Bibliography

[1] Al-kazemi, B., and Moham, C.K.: *Multi-phase Discrete Particle Swarm optimization* In: Proceedings of the International Workshop on Frontiers in Evolutionary Algorithms, pp. 622-625 (2002)

[2] Antonov, I.A. and Saleev, V.M.: *An economic method of computing LP_τ-sequences* Zh. vychisl. Mat. mat. **19** pp. 243-245. English translation: U.S.S.R. Comput. Math. Phys. 19 pp. 252-256. (1979)

[3] Bäck, T., Fogel, D.B., and Michalewicz, Z, eds. *Evolutionary Computation 2. 15.* Institute of Physics Publishers. (2000)

[4] Beasley, D., Bull, D.R., and Martin, R.R.: *A Sequential Niche Technique for Multimodal Function Optimization.* Evolutionary Computation, **1**(2): 101-125 (1993)

[5] Bird, S. and Li, X.: *Adaptively choosing niching parameters in a PSO* In: Proceeding of the Genetic and Evolutionary Computation Conference (GECCO 2006). pp. 3-9 Seattle, Washington, USA. (2006)

[6] Blackwell, T.: *Particle Swarm Optimization in Dynamic Environments* In: Evolutionary Computation in Dynamic and Uncertain Environments. Vol 51/2007 Springer Berlin/Heidelberg pp. 29-49 (2007)

[7] Blackwell, T.M., and Bentley, P.:*Don't Push Me! Collision-Avoiding Swarms* In: Proceedings of the IEEE Congress on Evolutionary Computation volume 2, PP. 1691-1696 (2002)

[8] Blackwell, T.M. and Bentley, P.J.: *Dynamic search with charged swarms* In: Genetic and Evolutionary Computation Conference. W.B.L. et al., Ed. Morgan Kaufmann, 19-26 (2000)

[9] Blackwell, T., Branke, J.: *Multi-Swarm Optimization in Dynamic Environments* In: Raidl, G.R. (ed.): Applications of Evolutionary Computation, Vol. 1281. Springer 489-500 (2004)

[10] Blackwell, T. and Branke, J.: *Multi-Swarms, Exclusion, and Anti-Convergence in Dynamic Environments* In: IEEE Transactions on Evolutionary Computation, **10**(4): 459-472 (August, 2006)

[11] Bratley, P. and Fox, B.L.:*Algorithm 659: Implementing Sobol's quasirandom sequence generator* ACM Trans. Math. Softw. **14**:88-100 (1988)

[12] Brits, R., Engelbrecht, A.P., and Van den Bergh, F.: *Solving systems of unconstrained equations using particle swarm optimizers* in: Proceedings of the IEEE Conference on Systems, Man and Cybernetics, pp. 102-107 (2002)

[13] Brits, R., Engelbrecht, A.P., and Van den Bergh, F.: *A niching particle swarm optimizer* In: Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning. pp. 692-696 Singapore (November, 2002)

[14] Brits, R.: *Niching strategies for particle swarm optimization, MSc Thesis*, University of Pretoria (2002)

[15] Brits, R., Engelbrecht, A.P., and Van den Bergh, F.: *Scalability of NichePSO* In: Proceedings of the IEEE Swarm Intelligence Symposium, pp. 228-234, (Indianapolis, USA), (2003)

[16] Brits, R., Engelbrecht, A.P. and Van den Bergh, F.: *Locating multiple optima using particle swarm optimization* Applied Mathematics and Computation, **189**(2): 1859-1883 (June 2007)

[17] Carlisle, A., Dozier, G.: *Adapting Particle Swarm Optimization to Dynamic Environments* In: Proceedings of the International Conference on Artificial Intelligence. Las Vegas Nevada USA pp. 429-434 (2000)

[18] Carlisle, A., Dozier, G.: *Tracking Changing Extrema with Particle Swarm Optimizer* In: Proceedings of the Fifth Biennial World Automation Congress. pp. 265-270 (2002)

[19] Clerc, M.: *The Swarm & the Queen. Towards a Deterministic and Adaptive Particle Swarm Optimization* In: Proceedings of the Congress on Evolutionary Computation, pp. 1951-1957 (Washington DC, USA) IEEE Service Center, Piscataway, NJ (1999)

