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
4 August 1998

3. REPORT TYPE AND DATES COVERED

4. TITLE AND SUBTITLE
NEURAL NETWORK AUTOPILOT SYSTEM FOR A MATHEMATICAL MODEL OF THE BOEING 747

5. FUNDING NUMBERS

6. AUTHOR(S)
Gerald C. Cottrill

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
West Virginia University

8. PERFORMING ORGANIZATION REPORT NUMBER
98-042

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
THE DEPARTMENT OF THE AIR FORCE
AFIT/CIA, BLDG 125
2950 P STREET
WPAFB OH 45433

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT
Unlimited distribution
In Accordance With AFI 35-205/AFIT Sup 1

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

14. SUBJECT TERMS

15. NUMBER OF PAGES
135

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT

18. SECURITY CLASSIFICATION OF THIS PAGE

19. SECURITY CLASSIFICATION OF ABSTRACT

20. LIMITATION OF ABSTRACT

Abstract

Artificial neural networks can be defined as approximate mathematical models of the human brain's learning activities. In recent years neural networks have demonstrated abilities to perform autopilot and fault tolerant control tasks when applied to non-linear numerical aircraft simulations. Five on-line learning neural network autopilot systems, trained with the Standard and Extended Back-Propagation algorithms, were applied to a six degree-of-freedom non-linear simulation of a Boeing 747-200. The performance of the autopilots was compared based on their abilities to perform maneuvers at linear conditions and to adapt at non-linear conditions to restore steady state conditions.

Linear maneuvers were performed by introducing reference values of altitude and speed, pitch angle, roll angle, or heading angle. The performance using the SBPA was satisfactory, but the EBPA performance was clearly superior throughout the entire range maneuvers while compensating for lightly damped phugoid and Dutch roll modes.

Non-linear adaptation investigations were performed by exciting the non-linear terms in the equations of motion. The non-linear conditions were achieved in two ways: by simultaneously exciting pitch and roll rates with maximum elevator and aileron inputs, and the other by simultaneously exciting roll, pitch, and yaw rates with maximum elevator, aileron, and rudder inputs. The EBPA based controllers were able to regain steady state conditions for both non-linear tests with better transient performance than their SBPA counterparts. The SBPA showed only limited ability to adapt in cases where all three angular rates were excited.

Artificial neural networks trained on-line using the Extended Back-Propagation algorithm are concluded to be better suited for autopilot systems for the 1/25 scale Boeing 747 based on their superior abilities to perform linear maneuvers and regain steady state conditions when at non-linear conditions.

19980811 036

DTIC QUALITY INSPECTED 1

References

[1] Hagan, M.T., Demuth, H.B., Beale, M. Neural Network Design, PWS Publishing Company, Boston, MA, 1996.

[2] Fausett, L. Fundamentals of Neural Networks: Architectures, Algorithms, and Applications, Prentice Hall, Upper Saddle River, NJ, 1994.

[3] Kincheloe, M.W. "On-Line Learning Neural Network Controllers for Flight Control Systems", Thesis WVU MAE Dept., Morgantown, WV, April 1996.

[4] Casanova, J.L. "Design and Comparison of Neural Network and Fuzzy Logic Actuator Failure Schemes for Flight Control Systems", Thesis WVU MAE Dept., Morgantown, WV, December 1996.

[5] Advisory Group for Aerospace Research & Development. "Artificial Neural Network Approaches in Guidance and Control", Lecture Series #179.

[6] Rao, V.B., Rao, H.V. C++ Neural Networks & Fuzzy Logic, 2nd Edition, MIS:Press, New York, 1995.

[7] Lippmann, R.P. "An Introduction to Computing with Neural Nets", IEEE ASSP Magazine, April, 1987, pp. 4-22.

[8] Windon II, D.A. "Design and Comparison of Neural Network and Kalman Predictor Based Sensor Validation Schemes for Implementation on the NASA/Aurora Theseus", Thesis WVU MAE Dept., Morgantown, WV, June 1996.

[9] Napolitano, M.R., Neppach, C., Casdorph, V., Naylor, S. "On-Line Learning Non-Linear Direct Neuro Controllers for Restructurable Flight Control Systems", AIAA Journal of Guidance, Control, and Dynamics, Vol. 18, No. 1, Jan-Feb. 1995, pp. 170-176.

[10] Napolitano, M.R., Casdorph, V., Neppach, C., Naylor, S., Innocenti, M., Silvestri, G. "On-Line Learning Neural Architectures and Cross-Correlation Analysis for Actuator Failure Detection and Identification", International Journal of Control, 1996, Vol. 63, No. 3, pp. 433-455.

[11] Napolitano, M.R., Chen, C.I., Naylor, S. "Aircraft Failure Detection and Identification Using Neural Networks", AIAA Journal of Guidance, Control, and Dynamics, Vol. 16, No. 6, Nov-Dec. 1993, pp. 999-1009.

[12] Napolitano, M.R., Neppach, C., Casdorph, V., Naylor, S., Innocenti, M., Silvestri, G. "Neural Network Based Scheme for Sensor Failure Detection, Identification, and Accommodation", AIAA Journal of Guidance, Control, and Dynamics, Vol. 18, No. 6, Nov-Dec. 1995, pp. 1280-1286.

[13] Napolitano, M.R., Kincheloe, M. "On-Line Learning Neural Network Controllers for Autopilot Systems", Proceedings of the 95 AIAA Guidance, Navigation, and Control Conference, AIAA Paper 95-3269, Baltimore, MD, August 1995.

References

[1] Hagan, M.T., Demuth, H.B., Beale, M. Neural Network Design, PWS Publishing Company, Boston, MA, 1996.

[2] Fausett, L. Fundamentals of Neural Networks: Architectures, Algorithms, and Applications, Prentice Hall, Upper Saddle River, NJ, 1994.

[3] Kincheloe, M.W. "On-Line Learning Neural Network Controllers for Flight Control Systems", Thesis WVU MAE Dept., Morgantown, WV, April 1996.

[4] Casanova, J.L. "Design and Comparison of Neural Network and Fuzzy Logic Actuator Failure Schemes for Flight Control Systems", Thesis WVU MAE Dept., Morgantown, WV, December 1996.

[5] Advisory Group for Aerospace Research & Development. "Artificial Neural Network Approaches in Guidance and Control", Lecture Series #179.

[6] Rao, V.B., Rao, H.V. C++ Neural Networks & Fuzzy Logic, 2nd Edition, MIS:Press, New York, 1995.

[7] Lippmann, R.P. "An Introduction to Computing with Neural Nets", IEEE ASSP Magazine, April, 1987, pp. 4-22.

[8] Windon II, D.A. "Design and Comparison of Neural Network and Kalman Predictor Based Sensor Validation Schemes for Implementation on the NASA/Aurora Theseus", Thesis WVU MAE Dept., Morgantown, WV, June 1996.

[9] Napolitano, M.R., Neppach, C., Casdorph, V., Naylor, S. "On-Line Learning Non-Linear Direct Neuro Controllers for Restructurable Flight Control Systems", AIAA Journal of Guidance, Control, and Dynamics, Vol. 18, No. 1, Jan-Feb. 1995, pp. 170-176.

[10] Napolitano, M.R., Casdorph, V., Neppach, C., Naylor, S., Innocenti, M., Silvestri, G. "On-Line Learning Neural Architectures and Cross-Correlation Analysis for Actuator Failure Detection and Identification", International Journal of Control, 1996, Vol. 63, No. 3, pp. 433-455.

[11] Napolitano, M.R., Chen, C.I., Naylor, S. "Aircraft Failure Detection and Identification Using Neural Networks", AIAA Journal of Guidance, Control, and Dynamics, Vol. 16, No. 6, Nov-Dec. 1993, pp. 999-1009.

[12] Napolitano, M.R., Neppach, C., Casdorph, V., Naylor, S., Innocenti, M., Silvestri, G. "Neural Network Based Scheme for Sensor Failure Detection, Identification, and Accommodation", AIAA Journal of Guidance, Control, and Dynamics, Vol. 18, No. 6, Nov-Dec. 1995, pp. 1280-1286.

[13] Napolitano, M.R., Kincheloe, M. "On-Line Learning Neural Network Controllers for Autopilot Systems", Proceedings of the 95 AIAA Guidance, Navigation, and Control Conference, AIAA Paper 95-3269, Baltimore, MD, August 1995.

Abstract

Artificial neural networks can be defined as approximate mathematical models of the human brain's learning activities. In recent years neural networks have demonstrated abilities to perform autopilot and fault tolerant control tasks when applied to non-linear numerical aircraft simulations. Five on-line learning neural network autopilot systems, trained with the Standard and Extended Back-Propagation algorithms, were applied to a six degree-of-freedom non-linear simulation of a Boeing 747-200. The performance of the autopilots was compared based on their abilities to perform maneuvers at linear conditions and to adapt at non-linear conditions to restore steady state conditions.

Linear maneuvers were performed by introducing reference values of altitude and speed, pitch angle, roll angle, or heading angle. The performance using the SBPA was satisfactory, but the EBPA performance was clearly superior throughout the entire range maneuvers while compensating for lightly damped phugoid and Dutch roll modes.

Non-linear adaptation investigations were performed by exciting the non-linear terms in the equations of motion. The non-linear conditions were achieved in two ways: by simultaneously exciting pitch and roll rates with maximum elevator and aileron inputs, and the other by simultaneously exciting roll, pitch, and yaw rates with maximum elevator, aileron, and rudder inputs. The EBPA based controllers were able to regain steady state conditions for both non-linear tests with better transient performance than their SBPA counterparts. The SBPA showed only limited ability to adapt in cases where all three angular rates were excited.

Artificial neural networks trained on-line using the Extended Back-Propagation algorithm are concluded to be better suited for autopilot systems for the 1/25 scale Boeing 747 based on their superior abilities to perform linear maneuvers and regain steady state conditions when at non-linear conditions.

[14] Napolitano, M.R., Kincheloe, M. "On-Line Learning Neural Network Controllers for Autopilot Systems", AIAA Journal of Guidance, Control, and Dynamics, Vol. 33, No. 6, Nov-Dec 1995, pp. 1008-1015.

[15] Bragg, D.E. "Application and Comparison of Neural Network and Fuzzy Logic Techniques for Design of Auto-Pilot Systems", Thesis WVU MAE Dept., Morgantown, WV, 1996.

[16] Napolitano, M.R., Martinelli, R.D., Windon, D.A., Casanova, J.L. "Using Neural Networks for Reconstruction of Control Deflections in Aircraft Crash Data", Proceedings of the "Transportation 96" conference, Queretaro, Mexico, November 14-16, 1996.

[17] Napolitano, M.R., Windon, D.A. "Applications of Neural Networks for Signal Reconstruction in Aircraft Crash Investigations", AIAA Paper 96-3802, Proceedings of the AIAA Guidance, Navigation, and Control 96 conference, San Diego, CA, July 1996.

[18] Napolitano, M.R., Silvestri, G., Windon, D.A., Casanova, J.L., Innocenti, M. "Sensor Validation Using Hardware-Based On-Line Learning Neural Networks", submitted for evaluation to the IEEE Transactions on Aerospace and Electronic Systems, November 1995 - accepted for publication in October 1996.

[19] Roskam, J. Airplane Flight Dynamics and Automatic Flight Controls, Part I, Design, Analysis and Research Corp., Lawrence, KS, 1995.

[20] McLean, D. Automatic Flight Control Systems, Prentice Hall, New York, 1990.

[21] Paris, A.C. "Estimation of the Longitudinal and Lateral-Directional Aerodynamic Parameters from Flight Data for the NASA F/A-18 HARV", Ph.D. Dissertation, WVU MAE Dept., Morgantown, WV, May, 1997.

[22] Anderson, J.D. Jr. Introduction to Flight, 3rd Edition, McGraw-Hill, New York, 1989.

[23] Roskam, J. Airplane Flight Dynamics and Automatic Flight Controls, Part II, Design, Analysis and Research Corp., Lawrence, KS, 1995.

[24] Franklin, G.F., Powell, J.D., Emami-Naeini, A. Feedback Control of Dynamic Systems, 3rd Edition, Addison-Wesley, New York, 1994.

[25] Kreyszig, Erwin. Advanced Engineering Mathematics, 7th Edition, John Wiley and Sons, New York, 1993.

[14] Napolitano, M.R., Kincheloe, M. "On-Line Learning Neural Network Controllers for Autopilot Systems", AIAA Journal of Guidance, Control, and Dynamics, Vol. 33, No. 6, Nov-Dec 1995, pp. 1008-1015.

[15] Bragg, D.E. "Application and Comparison of Neural Network and Fuzzy Logic Techniques for Design of Auto-Pilot Systems", Thesis WVU MAE Dept., Morgantown, WV, 1996.

[16] Napolitano, M.R., Martinelli, R.D., Windon, D.A., Casanova, J.L. "Using Neural Networks for Reconstruction of Control Deflections in Aircraft Crash Data", Proceedings of the "Transportation 96" conference, Queretaro, Mexico, November 14-16, 1996.

[17] Napolitano, M.R., Windon, D.A. "Applications of Neural Networks for Signal Reconstruction in Aircraft Crash Investigations", AIAA Paper 96-3802, Proceedings of the AIAA Guidance, Navigation, and Control 96 conference, San Diego, CA, July 1996.

[18] Napolitano, M.R., Silvestri, G., Windon, D.A., Casanova, J.L., Innocenti, M. "Sensor Validation Using Hardware-Based On-Line Learning Neural Networks", submitted for evaluation to the IEEE Transactions on Aerospace and Electronic Systems, November 1995 - accepted for publication in October 1996.

[19] Roskam, J. Airplane Flight Dynamics and Automatic Flight Controls, Part I, Design, Analysis and Research Corp., Lawrence, KS, 1995.

[20] McLean, D. Automatic Flight Control Systems, Prentice Hall, New York, 1990.

[21] Paris, A.C. "Estimation of the Longitudinal and Lateral-Directional Aerodynamic Parameters from Flight Data for the NASA F/A-18 HARV", Ph.D. Dissertation, WVU MAE Dept., Morgantown, WV, May, 1997.

[22] Anderson, J.D. Jr. Introduction to Flight, 3rd Edition, McGraw-Hill, New York, 1989.

[23] Roskam, J. Airplane Flight Dynamics and Automatic Flight Controls, Part II, Design, Analysis and Research Corp., Lawrence, KS, 1995.

[24] Franklin, G.F., Powell, J.D., Emami-Naeini, A. Feedback Control of Dynamic Systems, 3rd Edition, Addison-Wesley, New York, 1994.

[25] Kreyszig, Erwin. Advanced Engineering Mathematics, 7th Edition, John Wiley and Sons, New York, 1993.

Neural Network Autopilot Systems for a Mathematical Model of the Boeing 747

Thesis

Submitted to the College of Engineering and Mineral Resources

of

West Virginia University

In partial fulfillment of the requirements for

the Degree of Master of Science

in Aerospace Engineering

by

Gerald C. Cottrill

Department of Mechanical and Aerospace Engineering
West Virginia University
Morgantown, West Virginia

January 1998

Acknowledgments

First, and foremost, I would like to thank God, who in his infinite grace has given me everything I could ever possibly need. I have Him to thank for helping me through my educational career. He gave me the strength and perseverance to make it through when I thought I surely would fail.

I would like to thank my family for all of their loving support. Thank you Grandma for giving me a place to live, home cooked meals, and the grandmotherly love that so many students miss when they go to college. Thank you, Dad, for always encouraging me and pushing me to do my best. It is because of your sacrifices for me that I have had the chance to get the education I have. I would also like to thank Carl for being my best friend as well as my brother.

I would like to thank my committee chairman, Dr. Napolitano, whose instruction sparked my interest in control theory and has given me the desire to learn more and more. Additionally, I would like to thank Dr. Morris for agreeing to be on my committee and for helping make sure that the program at WVU met AFIT requirements. Thanks to Dr. Klinkhachorn for sitting on my committee and for expanding my knowledge of neural network theory.

Additionally, I would like to thank those who have called Aerolab room 103 home. I would especially like to thank Brad Seanor, Al Paris, and Jose Casanova for mentoring me and befriending me. I've enjoyed all of the laughs we've shared in the office and I appreciate the advice you have given me when I needed it.

Finally, I would like to thank Raylea for making the last 3 years of my life wonderful! You have been patient and supportive as I went through Air Force R.O.T.C and through my Masters program. The time we have spent together has been priceless and I look foreword to spending my lifetime with you.

Abstract

Artificial neural networks can be defined as approximate mathematical models of the human brain's learning activities. In recent years neural networks have demonstrated abilities to perform autopilot and fault tolerant control tasks when applied to non-linear numerical aircraft simulations. Five on-line learning neural network autopilot systems, trained with the Standard and Extended Back-Propagation algorithms, were applied to a six degree-of-freedom non-linear simulation of a Boeing 747-200. The performance of the autopilots was compared based on their abilities to perform maneuvers at linear conditions and to adapt at non-linear conditions to restore steady state conditions.

Linear maneuvers were performed by introducing reference values of altitude and speed, pitch angle, roll angle, or heading angle. The performance using the SBPA was satisfactory, but the EBPA performance was clearly superior throughout the entire range maneuvers while compensating for lightly damped phugoid and Dutch roll modes.

Non-linear adaptation investigations were performed by exciting the non-linear terms in the equations of motion. The non-linear conditions were achieved in two ways: by simultaneously exciting pitch and roll rates with maximum elevator and aileron inputs, and the other by simultaneously exciting roll, pitch, and yaw rates with maximum elevator, aileron, and rudder inputs. The EBPA based controllers were able to regain steady state conditions for both non-linear tests with better transient performance than their SBPA counterparts. The SBPA showed only limited ability to adapt in cases where all three angular rates were excited.

Artificial neural networks trained on-line using the Extended Back-Propagation algorithm are concluded to be better suited for autopilot systems for the 1/25 scale Boeing 747 based on their superior abilities to perform linear maneuvers and regain steady state conditions when at non-linear conditions.

Table of Contents

Acknowledgments ii

Abstract iii

Table of Contents iv

List of Tables vi

List of Figures vii

List of Symbols xi

Chapter 1: Introduction

1.1 Biological and Artificial Neural Networks 1

1.2 Historical Development of Artificial Neural Networks 2

1.3 Applications of Artificial Neural Networks 4

1.4 Research Objectives 6

Chapter 2: Aircraft Equations of Motion

2.1 Introduction 7

2.2 Derivation of Equations of Motion 7

2.3 Standard Atmosphere 14

Chapter 3: The Extended Back Propagation Algorithm

3.1 Introduction 16

3.2 The Extended Back Propagation Algorithm 16

3.3 On-Line Learning 21

Chapter 4: Numerical Simulation

4.1 Introduction 22

4.2 Aircraft Geometric and Aerodynamic parameters 22

4.3 Simulation of Aircraft Dynamics 23

4.4 Simulation of Linear and Non-linear Tests 24

Chapter 5: On-Line Learning Neural Network Autopilot Systems

5.1 Introduction 25

5.2 Design Considerations 26

5.3 Altitude Hold Autopilot 28

5.4 Pitch Hold Autopilot 29

5.5 Airspeed Hold Autopilot 29

5.6 Roll Hold Autopilot 29

5.7 Direction Hold Autopilot 30

Chapter 6: Evaluation of Adaptation Capabilities at Linear and Non-linear Conditions

6.1 Linear Maneuvers Using AHNN and SHNN Autopilots 32

6.2 Linear Maneuvers Using PHNN and RHNN Autopilots 33

6.3 Linear Maneuvers Using DHNN Autopilots 33

6.4 Non-linear Adaptation Capabilities with Excited p and q Rates 34

6.5 Non-linear Adaptation Capabilities with Excited p, q, and r Rates 34

6.6 Summary of Results 35

Chapter 7: Conclusions and Recommendations

7.1 Conclusions 36

7.2 Recommendations 36

References 38

Tables 40

Figures 51

Approval of Examining Committee 135

List of Tables

Table 4.1 Geometric, mass, and aerodynamic data for the B747 from Ref.[19]

Table 5.1 Final Neural Network Autopilot Structures.

Table 6.1 AHNN and SHNN reference input values for maneuvering at LC.

Table 6.2 AHNN and SHNN reference input values for maneuvering at HC.

Table 6.3 Settling times for maneuvers at LC using AHNN's and SHNN's.

Table 6.4 Settling times for maneuvers at HC using AHNN's and SHNN's.

Table 6.5 Settling times for maneuvers at LC using PHNN's.

Table 6.6 Settling times for maneuvers at HC using PHNN's.

Table 6.7 Settling times for maneuvers at LC using RHNN's.

Table 6.8 Settling times for maneuvers at HC using RHNN's.

Table 6.9 Settling times for maneuvers at LC using DHNN's.

Table 6.10 Settling times for maneuvers at HC using DHNN's.

Table 6.11 Settling times for p and q non-linear excitations at LC.

Table 6.12 Settling times for p and q non-linear excitations at HC.

Table 6.13 Settling times for p, q and r non-linear excitations at LC.

Table 6.14 Settling times for p, q and r non-linear excitations at HC.

List of Figures

Figure 2.1 Aircraft forces and moments about the body axis.

Figure 2.2 Aircraft orientation with respect to the Earth Fixed Inertial Axis coordinate system by the introduction of Euler angles.

Figure 2.3 Relationship between Body, Stability, and Wind axis systems.

Figure 3.1 General representation of a 3 layer neural network trained with the Extended Back-Propagation Algorithm.

Figure 3.2 Extended Back-Propagation sigmoid activation function for general U, L, and T arguments.

Figure 4.1 User interface with numerical simulation program.

Figure 4.2 General block diagram of simulation program with neural network autopilot controllers.

Figures 6.1 - 6.2 Altitude response to commanded altitude and speed maneuvers at LC.

Figures 6.3 - 6.4 Speed response to commanded altitude and speed maneuvers at LC.

Figures 6.5 - 6.7 Altitude response to commanded altitude and speed maneuvers at HC.

Figures 6.8 - 6.10 Speed response to commanded altitude and speed maneuvers at HC.

Figure 6.11 Pitch attitude response to commanded pitching maneuvers at LC.

Figure 6.12 Pitch attitude response to command pitching maneuvers at HC.

Figure 6.13 Roll angle response to commanded rolling maneuvers at LC.

Figure 6.14 Roll angle response to commanded rolling maneuvers at HC.

Figure 6.15 Heading angle response to commanded direction change maneuvers at LC.

Figure 6.16 Roll angle response to commanded heading angle meneuvers at LC.

Figure 6.17 Heading angle response to commanded direction change maneuvers at HC.

Figure 6.18 Roll angle response to commanded heading angle maneuvers at HC.

Figure 6.19 Aircraft response to non-linear conditions when p and q are above 2 deg/sec at LC.

Figure 6.20 NN control inputs in response to p and q excitations above 2 deg/sec at LC.

Figure 6.21 Aircraft response to non-linear conditions when p and q are above 4 deg/sec at LC.

Figure 6.22 NN control inputs in response to p and q excitations above 4 deg/sec at LC.

Figure 6.23 Aircraft response to non-linear conditions when p and q are above 6 deg/sec at LC.

Figure 6.24 NN control inputs in response to p and q excitations above 6 deg/sec at LC.

Figure 6.25 Aircraft response to non-linear conditions when p and q are above 8 deg/sec at LC.

Figure 6.26 NN control inputs in response to p and q excitations above 8 deg/sec at LC.

Figure 6.27 Aircraft response to non-linear conditions when p and q are above 10 deg/sec at LC.

Figure 6.28 NN control inputs in response to p and q excitations above 10 deg/sec at LC.

Figure 6.29 Aircraft response to non-linear conditions when p and q are above 12 deg/sec at LC.

Figure 6.30 NN control inputs in response to p and q excitations above 12 deg/sec at LC.

Figure 6.31 Aircraft response to non-linear conditions when p and q are above 14 deg/sec at LC.

Figure 6.32 NN control inputs in response to p and q excitations above 14 deg/sec at LC.

Figure 6.33 Aircraft response to non-linear conditions when p and q are above 16 deg/sec at LC.

Figure 6.34 NN control inputs in response to p and q excitations above 16 deg/sec at LC.

Figure 6.35 Aircraft response to non-linear conditions when p and q are above 18 deg/sec at LC.

Figure 6.36 NN control inputs in response to p and q excitations above 18 deg/sec at LC.

Figure 6.37 Aircraft response to non-linear conditions when p and q are above 2 deg/sec at HC.

Figure 6.38 NN control inputs in response to p and q excitations above 2 deg/sec at HC.

Figure 6.39 Aircraft response to non-linear conditions when p and q are above 4 deg/sec at HC.

Figure 6.40 NN control inputs in response to p and q excitations above 4 deg/sec at HC.

Figure 6.41 Aircraft response to non-linear conditions when p and q are above 6 deg/sec at HC.

Figure 6.42 NN control inputs in response to p and q excitations above 6 deg/sec at HC.

Figure 6.43 Aircraft response to non-linear conditions when p and q are above 8 deg/sec at HC.

Figure 6.44 NN control inputs in response to p and q excitations above 8 deg/sec at HC.

Figure 6.45 Aircraft response to non-linear conditions when p, q, and r are above 2 deg/sec at LC.

Figure 6.46 Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 2 deg/sec at LC.

Figure 6.47 NN control inputs in response to p, q, and r excitations above 2 deg/sec at LC.

Figure 6.48 Aircraft response to non-linear conditions when p, q, and r are above 4 deg/sec at LC.

Figure 6.49 Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 4 deg/sec at LC.

Figure 6.50 NN control inputs in response to p, q, and r excitations above 4 deg/sec at LC.

Figure 6.51 Aircraft response to non-linear conditions when p, q, and r are above 6 deg/sec at LC.

Figure 6.52 Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 6 deg/sec at LC.

Figure 6.53 NN control inputs in response to p, q, and r excitations above 6 deg/sec at LC.

Figure 6.54 Aircraft response to non-linear conditions when p, q, and r are above 8 deg/sec at LC.

Figure 6.55 Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 8 deg/sec at LC.

Figure 6.56 NN control inputs in response to p, q, and r excitations above 8 deg/sec at LC.

Figure 6.57 Aircraft response to non-linear conditions when p, q, and r are above 10 deg/sec at LC.

Figure 6.58 Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 10 deg/sec at LC.

Figure 6.59 NN control inputs in response to p, q, and r excitations above 10 deg/sec at LC.

Figure 6.60 Aircraft response to non-linear conditions when p, q, and r are above 12 deg/sec at LC.

Figure 6.61 Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 12 deg/sec at LC.

Figure 6.62 NN control inputs in response to p, q, and r excitations above 12 deg/sec at LC.

Figure 6.63 Aircraft response to non-linear conditions when p, q, and r are above 2 deg/sec at HC.

Figure 6.64 Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 2 deg/sec at HC.

Figure 6.65 NN control inputs in response to p, q, and r excitations above 2 deg/sec at HC.

Figure 6.66 Aircraft response to non-linear conditions when p, q, and r are above 4 deg/sec at HC.

Figure 6.67 Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 4 deg/sec at HC.

Figure 6.68 NN control inputs in response to p, q, and r excitations above 4 deg/sec at HC.

Figure 6.69 Aircraft response to non-linear conditions when p, q, and r are above 6 deg/sec at HC.

Figure 6.70 Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 6 deg/sec at HC.

Figure 6.71 NN control inputs in response to p, q, and r excitations above 6 deg/sec at HC.

Figure 6.72 Aircraft response to non-linear conditions when p, q, and r are above 8 deg/sec at HC.

Figure 6.73 Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 8 deg/sec at LC.

Figure 6.74 NN control inputs in response to p, q, and r excitations above 8 deg/sec at HC.

Figure 6.75 Aircraft response to non-linear conditions when p, q, and r are above 10 deg/sec at HC.

Figure 6.76 Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 10 deg/sec at HC.

Figure 6.77 NN control inputs in response to p, q, and r excitations above 10 deg/sec at HC.

LIST OF SYMBOLS

English

a	Acceleration	g
b	Wing Span	ft
$\bar{c}$	Mean aerodynamic chord	ft
C_A	Axial force coefficient	
C_D	Drag coefficient	
C_{D_0}	Drag coefficient at zero angle of attack	
$C_{D_{i_h}}$	Drag coefficient variation with stabilizer angle	rad^{-1}
C_{D_α}	Drag coefficient variation with angle of attack	rad^{-1}
C_{D_δ}	Drag coefficient variation with surface deflection	rad^{-1}
C_L	Lift coefficient	
C_{L_0}	Lift coefficient for zero angle of attack	
C_{L_δ}	Lift coefficient variation with control surface deflection	rad^{-1}
C_{L_α}	Lift coefficient variation with angle of attack	rad^{-1}
$C_{L_{\dot{\alpha}}}$	Lift coefficient variation with rate of change of angle of attack	rad^{-1}
C_{L_q}	Lift coefficient variation with pitch rate	
C_{L_u}	Lift coefficient variation with speed	rad^{-1}
C_l	Rolling moment coefficient	
C_{l_0}	Rolling moment coefficient for zero sideslip	
C_{l_β}	Rolling moment coefficient variation with sideslip angle	rad^{-1}
C_{l_p}	Rolling moment coefficient variation with roll rate	rad^{-1}
C_{l_r}	Rolling moment coefficient variation with yaw rate	rad^{-1}
C_{l_δ}	Rolling moment coefficient variation with control surface deflection	rad^{-1}
C_m	Pitching moment coefficient	

C_{m_0}	Pitching moment coefficient with zero angle of attack	
C_{m_α}	Pitching moment coefficient variation with angle of attack	rad^{-1}
$C_{m_{i_h}}$	Pitching moment coefficient variation with stabilizer angle	rad^{-1}
C_{m_δ}	Pitching moment coefficient variation with control surface deflection	rad^{-1}
$C_{m_{\dot{\alpha}}}$	Pitching moment coefficient variation with rate of change of angle of attack	rad^{-1}
C_{m_q}	Pitching moment coefficient variation with pitch rate	rad^{-1}
C_n	Yawing moment coefficient	
C_{n_0}	Yawing moment coefficient for zero sideslip	
C_{n_β}	Yawing moment coefficient variation with sideslip angle	rad^{-1}
C_{n_p}	Yawing moment coefficient variation with roll rate	rad^{-1}
C_{n_r}	Yawing moment coefficient variation with yaw rate	rad^{-1}
C_{n_δ}	Yawing moment coefficient variation with control surface deflection	rad^{-1}
C_{Norm}	Force coefficient normal to aircraft	
C_x	Force coefficient along x axis	
C_y	Side force coefficient	
C_{y_0}	Side force coefficient with zero sideslip	
C_{y_β}	Side force coefficient variation with sideslip angle	rad^{-1}
C_{y_δ}	Side force coefficient variation with control surface deflection	rad^{-1}
C_{y_p}	Side force coefficient variation with roll rate	rad^{-1}
C_{y_r}	Side force coefficient variation with yaw rate	rad^{-1}
d	Derivative	
F	Externally applied force (vector or scalar as indicated)	lbs
f	Sigmoid activation function	
g	Acceleration of gravity	ft/sec^2
H_i	Hidden layer neuron output	

h	Altitude	ft
I	Moment of inertia	sl-ft^2
J	Performance index	
L	Lower bound in modified sigmoid activation function	
M	Externally applied moment (vector or scalar as indicated)	ft-lbs
m	Mass	sl
net	Weighted sum input to sigmoid function	
O_j	Output of the output layer neurons	
p	Input data window size	
p	Roll rate	rad/s ; deg/s
q	Pitch rate	rad/s ; deg/s
$\bar{q}$	Dynamic pressure	lbs/ft^2
r	Yaw rate	rad/s ; deg/s
R	Universal ideal gas constant	(ft•lb)/(sl•°R)
S	Wing surface area	ft^2
t	Time	sec
T	Temperature	°R
T	Aircraft engine thrust	lbs
T	Slope of sigmoid activation function	
U	Upper bound of sigmoid activation function	
u	Velocity component along x-axis	ft/s
V	Total velocity (vector or magnitude as indicated)	ft/s
v	Velocity component along y-axis	ft/s
v	General weighting term in performance index	
v	Hidden layer connection weights	

w	Velocity component along z-axis	ft/s
w	General weighting term in performance index	
w	Output layer connection weights	
X	General variable in performance index	
Y	Target output of output layer neurons	

GREEK

α	Angle of attack	rad ; deg
α	Momentum term	
β	Sideslip angle	rad ; deg
Γ	Output layer threshold	
Δ	Increment	
δ	Error signals	
δ_A	Aileron surface deflection	rad ; deg
δ_E	Elevator surface deflection	rad ; deg
δ_R	Rudder surface deflection	rad ; deg
δ_T	Throttle input	%
η	Learning rate	
Θ	Hidden layer threshold	
Θ	Pitch angle	rad ; deg
ρ	Air density	sl/ft^3
Σ	Summation	
Φ	Bank (roll) angle	rad ; deg
Ψ	Heading (yaw) angle	rad ; deg
ω	Angular velocity (vector or scalar as indicated)	rad/s ; deg/s
∂	Partial derivative	

Subscripts

0	Sea level conditions
1	Initial condition
A	Aileron
E	Elevator
ext	Externally applied force
i	Index of hidden layer neurons
j	Index of output layer neurons
k	Number of hidden layer neurons
L	Lower bound of sigmoid activation function
l	Index of input values
n	Number of output layer neurons
net	Weighted sum input to sigmoid activation function
R	Rudder
T	Slope of sigmoid activation function
T	Throttle
U	Upper bound of sigmoid activation function

Overstrike

•	Time rate of change
→	Vector quantity

Acronyms

ADALINE	ADAptive LINear NEuron
AFDIA	Actuator Failure Detection, Identification, and Accommodation
AFIT	Air Force Institute of Technology

AHNN	Altitude Hold Neural Network
ANN	Artificial Neural Network
B747	Boeing 747
BPA	Back-Propagation Algorithm
CG	Center of Gravity
DHNN	Direction Hold Neural Network
EBPA	Extended Back-Propagation Algorithm
ECG	ElectroCardioGram
EEG	ElectroEncephaloGram
HC	High Cruise
LC	Low Cruise
LTI	Linear Time Invariant
MADALINE	Multiple ADAptive LINear NEuron
NASA	National Aeronautics and Space Administration
PC	Personal Computer
PD	Proportional + Derivative controller
PHNN	Pitch Hold Neural Network
PI	Proportional + Integral controller
PID	Proportional + Integral + Derivative controller
RHNN	Roll Hold Neural Network
SBPA	Standard Back-Propagation Algorithm
SFDIA	Sensor Failure Detection, Identification, and Accommodation
SHNN	Speed Hold Neural Network
WVU	West Virginia University

Chapter 1: Introduction

1.1 Biological and Artificial Neural Networks

It can be argued that the best control system is an experienced human operator. Human beings are able to learn to do very complex tasks. Once properly trained, humans are able to perform these tasks with minimal concentration while performing other tasks simultaneously. Consider a child learning to ride a bicycle or a teenager learning to drive a car. Initially, total focus and attention is required to control the bicycle or car. Any small disturbance may elicit a significant reaction. With time and experience the control of these systems requires less and less effort. Think of a novice pilot. It takes a significant amount of time to learn the purpose of the instruments and how to operate the aircraft. Initially, the workload may seem overwhelming. However, training and practice eventually will help the pilot become an expert to the point where flying an airplane becomes natural to him. The remarkable ability of the human brain to learn and perform complex tasks is the motivation for the development of Artificial Neural Networks (ANN's).

The human brain is made up of approximately 10^{11} processing elements called neurons. Each neuron has on the order of 10^4 connections to other neurons[1]. In other words, the brain is comprised of many interconnected parallel processing elements. Each neuron has three key elements: dendrites, soma or cell body, and axons. The dendrites serve as the input signal carriers. They carry the input signals to the neurons from other connected neurons and chemically alter the signals, which is similar to multiplying them by scaling factors. The cell body (soma) sums the scaled incoming signals. Once the potential reaches some threshold value the cell sends an output signal over the axons to other neurons[1,2].

Since the brain is so powerful, it is desirable to model it, in a limited way, to solve complex problems using ever advancing computer technology. ANN's are approximate mathematical models inspired by the brain's neural anatomy. ANN's are made up of many processing elements, also

referred to as neurons. These neurons share and transmit information through connection links. These connections also have associated weights that multiply the incoming signals to the neurons. The individual neuron sums the weighted input signals and applies a mathematical activation function to the weighted sum input to produce the output to be transmitted to other neurons[2,5]. ANN's are characterized by the arrangement of neurons and connections and the method used to update the connection weights. Detailed presentations of different architectures and training algorithms are presented in Refs.[1,2,6,7].

1.2 Historical Development of Artificial Neural Networks

Pioneer research work on artificial neural networks can be traced back to the 1940's when Warren McCulloch and Walter Pitts designed what are commonly considered the first neural networks. The McCulloch-Pitts neuron compared a weighted sum input to a threshold. If the weighted sum was greater than or equal to the threshold, the neuron output was assigned a value of 1. If the weighted sum was less than the threshold, the neuron output was 0. These neurons were arranged into a network and weights adjusted to produce outputs that were combinations of logical functions[1,2]. The 1940's also brought about the first learning law for ANN's. In 1949 psychologist Donald Hebb developed a learning law based on the premise that the connection weights between two simultaneously active neurons should be strengthened[2].

The 1950's and 1960's are referred to as "The First Golden Age of Neural Networks."[2] Frank Rosenblatt was interested in applying sets of interconnected neurons using non-linear hard-limiting activation functions using iterative weight adjustment to typed character recognition. Rosenblatt's neural network was a hardware based network known as the Perceptron. The Perceptron had the ability to separate data inputs into one of two classes as long as there was a suitable set of connection weights to solve the problem. Oscillatory behavior resulted when suitable weights did not exist[2,3].

During the same time frame, Bernard Widrow and Marcian Hoff developed the "Adaptive Linear Neuron" (ADALINE) and "Multiple Adaptive Linear Neuron" (MADALINE). The difference between the ADALINE, MADALINE, and Perceptron was in the activation function and training. Widrow used a continuous non-linear activation function and a Least Mean Square training method[2,3]. The MADALINE network has been applied to adaptive modems, nulling of RADAR jammers, and in adaptive equalizers in telephone lines.

The limitations of the aforementioned networks led to loss of interest in neural network research in the 1970's. The main limitation of these networks was that they were only able to solve linearly separable problems. Both Rosenblatt and Widrow proposed that multiple layer networks would be able to overcome this limitation, but neither was able to apply their learning rules to train more than their single layer networks[1]. Although interest in neural networks subsided, Teuvo Kohonen applied associative networks to speech recognition, musical composition, and solved the classical "Traveling Salesman Problem." James Anderson also applied associative memory networks to medical diagnosis[2].

The advent of the Backpropagation algorithm (BPA) sparked a renewed interest in neural networks in the 1980's. The BPA was designed to train a multiple layer network by "propagating" information about the output error to the hidden layer neurons. The idea of training a multiple layer network using the output error was first introduced by Werbos in 1974. The method was again discovered independently by Parker and LeCun in 1985 and 1986 respectively. Finally, David Rumelhart and James McClelland refined and publicized Parker's work. The BPA is a gradient descent method that minimizes the mean squared error between desired and actual network outputs or a cost function representing desired system behavior[2,3,4].

Other work during the 1980's included the development of the Hopfield network by John Hopfield and David Tank. The Hopfield network is characterized by fixed weights and adaptive activation functions and were applied to solve constraint satisfaction problems like the "Traveling

Salesman Problem." The Neocognitron was applied by Kunihiko Fukushima to character recognition problems where the character was distorted by position or rotation[2].

The BPA solved the problems associated with the early neural networks, namely the ability to train multi-layer networks and to solve linearly separable problems, but it too had some limitations. The BPA had problems in becoming trapped in local minima characterized by oscillating weights and outputs, resulting in network failure to converge to the desired output. The BPA also can be slow in learning large order or complex systems. These two problems with the BPA prompted the development of the Extended Back-Propagation Algorithm (EBPA) by Chih-Liang Chen in 1992 at West Virginia University (WVU). The EBPA is a heterogeneous network that increases the learning capabilities of the BPA by allowing each individual neuron in the hidden and output layers to update its output characteristics by adjusting the range and slope of the activation function[4,8].

1.3 Applications of Artificial Neural Networks

Artificial neural networks are being applied in many different disciplines. The banking and financial industries use them to read documents and checks, evaluate credit applications, and perform corporate financial analyses. The medical field utilizes ANN's to assist in making diagnoses, analyze breast cancer cell tests, and evaluate EEG and ECG readings. The defense industry uses neural networks in differentiating between friendly and hostile targets, weapon steering, and target tracking. Manufacturing companies use them in process control, dynamic modeling, and product quality analysis. These are just a few fields, and limited number of examples for each, where artificial neural networks are used. Other fields use neural networks for speech production, speech recognition, non-linear mathematical modeling, design analysis, quality assessment, and various types of optimization problems[1,2].

Another main interest area of applying neural network technology is in aircraft control. Neural networks are being applied to flight path simulations, autopilot systems, flight control systems, and

component fault detection, identification, and accommodation[1]. In 1991 C.M. Ha applied ANN's to a pitch rate controller and to the lateral control laws of a 6 degree-of-freedom linear state-space aircraft model. Ha trained his networks off-line using the linear model, froze the weights, and tested the resulting neural structure on-line[3]. In 1989 Dr. Marcello Napolitano at WVU began applying neural network algorithms to Actuator Failure Detection Identification and Accommodation (AFDIA) problems[9-11]. Later, he extended research to include Sensor Failure Detection Identification and Accommodation (SFDIA) schemes[12]. Neural network research at WVU has also included neural network autopilots for an F-15 simulation model[3,13,14], neural network autopilot and SFDIA schemes applied to the NASA/Aurora Thesis aircraft[4,8,15], and signal reconstruction from black box data for crash investigations[16,17]. SFDIA research has been extended by hardware implementation using a custom made neural network PC board interfaced with a Simulink model of the longitudinal dynamics of an F-14 aircraft[18]. These studies have concluded that software and hardware based artificial neural network technology has the ability to learn complex non-linear dynamics on-line to perform basic autopilot functions and to detect and accommodate for a variety of hardware failures.

The motivation for using neural network technology in aircraft control is to replace the existing linear time invariant (LTI) based methods with more flexible and adaptable systems. Current methods rely on feedback gains calculated off-line at many points in the flight envelope. These gains are then stored in the flight computer and interpolated between when moving between models in the flight envelope. These methods are computationally intensive in the scheduling of the feedback gains and are not suitable for highly non-linear conditions. Furthermore, performance of controllers designed using LTI methods critically depends on accurate modeling of the system dynamics[3,13,14]. Neural networks using an on-line learning approach have the ability to learn the aircraft dynamics throughout the entire flight envelope, regardless if the system is linear or non-linear, time variant or time invariant, and with or without system and measurement noise[3,4,8,15].

1.4 Research Objectives

The 1/25 scale B747 model is being used to test neural network based autopilot controllers, SFDIA, and AFDIA schemes. Phase I of the project is to design, construct, and conduct the first flight of the vehicle. Phase II is to carry out flight testing that will produce the aerodynamic information necessary to build a simulation code based on a non-linear mathematical model. Phase III and Phase IV are to be conducted simultaneously to develop the fault detection and autopilot neural systems using the simulation model of the aircraft. Lastly, the neural network algorithms will be implemented into a flight computer system and tested.

While Phase I was being carried out, it was decided to develop the autopilot systems using a non-linear simulation code for the actual B747 built from readily available information. Once the flight testing was complete, the simulation code would then be altered and the neural network autopilots adjusted for the 1/25 simulation model. Therefore, the objective of this research was to develop and evaluate the neural network autopilot systems using a computer simulation of the B747 dynamics.

Five neural network autopilot systems were developed using a six degree-of-freedom non-linear mathematical model of the B747. Concurrent designs were performed using the Standard and Extended Back-Propagation algorithms. These autopilot systems were compared based on their abilities to perform maneuvers by changing speed and altitude, pitch angle, roll angle, or direction angle. They were also evaluated on their abilities to adapt at non-linear conditions. Non-linear conditions were simulated by simultaneously exciting the angular rates to values where their products and squares are not negligible, as assumed in linearized models of the aircraft dynamics.

The rest of this document contains: the derivation of the six degree-of-freedom non-linear equations of motion, step by step presentation of the EBPA, description of the simulation program, discussion of the neural network autopilot designs, results of the linear and non-linear tests, and, finally, conclusions and recommendations.

Chapter 2: Aircraft Equations of Motion

2.1 Introduction

The simulation code that served as the basis for the development of the neural network autopilot systems was built using a non-linear 6 degree-of-freedom mathematical model. This chapter will present an overview of the derivation of the equations of motion. The full details of the derivation can be found in Refs.[8,19,20,21].

2.2 Derivation of Equations of Motion

The equations are derived with the following assumptions:

- The mass of the airplane remains constant with time
- The aircraft is symmetric about the XZ plane
- The aircraft is a rigid body
- Aerodynamic and thrust forces are the only external forces acting on the aircraft

The equations of motion development begins with Newtonian expressions of conservation of linear and angular momentum applied to the aircraft as a non-rotating inertial axis system:

$$\sum \vec{F}_{ext} = \frac{d}{dt}(m\vec{V}) \tag{2.1}$$

$$\sum \vec{M}_{ext} = \frac{d}{dt}(\vec{h}) \tag{2.2}$$

The external forces and moments acting on the aircraft are shown in Figure 2.1. The quantities $\vec{V}$ and $\vec{h}$ are vectors representing the velocity and angular momentum, respectively, about the aircraft center of gravity. It is first necessary to convert these expressions from the non-rotating axis system to the rotating axis system using:

$$\sum \vec{F}_{ext} = \frac{\partial}{\partial t}(m\vec{V}) + \vec{\omega} \times (m\vec{V}) \tag{2.3}$$

$$\sum \vec{M}_{ext} = \frac{\partial}{\partial t}\vec{h} + \vec{\omega} \times \vec{h} \tag{2.4}$$

In the equations above, $\vec{\omega}$ is the angular velocity vector of the body axis system and $\vec{h}$ is related to the angular velocity vector by the body axis inertia matrix as expressed below.

$$\vec{h} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \vec{\omega} \tag{2.5}$$

Equations 2.3 and 2.4 can also be expressed as scalar equations with respect to the body axis by carrying out the vector mathematics and substituting the components of $\vec{V}$ (u,v,w) and $\vec{\omega}$ (p,q,r). Theses operations result in the following scalar expressions for the external forces and moments:

$$F_x = m(\dot{u} + qw - rv) \tag{2.6}$$

$$F_y = m(\dot{v} + ru - pw) \tag{2.7}$$

$$F_z = m(\dot{w} + pv - qu) \tag{2.8}$$

$$M_x = I_{xx}\dot{p} - I_{xy}\dot{q} - I_{xz}\dot{r} + (I_{zz} - I_{yy})qr + I_{yz}(r^2 - q^2) - I_{xz}pq + I_{xy}pr \tag{2.9}$$

$$M_y = -I_{xy}\dot{p} + I_{yy}\dot{q} - I_{yz}\dot{r} + (I_{xx} - I_{zz})rp + I_{xz}(p^2 - r^2) - I_{xy}qr + I_{yz}pq \qquad 2.10)$$

$$M_z = -I_{xz}\dot{p} - I_{yz}\dot{q} + I_{zz}\dot{r} + (I_{yy} - I_{xx})pq + I_{xy}(q^2 - p^2) - I_{yz}pr + I_{xz}qr \tag{2.11}$$

The forces expressed on the left hand sides of Equations 2.6 - 2.8 can be expressed as functions of thrust, aerodynamic, and gravity forces as shown below.

$$F_x = \bar{q}SC_x - mg\sin(\Theta) + T \tag{2.12}$$

$$F_y = \bar{q}SC_y + mg\sin(\Phi)\cos(\Theta) \tag{2.13}$$

$$F_z = \bar{q}SC_z + mg\cos(\Phi)\cos(\Theta) \tag{2.14}$$

Similarly, the moments expressed on the left hand sides of equations 2.9-2.10 can be expressed using aerodynamic and thrust forces as follows:

$$M_x = \bar{q}SbC_l \tag{2.15}$$

$$M_y = \bar{q}S\bar{c}C_m \tag{2.16}$$

$$M_z = \bar{q}SbC_n \tag{2.17}$$

It is assumed in the force and moment equations above that the engine thrust line lies on the X body axis.

It is now necessary to express the aircraft orientation relative to the earth fixed inertial reference system through the introduction of the Euler angles Θ, Φ, Ψ. This is accomplished through a series of axis transformations described by Figure 2.2. Note that the order of the transformation sequence is important because they do not act as vectors, where the commutative property holds.

Step 1: Translate earth fixed inertial axis X'Y'Z' such that the axis origin lies at the aircraft center of gravity and rename it as $X_1Y_1Z_1$.

Step 2: Rotate $X_1Y_1Z_1$ about Z_1 through an angle Ψ (heading angle) to give $X_2Y_2Z_2$.

Step 3: Rotate $X_2Y_2Z_2$ about Y_2 through an angle Θ (pitch angle) to give $X_3Y_3Z_3$.

Step 4: Rotate $X_3Y_3Z_3$ about X_3 through an angle Φ (roll angle) to give the original body axis XYZ.

The resulting roll, pitch, and yaw rates can be expressed by this rotation sequence as functions of the Euler angles in matrix format.

