
(Truth = $1.12/2\pi$, Filter = $0.112/20\pi$)

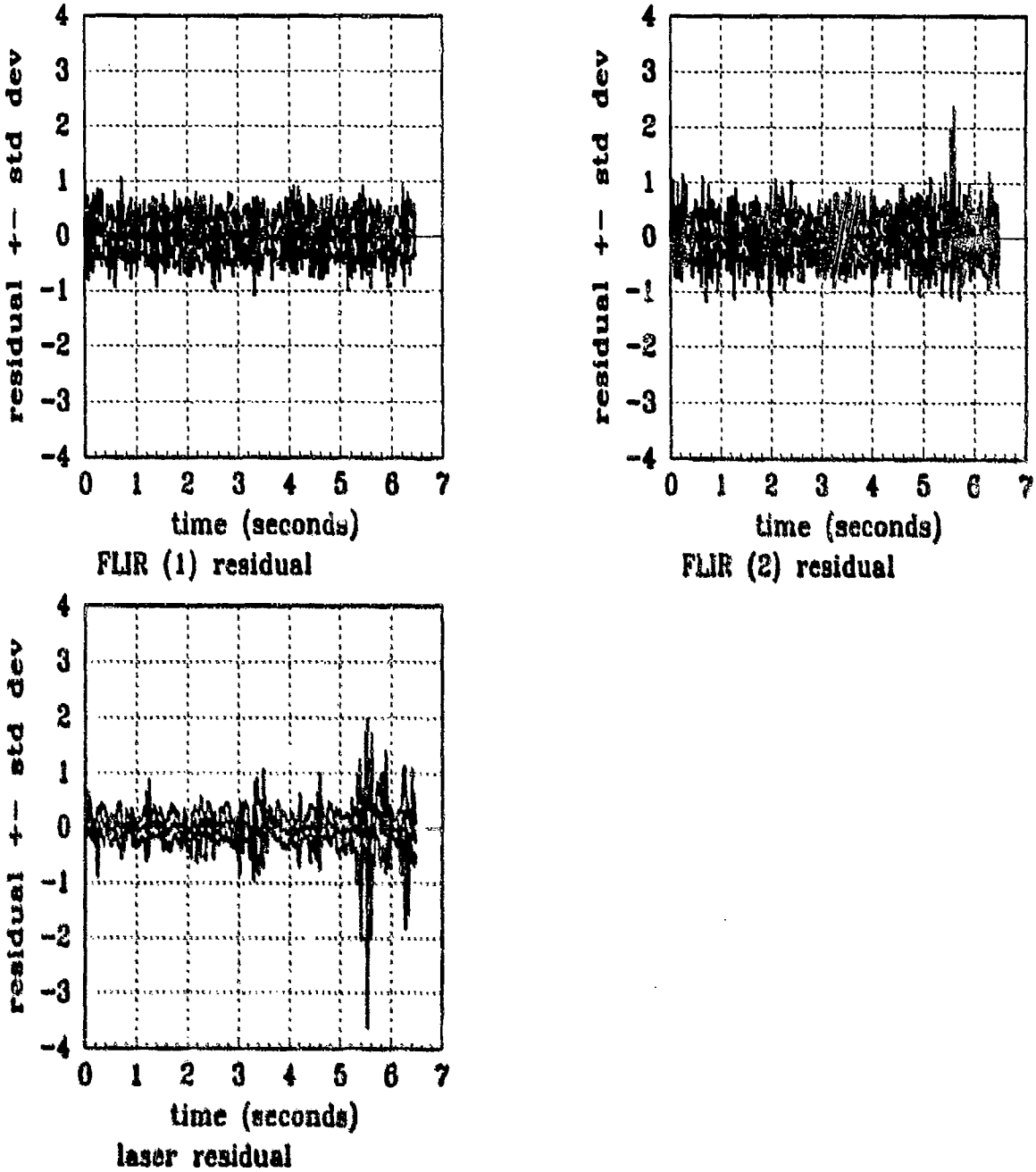


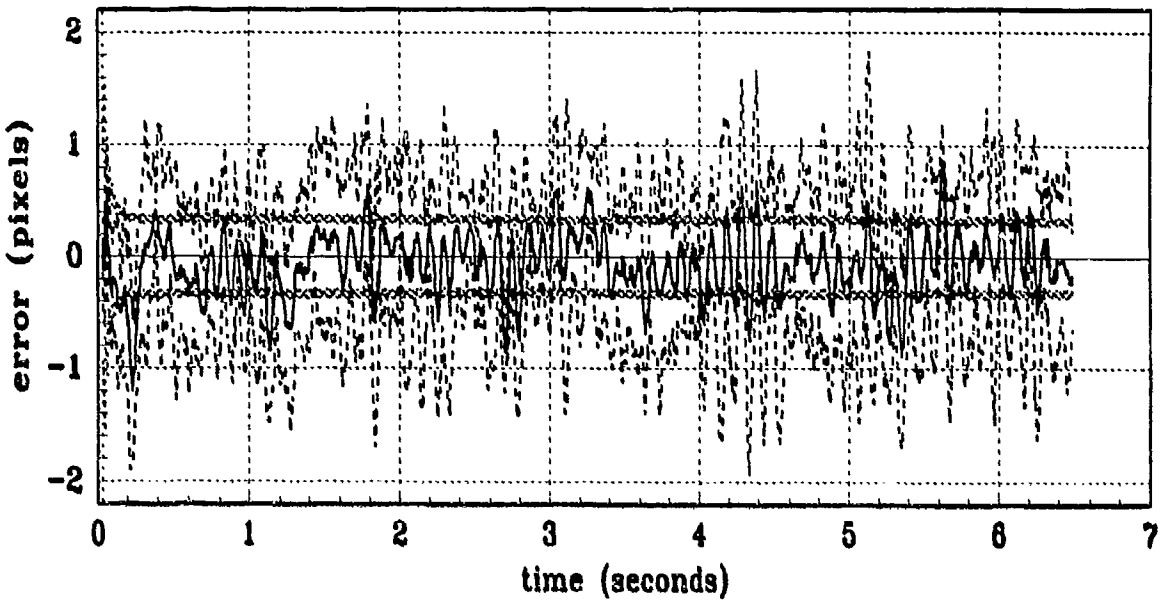
Figure E.18 Filter Residual Quantities
(Truth = 1. 12/2π, Filter = 0.112/20π)

Appendix F

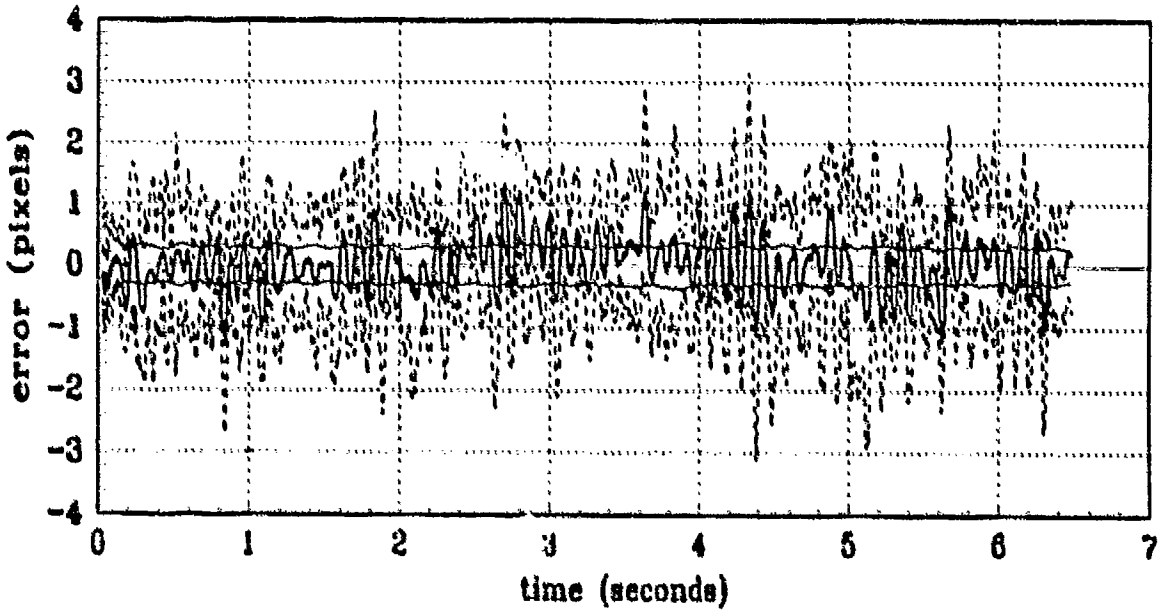
Filter Plots with Truth Parameter Values:

Amplitude = 1.12, Frequency = 20π

This appendix contains the state and error statistics plots of the nine-state elemental filters. The data depicted in the two types of plots in this appendix are explained in *Appendix A*. The state comparison plots show the truth state over the 5 *Monte Carlo* runs compared to the same statistic for the filter estimate. The error statistics plots represent the error mean $\pm$ standard deviation values in pixels (or pixel/second for velocity and pogo velocity), of the errors between the filter estimate and true state; true mean $\pm$ 1 true standard deviation are plotted, along with zero $\pm$ 1 filter-computed standard deviation.

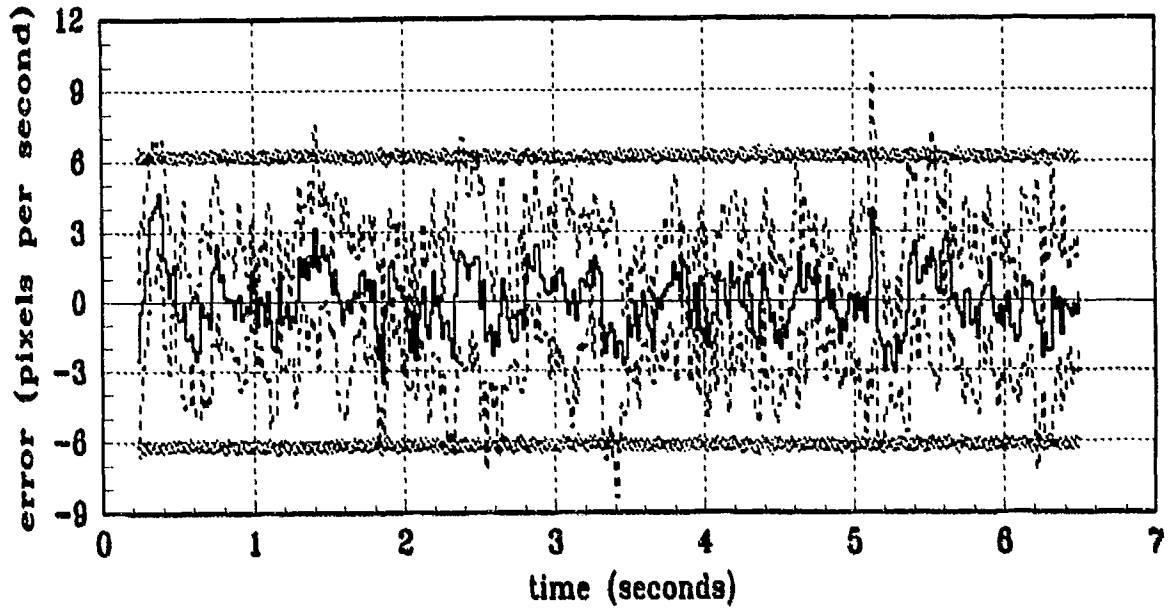


(a) FILTER vs ACTUAL ERROR (X-POSITION)

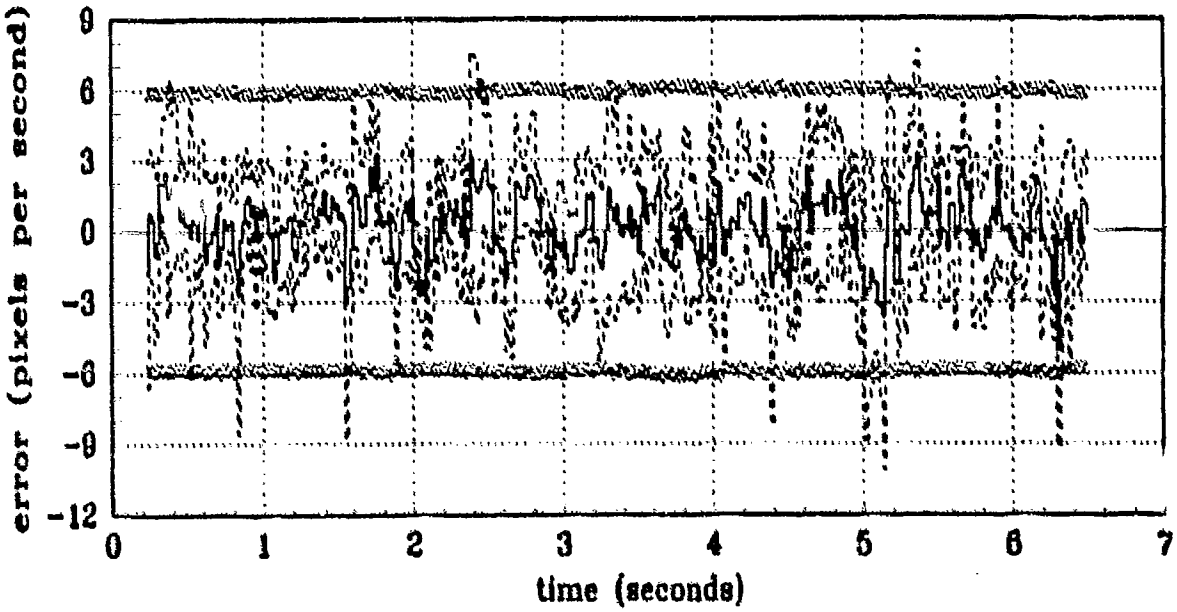


(b) FILTER vs ACTUAL ERROR (Y-POSITION)

Figure F.1 X/Y Position (Filter States 1 and 2) Error Statistics
(Truth = $1.12/20\pi$, Filter = $1.12/20\pi$)



(a) FILTER vs ACTUAL ERROR (X-VELOCITY)



(b) FILTER vs ACTUAL ERROR (Y-VELOCITY)

Figure F.2 X/Y Velocity (Filter States 3 and 4) Error Statistics
(Truth = $1.12/20\pi$, Filter = $1.12/20\pi$)

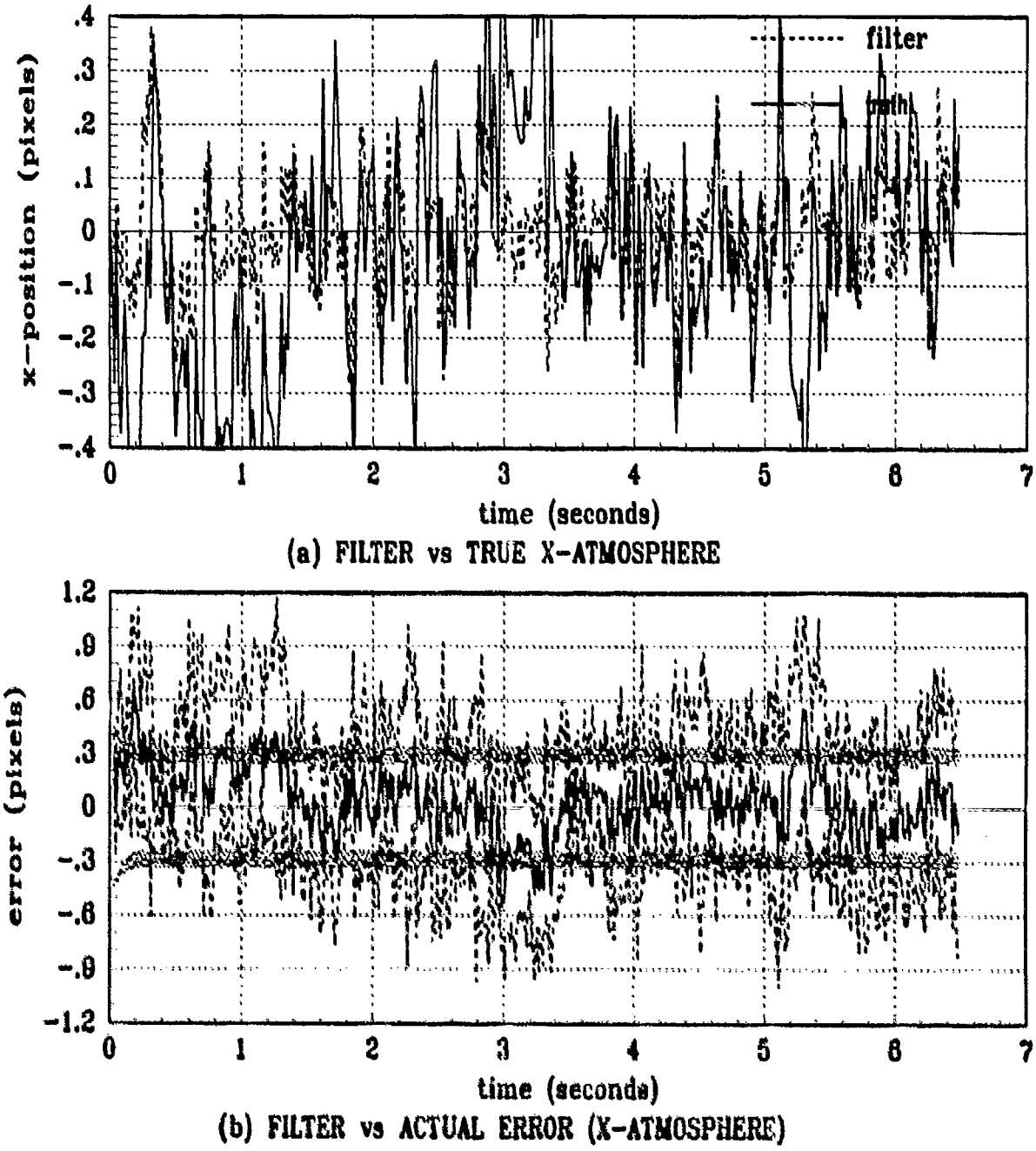


Figure F.3 X Atmospheric Jitter (Filter State 5)
(Truth = $1.12/20\pi$, Filter = $1.12/20\pi$)

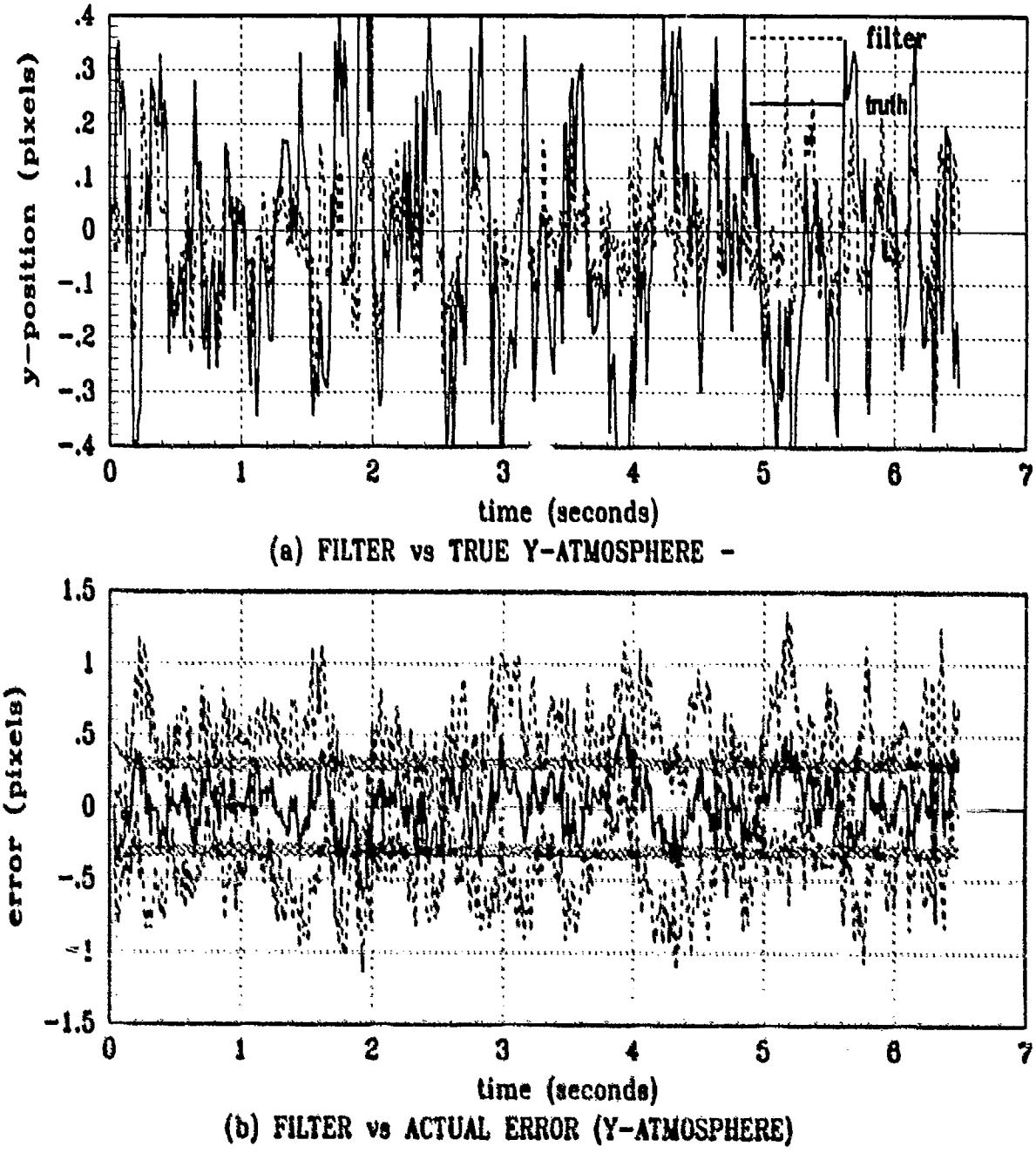


Figure F.4 Y Atmospheric Jitter (Filter State 6)
(Truth = $1.12/20\pi$, Filter = $1.12/20\pi$)

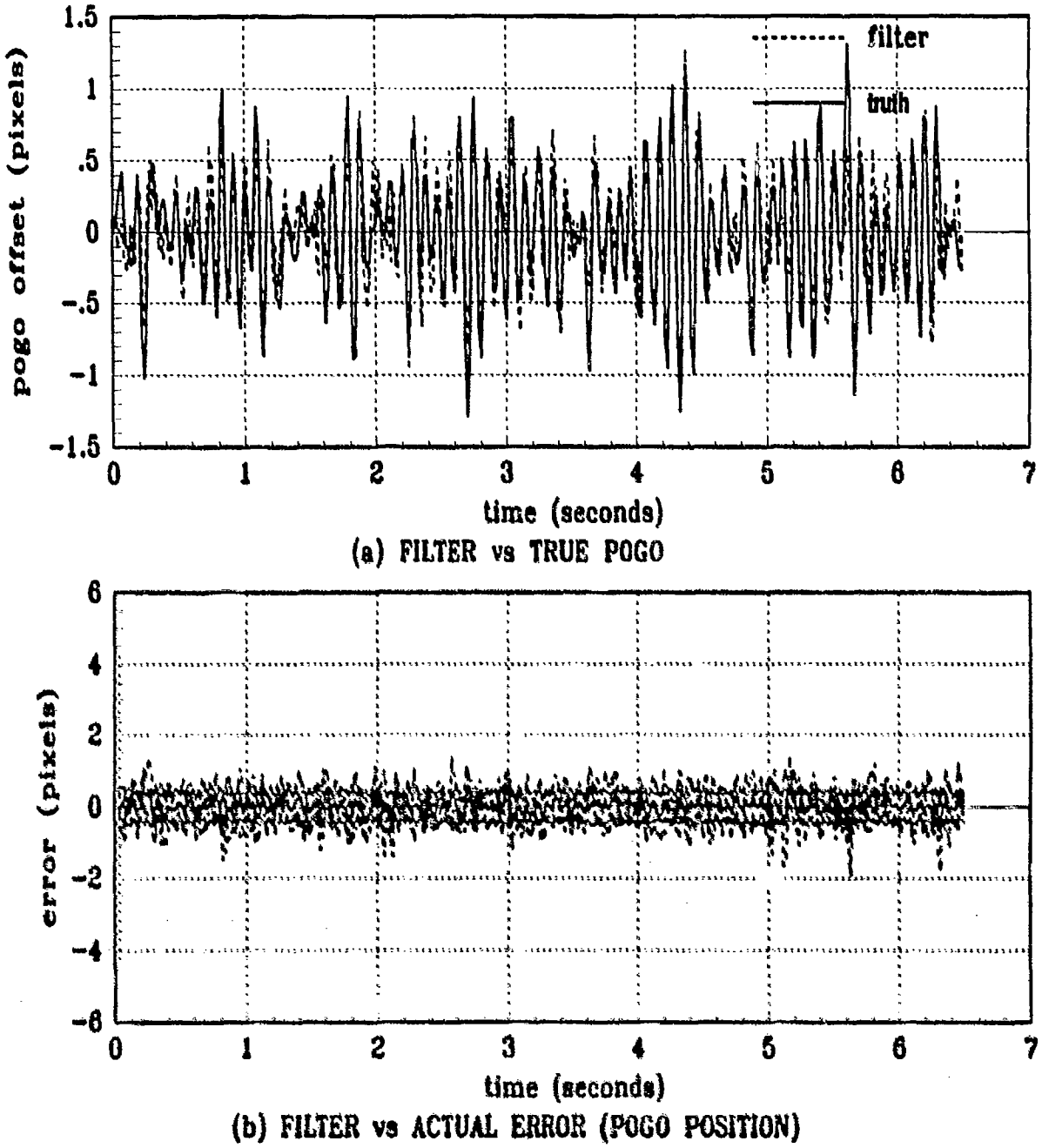
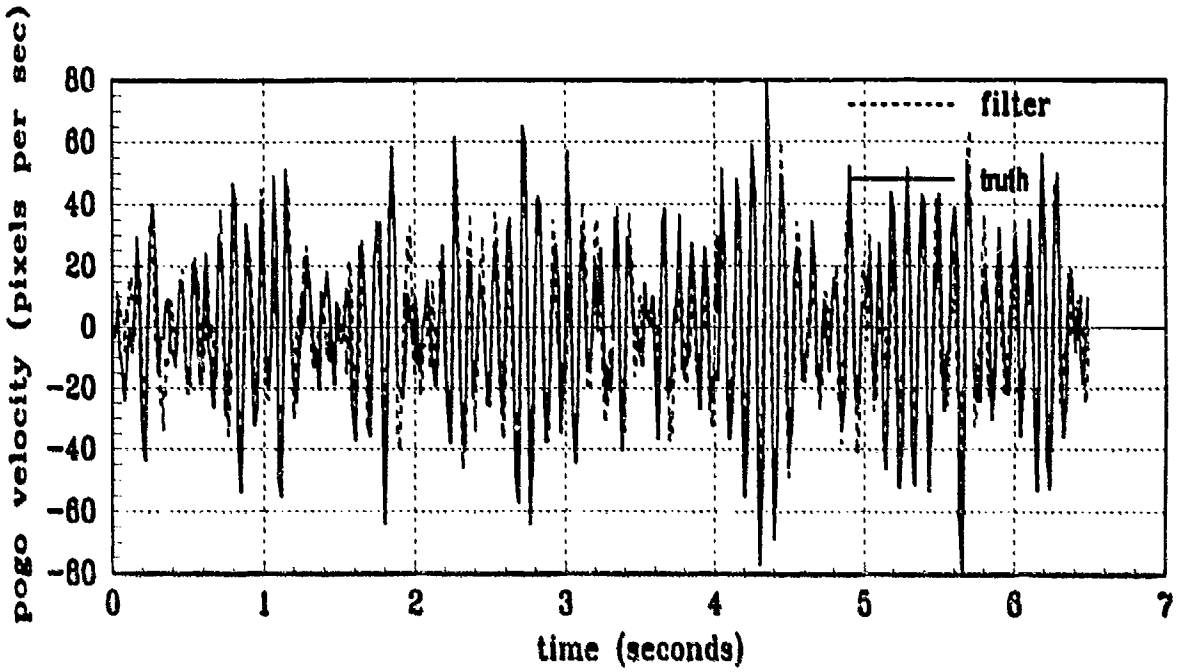
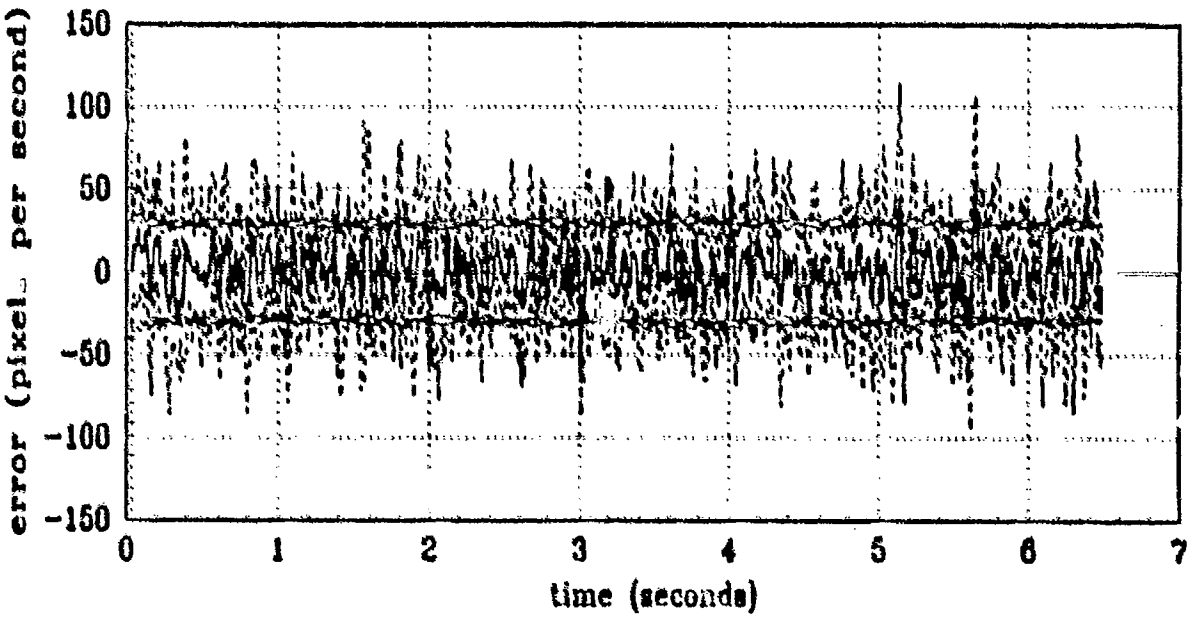


Figure F.5 Pogo Position Offset (Filter State 7)
(Truth = $1.12/20\pi$, Filter = $1.12/20\pi$)

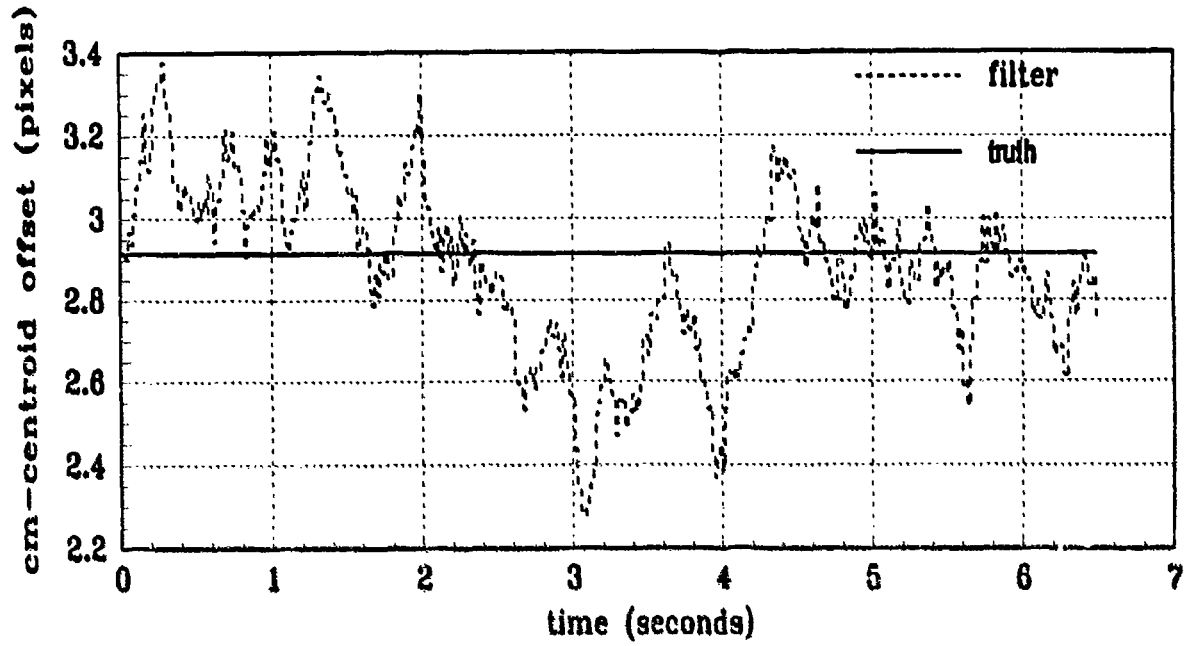


(a) FILTER vs TRUE POGO VELOCITY

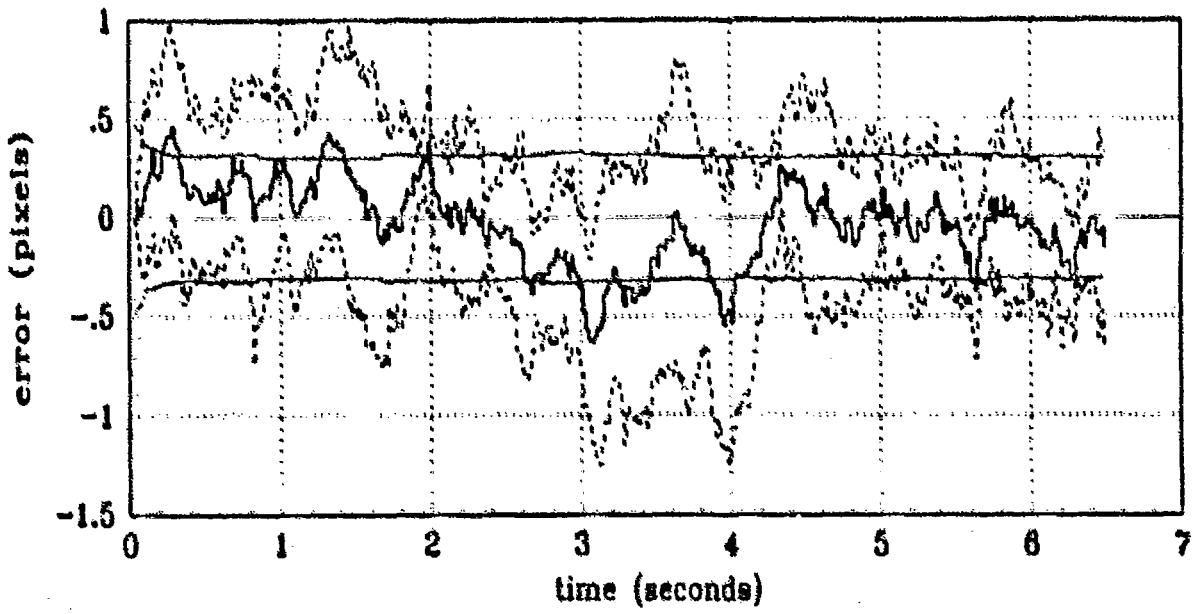


(b) FILTER vs ACTUAL ERROR (POGO VELOCITY)

Figure F.6 Pogo Velocity (Filter State 8)
(Truth = 1.12/20π, Filter = 1.12/20π)

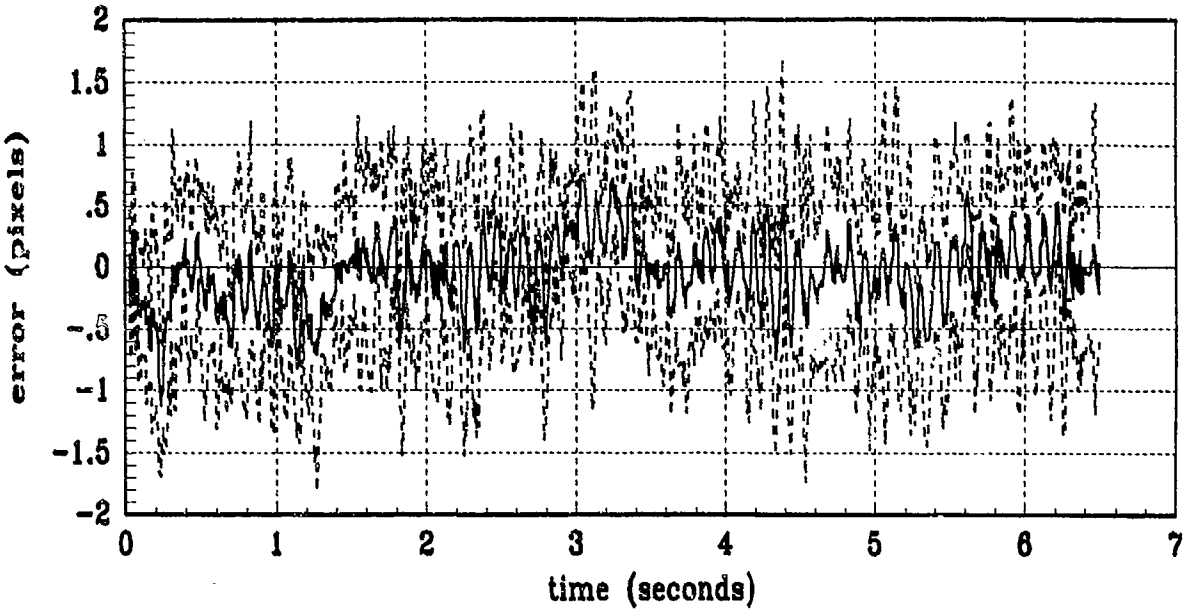


(a) FILTER vs TRUE COM-EQUILIBRIUM POINT OFFSET

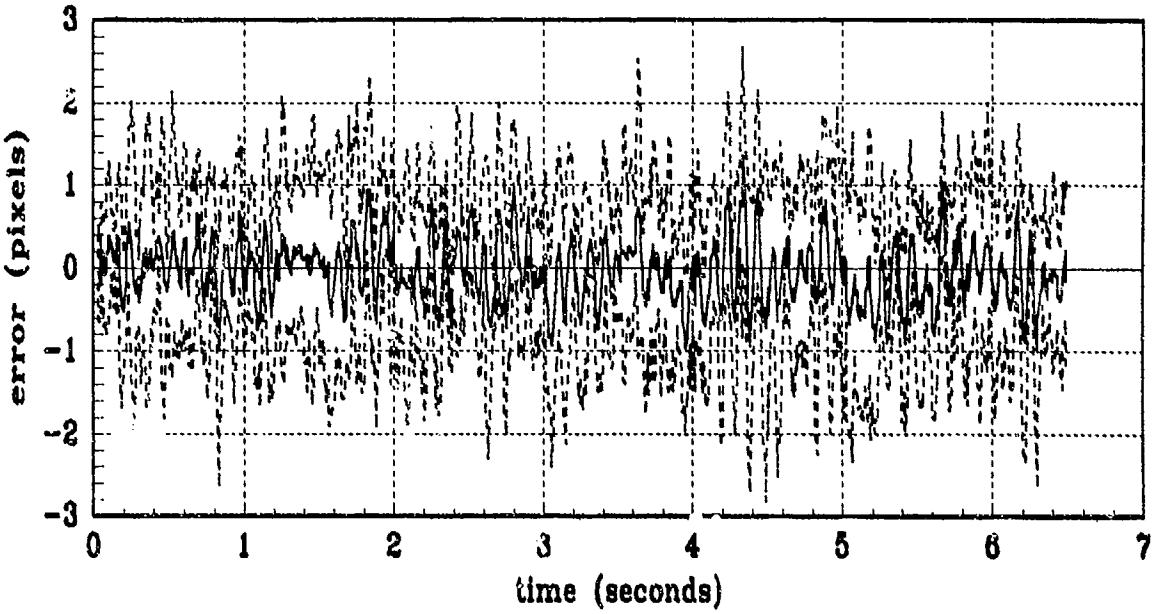


(b) FILTER vs ACTUAL ERROR (COM-EQUILIBRIUM POINT OFFSET)

Figure F.7 Center-of-Mass to Equilibrium Offset (Filter State 9)
(Truth = $1.12/20\pi$, Filter = $1.12/20\pi$)



(a) FILTER vs ACTUAL ERROR (X-CENTROID POSITION)



(b) FILTER vs ACTUAL ERROR (Y-CENTROID POSITION)

Figure F.8 Plume Centroid Error Statistics
(Truth = $1.12/20\pi$, Filter = $1.12/20\pi$)

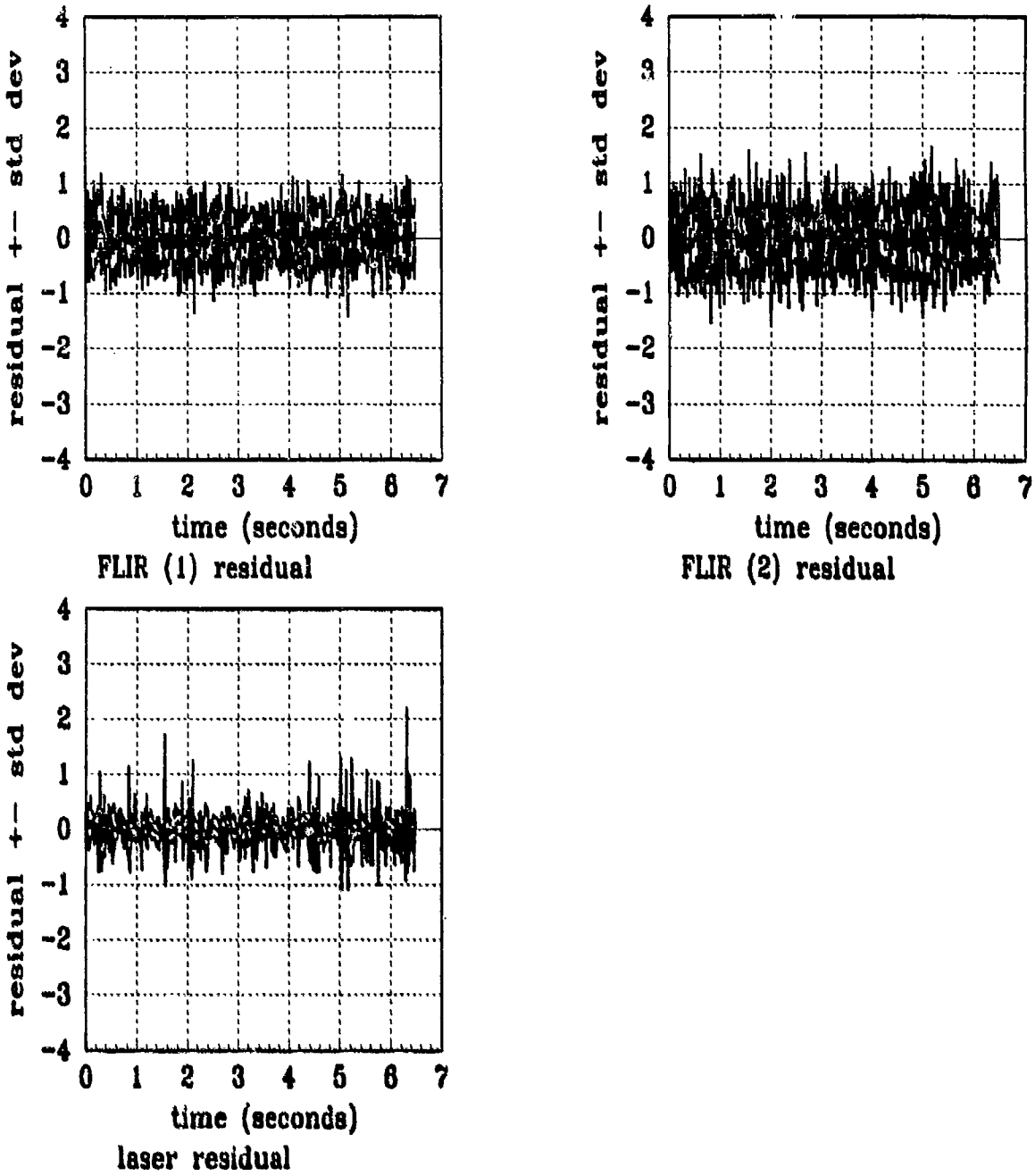
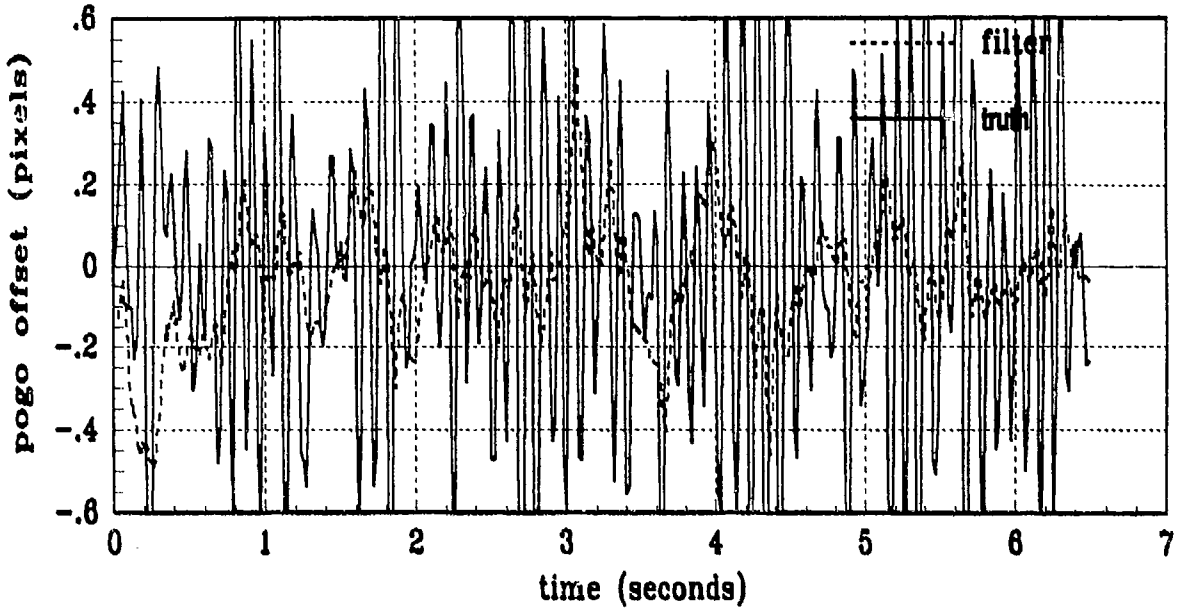
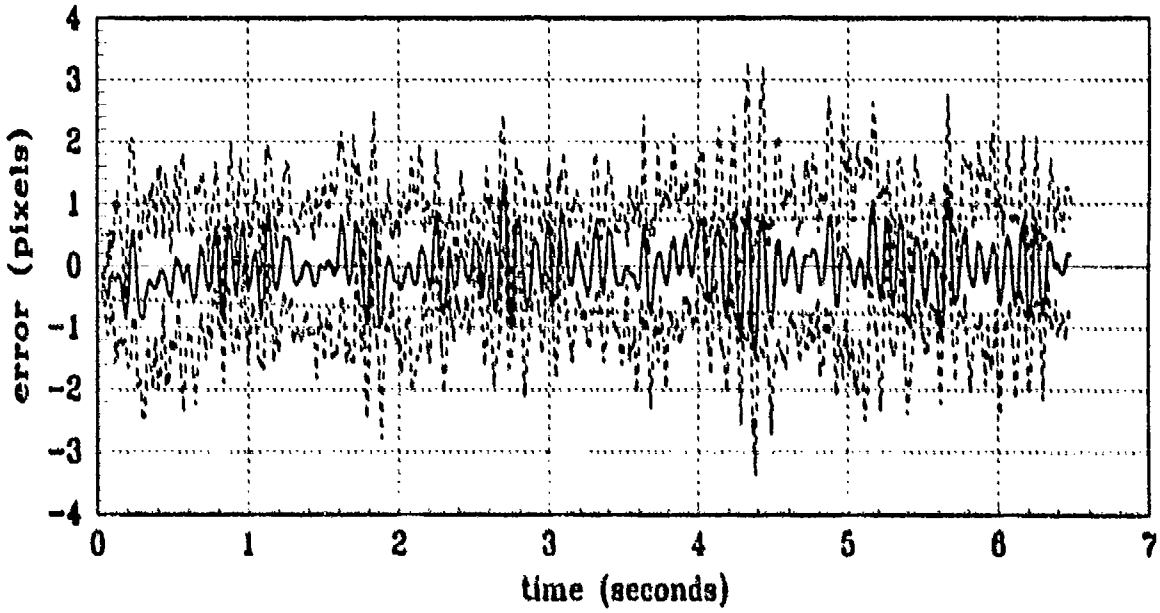


Figure F.9 Filter Residual Quantities
(Truth = $1.12/20\pi$, Filter = $1.12/20\pi$)

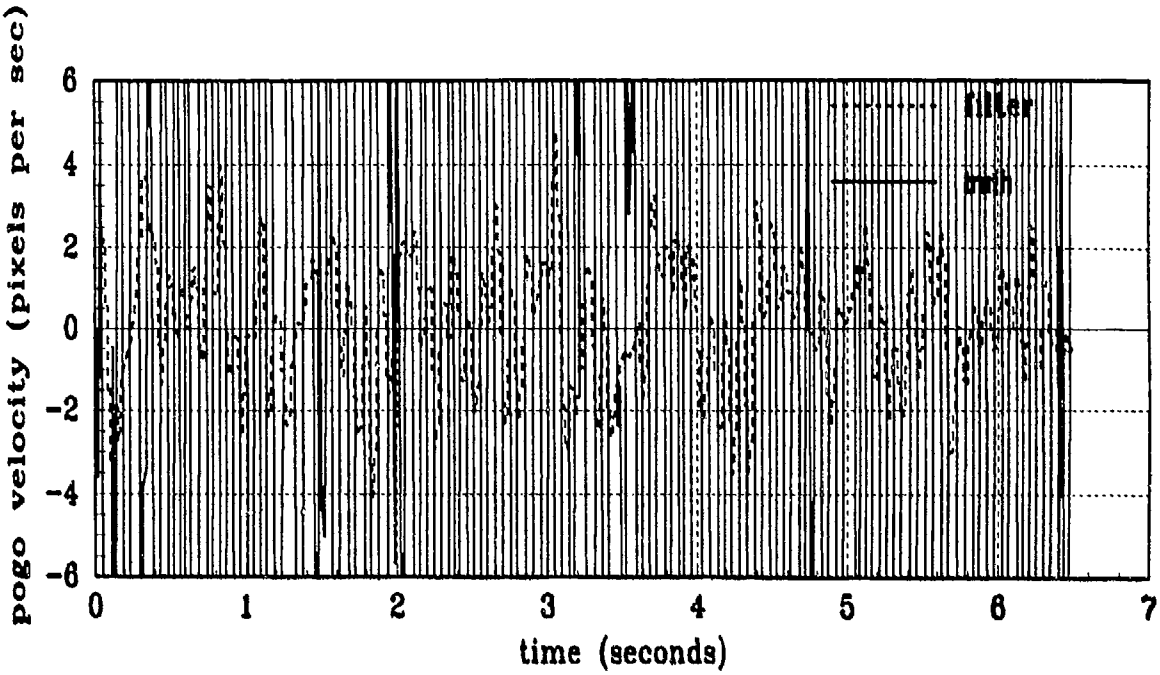


(a) FILTER vs TRUE POGO

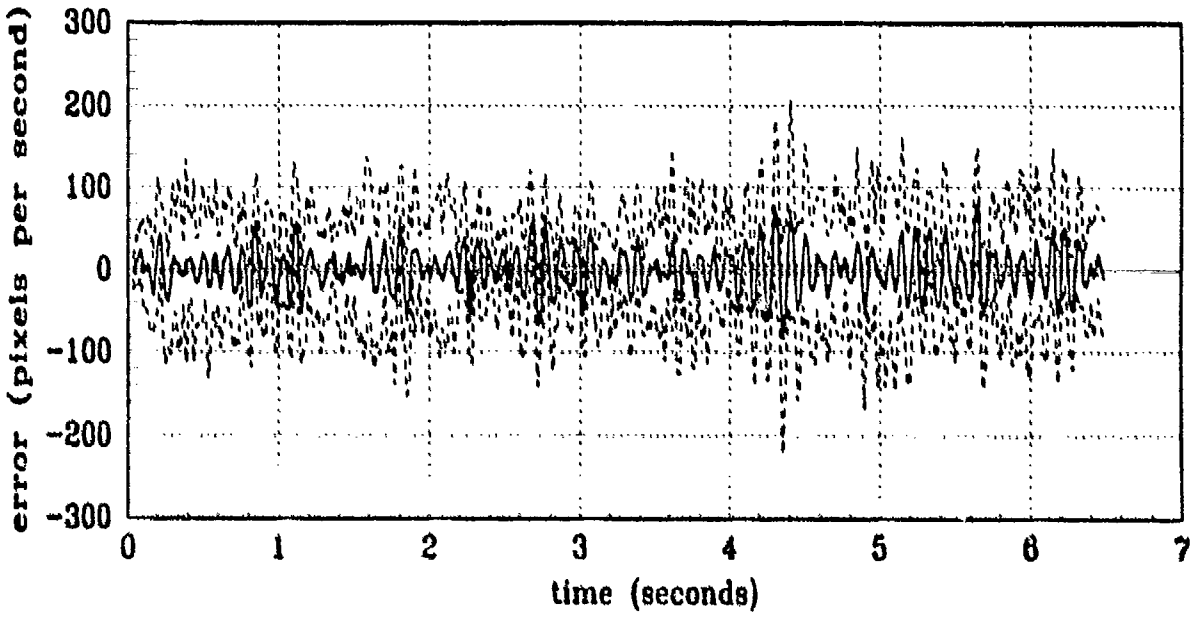


(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure F.10 Pogo Position (Filter State 7)
(Truth = $1.12/20\pi$, Filter = $1.12/2\pi$)

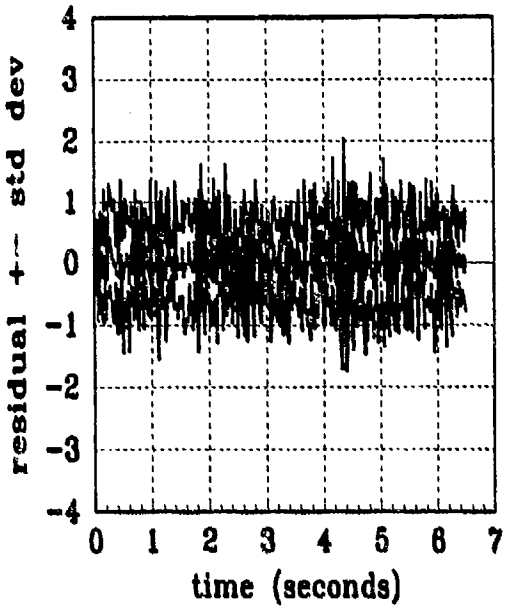


(a) FILTER vs TRUE POGO VELOCITY

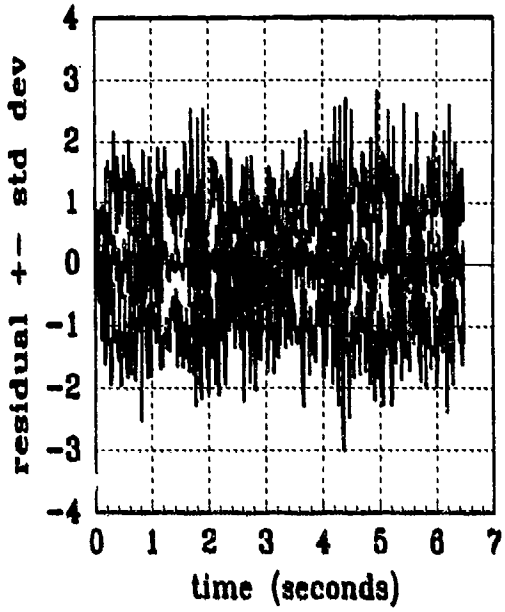


(b) FILTER vs ACTUAL ERROR (POGO VELOCITY)

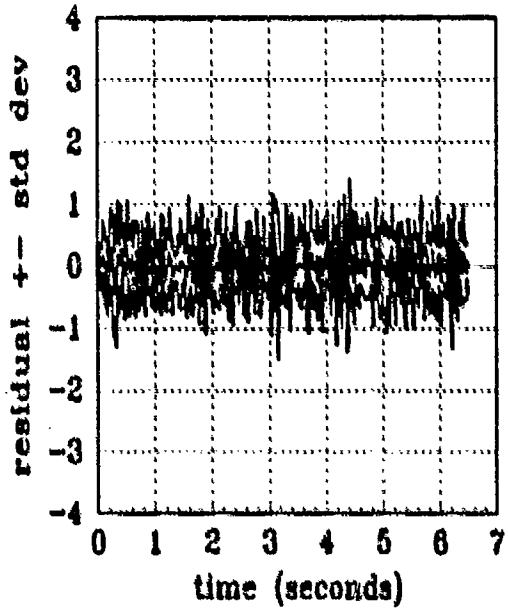
Figure F.11 Pogo Velocity (Filter State 8)
(Truth = $1.12/20\pi$, Filter = $1.12/2\pi$)



FLIR (1) residual

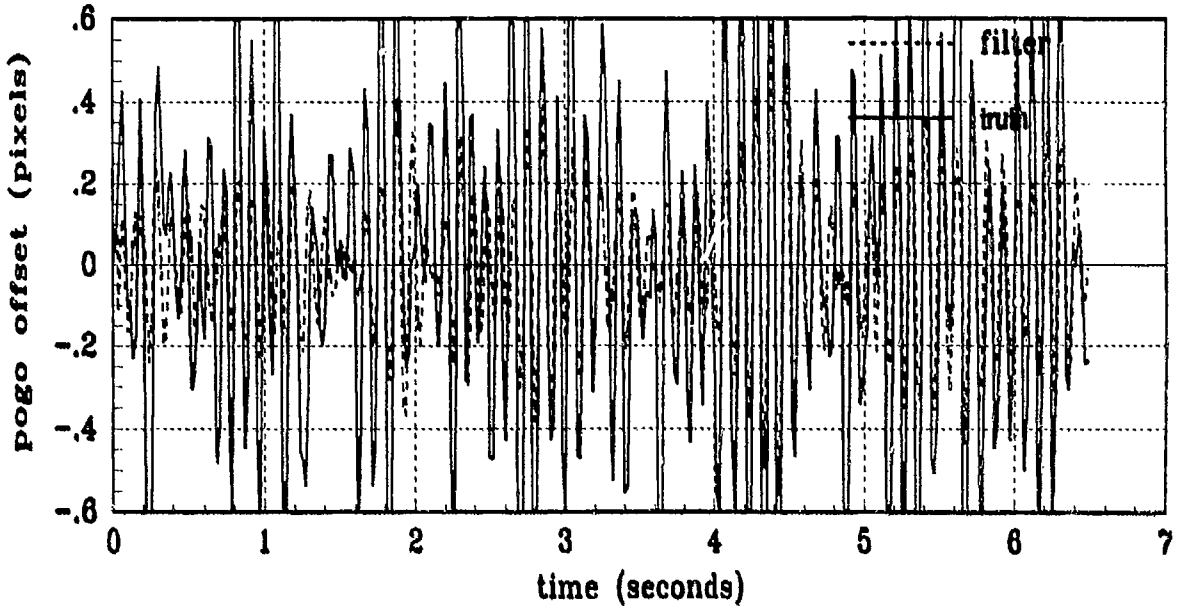


FLIR (2) residual

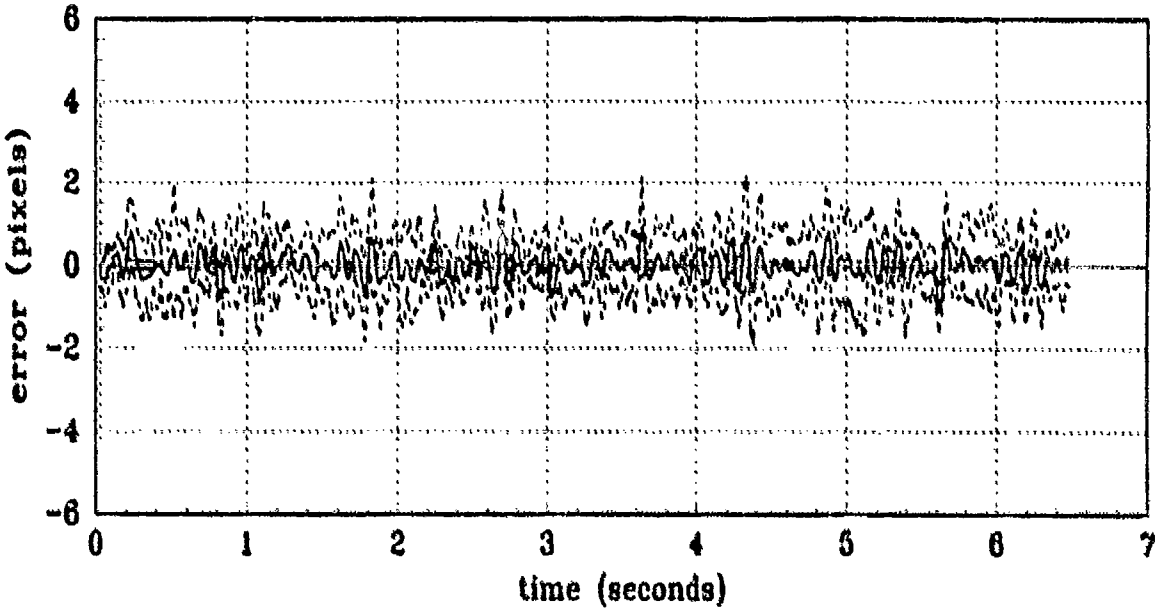


laser residual

Figure F.12 Filter Residual Quantities
(Truth = $1.12/20\pi$, Filter = $1.12/2\pi$)



(a) FILTER vs TRUE POGO



(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure F.13 Pogo Position (Filter State 7)
(Truth = $1.12/20\pi$, Filter = $0.112/20\pi$)