[20] Clerc, M. and Kennedy, J.: *The Particle Swarm - Explosion, Stability, and Convergence in a Multidimensional Complex Space*, IEEE Transactions on Evolutionary Computation, 6(1): pp. 58-73 (2002)

[21] Cramer, N.L.: *A representation for the Adaptive Generation of Sequential Programs* In: Proceedings of an International Conference on Genetic Algorithms and the Applications, Grefenstette, J.J. (ed.), Carnegie Mellon University (1985)

[22] Darwin, C.: *The Origin of Species*, P. F. Collier & Son (1909)

[23] Deb, K., and Goldberg, D.E.: *Natural Frequency Calculation using Genetic Algorithms.* In: Sathya V. Hanagud, Manohar P. Kamat, and Charles E. Ueng, editors, Proceedings of the 16th Southeastern Conference on Theoretical and Applied Mechanics, pp. 94-101, Atlanta, GA, (1990). The College of Engineering, Georgia Institute of Technology.

[24] De Jong, K.: *An analysis of the behavior of a class of genetic adaptive systems* Doctoral dissertation, University if Michigan, (1975)

[25] De Jong, K.A., Morrison, R.W.: *A Test Problem Generator for Non-Stationary Environment* In: Proceedings of the Congress on Evolutionary Computation. IEEE Press. pp. 2047-2053 (1999)

[26] Droste, S., Jansen, T. and Wegener, I.: *Perhaps not a free lunch but at least a free appetizer*

[27] Durham, W.: *Co-Evolution: Genes, Culture and Human Diversity*, Stanford University Press (1994)

[28] Eberhart, R.C., and Kennedy, J.: *A New Optimizer Using Particle Swarm Theory* Sixth International Symposium on Micro Machine and Human Science, pp.39-43 (Nagoya, Japan) IEEE Service Center (1995).

[29] Eberhart, R.C., and Shi, Y.: *Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization* In: Proceedings of the Congress on Evolutionary Computation, pp. 84-89 (San Diego, USA) IEEE Service Center, Piscataway, NJ (2000)

[30] Eberhart, R.C, and Shi, Y.: *Tracking and Optimizing Dynamic Systems with Particle Swarms* In: Proceedings of the 2001 Congress on Evolutionary Computation (CEC2001). pp. 94-100 (2001)

[31] Ellis, R., and Gulick, D.: *Calculus with Analytical Geometry* 5th ed., Saunders College Publishing (1994)

[32] Engelbrecht, A.P.: *Computational Intelligence - An Introduction*, John Wiley & Sons Ltd (2002)

[33] Engelbrecht, A.P.: *Fundamentals of Computational Swarm Intelligence*, John Wiley & Sons Ltd (2005)

[34] Engelbrecht, A.P., Masiye, B.S., and Pamparà, G.: *Niching Ability of Basic Particle Swarm Algorithms* (2005)

[35] Fogel, D.B.: *Evolutionary Computation*, 2nd ed., IEEE Press, Piscataway, NJ (2000)

[36] Fogel, L.J., Owens, A.J. and Walsh, M.J.: *Artificial Intelligence Through Simulated Evolution* Wiley (1966)

[37] Forsyth, D.B.: *BEAGLE A Darwinian Approach to Pattern Recognition* Kybernetes, **10**:159-166 (1981)

[38] Glover, F.: *Tabu Search - Part I*, ORSA Journal on Computing **1**(3):190-206 (1989)

[39] Goldberg, D.E., Deb, K., and Horn, J.: *Massive Multimodality, Deception, and Genetic Algorithms* In: R. Männer and B. Manderick (Eds.): Parallel Problem Solving from Nature, 2, 37-46 North-Holland (1992)

[40] Goldberg, D.E., and Richardson, J.: *Genetic Algorithms with Sharing for Multimodal Function Optimization* In: Proceedings of the 2nd International Conference on Genetic Algorithms, pp. 41-49 (1997)

[41] Hendtlass, T.: *WoSP: A Multi-Optima Particle Swarm Algorithm*, In: Proceedings of the IEEE Congress on Evolutionary Computation. pp. 727-734 Edinburgh, UK. (2005)

[42] Hoffman, R., Minkin, V.I., and Carpenter, B.K.: *Ockham's razor and Chemistry*. International Journal for the Philosophy of Chemistry. 3. 3-28 (1997)

[43] Holland, J.H.: *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor (1975)

[44] Hoos, H.H.: *Stochastic Local Search - Methods, Models, Applications, PhD Thesis*, University of Darmstadt (1998).

[45] Hu, X. and Eberhart, R.C.: *Tracking Dynamic Systems with PSO: Where's the Cheese?*, In: Proceedings of the Workshop on Particle Swarm Optimization. pp. 80-81 (2001)

[46] Hu, X. and Eberhart, R.C.: *Adaptive Particle Swarm Optimization: Detection and Response to Dynamic Systems* In: Proceedings of the IEEE Congress on Evolutionary Computation. Vol. 2 pp. 1666-1670, (May, 2002)

[47] Joe, S. and Kuo, F.Y.: *Remark on Algorithm 659: Implementing Sobol's quasirandom sequence generator* ACM Trans. Math. Softw. **29**:49-57 (2003)

[48] Kennedy, J., and Eberhart, R.C.: *Particle Swarm Optimization* In: Proceedings of the IEEE International Conference on Neural Networks (Perth, Australia), IEEE Service Center, Piscataway, NJ, IV: 1942-1948 (1995).

[49] Kennedy, J.: *The Particle Swarm: Social Adaptation of Knowledge* In: Proceedings of the International Conference on Evolutionary Computation, pp. 303-308, Indianapolis, IN, USA (1997)

[50] Kennedy, J.: *Why Does it Need Velocity?* In: Proceedings of the IEEE Swarm Intelligence Symposium, Pasadena CA (2005)

[51] Kennedy, J.: *Small Worlds and Mega-Minds: Effects of Neighborhood Topology on Particle Swarm Performance*, In: Proceedings of the Congress on Evolutionary Computation, pp. 1931-1938 (Washington DC, USA) IEEE Service Center, Piscataway, NJ (1999)

[52] Kennedy, J.: *Stereotyping: Improving Particle Swarm Performance with Cluster Analysis* In: Proceedings of the IEEE Congress on Evolutionary Compuation vol. 2 pp. 1507-1512 (2000)

[53] Kennedy, J., and Mendes, R.: *Population Structure and Particle Swarm Performance* In: Proceedings of the IEEE Congress on Evolutionary Computation, pp. 1671-1676 IEEE Press (2002)

[54] Kirkpatrick, S., Gelatt, C.D., and Vecchi, M.P.: *Optimization by simulated annealing*, Science, **220**4598: 671-680 (1983).

[55] Koza, J.R.: *Genetic Programming: On the programming of Computers by means of Natural Selection* MIT Press (1992)

[56] Li, J.P., Balazs, M.E., Parks, G.T., and Clarkson, P.J.: *A species conserving genetic algorithm for multi-modal function optimization* In: Evolutionary Computation, **10**(3):207-234 (2002)

[57] Li, X.: *Adaptively Choosing Neighbourhood Bests using Species in a Particle Swarm Optimizer for Multimodal Function Optimization.* In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2004). pp. 105-116 (2004)

[58] Li, X.: *A multimodal particle swarm optimzer based on fitness Euclidian-distance ratio* In: Proceeding of the Genetic and Evolutionary Computation Conference (GECCO 2007). pp. 78-85 London, England, UK. (2007)

[59] Li, X., Branke, J. and Blackwell, T.: *Particle Swarm and Adaptation in a Dynamic Environment* In: GECCO06, Seattle, Washington, USA (2006)

[60] Løvbjerg, M., Rasmussen, T.K., Krink,T.: *Hybrid particle swarm optimizer with subpopulations and breeding* In: Proceedings of the Genetic and Evolutionary Computation Conference, vol 1, San Francisco USA pp. 469-476 (July 2001)

[61] Mahfoud,S.: *Crowding and preselection revisited* In: Reinhard Männer and Bernard Manderick, editors, Parallel Problem Solving from Nature 2, pp. 27-37, (1992)

[62] Malan, K., and Engelbrecht, A.P.:*Algorithm Comparisons and the Significance of Population Size* Proceedings of the 2008 IEEE World Congress on Computational Intelligence (WCCI 2008). Hong Kong. (2008) To be published.

[63] Marais, E.N.: *The Soul of the White Ant*, (1937).

[64] Mathews, J.H.: *Numerical Methods for Mathematics, Science and Engineering*, Prentice Hall, Inc. 2nd ed. (1992).

[65] Mendes, R., Kennedy, J., and Neves, J.: *Watch Thy Neighbour or How the Swarm can Learn from its Environment* In: Proceedings of the IEEE Swarm Intelligence Symposium, pp. 88-94 (2003)

[66] Millonas, M.M.: *Swarms, Phase Transitions, and Collective Intelligence* In: C.G.Langton, Ed., Artificial Life III. Addison Wesley, Reading, MA. (1994)

[67] Nelder, J.A. and Mead, R.: *A simplex method for function minimization* In: The Computer Journal, volume 7, pp. 308-313 (1965)

[68] Ozcan, E., and Mohan, C.: *Analysis of a simple particle swarm optimization system* In: Intelligent Engineering Systems Through Artificial Neural Networks (ANNIE'98), Vol. 8 pp. 253-258 (1998)

[69] Ozcan, E., and Mohan, C.: *Particle Swarm Optimization: Surfing the Waves* In: Proceedings of the International Congress on Evolutionary Computation, pp.1939-1944 (Washington USA) (1999)

[70] Parrott, D., Li, X.: *A Particle Swarm Model for Tracking Multiple Peaks in a Dynamic Environment using Speciation.* In: Proceedings on the 2004 Congress of Evolutionary Computation (CEC2004) pp. 98-103 (2004)

[71] Parsopoulos, K.E., Plagianakos, V.P, Magoulas, G.D. and Vrahatis, M.N.: *Stretching Techniques for Obtaining Global Minimizers through Particle Swarm Optimization.* In: Proceedings of the Particle Swarm Optimization Workshop. Indianapolis USA pp. 22-29 (2001)

[72] Parsopoulos, K.E. and Vrahatis, M.N.: *Modification of the Particle Swarm Optimizer for Locating all the Global Minima.* In: Kurkova, V., Steele, N.C., Neruda, R. and Karny, M. (eds.): Artificial Neural Networks and Genetic Algorithms, 324-327 Springer (2001)

[73] Peer, E.S., Van den Bergh, F., and Engelbrecht, A.P.: *Using Neighbourhoods with the Guaranteed Convergence PSO* In: Proceedings of the IEEE Swarm Intelligence Symposium, pp. 235-242 IEEE Press (2003)

[74] Pétrowski, A.: *A clearing procedure as a niching method for genetic algorithms* In: Proceedings of the Third IEEE International Conference on Evolutionary Computation (ICEC'96), Piscataway, NJ, IEEE Press, pp. 798-803 (1996)

[75] Potts, W.K.: *The chorus-line hypothesis of coordination in avian flocks*, Nature **24**: 344-345 (1984)

[76] Press, W.H., Teukolsky, S.A., Vetterling, W.T., and Flannery, B.P.: *Numerical Recipes in C: the Art of Scientific Computing* Cambridge University Press, second edition (1992)

[77] Fowler, F.G., and Fowler, H.W.: *The Pocket Oxford Dictionary of Current English* 5th edition Oxford University Press, London (1969)

[78] Price, K.V.: *An Introduction to Differential Evolution*, In: Corne, D., Dorigo, M., and Glover, F. (eds.) New Ideas in Optimization, McGraw-Hill, London (1999)

[79] Rechenberg, I.: *Evolutionsstrategie: Optimierung Technischer Systeme nach Prinzipien der Biologischer Evolution.* Framman-Holzboog Verlag, Stuttgart (1973)

[80] Reynolds, C.W.: *Flocks, herds and schools: a distributed behavioural model*, Computer Graphics **21**(4): 25-34 (1987)

[81] Reynolds, R.G.: *Cultural algorithms* In: Corne, D., Dorigo, M., and Glover, F., editors, New Ideas in Optimization, p. 367, McGraw-Hill (1999)

[82] Schoeman, I.L., and Engelbrecht, A.P.: *Using Vector Operations to Identify Niches for Particle Swarm Optimization* Proceedings of the IEEE Conference on Cybernetics and Intelligent Systems. pp. 361-366 Singapore (2004)

[83] Schoeman, I.L., and Engelbrecht, A.P.: *A Parallel Vector-Based Particle Swarm Optimizer* Proceedings of the International Conference on Artificial Neural Networks and Genetic Algorithms. (ICANNGA2005) Coimbra Portugal, pp. 268-271 (2005)

[84] Schoeman, I.L., and Engelbrecht, A.P.: *Containing Particles inside Niches when Optimizing Multimodal Functions* Proceedings of SAICSIT2005. White River, South Africa pp. 78-85 (2005)

[85] Schoeman, I.L., and Engelbrecht, A.P.: *Niching for Dynamic Environments Using Particle Swarm Optimization* In: Proceedings of the Sixth International Conference on Simulated Evolution and Learning (SEAL'06). pp. Hefei, China, (October 2006)

[86] Schwefel, H-P.: *Evolutionsstrategie und Numerische Optimierung.* PhD Thesis, Technical University Berlin (1975)

[87] Sheskin, D.J.:
emphHandbook of Parametric and Nonparametric Statistical Procedures, Chapman & Hall/CRC (2007)

[88] Shi, Y., and Eberhart, R.C.: *A Modified Particle Swarm Optimizer* In: IEEE International Conference of Evolutionary Computation (Anchorage, Alaska) pp. 69-73 (1998)

[89] Shi, Y. and Eberhart, R.C.: *Parameter Selection in Particle Swarm Optimization* In: Evolutionary Programming VII: Proceedings of EP 98, pp. 591-600 (1998)

[90] Singh, G., and Deb, K.: *Comparison of Multi-Modal Optimization Algorithms Based on Evolutionary Algorithms* GECCO'06, (Seattle, USA) pp. 1305-1312 (2006)

[91] Smith, S.F.: *A learning System based on Genetic Adaptive Algorithms.* PhD Thesis, University of Pittsburgh (1980)

[92] Smith, R.E., Forrest, S., and Perelson, A.S.: *Searching for diverse, cooperative populations with genetic algorithms* TCGA Report No. 92002. The University of Alabama, Department of Engineering Mechanics. (1992)

[93] Snyman, J.A.: *Practical Mathematical Optimization*, Springer (2005) pp. 1-16.

[94] Storn, R., and Price, K.: *Differential Evolution - a Simple and Efficient Heuristic for Global Optimization over Continuous Spaces*, In: Journal of Global Optimization, **11**:341-359 (1997)

[95] Suganthan, P.N.: *Particle Swarm Optimizer with Neighbourhood Operator* In: Proceedings of the Congress on Evolutionary Computation, pp. 1958-1961 (Washington DC, USA) IEEE Service center, Piscataway NJ (1999)

[96] Thodberg, H.H.: *Improving Generalization of Neural Networks through Pruning.* International Journal of Neural Systems. 1(4), 317-326 (1991)

[97] Trelea, I.C.: *The Particle Swarm Optimization Algorithm: Convergence and Parameter Selection* Information Processing Letters, **85**(6):317-325 (2003)

[98] Van den Bergh, F.: *An Analysis of Particle Swarm Optimizers* PhD Thesis, University of Pretoria (2002).

[99] Van den Bergh, F. and Engelbrecht, A.P.: *A New Locally Convergent Particle Swarm Optimiser* In: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, (Hammamet, Tunisia), October 2002.

[100] Van den Bergh, F. and Engelbrecht, A.P.: *A Study of Particle Swarm Optimization Particle Trajectories*, Information Sciences **176**:937-971 (2006)

[101] Van den Bergh, F. and Engelbrecht, A.P.: *A Cooperative Approach to Particle Swarm Optimization* IEEE Transactions on Evolutionary Computation **8**(3):225-239 (2004)

[102] Veeramachaneni, K., Peram, T., Mohan, C., and Osadciw, L.A.: *Optimization Using Particle Swarms with Near Neighbour Interactions* In: Proceedings of the Genetic and Evolutionary Computation Conference, In: Lecture Notes in Computer Science **2723**:110-121. Springer-Verlag (2003)

[103] Wolpert, D.H., and Macready, W.G.: *No Free Lunch Theorems for Optimization*, IEEE Transactions on Evolutionary Computation **1**(1), pp. 67-82 (1997)

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Appendix A

Test Results

This appendix presents empirical results of tests that were run to determine specific settings used by the family of vector-based PSO algorithms. Section A.1 presents results of empirical tests to determine minimum sizes of small subswarms to prevent such swarms from stagnating. Section A.2 investigates the influence of interval sizes between calls to the merging procedure. This procedure is called several times while subswarms are updated in parallel in order to merge subswarms attempting to optimize the same optimal or suboptimal solution. Section A.3 presents results of empirical tests to find the minimum size of a subswarm capable of tracking a moving optimum. If the function landscape changes, positions of previous optima are retained and small subswarms are created around such optima. Fewer function evaluations are required resulting in a reduction in computational complexity.

A.1 Minimum subswarm sizes

To establish niche boundaries, the vector-based PSO calculates niche radii as described in section 5.4. A number of extra or false niches are formed adjacent to true niches. Some niches, especially the false niches, may contain subswarms consisting of only one or two particles. As explained in section 5.4, such subswarms may stagnate. To address this problem, these subswarms are extended to contain at least three particles.

A.1.1 Experimental procedure

A selection of two-dimensional functions with differing characteristics, namely the Himmelblau, Ursem F3 and Ackley functions, were used to investigate minimum subswarm sizes. Descriptions and illustrations of the functions were provided in section 6.2.3. The algorithms were run with minimal subswarm sizes of one, two, and three particles. The number of solutions that indicated optima was recorded as well as the number of extra or false solutions that did not indicate such positions. Each result is the average of 50 runs of the relevant algorithm. Results are presented for the parallel vector-based PSO as well as the enhanced parallel vector-based PSO.

Table A.1: Convergence of small subswarms using the parallel vector-based PSO

Function	Results					
	Subswarm size of 1		Subswarm size of 2		Subswarm size of 3	
	Success rate	Average # false solutions	Success rate	Average # false solutions	Success rate	Average # false solutions
Himmelblau	98%	5.6	99.5%	0.1	100%	0
Ursem F3	99%	5.7	100%	0.12	100%	0
Ackley (2-dim)	95.8%	2.7	98.7%	0.08	98.9%	0

A.1.2 Results and discussion

When subswarms stagnate, results show solutions that do not indicate optimal positions in the search space, since such subswarms do not converge on optimal positions. No such false solutions will result if subswarms keep moving and eventually merge to indicate true optimal positions. Thus, the number of such false solutions gives an indication of the ability of subswarms to keep moving.

Tables A.1 and A.2 show the average number of extra or false solutions for the parallel vector-based PSO and the enhanced parallel vector-based PSO. For each function and subswarm size combination that was tested, the success rate of the run (total number of solutions as a

Table A.2: Convergence of small subswarms using the enhanced parallel vector-based PSO

Function	Results					
	Subswarm size of 1		Subswarm size of 2		Subswarm size of 3	
	Success rate	Average # false solutions	Success rate	Average # false solutions	Success rate	Average # false solutions
Himmelblau	100%	0	100%	0	100%	0
Ursem F3	100%	0.8	100%	0.16	100%	0
Ackley	99.6%	0	99.6%	0	99%	0

percentage of the total number of possible solutions) and the average number of extra or false solutions were recorded. As result of the stochastic character of the VBPSO, small differences in success rates occurred. For each function, the average number of false solutions decreased until no false solutions were recorded when a subswarm contained a minimum of three particles. This lead to the conclusion that the subswarm sizes of at least three particles are sufficient to obtain good results. Note that the enhanced parallel vector-based PSO yielded fewer false solutions. This effect can be ascribed to the procedure where the particle position is updated (refer to section 5.3), which is the only difference between the two algorithms. The enhanced version tests each new position of a particle to ensure that the particle does not leave the niche. Thus, a subswarm is prevented from being diverted to, and absorbed by, a neighbouring subswarm, an effect causing a lower success rate. In addition, small subswarms are prevented from becoming still smaller, a situation where stagnation can occur more often.

The small selection of results presented here does not purport to represent an exhaustive analysis of the significance of choosing a minimum swarm size. False solutions may still be encountered, even with larger minimum swarm sizes. On the other hand, the improved performance of the enhanced version indicates that smaller minimum swarm sizes may suffice. However, keeping in mind that the introduction of another tunable parameter to indicate minimum swarm size is not an option, and too large swarm sizes increase computational complexity, the minimum swarm size was set to three for all subsequent experiments.

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

A.2 Merging intervals

The parallel vector-based PSO algorithms update subswarms in parallel, thereby differing from the earlier sequential version. During each iteration of the algorithm, all particles in all subswarms are updated. When the vector-based PSO initially identifies niches, a number of addtional or false niches are formed (see section 5.5). Subswarms in these niches are expected to converge on optimal or suboptimal solutions of adjacent true niches (containing an optimal solution), giving duplicate solutions. The parallel VBPSO merges these subswarms with subswarms in true niches to exclude duplicate solutions. If the merging procedure is called during the run, false niches are gradually absorbed by true niches, forming larger subswarms which converge more effectively. If the merging procedure is called only once at the end of the run, all subswarms have to be merged at once, which might be less effective. A selection of functions have been tested to observe the effect of different merging intervals. Descriptions and illustrations of the functions were provided in section 6.2. Only the enhanced parallel VBPSO was tested as the parallel and enhanced parallel algorithms use the same merging procedure. Averages of 50 runs have been reported for each setting. For each run the updating procedure is iterated 500 times, interspersed with a number of calls to the merging procedure.

Results presented in Table A.3 show that a small number of subswarms do not merge when the merging procedure is called once at the end of a run of 500 iterations. One extra call to the merging procedure (with differing intervals) improves the situation but some niches may still not merge. If the merging procedure is called more often, all niches merge. No significant difference in the success rates of the different functions was found, indicating that the algorithm is not sensitive to the exact size of the merging interval. In addition, the exact size of the interval does not exert any influence on the success of the merging process, provided that the merging procedure is called more than once during optimization and again at the end of the run. The calls are best spread evenly over the run. The vector-based PSO algorithms described in algorithms 10 and 11 calls the merging procedure at 10 equal intervals during the run. These intervals were chosen in order to trace the merging process and present it graphically. These results were presented in chapter 6.

Considering the above description, the size of the merging interval can rather be seen as a heuristic than a tunable parameter. Therefore, setting the size of the merging interval cannot be used as an argument that the principle of parsimony is violated.

Table A.3: Effect of merging intervals

Function	Results									
	Merge at end		Merge at interval 250		Merge at intervals 400, 100		Merge at 5 intervals of 100		Merge at 10 intervals of 50	
	Success rate	Average # extra solutions	Success rate	Average # extra solutions	Success rate	Average # extra solutions	Success rate	Average # extra solutions	Success rate	Average # extra solutions
Himmelblau (4 optima)	100%	0.2	100%	0	100%	0	100%	0	100%	0
Ursem F3 (4 optima)	100%	0.96	100%	0	100%	0	100%	0	100%	0
Ackley 2-dim (9 optima)	100%	1.44	99.8%	0.02	99.3%	0.06	99.8%	0	100%	0

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

A.3 Minimum swarm size for tracking optima

Section 7.6 presents an algorithm to track multiple optima in a dynamic environment. The initial stage locates optima using the VBPSO described in chapters 5 and 6. After each environment change, optima have to be tracked and new optimal positions located. To reduce computational complexity, only the optimal positions of the previous stage are retained, and small subswarms are created around those positions. This section presents tests designed to find a minimum swarm size capable of tracking optima in a dynamic environment.

A.3.1 Experimental procedure

Scenario 1 described in section 8.2 was used to observe the effect of different minimum swarm sizes on tracking capability. Three optima were tracked over six steps. For the sake of clarity, the positions of the peaks are listed in Table A.4. Experiments were conducted where the size of the small swarm created to track each optimum, were set to one (one particle at the position of each previous optimum), two, three and four. The average offline errors between the optimal positions found by the small swarms and the true optimal positions were calculated for each setting. These errors were compared to the average offline error for the initial stage of the algorithm (using a larger swarm). Errors of the same order of magnitude indicated a tracking ability of the small subswarms similar to that of the initial swarm.