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\Phi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\Phi) & \sin(\Phi) \\ 0 & -\sin(\Phi) & \cos(\Phi) \end{bmatrix} \begin{bmatrix} 0 \\ \dot{\Theta} \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\Phi) & \sin(\Phi) \\ 0 & -\sin(\Phi) & \cos(\Phi) \end{bmatrix} \begin{bmatrix} \cos(\Theta) & 0 & -\sin(\Theta) \\ 0 & 1 & 0 \\ \sin(\Theta) & 0 & \cos(\Theta) \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\Psi} \end{bmatrix} \quad (2.18)$$

Solving this matrix equation gives expressions for Euler angular rates known as the kinematic equations:

$$\dot{\Phi} = p + q\sin(\Phi)\tan(\Theta) + r\cos(\Phi)\tan(\Theta) \quad (2.19)$$

$$\dot{\Theta} = q\cos(\Phi) - r\sin(\Phi) \quad (2.20)$$

$$\dot{\Psi} = q\sin(\Phi)\sec(\Theta) + r\cos(\Phi)\sec(\Theta) \quad (2.21)$$

The six degree-of-freedom mathematical model is given by the force, moment, and kinematic equations. Notice that equations 2.6-2.8 are dependent on the velocity components u, v, and w. These equations need to be expressed in terms of a polar coordinate system defined by α, β, and V in order to be compatible with readily measured aircraft parameters. This system conversion can be accomplished by employing the following relationship between the aircraft body axis (XYZ), stability axis (differing from body axis by α), and wind axis (defined using $\vec{V}$ as the positive X axis). Figure 2.3 indicates the following:

$$V = \sqrt{u^2 + v^2 + w^2} \tag{2.22}$$

$$\alpha = \tan^{-1}\left(\frac{w}{u}\right) \tag{2.23}$$

$$\beta = \sin^{-1}\left(\frac{v}{V}\right) \tag{2.24}$$

Evaluating the derivatives of the above equations and working with equations 2.6-2.8 results in the following expressions for the wind axis translational accelerations.

$$\dot{V} = \frac{F_x}{m}\cos(\alpha)\cos(\beta) + \frac{F_y}{m}\sin(\beta) + \frac{F_z}{m}\sin(\alpha)\cos(\beta) \tag{2.25}$$

$$\dot{\alpha} = \frac{(F_z\cos(\alpha) - F_x\sin(\alpha))}{mV\cos(\beta)} + q - p\cos(\alpha)\tan(\beta) - r\sin(\alpha)\tan(\beta) \tag{2.26}$$

$$\dot{\beta} = -\frac{\cos(\alpha)\sin(\beta)}{mV}F_x + \frac{\cos(\beta)}{mV}F_y - \frac{\sin(\alpha)\sin(\beta)}{mV}F_z + p\sin(\alpha) - r\cos(\alpha) \tag{2.27}$$

The final six degree of freedom non-linear aircraft mathematical model can now be obtained by substituting the force and moment components into equations 2.9-2.11 and equations 2.25-2.27. The resulting expressions for the accelerations above are the following:

$$\dot{V} = -\frac{\overline{q}S}{m}C_{D_{wind}} + g\cos(\Phi)\cos(\Theta)\sin(\alpha)\cos(\beta) + g\sin(\Phi)\cos(\Theta)\sin(\beta)$$

$$- g\sin(\Theta)\cos(\alpha)\cos(\beta) + \frac{T}{m}\cos(\alpha)\cos(\beta) \tag{2.28}$$

$$\dot{\alpha} = q - p\tan(\beta)\cos(\alpha) - r\tan(\beta)\sin(\alpha) - \frac{\overline{q}S}{mV\cos(\beta)}C_L$$

$$+\frac{g\cos(\Theta)\cos(\Phi)\sin(\alpha)}{V\cos(\beta)} + \frac{g\sin(\Theta)\sin(\alpha)}{V\cos(\beta)} - \frac{T\sin(\alpha)}{mV\cos(\beta)} \tag{2.29}$$

$$\dot{\beta} = p\sin(\alpha) - r\cos(\alpha) + \frac{\overline{q}S}{mV}C_{Y_{wind}} + \frac{g}{V}\cos(\beta)\cos(\Theta)\sin(\Phi)$$

$$-\frac{g}{V}\sin(\beta)\cos(\Theta)\cos(\Phi)\sin(\alpha) + \frac{g}{V}\sin(\beta)\sin(\Theta)\cos(\alpha) - \frac{gT}{mV}\sin(\beta)\cos(\alpha) \tag{2.30}$$

The final equations for roll, pitch, and yaw rates are given next:

$$\dot{p} = \frac{I_{xy}}{I_{xx}}(\dot{q} - pr) + \frac{I_{xz}}{I_{xx}}(\dot{r} + pq) + \frac{(I_{yy} - I_{zz})}{I_{xx}}qr + \frac{I_{yz}}{I_{xx}}(q^2 - r^2) + \overline{q}SbC_l \tag{2.31}$$

$$\dot{q} = \frac{I_{xy}}{I_{yy}}(\dot{p} + qr) + \frac{I_{yz}}{I_{yy}}(\dot{r} - pq) + \frac{(I_{zz} - I_{xx})}{I_{yy}}pr + \frac{I_{xz}}{I_{yy}}(r^2 - p^2) + \frac{1}{I_{yy}}\overline{q}S\overline{c}C_m \tag{2.32}$$

$$\dot{r} = \frac{I_{xz}}{I_{zz}}(\dot{p} - qr) + \frac{I_{yz}}{I_{zz}}(\dot{q} + pr) + \frac{(I_{xx} - I_{yy})}{I_{zz}}pq + \frac{I_{xy}}{I_{zz}}(p^2 - q^2) + \overline{q}SbC_n \tag{2.33}$$

The kinematic equations are restated below:

$$\dot{\Phi} = p + q\sin(\Phi)\tan(\Theta) + r\cos(\Phi)\tan(\Theta) \tag{2.34}$$

$$\dot{\Theta} = q\cos(\Phi) - r\sin(\Phi) \tag{2.35}$$

$$\dot{\Psi} = q\sin(\Phi)\sec(\Theta) + r\cos(\Phi)\sec(\Theta) \tag{2.36}$$

Expressions for accelerations in the normal, x, y, and z directions in g units are given below.

$$a_{Norm} = \frac{\bar{q}S}{mg} C_{Norm} \tag{2.37}$$

$$a_x = -\frac{\bar{q}S}{mg} C_A + \frac{T}{mg} \tag{2.38}$$

$$a_y = \frac{\bar{q}S}{mg} C_Y \tag{2.39}$$

$$a_z = a_{Norm} \cos(\Theta)\cos(\Phi) \tag{2.40}$$

Finally, aerodynamic force and moment coefficients are defined using a linear component build-up method. The lift, drag, and side force coefficients are given below.

$$C_L = C_{L_0} + C_{L_\alpha}\alpha + C_{L_{\delta_E}}\delta_E + C_{L_{i_h}} i_h + C_{L_{\dot{\alpha}}}\left(\frac{\bar{c}\dot{\alpha}}{2V}\right) + C_{L_q}\left(\frac{\bar{c}q}{2V}\right) \tag{2.41}$$

$$C_D = C_{D_0} + C_{D_\alpha}\alpha + C_{D_{i_h}} i_h + C_{D_{\delta_E}}\delta_E \tag{2.42}$$

$$C_Y = C_{Y_0} + C_{Y_\beta}\beta + C_{Y_{\delta_A}}\delta_A + C_{Y_{\delta_R}}\delta_R + C_{y_p}\left(\frac{b}{2V}\right)p + C_{y_r}\left(\frac{b}{2V}\right)r \tag{2.43}$$

The lift and drag force coefficients are also related to the normal and axial force coefficients through the following equations.

$$C_L = C_{Norm}\cos(\alpha) - C_A \sin(\alpha) \tag{2.44}$$

$$C_D = C_{Norm}\sin(\alpha) + C_A \cos(\alpha) \tag{2.45}$$

Similarly, the wind axis force coefficients can be expressed as functions of the drag and side force coefficients.

$$C_{D_{wind}} = C_D \cos(\beta) - C_Y \sin(\beta) \tag{2.46}$$

$$C_{Y_{wind}} = C_D \sin(\beta) + C_Y \cos(\beta) \tag{2.47}$$

Lastly, the pitching, rolling, and yawing moment coefficients are listed below.

$$C_m = C_{m_0} + C_{m_\alpha}\alpha + C_{m_{i_h}} i_h + C_{m_{\delta_E}}\delta_E + C_{m_q}\left(\frac{\bar{c}q}{2V}\right) + C_{m_{\dot{\alpha}}}\left(\frac{\bar{c}\dot{\alpha}}{2V}\right) \tag{2.48}$$

$$C_l = C_{l_0} + C_{l_\beta}\beta + C_{l_{\delta_A}}\delta_A + C_{l_{\delta_R}}\delta_R + C_{l_p}\left(\frac{b}{2V}\right)p + C_{l_r}\left(\frac{b}{2V}\right)r \tag{2.49}$$

$$C_n = C_{n_0} + C_{n_\beta}\beta + C_{n_{\delta_A}}\delta_A + C_{n_{\delta_R}}\delta_R + C_{n_p}\left(\frac{b}{2V}\right)p + C_{n_r}\left(\frac{b}{2V}\right)r \tag{2.50}$$

2.3 Standard Atmosphere

One last detail needs to be introduced in reference to the equations of motion. These equations discussed in this chapter are dependent on the dynamic pressure, q, which is a function of the aircraft speed and density of the air at altitude. The standard atmosphere is a model of the air properties resulting from experiments conducted by the U.S. Air Force in 1959[22]. This model tabulates average values of pressure, temperature, and density as functions of altitude up to approximately 345,000 feet. This discussion will be limited to the first two regions of the standard atmosphere, which covers up to approximately 82,000 ft.

The first region of the standard atmosphere is a gradient region where the temperature decreases linearly with increasing altitude. This region begins at sea level and ends at 36,089 feet. Expressions for the density and temperature are given below.

$$\rho = \rho_0 \left(\frac{T}{T_0} \right)^{-\left[\left(g/_{-3.567\times10^{-3}R} \right)+1\right]} \tag{2.51}$$

$$T = T_0 - 3.567 \times 10^{-3}(h - h_0) \tag{2.52}$$

the second region of the standard atmosphere is an isothermal region that encompasses altitudes from 36,089 feet to 82,020 feet. The density as a function of altitude in this region is given as:

$$\rho = \rho_1 e^{-(g/RT)(h-h_1)} \tag{2.53}$$

Additional regions are not included here or in the numerical simulation code since the aircraft does not operate at altitudes that exceed the second region of the standard atmosphere. Full details on the definition of the standard atmosphere can be found in Refs.[8,22].

The equations of motion presented here are left in the most general form. These are usually reduced further to produce either linear s-domain transfer functions or linear time domain state space models. The general model presented here doesn't lend itself to discussion of dynamic modes expressed as the roots of the s-domain characteristic equations or the eigenvalues of the system state matrix. The full details of linearizing this model are not presented here but can be found in Refs.[8,19,20]. Mention will be made of the use of a linear model of the aircraft dynamics to give an awareness of the dynamic modes and handling qualities.

Chapter 3: The Extended Back Propagation Algorithm

3.1 Introduction

The diffusion of the Back-Propagation algorithm renewed interest in neural network research by introducing a method of training multi-layer networks. This removed the limitation of applying neural networks to linearly separable problems. However, the BPA, referred to here as the Standard Back-Propagation algorithm, has been shown to have limitations. In attempting to minimize a cost function the SBPA has a tendency to become trapped in local minima, characterized by oscillating weights and network outputs, which the network may or may not eventually break out of. In on-line learning control problems becoming trapped may cause system instability and divergence. Additionally, the SBPA can be slow in learning the behavior of complex systems characterized by non-linear dynamics. Quicker learning can be achieved by increasing the learning rate, but this also increases the tendency to become trapped in local minima. Many solutions to these problems have been proposed, most notably of which is the addition of a momentum term. This modification adds a portion of the previous weight change to the new connection weights[2]. In theory, when the network is moving in the direction toward the solution, this additional term will give "momentum" to weight changes in that direction. The pit fall of this method is that when the network is moving away from the desired solution, the momentum term will continue to push the network in the wrong direction. Again, this behavior is undesirable in on-line learning control applications. These problems with the SBPA prompted the development of the Extended Backpropagation Algorithm.

3.2 The Extended Back Propagation Algorithm (EBPA)

The heterogeneous structure of the EBPA trained network lends to its increased learning capabilities over the SBPA. Each neuron in the hidden and output layers has the ability to update the upper limit, lower limit, and slope of its individual sigmoid activation function. In contrast, every

hidden and output neuron in the SBPA network uses the same activation function and updates only the connection weights. This section presents the step by step process of the EBPA. A general neural network trained by the EBPA is given in Figure 3.1.

The difference between the EBPA and the SBPA is characterized by the activation function. The SBPA uses a bipolar sigmoid function given by the equation below.

$$f(net) = \frac{2}{1+e^{-net}} - 1 \tag{3.1}$$

The EBPA uses a "modified" sigmoid function. The modified sigmoid activation function is given by the following equation:

$$f(net, U, L, T) = \frac{U - L}{1+e^{-net/T}} + L \tag{3.2}$$

The U, L, and T arguments correspond to general upper bound, lower bound, and temperature (slope) of the sigmoid function respectively. Each of these parameters is updated for every hidden and output neuron during training. Note that the standard sigmoid function results if U, L, and T are 1, -1, and 1 respectively. Figure 3.2 describes the modified sigmoid function. The Extended Back-Propagation algorithm is presented step-by-step below. The full details of the EBPA are outlined in Ref.[11].

Step #0 (off-line): Initialize the connection weights (v and w), upper and lower bounds (U, L), temperatures (T), and thresholds (Θ, Γ) as random numbers between -0.5 and 0.5. Also assign values to the learning rates (η_{net}, η_{UL}, and η_T). Note that the SBPA only uses η_{net}.

Foreword Phase

Step #1: Calculate the input to the hidden layer neurons using the following equation.

$$net_i = \sum_{l}^{m} v_{li} I_l + \Gamma_i \; ; \quad \text{i} = 1,\ldots,\text{k (neurons in hidden layer)} \tag{3.3}$$

Step #2: Calculate hidden layer output using modified sigmoid function (Equation (3.2)).

Step #3: Calculate the input to the output layer using the equation below.

$$net_j = \sum_{i=1}^{k} w_{ij} H_i + \Theta_j \; ; \; \text{j} = 1,\ldots,\text{n (number of output neurons)} \tag{3.4}$$

Step #4: Calculate Output Layer Output using modified sigmoid function given by Equation (3.2).

Step #5: Evaluate the error or cost function to propagate back to the network.

Backward Phase

Step #6: Back-Propagation of error to output layer through error signals:

$$\delta_{net_j} = f'_{net_j}(Y_j - O_j) \tag{3.5}$$

$$\delta_{U_j} = f'_{U_j}(Y_j - O_j) \tag{3.6}$$

$$\delta_{L_j} = f'_{L_j}(Y_j - O_j) \tag{3.7}$$

$$\delta_{T_j} = f'_{T_j}(Y_j - O_j) \tag{3.8}$$

where:

$$f'_{net_i} = \frac{\partial f}{\partial net_j} = -\frac{(H_i - U_i)(H_i - L_i)}{T_i(U_i - L_i)} \tag{3.9}$$

$$f'_{U_i} = \frac{\partial f}{\partial U_i} = \frac{1}{1 + e^{\frac{-net_i}{T_i}}} \tag{3.10}$$

$$f'_{L_i} = \frac{\partial f}{\partial L_i} = 1 - f'_{U_i} \tag{3.11}$$

$$f'_{T_i} = \frac{\partial f}{\partial T_i} = -\frac{net_i f'_{net_i}}{T_i} \tag{3.12}$$

Step #7: Back-Propagation of the error to the hidden layer through error signals:

$$\delta_{net_i} = f'_{net_i} \sum_j^n \delta_{net_j} W_{ij} \tag{3.13}$$

$$\delta_{U_i} = f'_{U_i} \sum_j^n \delta_{U_j} W_{ij} \tag{3.14}$$

$$\delta_{L_i} = f'_{L_i} \sum_j^n \delta_{L_j} W_{ij} \tag{3.15}$$

$$\delta_{T_i} = f'_{T_i} \sum_j^n \delta_{T_j} W_{ij} \tag{3.16}$$

where:

$$f'_{net_j} = \frac{\partial f}{\partial net_j} = -\frac{(O_j - U_j)(O_j - L_j)}{T_j(U_j - L_j)} \tag{3.17}$$

$$f'_{U_j} = \frac{\partial f}{\partial U_j} = \frac{1}{1 + e^{\frac{-net_j}{T_j}}} \tag{3.18}$$

$$f'_{L_j} = \frac{\partial f}{\partial L_j} = 1 - f'_{U_j} \tag{3.19}$$

$$f'_{T_j} = \frac{\partial f}{\partial T_j} = -\frac{net_j f'_{net_j}}{T_j} \tag{3.20}$$

Step #8: Update connection weights:

$$\Delta W_{ij}(k+1) = \eta_{net}\delta_{net_j} O_j + \alpha_{net}\Delta W(k) \tag{3.21}$$

$$\Delta V_{ij}(k+1) = \eta_{net}\delta_{net_i} H_i + \alpha_{net}\Delta V(k) \tag{3.22}$$

Step #9: Update Thresholds:

$$\Delta\Gamma_i(k+1) = \eta_{net}\delta_{net_i} + \alpha_{net}\Delta\Gamma_i(k) \tag{3.23}$$

$$\Delta\Theta_j(k+1) = \eta_{net}\delta_{net_j} + \alpha_{net}\Delta\Gamma_j(k) \tag{3.24}$$

Step #10: Update Upper Bounds:

$$\Delta U_j(k+1) = \eta_{UL}\delta_{U_j} + \alpha_{UL}\Delta U_j(k) \tag{3.25}$$

$$\Delta U_i(k+1) = \eta_{UL}\delta_{U_i} + \alpha_{UL}\Delta U_i(k) \tag{3.26}$$

Step #11: Update Lower Bounds:

$$\Delta L_j(k+1) = \eta_{UL}\delta_{L_j} + \alpha_{UL}\Delta L_j(k) \tag{3.27}$$

$$\Delta L_i(k+1) = \eta_{UL}\delta_{L_i} + \alpha_{UL}\Delta L_i(k) \tag{3.28}$$

Step #12: Update Temperatures:

$$\Delta T_j(k+1) = \eta_T \delta_{T_j} + \alpha_T \Delta T_j(k) \tag{3.29}$$

$$\Delta T_i(k+1) = \eta_T \delta_{T_i} + \alpha_T \Delta T_i(k) \tag{3.30}$$

this procedure can be directly applied to the aircraft neural network autopilot controllers. The details of selecting the input data pattern, number hidden layer neurons, output, and performance index will be discussed in Chapter 5.

3.3 On-Line Learning

The typical training of a neural network involves presenting the network with sets of training data consisting of input and corresponding desired output values. Once the network has been trained to desired performance levels, the structure is "frozen" (learning disabled) and applied to perform the task it was trained for. This requires that enough training data be presented to represent the entire range of possible network inputs. In the case of an aircraft, this would require vast amounts of flight data covering the entire flight envelope. Although neural networks have remarkable capabilities of generalizing their training to new situations, there is no practical way for them to be trained to cover every circumstance. On-line learning is an indirect technique where the network learns the dynamics of the system as it operates. Instead of presenting the network with data sets of inputs and desired output pairs, the network is presented with the aircraft states for several previous time steps. The output of the network is the control input to the system. The network continuously (learning never disabled) updates its structure in order to find the control inputs necessary to minimize a cost function that is related to the desired aircraft state. This allows the neural network autopilot systems to learn the local dynamics, of the system; linear or non-linear, time invariant or time variant, at any point in the flight envelope.

Chapter 4: Numerical Simulation

4.1 Introduction

A simulation software was developed for the Boeing 747 using Microsoft® Visual Basic® version 5.0. This software combined the six degree-of-freedom non-linear mathematical model with neural network autopilot algorithms using the Standard and Extended Back-Propagation algorithms. This chapter will give details on the combination of these topics in the numerical simulation program. This chapter will also discuss the simulation of the linear and non-linear tests of the ANN autopilots.

4.2 Aircraft Geometric and Aerodynamic parameters

The B747 is a transport aircraft with commercial and military applications. The aircraft has a wing span of 196 feet and 4 engines giving a total thrust of 192,000 lbs. The primary control surfaces are: elevator, rudder, and ailerons. The elevator and rudder have maximum deflections of ±25 degrees while the maximum aileron deflection is ±20 degrees. The ailerons are assumed to move symmetrically giving the total aileron surface deflection. Aerodynamic data were not available to include spoilers and flaps in the simulation. The geometric properties, mass characteristics, and aerodynamic coefficients corresponding to low and high cruise conditions are given in Ref.[19] and summarized in Table 4.1.

The dynamic modes of the aircraft were reviewed by consulting linear approximations of the aircraft dynamics presented in Refs.[19,20]. The short period and phugoid are the longitudinal dynamic modes, which exhibit oscillatory response characteristics, described by 2 pairs of complex conjugate s-domain poles. The lateral dynamics are characterized by 2 real poles and 1 pair of complex conjugate poles corresponding to rolling, spiral, and Dutch roll modes. The longitudinal mode of concern is the phugoid at high cruise. It has a very low natural frequency (about 0.08) and small negative damping ratio (about -0.028) implying that the uncompensated system will tend to

slowly diverge in response to an elevator input. The Dutch roll mode is the lateral directional mode of concern for both low and high cruise conditions. This mode is characterized by coupled rolling and yawing motion which naturally tends to be lightly damped for swept wing aircraft[24]. The Dutch roll damping is approximately 0.1 and 0.05 for low and high cruise respectively. The high cruise phugoid is easily controlled by the pilot or by a longitudinal stability augmentation system. Likewise, the Dutch roll mode can be compensated for using a yaw damper control system. However, on-line learning neural networks have the ability to control these modes without using additional compensation.

4.3 Simulation of Aircraft Dynamics

The simulation program features a graphical interface utilizing various Windows features, such as scroll bars, menus, and buttons accessible through key strokes or mouse clicks. This user interface is shown in Figure 4.1. The simulation begins with the choice of flight condition through the "Flight Condition" menu. When a flight condition is chosen the aerodynamic, mass, and geometric properties are initialized. Additionally, the aircraft trim elevator deflection, angle of attack, and throttle settings are calculated. Once the flight condition is chosen the "Start" button is unlocked and can be clicked to begin the simulation.

Figure 4.2 is a block diagram describing the program steps. The system inputs are passed to the differential equations, which are evaluated and integrated using a fourth order Runge-Kutta technique to find the aircraft states[25]. These states are then displayed in the text boxes on the screen and saved to data files if the appropriate selection has been made from the "Save Data" menu. A check is then made to see if the autopilot systems have been engaged through the "Autopilots" menu. If the autopilots are engaged, the neural networks determine the control inputs for the next time step. Refer back to Figure 3.1 for details on the neural network autopilot block in Figure 4.2. If the autopilots have not been engaged, the program takes the user defined system inputs made by moving the throttle and control surface scroll bars. Other features of the software include user ability, through a menu

selection, to view plots on-line of any of the aircraft states. Additionally, the user has the abilities to change the default learning rates of the networks and to engage the momentum terms to aid network learning performance.

4.4 Simulation of Linear and Non-linear Tests

The neural networks were used to perform linear maneuvers involving commanded altitude and speed inputs, roll, pitch, or direction angle inputs. Modified definitions for settling times were introduced. In the case of the AHNN, the settling time is the time from when the network is engaged until oscillation within ±5 feet is achieved. The settling time for the SHNN is the time to achieve oscillation within ±1 ft/s. The settling times for the RHNN, PHNN, and DHNN are defined to be the time from network engagement until the plane reaches and remains within ±0.5 degrees of the input reference value.

Non-linear studies focused on two areas. The first focused on the coupling between the longitudinal and lateral directional dynamics by simultaneously exciting p and q. This state was simulated by moving the aileron scroll bar to -20 degrees and the elevator scroll bar to 25 degrees. Once both p and q reached a specified tolerance level, the RHNN and PHNN autopilots were engaged. They were instructed to return the aircraft to a constant pitch angle of three degrees and wings level condition. The settling times recorded for these runs were selected to reflect the time to reach and maintain ±0.5 degrees of the reference roll and pitch angles. The second study involved simultaneously exciting all of the non-linear terms in the equations of motion. All three angular rates, p, q, and r, were raised to various tolerance levels, at which time the PHNN and DHNN were engaged. These networks were instructed to return to constant three degrees of pitch, wings level, and zero yaw rate. The settling times recorded for these runs were selected to reflect the time to reach and maintain ±0.5 degrees of the reference roll and pitch angles and yaw rate oscillation between ±0.1 deg/s. The results of these tests are presented in Chapter 6.

Chapter 5: On-Line Learning Neural Network Autopilot Systems

5.1 Introduction

The previous chapter discussed the combination of the equations of motion with the neural network autopilot controllers in a simulation code. This chapter will discuss the design of the autopilots and the final network structures. The autopilot systems considered were:

- Altitude Hold (AHNN)
- Pitch Hold (PHNN)
- Airspeed Hold (SHNN)
- Roll Angle Hold (RHNN)
- Direction (heading) Angle Hold (DHNN)

On-line training using both SBPA and EBPA algorithms was chosen in order to avoid facing the same dilemmas as classical autopilots using gain scheduling. Although the networks could be trained off-line using non-linear simulation models, their effectiveness would still be limited to situations resembling the training model while requiring significant training time and massive amounts of training data. Furthermore, training the neural networks to have the abilities to perform maneuvers at linear maneuvers as well as regain steady state conditions when at non-linear conditions would be extremely challenging. On-line training adds flexibility by allowing the neural controllers to adjust their structures to learn the dynamics of the system at any given time to perform linear maneuvers or adapt at non-linear conditions while requiring no initial training. The final neural network designs are presented below. Their design parameters were chosen based on observing their performances at linear and non-linear conditions.

5.2 Design Considerations

There are many degrees of freedom in designing on-line learning neural network controllers. Careful consideration of these design parameters is important to the performance of the network controllers. These design parameters include:

- Performance index to be minimized
- Selection of input data pattern
- Network topology (single or multiple hidden layer, number of neurons per hidden layer)
- Learning rate

Neural network autopilot performance is most critically dependent on the selection of the performance index to be minimized. It seems logical that the quantities used in the feedback loops of classical controllers should be used in the performance indices to be minimized in neural network control systems. Previous investigations found these variables to be effective quantities to be used in the performance indices when arranged to resemble error formulations of classical PI, PD, or PID controllers[3,13]. See Refs.[20,23] for block diagrams of classical autopilot systems. This allows a theoretical link to be made between classical and neural network control methods. A general form for the performance indices is given by:

$$J = w(X - X_{ref}) + v(\dot{X} - \dot{X}_{ref}) \tag{5.1}$$

where w and v are weighting factors, X indicates the general variable to be controlled, and the reference values are chosen by the user. Terms can be added to this form to change the behavior of the controller. Although neural network controllers are not designed using the classical specifications of

rise time, settling time, peak overshoot, and delay time, they can be directly influenced by the selection of the weighting terms. The weighting terms in the performance indices presented below were chosen in order to minimize: the peak overshoot in linear maneuvers, transient oscillatory behavior for both linear maneuvers and non-linear excitations, and settling times.

Previous studies have suggested that careful selection of the input data patterns is another important factor in controller performance[13]. Since each autopilot system is being tasked to learn the non-linear relationship between the system control inputs and the resulting aircraft states, careful consideration of the inputs to the networks must be made. Selecting input data that is not strongly related to the controlled state and the network output will result in undesirable performance. With some exceptions in the AHNN and DHNN, the same quantities used in the performance indices and the associated control values are used as inputs to the neural networks. The aforementioned reference also indicates that on-line learning neural networks are robust to the size of the input data window. However, the data window should be large enough to allow the neural controllers to adequately learn the system dynamics, but small enough to keep computational costs to a minimum.

The studies mentioned in Ref.[13] mentioned that neural controllers are robust in the selection of the number of hidden layers. It has been shown that single hidden layer networks are as capable of performing the control task as multiple layer networks[13]. In fact, it has been shown that neural network architectures with at least one hidden layer have the ability to map any non-linear dynamics after being adequately trained[4,8,13,]. Therefore, single hidden layer networks were chosen for the autopilot systems. Additional hidden layers slow the process by adding more connections resulting in more multiplication operations in the foreword phase and more complex operations in the feedback phase as the output error has to be passed to an additional hidden layer to update its structure.

Selection of the network learning rates also strongly affects the performance of the neural autopilots. Since on-line learning is being used, it is necessary to select the largest possible learning rates to aid in the learning of the system dynamics. However, learning rates that are too large will

result in longer converge times caused by significant overshoot of target values due to large initial changes in the network free parameters. The networks will eventually converge at linear conditions, but at non-linear conditions where the network is trying to regain steady state conditions, oscillating behavior may cause an unrecoverable aircraft state. Observing simulation output indicated that the EBPA works well with learning rates that range from 0.2 to 0.5 while the SBPA performs best with learning rates on the order of 0.001. The network learning rates were chosen to minimize the occurrences of local minima trappings as well as satisfy criterion mentioned for selection of the weighting terms in the performance indices.

The considerations of performance index, input data pattern, network topology, and learning rate were used to design the autopilot systems mentioned below. The SBPA and EBPA networks have the same input data, performance indices, and output values. The two algorithms differ in the network structures and learning rates.

5.3 Altitude Hold Autopilot (AHNN)

The AHNN can be activated to maintain a specified cruising altitude or to perform a change in altitude maneuver by introducing a new reference altitude value. The inputs for this autopilot were chosen to reflect the relationships between altitude, altitude rate of change, pitch angle, and pitch rate, and elevator control inputs. The neural inputs, output, and performance index are given below:

Inputs: Θ , q, h, $\dot{h}, \delta_E$

Output: δ_E

PI: $J = 0.05(h - h_{ref}) + 0.75(\dot{h} - \dot{h}_{ref}) + 10(q - q_{ref})$

Notice that the performance index differs from equation 5.1 by the additional term dealing with the pitch rate. This term was chosen to penalize pitching motion, which would oscillate significantly

enough to cause the system behavior to become highly unstable when performing altitude change maneuvers.

5.4 Pitch Hold Autopilot (PHNN)

The pitch hold autopilot has several duties. Under linear conditions it is used to maneuver the aircraft to desired reference pitch angles. When engaged under non-linear excitation this network is used as a pitch damper. It cancels pitch rate and returns the aircraft to an arbitrary reference pitch angle (3 degrees in this research). The neural network input quantities were chosen to reflect the relationship between the pitch angle, pitch rate, and elevator control inputs. The neural inputs, output and performance index are stated below:

Inputs: Θ , q , δ_E

Output: δ_E

PI: $J = 0.25(\Theta - \Theta_{ref}) + 0.5(q - q_{ref})$

5.5 Airspeed Hold Autopilot (SHNN)

The job of this controller was to maintain or maneuver to a reference cruising speed. The inputs to the neural network allow the network to learn the relationship between the airspeed, acceleration, and thrust control inputs. The inputs, output, and performance index are given below:

Inputs: V , $\dot{V}$, δ_T

Output: δ_T

PI: $J = 0.5(V_{ref} - V) + 0.8(\dot{V} - \dot{V}_{ref})$

5.6 Roll Hold Autopilot (RHNN)

The Roll Hold neural network autopilot has similar function to the pitch hold network. Under linear conditions it can be used to maneuver and maintain a desired roll angle. When employed during non-linear excitations it is used as a roll damper whose job is to stop angular motion and return to steady state roll condition ($\Phi = 0$). The input variables were chosen to represent the relationships between roll angle, roll rate, aileron deflection, and rudder control inputs. The inputs, output and performance index are given below:

Inputs: $\Phi, p, \delta_A, \delta_R$

Output: δ_A

PI: $J = 0.25(\Phi_{ref} - \Phi) + 2.0(p_{ref} - p)$

5.7 Direction Hold Autopilot (DHNN)

The DHNN has the most difficult job of any neural controller designed for this project. This is due to the complex dynamic modes in the lateral directional dynamics. The lowly damped Dutch roll root necessitates the ability of the network to replace the yaw damping function in the typical flight control system. The DHNN serves two functions: to allow a direction change maneuvers by presenting the network with reference heading angles, and to compensate when subjected to non-linear excitations.

This network is unique because it has two outputs: δ_A and δ_R. This was chosen since the single rudder output version designed to stop yawing motion did not cooperate with the RHNN. The RHNN had difficulty compensating for the rolling motion induced by rudder inputs. Therefore, since a large correlation exists between rolling and rudder input, it was decided to try a multiple output network configuration. Additionally, each output neuron has a performance index relating to it. One performance index updates the network according to rolling motion and the other according to yawing.

Also, the network uses a separate set of performance indices for maneuvering and non-linear compensation. Classical heading angle command autopilot systems use roll rate, roll angle, and heading angle in feedback, while yaw dampers feedback yaw rate[23]. In other words, the heading angle command has an embedded roll command system in it. This was reflected in the choice of the performance indices for the DHNN. The inputs, outputs, and performance indices are stated below.

Inputs: $\Phi, p, \Psi, r\ \delta_A, \delta_R$

Output: δ_A, δ_R

PI: $J_1 = 0.1(\Psi - \Psi_{ref}) + 0.7(r - r_{ref}) + 0.25(\dot{r} - \dot{r}_{ref})$

$$J_2 = 0.1(\Phi_{ref} - \Phi) + 2.5(p_{ref} - p) + 0.25(\dot{p}_{ref} - \dot{p})$$

At non-linear conditions the performance indices are altered slightly. A penalty is not placed on the heading angle. The only concern is to damp out the yawing motion and return to level wing position. The performance indices used for this task are given as follows:

PI: $J_1 = 0.7(r - r_{ref}) + 0.25(\dot{r} - \dot{r}_{ref})$

$$J_2 = 0.3(\Phi_{ref} - \Phi) + 2.5(p_{ref} - p)$$

The final network structures presented in this chapter displayed the best observed performance in terms of transient response characteristics and settling times for the entire range of linear and non-linear tests. The designs began with considering the aforementioned guidelines for choosing neural network autopilot parameters and referring to neural network structures for autopilot designs presented in Refs.[3,13,14] as examples. These considerations were then applied to the B747 neural network autopilot designs. The weighting factors in the performance indices and learning rates were adjusted in order to reduce oscillating behavior while keeping settling times as small as possible.

Chapter 6: Evaluation of Adaptation Capabilities at Linear and Non-linear Conditions

6.1 Linear Maneuvers Using AHNN and SHNN Autopilots

The altitude and speed networks were engaged together to perform the maneuvers corresponding to the flight conditions listed in Table 6.1 for low cruise and Table 6.2 for high cruise. These maneuvers involved altitude changes up to 1500 feet and speed changes up to 130 ft/s. The initial cruising condition selected was presented to the AHNN and SHNN first in order to allow the networks to update their random structures. The resulting AHNN and SHNN responses are given in Figures 6.1-6.4 for low cruise and in Figures 6.5-6.10 for high cruise. The settling times for these maneuvers are reported in Tables 6.3 and 6.4. At low cruise, the SBPA and EBPA AHNN's results are similar. The EBPA does, however, move to the desired altitude with less oscillation, resembling a first order response. The speed responses differ significantly between the SBPA and EBPA SHNN's. The EBPA responses show moderate overshoot with quick convergence to the commanded value for all conditions. The SBPA follows similar trends for the first few maneuvers but soon shows difficulty in achieving the reference value and eventually fails. When failure occurred the throttle locked at maximum value. At this time the SHNN was disengaged and the throttle reset to the initial condition in order to continue AH maneuvers. Similar behavior is displayed at high cruise for both the SBPA and EBPA speed and altitude networks. The SBPA speed hold network did fail and was disengaged, as in the low cruise case, to allow maneuvering in altitude to continue. At both cruising conditions the AHNN using the EBPA appeared to perform consistently throughout the cycle where the SBPA performance improved slightly with continued training. The EBPA trained AHNN and SHNN were able to perform the maneuvers at high cruise while accommodating for the moderately unstable phugoid.

6.2 Linear Maneuvers Using PHNN and RHNN Autopilots

Similar procedures were used to separately test the ability of the PHNN and RHNN autopilots to perform maneuvers by presenting them with reference pitch or roll angles. They are both reported in this section due to the similarities in procedures and results. The reference pitch and roll angles were varied between ±20 degrees at both low and high cruise conditions. The settling times for the PHNN are reported in Tables 6.5 and 6.6 and the pitch responses are given in Figures 6.11 and 6.12. Similarly, the RHNN settling times are reported in Tables 6.7 and 6.8 and the roll responses are plotted in Figures 6.13 and 6.14. The pitch responses at both low cruise and high cruise are similar for both SBPA and EBPA with the SBPA actually reaching the desired targets sooner than the EBPA. The SBPA PHNN exhibits more oscillatory behavior at high cruise than does its EBPA counterpart. Although the SBPA reaches the target tolerances quicker than the EBPA, the EBPA oscillates less. The rolling responses show that the SBPA and EBPA based RHNN's performances are virtually identical. The settling times for the RHNN's are similar with the SBPA being slightly lower. As with the altitude network, the SBPA appears to improve its performance as time passes whereas the EBPA performs consistently. Again, the pitch responses show that the NN based controllers have the ability to compensate for the lowly damped high cruise phugoid.

6.3 Linear Maneuvers Using DHNN Autopilots

Heading angle maneuvers were performed similarly to the roll and pitch maneuvers. These maneuvers were performed at low and high cruise by presenting the DHNN with a series of reference heading angles corresponding to changes up to 20 degrees. Figures 6.15 and 6.16 show the heading angle and corresponding roll angle responses at low cruise. The same responses for high cruise are shown in Figures 6.17 and 6.18. Additionally, Tables 6.9 and 6.10 report the settling times for these maneuvers. The EBPA demonstrates the ability to achieve the commanded heading angles in every case for both flight conditions while reducing the roll angle to about 0 degrees. The SBPA also shows

the ability to achieve the desired heading angles at low cruise, but takes significantly longer than the EBPA. The SBPA does have difficulty returning the aircraft to level wing position, but seems to perform better as more maneuvers are performed. It does, however, diverge in the last case.

6.4 Non-linear Adaptation Capabilities with Excited p and q Rates

Non-linear conditions were simulated by setting the aileron and elevator positions to maximum settings. The PHNN and RHNN autopilots were engaged once both pitch and roll rates exceeded target values ranging from 1 to 18 deg/s. Figures 6.19-6.36 show the aircraft responses and neural control inputs at low cruise. High cruse aircraft responses and control inputs are given in Figures 3.37-6.44. Tables 6.11 and 6.12 report the corresponding settling times. The tablels report the settling times for all cases while the plots give the results for only the even cases. The SBPA has shorter settling times for the lower cases, but oscillates more than the EBPA. The EBPA has considerably shorter settling times for cases above 3 deg/s and moves more directly to the desired aircraft states. The SBPA has difficulty with the 4, 6, and 8 deg/s cases at low cruise and with the 6 and 8 deg/s cases at high cruise. At the 4 deg/s, the SBPA diverges as the neural aileron input to the system locks at maximum deflection. The SBPA behavior at the 6 and 8 deg/s cases at low and high cruise are characterized by oscillating aileron inputs, which cause high frequency low amplitude oscillations in roll rate about 0 deg/s. The roll angle, however, does not oscillate and moves to the 0 degrees of roll state. The SBPA fails to achieve steady state conditions beyond 12 deg/s. The EBPA performs consistently through the range of excitation except for the 16 deg/s case. At 16 deg/s the EBPA displays oscillating aileron inputs causing oscillation in the roll angle and roll rate.

6.5 Non-linear Adaptation Capabilities with Excited p, q, and r Rates

The most difficult task for the NN autopilots is to adapt when subjected to all three angular rates. The non-linear conditions were induced by giving maximum open loop elevator, aileron, and

rudder inputs. Once all three angular rates were above target values ranging from 1 to 12 deg/s, the DHNN and PHNN autopilots were engaged. Figures 6.45-6.62 show the low cruise aircraft responses and NN inputs to the system as they attempt to compensate for the non-linear conditions. Figures 6.63-6.77 show the high cruise aircraft responses and NN control inputs. Tables 6.13 and 6.14 report the corresponding settling times. Overall, the EBPA has quicker settling times and more direct movement toward the target aircraft state at low cruise. The EBPA does have difficulty with the 4 and 8 deg/s cases, which take the EBPA longer to damp the rolling and yawing rates. At high cruise, the EBPA clearly demonstrates superior ability to compensate for the non-linear disturbances. The EBPA converges for every case whereas the SBPA only converges for 4 out of 10. The EBPA generally converges with lower settling times and less oscillation. The EBPA does have some difficulty with the 6 and 7 deg/s cases displaying the same behavior as with the 4 and 8 deg/s cases at low cruise.

6.6 Summary of Results

These results can be summarized by stating that the EBPA based autopilot systems generally performed better than the SBPA autopilot systems. The EBPA neural networks were able to perform maneuvers and non-linear compensation for cases beyond where the SBPA networks failed. The EBPA based neural network controllers displayed less oscillating behavior over the entire range of linear maneuvers and non-linear compensation tests, even at conditions where the phugoid and Dutch roll dynamic modes were lightly damped. This is true even in the limited number of cases where the SBPA based autopilots reached the target conditions in less time. The ability of each hidden and output layer neuron in the EBPA based networks to adjust the slope and output range of its individual sigmoid activation function leads to this improved performance over the SBPA.

Chapter 7: Conclusions and Recommendations

7.1 Conclusions

Five on-line learning neural network autopilot systems were applied to a six degree-of-freedom non-linear simulation model of a Boeing 747. Two training algorithms were used: the SBPA and EBPA. They were tested and compared based on their abilities to perform maneuvers at linear conditions and adaptation capabilities at non-linear conditions. In general, on-line learning neural network algorithms have the abilities to perform the linear maneuvers and compensate for non-linear disturbances while dealing with moderately undesirable dynamic modes. Proper design choices of the performance indices, neural structure, and learning rates allow the networks to perform these tasks by quickly learning the dynamics of the system and control it without previous training. The training algorithm choice depends on the purpose of the autopilot. The SBPA performs adequately for linear maneuvers involving pitch and roll angles, or compensation at slightly non-linear conditions involving p and q. The EBPA clearly is more effective for maneuvers involving altitude and speed or direction changes, and for the entire range of non-linear compensation. Overall, the EBPA trained neural network autopilot systems have the ability to perform a wider range of linear maneuvers while compensating for undesirable dynamic modes and to adapt to non-linear excitations to regain steady state conditions.

7.2 Recommendations

The main suggestion for extending this research is to apply the simulated controllers to the physical system. The simulation program presented in this document should be updated to simulate the dynamics of the 1/25 scale B747 once the flight testing phase produces the necessary information. The neural network autopilot systems should then be evaluated and updated as necessary to perform the autopilot tasks of the scale model. The simulation program should be further updated by the

introduction of sensor and system noise models. Once the simulation code and the neural network autopilot systems have been updated, the networks could then be implemented via hardware and interfaced with the simulation code. After ensuring the hardware based neural network autopilot systems have the ability to operate as desired, they should then be placed in the flight computer to be used in the control laws.

Another extension of this research would be to design stability augmentation controllers, using either classical or neural network methods, to compensate for the high cruise phugoid and lowly damped Dutch roll. These controllers could be applied to increase the stability of the system, which then could serve as the basis for neural autopilot design. The neural autopilots designed using the stable platform could then be compared to the ones designed using the pure open loop dynamics.

Lastly, studies need to be performed to determine the optimal selection of the performance indices to be minimized in neural network autopilot systems. It has been shown that performance indices resembling the error formulations of classical PI, PD, and PID controllers are effective. However, the theoretical link between the selection of the weighting factors and meeting classical time domain specifications of rise time, settling time, overshoot, closed loop natural frequency, and closed loop damping is still vague. It would be beneficial to know how to select the weighting terms based on defined classical time domain specifications to reduce the interactive nature of adjusting the weighting factors.

References

[1] Hagan, M.T., Demuth, H.B., Beale, M. Neural Network Design, PWS Publishing Company, Boston, MA, 1996.

[2] Fausett, L. Fundamentals of Neural Networks: Architectures, Algorithms, and Applications, Prentice Hall, Upper Saddle River, NJ, 1994.

[3] Kincheloe, M.W. "On-Line Learning Neural Network Controllers for Flight Control Systems", Thesis WVU MAE Dept., Morgantown, WV, April 1996.

[4] Casanova, J.L. "Design and Comparison of Neural Network and Fuzzy Logic Actuator Failure Schemes for Flight Control Systems", Thesis WVU MAE Dept., Morgantown, WV, December 1996.

[5] Advisory Group for Aerospace Research & Development. "Artificial Neural Network Approaches in Guidance and Control", Lecture Series #179.

[6] Rao, V.B., Rao, H.V. C++ Neural Networks & Fuzzy Logic, 2nd Edition, MIS:Press, New York, 1995.

[7] Lippmann, R.P. "An Introduction to Computing with Neural Nets", IEEE ASSP Magazine, April, 1987, pp. 4-22.

[8] Windon II, D.A. "Design and Comparison of Neural Network and Kalman Predictor Based Sensor Validation Schemes for Implementation on the NASA/Aurora Theseus", Thesis WVU MAE Dept., Morgantown, WV, June 1996.

[9] Napolitano, M.R., Neppach, C., Casdorph, V., Naylor, S. "On-Line Learning Non-Linear Direct Neuro Controllers for Restructurable Flight Control Systems", AIAA Journal of Guidance, Control, and Dynamics, Vol. 18, No. 1, Jan-Feb. 1995, pp. 170-176.

[10] Napolitano, M.R., Casdorph, V., Neppach, C., Naylor, S., Innocenti, M., Silvestri, G. "On-Line Learning Neural Architectures and Cross-Correlation Analysis for Actuator Failure Detection and Identification", International Journal of Control, 1996, Vol. 63, No. 3, pp. 433-455.

[11] Napolitano, M.R., Chen, C.I., Naylor, S. "Aircraft Failure Detection and Identification Using Neural Networks", AIAA Journal of Guidance, Control, and Dynamics, Vol. 16, No. 6, Nov-Dec. 1993, pp. 999-1009.

[12] Napolitano, M.R., Neppach, C., Casdorph, V., Naylor, S., Innocenti, M., Silvestri, G. "Neural Network Based Scheme for Sensor Failure Detection, Identification, and Accommodation", AIAA Journal of Guidance, Control, and Dynamics, Vol. 18, No. 6, Nov-Dec. 1995, pp. 1280-1286.

[13] Napolitano, M.R., Kincheloe, M. "On-Line Learning Neural Network Controllers for Autopilot Systems", Proceedings of the 95 AIAA Guidance, Navigation, and Control Conference, AIAA Paper 95-3269, Baltimore, MD, August 1995.

[14] Napolitano, M.R., Kincheloe, M. "On-Line Learning Neural Network Controllers for Autopilot Systems", AIAA Journal of Guidance, Control, and Dynamics, Vol. 33, No. 6, Nov-Dec 1995, pp. 1008-1015.

[15] Bragg, D.E. "Application and Comparison of Neural Network and Fuzzy Logic Techniques for Design of Auto-Pilot Systems", Thesis WVU MAE Dept., Morgantown, WV, 1996.

[16] Napolitano, M.R., Martinelli, R.D., Windon, D.A., Casanova, J.L. "Using Neural Networks for Reconstruction of Control Deflections in Aircraft Crash Data", Proceedings of the "Transportation 96" conference, Queretaro, Mexico, November 14-16, 1996.

[17] Napolitano, M.R., Windon, D.A. "Applications of Neural Networks for Signal Reconstruction in Aircraft Crash Investigations", AIAA Paper 96-3802, Proceedings of the AIAA Guidance, Navigation, and Control 96 conference, San Diego, CA, July 1996.

[18] Napolitano, M.R., Silvestri, G., Windon, D.A., Casanova, J.L., Innocenti, M. "Sensor Validation Using Hardware-Based On-Line Learning Neural Networks", submitted for evaluation to the IEEE Transactions on Aerospace and Electronic Systems, November 1995 - accepted for publication in October 1996.

[19] Roskam, J. Airplane Flight Dynamics and Automatic Flight Controls, Part I, Design, Analysis and Research Corp., Lawrence, KS, 1995.

[20] McLean, D. Automatic Flight Control Systems, Prentice Hall, New York, 1990.

[21] Paris, A.C. "Estimation of the Longitudinal and Lateral-Directional Aerodynamic Parameters from Flight Data for the NASA F/A-18 HARV", Ph.D. Dissertation, WVU MAE Dept., Morgantown, WV, May, 1997.

[22] Anderson, J.D. Jr. Introduction to Flight, 3rd Edition, McGraw-Hill, New York, 1989.

[23] Roskam, J. Airplane Flight Dynamics and Automatic Flight Controls, Part II, Design, Analysis and Research Corp., Lawrence, KS, 1995.

[24] Franklin, G.F., Powell, J.D., Emami-Naeini, A. Feedback Control of Dynamic Systems, 3rd Edition, Addison-Wesley, New York, 1994.

[25] Kreyszig, Erwin. Advanced Engineering Mathematics, 7th Edition, John Wiley and Sons, New York, 1993.

Tables

Table 4.1: Geometric, mass, and aerodynamic data for the B747 from Ref.[19].

Reference Geometry		
S (ft^2)	5500	
$\bar{c}$ (ft)	27.3	
b (ft)	196	
Flight Condition	Low Cruise	High Cruise
h (ft)	20,000	40,000
Mach Number	0.65	0.9
U_1 (ft/s)	673	871
C.G. location (fraction of $\bar{c}$)	0.25	0.25
Angle of attack α_1 (deg)	2.5	2.4
Mass Data		
W (lbs)	636,636	636,636
I_{xx} (slugs ft^2)	18,200,000	18,200,000
I_{yy} (slugs ft^2)	33,100,000	33,100,000
I_{zz} (slugs ft^2)	49,700,000	49,700,000
I_{xz} (slugs ft^2)	970,000	970,000
Steady State Coefficients		
C_{L_1}	0.4	0.52
C_{D_1}	0.0250	0.0450
$C_{T_{x_1}}$	0.0250	0.0450
C_{m_1}	0	0
$C_{m_{\eta_1}}$	0	0
Longitudinal Coefficients		
C_{D_0}	0.0164	0.0305
C_{D_u}	0	0.22
C_{D_α}	0.20	0.50
$C_{T_{x_u}}$	-0.055	-0.950
C_{L_0}	0.21	0.29
C_{L_u}	0.13	-0.23
C_{L_α}	4.4	5.5
$C_{L_{\dot{\alpha}}}$	7.0	8.0
C_{L_q}	6.6	7.8
C_{m_0}	0	0
C_{m_u}	0.013	-0.09

C_{m_α}	-1.0	-1.6
$C_{m_{\dot{\alpha}}}$	-4.0	-9.0
C_{m_q}	-20.5	-25.5
$C_{m_{T_u}}$	0	0
$C_{m_{T_\alpha}}$	0	0
<u>Longitudinal Control Derivatives</u>		
$C_{D_{\delta_E}}\ (rad^{-1})$	0	0
$C_{L_{\delta_E}}\ (rad^{-1})$	0.32	0.30
$C_{m_{\delta_E}}\ (rad^{-1})$	-1.30	-1.20
$C_{D_{i_h}}\ (rad^{-1})$	0	0
$C_{L_{i_h}}\ (rad^{-1})$	0.70	0.65
$C_{m_{i_h}}\ (rad^{-1})$	-2.7	-2.5
<u>Lateral-Directional Coefficients</u>		
C_{l_β}	-0.160	-0.095
C_{l_p}	-0.340	-0.320
C_{l_r}	0.130	0.200
C_{y_β}	-0.90	-0.90
C_{y_p}	0	0
C_{y_r}	0	0
C_{n_β}	0.160	0.210
$C_{n_{T_\beta}}$	0	0
C_{n_p}	-0.026	0.020
C_{n_r}	-0.280	-0.330
<u>Lateral-Directional Control Derivatives</u>		
$C_{l_{\delta_A}}\ (rad^{-1})$	0.013	0.014
$C_{l_{\delta_R}}\ (rad^{-1})$	0.008	0.005
$C_{y_{\delta_A}}\ (rad^{-1})$	0	0
$C_{y_{\delta_R}}\ (rad^{-1})$	0.120	0.060
$C_{n_{\delta_A}}\ (rad^{-1})$	0.0018	-0.0028
$C_{n_{\delta_R}}\ (rad^{-1})$	-0.100	-0.095

Table 5.1: Final Neural Network Autopilot Structures.

AHNN

$J = 0.05(h - h_{ref}) + 0.75(\dot{h} - \dot{h}_{ref}) + 10(q - q_{ref})$ $\qquad \dot{h}_{ref} = q_{ref} = 0$

Inputs	Output	η	Window (SBPA\EBPA)	Structure (SBPA\EBPA)
Θ , q , h , $\dot{h}$, δ_E	δ_E	0.001\0.3	6\4	30-50-1\20-30-1

PHNN

$J = 0.25(\Theta - \Theta_{ref}) + 0.5(q - q_{ref})$ $\qquad q_{ref} = 0$

Inputs	Output	η	Window (SBPA\EBPA)	Structure (SBPA\EBPA)
Θ , q , δ_E	δ_E	0.01\0.5	4\4	12-15-1\12-15-1

SHNN

$J = 0.5(V_{ref} - V) + 0.8(\dot{V} - \dot{V}_{ref})$ $\qquad \dot{V}_{ref} = 0$

Inputs	Output	η	Window (SBPA\EBPA)	Structure (SBPA\EBPA)
V , $\dot{V}$, δ_T	δ_T	0.05\0.25	4\4	12-15-1\12-15-1

RHNN

$J = 0.25(\Phi_{ref} - \Phi) + 2.0(p_{ref} - p)$ $\qquad p_{ref} = 0$

Inputs	Output	η	Window (SBPA\EBPA)	Structure (SBPA\EBPA)
Φ, p, δ_A, δ_R	δ_A	0.01\0.5	4\4	16-19-1\16-19-1

DHNN

$J_1 = 0.1(\Psi - \Psi_{ref}) + 0.7(r - r_{ref}) + 0.25(\dot{r} - \dot{r}_{ref})$

$J_2 = 0.1(\Phi_{ref} - \Phi) + 2.5(p_{ref} - p) + 0.25(\dot{p}_{ref} - \dot{p})$ $\qquad$ (Linear) $r_{ref} = \dot{r}_{ref} = p_{ref} = \dot{p}_{ref} = 0$

$J_1 = 0.7(r - r_{ref}) + 0.25(\dot{r} - \dot{r}_{ref})$

$J_2 = 0.3(\Phi_{ref} - \Phi) + 2.5(p_{ref} - p)$ $\qquad$ (Nonlinear) $r_{ref} = \dot{r}_{ref} = p_{ref} = 0$

Inputs	Output	η	Window (SBPA\EBPA)	Structure (SBPA\EBPA)
Φ , p , δ_A Ψ , r , δ_R	δ_A δ_R	0.001\0.5	5\6	36-75-2\30-36-2

Table 6.1: AHNN and SHNN reference input values for maneuvering at LC.

Flight Cond	Altitude (ft)	Speed (ft/s)	Δ Altitude (ft)	Δ Speed (ft/s)
1	20000	680	0	0
2	20050	690	50	10
3	20150	685	100	-5
4	19950	650	-200	-35
5	20250	700	300	50
6	19900	725	-350	25
7	19500	700	-400	-25
8	20000	680	500	-20
9	20500	750	500	70
10	21000	780	500	30
11	19000	650	-2000	-130
12	20000	680	1000	30

Table 6.2: AHNN and SHNN reference input values for maneuvering at HC.

Flight Cond	Altitude (ft)	Speed (ft/s)	Δ Altitude (ft)	Δ Speed (ft/s)
1	40000	775	0	0
2	40050	789	50	14
3	40150	774	100	-15
4	39950	739	-200	-35
5	40250	789	300	50
6	39900	809	-350	20
7	39500	784	-400	-25
8	40000	775	500	-9
9	41000	875	1000	100
10	40500	844	-500	-31
11	39000	744	-1500	-100
12	40000	775	1000	31

Table 6.3: Settling times for maneuvers at LC using AHNN's and SHNN's.

Flight Cond	Ts (Altitude) (sec) SBPA	Ts (Altitude) (sec) EBPA	Ts (Speed) (sec) SBPA	Ts (Speed) (sec) EBPA
1	29.1	8.4	15.1	7.6
2	31.2	32.35	10.2	10.1
3	42.85	43.25	10.75	7.55
4	58.85	51.95	69.85	22.65
5	57.4	57.5	35.05	23.1
6	61.4	61.05	25.5	15
7	65.85	62.2	60.5	22.4
8	65.75	66.15	13.75	18.35
9	66.35	65.95	42.35	22.75
10	66.3	65.85	28.75	21
11	123.45	85.1	Throttle Locked	27.65
12	105.9	75.3	Throttle Locked	27.35

Note: Ts (altitude) is the time from AHNN engagement until oscillation within 5 feet.
Note: Ts (speed) is the time from SHNN engagement until oscillation within 1 ft/s.

Table 6.4: Settling times for maneuvers at HC using AHNN's and SHNN's.

Flight Cond	Ts (Altitude) (sec) SBPA	Ts (Altitude) (sec) EBPA	Ts (Speed) (sec) SBPA	Ts (Speed) (sec) EBPA
1	53.35	20.6	15.75	14.95
2	42.45	29.9	8.45	15.1
3	45.8	43.55	9	12.3
4	170.15	53.75		23.3
5	78.4	60.55		28.7
6	78.19	61.65	Throttle Locked	12.95
7	77.3	63.2	and set to 34%	27.9
8	89.45	65.85	to keep performing	19.4
9	104.3	77.85	altitude maneuvers	31
10	75.9	67.5		23.8
11	81.85	80.9		38.6
12	80.1	75.85		34.95

Note: Ts (altitude) is the time from AHNN engagement until oscillation within 5 feet.
Note: Ts (speed) is the time from SHNN engagement until oscillation within 1 ft/s.

Table 6.5: Settling times for maneuvers at LC using PHNN's.

