LEARNING IN CONNECTIONIST NETWORKS USING THE ALOPEX ALGORITHM

by

Kootala P. Venugopal

A Dissertation Submitted to the Faculty of

The College of Engineering

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy in Electrical Engineering

Florida Atlantic University
Boca Raton, Florida

April 1993

© Copyright by Kootala P. Venugopal 1993

ABSTRACT

Author:	***Kootala P. Venugopal***
Title:	***Learning in Connectionist Networks Using the Alopex Algorithm***
Institution:	***Florida Atlantic University***
Dissertation Advisors:	***Dr. A. S. Pandya and Dr. R. Sudhakar***
Degree:	***Doctor of Philosophy in Electrical Engineering***
Year:	***1993***

The Alopex algorithm is presented as a universal learning algorithm for connectionist models. It is shown that the Alopex procedure could be used efficiently as a supervised learning algorithm for such models. The algorithm is demonstrated successfully on a variety of network architectures. Such architectures include multilayer perceptrons, time–delay models, asymmetric, fully recurrent networks and memory neuron networks. The learning performance as well as the generalization capability of the Alopex algorithm are compared with those of the backpropagation procedure, concerning a number of benchmark problems, and it is shown that the Alopex has specific advantages over the backpropagation.

Two new architectures (gain layer schemes) are proposed for the on–line, direct adaptive control of dynamical systems using neural networks. The proposed schemes are shown to provide better dynamic response and tracking characteristics, than the other existing direct control schemes. A velocity reference scheme is introduced to improve the dynamic response of on–line learning controllers.

The proposed learning algorithm and architectures are studied on three practical problems; (i) Classification of handwritten digits using Fourier Descriptors; (ii) Recognition of underwater targets from sonar returns, considering temporal dependencies of consecutive returns and (iii) On–line learning control of autonomous underwater vehicles, starting with random initial conditions. Detailed studies are conducted on the learning control applications. Effect of the network learning rate on the tracking performance and dynamic response of the system are investigated. Also, the ability of the neural network controllers to adapt to slow and sudden varying parameter disturbances and measurement noise is studied in detail.

ACKNOWLEDGEMENT

I sincerely express my gratitude to my dissertation advisors Dr. A. S. Pandya and Dr. R. Sudhakar for their support and encouragement throughout the period of this research. Innumerable discussions with them, have helped me in concentrating the research effort and completing it in a shorter time–frame.

I wish to thank Dr. Stanley Dunn for his keen interest in this work, and providing the financial support throughout the period. The computing facilities at the Ocean Engineering Department had been vital in successfully completing this dissertation.

I extend my appreciation to Dr. Zvi Roth and Dr. P. S. Neelakanta for their helpful suggestions and encouragement. Part of this research work was performed during my stay at the General Motors Research Laboratories, during Summer–1992. I wish to thank Dr. K. P. Unnikrishnan for his guidance during this stay as well as on a number of other occasions. Also, I appreciate the suggestions from Andy Shein, Dr. Sam Smith and Dr. S. I. Sudharsanan at various stages.

Thanks are also due to the Super Computing Center, Florida State University, for providing their computing facility for this research.

LIST OF TABLES

Table 1.1	The Four Different Groups of Classifiers	20
Table 3.1	Comparison between Alopex and Backpropagation: Average number of iterations taken for convergence	39
Table 3.2	Encoder problem: average number of iterations taken for convergence by Alopex algorithm	41
Table 3.3	Fourier Descriptors for the digits shown in figure 3.8.	46

Figure 7.10 Controller performance under measurement noise
(a) Injection of white noise at the output
(b) Mean square error plotted as a function of noise to signal ratio 158

Figure 7.11 Comparison of Backpropagation and Alopex. Mean Square Error plotted as a function of the Noise to SIgnal Ratio. 159

Figure A.1 Representation of a plane curve. q(l) is the tangential direction distance l and f(l) the cumulative angular bend 171

Figure A.2 Description of a polygonal curve 172

Figure A.3 Following of the boundary of a simple closed curve 173

Figure 6.8 Backpropagation through time network
(a) A two neuron fully recurrent network
(b) The equivalent feedforward network for three time steps127

