ANT COLONY OPTIMIZATION IN DYNAMIC ENVIRONMENTS

CHEN FEI HUANG

PERPUSTAKAAN UMMERSITI MALAYSIA SABAH

THESIS SUBMITTED IN FULFILLMENT FOR THE DEGREE OF MASTER OF SCIENCE IN SOFTWARE ENGINEERING

SCHOOL OF ENGINEERING AND INFORMATION TECHNOLOGY UNIVERSITI MALAYSIA SABAH 2010



BORANG PENGESAHAN STATUS TESIS

JUDUL: ANT COLONY OPTIMIZATION IN DYNAMIC ENVIRONMENTS

DAZAH: MASTER OF SCIENCE IN SOFTWARE ENGINEERING

SAYA	CHEN	FEI HUANG	SESI PENGAJIAN:	2005	/2006
Sum.			SEST LENGI DIANT.		

(HURUF BESAR)

Mengaku membenarkan tesis (LPSM/Sarjana/Doktor Falsafah) ini disimpan di Perpustakaan Universiti Malaysia Sabah dengan syarat-syarat kegunaan seperti berikut:-

- 1. Tesis adalah hakmilik Universiti Malaysia Sabah.
- 2. Perpustakaan Universiti Malaysia Sabah dibenarkan membuat salinan untuk tujuan pengajian sahaja.
- Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.

4. Sila tandakan (/)



(Mengandungi maklumat yang berdarjah keselamatan atau Kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)

TERHAD

(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)



TIDAK TERHAD

Chen

(TANDATANGAN PENULIS)

Alamat Tetap: P.O Box A-377

89357 INANAM.

KOTA KINABALU, SABAH

Tarikh: 25 AUGUST 2010

CATATAN:- *Potong yang tidak berkenaan

(TANDATANGAN PUSTAKAWAN) DB JASON TEO

Disahkan Oleh

Nama Penyelia

Tarikh: 2 SEPTEMBER 2010

**Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh tesis ini perlu dikelaskan sebagai SULIT dan TERHAD.

@Tesis dimaksudkan sebagai tesis bagi Ijazah Doktor Falsafah dan Sarjana secara Penyelidikan atau disertai bagi pengajian secara kerja kursus dan Laporan Projek Sarjana Muda (LPSM).

DECLARATION

I hereby declare that the material in this thesis is my own except for quotations, excerpts, equations, summaries and references, which have been duly acknowledged. Any contribution made to research by others, with whom I have worked at UMS or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project's design and conception or in style, presentation and linguistic expression is acknowledged.

04 AUGUST 2010

CHEN FEI HUANG PS05-008-026



CERTIFICATION

NAME : CHEN FEI HUANG

MATRIC NUMBER : PS05-008-026

TITLE : ANT COLONY OPTIMIZATION IN DYNAMIC ENVIRONMENTS

DEGREE : MASTER OF SCIENCE (SOFTWARE ENGINEERING)

VIVA DATE : 4 JUNE 2010

DECLARED BY

1. SUPERVISOR

Dr. Jason Teo Tze Wi

18



ACKNOWLEDGEMENTS

I would like to thank all the people who had helped me along the way in completing this thesis. First of all, I would like to take the opportunity to thank Universiti Malaysia Sabah especially to the Dean of the School of Engineering and Information Technology, Associate Professor Dr. Rosalam Sabartly. I would like to express my sincere gratitude to my supervisor, Dr. Jason Teo for his precious guidance, advice and assistance during my research. I would also like to thank the academics and staffs of the School of Engineering and Information Technology that provided a lot of guidance and help.

My acknowledgement also goes to the PGD Scholarship provided by YTL, with further support from Centre for Postgraduate Studies, Universiti Malaysia Sabah.

I would also like to express my gratitude to my family and friends for their devotion, support and patience.

Last but not least, it would not be possible to list down everyone that has helped and lend their assistance in completing this work. I am deeply honored and your dedication is truly remembered.