Maneuver	Θref (deg)	ΔΘ (deg)	Ts (sec) SBPA	Ts (sec) EBPA
0	2.65	-	-	-
1	5	2.35	6.55	6.75
2	10	5	9.6	11.05
3	-5	-15	14.3	16.35
4	15	20	42.35	17.2
5	20	5	11.6	10.95
6	0	-20	27.9	18.25
7	-5	-5	8.7	11.4
8	5	10	11.85	15.4
9	-10	-15	14.15	16.15
10	-20	-10	10.3	13.15
11	-10	10	4.3	11.85
12	2.65	12.65	4.45	11.3
13	10	7.35	4.25	10.25
14	20	10	10.75	11.75
15	2.65	-17.35	21.3	14.8

Note: Ts is the time from PHNN engagement until oscillation within 0.5 deg.

Table 6.6: Settling times for maneuvers at HC using PHNN's.

Maneuver	Θref (deg)	ΔΘ (deg)	Ts (sec) SBPA	Ts (sec) EBPA
0	2.6	-	-	-
1	5	2.4	14.2	18.65
2	10	5	13.1	17.15
3	0	-10	16.9	21.45
4	15	15	31.25	36.65
5	20	5	-	45.6
6	0	-20	45.35	52.9
7	-5	-5	13.65	17.4
8	5	10	15.95	20.15
9	-10	-15	27.85	32.45
10	-20	-10	14.35	16.45
11	0	20	11.35	21.9
12	3	3	4.2	10.45

Note: Ts is the time from PHNN engagement until oscillation within 0.5 deg.

Table 6.7: Settling times for maneuvers at LC using RHNN's.

Maneuver	Φref (deg)	ΔΦ (deg)	Ts (sec) SBPA	Ts (sec) EBPA
0	0	-	-	-
1	5	5	15.8	19.75
2	15	10	18.3	20.1
3	30	15	-	-
4	20	-10	31.25	36.4
5	10	-10	23.45	26.8
6	0	-10	20.3	22.9
7	-5	-5	15	16.8
8	-15	-10	17.4	19.15
9	-20	-5	12.65	14.35
10	-10	10	23.7	26.3
11	0	10	20.5	22.9

Note: Ts is the time from RHNN engagement until oscillation within 0.5 deg.

Table 6.8: Settling times for maneuvers at HC using RHNN's.

Maneuver	Φref (deg)	ΔΦ (deg)	Ts (sec) SBPA	Ts (sec) EBPA
0	0	-	-	-
1	5	5	17.6	12.9
2	15	10	19.35	19
3	30	15	19.55	19.4
4	20	-10	29.55	28.5
5	10	-10	22.15	22.8
6	0	-10	20.85	20.35
7	-5	-5	17.1	17.75
8	-15	-10	19.45	19.25
9	-20	-5	11.2	15.25
10	-10	10	22.2	22.7
11	0	10	20.85	20.3

Note: Ts is the time from RHNN engagement until oscillation within 0.5 deg.

Table 6.9: Settling times for maneuvers at LC using DHNN's.

Maneuver	Ψref (deg)	ΔΨ (deg)	Ts (sec) SBPA	Ts (sec) EBPA
0	0	-	-	-
1	5	5	89	32
2	15	10	93.2	21.35
3	20	5	88.75	40.8
4	10	-10	94.95	49.7
5	0	-10	76.3	51.05
6	-5	-5	76.75	22.6
7	-15	-10	97.4	27.05
8	-20	-5	88.85	22.45
9	-10	10	-	22.8
10	0	10	93	20.4
11	18	18	-	35.65
12	0	-18	-	53.85

Note: Ts is the time from DHNN engagement until oscillation within 0.5 deg.

Table 6.10: Settling times for maneuvers at HC using DHNN's.

Maneuver	Ψref (deg)	ΔΨ (deg)	Ts (sec) SBPA	Ts (sec) EBPA
0	0	-	-	-
1	5	5	80.35	25.05
2	15	10	-	66.15
3	20	5	-	77
4	10	-10	86.9	63.3
5	0	-10	-	64.55
6	-5	-5	96.7	47.25
7	-15	-10	11.5	74.7
8	-20	-5	88.65	25.8
9	-10	10	-	92.95
10	0	10	-	92.35

Note: Ts is the time from DHNN engagement until oscillation within 0.5 deg.

Table 6.11: Settling times for p and q non-linear excitations at LC.

p, q (deg/s)	Ts (sec) SBPA	Ts (sec) EBPA	
1	12.1	12.2	Θref = 3 deg
2	12	12.8	Φref = 0 deg
3	16.55	13.1	
4	-	13.8	
5	19.65	15.45	
6	25.5	22.75	
7	38.15	23.8	
8	49.75	24.5	
9	39.45	25.3	
10	29.4	20.65	
11	44.95	20.8	
12	64.3	24.35	
14	-	25.3	
16	-	-	
18	-	39.15	

Note: Ts is the time from PHNN and RHNN engagement until Θ and Φ oscillation within 0.5 deg of reference values.

Table 6.12: Settling times for p and q non-linear excitations at HC.

p, q (deg/s)	Ts (sec) SBPA	Ts (sec) EBPA	
1	12.7	16.35	Θref = 3 deg
2	16.2	17.5	Φref = 0 deg
3	18.45	19.5	
4	26.75	20.1	
5	31.4	29.15	
6	35.45	30.45	
7	35.25	30.85	
8	35.75	30.85	

Note: Ts is the time from PHNN and RHNN engagement until Θ and Φ oscillation within 0.5 deg of reference values.

Table 6.13: Settling times for p, q and r non-linear excitations at LC.

p, q, r (deg/s)	Ts (sec) SBPA	Ts (sec) EBPA	
1	161.9	23.95	Θref = 3 deg
2	65.8	33.6	Φref = 0 deg
3	47	30.05	rref = 0 deg/s
4	46.6	82.35	
5	46.5	40.65	
6	46.25	43.05	
7	46.8	46.45	
8	47.55	52.75	
9	47.9	28.6	
10	48.5	33.35	
11	48.5	33.4	
12	49.55	37.25	

Note: Ts is the time from PHNN and DHNN engagement until Θ and Φ oscillation within 0.5 deg and r within 0.1 deg/s of reference values.

Table 6.14: Settling times for p, q and r non-linear excitations at HC.

p, q, r (deg/s)	Ts (sec) SBPA	Ts (sec) EBPA	
1	-	20.6	Θref = 3 deg
2	-	38.65	Φref = 0 deg
3	105.5	55.8	rref = 0 deg/s
4	87.2	28	
5	66.25	62	
6	67.25	108.2	
7	-	111.2	
8	-	32.1	
9	-	-	
10	-	39.25	

Note: Ts is the time from PHNN and DHNN engagement until Θ and Φ oscillation within 0.5 deg and r within 0.1 deg/s of reference values.

Figures

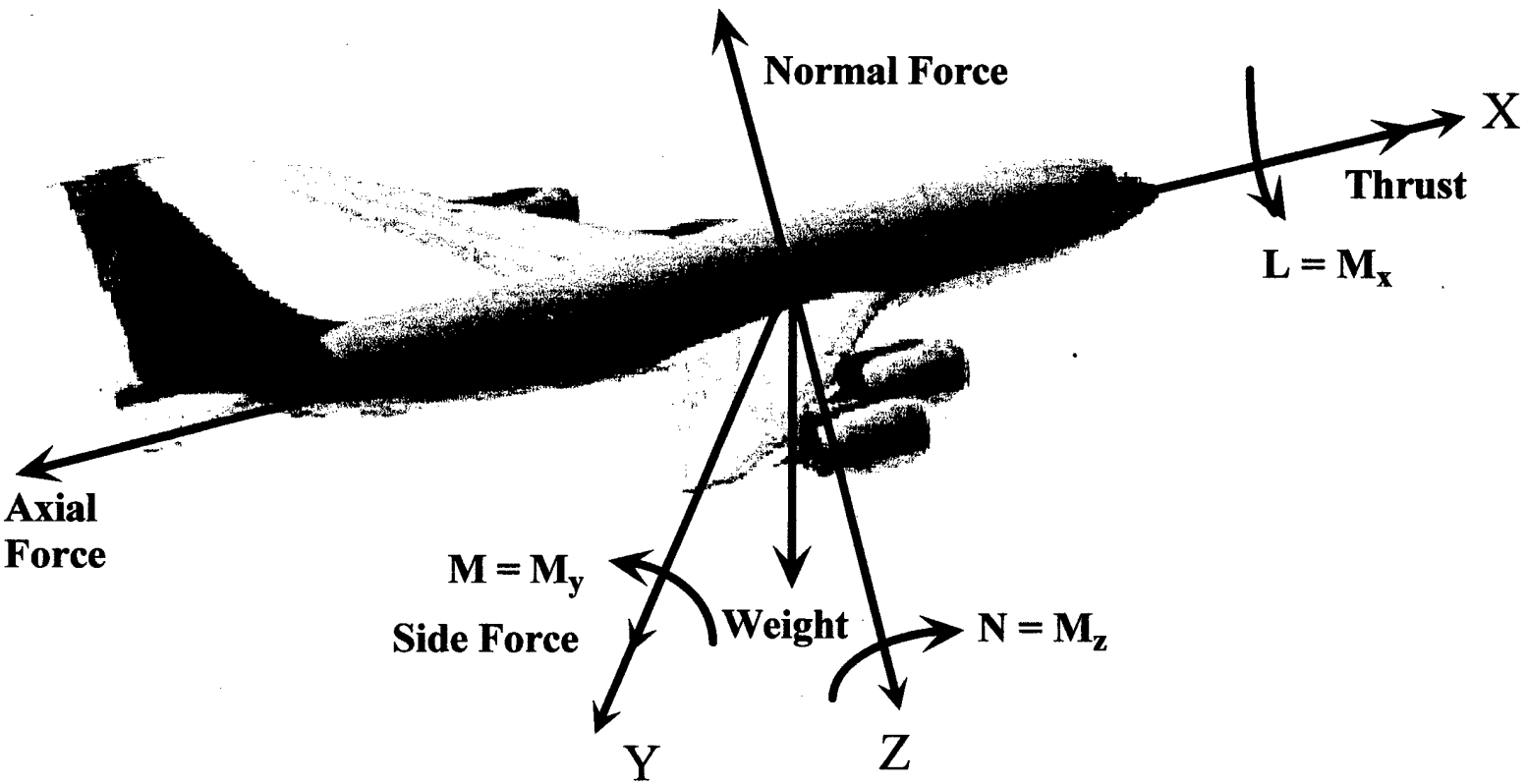


Figure 2.1: Aircraft forces and moments about the body axis.

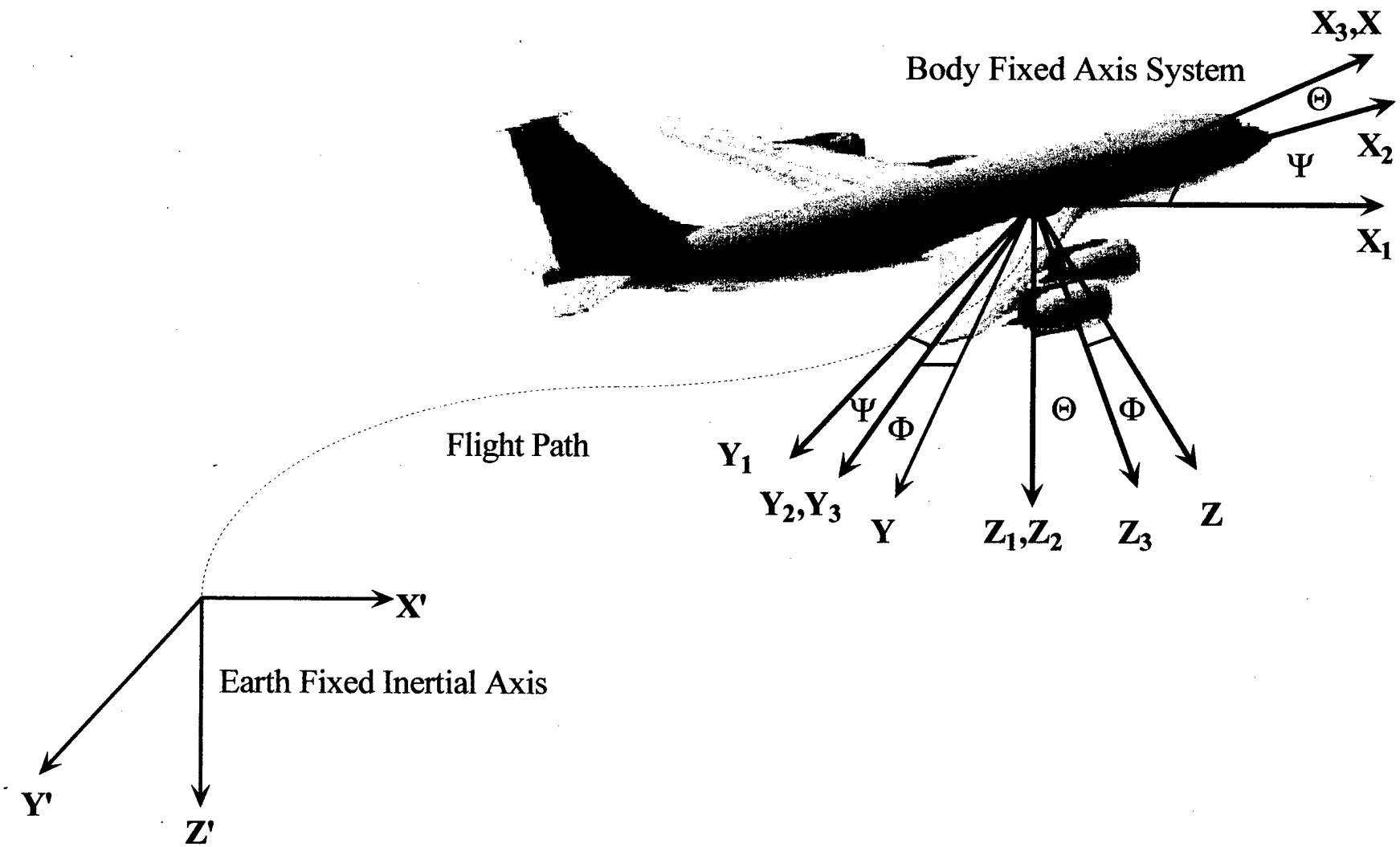


Figure 2.2: Aircraft orientation with respect to the Earth Fixed Inertial Axis coordinate system by the introduction of Euler Angles.

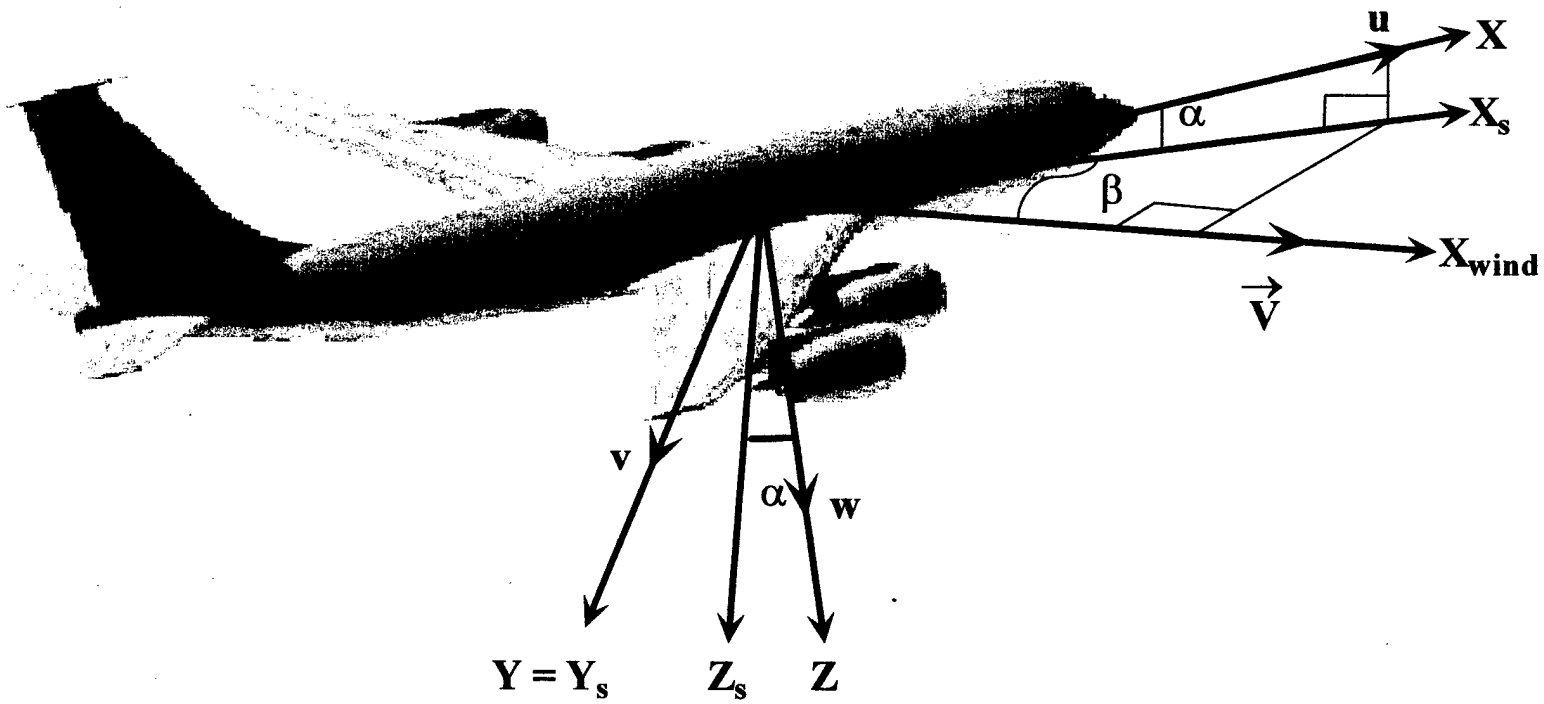


Figure 2.3: Relationship between Body, Stability, and Wind axis systems using polar coordinate system.

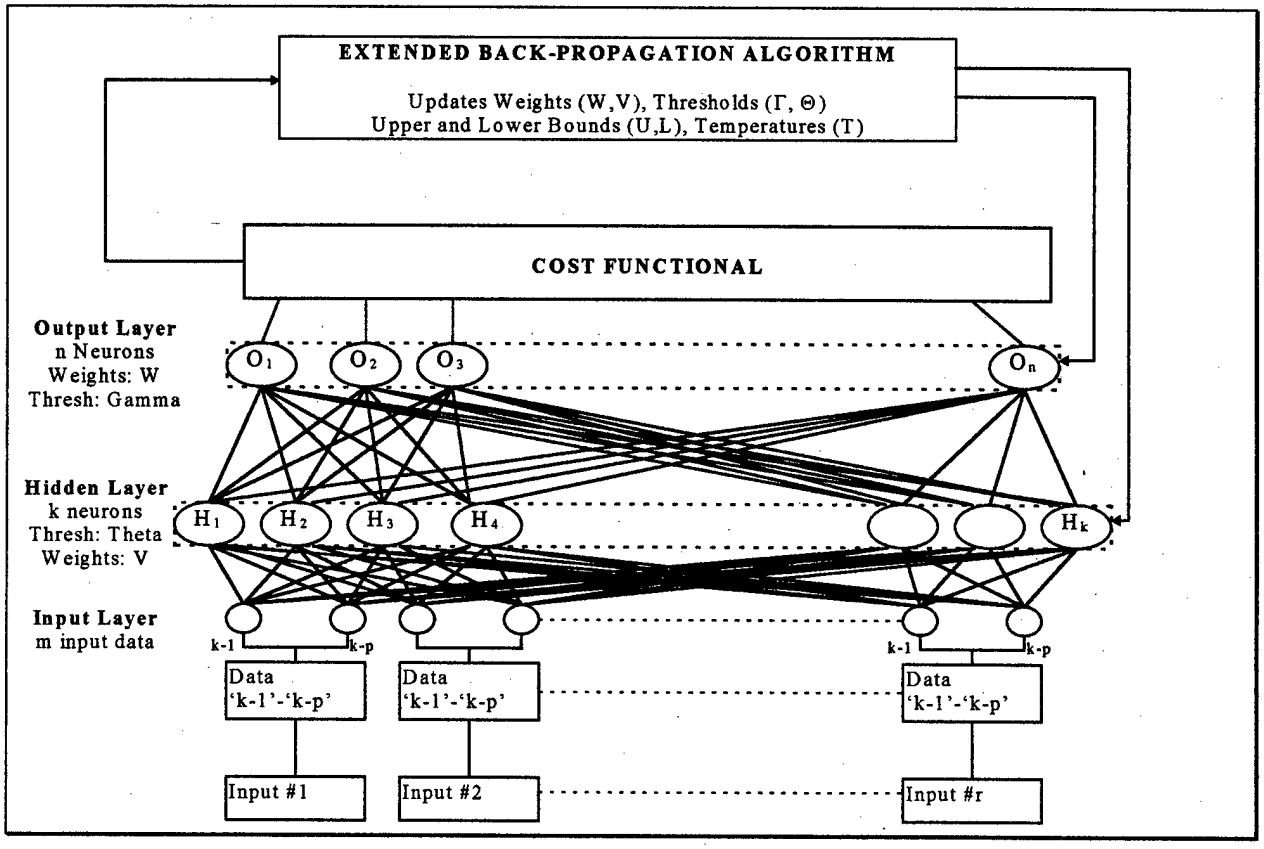


Figure 3.1: General representation of a 3 layer neural network trained with the Extended Back-Propagation Algorithm[14].

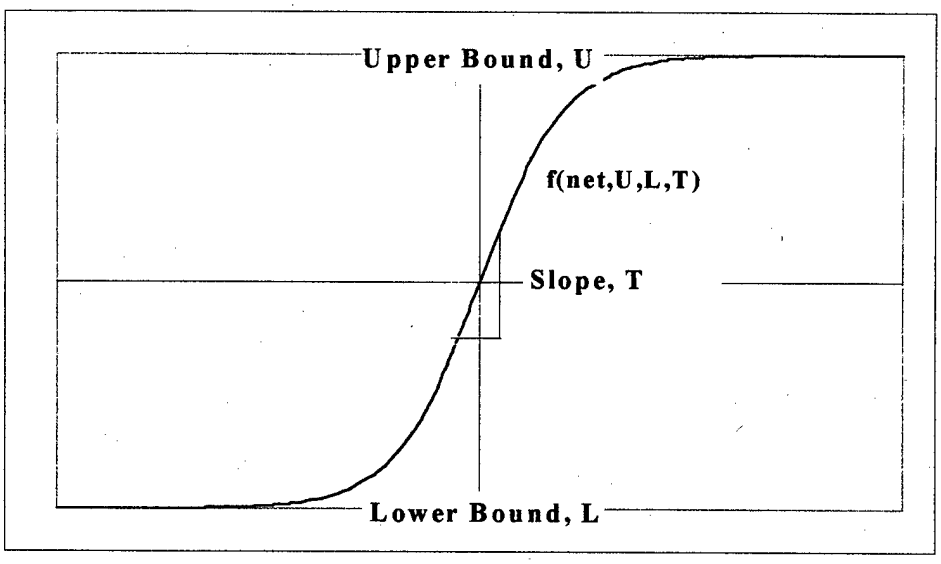


Figure 3.2: Extended Back-Propagation sigmoid activation function for general U, L, and T arguments.

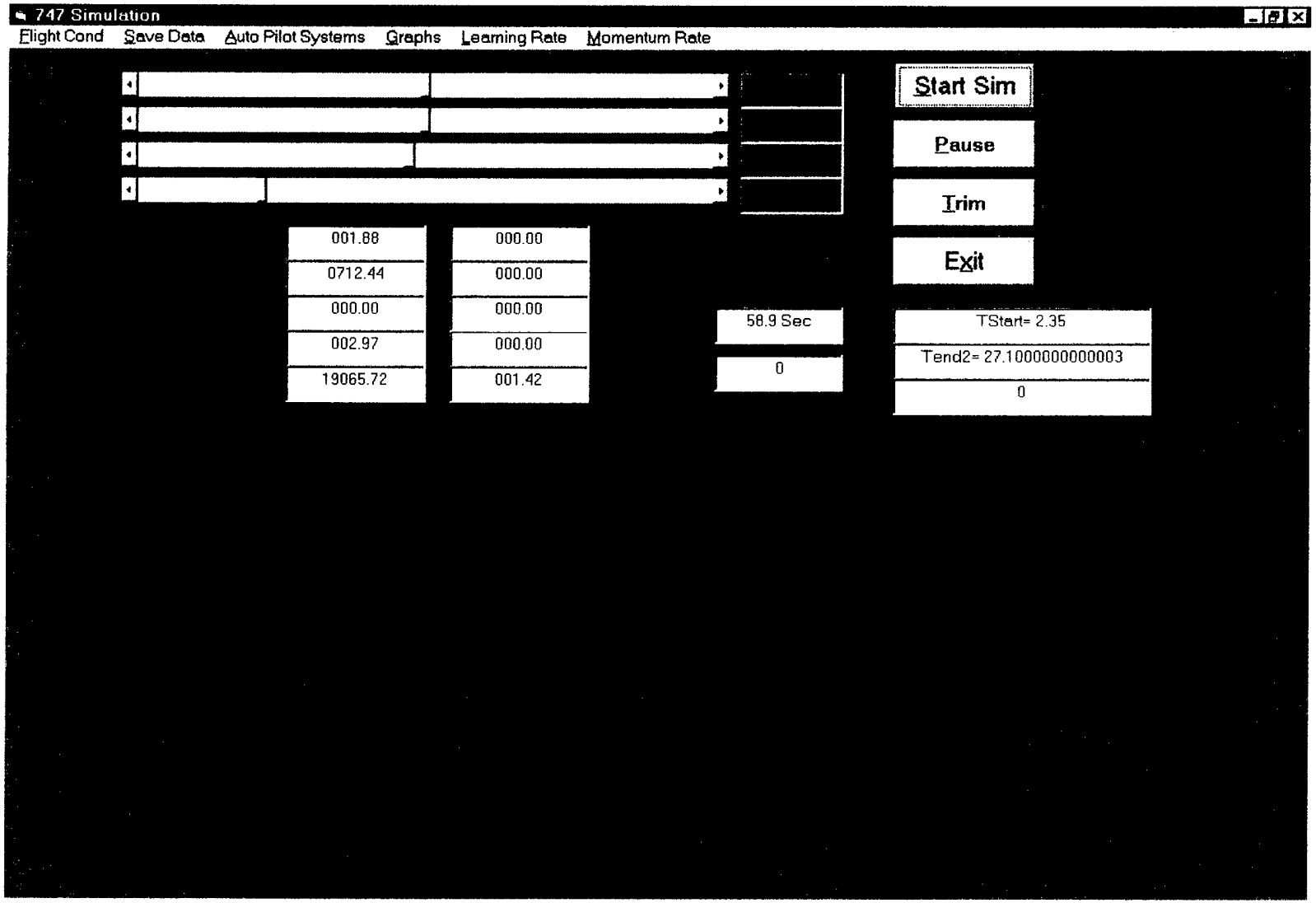


Figure 4.1: User interface with numerical simulation program.

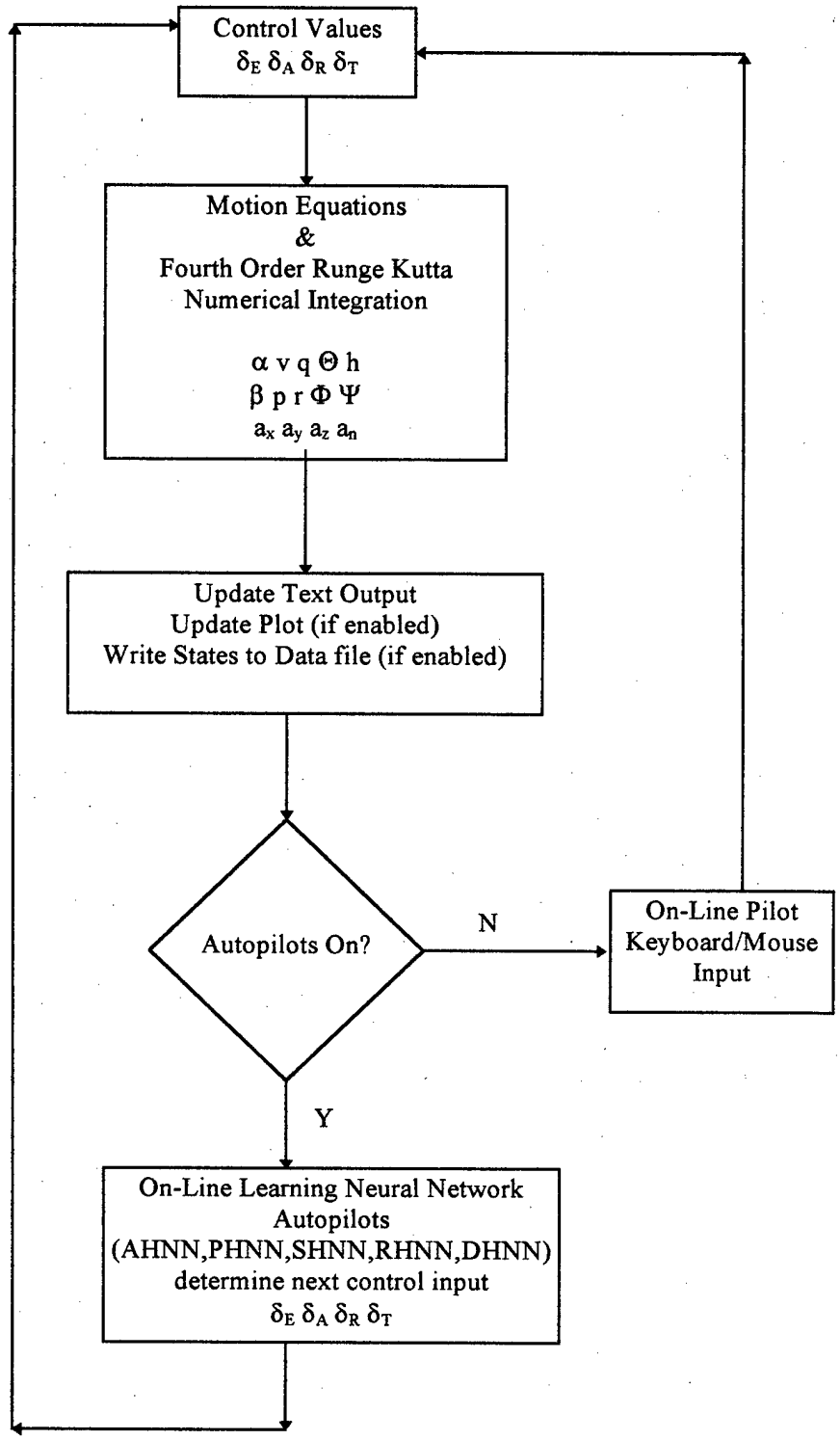


Figure 4.2: General block diagram of simulation program with neural network autopilot controllers.

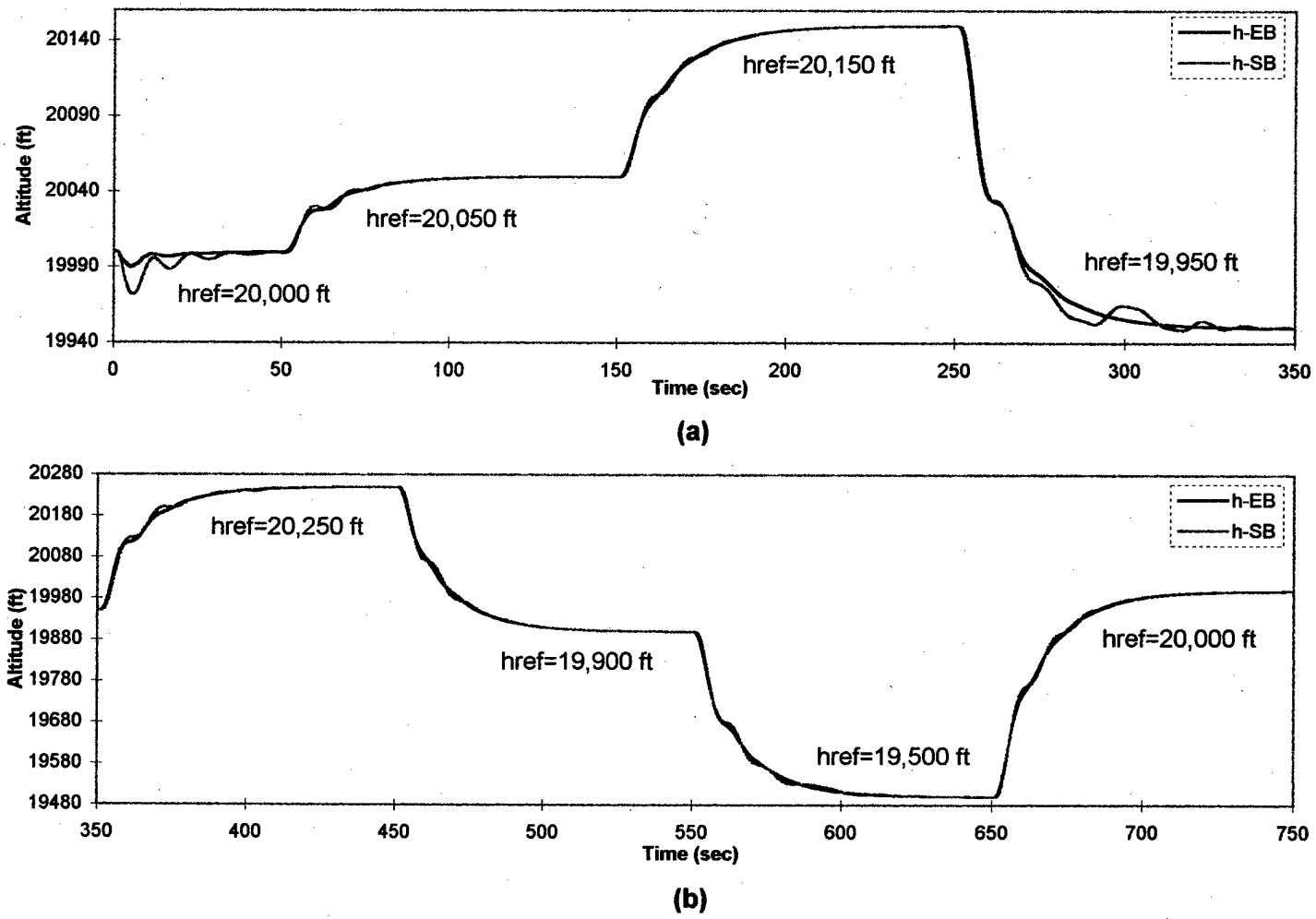


Figure 6.1: Altitude response to commanded altitude and speed maneuvers at LC (refer to Table 6.1).

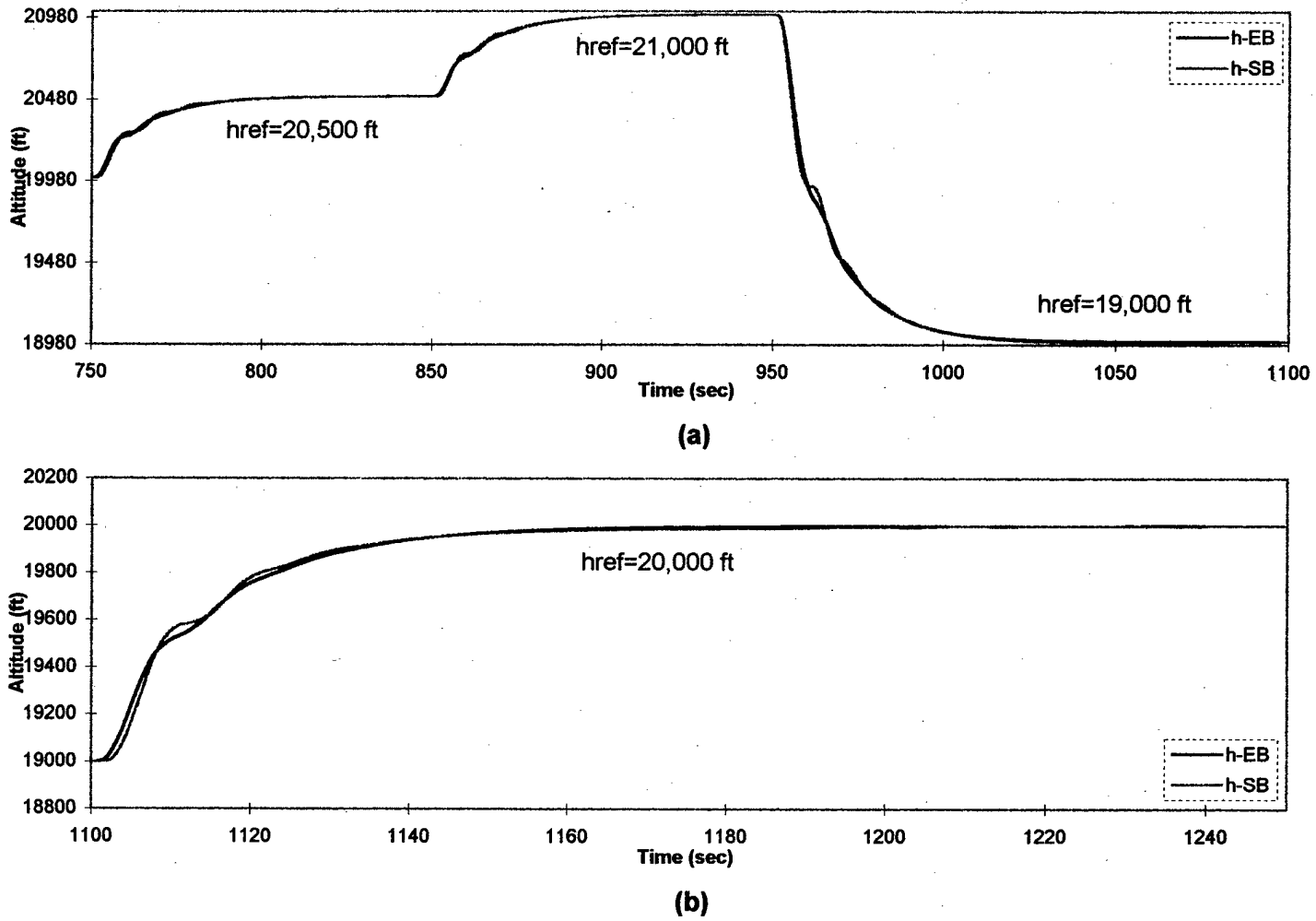


Figure 6.2: Altitude response to commanded altitude and speed maneuvers at LC (refer to Table 6.1).

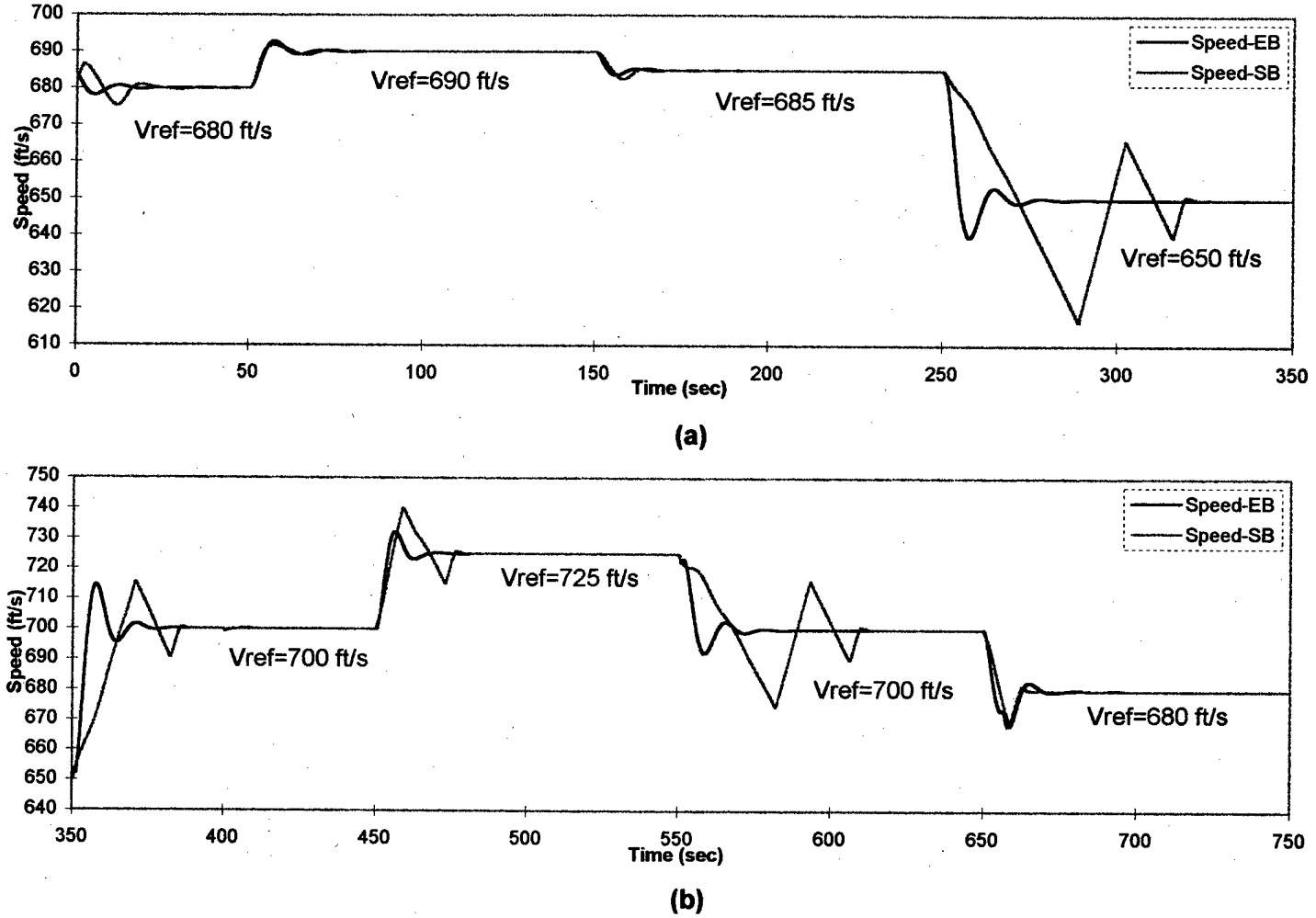


Figure 6.3: Speed response to commanded altitude and speed maneuvers at LC (refer to Table 6.1).

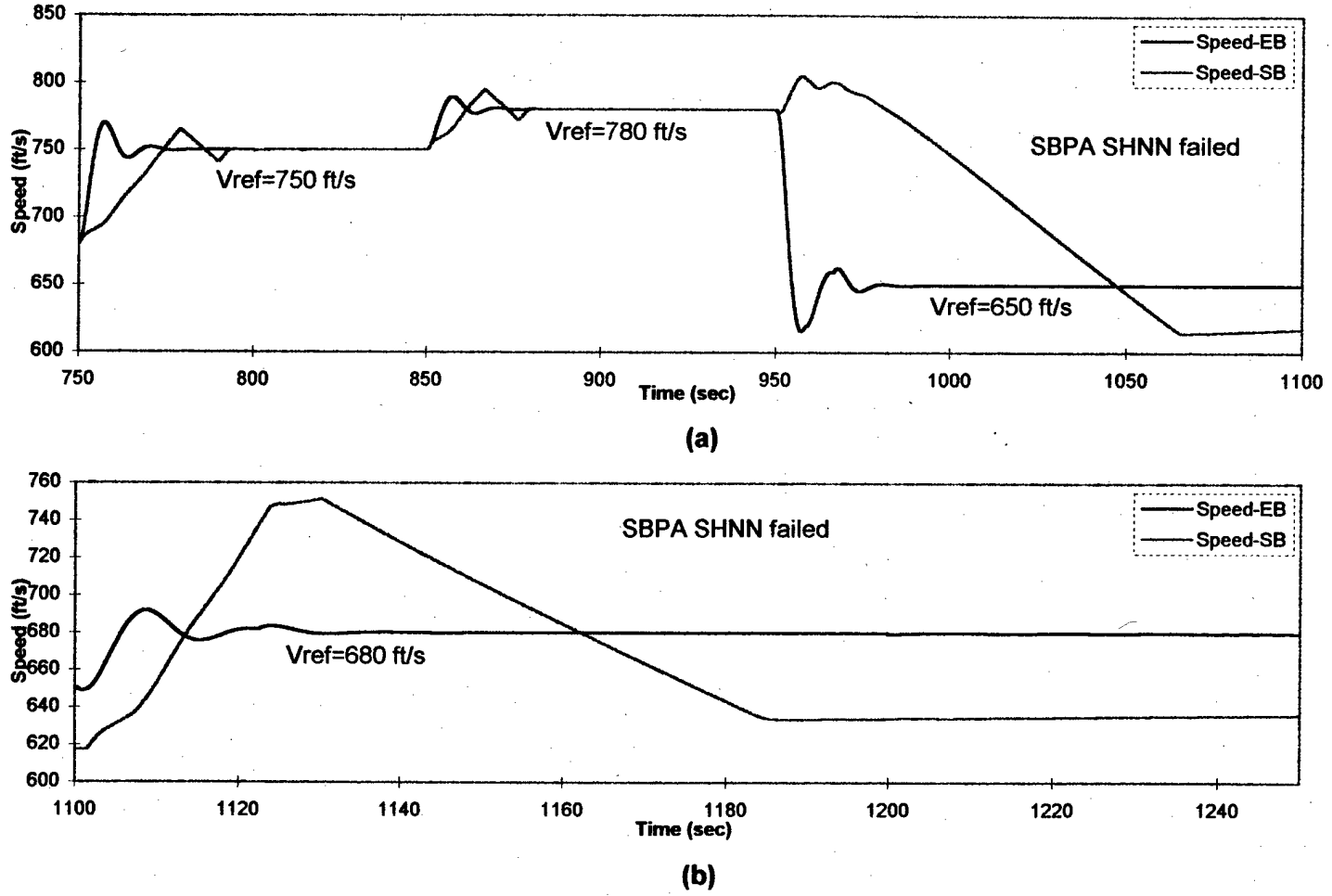


Figure 6.4: Speed response to commanded altitude and speed maneuvers at LC (refer to Table 6.1).

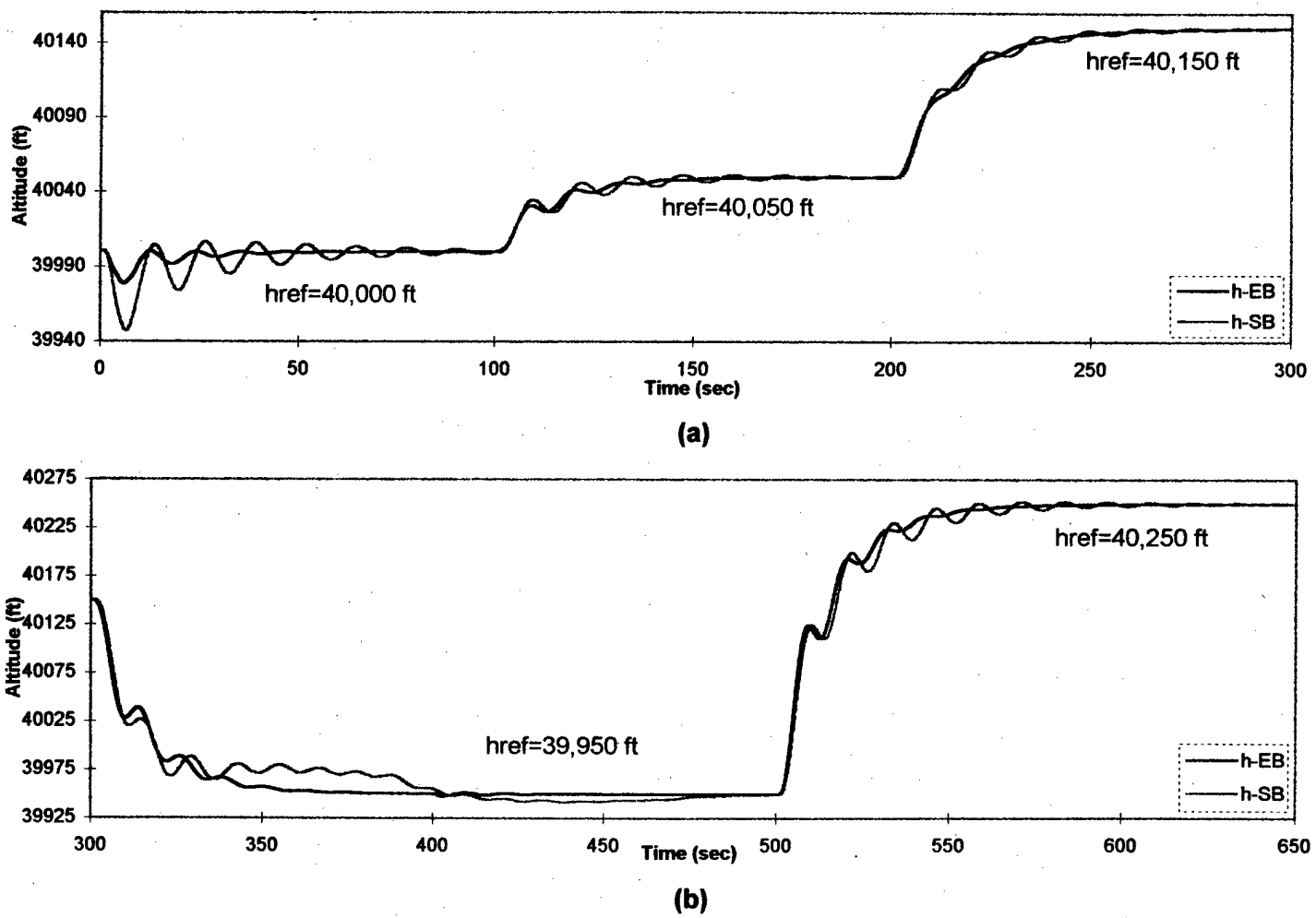


Figure 6.5: Altitude response to commanded altitude and speed maneuvers at HC (refer to Table 6.2).

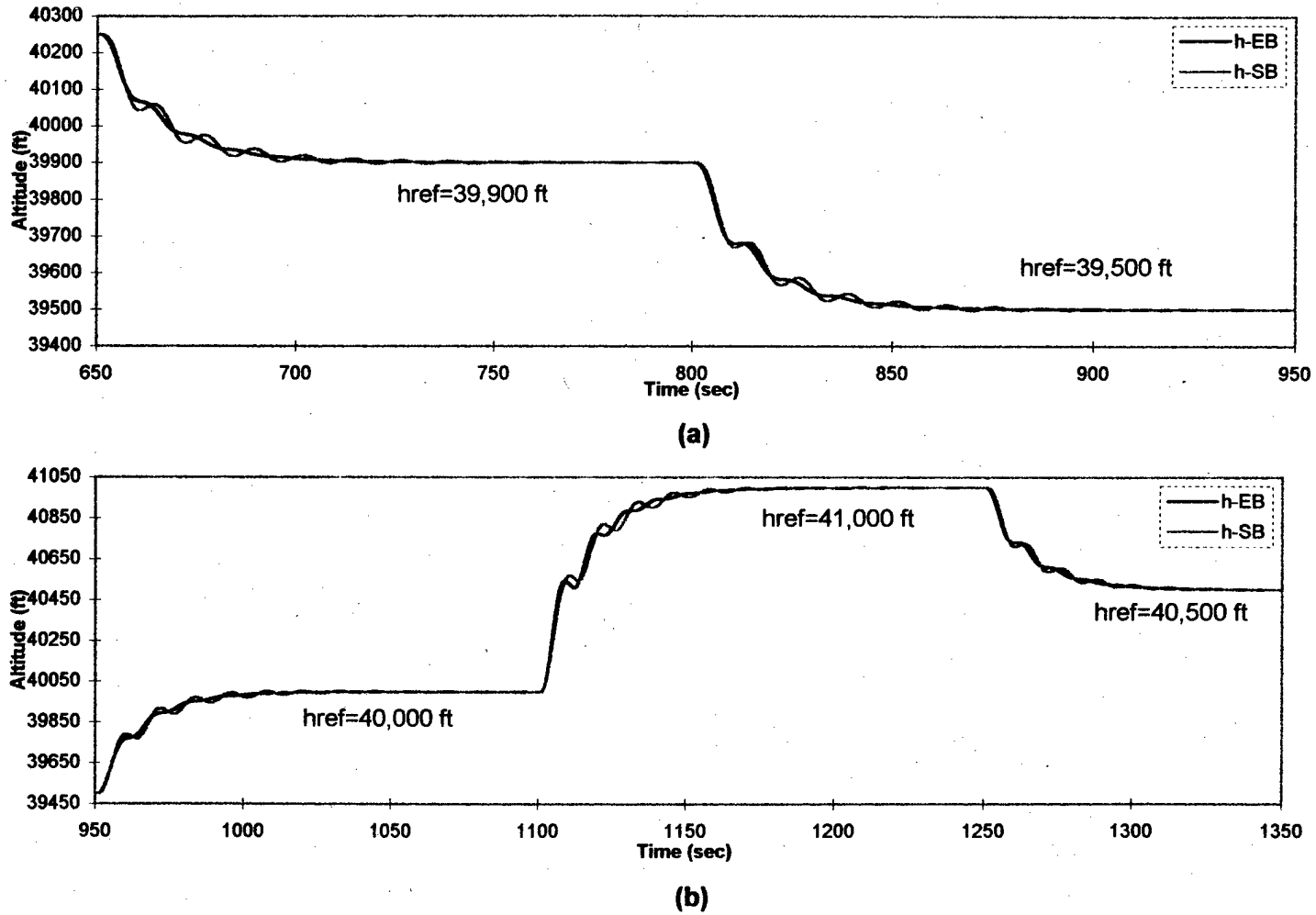


Figure 6.6: Altitude response to commanded altitude and speed maneuvers at HC (refer to Table 6.2).

