1

PM-1 3½"x4" PHOTOGRAPHIC MICROCOPY TARGET
NBS 1010a ANSI/ISO #2 EQUIVALENT

***Predicting Weld Features
Using
Artificial Neural Network Technology***

by

Billy Kwok-Hung Chan, B.Eng, M.Eng.

A thesis submitted to
the Faculty of Graduate Studies and Research
in partial fulfilment of
the requirements for the degree of
Doctor of Philosophy

**Department of Mechanical and Aerospace Engineering
Ottawa-Carleton Institute
for Mechanical and Aerospace Engineering**

Carleton University
Ottawa, Ontario

January 15, 1996.

copyright
1996, Billy Kwok-Hung Chan

National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file *Votre référence*

Our file *Notre référence*

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-612-13752-X

Canada

Name CHIAN KWOK HUNG BILLY

Dissertation Abstracts International is arranged by broad, general subject categories. Please select the one subject which most nearly describes the content of your dissertation. Enter the corresponding four-digit code in the spaces provided.

PREDICTING WEIRD FEATURES USING ARTIFICIAL NEURAL

SUBJECT TERM

NETWORK
TECHNOLOGY

SUBJECT CODE

0794

U·M·I

Subject Categories

THE HUMANITIES AND SOCIAL SCIENCES

COMMUNICATIONS AND THE ARTS

Architecture 0729
Art History 0377
Cinema 0900
Dance 0378
Fine Arts 0357
Information Science 0723
Journalism 0391
Library Science 0399
Mass Communications 0708
Music 0413
Speech Communication 0459
Theater 0465

EDUCATION

General 0515
Administration 0514
Adult and Continuing 0516
Agricultural 0517
Art 0273
Bilingual and Multicultural 0282
Business 0688
Community College 0275
Curriculum and Instruction 0727
Early Childhood 0518
Elementary 0524
Finance 0277
Guidance and Counseling 0519
Health 0680
Higher 0745
History of 0520
Home Economics 0278
Industrial 0521
Language and Literature 0279
Mathematics 0280
Music 0522
Philosophy of 0998
Physical 0523

Psychology 0525
Reading 0535
Religious 0527
Sciences 0714
Secondary 0533
Social Sciences 0534
Sociology of 0340
Special 0529
Teacher Training 0530
Technology 0710
Tests and Measurements 0288
Vocational 0747

LANGUAGE, LITERATURE AND LINGUISTICS

Language
 General 0679
 Ancient 0289
 Linguistics 0290
 Modern 0291
Literature
 General 0401
 Classical 0294
 Comparative 0295
 Medieval 0297
 Modern 0298
 African 0316
 American 0591
 Asian 0305
 Canadian (English) 0352
 Canadian (French) 0355
 English 0593
 Germanic 0311
 Latin American 0312
 Middle Eastern 0315
 Romance 0313
 Slavic and East European 0314

PHILOSOPHY, RELIGION AND THEOLOGY

Philosophy 0422
Religion
 General 0318
 Biblical Studies 0321
 Clergy 0319
 History of 0320
 Philosophy of 0322
Theology 0469

SOCIAL SCIENCES

American Studies 0323
Anthropology
 Archaeology 0324
 Cultural 0326
 Physical 0327
Business Administration
 General 0310
 Accounting 0272
 Banking 0770
 Management 0454
 Marketing 0338
Canadian Studies 0385
Economics
 General 0501
 Agricultural 0503
 Commerce-Business 0505
 Finance 0508
 History 0509
 Labor 0510
 Theory 0511
Folklore 0358
Geography 0366
Gerontology 0351
History
 General 0578

Ancient 0579
Medieval 0581
Modern 0582
Black 0328
African 0331
Asia, Australia and Oceania 0332
Canadian 0334
European 0335
Latin American 0336
Middle Eastern 0333
United States 0337
History of Science 0585
Law 0398
Political Science
 General 0615
 International Law and Relations 0616
 Public Administration 0617
Recreation 0814
Social Work 0452
Sociology
 General 0626
 Criminology and Penology 0627
 Demography 0938
 Ethnic and Racial Studies 0631
 Individual and Family Studies 0628
 Industrial and Labor Relations 0629
 Public and Social Welfare 0630
 Social Structure and Development 0700
 Theory and Methods 0344
Transportation 0709
Urban and Regional Planning 0999
Women's Studies 0453

THE SCIENCES AND ENGINEERING

BIOLOGICAL SCIENCES

Agriculture
 General 0473
 Agronomy 0285
 Animal Culture and Nutrition 0475
 Animal Pathology 0476
 Food Science and Technology 0359
 Forestry and Wildlife 0478
 Plant Culture 0479
 Plant Pathology 0480
 Plant Physiology 0817
 Range Management 0777
 Wood Technology 0746
Biology
 General 0306
 Anatomy 0287
 Biostatistics 0308
 Botany 0309
 Cell 0379
 Ecology 0329
 Entomology 0353
 Genetics 0369
 Limnology 0793
 Microbiology 0410
 Molecular 0307
 Neuroscience 0317
 Oceanography 0416
 Physiology 0433
 Radiation 0821
 Veterinary Science 0778
 Zoology 0472
Biophysics
 General 0786
 Medical 0760
EARTH SCIENCES
 Biogeochemistry 0425
 Geochemistry 0996

Geodesy 0370
Geology 0372
Geophysics 0373
Hydrology 0388
Mineralogy 0411
Paleobotany 0345
Paleoecology 0426
Paleontology 0418
Paleozoology 0985
Palynology 0427
Physical Geography 0368
Physical Oceanography 0415

HEALTH AND ENVIRONMENTAL SCIENCES

Environmental Sciences 0768
Health Sciences
 General 0566
 Audiology 0300
 Chemotherapy 0992
 Dentistry 0567
 Education 0350
 Hospital Management 0769
 Human Development 0758
 Immunology 0982
 Medicine and Surgery 0564
 Mental Health 0347
 Nursing 0569
 Nutrition 0570
 Obstetrics and Gynecology 0380
 Occupational Health and Therapy 0354
 Ophthalmology 0381
 Pathology 0571
 Pharmacology 0419
 Pharmacy 0572
 Physical Therapy 0382
 Public Health 0573
 Radiology 0574
 Recreation 0575

Speech Pathology 0460
Toxicology 0383
Home Economics 0386

PHYSICAL SCIENCES

Pure Sciences
 Chemistry
 General 0485
 Agricultural 0749
 Analytical 0486
 Biochemistry 0487
 Inorganic 0488
 Nuclear 0738
 Organic 0490
 Pharmaceutical 0491
 Physical 0494
 Polymer 0495
 Radiation 0754
 Mathematics 0405
 Physics
 General 0605
 Acoustics 0986
 Astronomy and Astrophysics 0606
 Atmospheric Science 0608
 Atomic 0748
 Electronics and Electricity 0607
 Elementary Particles and High Energy 0798
 Fluid and Plasma 0759
 Molecular 0609
 Nuclear 0610
 Optics 0752
 Radiation 0756
 Solid State 0611
Statistics 0463
Applied Sciences
 Applied Mechanics 0346
 Computer Science 0984

Engineering
 General 0537
 Aerospace 0538
 Agricultural 0539
 Automotive 0540
 Biomedical 0541
 Chemical 0542
 Civil 0543
 Electronics and Electrical 0544
 Heat and Thermodynamics 0348
 Hydraulic 0545
 Industrial 0546
 Marine 0547
 Materials Science 0794
 Mechanical 0548
 Metallurgy 0743
 Mining 0551
 Nuclear 0552
 Packaging 0549
 Petroleum 0765
 Sanitary and Municipal 0554
 System Science 0790
Geotechnology 0428
Operations Research 0796
Plastics Technology 0795
Textile Technology 0994

PSYCHOLOGY

General 0621
Behavioral 0384
Clinical 0622
Developmental 0620
Experimental 0623
Industrial 0624
Personality 0625
Physiological 0989
Psychobiology 0349
Psychometrics 0632
Social 0451

The undersigned recommends to the Faculty of Graduate Studies and Research, acceptance of the thesis: **Predicting Weld Features Using Artificial Neural Network Technology**, submitted by Billy Kwok-Hung Chan, in partial fulfilment of the requirements of the degree of Doctor of Philosophy.

Thesis Supervisor

Thesis Co-supervisor

Chairman
Department of Mechanical
and Aerospace Engineering

Abstract

The use of artificial neural network (ANN) technology for predicting the heat-affected zone (HAZ) hardness and fusion zone (FZ) cross-sectional weld shape is presented in this thesis. In particular, the backpropagation network (BPN) method is applied to these problems. Two learning enhancements for BPN termed modified dynamic and step-declining are proposed and tested together with the traditional methods available for this purpose. Finally, the BPN method is adapted to solve the inverse weld shape problem of estimating welding conditions to provide a given weld shape.

An extensive data base of measured bead-on-plate HAZ hardness values was assembled from the literature for "training" and testing. The data is primarily associated with the submerged arc (SA) welding process because among other things, it is widely recognized in the industry to be the best controlled and most reliable welding technique. Nonetheless, the database also includes hardness measurements from research investigations where gas metal arc welding (GMAW) and the shielded metal arc (SMA) process were used.

To predict HAZ hardness from the input welding conditions (voltage, current, wire travel speed, plate temperature and plate thickness) and steel chemical composition, the calculation of an intermediate cooling time is necessary. A BPN is presented for this purpose. It is then combined with the hardness network to generate hardness values directly from the input welding conditions and steel chemical composition.

The data for the weld shape problem was kindly provided by Mr. Jack Pacey from the measurements of an investigation in the Northern College (Kirkland Lake, Ontario). Various critical dimensions (bead width, height, penetration, "bay length", deposit area and fusion area) were extracted from the traced weld shapes. BPNs were constructed and trained to predict these parameters from the welding conditions (current, voltage, wire travel speed and plate thickness). From these, a shape is approximated by fitting a semi-ellipse ($\text{area} > 40 \text{ mm}^2$) or parabola ($\text{area} \leq 40 \text{ mm}^2$) to the deposit area and a double parabola to the lower part of the weld (below the plate surface). The Kirkland Lake data applies only to GMAW with C-25 and M-2 shielding but the network is sufficiently general that it can be applied to any welding process given measurements for training.

The advantages of the ANN technology versus traditional regression methods used for these problems are discussed. In general, ANN provides practitioners with a more flexible, more reliable tool than regression methods. It can be extended to include welding experience as it is gained. In addition, various assumptions that are necessary when physical modelling is used are implicit in the ANN method.

Acknowledgements

I would like to express my deepest appreciation to my thesis supervisors, Prof. Malcolm J. Bibby and Prof. Neil M. Holtz, for their continued interest, encouragement and guidance provided during this thesis program and throughout my study at Carleton University. I thank Prof. Bibby and the Department of Mechanical and Aerospace Engineering, Carleton University for their financial support. I would like to thank all my friends and the staff of the Department of Mechanical and Aerospace Engineering at Carleton University for their help and co-operation.

I would like to express a special thanks to Corby Nicholson from Pipe-Protech Corrosion Survey Ltd. and Jack Pacey from the Northern College at Kirkland Lake. I appreciate Michel Beaudoin for his excellent work in preparing all the experimental welds required for this research under the supervision of Mr. Pacey. I would also want to thank Dr. Roop Chandel (Nanyang Technological University, Singapore) for his helpful suggestions regarding the weld geometry problem and Prof. H. M. Hafez (Department of System and Computer Engineering, Carleton University) for his valuable advice about implementing the backpropagation network system.

Finally, I would like to present this thesis to my beloved parents and wife.

Table of Contents

	Page
Abstract	i
Acknowledgements	ii
Table of Contents	iii
List of Figures	vii
List of Tables	xiv
Nomenclature	xviii
 1) Introduction	 1
1.1 Weld Problems	2
1.1.1 HAZ Hardness	4
1.1.1.1 HAZ Hardness Regression Models	5
1.1.1.2 800 to 500 °C Cooling Time	6
1.1.2 Weld Bead Geometry	9
1.1.2.1 Weld Bead Geometry Regression Models	10
1.2 Software for Predicting Weld Features	13
1.2.1 Limitations of the Regression Models	14
1.3 Artificial Intelligence and Artificial Neural Networks	14
1.4 Objective	17
 2) Artificial Neural Networks	 18
2.1 Background	18
2.2 History	20
2.3 Properties of ANN	21
2.4 Basis of ANN	23
2.4.1 ANN Learning	24
2.4.2 Connecting Nodes Together - Paradigms	24
2.4.2.1 Perception	25
2.4.2.2 Hopfield's Network	26
2.4.2.3 Backpropagation Network	26
2.4.2.4 Self-Organizing Map	27
2.4.2.5 Counter-Propagation Network	28

3)	Backpropagation ANN Scheme	29
3.1	Introduction	29
3.2	Theory	29
3.2.1	Forward Propagation	31
3.2.2	BPN and the Generalized Delta Rule	32
3.2.3	Learning Rate	36
3.2.4	Transfer Function	37
3.2.5	Root Mean Squared Error	37
3.2.6	Enhancements to the BPN Learning Algorithms	38
3.2.6.1	Fahlman's Derivative	40
3.2.6.2	Differential Step Size	40
3.2.6.3	Dynamic Learning	40
3.2.6.4	Modified Dynamic Learning	41
3.2.6.5	Step Declining Learning Rate	42
3.3	Other Issues of Backpropagation Network	42
3.3.1	Knowledge Verification	43
3.3.2	Scalability	43
3.3.3	Appropriate Numbers of Nodes, Hidden Layers and Training Patterns	44
3.4	Summary for software Implementation	45
3.5	PC Based BPN Development Tools - NNWork	47
4)	Performance of Modified Learning Enhancements	49
4.1	Test Problems and Results	49
4.1.1	XOR Problem	50
4.1.2	HAZ Hardness Problem	52
4.1.3	Weld Bead Geometry Problem	53
4.2	Discussion	55
5)	Predicting Weld 800 to 500 °C Cooling Time and HAZ Hardness Using the BPN Method	57
5.1	Introduction	57
5.2	Weld 800 to 500 °C Cooling Time	58
5.2.1	Backpropagation Networks for Predicting $t_{8/5}$	58
5.2.2	Results and Discussion	60
5.2.2.1	Predicting $t_{8/5}$ for Other Welding Processes	65
5.2.2.2	Comments about Process Efficiency	67
5.3	HAZ Hardness	68
5.3.1	Backpropagation Networks for Predicting HAZ Hardness	69
5.3.2	Results and Discussion	70
5.3.2.1	Alternative Number of Training Patterns	72
5.3.2.2	Using a Single Carbon Equivalent	74
5.3.2.3	Including the Effect of Silicon (Si)	76

5.3.2.4	Comparing BPN Prediction with the Regression Methods	77
5.3.2.5	Applying BPN Knowledge to Independent Data	79
5.3.2.6	Adapting Existing Hardness Knowledge for Specific Job Shop Predictions	81
5.4	Combining $t_{8/5}$ and HAZ Hardness Prediction BPNs	83
5.4.1	Results and Discussion	84
5.5	Chapter Summary	86
6)	Predicting Weld Bead Geometry with BPNs	88
6.1	Introduction	88
6.2	Algorithms for Estimating the Weld Bead Shape	90
6.2.1	Results and Discussion	91
6.2.1.1	Comments about Bead Shape Continuity	96
6.3	Predicting C-25 Shielding Weld Bead Geometry	96
6.3.1	Results and Discussion	98
6.3.1.1	Bead Width - C-25 Shielding	98
6.3.1.2	Bead Height - C-25 Shielding	99
6.3.1.3	Penetration - C-25 Shielding	100
6.3.1.4	Bay Length - C-25 Shielding	101
6.3.1.5	Deposit Area (Upper-Bead) Area - C-25 Shielding	102
6.3.1.6	Plate Fusion (Lower-Bead) Area - C-25 Shielding	103
6.3.1.7	Pictorial Total Fusion Area Approximation Using Predicted Parameters	104
6.4	Predicting M-2 Shielding Weld Bead Geometry	106
6.4.1	Results and Discussion	106
6.5	Including A Non-Quantitative Parameter in the BPN	110
6.5.1	Results and Discussion	111
6.6	The Inverse Geometry Problem - Predicting Welding Parameters Given the Weld Bead Dimensions	112
6.6.1	Results and Discussion	114
6.6.1.1	Welding Conditions Estimated Using BW, BH, Pene and f^{25}	114
6.6.1.2	Welding Conditions Estimated Using BW, Pene and A2	116
6.6.1.3	Welding Conditions Estimated Using BW, Pene, A1 and A2	118
6.6.1.4	Welding Conditions Estimated Using BW, A1 and A2	119
6.6.1.5	Comparing the Four Different Weld Bead Criteria	121
6.7	Effect of Plate Thickness on Weld Dimensions	123
6.7.1	Results and Discussion	124
6.8	Chapter Summary	126

7)	General Discussion	128
7.1	Comparing BPN Predictions with Previous Works	128
7.2	BPN Prediction and Fundamental Analysis	129
7.3	Effects of Network Structure and Training Patterns	130
8)	General Conclusions	132
9)	Future Works	133
	References	134
	Appendix I Letters of WeldSoft Recognition	148
	Appendix II Yurioka et al Experimental HAZ Hardness Data	153
	Appendix III SAW Experimental Cooling Time	156
	Appendix IV Independent Cooling Time Data	159
	Appendix V AMCA Hardness Data	165
	Appendix VI Gas Metal Arc Weld Size Data	167
	Appendix VII NNWork User's Guide	184
	Appendix VIII Cooling Time Knowledge	205
	Appendix IX HAZ Hardness Knowledge	209
	Appendix X Terasaki HAZ Hardness Model	219
	Appendix XI Approximated Weld Shapes	220
	Appendix XII GMAW Bead Geometry Knowledge	229
	Appendix XIII Inverse Bead Geometry Examples	252

Lists of Figures

Figure	Page
1.1 Typical fusion welding set up.	1
1.2 Fe-C phase diagram with corresponding weld microstructures.	2
1.3 Schematic sketch of a typical butt weld cross section showing under-bead cold cracking in the HAZ.	3
1.4 Schematic HAZ characteristic hardness curve.	5
1.5 A schematic temperature-time profile of a weld thermal cycle.	6
1.6 A schematic sketch of a point, a line and an intermediate heat source.	7
1.7a A schematic illustration of 3-D heat flow - the thick plate situation.	7
1.7b A schematic illustration of 2-D heat flow - the thin plate situation.	7
1.7c A schematic illustration of 2.5-D heat flow - the intermediate plate thickness situation.	7
1.8 Values of C_r corresponding to H_r for computing cooling times in the transitional 2.5-D heat flow situation.	9
1.9 Definitions for describing weld geometry.	9
2.1 A schematic sketch of two neurons in a nervous system.	18
2.2 A typical artificial neural network.	19
2.3 A schematic illustration of a multi-perceptron network.	25
2.4 A 4-node Hopfield network which can store patterns consist of 4 binary fields.	26
2.5 A schematic sketch of a 3-layer backpropagation network.	27
3.1 A three-layer backpropagation network with input (X), network output (J) and desired output (Y) data.	30

3.2	The summation and transfer functions of a node in the network.	31
3.3	The search for a global minimum in weight-error space.	32
3.4	A schematic illustration of how a large learning rate may induce a danger of missing the global minimum.	36
3.5	A sigmoid transfer function, see equation (3-37).	37
3.6	A modified three-layer backpropagation network with bias nodes.	39
4.1	Graphical representation of XOR problem.	50
4.2	Schematic graphs of η and RMS error versus number of epochs for unsuccessful learning with dynamic learning.	55
4.3	Schematic graphs of η and RMS error versus number of epochs for successful learning without dynamic learning.	55
4.4	Schematic graphs of η and RMS error versus number of epochs for successful learning with modified dynamic learning.	56
4.5	Schematic graphs of η and RMS error versus number of epochs for successful learning with step-declining dynamic learning rate.	56
5.1	Proposed backpropagation network for predicting $t_{8/5}$.	59
5.2	Comparison of backpropagation networks (2-hidden layer with 5 and 3 hidden nodes respectively and a single-hidden with 5 hidden nodes) and Adams' cooling times, with experimental $t_{8/5}$.	62
5.3	Comparison of networks (2-hidden layer with 5 and 3 hidden nodes respectively and a single-hidden layer with 5 hidden nodes) and Adams' cooling times, with Kohno measured data.	62
5.4	Sensitivity of thermal conductivity of iron to temperature changes.	64
5.5	Sensitivity of volumetric specific heat of iron to temperature changes.	64
5.6	Comparison of GMAW $t_{8/5}$ predicted by using BPN (5&3-hidden node 2-hidden layer) and Adams' cooling time models (with valid data only)	65

5.7	Comparison of SMAW $t_{8/5}$ predicted by using BPN (5&3-hidden node 2-hidden layer) and Adams' cooling time models (process efficiency = 80%).	66
5.8	Comparison of SMAW $t_{8/5}$ predicted by using BPN (5&3-hidden node 2-hidden layer) and Adams' cooling time models (process efficiency = 70%).	66
5.9	A schematic diagram of BPN for predicting $t_{8/5}$ with welding process as one of the input parameters.	67
5.10	A schematic BPN diagram used for predicting HAZ hardness.	69
5.11	Comparison of BPNs predicted hardness with experimental values.	71
5.12	Characteristic hardness curves (CHCs) predicted by BPN for two different steels (A1&A7) - see Appendix II.	72
5.13	Comparison of predictions generated with the knowledge of 40 and 50 training patterns.	72
5.14	Comparison of predictions generated with the knowledge from 41 and 50 training patterns. Note: one of the patterns in the 41 group was specially selected.	73
5.15	Characteristic hardness curves of steels A1 and A7 (Appendix II), predicted with the knowledge from 40 and 41 training patterns.	73
5.16	Comparison of measured and BPNs with 4 hidden nodes and 50 training patterns (using different carbon equivalents) predicted hardness.	74
5.17	Comparison of different HAZ hardness prediction models.	78
5.18	BPN and NSC-S(CE) predicted CHCs with experimental values for steel B1 from AMCA data.	79
5.19	BPN and NSC-S(CE) predicted CHCs with experimental values for steel B2 from AMCA data.	80
5.20	BPN and NSC-S(CE) predicted CHCs with experimental values for steel B3 from AMCA data.	80
5.21	BPN (with adapted knowledge) and NSC-S(CE) predicted CHCs with experimental values for steel B1 from AMCA data.	82

5.22	BPN (with adapted knowledge) and NSC-S(CE) predicted CHCs with experimental values for steel B2 from AMCA data.	82
5.23	BPN (with adapted knowledge) and NSC-S(CE) predicted CHCs with experimental values for steel B3 from AMCA data.	83
5.24	A schematic diagram showing the combined network for predicting HAZ hardness from the welding conditions and steel chemical composition.	83
5.25	Comparison of BPNs (original=BPN-41, adapted=BPN-44) and regression models (Adams' & Suzuki) predicted HAZ hardness with AMCA data.	84
5.26	Comparison of $t_{8/5}$ predicted by using BPN and Adams' cooling time models for AMCA data.	85
6.1	Approximation of lower-bead shape with bead bay parameters included.	90
6.2	Approximation of low-bead using two parabolas.	90
6.3	Comparison of the actual and pictorial approximated (semi-ellipse and parabola) deposit areas with C-25 shielding.	91
6.4	Comparison of the actual and pictorial approximated (semi-ellipse and parabola) deposit areas with M-2 shielding.	92
6.5	Frequency diagram of the bay angle of C-25 shielding weld bead shape.	93
6.6	Comparison of the actual and represented (cosine, ellipses and parabolas) plate fusion areas with C-25 shielding.	94
6.7	Comparison of the actual and represented (cosine, ellipses and parabolas) plate fusion areas with M-2 shielding.	95
6.8	Experimental and approximated weld shape comparison (C-25 shielding).	95
6.9	Experimental and approximated weld shape comparison (M-2 shielding).	95
6.10	A schematic sketch of the approximated weld shape showing the discontinuities.	96
6.11	A schematic diagram of the BPN used for solving the weld bead geometry problem.	97

6.12	Comparison of BPN (single-hidden layer and 5 hidden nodes) predicted bead width with experimental values (C-25).	98
6.13	Comparison of BPN (single-hidden layer and 5 hidden nodes) predicted bead height with experimental values (C-25).	99
6.14	Comparison of BPN (2-hidden layer with 8 and 6 hidden nodes) predicted penetration with experimental values (C-25).	100
6.15	Comparison of BPN (2-hidden layer with 4 and 3 hidden nodes) predicted bay length with experimental values (C-25).	101
6.16	Comparison of BPN (single-hidden layer and 4 hidden nodes) predicted deposit area with experimental values (C-25).	102
6.17	Comparison of BPN (2-hidden layer with 5 and 3 hidden nodes) predicted plate fusion area with experimental values (C-25).	103
6.18	Comparison of parabola-ellipse approximated upper-bead shape using predicted dimensions with measured areas (C-25).	104
6.19	Comparison of double-parabola approximated lower-bead shape using predicted dimensions with measured areas (C-25).	104
6.20	Experimental and predicted weld shape approximation comparison (C-25 shielding).	105
6.21	Comparison of predicted and measured bead width of M-2 shielding using 6-node 1-hidden layer network.	108
6.22	Comparison of predicted and measured bead height of M-2 shielding using 6-node 1-hidden layer network.	108
6.23	Comparison of predicted and measured penetration of M-2 shielding using 5&3-node 2-hidden layer network.	108
6.24	Comparison of predicted and measured bay length of M-2 shielding using 5-node 1-hidden layer network.	108
6.25	Comparison of predicted and measured deposit area of M-2 shielding using 7-node 1-hidden layer network.	108

6.26	Comparison of predicted and measured bead width of M-2 shielding using 7-node 1-hidden layer network.	108
6.27	Comparison of ellipse approximated upper-bead shape with measured values (M-2).	109
6.28	Comparison of double-parabola approximated lower-bead shape with measured values (M-2).	109
6.29	Experimental and predicted weld shape approximation comparison (M-2 shielding).	109
6.30	A schematic diagram of the BPN tested for solving the inverse weld bead problem.	114
6.31	Comparison of measured arc current with values estimated by BPN using BW, BH, Pene, I^{225} and h_r .	115
6.32	Comparison of measured arc voltage with values estimated by BPN using BW, BH, Pene, I^{225} and h_r .	116
6.33	Comparison of measured wire travel speed with values estimated by BPN using BW, BH, Pene, I^{225} and h_r .	116
6.34	Comparison of measured arc current with values estimated by BPN using BW, Pene, A2 and h_r .	116
6.35	Comparison of measured arc voltage with values estimated by BPN using BW, Pene, A2 and h_r .	117
6.36	Comparison of measured wire travel speed with values estimated by BPN using BW, Pene, A2 and h_r .	117
6.37	Comparison of measured arc current with values estimated by BPN using BW, Pene, A1, A2 and h_r .	118
6.38	Comparison of measured arc voltage with values estimated by BPN using BW, Pene, A1, A2 and h_r .	118
6.39	Comparison of measured wire travel speed with values estimated by BPN using BW, Pene, A1, A2 and h_r .	118

6.40	Comparison of measured arc current with values estimated by BPN using BW, A1, A2 and h_t .	119
6.41	Comparison of measured arc voltage with values estimated by BPN using BW, A1, A2 and h_t .	120
6.42	Comparison of measured wire travel speed with values estimated by BPN using BW, A1, A2 and h_t .	120
6.43	Comparison of the desired and expected bead width using the results of the BW, Pene & A2 inverse network.	122
6.44	Comparison of the desired and expected penetration using the results of the BW, Pene & A2 inverse network.	122
6.45	Comparison of the desired and expected plate fusion area using the results of the BW, Pene & A2 inverse network.	122
6.46	The effect of plate thickness on bead width, height, penetration and bay length for $I_c=200$ amperes, $V_o=25$ volts, $S_p=5$ mm/s and with C-25 shielding.	125
6.47	The effect of plate thickness on deposit area and plate fusion area for $I_c=200$ amperes, $V_o=25$ volts, $S_p=5$ mm/s and with C-25 shielding.	125

Lists of Tables

Table	Page
1-1 Summary of Welding Process Efficiencies for Commonly Used Welding Heat Sources.	8
1-2 Tabulated Values of C_r Corresponding to H_r .	9
1-3a Coefficients for Chandel's Weld Bead dimension Models (C-25 Shielding Gas and +ve Electrode Polarity).	11
1-3b Coefficients for Chandel's Weld Bead dimension Models (C-25 Shielding Gas and -ve Electrode Polarity).	11
1-3c Coefficients for Chandel's Weld Bead dimension Models (M-2 Shielding Gas and +ve Electrode Polarity).	11
1-3d Coefficients for Chandel's Weld Bead dimension Models (M-2 Shielding Gas and -ve Electrode Polarity).	12
1-4 Coefficients for Chandel's Wire Melting Rate Equation.	12
2-1 Hebbian Learning.	20
2-2 Sample Applications of Different Paradigms.	25
4-1 XOR's Truth-Table.	50
4-2 Performance of Different Learning Enhancement Algorithms for the XOR Problem (Measured by the Number of Epochs Required for Successful Learning).	51
4-3 Performance of Different Learning Enhancement Algorithms for the HAZ Hardness Problem (Measured by the Number of Epochs Required for Successful Learning).	53
4-4 Performance of Different Learning Enhancement Algorithms for the Weld Bead Geometry Problem (Measured by the Number of Epochs Required for Successful Learning).	54
5-1 Welding Parameters Used to Train and Test the $t_{8/5}$ Prediction Networks.	60

5-2	Network Experimentation for Deciding the Appropriate Number of Training Patterns for the $t_{8/5}$ Prediction Using Group A Test Data (mean $t_{8/5}=20.50s$).	61
5-3	Network Experimentation for Testing the Accuracy of $t_{8/5}$ Prediction Using the Group A Test Data (mean $t_{8/5}=20.50s$).	63
5-4	Network Experimentation for Testing the Accuracy of $t_{8/5}$ Prediction Using the Group B Independent Data Set (mean $t_{8/5}=25.50s$).	63
5-5	Summary of GMAW $t_{8/5}$ Values Predicted by Using BPN (2-hidden layer with 5 and 3 hidden nodes) and the Adams' Cooling Time Models.	66
5-6	Summary of SMAW $t_{8/5}$ Values Predicted by Using BPN (2-hidden layer with 5 and 3 hidden nodes) and the Adams' Cooling Time Models (mean $t_{8/5}=14.35s$).	67
5-7	Ranges of $t_{8/5}$, Carbon Content, Carbon Equivalents and HAZ Hardness Values Used for the Training of the Hardness BPNs.	68
5-8	Network Experimentation for Determining the Appropriate Number of Training Patterns Required for the HAZ Hardness Problem (mean $H_{max}=309.48VHN$).	70
5-9	Results of BPN Experimentation for Verifying the Accuracy of the Knowledge of Different Network Structures with 50 Training Patterns (mean $H_{max}=309.48VHN$).	71
5-10	Comparison of HAZ Hardness Prediction Networks Performance Using Different Input Parameters (mean $H_{max}=309.48VHN$).	75
5-11	Results of HAZ Hardness Predictions Generated to Show the Effect of Silicon as an Explicit Parameter (single-hidden layer network with 4 hidden nodes was used) (mean $H_{max}=309.48VHN$).	77
5-12	Comparison of Different HAZ Hardness Prediction Models (mean $H_{max}=309.48VHN$).	78
5-13	Comparison of BPN and regression Model Predictions Applied to the AMCA Hardness Data (mean $H_{max}=283.78VHN$).	79

5-14	Comparison of the Results of Networks Trained with (43 patterns) and without (41 patterns) Additional AMCA Hardness Data.	81
5-15	Summary of Hardness Predicted by BPNs and Regression Models for AMCA Experimental Data (mean Hmax=283.78VHN) - see Appendix V.	85
5-16	Comparison of BPN and Adams' Cooling Time Models Predicted $t_{8/5}$ for AMCA Experimental Data (mean $t_{8/5}$ =22.37s) - see Appendix III.	86
6-1	Ranges of Arc Current, Voltage, Wire Travel Speed and Plate Thickness Used in the GMAW Bead Geometry Investigation - see Appendix VI for details.	88
6-2	Comparison of Different Upper-Bead Area Algorithms.	92
6-3	Comparison of Different Lower-Bead Area Algorithms.	94
6-4	A Summary of (C-25) Bead Width Predictions Generated by Different Networks Using 21 Training Patterns (mean BW=12.32mm).	98
6-5	A Summary of (C-25) Bead Height Predictions Generated by Different Networks (mean BH=3.11mm).	99
6-6	A Summary of (C-25) Penetration Predictions Generated by Different Networks Using 21 Training Patterns (mean Pene=3.25mm).	101
6-7	A Summary of (C-25) Bay Length at 22.5° Predictions Generated by Different Networks Using 21 Training Patterns (mean $l^{22.5}$ =3.02mm).	102
6-8	A Summary of (C-25) Deposit (Upper-Bead) Area Predictions Generated by Different Networks (mean A1=27.14mm ²).	103
6-9	A Summary of (C-25) Plate Fusion (Lower-Bead) Area Predictions Generated by Different Networks Using 21 Training Patterns (mean A2=19.90mm ²).	104
6-10	Comparison of Different Bead Shape Approximation Algorithms Using Predicted C-25 Weld Dimensions (BH, BW, Pene and $l^{22.5}$).	105
6-11	A Summary of Correlation Coefficients of M-2 Shielding Weld Bead Dimensions Predicted by Different Networks Using 21 Sets of Training Data.	107
6-12	A Summary of Means, Standard Deviations of Difference and Mean Percentage Errors of M-2 Shielding Weld Bead Dimensions Predicted by Different Networks Using 21 Sets of Training Data.	107

6-13	Comparison of Different Bead Shape Approximation Algorithms Using Predicted M-2 Weld Dimensions (Bead Width, Bead Height, Penetration and Bay Length at 22.5°).	110
6-14	A Summary of Correlation Coefficients of C-25 and M-2 Shielding Weld Bead Dimensions Predicted by Different Networks Using 42 Sets of Training Data.	111
6-15	A Summary of Means and S.D.s of Difference of C-25 and M-2 Shielding Weld Bead Dimensions Predicted by Different Networks Using 42 Sets of Training Data.	112
6-16	Summary of Estimates of Welding Parameters from Various Networks Using BW, BH, Pene, ρ^{25} and h_i for C-25 Shielding.	115
6-17	Summary of Estimates of Welding Parameters from Various Networks Using BW, Pene, A2 and h_i for C-25 Shielding.	117
6-18	Summary of Estimates of Welding Parameters from Various Networks Using BW, Pene, A1, A2 and h_i for C-25 Shielding.	119
6-19	Summary of Estimates of Welding Parameters from Various Networks Using BW, A1, A2 and h_i for C-25 Shielding.	120
6-20	The Summary of the Results of Weld Bead Dimensions Predicted by Using the Estimated Welding Conditions for C-25 Shielding.	123
6-21	Comparison of the Effects of Welding Parameters on C-25 Bead Width.	124
6-22	Networks Used for Predicting Weld Bead Dimensions with C-25 and M-2 Shielding.	127
6-23	Networks Used for Estimating Welding Conditions from Weld Bead Dimensions for C-25 Shielding.	127

Nomenclature

FZ - fusion zone

HAZ - heat-affected zone

IIW - the International Institute of Welding

PWHT - post-weld heat treated

GMAW - gas metal arc welding

SAW - submerged arc welding

GTAW - gas tungsten arc welding

FCAW - flux cored arc welding

t_{8/5} - weld 800 to 500°C cooling time (sec)

CHC - characteristic hardness curve

wt% - weight percent

CE - IIW carbon equivalent (wt%)

Pcm - Ito's carbon equivalent (wt%)

B - boron content (wt%)

C - carbon content (wt%)

Cr - chromium content (wt%)

Cu - copper content (wt%)

Fe - iron content (wt%)

Mn - manganese content (wt%)

Mo - molybdenum content (wt%)

Ni - nickel content (wt%)

Nb - niobium content (wt%)

Si - silicon content (wt%)

V - vanadium content (wt%)

Hmax - maximum HAZ hardness (VHN)

t_{8/5}[3-D] - thick plate cooling time from 800 to 500 °C (sec)

t_{8/5}[2.5-D] - intermediate plate cooling time from 800 to 500 °C (sec)

t_{8/5}[2-D] - thin plate cooling time from 800 to 500 °C (sec)

η_{proc} - welding process efficiency (%)

Q - heat input rate per unit length (J/s/mm)

h_t - plate thickness (mm)

k - thermal conductivity (J/(s.mm.K))
 C_p - volumetric specific heat (J/(mm³K))
 T_o - plate initial temperature (°C)
 C_r - ratio of cooling time on thick plate to that on finite plate
 I_c - arc current (amperes)
 V_o - arc voltage (volts)
 S_p - wire travel speed (mm/s)
 H_r - relative thickness
 L_e - electrode extension (mm)
 D_e - electrode diameter (mm)
 BW - bead width (mm)
 BH - bead height (mm)
 $Pene$ - penetration (mm)
 θ - lower-bead bay angle (°)
 l^p - lower-bead bay length (mm)
 $A1$ - deposit area (mm²)
 $A2$ - plate fusion area (mm²)
 A - total fusion area (mm²)
 $a, b, c, d, e, k, p, q, r$ - equations coefficients
 MR - wire melting rate (kg/hr)
 AI - artificial intelligence
 ANN - artificial neural network
 BPN - backpropagation network
 H - input layer node of the backpropagation network
 I - hidden layer node of the backpropagation network
 J - output layer node of the backpropagation network
 h - number of input data per pattern / # input layer node
 i - number of hidden layer node
 j - number of output data per pattern / # output layer node
 W_{ih}^i - weight of link connecting nodes H_i and I_i of backpropagation network
 W_{ji}^j - weight of link connecting nodes I_i and J_j of backpropagation network
 n, a, b - counter
 P_n - n-th pattern
 p - total number of training patterns

$X_i^{P_n}$ - input of node H_i for pattern P_n
 $Y_j^{P_n}$ - desired output at node J_j for pattern P_n
 F - transfer function
 S_i^I - summation / activation of node I_i
 S_j^J - summation / activation of node J_j
 E^P - pattern error of the network
 G - a general function notation
 η - learning rate
 ΔW - change in weight
 δ - delta term
 RMS error - root mean squared error
 α - momentum coefficient
 $[t]$ - pervious pass
 $[t+1]$ - current pass
 c - an arbitrary small integer
 S.D. - standard deviation

Chapter One

Introduction

Welding is one of the most commonly used joining processes for steel structures. In fusion welding, two pieces of metal are brought close together and the adjacent surfaces are heated to the melting point. In general, filler metal is added to the gap between the parts to be joined.