Chen Fei Huang 4 August 2010



ABSTRACT

ANT COLONY OPTIMIZATION IN DYNAMIC ENVIRONMENTS

Optimization is the process of finding the maximum or minimum of some objective function. During recent years, a new direction of research has emerged in optimization, with the research focus moving from the conventional static optimization problems to the dynamic optimization problems. Dynamic optimization problems differ from the static ones where the optimal solution is dynamic and changes over time. A similar trend can be seen in the application of Ant Colony Optimization (ACO). ACO is an optimization technique inspired by the ants' foraging behavior which optimizes their routes taken to food sources. This thesis is an investigation into the application of ACO for solving dynamic optimization problems. The first objective of this study is to identify which ant algorithm performs the best under a dynamic environment. In order to achieve this objective, six ant algorithms namely Ant System (AS), Ant Colony System (ACS), Best-Worst Ant System (BWAS), Elitist Ant System (EAS), Max-Min Ant System (MMAS) and Rank-Based Ant System (RBAS) were implemented to solve a dynamic optimization problem in the form of the dynamic Traveling Salesman Problem (TSP). Three different sizes of the dynamic TSP test sets were used: eil51 (small), lin318 (medium) and d1291 (large). Apart from the size of the optimization problem, how the swapping interval affects the dynamic optimization by the ant algorithms is also investigated. Swapping of cities in the dynamic TSP was done in the early, middle and late stages of the optimization process. A series of 30 test runs were conducted on each dynamic TSP instance and also for each swap condition. The second objective of the research is to investigate the suitability of applying local search algorithms to the best performing ant algorithm from the first objective. For this purpose, three local search algorithms namely 2-opt, 2.5-opt and 3-opt were chosen to be coupled with the ant algorithm in order to solve the dynamic TSP. The last objective of this thesis is to optimize the parameter settings of the best performing ant algorithm with local search. From the experimental analysis, it was found that ACS works best for solving the dynamic TSP compared to the other five ant algorithms. When coupled with a local search technique, a significant improvement can be seen for ACS using the 3-opt local search algorithm. Lastly, it was also found that different optimal parameter settings were required for solving the different sizes of the dynamic TSP problems.



ABSTRAK

Pengoptimuman merupakan satu proses pencarian maksimum atau minimum fungsi berobjektif. Sejak kebelakangan ini, suatu halatuju penyelidikan yang baru telah timbul dalam bidang pengoptimuman, di mana fokus penyelidikan telah berubah daripada masalah pengoptimuman statik kepada masalah pengoptimuman dinamik. Perbezaan di antara masalah dinamik dengan masalah statik adalah masalah dinamik mempunyai penyelesaian optimum yang dinamik dan berubah mengikut masa. Corak yang sama dapat dilihat dalam pengaplikasian "Ant Colony Optimization" (ACO). ACO adalah teknik pengoptimuman yang diilhamkan oleh sifat pencarian makanan semut yang mendoptimumkan penghalaan jalan-jalan mereka ke punca makanan. Tesis ini menerangkan satu penyelidikan yang dijalankan untuk mengenalpasti algoritma semut vang mempunyai pencapaian terbaik dalam keadaan dinamik. Objektif pertama penyelidikan ini ialah mengenal pasti algoritma semut yang berpencapaian terbaik di dalam keadaan dinamik. Untuk mencapai objektif ini, enam algoritma semut, iaitu "Ant System" (AS), "Ant Colony System" (ACS), "Best-Worst Ant System" (BWAS), "Elitist Ant System" (EAS), "Max-Min Ant System" (MMAS) dan "Rank-Based Ant System" (RBAS) telah dibangunkan untuk menyelesaikan masalah dinamik yang berbentuk masalah dinamik "Traveling Salesman" (TSP). Tiga set ujian TSP yang berlainan saiz telah digunakan: eil51 (kecil), lin318 (sederhana) dan d1291 (luas). Selain daripada saiz masalah yang berlainan, kesan masa penukaran bandar-bandar pada masa-masa berlainan ke atas pengoptimuman dinamik oleh algoritma semut juga dikaji. Penukaran bandar-bandar di dalam TSP dinamik telah dilakukan pada tahap awal, pertengahan dan akhir semasa proses pengoptimuman. Satu siri ujian yang merangkumi 30 perlaksanaan bagi setiap masalah TSP dinamik dan juga untuk setiap masa penukaran telah dijalankan. Objektif kedua penyelidikan ialah menyiasat kesesuaian penggunaan algoritma pencarian tempatan apabila diaplikasikan kepada algoritma terbaik hasil dapatan daripada objektif pertama untuk penyelesaian masalah dinamik. Untuk mencapai tujuan ini, tiga algoritma pencarian tempatan iaitu 2-opt, 2.5-opt dan 3opt telah dipilih untuk diaplikasikan bersama algoritma semut bagi menyelesaikan TSP dinamik. Untuk objektif terakhir, penentuan parameter terbaik bagi algoritma semut bersama dengan algoritma pencarian tempatan dilaksanakan. Analisa hasil kajian menunjukkan bahawa ACS mempunyai prestasi terbaik berbanding lima algoritma yang lain dalam keadaan dinamik. Peningkatan prestasi yang ketara dapat dilihat bagi algoritma semut apabila algoritma pencarian tempatan 3-opt diaplikasikan bersama ACS. Hasil ujian daripada objektif terakhir menunjukkan penentuan parameter terbaik yang berlainan diperlukan untuk menyelesaikan dinamik TSP yang berlainan saiz.