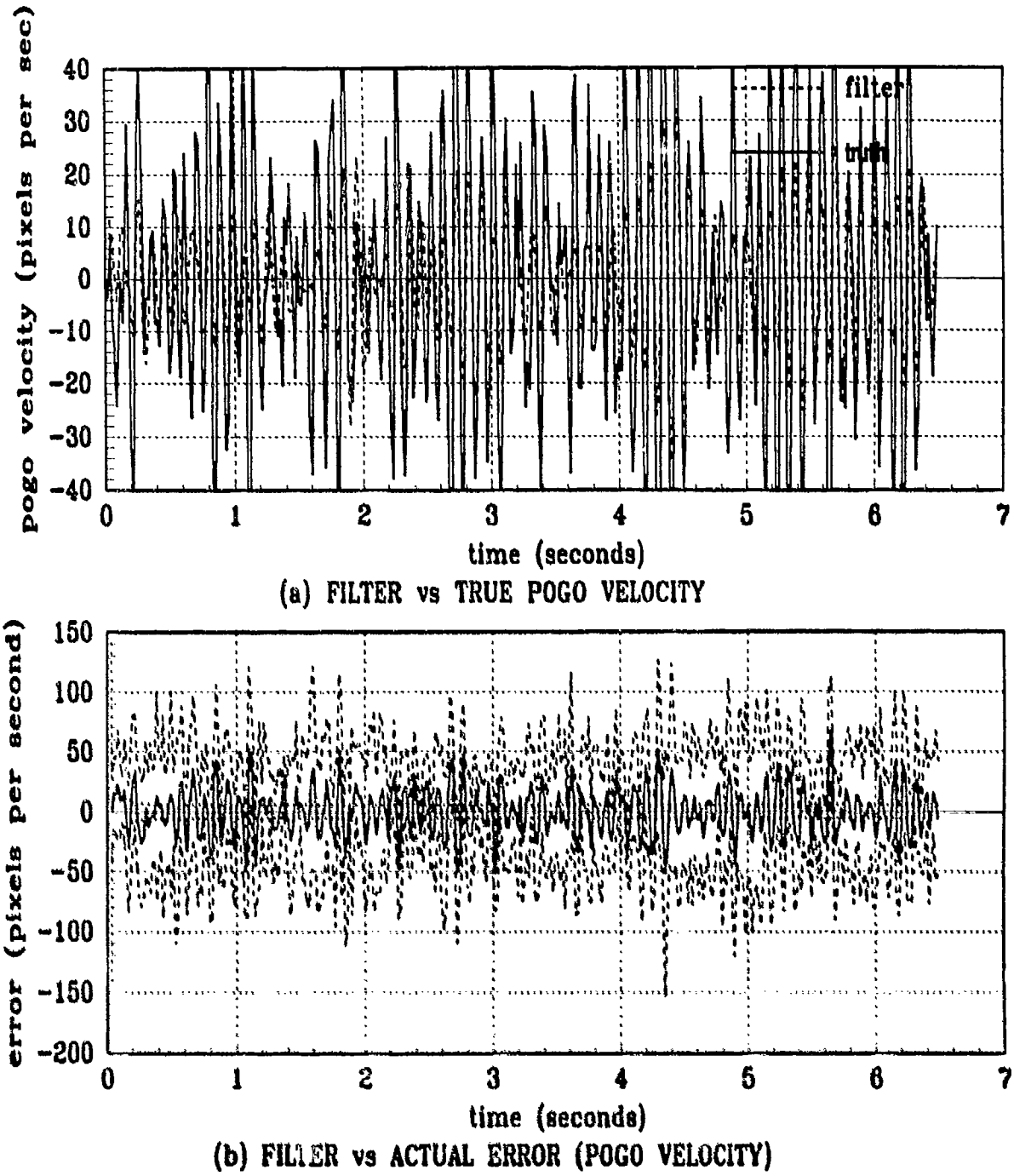


Figure F.14 Pogo Velocity (Filter State 8)
(Truth = $1.12/20\pi$, Filter = $0.112/20\pi$)

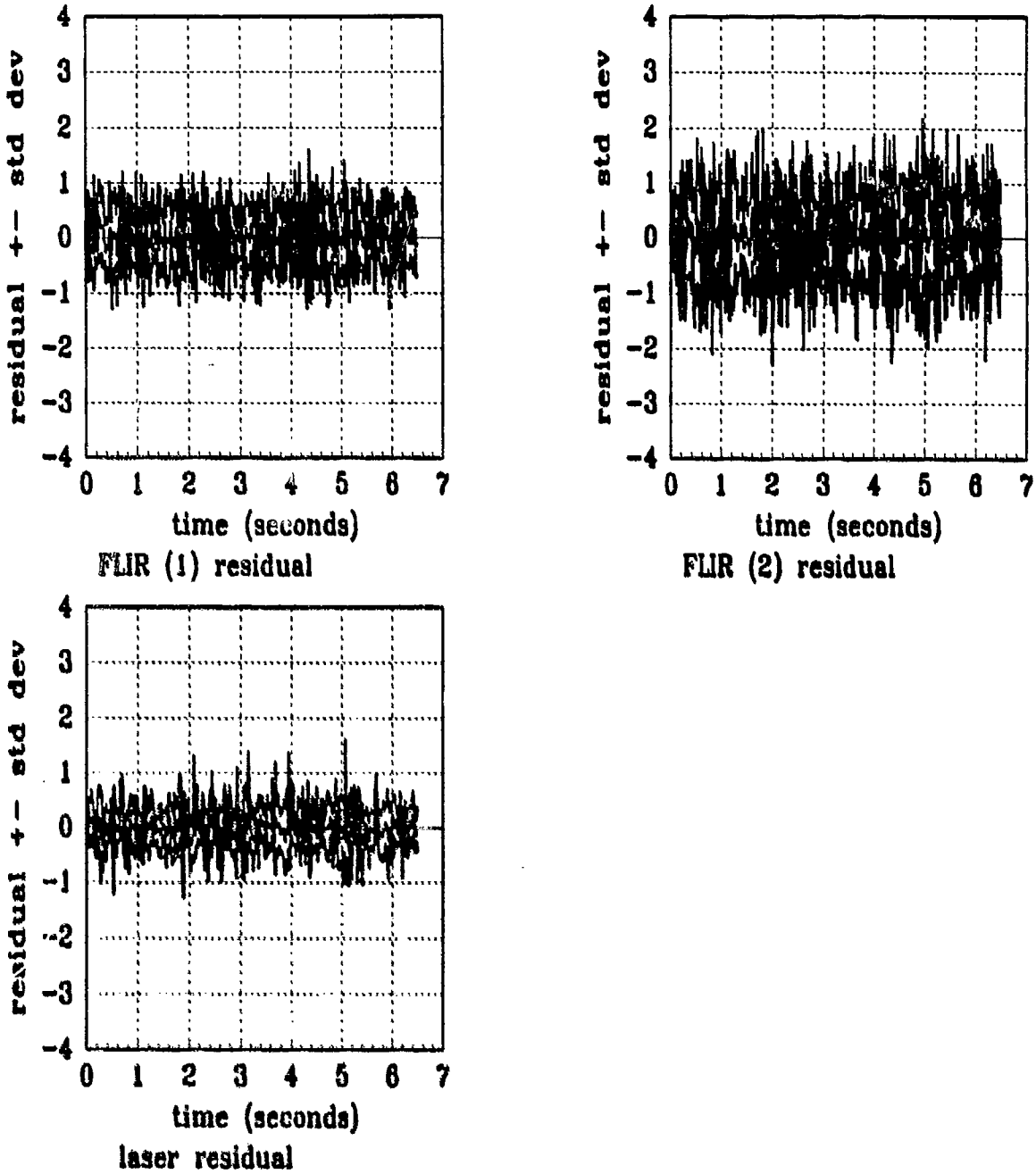
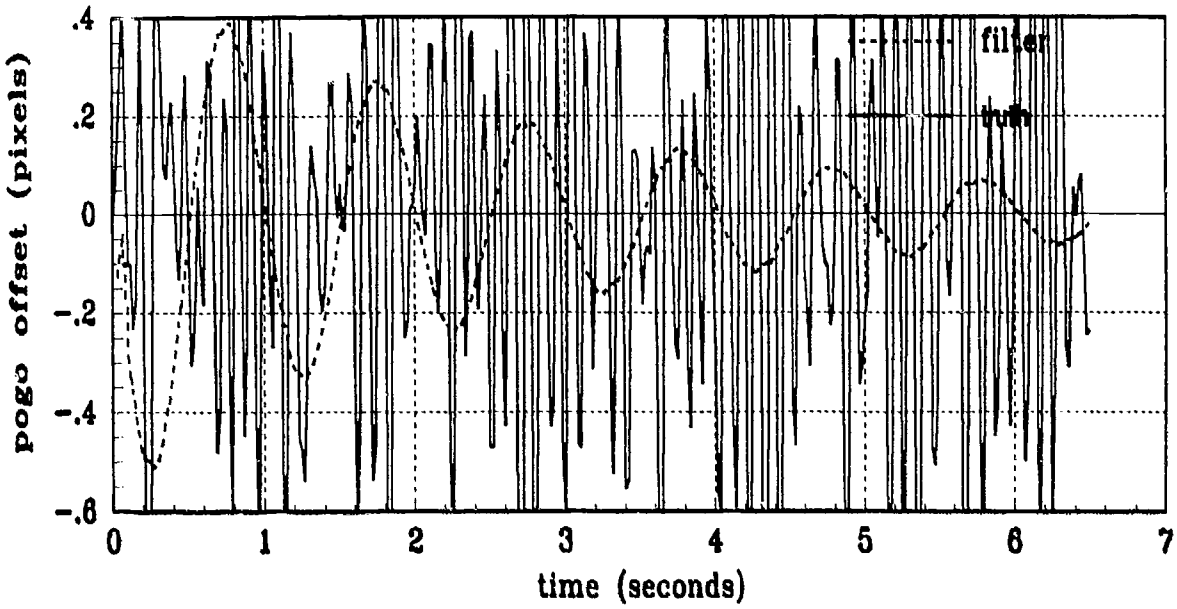
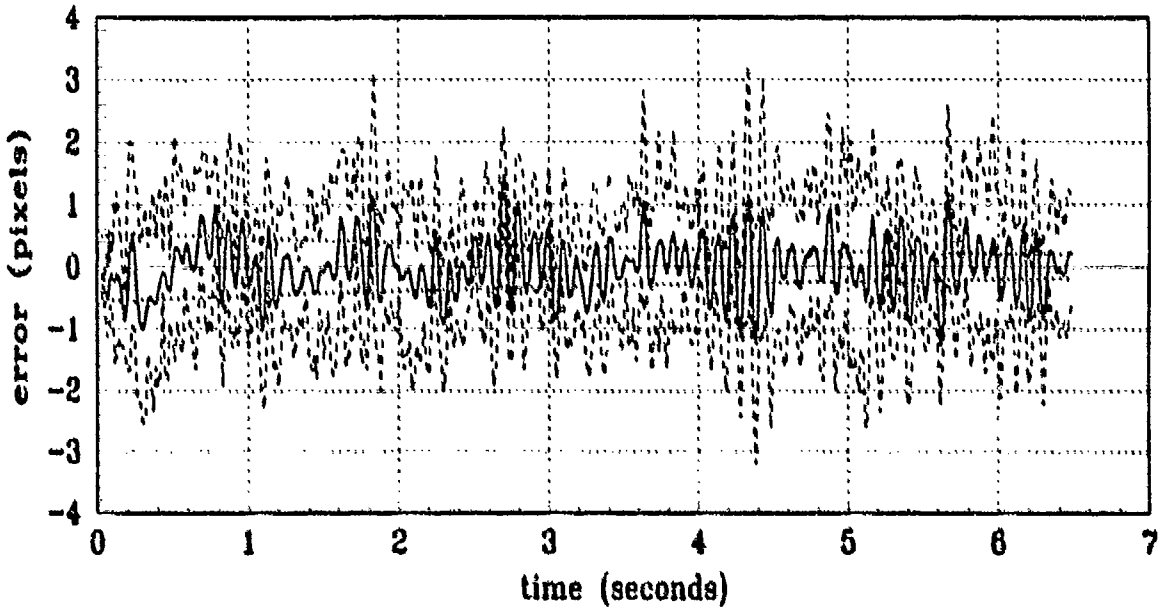


Figure F.15 Filter Residual Quantities
Truth = $1.12/20\pi$, Filter = $0.112/20\pi$)

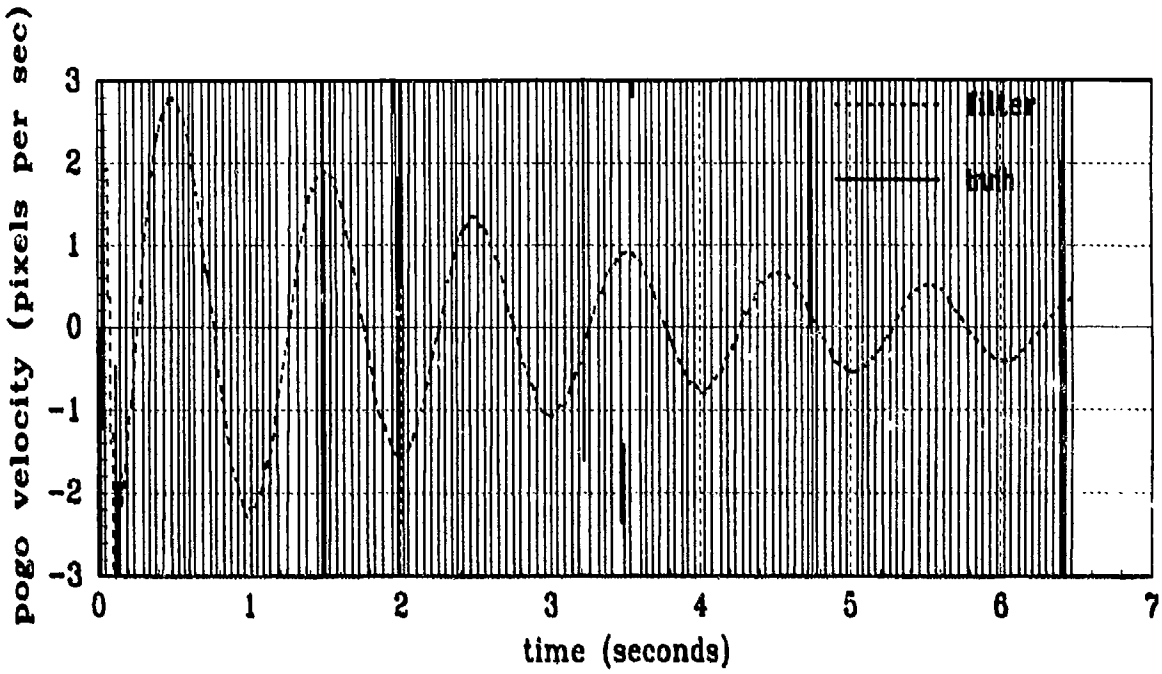


(a) FILTER vs TRUE POGO

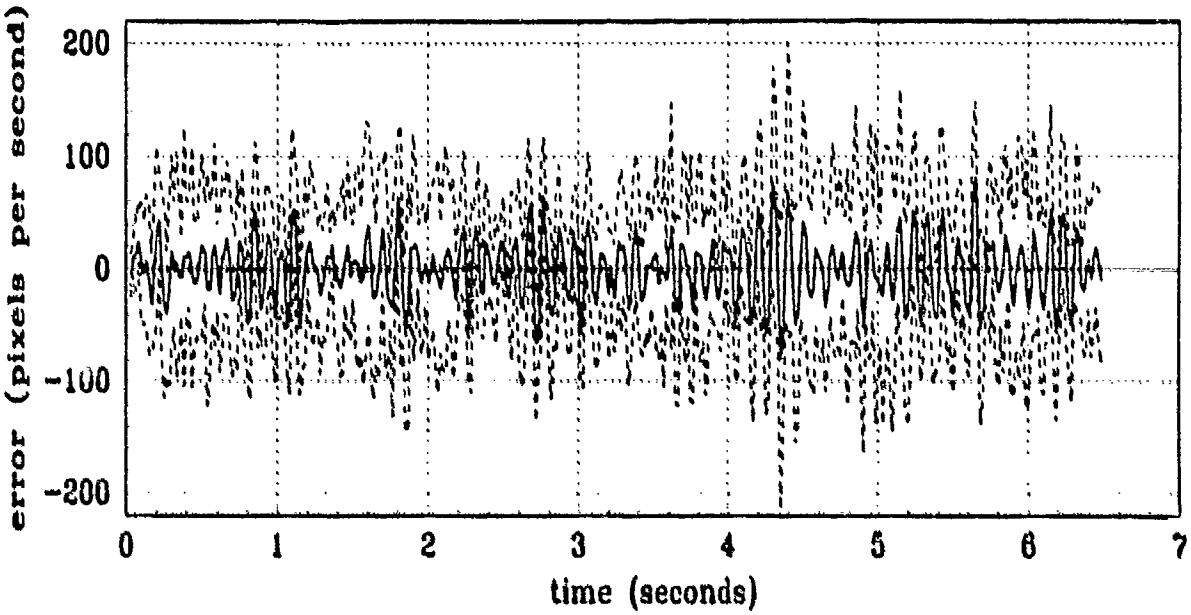


(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure F.16 Pogo Position (Filter State 7)
(Truth = $1.12/20\pi$, Filter = $0.112/2\pi$)



(a) FILTER vs TRUE POGO VELOCITY



(b) FILTER vs ACTUAL ERROR (POGO VELOCITY)

Figure F.17 Pogo Velocity (Filter State 8)
(Truth = $1.12/20\pi$, Filter = $0.112/2\pi$)

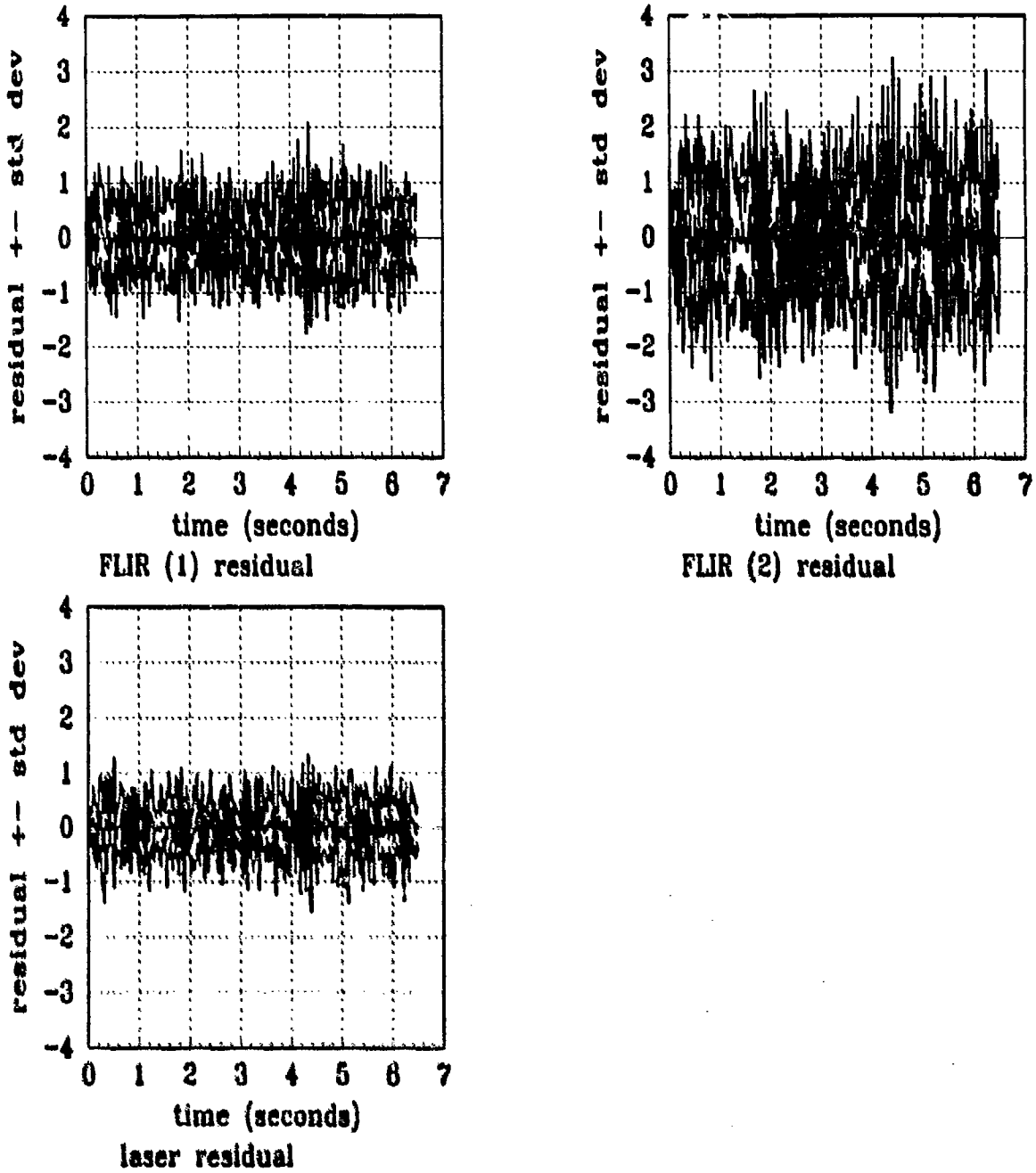
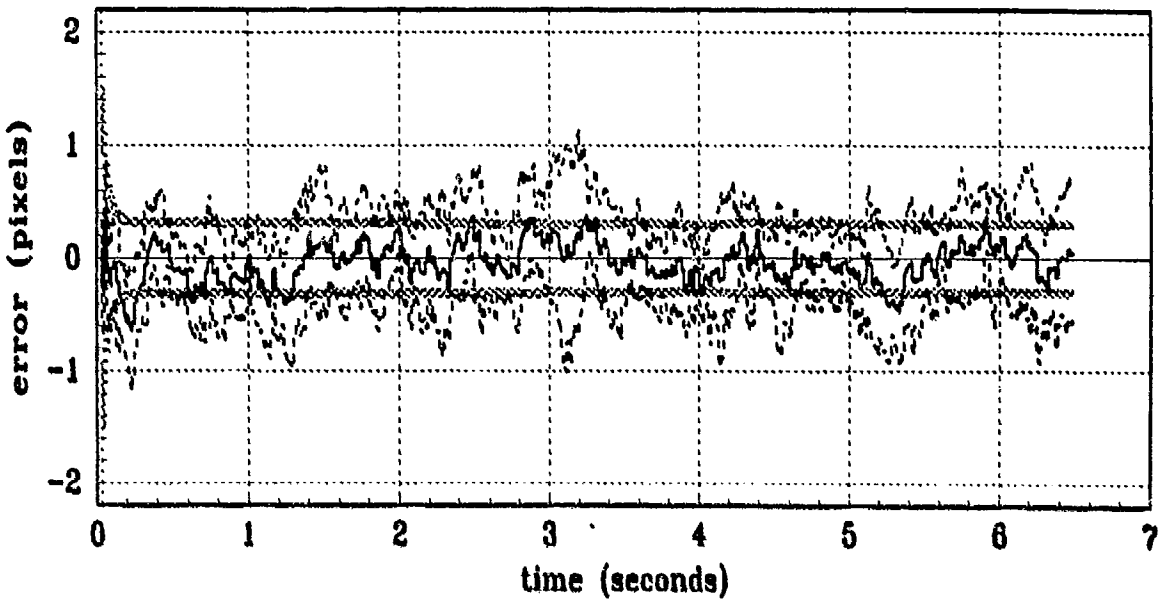


Figure F.18 Filter Residual Quantities
(Truth = $1.12/20\pi$, Filter = $0.112/2\pi$)

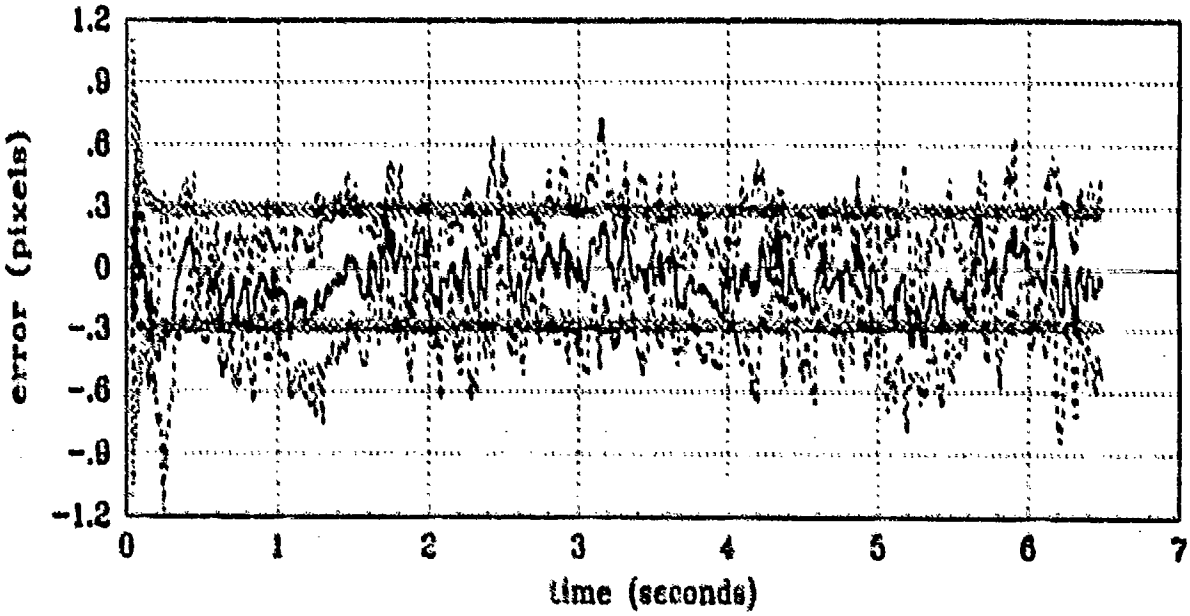
Appendix G

MMAE Configuration #1 Output Plots

This appendix contains the state and error statistics plots of the nine-state elemental filters. The data depicted in the two types of plots in this appendix are explained in ***Appendix A and B***. The state comparison plots show the sample mean truth state over the 5 ***Monte Carlo*** runs compared to the same statistic for the filter estimate. The error statistics plots represent the error mean $\pm$ standard deviation values in pixels (or pixel/second for velocity and pogo velocity), of the errors between the filter estimate and true state; true mean $\pm$ one true standard deviation are plotted, along with zero $\pm$ one filter-computed standard deviation. Note the filter covariance calculation is shown in Equation (4.5).

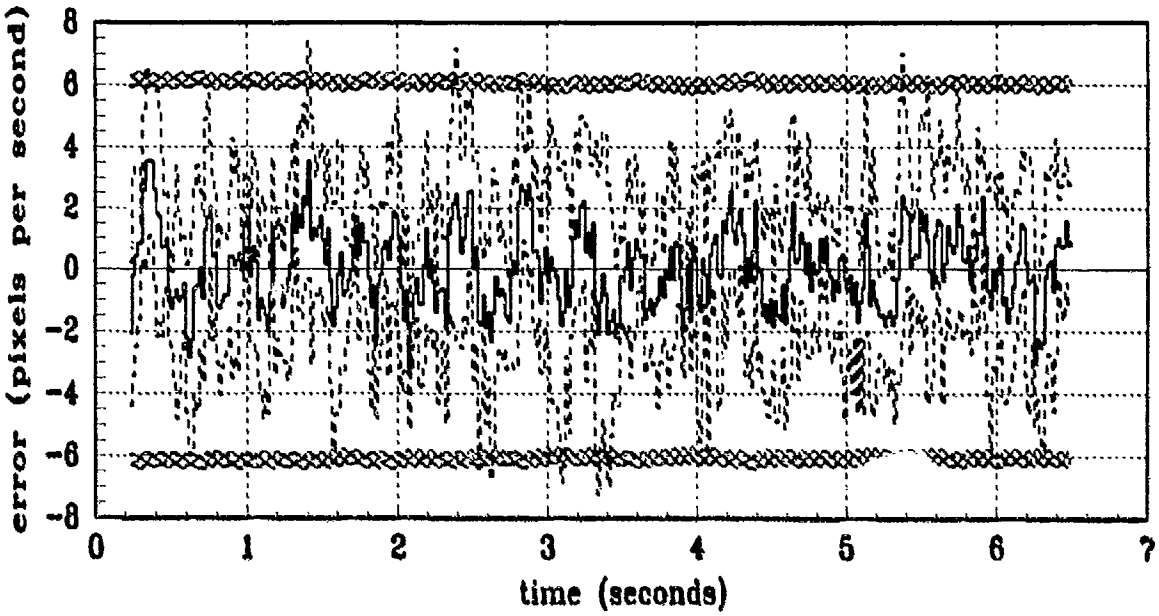


(a) FILTER vs ACTUAL ERROR (X-POSITION)

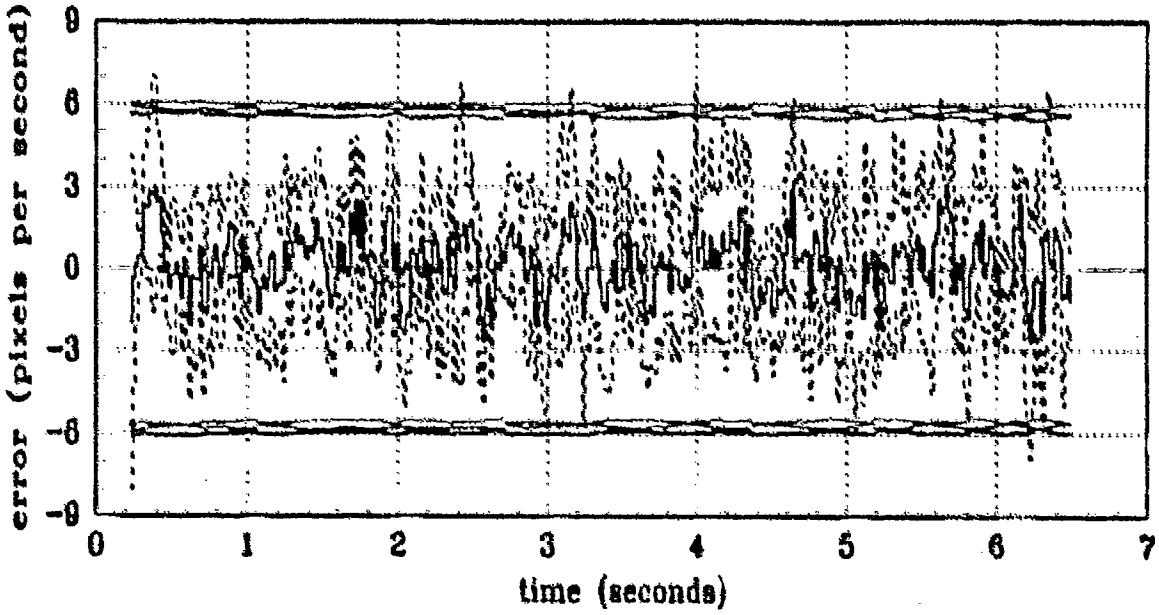


(b) FILTER vs ACTUAL ERROR (Y-POSITION)

Figure G.1 X/Y Position (Filter States 1 and 2) Error Statistics
(Truth = $0.112/20\pi$)

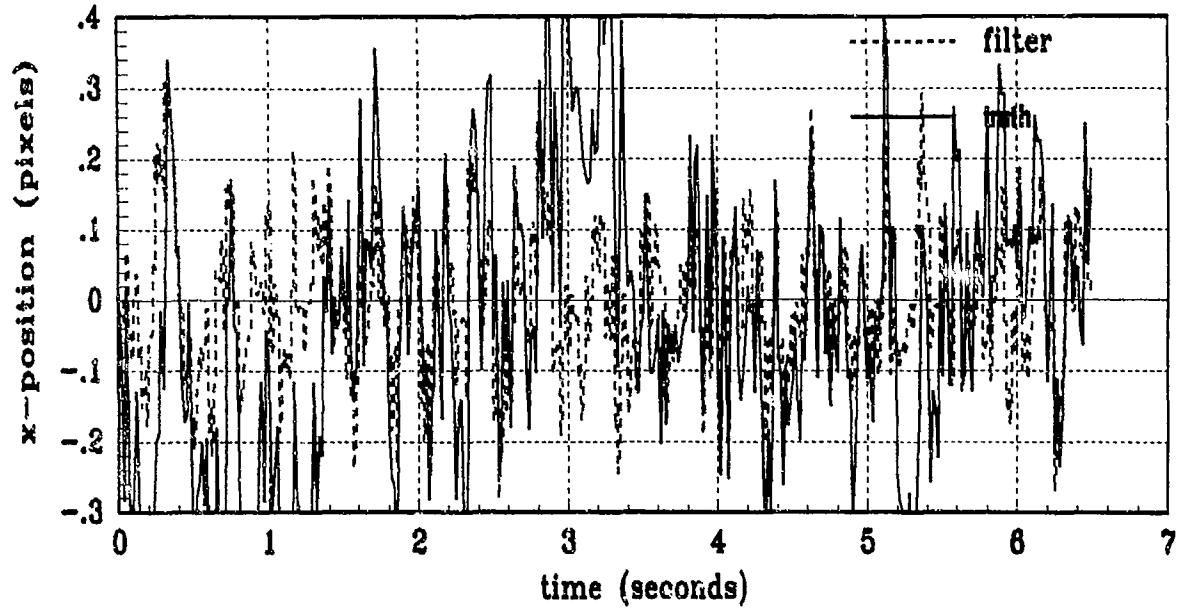


(a) FILTER vs ACTUAL ERROR (X-VELOCITY)

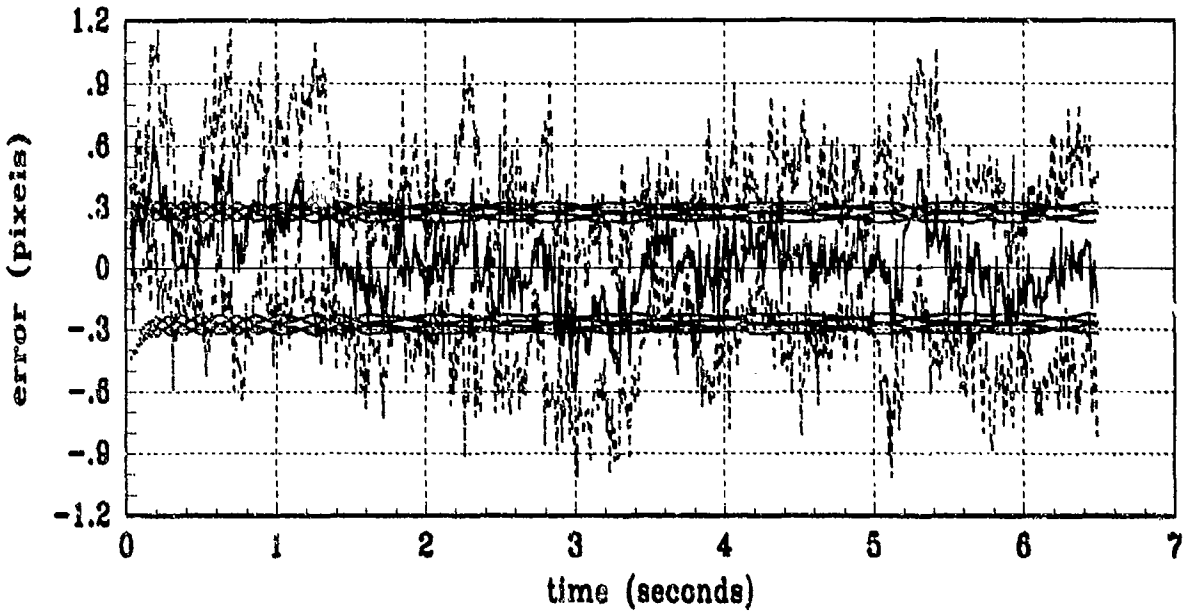


(b) FILTER vs ACTUAL ERROR (Y-VELOCITY)

Figure G.2 X/Y Velocity (Filter States 3 and 4) Error Statistics
(Truth = $0.112/20\pi$)

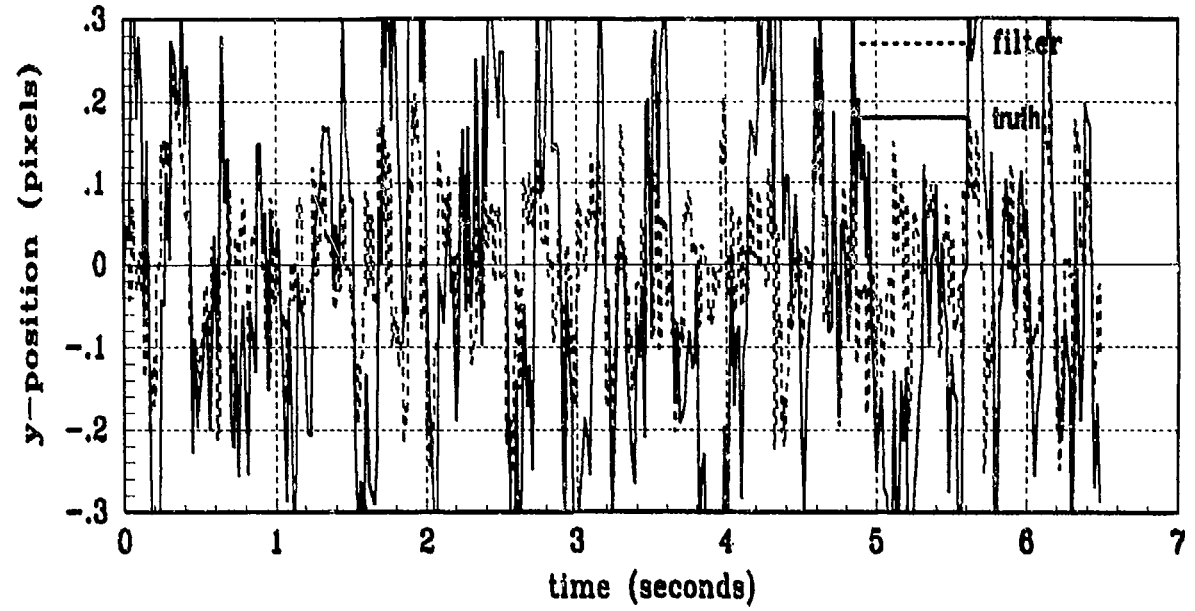


(a) FILTER vs TRUE X-ATMOSPHERE

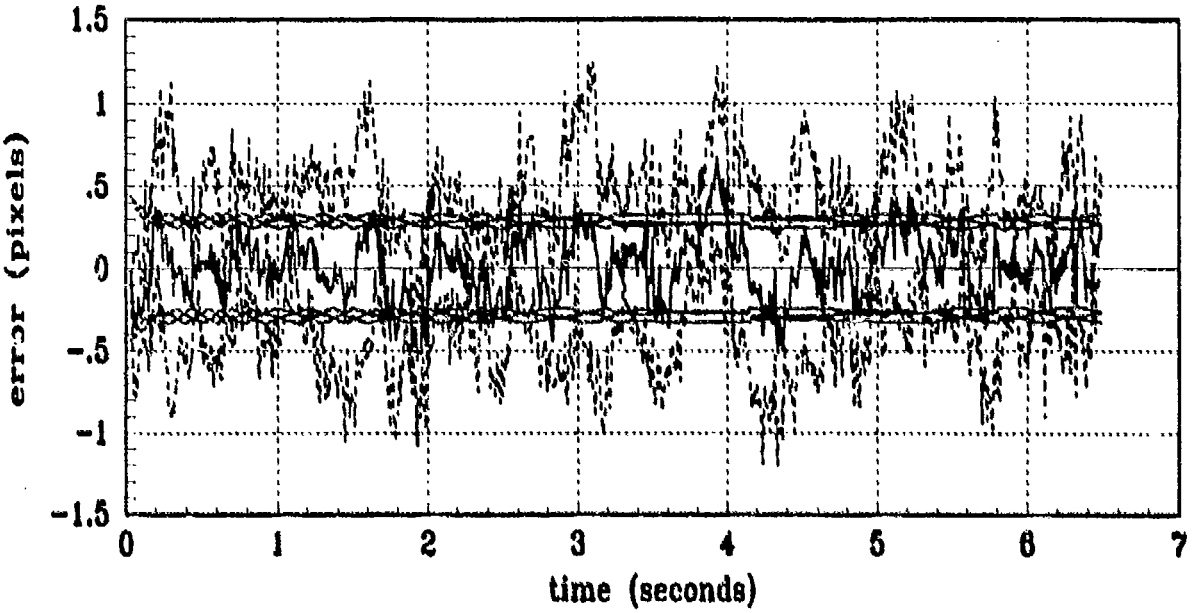


(b) FILTER vs ACTUAL ERROR (X-ATMOSPHERE)

Figure G.3 X Atmospheric Jitter (Filter State 5)
(Truth = $0.112/20\pi$)

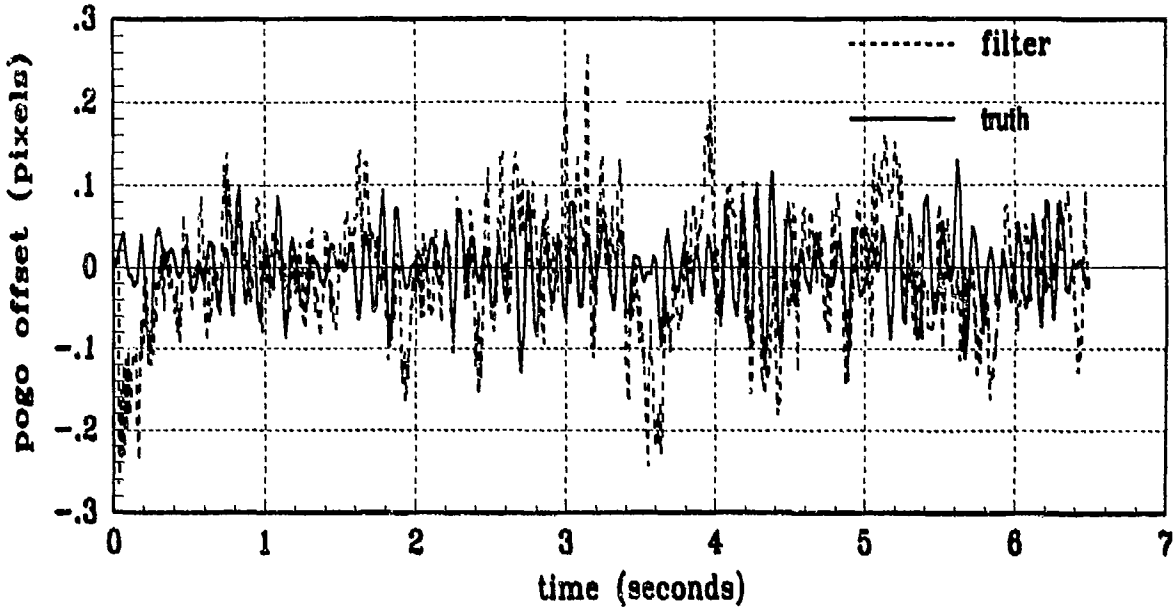


(a) FILTER vs TRUE Y-ATMOSPHERE -

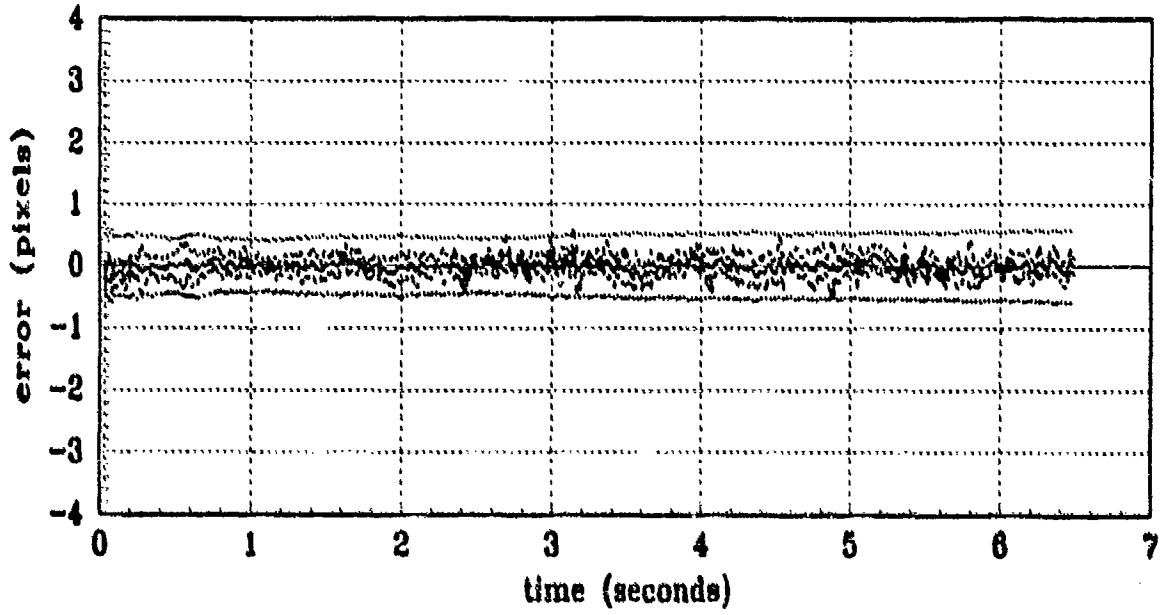


(b) FILTER vs ACTUAL ERROR (Y-ATMOSPHERE)

Figure G.4 Y Atmospheric Jitter (Filter State 6)
(Truth = 0.112/20π)

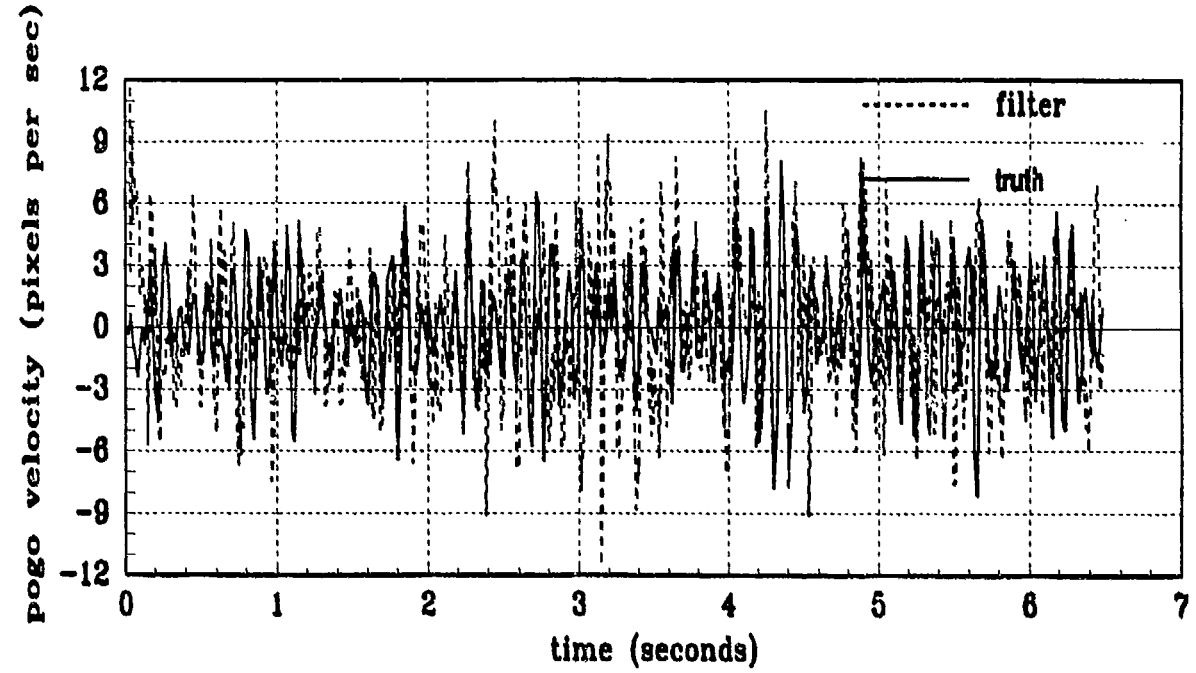


(a) FILTER vs TRUE POGO

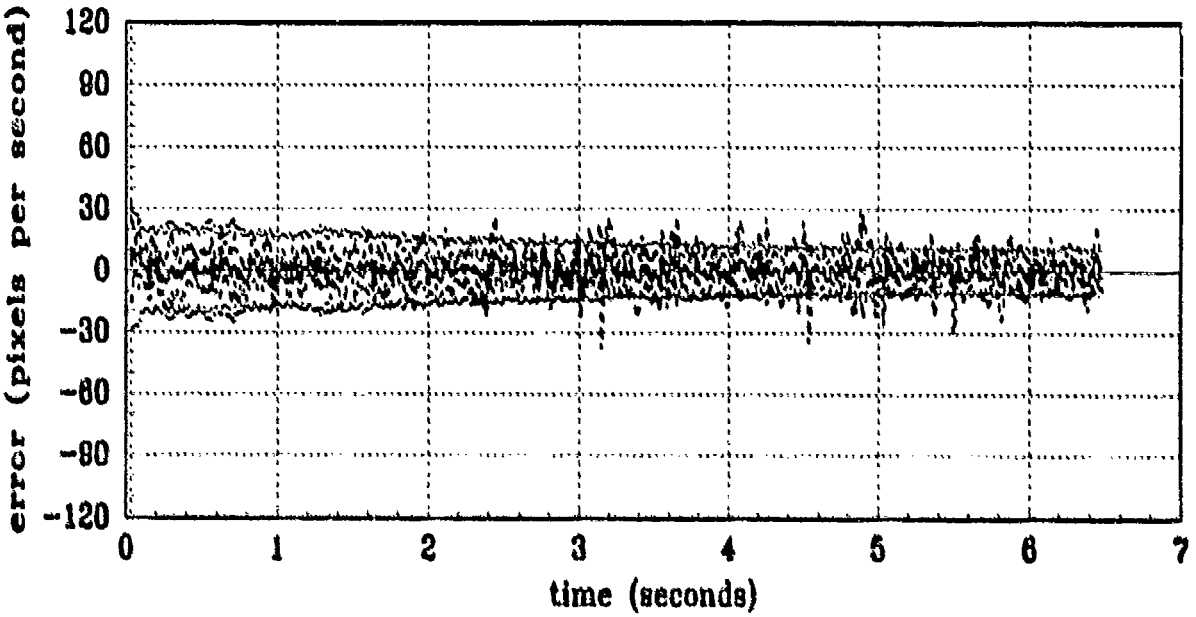


(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure G.5 Pogo Position Offset (Filter State 7)
(Truth = $0.112/20\pi$)

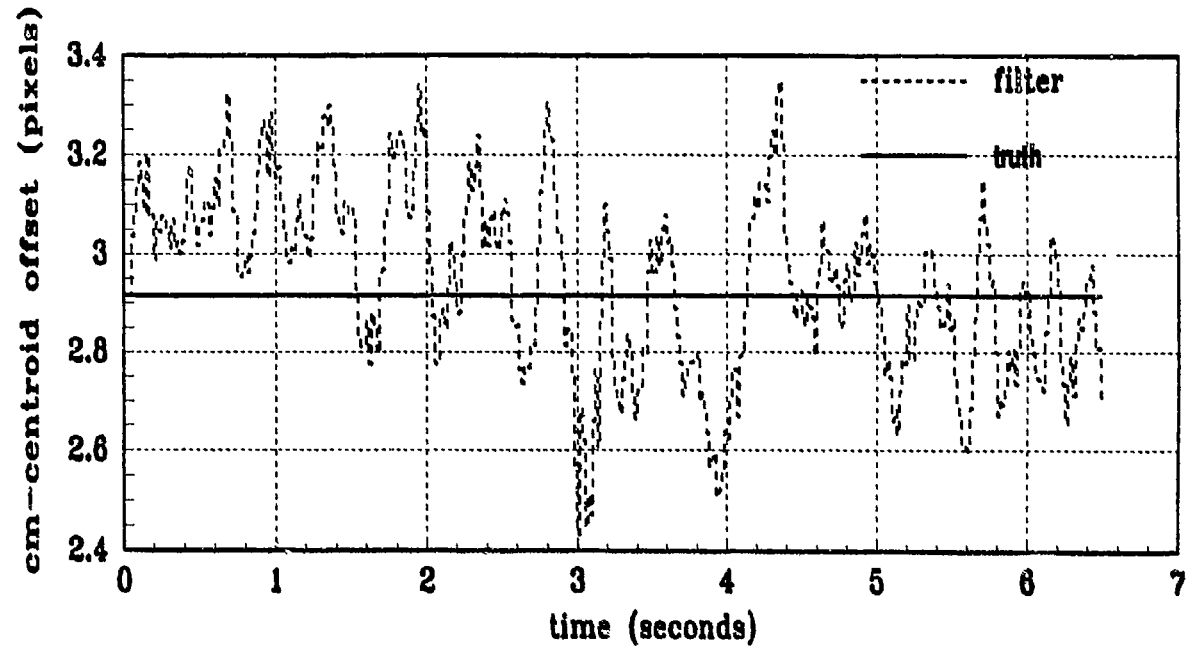


(a) FILTER vs TRUE POGO VELOCITY

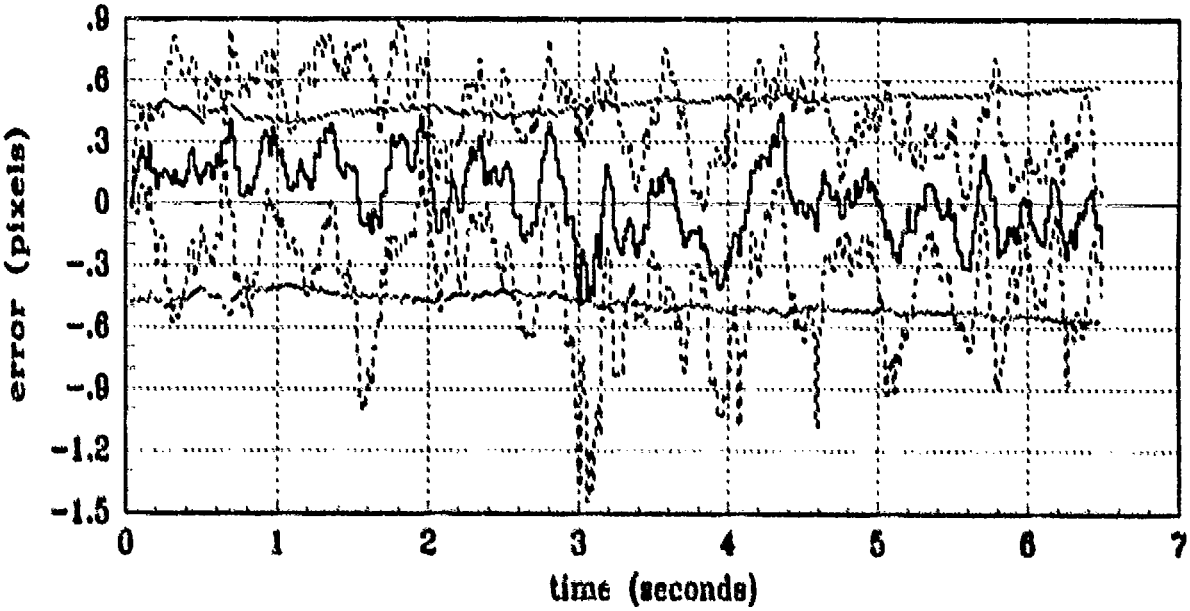


(b) FILTER vs ACTUAL ERROR (POGO VELOCITY)

Figure G.6 Pogo Velocity (Filter State 8)
(Truth = $1.12/20\pi$)

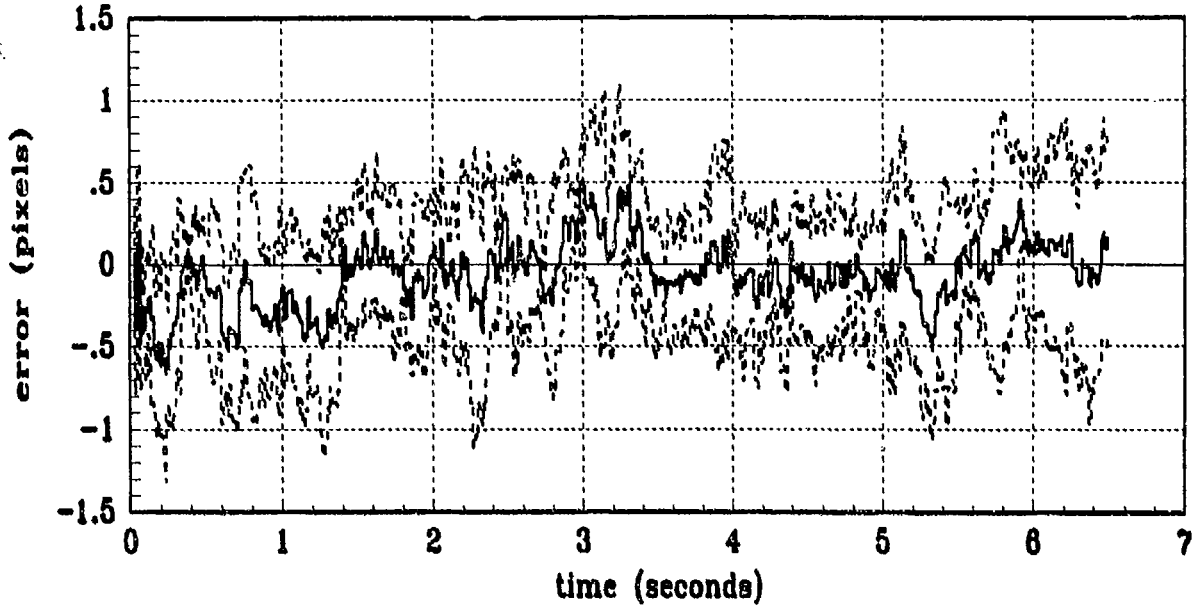


(a) FILTER vs TRUE COM-EQUILIBRIUM POINT OFFSET

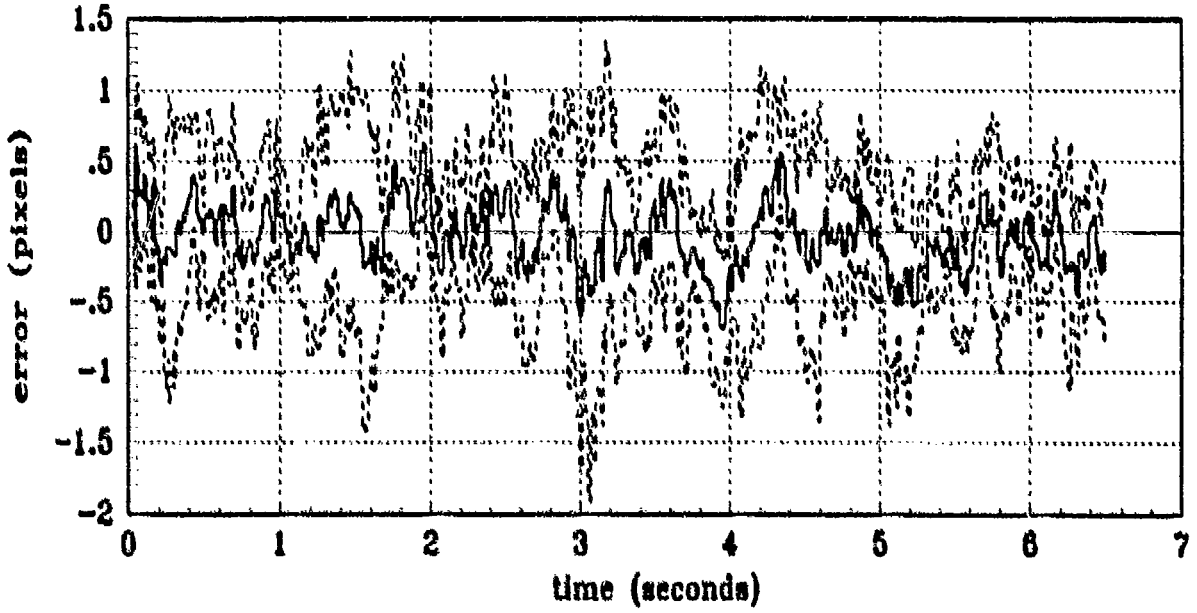


(b) FILTER vs ACTUAL ERROR (COM-EQUILIBRIUM POINT OFFSET)

Figure G.7 Center-of-Mass to Equilibrium Offset (Filter State 9)
(Truth = $0.112/20\pi$)



(a) FILTER vs ACTUAL ERROR (X-CENTROID POSITION)



(b) FILTER vs ACTUAL ERROR (Y-CENTROID POSITION)

Figure G.8 Plume Centroid Error Statistics
(Truth = $0.112/20\pi$)

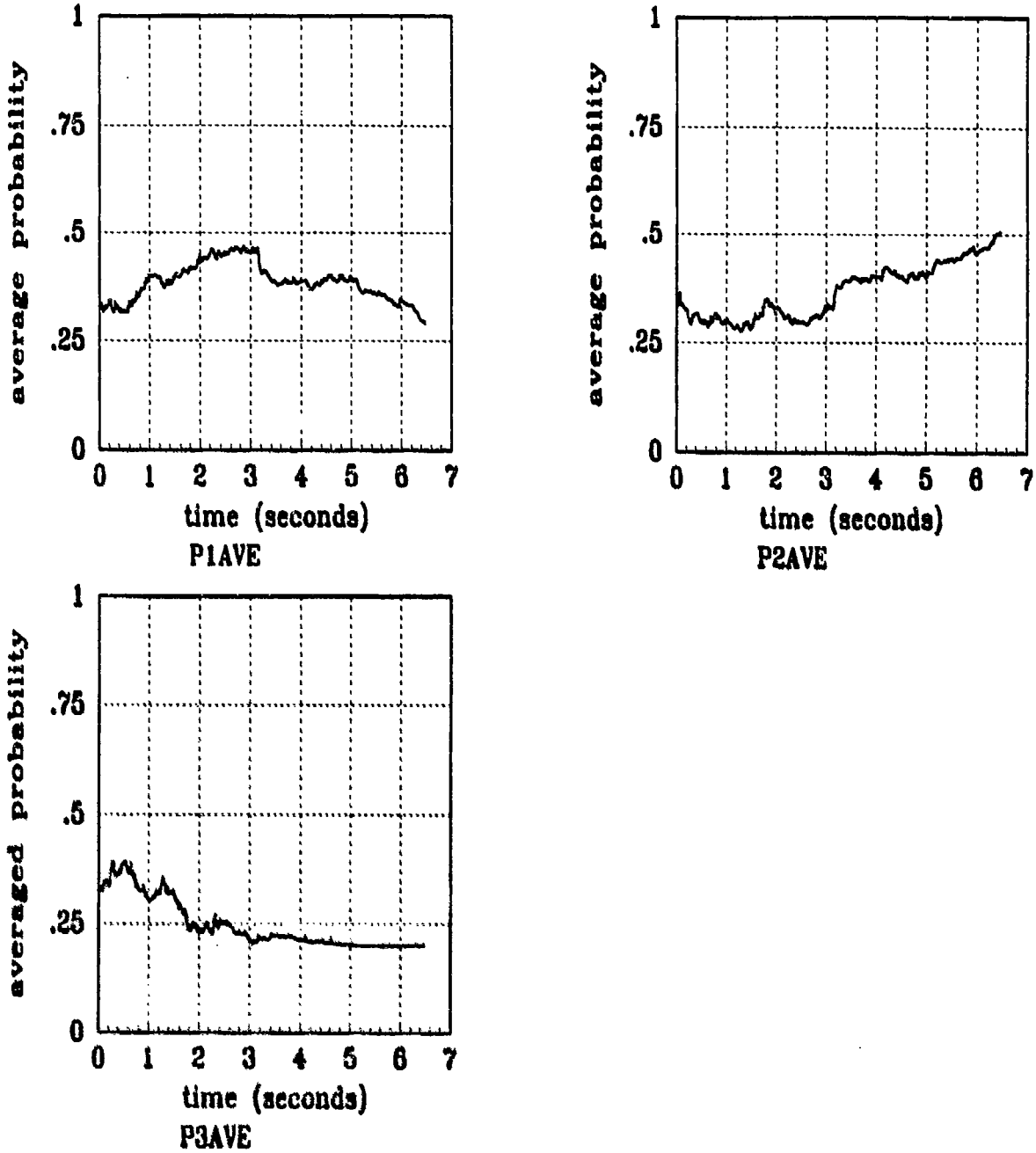


Figure G.9 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = $0.112/20\pi$)