Figure 1.1 Typical fusion welding set up.

The heat source is usually generated by a flow of electricity through an arc between an electrode and the joint metal, as shown in Figure 1.1. Flux and shielding gas are typical agents employed for protecting the filler core and the molten weld pool from oxidation. Since the base metal is subjected to elevated temperatures, the microstructure across the joint varies with position which is in general determined by peak temperature.

The microstructure of the welded joint is usually divided into three metallurgical zones, see Figure 1.1. The fusion zone (FZ) is formed from the re-solidification of a mixture of base metal and filler metal. The heat-affected zone (HAZ) is the metal adjacent to the fusion zone which has experienced elevated temperatures and excessive grain coarsening. The metal which is far away from the welding heat source and where the microstructure has not been affected by the heat source is termed unaffected parent metal.

A detailed description of the microstructural changes across a welded joint is illustrated in Figure 1.2. Region A (unaffected base metal) represents base metal which is

not affected by the heat source, i.e., the peak temperature experienced by this region is lower than the tempering temperature of steel (about 580 °C). Some grain growth or recrystallization may occur in region B if the base metal is previously cold rolled. In region C, a dual structure (coarse and enriched α +carbide) is present because of partial phase transformation. A relatively fine structure can be found in region D as a result of ferrite to austenite transformation on heating and the reverse on cooling. Austenitic grain growth occurs in region E which results in a relatively coarse structure adjacent to the weld fusion zone, region F. Regions B, C, D & E together is usually referred to as heat-affected zone.

Figure 1.2 Fe-C phase diagram with corresponding weld microstructures.

1.1 Weld Problems

The rapid heating and cooling cycle in the neighborhood of the welded joint causes residual stresses and strains. Furthermore, stresses are generated during the austenite-ferrite phase transformation. These residual stresses and strains may cause workpiece deflection and hence influence the serviceability of the joint. In addition, crystal growth, segregation and cellular and dendritic solidification are common problems associated with

the weld fusion zone. However, the HAZ is the principal source of trouble because of the formation of hard martensites and inhomogeneous microstructures in that region.

In general, as the hardness increases, the toughness of the HAZ decreases [1-1]. In addition to poor mechanical properties, the HAZ is also sensitive to cracking and corrosion problems [1-2]. Under-bead cold cracks form in the coarse grained region of the HAZ as it cools, see Figure 1.3. The combined effects of high tensile shrinkage stresses and hardened microstructures is believed to be the cause of this problem. Consequently, HAZ hardness is often used in the industry to provide some indication of the susceptibility of a welded joint to these problems. In most applications, HAZ hardness is limited to 320 VHN [1-2]. This is primarily because of the relationship between hardness and cold cracking is, in addition, related to the toughness and corrosion sensitivity of the welded joint.

Figure 1.3 Schematic sketch of a typical butt weld cross section showing under-bead cold cracking in the HAZ.

In addition to weld quality, the cost of fabricating a weld is another major concern for the engineer when specifying the welding procedure. To some extent, the mechanical properties (or HAZ hardness in the sense of this study) of the welded joint can be controlled by the welding parameters (arc voltage, arc current, wire traveling speed, etc.). However, the welding parameters also determine weld bead size and geometry. Increasing the heat input to the arc reduces the number of passes required to fill the gap and hence reduces production costs. However, a high heat input poses the danger of joint burn through. On the other hand, a low heat input may result in incomplete root penetration. In either case, the strength of the joint is reduced drastically.

Furthermore, some researchers have suggested that weld bead area can be used as an indirect measure of HAZ mechanical properties [1-3,4]. For example, Shultz and

Jackson [1-5] propose a linear inverse relationship between yield strength of the HAZ and the weld metal deposit area. At the same time, HAZ hardness decreases exponentially with fusion area.

From the foregoing discussion, it is apparent that determining optimal welding conditions for weld quality and productivity is complex because of the number of competing variables at work. In fact, at the moment relatively extensive field trials are necessary to fix the process in most cases. The ability to predict HAZ hardness level and weld bead geometry would reduce the number of trial runs for weld specification during the design stage of a fabrication job. This is essentially the objective of this thesis.

1.1.1 HAZ Hardness

The maximum HAZ hardness level depends on the base metal chemical composition and the weld thermal cycle (both the heating and cooling history experienced by the welded joint). Alloying elements in the base metal affect the hardenability of a steel. The most significant chemical component in this regard is carbon. So important is the carbon effect that a so-called carbon equivalent is assigned to reflect the effect of other alloying elements on the weldability of a steel. The determination of an appropriate carbon equivalent is based on regression analyses of experimental data. The two most commonly used carbon equivalent calculations are those proposed by Ito [1-6] and by the International Institute of Welding (IIW) [1-7].

The weld thermal cycle depends on the arc voltage, current, welding speed, thermal efficiency (process efficiency), plate temperature and joint geometry. It is widely believed that the temperature range from 800 to 500 °C is the most sensitive region with regard to martensite formation and in turn for the hardness level of a welded joint. Therefore, cooling time from 800 to 500 °C, $t_{8/5}$, is often used as a parameter to reflect the thermal cycle.

Several semi-empirical relationships [1-8,9] are available for predicting the HAZ hardness level of a welded joint given the carbon content, a carbon equivalent and the 800 to 500 °C cooling time.

1.1.1.1 HAZ Hardness Regression Models

Although the HAZ hardness prediction relationships [1-8,9] are relatively independent, the basic principles are similar. The maximum hardness occurs when the HAZ is a fully martensitic structure. On the other hand, the minimum hardness occurs when there is no martensite in the structure, i.e., the structure is 100% ferrite/pearlite.

Between these limits, the hardness varies according to the martensite content. A plot

Figure 1.4 Schematic HAZ characteristic hardness curve.

of HAZ hardness versus $t_{8/5}$ (on a log 10 scale) is usually used to describe the change. It is often called the characteristic hardness curve - CHC (Figure 1.4). Regression and curve fitting techniques are used to estimate the characteristic hardness curve for a given carbon equivalent and carbon content.

The two most commonly used carbon equivalents are P_{cm} - proposed by Ito-Bessyo [1-6] and CE (IIW carbon equivalent) [1-7]:

$$CE(\%) = C + \frac{Mn}{6} + \frac{Cu + Ni}{15} + \frac{Cr + Mo + V}{5} \quad (1-1)$$

$$P_{cm}(\%) = C + \frac{Si}{30} + \frac{Mn + Cu + Cr}{20} + \frac{Ni}{60} + \frac{Mo}{15} + \frac{V}{10} + 5B \quad (1-2)$$

With these carbon equivalents together with carbon content and $t_{8/5}$, Suzuki has suggested two sets of equations for predicting HAZ hardness [1-8].

The first may be referred to as the NSC-S(P_{cm}) system:

$$H_{max} = (187 + 64C + 485P_{cm}) - (97 + 680C - 441P_{cm})\tan^{-1}X \quad (1-3)$$

$$X = \frac{Y + (0.501 + 7.90C - 11.01Pcm)}{(0.543 + 0.55C - 0.76Pcm)} \quad (1-4)$$

$$Y = \log(t_{8/5}) \quad (1-5)$$

The second is the NSC-S(CE) relationship:

$$H_{max} = (192 + 397C + 153CE) - (92 + 377C - 139CE)\tan^{-1}X \quad (1-6)$$

$$X = \frac{Y + (0.385 + 0.34C - 3.47CE)}{(0.517 + 0.02C - 0.24CE)} \quad (1-7)$$

Suzuki's NSC-S(Pcm) and NSC-S(CE) models are just two of several regression models for predicting HAZ hardness. Similar relationships have been proposed by Terasaki [1-9,10,11], Suzuki [1-8,12,13], Arata [1-14], Yurioka [1-15,16], Beckert [1-17], Dueren [1-18] and Cottrel [1-19]. In the work to be presented in this thesis ANN (artificial neural network) technology is proposed as an alternative to the regression method.

1.1.1.2 800 to 500 °C Cooling Time

The weld 800 to 500 °C cooling time is shown schematically in Figure 1.5. It may be determined experimentally, calculated by employing finite element analysis [1-20] or in some cases approximated with simplified weld thermal models [1-21,22,23]. The classical heat flow approximation for the welding process was originally proposed by Rosenthal [1-24,25] and further modified by Adams et al [1-21,22] for determining the 800 to 500 °C cooling time. The Adams' cooling time models are widely used for their simplicity even though it is recognized that they are only approximate because several assumptions and simplifications inherent in the relationships. Yet HAZ hardness is

Figure 1.5 A schematic temperature-time profile of a weld thermal cycle.

sensitive to cooling times over three orders of magnitude for most steels and it is argued that considerable uncertainty in cooling time can be tolerated in most cases [1-2].

Because the heat flow mechanism changes with plate thickness, Adams [1-21,22] proposed three different cooling situations. If the arc energy input to plate thickness ratio is small, the arc can be viewed as a point source moving on the plate (Figure 1.6) and heat flows into the surroundings radially (Figure 1.7a). This is usually referred to as three-dimensional (3-D) semi-space heat flow. On the other hand, if the arc energy to plate thickness ratio is large, heat spreads laterally (Figure 1.7b) and the arc can be viewed as a line source moving within the plate (Figure 1.6). The line source situation is often referred as two-dimensional heat flow. In addition, a so-called 2.5-D heat flow is also suggested to describe the transitional behavior in a range intermediate between the 2-D and 3-D situations. In this case heat flux is visualized to be reflected from the bottom of the workpiece (Figure 1.7c).

Figure 1.6 A schematic sketch of a point, a line and an intermediate heat source.

Figure 1.7a A schematic illustration of 3-D heat flow - the thick plate situation.

Figure 1.7b A schematic illustration of 2-D heat flow - the thin plate situation.

Figure 1.7c A schematic illustration of 2.5-D heat flow - the intermediate plate thickness situation.

Cooling time relationships have been developed for the three heat flow situations; three-dimensional heat flow (thick plate) - $t_{8/5}[3-D]$, two-dimensional heat flow - $t_{8/5}[2-D]$, and transitional heat flow - $t_{8/5}[2.5-D]$, for intermediate welding cooling:

$$t_{8/5}[3-D] = \frac{\eta_{proc} Q}{2\pi k} \left[\left(\frac{1}{500 - T_o} \right) - \left(\frac{1}{800 - T_o} \right) \right] \quad (1-8)$$

$$t_{8/5}[2-D] = \frac{1}{2\pi k C_p} \left(\frac{\eta_{proc} Q}{h_t} \right)^2 \left[\left(\frac{1}{500 - T_o} \right)^2 - \left(\frac{1}{800 - T_o} \right)^2 \right] \quad (1-9)$$

$$t_{8/5}[2.5-D] = \frac{1}{C_r} \frac{\eta_{proc} Q}{2\pi k} \left[\left(\frac{1}{500 - T_o} \right) - \left(\frac{1}{800 - T_o} \right) \right] \quad (1-10)$$

Q is the welding process energy input rate per length in (J/mm):

$$Q = \frac{I_e \cdot V_o}{S_p} \quad (1-11)$$

η_{proc} is the welding process efficiency (Table 1-1 [1-15,26]), k is the thermal conductivity (a nominal value of 0.025 J/(mm.s.K) is used), T_o is the plate initial temperature, C_p is the volumetric specific heat (a nominal value of 0.006 J/(mm³K) is used), h_t is the plate thickness in mm, C_r is the ratio of the thick plate cooling time to that of the finite plate, all other conditions identical, and I_e , V_o and S_p are arc current in amperes, voltage in volts and wire traveling speed in mm/s respectively.

The appropriate relationship for any given situation is determined by a relative thickness parameter (H_r) defined as:

Table 1-1 Summary of Welding Process Efficiencies for Commonly Used Welding Heat Sources.

Process	SMAW	SAW	GMAW	GTAW	EBW
Efficiency (%)	80	95	70	40	85

$$H_r = \frac{C_p \cdot h_t^2 (650 - T_o)}{\eta_{proc} Q} \quad (1-12)$$

Three-dimensional cooling is suggested if H_r is greater than one. A two-dimensional cooling situation is appropriate if H_r is less than 0.3. Otherwise, the 2.5-D cooling relationship is considered applicable ($1.0 > H_r > 0.3$). Values of C_r may be obtained from Figure 1.8 or Table 1-2 with corresponding H_r values [1-21,22].

Figure 1.8 Values of C_r corresponding to H_r for computing cooling times in the transitional 2.5-D heat flow situation.

Table 1-2 Tabulated Values of C_r Corresponding to H_r .

H_r	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
C_r	0.30	0.43	0.63	0.76	0.85	0.94	0.94	0.97

In this thesis, an alternative (ANN - artificial neural network analysis) is proposed for determining $t_{8/5}$.

1.1.2 Weld Bead Geometry

Variations in welding conditions (arc current, arc voltage, wire travel speed, electrode extension, electrode diameter, electrode polarity, plate thickness and shielding gas) affect the weld bead shape. Furthermore, heat flow patterns during solidification and the surface tension of the molten weld metal increase the complexity of the formation of the

Figure 1.9 Definitions for describing weld geometry.

weld bead. Since the solution of a physics based mathematical model detailed enough to predict weld bead geometry is complex and the variables involved are highly coupled, several researchers have resorted to regression techniques (statistical analysis) based on large experimental data bases, e.g., Chandel et al [1-27,28]. Various critical dimensions (bead width, bead height, penetration, deposit area and fusion area) are determined, each as a function of the welding parameters (see Figure 1.9). The weld bead shape is approximated by fitting different curves (e.g., ellipse, quadratic or cosine curves) to the predicted dimensions (bead height, bead width, penetration, fusion area, etc.) [1-29,30,31,32].

In general, it is assumed that the alloying materials in low alloy steels have minimal effect on the weld bead geometry [1-33]. This author notes that such an assumption is not always justified although it is necessary at this point in research time to be able to proceed. Refinements to weld bead models will be possible as the effect of alloying is better understood. However, in general weld bead geometry depends on the fusion temperature, surface tension and thermal conductivity which in most cases are relatively independent of the alloying for low alloy structural steels.

1.1.2.1 Weld Bead Geometry Regression Models

As mentioned above Chandel et al [1-27,28] have proposed a series of regression models for predicting weld bead dimensions from welding conditions based on an enormous data base. Their work was mainly concerned with gas metal arc (GMA) bead-on-plate welds - M-2 (2% oxygen and 98% argon) or C-25 (25% carbon dioxide and 75% argon) shielding gases. The arc energy input was fixed at 3 kJ/mm. Most of Chandel's regression models are based on logarithm formulations as follows:

$$\text{dimension} = 10^k \cdot I_e^a \cdot V_o^b \cdot S_p^c \cdot D_e^d \cdot L_e^e \quad (1-13)$$

where 'dimension' is either bead width (BW) in mm, bead height (BH) in mm, penetration (Pene) in mm or total fusion area (A) in mm², and I_e , V_o , S_p , D_e , L_e are welding current (amperes), arc voltage (volts), wire travel speed (mm/s), electrode diameter (mm) and

electrode extension (mm) respectively (Figure 1.9). k , a , b , c , d and e are constants which depend on electrode polarity and shielding gas composition as listed in Table 1-3. The melting rate (MR) in kg/hr is estimated by:

$$MR = p \cdot I_c - q \cdot I_c \cdot V_o + r \cdot \frac{I_c^2 \cdot L_e}{D_e^2} + s \quad (1-14)$$

where p , q , r , s are constant listed in Table 1-4.

Table 1-3a Coefficients for Chandel's Weld Bead Dimension Models
(C-25 Shielding Gas and +ve Electrode Polarity).

Dimension	k	a	b	c	d	e
A	-2.350	1.689	0.260	-0.850	-0.630	0.2535
BW	-0.218	0.181	0.860	-0.614	0.567	0.0106
BH	-1.382	1.200	-0.690	-0.450	-1.360	0.3800
Pene	-4.030	2.050	0.142	-0.530	-0.860	-0.0630

Table 1-3b Coefficients for Chandel's Weld Bead Dimension Models
(C-25 Shielding Gas and -ve Electrode Polarity).

Dimension	k	a	b	c	d	e
A	-1.517	1.377	0.271	-0.905	-0.298	0.1900
BW	-1.500	0.520	0.272	-0.570	0.275	0.0001
BH	-0.460	0.690	-0.460	-0.360	-0.660	0.3370
Pene	-3.250	1.740	-0.093	-0.366	-0.460	-0.0630

Table 1-3c Coefficients for Chandel's Weld Bead Dimension Models
(M-2 Shielding Gas and +ve Electrode Polarity).

Dimension	k	a	b	c	d	e
A	-2.290	1.615	0.202	-0.835	-0.680	0.3400
BW	-1.004	0.534	0.660	-0.307	0.146	0.0258
BH	-1.440	1.070	-0.540	-0.450	-1.370	0.4900
Pene	-3.500	0.535	2.610	-0.340	-0.876	-0.6890

Table 1-3d Coefficients for Chandel's Weld Bead Dimension Models
(M-2 Shielding Gas and -ve Electrode Polarity).

Dimension	k	a	b	c	d	e
A	-0.544	1.123	-0.0378	-0.815	0.0320	0.166
BW	-0.900	1.030	-0.2300	-0.550	-0.0052	0.128
BH	-0.380	0.235	-0.0850	-0.320	-0.2070	0.120
Pene	4.420	1.246	1.8152	-0.183	-0.2360	-0.567

Table 1-4 Coefficients for Chandel's Wire Melting Rate Equation.

Shielding Gas	Electrode Polarity	p	q	r	s
C-25	+ve	0.0230	0.000460	0.00000306	-0.50
C-25	-ve	0.0195	0.000158	0.00000100	0.39
M-2	+ve	0.0200	0.000540	0.00000100	-0.56
M-2	-ve	0.0268	0.000360	0.00000100	0.11

The deposit area (A1) in mm² can be calculated from the relationship:

$$A1 = 35.47 \cdot \frac{MR}{S_p} \quad (1-15)$$

The plate fusion area (A2) mm² is calculated by subtracting the deposit area from the total fusion area (A):

$$A2 = A - A1 \quad (1-16)$$

Several other investigations have been conducted for the purpose of predicting the weld size, given the welding conditions. For example, Chandel et al [1-33,34,35] and Yang et al [1-31,36,37] have developed similar models for predicting submerged arc (SA) bead-on-plate welds. McGlone et al [1-31,38] suggest relationships for predicting submerged arc weld size with edge preparation (Y-groove). Quintino et al [1-39] and Raveendra et al [1-40] propose models to cover weld bead geometry for pulsed GMAW and flux cored arc welds respectively.

In this thesis, ANN (artificial neural network) methodology is applied to this problem.

1.2 Software for Predicting Weld Features

During this author's masters research program - "Software for Welding Engineers" [1-29,30,41,42,43], software based on available regression models was developed to permit welding engineers to predict as-welded HAZ hardness (WH), post-welded heat treated HAZ hardness (RH), preheat temperature to avoid under-bead cold cracking (PH), and gas metal arc (GMAW) bead-on-plate weld size and shape (WF). The as-welded HAZ hardness module was based on the experimental work of seven independent research groups (Suzuki et al [1-8,12,13], Arata et al [1-14], Terasaki et al [1-9,10,11], Yurioka et al [1-15,16], Beckert et al [1-17], Dueren et al [1-18] and Cottrell [1-19]). The post-welded heat treated HAZ hardness [1-44] and preheat [1-45,46,47,48] modules were based on the work of Yurioka et al. The GMAW weld features module was based on the work of Chandel et al [1-27,28]. It was considered important that the system should be user-friendly, i.e., easy to learn and use in practice. The system is basically function key or mouse driven and the practitioner requires only a basic personal computer knowledge to manipulate the software.

In order to strengthen the user-friendly graphics environment, a series of enhancements have been made since finishing the masters thesis. First, new modules for predicting submerged arc weld (SAW) size and shape with or without groove preparation have been added [1-31,32]. All modules (GMAW weld shape, SAW weld shape, as-welded HAZ hardness, post-welded heat treated HAZ hardness and preheat temperature to avoid cold cracking) have been integrated into a single menu-driven (mouse or function key) colour graphics computer environment which has been termed "WeldSoft" for commercial purposes [1-49]. *Currently, the commercial version of WeldSoft is being distributed in the United States through the American Welding Society and in Canada by the Pipe-Protech Corrosion Survey Inc. (PCS), Nepean. European and South African markets are handled by Gullco International Ltd. Comments on the value of the software to welding engineers in the form of supporting letters from the Canadian Welding Development Institute are attached to this thesis as Appendix I.*

1.2.1 Limitations of the Regression Models

Although the regression models designed to predict HAZ hardness level and weld bead geometry are reasonably comprehensive, there are some limits in practice. First, the algorithms are relatively lengthy and troublesome to apply. WeldSoft referred to above was generated for the purpose of making these models more useable for practitioners [1-29,30,41,42,43].

The second limitation is that in order to generate a regression model, the researcher must understand both the physical problem and regression techniques very well. The number of sensitive variables used in the analysis must be determined in advance. Non-sensitive parameters in the regression analysis may increase the complexity and in fact can reduce the accuracy of the final formulation [1-37]. The usefulness of the final model is usually restricted by the regression limits and the number of parameters. Consequently, the development of a regression model usually involves considerable expertise. Modifying a regression model to extend its range of applicability or otherwise improve its performance by changing either the regression limits or the number of parameters usually means re-analyzing the entire problem. As a result, it can be said that the model creation cycle is long but the life cycle of a model may be relatively short.

Finally, most models discussed previously require some pre-defined input-output relationships and assumptions. For example, the weld bead geometry relationship is often assumed to be logarithm-like and the HAZ hardness relationship usually has a backward logistic form. Furthermore, the assumptions necessary to calculate a reasonable cooling time that the thermal conductivity, k , and volumetric specific heat, C_p , are temperature independent constants together with the need to estimate relative thickness, H_r , render these relationships more suspect.

1.3 Artificial Intelligence and Artificial Neural Networks

Recent developments in artificial intelligence provide an attractive alternative to conventional regression techniques. Artificial intelligence (AI) has been defined under

certain circumstances as the study of human faculties through the use of computational models [1-50]. In some cases, AI researchers attempt to duplicate human intelligence (e.g. vision, natural language, learning, reasoning, etc.), imitate thinking and problem solving patterns. There are several distinct areas of AI research. The most significant developments include problem solving, logical reasoning, natural language processing, learning, expertise and vision. In the context of this work, learning through artificial neural networks (ANN) is the main concern.

ANN is a mathematical foundation that is said to emulate neural behavior. Originally, the basic concept of the model was to simulate the massive parallel processing units (neurons) of the brain. In mathematical terms, ANN can be best described as a highly complex, non-linear mapping function which transforms input to output in a given domain. ANN is the most suitable AI technique for handling situations such as pattern matching, data mapping and classification problems.

Recent ANN successes for solving non-trivial problems inspired the use of ANN for predicting weld size [1-51,52]. Although an ANN model may be very complex, the basic mathematics is relatively simple. Moreover, the end-user of ANN technology does not require specialized background to be able to apply the technology in practice. An ANN model may be viewed as a computational 'black box' which stores applicable knowledge for solving a problem. To the end-user (practitioner), the internal configuration of the black box is not a major concern; rather the prediction capability of the system is more important. With appropriate software, a practitioner can do the analysis with only basic ANN knowledge.

Knowledge of an ANN system is acquired through 'learning'. Therefore, a set of examples, termed the training data base, are required for learning. Similar to a conventional regression analysis, a sample data base covering the entire problem domain must be available. With sufficient learning, the network ascertains the relationship between the input (e.g., welding parameters) and output parameters (e.g., HAZ hardness level). In

some sense, ANN can also be described as a very complex non-linear statistical model with less pre-determined variable constraints than regression analysis.

The major advantage of ANN for users is the training concept. In conventional regression analysis, the relationship of input and output sets are determined and automated for the user's convenience by researchers. Whenever a new set of data is presented or added to the original data set, the analysis and the program amendments are usually conducted by someone other than the user. On the other hand, the ANN model may be generated and/or re-trained by the user if the appropriate software is at hand. In other words, the user has a computational black box which can be trained to handle new data by the end-user directly. While knowledge is built into a regression model directly, learning ability is built into an ANN program. Therefore, updating a model by the end-user may be continuous and the model life cycle is not usually an issue. Because of this, ANN modelling is said to be more flexible and to require less researcher involvement once it has been activated. Furthermore, with ANN, control of the complexity of the modelling relationship is easier to manage than it is with conventional regression analysis. Instead of pre-defined equations as required by regression technology, the complexity of the modelling relationship is represented by the network structure (a collection of procession units). As a result, ANN is more flexible than regression analysis.

There are two research investigations published in the literature that have been reported where ANN is used to predict to weld geometry [1-51,52]. The work of Andersen et al [1-51] based on gas tungsten arc weld (GTAW) may be the first attempt. This group used ANN to determine the relationship between the input welding conditions (voltage, current, electrode travel speed and wire feed speed) and weld dimensions (bead width, penetration, bead height and total fusion area). On average, predicted dimensions were said to be within 10%. The weld shape may then be approximated by fitting the predicted weld dimensions to pre-determined curves. Using a similar approach, Jones et al [1-52] generated a weld shape system, "WeldBead". Knowledge for WeldBead was a data base of on twenty-eight flux-cored arc (FCAW) L-joint fillet welds and GTAW square

butt welds. Furthermore, WeldBead includes an inverse network, i.e. the network can determine the required welding conditions by supplying the critical weld dimensions [1-52], i.e., bead width, bead height and penetration. However, using the three suggested weld dimensions to determine the welding parameters may not be sufficient nor convenient to a practitioner because the weld bead shape also depends on the fusion area and plate thickness.

1.4 Objective

In a general sense, the research described in this thesis is an extension of this author's masters research - a system for predicting HAZ hardness and weld bead size and shape. That work was based, for the most part, on regression analyses of large experimental data bases for this purpose. It is the contention of this author that ANN (artificial neural network) techniques can be used to provide a better, more convenient, more flexible system for this task. In particular, it should be possible to refine and extend the range of applicability of a system as data becomes available. It may also be possible for a practitioner to adapt the system to a particular job shop or manufacturing operation without the necessity of a lengthy and uncertain regression analysis.

In particular, an ANN system is to be generated for predicting HAZ hardness and weld size and shape given the input welding parameters. Data extracted from the investigations of Yurioka [1-15] - see Appendix II, Rodrigues [1-53], Van Adrichem et al [1-54], Bibby et al [1-55] and Bala et al [1-56] - see Appendix III, are used as training and test data for predicting HAZ hardness, given the welding parameters. The independent works of Kohno et al [1-57], Signes [1-58] - see Appendix IV, and AMCA [1-56] - see Appendix V, are used to further test the system.

Data generated from Northern College in Kirkland Lake [1-59] is to be used to train and test the system for predicting weld size and shape - see Appendix VI. Specifically the data refer to gas metal arc (GMA) welds with C-25 (25% carbon dioxide and 75% argon) and M-2 (2% oxygen and 98% argon) shielding.

Chapter Two

Artificial Neural Networks

2.1 Background

Artificial neural network (ANN) is one of the major fields of interest in current artificial intelligence (AI) research. There are two main thrusts in the study of ANN. Some researchers use it to understand and simulate the human nervous system, while others apply it to solve physical problems [2-1,2].

In biological terms, the neuron is the basic building block of the nervous system [2-2,3]. In general, a neuron is made up of a central cell body, dendrites and an axon, as shown in Figure 2.1. The point where the termination of the axon of one neuron comes into close proximity to the dendrites of other neurons is called synapse. A synapse is a microscopic gap.

Figure 2.1 A schematic sketch of two neurons in a nervous system.

Information or signals are collected from the dendrites and manipulated by the cell body. The out-going message is then relayed to the axon and passes on to the dendrites of other neurons. The information is transmitted in the form of micro-electrical signals within the cell. The cell body acts as an information gathering and filter centre in which the in-coming signals are summed. If the resultant is larger than a certain threshold, an out-going message is generated, i.e. the neuron is activated. The message from the activated neuron is passed on to other neurons through the axon, synapse and dendrites. At the synapse, an electrical signal from the axon is changed to a chemical signal. The synapse has a strength

which assigns the importance of the travelling signal. The signal is either excitatory or inhibitory at the synapse. The chemical signal is then changed back to an electrical signal at the dendrites and the signal passes on.

ANN is a crude duplication of the sophisticated biological nervous system described above. The neuron is represented by a processing unit termed a node. An ANN system is made up of a collection of nodes connected together by links.

Figure 2.2 A typical artificial neural network.

A weight is assigned to each link which serves as the function of the synaptic strength, as shown in Figure 2.2. The relationship of the two connected nodes depends on the weight of the link. Moreover, knowledge is also stored as weight. In general, information is entered from the input nodes. This information is transmitted through the links to the hidden nodes. The weighted information is summed for each individual hidden node. If the sum is greater than the threshold of that node, then the node fires.

Although there are similarities between ANN and the biological nervous system, there are also many differences. This is not surprising because there is an incomplete understanding of the biological process. Furthermore, there are many kinds of neurons in a biological system and these have not been characterized to the point where they can be simulated. The neuron system to be implemented in this thesis is a simple one. The number of neurons in a biological neural network is significantly larger than those in any ANN system. One of the major differences between a biological neural network and ANN is the learning speed of the two networks. In general, computer clock speed is about one

thousand times faster than the human brain. However, the human system is still able to outperform a digital computer primarily because of the brain's parallel architecture [2-4].

2.2 History

Neural networks originated as a crude model of how the brain works in the study of psychology. William James [2-5], a psychologist, and other researchers postulated the basis for early neural networks in the nineteenth century. The first artificial neural network model was proposed by McCulloch and Pitts [2-6,7] in 1943. However, their digital neurons did not have the very important learning ability of contemporary systems. In 1949, Hebb [2-8,9] introduced the idea of Hebbian learning which states that synaptic strengths are proportional to the activation of the connected neurons (see also Table 2-1). This is the basis of learning in modern ANN technology.

Table 2-1 Hebbian Learning.

Hebbian Learning:

- 1) If nodes A and B are simultaneously excited, the strength of the link between the two nodes is increased.
- 2) On the other hand, if the product of their activation is negative, the strength of the link between the two nodes is decreased.

In 1957, Rosenblatt [2-10,11,12] incorporated Hebbian learning into his ANN system. Such a network is called a perceptron. However, many problems cannot be solved with the perceptron network. A perceptron fails to learn in solving non-linear problems which is a major limitation. In 1969, Minsky and Papert [2-13] stated the limitations of the perceptron network and concluded that it was not a fruitful area of research. As a consequence, there was limited ANN research in the 1970's. It was not until 1982 and Hopfield's work [2-14,15,16] that ANN research was re-ignited. Grossberg [2-17,18] and Kohonen [2-19,20] have emphasized self-organizing networks to carry out unsupervised

learning. Kohonen [2-20] built a content-addressable memory network, in which separate indexing was not necessary to retrieve a particular item and it was referred to as associative memory.

The discovery of backpropagation by Werbos [2-21] in 1974 (termed dynamic feedback) and Parker [2-22] in 1982 (termed learning logic) was the next major advancement in ANN after Rosenblatt's perceptron. Backpropagation learning allows multi-layer network learning. In 1986, Rumelhart, Hinton and Williams exploited these algorithms in simulating the cognitive process [2-23,14,25]. Backpropagation learning is a powerful and practical tool for solving some difficult computer science problems, e.g. image processing, speech recognition, forecasting and optimization.

Since the emergence of backpropagation, many other models have been developed. In an extension of Kohonen's proposal, Kosko [2-26,27] developed a more powerful network that incorporated bi-directional associative memory (BAM), in 1988. In 1987, Hecht-Nielsen [2-28] combined backpropagation and competitive learning to initiate counter-propagation network which is said to be a faster learning algorithm.

Current ANN research is often limited by computer hardware. Many of the proposed algorithms are not practical for sequential digital computers. Implementing ANN algorithms on a parallel computer is another major degree of research freedom in ANN research.

There has been enormous progress in the use of ANN as a problem-solving tool in recent years. ANNs are used in the technical analysis of stocks and commodities, image recognition, speech recognition, credit card fraud detection, bomb detection systems, and many other applications [2-29].

2.3 Properties of ANN

Unlike most other AI methods, ANN requires a rigorous, mathematical foundation. The basic mathematics required has been relatively well established which

permits the formulation of algorithms for solving problems such as the prediction of weld shapes suggested in this thesis.

The processing sequence of ANN is at the same time parallel and simultaneous. Computation is distributed over more than one processing unit and is done simultaneously. Although, a sequential digital computer has been used in much of this work to simulate parallelism, there are many benefits. A sequential ANN model can isolate problems in at early model design stage as a result of the simplified approximation of the performance of a particular ANN model for a given application. However, only true neural network hardware can perform the operations in parallel with fast learning.

Knowledge is stored in the network as weights which are distributed throughout the system. Moreover, memory in ANN is associative. The network maps and associates data to data directly. Therefore, ANN is exceptionally good for pattern recognition but at the same time requires the computer to store a large number of patterns. In some sense, ANN does not compute the answer, it memorizes and generalizes the solution. Since knowledge is distributed in the system, ANN is fault tolerant. ANN can learn and decide, even if there is incomplete input data. However, the user must be aware of the limitations that this imposes.

The ability to learn in ANN has created intense research interest. Learning occurs when the weights in the network are adjusted in response to the training data. The weights and knowledge, can be stored apart from the network for a particular situation. Hence, an ANN can be used to solve different kinds of problems by applying the appropriate knowledge.

The ability to generalize with ANN is an interesting facility which allows the network to respond to input that the network has never experienced. Generalization can also be described as the ability to hypothesize a response. This ability permits the network to handle changing or unexpected environments. Generalization is also important for pattern recognition.

Like most other computer tools, there are many problems that ANN cannot handle. ANN does not provide precise answers. Furthermore, the results of an ANN computation cannot be justified, i.e. reasoning of the solution is not available. Learning is hard and time consuming. Since knowledge is distributed in the network, an error that occurs in prediction is difficult to correct locally. Usually the network needs to be re-trained. Finally, designing an ANN is still something of an art requiring a lot of trial and error before an effective algorithm is generated (to be discussed later).

2.4 Basis of ANN

As mentioned previously, the basic element of an ANN is the processing unit, i.e. the node. The input signal is evaluated at the node. The strength of each signal is determined. Signals are combined and the result is compared to a certain threshold level in order to determine the output. Therefore, a node is logically broken down into two functions: summation and transfer. The summation function handles the combining of weighted input signals and the total is evaluated by the transfer function. The threshold is represented by a transfer function. A transfer function can be a step function (output is restricted to $\{0,1\}$ or $\{-1,1\}$), a ramping function (a linear function that has been clipped to minimum and maximum values), a sigmoid or S-shaped curves. Such curves approach a minimum and maximum values at the asymptotes. The sigmoid and S-shaped curves are the most interesting since both the functions and their derivatives are continuous which is similar to biological response.

The inputs and outputs are normalized to values between zero and one which provides flexibility. In addition to the input data, a bias input is usually introduced to stabilize the system. Weights are also important to the network because they determine the impact of the input data. Moreover, weights are adaptive which enables learning. The weights are modified in response to various input values according to the network's learning rules or error correction algorithms. Learning may be viewed as the process by which a neural network weights are adjusted in response to external input.

2.4.1 ANN Learning

There are two types of learning used in ANN technology: supervised and unsupervised [2-4]. For supervised learning, the desired output is compared to the actual output of the ANN with identical input data. The weight of each link is adjusted according to the difference of the desired/computed output, to produce a closer match in the next iteration, i.e. the goal of supervised learning is to minimize the error between the desired output and the network output by continuously modifying the weights. Therefore, the training of a network consists of presenting input and output data to the network and adjusting connection weights. The training continues with the same set of input and output data for the purpose of reducing errors. The training is considered complete when the ANN produces a computed output with acceptable accuracy. At this time, the network is said to have acquired enough knowledge to solve a specific problem.

On the other hand, unsupervised learning requires no desired output comparison. Therefore, it is sometimes called self-supervised learning. Instead of comparing the computed output with known results, the network depends on regularities, similarities or trends in the input data, and adapts according to the function of the network, i.e. the network seeks natural demarcation amongst the input data. Learning is often made possible through some notion which is featured in the input set. Often, the features that are important, are not known in advance. Competitive learning is a biological-like unsupervised learning. Competition between processing units generates a natural impetus for learning. Based on the input pattern, one (or a few) output neurons are chosen as winners and only the weights associated with the winners are adjusted.

2.4.2 Connecting Nodes Together - Paradigms

One signal neuron is insignificant to the nervous system. Similarly one signal processing unit in ANN is not enough to solve a physical problem. In order to solve a problem, nodes are connected together, given one or more learning rules to form a

network. Different constructions (learning rules and nodal architectures) of such networks generate different ANN paradigms with characteristic behaviour. Furthermore, paradigms are good at solving different types of problems, see Table 2-2 for some examples [2-29]. Some well known paradigms will be described briefly, later in this Chapter.

Table 2-2 Sample Applications of Different Paradigms.

<u>Application</u>	<u>Paradigm</u>
Prediction	Backpropagation, Perceptron
Classification	Counter-propagation, Kohonen's net
Data Association	Hopfield net

Usually nodes are organized in layers. Such layers are classified into three main categories: input, hidden and output layers. The input and output layers are connected to the outside world while the hidden layer or layers handles all the computations. In some ways, it may be viewed as a black box or an internal representation of the solution.

2.4.2.1 Perceptron

The perceptron was the earliest useful ANN paradigm developed by Rosenblatt [2-9,10,11]. It is a two-layer network which was implemented through an electrical circuit, see Figure 2.3. A perceptron simulates a neuron by taking a weighted sum of its input and generating the output 1 if the sum is greater than the adjustable threshold value. The links are unidirectional and there is usually no interconnection between nodes within the same layer.

Figure 2.3 A schematic illustration of a multi-perceptron network.

Learning is the process of modifying the weights and threshold values. A perceptron network can learn to solve a problem, i.e. it can generalize and appropriately respond to stimuli it has never experienced. However, it is useful only for linear problems.

2.4.2.2 Hopfield's Network

Hopfield's network [2-13,14,15] is a fully interconnected signal-layer auto-associative pattern storage mechanism. Moreover, all connections are coupled with identical weights, as shown in Figure 2.4. The network shown in Figure 2.4 can store patterns containing four elements. Nodes are binary (active or inactive).

Figure 2.4 A 4-node Hopfield network which can store patterns consisting of 4 binary fields.

Nodes have positive weighted linking which tends to activate one another. Hopfield's net utilizes parallel relaxation for storing patterns. For parallel relaxation, a randomly selected node is activated if the weighted sum of all connected active neighbours of that node are positive; otherwise the node will stay inactive. Then, another random node is chosen and analyzed. The process continues until there is no change in state for any node.

2.4.2.3 Backpropagation Network

A similar algorithm for computer learning, backpropagation, has been proposed by several different individual researchers [2-21,22,23,24,25]. Backpropagation is a natural successor to the perceptron network. It is a multi-layer network, consisting of input and output layers with one (or more than one) hidden layer, as shown in Figure 2.5. Input layer nodes are connected to hidden layer nodes by weighted unidirectional links. Hidden layer nodes are connected to output layer nodes in a similar manner. If there is more than one layer of hidden nodes, the input nodes are connected to the first layer of hidden nodes. The first layer of hidden nodes are connected to the second layer of hidden nodes and so on. Finally, the last layer of hidden nodes are connected to the output nodes. There are no

interconnections within the same layer. Input data are fed forward to the network to compute an output. The network output is compared with a desired output and the comparison, error, is then back propagated. The weights are adjusted according to the magnitude of the error. The learning stops when the error is within tolerance. However, some difficulties are associated with backpropagation. Learning is slow and there is no promise that the learning will converge. The search for a minimum error may get trapped in local minima in weight-error space when the system finds an error that is lower than the surrounding but it may not represent the smallest possible error.

Figure 2.5 A schematic sketch of a 3-layer backpropagation network.

2.4.2.4 Self-Organizing Map

By utilizing the idea of competitive learning, Kohonen [2-19,20] developed a self-organizing network which learns without being given a target for an input pattern. The network is usually composed of two layers: an input layer and a competitive layer (also called Kohonen layer). The two layers are fully interconnected. However, there are no interconnections between units in the competitive layer. The winner, i.e. unit with the largest output, has the capability of inhibiting its competitors as well as exciting its neighbours. Only the winner generates an output and only the weights of the winner and its neighbours are adjusted.

2.4.2.5 Counter-propagation Network

If a set of input data is composed of reasonably distinct groupings, a counter-propagation network can be employed as a classifier. Counter-propagation network methodology was proposed by Hecht-Nielsen [2-28]. It apparently learns faster than the backpropagation network and is capable of a generalized solution. The network uses both supervised (Grossberg learning [2-17,18]) and unsupervised (Kohonen competitive learning [2-19,20]) learning. A typical counter-propagation network consists of three layers: an input layer, a middle Kohonen layer and an output Grossberg layer. Each layer is fully interconnected to the layer above but there is no interconnection within the same layer. Training of a counter-propagation network is similar to the training of a backpropagation scheme. A set of input/desired output pairs are iteratively exposed to the network until an acceptable accuracy is attained. Only the winner of the Kohonen (competitive) layer generates an output value of one, all other nodes are zero. The Grossberg layer's nodes have their weights adjusted with each pattern association based on the activation of the Kohonen layer's winners and the output so that the calculated output better agrees with the desired output. The training is fast because only the weights associated with the Kohonen layer's winners are adjusted and the rest are untouched. However, the result generated by the counter-propagation network is usually not as accurate as the one predicted by backpropagation.

Chapter Three

Backpropagation ANN Scheme

3.1 Introduction

The significant feature of a backpropagation network (BPN) is that, as it trains, the weight of the links between the layers adjust such that different links learn to recognize different features from the input. When the trained network is presented with an arbitrary input pattern (incomplete pattern and/or with noisy data) the nodes in the hidden layers will respond with an output if the input pattern contains information that resembles the features that the individual nodes have learned. On the other hand, nodes in the hidden layers inhibit output if the input pattern does not contain the feature that the network was trained to recognize. Therefore, the output pattern can be referred to as a feature map that provides an indication of the presence or absence of many different input patterns. This property of the backpropagation network is crucial for predicting weld features (t_s , HAZ hardness and weld bead geometry) from welding conditions. Weld features and welding condition relationships contain certain irregularities which may be identified as noise in the network. The uncertainties in the welding conditions (variations in the input voltage, current, welding speed, etc.) may be seen as incomplete input patterns. Furthermore, a good sense of topology (input and output patterns and the number and size of hidden layers) appropriate to the problem is essential before using the network.

3.2 Theory

The backpropagation network shown in Figure 3.1 is a schematic representation of a three-layer backpropagation network. The input pattern contains h parameters and the output pattern contains j parameters. Therefore, the input (H) and output (J) layers consist of h and j nodes respectively and the number of nodes in the hidden layer (I) is i . The weight of the links connecting between the input (H) - hidden (I) layers and between the

hidden (I) - output (J) layers are denoted as W_{ih}^I and W_{ji}^J respectively. Each of the n -th pattern prepared for training consists of an input pattern ($X^{Pa}_1, X^{Pa}_2, \dots, X^{Pa}_h$) and a matching output pattern ($Y^{Pa}_1, Y^{Pa}_2, \dots, Y^{Pa}_j$).

Figure 3.1 A three-layer backpropagation network with input (X), network output (J) and desired output (Y) data.

The learning of a BPN is composed of two stages: forward and backward propagation. During the forward propagation, input signals are analysed by the current network internal representation (weights) and hence the network responses are generated. Therefore, this is also the prediction function of a BPN. In the backward propagation, the network responses are compared with desired outputs. Weights in the hidden-output layers are adjusted according to the results of the comparison. Furthermore, the errors (results of the comparison) are back-propagated to the input-hidden layer for weight modification. Therefore, a back-propagation network is sometimes referred to as an error propagation network.

3.2.1 Forward Propagation

In general, a node carries out two functions: summation and transfer (Figure 3.2). A weighted sum is processed through the transfer function and forwarded to the nodes in the next layer. The weight of the links connecting the input signals and the nodes in the input layer are all equal to one, i.e. $W_{h-input} = 1$, and the transfer function of

Figure 3.2 The summation and transfer functions of a node in the network.

the nodes in the input layer is unity. Therefore, the activation of the nodes in the input layer is identical to the input values, i.e. $H_h = X_h^{P_n}$. Both input (X) and output (Y) values are normalized values, i.e. $0 \leq X_h^{P_n} \leq 1$ and $0 \leq Y_j^{P_n} \leq 1$.

The first pattern (P1) in the training set of p patterns is considered in the initial instance. The activation of the nodes in the input-layer (H) are assigned as:

$$H_h = X_h^{P_1} \quad (\text{for all } h) \quad (3-1)$$

The activation of the nodes in the input layer are passed on to the nodes in the hidden layer through weighted links. Usually, the initial weights of the links connecting the input-hidden and hidden-output layers are set by random values between -0.5 and 0.5 [3-3]. The activation of the nodes in the hidden layer (I) are determined by summation, equation (3-2), and the transfer functions, equation (3-3).

$$S_i^1 = \sum_{a=1}^h H_a W_{ia}^1 \quad (3-2)$$

$$I_i = F(S_i^1) \quad (\text{for all } i) \quad (3-3)$$

Computation of the activation of the nodes (J) in the output layer is similar to those of the hidden layer:

$$S_j^1 = \sum_{a=1}^i I_a W_{ja}^1 \quad (3-4)$$

$$J_j = F(S_j^1) \quad (\text{for all } j) \quad (3-5)$$

At this point, the network forward propagation is complete. In order to achieve learning, the results from the forward propagation, J_j , are compared with the desired output of the first pattern (Y_j^{P1}) to obtain the pattern error E^P . The pattern error is a function of the network and desired outputs, equation (3-6).

$$E^P = G(J_j, Y_j^P) \quad (3-6)$$

3.2.2 BPN and the Generalized Delta Rule

The learning of BPN can be viewed as an optimization problem. The objective of learning in the network is to seek a set of weights such that the total error of the network prediction is minimized. According to the standard delta rule [3-1], the weights are adjusted to reduce the pattern error. If there is no error, there is no learning necessary.

The search for the set of weights which generates a minimum error is the same as looking for a minimum in multi-dimensional weights-error space. To simplify the derivation, a two dimensional weight-error space may be considered (Figure 3.3). Line ab is constructed so that it is the slope (dE/dW) of the weight-error relationship at weight W_1 . To minimize the

error E , W_1 should move downward along the slope direction. Therefore,

$$W_2 = W_1 - \eta \frac{dE}{dW} \quad (3-7)$$

where η = step size, a small number, which is also termed the learning rate.

$$\Delta W = -\eta \frac{dE}{dW} \quad (3-8)$$

For a multi-dimensional weights-error space of the links connecting the hidden and output layers, equation (3-8) can be re-written as:

$$\Delta W_p^j = -\eta \frac{\partial E^p}{\partial W_p^j} \quad (3-9)$$

By applying the chain rule to equation (3-9), it can be re-written as:

$$\frac{\partial E^p}{\partial W_p^j} = \frac{\partial E^p}{\partial S_j^j} \cdot \frac{\partial S_j^j}{\partial W_p^j} \quad (3-10)$$

By substituting the result of equation (3-4) into the second term of the right-hand side of equation (3-10), the following is generated:

$$\frac{\partial S_j^j}{\partial W_p^j} = \frac{\partial}{\partial W_p^j} \left[\sum_{i=1}^I I_i W_p^j \right] \quad (3-11)$$

Because of the partial derivatives in equation (3-11), only the term $I_i W_p^j$ yields a non-zero term.

$$\frac{\partial S_j^j}{\partial W_p^j} = \frac{\partial I_i W_p^j}{\partial W_p^j} = I_i \quad (3-12)$$

Now considering the first term of the right-hand-side of equation (3-10), $\partial E^p / \partial S_j^j$, and applying chain-rule yields:

$$\frac{\partial E^p}{\partial S_j^j} = \frac{\partial E^p}{\partial J_j} \cdot \frac{\partial J_j}{\partial S_j^j} \quad (3-13)$$

Substituting the result of equation (3-5) into the second term of the right-hand-side of equation (3-13) produces:

$$\frac{\partial J_j}{\partial S_j^j} = \frac{\partial F(S_j^j)}{\partial S_j^j} = F'(S_j^j) \quad (3-14)$$

Next, let the error function E^p equal to half the square of the distance between the network and desired outputs. The assignment of the error function is called the least square or gradient descent method. It may be expressed as follows:

$$E^p = \frac{1}{2} (Y_j^p - J_j)^2 \quad (3-15)$$

By taking the partial derivative of E^p with respect to J_j :

$$\frac{\partial E^p}{\partial J_j} = -(Y_j^p - J_j) \quad (3-16)$$

Substituting equations (3-14) and (3-16) into (3-13):

$$\frac{\partial E^p}{\partial S_j^I} = -(Y_j^p - J_j)F'(S_j^I) \quad (3-17)$$

Substituting equations (3-12) and (3-17) into (3-10):

$$\frac{\partial E^p}{\partial W_{jp}^I} = -(Y_j^p - J_j)F'(S_j^I)I_p \quad (3-18)$$

The terms $(Y_j^p - J_j)F'(S_j^I)$ are usually labelled as delta, δ_j^I :

$$\delta_j^I = (Y_j^p - J_j)F'(S_j^I) \quad (3-19)$$

and also:

$$\delta_j^I = -\frac{\partial E^p}{\partial S_j^I} \quad (3-20)$$

Using the results of equations (3-18) and (3-19), the adjustments of the weights of the links connecting the hidden and the output layers can be re-written as:

$$\Delta W_{jp}^I = \eta \delta_j^I I_p \quad (3-21)$$

However, in order to adjust the weights between the input-hidden layers, the pattern errors need to be back-propagated. Therefore, BPN is also sometimes termed as an error propagation network. The derivation of the equations used for adjusting the weights of the connections between the input-hidden layers is similar to those for adjusting the hidden and the output layers. The change in weight is proportional to the partial derivative of the pattern error:

$$\Delta W_{ih}^I = -\eta \frac{\partial E^p}{\partial W_{ih}^I} \quad (3-22)$$

Applying the chain-rule to the right-hand-side of equation (3-22) results in:

$$\frac{\partial E^p}{\partial W_{ih}^I} = \frac{\partial E^p}{\partial S_i^I} \cdot \frac{\partial S_i^I}{\partial W_{ih}^I} \quad (3-23)$$

The second term on the right-hand-side of equation (3-23) can be re-written as:

$$\frac{\partial S_i^I}{\partial W_{ih}^I} = \frac{\partial}{\partial W_{ih}^I} \left[\sum_{a=1}^h H_a W_{ia}^I \right] \quad (3-24)$$

Equation (3-24) can be simplified to:

$$\frac{\partial S_i^l}{\partial W_{ih}^l} = H_h \quad (3-25)$$

Expanding the first term on the right-hand-side of equation (3-23):

$$\frac{\partial E^p}{\partial S_i^l} = \frac{\partial E^p}{\partial I_i} \cdot \frac{\partial I_i}{\partial S_i^l} \quad (3-26)$$

With equation (3-3), the second term of equation (3-26) can be re-written as:

$$\frac{\partial I_i}{\partial S_i^l} = \frac{\partial F(S_i^l)}{\partial S_i^l} = F'(S_i^l) \quad (3-27)$$

and the first term of equation (3-26) can be re-written as:

$$\frac{\partial E^p}{\partial I_i} = \sum_{\text{for all } j} \frac{\partial E^p}{\partial S_j^l} \cdot \frac{\partial S_j^l}{\partial I_i} \quad (3-28)$$

From equation (3-20), it is evident that the first term of equation (3-28) is equal to $-\delta_i^l$, and the second term can be re-written as:

$$\frac{\partial E^p}{\partial I_i} = \frac{\partial}{\partial I_i} \sum_{a=1}^j (I_a W_{ia}^l) = W_{ip}^l \quad (3-29)$$

Therefore, equation (3-28) can be re-written as:

$$\frac{\partial E^p}{\partial I_i} = - \sum_{a=1}^j (\delta_a^l W_{ia}^l) \quad (3-30)$$

From equations (3-27) & (3-28), equation (3-26) can be expressed as:

$$\frac{\partial E^p}{\partial S_i^l} = -F'(S_i^l) \sum_{a=1}^j (\delta_a^l W_{ia}^l) \quad (3-31)$$

From equations (3-25) & (3-31), equation (3-23) can be re-written as:

$$\frac{\partial E^p}{\partial W_{ih}^l} = -F'(S_i^l) \sum_{a=1}^j (\delta_a^l W_{ia}^l) H_h \quad (3-32)$$

Equation (3-32) can also be expressed in delta (δ^l) form:

$$\delta_i^l = F'(S_i^l) \sum_{a=1}^j (\delta_a^l W_{ia}^l) \quad (3-33)$$

and equation (3-32) can be written as:

$$\frac{\partial E^p}{\partial W_{ih}^l} = -\delta_i^l H_h \quad (3-34)$$

Substituting equation (3-34) into equation (3-22):

$$\Delta W_{ih}^l = \eta \delta_i^l H_h \quad (3-35)$$

Therefore, the network forward propagation can be computed by equations (3-1), (3-2), (3-3), (3-4) and (3-5). The learning of the network, backward propagation, can be achieved by implementing equations (3-19), (3-21), (3-33) and (3-35), where the new weight can be calculated by adding the change in weight, ΔW , to the original weight:

$$W_{\text{new}} = \Delta W + W_{\text{old}} \quad (3-36)$$

3.2.3 Learning Rate

The step size η controls how fast the weights move toward a global minimum. Therefore, η is also called the learning rate. The learning rate is usually between one and zero. It should be noted that it is not necessary for η to be the same for different layers of nodes or for different iterations. However, in general, η is set to be a constant value throughout the network.

The learning rate is a very important parameter for determining the learning time required for convergence. If η is small, the network learns slowly as each step toward the global minimum is very small. However, if η is too large, the search may miss the global minimum as illustrated in Figure 3.4. Finding a proper learning rate is not an easy task.

Figure 3.4 A schematic illustration of how a large learning rate may induce a danger of missing the global minimum.

If time is not a major constraint, a small learning rate is always recommended, e.g. 0.1 is commonly used. Learning rate is application dependent which in turn, depends on the network structure, the relationship of the input-output patterns and also the size of the

training set. Searching for a reasonable η always involves trial and error. There are also some learning enhancement algorithms (to be discussed later in this Chapter) available for assisting the selection of learning rate.

3.2.4 Transfer Function

As mentioned in Chapter 2 (Section 2.4), there are many choices for the transfer function of the network. The sigmoid function, as shown in Figure 3.5, is commonly used because of its response similarity to a biological system:

$$F(x) = \frac{1}{1 + e^{-x}} \quad (3-37)$$

Furthermore, the derivative of this sigmoid function is quite simple:

$$F'(x) = F(x)(1 - F(x)) \quad (3-38)$$

Using the sigmoid function as the transfer function, equations (3-19) and (3-20) can be re-written as:

$$\delta_j^I = F(S_j^I)(1 - F(S_j^I))(Y_j^P - J_j) \quad (3-39)$$

$$\delta_i^I = F(S_i^I)(1 - F(S_i^I)) \sum_{j=1}^J (\delta_j^I W_{ij}^I) \quad (3-40)$$

Figure 3.5 A sigmoid transfer function, see equation (3-37).

3.2.5 Root Mean Squared Error

So far, the network has been exposed to the first pattern of the entire training set only. In order to solve a problem in a specific domain, the network requires 'p' patterns. The number 'p' depends on the size and the complexity of the problem domain.

After the first pattern has been exposed to the network, the network adjusts its weights accordingly. Then the second, third, ..., and p-th patterns are submitted to the network and the weights are updated after each exposure. The network has finished one epoch of learning when the whole set of patterns (first to p-th) has been provided to the network. Similar to biological learning, the network requires many epochs of learning

before it yields an acceptable accuracy. The bench mark for acceptable accuracy is determined by the total average error for an epoch which is the root mean squared value (RMS error):

$$\text{RMS error} = \sqrt{\frac{\sum_{p=1}^P \sum_{b=1}^J (Y_b^{Pp} - J_b^{Pp})^2}{P \cdot J}} \quad (3-41)$$

If the RMS error drops below a tolerance, the learning is said to have converged, i.e., successful learning. An RMS error between 0.04 and 0.05 is usually used in practice [3-2,3]. However, the number of epochs required to complete learning does not depend on the tolerance alone. It also depends on the initial weights, the relationship between the input-output patterns and the number of patterns.

It is important to select a non-symmetric set of initial weights for the network. A symmetric set of initial weights may force the learning to generate a symmetric set of final weights which may not be achievable. The relationship in the input-output patterns is an important factor in determining the number of epochs required to complete the learning. A loose relationship in the input-output patterns may require more epochs to convince the network that there is a close relationship. Finally, the number of patterns presented to the network is also a major factor for controlling the learning time. More patterns may not be better. More patterns may have more conflicts embedded in the information between patterns. Conflicts confuse the network and the learning oscillates between conflicts.

3.2.6 Enhancements to the BPN Learning Algorithms

The backpropagation network discussed to this point is a very basic structure. Without some enhancements, the network usually learns slowly. Bias nodes [3-1,3] and momentum [3-1] are the two features that are always incorporated to enhance performance. Fahlman's derivative [3-9], differential step size [3-10] and dynamic learning [3-11] are also commonly used to improve a network's learning time.

Bias nodes [3-1,3] are extra nodes added to every layer except the output layer (Figure 3.6). The activation of a bias node is always one, and therefore, it has no inter-connection with the previous layer. The bias nodes serve as internal references to each layer which is important for stabilizing the search for the global minimum.

Figure 3.6 A modified three-layer backpropagation network with bias nodes.

The gradient descent method is prone to failure if local minima arise. Momentum is a popular modification of the generalized delta rule for avoiding a search trapped in the local minima. With the momentum modification, ΔW in equations (3-21) and (3-35) can be written as:

$$\Delta W_{ji}^{J[t+1]} = \eta \delta_j^J I_i + \alpha \Delta W_{ji}^{J[t]} \quad (3-42)$$

and

$$\Delta W_{ih}^{I[t+1]} = \eta \delta_i^I H_h + \alpha \Delta W_{ih}^{I[t]} \quad (3-43)$$

where α is the momentum coefficient, a value between 0 and 1.

It should be noted that the weight adjustment terms with momentum, $\Delta W^{[t+1]}$, refer to the previous pass, not the entire epoch. The momentum terms help the network continue to do what it was previously doing unless a unique feature is encountered. A higher learning rate can be used in the presence of momentum. Therefore, the system should converge to a solution much faster.

3.2.6.1 Fahlman's Derivative

When the transfer function, $F(S)$ approaches one or zero in equations (3-39) and (3-40), the derivative, $F'(S)$, goes to zero. This will be the case when the summation function, S ($\sum IW^I$ or $\sum JW^J$), is relatively large (>10) or small (<-10) as the sigmoid function defined in equation (3-37). Since the derivative of the sigmoid transfer function, $F'(S)$, approaches zero, the weight adjustments, ΔW , diminishes and no learning takes place. The output activation is said to be saturated.

Fahlman [3-9] proposed a slight adjustment to the derivative of the sigmoid function, equation (3-38), by adding 0.1 to it:

$$F'(S) = 0.1 + [1 - F(S)]F(S) \quad (3-44)$$

The value 0.1 is arbitrary but it is important to keep the network learning. The original proposed Fahlman's derivative was applied to the hidden-output layer. However, experience suggests that the modified derivative should be applied to all layers [3-3].

3.2.6.2 Differential Step Size

Because of diminishing weight adjustments as the summation, S , increases, Chen and Mars [3-10] suggested another modification to handle saturated output activation. The entire derivative is dropped from equation (3-19) for the hidden-output layer:

$$\delta_j^I = (Y_j^P - J_j) \quad (3-45)$$

$$\text{i.e.} \quad \Delta W_r^I = \eta (Y_j^P - J_j) I_r \quad (3-46)$$

In addition, these researchers proposed that by using a multiple coefficient, the learning performance might be improved:

$$\eta_{\text{hidden}} = \frac{1}{10} \eta_{\text{output}} \quad (3-47)$$

3.2.6.3 Dynamic Learning

Finally, dynamic learning [3-11] is another commonly used modification for enhancing the ability of the network. As mentioned before, the learning rate need not be

the same for different layers. Furthermore, it can be adjusted logically from epoch to epoch.

Initially, a moderate learning rate is selected. The RMS error of a previous epoch is compared to that of the current epoch. If the current epoch's RMS error is larger than the previous RMS error, the situation is alarming since the search is moving away from the minimum. Therefore, the learning rate is reduced geometrically by a factor of 0.95 immediately:

$$\eta_{\text{new}} = 0.95 \cdot \eta_{\text{old}} \quad (3-48)$$

An arbitrary multiplier factor, 0.95, is used in order to avoid a zero learning rate.

On the other hand, if the current epoch's RMS error is smaller than the previous RMS error, nothing is done until the fifth consecutive decrease. If after the fifth consecutive decrease in RMS error, the search for a minimum seems to be moving in the correct direction, the learning can be accelerated by increasing the learning rate by 0.05:

$$\eta_{\text{new}} = 0.05 + \eta_{\text{old}} \quad (3-49)$$

If the learning is increased with every indication of convergence and decreased with every indication of divergence, the network can become oscillatory. Therefore, a buffer, five times, is needed to be convinced that the network is searching in the right direction.

3.2.6.4 Modified Dynamic Learning

In the original dynamic learning [3-11], the learning rate is altered geometrically and arithmetically according to the comparison of the previous and current epochs' RMS errors despite of the current learning rate. An arithmetic increase may introduce instability to the learning in progress. Therefore, *the author of this study has suggested* a further modification to dynamic learning. The learning rate is altered geometrically only, i.e., the change of learning rate depends on the current learning rate. If the RMS error improves in five consecutive epochs, the learning rate is increased by a factor of 1.01. On the other hand, if the RMS error is deteriorating, the learning rate is reduced by a factor of 0.99. The learning rate is limited to a value between 0.005 and 0.900. This modification is found

to be more stable than the original dynamic learning proposed and still has the advantage of improving the learning time.

3.2.6.5 Step-Declining Learning

Step-declining learning is another modification *suggested by this author* for minimizing the instability of the learning process. In step-declining learning, the learning rate is reduced by 1% when the RMS error increases. The learning rate is limited to a value of 0.005. The step-declining learning rate method is a very conservative approach and a relatively large initial learning rate, e.g., 0.75, can be selected.

3.3 Other Issues of Backpropagation Network

In addition to the basic mathematics and enhancements of the backpropagation network, there are some other considerations for applying the network to real practical service. These considerations are scalability of the prototype, selection of the numbers of nodes and hidden layers, the number of training patterns, normalizing range and choice of transfer and error functions.

All input and output data are normalized between upper and lower boundaries. The selection of limits are important for the serviceability of a production-sized network. Since the network output is limited between zero and one, its extrapolation beyond the upper and lower limits is not possible. Therefore, if the range of the upper and lower limits is set narrowly, the applicability of the network will be limited. On the other hand, if the range is set too wide, the network near the end limits becomes unreliable. In this application, the output will be one or zero if the summation, S , approaches positive and negative infinity respectively for the sigmoid transfer function. Therefore, the maximum and minimum data are normalized to 0.1 and 0.9 of the span respectively. Changing the range may also improve the learning time and accuracy of the network prediction.

In general any continuous first degree differentiable function can be used as the transfer function [3-1,2,3], e.g. trigonometric sine, hyperbolic tangent, etc.. Different

transfer functions perform differently during the learning stage. Similar to the transfer function, the error function can be also altered [3-3]. The general form of the error function is as follow:

$$E^p = \frac{1}{n}(Y_j^p - J_j)^n \quad (3-50)$$

where n can be any even number. However, a larger value of n magnifies the error. The least squared method is a special case when the value n in equation (3-50) is 2.

3.3.1 Knowledge Verification

In some cases, the knowledge acquired from successful learning may fail to respond to unseen input, i.e., this happens when the network has acquired incorrect knowledge. This may be because the network fails to generalize the solution. Therefore, it is very important to verify the knowledge before submitting it to the network for practical use. Knowledge verification is usually achieved by exposing some known but unseen patterns (a set of testing patterns) to the network. If the knowledge does not return correct predictions, the network must be re-trained. If training continues to generate incorrect knowledge (even with different network structures), some kind of major conflict may present in the training data. Therefore, the training patterns must be examined and re-assembled.

3.3.2 Scalability

A major difficulty in putting a backpropagation network into practical service is scalability. Before the construction of a production-sized network, the real problem is usually scaled down to a prototype - a subset of the real problem. The number of input and output data used in the prototype may be less than the real problem. Hence, less nodes, hidden layers and training patterns are required.

The amount of computer memory, training time and whether or not the network will produce the correct answer are the three major concerns of the scalability of a backpropagation network. Since each node is fully connected to the nodes in the next

layer, the requirement for computer memory increases dramatically with increasing number of nodes and layers. As the number of hidden layers increases, the network may become unstable and fail to learn. In the sense of training time, it has been suggested that a backpropagation network scales quadratically as the number of training patterns is increased and scales exponentially as the total number of processing units is increased [3-1,3]. Finally, the question of whether or not a production-sized network will generate the correct answer is entirely out of the control of the network designer.

3.3.3 Appropriate Numbers of Nodes, Hidden Layers and Training Patterns

The numbers of nodes in the input and output layers are usually defined by the application. However, there is no general rule governing the number of nodes that is appropriate in the hidden layer [3-1,2,3]. The network cannot learn well with too few nodes since nodes contain features from the input-output patterns. Too few hidden nodes pose a danger of limiting the memory of the network. As a result, a network with too few hidden nodes may not be able to learn, i.e. the learning may not converge. On the other hand, if there are too many hidden nodes, the system may memorize all the training patterns and generalizing the solution may be difficult although the learning may converge. Furthermore, training time increases with the number of hidden nodes. The search for the appropriate number of hidden nodes is usually accomplished by trial and error starting with a reasonable number nodes [3-2]:

$$i = \sqrt{h + j + c} \quad (3-51)$$

where c is a small number. Usually the number of hidden nodes required should be less than the total number of input nodes, $i \leq 1/2 h$. In general, excessive nodes are necessary for handling noisy data or for fault tolerance reasons [3-12,13]. A systematic technique for determining the most appropriate number of hidden nodes was proposed by Hirose et al [3-14]. They propose that if the RMS error is not reduced by one percent after a hundred epochs, additional hidden nodes are required. On the other hand, if the RMS error reduces more than one percent after a hundred epochs, a hidden node can be removed. The

optimal number of hidden nodes is the minimal number obtained from trials which can keep the learning converging.

In general, a single-hidden-layer network is appropriate of most applications [3-3]. However, in principle [3-1,14], a two-hidden-layer network should be able to perform arbitrarily complex solutions. Sometimes, increasing the number of hidden layers can expedite learning by increasing the non-linearity of relationships between patterns. It should be noted that the learning rate is reduced with increasing numbers of hidden nodes or hidden layers [3-15].

The appropriate number of training patterns required depends on the complexity of the input/output relationship. As suggested by Hafez [3-22], each connection between two nodes requires about two training patterns. Therefore, the number of training patterns is about twice the number of all the connections in the network, e.g. the suggested number of training patterns required, p , for an $h \times i \times j$ network will be:

$$p = 2i(h + j) \quad (3-52)$$

This is a good suggestion as a starting point and training patterns can be added or removed from the training set as required.

3.4 Summary for Software Implementation

In brief, the following procedure is used to generate a practical backpropagation network:

- 1) Define the problem domain and prepare sets of training and testing patterns;
- 2) Determine and normalize the input and output data;
- 3) Design the hidden structure of the network, i.e. number of hidden layers and hidden nodes, and determine a moderate learning rate and momentum coefficient;
- 4) Proceed with network training (including bias nodes and momentum terms):
 - 4.1) Present the first input-output pattern to the network;
 - 4.2) Calculate the activation level of all nodes in all layers;

- 4.3) Compute the error for the nodes in the hidden-output and the input-hidden layers;
- 4.4) Update the weights of all connections;
- 4.5) Present the next input-output pattern to the network and repeat procedures 4.2 to 4.5 until the entire training pattern sets have been exposed to the network - one epoch;
- 4.6) Reckon the RMS error;
- 4.7) Repeat procedure 4.1 to 4.6 until the RMS error is smaller than a pre-determined tolerance value, i.e. successful learning, or number of epochs has exceeded a pre-determined limit, i.e. failure learning;
- 5) If the training fails, consider modifications to the network and/or training pattern sets, e.g. Fahlman's derivative, dynamic learning, differential step size, different hidden structure, different learning rate, different momentum coefficient, various normalizing ranges, different transfer functions or different error functions. Other learning enhancement algorithms are also available in the literature [3-16,17,18,19].
- 6) If the system does not yield a solution, using other available paradigms may be the only alternative;
- 7) If the learning does converge, the set of solution weights and network structure are extracted and stored for application. The application of the network is simply the feed forward stage of the backpropagation network.
- 8) Testing or verifying the completed network is also very important. Knowledge acquired from successful learning may generate undesired results when applied to unseen patterns. If this is the case, the network must be re-trained (with either the same or different network structure).

Creating a production-sized backpropagation network is usually preceded by a scaled down prototype. However, there is no guarantee that the production-sized network will learn and perform as the prototype. As mentioned in Chapter 2, implementing the network

on a parallel machine is the best means of accelerating the learning process. However, neural network hardware and neural accelerators [3-3] are also available in the marketplace for sequential computers.

3.5 PC Based BPN Development Tools - NNWork

In general, there are two types of training used by BPN practitioners: on-line and off-line training. NNWork is developed specifically for off-line training, i.e., the training patterns are pre-selected and once knowledge is acquired, the network will use it until amendment to the existing knowledge is required, e.g., ranges of parameters are changed. On the other hand, on-line training is usually used for applications such as signal processing where new patterns are added to the training sets when they are available and the network is required to learn and predict almost on a real-time basis. Therefore, because of the nature of the two different training situations, the off-line training time seems less important than the training time of the on-line situation.

In brief, the life cycle of a practical BPN involves the preparation of training data set (and testing data set), the training of the network (including knowledge verification) and the use of the knowledge obtained from the training to perform prediction. NNWork is a PC based graphics environment generated by this author which handles all three stages. NNWork contains three modules: normalization, training and prediction modules.

Experimental data can be normalized (between 0 and 1) in the normalization module. A maximum of 15 parameters are allowed for each input and output set. The training of the network is carried out by the training module using data from the normalization module.

Basically, the training module is the implementation of the BPN algorithms described previously in this chapter. The user has the freedom of selecting a network structure (numbers of hidden nodes and layers, to a maximum of 99 hidden-nodes per each of two hidden-layers), learning rate, momentum coefficient, learning enhancements (Fahlman's Derivative, modified dynamic learning, step-declining learning and differential

step size) and RMS error tolerance level. Moreover, modifier' 'dynamic or step declining learning can be activated or deactivated at an intermediate stage in the learning. This feature allows the practitioner to control the learning rate during training.

Finally, the prediction module is an automated version of the forward propagation part of BPN. Using the knowledge acquired from training, HAZ hardness and weld geometry are determined from the input welding parameters with the prediction module. In addition to basic prediction, for the convenience of the user, a batch testing function (for knowledge verification) and an automated plotting function (for parametric study) have been included. A detailed description of NNWork and the complied version are included in Appendix VII for easy reference.

Chapter Four

Performance of Modified Learning Enhancements

While investigating BPN for solving weld features problems (HAZ hardness and weld bead geometry predictions), different learning enhancement algorithms were found to perform differently. In general, the momentum coefficient [4-1] and Fahlman's derivative [4-2] were found to be quite helpful in most cases. However, the differential step size method [4-3] did not seem to help speed up the learning process as expected. Dynamic learning [4-4] was found to be unstable and performed inconsistently. *Therefore, the author of this study has suggested two dynamic learning modifications, namely modified dynamic learning and step-declining learning* (see Chapter 3, Sections 3.2.6.4 & 3.2.6.5 for detailed descriptions). The performance of all learning enhancement algorithms is discussed in the following.

4.1 Test Problems and Results

Three different networks were constructed for testing the performance of each individual learning enhancement algorithm: momentum, Fahlman's derivative, differential step size dynamic learning, modified dynamic learning and step-declining learning. The structure of the test networks ranged from simple (exclusive-or, XOR, problem), medium (HAZ hardness problem) to complex (weld geometry problem). Because the major concern of these tests was the performance of the learning enhancement algorithms, special attention was placed on the complexity of the network primarily rather than the accuracy of the prediction. The network as a prediction tool is reported later in this thesis.

For all tests, the network initial weights were set randomly between +0.5 and -0.5 (see Section 3.2.1) and all input and output parameters were normalized between 0.9 and 0.1 (see Section 3.3). Learning was considered unsuccessful when it failed to converge below the RMS error tolerance level in ten thousands epochs. The test on each learning

enhancement was terminated if the learning failed in five consecutive trials, or, when learning was successful after five trials. A 'relative performance' index was introduced to compare each learning enhancement with the control group. The relative performance is calculated by dividing the average numbers of epochs of the enhanced learning (excluding the largest and the smallest values) required for successful learning by the average numbers of epochs of the basic learning system (excluding the largest and the smallest values) required for successful learning. Therefore, the smaller the value of the relative performance index the better the learning enhancement.

4.1.1 XOR Problem

The exclusive-or, XOR, is a linearly non-separable problem (Figure 4.1), i.e., the black and white spots cannot be separated by a straight line graphically. Because XOR is the simplest non-linear problem, it is always used as a bench-mark test problem for BPN. The XOR problem has two input values (either zero or one) and one output value (either zero or one), the XOR's truth table is included in Table 4-1. The network consists of three layers: the input-layer has two nodes plus a bias node, the single hidden-layer consists of three nodes plus a bias node and the out-layer has only one node. Four training patterns are used, as listed in the truth table. A relatively large initial learning rate and a moderate momentum coefficient were used, 0.75 and 0.25 respectively. A relatively tight RMS error tolerance level is selected, 0.04 (see Section 3.2.5). The result of different tests are listed in Table 4-2.

Table 4-1 XOR Truth Table.

input 1	input 2	output
0	0	1
1	0	0
0	1	0
1	1	1

Figure 4.1 Graphical representation of XOR problem.

In the XOR test (see Table 4-2), it was found that both momentum and Fahlman's derivative speed up the learning process. Furthermore, Fahlman's derivative stabilizes the learning process, i.e., the number of failures decreased from three to one. Therefore, they were used extensively in developing modifications to the dynamic learning finally used in this investigation. On the other hand, the introduction of dynamic learning was so unstable that the learning failed to converge in five consecutive trials (Table 4-2) and RMS errors oscillated between 0.060 and 0.045 which may be considered to be close to successful. The differential step size helped the learning to converge by sacrificing learning time which took four times longer than basic learning with momentum. Therefore, it was not used in the final network for this investigation. Modified dynamic learning seemed to speed up the learning process consistently. On the other hand, step-declining learning did not show any obvious improvement. Learning failure when using the step-declining technique was primarily due to a low learning rate, 0.005. This may be traced to the nature of the algorithms. Nonetheless, the RMS error still decreased slowly. On the basis of these tests, modified dynamic learning seemed to provide the best overall performance.

Table 4-2 Performance of Different Learning Enhancement Algorithms for the XOR Problem (Measured by the Number of Epochs Required for Successful Learning).

	basic learning ^a	$\alpha=0.25^b$	Fahlman's Derivative ^c	Diff. Step Size ^d	Modified Dynamic ^e	Step- Declining ^f
Trial 1	3315	2076	2133	5222	1339	1856
Trial 2	2240	1637	1303	6350	1877	4486
Trial 3	2135	1843	1734	8461	1445	2194
Trial 4	2984	1408	1651	8912	2561	1809
Trial 5	3612	1474	1498	7647	1337	3362
# of Failures ^g	1	3	1	0	0	1
Relative Perform. ^h	1	0.58	0.58	2.63	0.55	0.87

- Notes: 1) The initial weights of all learning were set by random values between +0.5 and -0.5.
 2) All learning enhancement algorithms were tested until at least five successful learning was recorded.
 3) Network learning using the original dynamic learning failed to converge for five consecutive times. Therefore, it is not listed in the table. However, RMS error oscillated between 0.060 and 0.045, i.e., very close to the tolerance level.
 a) $\eta=0.75$, $\alpha=0.00$ and no other learning enhancement algorithms was used.

- b) $\eta=0.75$, $\alpha=0.25$ and no other learning enhancement algorithms was used.
- c) $\eta=0.75$, $\alpha=0.25$ and no other learning enhancement algorithms was used except Fahlman's derivative.
- d) $\eta=0.75$, $\alpha=0.25$ and no other learning enhancement algorithms was used except differential step size.
- e) $\eta=0.75$, $\alpha=0.25$ and no other learning enhancement algorithms was used except modified dynamic learning.
- f) $\eta=0.75$, $\alpha=0.25$ and no other learning enhancement algorithms was used except step-declining learning rate.
- g) The learning was considered to be unsuccessful when it failed to converge within 10000 epochs.
- h) Relative performance was calculated by dividing the average of the middle three epochs by the average of the middle three epochs of basic learning, i.e., the smaller the value of relative performance is, the better the learning enhancement performs.

4.1.2 HAZ Hardness Problem

The second test network was the HAZ hardness problem. It is considered to be more complicated than the XOR problem because it includes some noisy data and uncertainties in the training pattern sets. There were four input parameters (800 to 500 °C cooling time, carbon content, Ito's carbon equivalent and the IIW carbon equivalent) and one output parameter (HAZ hardness value) in this problem - see Chapter 5 for a detailed discussion of this problem. A four-layer network structure was used: the input-layer has four nodes and a bias node, the first hidden-layer consists of three hidden-nodes and a bias node, the second hidden-layer consists of two nodes and a bias node and the output-layer has a single node. Forty training patterns were used - see Appendix II. The initial learning rate was 0.75, the momentum coefficient was 0.5 and the RMS error tolerance level was 0.04. The results of testing with different learning enhancement algorithms are tabulated in Table 4-3. The results show that both the modified dynamic and step-declining algorithms helped speed up and stabilize the learning process with comparable relative performance, i.e., the number of failures decreased from three (original dynamic learning to zero in both cases). Once again, the modified dynamic method provided the best overall performance.

Table 4-3 Performance of Different Learning Enhancement Algorithms for the HAZ Hardness Problem
(Measured by the Number of Epochs Required for Successful Learning).

	without Dynamic Learn.	Dynamic Learning	Modified Dynamic	Step-Declining Learning
Trial 1	1973	539	1099	551
Trial 2	756	833	524	547
Trial 3	1450	852	845	841
Trial 4	548	769	458	6533
Trial 5	1394	838	593	943
# of Failure	2	3 ^a	0	0
Relative Performance	1	0.68	0.55	0.65

- Notes: 1) The initial weights of all learning were set by random values between +0.5 and -0.5.
 2) All learning enhancement algorithms were tested until at least five successful learning was recorded.
 3) The initial learning rate, momentum coefficient and RMS error tolerance level were 0.75, 0.5 and 0.04 respectively and Fahlman's derivative was used in all cases.
 4) The learning was considered to be unsuccessful when it failed to converge within 10000 epochs.
 a) For these three unsuccessful learning, RMS errors oscillated around 0.076 and 0.042.

4.1.3 Weld Bead Geometry Problem

The most complex network used for testing different learning enhancement algorithms in this investigation consisted of four layers and twenty-eight training patterns, see Appendix VI. The network constructed for solving the weld bead geometry problem with C-25 shielding gas is detailed discussion in Chapter 6. The input-layer has four nodes (current, voltage, welding speed and plate thickness) and a bias node. The first and second hidden-layers have seven and five hidden-nodes respectively and each layer has one bias node. The output-layer consists of three nodes (bead height, bead width and penetration). As defined by the network input and output structure, this network was intended for predicting weld bead geometry. The initial learning rate and momentum coefficient were 0.5 and 0.5 respectively. Because of uncertainties in the input-output relationship, a higher

RMS error tolerance level was used, 0.05, in this case. The results are included in Table 4-4.

Table 4-4 Performance of Different Learning Enhancement Algorithms for the Weld Bead Geometry Problem
(Measured by the Number of Epochs Required for Successful Learning).

	without Dynamic Learn.	Dynamic Learning	Modified Dynamic	Step-Declining Learning
Trial 1	2363	2483	4267	4475
Trial 2	N/A	N/A	2730	2766
Trial 3	N/A	N/A	3450	2251
Trial 4	N/A	N/A	6818	3912
Trial 5	N/A	N/A	3722	1522
# of Failures	5 ^a	5 ^b	2	1
Relative Performance ^c	1	1.05	1.61	1.26

- Notes:
- 1) The initial weights of all learning were set by random values between +0.5 and -0.5.
 - 2) All learning enhancement algorithms were tested until at least five successful learning was recorded or five unsuccessful learning had occurred.
 - 3) The initial learning rate, momentum coefficient and RMS error tolerance level were 0.5, 0.5 and 0.05 respectively and Fahlman's derivative was used in all cases.
 - 4) The learning was considered to be unsuccessful when it failed to converge with 10000 epochs.
 - a) In total, eight unsuccessful learning was recorded.
 - b) For all of the unsuccessful learning, RMS errors oscillated between 0.083 and 0.053. Seven unsuccessful learning was recorded.
 - c) Wherever there are five data available for comparison, the average of the middle three values are used for calculating the relative performance. Otherwise, the only datum is used.

Where the network is complex (see Table 4-4) both modified dynamic learning and step-declining learning have clear superiority over the original dynamic learning and basic learning with momentum and Fahlman's derivative. Although the relative performances of both the modified learning algorithms do not out-perform the original dynamic learning or basic learning with momentum and Fahlman's derivative, they have the advantage of learning consistently.

4.2 Discussion

Some general characteristics of different dynamic learning algorithms were observed. When dynamic learning was used, the time required for successful network learning was shortened considerably in some cases because a larger step size (learning rate, η) was introduced. However, sometimes learning failed to converge because of a changing learning rate, see Tables 4.2, 4.3 & 4.4. With dynamic learning, learning rate was increased arithmetically despite the magnitude of the current learning rate. Therefore, the relative change of learning rate can be dramatic, e.g. η changing from 0.10 to 0.15 represents a 50% increase. Such a large change may introduce instability and may cause the learning to oscillate (Figure 4.2 and Table 4-2 note 3). A schematic illustration of the basic learning process is included in Figure 4.3 for comparison with Figure 4.2.

On the other hand, when modified dynamic learning is used, the learning rate is altered according to the current learning rate. Therefore, the change is less dramatic, especially when learning is close to the finish and the learning rate is relatively small. The proportional increase in learning rate minimizes the instability caused by the change. A schematic sketch of the learning process with

Figure 4.2 Schematic graphs of η and RMS error versus number of epochs for unsuccessful learning with dynamic learning.

Figure 4.3 Schematic graphs of η and RMS error versus number of epochs for successful learning without dynamic learning.

modified dynamic learning is shown in Figure 4.4. With modified dynamic learning, the learning time still benefits from the increase of step size without sacrificing stability.

Step-declining learning rate is the most conservative learning enhancement algorithm. Usually, the learning begins with a fairly large learning rate. As difficulties are encountered, the learning rate is reduced proportionally (Figure 4.5). Because the learning rate never increases, the learning process is considered to be very stable. However, the learning rate sometimes settles to a very small value (a pre-set minimum, e.g., 0.005) and the learning process can be lengthy

If the network learning becomes saturated (oscillating about a certain RMS error) close to convergence (within 5%) and there is a relatively small learning rate (equal to or less than 0.10) slightly increasing the learning rate (introducing the modified dynamic learning) might help the learning to converge. The introduction of a slight instability acts as a small shaking force on the system and helps the learning converge by moving the current weights through tiny ripples around the global minimum.

Both modified dynamic and step-declining learning rate algorithms would be used extensively in the HAZ hardness prediction and weld bead geometry problems.

Figure 4.4 Schematic graphs of η and RMS error versus number of epochs for successful learning with modified dynamic learning.

Figure 4.5 Schematic graphs of η and RMS error versus number of epochs for successful learning with step-declining dynamic learning rate.

Chapter Five

Predicting Weld 800 to 500 °C Cooling Time and HAZ Hardness Using the BPN Method

5.1 Introduction

Heat-affected zone (HAZ) hardness is widely used as an indicator of weld quality in practice as explained in Chapter 1. It depends on the percentage of hard martensite structure found in this zone. The formation of martensite structure in turn depends on the base metal chemical composition and the thermal history of the weld. The weld 800 to 500°C cooling time, $t_{8/5}$, is often selected as a measure of the thermal cycle because this is the transformation range of most structural steels, i.e., the cooling region where martensite forms. Several investigations correlate $t_{8/5}$ and base metal chemical composition through a so-called carbon equivalent, see Chapter 1, Section 1.2.2, with HAZ hardness [5-1,2,3]. Therefore, in order to obtain a practical and accurate prediction of HAZ hardness, the relationship between $t_{8/5}$ and welding conditions (e.g., arc current, voltage, wire travel speed, plate temperature, plate thickness, welding process efficiency, etc.) must be established in advance.

The algorithm for predicting HAZ hardness using ANN technology is a two part system. Cooling time, $t_{8/5}$, is first determined from the input welding conditions and then HAZ hardness is predicted from $t_{8/5}$ and the steel chemical composition. The $t_{8/5}$ network correlates heat input, plate thickness and plate temperature to cooling time. The HAZ hardness network maps $t_{8/5}$ and the base metal chemical composition (in terms of carbon content and carbon equivalents, see Chapter 1, Section 1.2.2) to HAZ hardness. Finally, the knowledge of both networks is combined to permit the practitioner to estimate HAZ hardness directly from the welding conditions and steel chemical composition as it is usually done in the industry.

5.2 Weld 800 to 500 °C Cooling Time

The $t_{8/5}$ cooling time is sometimes calculated from the Adams' cooling time relationships [5-4,5] (see Chapter 1, Section 1.2.2.1). In practice, the Adams' models are instructive and relatively easy to apply. However, difficulties arise when assigning fixed thermal conductivity and volumetric specific heat values which, in reality, vary with temperature, among other things. Other more sophisticated finite element methods [5-6,7] and the Kohira model [5-8] are sometimes used but even then, several approximations are necessary to achieve reasonable accuracy.

The purpose of this part of the thesis study is to investigate the use of the backpropagation network method as an alternative for predicting the 800 to 500°C cooling time for submerged arc bead-on-plate welds (SAW). This can eventually be extended to other processes if the technique is proven useful. Once calculated, it is input to network for predicting HAZ hardness. Briefly the operating parameters (arc voltage, current, welding speed, plate thickness, plate temperature & welding process efficiency) are submitted to a trained network and a cooling time is received. During training, the network learns the relationship between an input data set and output cooling time by exposing the network to 'examples'. The network stores the knowledge as internal network weights - see Chapter 3 for a detailed discussion. Although the knowledge is based on experimental submerged arc bead-on-plate welds in the following investigation, approximating cooling times for other welding processes, e.g., gas metal arc welding (GMAW) and shielded metal arc welding (SMAW), are possible by assigning appropriate welding process efficiencies [5-9,10].

5.2.1 Backpropagation Networks for Predicting $t_{8/5}$

To predict the cooling time of a bead-on-plate weld, the network input parameters are energy input rate per unit length (Q), plate thickness (h_i), plate temperature (T) and

welding process efficiency (η_{proc}). The process energy input rate per length (Q) is determined from the arc current (I_c), arc voltage (V_o) and wire travel speed (S_p):

$$Q = \frac{I_c \cdot V_o}{S_p} \quad (5-1)$$

In reality, the heat input to a workpiece (H_i) is a more fundamental parameter and may be expressed as:

$$H_i = \eta_{proc} Q \quad (5-2)$$

where η_{proc} is the process efficiency ($\eta_{proc}=0.95$ for SAW [5-9]). Although all the training data are based on submerged arc welds, a process efficiency is included in the scheme so that it can be applied to other welding processes in the absence of data bases specific to these joining methods. However, if the network is used to predict SAW cooling time only, the process efficiency ($\eta_{proc}=0.95$) need not be included. Yet there is no difficulty including the efficiency because it is simply a scaling factor. Because the output of the transfer function is always between zero and one, all input and output values are normalized between 0.1 and 0.9 in any event before submitting to the network for training, see Chapter 3, Section 3.3.

The basic network architecture used in this study is shown schematically in Figure 5.1. It is basically a single-hidden layer network with bias nodes. The training data is an experimental data base

sub-set (20 out of 59) from the work of four different independent investigators:

Rodrigues et al [5-11],
Van Adrichem et al [5-12], Bibby et al [5-13] and Bala et al [5-14] - see Appendix

Figure 5.1 Proposed backpropagation network for predicting $t_{8/5}$.

III. The heat input, plate thickness, plate temperature and $t_{8/5}$ cooling time ranges for this submerged arc weld data set are listed in Table 5-1. The network was trained with 16, 18 and 20 parameter sets (Appendix III) to test the accuracy of the system to the number of training patterns (Table 5-2). Furthermore, different node and layer arrangements were similarly tested (Table 5-3).

Initial weights were set in the range between -0.5 and +0.5 randomly with a computer algorithm for this purpose [5-15]. Furthermore, modified dynamic learning and Fahlman's derivative were incorporated. A moderate learning rate and momentum coefficient (0.5 and 0.5 respectively) were selected. The root mean squared error (RMS error) tolerance level was set at 0.04 to ensure a good fit. The rest of the experimental data (39 out of 59) in group A were used to test and verify the trained network. In addition to these data, a fifth independent set (25), group B from Kohno et al [5-16] was used for further verification (Appendix IV). The data ranges are listed in Table 5-1.

Table 5-1 Welding Parameters Used to Train and Test the $t_{8/5}$ Prediction Networks.

		Voltage (Volts)	Current (Amps)	Speed (mm/s)	Plate Thick. (mm)	Heat Input (kJ/mm)	Plate Temp. (°C)	$t_{8/5}$ (s)
A ¹	Min	20	150	2.5	12.7	0.51	15	3.1
	Max	35	2684	10	63	10.2	155	96.8
B ²	Min	30	480	3.5	25	1.45	20	5.15
	Max	40	920	8.5	25	6.2	20	73.33

Note 1: Group A test set range: references [5-11,12,13,14]

Note 2: Group B test set range: reference [5-16]

5.2.2 Results and Discussion

Although there is no general rule for deciding the sequence of selecting the appropriate numbers of training patterns, hidden nodes or hidden layers, this author suggests tackling the number of training patterns first and then the network structure. Furthermore, starting with a minimum number of training patterns, hidden nodes or hidden layers guarantees the problem is not over-stated. Since the network design is a trial and

error process, training patterns, nodes and layers can be added or removed as required. The selection of learning rate, momentum coefficient and learning enhancements are left to the practitioner (see Chapter 4). Moreover, knowledge verification is extremely important (see Chapter 3, Section 3.3.1) since successful learning does not always imply accurate knowledge. Therefore, all networks were trained at least three times, and only the network with the best accuracy is reported.

The number of training patterns should be kept to a minimum, but at the same time it should cover the entire problem domain. During prediction, any parameters beyond the training range would normally be considered to be beyond the knowledge of the network. In this specific study, a 2-hidden layer network with 4 and 3 hidden nodes (including bias nodes) respectively was used to determine the appropriate number of training patterns required. The use of a 2-hidden layer network was used to ensure consistent successful learning. The network was initially trained with 16 training patterns - see Appendix III. Additional training patterns were used to improve the prediction accuracy until a satisfactory accuracy was achieved. The results are tabulated in Table 5-2. It was determined that 20 training patterns provided the knowledge necessary for the cooling time problem.

Table 5-2 Network Experimentation for Deciding the Appropriate Number of Training Patterns for the $t_{8/5}$ Prediction Using Group A Test Data (mean $t_{8/5}$ =20.50 s) .

# of Training Patterns Used	Correlation Coefficient	Absolute Difference (s)	
		Mean	Standard Deviation
16	0.928	4.29	4.38
18	0.954	3.78	3.41
20	0.965	2.40	3.44

- Note: 1) A 2-hidden layer network with 4&3 hidden nodes respectively (including bias nodes) was used in all cases.
2) The results presented were based on the best of three trials.

This study suggests that a network of 2-hidden layer with 5 and 3 hidden nodes in each layer respectively provides reliable SAW $t_{8/5}$ predictions. The knowledge (weights) for this learning is listed in Appendix VIII. The results of the $t_{8/5}$ network predictions are

shown in Table 5-3 and Figure 5.2, where a correlation factor, mean and standard deviation of absolute difference of 0.965, 2.87 s and 3.19 s respectively is reported for Group A data, i.e., the network is expected to have 14% mean error.

The network knowledge was further verified with independent data (Group B) (Table 5-4 and Figure 5.3). A correlation factor, mean and standard deviation of absolute difference values of 0.991, 3.58 s and 2.79 s respectively are obtained in this case, i.e., mean error of 14%. However, even the simpler network configuration of a single-hidden layer network with 5 hidden nodes is quite comparable (Tables 5-3 & 5-4) with mean errors of

Figure 5.2 Comparison of backpropagation networks (2 hidden layers with 5 and 3 hidden nodes respectively and a single-hidden layer with 5 hidden nodes) and Adams' cooling times, with experimental [5-11,12,13,14] $t_{8/5}$.

Figure 5.3 Comparison of networks (2 hidden layers with 5 and 3 hidden nodes respectively and a single-hidden layer with 5 hidden nodes) and Adams' cooling times, with Kohno [5-16] measured data.

16% and 12% when it is tested on Groups A and B data respectively. The knowledge of the 5-hidden node network is included in Appendix VIII. Furthermore, a similar 2-hidden layer network with 4 and 3 hidden nodes in each layer provided comparable results when tested against Group A data with a mean error of 17%. The weights of the 2-hidden layer network is listed in Appendix VIII. Therefore, the accuracy of the prediction is sensitive to but not a paramount factor regarding network structure.

Table 5-3 Network Experimentation for Testing the Accuracy of $t_{8/5}$ Prediction Using the Group A Test Data (mean $t_{8/5}$ = 20.50s).

# of Hidden Layer(s)	# of Nodes (1st/2nd Layer)	Correlation Coefficient	Absolute Difference (s)	
			Mean	S.D.
1	3	0.939	3.90	4.31
1	4	0.956	3.17	3.53
1	5 ^a	0.959	3.04	3.31
2	3/3	0.937	4.02	4.06
2	4/3	0.965	2.90	3.44
2	5/3 ^a	0.965	2.87	3.19
2	5/4	0.961	3.40	3.80
Adams' Cooling Time ^a		0.881	5.38	6.06

- Notes: 1) Bias node(s) is included.
 2) Results reported are the best out of three trials.
 a) Results of these entries are plotted in Figure 5.2.

Table 5-4 Network Experimentation for Testing the Accuracy of $t_{8/5}$ Prediction Using the Group B Independent Data Set (mean $t_{8/5}$ = 25.50s).

# of Hidden Layer(s)	# of Nodes (1st/2nd layer)	Correlation Coefficient	Absolute Difference (s)	
			Mean	S.D.
2	5/3	0.991	2.58	2.79
1	5	0.996	3.18	2.06
Adams' Cooling Time		0.990	8.25	9.84

Note: Comparison of the calculated and measured values are plotted in Figure 5.3.

It should be observed from Tables 5-3 & 5-4 that all network configurations outperformed the Adams' cooling time models. In some situations there is more than a 50% improvement - see Tables 5-3 & 5-4 and Figures 5.2 & 5.3.

However, the predictions of Group A data scattered more than those of Group B data probably because Group A data is a combination of results from four different individual researchers. Consequently, there might well be unseen and uncontrollable differences between these investigations which could cause such a variation in the results. For example, workmanship might affect the process efficiency. An efficiency of 95% is used; yet this is a nominal value which could vary from operation to operation. On the other hand, Group B data comes from only one source which would likely be more consistent but a bias might be present in the experiment. Therefore, predictions from Group B data showed signs of systematic error in the results.

The major advantage of using BPN is that assumptions such as that of fixed thermal conductivity and volumetric heat capacity necessary in the Adams' formulations are not required. These values vary throughout the thermal cycle (Figures 5.4 & 5.5) and yet a simple constant value must be assigned. This may contribute significant error to the cooling time calculations. Using the backpropagation network, the material properties are implicitly

Figure 5.4 Sensitivity of thermal conductivity of iron to temperature changes.

Figure 5.5 Sensitivity of volumetric specific heat of iron to temperature changes.

embedded in the network knowledge during the learning process. As mentioned previously, BPN takes into account these kinds of unseen or unknown relationships between input and output data sets. This property is of considerable importance in analyzing empirical data such as weld features.

Eliminating the need for relative thickness, H_r , is yet a further advantage of the neural approach for determining an appropriate $t_{8/5}$. Estimates of H_r (Table 1.2 and Figure 1.8) and C_r in the Adams' cooling time models are considered questionable. There are several other difficulties in the conventional models that require assumptions, e.g., the necessity of assuming a point heat source, insignificant radiation from the heat source, etc.. In fact, information on all these factors is implicitly embedded in the network.

5.2.2.1 Predicting $t_{8/5}$ for Other Welding Processes

Since the process efficiency is embedded into the heat input to the workpiece (H_i) and the process efficiency of the training data is set at 95% for SAW, approximating $t_{8/5}$ for other welding processes can be determined from the knowledge acquired from SAW cooling by assigning an appropriate process efficiency (Table 1-1). Data from two investigations based on gas metal arc weld (GMAW) [5-17] and shielded metal arc weld (SMAW) [5-9] (Appendix IV) have been prepared to test this contention.

The experimental data of $t_{8/5}$ of GMAW are extracted from the work of Signes [5-17] (Appendix IV). As suggested by Yurioka [5-9], a 70% process efficiency is appropriate for GMAW and 80% for SMAW. The knowledge of the SAW 2-hidden layer network with 5 and 3 hidden nodes was used and a summary of the results for the entire (both valid and invalid) experimental data set

Figure 5.6 Comparison of GMAW $t_{8/5}$ predicted by using BPN (5&3-hidden node 2-hidden layer) and Adams' cooling time models (with valid data only).

is listed in Table 5-5. The Adams' cooling time models seem to correlate better than the BPN approach for the entire data set because BPN does not perform well where there is extrapolation (invalid range). When the data beyond the valid range is removed from the data set, BPN out-performs the Adams' cooling time models (Table 5-5 and Figure 5.6).

Table 5-5 Summary of GMAW $t_{8/5}$ Values Predicted by Using BPN (2-hidden layer with 5 and 3 hidden nodes) and the Adams' Cooling Time Models.

Data Set	Prediction	Correlation Coefficient	Absolute Difference (s)	
			Mean	S.D.
Entire data set (mean $t_{8/5}$ =37.24s)	BPN	0.882	11.41	16.86
	Adams'	0.944	24.80	56.06
Valid data only (mean $t_{8/5}$ =14.44s)	BPN ^a	0.943	2.50	2.21
	Adams' ^a	0.952	2.86	4.98

Note: a) Results of these entries are also plotted in Figure 5.6.

The same network knowledge was also used to predict $t_{8/5}$ for shielded metal arc weld (SMAW) [5-9] (Appendix IV) by assigning 80% to the process efficiency. However, both the BPN and Adams' techniques over-estimate the cooling time (Table 5-6 and Figure 5.7). By reducing the process efficiency to 70%, the mean absolute differences

Figure 5.7 Comparison of SMAW $t_{8/5}$ predicted by using BPN (5&3-hidden node 2-hidden layer) and Adams' cooling time models (process efficiency = 80%).

Figure 5.8 Comparison of SMAW $t_{8/5}$ predicted by using BPN (5&3-hidden node 2-hidden layer) and Adams' cooling time models (process efficiency = 70%).

improved drastically (Table 5-6 and Figure 5.8). This suggests that in fact the efficiency of 80% suggested by Yurioka is high in this case. In general, the BPN provided a comparable correlation coefficient and a better mean of absolute difference than the Adams' cooling time models.

Table 5-6 Summary of SMAW $t_{8/5}$ Values Predicted by Using BPN (2-hidden layer with 5 and 3 hidden nodes) and the Adams' Cooling Time Models (mean $t_{8/5}$ =14.35s).

Process Efficiency	Prediction Model	Correlation Coefficient	Absolute Difference (s)	
			Mean	S.D.
80%	BPN	0.985	6.69	8.04
	Adams'	0.997	7.49	11.47
70%	BPN	0.972	3.20	2.27
	Adams'	0.997	3.33	4.46

Notes: a) Results of these entries are also plotted in Figure 5.7.
b) Results of these entries are also plotted in Figure 5.8.

5.2.2.2 Comments about Process Efficiency

The results of this investigation would suggest that the welding process efficiency is a reasonably robust assignment. However, at least in principle it depends on the welding heat source, workmanship, equipment setup, etc.. Therefore,

Figure 5.9 A schematic diagram of BPN for predicting $t_{8/5}$ with welding process as one of the input parameters.

assigning a nominal efficiency for each welding process might be suspect but it is sometimes necessary when data is not sufficient to carry out independent training. With sufficient data, the welding process can be treated as one of the input parameters (Figure 5.9). In this investigation, a numerical value can be used to represent each process, e.g. SAW=0.1, SMAW=0.5, GMAW=0.9. The selection of these values is quite arbitrary but

the assignment must be consistent and the values must be distinctive. About twenty well distributed sets of data are required for each welding process as suggested in this study. At this time, there is insufficient data for this trial and therefore as an alternative, efficiency is combined with SAW knowledge for predicting $t_{8/5}$ as discussed in Section 5.2.2. At the same time, the necessity of achieving precise $t_{8/5}$ values is debatable because the HAZ hardness is sensitive to cooling time over three orders of magnitude for most steels.

Finally, the nature of BPN is suitable for custom tailored $t_{8/5}$ models for specific job shop operations. Depending on the scope of the domain, knowledge about $t_{8/5}$ can be correlated with the welding conditions with as few as 20 trial runs.

5.3 HAZ Hardness

Because of the importance of HAZ hardness, it has been studied extensively in the past twenty years [5-18,19]. As discussed in Chapter 1, Section 1.1.1, these studies purport to be able to estimate the HAZ hardness level given $t_{8/5}$ and steel chemical composition by means of regression relationships. The hardness, $t_{8/5}$ and chemical composition relationships are bounded by pre-defined mathematical equations as required by the regression method. The major objective of this study is to explore the use of BPN for predicting HAZ hardness given $t_{8/5}$ and steel chemical composition.

Table 5-7 Ranges of $t_{8/5}$, Carbon Content, Carbon Equivalents and HAZ Hardness Values Used for the Training of the Hardness BPNs.

Range	C (wt%) ^a	Pcm (wt%) ^b	CE (wt%) ^c	$t_{8/5}$ (s)	Hardness (VHN)
Minimum	0.034	0.166	0.372	3	200
Maximum	0.254	0.359	0.560	60	485

Note: a) Carbon content in weight percent.
b) Ito's carbon equivalent [5-20].
c) IIW carbon equivalent [5-21].

Experimental data taken from the work of Yurioka et al [5-9] is used as the basis for this part of the thesis study. One hundred and forty HAZ hardness values are reported

for fourteen low alloy steel covering a wide range of materials and cooling times used in practice - see Table 5-7 for the data ranges and Appendix II for the original data.

5.3.1 Backpropagation Networks for Predicting HAZ Hardness

The input parameters of a BPN for predicting HAZ hardness are $t_{8/5}$, carbon content (C wt%) and carbon equivalents (Pcm - proposed by Ito et al [5-20] and CE - suggested by

Figure 5.10 A schematic BPN diagram used for predicting HAZ hardness.

International Institute of Welding [5-21]), see Chapter 1, Section 1.1.1.1 for details. Two different carbon equivalents are used because, as suggested by Suzuki [5-1], Pcm is superior to CE where the cooling is fast, while CE is a better indicator than Pcm for slow cooling conditions. Carbon content is also submitted to the networks as an independent parameter because it reflects maximum hardness whereas carbon equivalents indicate the hardenability of the steel. Because of the nature of the transfer function used in the BPN technique, all input and output parameters are normalized between 0.05 and 0.95 before submitting to the network - see Chapter 3, Section 3.3 for details. Figure 5.10 is a schematic sketch of the network used for predicting HAZ hardness. In general, there are five (including a bias node) hidden nodes and one node in each of the hidden and output layers respectively.

Fifty sets of data were set aside for training purposes from the one-hundred and forty available and the remainder (90 sets of data) were used for knowledge verification (Appendix II). The initial weights of the networks were set by random algorithm between

-0.5 and 0.5 [5-15]. Step-declining learning and Fahlman's derivative were employed throughout with initial learning rate, momentum coefficient and RMS error tolerance level set at 0.75, 0.5 and 0.04 respectively. In every case, the results reported are based on the best of three trials.

5.3.2 Results and Discussion

A single-hidden layer network with 4 hidden nodes (including bias node) was used to determine the appropriate number of training patterns required for the HAZ hardness problem. The network was trained with 30, 40 and 50 sets of data (Appendix II). As mentioned above, the remaining 90 sets of data (unseen during the training stage) were used for verifying the network knowledge, i.e., experimental values were compared with the network prediction using the knowledge acquired previously. The results are summarized in Table 5-8. Experimentation suggests that at least 40-50 sets of data were necessary to cover the HAZ hardness problem domain studied in this investigation.

Table 5-8 Network Experimentation for Determining the Appropriate Number of Training Patterns Required for the HAZ Hardness Problem (mean Hmax=309.48VHN).

# of Training Patterns Used	Correlation Coefficient	Absolute Difference (VHN)	
		Mean	Standard Deviation
30	0.955	16.99	14.50
40	0.984	11.99	8.04
50	0.988	8.61	7.21

Note: 1) A single-hidden layer network with 4 hidden nodes was used.
2) The results were based on the best of three trials.

After an appropriate number of training patterns was determined, different network structures were examined for improving the accuracy of the system. In total, five different networks with 50 training data sets were constructed for the study of the accuracy of predicting HAZ hardness using BPN given carbon content, carbon equivalents (Pcm & CE) and $t_{8/5}$. A comparison of network predictions with experimental values using the 90 sets of data (unseen during the learning stage) is shown in Table 5-9. The

knowledge of a single-hidden layer with 4 hidden nodes (including bias node) generates the best results with values of 0.988, 8.61 VHN and 7.21 VHN for the correlation coefficient, mean and standard deviation of absolute difference respectively (Figure 5.11), i.e., mean error of 3%. The weights (knowledge) for this network are listed in

Figure 5.11 Comparison of BPNs predicted hardness with experimental values [5-9].

Appendix IX. The knowledge, listed in Appendix IX, for a 2-hidden layer network with 5 and 3 hidden nodes (including bias nodes) respectively provides comparable prediction values of 0.987 and 9.38 VHN for the correlation coefficient and mean of absolute difference respectively (Figure 5.11). Measured and BPNs (both single- and 2-hidden layer) predicted hardness values are in general agreement.

Table 5-9 Results of BPN Experimentation for Verifying the Accuracy of the Knowledge of Different Network Structures with 50 Training Patterns (mean Hmax=309.48VHN).

# of Hidden Layer(s)	# of Nodes (1st/2nd Layer)	Correlation Coefficient	Absolute Difference (VHN)	
			Mean	S.D.
1	4 ^a	0.988	8.61	7.21
1	5	0.986	9.86	7.54
2	4/3	0.968	13.30	12.25
2	5/3 ^a	0.987	9.38	7.55
2	5/4	0.986	9.86	7.54

Note: 1) Results of 90 test data sets used for the evaluation.

a) Results of these entries are plotted in Figure 5.11.

In Figure 5.12, two steels of high and low carbon equivalents (A1 & A7) were selected to generate characteristic hardness curves (CHCs) predicted by the single-layer network with 4 hidden nodes. Without providing any knowledge of the shape of the curves, the network is able to interpret the input-output relationship which resembles the CHC rather well. This demonstrates one of the principal advantages of BPN versus conventional regression technology which usually requires a pre-defined mathematical relationship.

Figure 5.12 Characteristic hardness curves (CHCs) predicted by BPN for two different steels (A1&A7) - see Appendix II.

5.3.2.1 Alternative Number of Training Patterns

Comparing the knowledge acquired from 40 and 50 training patterns (Table 5-8) shows that both are comparable. When they are further examined in Figure 5.13, both perform similarly at the lower end of the hardness scale. However, there are differences at the higher end of the hardness scale.

Figure 5.13 Comparison of predictions generated with the knowledge of 40 and 50 training patterns.

knowledge using a minimum number of training patterns, a single training pattern of large hardness (steel A7: C = 0.254 wt%, Pcm = 0.359 wt%, CE = 0.560 wt% & hardness = 485 VHN) was added to the set of forty. A single-hidden layer BPN with 4 hidden nodes was used for this investigation.

The results show significant improvement, with correlation coefficient, mean and standard deviation of absolute difference values of 0.985, 10.17 VHN and 7.96 VHN respectively - see Figure 5.14 and Appendix IX for the weights listing. Among other things, this demonstrates the trial and error nature of putting together an effective BPN network in general terms. Characteristic hardness curves generated by the 4-hidden node network with 41 training patterns for steels A1 and A7 are included in Figure 5.15 for

Figure 5.14 Comparison of predictions generated with the knowledge from 41 and 50 training patterns. Note: one of the patterns in the 41 group was specially selected.

Figure 5.15 Characteristic hardness curves of steels A1 and A7 (Appendix II), predicted with the knowledge from 40 and 41 training patterns.

2 OF 2

PM-1 3 1/2"x4" PHOTOGRAPHIC MICROCOPY TARGET
NBS 1010a ANSI/ISO #2 EQUIVALENT

comparison purposes. Although the prediction is not as accurate as the prediction provided by the knowledge of 50 training patterns, where the correlation coefficient, mean and standard deviation of absolute difference are 0.988, 8.61 VHN and 7.21 VHN respectively, the knowledge of 41 training patterns is very acceptable and therefore, 40 training patterns, judiciously selected, would be sufficient for the HAZ hardness problem.

The selection of training patterns is very important. Insufficient training patterns may cause bias in the knowledge. However, increasing the number of training patterns increases the learning time significantly. Keeping the number of training patterns to a minimum is important in practice for efficiency purposes.

5.3.2.2 Using a Single Carbon Equivalent

Although the two carbon equivalents (Pcm & CE) used as input parameters for determining HAZ hardness are believed to be sensitive to different cooling rates [5-1]. Suzuki suggests that there is an approximate relationship [5-22]:

$$P_{cm} = \frac{(2C + CE)}{3} \div 0.005 \quad (5-3)$$

This in turn suggests that only a single carbon equivalent (Pcm or CE) may be possible with the BPN technique. A single-hidden layer network with 4 hidden nodes (including bias node) and a 2-hidden layer network with 5 and 3 hidden nodes (including bias nodes) were trained using the same 50 training data sets - carbon content, Pcm and $t_{8/5}$ or with carbon content, CE and $t_{8/5}$ as input parameters. Fahlman's derivative

Figure 5.16 Comparison of measured and BPNs with 4 hidden nodes and 50 training patterns (using different carbon equivalents) predicted hardness.

and step-declining learning were used. Modified dynamic learning was introduced in the middle of the learning process when required. The initial learning rate, momentum and tolerance level were set at 0.75, 0.5 and 0.04 respectively. Only the best results out of three trials are reported in every case. A summary of the results is included in Table 5-10.

Table 5-10 Comparison of HAZ Hardness Prediction Networks Performance Using Different Input Parameters (mean Hmax=309.48VHN).

Input Parameters (50 train. patterns)	# of hidden nodes	Correlation Coefficient	Absolute Difference (VHN)	
			Mean	S.D.
C, Pcm, CE & $t_{8/5}$ ^{a,b}	4 (1-hidden layer)	0.988	8.61	7.21
C, Pcm & $t_{8/5}$ ^{a,b,c}	4	0.983	10.59	9.34
C, CE & $t_{8/5}$ ^{a,b,c}	4	0.987	9.07	7.27
C, Pcm, CE & $t_{8/5}$ ^{b,c}	5/3 (2-hidden layer)	0.987	9.38	7.55
C, Pcm & $t_{8/5}$ ^{b,c}	5/3	0.985	9.59	8.24
C, CE & $t_{8/5}$ ^{b,c}	5/3	0.985	10.55	7.83
C, CE & $t_{8/5}$ (41) ^{c,d}	4 (1-hidden layer)	0.983	12.15	7.69

Note: a) Results of these entries are plotted in Figure 5.16.

b) 50 training sets.

c) Weights are listed in Appendix IX.

d) 41 training sets.

In general, all seven networks perform very well with mean error less than 4%. In most cases, a single-hidden layer network out-performs the 2-hidden layer network given identical input parameters. Therefore, the second hidden layer might be considered redundant. On the other hand, the HAZ hardness prediction generated from a knowledge of 41 training patterns is comparable with those generated from 50 training patterns (Table 5-10). This is further evidence that about 40 training patterns may be sufficient for generalizing the HAZ hardness problem.

Where only one carbon equivalent is incorporated, the networks using CE seem more accurate than those using Pcm. This is because Pcm was developed based on HT50 to HT100 steels with carbon content ranging between 0.09 to 0.17% which is about half of the carbon range in this investigation. Furthermore, as suggested by Suzuki [5-1], Pcm is more appropriate for fast cooling situations, e.g., $t_{8/5}$ less than 8.5 seconds, whereas

more than half of the cooling times in the Yurioka data base (test data used here) are beyond this range.

The networks that contain both carbon equivalents (Pcm & CE) always outperform those where only one is incorporated. Although Pcm is mostly used for fast cooling situations, the hardening effects of silicon (Si) and Boron (B) are included whereas they are not part of CE. Therefore, Pcm can be used to supplement to CE when the effects of silicon and boron become significant or fast cooling is encountered.

5.3.2.3 Including the Effect of Silicon (Si)

In the ideal case, instead of using carbon equivalents as input to the network, the actual alloying content should be incorporated because all carbon equivalents are themselves the result of regression analyses. However, the number of training patterns required is drastically increased. On the other hand, the carbon equivalents (Pcm and CE) used in this investigation are well established and are highly regarded in the welding industry for evaluating cold cracking susceptibility.

The results of the last section would suggest that Pcm might be dropped as one of the input parameters in most cases. However, this results in ignoring the effects of silicon and boron which can give rise to some inaccuracy for certain steels where these two alloying elements are important. Yet, including both Pcm and CE may hamper BPN learning and may result in conflicts between fast and slow cooling situations. Therefore, the purpose of this part of the study is to use carbon content, silicon content, CE and $t_{8/5}$ to predict HAZ hardness. Although Pcm reflects both silicon and boron, only silicon is included in this investigation because it is believed that silicon is a more common alloying element in steel than boron.

A single-hidden layer network with 4 hidden nodes (including bias node) was used for this study. Forty-one training patterns were used to generate the BPN knowledge. The range of silicon content in the training patterns set is between 0.15 to 0.45 wt% with all other parameter ranges the same. The results are tabulated in Table 5-11 together with the

results using knowledge generated with Pcm & CE and then CE alone for comparison. The weights of the network are listed in Appendix IX. Comparing the prediction knowledge of different networks with 41 training patterns, the one used both Pcm and CE is still favorable. However, the induction of Si content increases the accuracy when compared to using a single carbon equivalent.

Table 5-11 Results of HAZ Hardness Predictions Generated to Show the Effect of Silicon as an Explicit Parameter (single-hidden layer network with 4 hidden nodes was used).

Input Parameters	# of Training Patterns	Correlation Coefficient	Absolute Difference (VHN)	
			Mean	S.D.
C, Pcm, CE & $t_{8/5}$	50	0.988	8.61	7.21
C, Pcm, CE & $t_{8/5}$	41	0.985	10.17	7.96
C, CE & $t_{8/5}$	41	0.983	12.15	7.69
C, Si, CE & $t_{8/5}$	41	0.986	11.36	7.04
C, Si, B, CE & $t_{8/5}$	41	0.986	10.74	8.02

Note: 1) mean Hmax = 309.48 VHN.

Attempts have been made to include boron as an input parameter using the same network structure suggested above - see Appendix IX for weights listing. The range of boron content in the Yurioka hardness data base is ranging from 0.00 to 0.0018 wt%. However, the majority of steels (11 out of 14) reported in Yurioka's study contain less than 0.0006 wt% boron. Although predictions using the knowledge with explicit boron content are comparable to other networks tested (Table 5-11), the knowledge is suspect because the unevenly distributed boron content in the training set may cause knowledge bias.

5.3.2.4 Comparing BPN Prediction with the Regression Methods

There are several HAZ hardness regression models used in industry. The most common ones are those generated by Yurioka [5-3,9], Suzuki [5-1,22] and Terasaki [5-2].

among others. A comparison of the performance of the regression relationships of Suzuki [5-1] and Terasaki [5-2] with the BPN technique is given in Table 5-12 and Figure 5.17.

Table 5-12 Comparison of Different HAZ Hardness Prediction Models
(mean Hmax=309.48VHN).

Model	Correlation Coefficient	Absolute Difference (VHN)	
		Mean	S.D.
Suzuki Pcm, NSC-S(Pcm)	0.979	10.61	11.07
Suzuki CE, NSC-S(CE) ^a	0.982	11.89	8.81
Terasaki II	0.973	12.99	12.87
BPN, Pcm & CE with 50 patterns ^{a,b}	0.988	8.61	7.21
BPN, CE with 41 patterns ^b	0.984	12.15	7.50
BPN, Si & CE with 41 patterns ^{a,b}	0.986	11.36	7.04

Note: a) Results of these entries are also plotted in Figure 5.17.

b) Weights of this 4-hidden node single-hidden layer network are listed in Appendix IX.

In general, the BPNs predict more accurately than the conventional regression models. The 4-hidden node BPN with 50 training patterns has the best prediction with 3% mean error. Because the Suzuki NSC-S(Pcm), and the Suzuki NSC-S(CE) [5-1], hardness models are based on

Figure 5.17 Comparison of different HAZ hardness prediction models.

Yurioka hardness data base and only a single carbon equivalent (Pcm and CE respectively) is used (Chapter 1, Section 1.1.1.1), the results may be compared directly. The Terasaki II

model [5-2] is one of the most recent independent hardness regression relationships (Appendix X). Differences between the BPN approach and that of Terasaki II may be expected because different hardness data bases were used in the development of these models.

5.3.2.5 Applying BPN Knowledge to Independent Data

The experimental data used for this part of the investigation are extracted from a study by the AMCA Corporation [5-14]. Submerged-arc bead-on-plate welds were deposited with a range of heat input for the three steels as documented in Appendix V.

Table 5-13 Comparison of BPN and Regression Model Predictions Applied to the AMCA [5-14] Hardness Data (mean Hmax=283.78VHN).

Model	Correlation Coefficient	Absolute Difference (VHN)	
		Mean	S.D.
Suzuki NSC-S(CE) ^a	0.916	20.03	10.97
Terasaki II	0.829	25.96	18.72
BPN Pcm & CE-50 patterns ^b	0.827	24.79	14.08
BPN Pcm & CE-41 patterns ^b	0.856	22.64	14.43
BPN Si & CE-41 patterns ^{a,b}	0.931	16.81	12.08

Note: a) CHCs of these entries are plotted in Figures 5.18, 19 & 20.
b) A 4-hidden node network is used.

Predictions from three different networks and two regression models, Suzuki NSC-S(CE) and Terasaki II, are compared with the experimental AMCA data. All BPNs used were 4-hidden node single-layer system- but with different training as described in Table 5-13. The knowledge of the first network was acquired from 50 training patterns with C, Pcm, CE and $t_{8/5}$ as

Figure 5.18 BPN and NSC-S(CE) predicted CHCs with experimental values for steel B1 from AMCA data.

input parameters - see Appendix IX for weights listing. The knowledge of the second BPN was the result of training from 41 training patterns with C, Pcm, CE and $t_{8/5}$ as input parameters - see Appendix IX for the weights listing. The knowledge of the third network was generated from 41 training patterns with C, Si, CE and $t_{8/5}$ as input parameters - see Appendix IX. A summary of the results is listed in Table 5-13. BPN (Si & CE - 41 patterns) and NSC-S (CE) characteristic hardness curves are also included as Figures 5.18, 19 & 20.

The BPN method seems comparable to the Suzuki NSC-S(CE) regression model and somewhat better than the Terasaki II hardness model as indicated by correlation coefficient, mean and standard deviation of absolute difference. Furthermore, when comparing the predictions generated

from the knowledge of 50 and 41 training patterns (first and second networks), seemingly the extra 9 patterns seem to cause the network knowledge bias to the Yurioka data base since it provides the best prediction with Yurioka data and the worst prediction with AMCA data. On the other hand, the knowledge generated with Si content as an independent input instead of through Pcm seems the most accurate (mean error of 6%)

Figure 5.19 BPN and NSC-S(CE) predicted CHCs with experimental values for steel B2 from AMCA data.

Figure 5.20 BPN and NSC-S(CE) predicted CHCs with experimental values for steel B3 from AMCA data.

when compared to other models. Therefore, this network is considered to be a more general solution to HAZ hardness problem.

A more detailed comparison of BPN using the knowledge of 41 training patterns with C, Si, CE and $t_{8/5}$ as input parameters and NSC-S (CE) hardness prediction model is shown by the CHCs in Figures 5.18, 19 & 20. In general, the BPN method predicts more accurately than the Suzuki NSC-S (CE) model for steel B1. For steels B2 and B3, the BPN method generates similar results when $t_{8/5}$ is larger than 8s. In conclusion, the BPN method examined is recommended with minor difficulties in fast cooling situations.

5.3.2.6 Adapting Existing Hardness Knowledge for Specific Job Shop Functions

As shown in Table 5-13 and Figures 5.19 & 20, predictions for the AMCA steels are not as accurate as those taken from the Yurioka data base in the fast cooling range although the applicable ranges of both data sets are within the knowledge limits. It should be remembered at this point that the knowledge was acquired from the data contained in the Yurioka data base. It is believed that there are uncontrollable and unforeseeable differences, e.g., different steels, experiment setup, workmanship, etc., between the two studies and these factors would not be accounted for during network training. For the same reason, it is recommended that independent job shop HAZ hardness knowledge should be used for the training base where possible. However, the time and cost involved may be considerable.

Two hardness values from the AMCA study were added to the original training data set (41 patterns) to examine the possibility of improving the existing hardness knowledge (4-hidden node, single-hidden layer network with C, Si, CE and $t_{5/8}$ as input parameters) for predicting AMCA steels. The extra data were extracted from steels B2 and B3 with 3.1 s of $t_{8/5}$ respectively - see Appendix V for more information about the hardness data. A summary of results is listed in Table 5-14.

Table 5-14 Comparison of the Results of Networks Trained with (43 patterns) and without (41 patterns) Additional AMCA Hardness Data.

Test Data Base	Model	Correlation	Absolute Diff. (VHN)	
		Coefficient	Mean	S.D.
AMCA Steels (mean Hmax=284VHN)	Suzuki NSC-S (CE)	0.916	20.03	10.97
	BPN (41 patterns)	0.931	16.81	12.08
	BPN (43 patterns)	0.948	16.24	9.86
Yurioka Steels (mean Hmax=309VHN)	BPN (41 patterns)	0.986	11.36	7.04
	BPN (43 patterns)	0.976	13.12	9.58

Notes: a) CHCs of these entries are also shown in Figures 5.21, 5.22 and 5.23.

b) The input parameters of the 4-hidden node BPN are C, Si, CE and $t_{8/5}$.

The results of the predictions using the adapted knowledge are very promising - correlation coefficient, mean and standard deviation of absolute difference of 0.948, 16.24 VHN and 9.86 VHN respectively when predicting the AMCA data. This may be compared to the results using the original knowledge ($r=0.931$, mean=16.81 VHN & SD=12.08 VHN). By adding extra information to the original knowledge there is considerable improvement. Furthermore, the adapted knowledge out-performs the best regression model - Suzuki NSC-S(CE), ($r=0.916$, mean=20.03 VHN & SD=10.97 VHN). Predicted CHCs of the three steels are shown in Figures 5.21, 5.22 & 5.23.

On the other hand, when the adapted knowledge was used to predict the original Yurioka test data set, correlation coefficient, mean and standard deviation of absolute difference of 0.976, 13.12 VHN and 9.58 VHN are reported respectively. Compared to the results using the original knowledge ($r=0.986$, mean=11.36 VHN & SD=7.04 VHN), the addition of the two data sets does not affect the original network prediction from Yurioka data.

Figure 5.21 BPN (with adapted knowledge) and NSC-S(CE) predicted CHCs with experimental values for steel B1 from AMCA data.

Figure 5.22 BPN (with adapted knowledge) and NSC-S(CE) predicted CHCs with experimental values for steel B2 from AMCA data.

Figure 5.23 BPN (with adapted knowledge) and NSC-S(CE) predicted CHCs with experimental values for steel B3 from AMCA data.

However, considerable caution must be excised when extending the boundaries of the original knowledge. If the additional data are not fully compatible, i.e., there may be fundamental differences in the data bases, merging of data may cause conflict in the training data set and may increase the network training time without any benefit. Furthermore, the extra data may lead to incorrect knowledge. Therefore, it is always wise to verify the adapted knowledge with both the original test data and the new data set.

5.4 Combining $t_{8/5}$ and HAZ Hardness Prediction BPNs

By combining the BPN knowledge of $t_{8/5}$ and HAZ hardness, an estimate HAZ hardness directly from the welding conditions and base metal chemical composition is possible. A schematic diagram of the combined network is illustrated in Figure 5.24. The input parameters to the system are arc current, voltage, wire travel speed, process efficiency, plate temperature, plate thickness and steel chemical composition (in terms of C, Si and CE). A HAZ hardness is directly output.

Figure 5.24 A schematic diagram showing the combined network for predicting HAZ hardness from the welding conditions and steel chemical composition.

The BPN for predicting $t_{8/5}$ (given arc current, voltage, wire travel speed, process efficiency, plate thickness and plate temperature) is a 2-hidden layer network with 5 and 3 hidden nodes respectively and the knowledge is based on the training of 20 patterns as described in Section 5.2 of this Chapter and the weights are listed in Appendix VIII. The HAZ hardness prediction (given C, Si, CE and $t_{8/5}$) is performed by a single-hidden layer 4 hidden nodes network with knowledge acquired from 41 training patterns as detailed in Section 5.2 of this Chapter and the weights are listed in Appendix IX. The prediction of the combined network is compared with estimations from Adams' cooling time models, the Suzuki NSC-S(CE) (Chapter 1, Section 1.1.1.1), and the Terasaki II regression models (Appendix X). The experimental data used are taken from the AMCA study [5-14] (partly used in Section 5.3.2.5) - see Appendix V. The study reports the SAW $t_{8/5}$ and the HAZ hardness of three microalloy steels for various welding conditions.

5.4.1 Results and Discussion

A 95% process efficiency of SAW was used for cooling time calculation. The results of the $t_{8/5}$ prediction network were submitted to the HAZ hardness prediction networks. The comparison of the predicted measured HAZ hardness using BPNs and regression models (Adams' cooling time models and Suzuki NSC-S(CE) HAZ hardness

prediction model) is shown in Figure 5.25. A summary of the results is shown in Table 5-15. Furthermore, the results from the adapted knowledge (previous Section) is included for comparison.

Table 5-15 Summary of Hardness Predicted by BPNs and Regression Models for AMCA Experimental Data [5-14] (mean Hmax=283.78VHN) - see Appendix V.

$t_{8/5}$ Model	Hardness Model	Correlation Coefficient	Absolute Difference (VHN)	
			Mean	S.D.
BPN	BPN ^a	0.893	18.06	15.13
BPN	adapted BPN ^a	0.919	17.57	11.86
Adams'	NSC-S(Pcm)	0.869	23.45	17.63
Adams'	NSC-S(CE) ^a	0.915	19.90	11.85
Adams'	Terasaki II	0.843	27.59	20.36

Note: a) Results of these entries are plotted in Figure 5.25.

The overall hardness prediction using the combined network given the welding conditions and steel chemical composition out-performs the HAZ hardness estimated by using Adams' cooling time models and Suzuki NSC-S (CE) HAZ hardness prediction model with an overall mean error of 6% versus 7%. This finding is consistent with the previous observation. The combined network predicts more accurately in the low hardness region, e.g. less than 350 VHN, that it does for hardness values in excess of this value.

Figure 5.25 Comparison of BPNs (original=BPN-41, adapted=BPN-43) and regression models (Adams' & Suzuki) predicted HAZ hardness with AMCA data.

Moreover, when intermediate results from the combined network ($t_{8/5}$) are examined, the network prediction is still favorable (Table 5-16 and Figure 5.26).

However, because the $t_{8/5}$ prediction using BPN is more sensitive to plate thickness than the Adams' formulations, different cooling times are recorded for the same heat input but various plate thickness, see Figure 5.26.

Figure 5.26 Comparison of $t_{8/5}$ predicted by using BPN and Adams' cooling time models for AMCA data.

In general, the combined network (2-hidden layer, 5 & 3-hidden node BPN with 20 training patterns for predicting $t_{8/5}$ given heat input, plate thickness & plate temperature and 1-hidden layer, 4-hidden

node BPN with 41 training patterns for predicting HAZ hardness given C, Si, CE & $t_{8/5}$) can predict HAZ hardness directly from welding conditions and steel chemical composition with reasonable accuracy. Furthermore, the use of adapted hardness knowledge demonstrates that BPN is flexible and can be used for example in situations where the use of regression relationships would be very cumbersome.

Table 5-16 Comparison of BPN and Adams' Cooling Time Models Predicted $t_{8/5}$ for AMCA Experimental Data [5-14] (mean $t_{8/5}$ =22.37s) - see Appendix III.

Model	Correlation Coefficient	Absolute Difference (s)	
		Mean	S.D.
2-hidden layer, 5&3-hidden node BPN with 20 training patterns	0.992	2.02	1.77
Adams' cooling time models	0.999	3.95	3.61

5.5 Chapter Summary

A 2-hidden layer network with 5 and 3 hidden nodes (including bias nodes) respectively is recommended for predicting 800 to 500 °C cooling time from heat input (calculated from arc current, voltage, wire travel speed and process efficiency), plate thickness and plate temperature as input parameters. Furthermore, a single-hidden layer network with 5 hidden nodes (including bias node) is found to be comparable to the 2-

hidden layer network for predicting $t_{8/5}$. About 20 training sets are required to achieve the appropriate knowledge. The training set and weights (knowledge) of the network described are listed in Appendices III and VIII respectively. Although the cooling time knowledge is acquired from the SAW process ($\eta_{\text{proc}} = 95\%$ of SAW), it is applicable to other welding processes by assigning an appropriate process efficiency. In general, predicting cooling times with the BPN technique is found to be more accurate than predictions made with the Adam's cooling models. This follows because most of the assumptions required in the Adam's models are not necessary with the BPN technique.