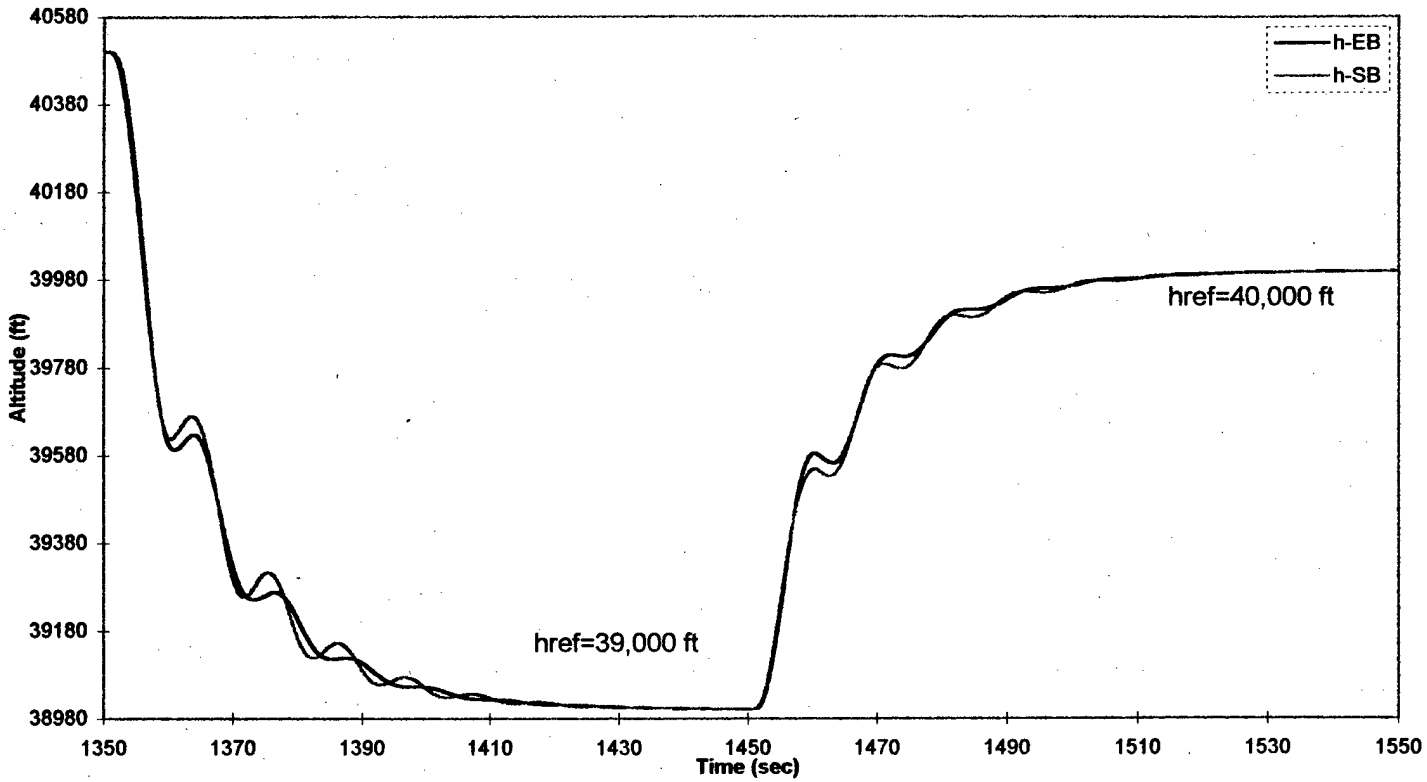


Figure 6.7: Altitude response to commanded altitude and speed maneuvers at HC (refer to Table 6.2).

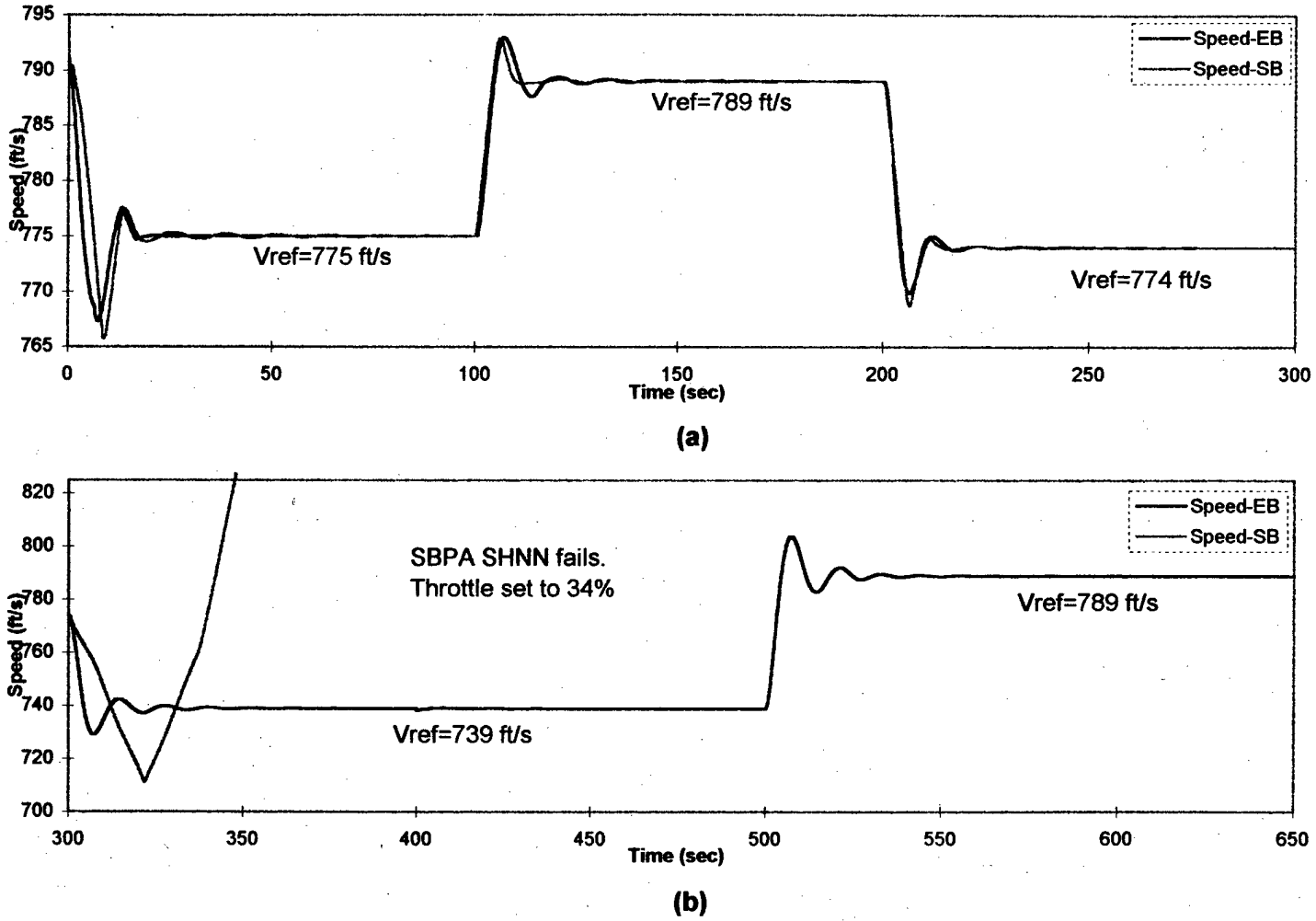


Figure 6.8: Speed response to commanded altitude and speed maneuvers at HC (refer to Table 6.2).

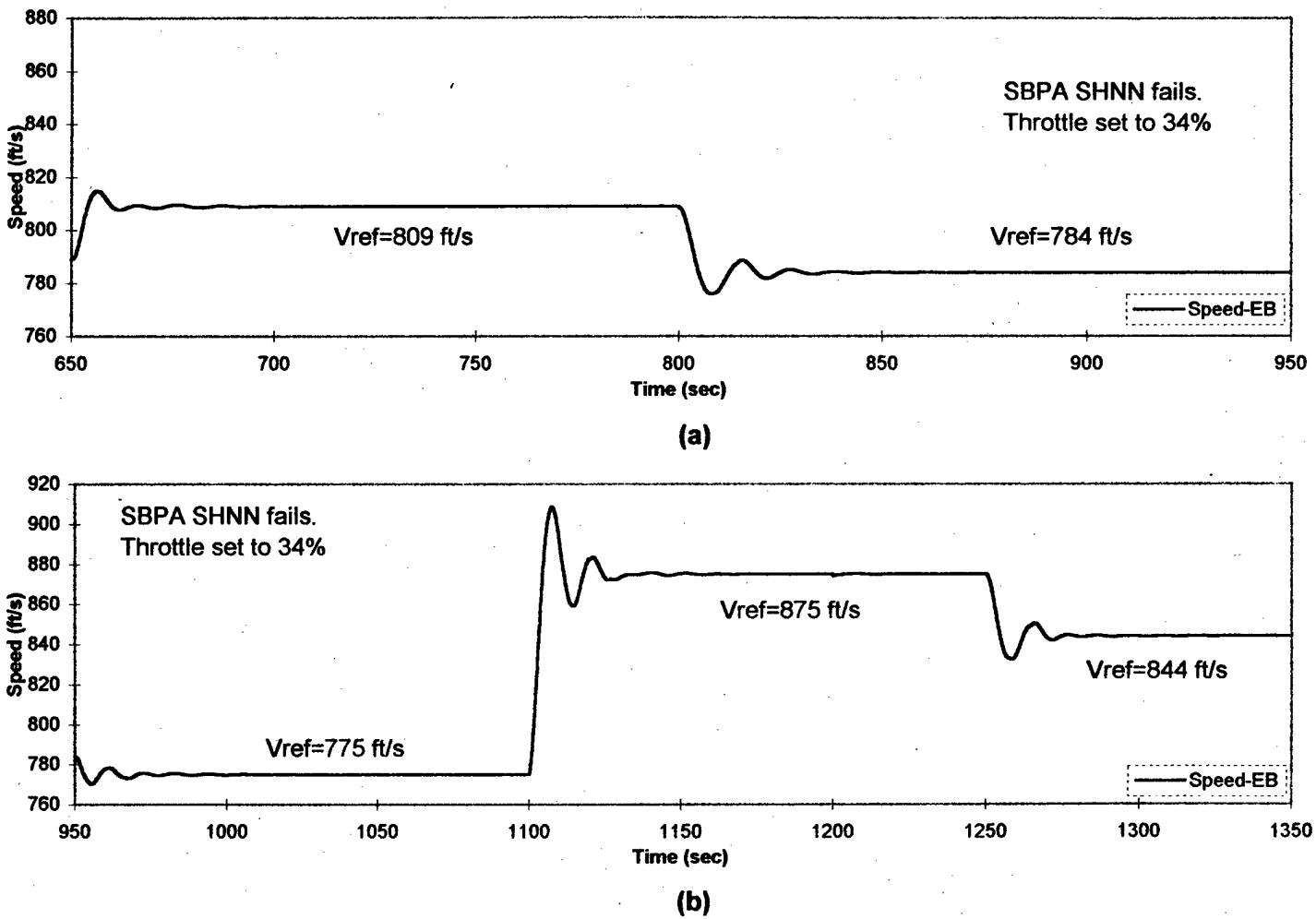


Figure 6.9: Speed response to commanded altitude and speed maneuvers at HC (refer to Table 6.2).

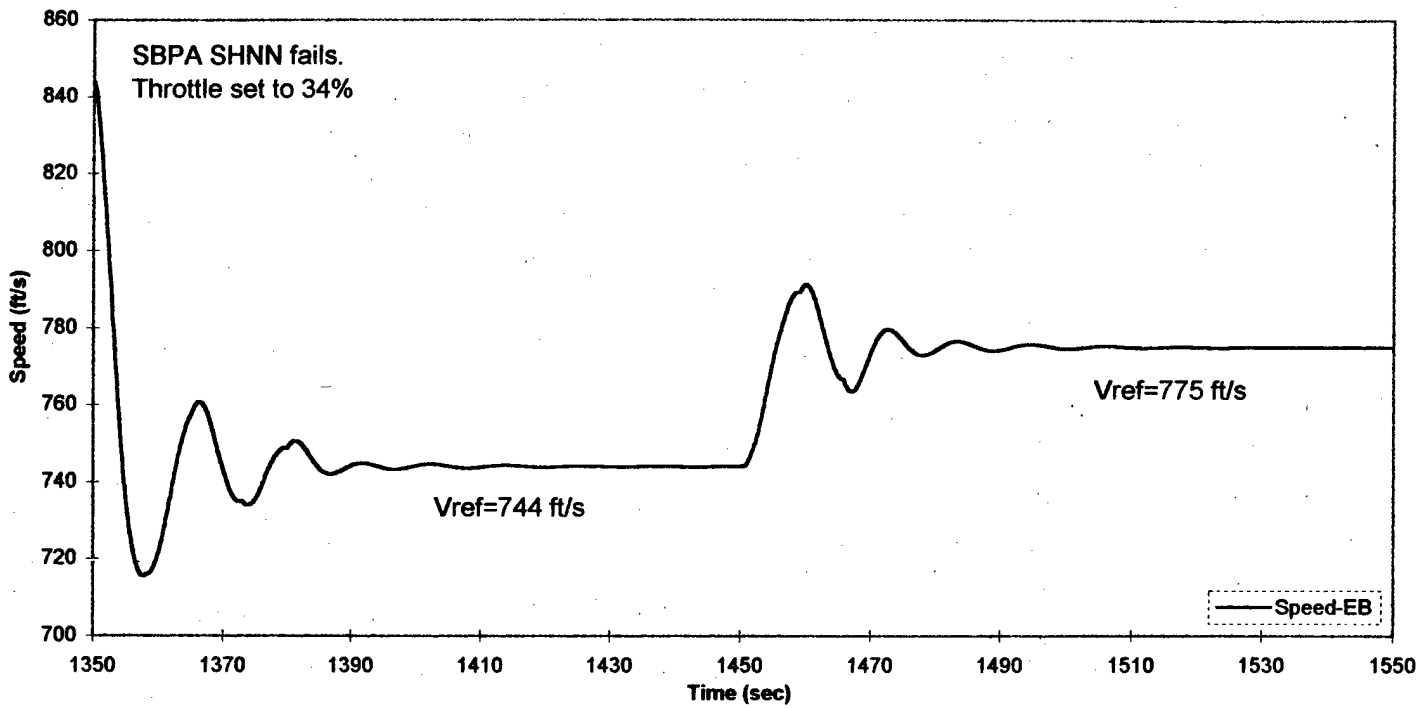
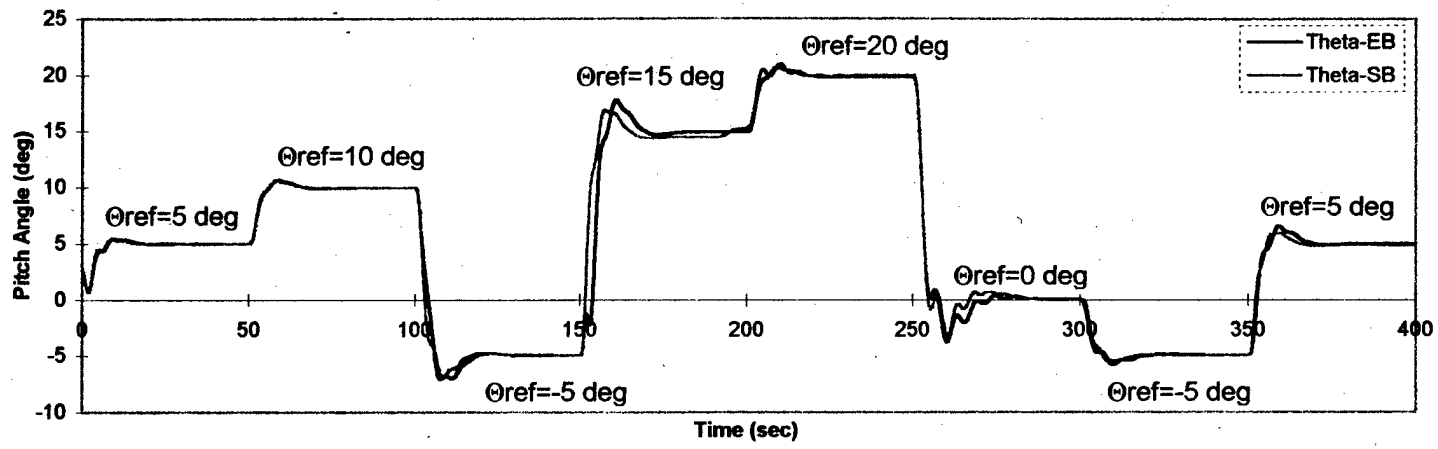
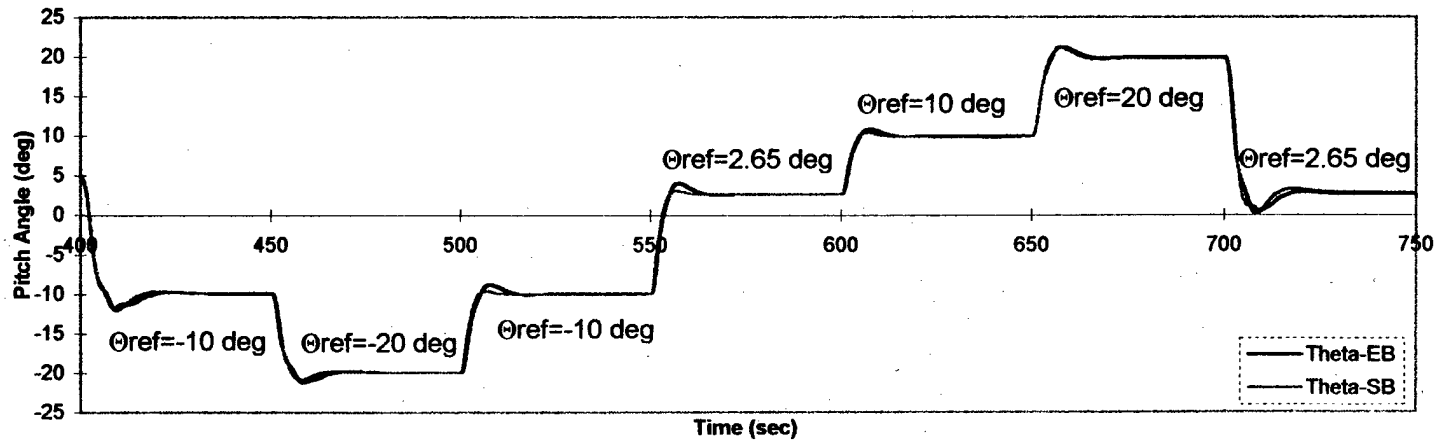


Figure 6.10: Speed response to commanded altitude and speed maneuvers at HC (refer to Table 6.2).

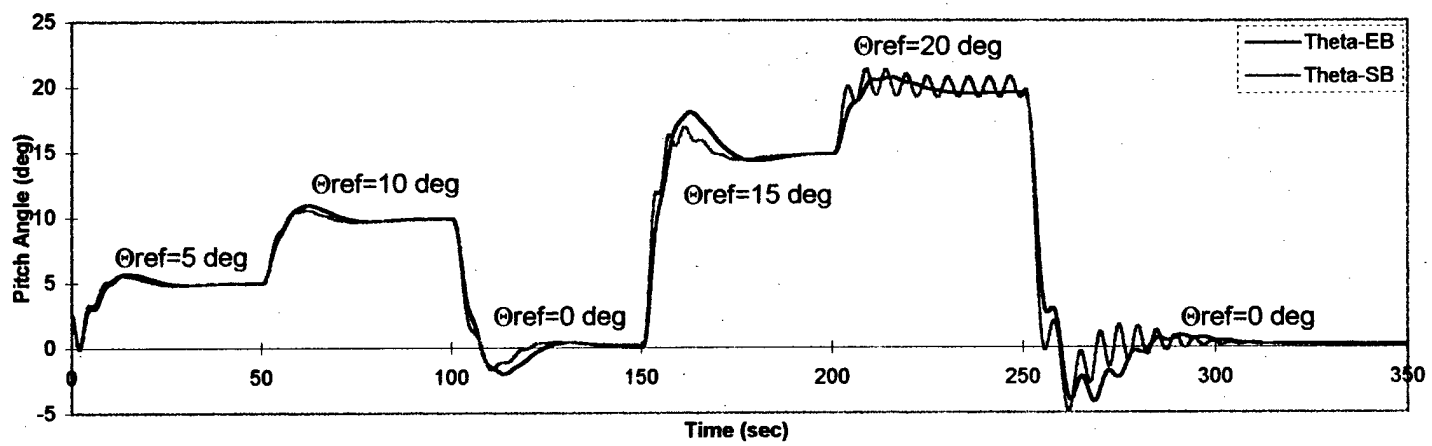


(a)

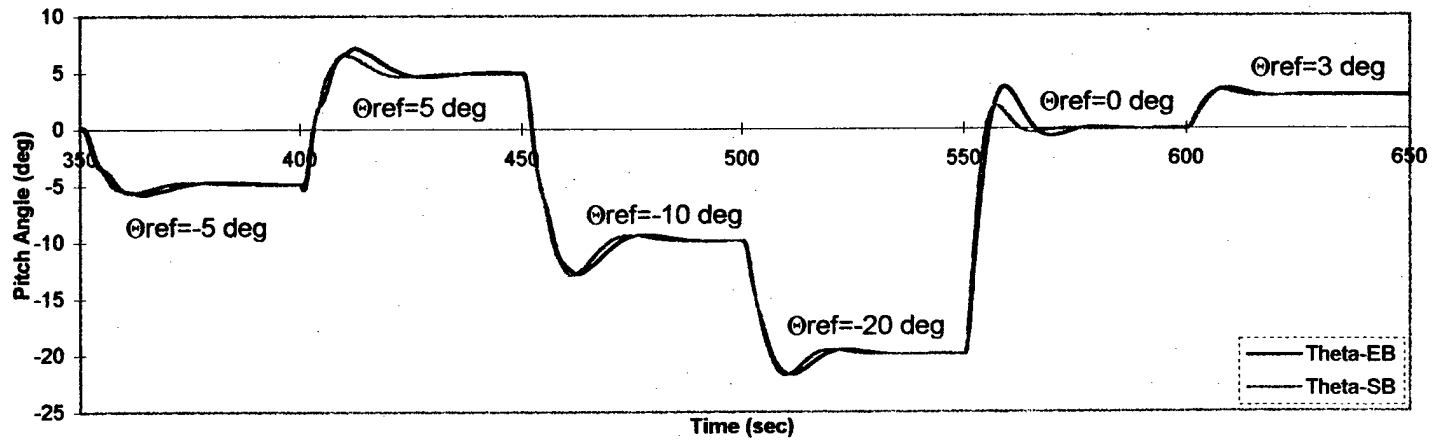


(b)

Figure 6.11: Pitch attitude response to commanded pitching maneuvers at LC.

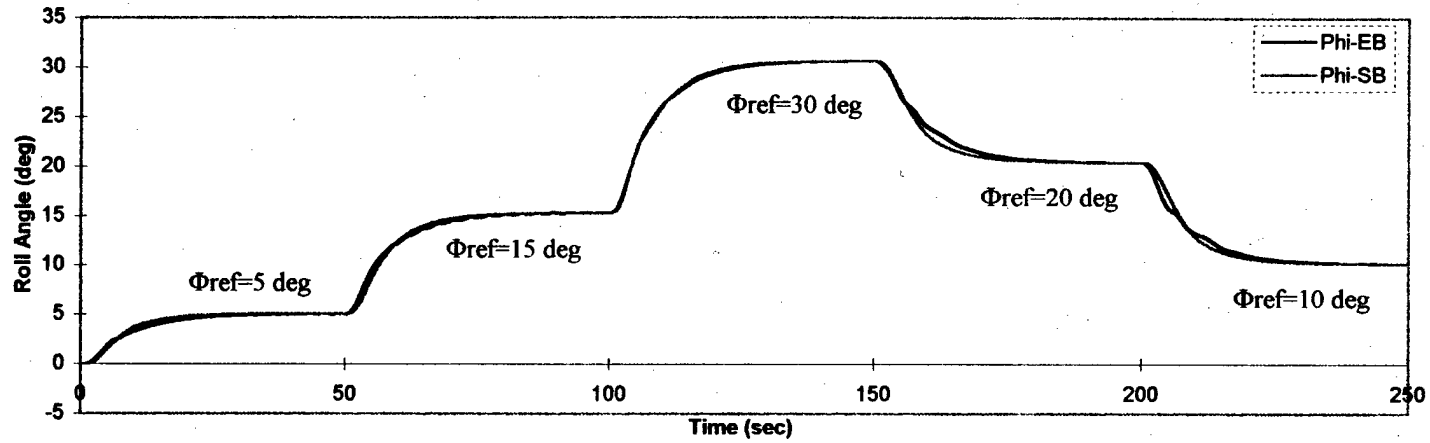


(a)

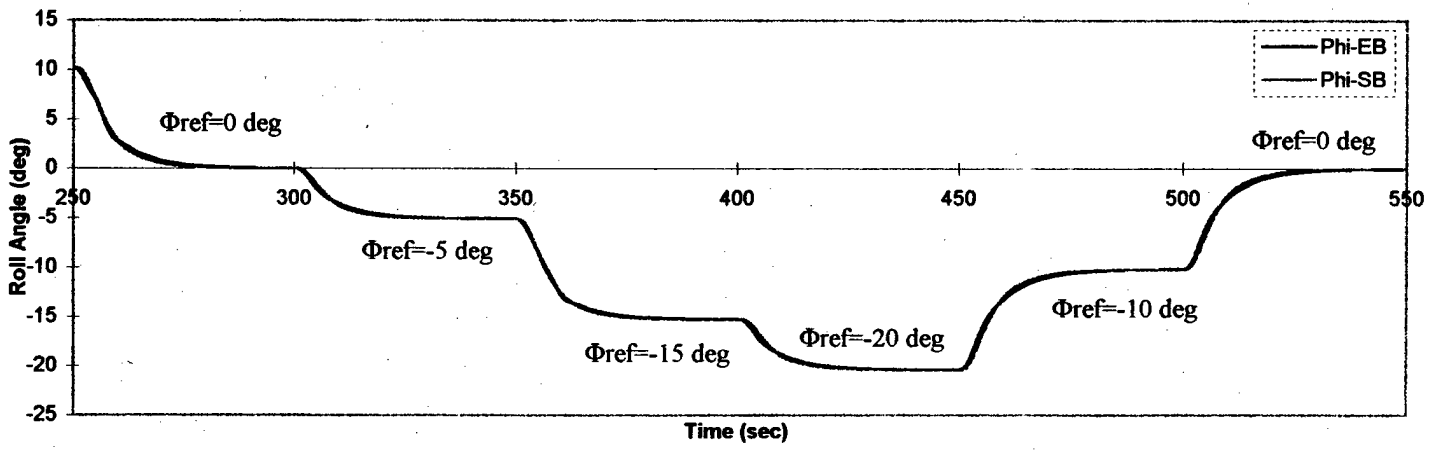


(b)

Figure 6.12: Pitch attitude response to command pitching maneuvers at HC.

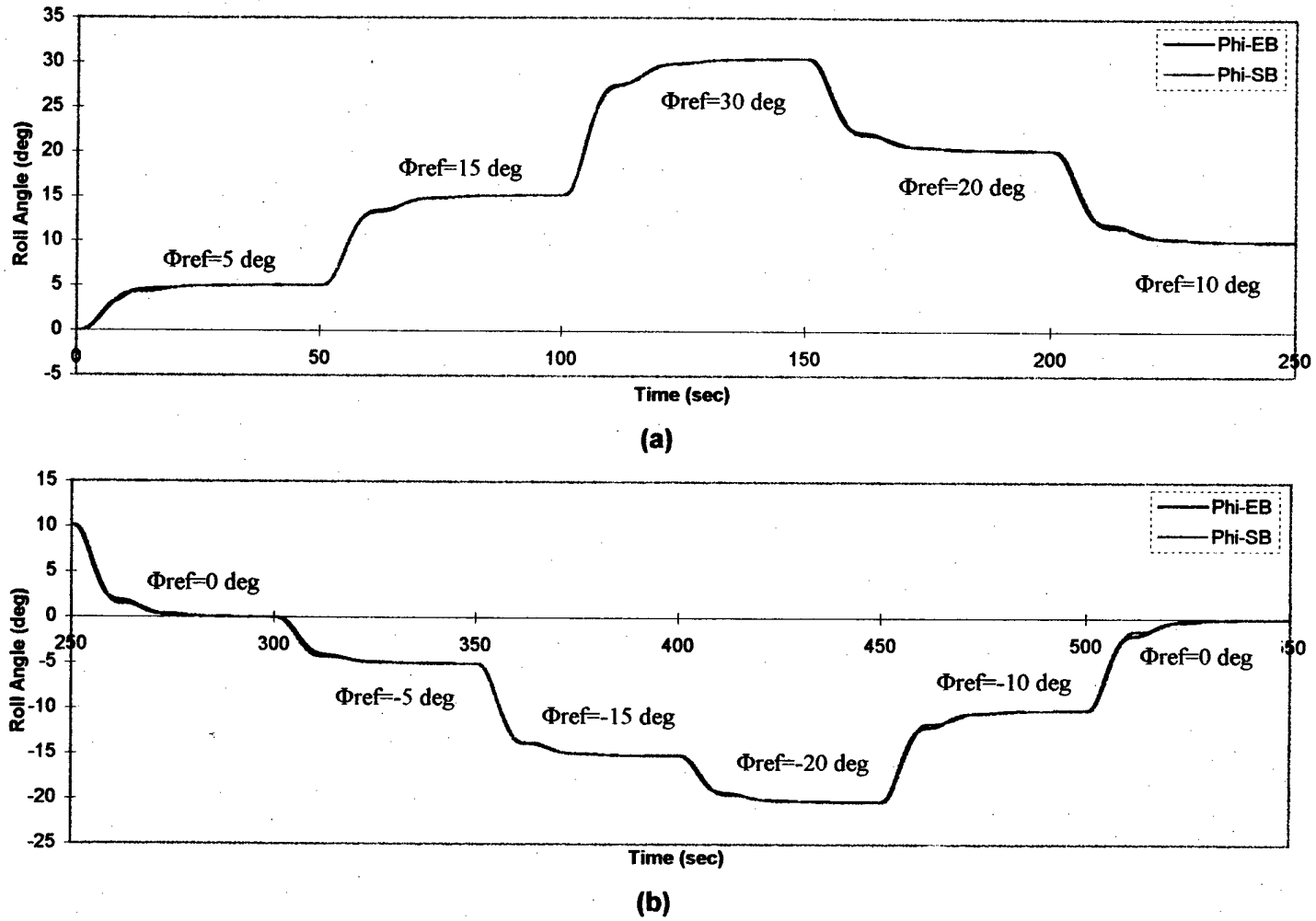


(a)



(b)

Figure 6.13: Roll angle response to commanded rolling maneuvers at LC.



Figrue 6.14: Roll angle response to commanded rolling maneuvers at HC.

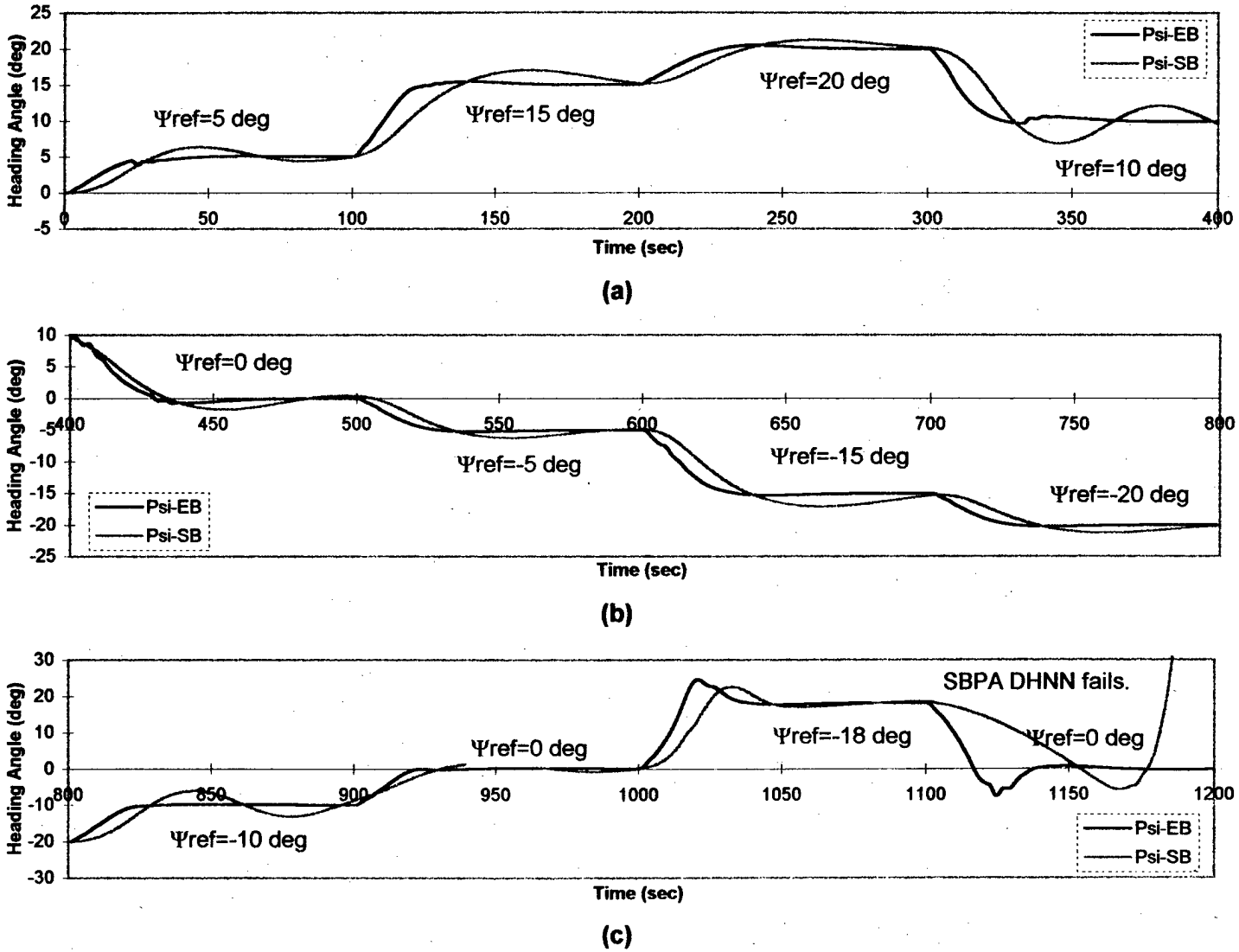
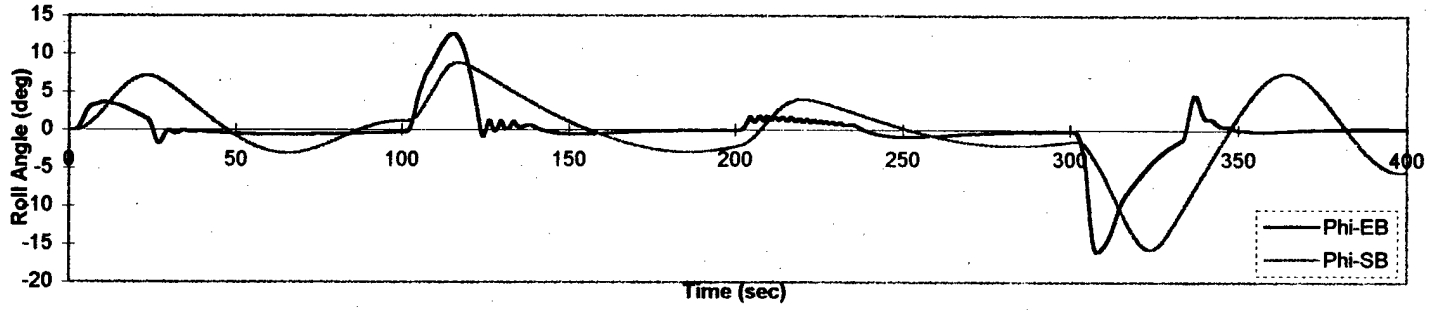
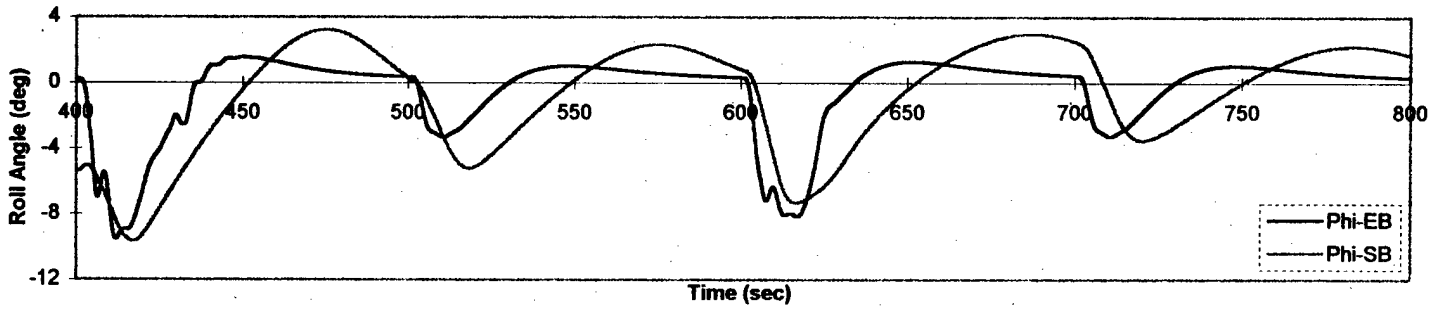


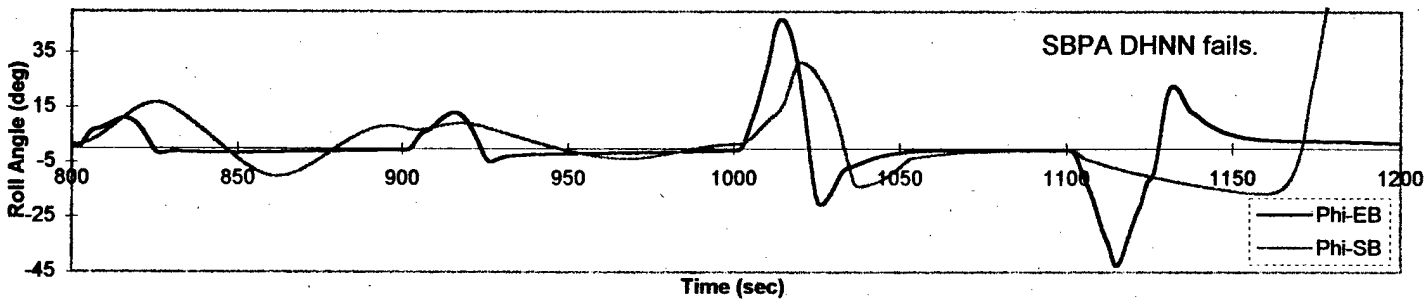
Figure 6.15: Heading angle response to commanded direction change maneuvers at LC.



(a)



(b)



(c)

Figure 6.16: Roll angle response to commanded heading angle maneuvers at LC.

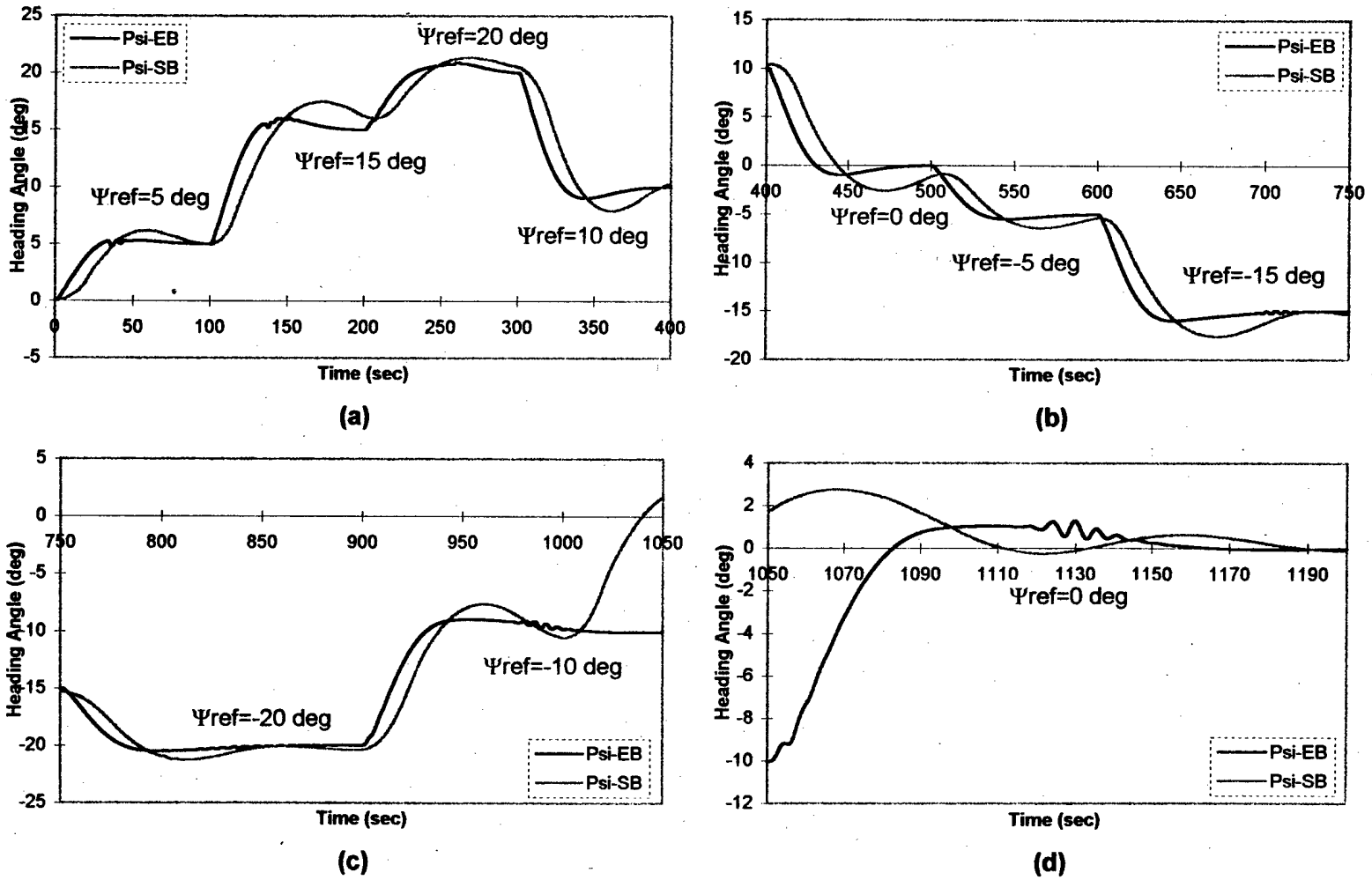


Figure 6.17: Heading angle response to commanded direction change maneuvers at HC.

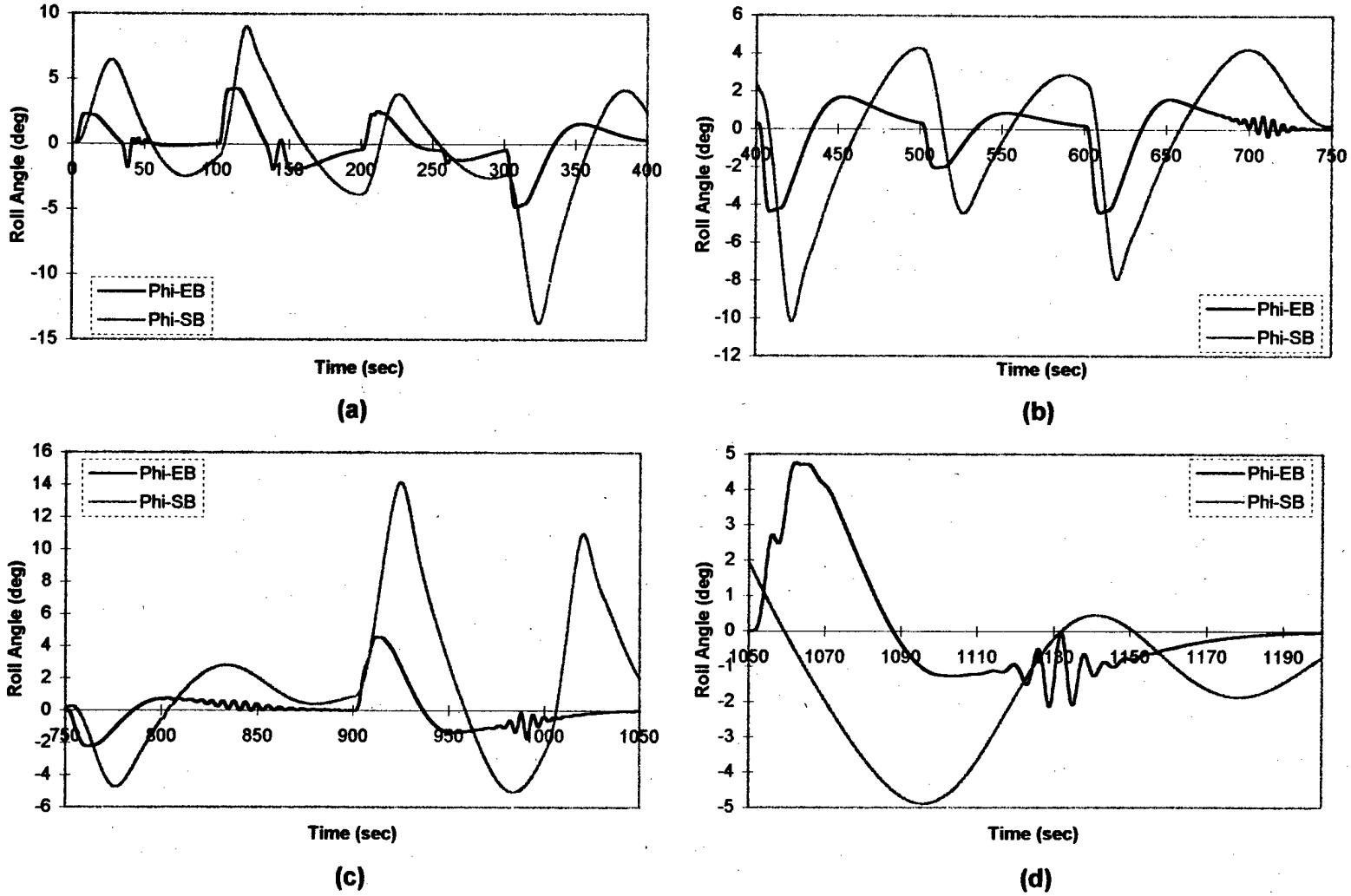


Figure 6.18: Roll angle response to commanded heading angle maneuvers at HC.

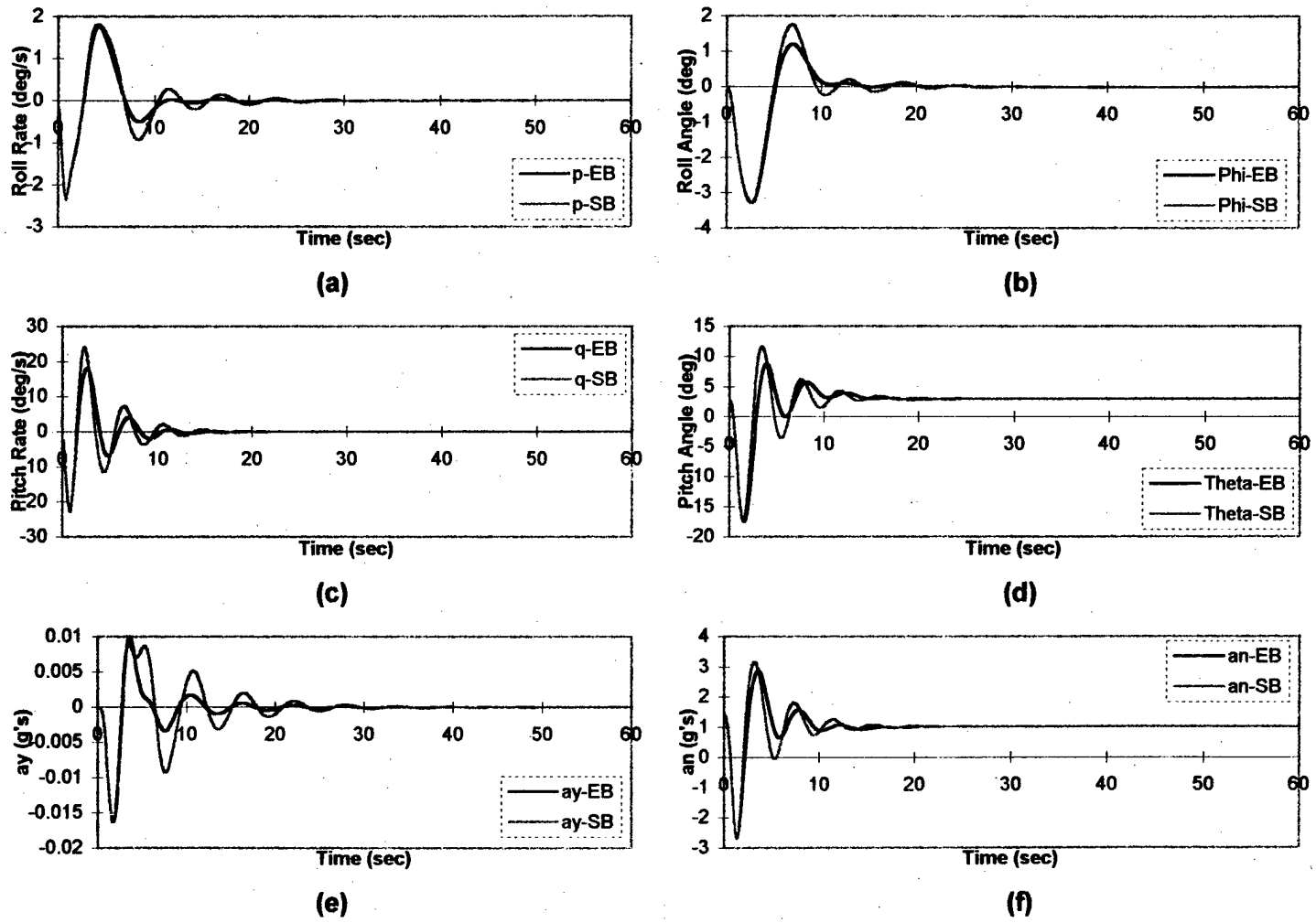
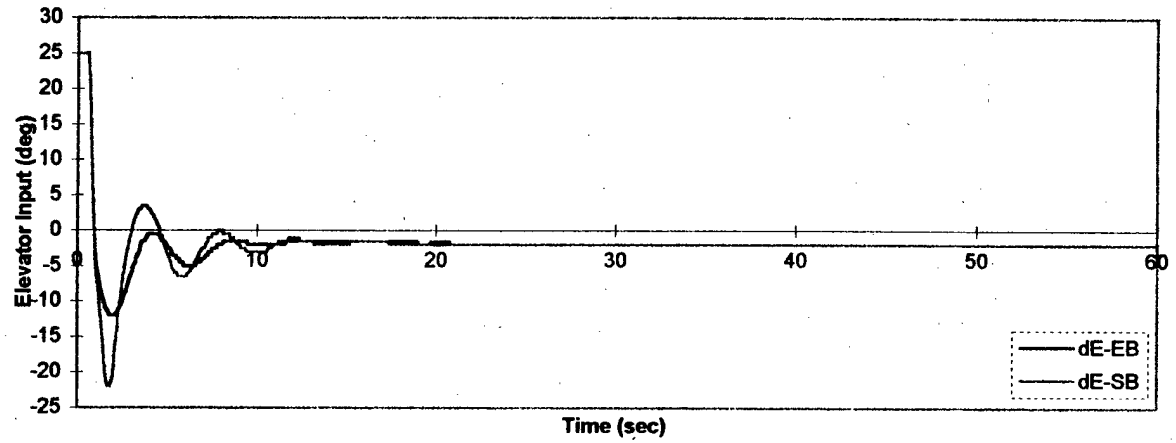
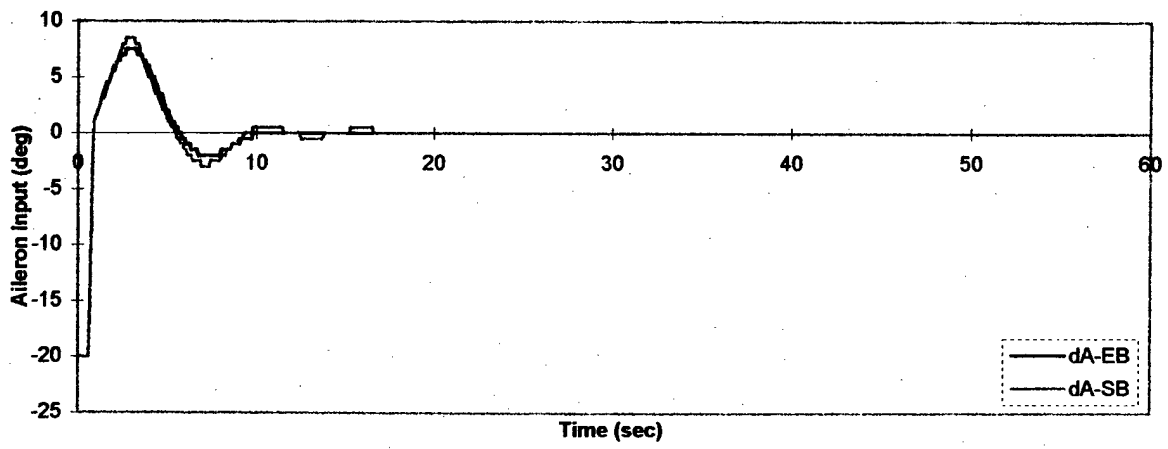


Figure 6.19: Aircraft response to non-linear conditions when p and q are above 2 deg/sec at LC.