Figure 6.9 A two neuron recurrent network, with the dynamics represented by equations (6.22) and (6.23) 128

Figure 6.10 A simple network architecture. The neuron j has input connections from the lower layer as well as from external inputs 131

Figure 6.11 (a) A simple recurrent network.
(b) The error propagation network of figure (a). The dotted lines are not needed for the error propagation 134

Figure 6.12 Gradient Generation Scheme for the system 136

Figure 6.13 Gradient generation scheme for the nonlinear system 138

Figure 6.14 Feedforward Control Scheme using Neural Network 139

Figure 6.15 Gradient generation scheme for the control scheme shown in figure 6.14. 140

Figure 6.16 Feedback Control Scheme 141

FIgure 6.17 The Gradient Generation Scheme 142

Figure 7.1 Memory neuron network architecture. The details of the information transfer between the network neuron and memory neuron is shown in the inset. 146

Figure 7.2 Identification Scheme 149

Figure 7.3 Output of the plant and the identified model, after stopping the adaptation at 75, 000 iterations. 149

Figure 7.4 Recurrent neural network based direct control scheme. The network is adapted using the Alopex algorithm
(a) The control scheme
(b) The network architecture. The hidden layer is fully recurrent. All weights are updated synchronously based on the system error ($y_d - y$) 152

Figure 7.5 Control of the Pitch angle
(a) Response of the vehicle for a commanded input of 10 degrees
(b) Control signal generated by the neural network controller 153

Figure 7.6 Tracking of a time varying command signal (pitch)
(a) Command signal and the vehicle response
(b) Control input to the dynamics 154

Figure 7.7 Heading control
(a) Vehicle response for a commanded heading of 30 degrees
(b) Corresponding control signal. 155

Figure 7.8 Depth Control
(a) Response of the system for a commanded depth of 150 ft.
(b) Neural network output (control input) 156

Figure 7.9 Performance of the controller under sudden disturbance
(a) Change in forward thruster rpm.
(b) Response of the vehicle 157

Figure 5.16 Tracking of the pitch command under sudden disturbance. The forward thruster rpm is changed from 400 to 1200 at t=500 sec. and back to 400 at t=1000 sec. 102

Figure 5.17 Controlling the heading of the AUV using the gain layer scheme
(a) Response of the vehicle (heading) for a constant command of 20 degrees
(b) The control input to the rudder generated by the neural network for the response in (a) 103

Figure 5.18 Heading Control for oscillatory command signal
(a) Vehicle Response
(b) Corresponding control input 103

Figure 5.19 Tracking of the command signal when a slow varying disturbance is injected into the dynamics.
(a) Variation in the forward thruster rpm
(b) Response of the vehicle (heading) for two different learning rates 104

Figure 5.20 Tracking of a constant depth command. Learning rate for the neural network is 0.001 105

Figure 5.21 Depth Control for a time varying command signal
(a) Command Signal and the corresponding Vehicle Response
(b) Control input generated by the neural network controller 106

Figure 5.22 Direct Control Scheme for a 2–Input 2–Output MIMO System using gain layer 108

Figure 5.23 Control of the MIMO system. Tracking of step command inputs
(a) Output 1
(b) Output 2 109

Figure 5.24 Simultaneous control of pitch (θ) and vertical velocity (w), using the scheme in Figure 5.2.
(a) Tracking of the pitch command
(b) Tracking of the vertical velocity command 110

Figure 6.1 Time–delay Neural Network
(a) Tapped delay lines for the input
(b) Time–delayed interconnections 116

Figure 6.2 Time delay network for speech recognition, proposed by Tank and Hopfield[124]. Adopted from [72]. 117

Figure 6.3 The delay kernel for n=5 118

Figure 6.4 Partially recurrent network architectures.
(a) Feedback from the hidden layer to the input layer[62]
(b) Feedback from the output layer to the input layer[113] 121

Figure 6.5 Memory neuron network 124

Figure 6.6 Memory Kernel for different values of a. The effective time window for the past neuron states becomes shorter as a approaches unity. 124