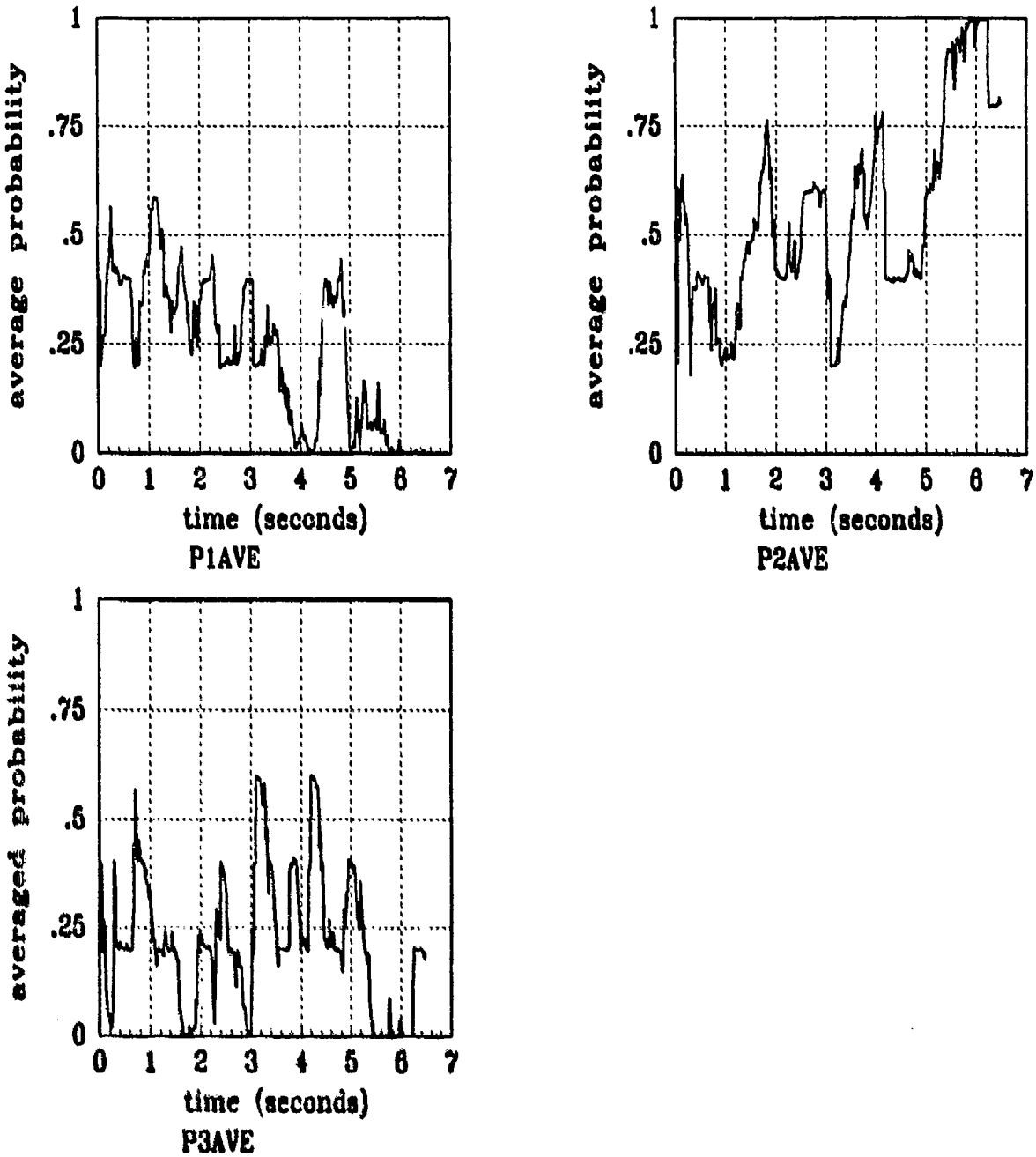


Figure G.10 Filter Residual Quantities; Computed *with* leading coefficient and $A_k(t_i)$ (Truth = $0.112/20\pi$)

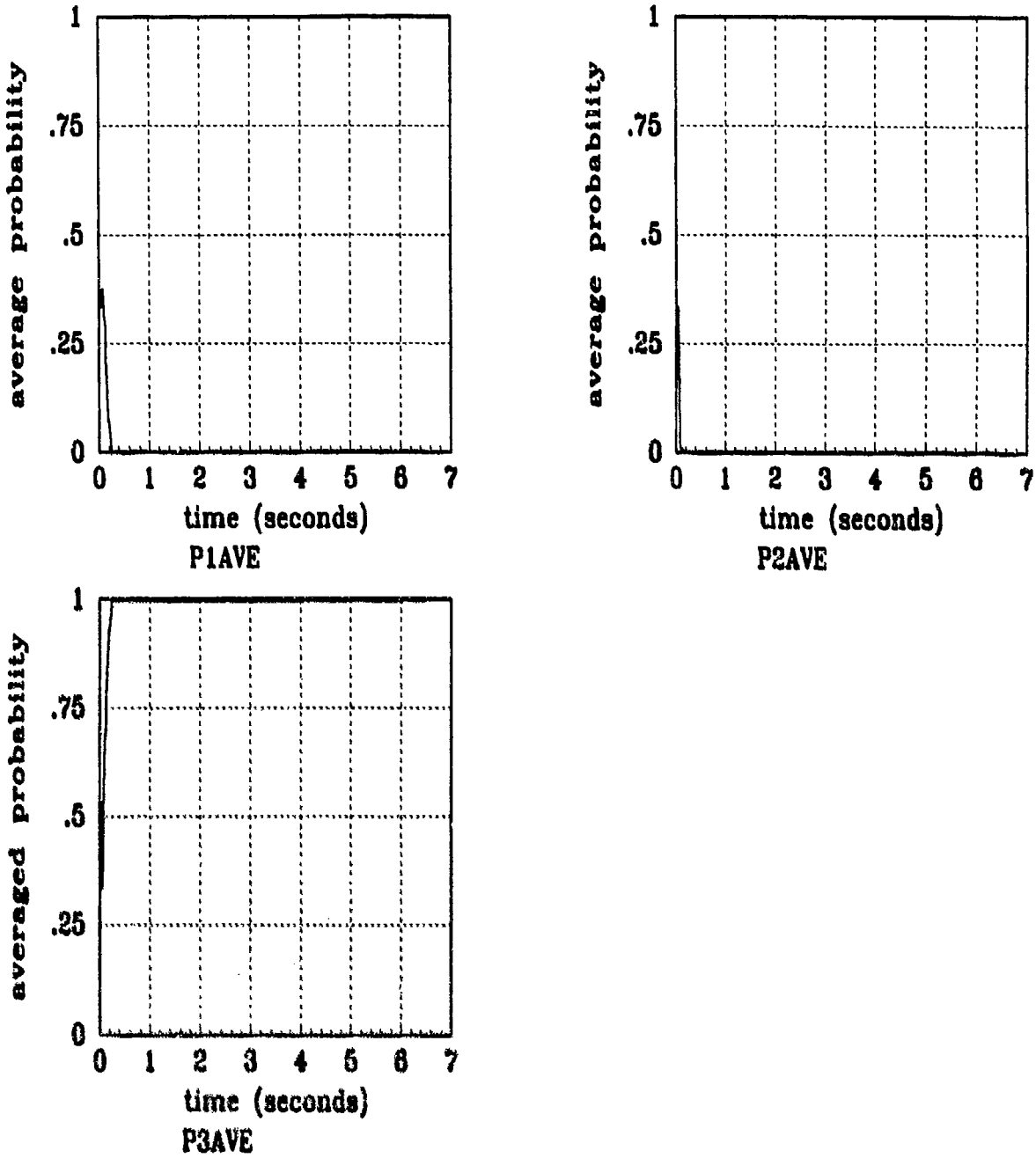
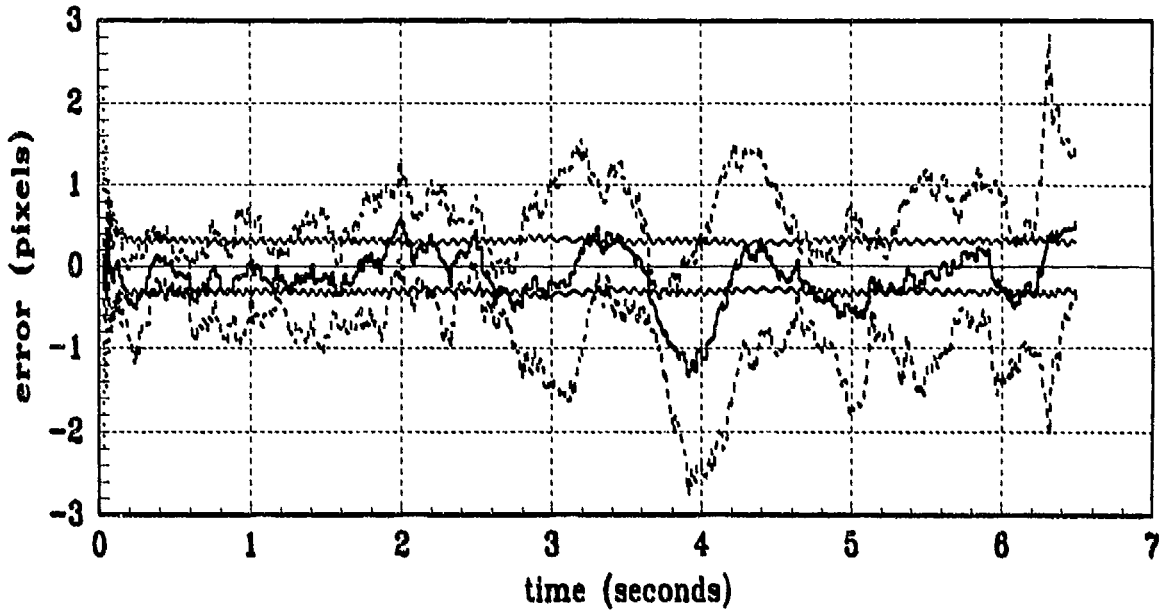
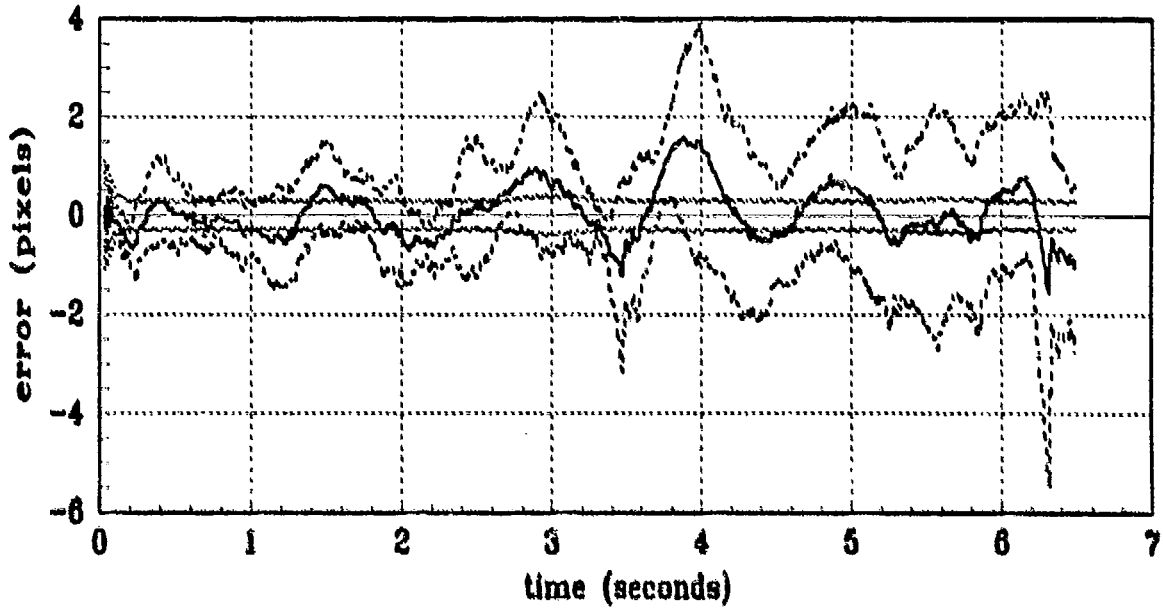


Figure G.11 Filter Residual Quantities; Computed *without* leading coefficient (Truth = $0.112/20\pi$)

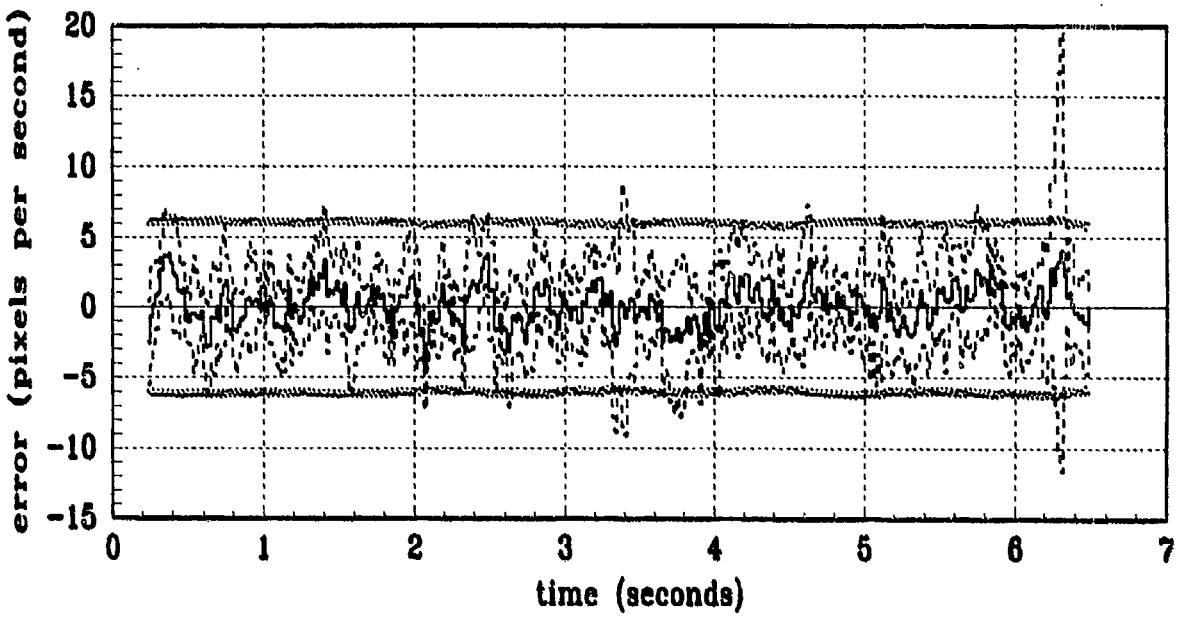


(a) FILTER vs ACTUAL ERROR (X-POSITION)

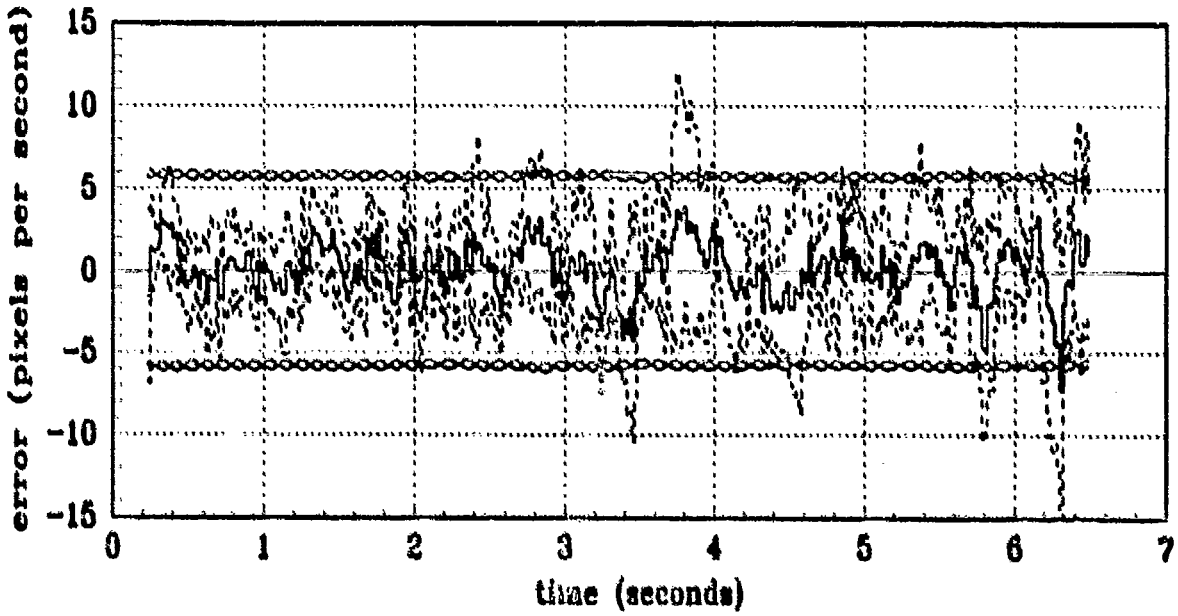


(b) FILTER vs ACTUAL ERROR (Y-POSITION)

Figure G.12 X/Y Position (Filter States 1 and 2) Error Statistics
(Truth = $1.12/2\pi$)



(a) FILTER vs ACTUAL ERROR (X-VELOCITY)



(b) FILTER vs ACTUAL ERROR (Y-VELOCITY)

Figure G.13 X/Y Velocity (Filter States 3 and 4) Error Statistics
(Truth = $1.12/2\pi$

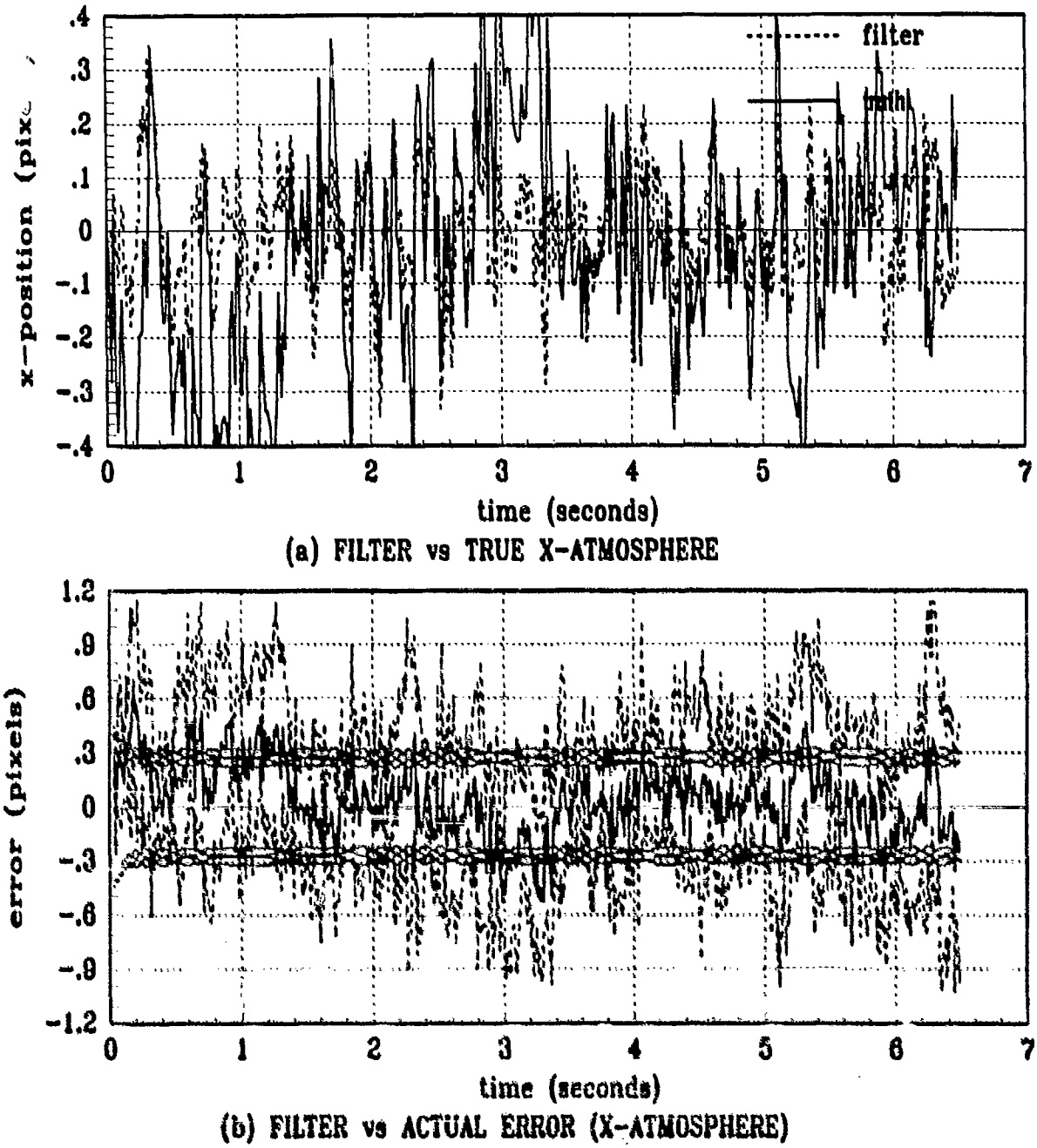


Figure G.14 X Atmospheric Jitter (Filter State 5)
(Truth = $1.12/2\pi$)

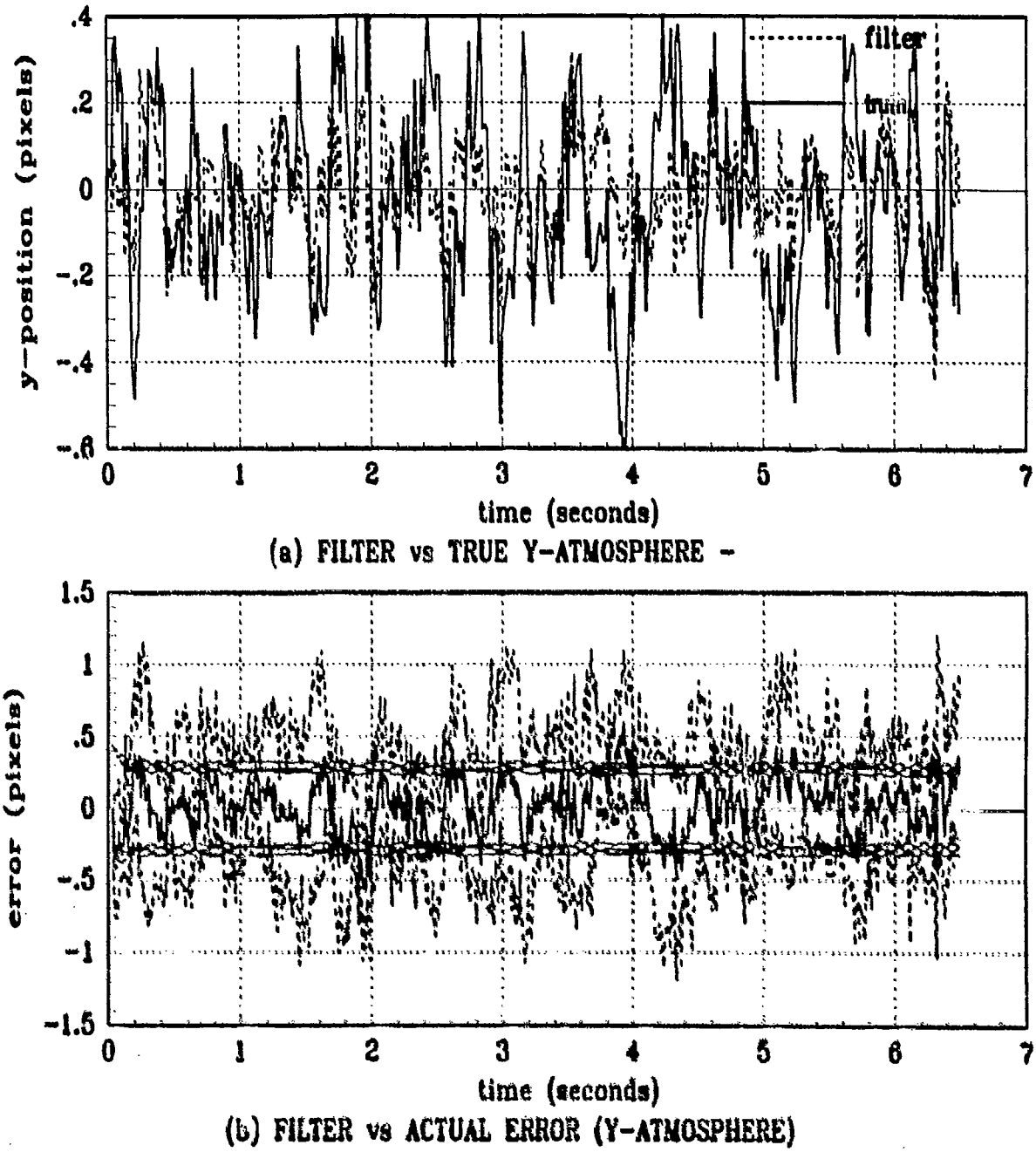
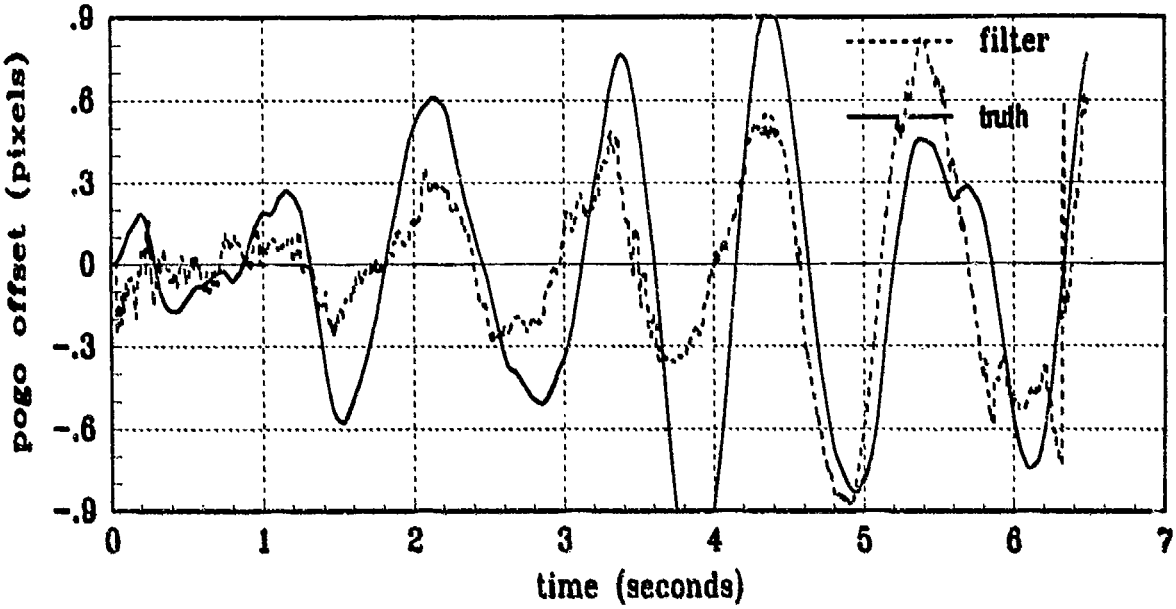
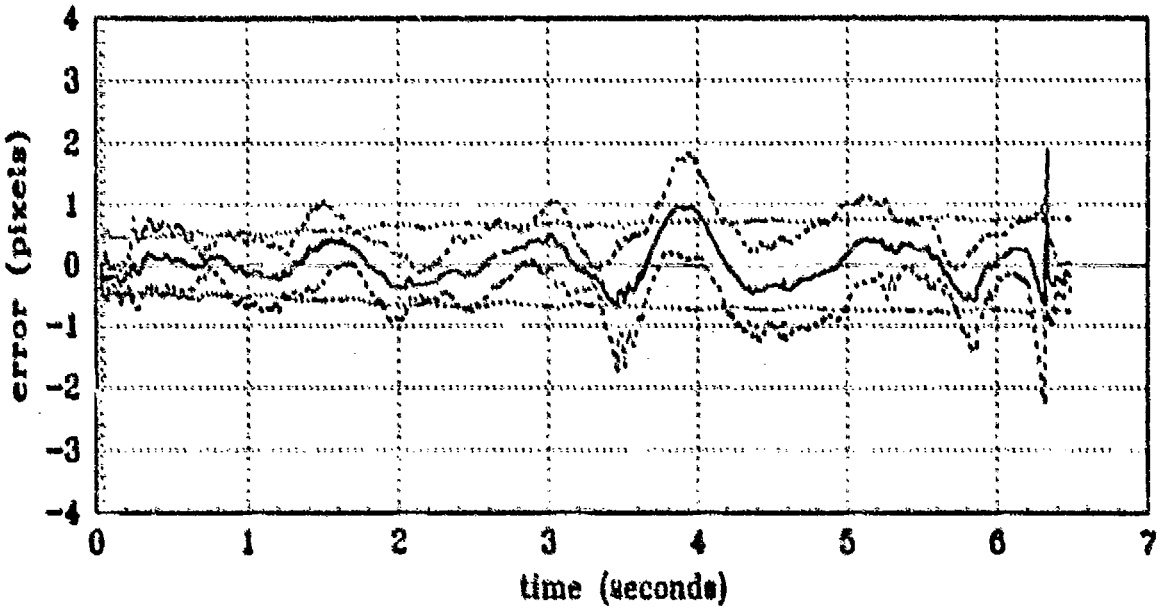


Figure G.15 Y Atmospheric Jitter (Filter State 6)
(Truth = 1.12/2π)

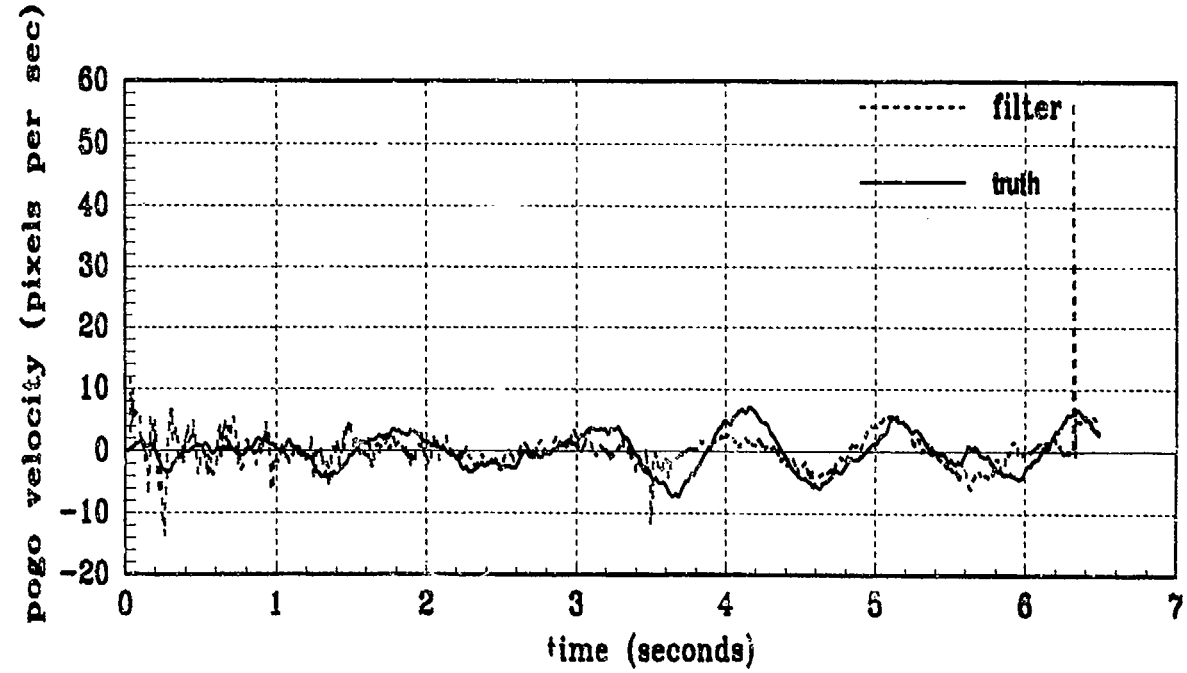


(a) FILTER vs TRUE POGO

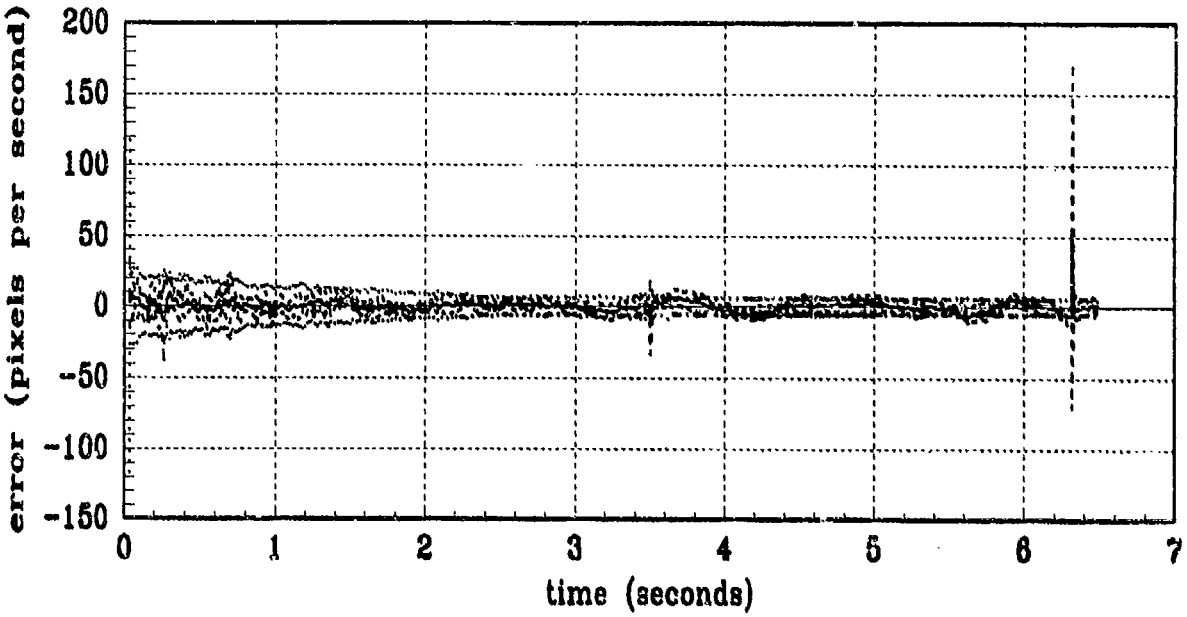


(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure G.16 Pogo Position Offset (Filter State 7)
(Truth = 1.12/2π)

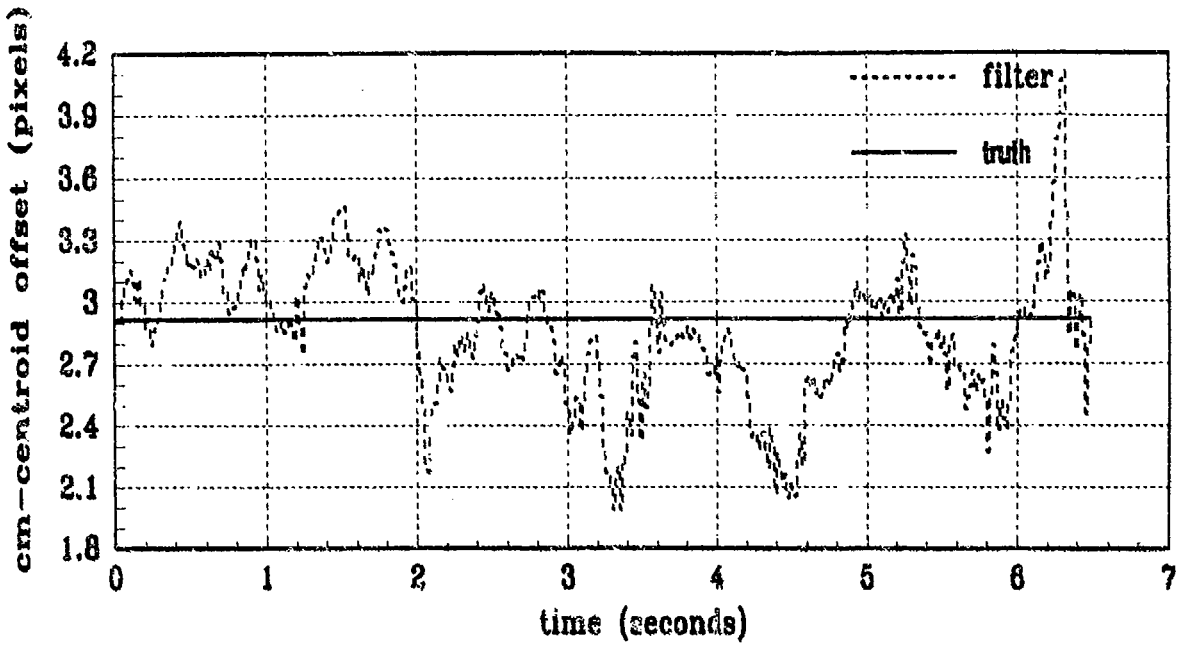


(a) FILTER vs TRUE POGO VELOCITY

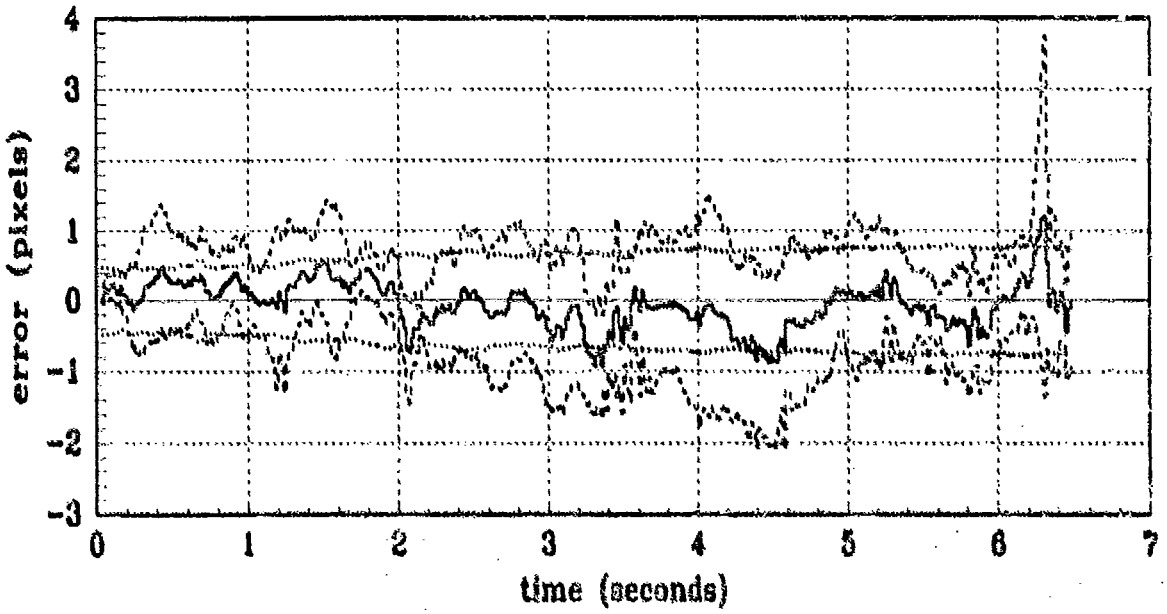


(b) FILTER vs ACTUAL ERROR (POGO VELOCITY)

Figure G.17 Pogo Velocity (Filter State 8)
(Truth = $1.12/2\pi$)

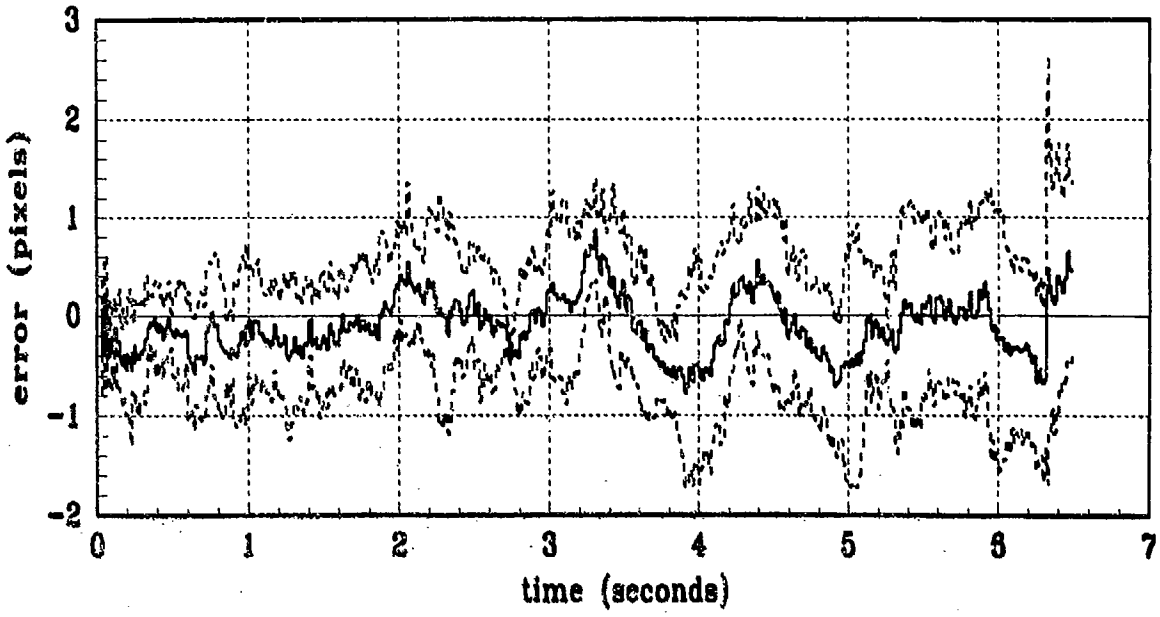


(a) FILTER vs TRUE COM-EQUILIBRIUM POINT OFFSET

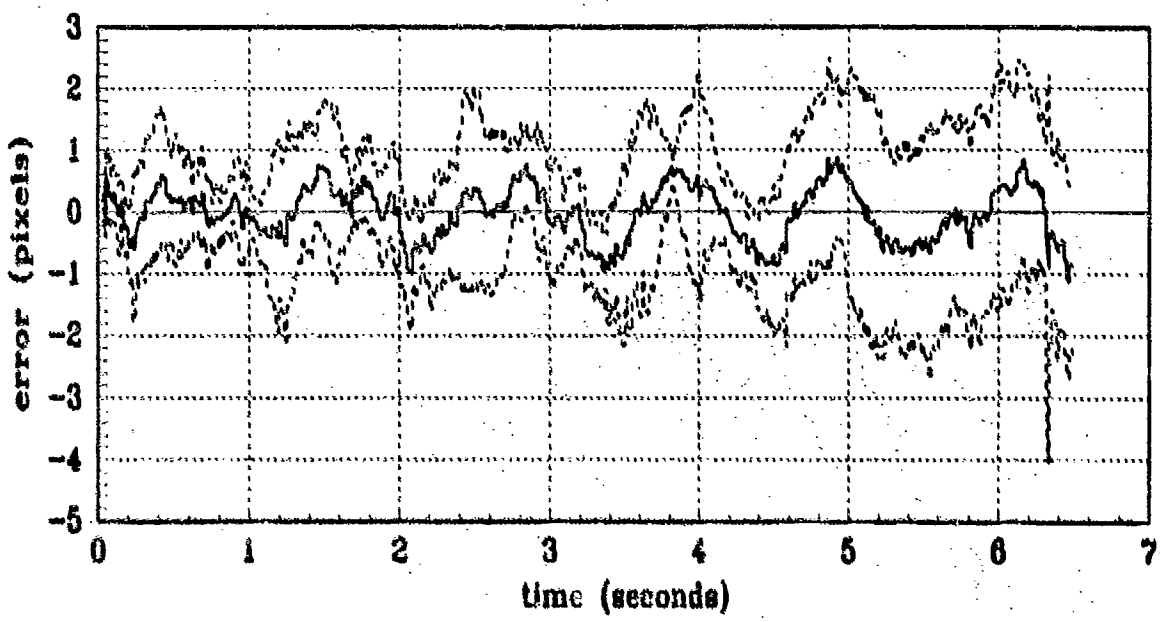


(b) FILTER vs ACTUAL ERROR (COM-EQUILIBRIUM POINT OFFSET)

Figure G.18 Center-of-Mass to Equilibrium Offset (Filter State 9)
(Truth = $1.12/2\pi$)



(a) FILTER vs ACTUAL ERROR (X-CENTROID POSITION)



(b) FILTER vs ACTUAL ERROR (Y-CENTROID POSITION)

Figure G.19 Plume Centroid Error Statistics
(Truth = $1.12/2\pi$)

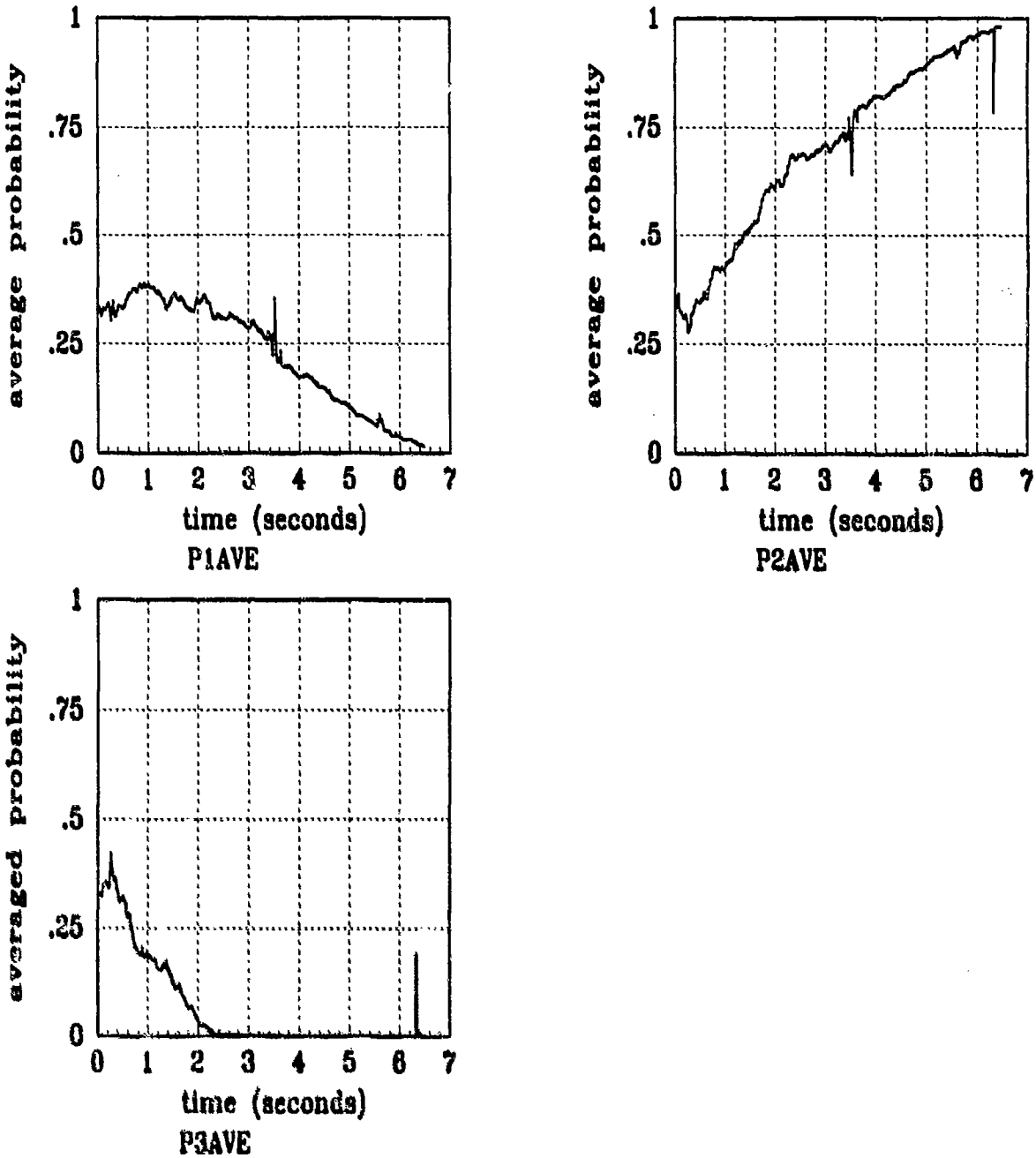


Figure G.20 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = $1.12/2\pi$)

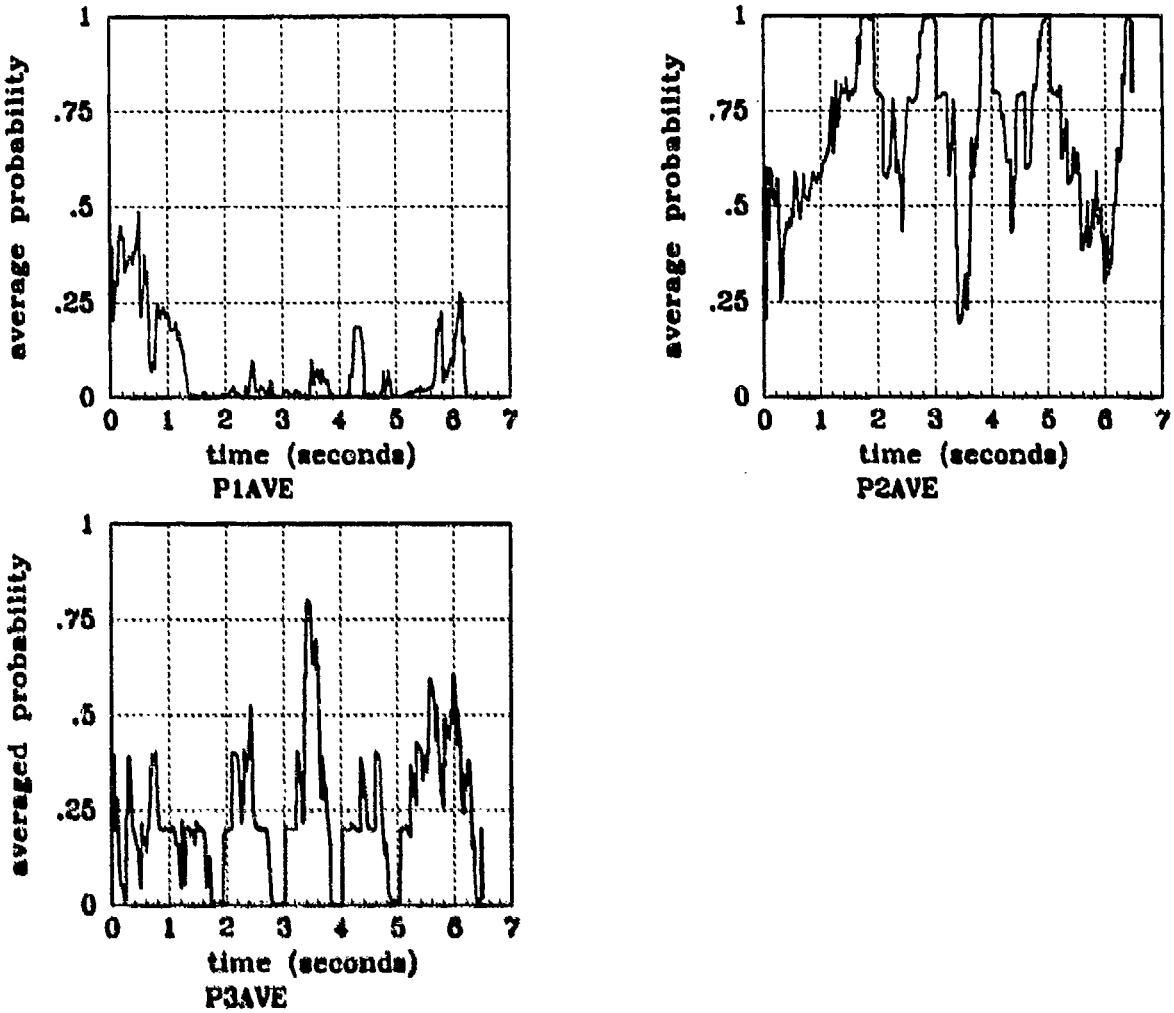


Figure G.21 Filter Residual Quantities; Computed *with* leading coefficient and $A_1(t_i)$ (Truth = 1.12/2π)

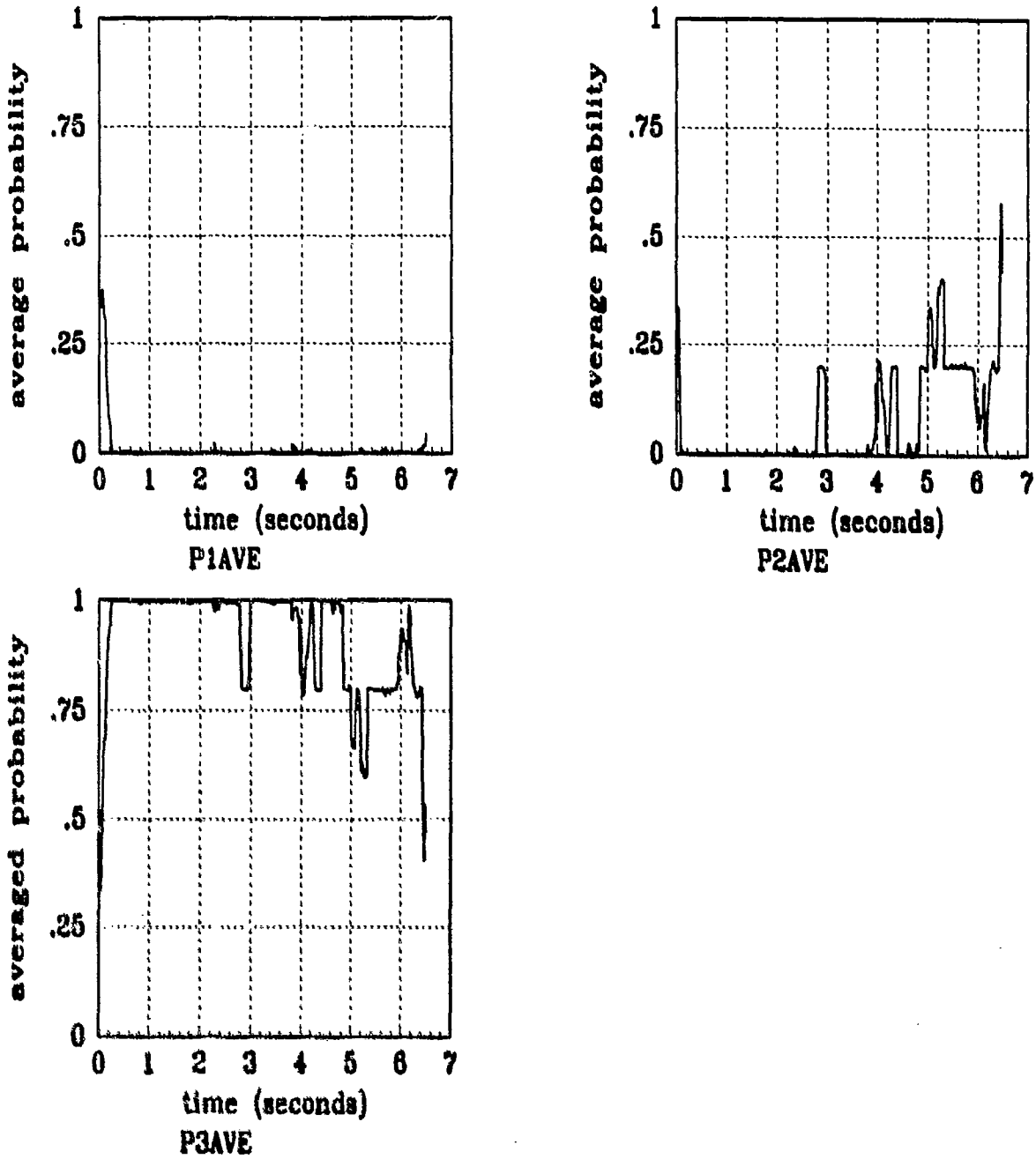
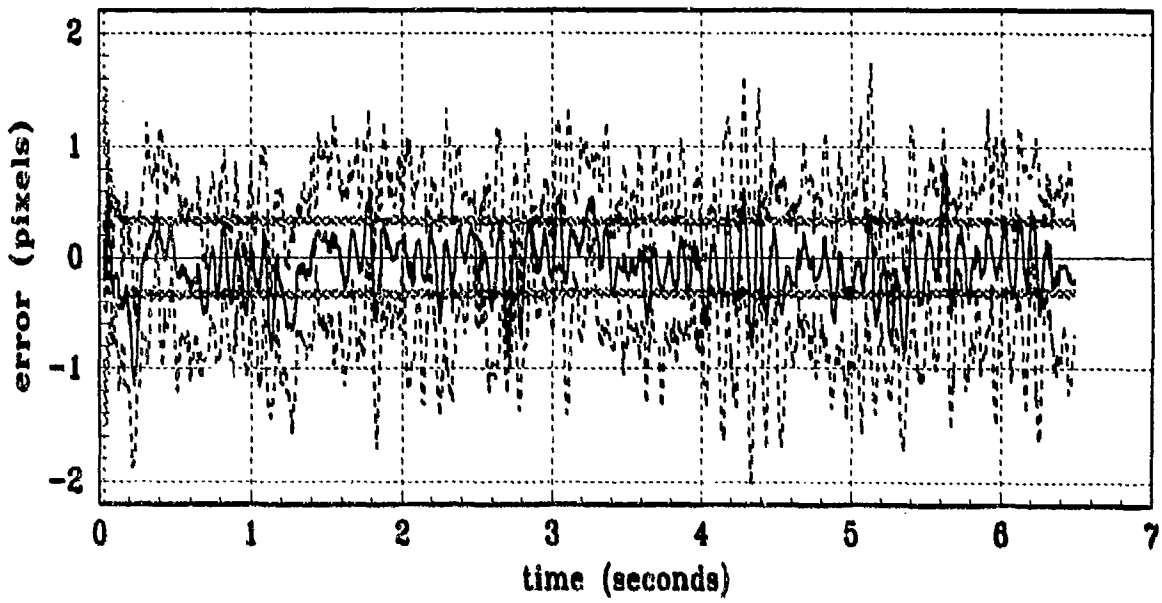
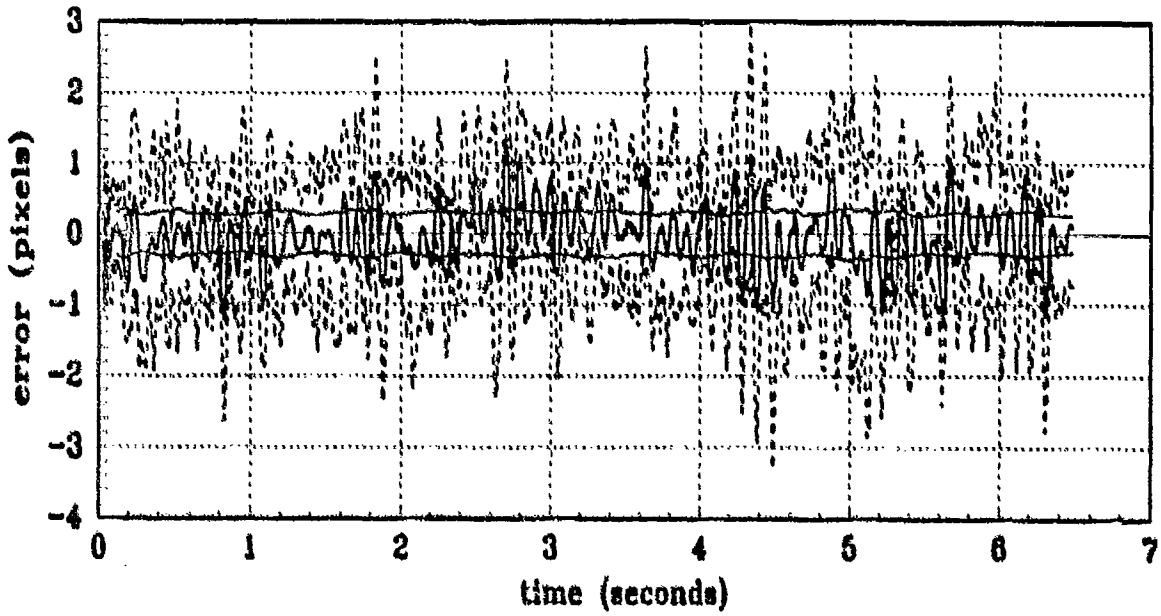


Figure G.22 Filter Residual Quantities; Computed *without* leading coefficient (Truth = $1.12/2\pi$)

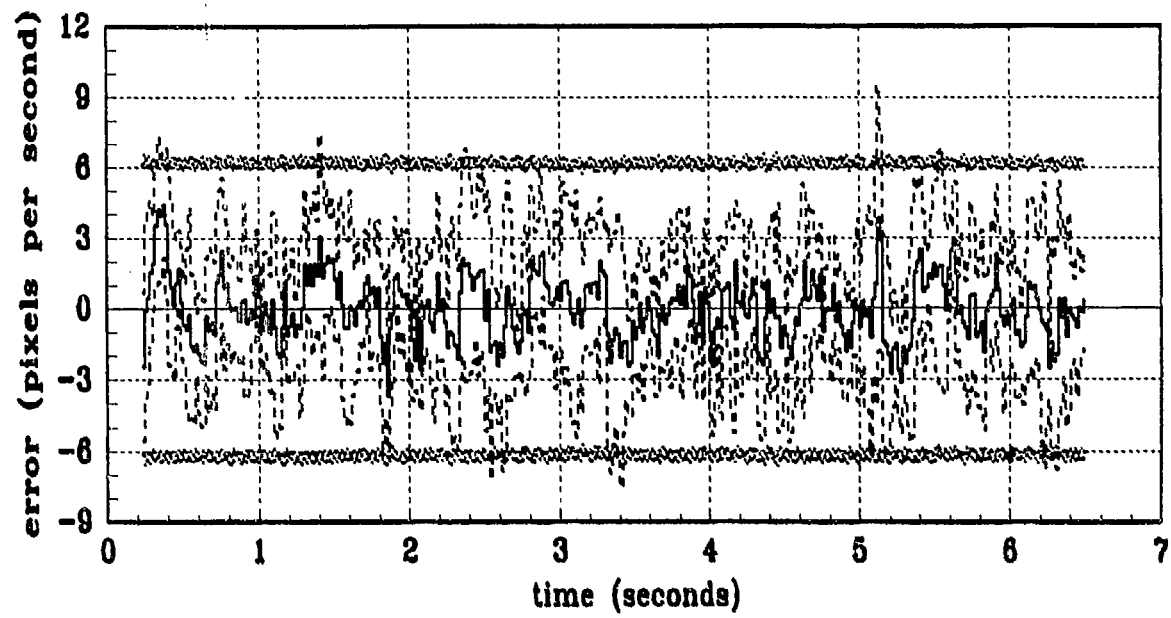


(a) FILTER vs ACTUAL ERROR (X-POSITION)