(a)



(b)

Figure 6.20: NN control inputs in response to p and q excitations above 2 deg/sec at LC.

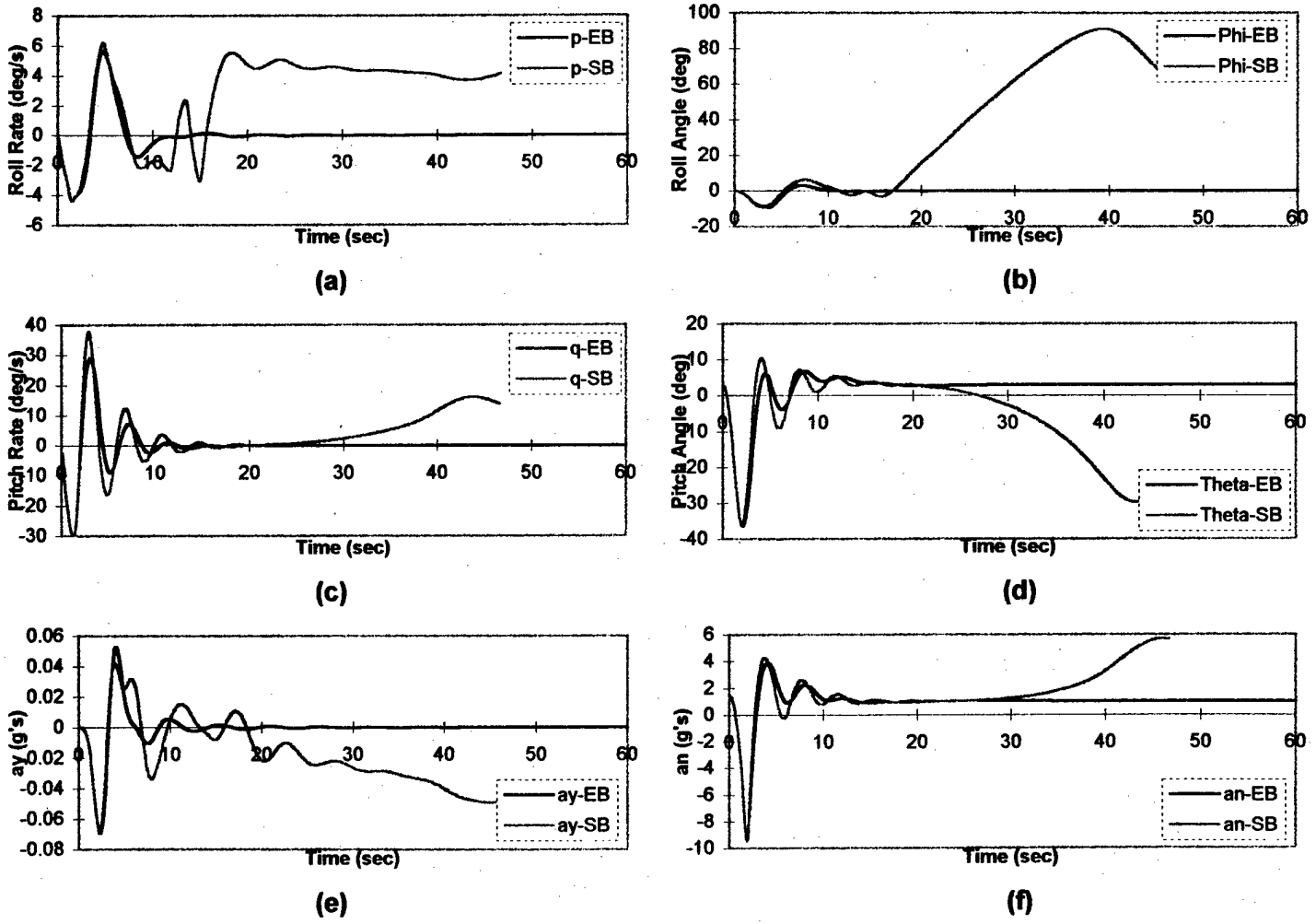


Figure 6.21: Aircraft response to non-linear conditions when p and q are above 4 deg/sec at LC.

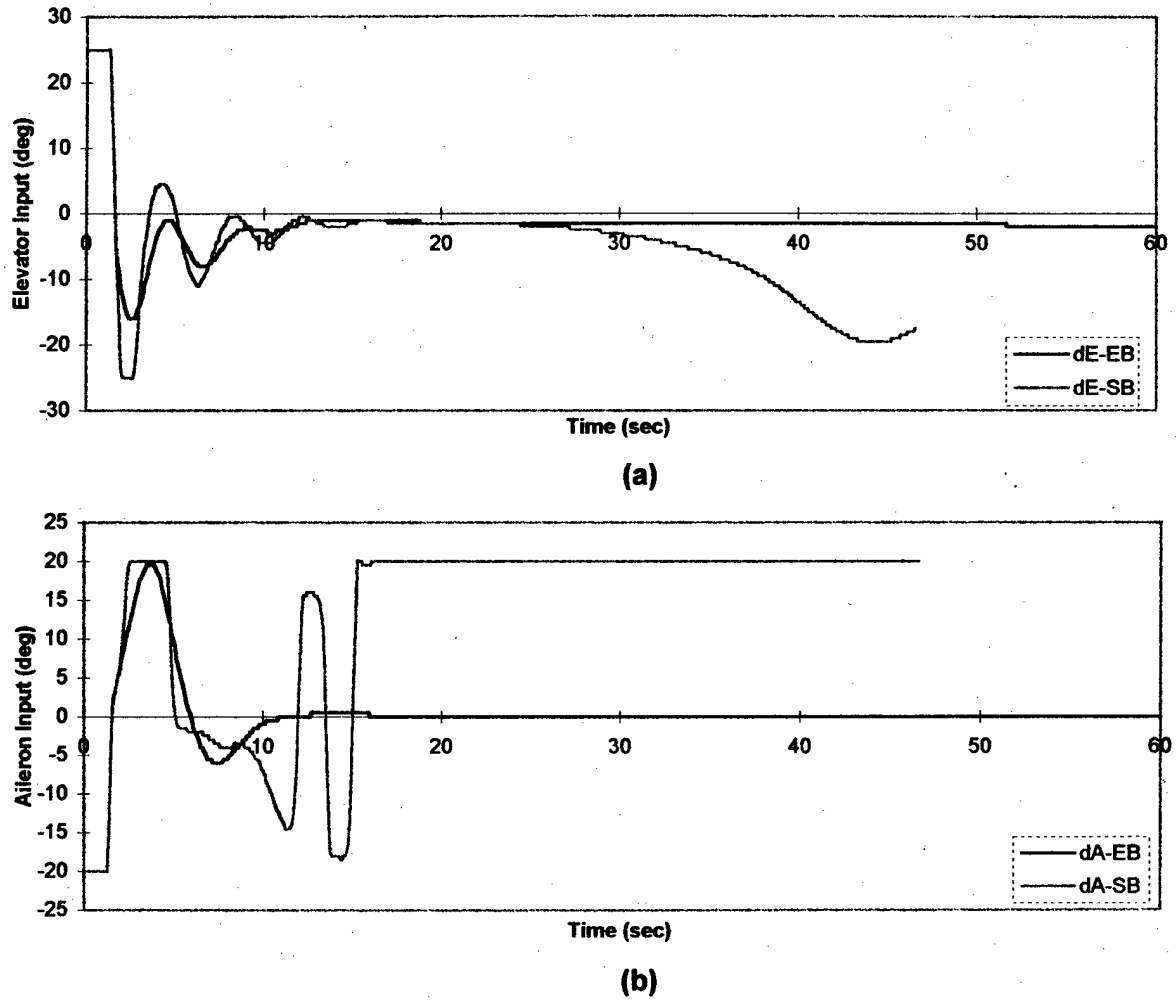


Figure 6.22: NN control inputs in response to p and q excitations above 4 deg/sec at LC.

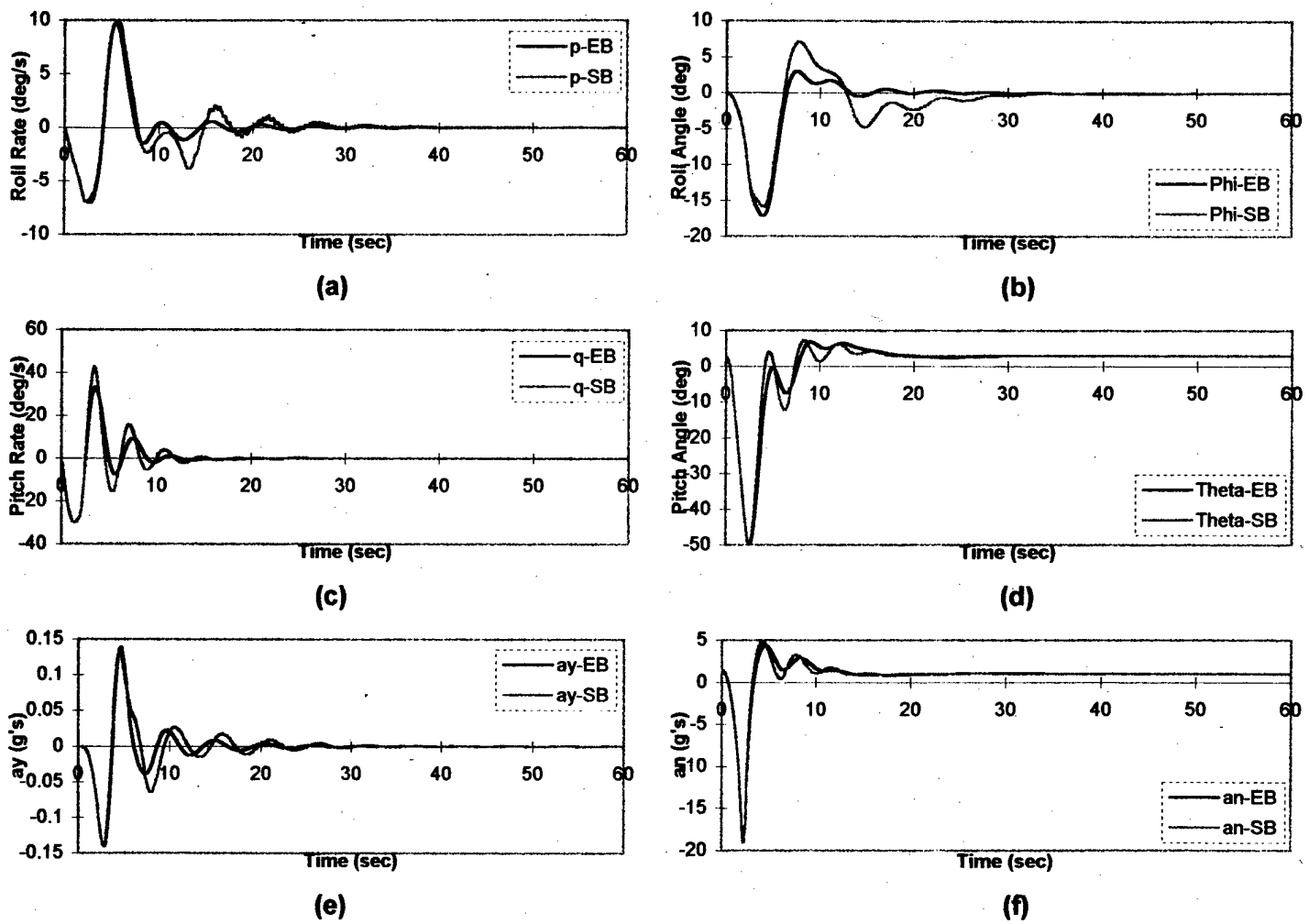
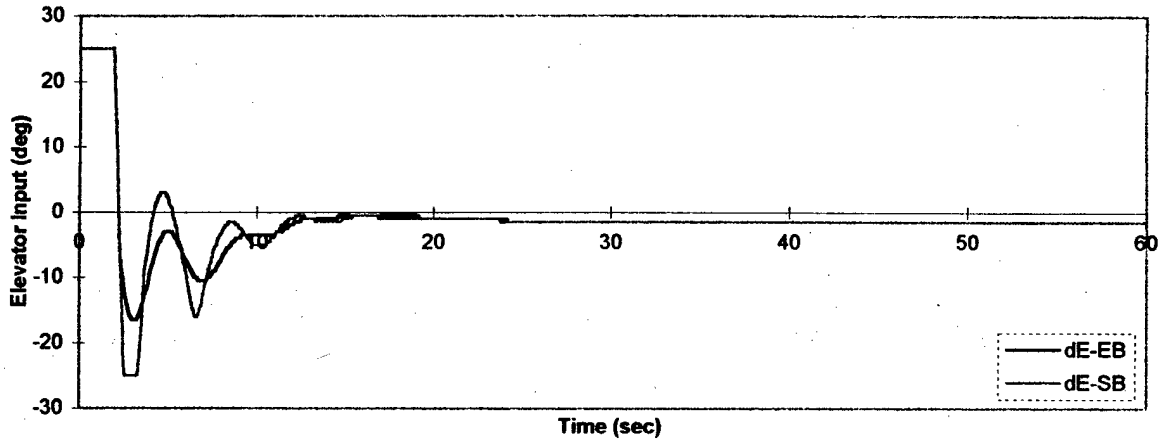
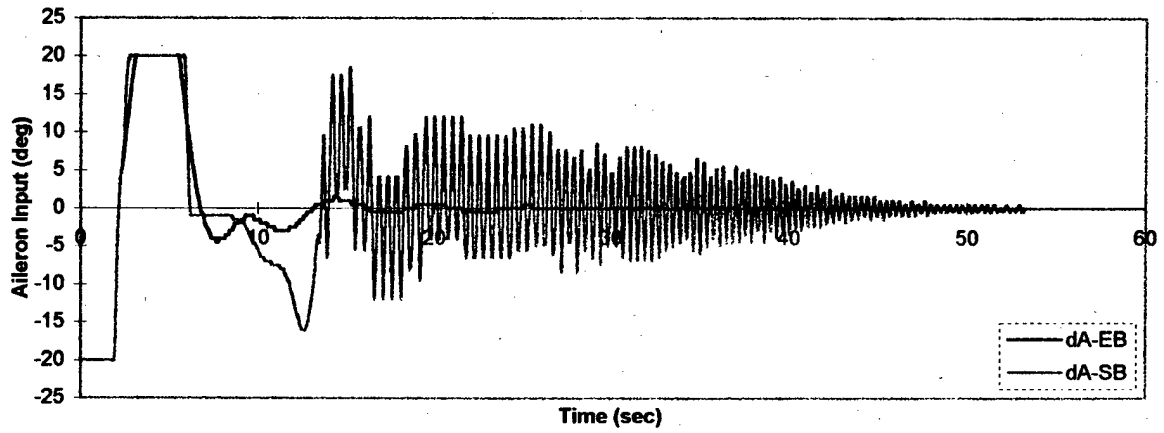


Figure 6.23: Aircraft response to non-linear conditions when p and q are above 6 deg/sec at LC.



(a)



(b)

Figure 6.24: NN control inputs in response to p and q excitations above 6 deg/sec at LC.

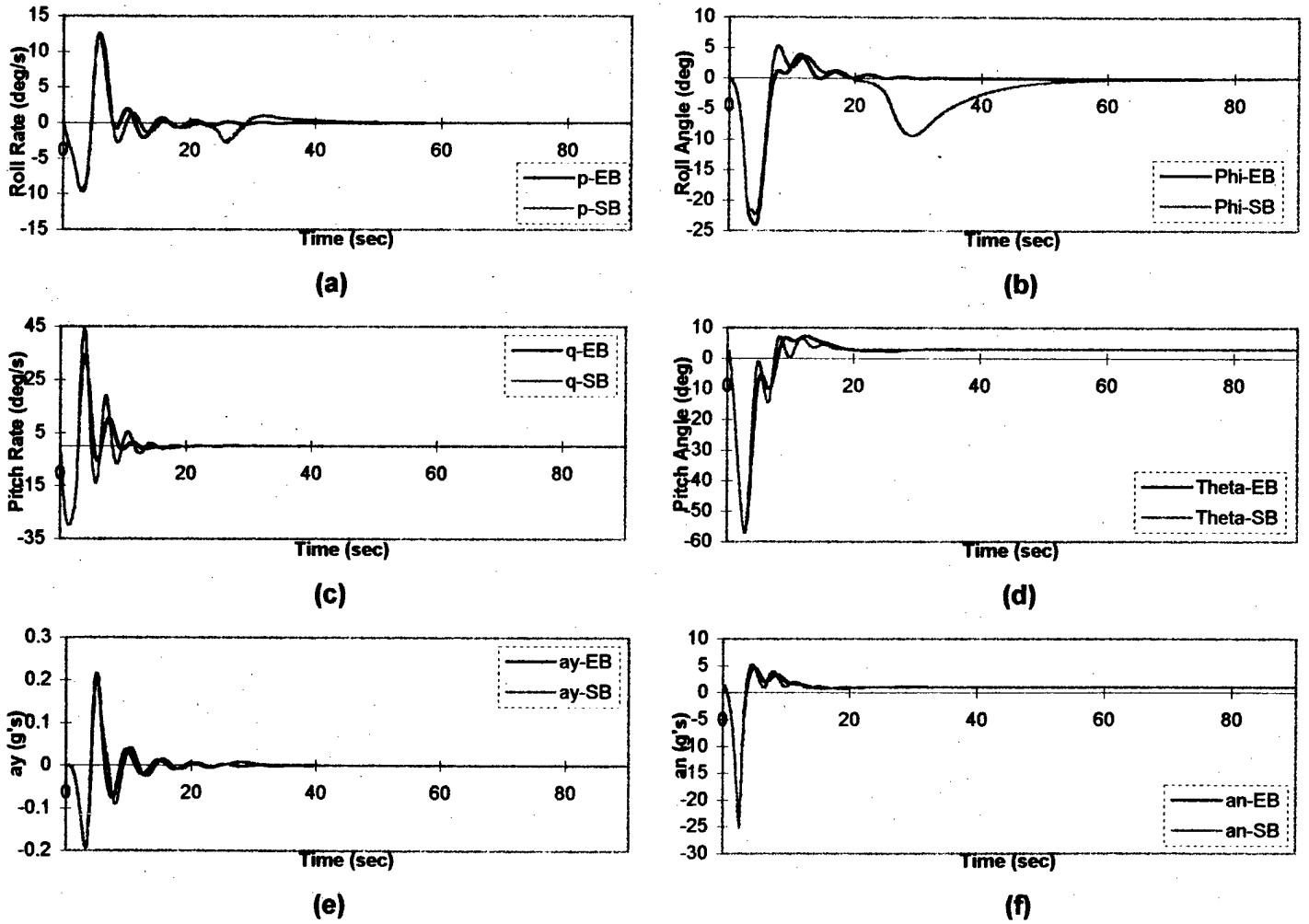
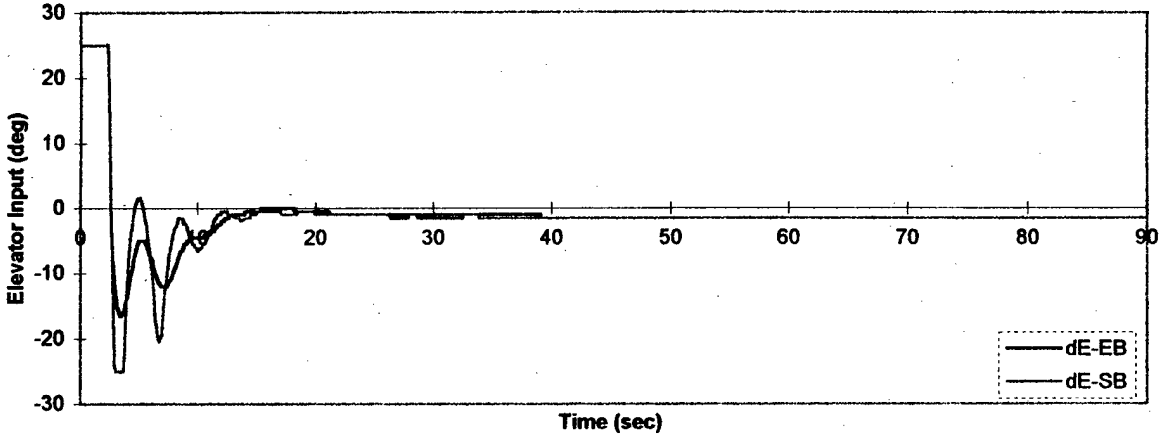
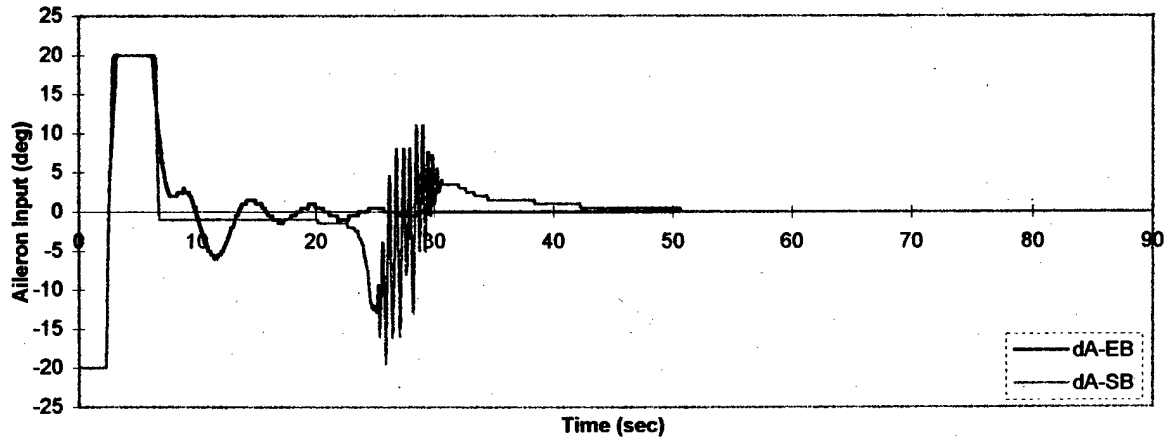


Figure 6.25: Aircraft response to non-linear conditions when p and q are above 8 deg/sec at LC.



(a)



(b)

Figure 6.26: NN control inputs in response to p and q excitations above 8 deg/sec at LC.

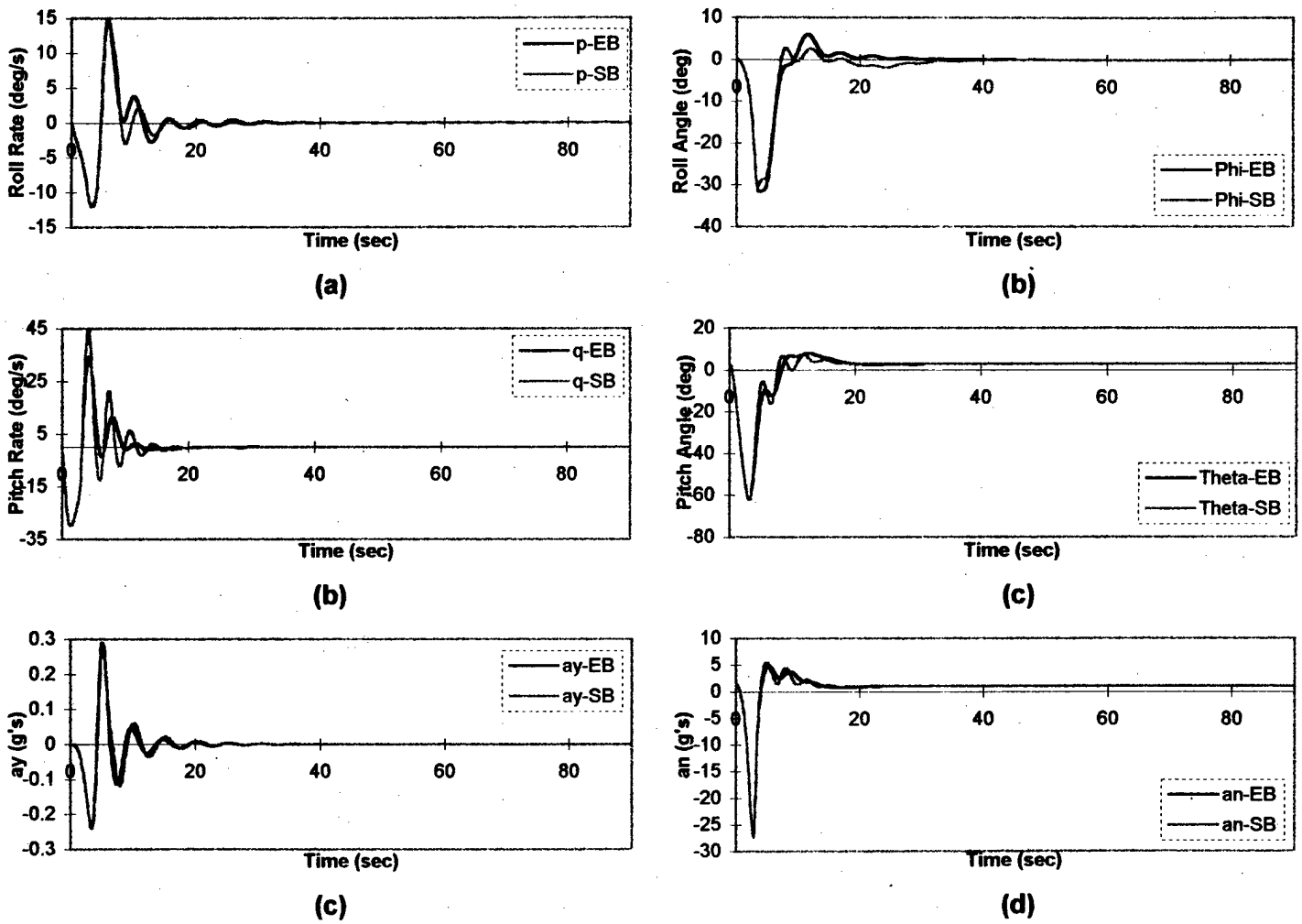
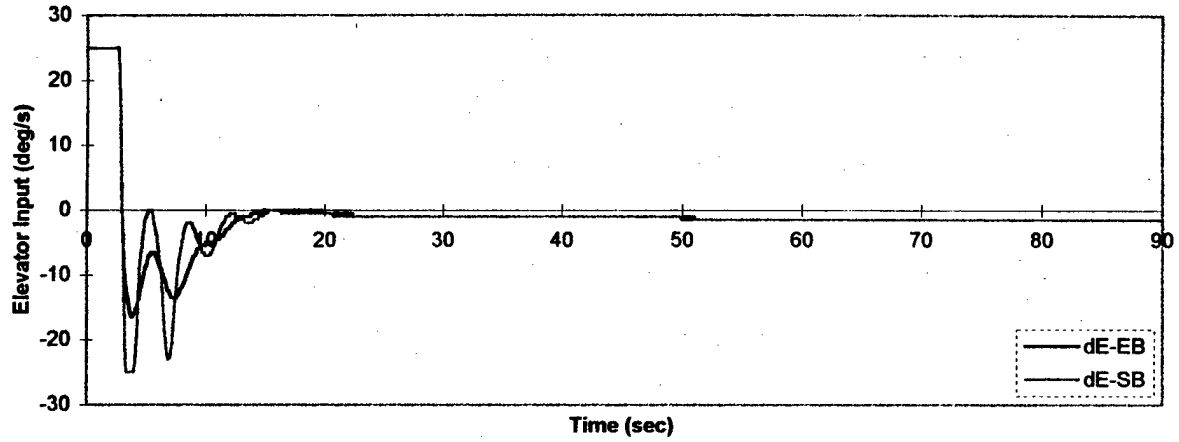
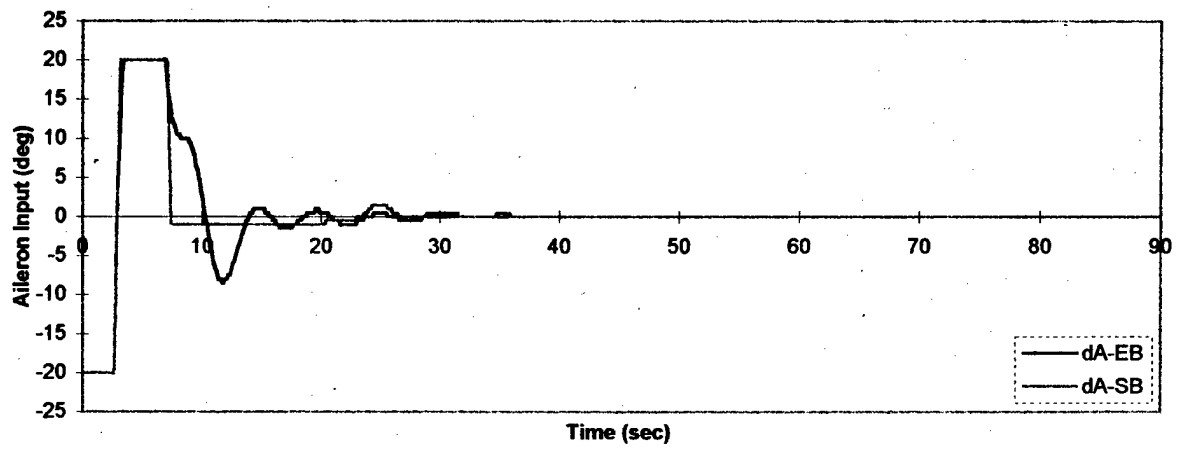


Figure 6.27: Aircraft response to non-linear conditions when p and q are above 10 deg/sec at LC.



(a)



(b)

Figure 6.28: NN control inputs in response to p and q excitations above 10 deg/sec at LC.

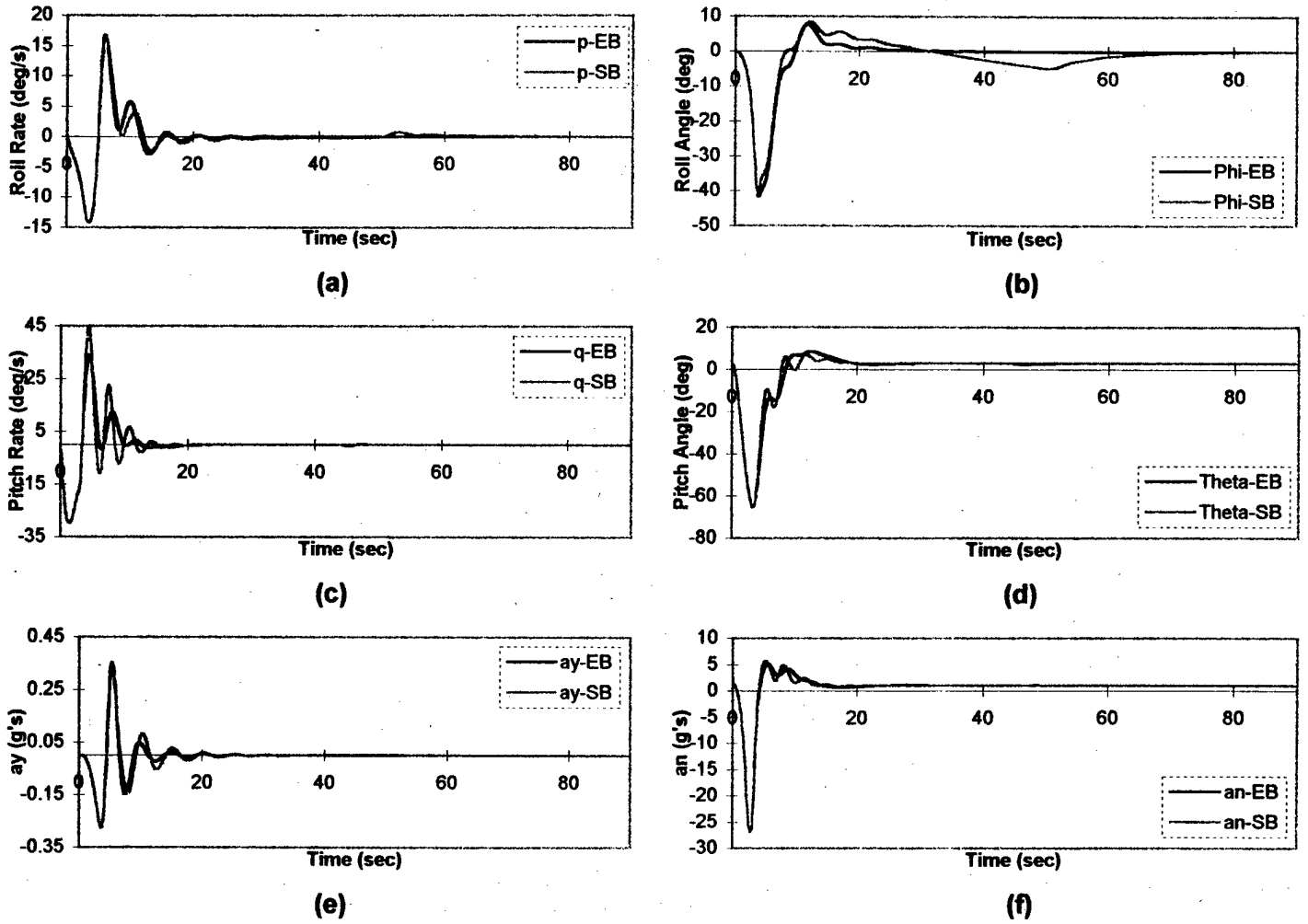


Figure 6.29: Aircraft response to non-linear conditions when p and q are above 12 deg/sec at LC.

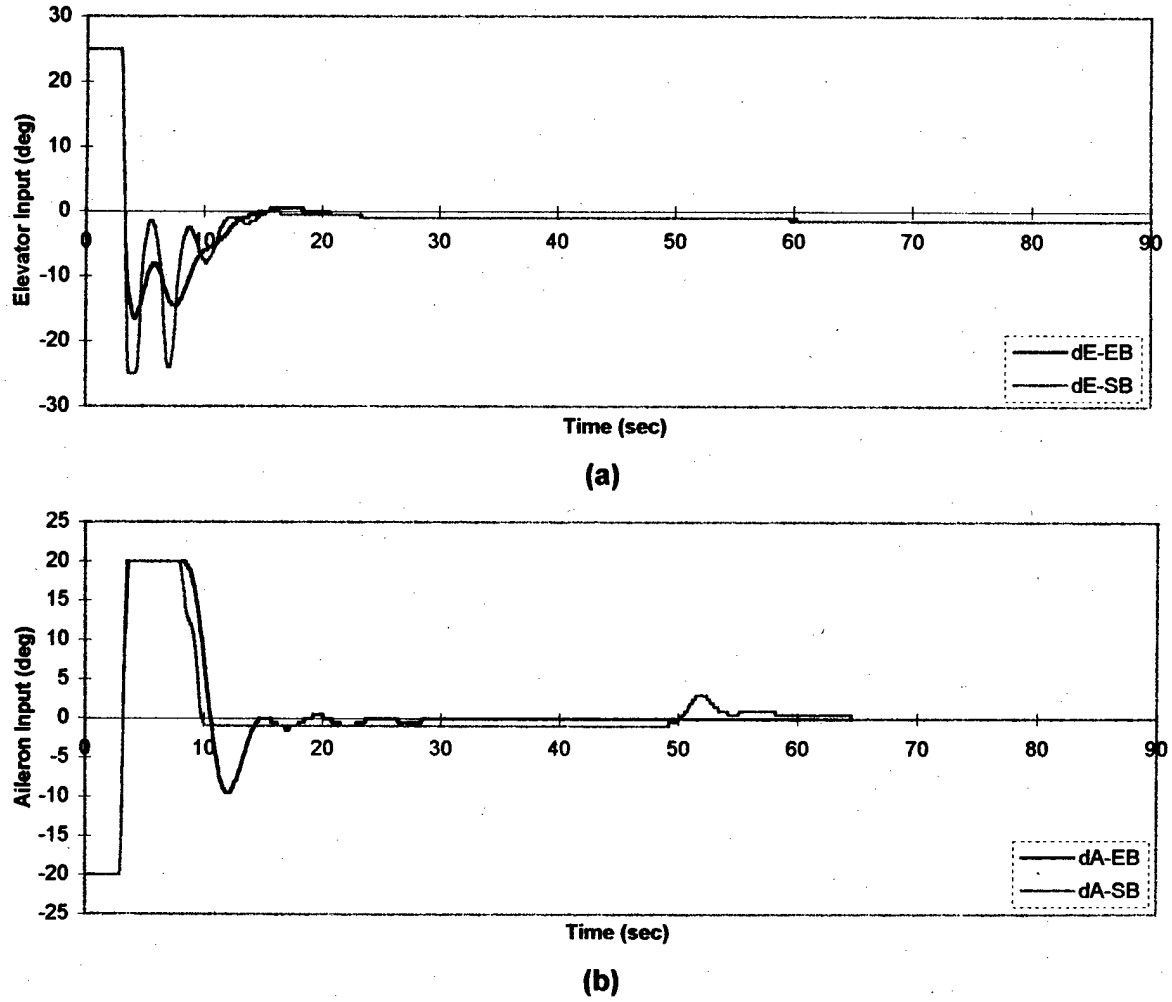


Figure 6.30: NN control inputs in response to p and q excitations above 12 deg/sec at LC.

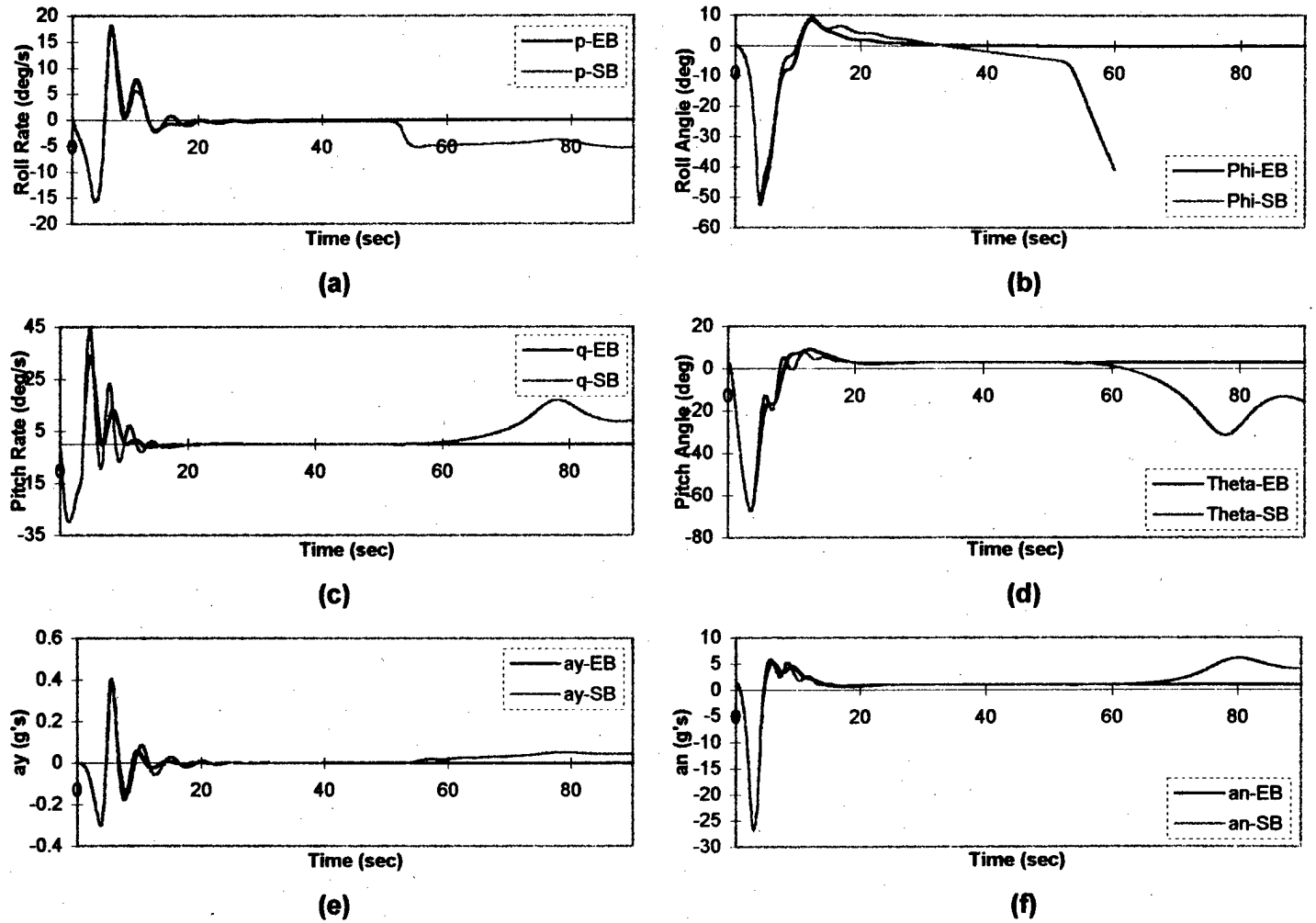
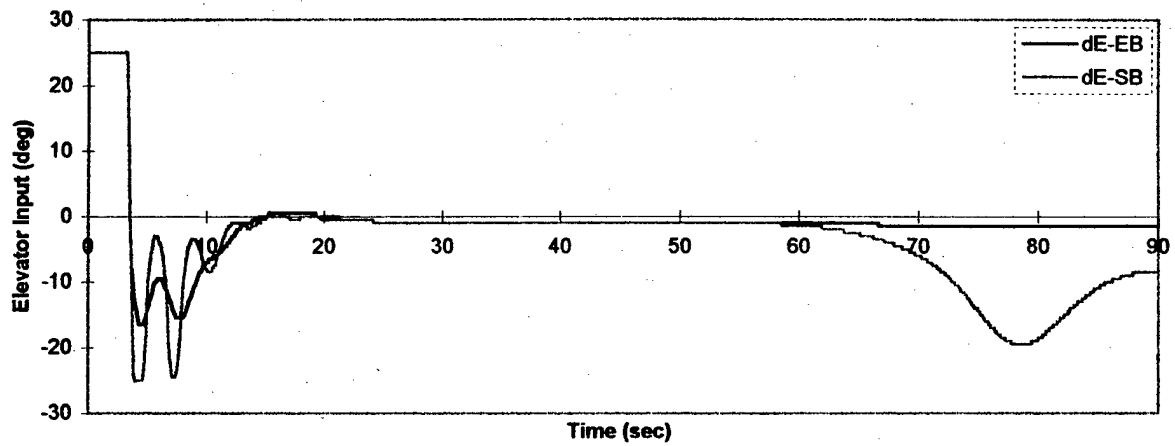
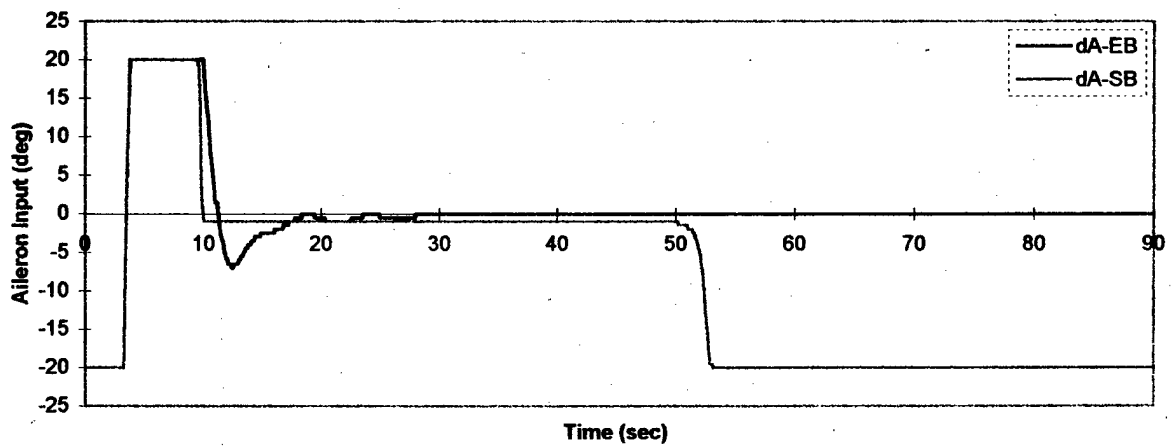


Figure 6.31: Aircraft response to non-linear conditions when p and q are above 14 deg/sec at LC.



(a)



(b)

Figure 6.32: NN control inputs in response to p and q excitations above 14 deg/sec at LC.

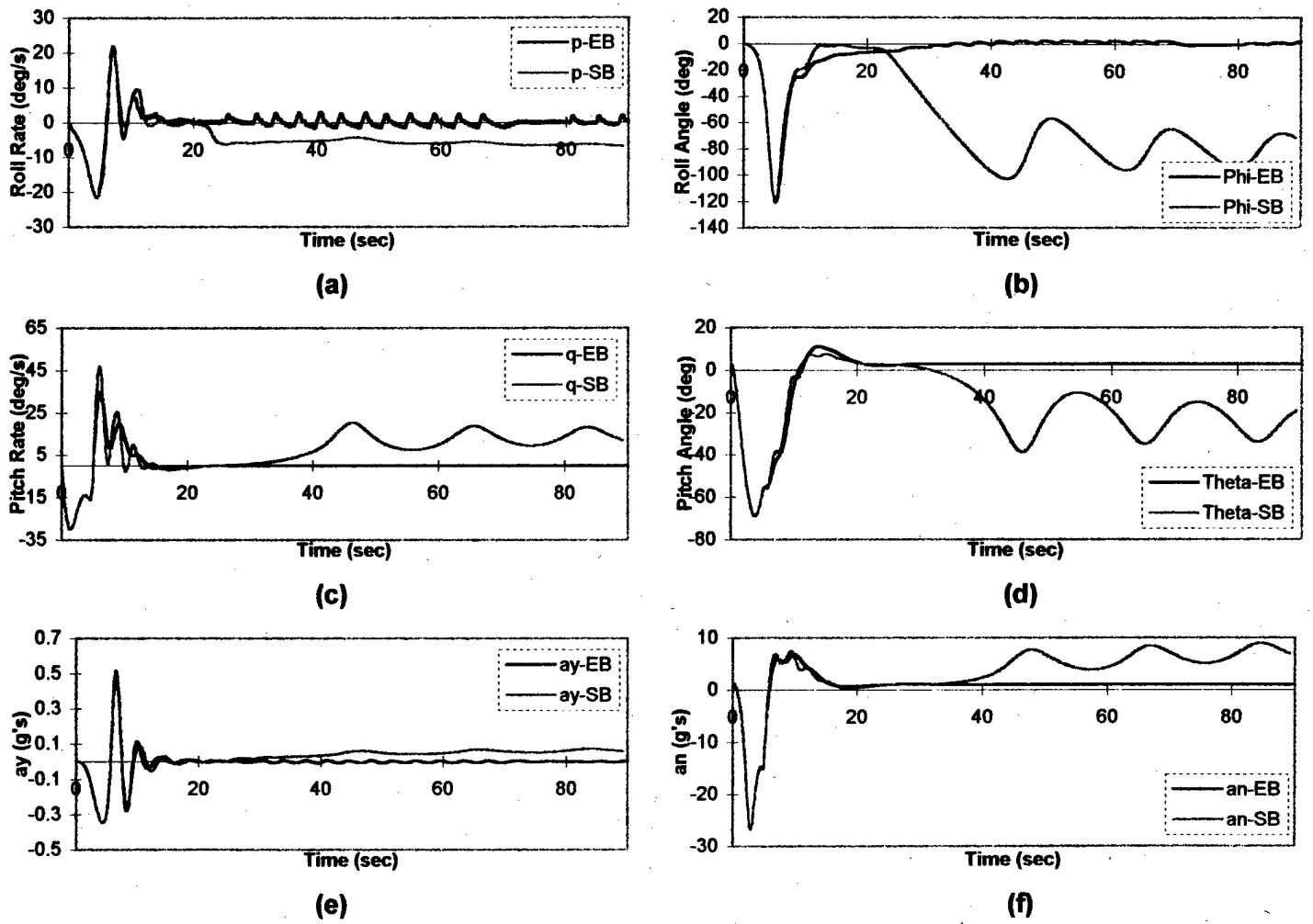
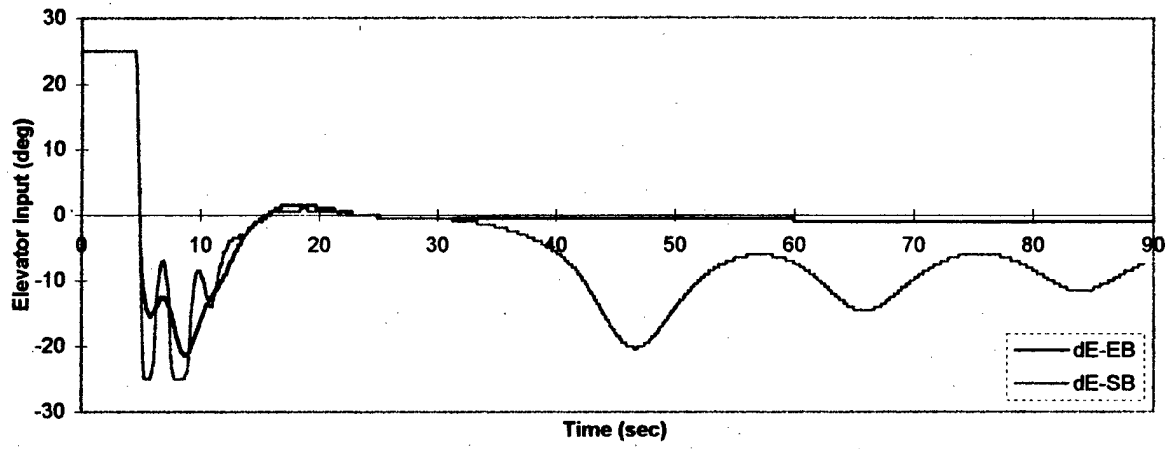
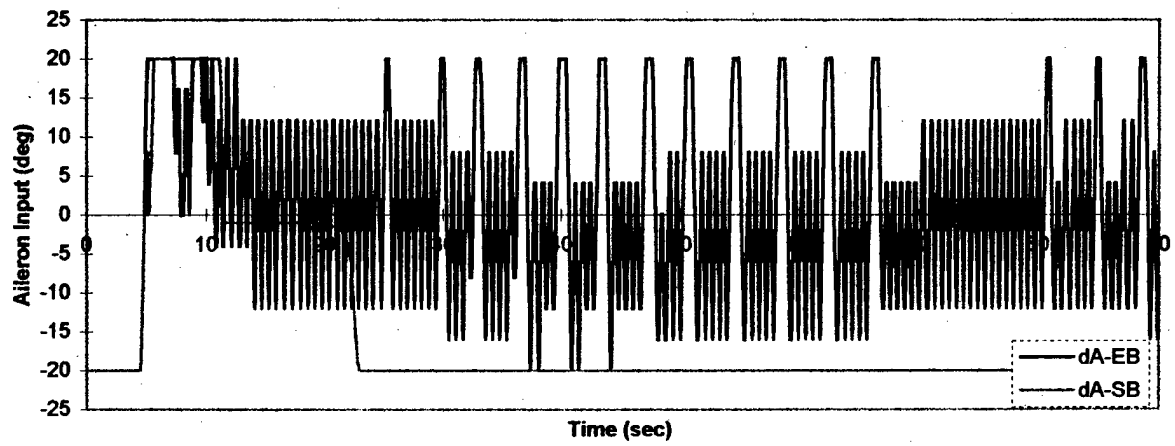


Figure 6.33: Aircraft response to non-linear conditions when p and q are above 16 deg/sec at LC.



(a)



(b)

Figure 6.34: NN control inputs in response to p and q excitations above 16 deg/sec at LC.