Figure 6.7 Higher order memory kernel. The gamma network. 125

Figure 4.13 Vehicle response for a commanded pitch of 10 degrees

Figure 4.14 System trajectory for the response shown in Figure 4.13 83
The solid lines represent the response with velocity reference. . . . 84

Figure 4.15 The velocity reference signal and its effect on the change in the error convergence 85

Figure 4.16 The control signals generated by the neural network 85

Figure 5.1 Details of the vehicle coordinate system (adopted from [96]) 90

Figure 5.2 The controller architecture for the lateral motion. Each neural network and associated gain factor are trained with the correspondin error between the command signal and actual output. The vehicle states fed back to the path planner are the linear and angular velocities. 91

Figure 5.3 Controller architecture for the longitudinal motion 92

Figure 5.4 Tracking of the pitch command signal
Solid lines correspond to the vehicle response. 94

Figure 5.5 Tracking of the vertical velocity command signal. Note that the initial overshoot could be controlled by changing the network learning rate. The response shown is for $\eta = 0.002$. 94

Figure 5.6 Heading control of the vehicle. The learning rate for the neural network is 0.02 96

Figure 5.7 Tracking of the roll angle command signal.
The learning rate used is 0.5 96

Figure 5.8 Tracking of the side velocity command input with a neural network learning rate of 0.2. 96

Figure 5.9 Effect of the learning rate on the tracking performance of the system 97

Figure 5.10 Effect of initial weight distribution on the tracking performance of the system. 97

Figure 5.11 Response of the vehicle to step pitch command 98

Figure 5.12 Response of the vehicle (pitch) for a rectangular command signal. Learning rate for the controller is 0.008. 98

Figure 5.13 Effect of the controller learning rate on the system response (pitch) 99

Figure 5.14 . State space representation of the error convergence for the step pitch command. 99

Figure 5.15 Effect of learning rate on the adaptation of the controller under disturbance
(a) Slow varying disturbance on the dynamics: change in forward thruster rpm.
(b) Vehicle response for three different learning rates
(c) Vehicle response and corresponding control signal generated by the neural network
(d) State space representation of the error convergence 101

Figure 3.10 Sidescan sonar returns
(a) from the wooden box
(b) from the metallic drum
(c) from the background 48

Figure 3.11 The neural network architecture for sonar target recognition 49

Figure 3.12 Change in the error convergence when the number of neurons in first hidden layer is changed 51

Figure 3.13 Change in the error convergence as the learning rate h is changed. 51

Figure 3.14 Change in the error convergence as the stepsize is changed 51

Figure 3.15 Recurrent network architecture to solve temporal XOR Problem . 52

Figure 3.16 The output of a 1x2x1 network trained to solve τ=0 temporal XOR problem 53

Figure 3.17 . The output of a 1x4x1 network trained to solve τ=1 temporal XOR problem. 53

Figure 3.18 Output of a 1x6x1 network trained to solve τ=2 temporal XOR problem 54

Figure 3.19 The average error per bit for the training set for the three temporal XOR problems. 54

Figure 3.20 Error surfaces for 2–2–1 XOR problem (from [55]).
The network architecture is shown in Figure 3.1.
(a) Square error surface with respect to the weights w3 & w6
(b) Square error surface with respect to the weights w4 & w5
(c) Log error surface with respect to the weights w3 & w6
(d) Log error surface with respect to the weights w4 & w5 . . 55

Figure 4.1 Learning controller using linear discriminant function (adopted from [78]) 60

Figure 4.2 The control scheme 62

Figure 4.3 Tracking of the roll angle command signal 64

Figure 4.4 Tracking of the vertical velocity command signal 65

Figure 4.5 Inverse Transfer Matrix Scheme 71

Figure 4.6 Feedback Gain Scheme 73

Figure 4.7 Gain Layer Scheme 75

Figure 4.8 The Gain Layer Scheme with error feedback 78

Figure 4.9 System response for a commanded pitch of 8.5 degree.
(1) Specialized learning scheme
(2) Inverse Transfer matrix scheme
(3) Feedback gain scheme
(4) Gain layer scheme (without error feedback)
(5) Gain layer scheme (with error feedback) 79