Table A.4: Positions of 3 optima over 6 steps

Step	Peak 1				Peak 2				Peak 3			
x_1	x_2	H	R	x_1	x_2	H	R	x_1	x_2	H	R	
1	-0.6	-0.8	1	2	-0.5	0.3	0.6	2	0.5	0	0.8	2
2	-0.4	-0.8	1	2	-0.3	0.4	0.6	2	0.4	0	0.8	2
3	-0.2	-0.8	1	2	-0.1	0.5	0.6	2	0.3	0	0.8	2
4	0	-0.8	1	2	0.1	0.6	0.6	2	0.2	0	0.8	2
5	0.2	-0.8	1	2	0.3	0.7	0.6	2	0.1	0	0.8	2
6	0.4	-0.8	1	2	0.5	0.8	0.6	2	0	0	0.8	2

A.3.2 Results and discussion

Results of experiments conducted to test the tracking ability of small subswarms are presented in Table A.5. Averages of 50 experiments are listed. No significant difference was observed between the average offline errors of the initial swarm (consisting of 30 particles) for the different experiments. However, for stages two to six of the algorithm, the offline error decreased with an increase in the sizes of the small subswarms. For a subswarm of three particles, the offline error was marginally larger than that of the initial swarm. For a subswarm of four particles, the offline error was marginally smaller than that of the initial swarm. Therefore, given that the severity did not exceed the niche radius (refer to section 8.4), a subswarm size of three or four particles was sufficient to track moving optima in a dynamic environment. For the dynamic VBPSO, it was decided to create subswarms at previous optimal positions where each subswarm consisted of a particle at one of those positions as well as three additional particles.

Table A.5: Tracking ability of small subswarms

Subswarm size (particles)	Success rate	Offline error	
		Initial swarm	Small subswarms
1	100%	5.17E-18 ± 1.77E-18	3.48E-03 ± 1.33E-03
2	100%	2.40E-17 ± 1.17E-17	5.34E-06 ± 1.90E-06
3	100%	7.57E-18 ± 2.61E-18	9.38E-17 ± 4.67E-17
4	100%	2.62E-17 ± 1.98E-17	1.09E-17 ± 1.62E-18

Appendix B

Derived publications

This appendix lists all papers that have been published or are currently under review, that led to, or are derived from the work presented in this thesis.

Schoeman, I.L., and Engelbrecht, A.P.: *Using vector operations to identify niches for particle swarm optimization.* In: Proceedings of the IEEE Conference on Cybernetics and Intelligent Systems. pp. 361-366 Singapore (2004)

Schoeman, I.L., and Engelbrecht, A.P.: *A parallel vector-based particle swarm optimizer.* In: Proceedings of the International Conference on Artificial Neural Networks and Genetic Algorithms. (ICANNGA2005) Coimbra Portugal, pp. 268-271 (2005)

Schoeman, I.L., and Engelbrecht, A.P.: *Containing particles inside niches when optimizing multimodal functions.* In: Proceedings of SAICSIT2005. White River, South Africa pp. 78-85 (2005)

Schoeman, I.L., and Engelbrecht, A.P.: *Niching for dynamic environments using particle swarm optimization.* In: Proceedings of the Sixth International Conference on Simulated Evolution and Learning (SEAL'06). Hefei, China, (October 2006)

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Schoeman, I.L., and Engelbrecht, A.P.: *A novel particle swarm niching technique based on extensive vector operations.* Natural Comuputing, Springer, 1567-7818 (print) 1572-9796 (online) (December 23, 2009)

Schoeman, I.L., and Engelbrecht, A.P.: *Scalability of the vector-based PSO.* In: Proceedings of the Congress of Evolutionary Computation (CEC2009) Trondheim, Norway, (May 2009)

Isabella Schoeman and Andries Engelbrecht: *Effect of Particle Initialization on the Performance of Particle Swarm Niching Algorithms.* Accepted as an extended abstract in ANTS 2010

Schoeman, I.L., and Engelbrecht, A.P.: *Tracking Multiple Optima in Dynamic Environments using Particle Swarm Optimization.* Under review.