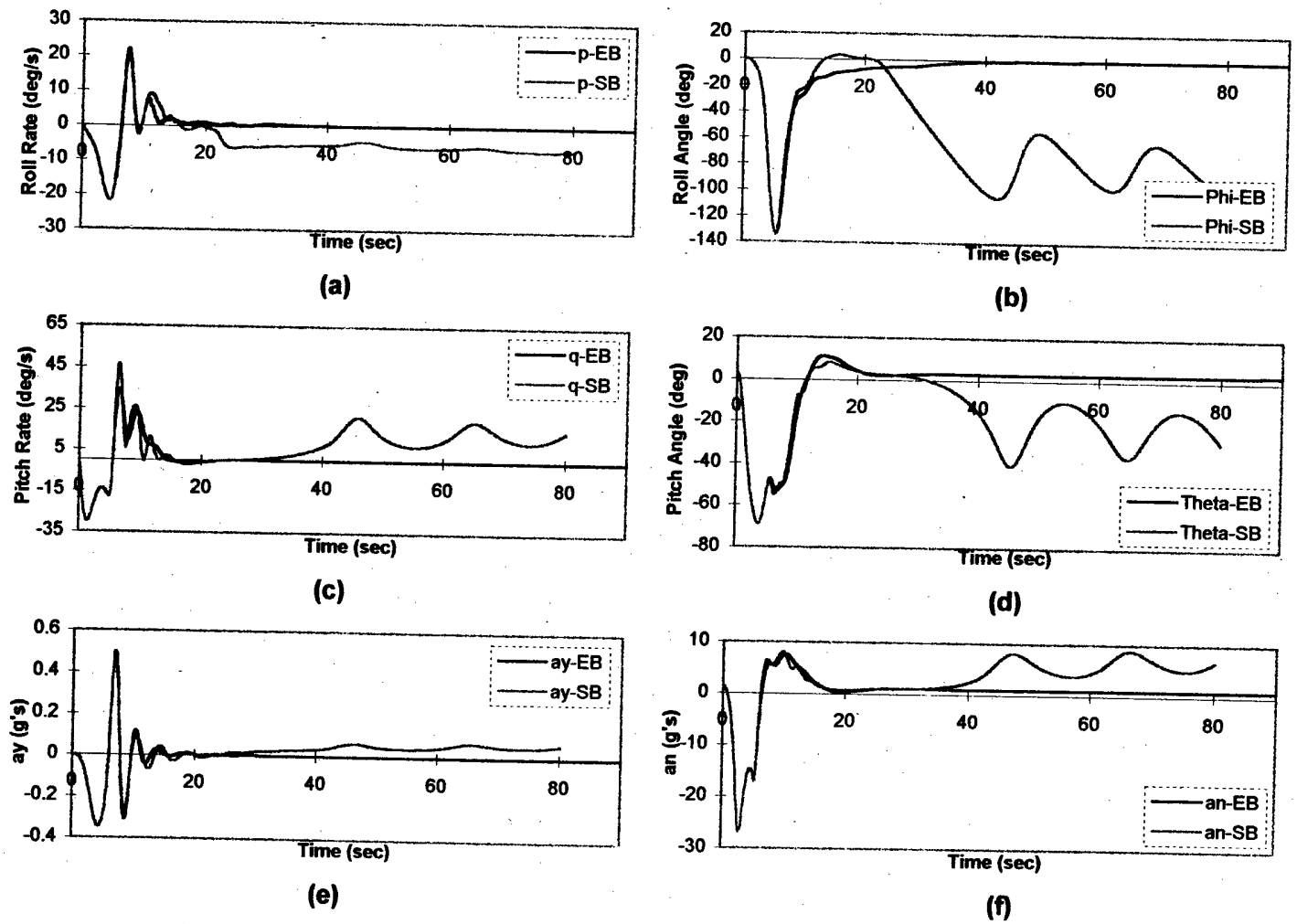
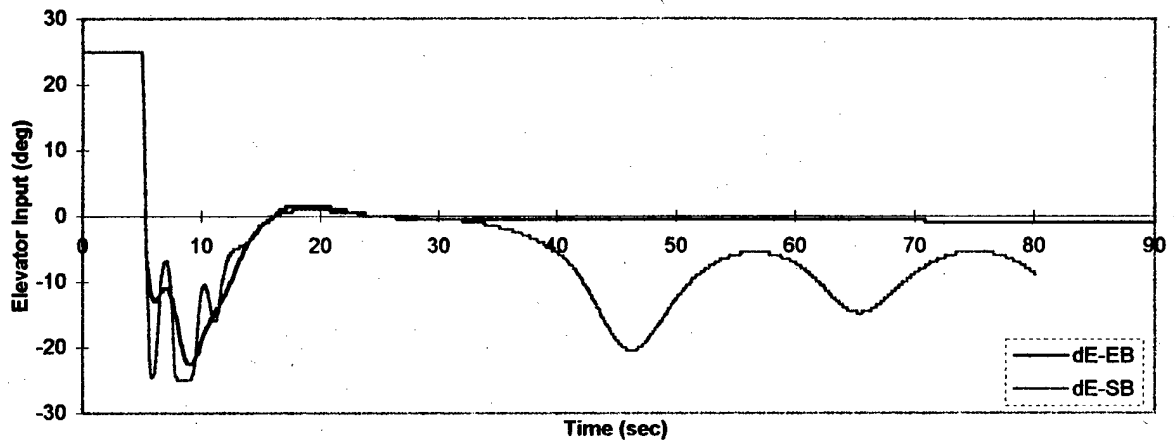
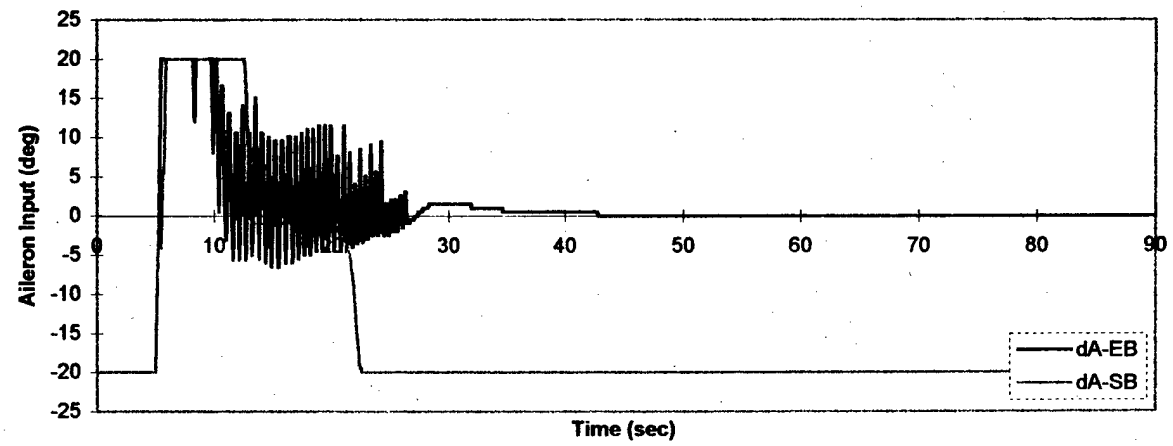


Figure 6.35: Aircraft response to non-linear conditions when p and q are above 18 deg/sec at LC.



(a)



(b)

Figure 6.36: NN control inputs in response to p and q excitations above 18 deg/sec at LC.

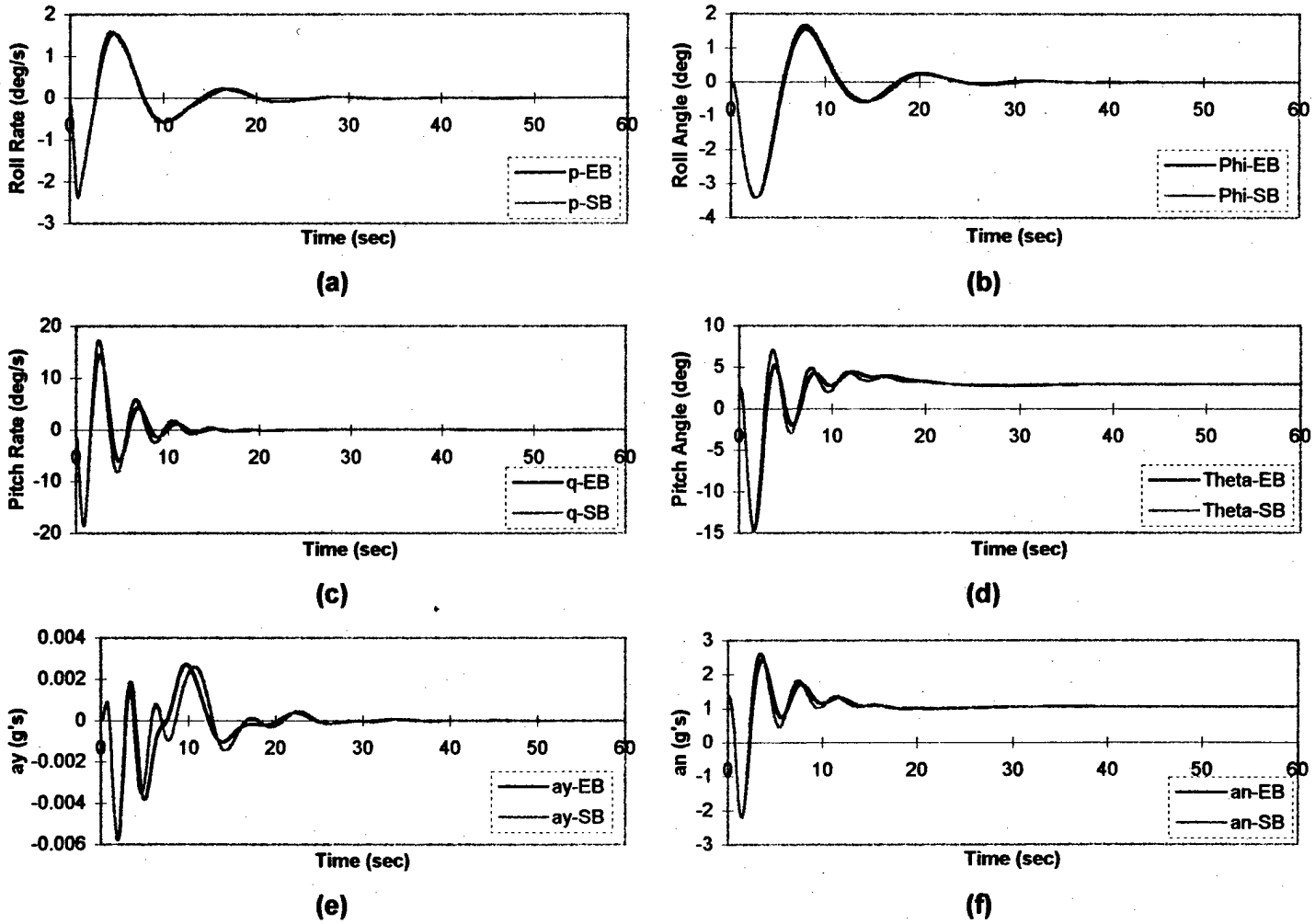


Figure 6.37: Aircraft response to non-linear conditions when p and q are above 2 deg/sec at HC.

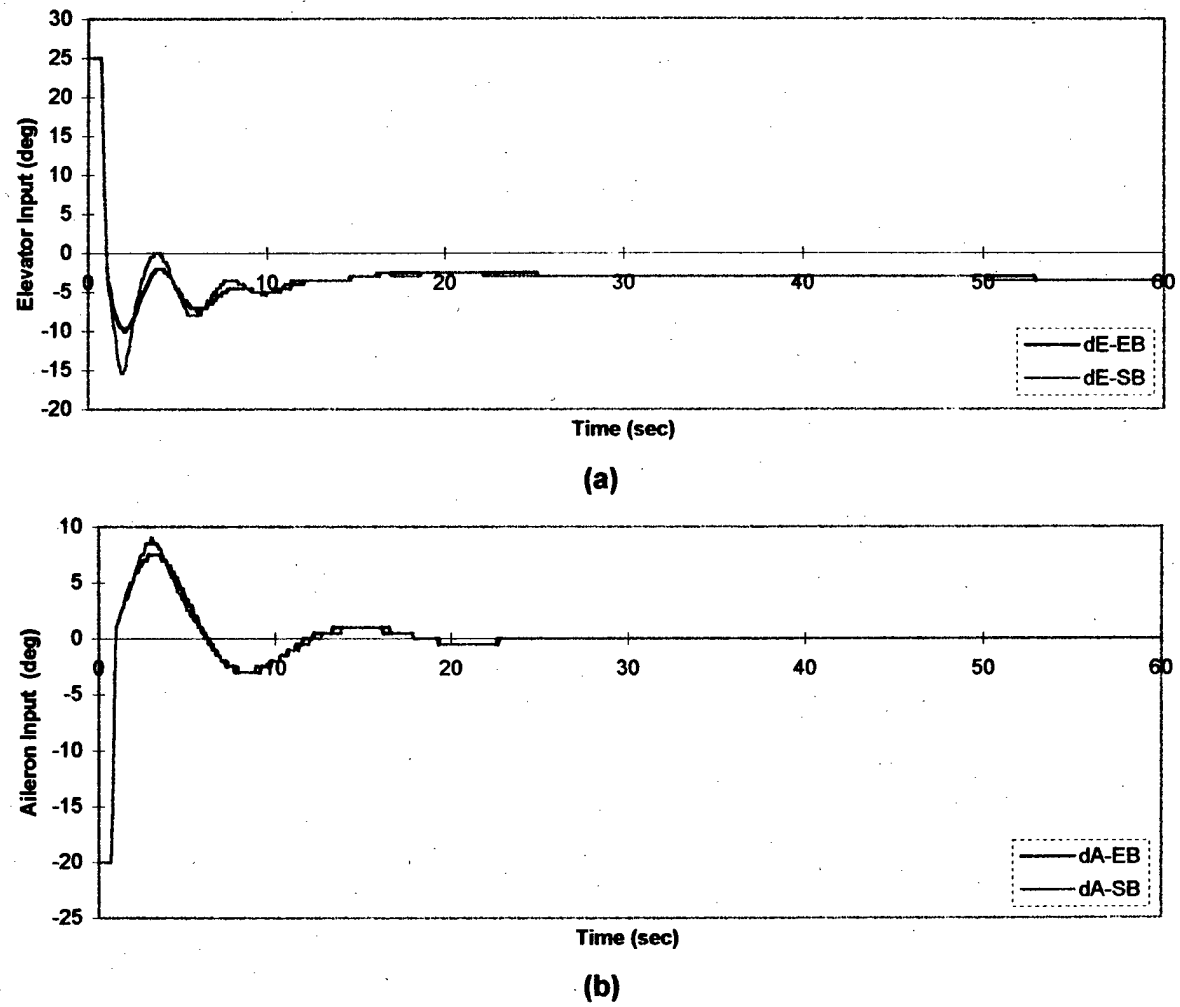


Figure 6.38: NN control inputs in response to p and q excitations above 2 deg/sec at HC.

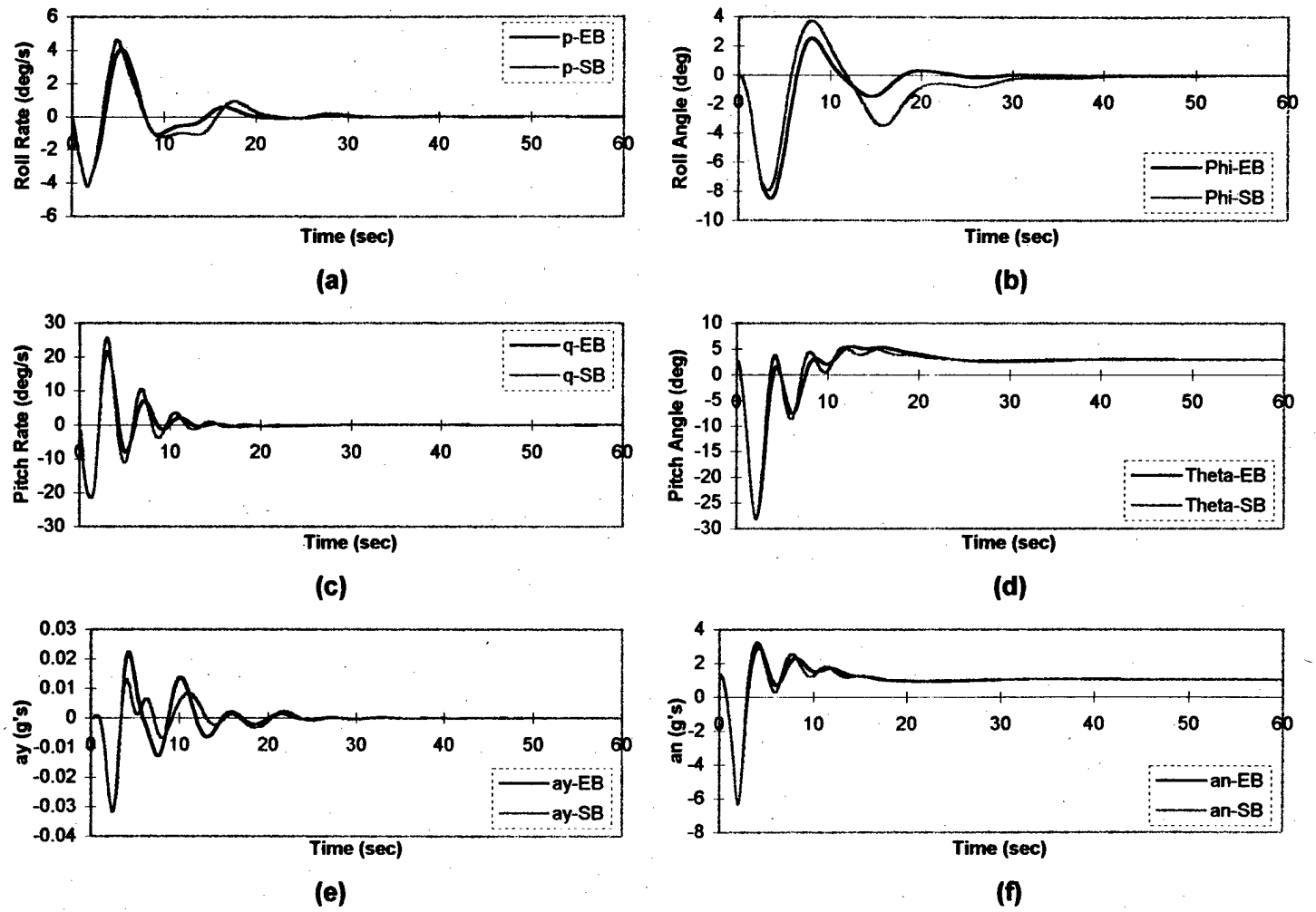
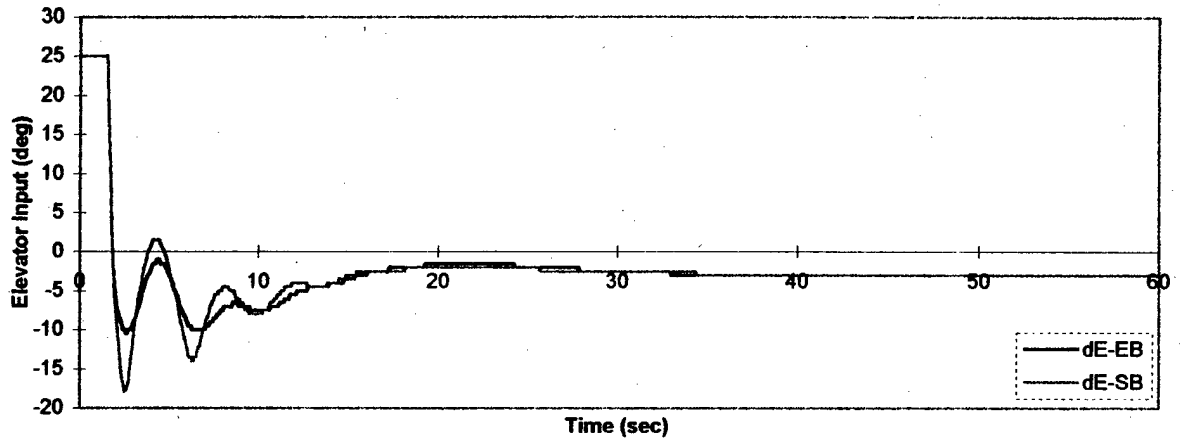
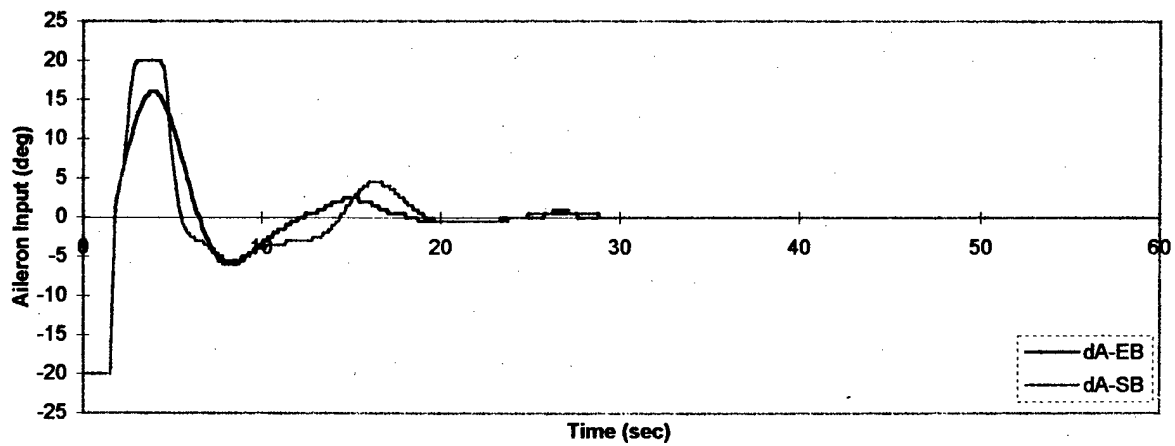


Figure 6.39: Aircraft response to non-linear conditions when p and q are above 4 deg/sec at HC.



(a)



(b)

Figure 6.40: NN control inputs in response to p and q excitations above 4 deg/sec at HC.

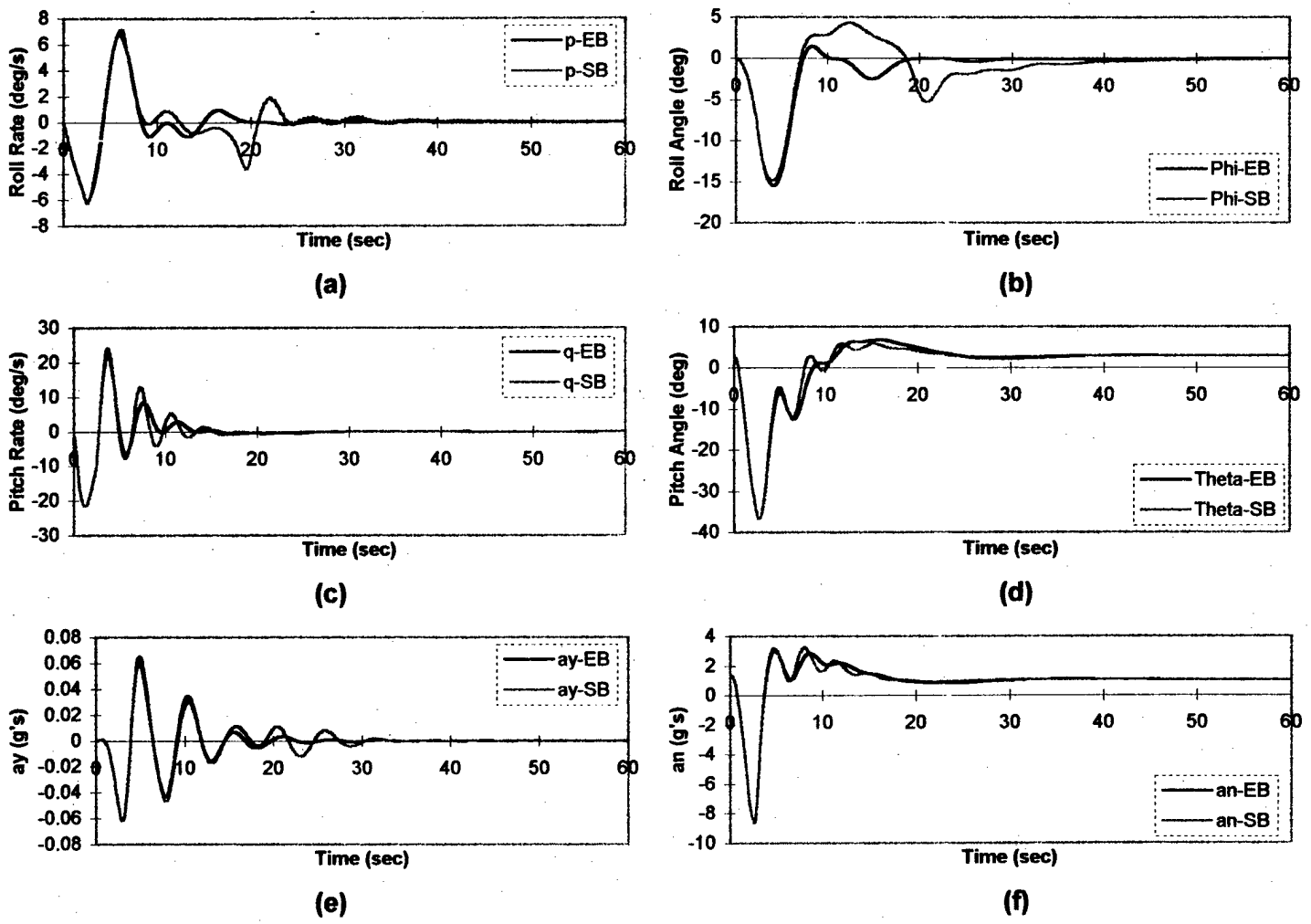


Figure 6.41: Aircraft response to non-linear conditions when p and q are above 6 deg/sec at HC.

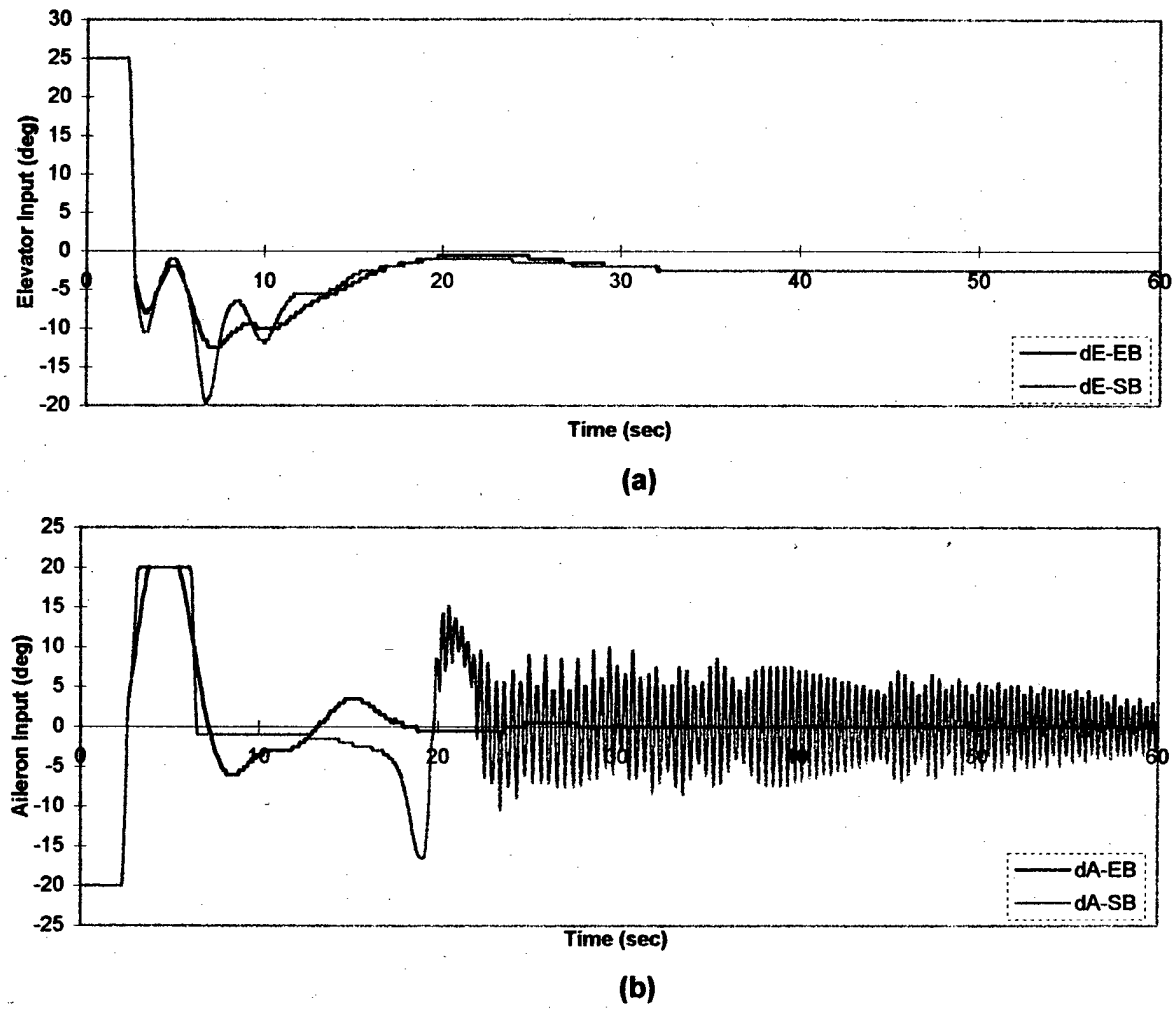


Figure 6.42: NN control inputs in response to p and q excitations above 6 deg/sec at HC.

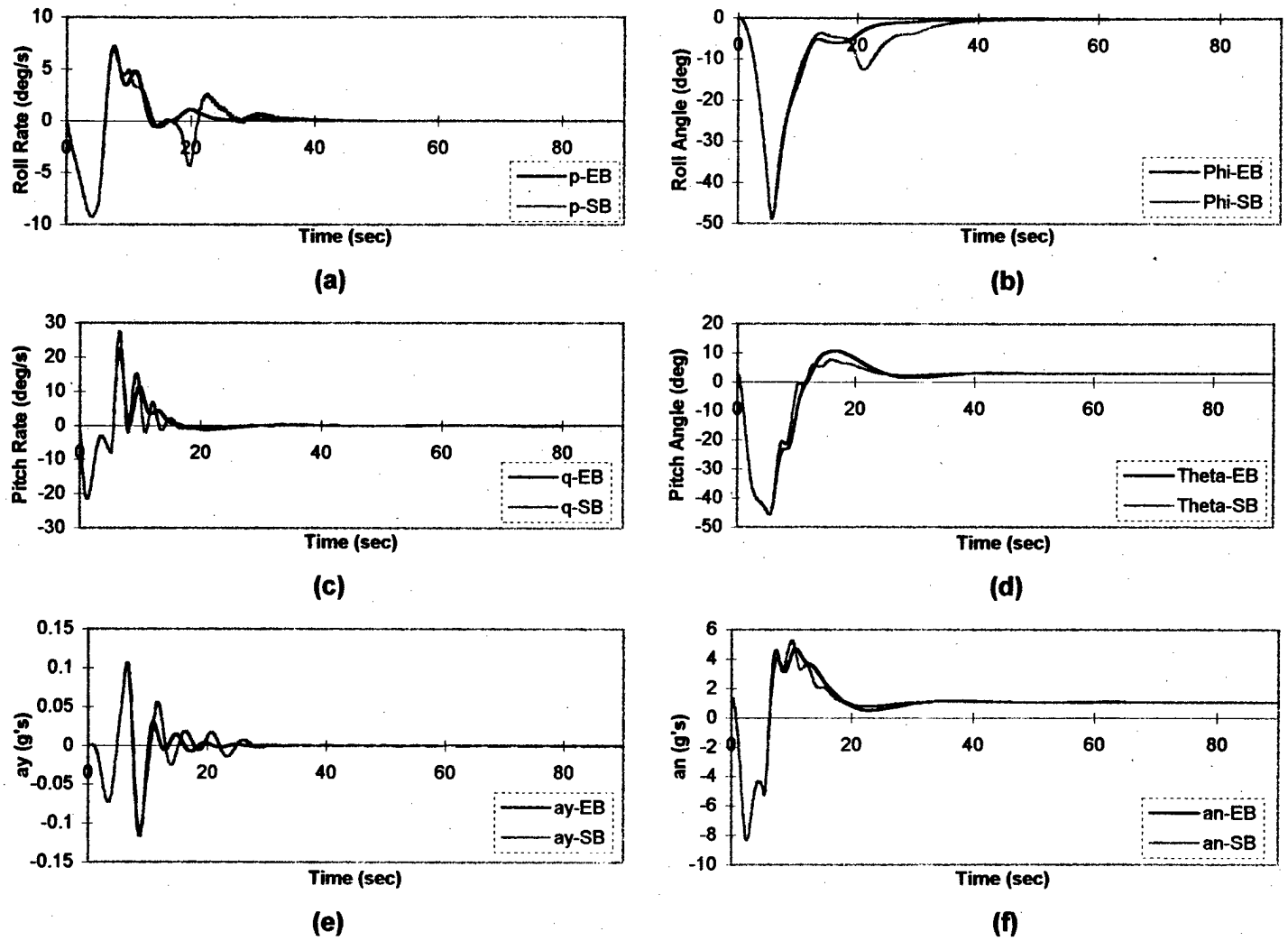
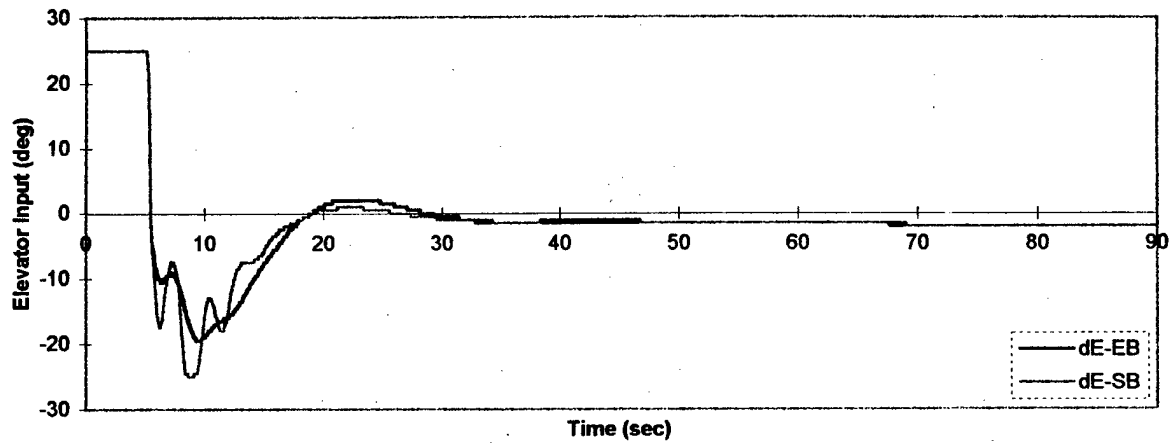
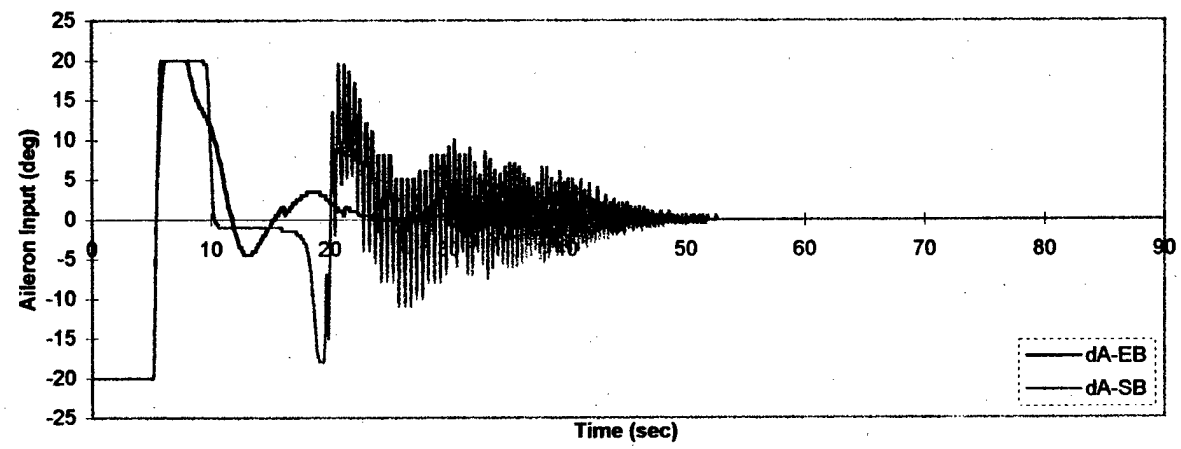


Figure 6.43: Aircraft response to non-linear conditions when p and q are above 8 deg/sec at HC.



(a)



(b)

Figure 6.44: NN control inputs in response to p and q excitations above 8 deg/sec at HC.

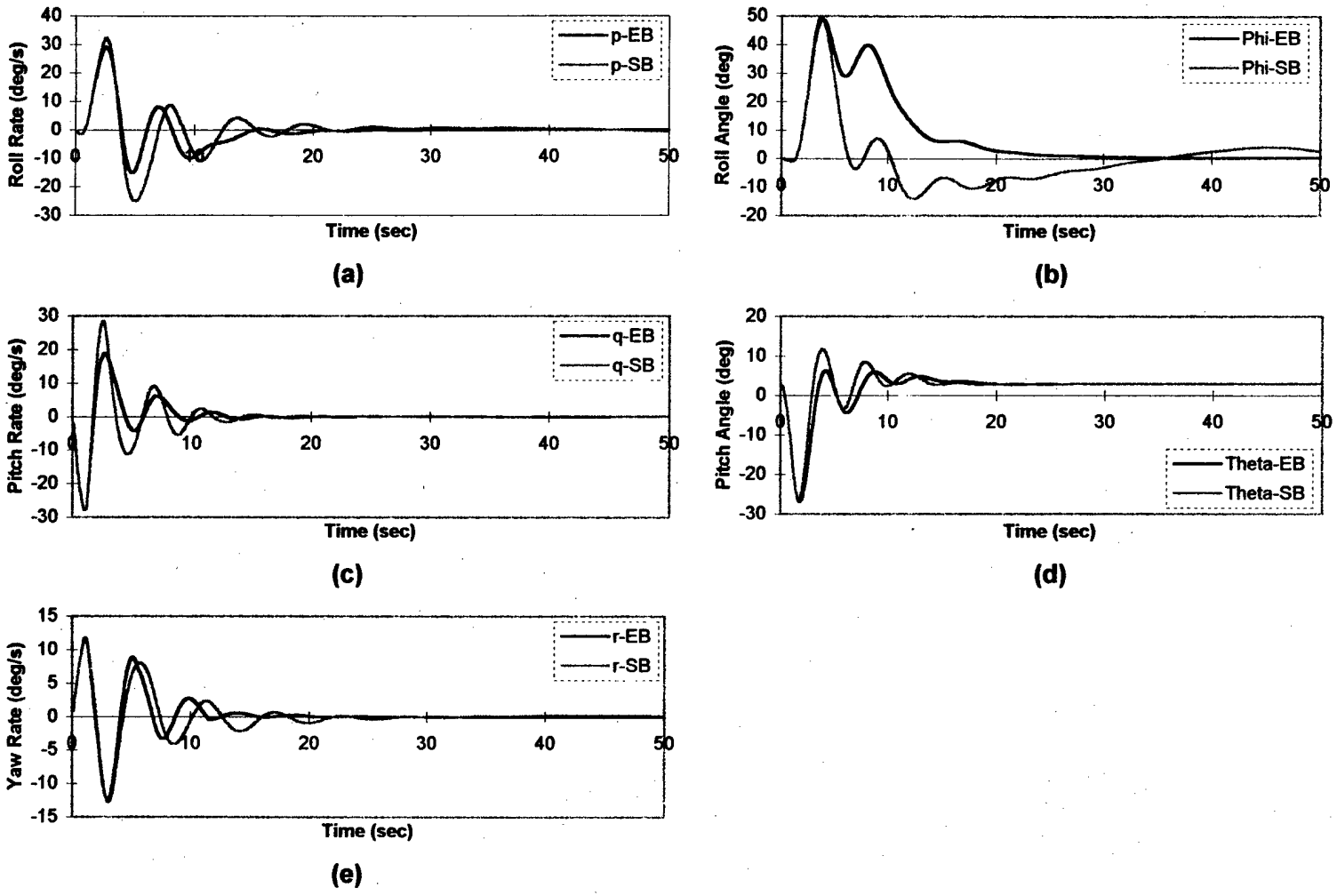
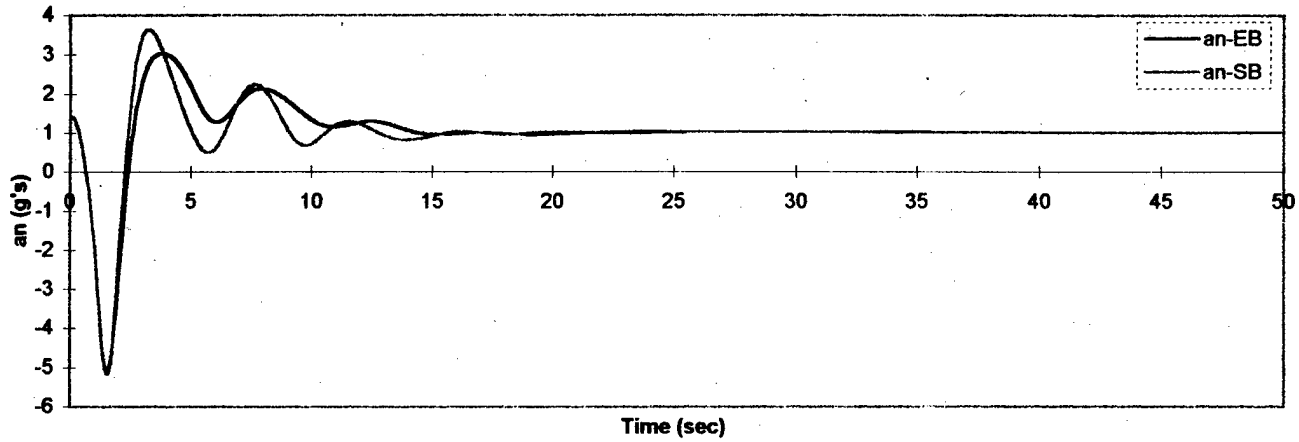
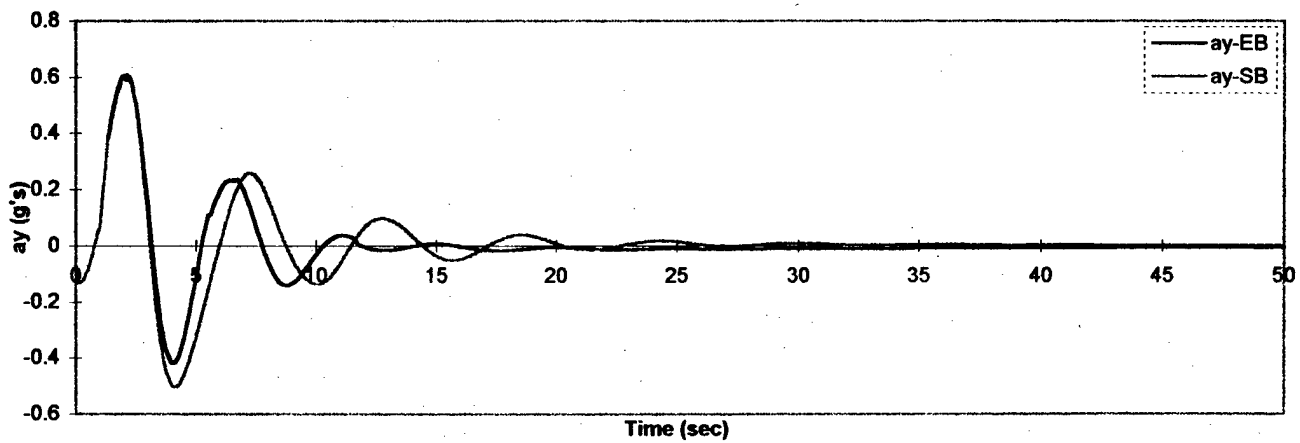


Figure 6.45: Aircraft response to non-linear conditions when p, q, and r are above 2 deg/sec at LC.



(a)



(b)

Figure 6.46: Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 2 deg/sec at LC.

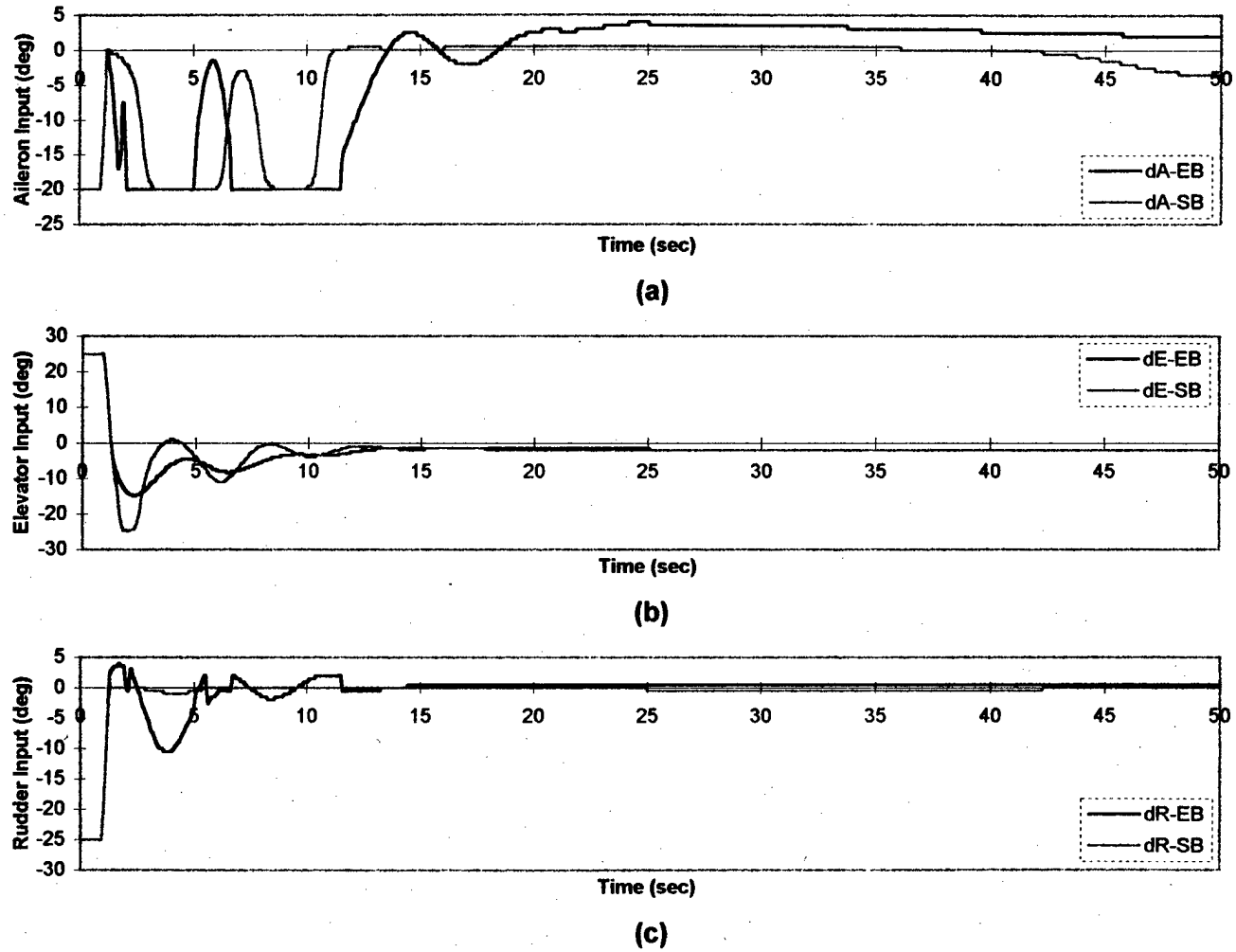


Figure 6.47: NN control inputs in response to p, q, and r excitations above 2 deg/sec at LC.

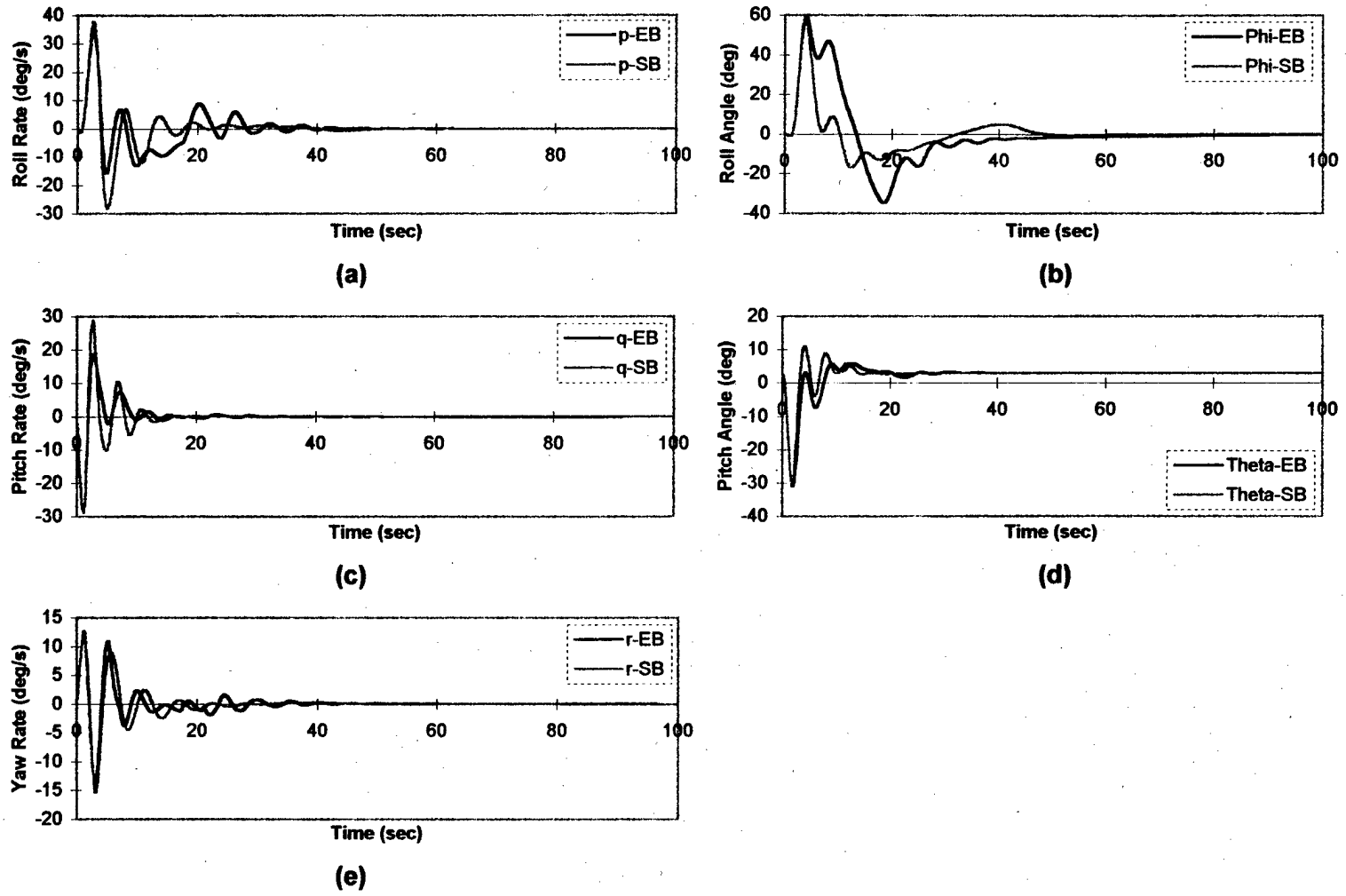
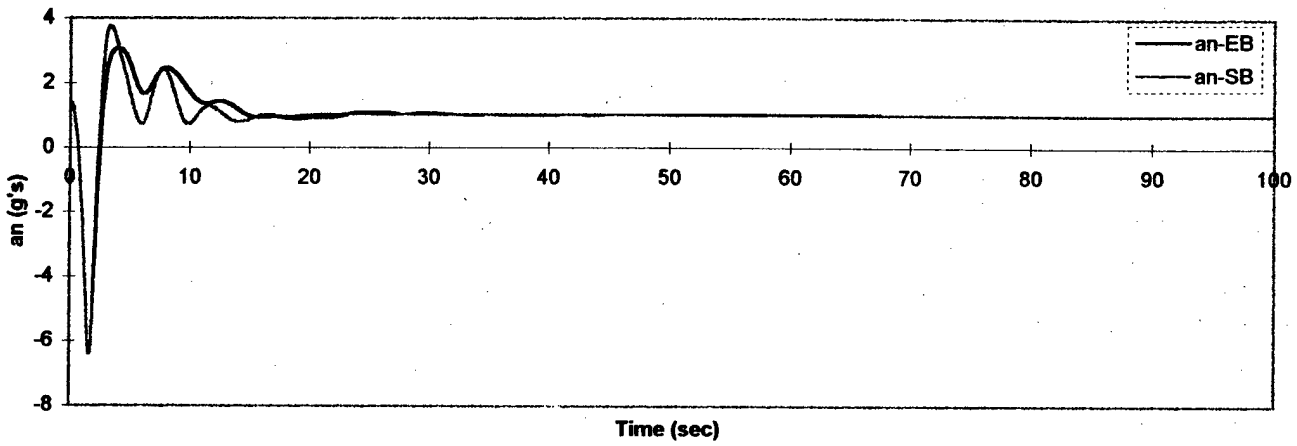
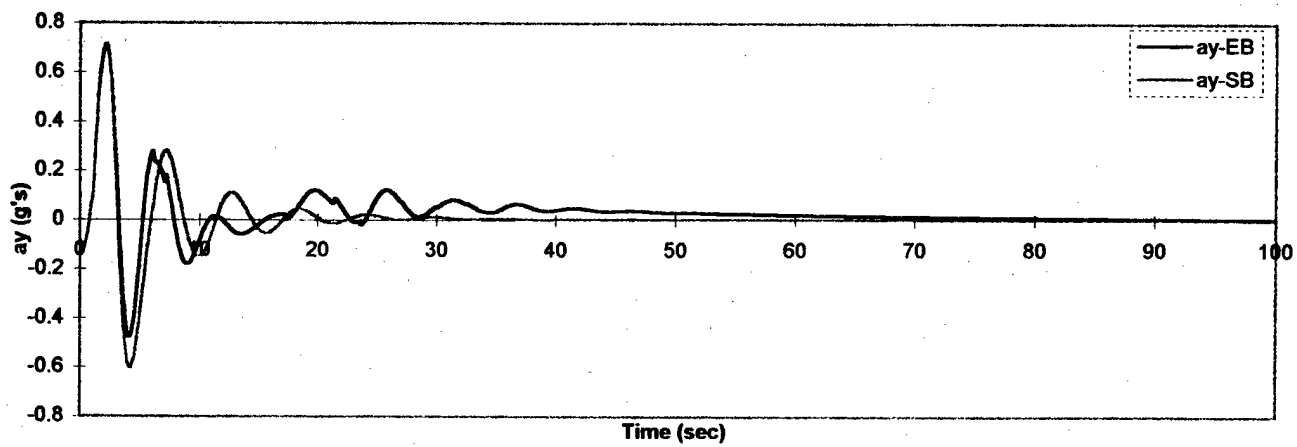


Figure 6.48: Aircraft response to non-linear conditions when p, q, and r are above 4 deg/sec at LC.