Figure 4.10 Variation of the gain factor for a response as shown in Figure 4.9. 80

Figure 4.11 Comparison of Kawato's scheme and the 'gain layer' scheme for a time varying command signal. 81

Figure 4.12 Controller architecture using neural network. 83

LIST OF FIGURES

Figure 2.1	Categorization of different learning algorithms	13
Figure 2.3	Mean Square Error surfaces for different transfer functions (adopted from [75]) (a) Linear Transfer Function (b) Sigmoid Transfer Function (c) Signum Transfer Function	16
Figure 2.4	Diagrammatic Representation of Backpropagation (adopted from [77])	17
Figure 2.5	Relative difference between classifier training time and memory requirements(adopted from [46])	21
Figure 2.6	Identification of dynamical systems using neural networks (a) Parallel Identification (b) Series–Parallel Identification	23
Figure 2.7	Identification of dynamical systems using neural networks (a) Identification of a static nonlinear system using parallel scheme (b) Identification of a nonlinear dynamic system using series parallel scheme	24
Figure 2.8	Adaptive control techniques for dynamical systems (a) Direct Control (b) Indirect Control	26
Figure 3.1	The 2–2–1 network architecture for XOR problem	37
Figure 3.2	Convergence of the output error with Alopex algorithm using different error functions and backpropagation using square error (on XOR problem)	38
Figure 3.3	Effect of the stepsize on the convergence of the output error; Parity–3 Problem	38
Figure 3.6	Recognized patterns after the convergence for encoder problem (a) using log error. (b) using square error	41
Figure 3.7a.	Output of the network for the test pattern 1	42
b.	Output of the network for the test pattern 2	43
c.	Output of the network for the test pattern 3	43
Figure 3.8	Examples from the training set. Corresponding FDs are shown in Table 3.3.	45
Figure 3.9a.	Convergence of the output error during training	47
b.	Output error after convergence over all the samples.	47

7.2.2. Batch Updating 152
7.3. RESULTS 153
7.3.1. Experiment: 1 153
7.3.2. Experiment: 2 155
7.3.3. Experiment: 3 155
7.3.4. Controller Performance under Disturbance 156
7.4. COMPARISON OF BACKPROPAGATION AND ALOPEX ALGORITHMS 158
7.5. SUMMARY 160

8. SUMMARY AND CONCLUSIONS 161

APPENDIX A
FOURIER DESCRIPTORS 170

APPENDIX B
CONTROL SCHEMES FOR UNDERWATER VEHICLES . 175

APPENDIX C
NONLINEAR EQUATIONS OF MOTION FOR THE AUV . 179

REFERENCES 183

5.3.1.3. Heading Control 95
5.3.1.4. Control of Other Parameters 95
5.3.2. Effect of the Learning Rate 97
5.3.3. Effect of Initial Weight Distribution 97
5.3.4. Nonlinear Dynamics 98
5.3.4.1. Pitch Control 98
5.3.4.2. Heading Control 102
5.3.4.3. Depth Control 105
5.4. GAIN LAYER SCHEME FOR THE LEARNING CONTROL OF MIMO SYSTEMS 107
5.4.1. Indirect Control Scheme for MIMO Systems 107
5.4.2. Direct Control Scheme for MIMO Systems 108
5.5. SUMMARY 110

6. RECURRENT NEURAL NETWORK ARCHITECTURES AND ALGORITHMS FOR APPLICATIONS INVOLVING DYNAMICAL SYSTEMS 113

6.1. IMPORTANCE OF TIME AND ITS REPRESENTATION 114
6.2. PARALLELIZING THE TIME AND TIME DELAY NETWORKS 115
6.3. NEURAL NETWORKS WITH MEMORY 120
6.3.1. Memory Neuron Networks 122
6.4. BACKPROPAGATION THROUGH TIME (BTT) NETWORKS: 126
6.5. FULLY RECURRENT NEURAL NETWORKS: THE REAL–TIME RECURRENT LEARNING ALGORITHM (RTRL) 128
6.6. RECURRENT BACKPROPAGATION 130
6.7. THE DYNAMIC BACKPROPAGATION ALGORITHM 134
6.7.1. Gradient Methods for Linear Systems 135
6.7.2. Gradient Methods for Nonlinear Systems 137
6.7.3. Gradient Methods using Backpropagation Networks 139
6.8. SUMMARY 142