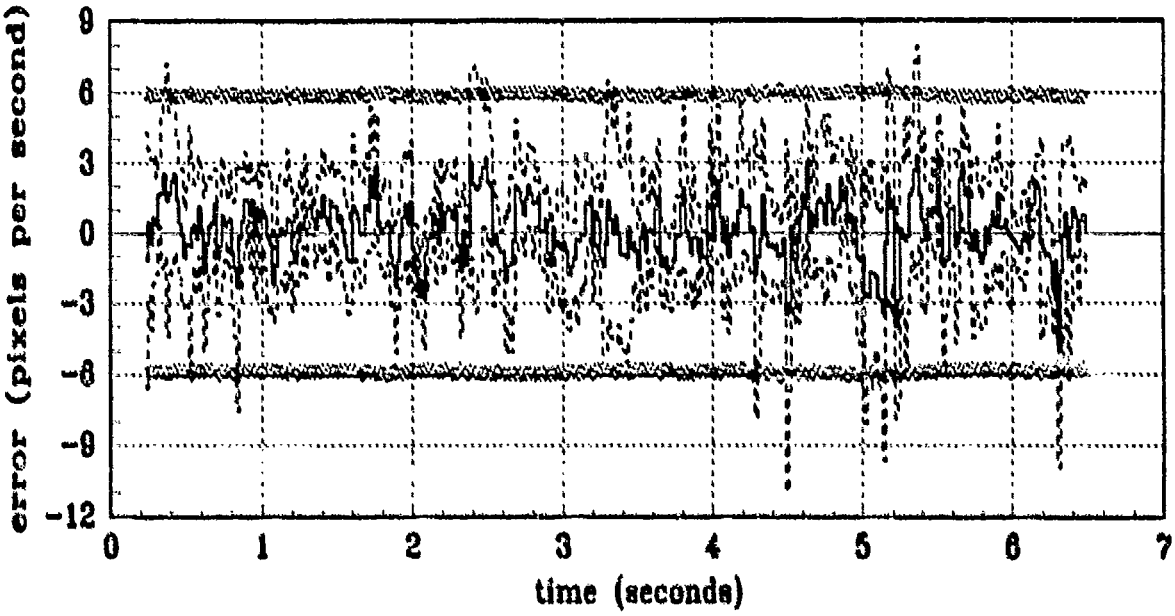


(b) FILTER vs ACTUAL ERROR (Y-POSITION)

Figure G.23 X/Y Position (Filter States 1 and 2) Error Statistics
(Truth = $1.12/20\pi$)

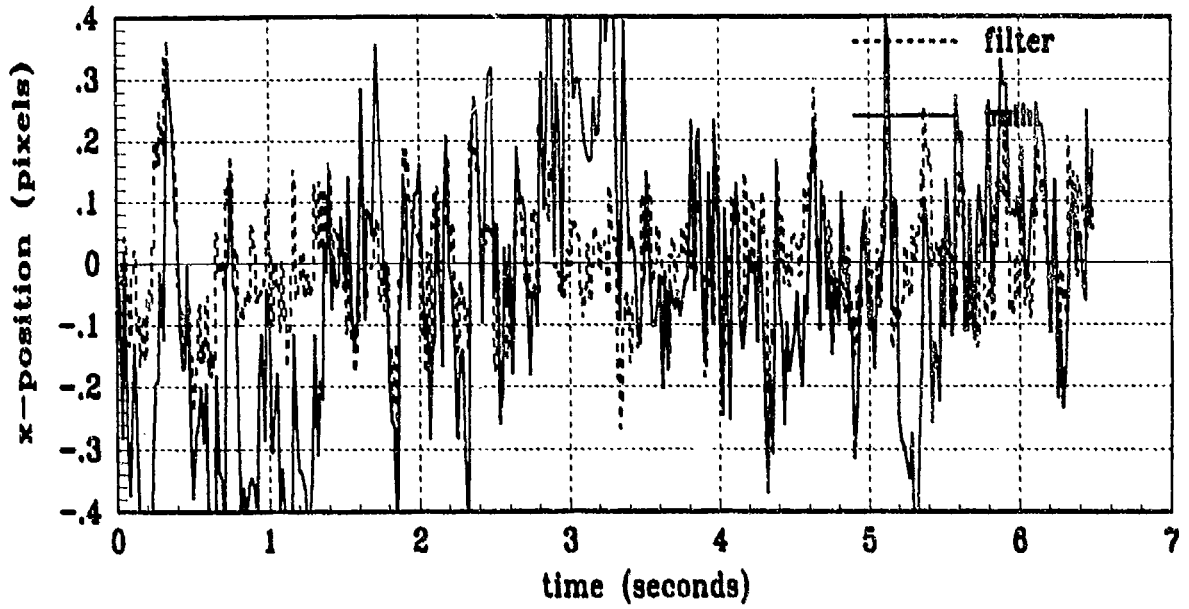


(a) FILTER vs ACTUAL ERROR (X-VELOCITY)

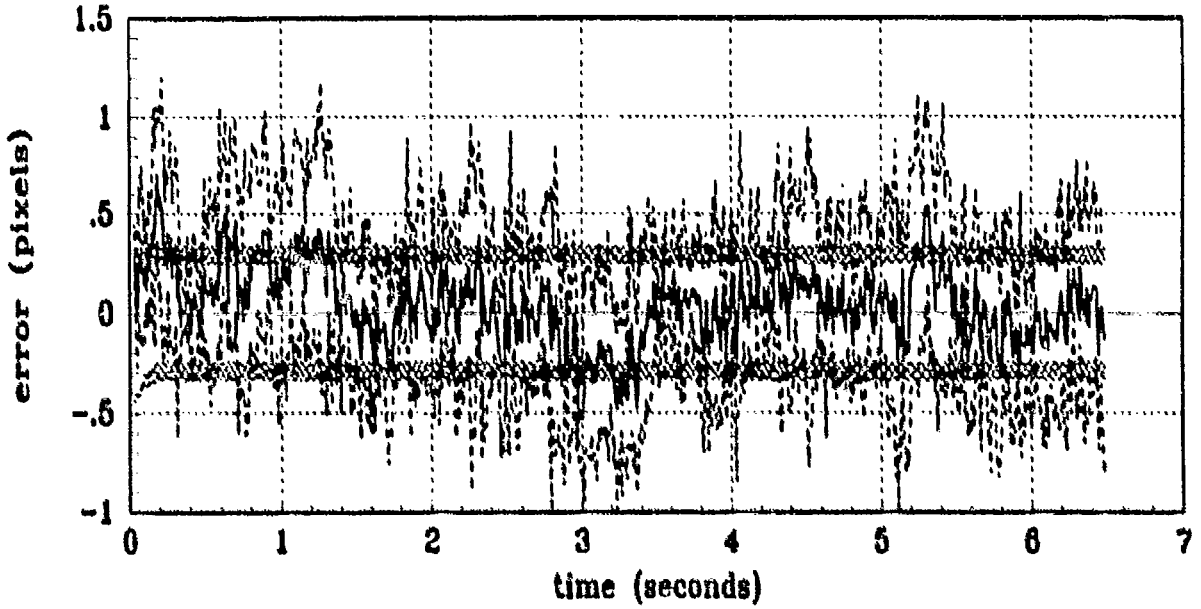


(b) FILTER vs ACTUAL ERROR (Y-VELOCITY)

Figure G.24 X/Y Velocity (Filter States 3 and 4) Error Statistics
(Truth = $1.12/20\pi$)

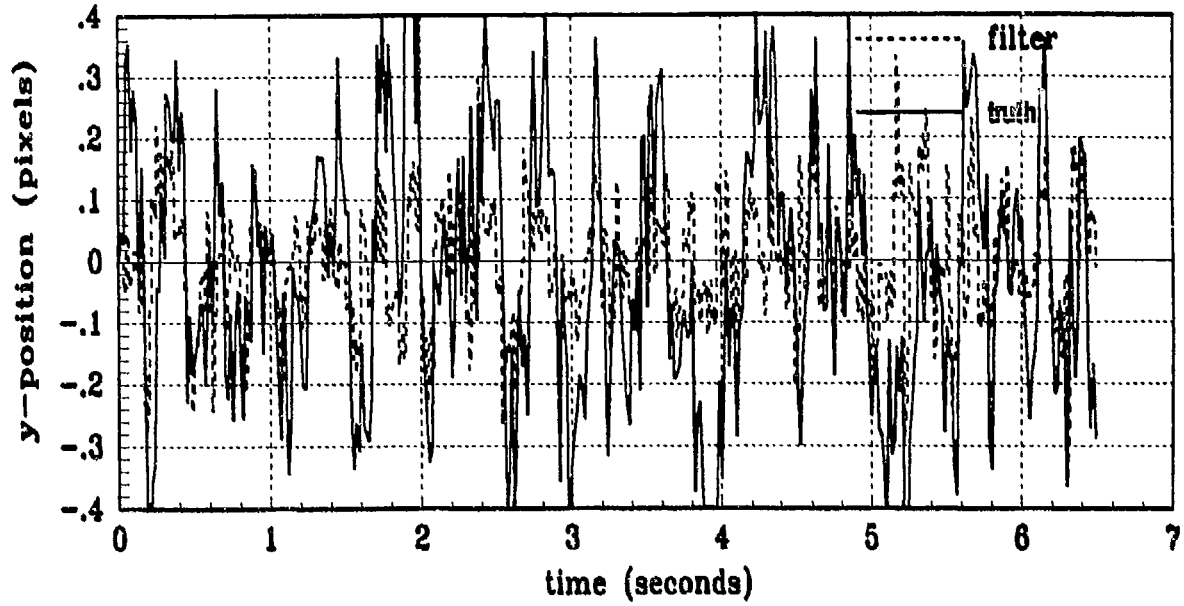


(a) FILTER vs TRUE X-ATMOSPHERE

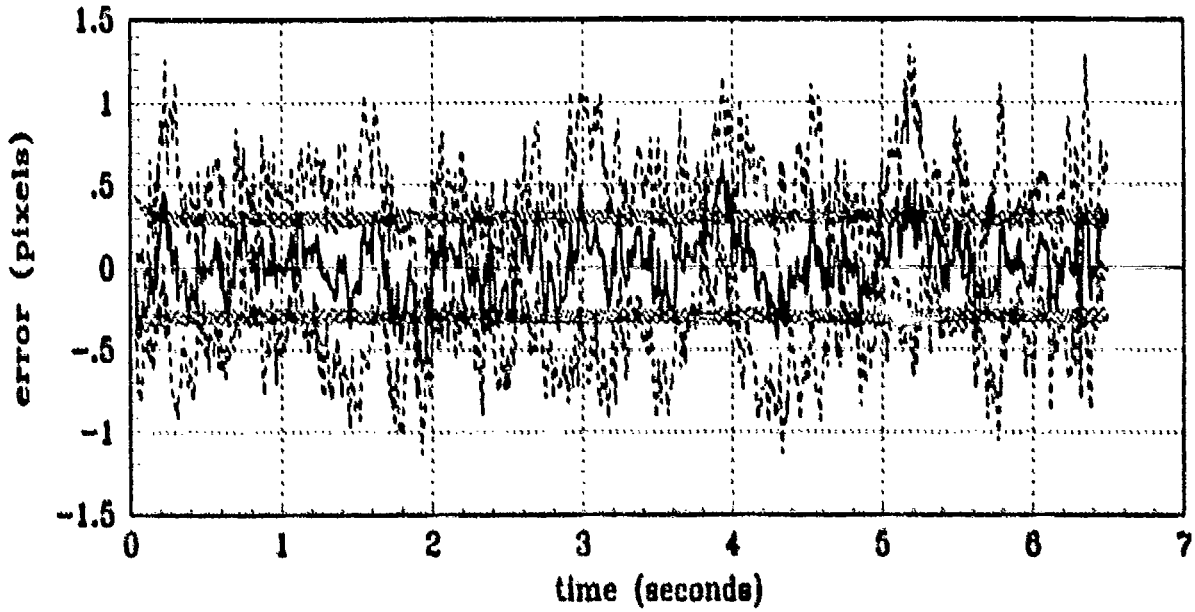


(b) FILTER vs ACTUAL ERROR (X-ATMOSPHERE)

Figure G.25 X Atmospheric Jitter (Filter State 5)
(Truth = $1.12/20\pi$)

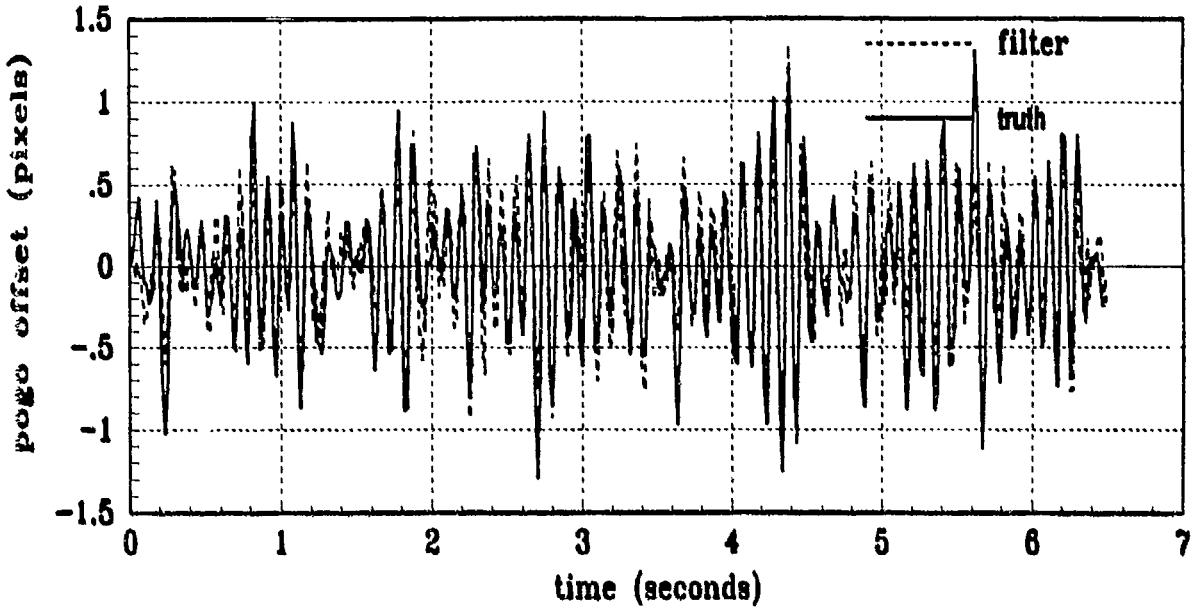


(a) FILTER vs TRUE Y-ATMOSPHERE -

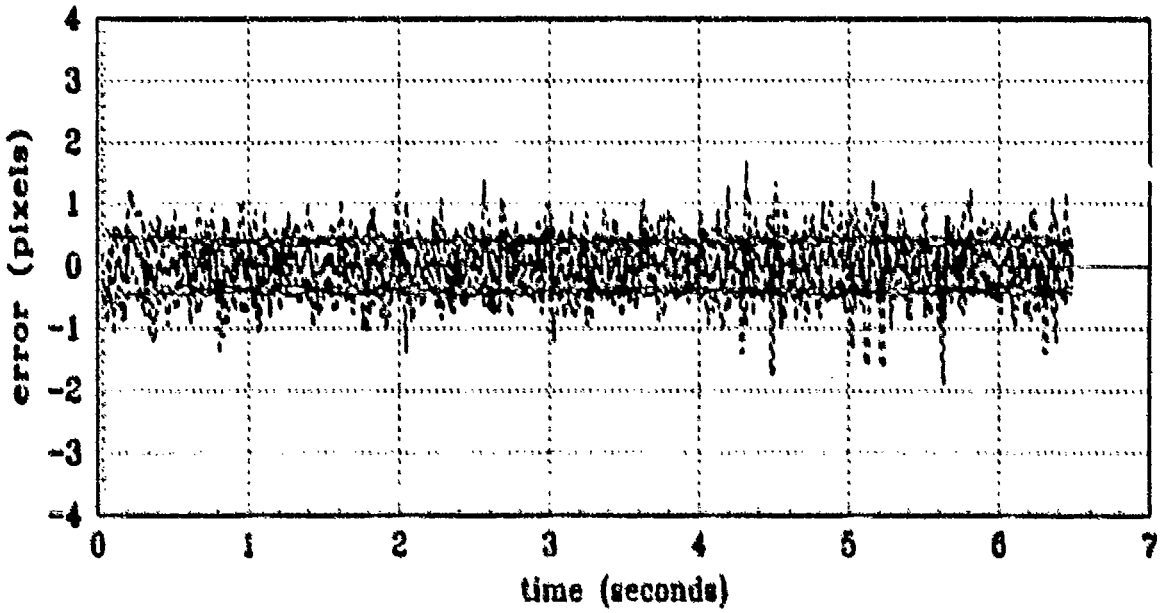


(b) FILTER vs ACTUAL ERROR (Y-ATMOSPHERE)

Figure G.26 Y Atmospheric Jitter (Filter State 6)
(Truth = 1.12/20π)

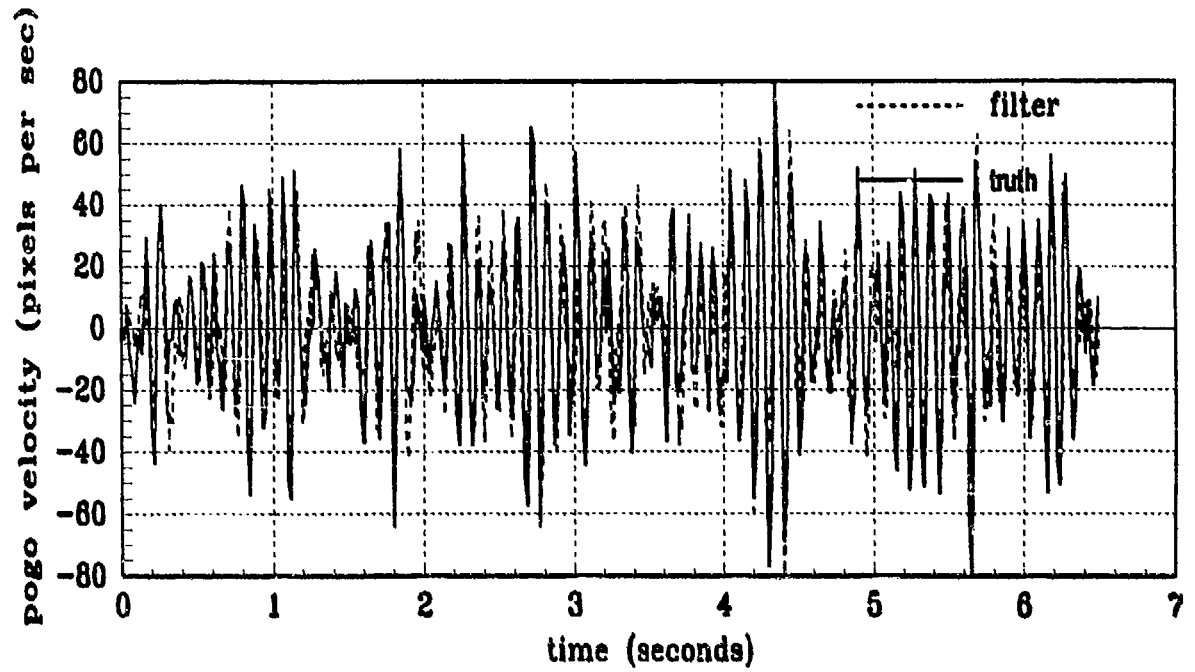


(a) FILTER vs TRUE POGO

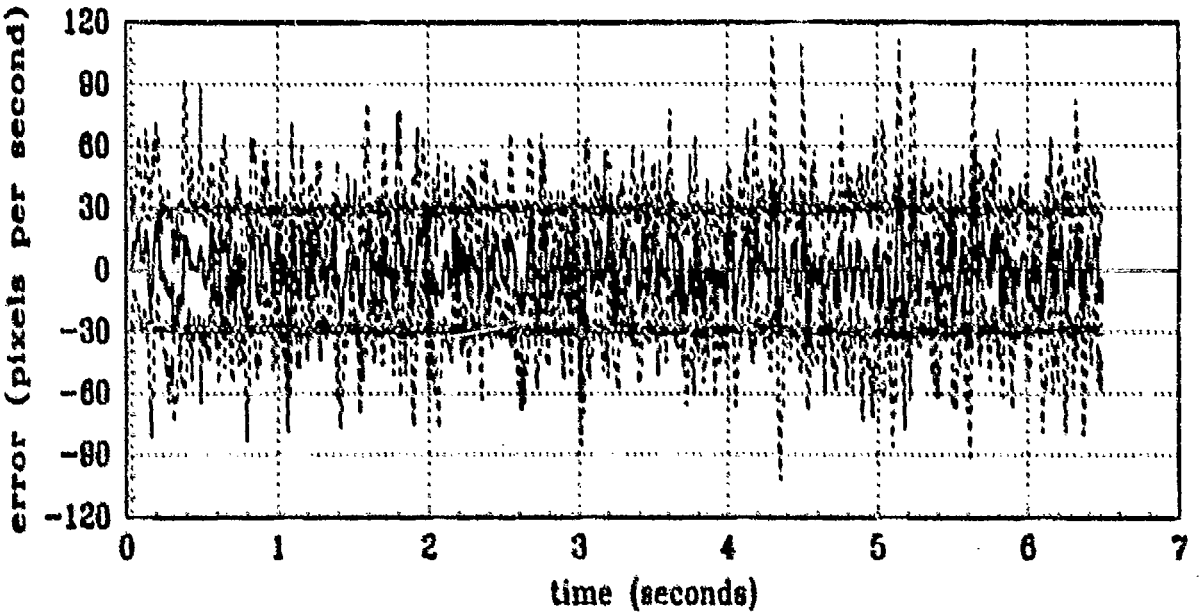


(b) FILTER vs ACTUAL ERROR (POGO POSITION)

Figure G.27 Pogo Position Offset (Filter State 7)
(Truth = 1.12/20π)

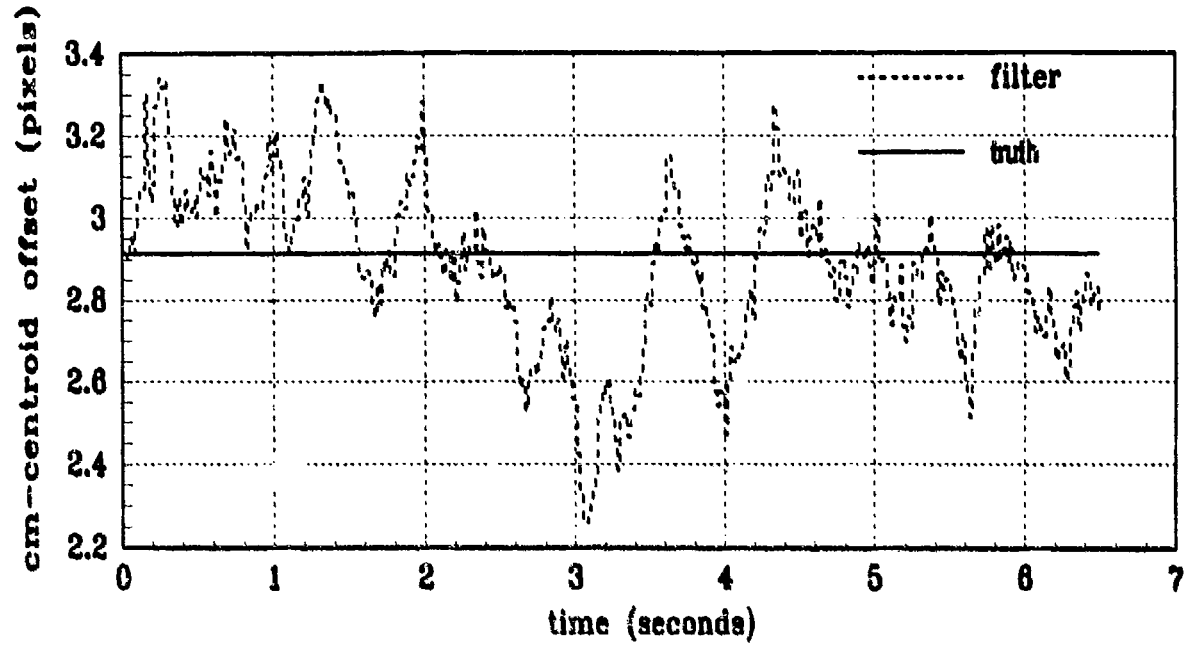


(a) FILTER vs TRUE POGO VELOCITY

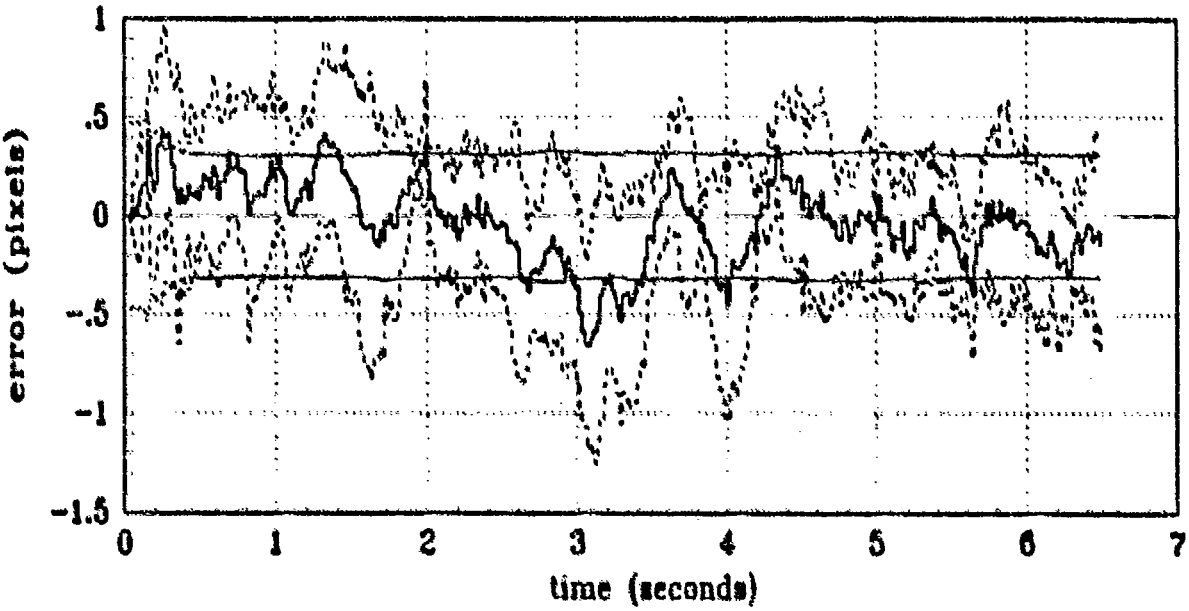


(b) FILTER vs ACTUAL ERROR (POGO VELOCITY)

Figure G.28 Pogo Velocity (Filter State 8)
(Truth = $1.12/20\pi$)

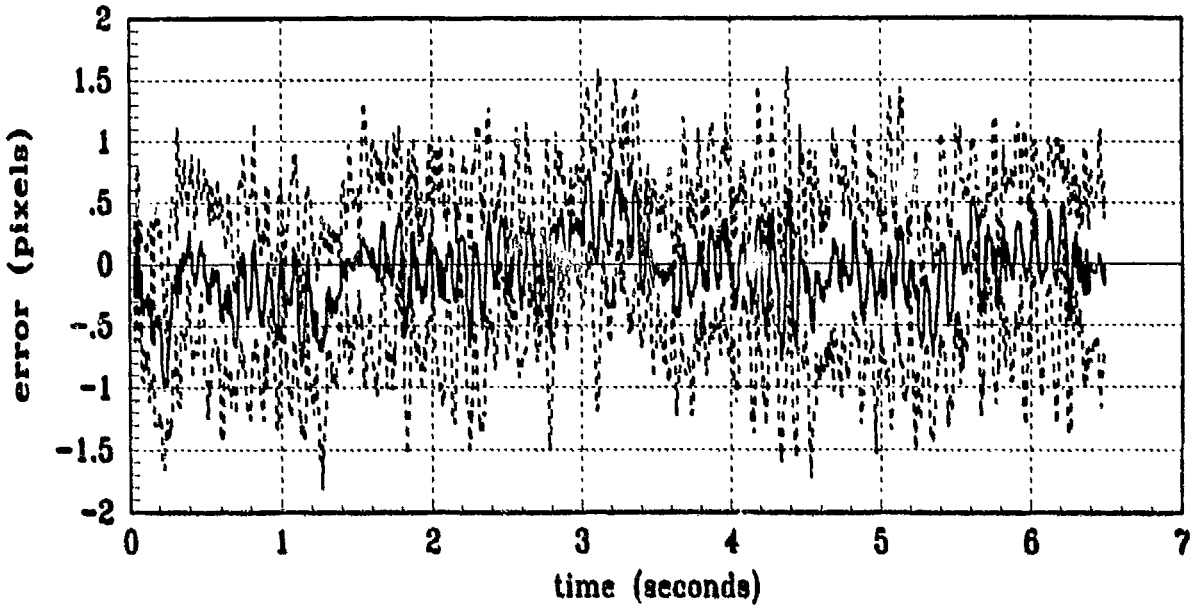


(a) FILTER vs TRUE COM-EQUILIBRIUM POINT OFFSET

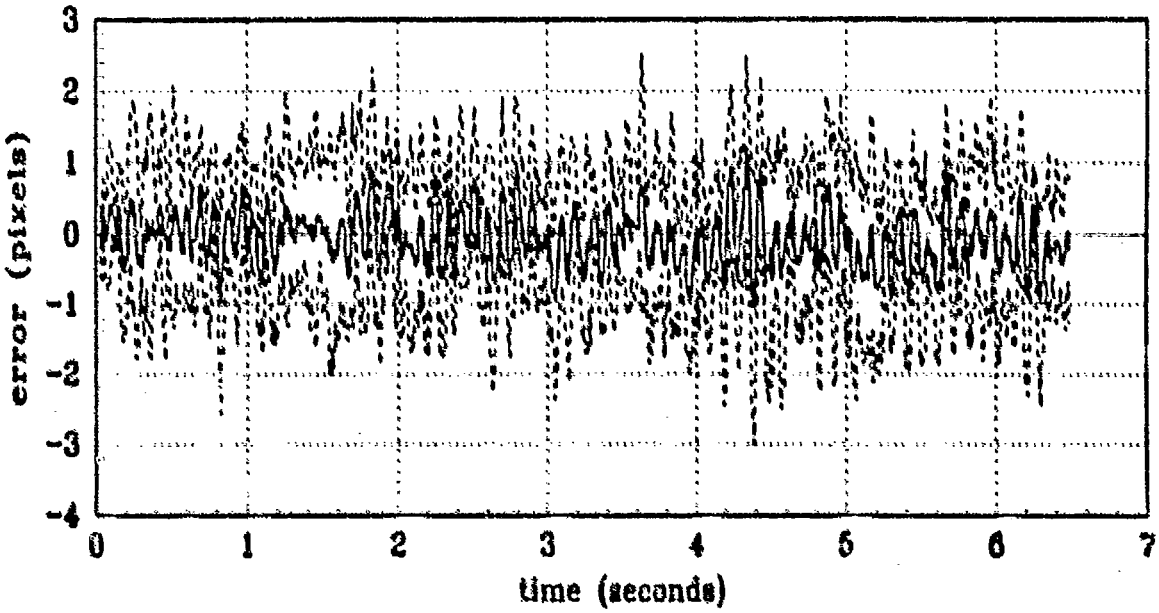


(b) FILTER vs ACTUAL ERROR (COM-EQUILIBRIUM POINT OFFSET)

Figure G.29 Center-of-Mass to Equilibrium Offset (Filter State 9)
(Truth = $1.12/20\pi$)



(a) FILTER vs ACTUAL ERROR (X-CENTROID POSITION)



(b) FILTER vs ACTUAL ERROR (Y-CENTROID POSITION)

Figure G.30 Plume Centroid Error Statistics
(Truth = 1.12/20π)

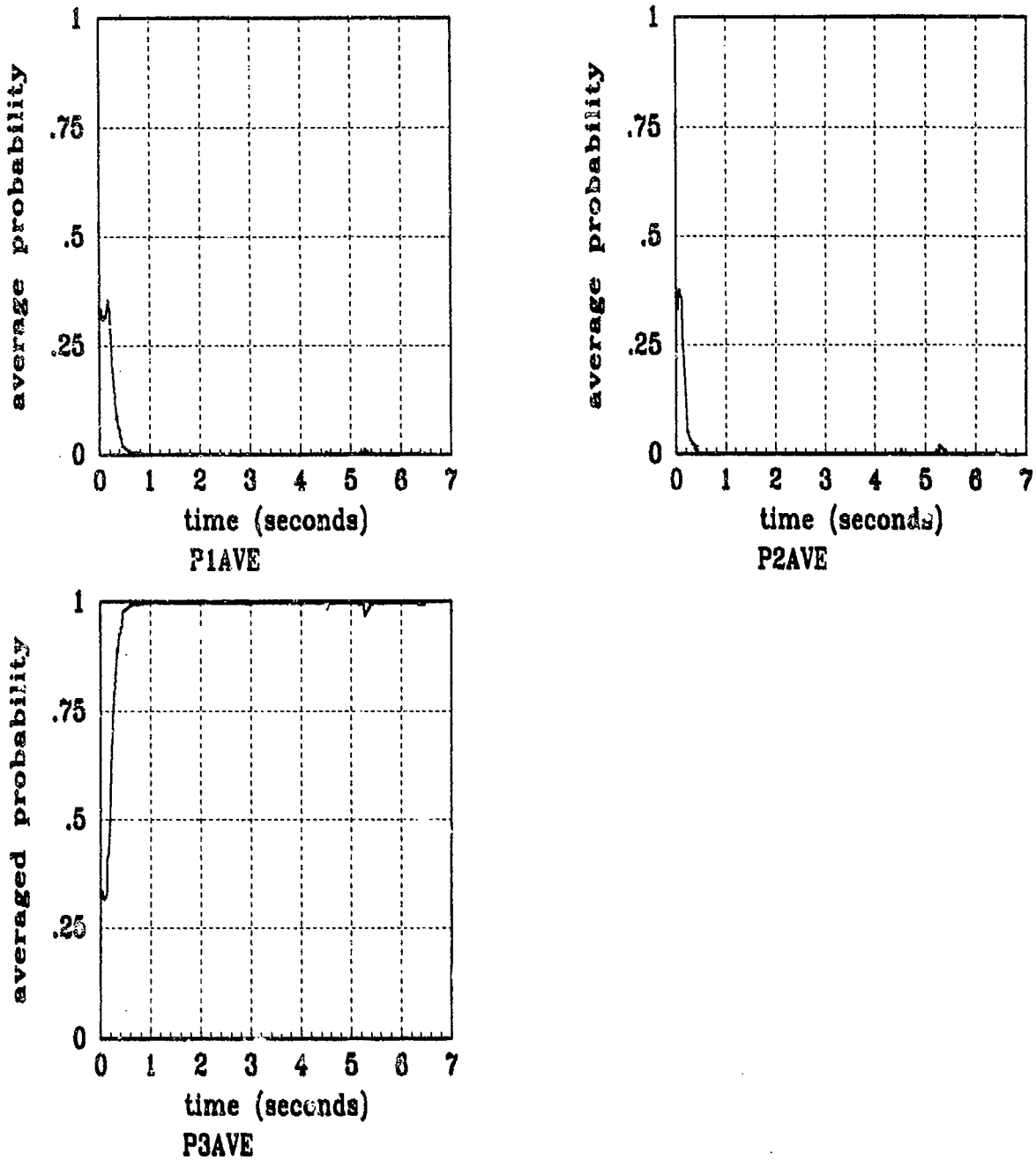


Figure G.31 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = $1.12/20\pi$)

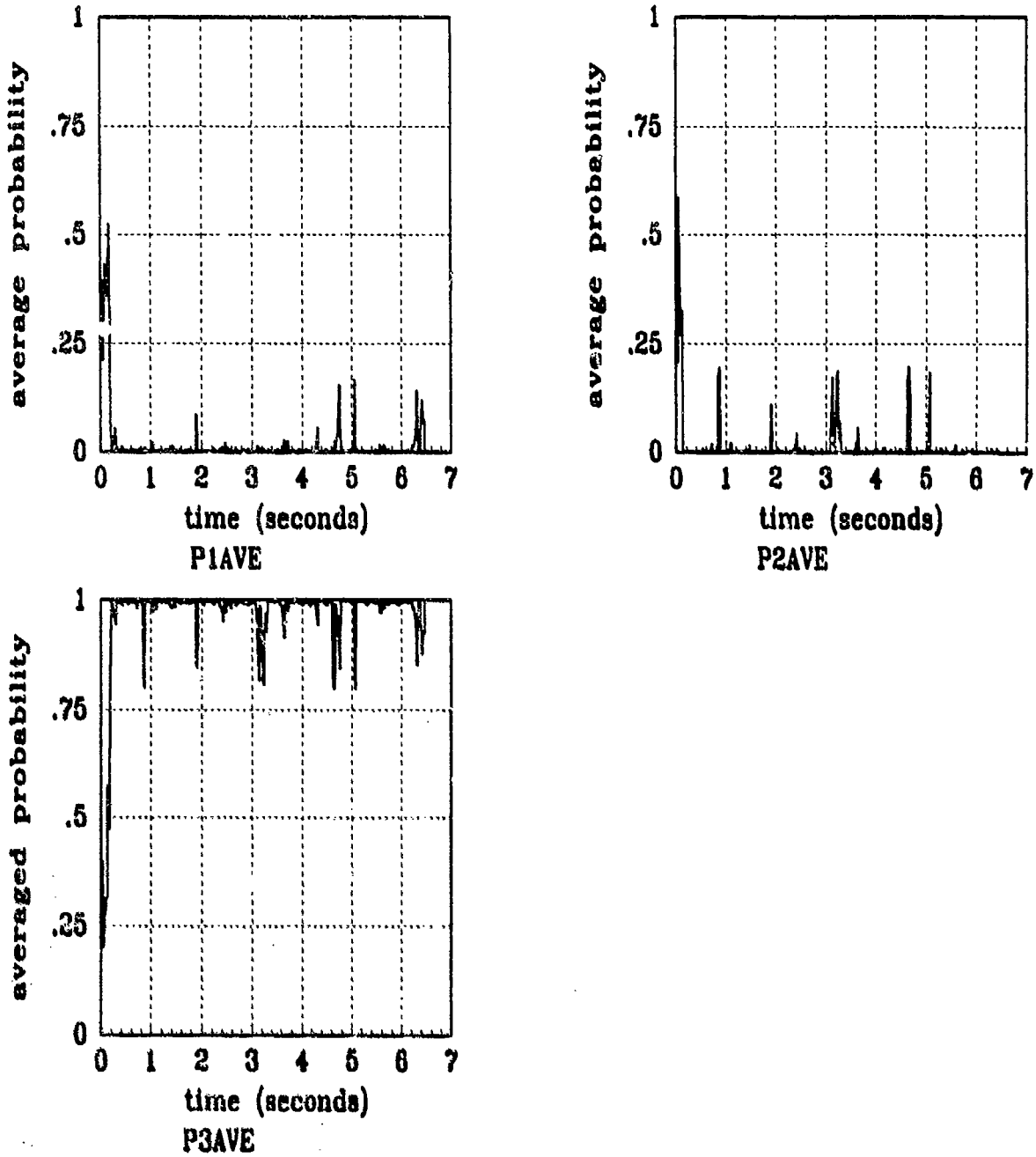


Figure G.32 Filter Residual Quantities; Computed *with* leading coefficient and $A_k(t_i)$ (Truth = $1.12/20\pi$)

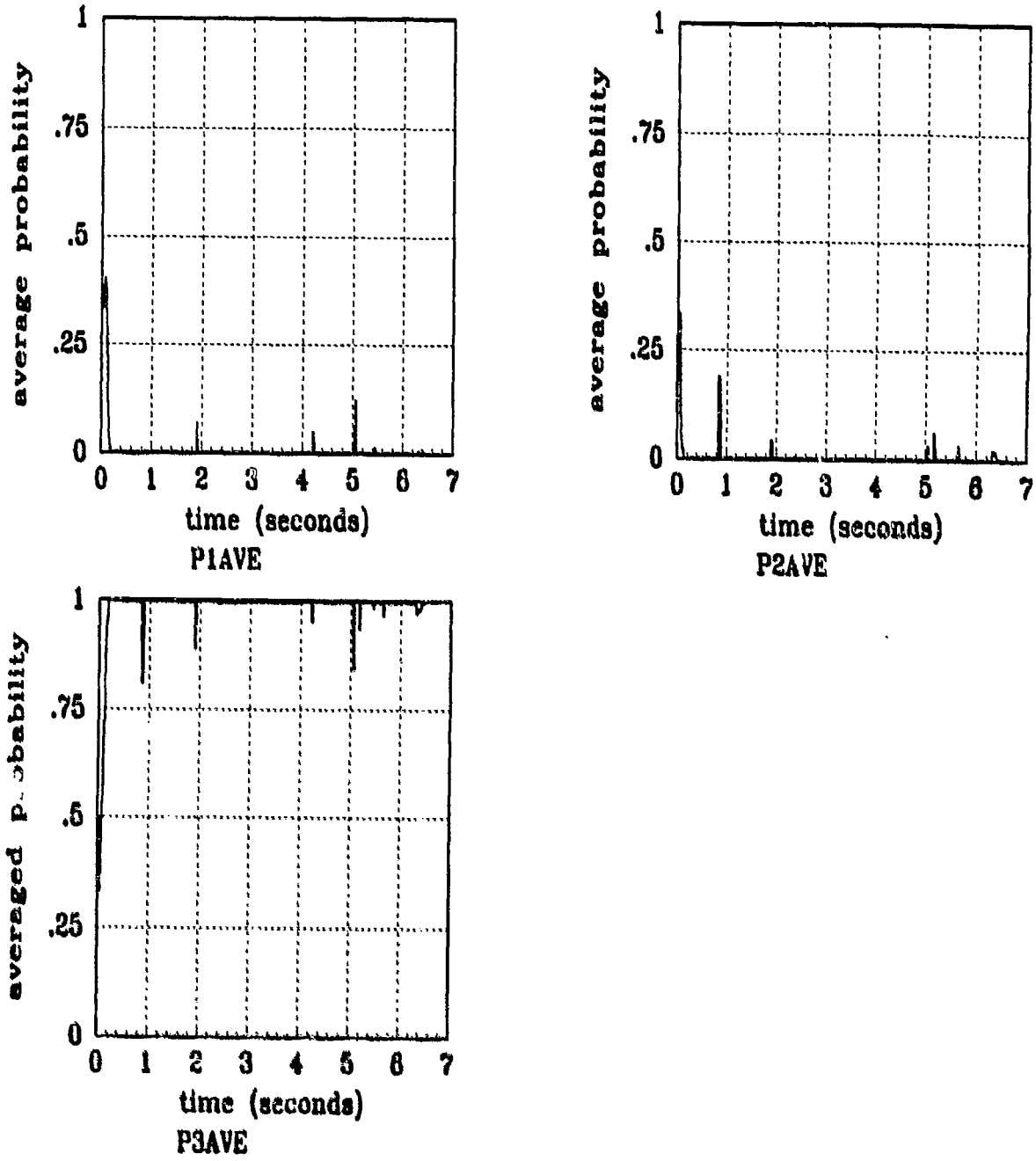


Figure G.33 Filter Residual Quantities; Computed *without* leading coefficient (Truth = $1.12/20\pi$)

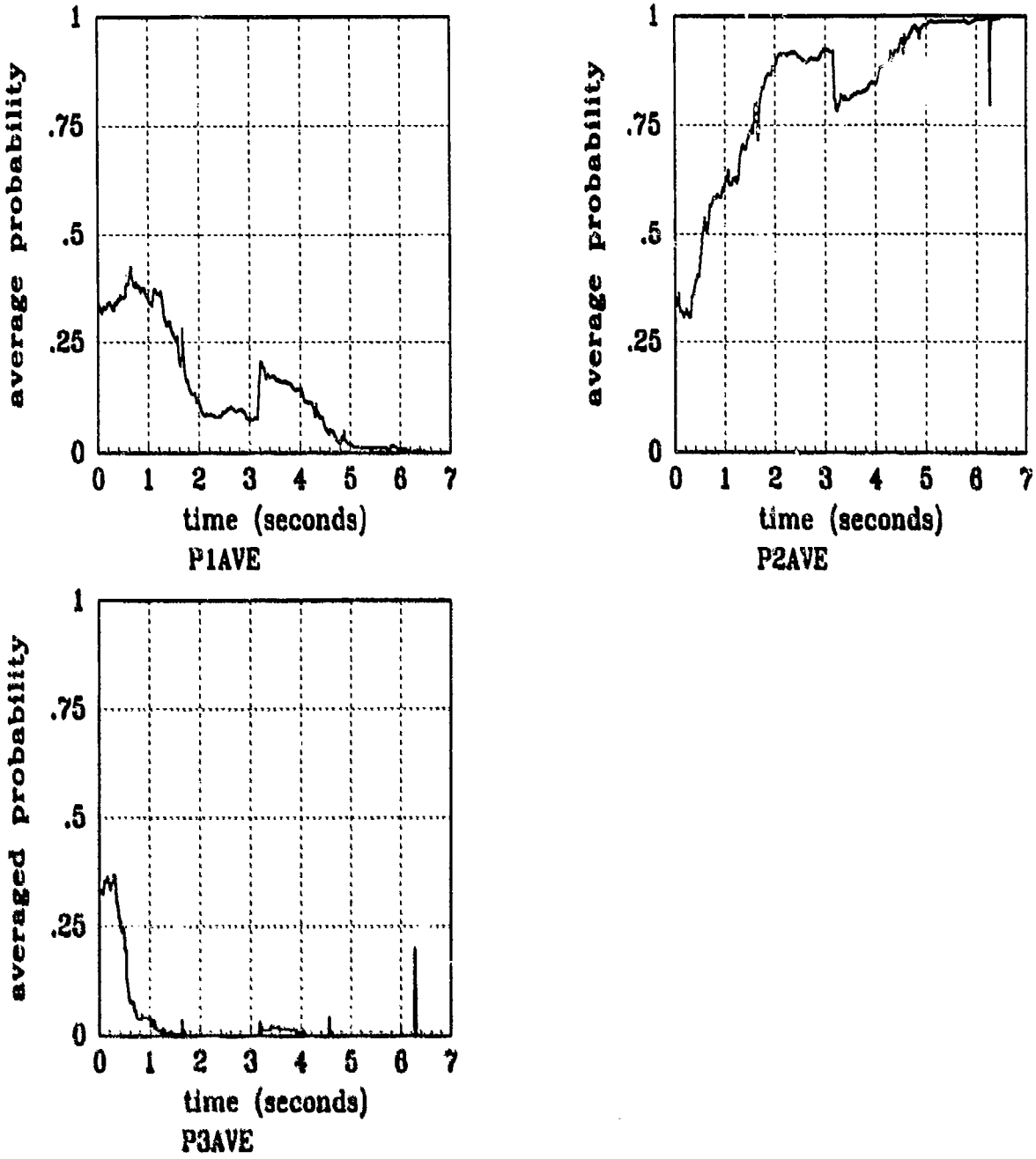


Figure G.34 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1. 12/4π)

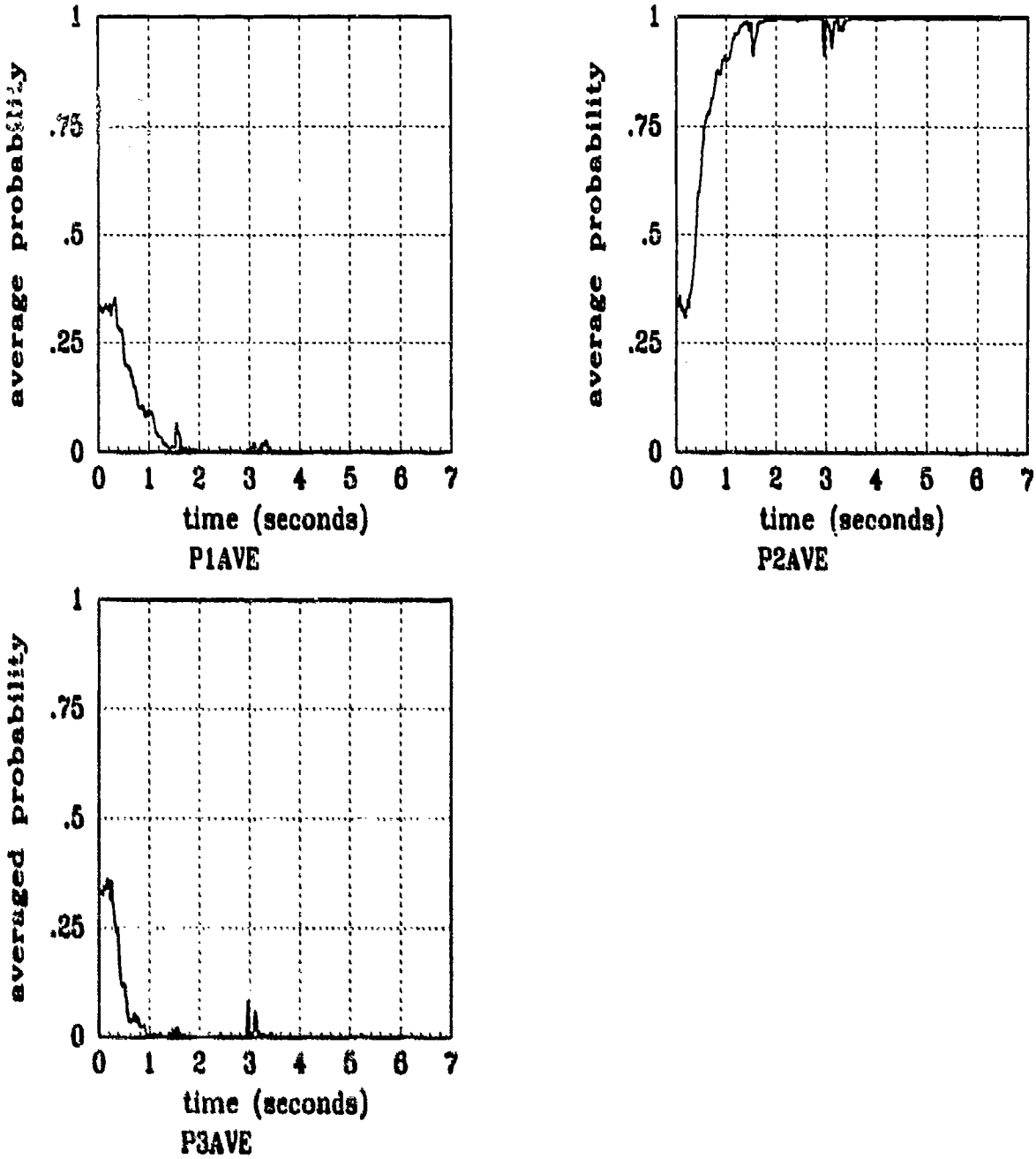


Figure G.35 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = $1.12/8\pi$)

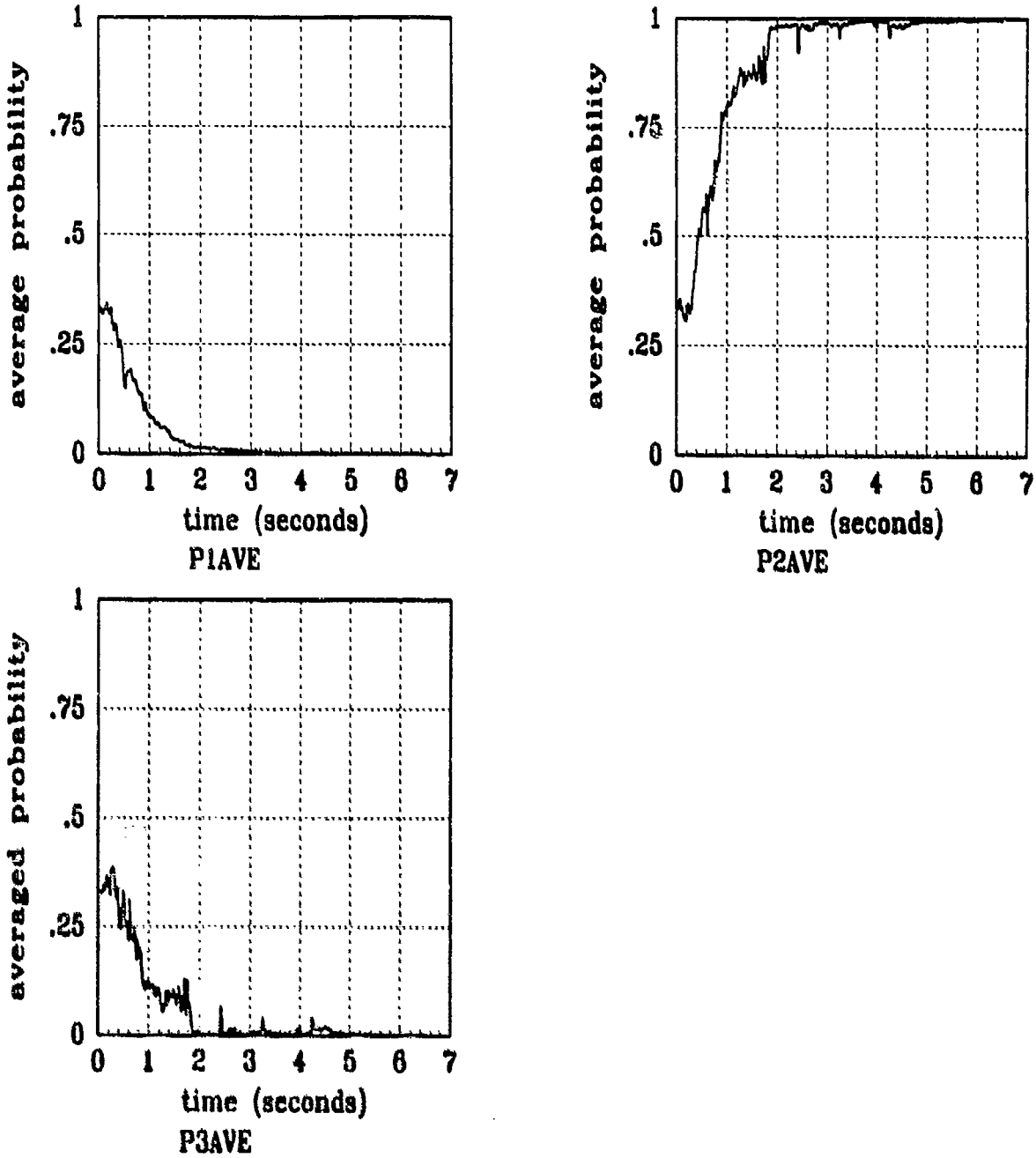


Figure G.36 Filter Residual Quantities; Computed *without* leading coefficient or $A_1(t)$ (Truth = 1.12/9.5π)

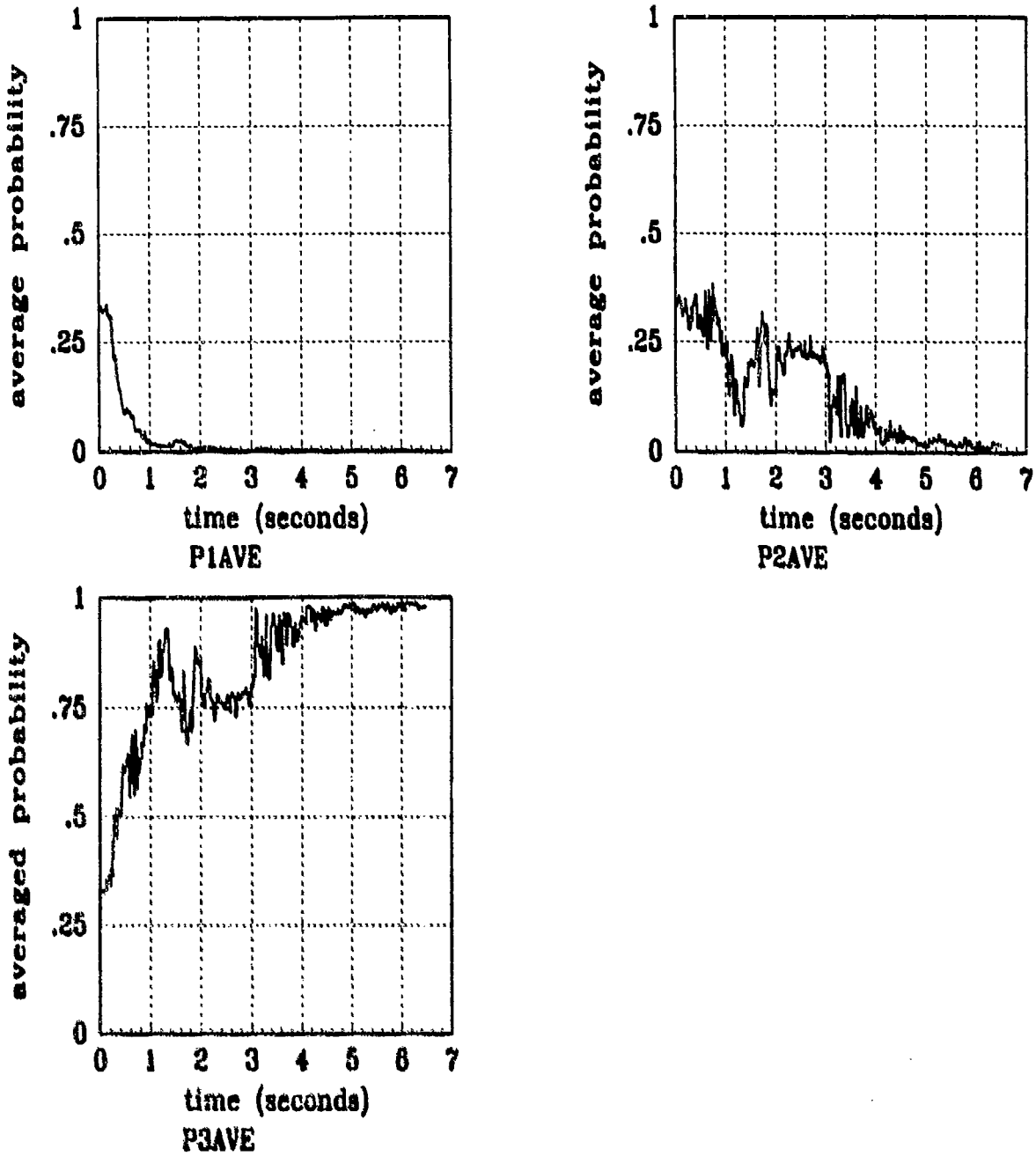


Figure G.37 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = $1.12/11\pi$)

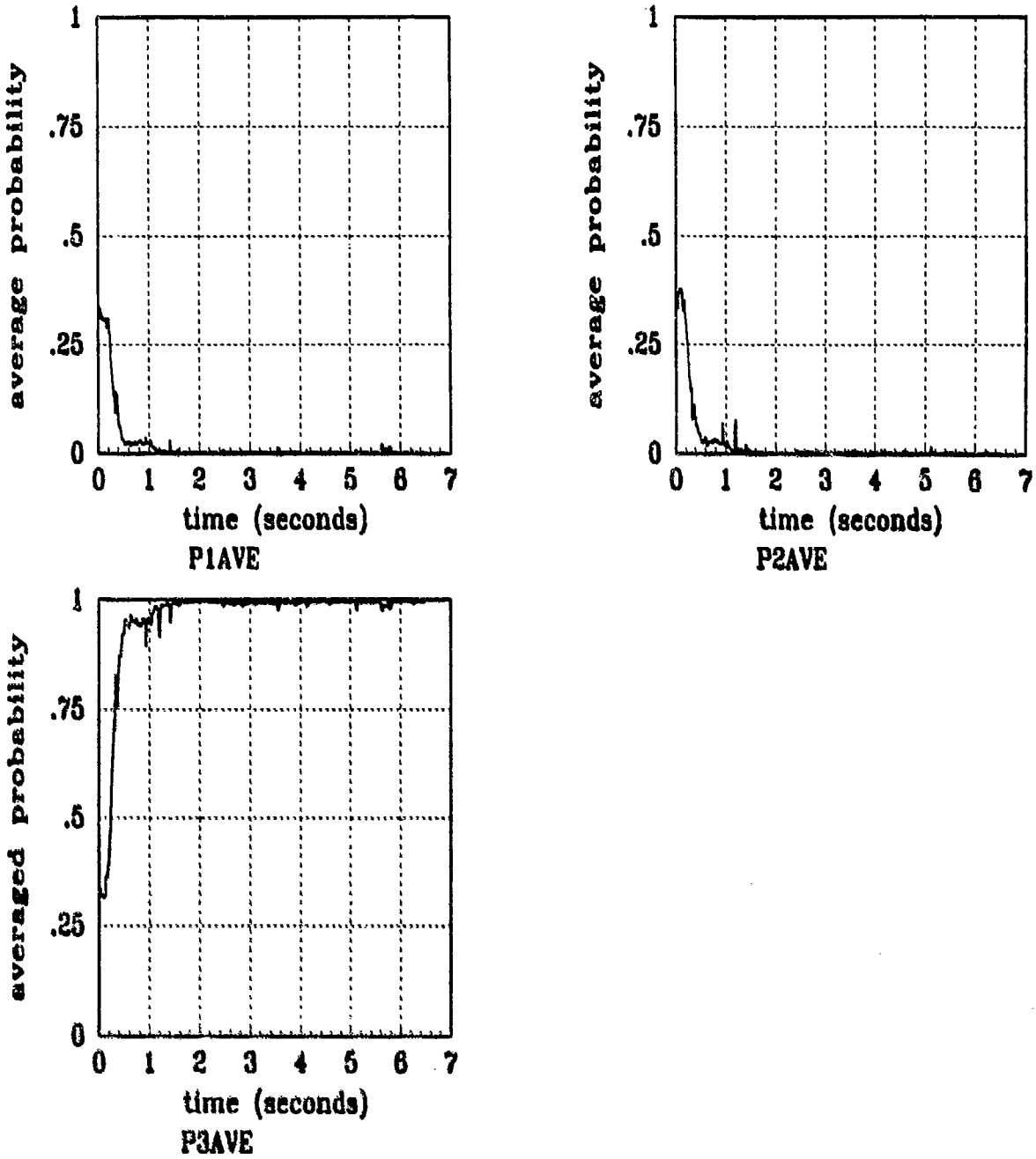


Figure G.38 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = 1. 12/14π)

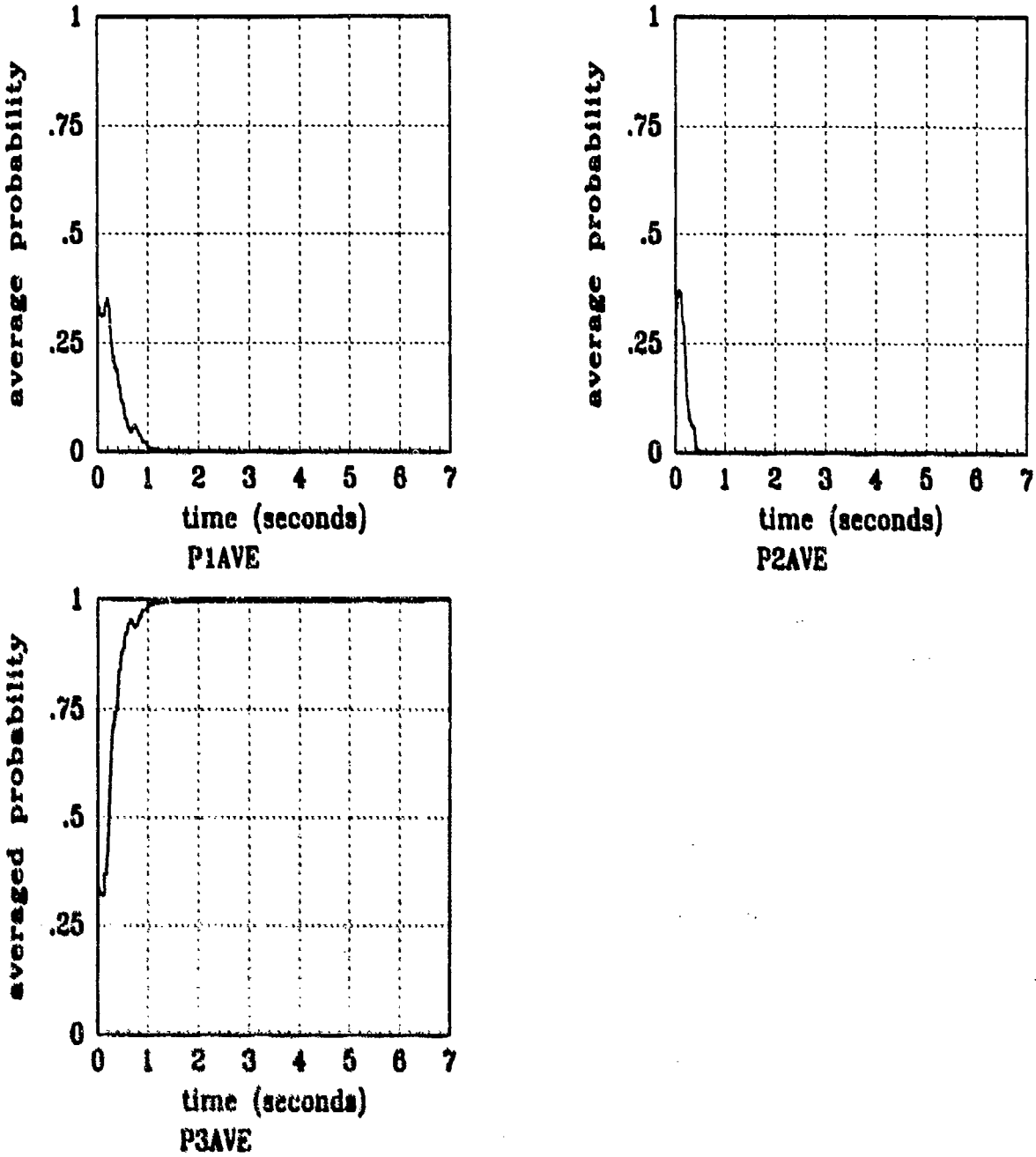


Figure G.39 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = .784/20π)