(a)



(b)

Figure 6.49: Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 4 deg/sec at LC.

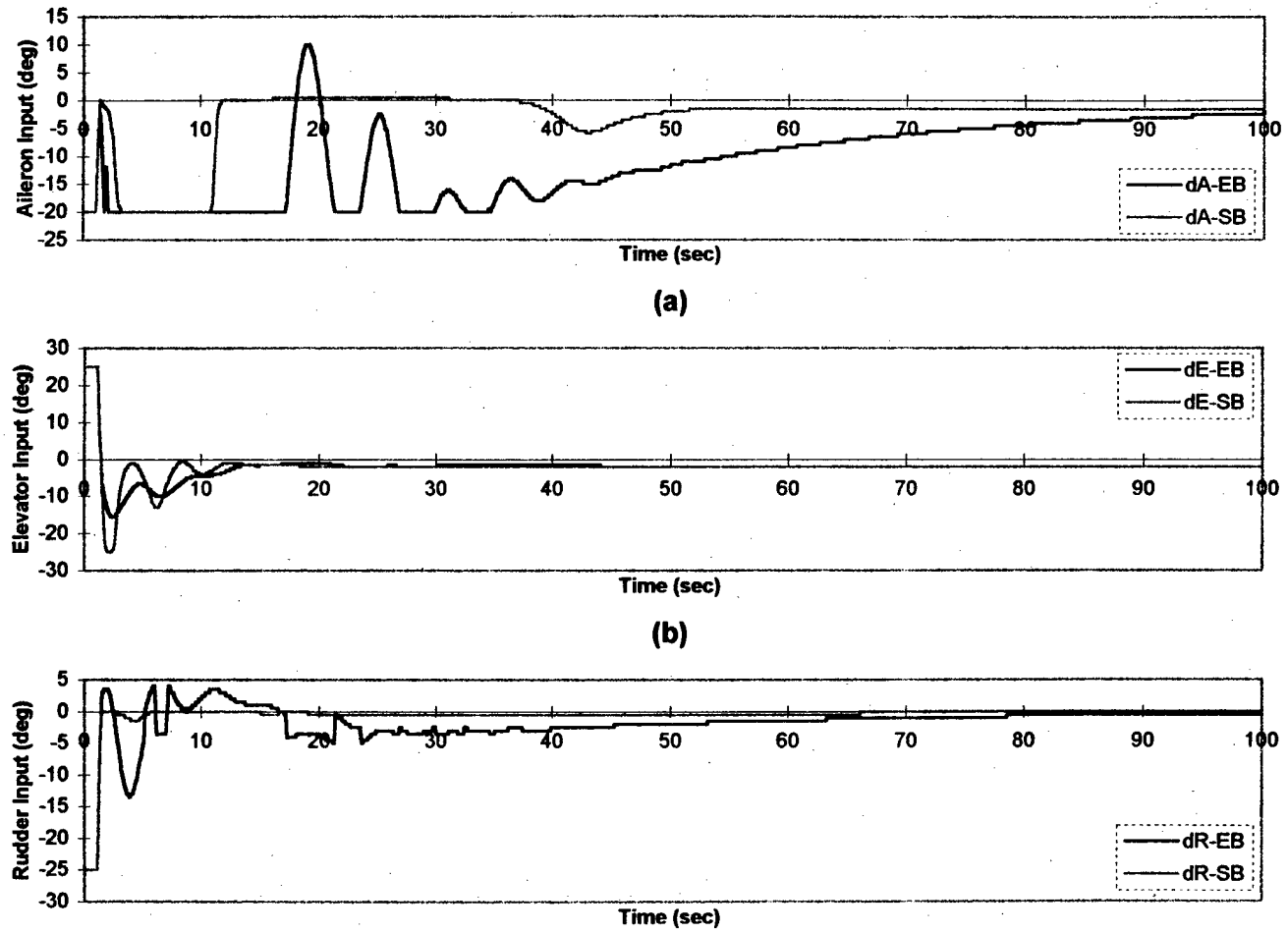


Figure 6.50: NN control inputs in response to p, q, and r excitations above 4 deg/sec at LC.

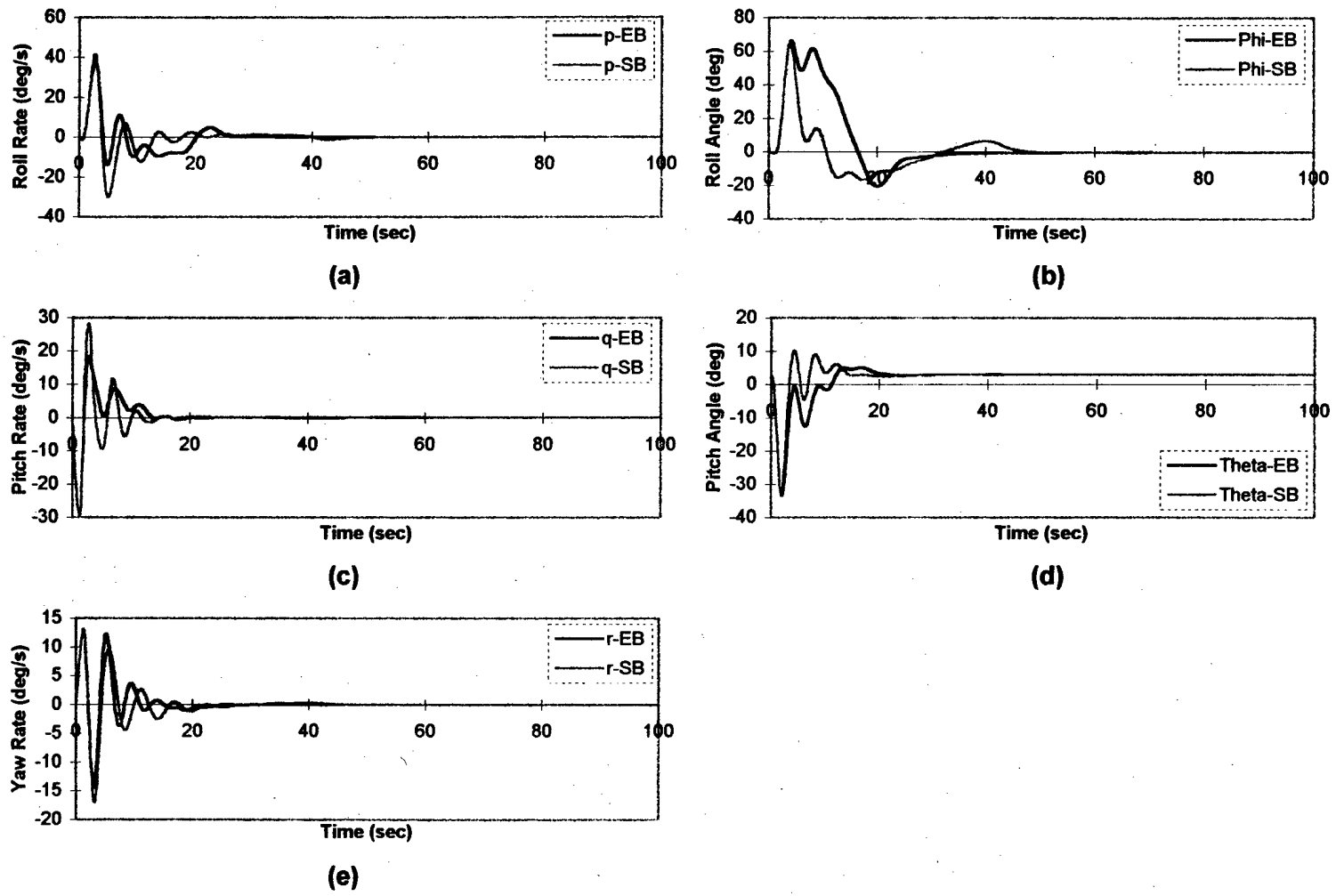


Figure 6.51: Aircraft response to non-linear conditions when p, q, and r are above 6 deg/sec at LC.

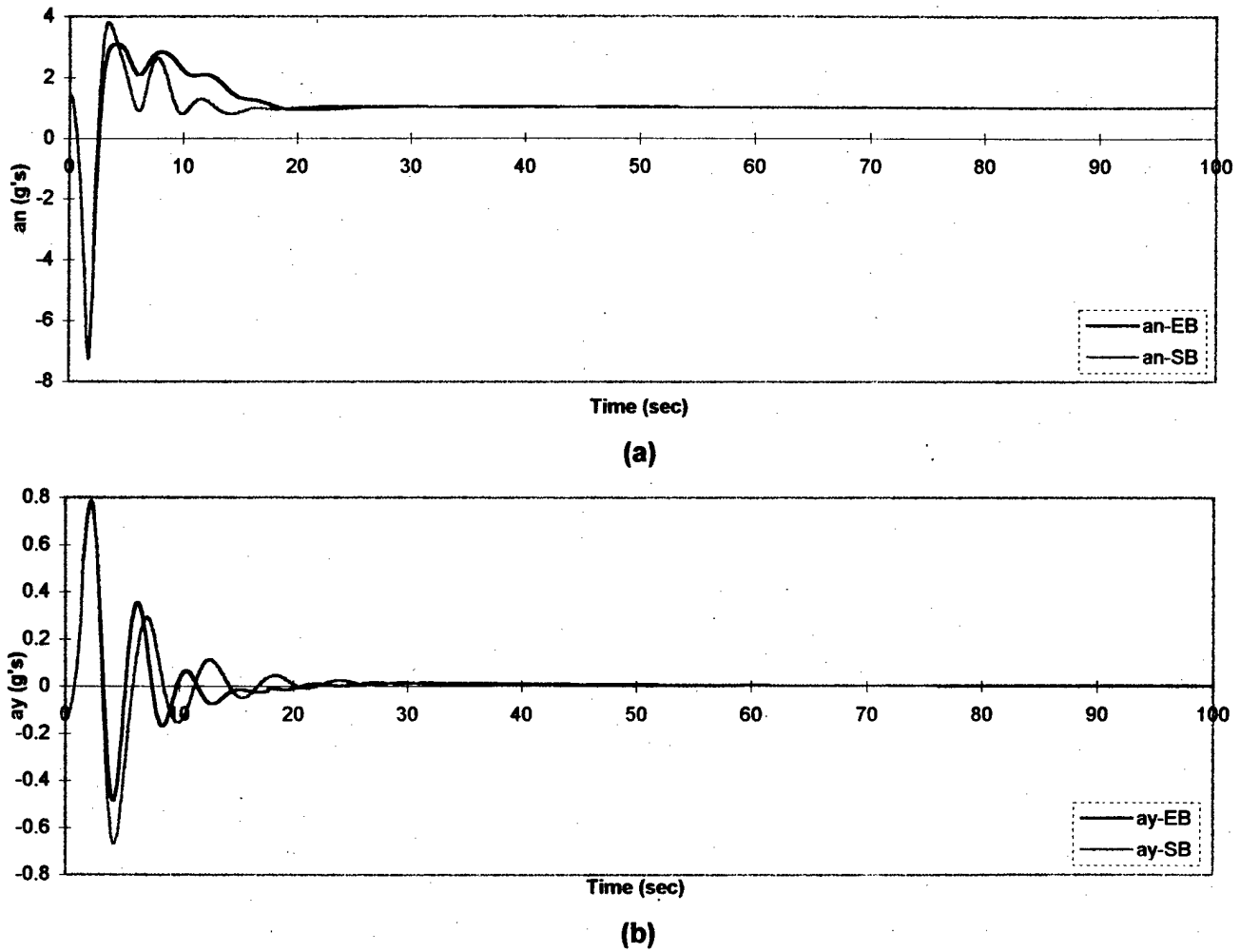


Figure 6.52: Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 6 deg/sec at LC.

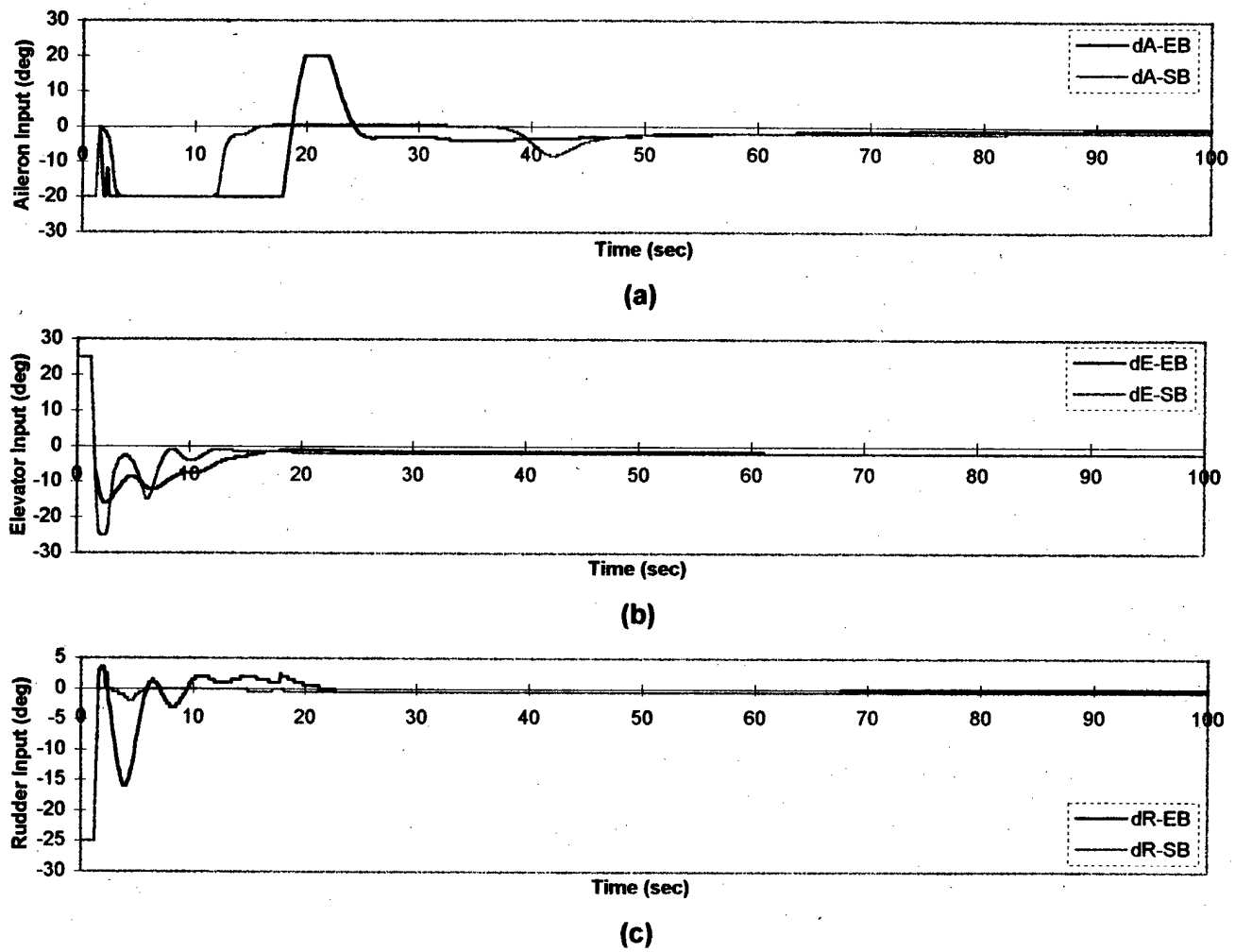


Figure 6.53: NN control inputs in response to p, q, and r excitations above 6 deg/sec at LC.

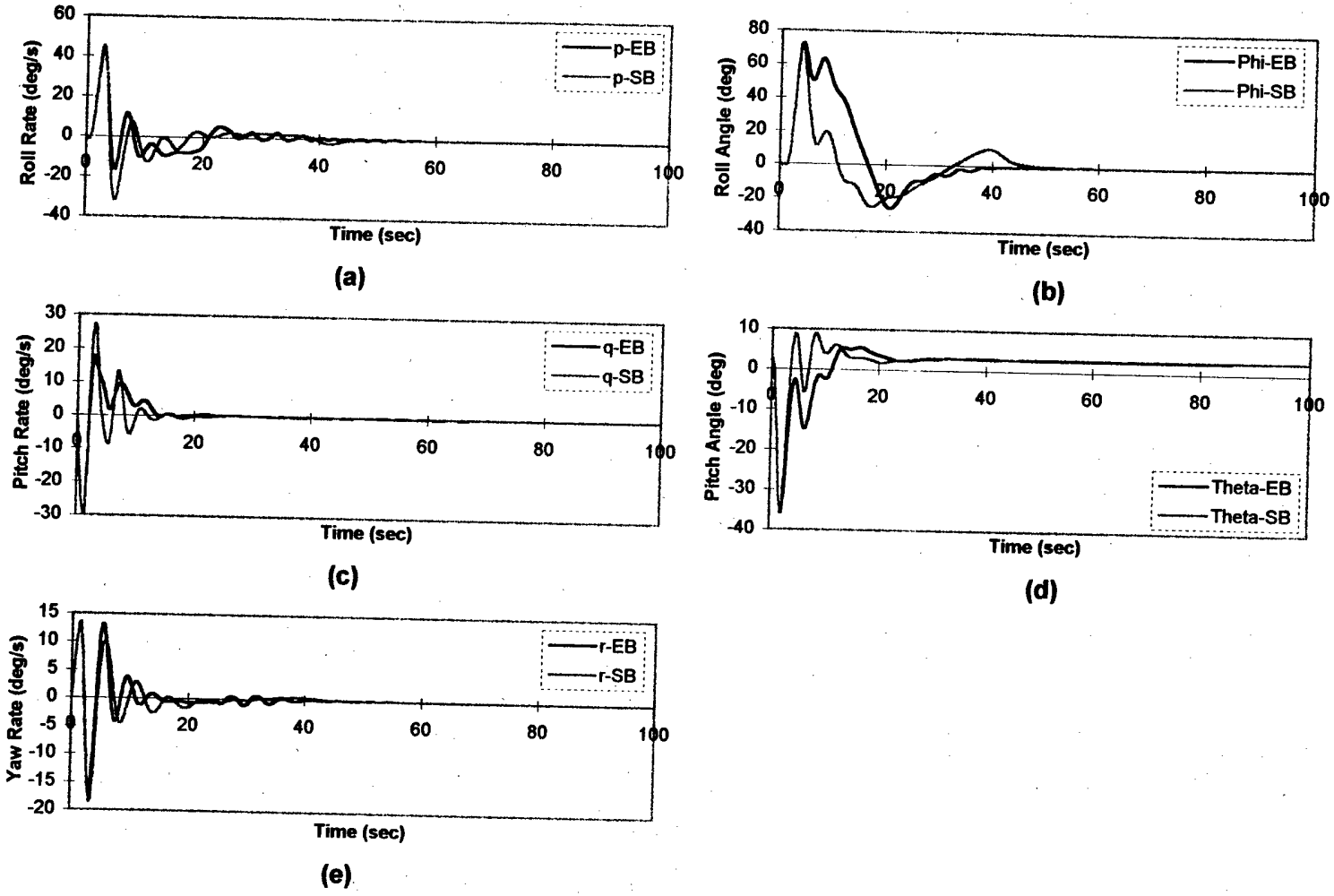


Figure 6.54: Aircraft response to non-linear conditions when p, q, and r are above 8 deg/sec at LC.

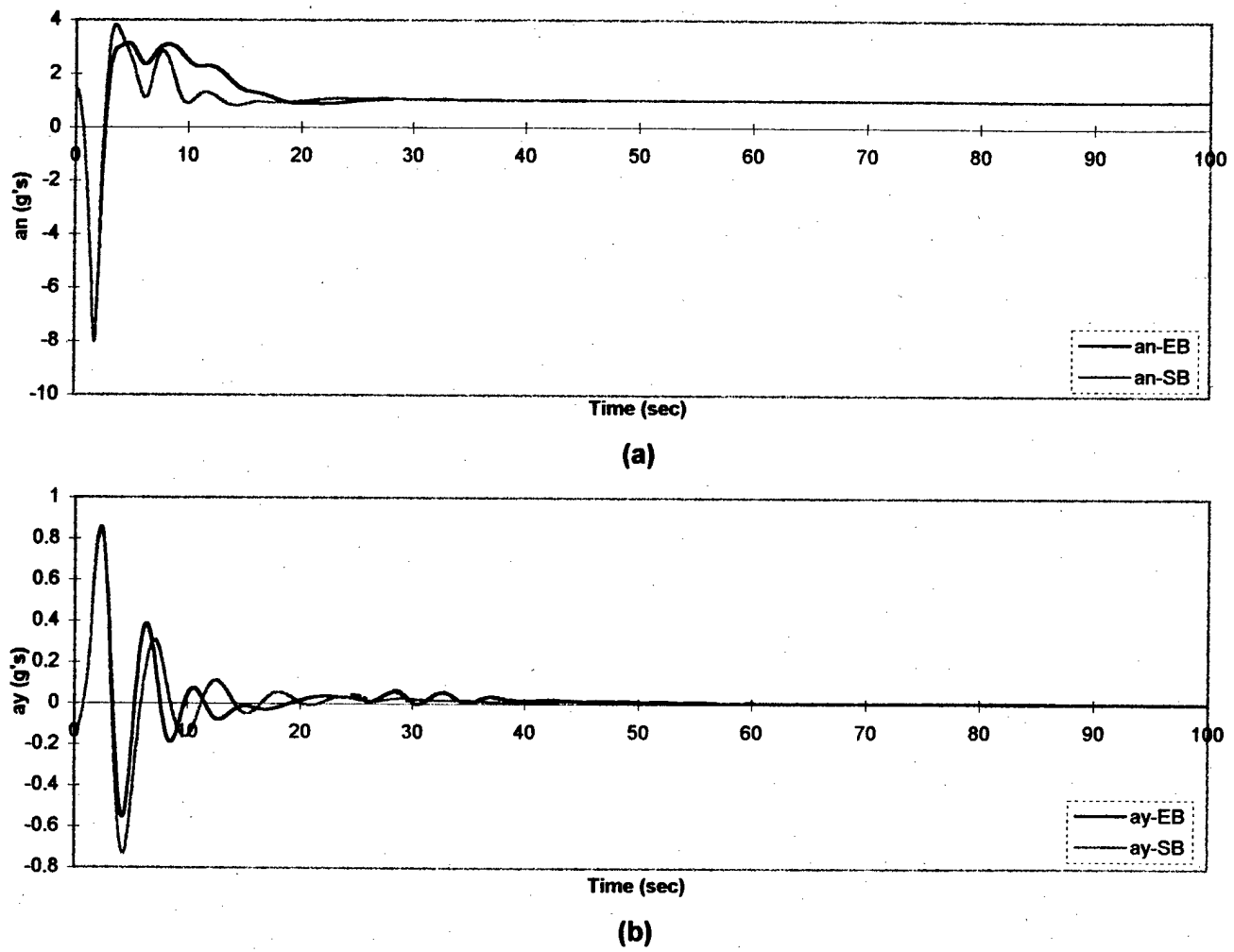


Figure 6.55: Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 8 deg/sec at LC.

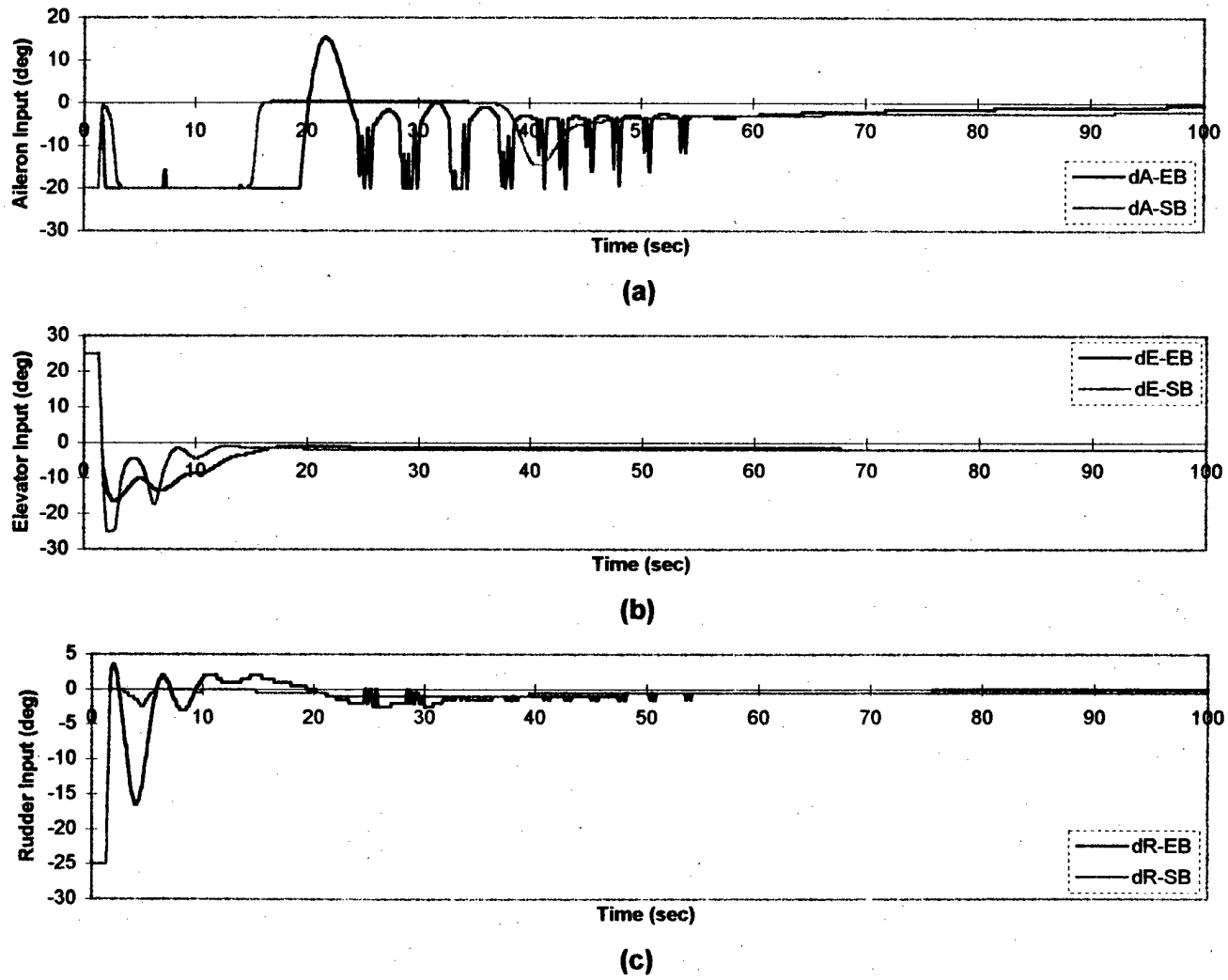


Figure 6.56: NN control inputs in response to p, q, and r excitations above 8 deg/sec at LC.

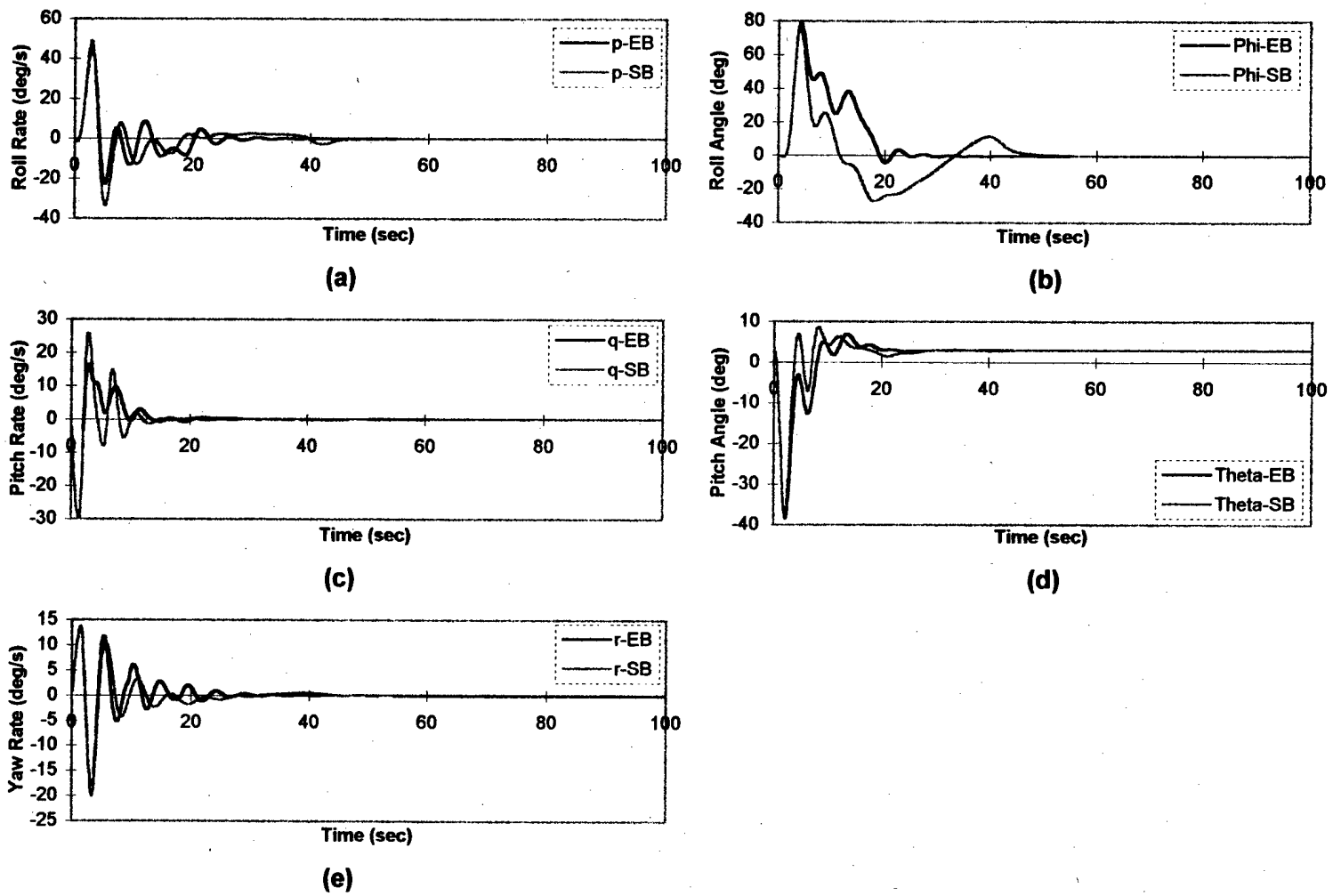
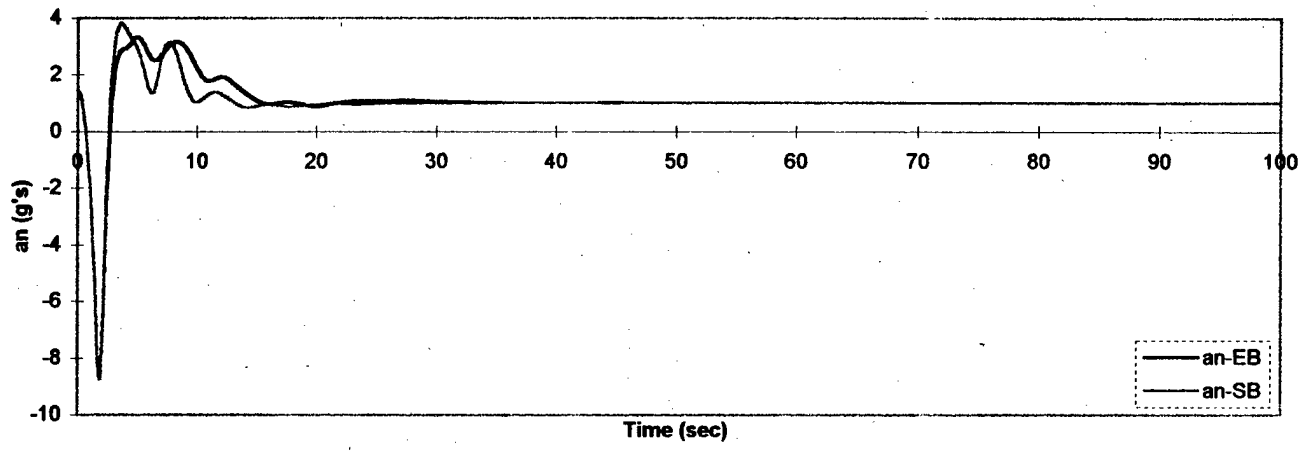
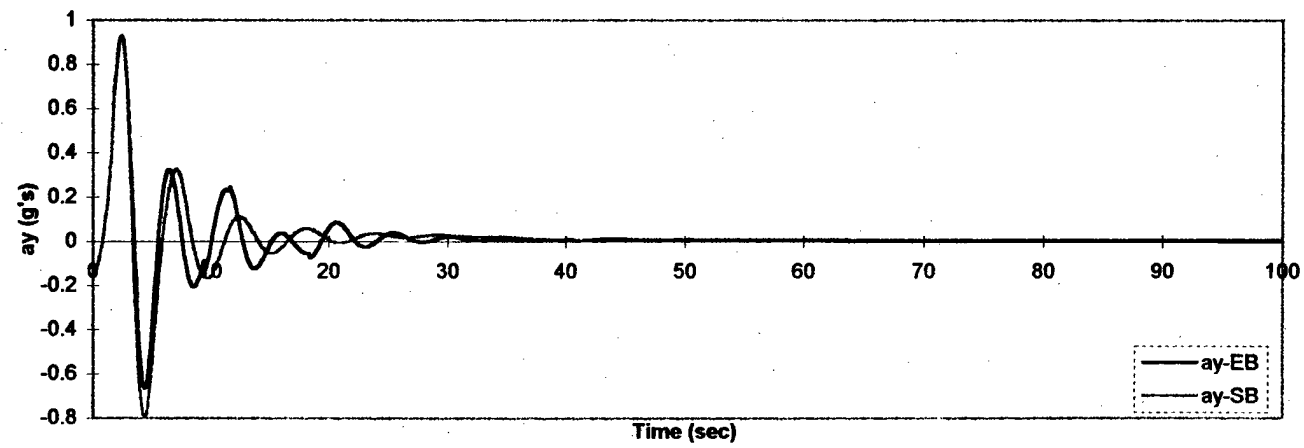


Figure 6.57: Aircraft response to non-linear conditions when p, q, and r are above 10 deg/sec at LC.



(a)



(b)

Figure 6.58: Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 10 deg/sec at LC.

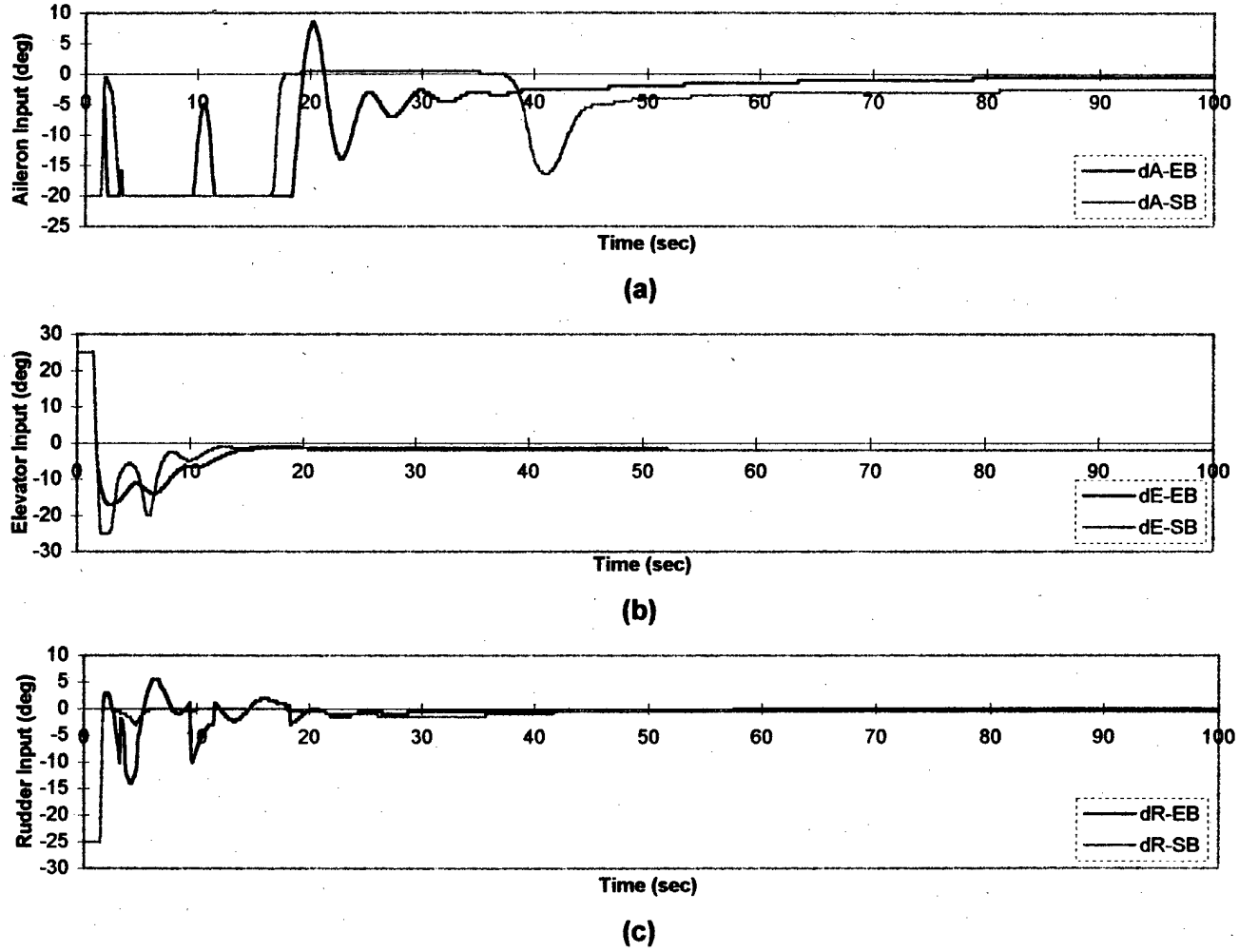


Figure 6.59: NN control inputs in response to p, q, and r excitations above 10 deg/sec at LC.

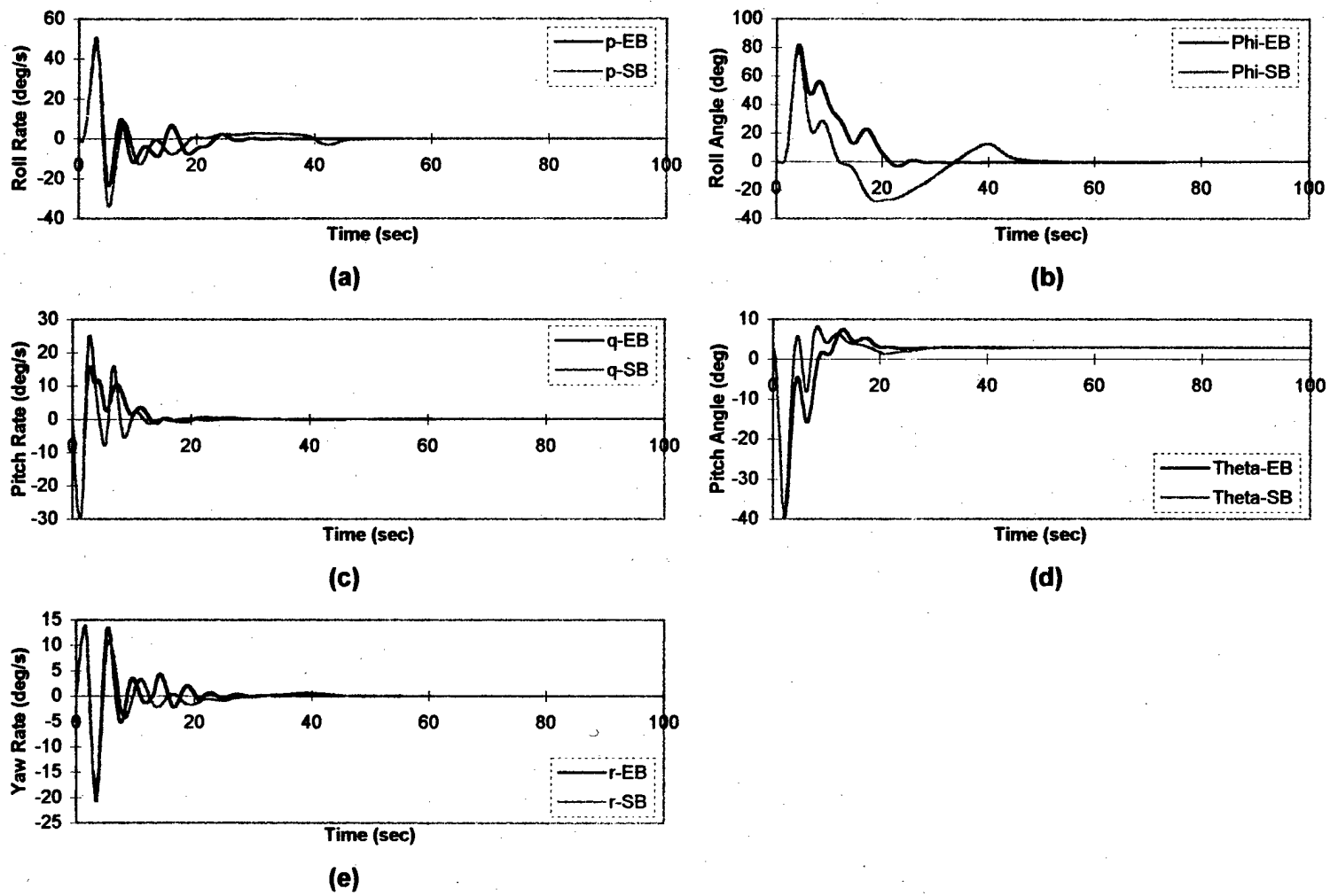
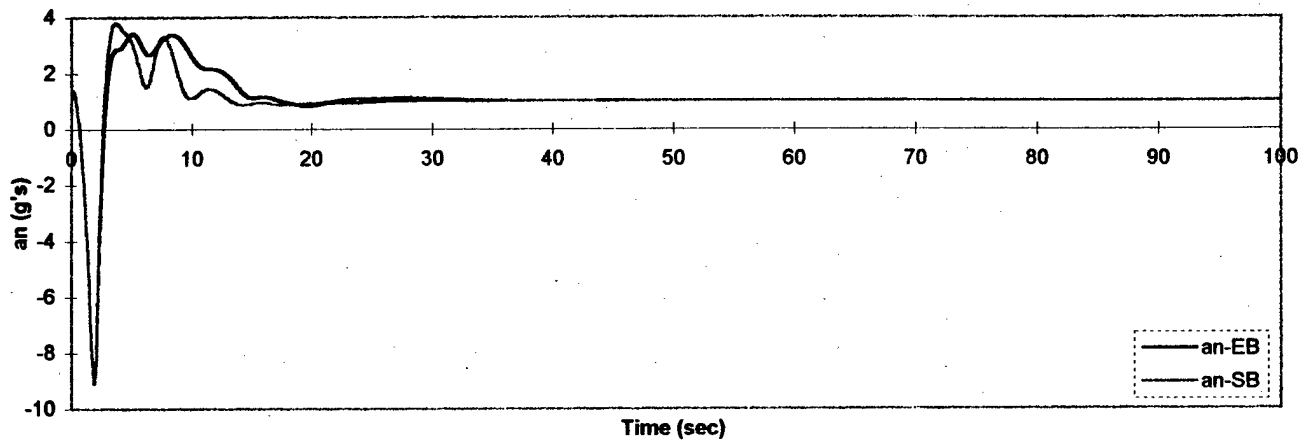
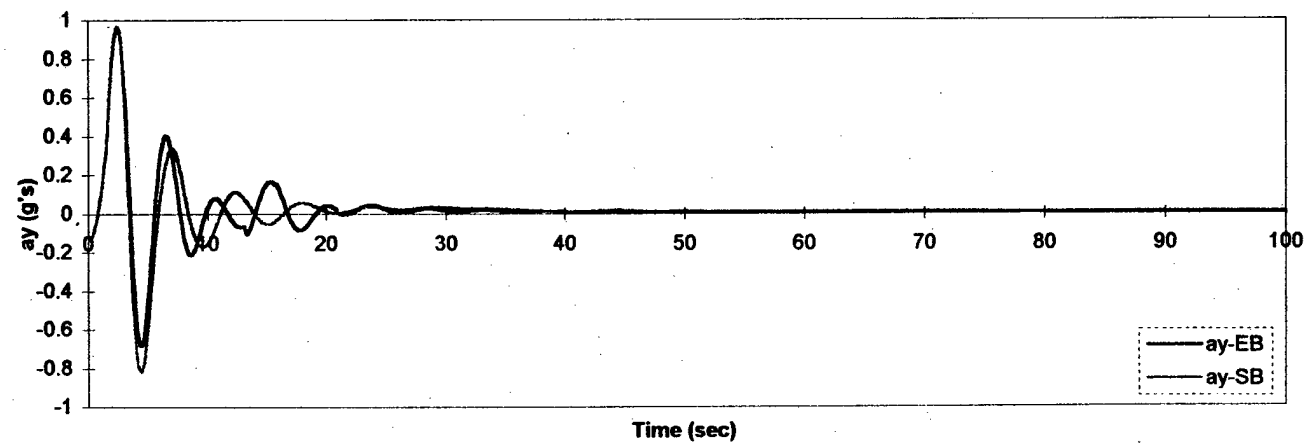


Figure 6.60: Aircraft response to non-linear conditions when p, q, and r are above 12 deg/sec at LC.



(a)



(b)

Figure 6.61: Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 12 deg/sec at LC.

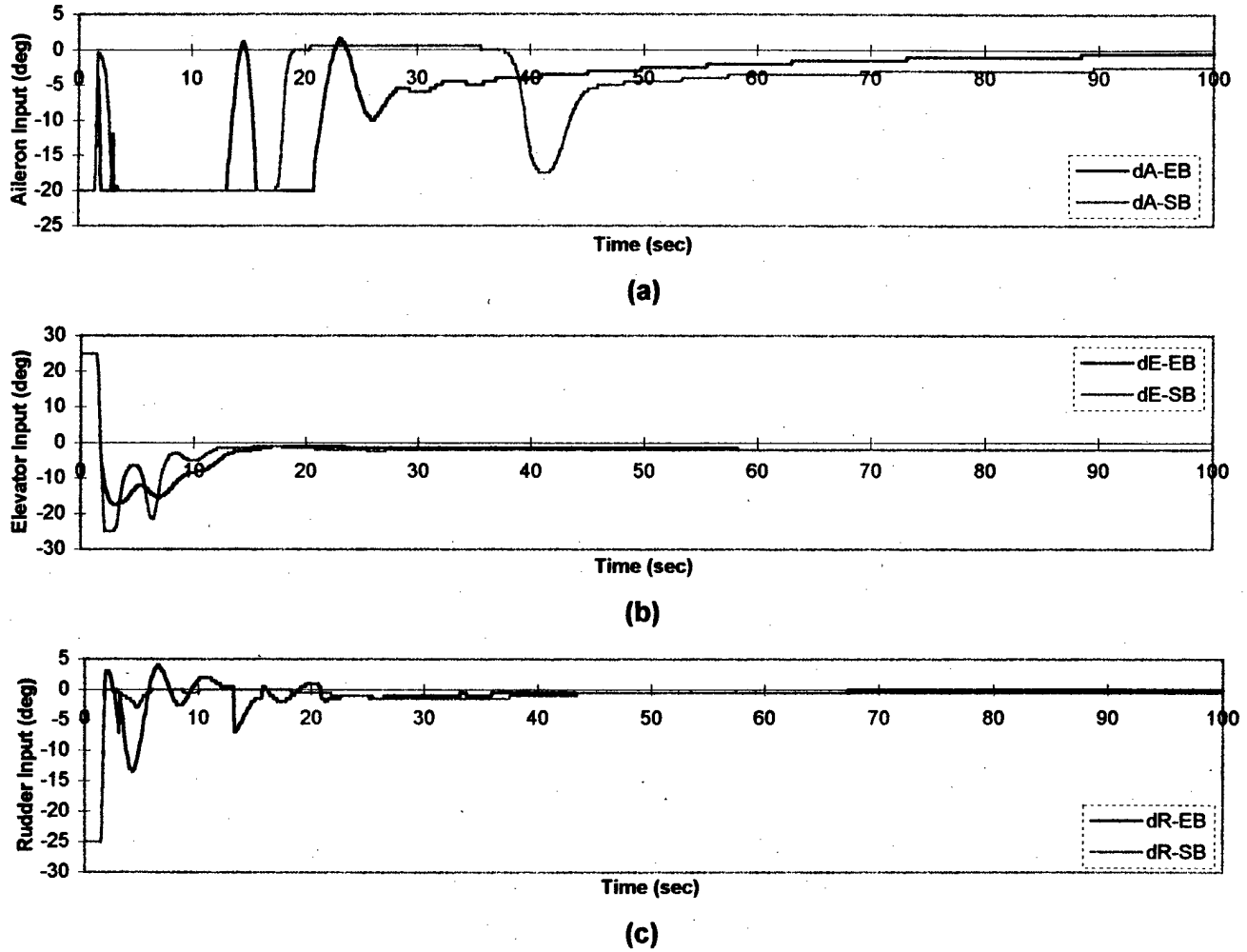


Figure 6.62: NN control inputs in response to p, q, and r excitations above 12 deg/sec at LC.