TABLE OF CONTENTS

		Page
FRO	NT PAGE	1
DEC	LARATION	ii
CER	TIFICATION	m
АСК	NOWLEDGEMENT	iv
ABS	TRACT	v
ABS	TRAK	vi
ТАВ	LE OF CONTENTS	vii
LIST	T OF FIGURES	×
LIST	r of tables	xi
LIST	T OF ABBREVIATIONS	xii
LIST	T OF SYMBOLS	×iii
СНА	PTER 1: INTRODUCTION	1
1.1	Overview	1
1.2	Ant Colony Optimization	1
1.3	Dynamic Environment	3
	1.3.1 Traveling Salesman Problem and Dynamic Traveling Salesman Problem	3
1.4	Motivation Of Research	4
1.5	Objectives Of Research	5
1.6	Scope And Limitations Of The Research	6
1.7	Expected Outcomes Of The Research	6
1.8	Structure Of Thesis	6
1.9	Conclusion	7
CHA	PTER 2: LITERATURE REVIEW	9
2.1	Overview	9
2.2	Swarm Intelligence	9
2.3	Ant Colony	9
	2.3.1 The artificial ants compared to real ants	10
2.4	Ant Colony Optimization Metaheuristic	11
	2.4.1 The basic framework of the ACO algorithm	12
2.5	Ant Colony Optimization Algorithm	15
	2.5.1 Ant System (AS)	15
	2.5.2 Ant Colony System (ACS)	16
	2.5.3 Elitist Ant System (EAS)	18
	2.5.4 Max-Min Ant System (MMAS)	18
	2.5.5 Kank-based Ant System (RBAS)	19
20	2.5.0 Best-Worst Ant System (BWAS)	20
2.0	ACO ADDIICATION TO SOIVE ODUITIZATION PRODIEMS	/1



2.7	ACO Application In Dynamic Optimization Problems	22
2.8	Local Search Techniques	24
	2.8.1 2-opt	24
	2.8.2 2.5-opt	26
	2.8.3 3-opt	29
2.9	Critical Summary Of Literature Review	31
2.10	Conclusion	32
CHAR	PTER 3: METHODOLOGY	33
3.1	Overview	33
3.2	Hardware And Software Specifications	33
3.3	Test Bed	34
3.4	Pheromone Equalization Strategy	35
3.5	Local Search Techniques	36
3.6	Experimental Setup	38
3.7	Parameter Tuning Setup	40
3.8	Performance Metrics	40
3.9	Conclusion	41
CHAI	PTER 4: ANT COLONY OPTIMIZATION FOR DYNAMIC TSP	42
4.1	Overview	42
4.2	Pheromone Equalization Strategy	42
4.3	Experimental Setup	43
4.4	Results And Discussion	44
4.5	Comparison Of ACS And EAS	49
4.6	Experimental Findings	53
4.7	Conclusion	54
CHAI	PTER 5: LOCAL SEARCH TECHNIQUES IN ANT COLONY SYSTEM FOR DYNAMIC TSP	55
5.1	Overview	55
5.2	Experimental Setup	55
5.3	Results And Discussion	56
5.4	Conclusion	62
CHAI	PTER 6: OPTIMIZING THE PARAMETER SETTINGS OF ANT COLONY SYSTEM FOR DYNAMIC TSP	64
6.1	Overview	64
6.2	Experimental Objectives	64
6.3	Parameter Tuning	65
6.4	Results And Discussion	67
	6.4.1 Number of ants	67
	6.4.2 β value	71
	6.4.3 <i>p</i> value	79
6.5	ACS With Optimal Parameter Setting Vs ACS With Previous Parameter Setting	85
6.6	Conclusion	89
CHA	PTER 7: CONCLUSION	00
71	Overview	90
72	Summary Of Results	90
73	Research Contributions	92
74	Future Works	92
7.5	Conclusion	93
	viii	55