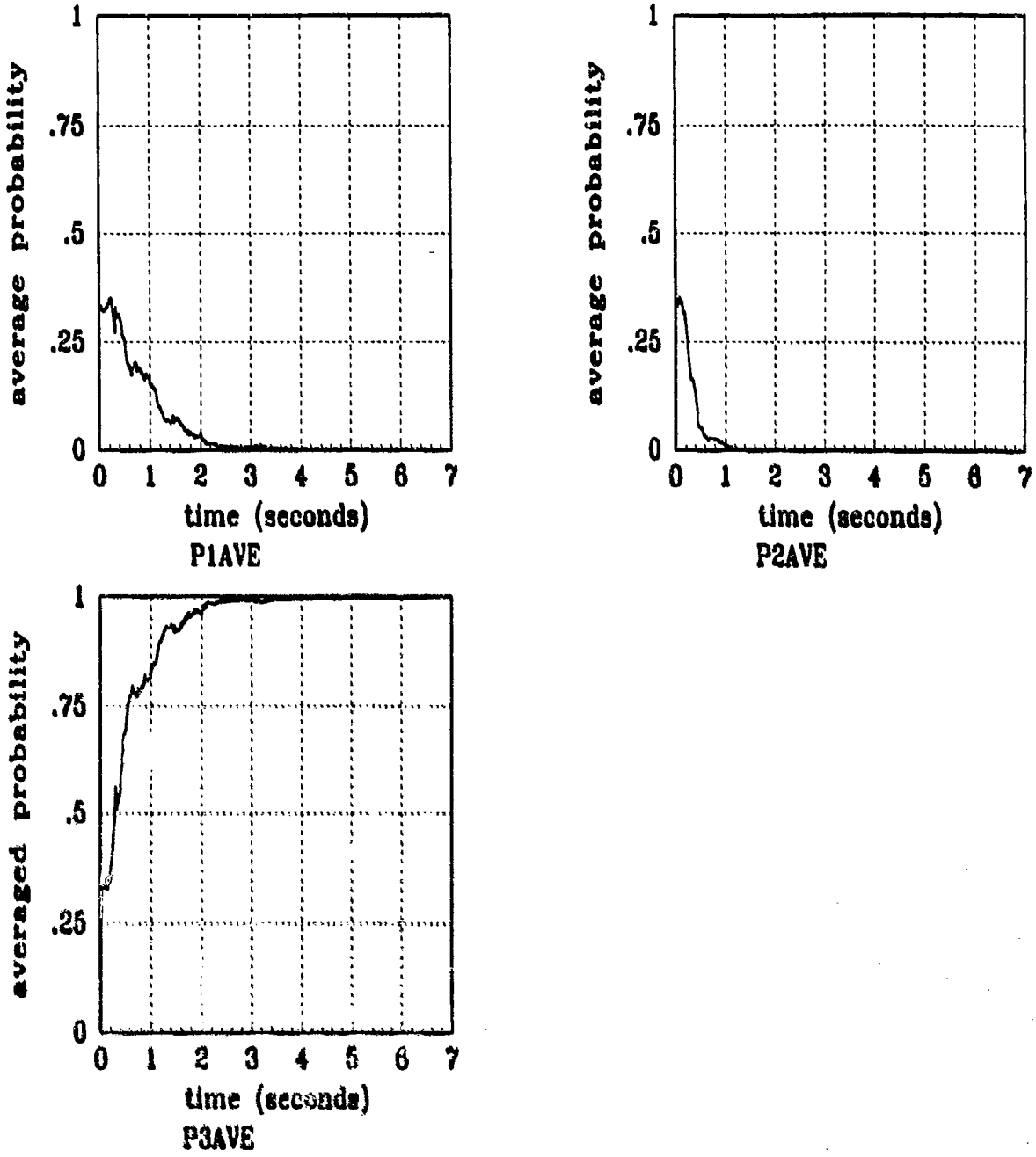


Figure G.40 Filter Residual Quantities; Computed *without* leading coefficient or $A_1(t_i)$ (Truth = 0.448/20π)

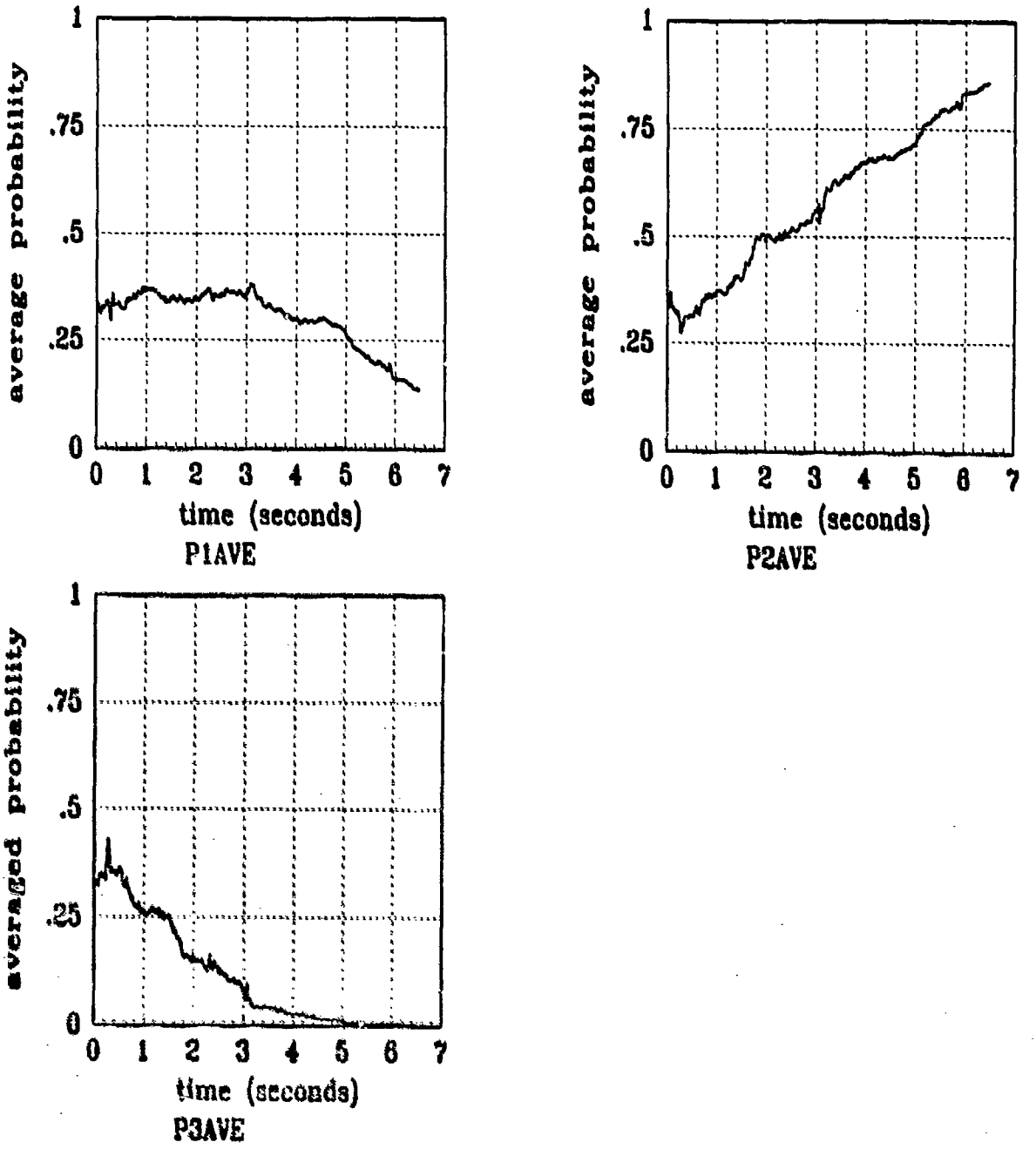


Figure G.41 Filter Residual Quantities; Computed *without* leading coefficient or $A_1(t_i)$ (Truth = $0.112/2\pi$)

Appendix H

MMAE Configuration #2 Output Plots

This appendix contains the state and error statistics plots of the nine-state elemental filters. The data depicted in the two types of plots in this appendix are explained in *Appendix A*. The state comparison plots show the truth state over the 5 *Monte Carlo* runs compared to the same statistic for the filter estimate. The error statistics plots represent the error mean $\pm$ standard deviation values in pixels (or pixel/second for velocity and pogo velocity), of the errors between the filter estimate and true state; true mean $\pm$ one true standard deviation are plotted, along with zero $\pm$ one filter-computed standard deviation. Note the filter covariance calculation is shown in Equation (4.5).

All probability calculations, from this point on, will *exclude* the leading coefficient and $A_k(t_i)$ from Equations (2.30) and (2.31).

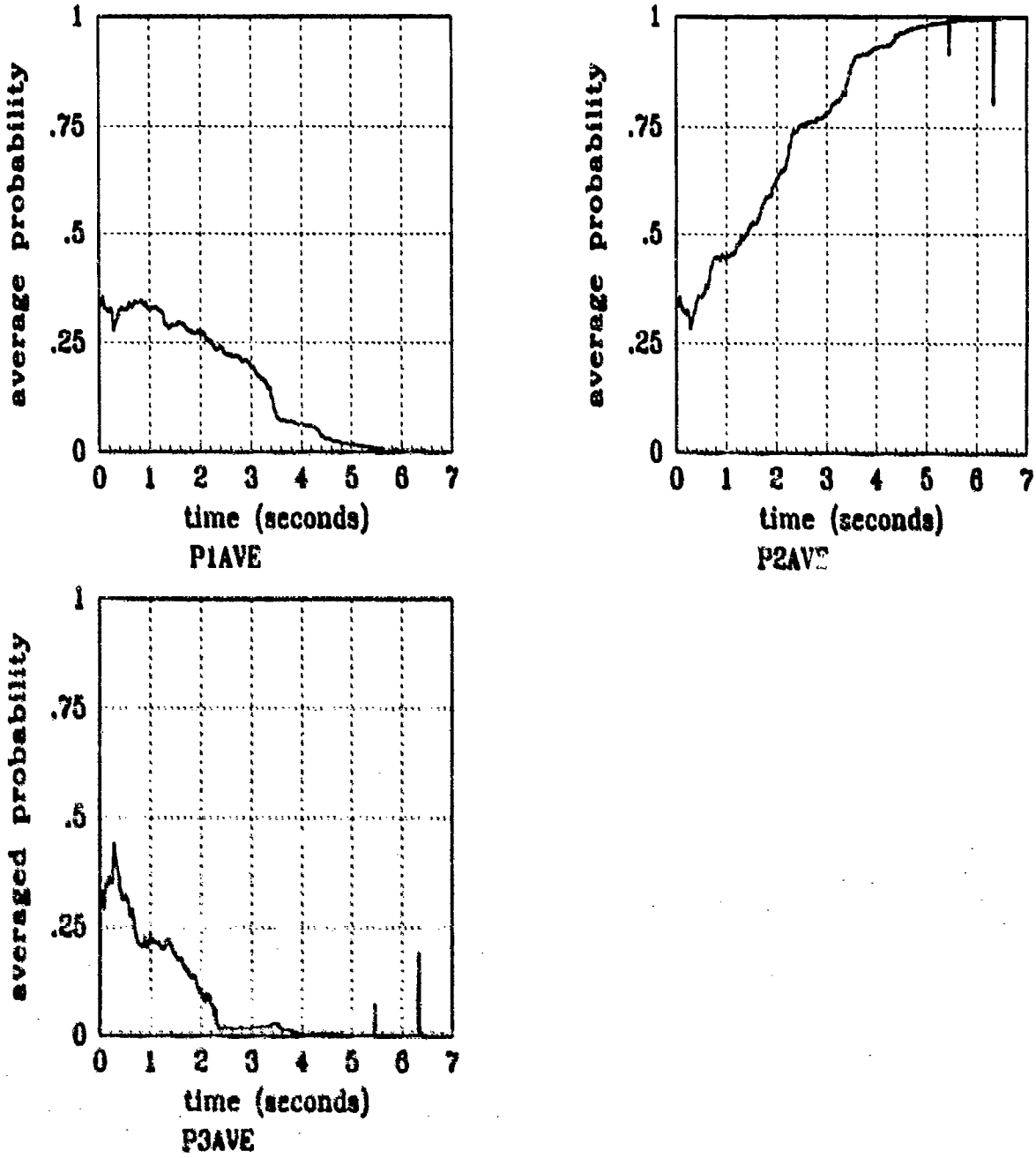


Figure H.1 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = $1.12/2\pi$)

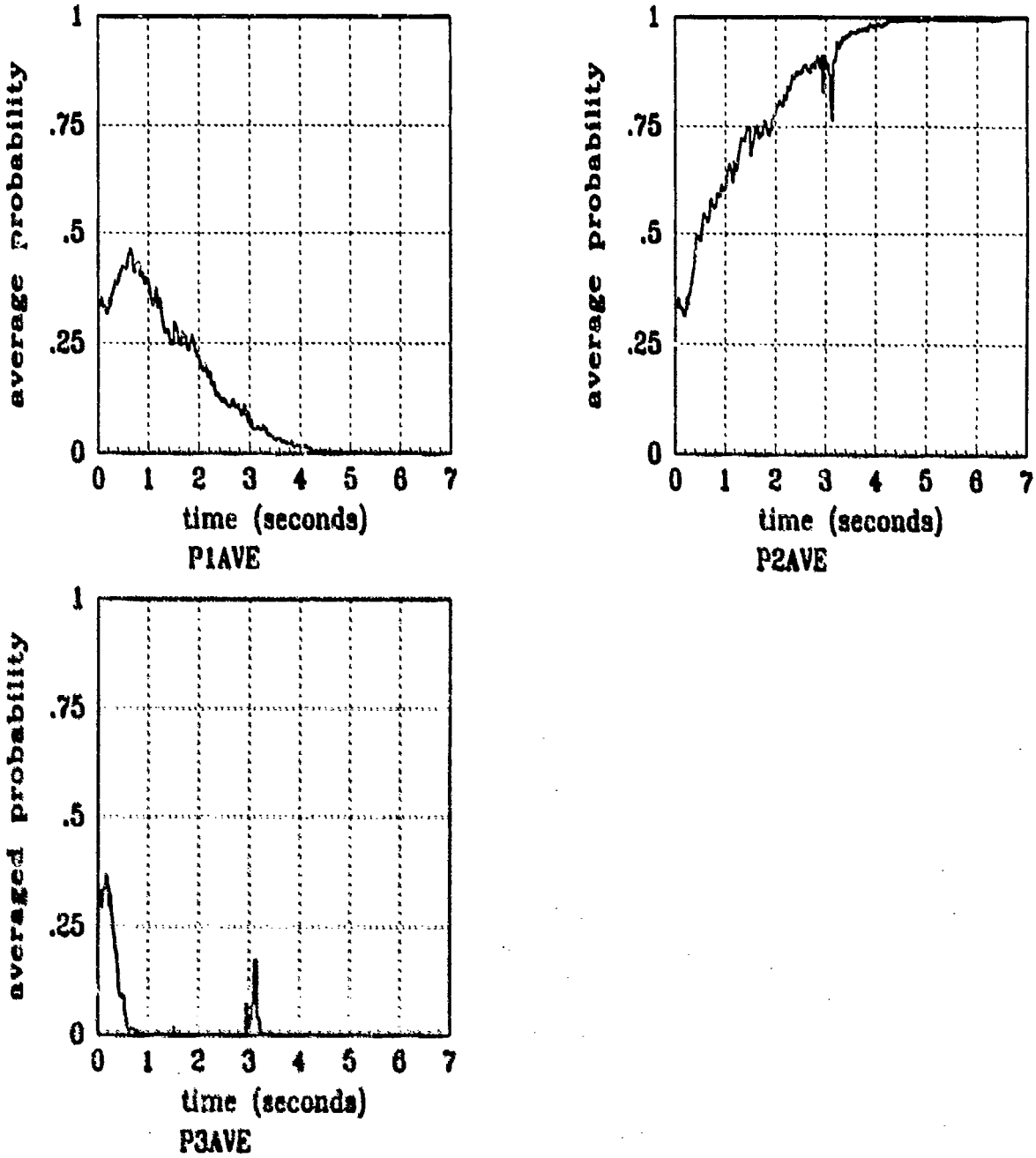


Figure H.2 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = $1.12/8\pi$)

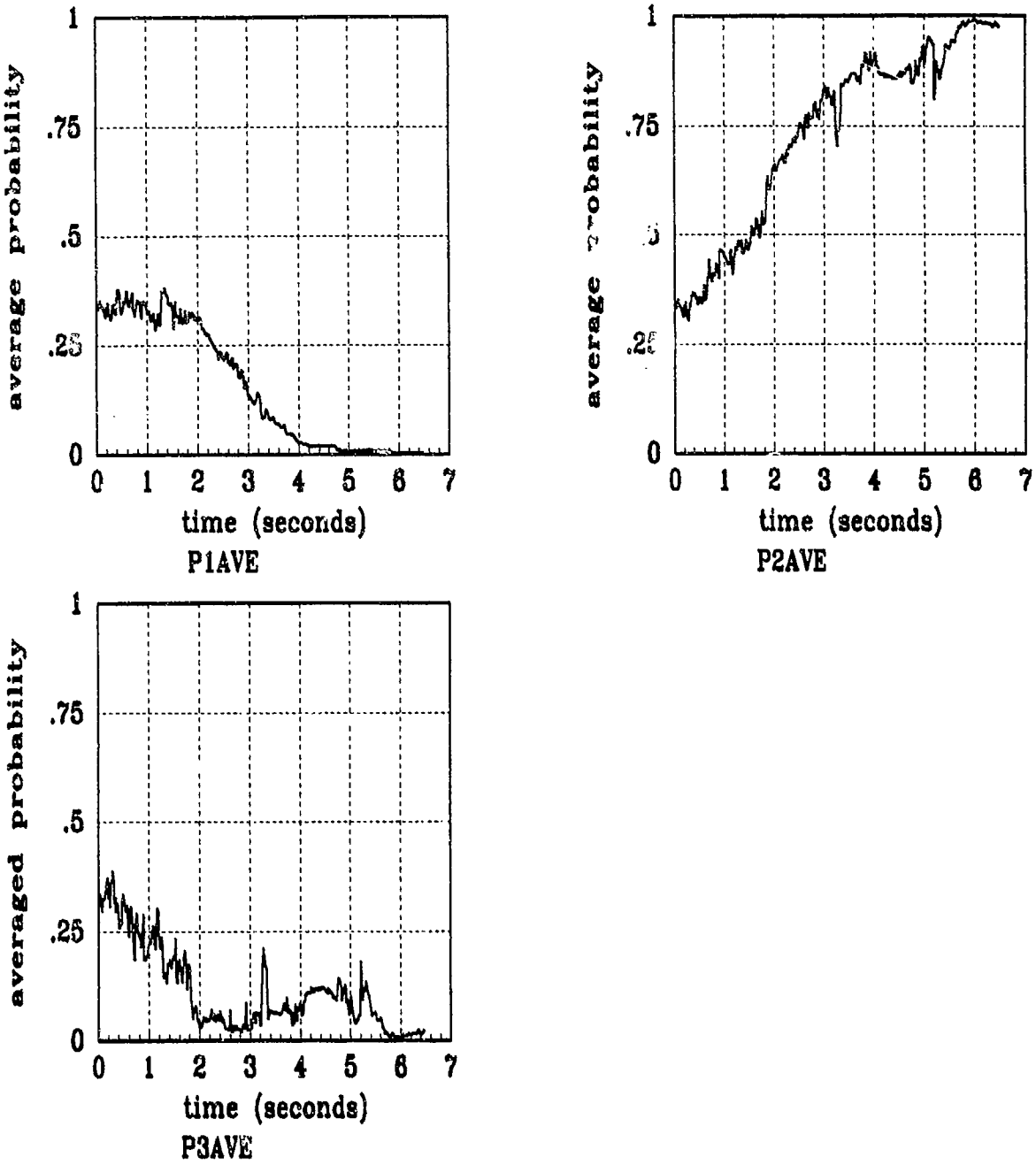


Figure H.3 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = $1.12/10\pi$)

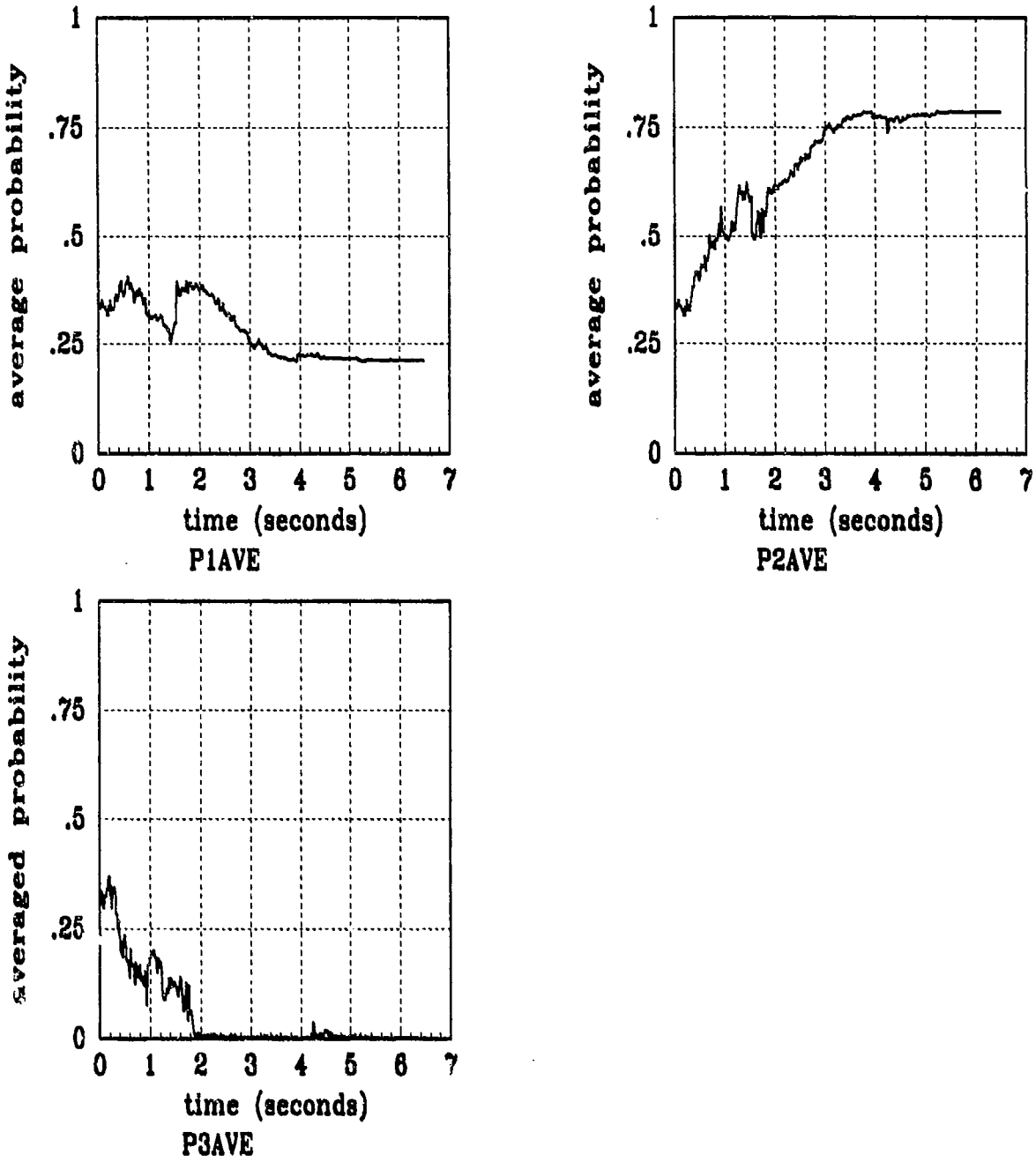


Figure H.4 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = $1.12/10.5\pi$)

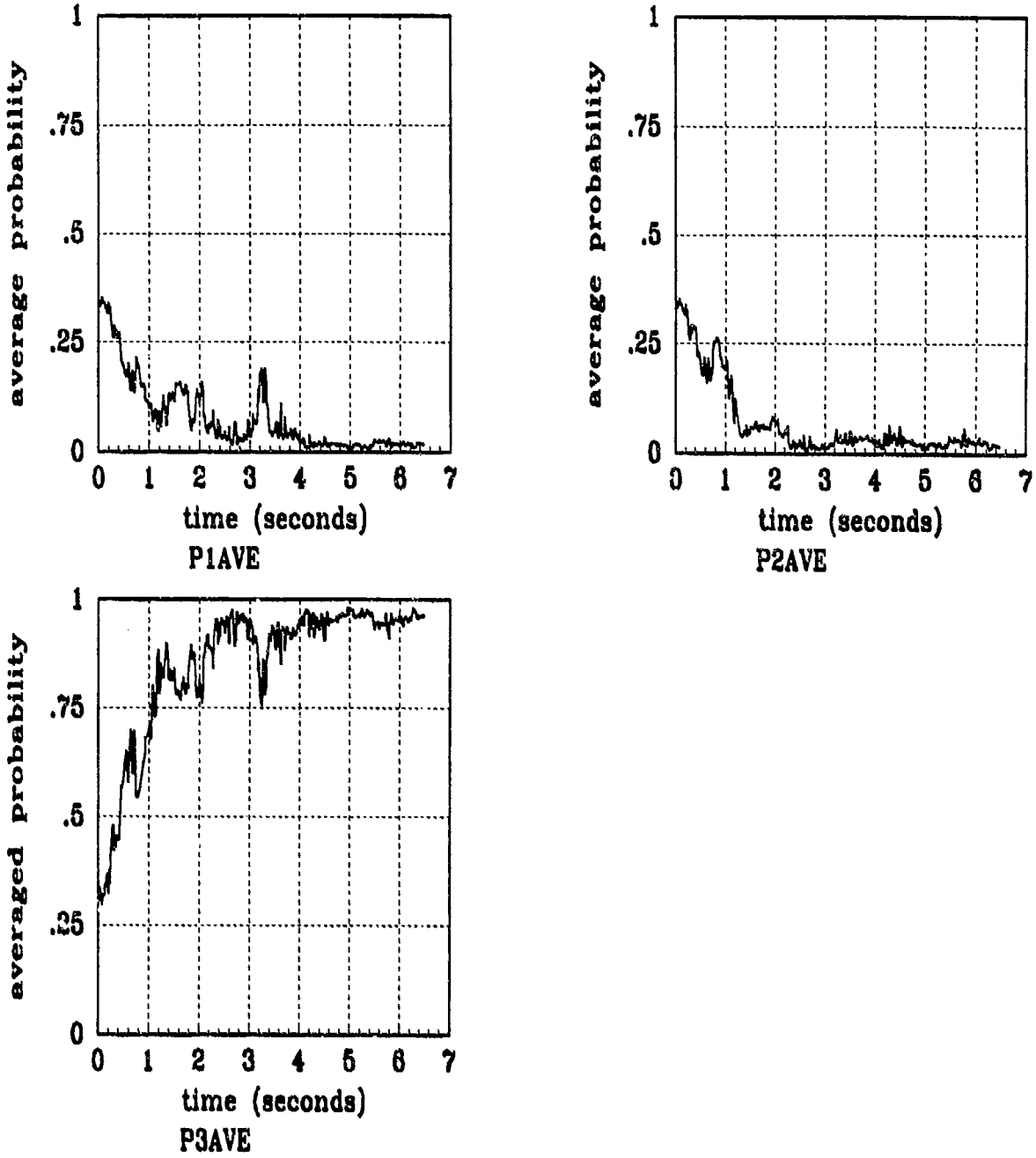


Figure H.5 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = $1.12/11\pi$)

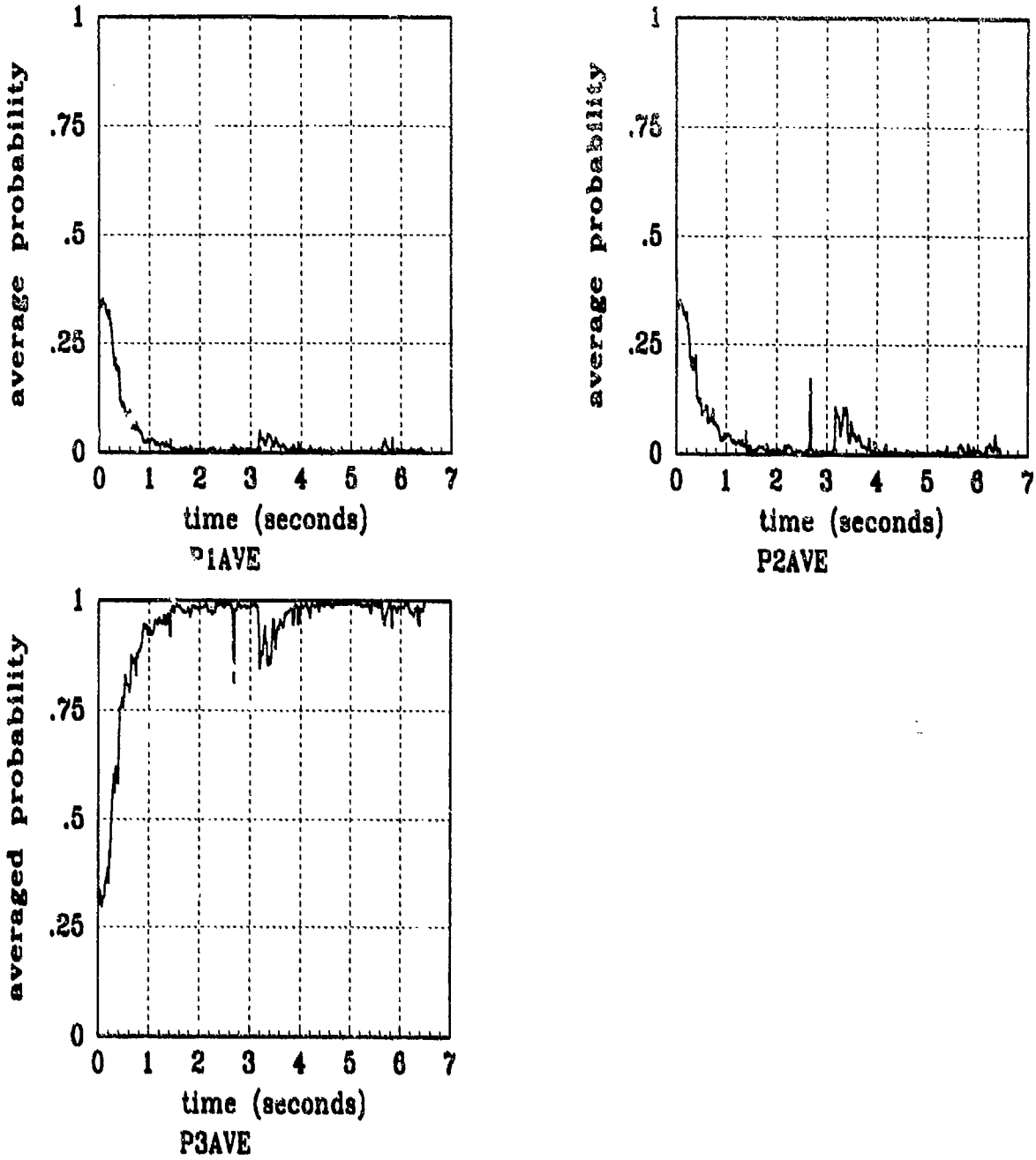


Figure H.6 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$
(Truth = $1.12/12\pi$)

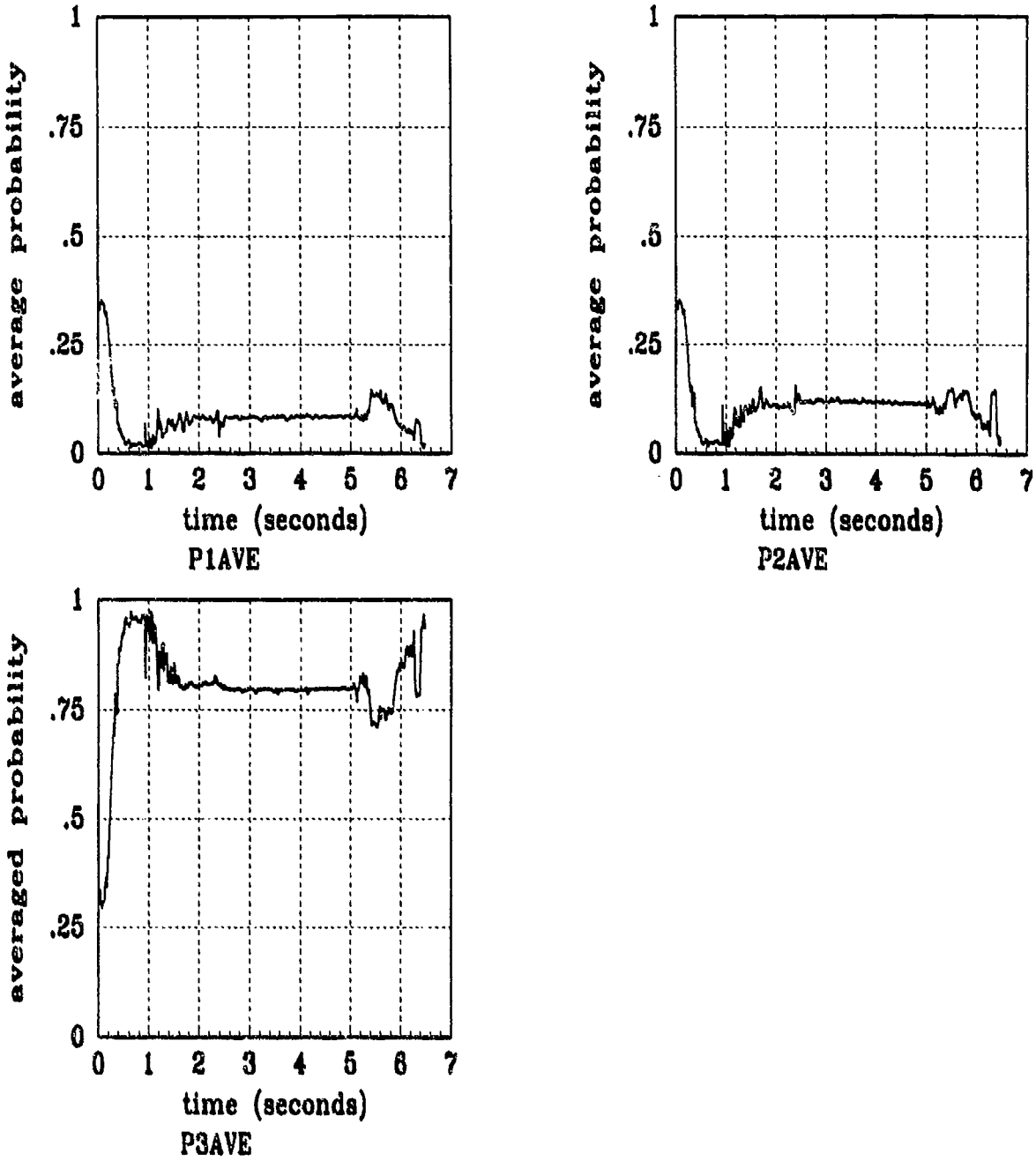


Figure H.7 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = $1.12/14\pi$)

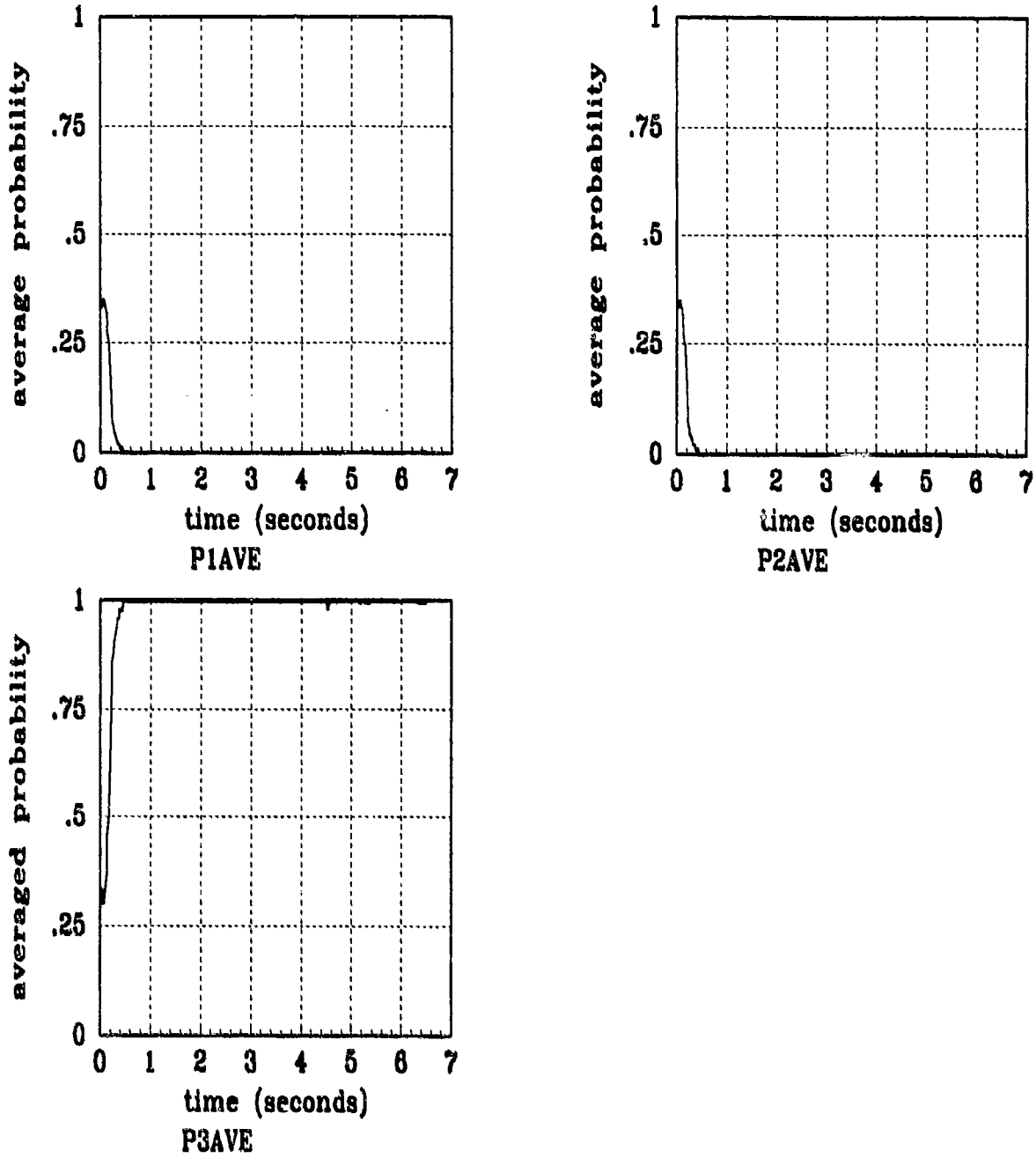


Figure H.8 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$
(Truth = $1.12/20\pi$)

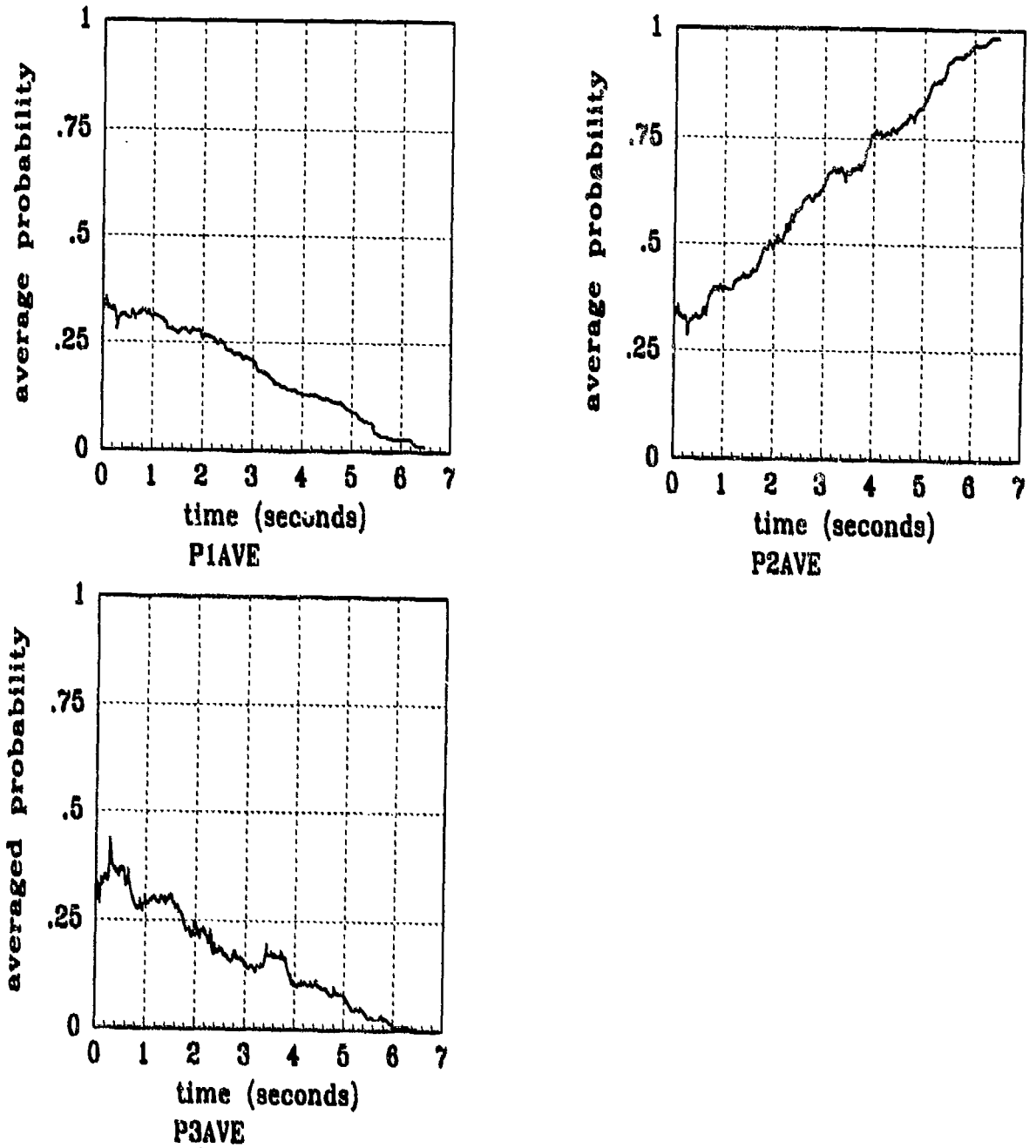


Figure H.9 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = $0.448/2\pi$)

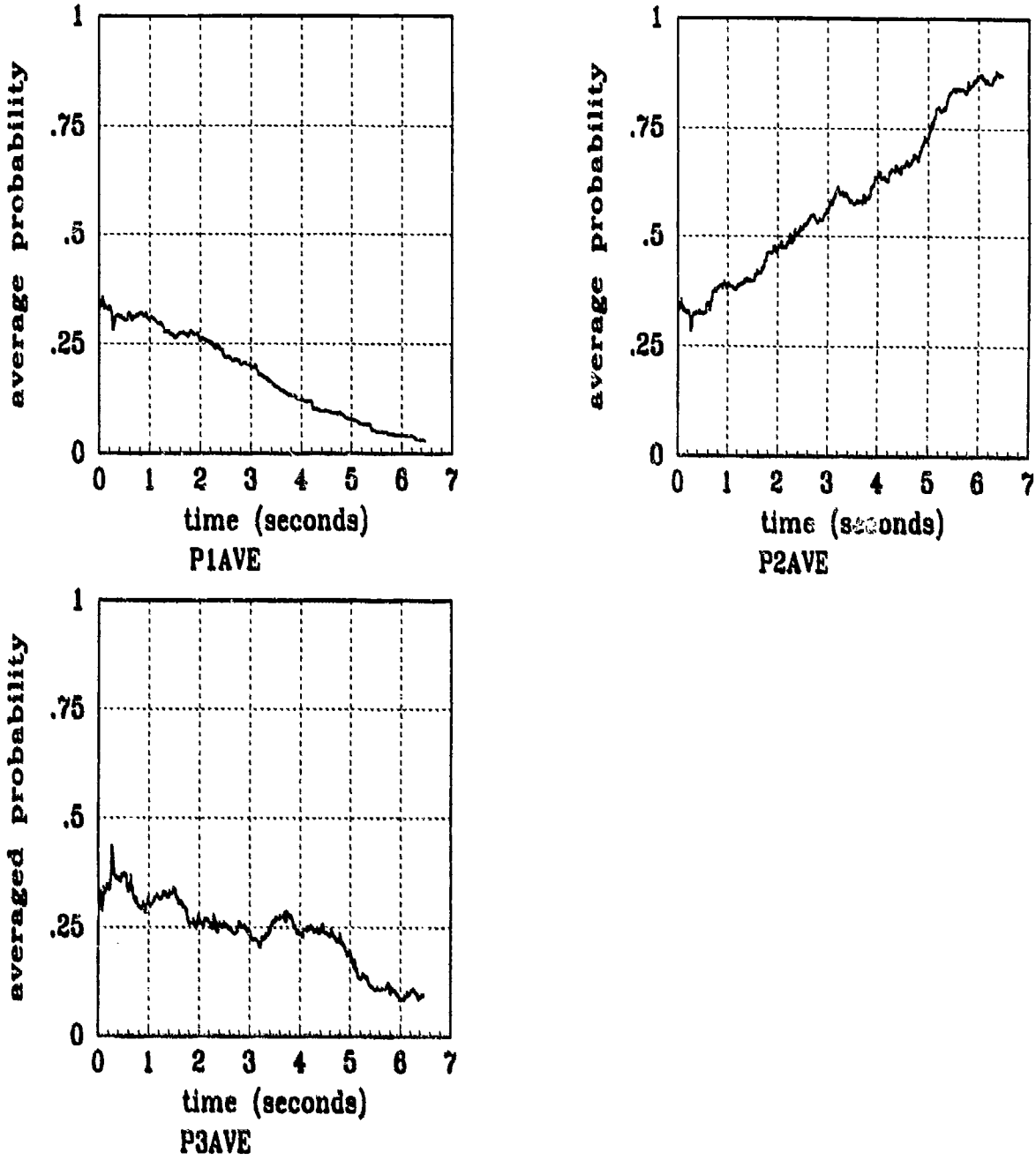


Figure H.10 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = $0.112/2\pi$)

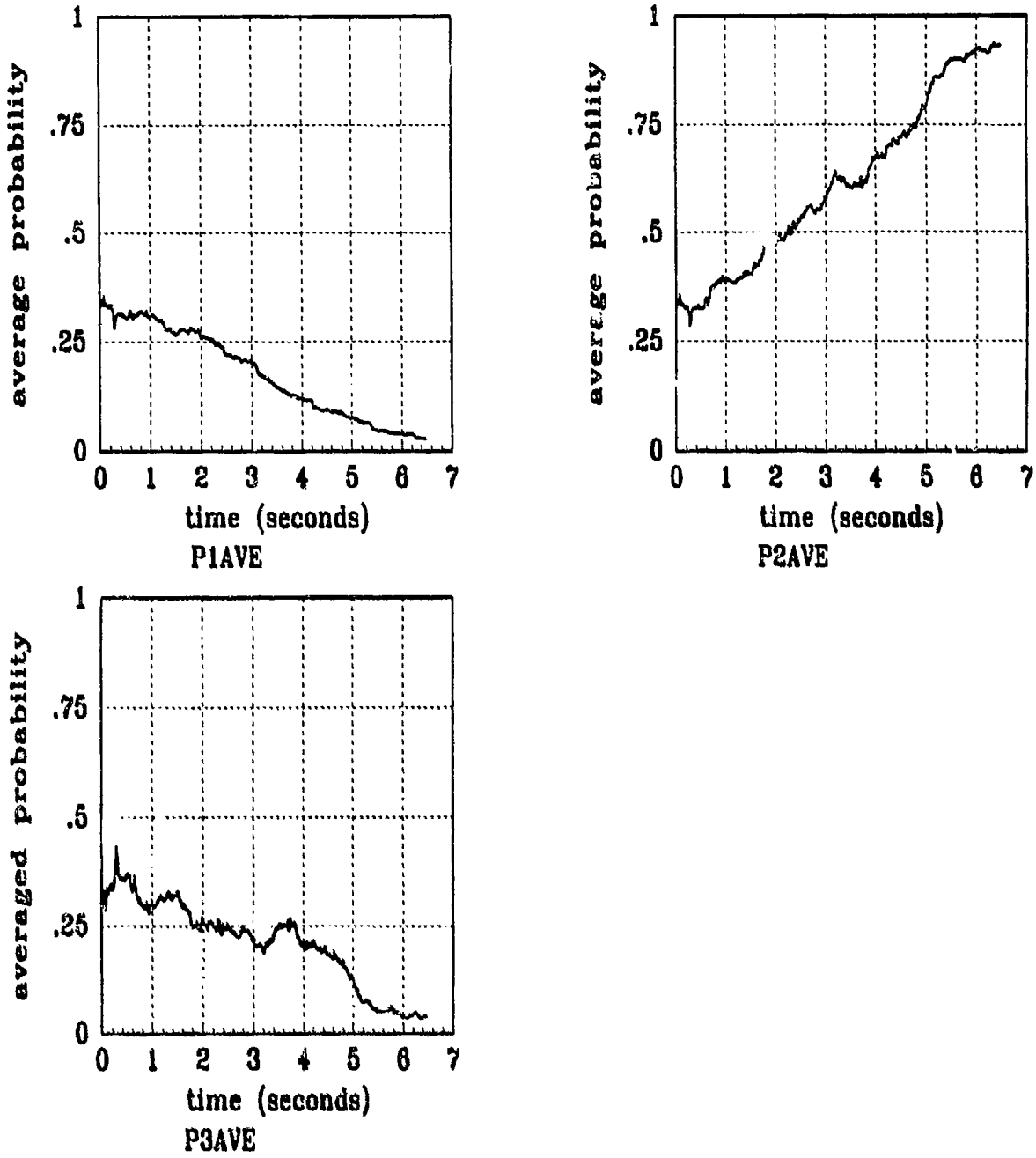


Figure H.11 Filter Residual Quantities; Computed *without* leading coefficient or $A_2(t_i)$ (Truth = $0.05/2\pi$)

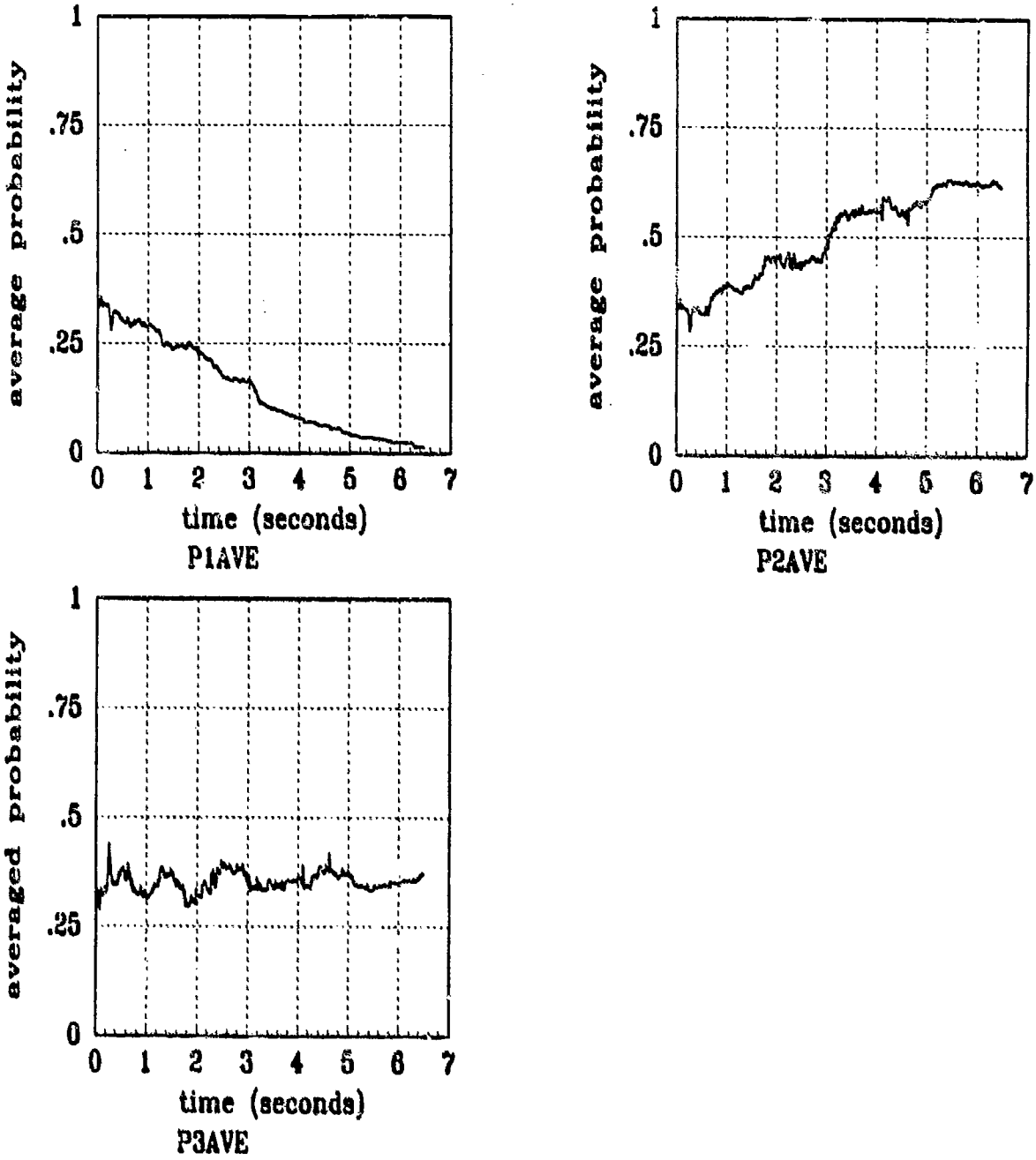


Figure H.12 Filter Residual Quantities; Computed *without* leading coefficient or $A_k(t_i)$ (Truth = $0.112/20\pi$)

Appendix I

MMAE Configuration #3 Output Plots

This appendix contains the state and error statistics plots of the nine-state elemental filters. The data depicted in the two types of plots in this appendix are explained in *Appendix A*. The state comparison plots show the sample mean truth state over the 5 *Monte Carlo* runs compared to the same statistic for the filter estimate. The error statistics plots represent the error mean $\pm$ standard deviation values in pixels (or pixel/second for velocity and pogo velocity), of the errors between the filter estimate and true state; true mean $\pm$ one true standard deviation are plotted, along with zero $\pm$ one filter-computed standard deviation.

Note the filter covariance calculation is shown in Equation (4.5).