7. LEARNING IN RECURRENT NEURAL NETWORKS USING THE ALOPEX ALGORITHM 144

7.1. ALOPEX ALGORITHM FOR TRAINING MEMORY NEURON NETWORKS 145
7.1.1. Memory Neuron Network Architecture 145
7.1.2. Learning 146
7.1.3. Results 148
7.2. ALOPEX ALGORITHM FOR TRAINING RECURRENT NEURAL NETWORKS 150
7.2.1. Control Scheme 151

3.6. LEARNING AND CLASSIFICATION OF DYNAMIC PATTERNS 47
3.6.1. 'Parallelizing the time' Approach: Recognition of underwater objects from side scan sonar returns 47
3.6.2. Recurrent Network architecture: Classification of Temporal XOR 52
3.7. SUMMARY 54

4. LEARNING CONTROL OF DYNAMICAL SYSTEMS USING FEEDFORWARD NEURAL NETWORKS: PART I 57

4.1. PATTERN CLASSIFIERS AND LEARNING CONTROLLERS 58
4.2. THE NEURAL NETWORK BASED LEARNING CONTROLLERS 60
4.2.1. On–line and Off–line Controllers 61
4.3. ALOPEX ALGORITHM FOR THE LEARNING CONTROL OF DYNAMICAL SYSTEMS 62
4.3.1. The Vehicle Dynamics 63
4.3.2. Results 64
4.4. THE BACKPROPAGATION ALGORITHM FOR LEARNING CONTROL APPLICATIONS 66
4.4.1. The network error and the system error 67
4.4.2. Backpropagating the error through the dynamics 68
4.4.3. The Inverse Transfer Matrix Scheme 69
4.4.4. The Feedback Gain Scheme 71
4.4.5. The Gain Layer Schemes 74
4.4.6. Updating the Gain factor 75
4.4.7. Effect of the error feedback 76
4.4.8. Simulation Results 78
4.5. IMPROVING THE DYNAMIC RESPONSE USING VELOCITY REFERENCE FEEDBACK 82
4.5.1. The Velocity Reference Scheme 82
4.5.2. Simulation Results 84
4.6. SUMMARY 86

5. LEARNING CONTROL OF DYNAMICAL SYSTEMS USING FEEDFORWARD NEURAL NETWORKS: PART II 88

5.1. THE VEHICLE DYNAMICS 89
5.2. CONTROLLER IMPLEMENTATION 90
5.2.1. Scaling of the command signal 93
5.3. SIMULATION RESULTS 93
5.3.1. Linear Dynamics 94
5.3.1.1. Pitch Control 94
5.3.1.2. Depth Control 95