REFERENCES

APPENDIXES

94



LIST OF FIGURES

		Page
Figure 1.1	The process how ants find the shorter path.	2
Figure 2.1	The ACO metaheuristic in pseudo-code.	13
Figure 2.2	An example of 2-opt move on the original tour.	25
Figure 2.3	The 2-opt algorithm.	26
Figure 2.4	An example of 2.5-opt move on the original tour.	27
Figure 2.5	The 2.5-opt algorithm.	28
Figure 2.6	Examples of 3-opt move on the original tour.	29
Figure 2.7	The 3-opt algorithm.	30
Figure 3.1	A 2-opt move on the original tour.	36
Figure 3.2	A 2.5-opt move on the original tour.	37
Figure 3.3	A 3-opt move on the original tour.	38
Figure 4.1	Average optimal value for 30 test runs on eil51 TSP.	45
Figure 4.2	Average optimal value for 30 test runs on lin318 TSP.	46
Figure 4.3	Average optimal value for 30 test runs on d1291 TSP.	48
Figure 4.4	Average optimal value comparison for ACS and EAS on eil51.	50
Figure 4.5	Average optimal value comparison for ACS and EAS on lin318.	52
Figure 4.6	Average optimal value comparison for ACS and EAS on d1291.	53
Figure 5.1	Average optimal value on experiment conducted on eil51 TSP.	57
Figure 5.2	Average optimal value on experiment conducted on lin318 TSP.	59
Figure 5.3	Average optimal value on experiment conducted on d1291 TSP.	61
Figure 5.4	ACS-3opt running on d1291 TSP.	62
Figure 6.1	Average optimal value on number of ants tuning on eil51 TSP.	68
Figure 6.2	Average optimal value on number of ants tuning on lin318 TSP.	69
Figure 6.3	Average optimal value on number of ants tuning on d1291 TSP.	70
Figure 6.4	Relationship between the optimal number of ants and TSP size.	71
Figure 6.5	Average optimal value on β tuning on eil51 TSP (20 ants setting).	73
Figure 6.6	Average optimal value on β tuning on eil51 TSP (30 ants setting).	73
Figure 6.7	Average optimal value on β tuning on lin318 TSP.	75
Figure 6.8	Average optimal value on β tuning on d1291 TSP.	77
Figure 6.9	Relationship between the best Beta setting and TSP size.	79
Figure 6.10	Average optimal value on ρ value tuning on eil51 TSP.	80
Figure 6.11	Average optimal value on ρ value tuning on lin318 TSP.	81
Figure 6.12	Average optimal value on ρ value tuning on d1291 TSP.	83
Figure 6.13	Relationship between the optimal ρ value and TSP size.	85
Figure 6.14	Average optimal value comparison for eil51 TSP.	86
Figure 6.15	Average optimal value comparison for lin318 TSP.	87
Figure 6.16	Average optimal value comparison for d1291 TSP.	88



LIST OF TABLES

		Page
Table 2.1	Applications of ACO.	21
Table 2.2	Recent applications of ACO on dynamic optimization problems.	24
Table 3.1	The number of cities on TSP instance and spare cities pool.	39
Table 3.2	Total of iteration run for each TSP instance for each test run.	39
Table 3.3	Parameter settings for each TSP instance.	40
Table 4.1	Parameter setting for the six ant algorithms.	43
Table 4.2	Results of 30 test runs on eil51 TSP.	46
Table 4.3	Results of 30 test runs on lin318 TSP.	47
Table 4.4	Results of 30 test runs on d1291 TSP.	48
Table 4.5	Experimental results for eil51 TSP.	50
Table 4.6	Experimental results for lin318 TSP.	51
Table 4.7	Experimental results for d2191 TSP.	52
Table 5.1	Parameter settings for ACS.	56
Table 5.2	Result of 30 runs for eil51 TSP.	56
Table 5.3	Result of 30 runs for lin318 TSP.	58
Table 5.4	Result of 30 runs for d1291 TSP.	60
Table 6.1	Parameter tuning for each TSP instance.	66
Table 6.2	Parameters setting of ACS on previous experiment.	67
Table 6.3	Best optimal value on number of ants tuning on eil51 TSP.	68
Table 6.4	Best optimal value on number of ants tuning on lin318 TSP.	69
Table 6.5	Best optimal value on number of ants tuning on d1291 TSP.	70
Table 6.6	Best optimal value on number of ants tuning on eil51 TSP.	72
Table 6.7	Best optimal value on β tuning on lin318 TSP.	74
Table 6.8	Best optimal value on β tuning on d1291 TSP.	76
Table 6.9	Best optimal value on ρ tuning on d1291 TSP.	82
Table 6.10	ACS parameter settings comparison.	88