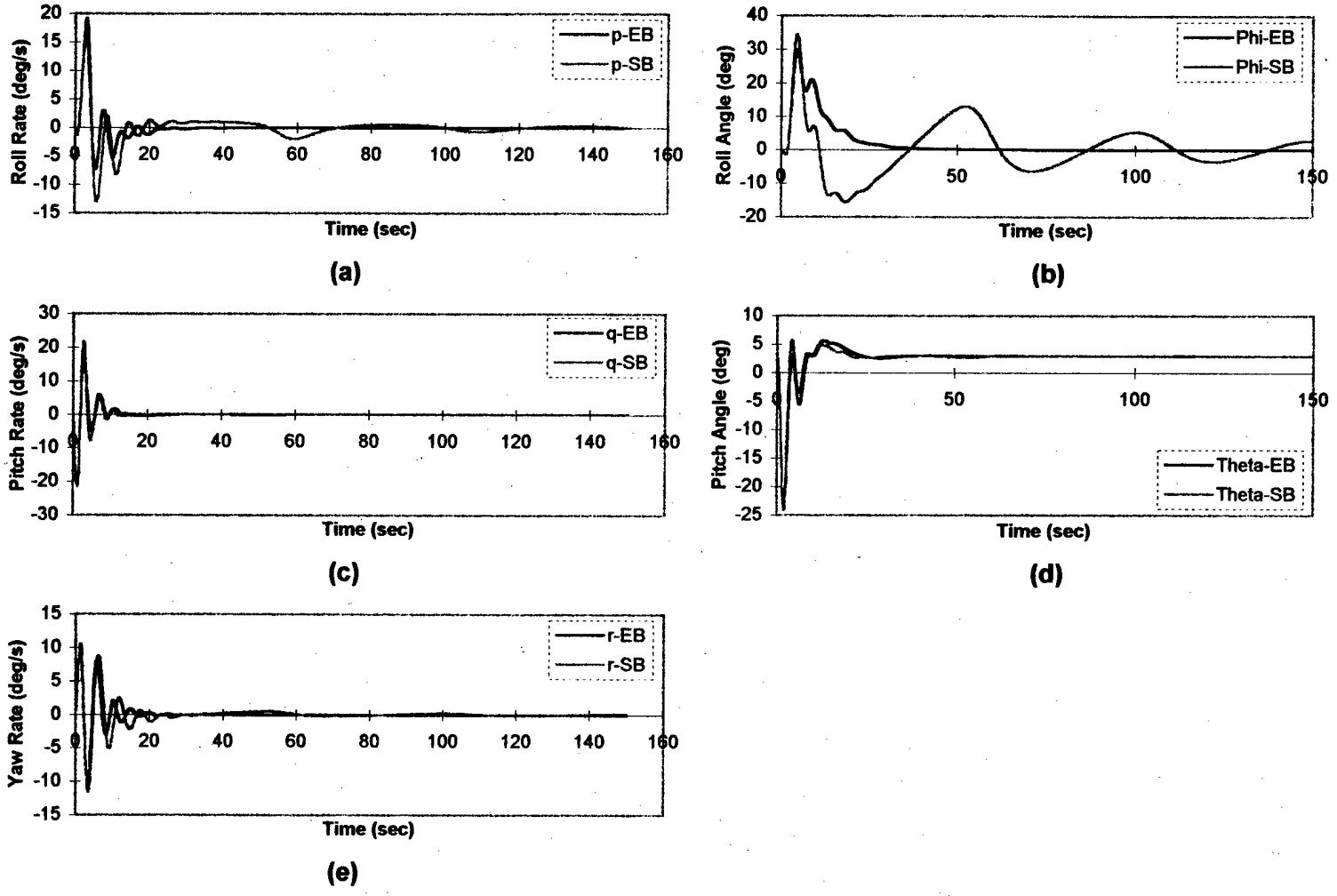
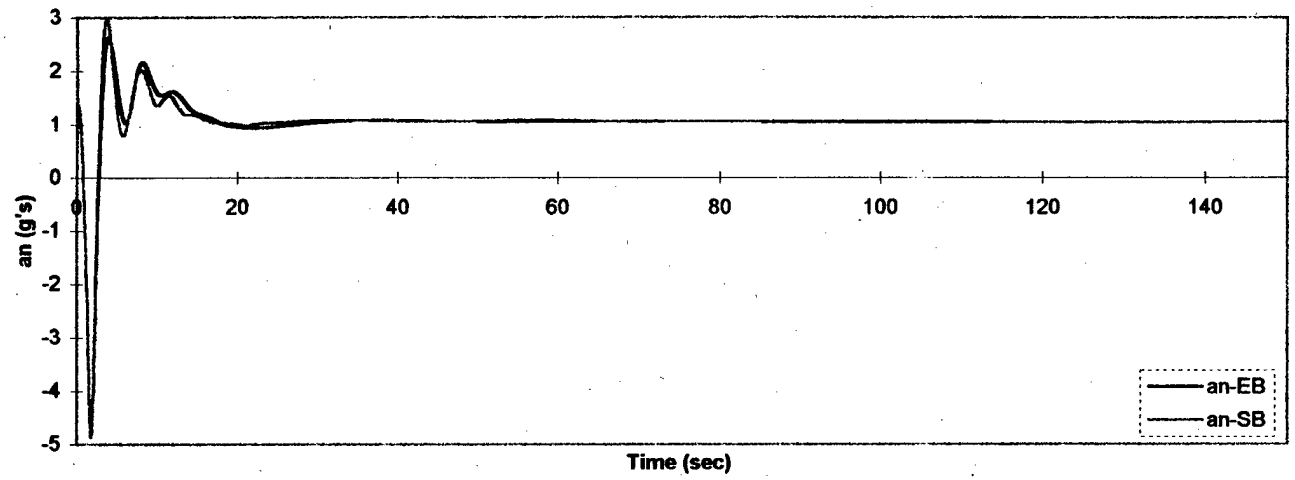
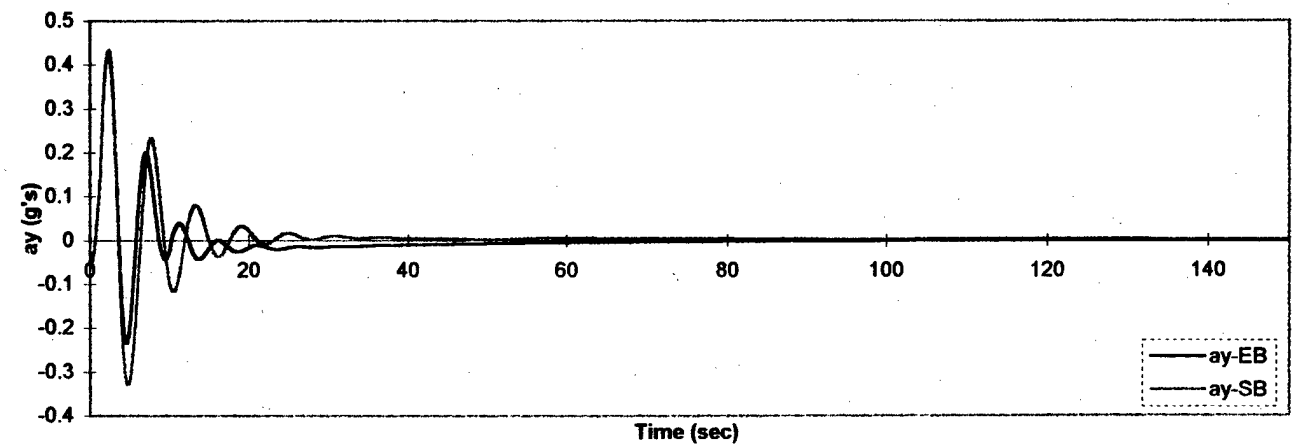


Figure 6.63: Aircraft response to non-linear conditions when p, q, and r are above 2 deg/sec at HC.



(a)



(b)

Figure 6.64: Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 2 deg/sec at HC.

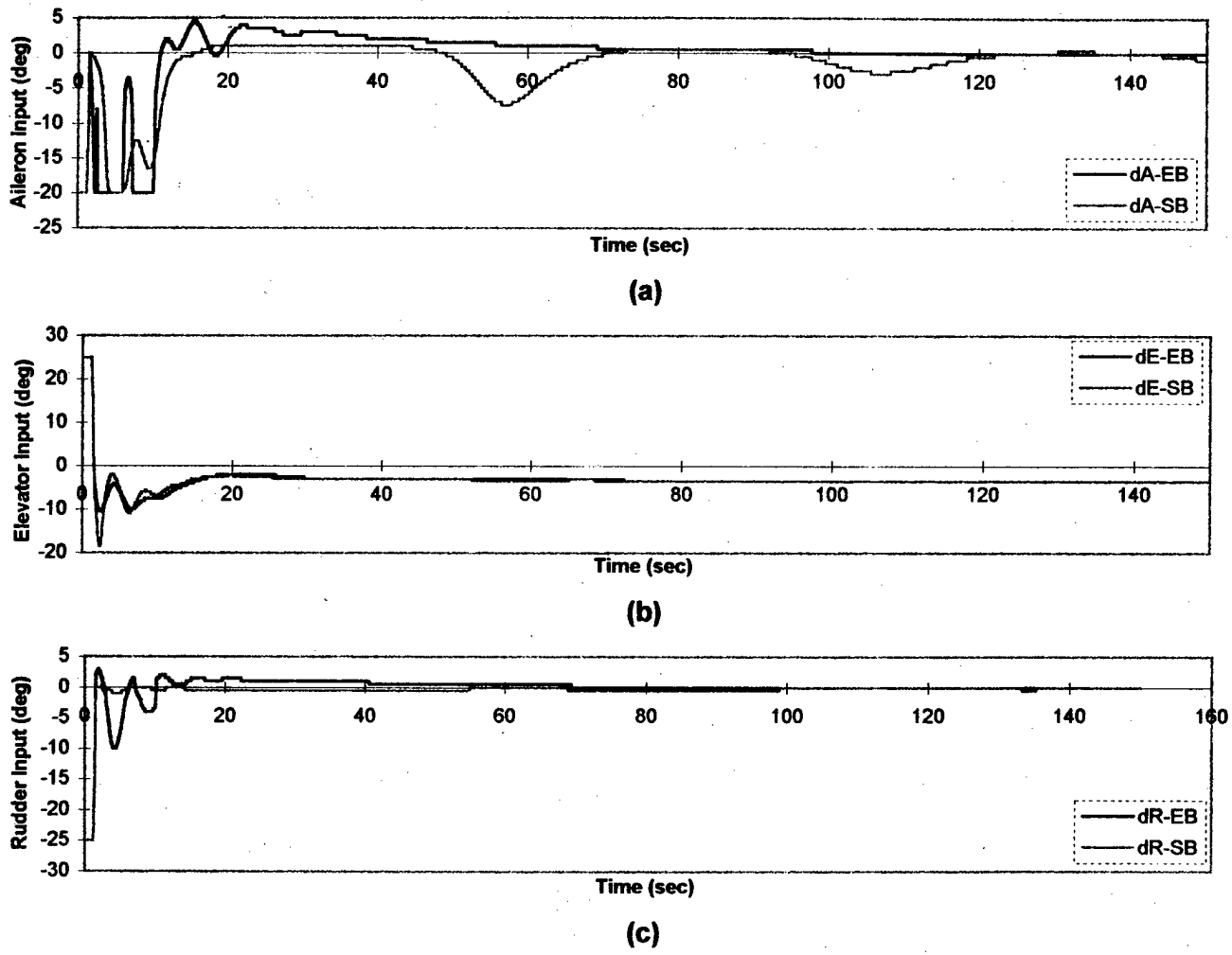


Figure 6.65: NN control inputs in response to p, q, and r excitations above 2 deg/sec at HC.

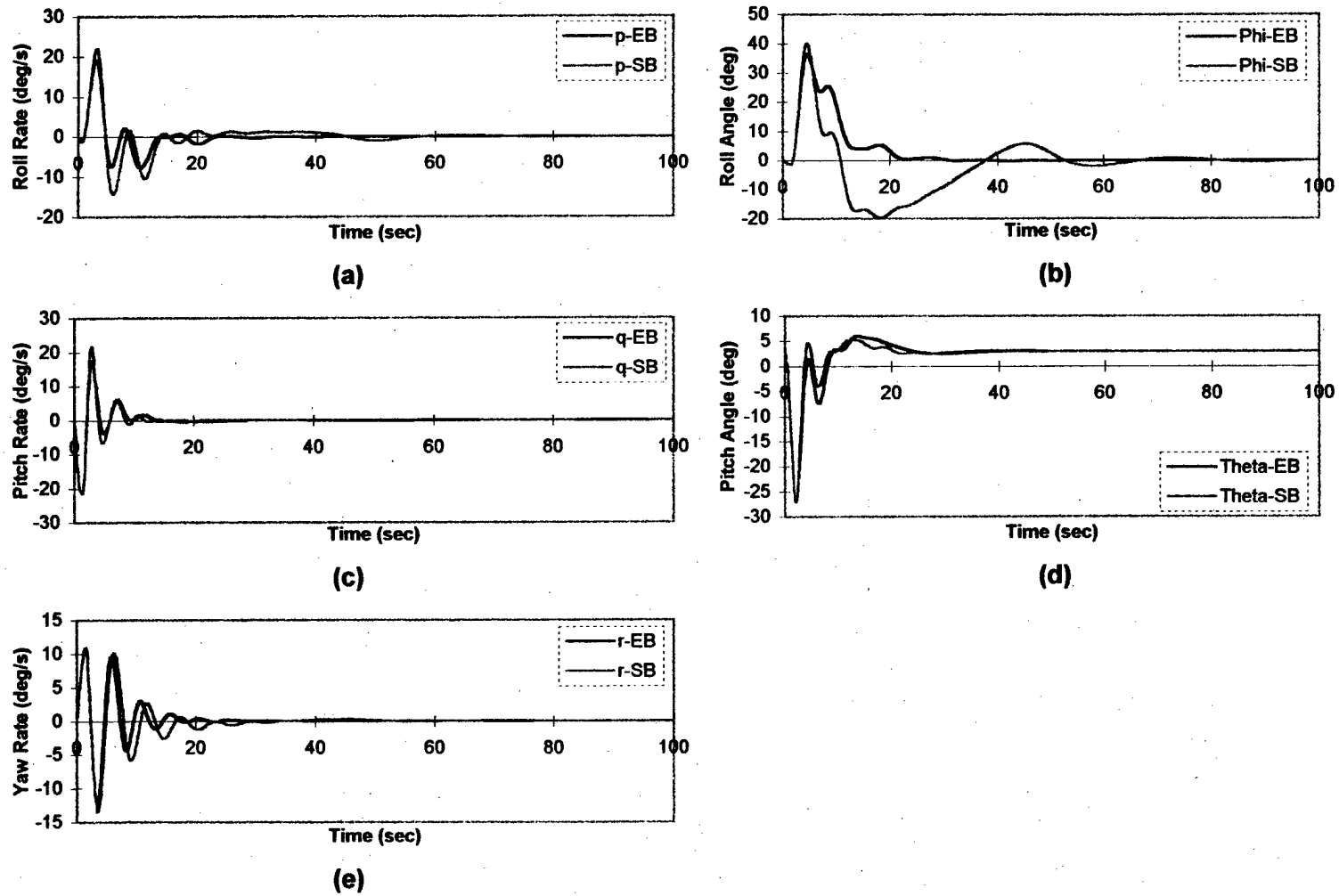


Figure 6.66: Aircraft response to non-linear conditions when p, q, and r are above 4 deg/sec at HC.

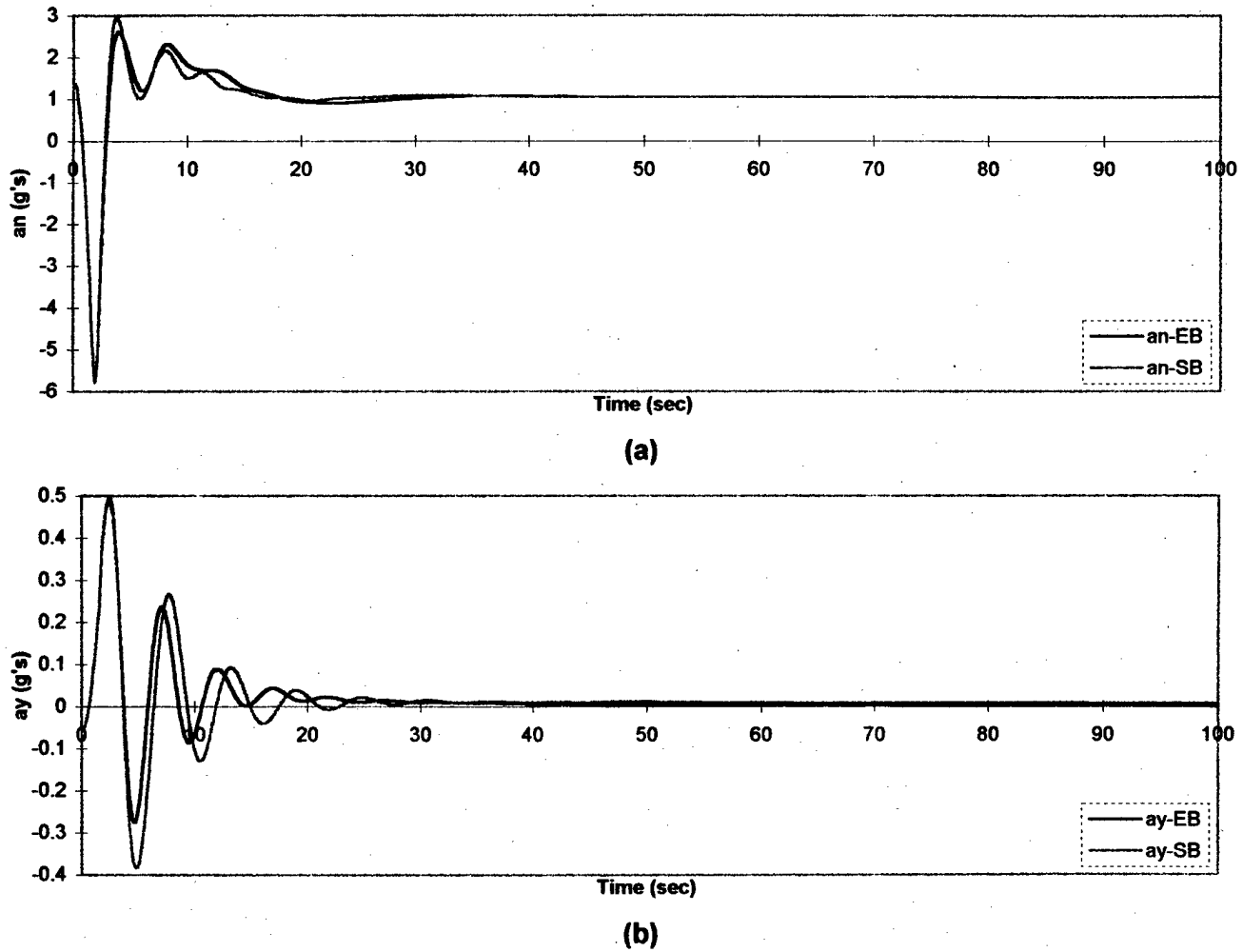


Figure 6.67: Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 4 deg/sec at HC.

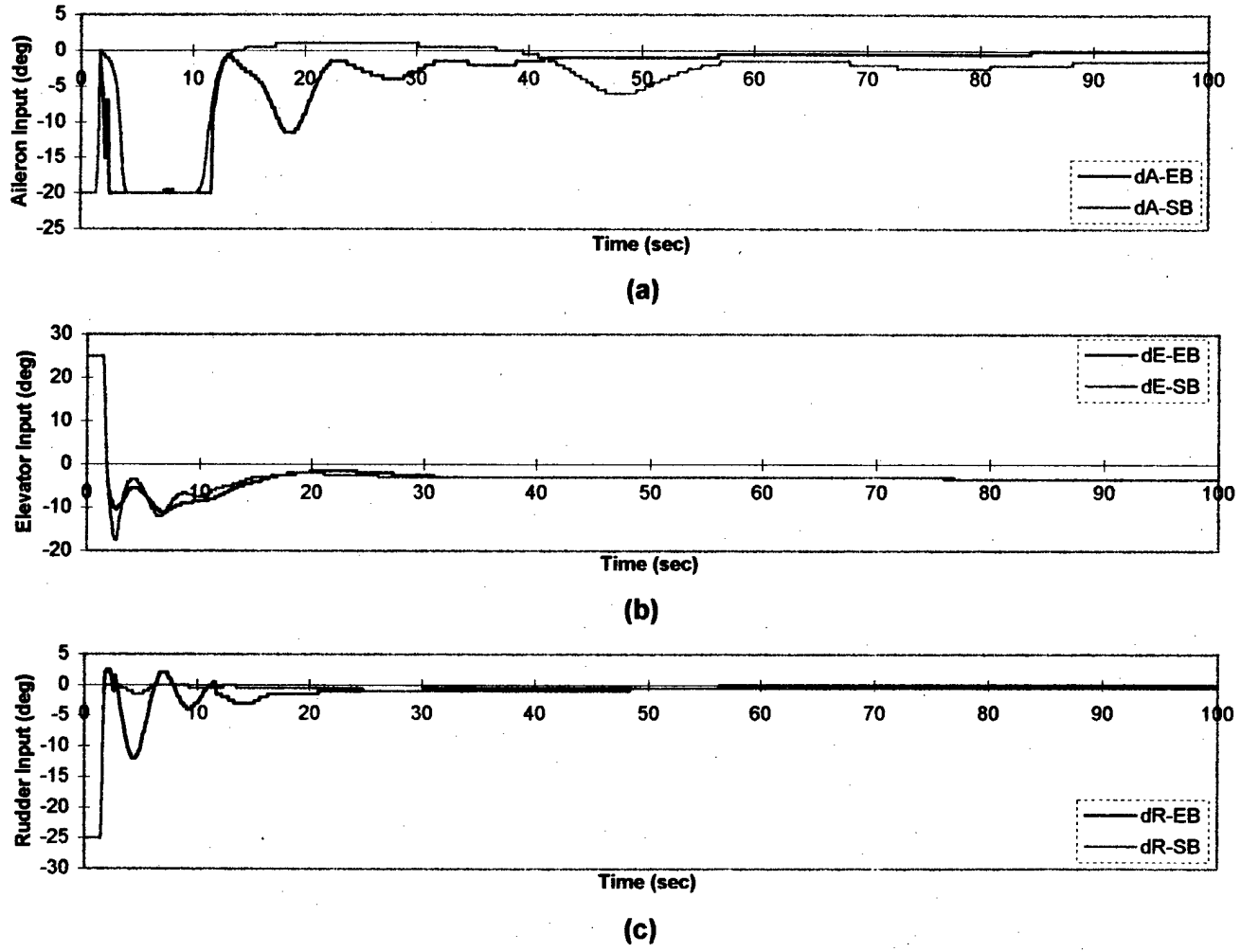


Figure 6.68: NN control inputs in response to p, q, and r excitations above 4 deg/sec at HC.

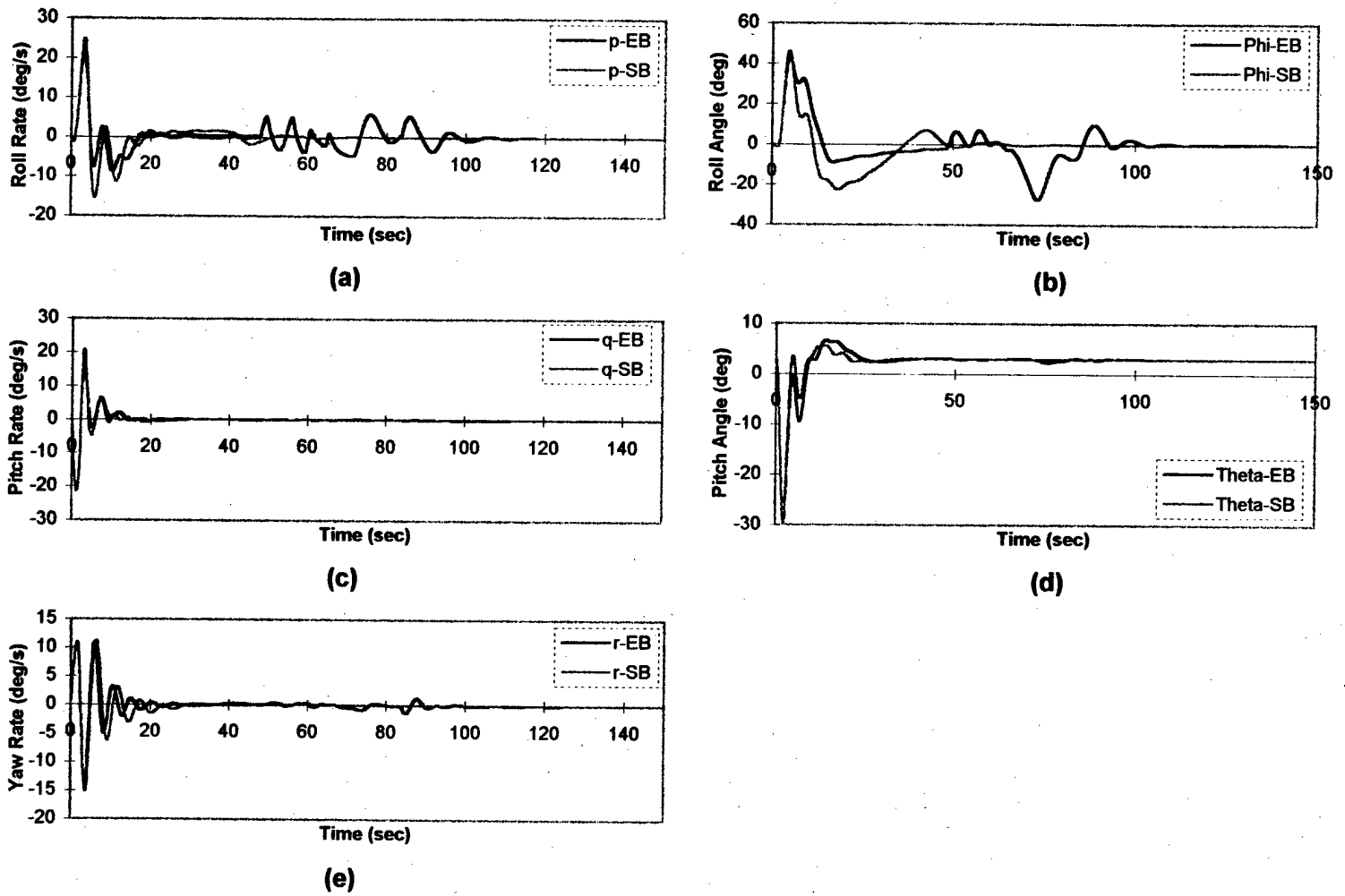


Figure 6.69: Aircraft response to non-linear conditions when p, q, and r are above 6 deg/sec at HC.

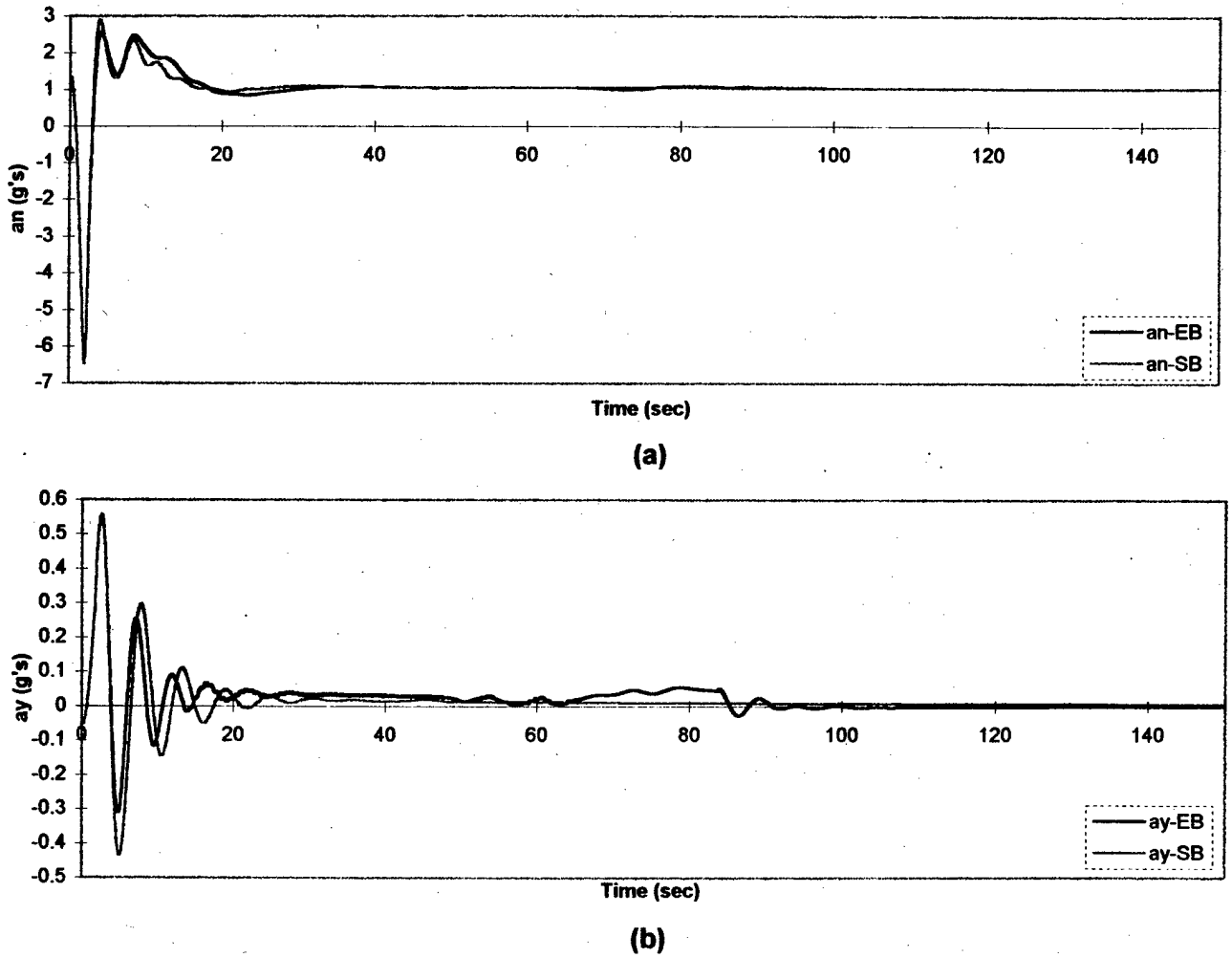


Figure 6.70: Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 6 deg/sec at HC.

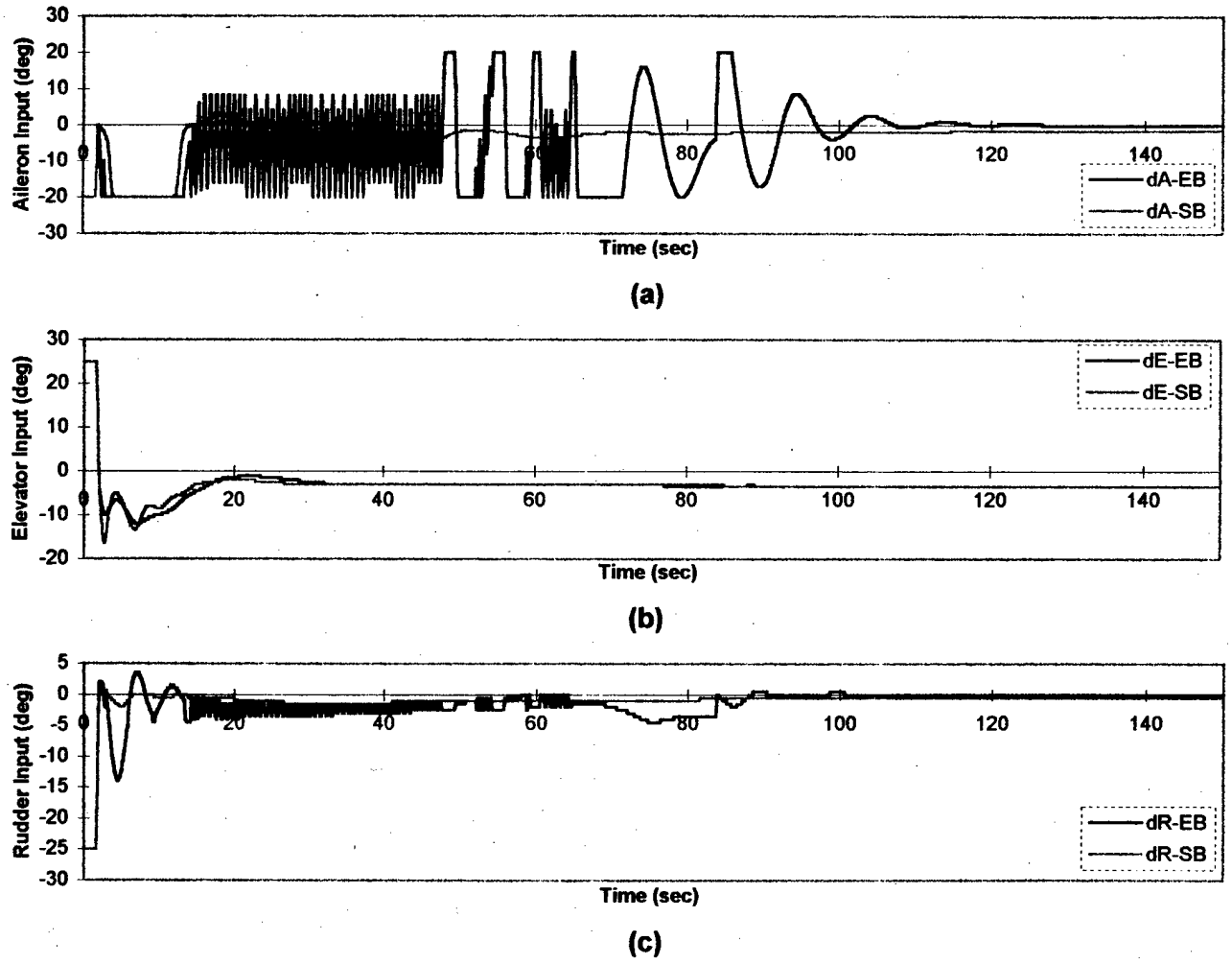


Figure 6.71: NN control inputs in response to p, q, and r excitations above 6 deg/sec at HC.

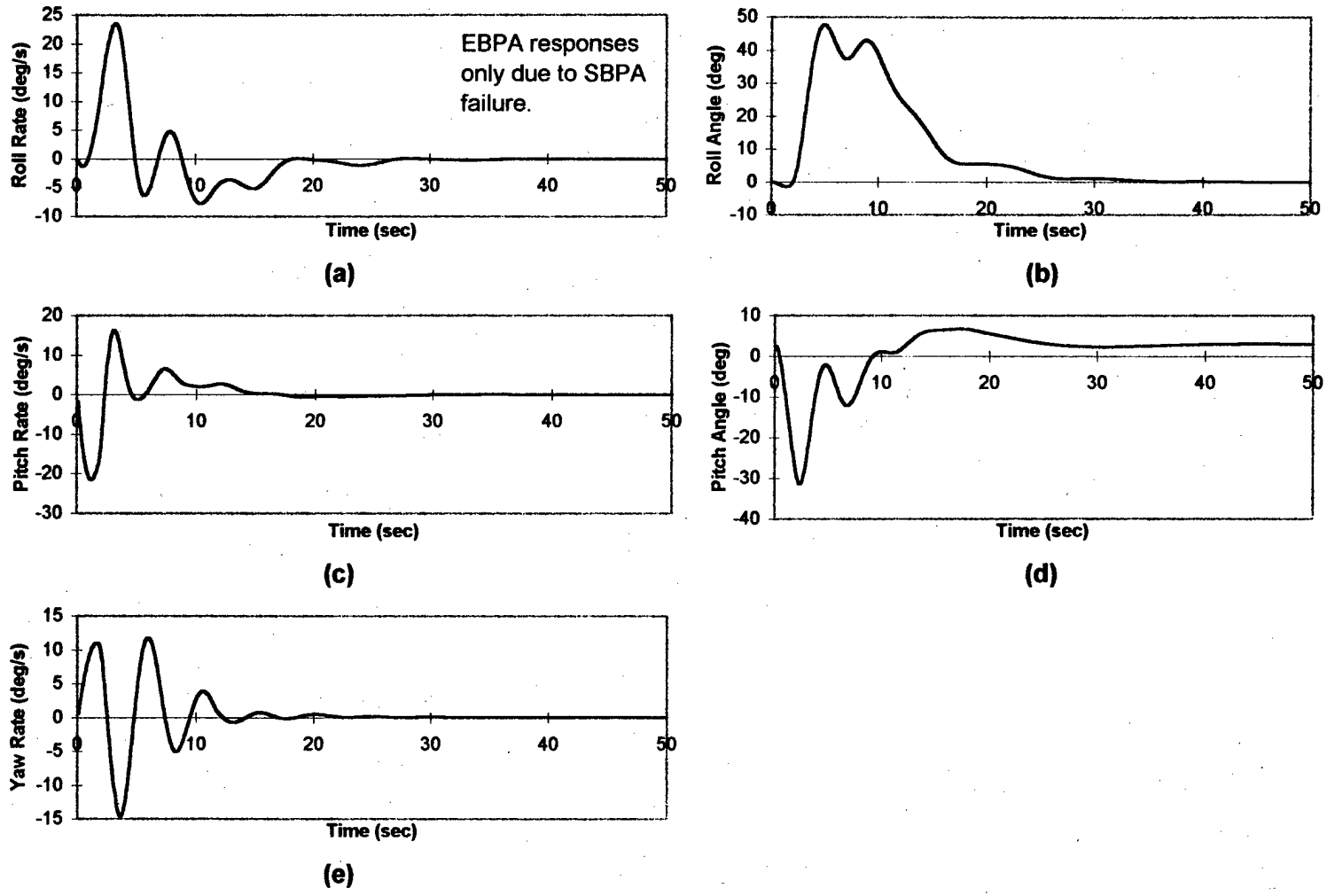
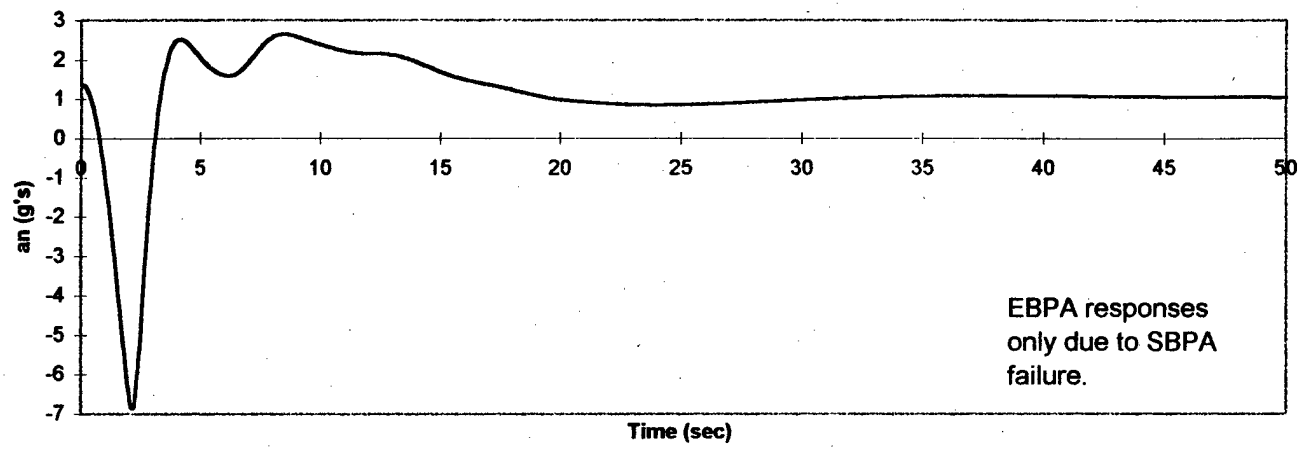
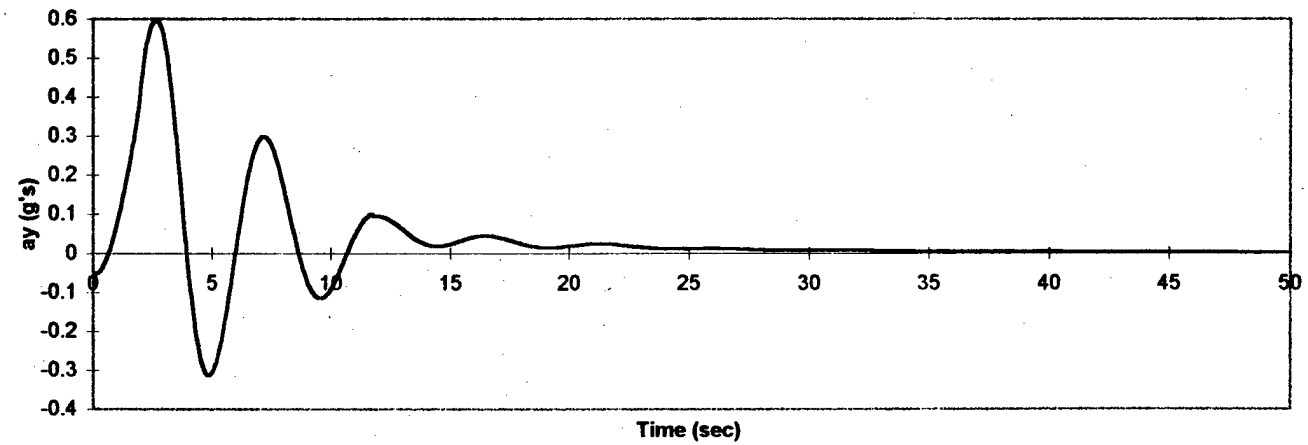


Figure 6.72: Aircraft response to non-linear conditions when p, q, and r are above 8 deg/sec at HC.



(a)



(b)

Figure 6.73: Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 8 deg/sec at LC.

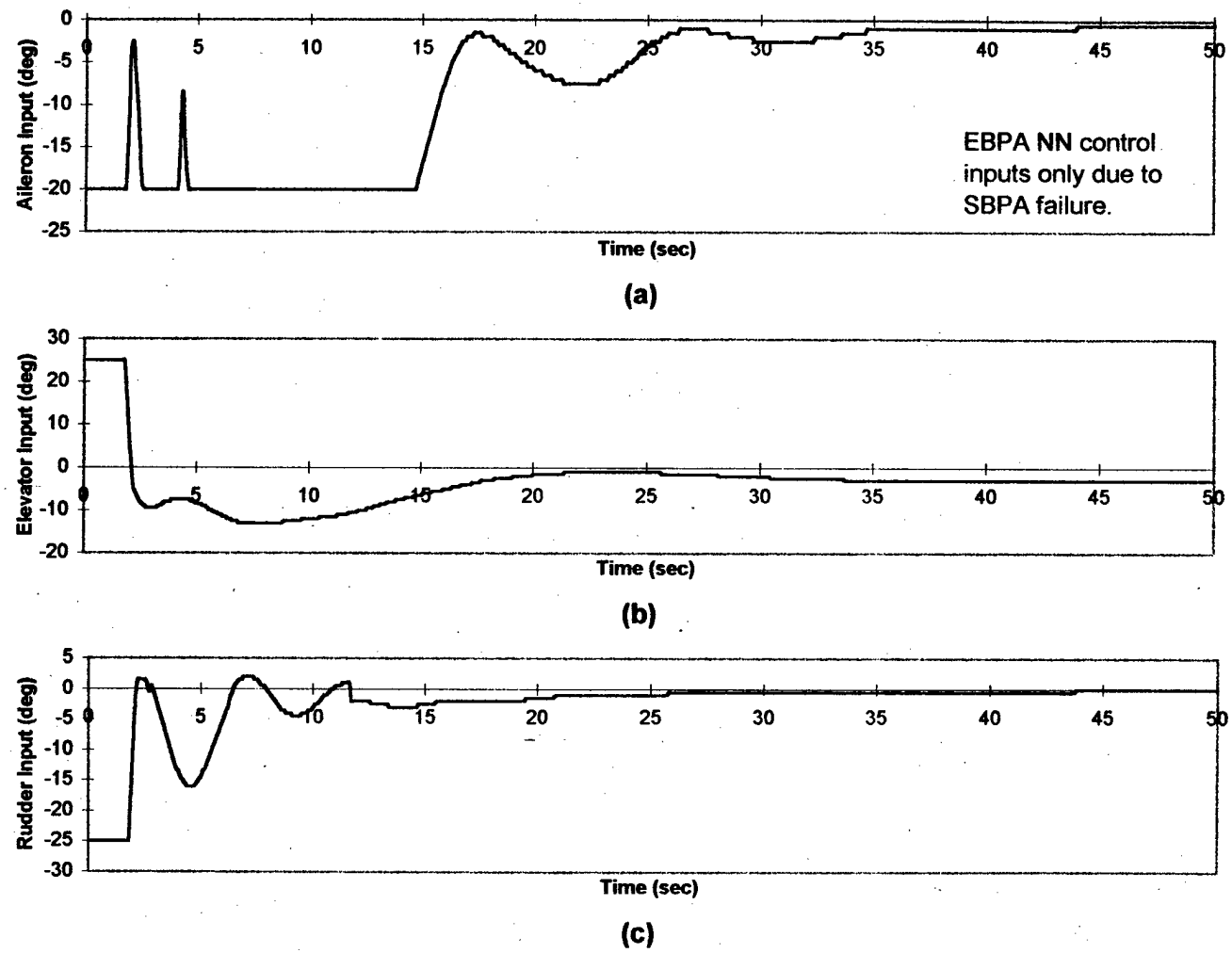


Figure 6.74: NN control inputs in response to p, q, and r excitations above 8 deg/sec at HC.

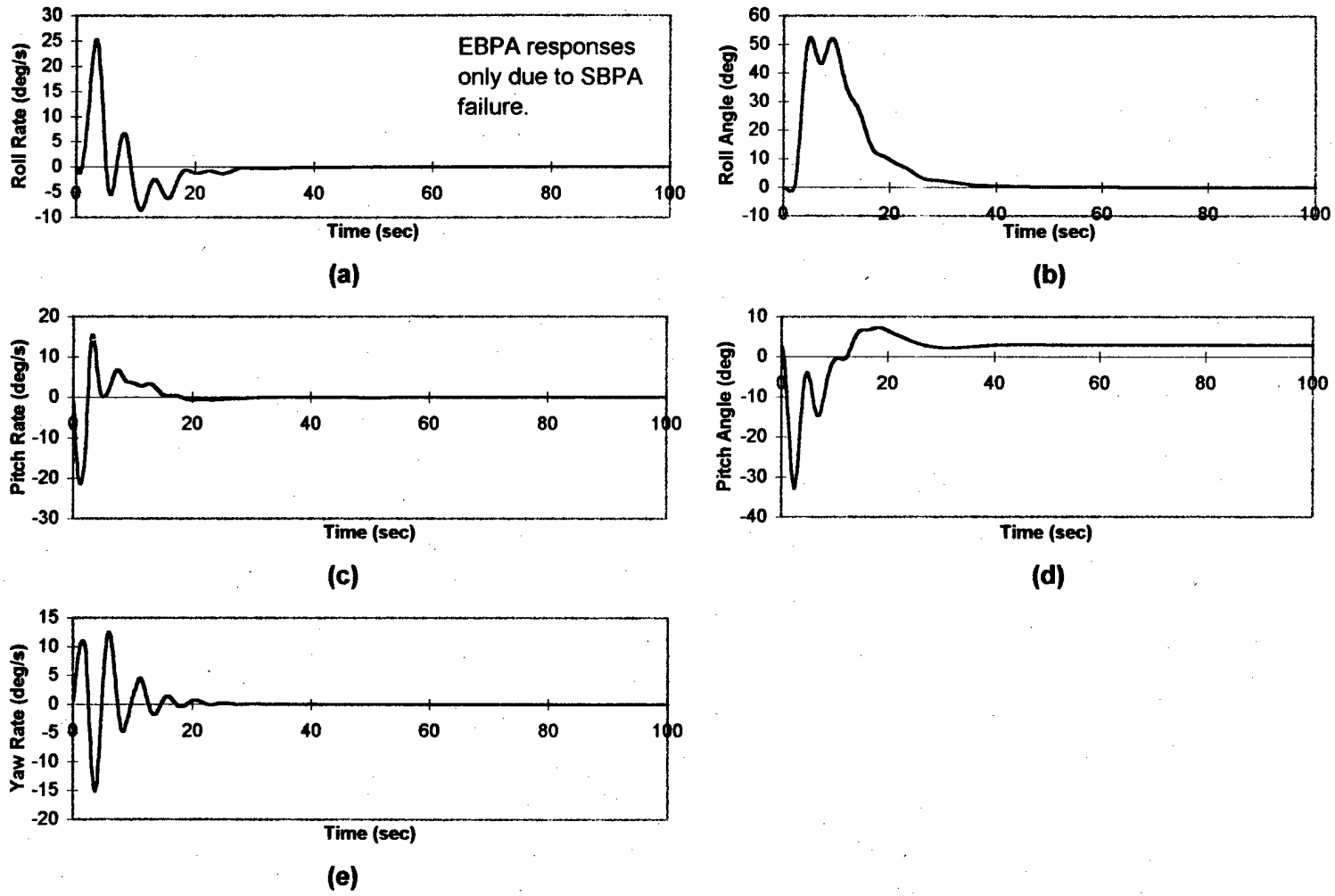
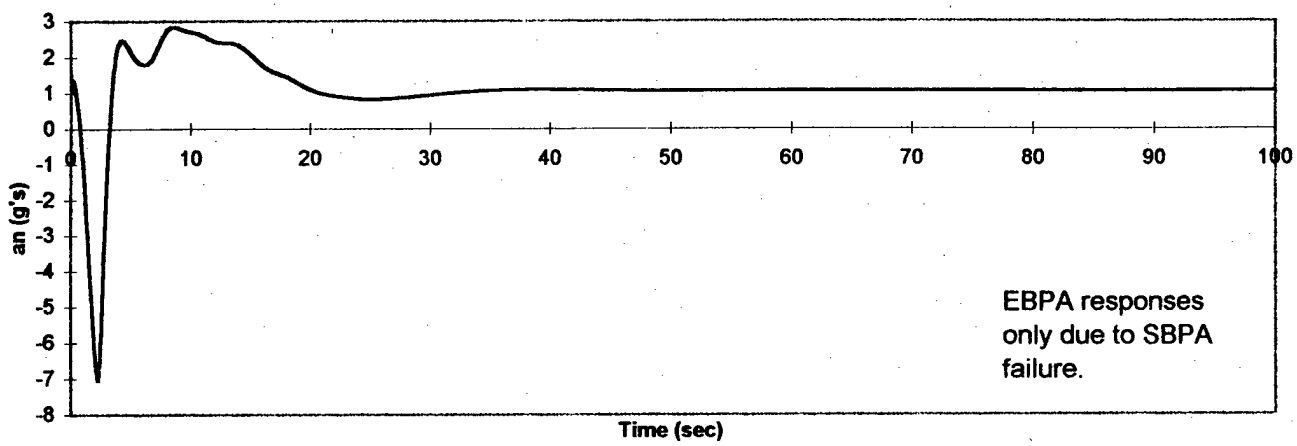
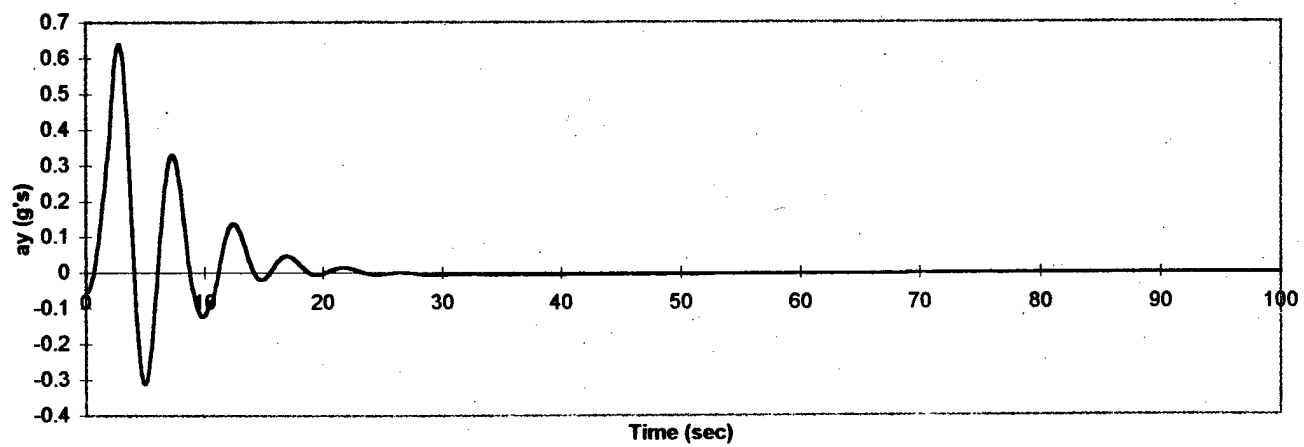


Figure 6.75: Aircraft response to non-linear conditions when p, q, and r are above 10 deg/sec at HC.



(a)



(b)

Figure 6.76: Aircraft normal and lateral acceleration responses to non-linear conditions when p, q, and r are above 10 deg/sec at HC.

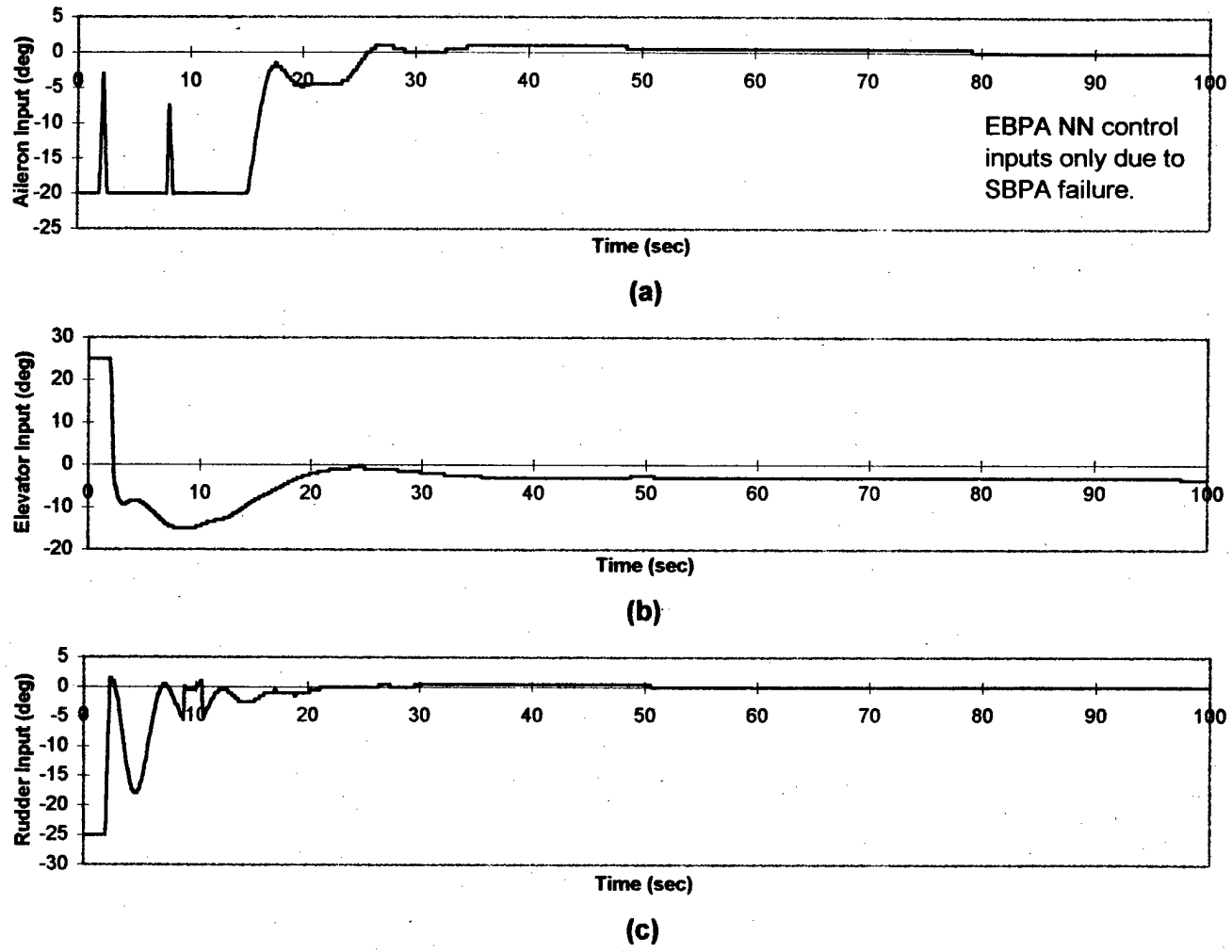


Figure 6.77: NN control inputs in response to p, q, and r excitations above 10 deg/sec at HC.

Approval of Examining Committee

Dr. Gary Morris, Ph.D.
Committee Member

Dr. Powsiri Klinkhachorn, Ph.D.
Committee Member

Dr. Marcello Napolitano, Ph.D.
Committee Chairman

27 Jan 98
Date