TABLE OF CONTENTS

1. INTRODUCTION 1

1.1. RATIONALE OF THE PRESENT RESEARCH 4

1.2. THE RESEARCH APPROACH 6

1.3. CONTRIBUTIONS 9

1.4. ORGANIZATION OF THE DISSERTATION 10

2. LEARNING IN CONNECTIONIST MODELS 12

2.1. SUPERVISED LEARNING ALGORITHMS 13

2.1.1. Multilayer Neural Networks: Backpropagation algorithm 15

2.1.2. The Boltzmann Machine 18

2.2. NEURAL NETWORKS FOR PATTERN CLASSIFICATION 19

2.3. NEURAL NETWORKS FOR IDENTIFICATION AND CONTROL 22

2.3.1. System Identification 22

2.3.2. Control 25

2.4. SUMMARY 27

3. THE ALOPEX ALGORITHM FOR SUPERVISED LEARNING IN NEURAL NETWORKS 28

3.1. THE ALOPEX PROCESS 29

3.2. THE ALOPEX ALGORITHM AS AN OPTIMIZATION ALGORITHM . 30

3.3. THE ALOPEX ALGORITHM AS A LEARNING ALGORITHM FOR NEURAL NETWORKS 31

3.3.1. Temperature Perturbation 34

3.3.2. Batch Updating 34

3.4. IMPORTANT PROPERTIES OF THE ALOPEX ALGORITHM 35

3.5. SIMULATION RESULTS FOR STATIC LEARNING TASKS 36

3.5.1. Parity Learning 37

3.5.2. Symmetry Problem 39

3.5.3. Encoder Problem 40

3.5.4. MONKS Problems 41

3.5.5. 2D object recognition using Alopex algorithm 44

LIST OF TABLES

Table 1.1	The Four Different Groups of Classifiers	20
Table 3.1	Comparison between Alopex and Backpropagation: Average number of iterations taken for convergence	39
Table 3.2	Encoder problem: average number of iterations taken for convergence by Alopex algorithm	41
Table 3.3	Fourier Descriptors for the digits shown in figure 3.8.	46

For: venu
Printed on: Sat, Feb 5, 1994 15:22:15

Document: tbl_doc
Last saved on: Sat, Feb 5, 1994 15:14:43

Figure 7.2 Identification Scheme 149

Figure 7.3 Output of the plant and the identified model,
after stopping the adaptation at 75, 000 iterations. 149

Figure 7.4 Recurrent neural network based direct control scheme.
The network is adapted using the Alopex algorithm
(a) The control scheme
(b) The network architecture. The hidden layer is fully recurrent.
All weights are updated synchronously based on the system
error ($y_d - y$) 152

Figure 7.5 Control of the Pitch angle
(a) Response of the vehicle for a commanded input of
10 degrees
(b) Control signal generated by the neural network controller 153

Figure 7.6 Tracking of a time varying command signal (pitch)
(a) Command signal and the vehicle response
(b) Control input to the dynamics 154

Figure 7.7 Heading control
(a) Vehicle response for a commanded heading of 30 degrees
(b) Corresponding control signal. 155

Figure 7.8 Depth Control
(a) Response of the system for a commanded depth of 150 ft.
(b) Neural network output (control input) 156

Figure 7.9 Performance of the controller under sudden disturbance
(a) Change in forward thruster rpm.
(b) Response of the vehicle 157

Figure 7.10 Controller performance under measurement noise
(a) Injection of white noise at the output
(b) Mean square error plotted as a function of noise
to signal ratio 158

Figure 7.11 Comparison of Backpropagation and Alopex. Mean Square
Error plotted as a function of the Noise to SIgnal Ratio. 159

Figure A.1 Representation of a plane curve. q(l) is the tangential direction
distance l and f(l) the cumulative angular bend 171

Figure A.2 Description of a polygonal curve 172

Figure A.3 Following of the boundary of a simple closed curve 173

Figure 5.21 Depth Control for a time varying command signal
(a) Command Signal and the corresponding Vehicle Response
(b) Control input generated by the neural network controller 106

Figure 5.22 Direct Control Scheme for a 2–Input 2–Output MIMO System using gain layer 108

Figure 5.23 Control of the MIMO system. Tracking of step command inputs
(a) Output 1
(b) Output 2 109

Figure 5.24 Simultaneous control of pitch (θ) and vertical velocity (w), using the scheme in Figure 5.2.
(a) Tracking of the pitch command
(b) Tracking of the vertical velocity command 110

Figure 6.1 Time–delay Neural Network
(a) Tapped delay lines for the input
(b) Time–delayed interconnections 116

Figure 6.2 Time delay network for speech recognition, proposed by Tank and Hopfield[124]. Adopted from [72]. 117

Figure 6.3 The delay kernel for n=5 118

Figure 6.4 Partially recurrent network architectures.
(a) Feedback from the hidden layer to the input layer[62]
(b) Feedback from the output layer to the input layer[113] 121

Figure 6.5 Memory neuron network 124

Figure 6.6 Memory Kernel for different values of a. The effective time window for the past neuron states becomes shorter as a approaches unity. 124