LIST OF ABBREVIATIONS

ACO	Ant Colony Optimization
ACS	Ant Colony System
AS	Ant System
BWAS	Best-Worst Ant System
EAS	Elitist Ant System
MMAS	Max-Min Ant System
PSO	Particle Swarm Optimization
OAP	Quadratic Assignment Problem
RBAS	Rank-based Ant System
TSP	Traveling Salesman Problem



LIST OF SYMBOLS

α	Parameter that controls the pheromone information
β	Parameter that controls the heuristic information
d_{ij}	The distance between city i and city j
ρ	Parameter that control the evaporation rate
$ au_{ij}$	The pheromone information between city i and city j
η_{ii}	The heuristic information between city i and city j
S _{global-best}	The global-best ant



CHAPTER 1

INTRODUCTION

1.1 Overview

Over the years, Swarm Intelligence has been widely studied by researchers and scholars alike. Swarm Intelligence is a field that adopts the natural behaviour of the real animal or insects to solve many problems such as the NP-hard optimization problems (Dorigo & Stützle, 2004). One of the examples of Swarm Intelligence techniques is Ant Colony Optimization (ACO). Ant Colony Optimization is a well-known technique for solving optimization problems. Many studies have been successful in applying ACO to solve optimization problems (Gambardella & Dorigo, 1995; Stützle & Hoos, 1997; Dorigo *et al.*, 1999).

1.2 Ant Colony Optimization

The origin of the ACO algorithm lies within the real ants itself. The behavior of the ants to find the shortest route from the nest to food is the main motivation for Dorigo et al. to create the ACO algorithm (Dorigo & Stützle, 2001). An important fact to keep in mind is that many ants species are almost blind, which avoids the exploitation of visual clues (Cordon et al., 2002). The ants are able to find the shortest route to the food source when it interacts with other ants in the colony through pheromone (a chemical that the ants can smell). This special behavior was studied by Deneubourg and colleagues in the year 1989 and 1990 (Goss et al., 1989; Deneubourg et al., 1990). In the experimental report, Deneubourg and colleagues used a double bridge connecting a nest of ants and a food source to study the pheromone trail and following behavior in controlled experimental conditions. In that experiment, a colony of Linepithema humile ants (also known as Argentine ants) was used. The Argentine ants are known for depositing pheromone both when leaving and when returning to the nest. A number of experiments were conducted in which the ratio of the two branches of the bridge varied. The interesting fact in the experiments is that one of the branches was longer than the other. In the experiment, at start the ants were left free to move between the nest and the food source and the percentage of ants that chose one or the other of the two branches was observed over time. The outcome of the experiments showed that almost all ants ended up using the shorter branch despite the random start.



The result of the experiments can be explained as per Figure 1.1. Before the start of the experiment, there is no pheromone on the two branches. Since there is no preference on which route to take, it is expected on average, half of the ants choose the short branch and the other half the long branch. As the ants move along the branch, it will deposit pheromone along the way. The ants on the shorter branch will arrive at the food source faster compared to the other ants. But then, the ants must decide which branch to take to travel back to the nest. Because the ants choosing the short branch arrived first, the short branch has a trail of pheromone. The higher level of pheromone on the short branch biases the ants' decision in its favor. Therefore, the ants will take the branch with higher level of pheromone (the short branch) to bring back the food to the nest. With this happening, the pheromone of the short branch accumulates faster and eventually will be used by majority of the ants.



Figure 1.1: The process how ants find the shorter path.



Based on the experimental results on the real ants, Dorigo and Di Caro (1999) came up with the ACO Metaheuristic. A more detailed view on the ACO metaheuristic will be discussed in Chapter 2.