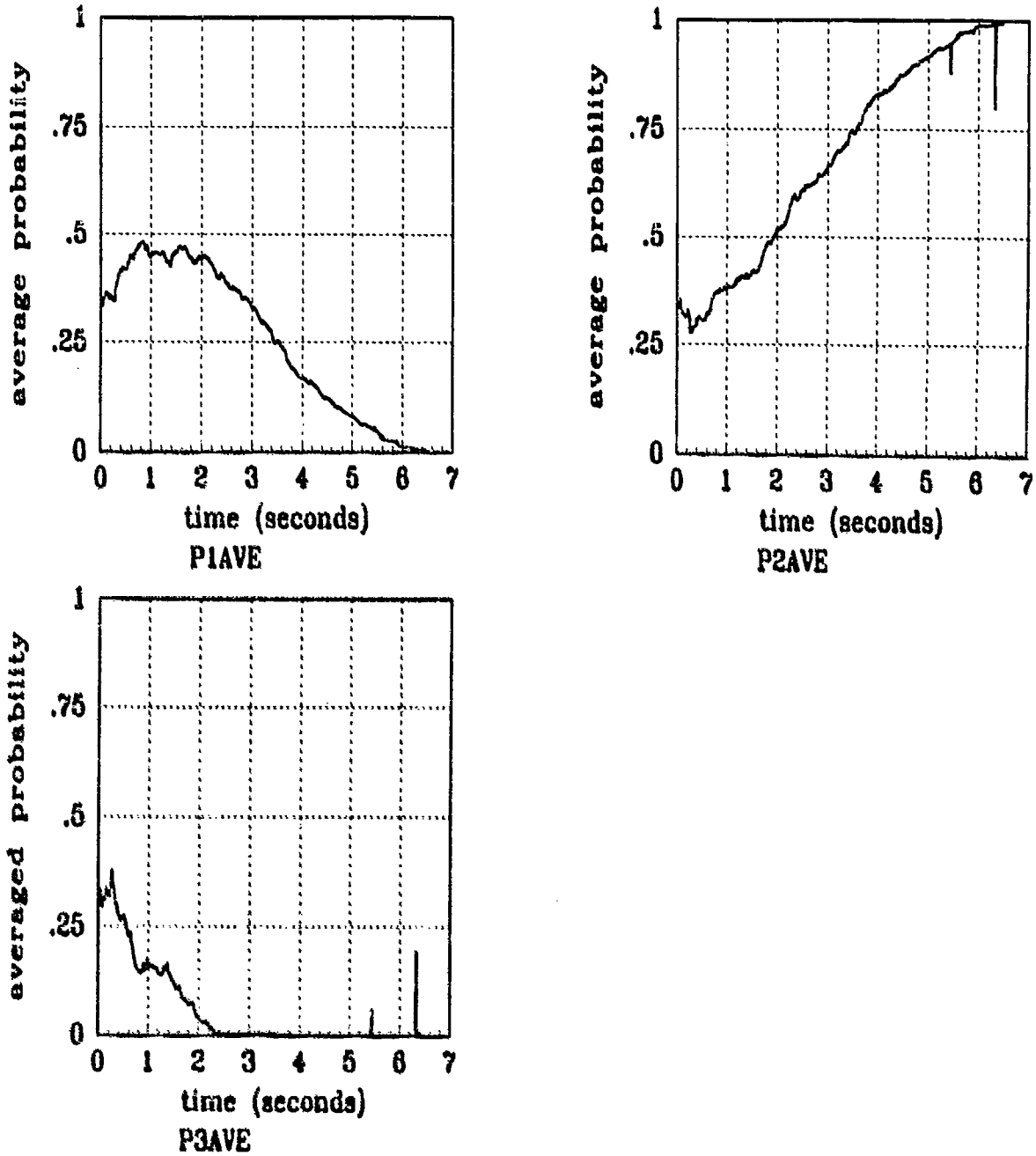


Figure I.1 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t_i)$ (Truth = $1.12/2\pi$)

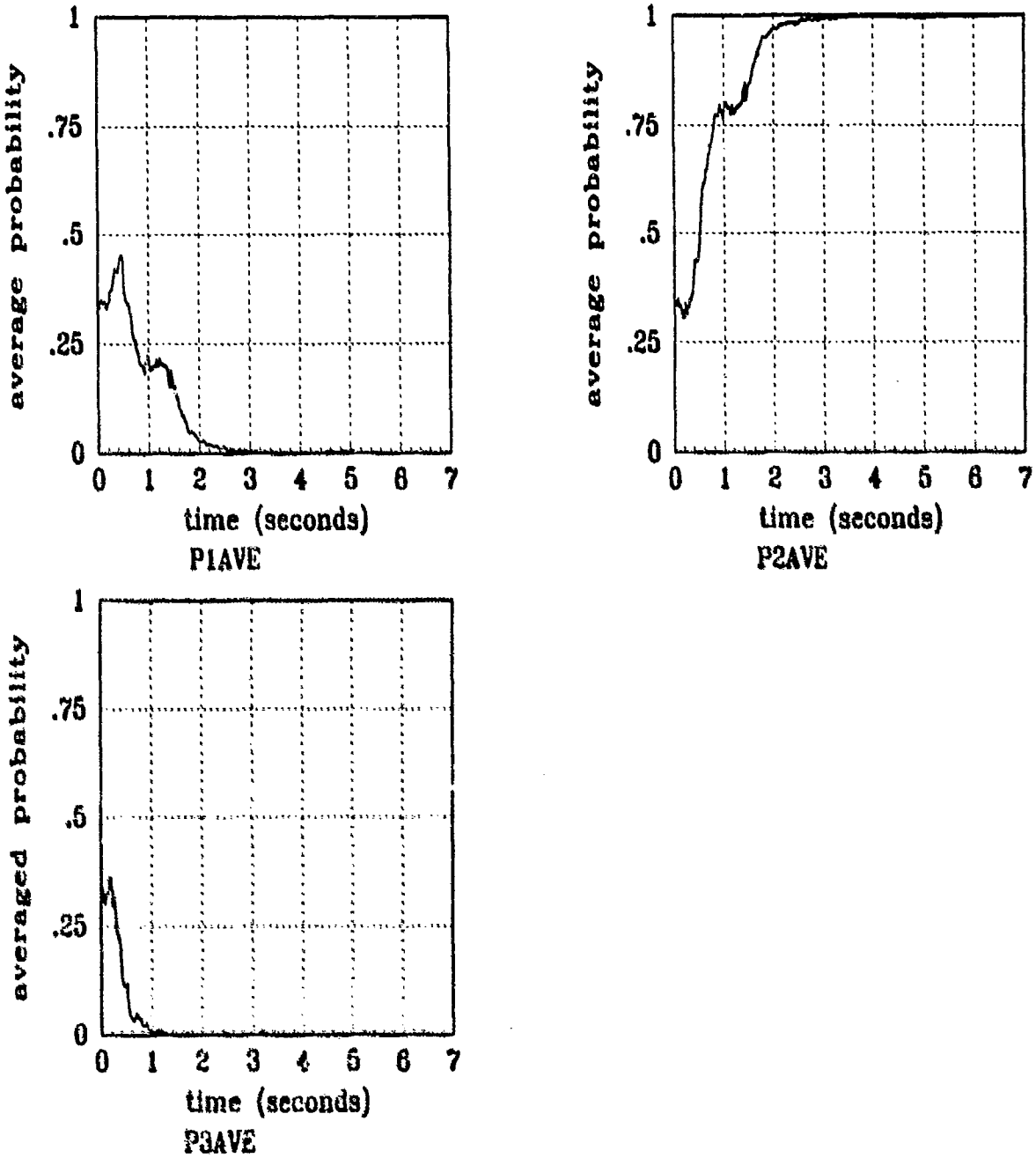


Figure I.2 Filter Residual Quantities; Co ited *without* leading coefficient or $A_2(t)$
(Truth = $1.12/8\pi$)

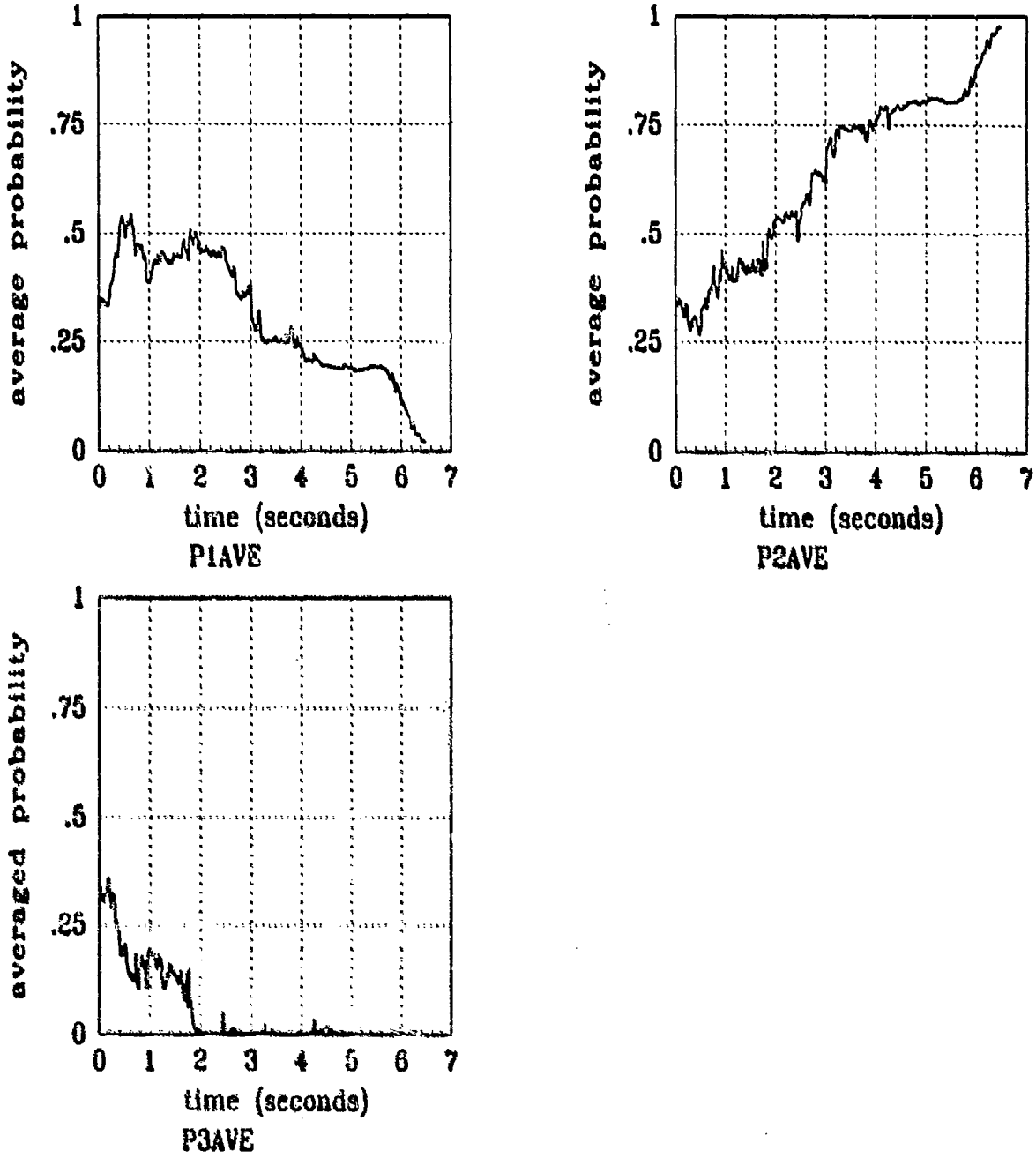


Figure I.3 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t)$
(Truth = 1.12/9.5π)

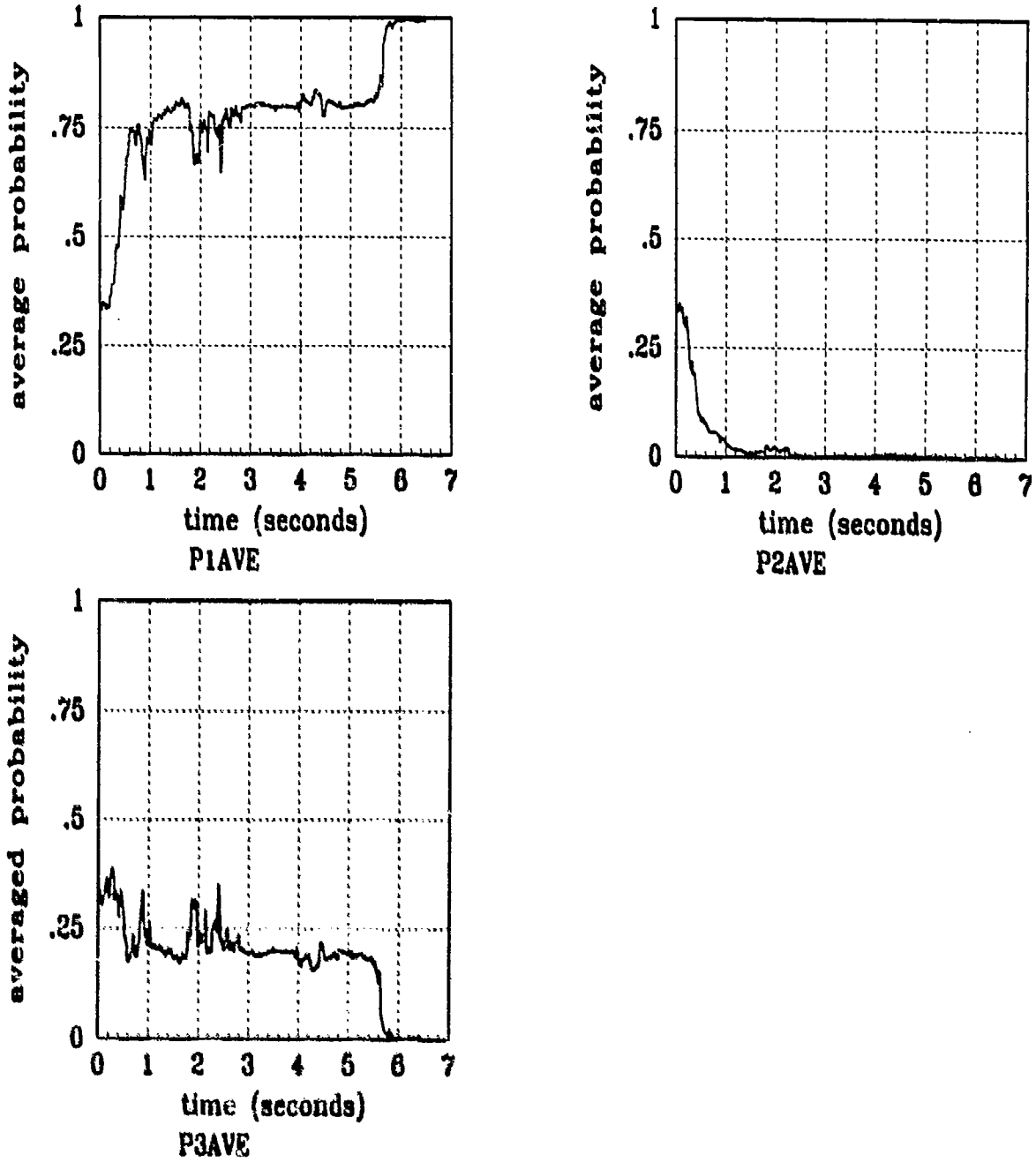


Figure I.4 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$
(Truth = $1.12/11\pi$)

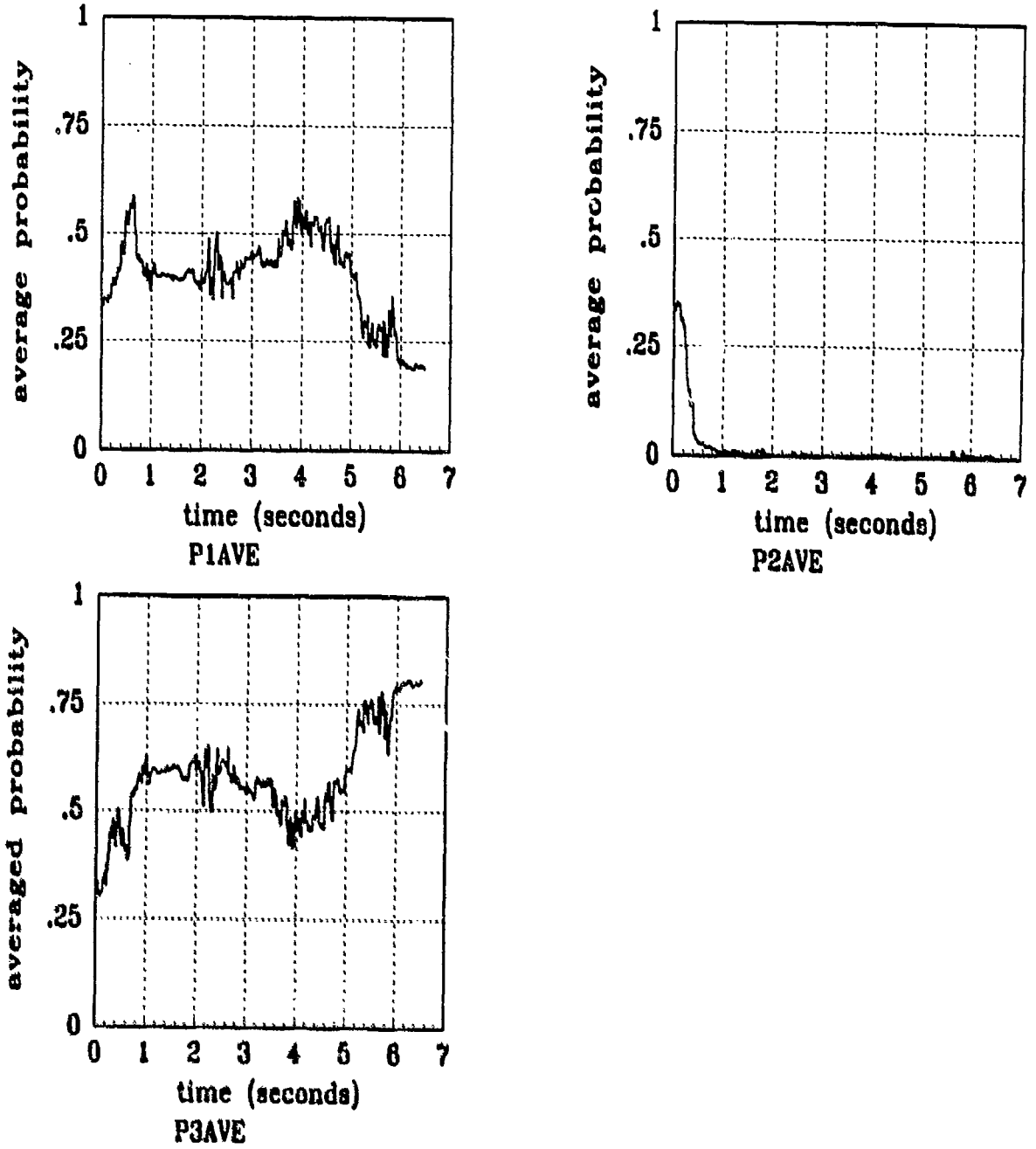


Figure I.5 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 1.12/12.5π)

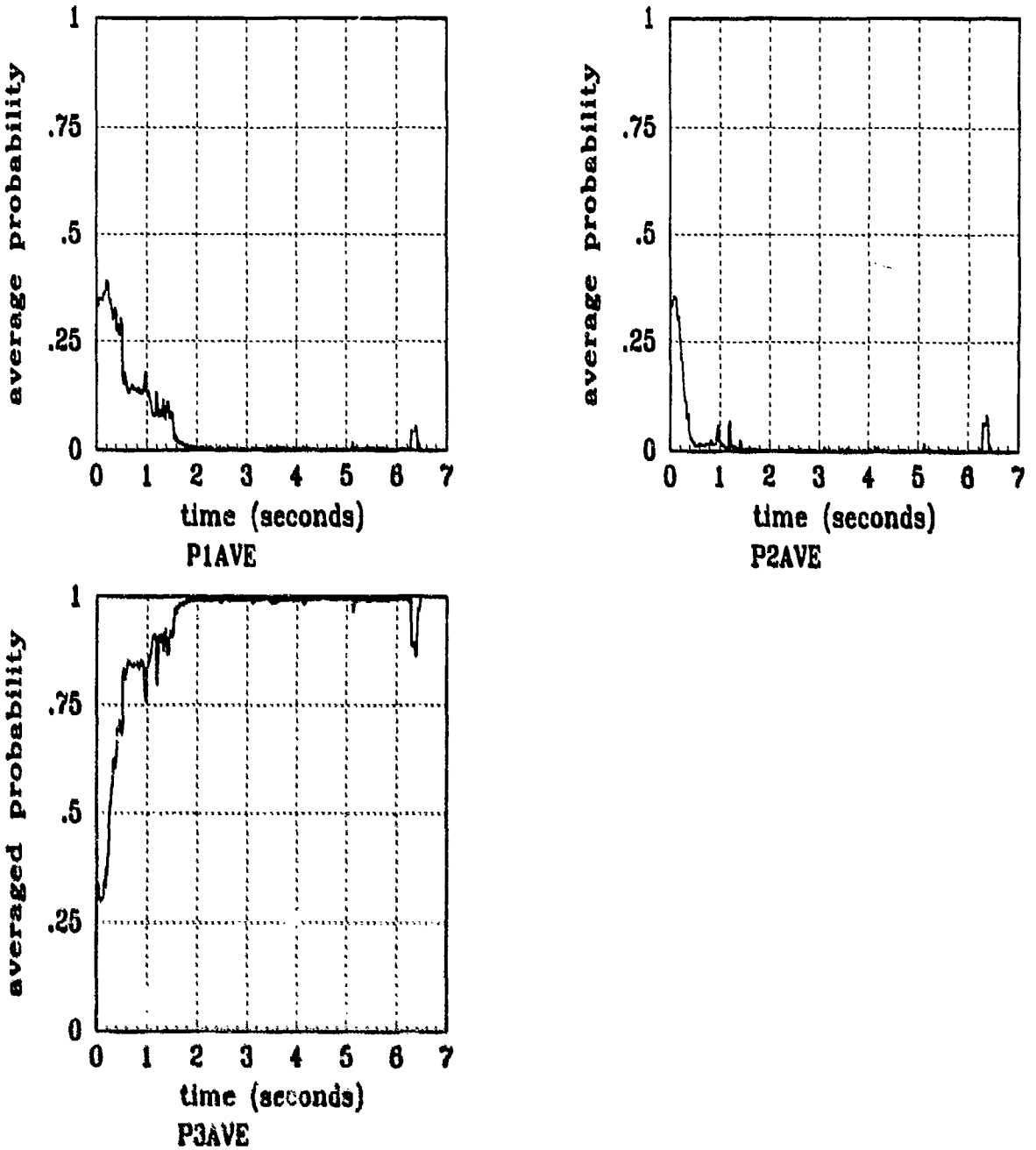


Figure I.6 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = $1.12/14\pi$)

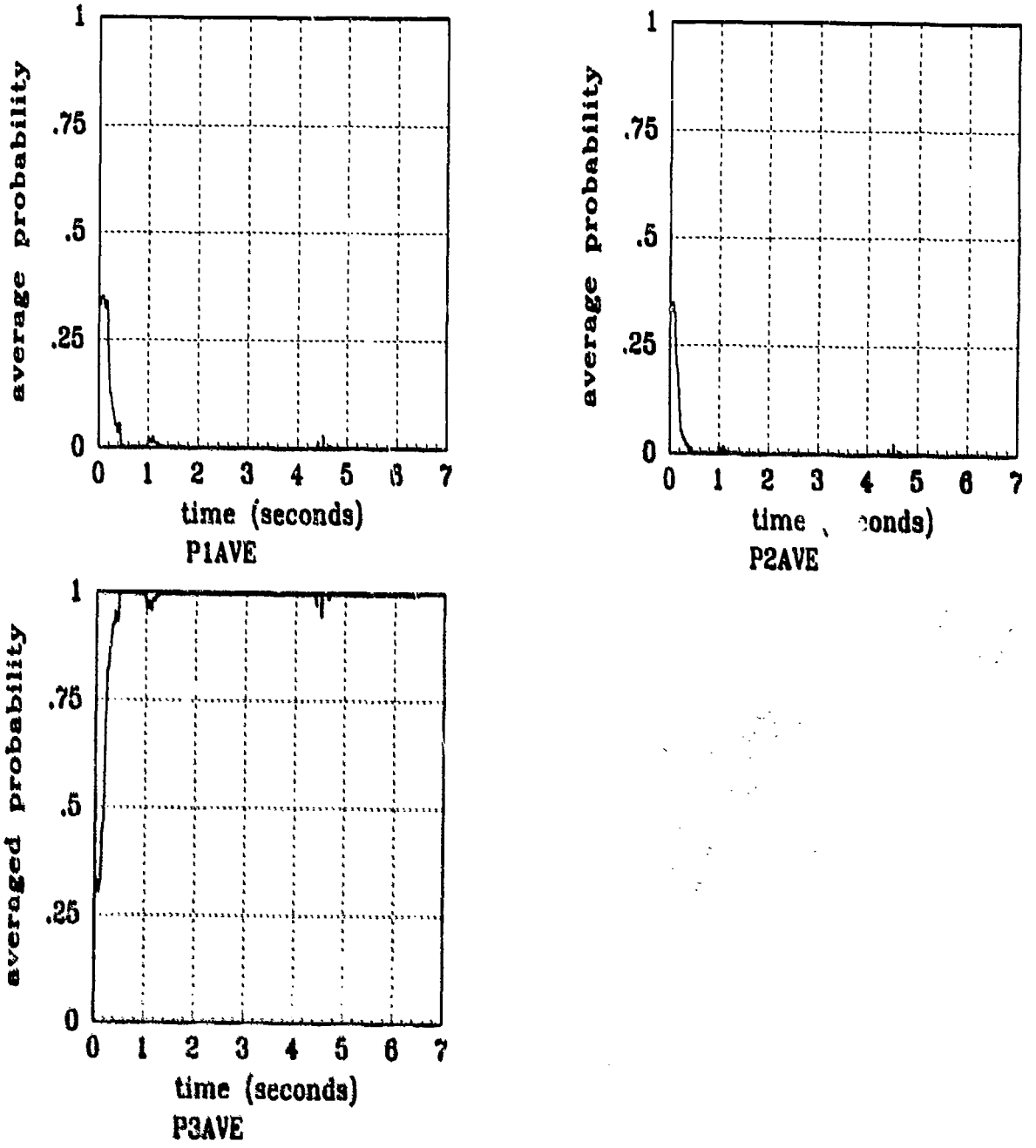


Figure I.7 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t_i)$ (Truth = 1.12/20π)

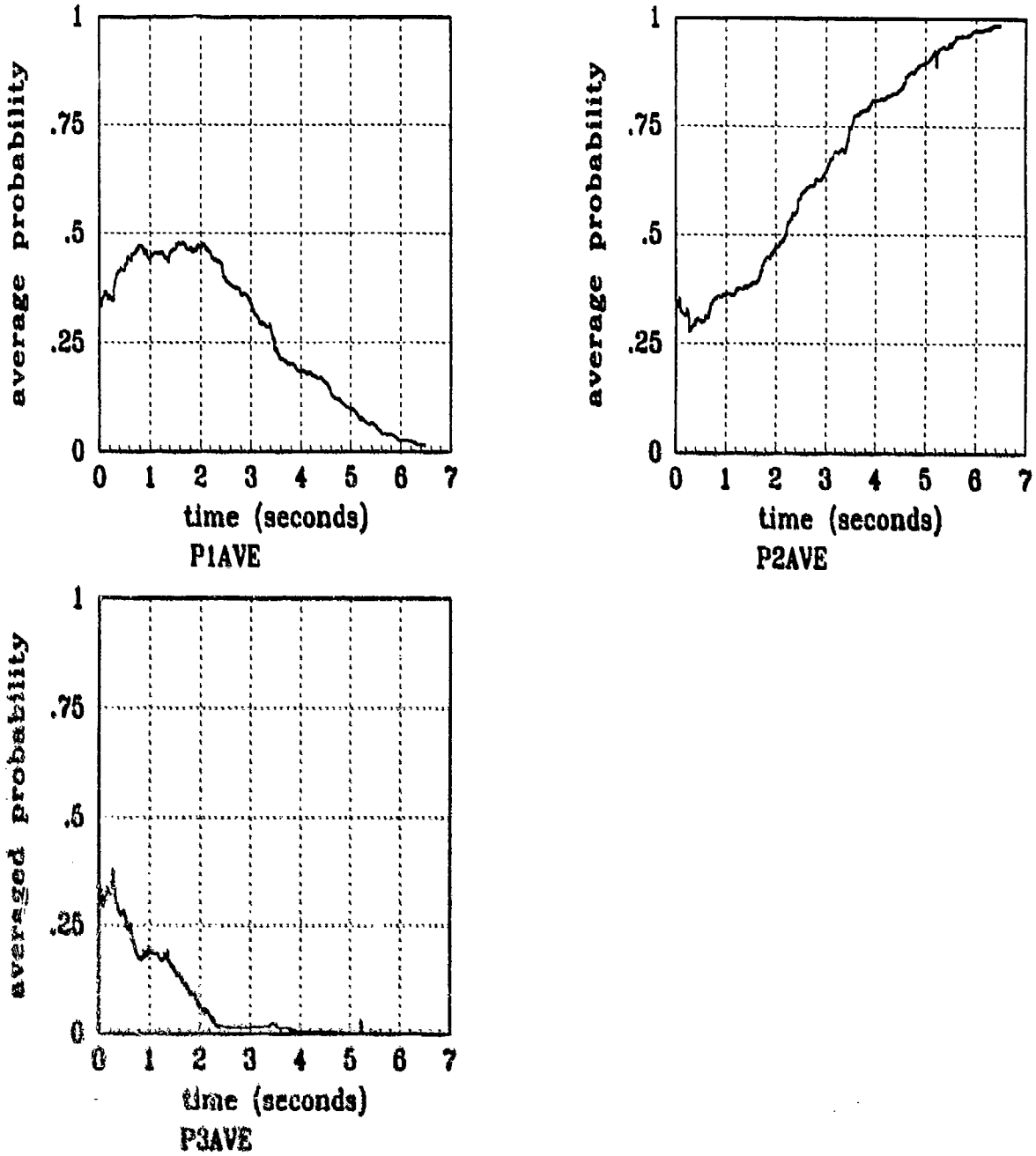


Figure I.8 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t_i)$
(Truth = $0.784/2\pi$)

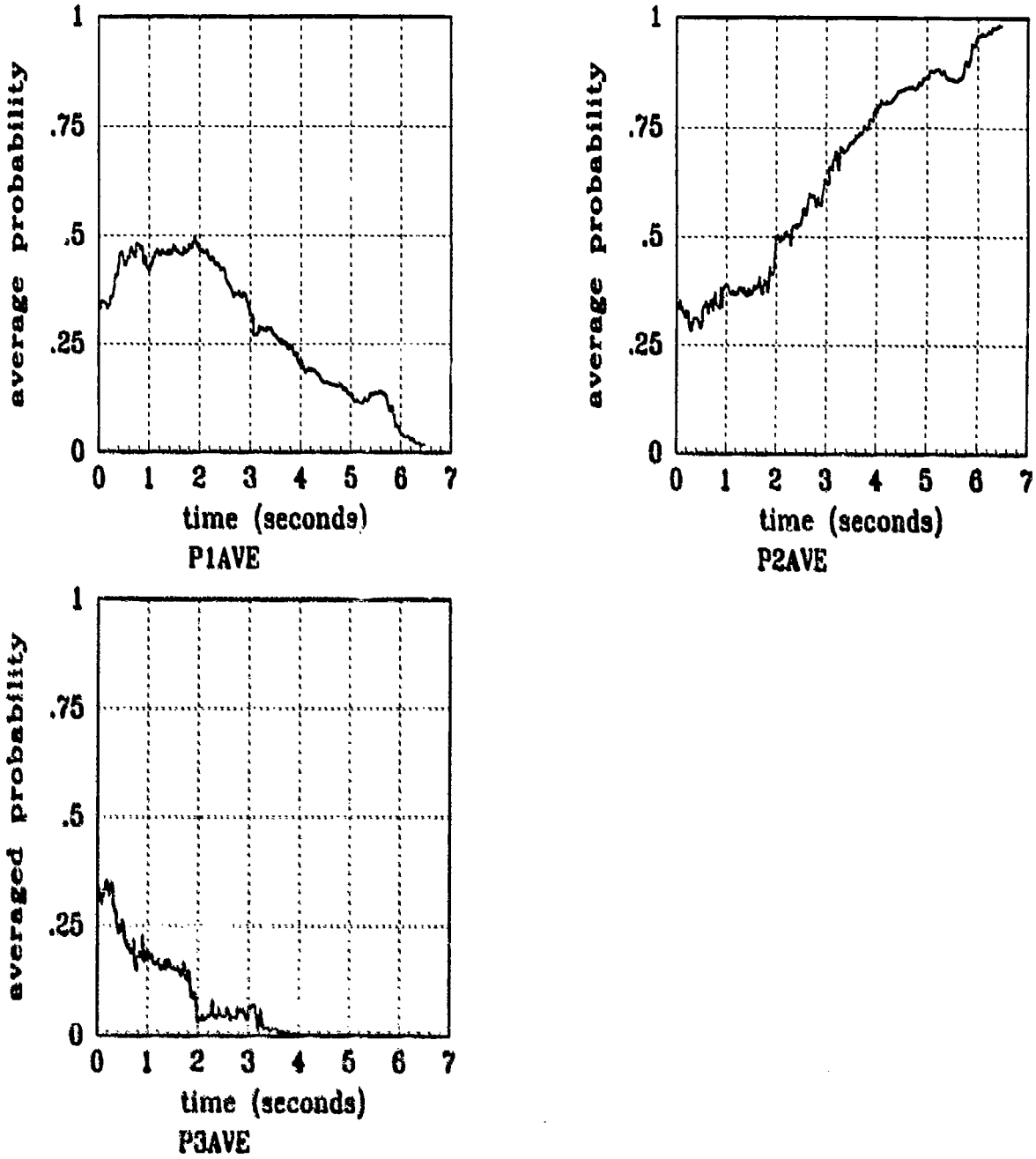


Figure I.9 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$
(Truth = 0.784/9.5π)

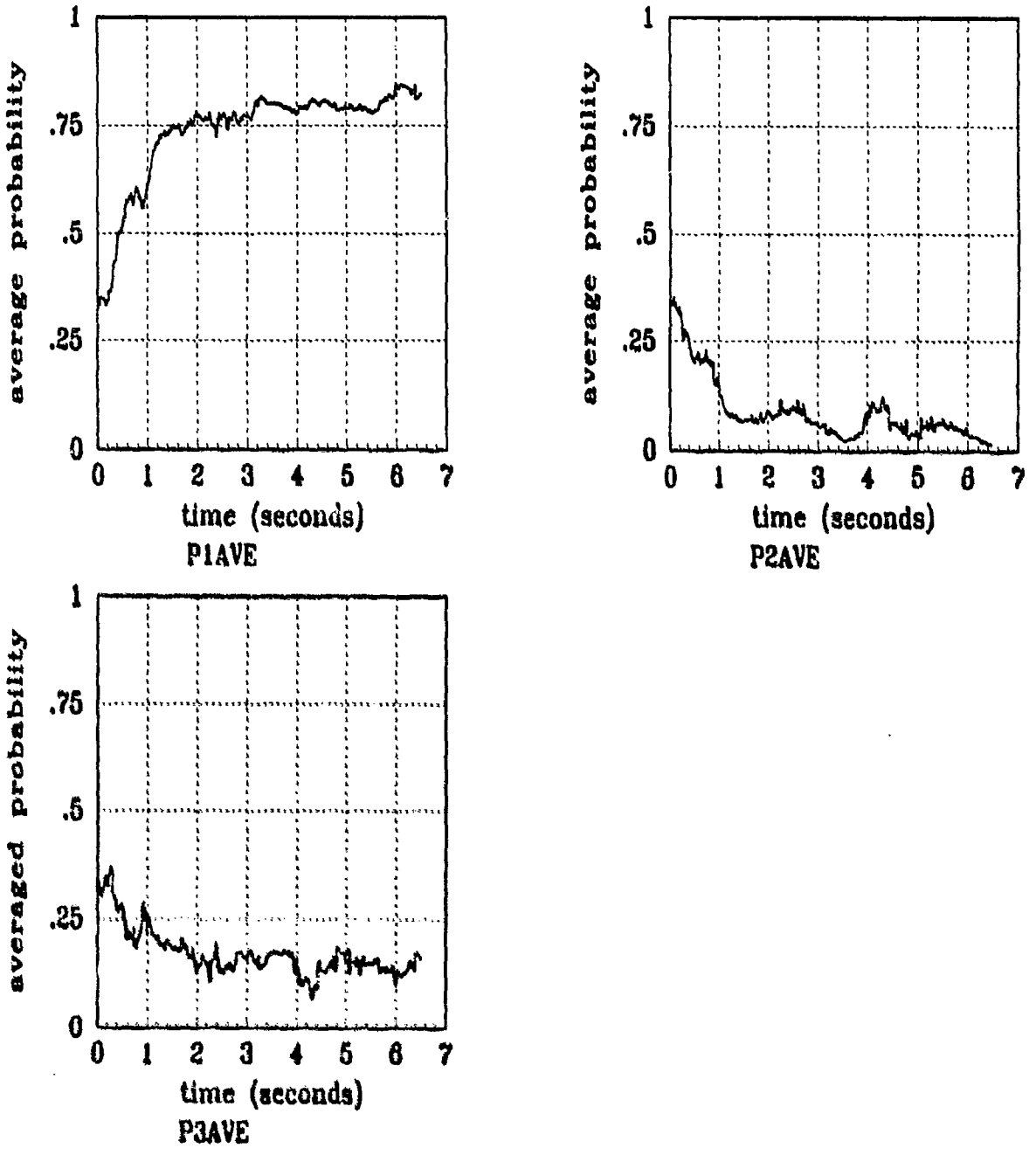


Figure I.10 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t_i)$
(Truth = 0.784/10.5π)

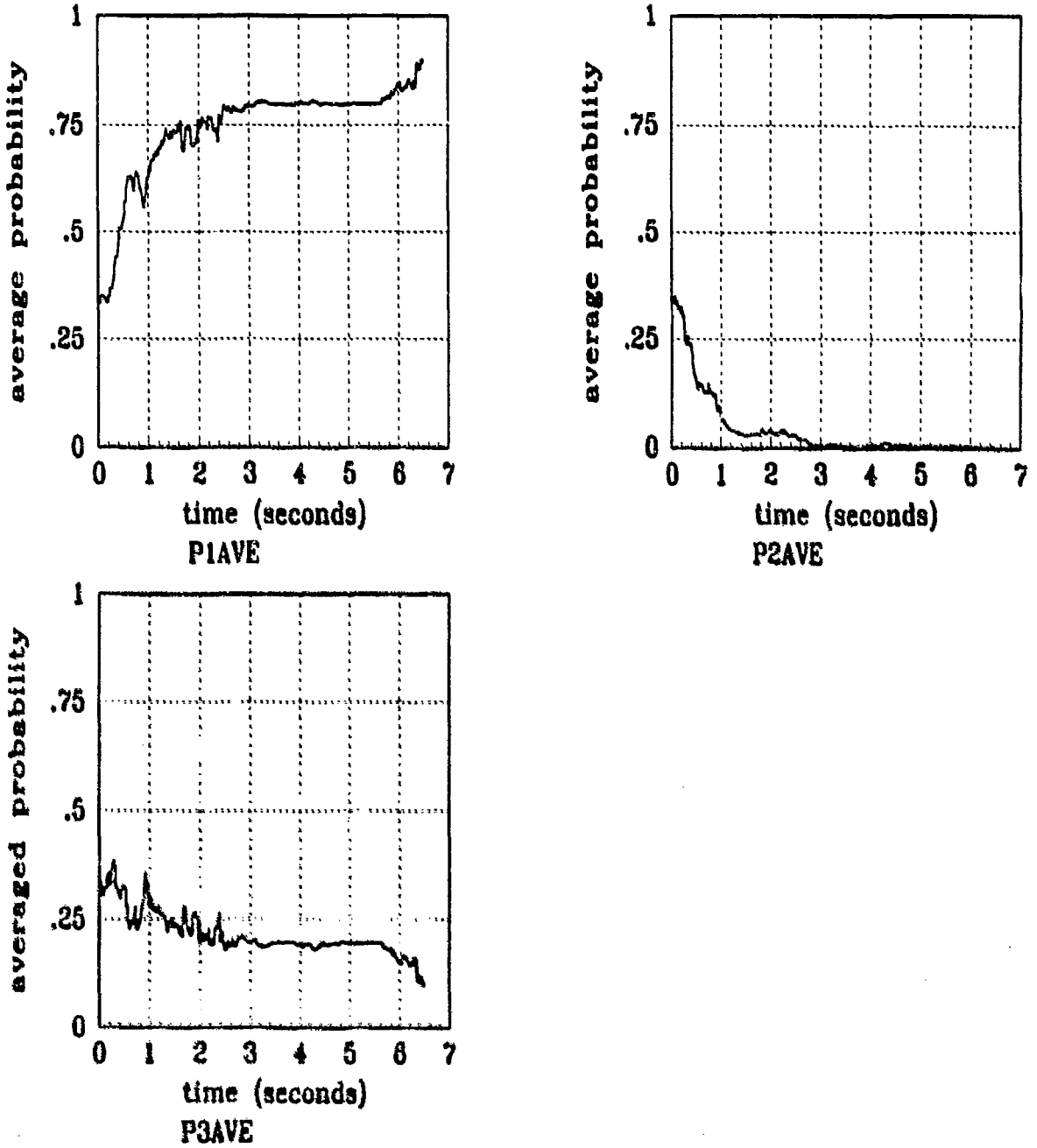


Figure I.11 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_2(t_i)$
(Truth = $0.784/11\pi$)

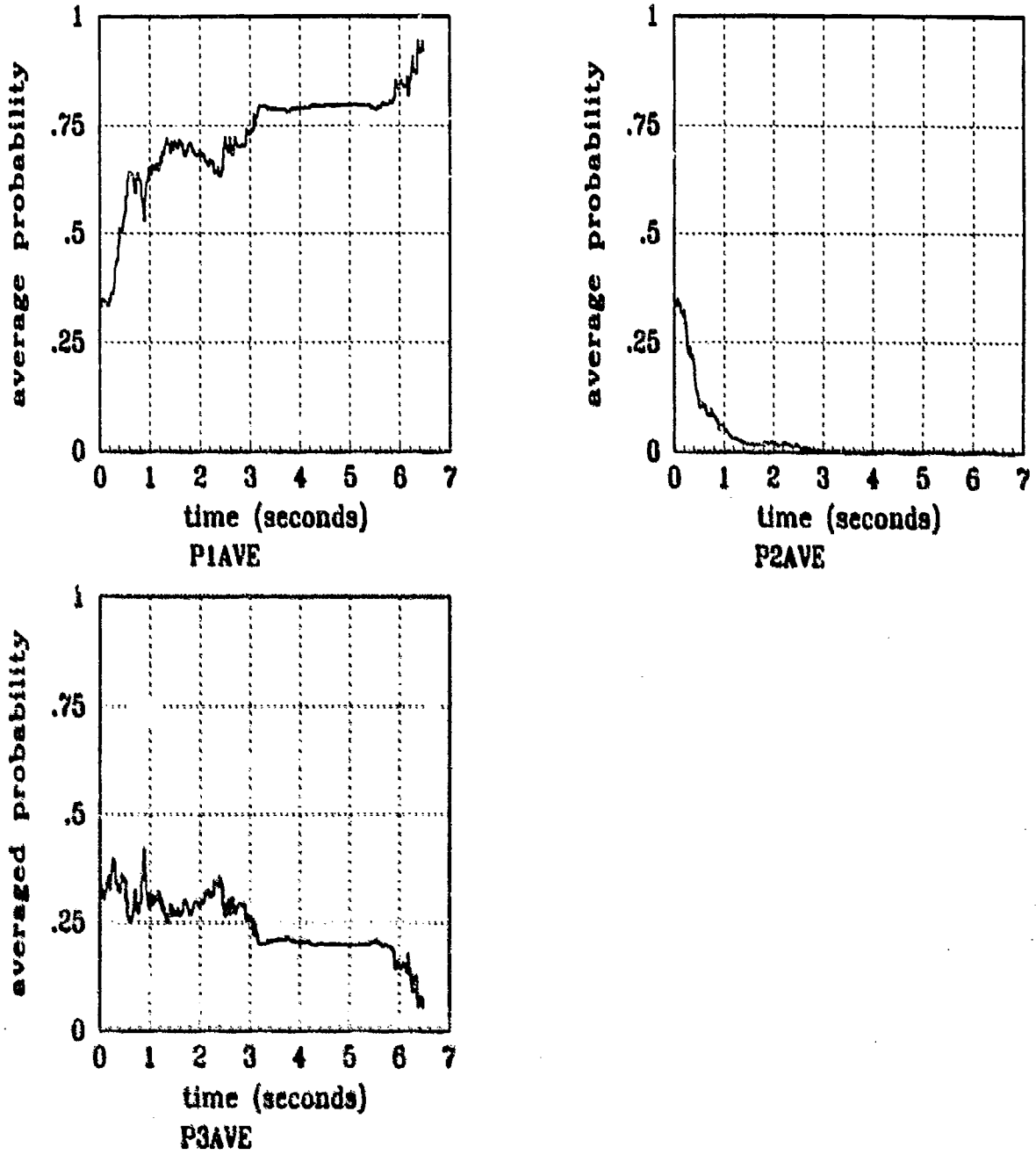


Figure I.12 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$
(Truth = $0.784/11.5\pi$)

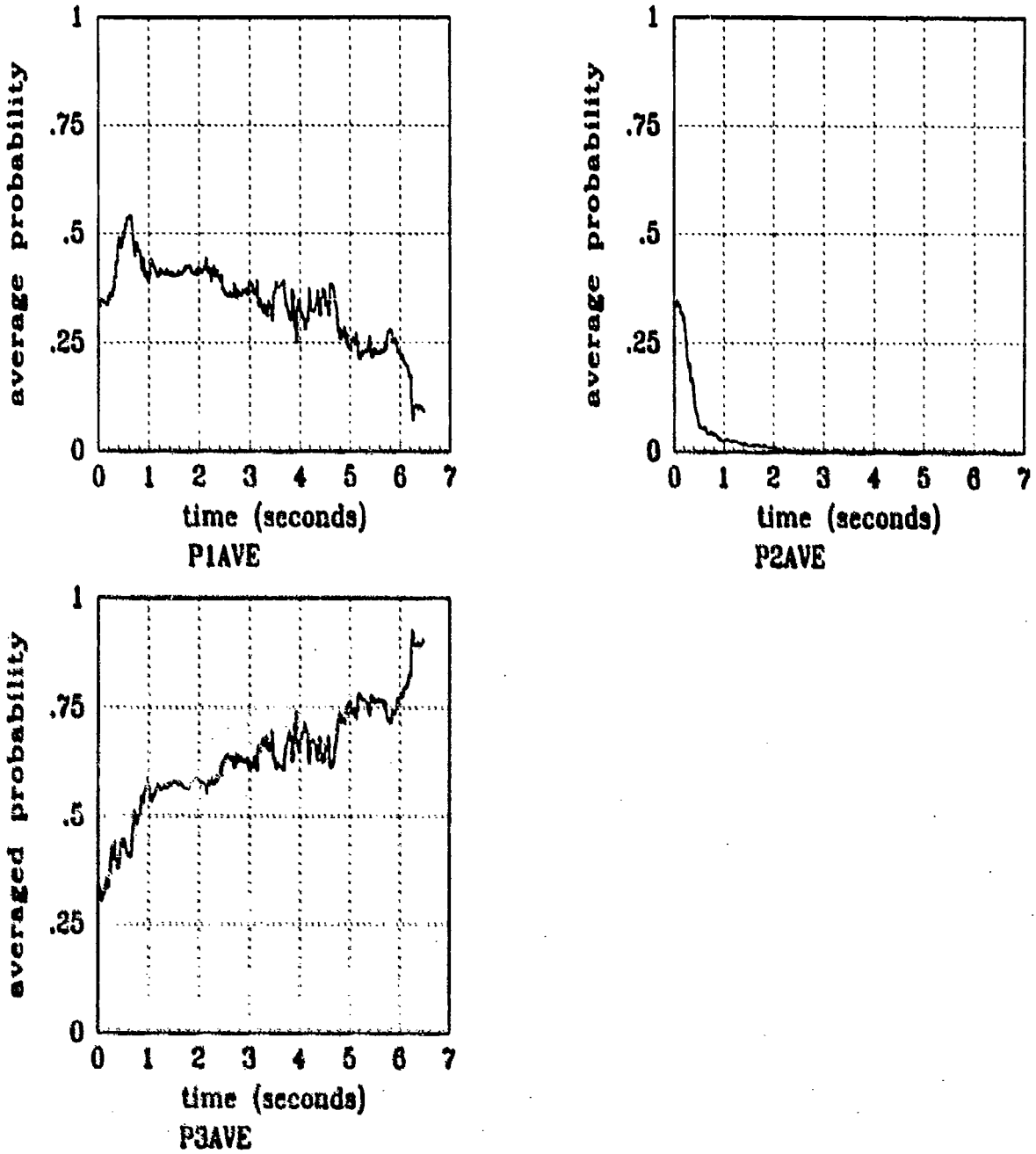


Figure I.13 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_s(t)$
(Truth = $0.784/12.5\pi$)

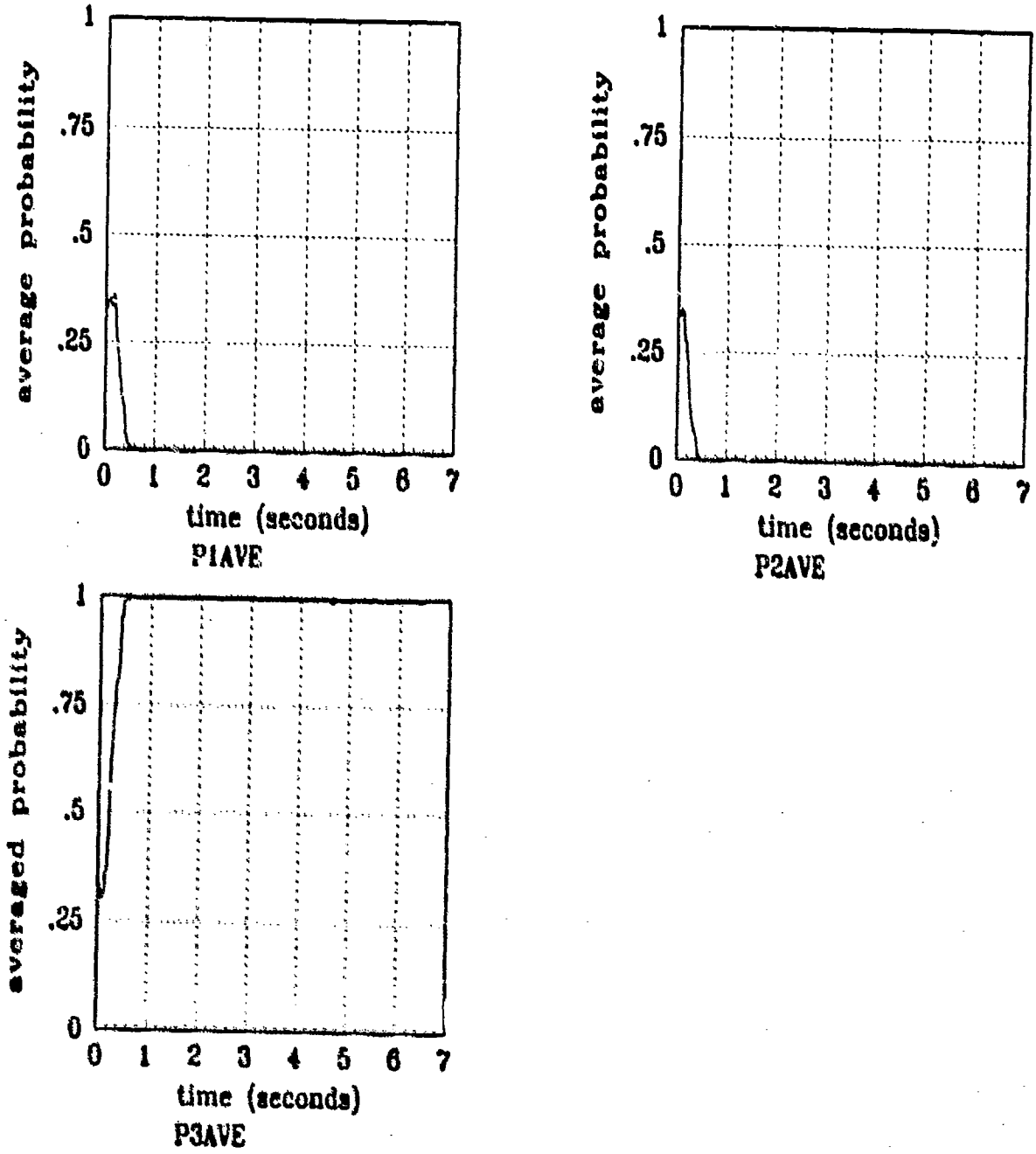


Figure I.14 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t_i)$
(Truth = $0.784/20\pi$)

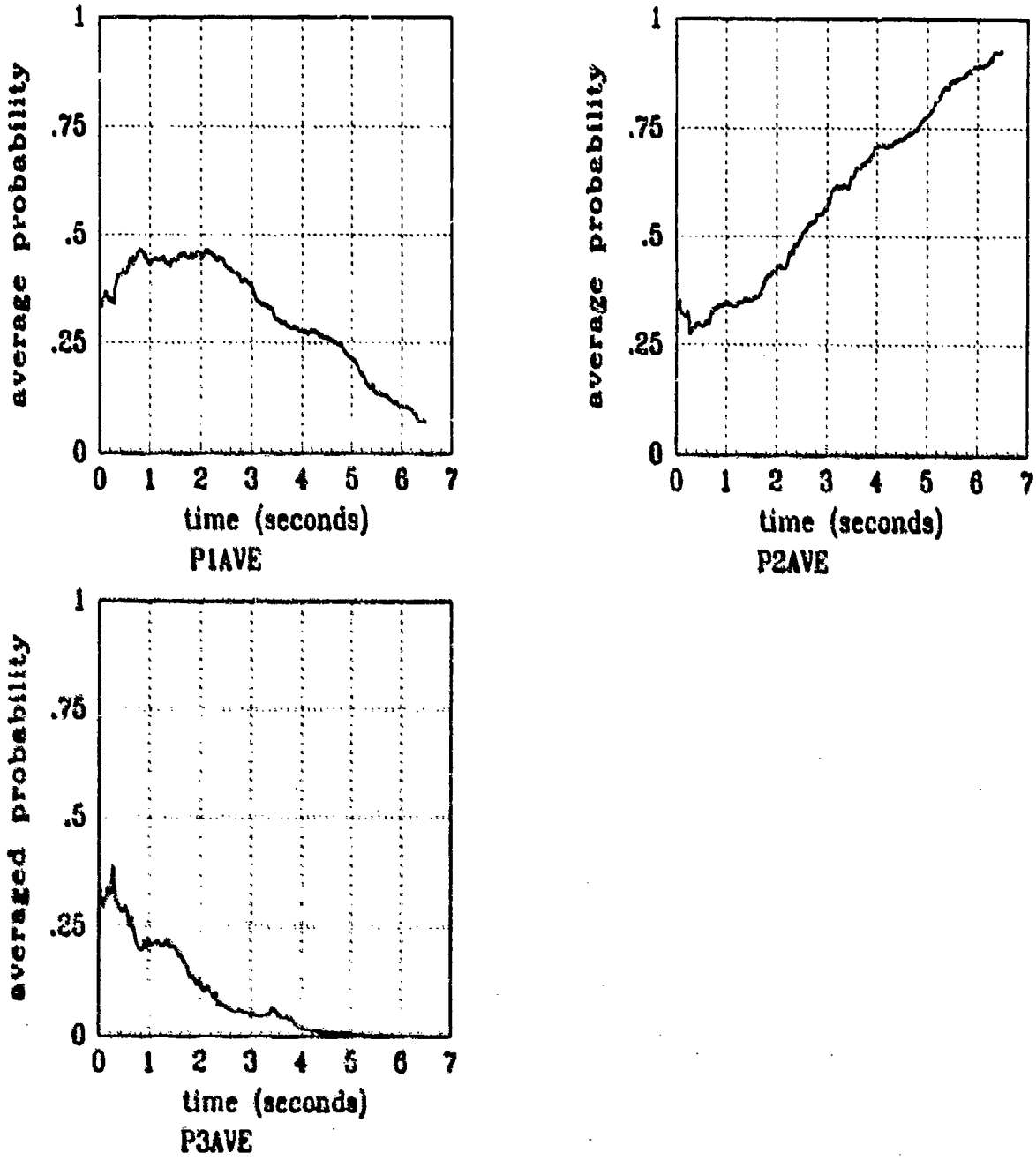


Figure I.15 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = $0.448/2\pi$)

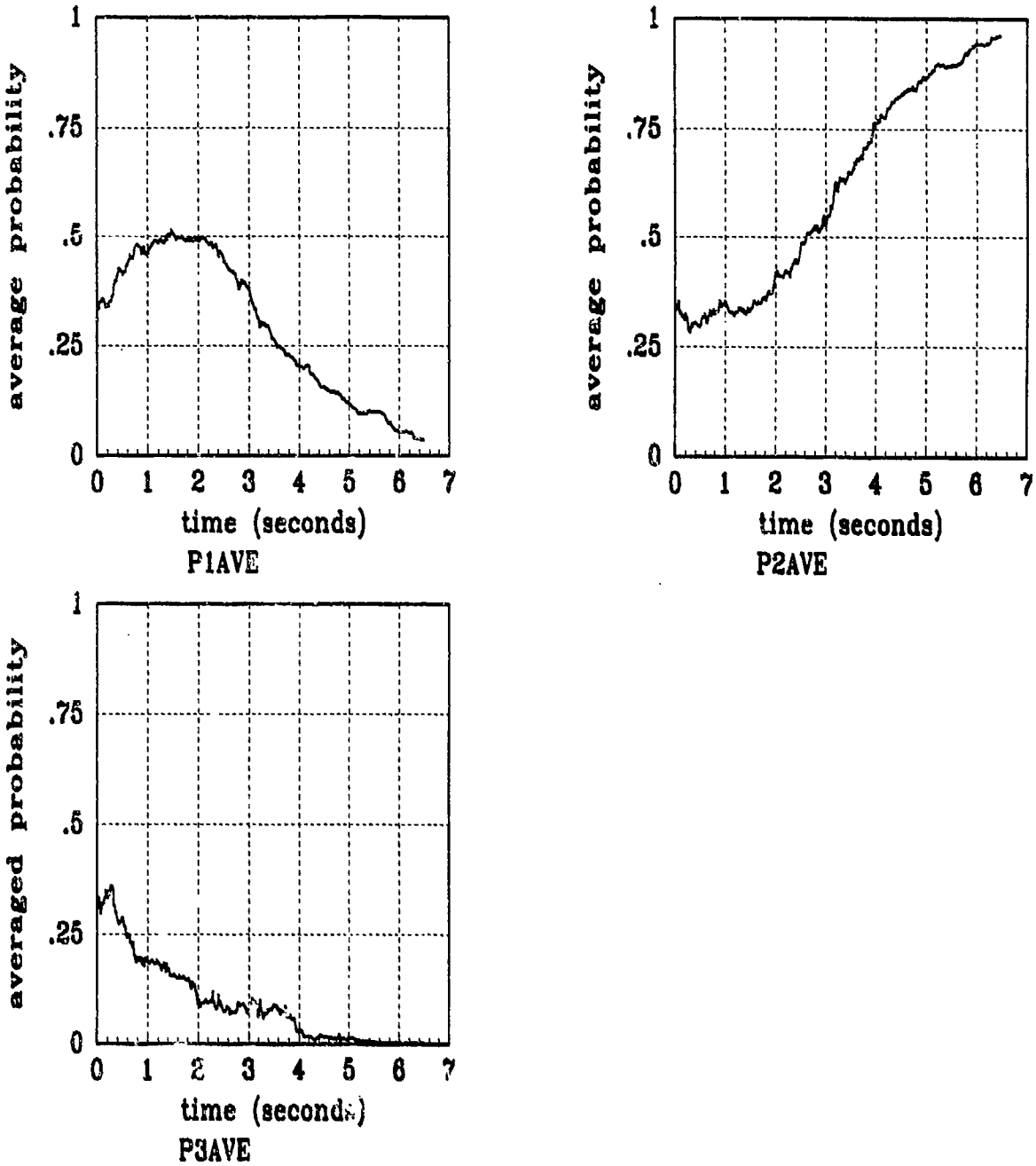


Figure I.16 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.448/9.5π)

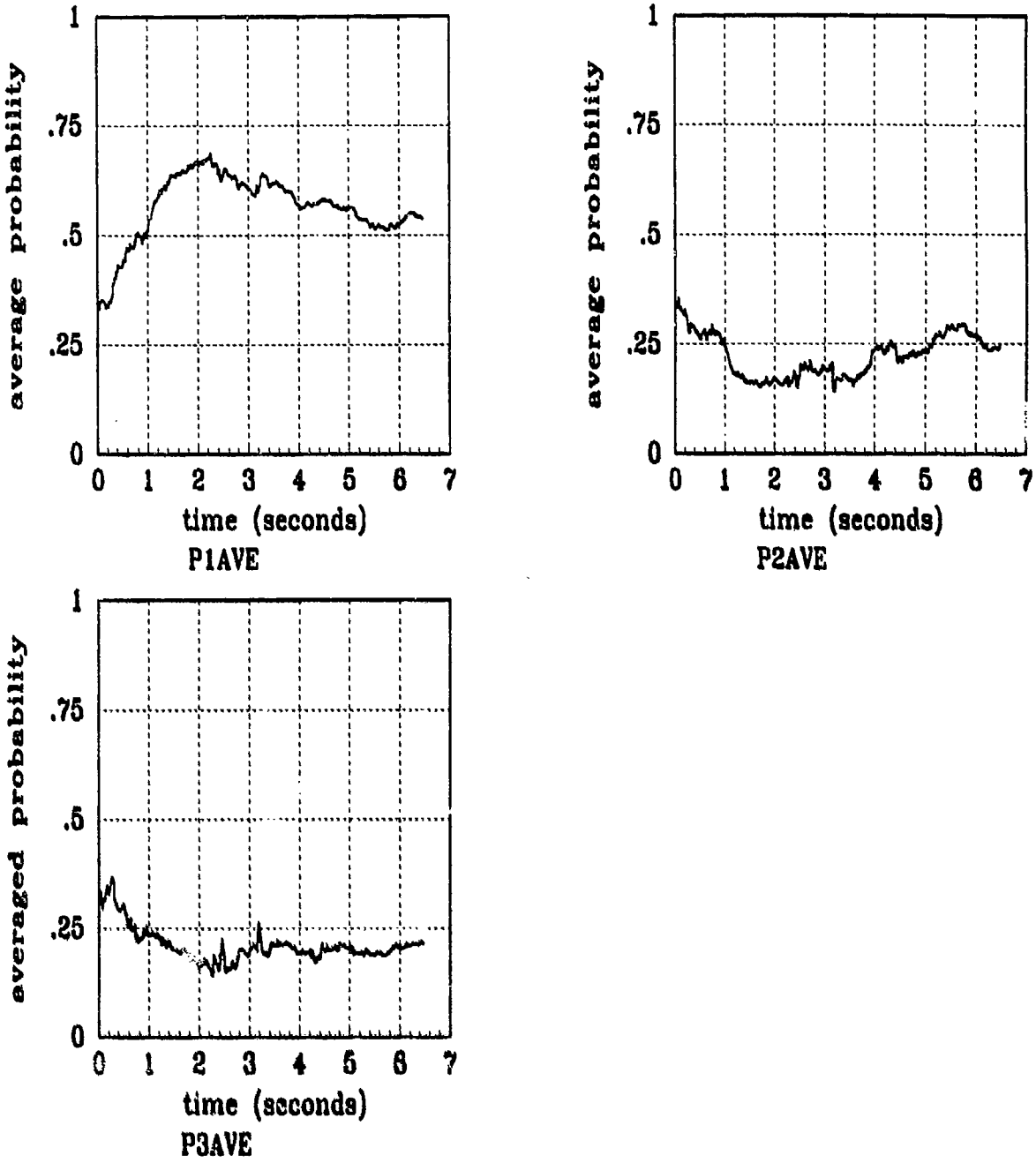


Figure I.17 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = 0.448/10.5π)

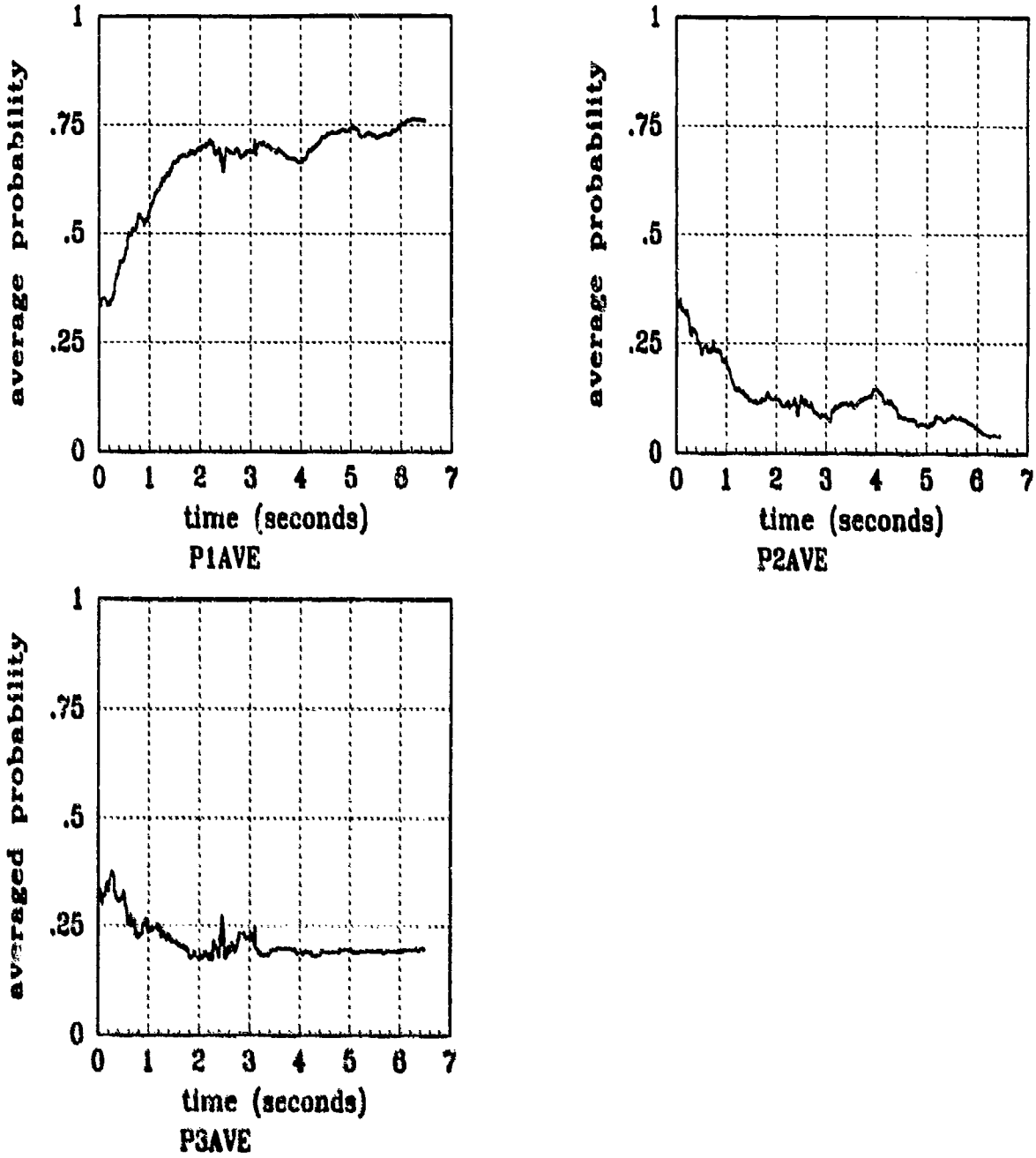


Figure I.18 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$
(Truth = $0.448/11\pi$)

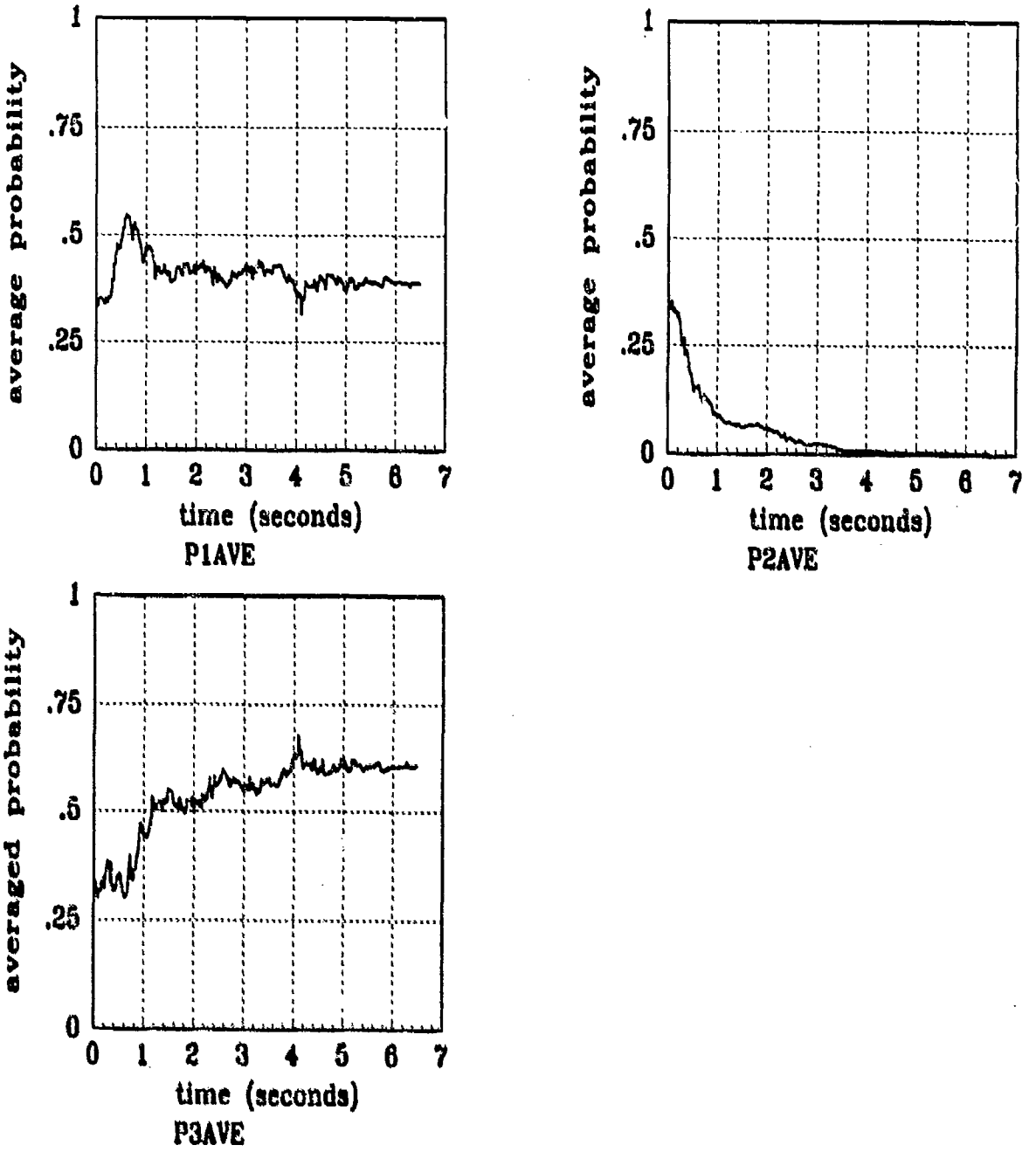


Figure I.19 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_s(t_i)$
(Truth = 0.448/12.5π)