Figure 6.7 Higher order memory kernel. The gamma network. 125

Figure 6.8 Backpropagation through time network
(a) A two neuron fully recurrent network
(b) The equivalent feedforward network for three time steps127

Figure 6.9 A two neuron recurrent network, with the dynamics represented by equations (6.22) and (6.23) 128

Figure 6.10 A simple network architecture. The neuron j has input connections from the lower layer as well as from external inputs 131

Figure 6.11 (a) A simple recurrent network.
(b) The error propagation network of figure (a). The dotted lines are not needed for the error propagation 134

Figure 6.12 Gradient Generation Scheme for the system 136

Figure 6.13 Gradient generation scheme for the nonlinear system 138

Figure 6.14 Feedforward Control Scheme using Neural Network 139

Figure 6.15 Gradient generation scheme for the control scheme shown in figure 6.14. 140

Figure 6.16 Feedback Control Scheme 141

FIgure 6.17 The Gradient Generation Scheme 142

Figure 7.1 Memory neuron network architecture. The details of the information transfer between the network neuron and memory neuron is shown in the inset. 146

Figure 5.3 Controller architecture for the longitudinal motion 92

Figure 5.4 Tracking of the pitch command signal
Solid lines correspond to the vehicle response. 94

Figure 5.5 Tracking of the vertical velocity command signal. Note that the initial overshoot could be controlled by changing the network learning rate. The response shown is for $\eta = 0.002$. 94

Figure 5.6 Heading control of the vehicle. The learning rate for the neural network is 0.02 96

Figure 5.7 Tracking of the roll angle command signal.
The learning rate used is 0.5 96

Figure 5.8 Tracking of the side velocity command input with a neural network learning rate of 0.2. 96

Figure 5.9 Effect of the learning rate on the tracking performance of the system 97

Figure 5.10 Effect of initial weight distribution on the tracking performance of the system. 97

Figure 5.11 Response of the vehicle to step pitch command 98

Figure 5.12 Response of the vehicle (pitch) for a rectangular command signal. Learning rate for the controller is 0.008. 98

Figure 5.13 Effect of the controller learning rate on the system response (pitch) 99

Figure 5.14 State space representation of the error
. convergence for the step pitch command. 99

Figure 5.15 Effect of ;earning rate on the adaptation of the controller under disturbance
(a) Slow varying disturbance on the dynamics: change in forward thruster rpm.
(b) Vehicle response for three different learning rates
(c) Vehicle response and corresponding control signal generated by the neural network
(d) State space representation of the error convergence 101

Figure 5.16 Tracking of the pitch command under sudden disturbance. The forward thruster rpm is changed from 400 to 1200 at t=500 sec. and back to 400 at t=1000 sec. 102

Figure 5.17 Controlling the heading of the AUV using the gain layer scheme
(a) Response of the vehicle (heading) for a constant command of 20 degrees
(b) The control input to the rudder generated by the neural network for the response in (a) 103

Figure 5.18 Heading Control for oscillatory command signal
(a) Vehicle Response
(b) Corresponding control input 103

Figure 5.19 Tracking of the command signal when a slow varying disturbance is injected into the dynamics.
(a) Variation in the forward thruster rpm
(b) Response of the vehicle (heading) for two different learning rates 104

Figure 5.20 Tracking of a constant depth command. Learning rate for the neural network is 0.001 105

