Niching Ant Colony Optimisation

Daniel John Angus, BEng. BSci.

Complex Intelligent Systems Laboratory
Faculty of Information & Communication Technologies
Swinburne University of Technology
Melbourne, Australia

Submitted in partial fulfilment for the degree of
Doctor of Philosophy

July 2008

Abstract

Optimisation, in the mathematical sense, is the process of finding solutions to a problem such that one or many objectives are minimised or maximised. Optimisation problems are diverse in form, necessitating the need for many different optimisation algorithms. These algorithms can be defined in two categories: deterministic and non-deterministic algorithms. Deterministic algorithms usually have set execution schedules and are fairly exhaustive search methods. Non-deterministic algorithms use randomness and prove useful for problems where it may not be possible to execute a deterministic algorithm due to the size, or nature of the problem search space. In these cases a deterministic algorithm may take days or months to find an optimal solution, whereas a non-deterministic algorithm can usually find an approximate but still near-optimal solution in a matter of minutes or seconds.

Ant Colony Optimisation (ACO), a non-deterministic algorithm class, aims to mimic (and exploit) the behaviours of real ant colonies to solve real-world optimisation problems. ACO algorithms are a class of constructive heuristic algorithms, which build solutions to a given optimisation problem, one solution component at a time, according to a defined set of rules (heuristics), i.e. starting with an 'empty' solution add solution components until a complete solution is built. ACO algorithms are unique in this class by their use of past solutions in manipulating an artificial 'pheromone'. The pheromone being a measure associated to every unique solution component which reflects the estimated utility of this solution component. These pheromone values are used to bias solution construction by influencing the probability of a solution component being added to a growing solution based on the amount of pheromone it contains.

The Population-based Ant Colony Optimisation (PACO) algorithm is a recently developed ant-inspired algorithm which, unlike traditional ACO algorithms, maintains a finite population of solutions as well as pheromone information. It has been demonstrated to be an efficient optimisation algorithm when applied to a range of difficult single-objective, multi-objective and dynamic problem instances. In this thesis a review of existing PACO algorithms is offered and an identification of common features is used in the development of a Population-based ACO framework.

Using the new Population-based ACO framework, several new PACO algorithms imbued with a diversity preservation technique known as niching are defined. Niching has been used extensively in the field of Evolutionary Computation, but to the best knowledge of the author, has never been explicitly applied to an ACO algorithm per se. An empirical analysis of these novel

implementations is presented using a variety of single and multiple objective continuous function and combinatorial optimisation problems. These optimisation problems have been chosen since they demonstrate the advantages and disadvantages of adding niching to a PACO algorithm. To conclude, two of these new PACO algorithms are applied to a real-world optimisation problem.

Acknowledgments

A body of work such as this is never completed without a number of challenges and difficulties along the way. By far my greatest challenge was pursuing my studies without a scholarship for my first year. Although tricky, stressful and at times depressing this first year taught me many lessons on life, love and how fulfilling it can be to feed yourself for a week on $5. My ability to sense the free campus BBQ also became finely tuned! I wish to firstly thank those people who were there for me during this time who ensured that I continued to pursue my studies. Although I can't say it has been clear sailing from there, it has certainly been made easier thanks to the understanding and care of so many family members, friends and colleagues.

To my wife Katherine, your patience, care and understanding have made these last three years some of the most memorable in my life so far. You were with me from day one, and have stood beside me the whole way, for this I thank you.

Anne and Clive Angus, as my parents you have always encouraged me and given me the support I need. You have given me the strength and courage to pursue my dreams, for this I thank you. To Kate and Leigh Thomson, as my sister and brother in law you have always encouraged my studies and showed a genuine interest in my academic pursuits, thank you.

My colleagues at the Complex Intelligent Systems group, you have all helped me at some point and I thank you all for the many meaningful discussions shared over a beer. To my supervisor Prof. Tim Hendtlass, you inspired me from early in my undergraduate research days to pursue a PhD, and here I am now. You have played a most important role as my supervisor and mentor. Thank you for your trust, patience and understanding. To my co-supervisor Assoc. Prof. Marcus Randall your support is also appreciated. To my colleague and old room-mate Jason Brownlee, I have valued your input highly and you deserve an appropriate amount of thanks for your contribution. Your ideas about anything research related always seem to be light-years ahead of mine and I thank you for the positive contribution you have made to my research, and to my life in general.

Family and friends be they university colleagues past or present, high school chums, Rovers and the like, you have all been there at some point to offer encouragement, thank you all.

On a professional note I must thank Adam Deller for his help with Chapter 8 which was derived from his PhD work with the Centre for Astrophysics and Supercomputing at Swinburne University. I also acknowledge the Australian Government for providing me with an Australian Postgraduate Award for a partial component of my PhD program.

Declaration

I declare that this thesis contains no material that has been accepted for the award of any other degree or diploma and to the best of my knowledge contains no material previously published or written by another person except where due reference is made in the text of this thesis.

Daniel John Angus

Publications Arising from this Thesis

Angus, D. *Niching for Population-based Ant Colony Optimization*, Proceedings of the 2nd International IEEE Conference on e-Science and Grid Computing, Workshop on Biologically Inspired Optimisation Methods for Parallel and Distributed Architectures: Algorithms, Systems and Applications, 2006.

Angus, D. *Crowding Population-based Ant Colony Optimisation for the Multi-objective Travelling Salesman Problem*, Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making (MCDM 2007), 2007, pp. 333–340.

Angus, D. *Population-based Ant Colony Optimisation for Multi-objective Function Optimisation*, Proceedings of the 3rd Australian Conference on Artificial Life (ACAL2007), 2007, pp. 232–244.

Contents

List of Figures

List of Tables

Commonly Used Acronyms

ACO	Ant Colony Optimisation
ACOCD	Ant Colony Optimisation for Continuous Domains
ACS	Ant Colony Systems
AS	Ant Systems
CFO	Continuous Function Optimisation
CGA	Deterministic Crowding Genetic Algorithm
CPACO	Crowding Population-based Ant Colony Optimisation
DE	Differential Evolution
EC	Evolutionary Computation
EDA	Estimation of Distribution Algorithm
ES	Evolutionary Strategies
FSPACO	Fitness Sharing Population-based Ant Colony Optimisation
GA	Genetic Algorithm
GP	Genetic Programming
MMAS	$\mathcal{MAX} - \mathcal{MIN}$ Ant Systems
MOEA	Multiple Objective Evolutionary Algorithm
MOFO	Multiple Objective Function Optimisation
MOGA	Multiple Objective Genetic Algorithm
MOO	Multiple Objective Optimisation
MOTSP	Multiple Objective Travelling Salesman Problem
NPGA	Niched Pareto Genetic Algorithm
NSGA	Non-dominated Sorting Genetic Algorithm
NSGA-II	Non-dominated Sorting Genetic Algorithm (version 2)
PACO	Population-based Ant Colony Optimisation
PBIL	Population Based Incremental Learning
PCB	Printed Circuit Board
PMBGA	Probabilistic Model-based Genetic Algorithm
PSO	Particle Swarm Optimisation
QAP	Quadratic Assignment Problem
RBAS	Rank-based Ant Systems
RTS	Restricted Tournament Selection
RVGA	Real Value Genetic Algorithm
SA	Simulated Annealing
SNGA	Sequential Niching Genetic Algorithm
TSP	Travelling Salesman Problem
VEGA	Vector Evaluated Genetic Algorithm
VLBI	Very Long Baseline Interferometry