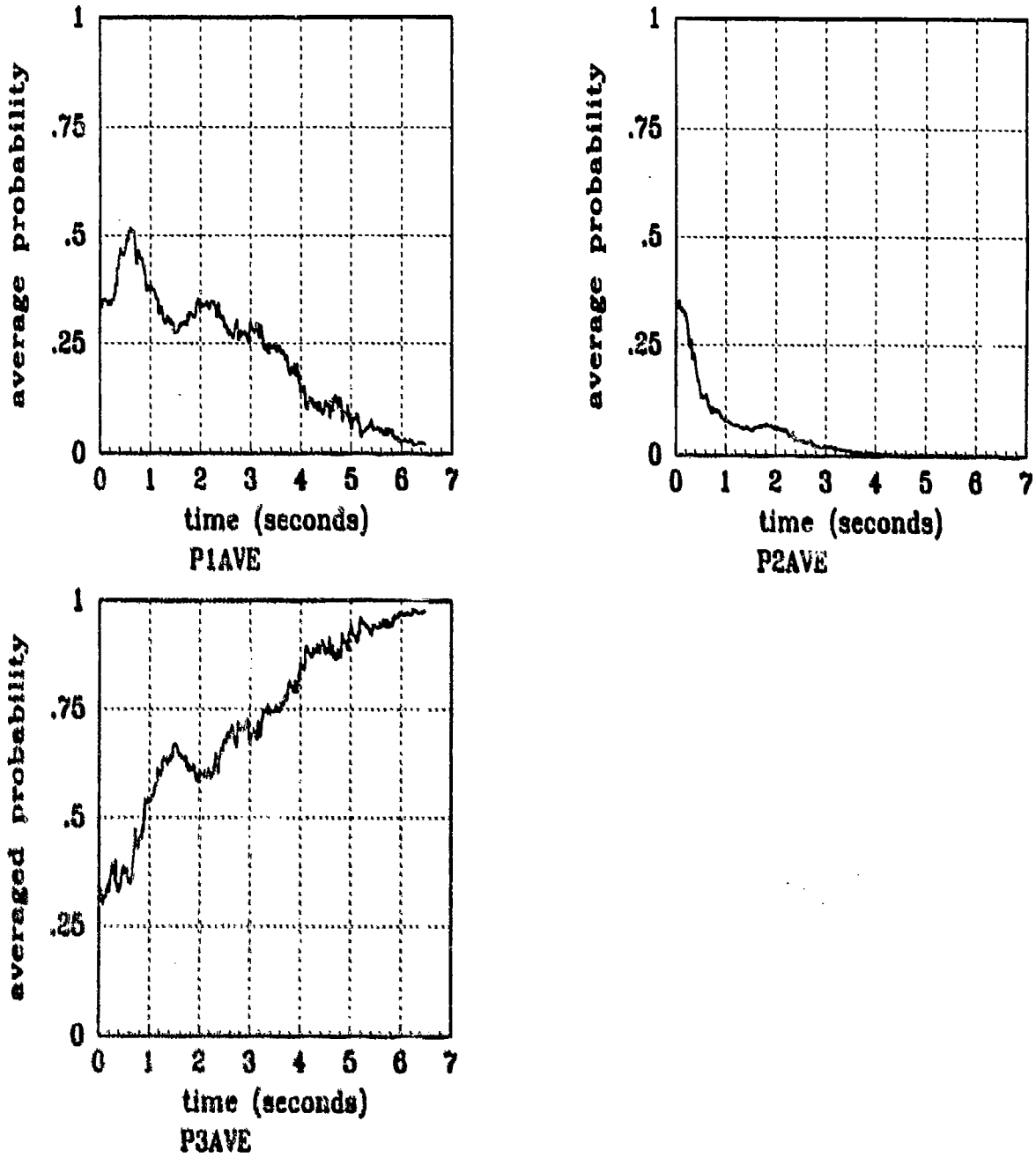


Figure I.20 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t)$
(Truth = $0.448/13\pi$)

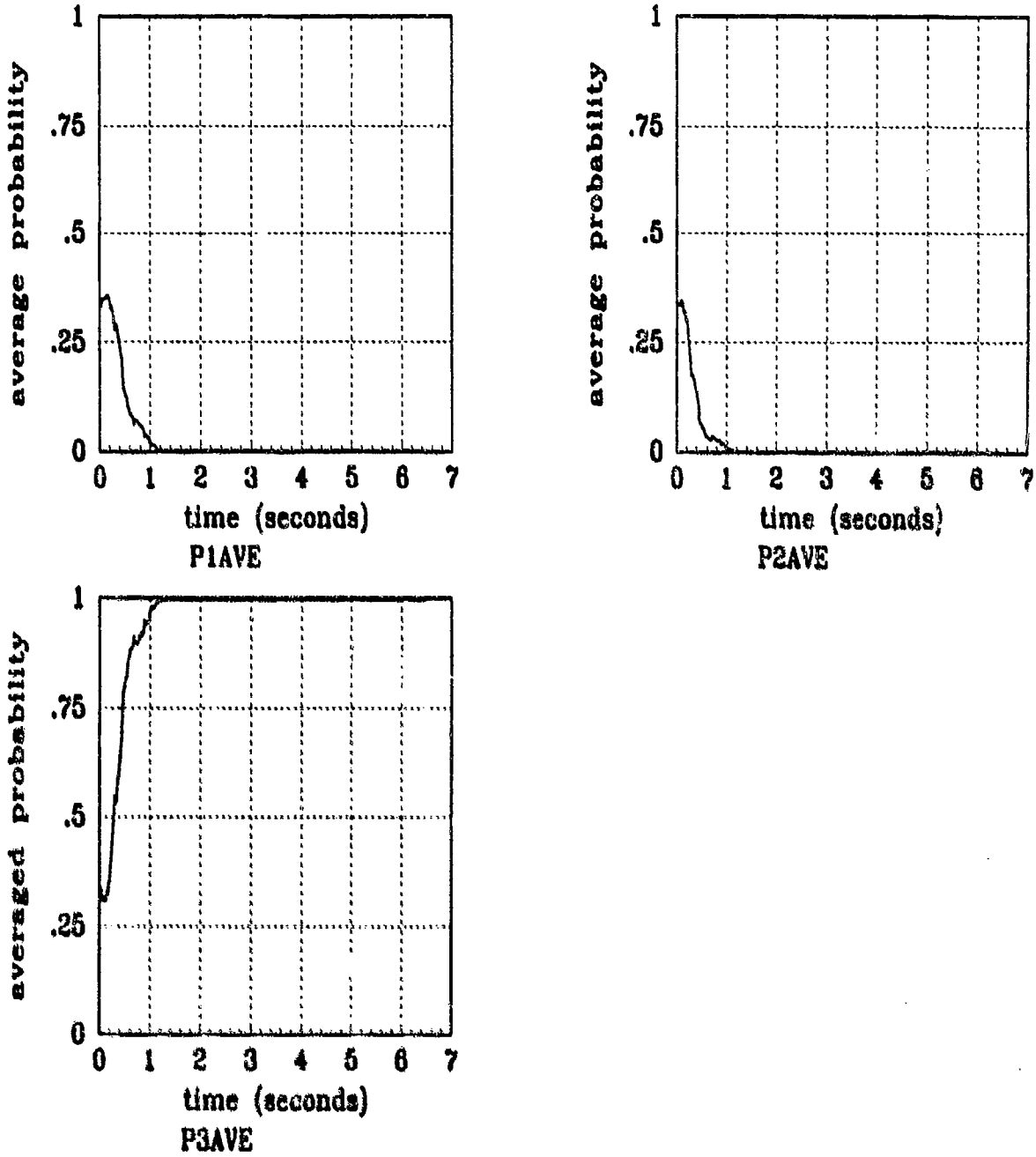
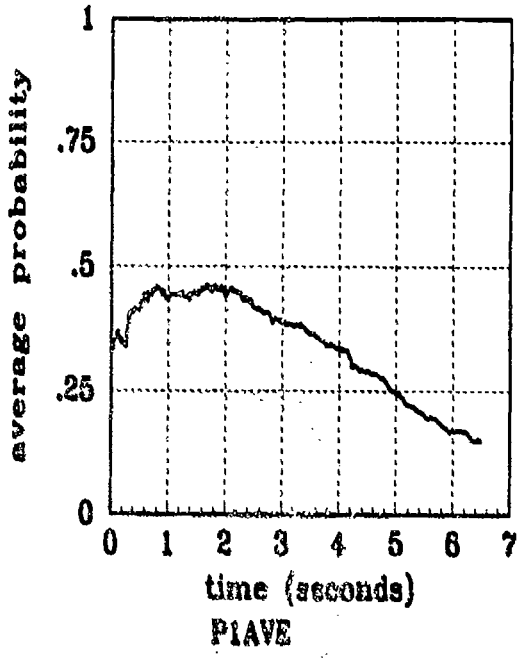
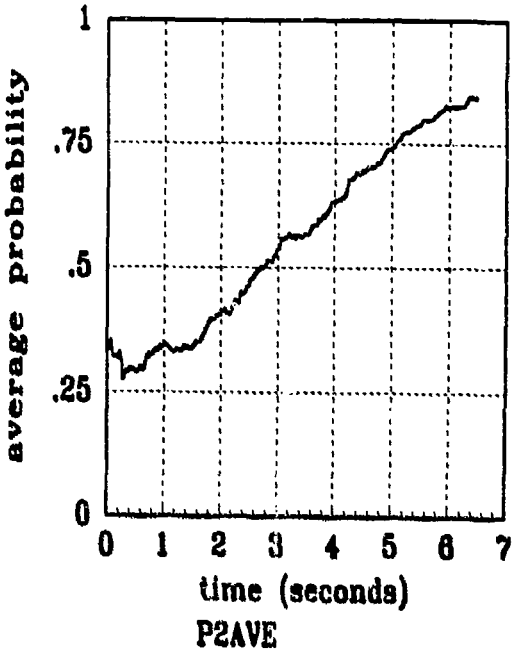


Figure I.21 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t_i)$ (Truth = $0.448/20\pi$)





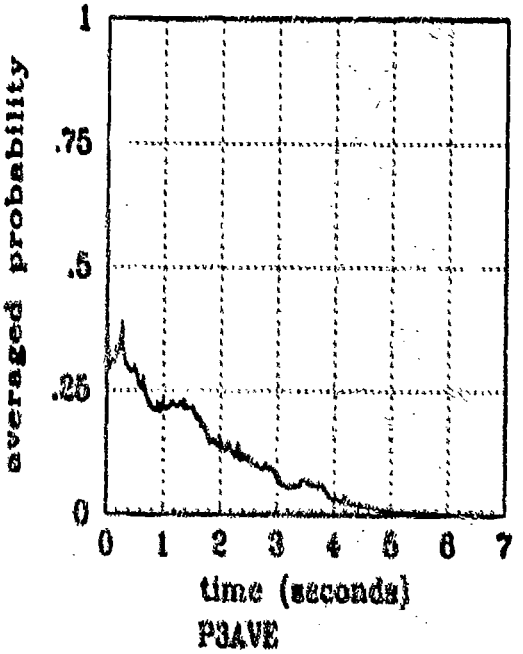


Figure I.22 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = $0.112/2\pi$)

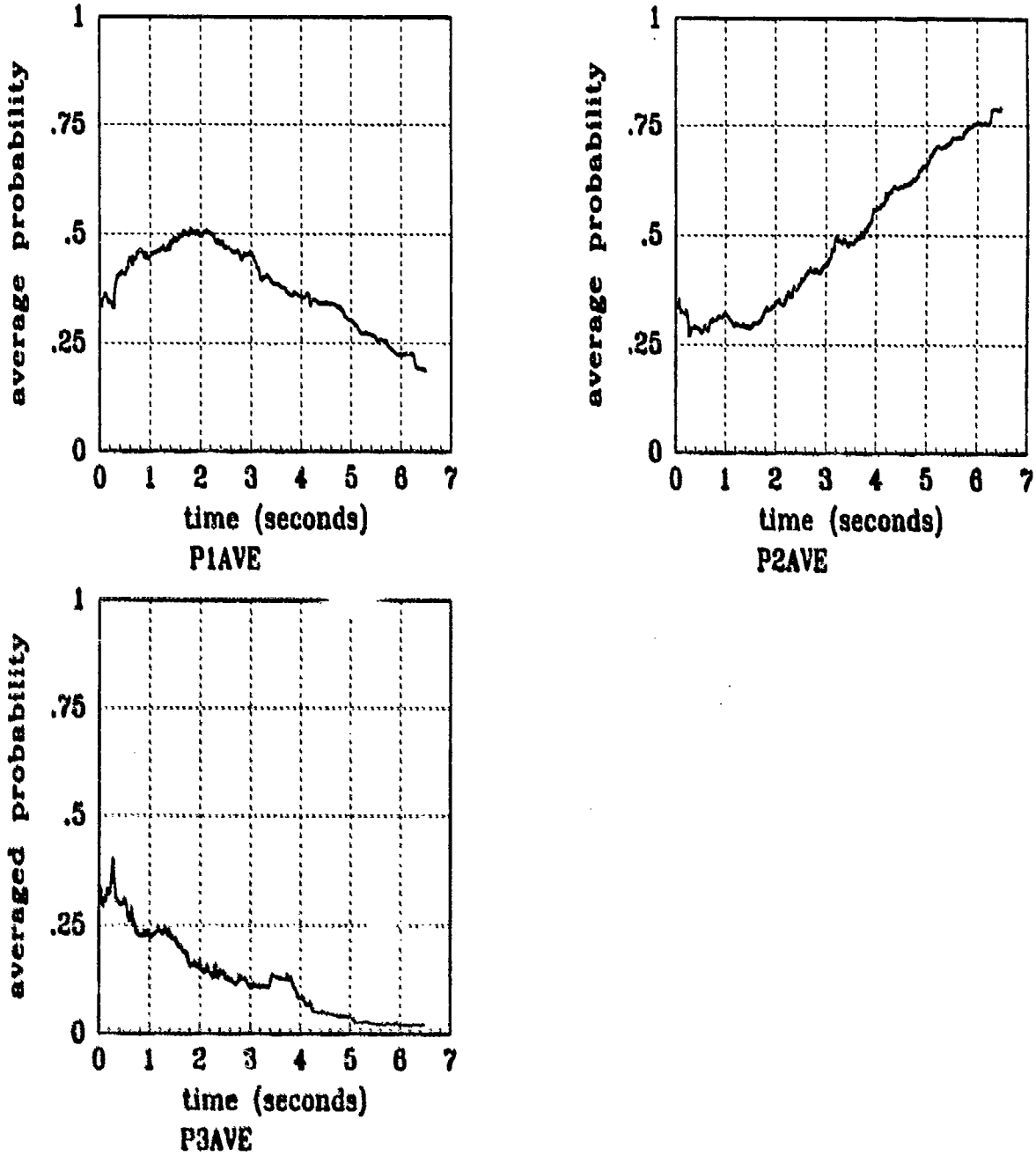


Figure I.23 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t_i)$ (Truth = $0.112/11\pi$)

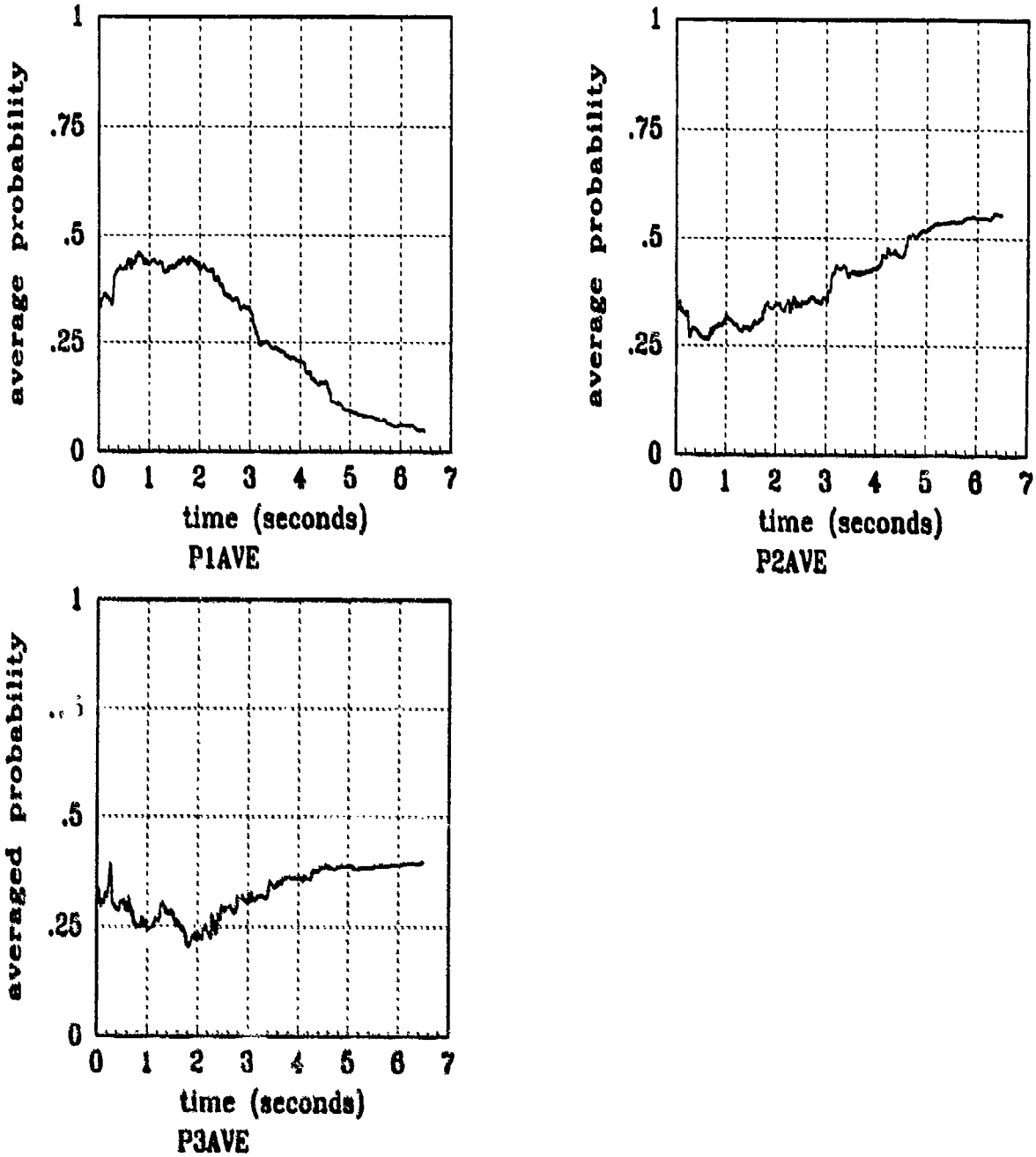


Figure I.24 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = $0.112/20\pi$)

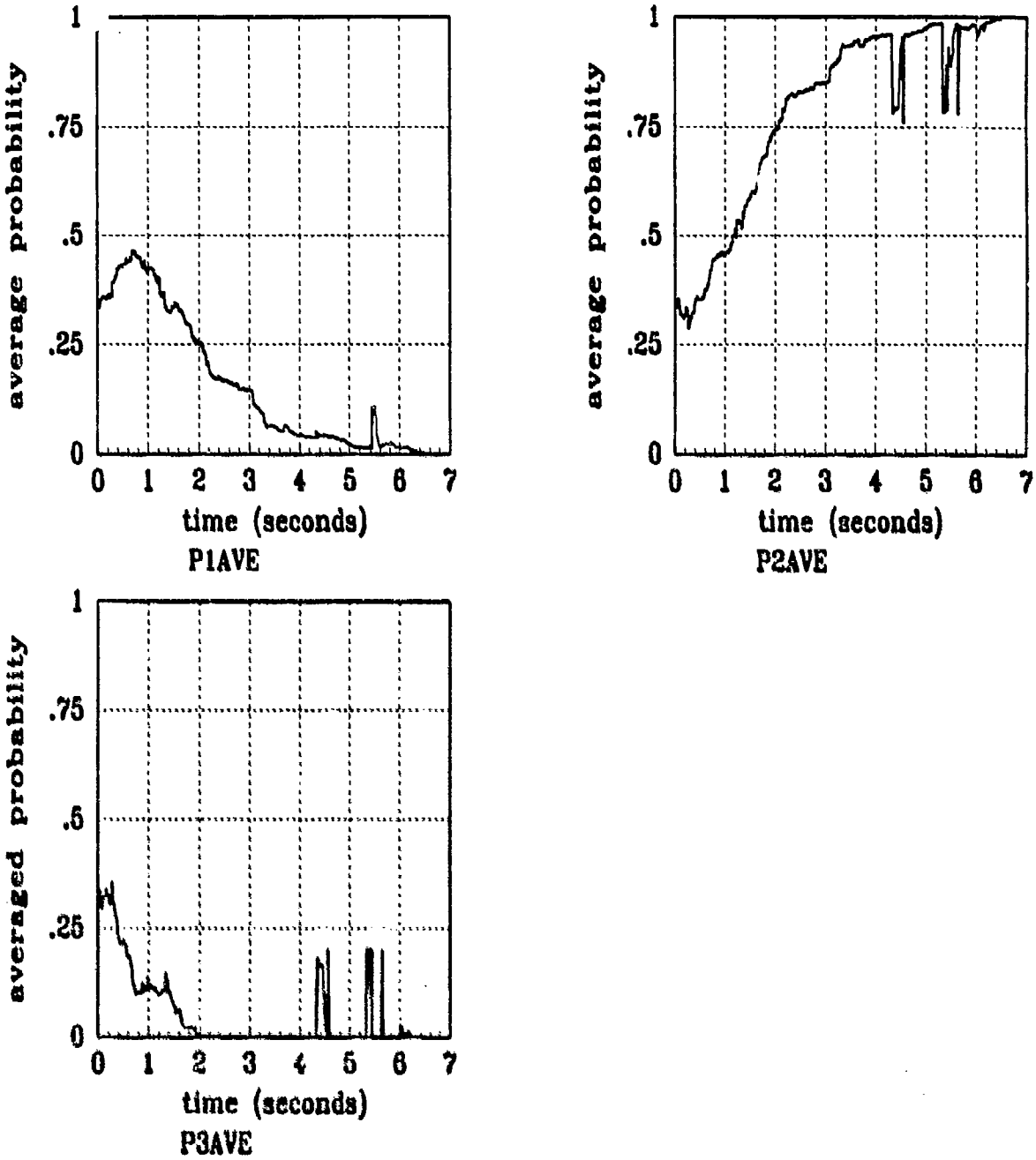


Figure I.25 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t_i)$
(Truth = $2.24/2\pi$)

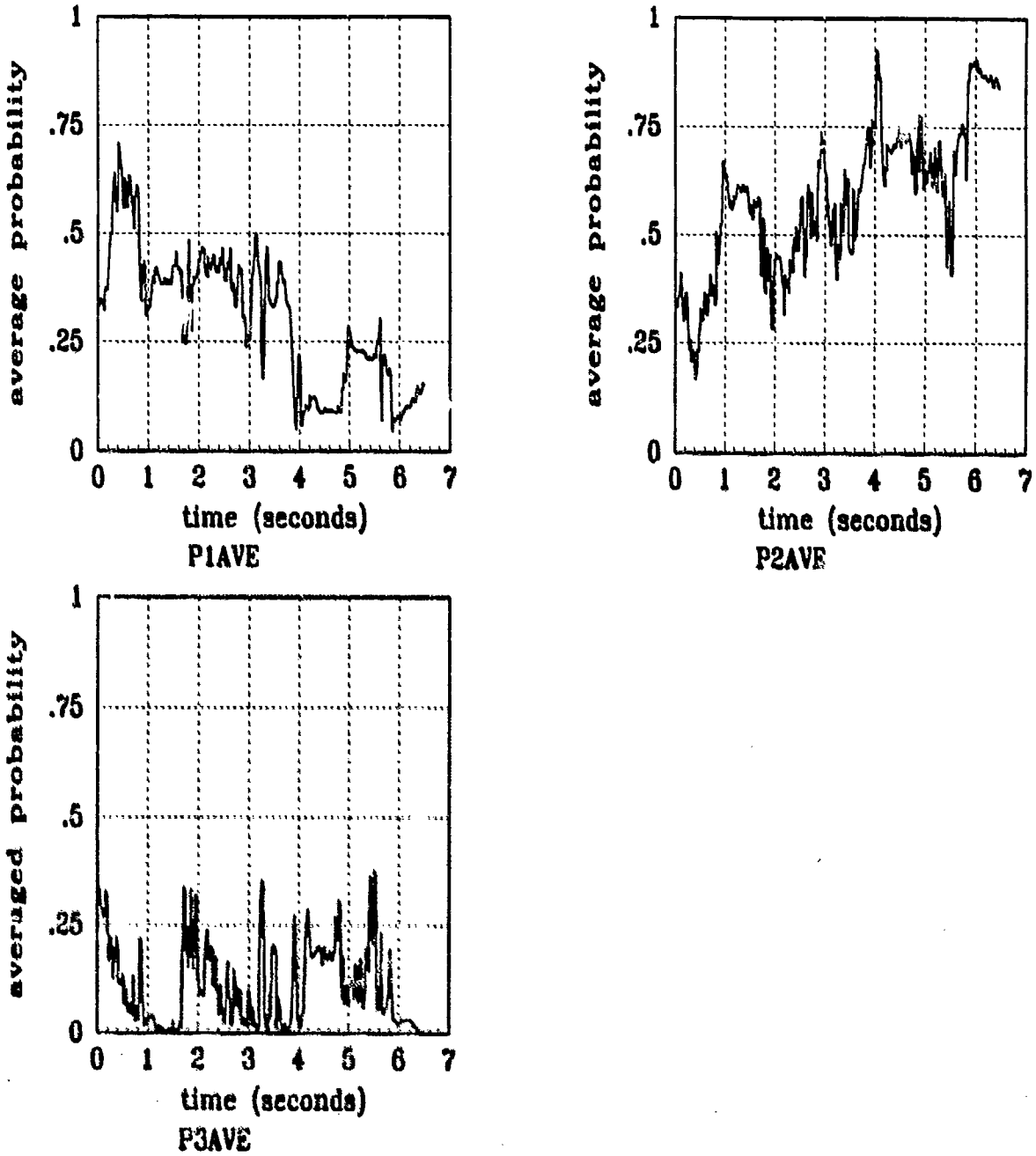


Figure I.26 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t)$
(Truth = 2.24/9.5π)

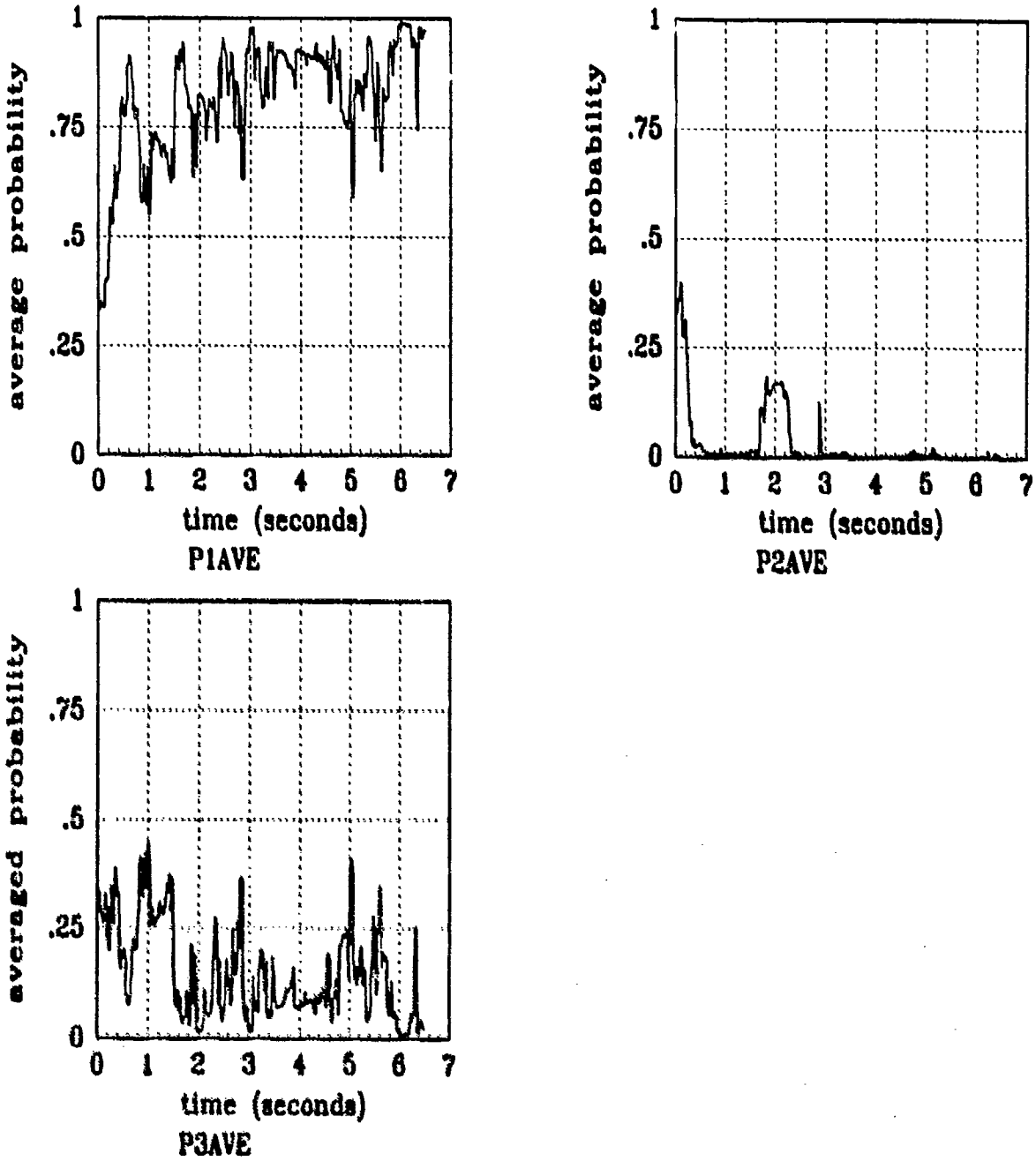


Figure I.27 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$
(Truth = $2.24/11\pi$)

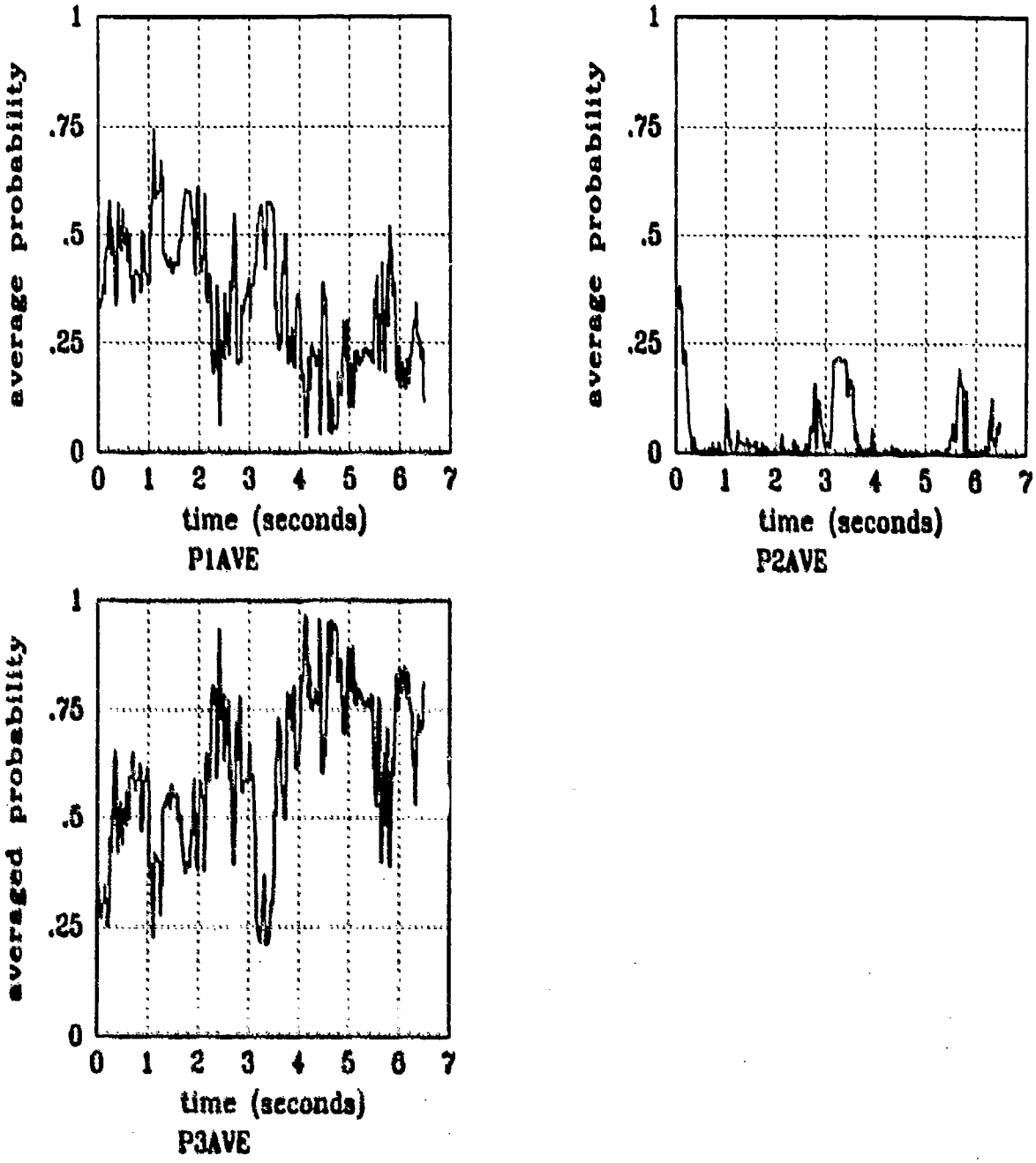
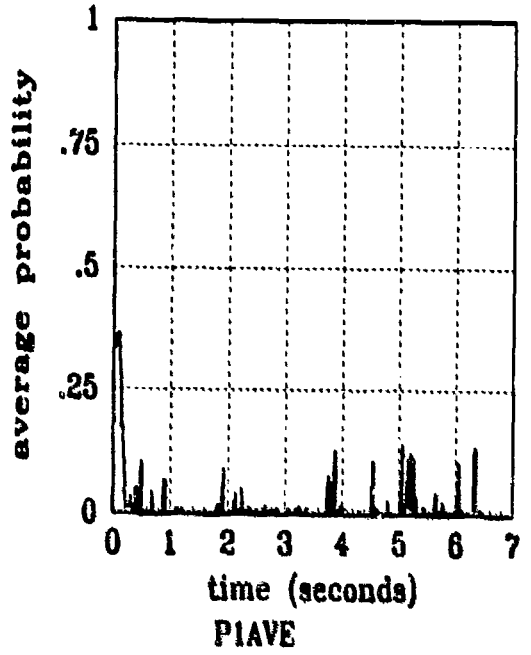
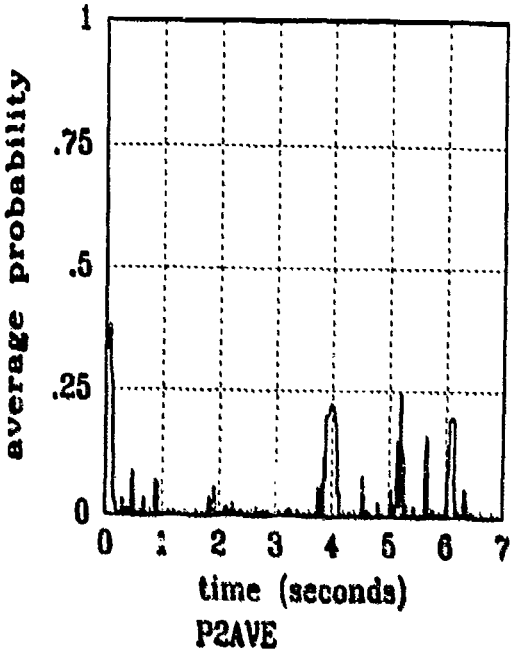


Figure I.28 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t)$
(Truth = 2.24/12.5π)





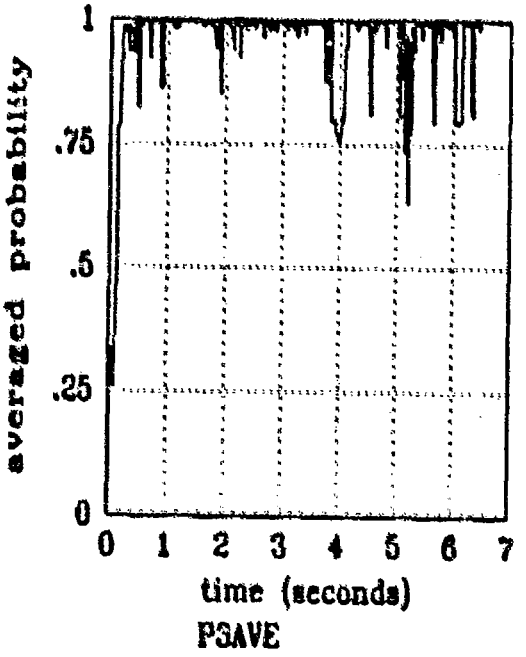


Figure I.29 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = $2.24/20\pi$)

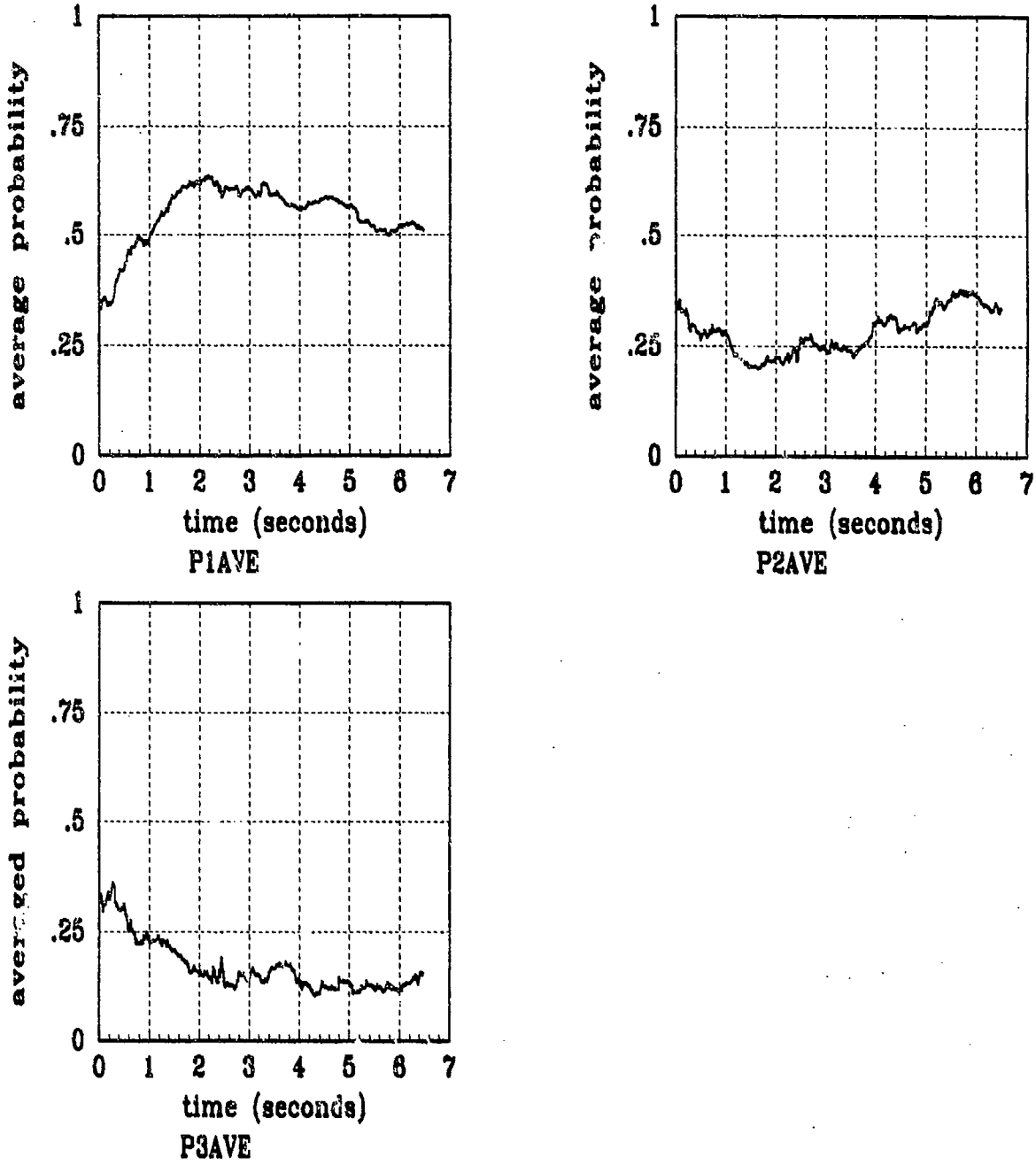


Figure I.30 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = $0.280/11\pi$)

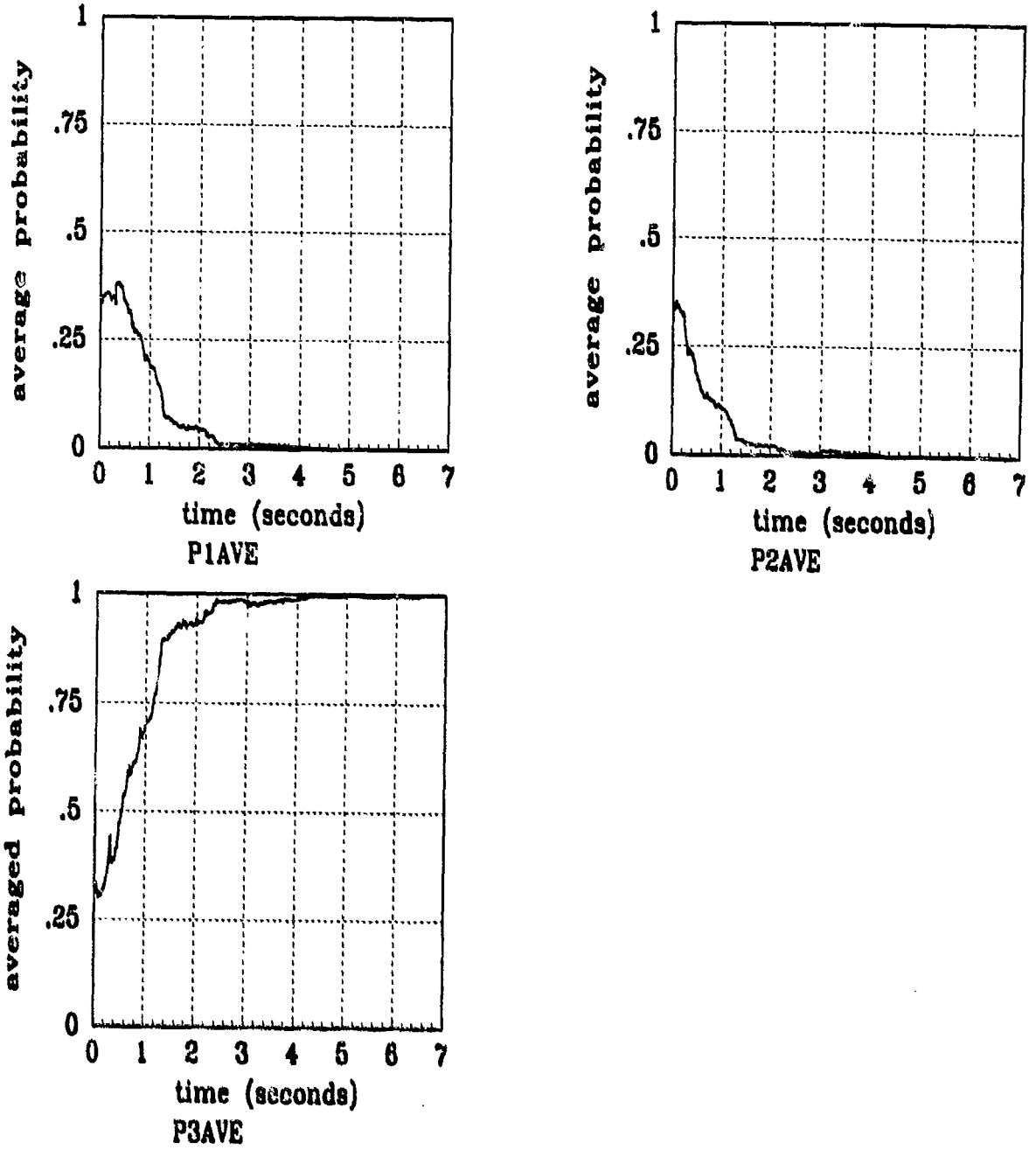


Figure I.31 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$
(Truth = $0.28/20\pi$)

Appendix J

MMAE Configuration #4 Output Plots

This appendix contains the state and error statistics plots of the nine-state elemental filters. The data depicted in the two types of plots in this appendix are explained in *Appendix A*. The state comparison plots show the sample mean truth state over the 5 *Monte Carlo* runs compared to the same statistic for the filter estimate. The error statistics plots represent the error mean $\pm$ standard deviation values in pixels (or pixel/second for velocity and pogo velocity), of the errors between the filter estimate and true state; true mean $\pm$ one true standard deviation are plotted, along with zero $\pm$ one filter-computed standard deviation.

Note the filter covariance calculation is shown in Equation (4.5)..

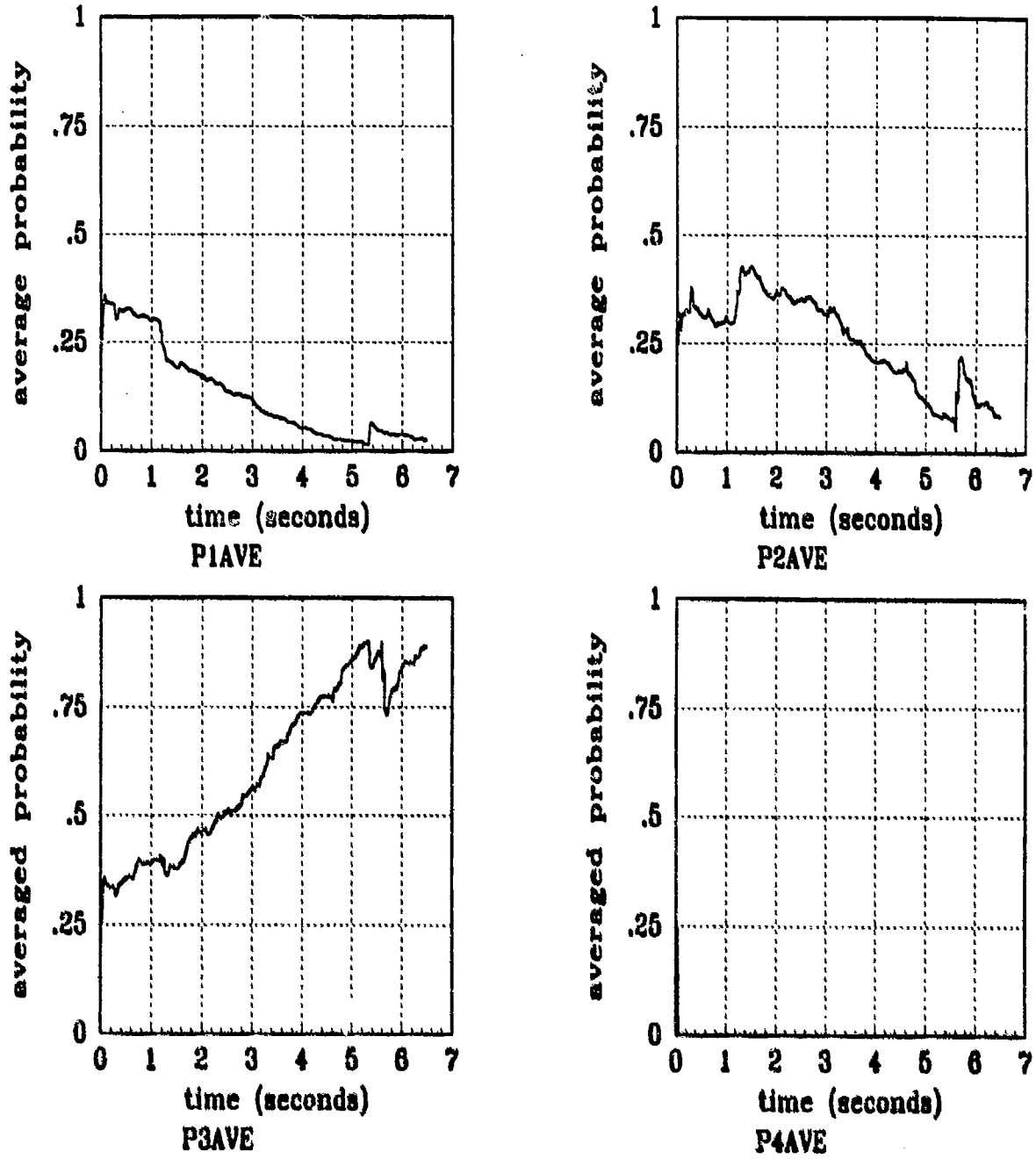


Figure J.1 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t_i)$
(Truth = $1.12/2\pi$)

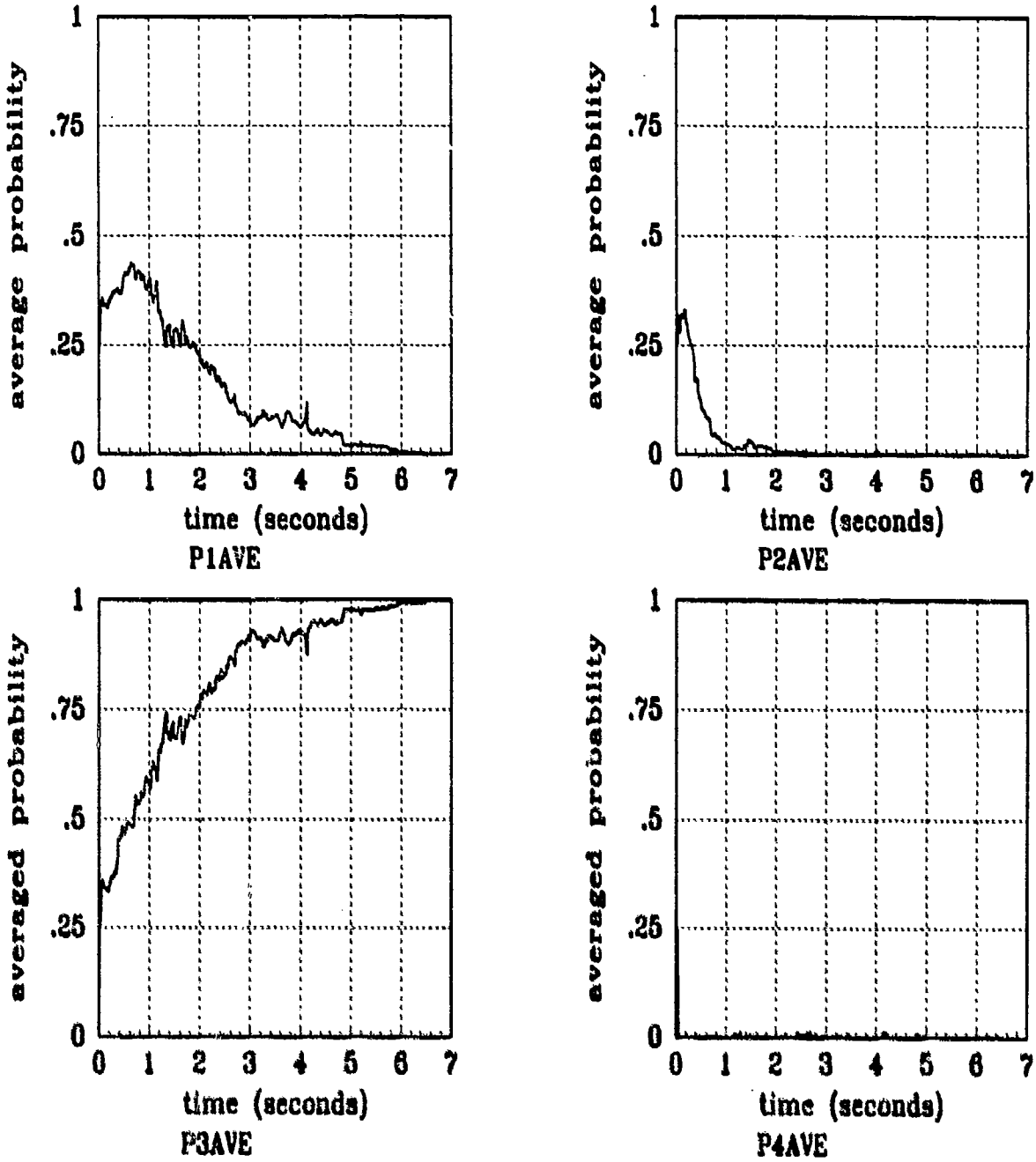


Figure J.2 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$
(Truth = $1.12/8\pi$)

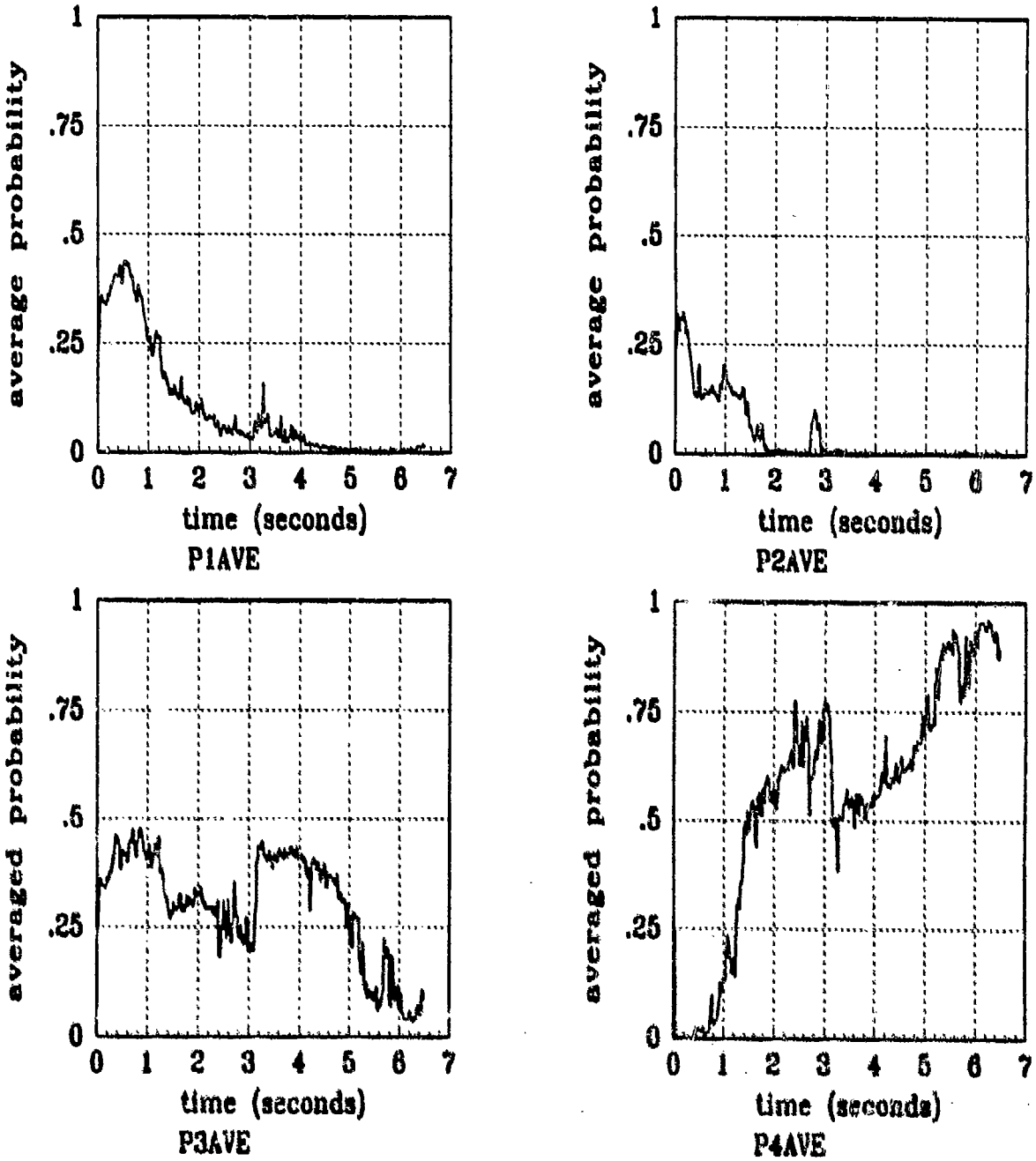


Figure J.3 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_x(t_i)$
(Truth = $1.12/11\pi$)

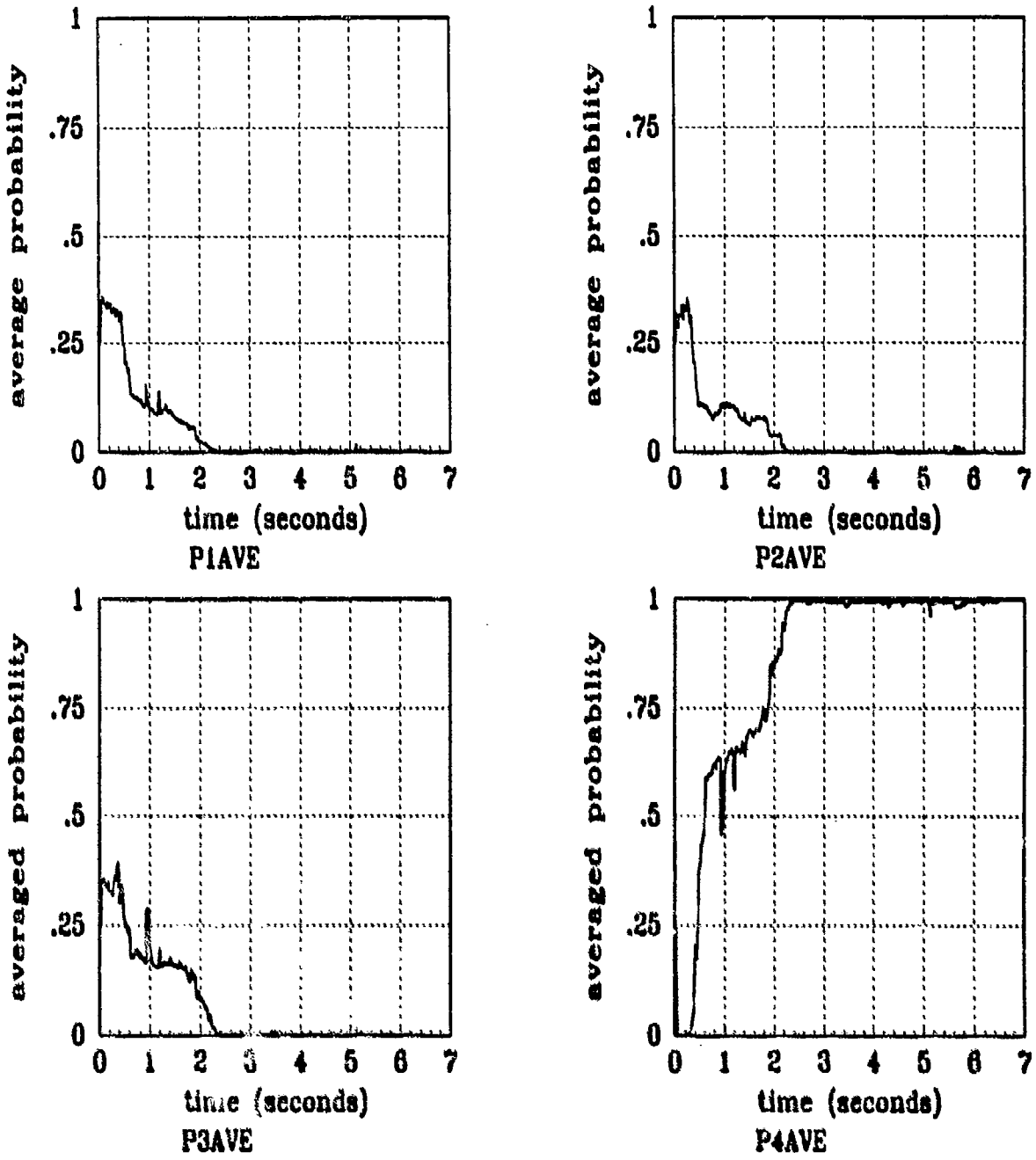


Figure J.4 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t_i)$
(Truth = $1.12/14\pi$)