1.3 Dynamic Environment

In recent years, there has been a growing interest in studying ACO for dynamic environments. When dealing with dynamic environments, many types of uncertainties need to be taken into account. According to Jin and Branke (2005), uncertainties in the environments can be categorized into four classes. The first one is noise. Noise is subject to the fitness evaluation and may come from many different sources. The second class of uncertain environments is robustness. The design variables are subject to changes after an optimization process. Therefore, the requirement that the solution will work satisfactorily when the changes occur is called robust solutions. The third class is fitness approximation. In this class, the approximation approach is used based on data generated by simulation and experiments. This occurs when the fitness function is expensive to evaluate or the analytical fitness function is not available. The final class is time-varying fitness functions where the optimum point changes over time, which is the focus of this thesis.

1.3.1 Traveling Salesman Problem and Dynamic Traveling Salesman Problem

One of the key areas of research in ACO is to solve the Traveling Salesman Problem (TSP). The first ant algorithm, Ant System (AS) was designed to solve TSP (Gambardella & Dorigo, 1995). Looking at the recent trend, more and more researchers put their effort in investigating dynamic problems instead of the conventional static problems. The TSP idea originated from a salesman whom, starting from city *a*, would like to visit all the cities in his route exactly once before returning to city *a*. The main goal of TSP is to find the shortest length to travel to each city from the graph exactly once. This is also known as the Hamiltonian circuit and often referred to as a tour.

The dynamic TSP however is different from the conventional TSP, where cities can be added or removed during the optimization process (Dorigo & Stützle, 2004). For example, at iteration 100 during run time, 5 cities were deleted from the TSP instance and 3 new cities were added in. These sudden changes on the TSP instances make the problem dynamic and hence remit in a time-varying fitness function. The main objective remained the same, which is to find the shortest tour to visit all cities.



3

1.4 Motivation Of The Research

Throughout the years, ACO described in section 1.2 had been applied to solve many NPhard problems such Traveling Salesman Problem (TSP), Quadratic Assignment Problem (QAP), vehicle routing problems and many more. These problems however are often static where all data are known in advance before the optimization has started. Real world problems are often dynamic where the optimal point varies from time to time (Gambardella & Dorigo, 1995; Stützle & Hoos, 1997).

In Cordon et al. (2002) the authors had pointed out the current research direction of ACO. Researchers are now focusing on applying ACO to solve multi-objective optimization problems, Another research direction stated by the authors is the application of ACO to time-varying variants of classical NP-hard optimization problems like the TSP or the QAP. Throughout the literature review process, it was found that not many research works have been conducted in applying ACO to dynamic problems. According to a technical report written by Angus (2006), several research works were done to apply ACO into the dynamic environment. In the paper, the author explained two research works that had been applied to dynamic problems. The first one is FIFO-Queue ACO which was introduced by Guntsch and Middendorf (2002). The second research discussed in the paper was Ant Systems for dynamic TSP done by Eyckelhof and Snoek (2002). The version of dynamic TSP used by the researchers introduced a "traffic jam" to the current best path. This was done by increasing the weight of the edge on the current best path. But the main question here is whether Ant System is the best ant algorithm among all the different ACO algorithms to apply to when solving dynamic problems? As reported in (Angus, 2006), only a few research work had been done on ACO application in the dynamic Traveling Salesman Problem.

Thus, the motivation of this research is to find out which is the best ant algorithm to use when dealing with dynamic environments. There is no recommendation so far on which ant algorithm is to be used when dealing with dynamic environments. Therefore, this research will investigate the best ant algorithm to apply when dealing with dynamic environments. This research will also attempt to find out which local search algorithm applied to ant algorithm performs best in dynamic environments. The last objective of the research is to find out the best parameter settings for the ant algorithms when applied to dynamic environments. There is no guarantee that the parameter settings that work well on static environment will also perform well in dynamic environment.



4

1.5 Objectives Of Research

The main objectives of this research are outlined as follows:

1. Comparison of six ACO algorithms in dynamic environments

The first objective of this research is to investigate which ACO algorithm performs best in dynamic environments. Investigation on this was done since there is no report yet on which ACO algorithm performs well in dynamic environments. Six ACO algorithms were chosen to run on the same dynamic environment for a fair performance comparison. The dynamic Traveling Salesman Problem was chosen as the test bed that portrayed the dynamic environment. The idea of dynamic Traveling Salesman Problem originated from Guntsch *et al.* (2001). A slight modification was done to the Traveling Salesman Problem to generate a dynamic version of the TSP. Before the optimization process begins, a number of *n* cities from the total *N* were taken out and stored in a spare cities pool. The cities in the spare cities pool will swap into the TSP instance during run time at iteration *t*. With these changes on the problem instance, the TSP problem is now dynamic. Results from the experiment were analyzed to determine which ant algorithm works well under dynamic environment. The better algorithm will be used in the second objective of the research.