In order to predict HAZ hardness using the BPN technique, the input parameters suggested are carbon content (C), silicon content (Si), the International Institute of Welding's carbon equivalent (CE) and $t_{8/5}$. Pcm (Ito's carbon equivalent) can be left out if Si content is included. A single-hidden layer network with 4 hidden nodes (including bias node) generates the most accurate hardness predictions. Predictions with a network of 2 hidden layers with 5 and 3 hidden nodes (including bias nodes) is comparable to that of the single-hidden layer network. About 50 sets of data are necessary for generating the required knowledge to cover the HAZ hardness problem studied. However, it seems that the knowledge acquired from 41 sets of training data is more applicable for general situation, i.e., low alloy steels that are not covered in Yurioka data base. The training sets and weights of the two networks are listed in Appendices II and IX respectively. Moreover, it was found that the existing hardness knowledge can be adapted to cover another set of steels by adding a few judiciously selected patterns from the new data set. The BPNs generated in this thesis out-performed comparable HAZ hardness regression models in most cases.

Finally, combining the $t_{8/5}$ (2-hidden layer 5&3-hidden node) and HAZ hardness (single-hidden layer 4-hidden node) networks can be used to predict HAZ hardness directly from the welding conditions (arc current, voltage, wire travel speed, plate thickness, plate temperature and process efficiency) and steel chemical composition (in terms of C, Si & CE) with acceptable accuracy.

Chapter Six

Predicting Weld Bead Geometry with BPNs

6.1 Introduction

As mentioned in Chapter 1, in a fabrication operation both the production costs and welded joint strength depend greatly on weld bead geometry. Therefore, it is apparent that models for predicting the bead geometry (bead width, bead height, penetration, deposit area and fusion area) are of considerable importance to welding or materials engineers. The purpose of this investigation is to study the use of BPN as an alternative to traditional regression techniques used by many other researchers - see Chapter 1, Section 1.1.2, for a discussion of the importance of predicting weld bead size.

About 200 bead-on-plate GMA welds with different shielding gases (C-25, 25% carbon dioxide and 75% argon, and M-2, 2% oxygen and 98% argon) were prepared by the Northern College in Kirkland Lake [6-1] (included in Appendix VI) for this study. The data base recorded weld bead shapes for various welding conditions (shielding gas, arc current, voltage, wire travel speed and plate thickness) with electrode extension, electrode diameter and electrode polarity kept constant (19.05 mm, 0.89 mm and negative electrode polarity respectively). The welding condition ranges are listed in Table 6-1. Of considerable significance is that the welding conditions cover the three heat flow ranges (2-D, 2.5-D and 3-D) suggested by Adams et al [6-2].

Table 6-1 Ranges of Arc Current, Voltage, Wire Travel Speed and Plate Thickness Used in the GMAW Bead Geometry Investigation - See Appendix VI for Details.

Shielding Gas	Range	Current (amperes)	Voltage (volts)	Speed (mm/s)	Thickness (mm)	Heat Input ^a (kJ)
C-25	Min.	160	20	3.39	6.35	0.53
	Max.	320	42	11.0 ^a	15.87	1.22
M-2	Min.	160	24	4.23	6.35	0.45
	Max.	320	37	9.31	15.87	1.43

Note: a) $\eta_{\text{proc}} = 70\%$

The pictorial representation of the weld shape is conventionally estimated by fitting a semi-ellipse for the upper-bead (determined by the bead height and width) and a cosine curve for the lower-bead (determined by the bead width and penetration) [6-3,4,5,6]. However, these pictorial representations usually overestimate the deposit and plate fusion areas [6-3,4]. New algorithms are examined in this investigation to improve the approximations of the pictorial representation for both the upper- and lower-bead shapes.

Because the experimental data base is easily identified in two distinctive groups (bead-on-plate weld bead shapes of C-25 and M-2 shielding gases), the weld bead geometry problem is divided into two sub-problems for practical reasons. The bead geometry using C-25 shielding problem was studied first. Based on the experience accumulated from this study, the M-2 shielding bead geometry problem was then investigated. Finally, in order to examine the ability of BPN for handling non-quantitative parameters, e.g., different shielding, C-25 and M-2 shielding bead geometry problems were combined and analyzed as a single problem. The input parameters to the combined network include shielding gas, arc current, voltage, wire travel speed and plate thickness. The output parameters are the weld bead dimensions as described previously.

Since controlling weld bead geometry by adjusting the welding conditions is a major concern to industry, the use of BPNs for predicting the welding parameters (arc current, voltage and wire travel speed) given the weld bead dimensions (e.g., bead height, bead width, penetration, plate thickness... etc.) has also been investigated using the C-25 shielding weld bead data. This may be termed the inverse weld bead geometry problem.

In addition to arc current, voltage and wire travel speed, plate thickness was also varied to study its effect on weld bead geometry. Plate thickness is a parameter that most practitioners would contend is very important; yet it has not been included in prior regression studies. Thick section welding has been assumed in all previous investigations which limits the usefulness of those analyses.

6.2 Algorithms for Estimating the Weld Bead Shape

As mentioned above, a semi-ellipse has been used to represent the deposit area. However, it is found that the area is usually overestimated by this configuration. Therefore, a parabolic curve is proposed:

$$y = -ax^2 + b \quad (6-1)$$

and
$$a = \frac{4 \cdot BH}{BW^2} \quad (6-2)$$

$$b = BH \quad (6-3)$$

At the same time the lower section of weld has conventionally been represented by a cosine relationship. However lower-bead area is usually overestimated [6-3,4] because among other things the bay that appears in most welds is not incorporated into the shape algorithm. Including a bay approximation for the pictorial representation of the lower-bead shape would seem to be a necessary improvement. In the lower-bead shape algorithm, the location and depth of the bay is defined by the angle (θ) and length (l^b) as illustrated in Figure 6.1. With these extra bead geometry dimensions, the lower-bead can be approximated by two overlapping parabolas with one rotated 90° as shown in Figure 6.2.

$$y^2 = c(x + d) \quad (6-4)$$

and
$$y = ex^2 - f \quad (6-5)$$

$$c = \frac{y_1^2}{-x_1 + BW / 2} \quad (6-6)$$

$$d = BW / 2 \quad (6-7)$$

Figure 6.1 Approximation of lower-bead shape with bead bay parameters included.

Figure 6.2 Approximation of lower-bead using two parabolas.

$$e = \frac{-y_1 + \text{Pene}}{x_1^2} \quad (6-8)$$

$$f = \text{Pene} \quad (6-9)$$

where (x_1, y_1) is the point of intersection of the two parabolas:

$$x_1 = f \sin \theta \quad (6-10)$$

$$y_1 = f \cos \theta \quad (6-11)$$

In a shallow weld, the lower-bead shape is more like a single parabola. The double-parabola can accommodate this situation since a shallow weld is usually accompanied by a relatively large bay length and hence the double-parabola becomes similar to a semi-ellipse.

6.2.1 Results and Discussion

The experimental data based on C-25 shielding was used to evaluate the weld bead shape algorithms. The use of a parabola to approximate the upper-bead improves by some 40% with no significant change in correlation factor - see Table 6-2 and Figure 6.3. It is, therefore apparent that a parabola is a better representation for the upper-bead shape than the conventional semi-ellipse for C-25 shielding.

Figure 6.3 Comparison of the actual and pictorial approximated (semi-ellipse and parabola) deposit areas with C-25 shielding.

When the measured and pictorial approximated (both semi-ellipse and parabola) deposit areas with C-25 shielding are examined in detail (Figure 6.3), the use of the semi-ellipse always overestimates the upper-bead shape, while the use of a parabola tends to underestimate the actual deposit area. In general, it seems that the upper-bead shape is approximated more closely by a parabola if the deposit area is less than 40 mm^2 . On the other hand, a semi-ellipse seems to be more suitable when the deposit area is larger than 40 mm^2 . By using a parabola for the smaller welds and an ellipse for the larger welds there is significant improvement in the correlation coefficient, mean and standard deviation of absolute difference as listed in Table 6-2.

Table 6-2 Comparison of Different Upper-Bead Area Algorithms.

Shielding	Algorithm	Correlation Coefficient	Absolute Difference (mm^2)	
			Mean	S.D.
C-25 (mean $A_1=26.98\text{mm}^2$)	Semi-ellipse	0.945	3.29	2.13
	Parabola	0.945	1.96	1.44
	Parabola-ellipse	0.949	1.91	1.39
M-2 $A_1=28.76\text{mm}^2$	Semi-ellipse	0.959	1.91	1.97
	Parabola	0.959	3.23	1.87

When the pictorial approximated upper-bead shapes with M-2 shielding are compared with measured values, the semi-ellipse approximation is more accurate in general (Table 6-2 and Figure 6.4). It is apparent from Figure 6.4 that the semi-ellipse usually overestimates the actual deposit area and a parabola approximation has a tendency to understate the actual deposit area.

Figure 6.4 Comparison of the actual and pictorial approximated (semi-ellipse and parabola) deposit areas with M-2 shielding.

The C-25 shielding weld bead shape data were used to determine the appropriate lower-bead bay angle since C-25 shielding lower-bead shapes are more symmetrical and there are deeper bays in general. One hundred and ninety-two bay angles were measured (left and right sides of the lower-bead, 96 shapes X 2 angles/shape = 192 angles) and

Figure 6.5 Frequency diagram of the bay angle of C-25 shielding weld bead shape.

listed in Appendix VI. From the frequency diagram (histogram), Figure 6.5, most of the bay angles occur in the range of 15° and 30° with an overall mean value of 22.4° and standard deviation of 8.2°. Therefore, a value of $\pi/8$ (or 22.5°) is assumed to be the nominal bay angle (θ) for pictorial representation purposes.

Having fixed the bay angle at 22.5°, the area of lower-bead shape depends only on the bead width (BW), penetration (Pene) and bay length ($l^{22.5}$). The results of the areas approximated by conventional cosine and proposed double parabolic curves are tabulated in Table 6-3.

In general, a lower-bead bay angle of 22.5° should also be applied to the weld bead shape with M-2 shielding since the bay effect improves the lower-bead shape approximations significantly (Table 6-3). In addition to the double-parabola, the use of a double-ellipse to represent the lower-bead was also examined with the major axis of one ellipse defined by the bead width and the minor axis of the other ellipse determined by the penetration. The intersection of the two ellipses was fixed by the bay parameters ($\theta=22.5^\circ$ & $l^{22.5}$). A comparison of this approximation with measured deposit area is listed in Table 6-3.

Table 6-3 Comparison of Different Lower Bead Area Algorithms.

Shielding	Algorithm	Correlation Coefficient	Absolute Difference (mm ²)	
			Mean	S.D.
C-25 (mean A2=19.70mm ²)	Cosine Curve	0.898	2.73	2.67
	Double-Ellipse	0.965	1.77	1.53
	Double-Parabola	0.968	1.19	1.23
M-2 (mean A2=14.71mm ²)	Cosine Curve	0.750	4.83	2.86
	Double-Ellipse	0.926	2.44	1.79
	Double-Parabola	0.931	1.78	1.72

There is no doubt that the use of a double-parabola (with $\theta=22.5^\circ$) improves the accuracy of lower-bead shape approximation for both C-25 and M-2 shielding. This combination out-performs both the cosine and double-ellipse algorithms in the entire plate fusion area range. On the other hand, using the double-ellipse approximates the lower-bead shape better

than a cosine curve but it is not as accurate as the double-parabola

(Figures 6.6 & 6.7). It should be

noticed that there is almost no

difference between the double-ellipse

or double-parabola

pictorial

representation

where the plate

fusion area is

Figure 6.6 Comparison of the actual and represented (cosine, ellipses and parabolas) plate fusion areas with C-25 shielding.

smaller than 25 mm^2 . However, in the larger plate fusion area region, the use of the double-parabola is obviously more suitable. Some weld bead shape comparisons using the newly proposed algorithms (parabola-ellipse for the upper-bead and double-parabola for the lower

Figure 6.7 Comparison of the actual and represented (cosine, ellipses and parabolas) plate fusion areas with M-2 shielding.

Figures 6.8 & 6.9 and Appendix XI to provide some idea of how faithful the pictorial representation is to the actual shape.

Figure 6.8 Experimental and approximated weld shape comparison (C-25 shielding).

Figure 6.9 Experimental and approximated weld shape comparison (M-2 shielding).

6.2.1.1 Comments about Bead Shape Continuity

It is generally believed that the real weld bead shape is continuous. However, in the case of parabola upper-bead (C-25 shielding) and double-parabola lower-bead approximations, the weld shape becomes first degree discontinuous at the plate surface where the upper- and lower-beads meet (Figure 6.10). Furthermore, the proposed lower-bead algorithm (double-parabola) introduces

Figure 6.10 A schematic sketch of the approximated weld shape showing the discontinuities.

first degree discontinuity at the bay where the two parabolas intersect. A discontinuity between the upper- and lower-beads can be justified since most of the experimental weld shapes show sharp corners at the intersections. Nonetheless, the lower-bead double-parabola algorithm is just a close approximation. The discontinuity introduced by the intersection of the two parabolas is not realistic and can be smoothened by adding an extra curve. However, the extra complexity may not be justified because shape detail is usually not necessary.

6.3 Predicting C-25 Shielding Weld Bead Geometry

For predicting the C-25 shielding weld bead geometry using BPNs, thirty out of the original ninety-six sets of C-25 shielding weld bead geometry data were selected for training and the remainder was used for knowledge verification. They are listed in Appendix VI. The input to the BPN is arc current, voltage, wire travel speed, plate thickness and a bias value (which is always one - see Chapter 3, section 3.2.6). The output parameters are the weld bead dimensions, i.e., bead width, bead height, penetration,

lower-bead bay length (the bay angle is assumed to be constant - 22.5°), deposit area and plate fusion area.

A schematic representation of the network is shown in Figure 6.11 with the hidden network structure represented as a variable. The hidden network structure may contain either one or two hidden layers with a bias node. The output of the network is the relevant weld bead dimension, e.g., head width, bead height, penetration, bay length at 22.5° , deposit or plate fusion areas.

Figure 6.11 A schematic diagram of the BPN used for solving the weld bead geometry problem.

The initial weights were selected randomly between -0.5 and 0.5 as described in Chapter 3, Section 3.2.1. Step-declining learning and Fahlman's derivative were used for all tests. A moderate learning rate, momentum coefficient and RMS tolerance value were used (0.5, 0.25 and 0.05 respectively) and all data were normalized between 0.1 and 0.9 (Chapter 3, Section 3.3). In all tests, only the results of the best of three trials are reported because of the nature the of BPN method (Chapter 3, Section 3.3.1). All networks were initially trained with 21 (out of 30) sets of weld bead geometry data. The number of training patterns was increased until the desired accuracy was achieved - see Appendix VI.

6.3.1 Results and Discussion

It is likely that the relationships between the input welding conditions and the output weld bead dimensions would have different levels of complexity. Therefore, instead of using one large network to solve the entire C-25 shielding weld bead geometry problem, it is further simplified to six independent networks, i.e., each network is responsible for predicting one and only one weld bead dimension.

6.3.1.1 Bead Width - C-25 Shielding

Using a single-hidden layer network with five hidden nodes (including a bias node) and 21 training patterns, the network is able to predict bead width with reasonable precision (correlation coefficient = 0.903 and mean of difference = 0.60 mm or mean percentage error = 5%) - see Table 6-4 and Figure 6.12. The weights for this network are listed in Appendix XII.

Figure 6.12 Comparison of BPN (single hidden layer and 5 hidden nodes) predicted bead width with experimental values (C-25).

Table 6-4 A Summary of (C-25) Bead Width Predictions Generated by Different Networks Using 21 Training Patterns (mean BW=12.32mm).

# of Hidden Layers	# of Nodes ^a (1st/2nd layer)	Correlation Coefficient	Absolute Difference (mm)	
			Mean	S.D.
1	4	0.852	0.81	0.76
1	5 ^b	0.903	0.60	0.55
2	5/3	0.858	0.77	0.75

Notes: a) Bias node is included.

b) Weights of this network are listed in Appendix XII.

6.3.1.2 Bead Height - C-25 Shielding

The welding parameters and bead height are believed to be loosely related and there would likely be more uncertainty than the bead width-welding parameters relationship. In fact all networks trained with 21 and 23 patterns failed to perform the prediction accurately. Only the knowledge of a single-hidden layer network with 5 hidden nodes (containing bias node) acquired from 28 training patterns yielded acceptable results with a mean percentage error of 7% - see Table 6-5 and Figure 6.13. Weights of the network are included in Appendix XII. Networks with various

Figure 6.13 Comparison of BPN (single hidden layer and 5 hidden nodes) predicted bead height with experimental values (C-25).

Table 6-5 A Summary of (C-25) Bead Height Predictions Generated by Different Networks (mean BH=3.11mm).

# of Training Patterns	# of Hidden Layers	# of Hidden Nodes ^a	Correlation Coefficient	Absolute Difference (mm)	
				Mean	S.D.
21	1	5	0.372	0.57	0.41
	2	5/3	0.403	0.50	0.43
	2	8/6	0.492	0.53	0.42
23	1	5	0.640	0.32	0.30
	2	5/3	0.616	0.29	0.31
	2	8/6	0.692	0.29	0.27
28	1	5 ^b	0.730	0.22	0.23
	2	5/3	0.617	0.27	0.31
	2	6/4	0.697	0.23	0.25
	2	8/6	0.601	0.29	0.31

Notes: a) Bias node is included.

b) Weights of this network are listed in Appendix XII.

hidden structures (maximum of 10 hidden nodes were used in each hidden layer) were trained 30 patterns. However, all learning failed to converge within 30000 epochs.

Where the network is kept, it is apparent that the knowledge acquired from 28 patterns always performs more accurately than the knowledge from either 21 or 23 patterns (Table 6-5). This strongly suggests that the selection of the number of training patterns should be done before choosing the hidden structure. In the end, it would seem that 18 training sets and a single-hidden layer 5-hidden node structure provides the best bead height prediction.

6.3.1.3 Penetration - C-25 Shielding

The knowledge gained from 21 training patterns was found to be adequate for predicting penetration. A two-hidden layer network with 8 and 6 hidden nodes (including bias nodes) was found to be the most appropriate structure with (mean percentage error of 12%). Values of 0.839 and 0.39 mm for the correlation coefficient and mean of difference respectively are reported in Table 6-6 and a plot of measured versus predicted penetration is shown in Figure 6.14. The network weights are listed in Appendix XII.

Figure 6.9 Comparison of BPN (2 hidden layers and 8&6 hidden nodes) predicted penetration with experimental values (C-25).

Table 6-6 A Summary of (C-25) Penetration Predictions Generated by Different Networks Using 21 Training Patterns (mean Pene=3.25mm).

# of Hidden Layers	# of nodes (1st/2nd layer) ^a	Correlation Coefficient	Absolute Difference (mm)	
			Mean	S.D.
1	4	0.802	0.45	0.45
1	5	0.821	0.44	0.42
2	5/3	0.733	0.62	0.57
2	8/6 ^b	0.838	0.39	0.44

Notes: a) Bias node is included.

b) Weights of this network are listed in Appendix XII.

6.3.1.4 Bay Length - C-25 Shielding

As defined previously in Section 6.2, the use of double-parabola lower-bead shape algorithm requires bay information about the lower-bead. The bay length is the length from the center to the edge of the weld bead at 22.5°. Using only 21 sets of training data, reasonable prediction (7% mean error) could be obtained with a 2-hidden layer 4&3 hidden node (including bias node) network. The comparison of the network predicted bay length with the measured values is shown in Figure

Figure 6.15 Comparison of 2-hidden layer 4&3-hidden node BPN predicted bay length with experimental values (C-25).

6.15. The correlation coefficients and mean of absolute difference of different network structures tested are listed in Table 6-7. The accuracy of fusion area approximation using the predicted bay length is presented later in this chapter, Section 6.3.1.7.

Table 6-7 A Summary of (C-25) Bay Length at 22.5° Predictions Generated by Different Networks Using 21 Training Patterns (mean $r^{2.5}=3.02\text{mm}$).

# of Hidden Layers	# of Nodes (1st/2nd layer) ^a	Correlation Coefficient	Absolute Difference (mm)	
			Mean	S.D.
1	4	0.825	0.22	0.20
1	5	0.745	0.24	0.25
2	4/3 ^b	0.836	0.20	0.20
2	5/3	0.804	0.21	0.22

Notes: a) Bias node is included.

b) Weights of this network are listed in Appendix XII.

6.3.1.5 Deposit (Upper-Bead) Area - C-25 Shielding

The uncertainty in the bead height prediction is also reflected in the deposit (upper-bead) area prediction as might be expected since the two are closely associated. Therefore, the complexities and uncertainties between the welding conditions and bead height or deposit area are probably common. Twenty-one, twenty-five and thirty sets of data were used for training the deposit area networks. The results are summarized in Table 6-8. A single-hidden network with four hidden nodes (including bias node) and 25 training patterns produces the most accurate bay length prediction (mean error of 12%), as shown in Figure 6.16.

Figure 6.16 Comparison of BPN (single hidden layer and 4 hidden nodes) predicted deposit area with experimental values (C-25).

Table 6-8 A Summary of (C-25) Deposit (Upper-Bead) Area Predictions Generated by Different Networks (mean A1=27.14mm²).

# of Training Patterns	# of Hidden Layers	# of Hidden Nodes ^a	Correlation Coefficient	Absolute Difference (mm ²)	
				Mean	S.D.
21	1	4	0.652	5.62	5.42
	1	5	0.532	6.46	6.52
	2	5/3	0.543	6.36	7.00
25	1	4 ^b	0.795	3.32	3.22
	1	5	0.650	4.92	5.24
	2	5/3	0.696	3.36	5.16
	2	6/4	0.713	4.22	5.00
30	1	5	0.707	3.76	3.82
	2	5/3	0.711	3.38	4.08

Notes: a) Bias node is included.

b) Weights of this network are listed in Appendix XII.

6.3.1.6 Plate Fusion (Lower-Bead) Area - C-25 Shielding

The BPN prediction of plate fusion area is similar to the BPN prediction of penetration as might be expected since these two parameters are closely associated. Twenty-one training patterns seems sufficient to provide the knowledge necessary for predicting the lower-bead area with C-25 shielding. A 2-hidden layer network with 5 and 3 hidden (including bias nodes) nodes in each hidden layer respectively provides the most reliable results (mean error of 12%). A comparison of the network predicted plate fusion areas with measured values is shown in Figure

Figure 6.17 Comparison of BPN (2-hidden layer with 5&3 hidden nodes) predicted plate fusion area with experimental values (C-25).

6.17. A summary of the results of other network structures is included in Table 6-9.

Table 6-9 A Summary of (C-25) Plate Fusion (Lower-Bead) Area Predictions Generated by Different Networks Using 21 Training Patterns (mean A2=19.70mm²).

# of Hidden Layers	# of Nodes (1st/2nd layer) ^a	Correlation Coefficient	Absolute Difference (mm ²)	
			Mean	S.D.
1	4	0.803	3.24	6.54
1	5	0.832	2.98	7.12
2	5/3 ^b	0.895	2.32	7.08

Notes: a) Bias node is included.

b) Weights of this network are listed in Appendix XII.

6.3.1.7 Pictorial Total Fusion Area Approximations Using Predicted Parameters

Finally, after all critical weld bead dimensions (BW, BH, Pene, $l^{22.5}$, A1 and A2) have been estimated by the networks, pictorial weld shape can be obtained by applying these dimensions to weld shape approximation algorithms as described in Section 6.2. Comparisons of the pictorial approximate upper- (parabola-ellipse) and lower-bead (double-parabola) areas using network predicted weld dimensions with the measured areas are included in Figures 6.18 & 6.19 respectively. In general, these shapes are more

Figure 6.18 Comparison of parabola-ellipse approximated upper-bead shape using predicted dimensions with measured areas (C-25).

Figure 6.19 Comparison of double-parabola approximated lower-bead shape using predicted dimensions with measured areas (C-25).

accurate than their conventional semi-ellipse - upper-bead and cosine - lower-bead counterparts. A summary of the results is included in Table 6-10.

For upper-bead approximate using network predicted weld dimensions (bead height and bead width), both correlation coefficient and mean of absolute difference suggest that the use of parabola-ellipse algorithm is the most appropriate (Table 6-10) among the three algorithms tested with a mean percentage error of 11%. These findings are consistent with those reported in Section 6.2.1 where the measured weld dimensions were

Figure 6.20 Experimental and predicted weld shape approximation comparison (C-25 shielding).

used. A semi-ellipse shape always over-estimates the actual deposit area and a parabola shape usually understates the actual area. On the other hand, the use of a double-parabola arrangement (mean error of 13%) to approximate the lower-bead shape using predicted weld dimensions (bead width, penetration and bay length at 22.5°) clearly is better than the use of a cosine or double-ellipse shape. Comparisons of predicted with experimental weld

Table 6-10 Comparison of Different Bead Shape Approximation Algorithms Using Predicted C-25 Weld Dimensions (BH, BW, Pene & $I^{22.5}$).

Area	Algorithm	Correlation Coefficient	Absolute Difference (mm ²)	
			Mean	S.D.
Upper-Bead (mean $A1=26.98\text{mm}^2$)	semi-ellipse	0.815	3.08	3.38
	parabola	0.815	3.08	2.45
	parabola-ellipse	0.837	2.94	2.41
Lower-Bead (mean $A2=19.70\text{mm}^2$)	cosine	0.884	3.00	2.90
	double-ellipse	0.890	2.92	3.01
	double-parabola	0.893	2.45	2.51

shapes are included in Figure 6.20 and Appendix XI.

6.4 Predicting M-2 Shielding Weld Bead Geometry

Twenty-five sets of data were selected from the original M-2 shielding weld bead geometry data base (86 sets of data) for network training purposes (as listed in Appendix VI). As in the previous C-25 shielding weld bead geometry investigation, the network input contains arc current, voltage, wire travel speed, plate thickness and a bias value (which is always one - see Chapter 3, section 3.2.6). The output parameters are the weld bead dimensions, i.e., bead width, bead height, penetration, lower-bead bay length at 22.5°, deposit area and plate fusion area. Network knowledge verification is accomplished by using the remaining 66 weld bead geometry data sets.

The schematic network structure parallels to the geometry problem with C-25 shielding (Figure 6.11). Data were normalized between 0.1 and 0.9 (Chapter 3, Section 3.3). All of the learning parameters used are similar to those used in the C-25 shielding problem, i.e., learning rate, momentum coefficient and RMS error tolerance were 0.5, 0.25 and 0.05 respectively. Step-declining learning and Fahlman's derivative were used and the initial weights were set randomly between -0.5 and 0.5. The results reported were the best of three trials (Chapter 3, Section 3.3.1).

6.4.1 Results and Discussion

A summary of correlation coefficients for all weld bead dimensions (bead width, height, penetration, bay length, deposit and plate fusion areas) predictions is listed in Table 6-11. A table of means and standard deviations of absolute difference is also included for comparing the predicted and measured values (Table 6-12). The detailed comparisons of predicted and measured values (BW, BH, Pene, $\ell^{22.5}$, A1 & A2) are also available in Figures 6.21 to 6.26. In general, 21 sets of data are sufficient for training in all cases. For predicting bead width and bead height, a single-layer network with six hidden nodes (including bias node) is the most appropriate. A seven-hidden node (including bias node)

single-hidden layer network is found to be the best structure for predicting deposit area and plate fusion area. For predicting penetration and bay length at 22.5°, a 5&3-hidden node (including bias nodes) 2-hidden layer network and a 5-hidden node (including bias node) single-hidden layer network are believed to be the best. All correlation coefficients of the results generated by the networks mentioned above are larger than 0.75 with the bead width prediction being the most accurate (correlation coefficient = 0.86). The weights of the networks described above are listed in Appendix XII.

Table 6-11 A Summary of Correlation Coefficients of M-2 Shielding Weld Bead Dimensions Predicted by Different Networks Using 21 Sets of Training Data.

# of Hidden Layers	# of Hidden Nodes	Bead Width	Bead Height	Penetration	Bay Length	Deposit Area	Plate Fusion Area
1	4	0.764	0.635	0.703	0.608	0.822	0.778
	5	0.842	0.729	0.728	0.755^a	0.671	0.812
	6	0.855^a	0.760^a	0.742	0.728	0.828	0.847
	7	0.823	0.743	0.750	0.565	0.848^a	0.847^a
2	4/3	0.694	0.643	0.749	0.421	0.644	0.826
	5/3	0.829	0.731	0.768^a	0.636	0.761	0.800

Notes: 1) High-lighted values are the most accurate weld dimension predictions.

a) The results of these entries are also plotted in Figures 6.21 to 6.26 respectively.

Table 6-12 A Summary of Means, Standard Deviations of Difference and Mean Percentage Errors of M-2 Shielding Weld Bead Dimensions Predicted by Different Networks Using 21 Sets of Training Data.

		BW (mm)		BH (mm)		Pene (mm)		f^{225} (mm)		A1 (mm ²)		A2 (mm ²)	
Mean		12.48		3.06		2.83		2.50		28.76		14.70	
# of layers	# of nodes	mean	S.D.	mean	S.D.	mean	S.D.	mean	S.D.	mean	S.D.	mean	S.D.
1	4	1.23	0.99	0.30	0.30	0.55	0.46	0.50	0.48	3.40	3.66	3.48	3.14
	5	0.95	0.83	0.29	0.27	0.50	0.46	0.37	0.33	6.22	5.48	3.12	2.92
	6	0.85	0.79	0.25	0.24	0.49	0.45	0.38	0.40	3.42	3.28	2.72	2.82
	7	0.89	0.85	0.25	0.25	0.50	0.44	0.55	0.48	3.40	3.04	2.62	2.72
2	4/3	1.51	1.23	0.30	0.32	0.52	0.44	0.63	0.62	5.42	5.48	2.74	3.06
	5/3	0.97	0.83	0.34	0.27	0.47	0.39	0.43	0.45	4.22	4.70	3.22	3.30
mean error		7%		8%		17%		15%		12%		18%	

Note: 1) Bias node is included.

Figure 6.21 Comparison of predicted and measured bead width of M-2 shielding using 6-node 1-hidden layer network.

Figure 6.22 Comparison of predicted and measured bead height of M-2 shielding using 6-node 1-hidden layer network.

Figure 6.23 Comparison of predicted and measured penetration of M-2 shielding using 5&3-node 2-hidden layer network.

Figure 6.24 Comparison of predicted and measured bay length of M-2 shielding using 5-node 1-hidden layer network.

Figure 6.25 Comparison of predicted and measured deposit area of M-2 shielding using 7-node 1-hidden layer network.

Figure 6.26 Comparison of predicted and measured plate fusion area of M-2 shielding using 7-node 1-hidden layer network.

Figure 6.27 Comparison of ellipse approximated upper-bead shape with measured values (M-2).

Figure 6.28 Comparison of double-parabola approximated lower-bead shape with measured values (M-2).

Because some of the M-2 shielding weld shapes are irregular and unsymmetrical, the use of network predicted weld dimensions for approximating the bead shape is expected to be less accurate than those for the C-25 shielding weld shape (Table 6-13 and Figures 6.27 & 28). However, the pictorial approximation is still reasonable as shown in Figure 6.29 and Appendix XI. For the upper-bead shape, the conventional semi-ellipse and suggested parabola algorithms provide comparable results (mean error of 13%). The lower-bead shape is more accurately estimated by the newly proposed double-parabola algorithms with a mean error of 26%.

Figure 6.29 Experimental and predicted weld shape approximation comparison (M-2 shielding).

Table 6-13 Comparison of Different Bead Shape Approximation Algorithms Using Predicted M-2 Weld Dimensions
(Bead Width, Bead Height, Penetration and Bay Length at 22.5°).

Area	Algorithm	Correlation Coefficient	Absolute Difference (mm ²)	
			Mean	S.D.
Upper-Bead (mean A1=28.76mm ²)	semi-ellipse parabola	0.836	7.50	6.76
		0.836	7.56	6.22
Lower-Bead (mean A2=14.71mm ²)	cosine	0.570	11.82	8.74
	double-ellipse	0.712	8.30	7.46
	double-parabola	0.714	7.56	6.92

6.5 Including A Non-Quantitative Parameter in the BPN

As mentioned in Chapter 5, Section 5.2.2.2, a non-quantitative parameter can be included in the network as input (or output) parameter. For example, the welding process is included as one of the input parameters for predicting 800 to 500°C cooling time, numerical values such as 0.1, 0.5 and 0.9 can be designated to the SAW, SMAW and GMAW processes respectively. Similarly, shielding can be included in the GMAW weld bead geometry problem by assigning a distinctive value to each gas type (C-25 or M-2). This is demonstrated in the following study.

The schematic network structure is the same as that used for analyzing the weld bead geometry problem where the shielding gas is fixed (Figure 6.11), except that there is an additional input node for entering the shielding gas. Values of 0.9 and 0.1 are assigned for C-25 and M-2 shielding respectively. The selection of the values (0.9 and 0.1) is arbitrary but the values must be as distinct as possible.

The entire weld bead shape data base (96 sets of C-25 shielding data and 86 sets of M-2 shielding data) was used in this investigation. Forty-two sets of weld shape data (21 sets of data from each shielding, Appendix VI) were selected for network training purpose and the remaining sets were used for knowledge verification. All data were normalized between 0.1 and 0.9 (Chapter 3, Section 3.3). Learning rate, momentum coefficient and

RMS error tolerance used were 0.5, 0.25 and 0.05 respectively. Fahlman's derivative, modified dynamic and step-declining learning were employed. Only the results of the best of three trials were reported.

6.5.1 Results and Discussion

The correlation coefficients, means and standard deviations of absolute difference of the network predictions are tabulated in Tables 6-14 and 15 respectively and the weights of the most accurate prediction networks are listed in Appendix XII. Although the predictions are not as accurate as those analyzed with shielding fixed, the results are comparable. The results of these tests shows that non-quantitative parameters can also be included in a BPN approach.