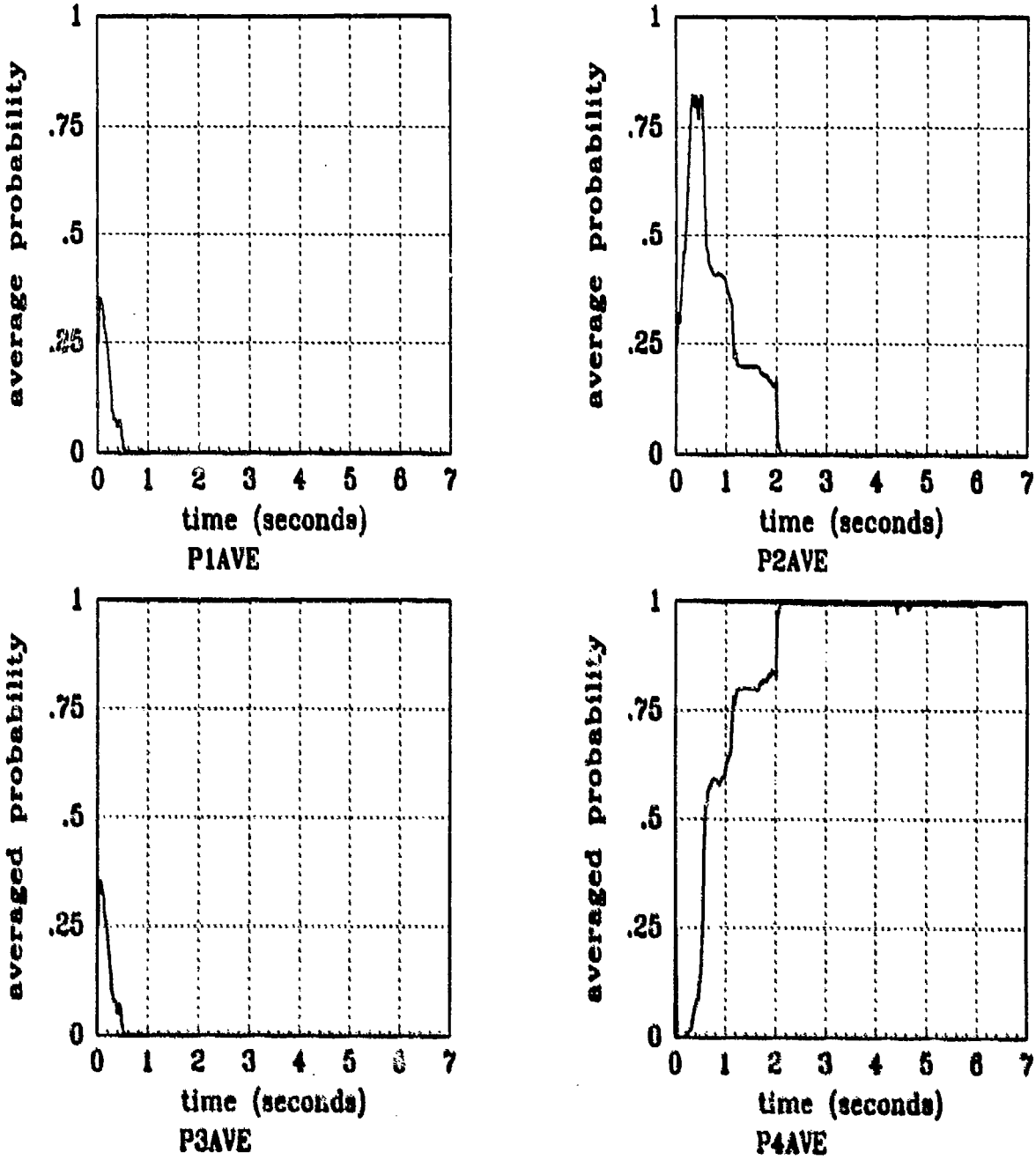


Figure J.5 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_s(t_i)$
(Truth = $1.12/20\pi$)

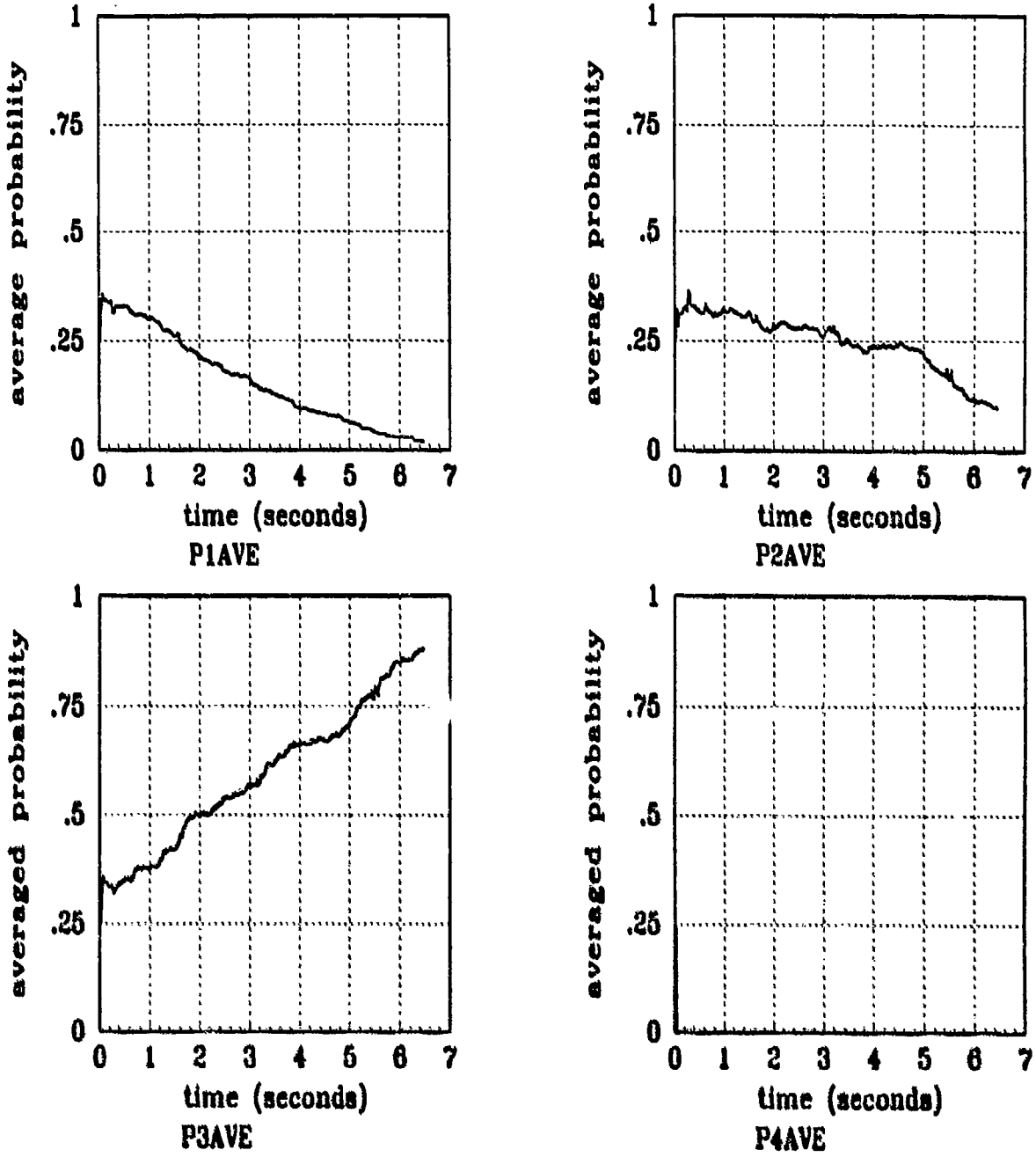


Figure J.6 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_x(t_i)$
(Truth = $0.112/2\pi$)

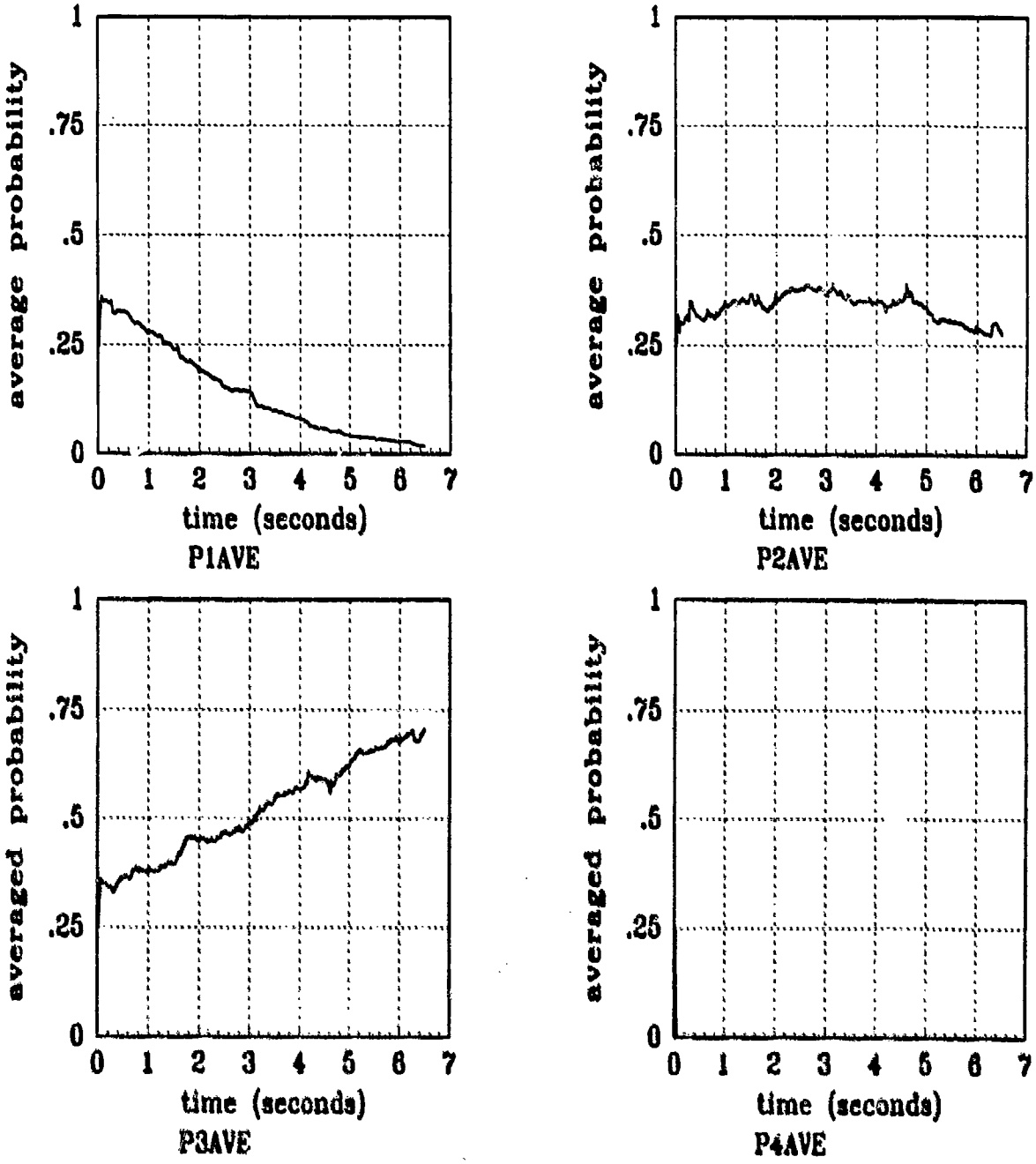


Figure J.7 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t)$
(Truth = $0.112/20\pi$)

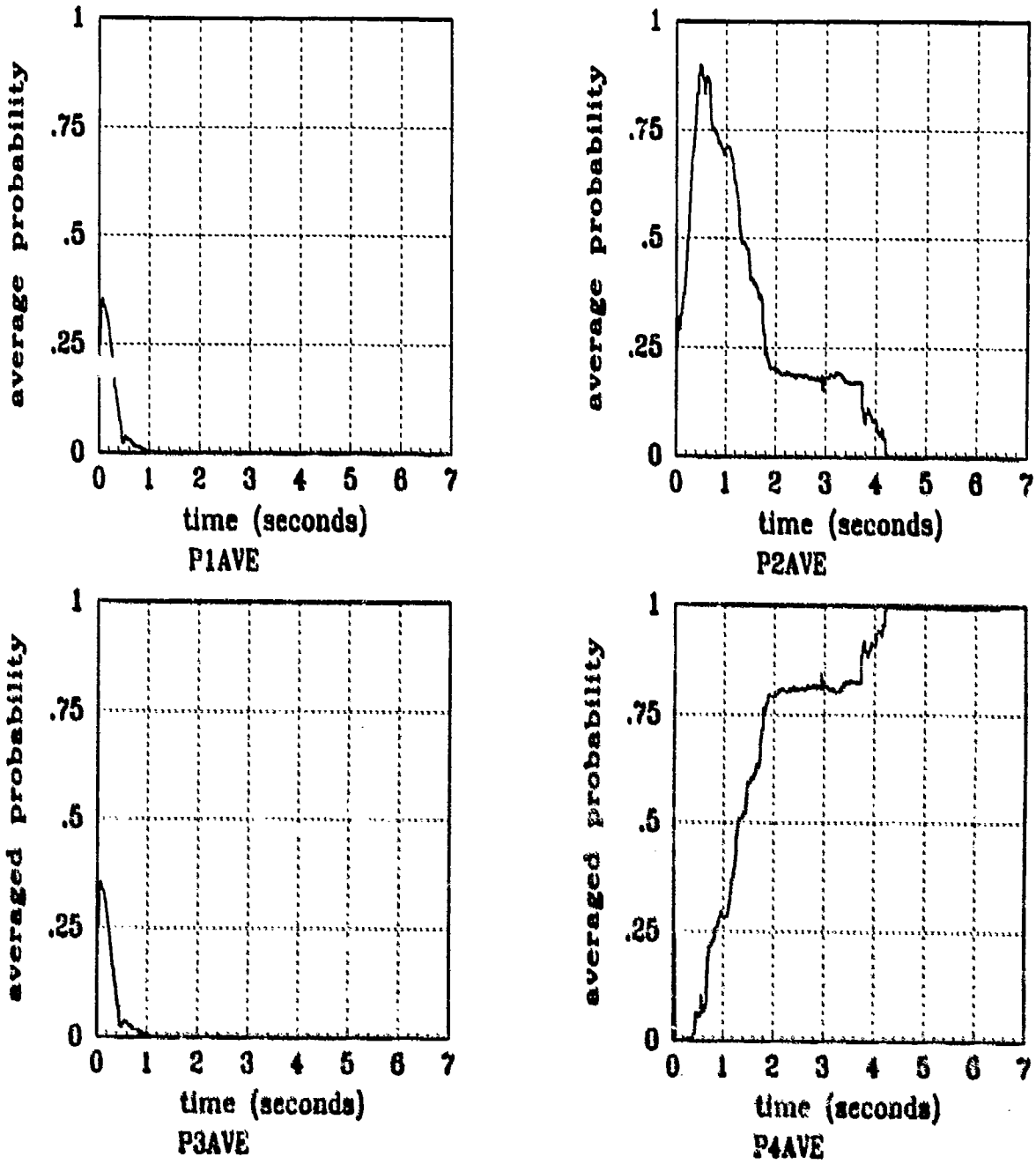


Figure J.8 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t_i)$
(Truth = $0.784/20\pi$)

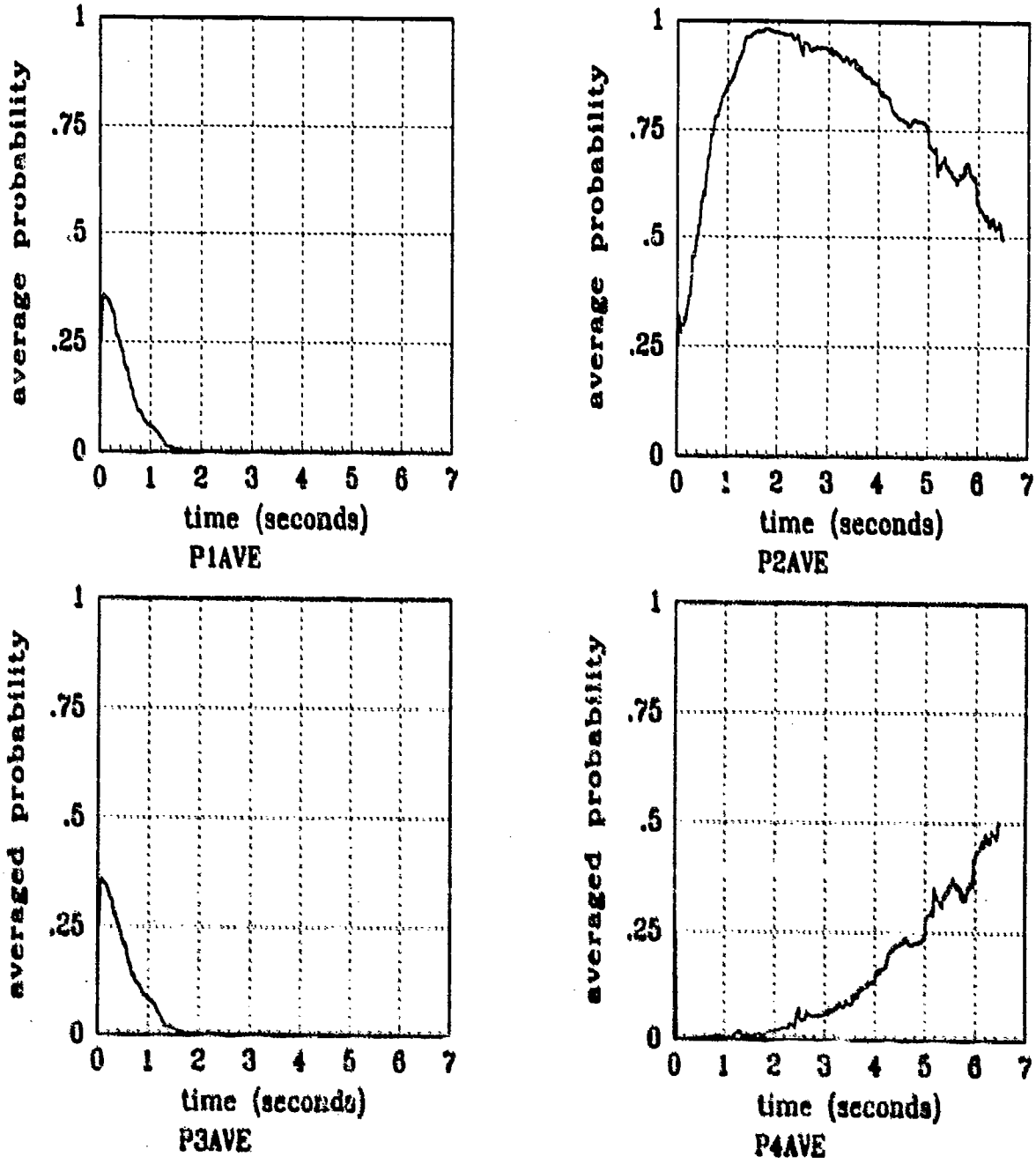


Figure J.9 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_x(t)$
(Truth = $0.448/20\pi$)

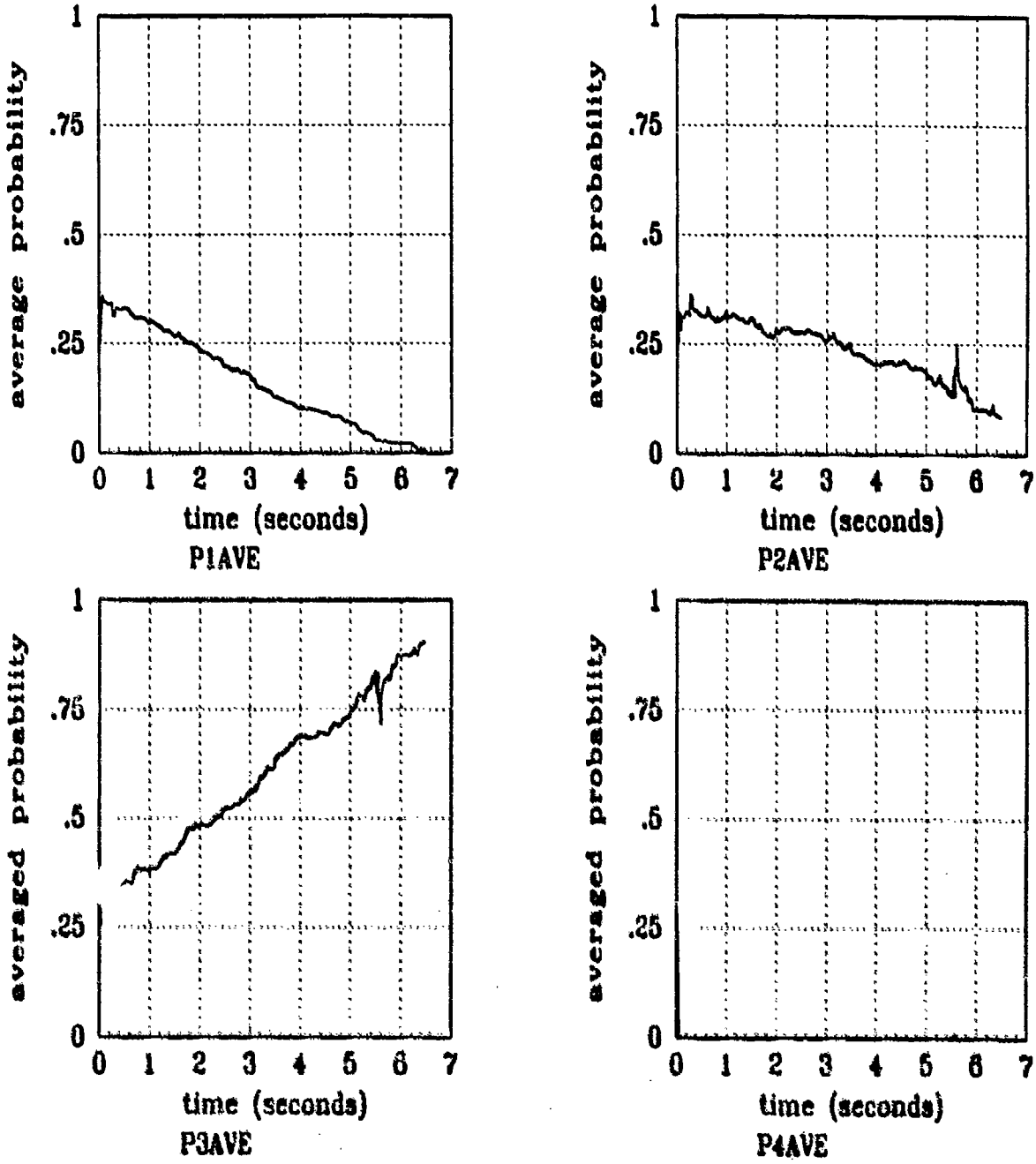


Figure J.10 Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t)$
(Truth = $0.448/2\pi$)

Appendix K

MMAE Configuration #5 Outp'·' Plots

This appendix contains the state and error statistics plots of the nine-state elemental filters. The data depicted in the two types of plots in this appendix are explained in *Appendix A*. The state comparison plots show the sample mean truth state over the 5 *Monte Carlo* runs compared to the same statistic for the filter estimate. The error statistics plots represent the error mean $\pm$ standard deviation values in pixels (or pixel/second for velocity and pogo velocity), of the errors between the filter estimate and true state; true mean $\pm$ one true standard deviation are plotted, along with zero $\pm$ one filter-computed standard deviation.

Note the filter covariance calculation is shown in Equation (4.5).

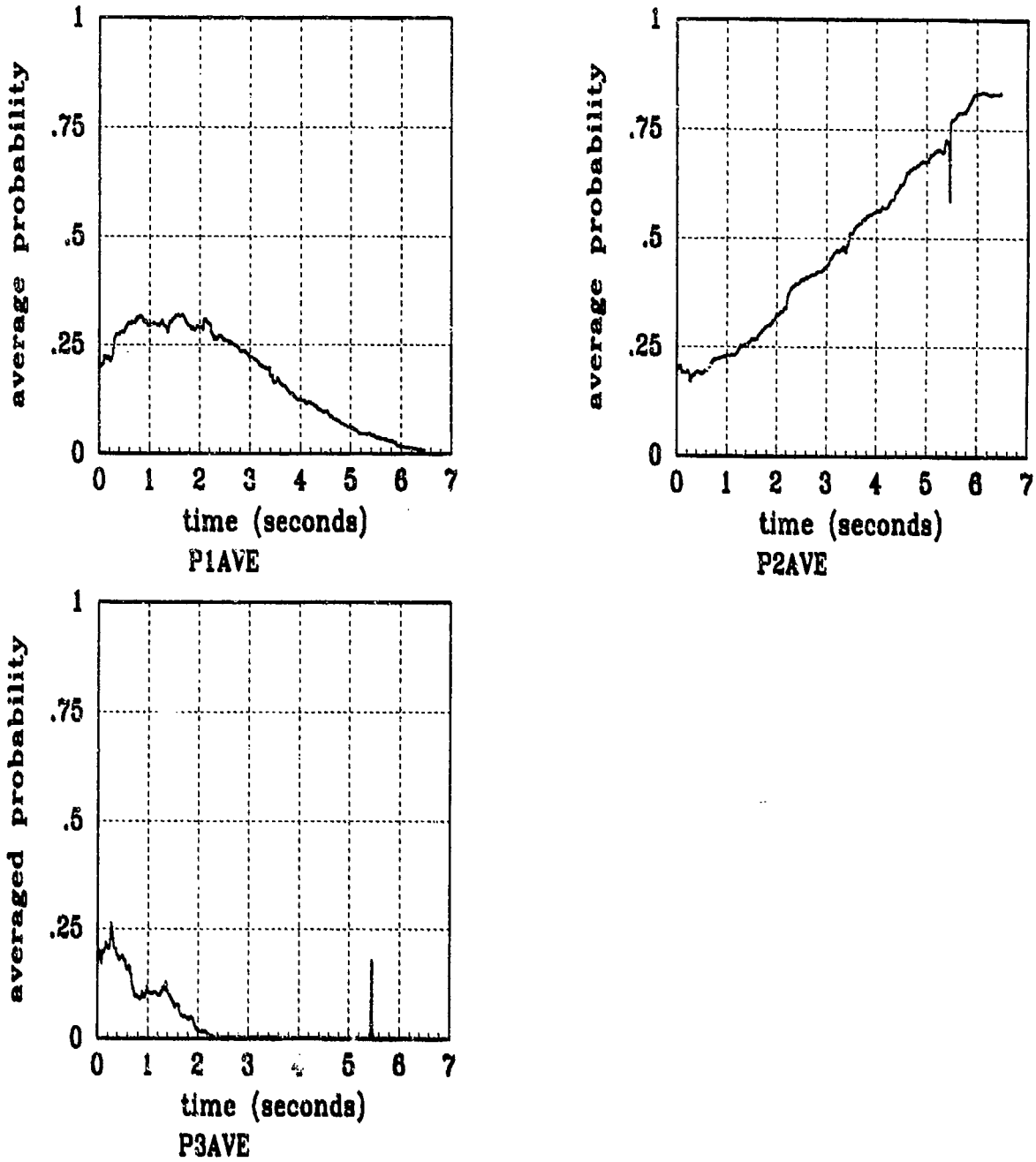
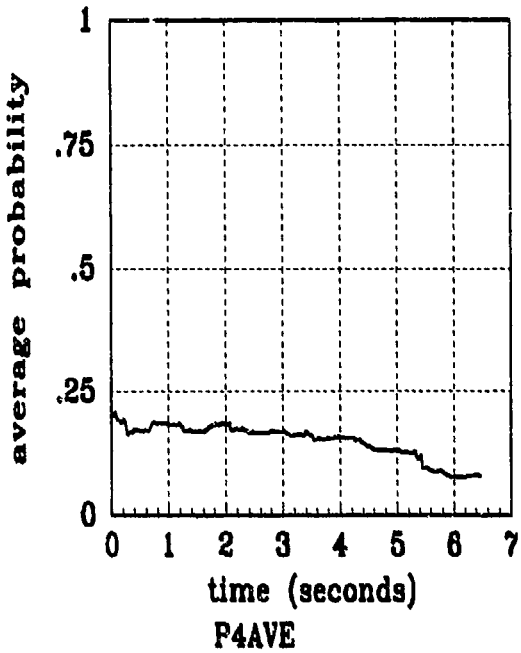


Figure K.1a Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$
(Truth = $1.12/2\pi$)



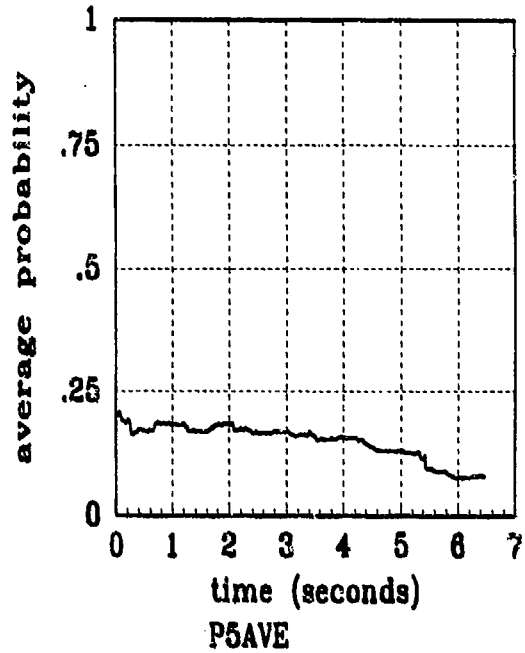


Figure K.1b Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$
(Truth = $1.12/2\pi$)

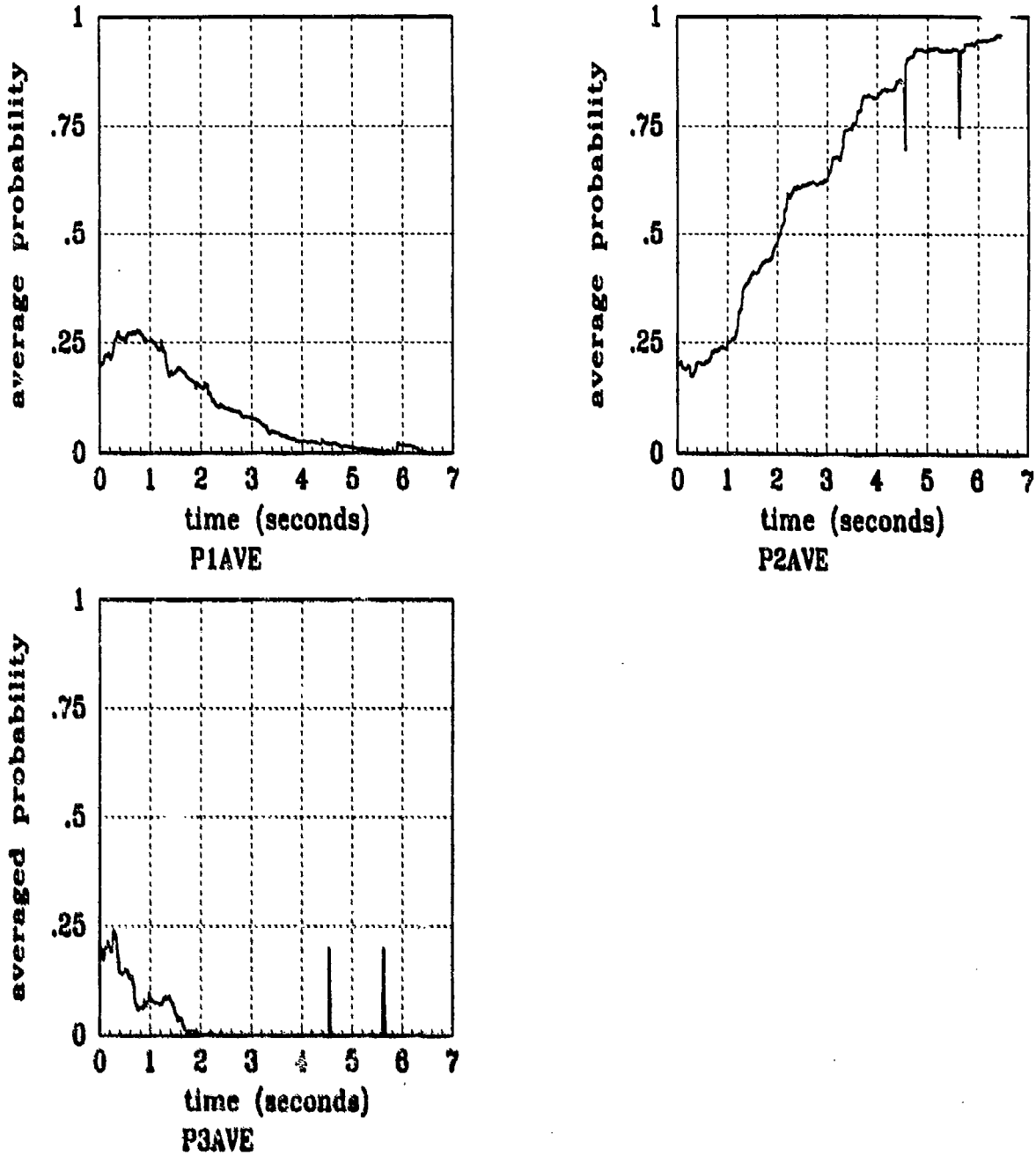
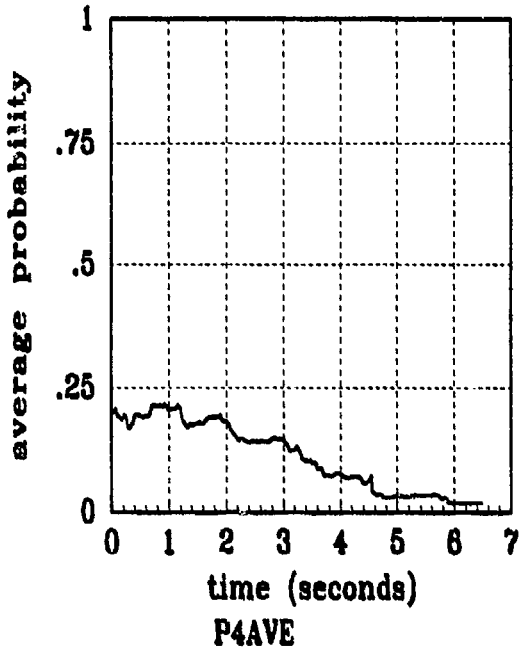


Figure K.2a Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = $2.24/2\pi$)



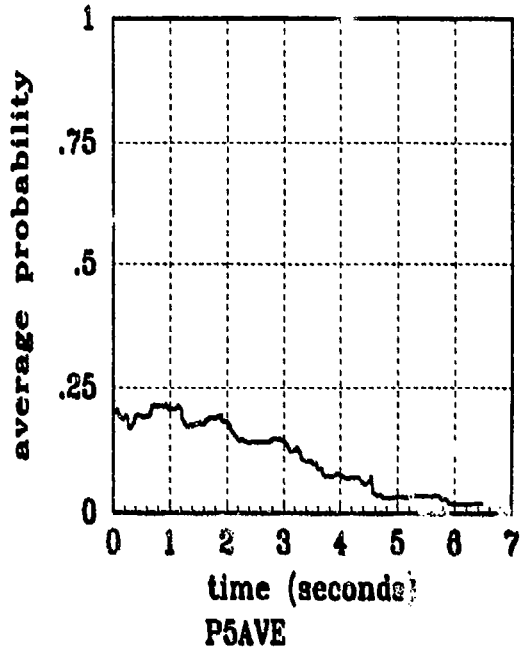


Figure K.2b Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t_i)$
(Truth = $2.24/2\pi$)

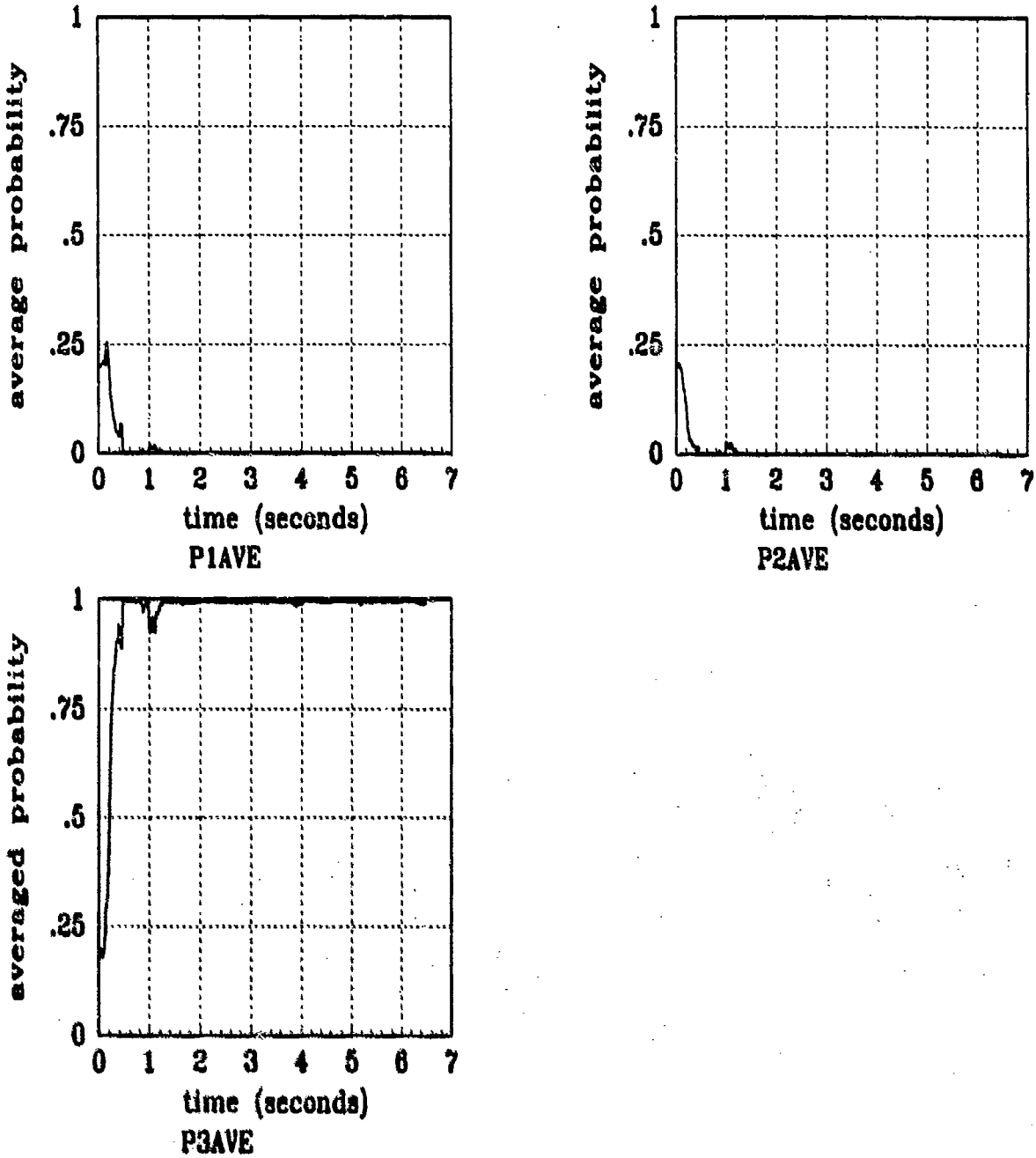
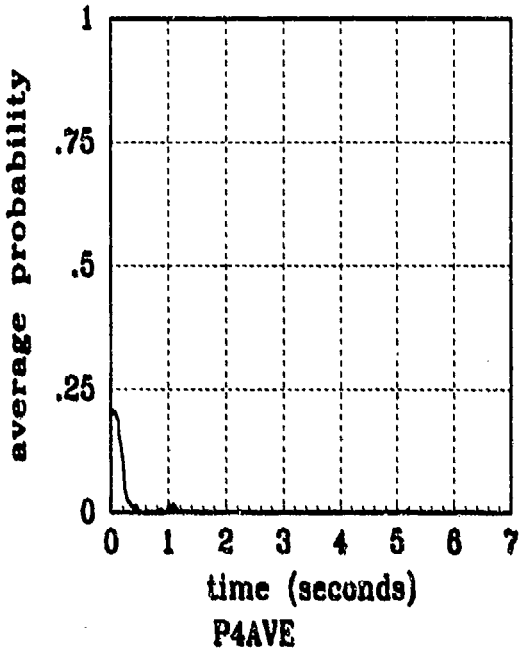


Figure K.3a Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_1(t_i)$
(Truth = $1.12/20\pi$)



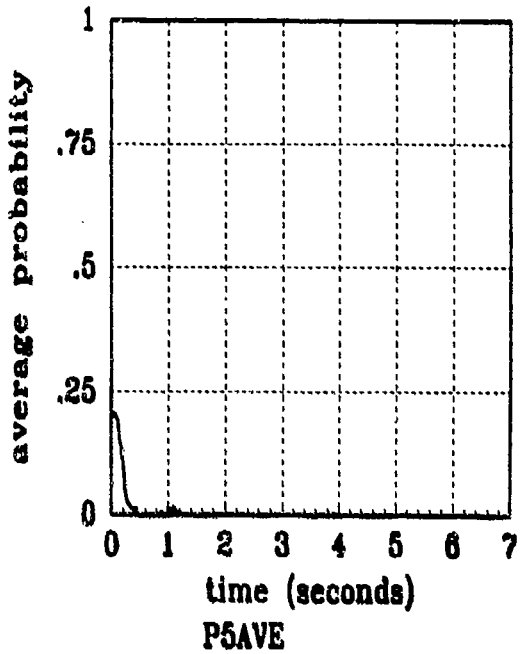


Figure K.3b Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$
(Truth = $1.12/20\pi$)

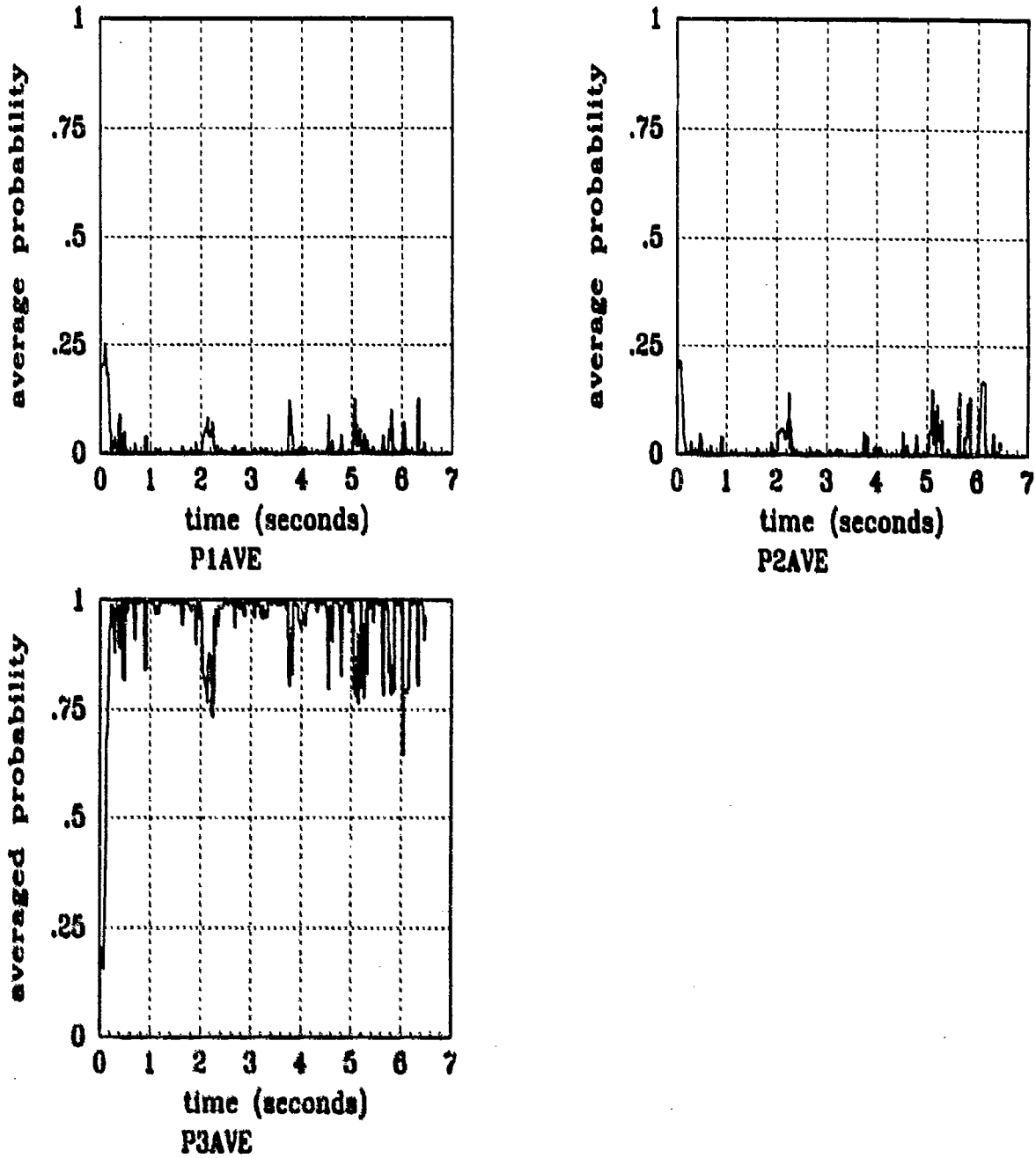
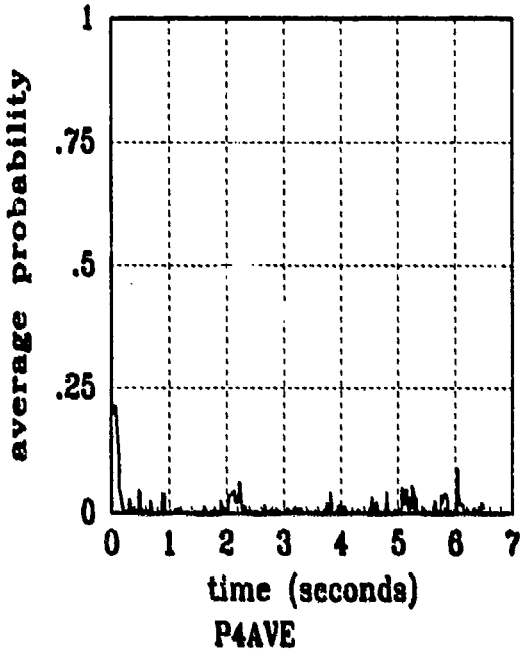


Figure K.4a Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$
(Truth = $2.24/20\pi$)



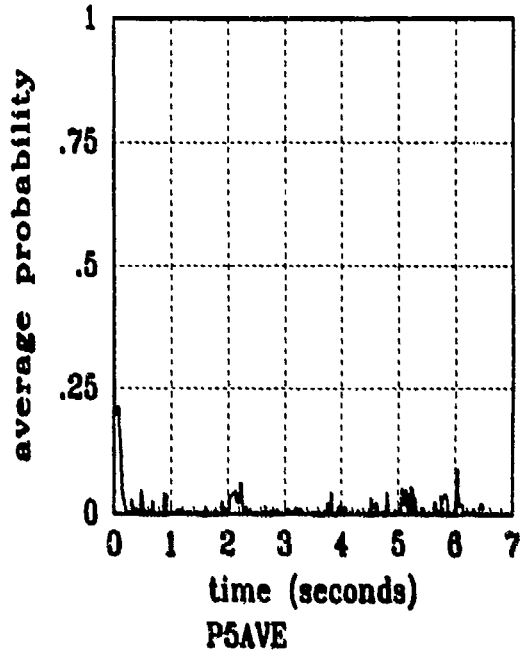


Figure K.4b Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$
(Truth = $2.24/20\pi$)

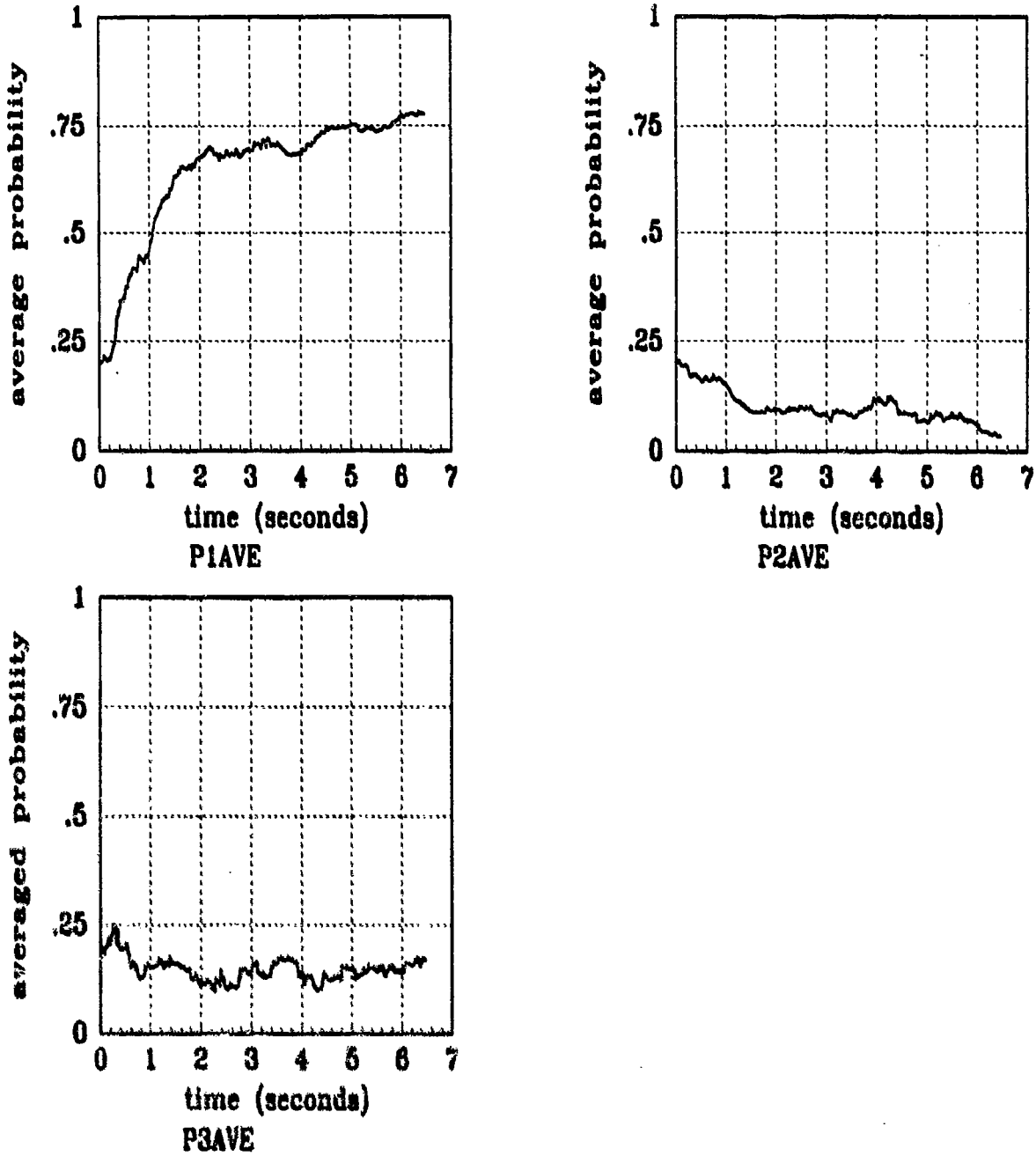
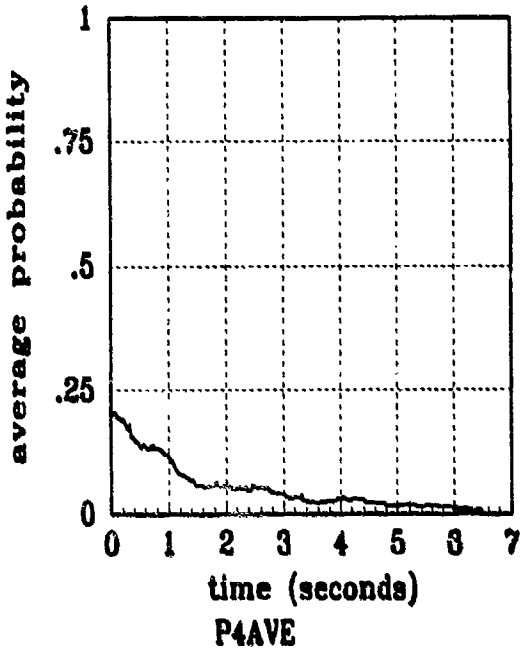


Figure K.5a Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_k(t_i)$ (Truth = $0.448/11\pi$)



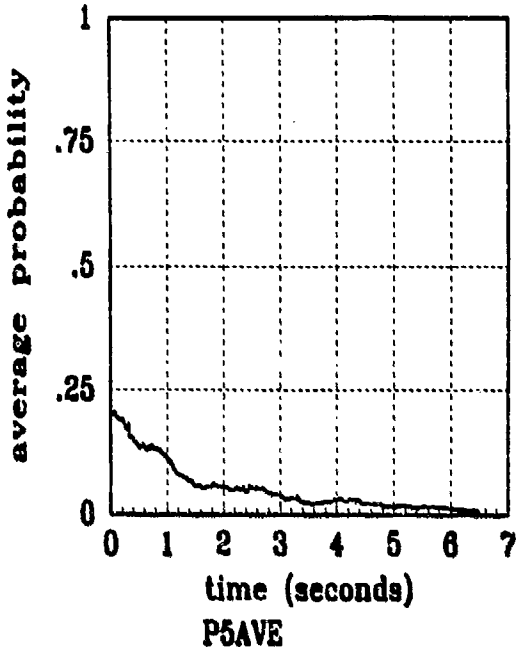


Figure K.5b Filter Residual Quantities; Computed *Without* Leading Coefficient or $A_4(t_i)$
(Truth = $0.448/11\pi$)

1. Ching, Michael L. *Kalman Filter Tracking of a Ballistic Missile Using Forward Looking Infrared Measurements and Doppler Returns.* MS Thesis, AFIT/GE/ENG/92D-09. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1992.

2. Durst, F., A. Melling, and J. H. Whitelaw. *Principles and Practice of Laser-Doppler Anemometry.* New York: Academic Press, 1981.

3. Eden, Claude W. *Enhanced Tracking of Ballistic Targets Using Forward Looking Infrared Measurements With Active Target Illumination.* MS Thesis, AFIT/GE/ENG/89D-11. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH December 1989.

4. Evans, Roger J. *Kalman Filter Tracking of a Reflective Target Using Forward Looking Infrared Measurements and Laser Illumination.* MS Thesis, AFIT/GE/ENG/90D-20. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1990.

5. Fairbanks, Ronald R., Capt, Headquarters 6585th Test Group. Holloman AFB, NM. Letter and Unclassified Empirical Radar Information. 19 June 1990.

6. Flynn, P. M. *Alternative Dynamics Models and Multiple Model Filtering for a Short Range Tracker.* MS Thesis, AFIT/GE/ENG/81D. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1981.

7. Gill, T. P. *The Doppler Effect.* New York: Academic Press, 1965.

8. Harnly, Douglas A., and Robert L. Jensen. *An Adaptive Distributed-Measurement Extended Kalman Filter for a Short Range Tracker.* MS Thesis, AFIT/GA/EE/79-1. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1979.

9. Herrera, Theodore D. *Kalman Filter Tracking of a Reflective Target Using Forward Looking Infrared Measurements and Doppler Returns.* MS Thesis, AFIT/GE/ENG/91D-25. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1991.

10. Integrated Systems Inc. *MATRIX$_x$ 7.0 Release Notes.* Integrated Systems Inc., Santa Clara, California, October 1988.

11. Kozemchak, Mark R. *Enhanced Image Tracking: Analysis of Two Acceleration Models in Tracking Multiple Hot-Spot Images.* MS Thesis, AFIT/GE/ENG/82D. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1982.

12. Leeney, T. A. *A Multiple Model Adaptive Tracking Algorithm Against Airborne Targets.* MS Thesis, AFIT/GE/ENG/87D-37. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1987.

13. Levanon, Nadav. *Radar Principles.* New York: John Wiley and Sons, 1988.

14. Loving, Phyllis A. *Bayesian vs. MAP Multiple Model Adaptive Estimation for Field of View Expansion in Tracking Airborne Targets.* MS Thesis, AFIT/GE/ENG/85M. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, March 1985.

15. McManamon, Paul F. *Active/Passive Sensor for Strategic Defense.* Electro-Optics Branch, Mission Avionics Division, Wright Laboratories, Wright-Patterson AFB, OH, 16 April 1991.

16. Maybeck Peter S. *Stochastic Models, Estimation and Control, Volume I.* New York: Academic Press, 1979.

17. - - - - -. *Stochastic Models, Estimation and Control, Volume II.* New York: Academic Press, 1982.

18. - - - - -. "Adaptive Tracking of Maneuvering Targets Based on IR Image Data." *Kalman Filter Integration of Modern Guidance and Navigation Systems*, Proceedings of the North Atlantic Treaty Organization (NATO), AGARD Lecture Series No. 166, 7-1 - 7-18, London, England, 1989.

19. - - - - -. Professor of Electrical Engineering. Personal Interviews. Air Force Institute of Technology (AU), School of Engineering, Wright-Patterson AFB, OH, January through December, 1993.

20. - - - - -, D. A. Harnly and R. L. Jensen. "An Adaptive Extended Kalman Filter for Target Image Tracking," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-17, No. 2, 173-180, March 1981.

21. - - - - - and D. E. Mercier. "A Target Tracker Using Spatially Distributed Infrared Measurements," *IEEE Transactions on Automatic Control*, Col. AC-25, No. 2, 222-225, April 1980.

22. - - - - - and S. K. Rogers. "Adaptive Tracking of Multiple Hot-Spot Target IR Images," *IEEE Transactions on Automatic Control*, Vol. AC-28, No. 10, 937-943, October 1983.

23. - - - - - and R. I. Suizu. "Adaptive Tracker Field of View Variation Via Multiple Model Filtering." *IEEE Transactions of Aerospace and Electronic Systems*, Vol. AES-21, No. 4, 529-539, July 1985.

24. - - - - -, W. H. Worsley, and P. M. Flynn. "Investigation of Constant Turn-Rate Dynamics Models in filters for Airborne Vehicle Tracking," *Proceedings of the IEEE National Aerospace and Electronics Conference*, Dayton, OH, 896-903, May 1982.

25. - - - - - and others. AFIT Graduate Student Developed Software for Kalman Filter Tracking of Airborne Targets Using Infrared and Laser Sensors. Fortran Software, Air Force Institute of Technology (AU), School of Engineering, Wright-Patterson AFB, OH, 1978 - December 1993.

26. Mercier, D. E. *An Extended Kalman Filter for Use in a Shared Aperture Medium Range Tracker.* MS Thesis, AFIT/GA/EE/78D-3. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1978.

27. Miller, Judith. *Doppler Tracking Program.* Briefing Slides. Phillips Laboratory, Kirtland AFB, New Mexico, 27 February 1991.

28. Millner, P. P. *Enhanced Tracking of Airborne Targets Using a Correlator/Kalman Filter.* MS Thesis, AFIT/GE/ENG/82M. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1982.

29. Murphy, P. J. *Laser Scattering and Extinction Techniques for Rocket Plume Diagnostics.* Technical Report AEDC-TR-89-16. Arnold Engineering Development Center, Air Force Systems Command, Arnold Air Force Base, Tennessee, April 1990.

30. Nawrocki, Paul J. and Robert Papa. *Atmospheric Processes.* New Jersey: Prentice-Hall Inc., 1963.

31. Netzer, Allan S. *Characteristics of Bayesian Multiple Model Adaptive Estimation for Tracking Airborne Targets.* MS Thesis, AFIT/GAE/ENG/85D-2. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH < December 1985.

32. Norton, John E. *Multiple Adaptive Tracking of Airborne Targets.* MS Thesis, AFIT/GA/ENG/88D-2. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1988.

33. Nowakowski, J. and A. Kalisz. *Measurement of Doppler Signatures.* Technical Report FR-1/B78-705. Air Force Weapons Laboratory (AFSC), Kirtland AFB, New Mexico, 15 August 1990.

34. Rizzo, David R. *Enhanced Tracking of Ballistic Targets Using Forward Looking Infrared Measurements.* MS Thesis, AFIT/GE/ENG/89M-7. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, March 1989.

35. Rogers, Steven K. *Advanced Tracking of Airborne Targets Using Forward Looking Infrared Measurements.* MS Thesis, AFIT/GE/ENG/81D. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1981.

36. Sheldon, S. N. "An Optimal Parameter Discretization Strategy for Multiple Model Adaptive Estimation and Control," Ph. D. Dissertation, Air Force Institute of Technology, December 1989.

37. Singletery, J. *Adaptive Laser Pointing and Tracking Problem.* MS Thesis, AFIT/GEO/EE/ENG/80D-12. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1980.

38. Skolnik, Merril I. *Introduction to Radar Systems.* New York: McGraw-Hill, Inc., 1962.

39. Stimson, George W. *Introduction to Airborne Radar.* Hughes Aircraft Company, El Segundo, CA., 1983.

40. Suizu, Robert I. *Advanced Tracking of Airborne Targets Using Multiple Model Adaptive Filtering Techniques for Adaptive Field of View Expansion.* MS Thesis, AFIT/GE/ENG/83D. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1983.

41. Tobin, David M. *A Multiple Model Adaptive Tracking Algorithm for a High Energy Laser Weapon System.* MS Thesis, AFIT/GE/ENG/86D-37. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1987.

42. Tobin, David M., and Peter S. Maybeck. "Substantial Enhancements to a Multiple Model Adaptive Estimator for Tracking Image Tracking," *Proceedings of the IEEE Conference on Decision and Control*, Los Angelos, CA, 2002-2011, December 1987.

43. Tobin, David M., and Peter S. Maybeck. "Enhancements to a Multiple Model Adaptive Estimator/Image Tracker," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-24, No. 4, pp 417-426, July 1988.

44. Toomay, J. C. *Radar Principles for the Non-Specialist.* Belmont, Ca: Wadsworth, Inc., 1982.

45. Wlodawski, M., et. al. *Measurements of Scattering by Particulates in Rocket Exhaust Plumes.* Technical Report TR-1/B78-7-05, Air Force Weapons Laboratory, Kirtland AFB, New Mexico.

Vita

Captain Patrick Joseph Grondin was born on June 6, 1959 in Sandusky, Ohio. He graduated from Apopka Memorial High School, Apopka, Florida in 1976 and enlisted in the United States Navy the same year. Patrick was a member of the Navy Band for eight years, stationed at Honolulu, Hawaii; Yokosuka, Japan; and Orlando, Florida. While stationed in Orlando in 1982, he began his engineering studies at Valencia Community College. In 1984, he was honorably discharged from the Navy and accepted into Air Force ROTC at the University of Central Florida in Orlando. In July, 1987, he received his B.S in Engineering from UCF and commission in the United States Air Force. His engineering career began in 1988 where he flew with the 4950th Test Wing as a Mission Coordinator on the Advanced Range Instrumentation Aircraft (ARIA) at Wright-Patterson AFB, OH. In 1992, Patrick was accepted to the Air Force Institute of Technology and began pursuing his Master's degree in Electrical Engineering. His follow-on assignment is to the Satellite Directorate at Los Angeles AFB, CA.

Permanent Address:

1632 Cherrystone Lane

Orlando, FL 32813

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	December 1993	Master's Thesis

4. TITLE AND SUBTITLE
ENHANCED TRACKING OF AIRBORNE TARGETS USING FORWARD LOOKING INFRARED AND LASER RETURN MEASUREMENTS

5. FUNDING NUMBERS

6. AUTHOR(S)
Patrick J. Grondin, Captain, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology
Wright-Patterson AFL, OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GE/ENG/93D-11

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Capt J. Bergen
Phillips Lab/LITT
Kirtland AFB, NM 87117

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

The Air Force Institute of Technology has been involved in developing Kalman filter based trackers of ballistic missiles for 15 years. The goal of this thesis is to develop a Multiple Model Adaptive Estimator (MMAE) that tracks the missile plume (using a forward looking infrared sensor) and the missile hardbody center-of-mass (additionally using low energy laser returns) for the purpose of directing a high power laser to incapacitate the missile. The missile plume "pogos" about an offset equilibrium point (relative to the hardbody center-of-mass) with an amplitude and frequency of oscillation that are not precisely known *a priori*. The MMAE algorithm estimates these parameters to improve performance in tracking the hardbody center-of-mass. To accomplish this MMAE structure, single Kalman filters were developed and tested at the different parameter values. A Kalman filter residual analysis was used on these working single filters to define the MMAE structure that provided the most effective adaptation and most accurate target tracking. A three-filter MMAE structure gave the lowest hardbody center-of-mass tracking errors. The two-dimensional parameter space, pogo amplitude and frequency, was successfully partitioned according to the frequency of oscillation. When the plume pogo amplitude is large, the MMAE structure substantially reduces the tracking errors of the hardbody center-of-mass, compared to a tracker without adaptive pogo estimation.

14. SUBJECT TERMS
Kalman Filter, Tracking, Infrared, Doppler, Laser

15. NUMBER OF PAGES
441

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL