Bibliography

[Ackley *et al* 1985] DG Ackley, G Hinton and T Sejnowski, *A Learning Algorithm for Boltzmann Machines*, Cognitive Science, 9, pp 147-169, 1985.

[Aleksander *et al* 1990] I Aleksander and H Morton, *An Introduction to Neural Computing,* Chapman and Hall, 1990.

[Baker 1987] JA Baker, *Reducing Bias and Inefficiency in the Selection Algorithm,* Proceedings of the Second International Conference on Genetic Algorithms: Genetic Algorithms and Their Applications, Lawrence ErlBaum Associates, 1987.

[Baldi 1991] P Baldi, *Computing with Arrays of Bell-Shaped and Sigmoid Functions,* Neural Information Processing Systems, 3, (RP Lippmann, JE Moody and DS Touretzky (eds.)), Morgan Kaufmann, pp 735-742, 1991.

[Barto 1992] AG Barto, *Reinforcement Learning and Adaptive Critic Methods,* in Handbook of Intelligent Control, (DA White and DA Sofge (eds.)), Van Nostrand Reinhold, pp 469-491, 1992.

[Battiti 1992] R Battiti, *First- and Second-OrderMethods for Learning: Between Steepest Descent and Newton's Method,* Neural Computation, 4, pp 141-166, 1992.

[Baum *et al* 1989] EB Baum and D Haussler, *What Size Net Gives Valid Generalizations?*, Neural Computation, 1, pp 151-160, 1989.

[Billings *et al* 1995] SA Billings and GL Zheng, *Radial Basis Function Network Configuration Using Genetic Algorithms,* Neural Networks, 8(6), pp 877-890, 1995.

[Bilbro *et al* 1989] GL Bilbro and WE Snyder, *Range Image Restoration using Mean Field Annealing,* (DS Touretzky, ed.), Advances in Neural Information Processing Systems, 1, Morgan Kaufmann, pp 594-601, 1989.

[Bryson *et al* 1969] AE Bryson and Y-C Yo, *Applied Optical Control,* Hemisphere Publishing, 1969.

[Burdsall *et al* 1997] B Burdsall and C Giraud-Carrier, *GA-RBF: A Self-Optimizing RBF Network,* In Proceedings of the Third International Conference on Artificial Neural Networks and Genetic Algorithms, Springer-Verlag, pp 348-351, 1997.

[Chakraborty *et al* 1992] K Chakraborty, K Mehtotra, CK Mohan and S Ranka, *Forecasting the Behavior of Multivariate Time Series using Neural Networks,* Neural Networks, 5, pp 961-970, 1992.

[Chan *et al* 1987] LW Chan and F Fallside, *An Adaptive Training Algorithm for Backpropagation Networks,* Nature, 264, pp 705-712, 1987.

[Chang *et al* 1991] EI Chang and RP Lippmann, *Using Genetic Algorithms to Improve Pattern Classification Performance,* In Advances in Neural Information processing Systems, 3, Morgan Kaufmann, pp 797-803, 1991.

[Churchland *et al* 1992] PS Churchland and TJ Sejnowski, *The Computational Brain,* MIT Press, 1992.

[Cibas *et al* 1996] T Cibas, F Fogelman, P Soulié and S Raudys, *Variable selection with Neural Networks,* Neurocomputing, 12, pp 223-248, 1996.

[Cohen *et al* 1993] M Cohen, H Franco, N Morgan, D Rumelhart and V Abrash, *Context Dependent Multiple Distribution Phonetic Modeling with MPLs,* In Advances in Neural Information Processing Systems, 5, (SJ Hanson, JD Cowan and C Lee Giles (eds.)), Morgan Kaufmann, pp 649-657, 1993.

[Cybenko 1969] G Cybenko, *Approximation by Superpositions of a Sigmoidal Function,* Mathematical Control Signals Systems, 2, pp 203-204, 1969.

[Dagli *et al* 1995] CH Dagli and J James, *Use of Genetic Algorithms for Encoding Efficient Neural Network Architectures: Neuro-Computer Implementation,* In SPIE Proceedings of Applications and Science of Artificial Neural Networks, 2492, pp 323-330, 1995.

[Davis 1991] L Davis, *Handbook of Genetic Algorithms,* Von Nostrand Reinhold, 1991.

[De Jong 1975] KA De Jong, *An Analysis of the Behavior of a Class of Genetic Adaptive Systems,* PhD Thesis, University of Michigan, 1975.

[Durbin *et al* 1989] R Durbin and DE Rumelhart, *Product units: A Computationally Powerful and Biologically Plausible Extension to Backpropagation Networks,* Neural Computation, 1, pp 133-142, 1989.

[Eberhart *et al* 1990] RC Eberhart and RW Dobbins, *Case Study I: Detection of Electrocephalogram Spikes,* in Neural Networks PC Tools, (RC Eberhart, RW Dobbins (eds.)), Academic Press, 1990.

[Eberhart *et al* 1995] RC Eberhart and J Kennedy, *A New Optimizer Using Particle Swarm Theory,* Proceedings of the Sixth International Symposium on Micro Machine and Human Science, IEEE Service Center, Piscataway, NJ, pp 39-43, 1995.

[Eberhart *et al* 1996] RC Eberhart, P Simpson and R.W Dobbins, *Computational Intelligence PC Tools*, Academic Press Limited, 1996.

[Eberhart *et al* 1998] RC Eberhart, Z He, C Wei, L Yang, X Gao, S Yao and Y Shi, *Extracting Rules from Fuzzy Neural Networks by Particle Swarm Optimization,* IEEE International Conference on Evolutionary Computation, Alaska, 1998.

[Eberhart *et al* 1999] RC Eberhart and X Hu, *Human Tremor Analysis using Particle Swarm Optimization,* In Proceedings of Congress on Evolutionary Computation 1999, Piscataway, pp 1951-1957, 1999.

[Eberhart *et al* 2000] RC Eberhart and Y Shi, *Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization,* In Proceedings of International Congress on Evolutionary Computation 2000, San Diego, CA, pp 84-88, 2000.

[Elman 1990] JL Elman, *Finding Structure in Time*, Cognitive Science, 14, pp 179-211, 1990.

[Engelbrecht *et al* 1995a] AP Engelbrecht, I Cloete, J Geldenhuys, JM Zurada, *Automatic Scaling using Gamma Learning in Feedforward Neural Networks*, In International Workshop on Artificial Neural Networks, (J Mira and F Sandoval (eds.)), 'From Natural Science to Artificial Neural Computing' in the series 'Lecture notes in Computer Science', 930, pp 374-381, 1995.

[Engelbrecht *et al* 1995b] AP Engelbrecht, I Cloete and JM Zurada, *Determining the Significance of Input Parameters using Sensitivity Analysis,* International Workshop on Artificial Neural Networks, In 'From Natural Science to Artificial Neural Computing' (J Mira and F Sandoval (eds.)), In the Series 'Lecture Notes in Computer Science', 930, pp 382-388, 1995.

[Engelbrecht *et al* 1999a] AP Engelbrecht and A Ismail, *Training Product Unit Neural Networks,* Stability and Control: Theory and Applications, 2 (1/2), pp 59-74, 1999.

[Engelbrecht *et al* 1999b] AP Engelbrecht and I Cloete, *A Sensitivity Analysis Algorithm for Pruning Feedforward Neural Networks,* IEEE International Conference on Neural Networks, 2, Washington DC, USA, pp 1274-1277, 1996.

[Engelbrecht *et al* 1999c] AP Engelbrecht, L Fletcher, I Cloete, *Variance Analysis of Sensitivity Information for Pruning Multilayer Feedforward Neural Networks,* IEEE International Joint Conference on Neural Networks, Washington DC, USA, paper 379, 1999.

[Engelbrecht *et al* 1999d] AP Engelbrecht and I Cloete, *Incremental Learning using Sensitivity Analysis,* IEEE Joint Conference on Neural Networks, Washington, paper 380, 1999.

[Engelbrecht 2001] AP Engelbrecht, *A New Pruning Heuristic Based on Variance Analysis of Sensitivity Information,* IEEE Transactions on Neural Networks, 12(6), 2001.

[Fahlman 1989] SE Fahlman, *Fast Learning Variations on Back-Propagation: An Empirical Study,* In Proceedings of the 1988 Connectionist Models Summer School (Pittsburgh, 1988), (D Touretzky, G Hinton and T Sejnowski (eds.)), Morgan Kaufmann, pp 38-51, 1989.

[Fahlman *et al* 1990] SE Fahlman and C Lebiere, *The Cascade-Correlation Learning Architecture,* In Advances in Neural Information Processing Systems 2 (DS Touretzky, ed.), Morgan Kaufmann, pp 524-532, 1990.

[Fausett 1994] L Fausett, *Fundamentals of Neural Networks: Architectures, Algorithms and Applications,* Prentice Hall, 1994.

[Finnoff 1993a] W Finnoff, *Diffusion Approximations for the Constant Learning Rate of Backpropagation Algorithm and Resistance to Local Minima,* Advances in Neural

Information Processing Systems, 5, (SJ Hanson, JD Cowan and C Lee Giles (eds.)), Morgan Kaufmann, pp 459-466, 1993.

[Finnoff 1993b] W Finnoff, F Hergert and HG Zimmermann, *Improving Model Selection by Nonconvergent Methods,* Neural Networks, 6, pp 771-783, 1993.

[Fletcher *et al* 1995] J Fletcher and Z Obradovic, *A Discrete Approach to Constructive Neural Network Learning,* Neural, Parallel and Scientific Computations, 3(3), pp 307-320, 1995.

[Foo *et al* 1988] YPS Foo and Y Takefuji, *Integer Linear Programming Neural Networks for Job-Shop Scheduling,* Proceedings of the IEEE International Conference on Neural Networks, San Diego, California, 1988.

[Franzini 1987] MA Franzini, *Speech Recognition with Back Propagation,* Proceedings of the Nineth Annual Conference of the IEEE Engineering in Medicine and Biology Society, New York, pp 1702-1703, 1987.

[Frean 1990] M Frean, *The Upstart Algorithm: A Method for Constructing and Training Feedforward Neural Networks,* Neural Computation, 2, pp 198-209, 1990.

[Fu *et al* 1993] L Fu and T Chen, *Sensitivity Analysis for Input Vector in Multilayer Feedforward Neural Networks,* IEEE International Conference on Neural Networks, 1, 1993, pp 215-218, 1993.

[Fukushima 1975] K Fukushima *Cognitron: A Self-Organizing Multilayered Neural Network,* Biological Cybernetics, 20, pp 121-136, 1975.

[Funahashi 1989] KI Funahashi, *On the Approximate Realization of Continuous Mappings by Neural Networks*, Neural Networks, 2, pp 183-192, 1989.

[Gallant 1992] AR Gallant and H White, *On Learning the Derivatives of an Unknown Mapping with Multilayer Feedforward Networks*, Neural Networks, 5, pp 129-138, 1991.

[Gallant 1986] SI Gallant, *Three Constructive Algorithms for Network Learning*, In Proceedings of the Eighth Annual Conference of the Cognitive Science Society, pp 652-660, 1986.

[Ghosh *et al* 1992] J Ghosh and Y Shin, *Efficient High-Order Neural Networks for Classification and Function Approximation*, International Journal of Neural Systems, 3(4), pp 323-325, 1992.

[Ghosh *et al* 1994] J Ghosh and K Turner, *Structural Adaptation and Generalization in Supervised Feed-Forward Networks*, Journal of Artificial Neural Networks, 1(4), pp 431-458, 1994.

[Girosi *et al* 1995] F Girosi and T Poggio, *Regularization Theory and Neural Networks Architectures*, Neural Computation, 7, pp 219-269, 1995.

[Goldberg 1989] DE Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, 1989.

[Grossberg 1987] S Grossberg, *Competitive Learning: from Interactive Activation to Adaptive Resonance*, Cognitive Science, 11, pp 23-63, 1987.

[Guo *et al* 1992] Z Guo and RO Uhrig, *Use of Genetic Algorithms to Select Inputs for Neural Networks*, In Proceedings on Combination of Genetic Algorithms and Neural Networks, pp 223-234, 1992.

[Gurney 1992] KN Gurney, *Training Nets of Hardware Realizable Sigma-Pi Units*, Neural Networks, 5, pp 289-303, 1992.

[Guyon 1990] I Guyon, *Neural Networks and Applications*, Computer Physics Reports, Amsterdam: Elsevier, 1990.

[Haffner *et al* 1988] P Haffner, A Waibel, H Sawai and K. Shikano, *Fast Back-Propagation Learning Methods for Neural Networks in Speech,* Technical Report TR-1-0058, ATR Interpreting Telephony Research Laboratories, Osaka, Japan, 1988.

[Harrison *et al* 1991] R Harrison, S Marshall and R Kennedy, *The Early Diagnosis of Heart Attacks: A Neurocomputational Approach,* IEEE International Joint Conference on Neural Networks, 1, Seattle, pp 1-5, 1991.

[Hanson *et al* 1989] SJ Hanson and LY Pratt, *Comparing Biases for Minimal Network Construction with Back-Propagation*, Advances in Neural Information Processing Systems, 1, (DS Touretzky ed.), pp 177-185, 1989.

[Hassibi *et al* 1994] B Hassibi and DG Stork, *Second Order Derivatives for Network Pruning: Optimal Brain Surgeon,* Advances in Neural Information Processing Systems, 5, (SJ Hanson, JD Cowan, C Lee Giles (eds.)), Morgan Kaufmann, 1994.

[Hassoun 1995] MH Hassoun, *Fundamental Artificial Neural Networks,* MIT Press, 1995.

[Haykin *et al* 1992] S Haykin and TK Bhattacharya, *Adaptive Radar Detection Using Supervised Learning Networks,* Computational Neuroscience Symposium, Indiana University - Purdue University at Indianapolis, pp 35-51, 1992.

[Haykin 1994] S Haykin, *Neural Networks: A Comprehensive Foundation*, MacMillan Publishing Company, 1994.

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

[Heppner *et al* 1990] F Heppner and U Grenander, *A Stochastic Nonlinear Model for Coordinated Bird Flocks*, in The Ubiquity of Chaos (S Kraner, ed.), AAAS Publications, 1990.

[Hinton 1987] GE Hinton, *Connectionist Learning Procedures,* Reproduced in Machine Learning: Paradigms and Methods (J Carbonell ed.), MIT Press, pp 185-234, 1990.

[Hirose *et al* 1991] Y Hirose, K Yamashita and S Hijiya, *Back-propagation Algorithm which Varies the Number of Hidden Units*, Neural Networks, 4, pp 61-66, 1991.

[Holland 1992] JH Holland, *Adaptation in Natural and Artificial Systems,* MIT Press, 1992.

[Holm *et al* 1999] JEW Holm and EC Botha, *Leap-frog is a Robust Algorithm for Training Neural Networks,* Network Computation in Neural Systems, 10, pp 1-13, 1999.

[Holtzman 1992] JM Holtzman, *On Using Perturbation Analysis to do Sensitivity Analysis: Derivatives versus Difference,* IEEE Transactions on Automatic Control, 37(2), pp 243-247, 1992.

[Hopfield 1982] JJ Hopfield, *Neural Networks and Physical Systems with Emergent Collective Computational Properties,* Proceedings of the National Academy of Sciences of the USA, 79, pp 2554-2588, 1982.

[Hornik *et al* 1989a] K Hornik, M Stinchcombe and H White, *Universal Approximation of an Unknown Mapping and its Derivatives Using Multilayer Feedforward Networks*, Neural Networks, 3, pp 551-560, 1989.

[Hornik *et al* 1989b] K Hornik, M Stinchcombe and H White, *Multilayer Feedforward Networks are Universal Approximators*, Neural Networks, 2(5), pp 359-366.

[Hsieh *et al* 1998] WW Hsieh and B Tang, *Applying Neural Network Models to Prediction and Data Analysis in Meteorology and Oceanography,* Bulletin of the American Meteorological Society, 79, pp 1855-1870, 1998.

[Hush *et al* 1991] DR Hush, JM Salas and B Horne, *Error Surfaces for Multilayer Perceptrons,* IEEE International Joint Conference on Neural Networks, Seattle, I, pp 759-764, 1991.

[Hush *et al* 1993] DR Hush and GG Horne, *Progress in Supervised Neural Networks: What's New Since Lippmann?,* IEEE Signal Processing Magazine, 10, pp 8-39, 1993.

[Hussain *et al* 1997] A Hussain, JJ Soraghan and TS Durbani, *A New Neural Network for Nonlinear Time-Series Modelling*, Neurovest Journal, pp 16-26, 1997.

[Ismail *et al* 1999] A Ismail and AP Engelbrecht, *Training Product Units in Feedforward Neural Networks using Particle Swarm Optimization,* In Development and Practice of Artificial Intelligence Techniques, (VB Bajic and D Sha (eds.)), Proceedings of the International Conference on Artificial Intelligence, pp 36-40, 1999.

[Ismail *et al* 2000] A Ismail and AP Engelbrecht, *Global Optimization Algorithms for Training Product Unit Neural Networks,* IEEE International Joint Conference on Neural Networks, paper 032, 2000.

[Janson *et al* 1993] DJ Janson and JF Frenzel, *Training Product Unit Neural Networks with Genetic Algorithms*, IEEE Expert Magazine, pp 26-33, 1993.

[Jordan 1986] *Attractor Dynamics and Parallelism in a Connectionist Sequential Machine,* In Proceedings of the Eighth Annual Conference of the Cognitive Science Society, Erlbaum, pp 531-546, 1986.

[Jordan *et al* 1990] MI Jordan and RA Jacobs, *Learning to Control an Unstable System with Forward Modeling,* In Advances in Neural Information Processing Systems, 2 (DS Touretzky, ed.), Morgan Kaufmann, pp 324-331, 1990.

[Karnin 1990] ED Karnin, *A Simple Procedure for Pruning Back-Propagation Trained Neural Networks,* IEEE Transactions on Neural Networks, 1(2), pp 239-242, 1990.

[Kawato 1990] M Kawato, *Computational Schemes and Neural Network Models for Formation and Control of Multijoint Arm Trajectory,* In Neural Networks for Robotics and Control, (T Miller, R Sutton and P Werbos (eds.)), MIT Press, 1990.

[Kennedy 1995a] J Kennedy, *The Particle Swarm: Social Adaptation of Knowledge,* Proceedings of the IEEE International Conference on Evolutionary Computation, Indianapolis, Indiana, IEEE Service Center, Piscataway, NJ, pp 303-308, 1995.

[Kennedy *et al* 1995b] J Kennedy and RC Eberhart, *Particle Swarm Optimization,* Proceedings of the IEEE International Conference on Neural Networks (Perth, Australia), IEEE Service Center, Piscataway, NJ, IV, pp 1942-1948, 1995.

[Kirkpatrick *et al* 1983] S Kirkpatrick, C Gelatt and M Vecchi, *Optimization by Simulated Annealing*, Science, 220, pp 671-680, 1983.

[Klassen *et al* 1988] MS Klassen and YH Pao, *Characteristics of the Functional-link Net: A Higher Order Delta Rule Net,* IEEE Proceedings of Second Annual International Conference on Neural Networks, 1988.

[Kohonen 1988a] T Kohonen, *The Neural Phonetic Typewriter,* IEEE Computer, 27(3), pp 11-22, 1988.

[Kohonen 1988b] T Kohonen, *Self-organization and Associative Memory,* Springer-Verlag, 1988.

[Kolen *et al* 1990] JF Kolen and JB Pollack, *Back Propagation is Sensitive to Initial Conditions,* Technical Report TR 90-JK-BPSIC, Laboratory for Artificial Intelligence Research, Computer and Information Science Department, The Ohio State University, Columbus, 1990.

[Kramer *et al* 1989] AH Kramer and A Sangiovanni-Vincentelli, *Efficient Parallel Learning Algorithms for Neural Networks,* Advances in Neural Information Processing Systems, 1, (DS Touretzky, ed.), Morgan Kaufmann, pp 40-48, 1989.

[Kuo *et al* 1994] IE Kuo and SS Melsheimer, *Using Genetic Algorithms to Estimate the Optimum Width Parameter in Radial Basis Function Networks,* In Proceedings of the 1994 American Control Conference, 1994.

[Kwok *et al* 1995] T-Y Kwok and D-Y Yeung, *Constructive Feedforward Neural Networks for Regression Problems: A Survey,* Technical Report HKUST-CS95-43, Department of Computer Science, The Hong Kong University of Science and Technology, 1995.

[Lange *et al* 1996] S Lange and T Zeugmann, *Incremental Learning from Positive Data,* Journal of Computer and System Sciences, 53, pp 88-103, 1996.

[Lawrence *et al* 2000] S Lawrence and C Lee Giles, *Overfitting and Neural Networks: Conjugate Gradient and Backpropagation,* In Proceedings of the IEEE International Joint Conference on Neural Networks, Como, Italy, July 2000.

[Le Cun 1989] Y Le Cun, *Generalization and Network Design Strategies,* In Connectionism in Perspective, (R Pfeifer, Z Schreter, F Fogelman-Soulié and L. Steels (eds.)), Amsterdam: North-Holland, pp 143-155, 1989.

[Le Cun *et al* 1990] Y Le Cun, JS Denker and SA Solla, *Optimal Brain Damage,* Advances in Neural Information Processing Systems, 2, (DS Touretzky, ed.), Morgan Kaufmann, pp 598-605, 1990.

[Lee *et al* 1991] Y Lee, S Oh and M Kim, *The Effect of Initial Weights on Premature Saturation in Back-propagation Learning,* International Joint Conference on Neural Networks, 1, Seattle, pp 765-770, 1991.

[Lee Giles 1987] C Lee Giles, *Learning, Invariance and Generalization in Higher-Order Neural Networks,* Applied Optics, 26(23), pp 4972-4978, 1987.

[Leerink *et al* 1995] LR Leerink, C Lee Giles, BG Horne and MA Jabri, *Learning with Product Units,* Advances in Neural Information Processing Systems, (G Tesauro, D Touretzky and T Leen (eds.)), 7, pp 537-544, 1995.

[Li *et al* 1996] JY Li and TWS Chow, *Functional Approximation of Higher-Order Neural Networks,* Journal of Intelligent Systems, (R Paul and R Macredie (eds.)), 6, 1996.

[MacLeod 1990] K MacLeod, *An Application Specific Neural Model for Document Clustering,* In Proceedings of the Fourth Annual Parallel Processing Symposium, 1, pp 5-16, 1990.

[Maxwell *et al* 1986] T Maxwell, C Lee Giles, YC Lee and HH Chen, *Nonlinear Dynamics of Artificial Neural Systems,* In Neural Networks For Computing, (J Denker (ed.)), New York: American Institute of Physics, p 299, 1986.

[Milenković *et al* 1996] S Milenković, Z Obradović and V Litovski, *Annealing Based Dynamic Learning in Second-Order Neural Networks,* Technical Report, Department of Electronic Engineering, University of Nis, Yugoslavia, 1996.

[Minsky 1961] ML Minsky, *Steps Towards Artificial Intelligence,* In Proceedings of the Institute of Radio Engineers, 49, pp 8-30, 1961.

[Minsky *et al* 1969] ML Minsky and SA Papert, *Perceptrons: An Introduction to Computational Geometry,* MIT Press, 1988.

[Mitchel 1996] M Mitchel, *An Introduction to Genetic Algorithms,* MIT-Press, 1996.

[Moody *et al* 1996] J Moody and PJ Antsaklis, *The Dependence Identification Neural Network Construction Algorithm,* IEEE Transactions on Neural Networks, 7(1), pp 3-15, 1996.

[Moreira *et al* 1995] M Moreira and E Fiesler, *Neural Networks with Adaptive Learning Rate and Momentum Terms,* In Technical Report 95-04 Institut Dalle Molle d'Intelligence Artificielle Perceptive, Martigny, Switzerland, 1995.

[Mozer *et al* 1989] MC Mozer and P Smolensky, *Skeletonization: A Technique for Trimming the Fat from a Network Via Relevance Assessment,* Advances in Neural Information Processing, 1, (DS Touretzky ed.), Morgan Kaufmann, pp 107-115, 1989.

[Nigrin 1993] A Nigrin, *Neural Networks for Pattern Recognition,* MIT Press, p 11, 1993.

[Oh *et al* 1995] S-H Oh and Y Lee, *Sensitivity Analysis of Single Hidden-Layer Neural Networks with Threshold Functions,* IEEE Transactions on Neural Networks, 6(4), pp 1005-1007, 1995.

[Opitz *et al* 1994] DW Opitz and JW Shavlik, *Genetically Refining Topologies of Knowledge-based Neural Networks,* In International Symposium on Integrating Knowledge and Neural Heuristics, pp 57-66, 1994.

[Pao 1989] YH Pao, *Adaptive Pattern Recognition and Neural Networks,* Addison Wesley Publishing, 1989.

[Pao *et al* 1992] YH Pao and Y Takefuji, *Functional-Link Net Computing: Theory, System Architecture and Functionalities*, IEEE Computer, 25(5), pp 76-79, 1992.

[Parker 1985] DB Parker, *Learning-logic: Casting the Cortex of the Human Brain in Silicon*, Technical Report TR-47, Center for Computational Research in Economics and Management Science, MIT, Cambridge, 1985.

[Pitts *et al* 1943] W Pitts and WS McCulloch, *A Logical Calculus of the Ideas Imminent in Nervous Activity*, Bulletin of Mathematical Biophysics, 5, pp 115-133, 1943.

[Poggio *et al* 1990] T Poggio and F Girosi, *Networks for Approximation and Learning*, In Proceedings of the IEEE, 78(9), pp 1481-1497, 1990.

[Pomerlau 1989] DA Pomerlau, *ALVINN: An Autonomous Land Vehicle in a Neural Network*, Advances in Neural Information Processing, 1, (D Touretzky ed.), Morgan-Kaufmann, 1989.

[Prechelt 1994] L Prechelt, *PROBEN1- A Set of Benchmarks and Benchmarking Rules for Neural Network Training Algorithms*, Technical Report 21/94, Fakultät für Informatik, Universität Karlsruhe, D-76128 Karlsruhe, Germany, Anonymous FTP:/pub/papers/techreports/1994/1994-21.ps.Z on ftp.ira.uka.de, 1994.

[Redding *et al* 1993] NJ Redding, A Kowalczyk and T Downs, *Constructive Higher-Order Network Algorithm that is Polynomial in Time*, Neural Networks, 6, pp 997-1010, 1993.

[Reed 1994] R Reed, *Pruning Algorithms - A Survey*, IEEE Transactions on Neural Networks, 4(5), pp 740-747, 1993.

[Reynolds 1987] CW Reynolds, *Flocks, Herds and Schools: A Distributed Behavioral Model*, Computer Graphics, 21(4), pp 25-34, 1987.

[Rojas 1996] R Rojas, *Neural Networks: A Systematic Introduction*, Springer-Verlag, 1996.

[Rosenblatt 1962] F Rosenblatt, *Principles of Neurodynamics*, Spartan Books, 1962.

[Rowley *et al* 1996] H Rowley, S Baluja, and T Kanade, *Human Face Detection in Visual Scenes*, Advances in Neural Information Processing Systems, 8, pp 875-881, 1996.

[Rumelhart *et al* 1985] DE Rumelhart and Zipser, *Feature Discovery by Competitive Learning*, Cognitive Science, 9, pp 75-112, 1985.

[Rumelhart *et al* 1986a] DE Rumelhart and JL McClelland, *Parallel Distributed Processing: Exploration in the Microstructure of Cognition*, 1, MIT Press, 1986.

[Rumelhart *et al* 1986b] DE Rumelhart, GE Hinton and RJ Williams. *Learning Internal Representations by Back-propagation Errors*, Nature, 323, pp 533-536, 1986.

[Saund 1989] E Saund, *Dimensionality Reduction Using Connectionist Networks*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 11, pp 304-314, 1989.

[Salerno 1997] J Salerno, *Using the Particle Swarm Optimization Technique to Train a Recurrent Neural Model*, Proceedings of the Nineth IEEE International Conference on Tools with Artificial Intelligence, 1997.

[Schalkoff 1997] RJ Schalkoff, *Artificial Neural Networks*, McGraw-Hill Series in Computer Science, 1997.

[Schiffman *et al* 1992] W Schiffmann, M Joost and R Werner, *Synthesis and Performance Analysis of Multilayer Neural Network Architectures*, Technical Report, Institut für Physik, Universität Koblenz, 16/1992, 1992.

[Sejnowski *et al* 1987] TJ Sejnowski and CR Rosenberg, *Parallel Networks that Learn to Pronounce English Text,* Complex Systems, 1, pp 145-168, 1987.

[Shi *et al* 1998] Y Shi and RC Eberhart, *Parameter Selection in Particle Swarm Optimization,* The Seventh Annual Conference on Evolutionary Programming, pp 591-600, 1998.

[Shi *et al* 1999] Y Shi and RC Eberhart, *Empirical Study of Particle Swarm Optimization,* In Proceedings of Congress on Evolutionary Computation 1999, pp 1945-1950, 1999.

[Shin *et al* 1995] Y Shin and J Ghosh, *Ridge Polynomial Networks,* IEEE Transactions on Neural Networks, 6(2), pp 610-622, 1995.

[Shittenkopf *et al* 1997] C Schittenkopf, G Deco and W Brauer, *Two Strategies to Avoid Overfitting in Feedforward Neural Networks,* Neural Networks, 10(30), pp 505-516, 1997.

[Sietsma *et al* 1988] J Sietsma and RJF Dow, *Neural Net Pruning - Why and How,* IEEE International Conference on Neural Networks, I, pp 325-333, 1988.

[Sietsma *et al* 1991] J Sietsma and RJF Dow, *Creating Artificial Neural Networks that Generalize,* Neural Networks, 4(1), pp. 67-69, 1991.

[Silva *et al* 1990] FM Silva and LB Almeida, *Acceleration Techniques for the Backpropagation Algorithm,* Neural Networks, Europe Lecture Notes in Computer Science (LB Almeida and Wellekens (eds.)), Springer-Verlag, pp 110-119, 1990.

[Snyman 1982a] JA Snyman, *A New and Dynamic Method for Unconstrained Minimization,* Applied Mathematical Modelling, 6, pp 449-462, 1982.

[Snyman 1982b] JA Snyman, *An Improved Version of the Original LeapFrog Dynamic Method for Unconstrained Minimization: LFOP1(b)*, Applied Mathematical Modelling, 7, pp 216-218, 1983.

[Stinchcombe *et al* 1989] M Stinchcombe and H White, *Universal Approximations using Feedforward Networks with Non-Sigmoid Hidden Layer Activation Functions*, Proceedings of the International Joint Conference on Neural Networks, I, pp 613-617, 1989.

[Sutton 1986] R Sutton, *Two Problems with Backpropagation and Other Steepest-Descent Learning Procedures for Networks*, In Proceedings of the 8th Annual Conference on the Cognitive Science Society, Amherst, pp 823-831, 1986.

[Syswerda 1991] G Syswerda *Schedule Optimization using Genetic Algorithms*, In Handbook of Genetic Algorithms, (L Davis (ed.)), Van Nostrand Reinhold, 1991.

[Thimm *et al* 1995] G Thimm and E Fiesler, *Evaluating Pruning Methods*, Proceedings of the International Symposium on Artificial Neural Networks, pp 20-25, 1995.

[Van den Bergh 1999] F van den Bergh, *Particle Swarm Weight Initialization in Multi-layer Perceptron Artificial Neural Networks*, In Development and Practice of Artificial Intelligence Techniques, Proceedings of International Conference on Artificial Intelligence, pp 41-45, 1999.

[Van den Bergh *et al* 2000] F van den Bergh and AP Engelbrecht, *Cooperative Learning in Neural Networks using Particle Swarm Optimizers*, South African Computer Journal, 26, pp 84-90, 2000.

[Van den Bergh 2001a] F van den Bergh, *Analysis of Particle Swarm Optimizers*, PhD Thesis, Department of Computer Science, University of Pretoria, submitted 2001.

[Van den Bergh *et al* 2001b] F van den Bergh and AP Engelbrecht, *Using Cooperative Particle Swarm Optimization to Train Product Unit Neural Networks,* in Proceedings of International Joint Conference on Neural Networks, Washington, 2001.

[Van den Bergh *et al* 2001c] F van den Bergh, AP Engelbrecht and DG Kourie, *A Convergence Proof for Particle Swarm Optimizers,* submitted to IEEE Transactions on Evolutionary Computing, 2001.

[Van den Bergh *et al* 2001d] F van den Bergh and AP Engelbrecht,*Effects of Swarm Size on Cooperative Particle Swarm Optimizers,* In Proceedings of Genetic and Evolutionary Computation Conference 2001, San Francisco, 2001.

[Von Lehman *et al* 1988] A Von Lehman, EG Liao, PF Marrakchi and JS Patel, *Factors Influencing Learning by Back-propagation,* IEEE International Conference on Neural Networks, I, pp 765-770, 1988.

[Weigend *et al* 1991] AS Weigend, DE Rumelhart and BA Huberman, *Generalization by Weight Elimination with Application to Forecasting,* Advances in Neural Information Processing Systems, 3, (R Lippman and J Moody and DS Touretzky (eds.)), pp 872-882, 1991.

[Werbos 1974] PJ Werbos, *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences,* Ph.D Thesis, Harvard University, Cambridge, 1974.

[Werbos 1989] PJ Werbos, *Backpropagation and Neurocontrol: A Review and Prospectus,* International Joint Conference on Neural Networks, 1, pp 209-216, 1989.

[Wessels *et al* 1992] LFA Wessels and E Barnard, *Avoiding False Local Minima by Proper Initialization of Connections,* IEEE Transactions on Neural Networks, 3(6), pp 899-905, 1992.

[Whitehead *et al* 1996] BA Whitehead and TD Choate, *Cooperative-Competitive Genetic Evolution of Radial Basis Function Centers and Widths for Time Series Prediction,* IEEE Transactions on Neural Networks, 7(4), pp 869-881, 1996. D.

[Whitley *et al* 1990] D Whitley and C Bogart, *The Evolution of Connectivity: Pruning Neural Networks using Genetic Algorithms*, International Joint Conference on Neural Networks, 1, pp 134-137, 1990.

[Widrow *et al* 1960] B Widrow and ME Hoff, *Adaptive Switching Circuits,* IRE WESCON Convention Record, pp 96-104, 1960.

[Wieland *et al* 1987] A Wieland and R Leighton, *Geometric Analysis of Neural Network Capabilities*, First IEEE International Conference on Neural Networks, 3, pp 385-392, 1987.

[Wilson 1975] EO Wilson, *Sociology: The New Synthesis*, Belknap Press, 1975.

[Zamparelli 1997] M Zamparelli, *Genetically Trained Cellular Networks,* Neural Networks, 10(6), pp 1143-1151, 1997.

[Zhang 1994] B-T Zhang, *Accelerated Learning by Active Example Selection,* International Journal of Neural Systems, 5(1), pp 67-75, 1994.

[Zhang *et al* 1997] J Zhang and A Morris, *A Sequential Learning Approach for Single Hidden Layer Neural Networks*, Neural Networks, 11, pp 65-80, 1997.

[Zurada 1992] JM Zurada, *Introduction to Artificial Neural Systems,* West Publishing Company, 1992.

[Zurada *et al* 1997] JM Zurada, A Malinowski and S Usui, *Perturbation Method for Deleting Redundant Inputs of Perceptron Networks,* Neurocomputing, 4, pp 177-193, 1997.

Appendix A

Derivation of learning rules for PUNNs

The product learning equations for the feed-forward neural network type used in this thesis are derived in this appendix. This thesis assumes a network architecture which consists of an input layer, a hidden layer consisting of product units and an output layer consisting of summation units. Linear activations are assumed for all units. This thesis assumes a PUNN architecture with a bias to the output units and no bias to the hidden units. Instead, an extra unit referred to as a 'distortion unit', is included to the hidden units. For a discussion on the 'distortion unit', refer to section 3.7.3 on page 72. Section A.1 derives the learning equations for a PUNN architecture with biases to both the hidden and output units. In section A.2 the equations of section A.1 are then adapted for a PUNN where a 'distortion unit' replaces the bias in the hidden layer. The derivations assume gradient descent as optimization algorithm and on-line learning.

The mean squared error (MSE) function is assumed as the objective function, with linear activation functions in both, the hidden and output layers of the product unit neural network (PUNN).

The objective function is expressed as,

$$E = \frac{\sum_{p=1}^{P} E_p}{PK} \tag{A.1}$$

where P is the total number of patterns in the training set, K is the number of output units, and E_p is the error of pattern p, defined as

$$E_p = \frac{1}{2} \cdot \sum_{k=1}^{K} (t_{k,p} - o_{k,p})^2 \tag{A.2}$$

where $t_{k,p}$ and $o_{k,p}$ are respectively the target and actual output values of the k^{th} output unit, O_k, when pattern p is presented to the neural network.

The derivations in this appendix refer to individual patterns. For the sake of notational convenience the superscript p, that refers to a specific pattern, is removed. Throughout this appendix I, J and K refer, respectively, to the number of input, hidden and output units excluding biases.

The output of the k^{th} output unit is (under the assumption of linear activated outputs)

$$\begin{aligned} o_k &= f(net_{o_k}) \\ &= net_{o_k} \end{aligned} \tag{A.3}$$

and

$$f'(net_{o_k}) = 1 \tag{A.4}$$

The net input signal is calculated as

$$net_{o_k} = \sum_{j=1}^{J+1} w_{kj} y_j \tag{A.5}$$

The $(J+1)^{th}$ unit represents the bias to each output unit; w_{kj} is the weight between the j^{th} hidden and k^{th} output units; y_j is the output of the j^{th} hidden unit, defined as

(assuming linear activation)

$$y_j = f(net_{y_j}) \tag{A.6}$$
$$= net_{y_j} \tag{A.7}$$

and

$$f'(net_{y_j}) = 1 \tag{A.8}$$

A.1 Learning rules for a PUNN using a bias unit

This section derives the learning equations for a PUNN where it is assumed that bias units occur in both the input and hidden layers, that respectively serve as bias to hidden units and bias to output units.

The net input of the hidden units of a PUNN that contains a bias unit in the hidden layer is given by,

$$net_{y_j} = \prod_{i=1}^{I} z_i^{v_{ji}} + v_{j,I+1} \cdot z_{I+1} \tag{A.9}$$

The $(I+1)^{th}$ unit represents the bias unit to each hidden unit; v_{ji} is the weight between the i^{th} input and j^{th} hidden units; z_i is the value of the i^{th} input unit.

Weights are updated according to the following equations:

$$w_{kj}(t) = \Delta w_{kj}(t) + \alpha \cdot w_{kj}(t-1) \tag{A.10}$$
$$v_{ji}(t) = \Delta v_{ji}(t) + \alpha \cdot v_{ji}(t-1) \tag{A.11}$$

where α is the momentum, w_{kj} is the weight between the j^{th} hidden unit, Y_j, and k^{th} output unit, O_k and v_{ji} is the weight between the i^{th} input unit, Z_i, and j^{th} hidden unit, Y_j.

In the remainder of this appendix the equations for calculating $\Delta w_{kj}(t)$ and $\Delta v_{ji}(t)$ are derived. For notational convenience, the reference to time, t is omitted.

The error with respect to weight v_{ji} is calculated, applying the chain rule of differentiation,

$$\begin{aligned}
\frac{\partial E}{\partial v_{ji}} &= \frac{\partial E}{\partial net_{y_j}} \cdot \frac{\partial net_{y_j}}{\partial v_{ji}} \\
&= (\sum_{k=1}^{K} \frac{\partial E}{\partial net_{o_k}} \cdot \frac{\partial net_{o_k}}{\partial net_{y_j}}) \cdot \frac{\partial net_{y_j}}{\partial v_{ji}} \\
&= \sum_{k=1}^{K} \frac{\partial E}{\partial net_{o_k}} \cdot \frac{\partial net_{o_k}}{\partial y_j} \cdot \frac{\partial y_j}{\partial net_{y_j}} \cdot \frac{\partial net_{y_j}}{\partial v_{ji}} \qquad \text{(A.12)}
\end{aligned}$$

Now define,

$$\delta_{o_k} = -\frac{\partial E}{\partial net_{o_k}} \qquad \text{(A.13)}$$

Substitution of (A.13), (A.5) and (A.7) in (A.12) yields,

$$\begin{aligned}
\frac{\partial E}{\partial v_{ji}} &= \sum_{k=1}^{K} -\delta_{o_k} \cdot \frac{\partial(\sum_{j=1}^{J+1} w_{kj} y_j)}{\partial y_j} \cdot \frac{\partial(f(net_{y_j}))}{\partial net_{y_j}} \cdot \frac{\partial y_j}{\partial v_{ji}} \\
&= -\sum_{k=1}^{K} \delta_{o_k} \cdot w_{kj} \cdot f'(net_{y_j}) \cdot \frac{\partial y_j}{\partial v_{ji}} \\
&= -\sum_{k=1}^{K} \delta_{o_k} \cdot w_{kj} \cdot \frac{\partial y_j}{\partial v_{ji}} \qquad \text{(A.14)}
\end{aligned}$$

The output of hidden unit Y_j, is calculated next, where $v_{j,I+1}$ is the bias to Y_j and z_{I+1} refers to the bias unit with a constant value of -1.

Substitution of (A.9) in (A.7), results in,

$$\begin{aligned}
y_j &= \prod_{i=1}^{I} z_i^{v_{ji}} + z_{I+1} \cdot v_{j,I+1} \\
&= e^{\ln(\prod_{i=1}^{I} z_i^{v_{ji}})} + z_{I+1} \cdot v_{j,I+1} \\
&= e^{\sum_{i=1}^{I} v_{ji} \ln z_i} + z_{I+1} \cdot v_{j,I+1} \qquad \text{(A.15)}
\end{aligned}$$

If $z_i < 0$, then z_i can be written as the complex number $z_i = \imath^2 |z_i|$ which, substituted in equation (A.15), yields

$$y_j = e^{\sum_{i=1}^{I} v_{ji} \ln|z_i|} \cdot e^{\sum_{i=1}^{I} v_{ji} \ln \imath^2} + z_{I+1} \cdot v_{j,I+1} \qquad \text{(A.16)}$$

Let $c = 0 + \imath = a + b\imath$ be a complex number representing $\imath$. Then,

$$\ln c = \ln re^{\imath\theta} = \ln r + \imath\theta + 2\pi k\imath \tag{A.17}$$

where $r = \sqrt{a^2 + b^2} = 1$.

Considering only the main argument, $arg(c)$, $k = 0$, which implies that $2\pi k\imath = 0$. Also, is $\ln r = 0$, if $r = 1$. Furthermore $\theta = \frac{\pi}{2}$ for $\imath = (0, 1)$. Therefore, $\imath\theta = \imath\frac{\pi}{2}$, which simplifies equation (A.17) to $\ln c = \imath\frac{\pi}{2}$, and consequently,

$$\ln \imath^2 = \imath\pi \tag{A.18}$$

Substitution of (A.18) in (A.16) yields

$$\begin{aligned} y_j &= e^{\sum_{i=1}^{I} v_{ji} \ln|z_i|} \cdot e^{\sum_{i=1}^{I} v_{ji}\pi\imath} + z_{I+1} \cdot v_{j,I+1} \\ &= e^{\sum_{i=1}^{I} v_{ji} \ln|z_i|}(\cos(\pi \sum_{i=1}^{I} v_{ji}) + \imath \cdot \sin(\pi \sum_{i=1}^{I} v_{ji})) + z_{I+1} \cdot v_{j,I+1} \end{aligned} \tag{A.19}$$

Omitting the imaginary part, which is allowed since its inclusion did not result in any substantial improvement as reported by Durbin *et al* [Durbin *et al* 1989], reduces (A.19) to

$$y_j = e^{\sum_{i=1}^{I} v_{ji} \ln|z_i|} \cdot \cos(\pi \sum_{i=1}^{I} v_{ji}) + z_{I+1} \cdot v_{j,I+1} \tag{A.20}$$

Let

$$\rho = \sum_{i=1}^{I} v_{ji} \ln|z_i| \tag{A.21}$$

and

$$\phi = \sum_{i=1}^{I} v_{ji}\mathcal{I}_i \tag{A.22}$$

where

$$\mathcal{I}_i = \begin{cases} 0 & \text{if } z_i \geq 0 \\ 1 & \text{if } z_i < 0 \end{cases} \tag{A.23}$$

Then equation (A.20) becomes,

$$y_j = e^{\rho} \cdot \cos(\pi\phi) + z_{I+1} \cdot v_{j,I+1} \tag{A.24}$$

Now, applying differentiation w.r.t v_{ji} in equation (A.24),

$$\frac{\partial y_j}{\partial v_{ji}} = e^{\rho}\frac{\partial \rho}{\partial v_{ji}} \cdot \cos(\pi\phi) + \frac{\partial \cos(\pi\phi)}{\partial v_{ji}} \cdot e^{\rho} + \frac{\partial z_{I+1} \cdot v_{j,I+1}}{\partial v_{ji}} \tag{A.25}$$

$$\frac{\partial y_j}{\partial v_{ji}} = \begin{cases} e^{\rho} \cdot \ln|z_i| \cdot \cos(\pi\phi) - \pi\mathcal{I}_i \cdot \sin(\pi\phi) \cdot e^{\rho} & \text{if } i < I+1 \\ z_{I+1} & \text{if } i = I+1 \end{cases} \tag{A.26}$$

Substitution of (A.26) in (A.14), results in,

$$\frac{\partial E}{\partial v_{ji}} = \begin{cases} -\sum_{k=1}^{K} \delta_{o_k} \cdot w_{kj} \cdot e^{\rho}(\ln|z_i| \cdot \cos(\pi\phi) - \pi\mathcal{I}_i \cdot \sin(\pi\phi)) & \text{if } i < I+1 \\ -\sum_{k=1}^{K} \delta_{o_k} \cdot w_{kj} \cdot z_{I+1} & \text{if } i = I+1 \end{cases} \tag{A.27}$$

The changes to input-to-hidden weights are calculated as,

$$\Delta v_{ji} = -\eta \cdot \frac{\partial E}{\partial v_{ji}} \tag{A.28}$$

Substitution of (A.27) in (A.28) yields,

$$\Delta v_{ji} = \begin{cases} \eta \cdot \sum_{k=1}^{K} \delta_{o_k} \cdot w_{kj} \cdot e^{\rho} \cdot (\ln|z_i| \cdot \cos(\pi\phi) - \pi\mathcal{I}_i \cdot \sin(\pi\phi)) & \text{if } i < I+1 \\ \eta \cdot \sum_{k=1}^{K} \delta_{o_k} \cdot w_{kj} \cdot z_{I+1} & \text{if } i = I+1 \end{cases} \tag{A.29}$$

The error at the hidden layer, δ_{y_j} is now defined as ,

$$\delta_{y_j} = -\frac{\partial E}{\partial net_{y_j}} \tag{A.30}$$

$$= -\frac{\partial E}{\partial y_j} \cdot \frac{\partial y_j}{\partial net_{y_j}} \tag{A.31}$$

$$= -\frac{\partial E}{\partial y_j} \cdot f'(net_{y_j})$$

$$= -\frac{\partial E}{\partial y_j} \tag{A.32}$$

Next, $\frac{\partial E}{\partial y_j}$, is calculated applying the chain rule for differentiation,

$$\begin{aligned}
\frac{\partial E}{\partial y_j} &= \sum_{k=1}^{K}\left(\frac{\partial E}{\partial o_k}\cdot\frac{\partial o_k}{\partial y_j}\right) \\
&= \sum_{k=1}^{K}\left(\frac{\partial E}{\partial o_k}\cdot\frac{\partial o_k}{\partial net_{o_k}}\cdot\frac{\partial net_{o_k}}{\partial y_j}\right) \qquad \text{(A.33)} \\
&= \sum_{k=1}^{K}\left(\frac{\partial(\frac{1}{2}\sum_{k=1}^{K}(t_k-o_k)^2)}{\partial o_k}\cdot\frac{\partial o_k}{\partial net_{o_k}}\cdot\frac{\partial(\sum_{j=1}^{J+1}w_{kj}y_j)}{\partial y_j}\right) \\
&= \sum_{k=1}^{K}(-(t_k-o_k)\cdot f'(net_{o_k})\cdot w_{kj}) \\
&= -\sum_{k=1}^{K}(t_k-o_k)\cdot w_{kj} \qquad \text{(A.34)}
\end{aligned}$$

where (A.2) and (A.5) have been substituted in equation (A.33).

Substitution of (A.34) in (A.32) results in,

$$\delta_{y_j} = \sum_{k=1}^{K}(t_k-o_k)\cdot w_{kj} \qquad \text{(A.35)}$$

The equation above reduces (A.35) to,

$$\delta_{y_j} = \sum_{k=1}^{K}\delta_{o_k}\cdot w_{kj} \qquad \text{(A.36)}$$

Substitution of (A.36) in (A.29) results in,

$$\Delta v_{ji} = \eta\cdot\delta_{y_j}\cdot D_{ji} \qquad \text{(A.37)}$$

where D_{ji} is defined as,

$$D_{ji} = \begin{cases} e^{\rho}\cdot(\ln|z_i|\cdot\cos(\pi\phi)-\pi\mathcal{I}_i\cdot\sin(\pi\phi)) & \text{if } i < I+1 \\ z_{I+1} & \text{if } i = I+1 \end{cases} \qquad \text{(A.38)}$$

The error with respect to weight w_{kj} is calculated in the same way as for summation multilayer networks using gradient descent, i.e.

$$\Delta w_{kj} = -\eta\cdot\frac{\partial E}{\partial w_{kj}}$$

$$
\begin{aligned}
&= -\eta \cdot \frac{\partial E}{\partial o_k} \cdot \frac{\partial o_k}{\partial w_{kj}} \\
&= -\eta \cdot \frac{\partial}{\partial o_k}(\frac{1}{2}\sum_{k=1}^{K}(t_k - o_k)^2) \cdot \frac{\partial o_k}{\partial w_{kj}} \\
&= -\eta \cdot (-(t_k - o_k) \cdot \frac{\partial}{\partial w_{kj}}(\sum_{j=1}^{J+1} w_{kj} y_j)) \\
&= \eta \cdot (t_k - o_k) \cdot y_j
\end{aligned} \tag{A.39}
$$

Define the error that needs to be back-propagated as $\delta_{o_k} = -\frac{\partial E}{\partial net_{o_k}}$.

Then,

$$
\begin{aligned}
\delta_{o_k} &= -\frac{\partial_E}{\partial net_{o_k}} \\
&= -\frac{\partial E}{\partial o_k} \cdot \frac{\partial o_k}{net_{o_k}} \\
&= -\frac{\partial}{\partial o_k}(\frac{1}{2}\sum_{k=1}^{K}(t_k - o_k)^2) \cdot \frac{\partial o_k}{\partial net_{o_k}} \\
&= -(-(t_k - o_k) \cdot f'(net_{o_k})) \\
&= (t_k - o_k)
\end{aligned} \tag{A.40}
$$

since for linear activation, $f'(net_{o_k}) = 1$. Substitution of (A.40) in (A.39) yields,

$$
\Delta w_{kj} = \eta \cdot \delta_{o_k} \cdot y_j \tag{A.41}
$$

A.2 Learning rules for PUNN using a distortion unit

In this section the learning equations for a PUNN using a distortion unit are derived. In the case where the bias unit is replaced by a distortion unit in the hidden layer, only the equations influencing (A.29) need to be modified. Thus, equation (A.9) becomes,

$$
net_{y_j} = \prod_{i=1}^{I+1} v_{ji} z_i \tag{A.42}
$$

The $(I+1)^{th}$ input now represents the distortion to each hidden unit (refer to section 3.7.3 on page 72 for a discussion on the distortion unit). The input to the distortion unit is -1, i.e. $z_{I+1} = -1$.

Equation (A.15) now becomes,

$$y_j = e^{\sum_{i=1}^{I+1} v_{ji} \ln |z_i|} \tag{A.43}$$

To include the distortion unit in the product, equations (A.20), (A.21) and (A.22) become,

$$y_j = e^{\sum_{i=1}^{I+1} v_{ji} \ln|z_i|} \cdot \cos(\pi \sum_{i=1}^{I+1} v_{ji}) \tag{A.44}$$

which can be written as,

$$y_j = e^{\rho} \cdot \cos(\pi\phi) \tag{A.45}$$

where

$$\rho = \sum_{i=1}^{I+1} v_{ji} \ln |z_i| \tag{A.46}$$

$$\phi = \sum_{i=1}^{I+1} v_{ji} \mathcal{I}_i \tag{A.47}$$

where

$$\mathcal{I}_i = \begin{cases} 0 & \text{if } z_i \geq 0 \\ 1 & \text{if } z_i < 0 \end{cases} \tag{A.48}$$

Equations (A.29) and (A.38) become,

$$\Delta v_{ji} = \eta \cdot \sum_{k=1}^{K} \delta_{o_k} \cdot w_{kj} \cdot e^{\rho} \cdot (\ln |z_i| \cdot \cos(\pi\phi) - \pi \mathcal{I}_i \cdot \sin(\pi\phi)) \tag{A.49}$$

$$D_{ji} = e^{\rho} \cdot (\ln |z_i| \cdot \cos(\pi\phi) - \pi \mathcal{I}_i \cdot \sin(\pi\phi)) \tag{A.50}$$

where $i \leq I+1$ in equations (A.49) and (A.50).

The weight adjustment for weights between the hidden and output layer remains the same as the adjustment for PUNNs with a bias unit, i.e.

$$\Delta w_{kj} = \eta \cdot \delta_{o_k} \cdot y_j \tag{A.51}$$

This concludes the discussion on learning rules for PUNNs that contain either a bias or a distortion unit.

Appendix B

Publications from this thesis

A Ismail, AP Engelbrecht, *Training Product Units in Feedforward Neural Networks using Particle Swarm Optimization,* In: Development and Practice of Artificial Intelligence Techniques, V Bajic, D Sha (eds), Proceedings of the International Conference on Artificial Intelligence, Durban, South Africa, pp 36-40, 1999.

AP Engelbrecht, A Ismail, *Training Product Unit Neural Networks,* Stability and Control: Theory and Applications, Vol 2, No 1/2, pp 59-74, 1999.

A Ismail, AP Engelbrecht, *Global Optimization Algorithms for Training Product Unit Neural Networks,* IEEE International Joint Conference on Neural Networks, 24-27 July 2000, Como Italy, paper 032.

A Ismail, AP Engelbrecht, *Pruning Product Unit Neural Networks,* submitted to IEEE World Congress on Computational Intelligence, 2002.

A Ismail, AP Engelbrecht, *Improved Product Neural Networks,* submitted to IEEE World Congress on Computational Intelligence, 2002.

Appendix C

Symbols and notation

Symbols	Meaning
ANN	artificial neural network
BP	back-propagation by gradient descent
FLN	functional link network
FNN	feed-forward neural network
GA	genetic algorithm
LFOP	leapfrog optimization algorithm
NN	neural network
PSN	pi-sigma network
PSO	particle swarm optimization
PSO:PUs	product unit using product units
PSO:SUs	product unit using summation units
PU	product unit
PUNN	product unit neural network
RNN	recurrent neural network
SU	summation unit
SUNN	summation unit neural network

Symbols	Meaning
$\vec{v}_p$	the currents velocity of particle p
$\vec{x}_p = (x_{p,1}, x_{p,2}, ..., x_{p,D})$	the current position of particle p
$BEST_p$	the current best fitness achieved by particle p
$\overrightarrow{BESTx_p}$	the position that produced the best fitness value of the p^{th} particle
$GBEST$	the index of the best particle among all the particles in the population
δ_{y_j}	the error at the hidden layer
δ_{o_k}	the error at the output layer
z_i	i^{th} input value
Z_i	i^{th} input unit
y_j	activation of j^{th} hidden unit
Y_j	j^{th} hidden unit
o_k	activation of k^{th} output unit
O_k	k^{th} output unit
v_{ji}	weight between i^{th} input unit and j^{th} hidden unit
w_{kj}	weight between j^{th} hidden unit and k^{th} output unit
$f(net_{o_k})$	the activation for the k^{th} output unit
$f(net_{y_j})$	the activation for the j^{th} hidden unit
net_{o_k}	the net input for the k^{th} output unit
net_{y_j}	the net input for the j^{th} hidden unit
$\pm[2.0, 5.0]$	interval $[-5.0, -2.0]$ and interval $[2.0, 5.0]$