Table 6-14 A Summary of Correlation Coefficients of C-25 and M-2 Shielding Weld Bead Dimensions Predicted by Different Networks Using 42 Sets of Training Data.

# of Hidden Layers	# of Hidden Nodes	Bead Width	Bead Height	Penetration	Bay Length	Deposit Area	Plate Fusion Area
1	5	0.829^b	NA ^a	0.833	0.817	0.762^b	0.821
	6	0.765	0.635^b	0.814	0.797	0.711	0.846
2	5/4	0.654	0.562	0.843^b	0.777	0.744	0.841
	6/4	0.755	0.571	0.812	0.820^b	0.594	0.857^b
C-25 or M-2 only ^c		0.869	0.720	0.816	0.824	0.824	0.888

Notes: 1) The highlighted values are the most accurate weld dimension predictions.

a) NA - not applicable because learning failed to converge.

b) Weights of these entries are listed in Appendix XII.

c) Details are listed in Tables 6-4 to 6-9 and 6-11.

The bead height prediction is the least accurate. This is consistent where only C-25 shielding is considered (Section 6.6.3.1.2). However, the penetration prediction seems more accurate than those where the shielding is fixed. The reduction in accuracy in other parameters is probably due to the increase in the complexity from the number of variables involved. The accuracy is inversely proportional to the number of parameters for a

constant RMS error tolerance level. Therefore, for accuracy reasons, the problem should be kept to a minimum size if possible.

Table 6-15 A Summary of Means and S.D.s of Difference of C-25 and M-2 Shielding Weld Bead Dimensions Predicted by Different Networks Using 42 Sets of Training Data.

		BW (mm)		BH (mm)		Pene (mm)		$l^{2.5}$ (mm)		A1 (mm ²)		A2 (mm ²)	
Mean		12.39		3.08		3.05		2.78		27.90		17.34	
# of Layers	# of Nodes	mean	S.D.	mean	S.D.	mean	S.D.	mean	S.D.	mean	S.D.	mean	S.D.
1	5	0.88^b	0.80	NA ^a	NA	0.38	0.35	0.29	0.25	4.38 ^b	3.83	3.12	3.10
	6	1.06	0.99	0.36 ^b	0.34	0.42	0.39	0.30	0.29	4.58	4.42	3.14	2.96
2	5/4	1.24	1.34	0.34	0.30	0.38^b	0.34	0.31	0.29	3.78	4.30	3.94	2.62
	6/4	1.09	0.89	0.30	0.27	0.41	0.40	0.29^b	0.26	5.48	5.74	3.92^b	2.68
C-25 or M-2 only ^c		0.72	0.68	0.24	0.24	0.43	0.41	0.29	0.28	3.36	3.12	2.46	2.36

Notes: 1) The highlighted values are the most accurate weld dimension predictions.

a) NA - not applicable because learning failed to converge.

b) Weights of these entries are listed in Appendix XII.

c) Details are listed in Tables 6-4 to 6-9 and 6-12.

6.6 The Inverse Geometry Problem - Predicting Welding Parameters Given the Weld Bead Dimensions

Controlling the welding parameters to achieve a desired weld bead shape is the problem of major interest to most practitioners. However, the problem is non-trivial because there may be more than one set of welding conditions that can provide a certain weld shape. Therefore, conventional regression techniques may not be easily applied. The BPN technique described in this report may be used and the problem is termed the inverse weld bead network. Some researchers [6-7] have reported using BPN to estimate optimal welding conditions (arc current, voltage and wire travel speed) for achieving certain weld bead characteristics (bead height, bead width and penetration). This work applied specifically to the gas tungsten arc process (Chapter 1, Section 1.3). However, this work was limited to a single plate thickness (0.125 inches) and narrowly defined welding conditions. Furthermore, the author of this study believes that the use of bead width,

height and penetration may be not descriptive enough to use in practice. The bay shape was important and was not considered in the study.

Moreover, different practitioners may need to define one or another weld bead differently. For example, instead of defining a weld bead by its width, height, penetration and bay length, the interest may be in deposit or fusion areas, etc.. Therefore, the extent of this study provides more choices for estimating near welding conditions given the plate thickness (range of plate thickness is listed in Table 6-1) and weld bead characteristics. For the input to the inverse weld bead networks, the weld bead can be defined by:

- 1) bead width, bead height, penetration and bay length at 22.5° ,
 - 2) bead width, penetration and plate fusion area,
 - 3) bead width, penetration, deposit area and plate fusion area, or
 - 4) bead width, deposit area and plate fusion area,
- together with the plate thickness.

Figure 6.30 A schematic diagram of the BPN tested for solving the inverse weld bead problem.

A schematic diagram of the network used in this part of the study is shown in Figure 6.30. The input layer is defined by the weld bead characteristics and plate thickness. The hidden structure is composed of one or two hidden layers with various

hidden nodes. The output layer contains arc current, voltage and wire travel speed. Bias nodes are included in all hidden and input layers. Twenty-one out of ninety-six sets (C-25 shielding) weld bead data were used for training (Appendix VI). The normalization range for all data is between 0.1 and 0.9 (Chapter 3, Section 3.3). Learning rate, momentum coefficient and RMS error tolerance used were 0.5, 0.25 and 0.05 respectively. Fahlman's derivative, modified dynamic and step-declining learning were employed. Only the results of the best of three trials were reported.

6.6.1 Results and Discussion

A single network (instead of three separate networks as previously used) was used to estimate weld specifications (arc current, voltage and wire travel speed) for a given specific set of weld bead characteristics because it is believed that the welding parameters and weld characteristics are somewhat inter-related and the solution (welding parameters) should always be considered as a set. Therefore, solving all three parameters with one network is the more logical approach. However, in order to compensate for the complexity of the welding parameters-weld dimensions relationship, a larger network structure is required to accommodate the extra information. As the network size increases, the learning time increases exponentially because the number of weight combinations increases significantly.

6.6.1.1 Welding Conditions Estimated from BW, BH, Pene and $l^{22.5}$

It is conventional to define a weld bead by its width, height and penetration. However, with the extra lower-bead bay information ($l^{22.5}$), the weld bead description is more complete. The results of various networks tested are summarized in Table 6-16. A 6&4-hidden node 2-hidden layer network generates the most accurate overall estimation with a correlation coefficient value of 0.986. Comparisons of estimated arc current, voltage and wire travel speed with the measured values are included in Figures 6.31 to 6.33.

Table 6-16 Summary of Estimates of Welding Parameters from Various Networks Using BW, BH, Pene, $\bar{r}^{22.5}$ and h_i for C-25 Shielding.

	# of Hidden Layers (nodes)	Arc Current (amperes)	Arc Voltage (volts)	Wire Travel Speed (mm/s)	Overall Correlation Coefficient
Correlation Coefficient	1 (4)	0.808	0.838	0.713	0.981
	1 (5)	0.736	0.863	0.661	0.977
	2 (5/3)	0.830	0.895	0.728	0.985
	2 (5/4)	0.838	0.848	0.758	0.985
	2 (6/3)	0.784	0.825	0.727	0.980
	2 (6/4) ^a	0.839	0.862	0.773	0.986
	2 (6/5)	0.837	0.843	0.724	0.985
Mean	-	221.56	33.15	6.66	-
Mean (S.D.) of Absolute Difference	1 (4)	23.40 (24.97)	2.36 (3.47)	0.92 (1.46)	-
	1 (5)	26.87 (25.26)	2.22 (2.15)	1.52 (1.20)	-
	2 (5/3)	20.71 (21.44)	1.98 (1.70)	1.02 (0.81)	-
	2 (5/4)	21.77 (22.31)	2.31 (2.32)	1.09 (0.84)	-
	2 (6/3)	23.54 (27.12)	2.51 (2.38)	1.02 (0.88)	-
	2 (6/4) ^a	20.29 (20.31)	2.24 (1.97)	0.83 (0.68)	-
	2 (6/5)	24.22 (20.48)	2.61 (2.31)	1.22 (0.94)	-

Notes: 1) Highlighted values are the closest estimations.

a) The weights of this entry can be found in Appendix XII.

 Figure 6.31 Comparison of measured arc current with values estimated by BPN using BW, BH, Pene, $\bar{r}^{22.5}$ and h_i .

 Figure 6.32 Comparison of measured arc voltage with values estimated by BPN using BW, BH, Pene, $\bar{r}^{22.5}$ and h_i .

Figure 6.33 Comparison of measured wire travel speed with values estimated by BPN using BW, BH, Pene, $l^{22.5}$ and h_t .

6.6.1.2 Welding Conditions Estimated Using BW, Pene and A2

Sometimes in practice, the filler metal (deposited material) is eliminated (ground off) to provide a smooth surface. Therefore, the bead height and deposit area become less important. In this case, the weld bead characteristics - bead width, penetration and plate fusion area can be used for defining the input space to the network. They are specially chosen to fix the plate fusion parameters where there is no interest in the geometry developed above the plate surface. The results from various networks tested are listed in Table 6-17. In general, a

Figure 6.34 Comparison of measured current with values estimated by BPN using BW, Pene, A2 and h_t .

6&4 hidden node 2-hidden layer network provides the most accurate welding condition

estimates using bead width, penetration and plate fusion area. Estimated and measured welding parameters comparisons are similar to those shown in Figures 6.31 to 6.33 and are included in Figures 6.34 to 6.36.

Figure 6.35 Comparison of measured voltage with values estimated by BPN using BW, Pene, A2 and h_t .

Figure 6.36 Comparison of measured wire travel speed with values estimated by BPN using BW, Pene, A2 and h_t .

Table 6-17 Summary of Estimates of Welding Parameters from Various Networks Using BW, Pene, A2 and h_t for C-25 Shielding.

	# of Hidden Layers (nodes)	Arc Current (amperes)	Arc Voltage (volts)	Wire Travel Speed (mm/s)	Overall Correlation Coefficient
Correlation Coefficient	1 (5)	0.832	0.830	0.748	0.986
	1 (6)	0.795	0.850	0.689	0.979
	1 (7)	0.795	0.845	0.631	0.980
	2 (5/4)	0.757	0.780	0.669	0.977
	2 (6/4) ^a	0.876	0.860	0.759	0.989
	2 (6/5)	0.817	0.833	0.701	0.981
Mean	-	221.56	33.15	6.66	-
Mean (S.D.) of Absolute Difference	1 (7)	22.97 (22.12)	2.68 (2.26)	1.02 (0.77)	-
	1 (6)	24.90 (26.03)	2.27 (2.32)	1.24 (1.09)	-
	1 (7)	24.64 (25.82)	2.38 (2.18)	1.42 (1.28)	-
	2 (5/4)	26.91 (25.82)	2.99 (2.59)	1.17 (0.91)	-
	2 (6/4) ^a	18.67 (17.05)	2.03 (2.21)	0.89 (0.73)	-
	2 (6/5)	23.96 (23.25)	2.32 (2.28)	1.03 (0.87)	-

Notes: 1) Highlighted values are the closest estimations.

a) The weights of this entry can be found in Appendix XII.

6.6.1.3 Welding Conditions Estimates Using BW, Pene, A1 and A2

In some cases deposit area together with bead width, penetration, plate fusion area and plate thickness are convenient in practice to describe a desired weld bead. The input space to the network is similar to the one mentioned in the previous section with the addition of the deposit area constraint. The results from different networks are listed in Table 6-18. A 6&4-hidden node 2-hidden layer network is recommended. The weights of this network are listed in Appendix XII. Comparisons of the network predicted and measured values are included in Figures 6.37 to 6.39.

Figure 6.37 Comparison of measured current with values estimated by BPN using BW, Pene, A1, A2 and h_r .

Figure 6.38 Comparison of measured voltage with values estimated by BPN using BW, Pene, A1, A2 and h_r .

Figure 6.39 Comparison of measured wire travel speed with values estimated by BPN using BW, Pene, A1, A2 and h_r .

Table 6-18 Summary of Estimates of Welding Parameters from Various Networks Using BW, Pene, A1, A2 and h_1 for C-25 Shielding.

	# of Hidden Layers (nodes)	Arc Current (amperes)	Arc Voltage (volts)	Wire Travel Speed (mm/s)	Overall Correlation Coefficient
Correlation Coefficient	1 (4)	0.758	0.832	0.668	0.976
	1 (5)	0.796	0.752	0.672	0.981
	1 (6)	0.777	0.676	0.602	0.976
	2 (5/4)	0.810	0.791	0.666	0.982
	2 (6/3)	0.718	0.775	0.562	0.968
	2 (6/4) ^a	0.848	0.854	0.750	0.986
	2 (6/5)	0.789	0.830	0.658	0.979
Mean	-	221.56	33.15	6.66	-
Mean (S.D.) of Absolute Difference	1 (4)	27.59 (26.19)	2.51 (2.34)	1.15 (0.98)	-
	1 (5)	25.89 (24.15)	3.08 (2.69)	1.25 (1.02)	-
	1 (6)	27.88 (28.54)	3.55 (3.45)	1.47 (1.36)	-
	2 (5/4)	23.52 (34.58)	2.76 (2.63)	1.17 (1.06)	-
	2 (6/3)	32.29 (30.43)	3.04 (2.82)	1.54 (1.35)	-
	2 (6/4) ^a	22.46 (22.78)	2.37 (2.15)	1.09 (0.92)	-
	2 (6/5)	27.11 (26.71)	2.54 (2.46)	1.28 (1.19)	-

Notes: 1) Highlighted values are the closest estimations.

a) The weights of this entry can be found in Appendix XII.

6.6.1.4 Welding Conditions Estimates Using BW, A1 and A2

Using bead width, deposit area and plate fusion area to define the weld bead is believed to be the most simplified form since there is no constraint regarding the height or penetration. The results are tabulated in Table 6-19. A 6-hidden node single-hidden layer network is found to be the most appropriate in this case and the weights are listed in Appendix XII. The comparisons of estimated and measured welding parameters are included in Figures 6.40 to 6.42.

Figure 6.40 Comparison of measured current with values estimated by BPN using BW, A1, A2 and h_1 .

Figure 6.41 Comparison of measured voltage with values estimated by BPN using BW, A1, A2 and h_t .

Figure 6.42 Comparison of measured wire travel speed with values estimated by BPN using BW, A1, A2 and h_t .

Table 6-19 Summary of Estimates of Welding Parameters from Various Networks Using BW, A1, A2 and h_t for C-25 Shielding.

	# of Hidden Layers (nodes)	Arc Current (amperes)	Arc Voltage (volts)	Wire Travel Speed (mm/s)	Overall Correlation Coefficient
Correlation Coefficient	1 (5)	0.688	0.695	0.611	0.972
	1 (6) ^a	0.766	0.855	0.705	0.978
	2 (5/4)	0.716	0.780	0.599	0.973
	2 (6/4)	0.752	0.849	0.689	0.977
	2 (7/5)	0.705	0.763	0.631	0.970
Mean	-	221.56	33.15	6.66	-
Mean (S.D.) of Absolute Difference	1 (5)	29.30 (28.75)	3.33 (3.07)	1.29 (0.99)	-
	1 (6) ^a	24.31 (27.09)	2.31 (2.17)	1.03 (1.02)	-
	2 (5/4)	27.01 (29.56)	2.45 (2.88)	1.16 (1.01)	-
	2 (6/4)	25.62 (27.25)	2.36 (2.29)	1.03 (0.95)	-
	2 (7/5)	29.84 (29.68)	2.96 (2.98)	1.20 (1.13)	-

Notes: 1) Highlighted values are the closest estimations.

a) The weights of this entry can be found in Appendix XII.

6.6.1.5 Comparing the Four Different Weld Bead Criteria

It is to be expected that the use of weld bead defined by its width, deposit area, plate fusion area and plate thickness to estimate the appropriate welding parameters would yield the least accurate results because the definitions contain the least information and constraints. On the other hand, the use of traditional weld bead definitions (bead width, bead height, penetration, bay length at 22.5° and plate thickness) offers the best results since the definitions are the most complete and descriptive. The use of bead width, penetration, deposit area, plate fusion area and plate thickness to estimate the welding parameters generates comparable results because deposit area seems somewhat coupled to bead height (Section 6.2) and plate fusion area can be closely described by penetration and bay length at 22.5° (Section 6.2).

The most accurate welding condition estimates are provided by using only bead width, penetration and plate fusion area when the upper-bead is not included. Eliminating the upper-bead for the flat plate (ground bead) situation gives the network more flexibility in selecting the most appropriate welding conditions. Furthermore, as mentioned in Sections 6.3.1.2 and 6.3.1.5, the experimental upper-bead information is somewhat uncertain. By neglecting the upper-bead information, some of the uncertainties of the upper-bead information can be avoided and hence a better result can be achieved.

The use of inverse weld bead networks provides near approximations to the welding conditions necessary for generating desired weld shapes as shown in Figures 6.43 to 6.45. Table 6-20 shows network predicted weld bead dimensions using the estimations of welding conditions when compared to the desired weld bead

Figure 6.43 Comparison of the desired and expected bead width using the results of the BW, Pene & A2 inverse network.

dimensions. In other words, the desired weld bead dimensions are submitted to the inverse networks, the results (welding conditions) of the inverse network are then used to predict weld bead dimensions and the final output (networks predicted weld bead dimensions) are in turn compared with the original desired dimensions. Figures 6.43 to 6.45 compare the desired and expected weld dimensions (predicted from the welding parameters of the BW, Pene & A2 inverse network). The networks used for weld bead geometry and inverse weld bead problems are those described in Sections 6.3.1.1 to 6.3.1.6 and Sections 6.6.1.1 to 6.6.1.4 respectively. Some sample estimated and desired weld bead dimensions are included in Appendix XIII.

Figure 6.44 Comparison of the desired and expected penetration using the results of the BW, Pene & A2 inverse network.

Figure 6.45 Comparison of the desired and expected plate fusion area using the results of the BW, Pene & A2 inverse network.

All the inverse weld bead networks presented in Table 6-20 provide close weld bead dimension predictions from of welding conditions. The average performance of inverse network 'B' (estimated by using BW, Pene, A2 and h_t) is the best among the results of the four networks examined with a correlation coefficient of every weld dimension over 0.70. The weld bead dimension predictions using welding conditions from the traditional weld bead definitions (BW, BH, Pene, $t^{2.5}$ & h_t) offer excellent accuracy for

bead width, penetration and lower-bead bay length at 22.5°. However, the results of these same networks suggest less accurate bead height and deposit area predictions. The difficulties can be traced back to the original experimental weld data as described in Section 6.3.1.2.

Table 6-20 The Summary of Weld Bead Dimensions Predicted by Using the Estimated Welding Conditions for C-25 Shielding.

Inverse Network	Bead Width (mm)	Bead Height (mm)	Penetration (mm)	Bay Length (mm)	Deposit Area (mm ²)	Plate Fusion Area (mm ²)	Overall Corr. Coeff.
A	0.858	0.671	0.857	0.841	0.677	0.913	0.958
B	0.892	0.717	0.807	0.848	0.728	0.886	0.966
C	0.839	0.694	0.796	0.524	0.722	0.868	0.964
D	0.806	0.783	0.805	0.656	0.832	0.871	0.969

Notes: 1) The weld bead geometry problem networks are described in Sections 6.3.1.1 to 6.3.1.6.

A) The inverse network input is BW, BH, Pene, $\bar{r}^{22.5}$ and h_i (Section 6.6.1.1).

B) The inverse network input is BW, Pene, A2 and h_i (Section 6.6.1.2).

C) The inverse network input is BW, Pene, A1, A2 and h_i (Section 6.6.1.3).

D) The inverse network input is BW, A1, A2 and h_i (Section 6.6.1.4).

6.7 Effect of Plate Thickness on Weld Dimensions

The weld bead geometry of GMAW is closely related to the arc heat input, shielding gas, electrode and joint geometry since all these parameters affect the heat flow during welding. The effect of heat input can be viewed in terms of the parameters: current, voltage and wire travel speed because each parameter has an effect on the weld bead geometry. Furthermore, the electrode effect is usually characterized by its polarity, diameter and extension (distance between the workpiece and the electrode nozzle). In general, as the heat input increases so do the weld dimensions. Therefore, weld dimensions increase with increasing arc current and voltage or decreasing wire travel speed and electrode extension. Negative polarity and smaller electrode diameter usually result in

deeper penetration. Shielding gas influences the workpiece surrounding environment and hence the heat transfer. Joint geometry directly affects the heat flow within the workpiece.

The effects of the welding parameters discussed above have been studied and reported by many researchers [6-8,9,10,11], except for the effect of the workpiece thickness, since most investigation have been assumed thick plate situations (described in Chapter 1, Section 1.1.1.2). This part of the study is intended to provide a preliminary investigation regarding the effect of plate thickness on weld bead geometry. The weld bead geometry prediction models developed previously in this Chapter (Sections 6.3 & 6.4 and the weights of the networks are listed in Appendix XII) are used in this analysis.

6.7.1 Results and Discussion

In general, the effect of plate thickness is not as great as those of arc current, voltage or wire travel speed (Table 6-21). By doubling the plate thickness, the bead width only increases by some 5%. Moreover, from the same Table, where input energy is held constant, various welding parameters affect the bead width for C-25 shielding differently. The greatest effect on bead width is arc current while wire travel speed has the least effect.

Table 6-21 Comparison of the Effects of Welding Parameters on C-25 Bead Width.

Current I_c (amperes)	Voltage V_o (volts)	Speed S_p (mm/s)	Input Energy Q (kJ)	Thickness h_i (mm)	Bead Width BW (mm)	% Difference (%)
160	25	5	0.80	6.5	9.06	83
320	25	5	1.60	6.5	16.61	
200	20	5	0.80	6.5	10.37	44
200	40	5	1.60	6.5	14.91	
320	25	10	0.80	6.5	15.34	8.3
320	25	5	1.60	6.5	16.61	
160	25	5	0.80	6.5	9.06	4.3
160	25	5	0.80	13	9.45	

The following observations are based on a specific set of intermediate welding conditions ($I_c=200$ amperes, $V_o=25$ volts, $S_p=5$ mm/s and with C-25 shielding). This covers the three different heat flow situations (2-D, 2.5-D & 3-D as described in Section

1.1.1.2 and Equations 1-12) evenly when the plate thickness increases from 6.5 mm to 15.5 mm. For GMAW, a 70% process efficiency (η_{proc}) [6-12] is used for calculating the relative thickness (H_r). In this case, plate thicknesses less than 7.5 mm can be considered to be 2-D, while plate thicknesses greater than 13.6 mm are 3-D and plate thicknesses between 7.5 mm and 13.6 mm are said to be 2.5-D. The predicted weld dimensions are listed in Appendix XIII. The effect of the plate thickness on the bead width, bead height, penetration, bay length, deposit area and fusion area are shown in Figures 6.46 & 6.47.

From Figure 6.46, bead width seems to increase with the plate thickness. However, the plate thickness does not have any significant effect on the bead height, penetration or bay length although some variation is observed in the 2.5-D region. The most significant effect caused by plate thickness seems to be on the

Figure 6.46 The effect of plate thickness on bead width, height, penetration and bay length for $I_c=200$ amperes, $V_o=25$ volts, $S_p=5$ mm/s and with C-25 shielding.

Figure 6.47 The effect of plate thickness on deposit area and plate fusion area for $I_c=200$ amperes, $V_o=25$ volts, $S_p=5$ mm/s and with C-25 shielding.

deposit area (Figure 6.47). In the 2-D situation, the deposit area decreases significantly with increasing plate thickness but its effect seems to taper off in the 2.5- and 3-D regions. Yet, plate thickness seems to have a slightly positive effect on the plate fusion area.

6.8 Chapter Summary

For estimating the upper-bead (deposit area) shape of C-25 shielding, if the deposit area is greater than 40 mm^2 , a semi-ellipse (determined by bead width and height) is the most appropriate choice. Otherwise a parabola is the better choice (deposit area is greater than 40 mm^2). However, for the pictorial representation of the upper-bead shape of M-2 shielding, a semi-ellipse usually provides a closer approximation. On the other hand, the pictorial approximation of the lower-bead (plate fusion area) of both C-25 and M-2 shielding can be accurately represented by two parabolas, one rotated 90° (defined by bead width and penetration). Furthermore, the intersection of the two parabolas is located by the bay angle (22.5° from the horizontal plane) and length (from the center to the weld edge at this angle).

For solving the weld bead geometry problem where C-25 is the shielding, six networks are required and each one of them is responsible for one and only one of the weld dimensions. The network structure for each weld dimension is listed in Table 6-22 and weights are included in Appendix XII. Most of the network predictions provide reasonable accuracy except for the upper-bead parameters (bead height and deposit area). Similarly, the weld bead geometry problem for M-2 shielding requires six individual networks which are also listed in Table 6-22 and Appendix XII. It has also been shown that the M-2 and C-25 shielding weld bead geometry problems can be combined and solved with the BPN technique using the shielding gas as one of the input parameters. However, prediction accuracy is sacrificed by so doing.

Estimating the welding conditions given the weld bead characteristics can also be achieved with BPN method. Four networks are proposed for estimating the near welding conditions given different input weld dimension requirements (Table 6-23 and Appendix

XIII). The best accuracy is found by using bead width, penetration, plate fusion area and plate thickness. Because welding conditions to weld bead shape relationship is non-trivial, i.e., more than one set of welding conditions is able to provide the desired weld shape, the set of welding parameters are treated as a unique solution and hence all inverse networks contain three output parameters (arc current, voltage and wire travel speed).

**Table 6-22 Networks Used for Predicting Weld Bead Dimensions
with C-25 and M-2 Shielding.**

Shielding	Weld Dimension	Network Structure		Mean Error (%)
		# of Layers	# of Nodes	
C-25	Bead Width	1	5	5
	Bead Height	1	5	7
	Penetration	2	8/6	12
	Bay Length at 22.5°	2	4/3	7
	Deposit Area	1	4	12
	Plate Fusion Area	2	5/3	12
M-2	Bead Width	1	6	7
	Bead Height	1	6	8
	Penetration	2	5/3	17
	Bay Length at 22.5°	1	5	15
	Deposit Area	1	7	12
	Plate Fusion Area	1	7	18

**Table 6-23 Networks Used for Estimating Welding Conditions
from Weld Bead Dimensions for C-25 Shielding.**

Network Input	# of Hidden Layers	# of Hidden Nodes
BW, BH, Pene, $t^{22.5}$ & h_t	2	6/4
BW, Pene, A2 & h_t	2	6/4
BW, Pene, A1, A2 & h_t	2	6/4
BW, A1, A2 & h_t	1	6

Finally, plate thickness is found to have minimal effect on the bead height, penetration, bay length at 22.5° and plate fusion area. Bead width increases and deposit area decreases with plate thickness for thin plate 2-D/2.5-D situations.

Chapter Seven

General Discussion

In general, the use of the BPN technique to predict HAZ hardness and weld bead geometry provides significant advantages over the use of conventional regression analysis. The major advantage is ability of the BPN method to ferret out a relationship from the given training patterns, e.g., the characteristic hardness curves in Chapter 5, Section 5.3.2. If regression technology is used, that relationship must be ascertained by the user and may bias the entire analysis. BPN also eliminates the need to assign some physical constants required when a physical solution is used, e.g., cooling time prediction in Chapter 5, Section 5.2.2.

7.1 Comparing BPN Predictions with Previous Works

The use of BPN method to predict $t_{8/5}$ has demonstrated significant advantages over the Adams' cooling time models [7-1]. In most cases, the network predicted $t_{8/5}$ is more accurate. Although the network cooling time knowledge is acquired from SAW data only, it can be applied to other welding process by assigning an appropriate process efficiency. However, it seems that the process efficiency depends not only on the type of welding process but also on workmanship and equipment setup. Therefore, assigning a nominal efficiency for different welding process may give rise to a certain amount of inaccuracy. However it may be necessary to use an efficiency when experimental cooling time data is limited or when the cost and time to produce such information are also involved into consideration.

The HAZ hardness network is found to be more accurate than several recent regression HAZ hardness prediction models [7-2,3]. Moreover, the network knowledge developed in this work covers a larger range of low alloy steels than previously considered. Specifically, existing network hardness knowledge can also be fine tuned by

injecting new hardness data other than the original data (Yurioka in this case) into the network training data set. Finally, the HAZ hardness can be estimated directly from welding conditions and steel chemical composition by combining the cooling time network and HAZ hardness network. The results are more accurate and generally better based in principle than those generated from using Adams' cooling time models and regression hardness models. For example, BPN technique eliminates the need to pre-define the cooling time-plate thickness relationship, and eliminates the need to assign physical thermal parameters which are not well known.

Similarly the use of BPN technique to predict weld bead geometry instead of the conventional regression formulation eliminates the need for pre-defining the weld dimensions-welding parameters relationships, e.g., the logarithmic function assumed by Chandel et al [7-4,5]. There are two other publications in the literature, Andersen et al [7-6] and Jones et al [7-7], where BPN for predicting weld size is used. However, they are based on GTAW and FCAW. These two welding processes are not as commonly used as GMAW used in this study. Furthermore, there are few details regarding the development of the BPN technology for this purpose. In fact, the work of Jones et al is currently commercial and specifically denies the welding community the algorithms and methodology.

7.2 BPN Prediction and Fundamental Analysis

The main disadvantage of the BPN technique is the absence of reasoning. Unlike a fundamental physical solution [7-8,9], the weld mechanics are not included in an ANN solution and hence explanations of predicted phenomena are almost impossible. Therefore, BPN can be used as a regression analysis alternative for predicting HAZ hardness and weld bead geometry but it does not replace a fundamental physical solution of the weld mechanics of such a problem. It is always going to be necessary to understand the fundamentals of such situations in the end.

Furthermore, there is no control of the network predicted input-output relationship which is solely dependent on the training pattern provided. As a result, the network generated relationship can be very unpredictable. For the same reason, the network generated relationship may perform poorly beyond the training pattern limits.

7.3 Effects of the Network Structure and Training Patterns

As shown in Sections 5.2.2, 5.3.2, 6.3.1 and 6.4.1, the effect of the training patterns is more significant than the network hidden structure. A good set of training patterns ensures the network has been provided with the necessary knowledge to solve the problem. The hidden structure can be viewed as a way of fine tuning the input-output relationship once the knowledge size has been determined.

Despite different degrees of uncertainty presented in the experimental data used in this study, all BFNs were capable of predicting the required weld features with acceptable accuracy. Although the BPN technique is designed to handle noisy data, uncertainties in the training patterns should be minimized if possible because they affect both the learning time and the accuracy of the knowledge. In general, the HAZ hardness and cooling time predictions (with an average correlation coefficient about 0.90) are more accurate than weld bead geometry prediction (with an average correlation coefficient of 0.85). This is because the hardness and cooling time values used are the mean values of several measurements [7-10,11,12], i.e., variations in the experimental values due to uncertainty are reduced. On the other hand, the weld bead data used are the measurements of single runs and a greater uncertainty is to be expected.

Even though BPN can handle some uncertainty, it has limited tolerance for conflicts within the training data. It is suspected that this might be the cause for rather poor C-25 shielding bead height predictions (Chapter 6, Section 6.3.1.2). At least in one case from the C-25 shielding data sets 1 & 5 (Appendix VI), the measured data suggests that bead height is reduced as the voltage increases. However, from the C-25 shielding data sets 9 & 13, the measurements suggest the opposite phenomena, i.e., bead height

increases with voltage. Although the learning converged, predictions were not all that satisfactory (Table 6-5). In order to solve this, two training patterns (#5 & 72) were removed from the training set by this researcher after examining the entire data base for apparent conflicting measurements.

As mentioned in Chapter 3, Section 3.3.3, the least complex hidden structure that provides reasonable prediction is always preferred, since a smaller hidden structure is always a sub-set of a larger one. As the combination of weights for satisfying the training criteria increases with a larger hidden structure, the probability of finding an optimal set of weights may be reduced and hence the actual time required to acquire the correct knowledge may increase even though the individual learning time may seem shorter, i.e., the chance of successful learning is larger for a larger network, and, the knowledge may not be as accurate as that of a simpler one. Furthermore, as stated in Sections 5.2.2 (Table 5-3), 5.3.2 (Table 5-9) and 6.4.1 (Table 6-11), the hidden structure usually has minimal effect on the final network prediction accuracy.

Finally, the accuracy of network prediction is also affected by the normalization range. By normalizing the input and output data between 0.1 and 0.9, some information may have lost because of the shrinkage of the problem domain. However, the practice could be justified by improving the training time. At the same time, the work should be constrained to this range, i.e., extrapolation should be discouraged.

Chapter Eight

General Conclusions

- 1) The backpropagation network technique is found to be an effective alternative to conventional regression analysis for modeling HAZ hardness and weld bead geometry problems.
- 2) The heat-affected zone hardness of low-alloyed steel can be accurately predicted using the backpropagation network technology given the carbon content, carbon equivalent (IIW), silicon content and 800 to 500 °C cooling time. Furthermore, the 800 to 500 °C cooling time can also be determined by the backpropagation network method given the arc current, voltage, wire travel speed, plate thickness and process efficiency. In brief, HAZ hardness can be estimated by the combining cooling time and HAZ hardness networks, given welding conditions and steel chemical composition.
- 3) Networks have been successfully trained for predicting gas metal arc weld bead dimensions in terms of bead height, bead width, penetration, lower-bead bay length at 22.5°, deposit area and plate fusion area given arc current, voltage, wire travel speed and plate thickness.
- 4) A double-parabola algorithm is proposed for representing the lower-bead shape pictorially, defined by the bead width, penetration and bay length at 22.5°. The upper-bead shape can be approximated by the conventional semi-ellipse for M-2 and for C-25 shielding with deposit area greater than 40 mm². However, for C-25 shielding, a parabola is more appropriate if the deposit area is less than 40 mm².
- 5) The inverse weld bead geometry problem, i.e., estimating the welding conditions given a specific weld shape, can also be solved by the backpropagation network method. It is found that using bead width, penetration and plate fusion area provides the best welding conditions estimates.