Figure 3.13	Change in the error convergence as the learning rate h is changed.	51
Figure 3.14	Change in the error convergence as the stepsize is changed	51
Figure 3.15	Recurrent network architecture to solve temporal XOR Problem	52
Figure 3.16	The output of a 1x2x1 network trained to solve τ=0 temporal XOR problem	53
Figure 3.17 .	The output of a 1x4x1 network trained to solve τ=1 temporal XOR problem.	53
Figure 3.18	Output of a 1x6x1 network trained to solve τ=2 temporal XOR problem	54
Figure 3.19	The average error per bit for the training . set for the three temporal XOR problems.	54
Figure 3.20	Error surfaces for 2–2–1 XOR problem (from [55]). The network architecture is shown in figure 3.1. (a) Square error surface with respect to the weights w3 & w6 (b) Square error surface with respect to the weights w4 & w5 (c) Log error surface with respect to the weights w3 & w6 (d) Log error surface with respect to the weights w4 & w5 . .	55
Figure 4.1	Learning controller using linear discriminant function (adopted from [78])	60
Figure 4.2	The control scheme	62
Figure 4.3	Tracking of the roll angle command signal	64
Figure 4.4	Tracking of the vertical velocity command signal	65
Figure 4.5	Inverse Transfer Matrix Scheme	71
Figure 4.6	Feedback Gain Scheme	73
Figure 4.7	Gain Layer Scheme	75
Figure 4.8	The Gain Layer Scheme with error feedback	78
Figure 4.9	System response for a commanded pitch of 8.5 degree. (1) Specialized learning scheme (2) Inverse Transfer matrix scheme (3) Feedback gain scheme (4) Gain layer scheme (without error feedback) (5) Gain layer scheme (with error feedback)	79
Figure 4.10	Variation of the gain factor for a response as shown in Figure 4.9.	80
Figure 4.11	Comparison of Kawato's scheme and the 'gain layer' scheme for a time varying command signal.	81
Figure 4.12	Controller architecture using neural network.	83
Figure 4.13	Vehicle response for a commanded pitch of 10 degrees	
Figure 4.14	System trajectory for the response shown in Figure 4.13 . . . . . The solid lines represent the response with velocity reference. . . .	83 84
Figure 4.15	The velocity reference signal and its effect on the change in the error convergence	85
Figure 4.16	The control signals generated by the neural network	85
Figure 5.1	Details of the vehicle coordinate system (adopted from [96])	90
Figure 5.2	The controller architecture for the lateral motion. Each neural network and associated gain factor are trained with the correspondin error between the command signal and actual output. The vehicle states fed back to the path planner are the linear and angular velocities.	91

LIST OF FIGURES

Figure 2.1	Categorization of different learning algorithms	13
Figure 2.3	Mean Square Error surfaces for different transfer functions (adopted from [75]) (a) Linear Transfer Function (b) Sigmoid Transfer Function (c) Signum Transfer Function	16
Figure 2.4	Diagrammatic Representation of Backpropagation (adopted from [77])	17
Figure 2.5	Relative difference between classifier training time and memory requirements(adopted from [46])	21
Figure 2.6	Identification of dynamical systems using neural networks (a) Parallel Identification (b) Series–Parallel Identification	23
Figure 2.7	Identification of dynamical systems using neural networks (a) Identification of a static nonlinear system using parallel scheme (b) Identification of a nonlinear dynamic system using series parallel scheme	24
Figure 2.8	Adaptive control techniques for dynamical systems (a) Direct Control (b) Indirect Control	26
Figure 3.1	The 2–2–1 network architecture for XOR problem	37
Figure 3.2	Convergence of the output error with Alopex algorithm using different error functions and backpropagation using square error (on XOR problem)	38
Figure 3.3	Effect of the stepsize on the convergence of the output error; Parity–3 Problem	38
Figure 3.6	Recognized patterns after the convergence for encoder problem (a) using log error. (b) using square error	41
Figure 3.7a.	Output of the network for the test pattern 1	42
b.	Output of the network for the test pattern 2	43
c.	Output of the network for the test pattern 3	43
Figure 3.8	Examples from the training set. Corresponding FDs are shown in Table 3.3.	45
Figure 3.9a.	Convergence of the output error during training	47
b.	Output error after convergence over all the samples.	47
Figure 3.10	Sidescan sonar returns (a) from the wooden box (b) from the metallic drum (c) from the background	48
Figure 3.11	The neural network architecture for sonar target recognition	49
Figure 3.12	Change in the error convergence when the number of neurons in first hidden layer is changed	51

For: venu
Printed on: Sat, Feb 5, 1994 15:22:19

Document: fig_doc
Last saved on: Sat, Feb 5, 1994 15:14:30