Chapter Nine

Future Works

From the results of this study, it seems that BPN is a useful technology for the welding industry and for that matter for the fabrication and manufacturing industry in general. Numerous application can be imagined just by considering the use of BPN as an alternative to regression analysis and at least in principle it goes well beyond that.

The cooling time prediction knowledge base should be extended to include other welding processes as suggested in Section 5.2.2.2. Instead of using an efficiency factor, the input should be the welding parameters directly. Similarly, instead of using a carbon equivalent to predict HAZ hardness of steel, alloy content should be used as direct input to the network, at least in principle, since carbon equivalents are the products of prior regression analyses. Moreover, predicting HAZ hardness directly from welding conditions (arc current, voltage, wire travel speed, plate temperature, plate geometry and welding process) and steel chemical composition (C, Si, Mn, Cu, Cr Ni, Mo, V & B) would be an ideal approach with the cooling time prediction eliminated since it is only an intermediate step in any case. Finally, an inverse HAZ hardness network would also be interesting, i.e., predicting the arc energy input given plate temperature, plate geometry, welding process and steel chemical composition to achieve a given hardness level.

The GMAW bead geometry problem could be expanded to cover the effect of electrode extension, electrode diameter and electrode polarity which are believed to have significant effects. In addition, including the workpiece edge preparation and preheat temperature to the bead geometry problem would provide a more practical knowledge base. Finally, the most ambitious improvement for the weld bead geometry problem would be to analyze the effect of steel chemical composition using BPN.

References

References for Chapter 1

- 1-1) Shigley J. E., Mitchell L. O., Mechanical Engineering Design, 4th edition, McGraw-Hill Book Company, 1983, pp 175-176.
- 1-2) Yurioka N., Impact of Welding Research on Steel Composition Development, Materials and Design, Vol. VI, No. 4, 1985, 19 pgs.
- 1-3) Jackson C. E., The Science of Arc Welding, Welding Journal, Part 1, Vol. 39, No. 4, 1960, pp 129s, Part 2, Vol. 39, No. 5, pp 177s, Part 3, Vol. 39, No. 6, pp 225s.
- 1-4) Naker B. H., et al, Mild Steel Welding with Argon/CO₂ Mixtures, British Welding Journal, Vol. 8, No. 4, 1961, pp 151-161.
- 1-5) Shultz B. L., Jackson C. E., Influence of Weld Bead Area on Weld Metal Mechanical Properties, Welding Journal, January 1973, pp 26s.
- 1-6) Ito Y., Bessyo K., Weldability Formula of High-Strength Steels Related to Heat-Affected-Zone Cracking, IIW Doc. IX-576-68, 1968, 45 pgs., Transaction of Japanese Welding Society, 1968, Vol. 37, No. 9, pp 983-991, IIW Doc. IX-631-69, 1969, 18 pgs.
- 1-7) The IIW Formula for Carbon Equivalent, Technical Report, 1967, IIW Doc IX-535-67.
- 1-8) Suzuki H., Carbon Equivalent and Maximum Hardness, Transaction of Japanese Welding Society, 1984, Vol. 15, No. 1, pp 25-33.
- 1-9) Terasaki T., Nomura T., Kitada T., Proposal of Predictive Equation for Maximum Hardness in the Heat-Affected-Zone, Japan Welding Society Journal Quarterly, 1988, Vol. 6, No. 1, pp 139-143.
- 1-10) Satoh K., Terasaki T., Estimation of HAZ Hardness, Japanese Welding Society General Meeting, Vol. 25, 1979, pp 98.
- 1-11) Terasaki T., Study of Predictive Equations of Thermal Factor and Hardness Related with Weld Cold Cracking, Journal of Iron and Steel Institute of Japan, Vol. 16, 1981, pp 2715-2723.

- 1-12) Suzuki H., Root Cracking and Maximum Hardness in High-Strength Steel Welds, Trans. Jap. Weld. Soc., Vol. 15, No. 1, 1984, pp 34-42, IIW Doc. IX-1280-83.
- 1-13) Suzuki H., A New Formula for Estimating HAZ Hardness in Welded Steels, IIW DOC IX-1351-85, 1985, 15 pgs.
- 1-14) Arata Y., Nishiguchi K., Ohij T., Kohsai N., Weldability Concept on Hardness Prediction, Transaction of Journal of Japanese Welding Research Institute, Vol. 8, No. 1, 1979, pp 43-52, IIW Doc. IV-263-72.
- 1-15) Yurioka N., Ohshita S., Tamehiro H., Study on Carbon Equivalent to Assess Cold Cracking Tendency and Hardness in Steel Welding, Proc. Conf. on Pipeline Welding in the 80's, 1981, 15 pgs., JWS WM-784-80, 11 pgs, 1980.
- 1-16) Yurioka N., Okumura M., Kasuya T., Prediction of HAZ Hardness of Transformable Steel, Metal Construction, Vol. 19, No. 4, 1987, pp 217R-223R.
- 1-17) Beckert M., Holz R., Use of Carbon Equivalent and Prediction of the Weld HAZ Hardness, Schweiss Technik, Vol. 23, No. 8, 1973, pp 344-346.
- 1-18) Dueren C., Equation for the Prediction of Cold Cracking Resistance in Field Welding Large Diameter Pipes, IIW Doc. IX-1358-85, 1985, 15 pgs.
- 1-19) Cottrell C. L. M., Hardness Equivalent May Lead to a More Critical Measure of Weldability, Metal Construction, Vol. 16, No. 12, 1984, pp 740-744.
- 1-20) Goldak J., Chakravarti A., Bibby M., A New Finite Element Model for Welding Heat Sources, Metallurgical Transaction, Vol. 15B, No. 2, 1984, pp 299-305.
- 1-21) Adams C. M. Jr., Cooling Rates and Peak Temperatures in Fusion Welding, AWS Welding Journal, Vol. 37, No. 5, 1958, pp 210.
- 1-22) Jhaveri P., Moffat W. G. Adams C. M. Jr., The Effect of Plate Thickness and Radiation on Heat Flow in Welding and Cutting, Welding Journal, Vol. 41, No. 1, 1962, pp 12.
- 1-23) Satoh K., Matsui S., Determination of Preheating Condition to Avoid Weld Cracking in Steel Constructions, Journal of JWES, Vol. 40, No. 11, 1971, pp 1117.
- 1-24) Rosenthal D., Mathematical Theory of heat Distribution During Welding and Cutting, AWS Welding Journal, Vol. 21 No. 5, May 1941, pp 220s-234s.

- 1-25) Rosenthal D., The Theory of Moving Sources of Heat and Its Application to Metal Treatments, Transaction of ASME, Vol. 43, No. 11, 1946, pp 849-866.
- 1-26) Christensen N., Davies V. de L., Gjermundsen K., Distribution of Temperature in Arc Welding, British Welding Journal, Vol. 12, No. 2, 1965, pp 54-75.
- 1-27) Chandel R. S., Mathematical Modelling of GMAW Features, Modelling and Control of Casting and Welding Process IV, 1988, pp 109-120.
- 1-28) Chandel R. S., An Investigation into the Electrode and Plate Melting Efficiencies of SAW and GMAW Processes, Report MTL 89-10(J), Metals Technology Laboratories, CANMET/EMR, Government of Canada, Ottawa. Ontario, K1A 0G1, 1989.
- 1-29) Chan B., Software for Welding Engineers, Master Thesis. Department of Mechanical and Aerospace Engineering, Carleton University, December, 1990.
- 1-30) Chan B., Chandel R., Bibby M., A Software System for Computing the Size and Shape of GMA (Gas Metal Arc) Welds, ASME Journal of Engineering for Industry, presented at ASME, Production Engineering Division, 1991 Winter Annual Meeting, Atlanta Hilton and Towers, December 1-6, 1991. 'Welding and Joining Processes', ASME PED Vol. 51, Welding and Joining Processes, 1991, pp 215-225.
- 1-31) Chan B., Bibby M., Yang L. J., Chandel R., A Discussion of Algorithms of Predicting Submerged Arc Weld Shapes with Workpiece Edge Preparation, Manufacturing Science and Engineering, PED-vol. 64, presented at the 114th annual ASME Annual Winter Meeting in New Orleans, Nov. 29 - Dec. 3, 1993.
- 1-32) Chan B., Bibby M., Chandel R., A Software System for Anticipating the Size and Shape of Submerged Arc Welds, Journal of Materials Processing Technology, Vol. 40, No. 3-4, pp 249-262, Jan. 1994 (also presented at the Asia Pacific Conference on Material Processing, Singapore, March, 1993).
- 1-33) Chandel R. S., Bala S. R., Relationship between Submerged Arc Welding Parameters and Weld Bead Size, Welding and Cutting Journal. Vol. 2, 1988, pp E28-30.
- 1-34) Chandel R. S., Mathematical of Melting Rates for Submerged Arc Welding, Welding Journal, May 1987, pp 135s-140s.
- 1-35) Chandel R. S., Melting Efficiency of SAW and Its Prediction, Australian Welding Journal, 1st Quarter 1989, pp 23-25.

- 1-36) Yang L. J., Bibby M. J., Chandel R. S., Modelling the Submerged Arc Weld Process Using Linear Regression Equations, Proceeding of the 3rd Conference on Trends in Welding Research, Gatlinburg, Tennessee, June 1992.
- 1-37) Yang L. J., A Study on the Submerged Arc Welding Process Variables, Internal Report, School of Mechanical & Production Engineering, Nanyang Technological Institute, Singapore, April 30, 1990.
- 1-38) McGlone J. C., Chadwick B. D., The Submerged Arc Butt Welding of Mild Steel, Part I: The Influence of Procedure Parameters on Weld Bead Geometry, The Welding Institute Report No 80/1978/PR, Dec 1978, 26 pgs.
- 1-39) Quintino L., Allun C. J., Pulsed GMAW : Interaction between Process Parameters Part 2, Welding and Metal Fabrication, Vol. 52, No. 3, 1984, pp 126-129.
- 1-40) Raveendra J., Parmar R. S., Mathematical Models to Predict Weld Bead Geometry of Flux Cored Arc Welding Metal Construction, Vol. 19, No. 1, 1987, pp 31R.
- 1-41) Chan B., Bibby M. J., Goldak J. A., A Software System for Determining the Preheat Temperature Required to Avoid Underbead Cracking - I, Canadian Metallurgical Quarterly, Vol. 31, No. 3, 1992, pp 195-202.
- 1-42) Chan B., Bibby M. J., Goldak J. A., A Software System for Computing the HAZ Hardness after Post Weld Heat Treatment, Canadian Metallurgical Quarterly, Vol.31, No. 3, 1992, pp 203-209.
- 1-43) Chan B., Bibby M. J., A Software Environment for Predicting Weld Features, AWS International Conference on Computerization for Welding Information IV, Orlando, Florida, USA, 1992.
- 1-44) Okumura M., Yurioka N., Kasuya T., Cotton H. J. V., Prediction of HAZ Hardness after PWHT, Proceedings of IIW International Conference on Residual Stress and Stress Relieving, Sofia, Bulgaria, 1987, pp 61-68.
- 1-45) Yurioka N., Suzuki H., Ohshita S., Saito S., Determination of Necessary Preheat Temperature in Steel Welding, Welding Journal, Vol. 62, No. 6, 1983, pp 147s.
- 1-46) Yurioka N., Okumura M., Ohshita S., Saito S., On the Methods Determining Preheat Temperature Necessary to Avoid Cold Cracking in Steel Welding, IIW DOC XII-E-10-81, 1981.

- 1-47) Suzuki H., Yurioka N., Weldability of Line-Pipe Steels and Prevention of Cracking in Field Welding, IIW DOC IX-1458-87, 1987.
- 1-48) Mehrotra V., Bibby M., Goldak J., Moore J., A Microcomputer Method for Predicting Preheating Temperature, published in "Welding for Challenging Environments", edited by the Welding Institute of Canada, Pergamon Press, 1986, pp 233-244.
- 1-49) Chan B., Weldsoft - a Software System for Busy Welding Engineers: User's Guide, distributed by American Welding Society and Pipe-Protech Corrosion Survey Inc. 174 Colonnade Rd. S., Unit 21, Nepean, Ontario K2E 7J5, 1992.
- 1-50) Charniak E., McDermott D., Introduction to Artificial Intelligence, Addison-Wesley Publishing Company Inc., 1985.
- 1-51) Andersen K., Cook G. E., Ramaswamy K., Karsai G., Artificial Neural Networks Applied to Arc Welding Process Modelling and Control, IEEE Industry Applications -Conference Record of the 1992 IEEE, Part 2, 1992, pp 2327-2331.
- 1-52) Madden S., Rock A., Xu X., Jones J., Neural Networks for Sensor Data Processing in Welding Application, American Welding Institute, Internal Report, 1992.
- 1-53) Rodrigues P.E.L.B., Rogerson J.H., How Weld Metal Thermal Cycles Determine the Toughness of SAWs, Welding and Metall Fabrication, April 1980, pp 183-192.
- 1-54) Van Adrichem Th. J., Kas J., Calculation, Measurement and Simulation of Weld thermal Cycles, Smit-Weld N.V. Nijmegen, Holecctecniek, Netherlands, June 1971. Vol. 1, No.1, pp 2-11.
- 1-55) Bibby M. J., St. Louis M., Trauttmansdorff F., Goldak J. A., Predicting the Fusion and HAZ Hardness of Welds, A Report to AMCA Corporation, Carleton University, February 18, 1983.
- 1-56) Bala S. R., Pond S., Weldability and Formability of Two Candidate Steels for Offshore Structure and Arctic Vessels, AMCA International Ltd., Report No. 1-183-014/F, December 1983, Part B: Table3, Table B2.1.
- 1-57) Kohno R., Jones S. B., An Initial Study of Arc Energy and thermal Cycles in the Submerged Arc Welding of Steel, the Welding Institute, IIW DOC 81/1978/PE, Dec. 1978.

- 1-58) Signes E. G., A Simplified Method of Calculating Cooling Rates in Mild and Low Alloy Steel Weld Metals, *Welding Journal*, Vol. 51, October 1972, pp 475s.
- 1-59) Beaudoin M., Expanding the Carleton University Weldsoft Package, Graduate Technical Report, Welding Engineering Technology Department, Northern College at Kirkland Lake, April, 1993.

References for Chapter 2

- 2-1) Rich E., Knight K., *Artificial Intelligence*, 2nd ed., McGraw-Hill, 1991, pp 487-525.
- 2-2) Muller B., Reinhardt J., *Neural Networks: an Introduction*, Springer-Verlag, 1991, pp 2-21.
- 2-3) Peretto P., *An Introduction to the Modelling of Neural Networks*, Cambridge University Press, 1992.
- 2-4) McCord Nelson M., Illingworth W. T., *A Practical Guide to Neural Nets*, Addison-Wesley Publishing Company, Inc., 1991.
- 2-5) James W., *Psychology (Briefer Course)*, New York, Holt, Chapter XVI, Association, 1890, pp 253-279.
- 2-6) McCulloch W. S., Pitts W., A Logical Calculus of the Ideas Immanent in Nervous Activity, *Bulletin of Mathematical Biophysics* 5, 1943, pp 115-133.
- 2-7) Pitts W., McCulloch W. S., How We Know Universals: the Perception of Auditory and Visual Forms, *Bulletin of Mathematical Biophysics* 9, 1947, pp 127-147.
- 2-8) Hebb D. O., *The Organization of Behaviour: Introduction*, New York, John Wiley & Sons, 1949, pp xi-xix.
- 2-9) Hebb D. O., *The Organization of Behaviour: Chapter 4 The First Stage of Perception: Growth of the Assembly*, New York, John Wiley & Sons, 1949, pp 60-78.
- 2-10) Rosenblatt F., *The Perceptron: A Perceiving and Recognizing Automaton*, Cornell Aeronautical Laboratory, 1957.

- 2-11) Rosenblatt F., The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain, Psychological Review 65, 1958, pp 386-408.
- 2-12) Rosenblatt F., Principles of Neurodynamics: Perceptron and the Theory of Brain Mechanism, Spartan Books, New York, 1962.
- 2-13) Minsky M., Papert S., Perceptrons, Cambridge MA, the MIT Press, 1969.
- 2-14) Hopfield J. J., Neural Networks and Physical Systems with Emergent Collective Computational Abilities, Proceedings of National Academy of Sciences 79, 1982, pp 2554-2558.
- 2-15) Hopfield J. J., Neurons with Graded Response have Collective Computational Properties Like those of Two-State Neurons, Proceedings of the National Academy of Sciences 81, 1984, pp 3088-3092.
- 2-16) Hopfield J. J., Tank D. W., Computing with Neural Circuits: a Model. Science, Vol. 233, Aug 8 1986, pp 625-632, Vol. 235, March 6, 1987, pp 1226-1229.
- 2-17) Grossberg S., How Does a Brain Build a Cognitive Code, Psychological Review 87, 1980, pp 1-51.
- 2-18) Grossberg S., The Adaptive Brain, Vol. I & II, Reidel Press, Boston, 1982.
- 2-19) Kohonen T., Self-Organized Formation of Topologically Correct Feature Maps, Biological Cybernetics 43, 1982, pp 59-69.
- 2-20) Kohonen T., Self-Organization and Associative Memory, 3rd ed., Springer-Verlag, 1988.
- 2-21) Werbos P., Beyond Regression: New Tools for Prediction and Analysis in the Behaviour Science, Ph.D. dis, Harvard University, 1974.
- 2-22) Parker D. B., Lemma Logic, Paper, Stanford University, 1982.
- 2-23) Rumelhart D. E., Hinton G., Williams R. J., Learning Representations by Back Propagating Errors, Nature, 1986, pp 323.
- 2-24) Hinton G. E., McClelland J. L., Rumelhart D. E., Distributed Representation, in Parallel Distributed Processing, Vol. I, edited by Rumelhart and McClelland, Cambridge MA, the MIT Press, 1986, pp 77-109.

- 2-25) Rumelhart D. E., McClelland J. L., ed., Explorations in Parallel Distributed Processing, Cambridge MA, the MIT Press, 1989.
- 2-26) Kosko B., Bidirectional Associative Memories, IEEE Trans., Systems, Man, Cybernetics, SMC-L8, Jan/Feb 1988, pp 46-60.
- 2-27) Kosko B., Constructing an Associative Memory, Byte, Vol. 12, No. 10, Sept 1987, pp 137-144.
- 2-28) Hecht-Nielsen R., Counterpropagation Networks, Proceedings of the IEEE 1st International Conference on Neural Network, 1987.
- 2-29) Dixon R., Neural Nets: Building on Nature's Design, Course Notes, Carleton Professional Development Centre, Ottawa, Ont., April 1993.
- 2-30) Anderson J. A., Rosenfeld E., ed., Neurocomputing: Foundations of Research, Cambridge MA, The MIT Press, 1988.

References for Chapter 3

- 3-1) Rumelhart D. E., Hinton G., Williams R. J., Learning Internal Representation by Error Propagation, in Parallel Distributed Processing, Vol. I, edited by Rumelhart and McClelland, Cambridge MA, the MIT Press, 1986, pp 318-362.
- 3-2) Eberhart R. C., Dobbins R. W., (editors), Neural Network PC Tools: A Practical Guide, Academic Press, Inc., San Diego, CA, 1990, pp 40-43, 240-242.
- 3-3) Dixon R., Neural Nets: Building on Nature's Design, Course Notes, Carleton Professional Development Centre, Ottawa, April 1993.
- 3-4) Lisboa P. G. B. (editor), Neural Networks Current Applications, Chapman & Hall, 1992.
- 3-5) Gelenbe E. (editor), Neural Networks Advances and Application 2, North-Holland, 1992.
- 3-6) Hammerstorm D., Neural Networks at Work, IEEE Spectrum, Computers & Applications, 1993, pp 26-32.

- 3-7) Madden S., Rock A., Xu X., Jones J., Neural Networks for Sensor Data Processing in Welding Application, American Welding Institute, Internal Report, 1992.
- 3-8) Aleksander I., Taylor J., (editors), Artificial Neural Network 2, Vol. 1 & 2, Proceeding of the 1992 International Conference on ANN (ICANN-92), North-Holland, 1992.
- 3-9) Fahlman S. E., Faster Learning Variations on Backpropagation: An Empirical Study, in Proceedings of 1988 Connectionist Models Summer School, D.S. Touretzky, G. Hinton, T. Sejnowski, (editors), Morgan, Kaufmann, 1988.
- 3-10) Chen R., Mars P., Step-Size Variation Methods for Accelerating the Backpropagation Algorithm, International Joint Conference on Neural Network, IJCNN-90-WASH-DC, 1990.
- 3-11) Jacobs R., Increased Rates of Convergence through Learning Rate Adaption, Neural Networks, Vol. 1, 1988.
- 3-12) Segein C. H., Clay R. D., Fault Tolerance in Feed-Forward Artificial Neural Networks, in Neural Networks: Concepts, Application and Implementation, Vol. IV, edited by P. Autognetti, V. Milutinovic, Prentice Hall, pp 111-141.
- 3-13) Solla S. A., le Cun Y., Constrained Neural Networks for Pattern Recognition, in Neural Networks: Concepts, Application and Implementation, Vol. IV, edited by P. Autognetti, V. Milutinovic, Prentice Hall, pp 142-161.
- 3-14) Hirose Y., Yamashita K., Hijiya S., Back-propagation Algorithm which Varies the Number of Hidden Units, Neural Networks, Vol. 4, 1991, pp 61-66.
- 3-15) Pao Y. H., Adaptive Pattern Recognition and Neural Networks, Addison-Wesley Publishing Company, Inc., 1989, pp 129-131, 136-137.
- 3-16) Vogl T. P., Mangis J. K., Rigler A. K., Accelerating the Convergence of the Backpropagation Method, Biological Cybernetics, Vol. 59, 1988, pp 257-262.
- 3-17) Haario H., Jokinen P., Increasing the Learning Speed of Backpropagation Algorithm by Linearization, Proceeding of International Conference on Artificial Neural Network, Finland, 1991, pp 629-634.
- 3-18) Izui Y., Pertland A., Speeding Up Back Propagation, Proceeding of the International Joint Conference on Neural Network, IEEE, Washington, DC, 1990, pp 639-642.

- 3-19) Silva F. M., Almeida L. B., Acceleration Techniques for the Backpropagation Algorithm, Proceedings of Neural Networks EURASIP Workshop, Springer-Verlag, 1990, pp 110-119.
- 3-20) Tveter D., Getting a Fast Break with Backpropagation, AI Expert, July, 1991.
- 3-21) Hammerstorm D., Working with Neural Networks, IEEE Spectrum, Computers & Applications, July 1993, pp 46-53.

References for Chapter 4

- 4-1) Rumelhart D. E., Hinton G., Williams R. J., Learning Internal Representation by Error Propagation, in Parallel Distributed Processing, Vol. I, edited by Rumelhart and McClelland, Cambridge MA, the MIT Press, 1986, pp 318-362.
- 4-2) Fahlman S. E., Faster Learning Variations on Backpropagation: An Empirical Study, in Proceedings of 1988 Connectionist Models Summer School, D.S. Touretzky, G. Hinton, T. Sejnowski, (editors), Morgan, Kaufmann, 1988.
- 4-3) Chen R., Mars P., Step-Size Variation Methods for Accelerating the Backpropagation Algorithm, International Joint Conference on Neural Network, IJCNN-90-WASH-DC, 1990.
- 4-4) Jacobs R., Increased Rates of Convergence through Learning Rate Adaption, Neural Networks, Vol. 1, 1988.

References for Chapter 5

- 5-1) Suzuki H., Carbon Equivalent and Maximum Hardness, Transaction of Japanese Welding Society, 1984, Vol. 15, No. 1, pp 25-33.
- 5-2) Terasaki T., Nomura T., Kitada T., Proposal of Predictive Equation for Maximum Hardness in the Heat-Affected-Zone, Japan Welding Society Journal Quarterly, 1988, Vol. 6, No. 1, pp 139-143.
- 5-3) Yurioka N., Okumura M., Kasuya T., Prediction of HAZ Hardness of Transformable Steel, Metal Construction, Vol. 19, No. 4, 1987, pp 217R-223R.
- 5-4) Adams C. M. Jr., Cooling Rates and Peak Temperatures in Fusion Welding, AWS Welding Journal, Vol. 37, No. 5, 1958, pp 210.

- 5-5) Jhaveri P., Moffat W. G. Adams C. M. Jr., The Effect of Plate Thickness and Radiation on Heat Flow in Welding and Cutting, *Welding Journal*, Vol. 41, No. 1, 1962, pp 12.
- 5-6) Goldak J., Chakravarti A., Bibby M., A New Finite Element Model for Welding Heat Sources, *Metallurgical Transaction*, Vol. 15B, No. 2, 1984, pp 299-305.
- 5-7) Kasuya T., Yurioka N., Predicting of Welding Thermal History by a Comprehensive Solution, Nippon Steel Corporation, Internal Report TM-40, April 1991.
- 5-8) Kohira K., Yatake T., Yurioka N., A Numerical Analysis of the Diffusion and Trapping of Hydrogen in Steels and Steel Weldments, *IIW Doc. IX-951-76*, 1976.
- 5-9) Yurioka N., Ohshita S., Tamehiro H., Study on Carbon Equivalent to Assess Cold Crack ing Tendency and Hardness in Steel Welding, *Proc. Conf. on Pipeline Welding in the 80's*, 1981, 15 pgs., *JWS WM-784-80*, 11 pgs, 1980.
- 5-10) Christensen N., Davies V. de L., Gjermundsen K., Distribution of Temperature in Arc Welding, *British Welding Journal*, Vol. 12, No. 2, 1965, pp 54-75.
- 5-11) Rodrigues P.E.L.B., Rogerson J.H., How Weld Metal Thermal Cycles Determine the Toughness of SAWs, *Welding and Metall Fabrication*, April 1980, pp 183-192.
- 5-12) Van Adrichem Th. J., Kas J., Calculation, Measurement and Simulation of Weld thermal Cycles, *Smit-Weld N.V. Nijmegen, Holecctecniek, Netherlands*, June 1971, Vol. 1, No.1, pp 2-11.
- 5-13) Bibby M.J., St. Louis M., Trauttmansdorff F., Goldak J.A., Predicting the Fusion and HAZ Hardness of Welds, A Report to AMCA Corporation, Carleton University, February 18, 1983.
- 5-14) Bala S. R., Pond S., Weldability and Formability of Two Candidate Steels for Offshore Structure and Arctic Vessels, *AMCA International Ltd., Report No. 1-183-014/F*, December 1983, Part B: Table3, Table B2.1.
- 5-15) Turbo Pascal 7.0 Reference Guide & User's Guide, Borland International. USA. 1992.
- 5-16) Kohno R., Jones S. B., An Initial Study of Arc Energy and thermal Cycles in the Submerged Arc Welding of Steel, the Welding Institute, *IIW DOC 81/1978/PE*, Dec. 1978.

- 5-17) Signes E. G., A Simplified Method of Calculating Cooling Rates in Mild and Low Alloy Steel Weld Metals, *Welding Journal*, Vol. 51, October 1972, pp 475s.
- 5-18) Yurioka N., Impact of Welding Research on Steel Composition Development, *Materials and Design*, Vol. VI, No. 4, 1985, 19 pgs.
- 5-19) B. Chan, M.J. Bibby, A Software Environment for Predicting Weld Features, AWS International Conference on Computerization for Welding Information IV, Orlando, Florida, USA, 1992.
- 5-20) Ito Y., Bessyo K., Weldability Formula of High-Strength Steels Related to Heat-Affected-Zone Cracking, IIW Doc. IX-576-68, 1968, 45 pgs., Transaction of Japanese Welding Society, 1968, Vol. 37, No. 9, pp 983-991, IIW Doc. IX-631-69, 1969, 18 pgs.
- 5-21) The IIW Formula for Carbon Equivalent, Technical Report, 1967, IIW Doc IX-535-67.
- 5-22) Suzuki H., Carbon Equivalent for Steel Weldability, IIW Doc. IX-1230-82, 1982, 13 pgs.

References for Chapter 6

- 6-1) Beaudoin M., Expanding the Carleton University Weldsoft Package, Graduate Technical Report, Welding Engineering Technology Department, Northern College at Kirkland Lake, April, 1993.
- 6-2) Jhaveri P., Moffat W. G. Adams C. M. Jr., The Effect of Plate Thickness and Radiation on Heat Flow in Welding and Cutting, *Welding Journal*, Vol. 41, No. 1, 1962, pp 12.
- 6-3) Chan B., Software for Welding Engineers, Master Thesis, Department of Mechanical and Aerospace Engineering, Carleton University, December, 1990.
- 6-4) Chan B., Chandel R., Bibby M., A Software System for Computing the Size and Shape of GMA (Gas Metal Arc) Welds, ASME Journal of Engineering for Industry, presented at ASME, Production Engineering Division, 1991 Winter Annual Meeting, Atlanta Hilton and Towers, December 1-6, 1991, 'Welding and Joining Processes', ASME PED Vol. 51, Welding and Joining Processes, 1991, pp 215-225.

- 6-5) Chan B., Bibby M., Yang L. J., Chandel R., A Discussion of Algorithms of Predicting Submerged Arc Weld Shapes with Workpiece Edge Preparation, Manufacturing Science and Engineering, PED-vol. 64, presented at the 114th annual ASME Annual Winter Meeting in New Orleans, Nov. 29 - Dec. 3, 1993.
- 6-6) Chan B., Bibby M., Chandel R., A Software System for Anticipating the Size and Shape of Submerged Arc Welds, Journal of Materials Processing Technology, Vol. 40, No. 3-4, pp 249-262, Jan. 1994 (also presented at the Asia Pacific Conference on Material Processing, Singapore, March, 1993).
- 6-7) Madden S., Rock A., Xu X., Jones J., Neural Networks for Sensor Data Processing in Welding Application, American Welding Institute, Internal Report, 1992.
- 6-8) Chandel R. S., Mathematical Modelling of GMAW Features, Modelling and Control of Casting and Welding Process IV, 1988, pp 109-120.
- 6-9) Chandel R. S., An Investigation into the Electrode and Plate Melting Efficiencies of SAW and GMAW Processes, Report MTL 89-10(J), Metals Technology Laboratories, CANMET/EMR, Government of Canada, Ottawa, Ontario, K1A 0G1, 1989.
- 6-10) Yang L. J., A Study on the Submerged Arc Welding Process Variables, Internal Report, School of Mechanical & Production Engineering, Nanyang Technological Institute, Singapore, April 30, 1990.
- 6-11) McGione J. C., Chadwick B. D., The Submerged Arc Butt Welding of Mild Steel. Part I: The Influence of Procedure Parameters on Weld Bead Geometry, The Welding Institute Report No 80/1978/PR, Dec 1978, 26 pgs.
- 6-12) Yurioka N., Ohshita S., Tamehiro H., Study on Carbon Equivalent to Assess Cold Cracking Tendency and Hardness in Steel Welding, Proc. Conf. on Pipeline Welding in the 80's, 1981, 15 pgs., JWS WM-784-80, 11 pgs, 1980.

References for Chapter 7

- 7-1) Adams C. M. Jr., Cooling Rates and Peak Temperatures in Fusion Welding, AWS Welding Journal, Vol. 37, No. 5, 1958, pp 210.
- 7-2) Suzuki H., Carbon Equivalent and Maximum Hardness, Transaction of Japanese Welding Society, 1984, Vol. 15, No. 1, pp 25-33.

- 7-3) Terasaki T., Nomura T., Kitada T, Proposal of Predictive Equation for Maximum Hardness in the Heat-Affected-Zone, Japan Welding Society Journal Quarterly, 1988, Vol. 6, No. 1, pp 139-143.
- 7-4) Chandel R. S., Mathematical Modelling of GMAW Features, Modelling and Control of Casting and Welding Process IV, 1988, pp 109-120.
- 7-5) Chandel R. S., An Investigation into the Electrode and Plate Melting Efficiencies of SAW and GMAW Processes, Report MTL 89-10(J), Metals Technology Laboratories, CANMET/EMR, Government of Canada, Ottawa, Ontario, K1A 0G1, 1989.
- 7-6) Andersen K., Cook G. E., Ramaswany K., Karsai G., Artificial Neural Networks Applied to Arc Welding Process Modelling and Control, IEEE Industry Applications -Conference Record of the 1992 IEEE, Part 2, 1992, pp 2327-2331.
- 7-7) Madden S., Rock A., Xi X., Jones J., Neural Networks for Sensor Data Processing in Welding Application, American Welding Institute, Internal Report, 1992.
- 7-8) Goldak J. A., Patel B., Bibby M. J., Moore, J. E., Computational Weld Mechanics, AGARD Workshop - Structures and Materials 61st Panel Meeting, Oberammergau, Germany, September 8-13, 1985, 32 pgs.
- 7-9) Easterling K. E., Introduction to Physical Metallurgy of Welding, Butterworths & Co. (Publishers) Ltd., London, 1983.
- 7-10) Yurioka N., Ohshita S., Tamehiro H., Study on Carbon Equivalent to Assess Cold Cracking Tendency and Hardness in Steel Welding, Proc. Conf. on Pipeline Welding in the 80's, 1981, 15 pgs., JWS WM-784-80, 11 pgs, 1980.
- 7-11) Bala S. R., Pond S., Weldability and Formability of Two Candidate Steels for Offshore Structure and Arctic Vessels, AMCA International Ltd., Report No. 1-183-014/F. December 1983, Part B: Table3, Table B2.1.
- 7-12) Kohno R., Jones S. B., An Initial Study of Arc Energy and thermal Cycles in the Submerged Arc Welding of Steel, the Welding Institute. IIW DOC 81/1978/PE, Dec. 1978.

Appendices

Appendices I to XIII and the backpropagation network software, NNWork, are included in another volume - *Appendices for Predicting Weld Features Using Artificial Neural Network Technology*. It is available at the Department of Mechanical and Aerospace Engineering, Carleton University or by contacting Dr. Malcolm Bibby or Billy Chan.

END

1 9 - 1 1 - 9 / 6

FIN