2. Local search optimization for ACO in dynamic environments

The best performing ant algorithm from objective one will be used in the second objective. In objective two, three local search techniques are applied to the ACO and a comparison was done between the three techniques. The three local search techniques that were applied to the ACO were 2-opt local search, 2.5-opt local search and 3-opt local search. Local search techniques have been reported to assist ant algorithm to produce good solutions. Therefore, it is an interesting prospect to investigate the outcome of the local search technique when applied to ant algorithms in a dynamic environment. Besides that, it is important to find out which is the best local search technique to be coupled with the ant algorithm in order to solve dynamic problems. The best local search technique will be implemented for the next objective of the research.

3. Fine tune the parameter settings of ACO in dynamic environments

For the last objective, the parameter settings of the ant algorithm will be fine tuned. This will give the optimal performance of the ant algorithm while working on dynamic



environments. The results of the optimized ant algorithm will be compared with the result from the second objective. It is interesting to see if the parameter settings recommended for static TSP will work well under the dynamic environment.

1.6 Scope And Limitations Of The Research

This research is limit to discrete optimization for dynamic environments especially to ACO metaheuristic. The aim of this research is to determine the best ant algorithm to use when dealing with dynamic environment. Six of the more common used ant algorithms were selected to solve the dynamic environment. Dynamic TSP will be used to emulate the dynamic environment. Three different sizes of dynamic TSP test set were used in this research.

The main limitation of this research is the time of the experiment runs will not be taken into account. The timing of the runs will not be taken as the main focus of this research is on the solution's quality. The six ant algorithms selected cannot represent all the other ant algorithm variation. With this, it is not guaranteed that other ant algorithms, beside the six selected in the research, will work well in solving dynamic environment.

1.7 Expected Outcomes Of The Research

The expected output for this research is to identify the best ant algorithm when dealing with a dynamic environment. The next expected outcome will be the suitability of a local search technique coupled together with ant algorithm when applied to dynamic problems. Lastly, the optimal parameter settings for the ant algorithm that performs best in a dynamic environment will be produced.

1.8 Structure Of Thesis

This thesis has seven chapters and is organized as follows:

In Chapter 1, the introduction of the thesis is presented. An overview of the research field is shown followed by the motivations and the objectives of the research. An outline of the expected research output and conclusion is given at the end of this chapter.

For Chapter 2, a literature review was conducted on ACO and dynamic



environments especially for dynamic TSP. The review focuses on the history, background, varieties and techniques of the ants' algorithm. Some previous works done on applying ant algorithm to dynamic environments are discussed. The outcome of the literature review is discussed and the chapter closes with a summary.

Chapter 3 explains in detail the methodology used throughout the research. Experimental setup and equipment used will be discussed. The dynamic TSP that is used throughout the research will be explained in detail.

In Chapter 4, the experiment conducted for the first objective is presented. In this chapter, six ant algorithms were tested on the dynamic TSP. Three different sizes of the dynamic TSP were used as the test bed. The results of the experiment run for each ant algorithm was analyzed and compared against the rest of the algorithms. The ant algorithm that produced the best result was applied to the next objective.

Chapter 5 explains the local search techniques that were applied to the ant algorithm. Local search techniques that were applied to the ant algorithm were 2-opt, 2.5-opt and 3-opt local search. The outcome of the experiment will be compared with the basic ant algorithm without application of local search technique. The best local search technique will be applied to the next objective.

Chapter 6 discusses the parameter tuning of the ant algorithm. Three main parameter settings were explicitly hand-tuned to obtain the maximum performance out of the ant algorithm when applied to a dynamic environment. Analysis of the experimental result will give the best parameter settings applied for ant algorithm in a dynamic environment. The parameter settings will be compared with the recommended settings for static TSP problems.

The conclusion of the research is discussed in Chapter 7. The important findings from the research are presented in this chapter. The directions of future work arising from this thesis are also discussed on Chapter 7.

1.9 Conclusion

An overview of the research topic was discussed in this chapter. Then, the main motivations of the research were discussed. This chapter also highlighted the main



7

objectives of this research as well as the expected output. The scope and limitation of the thesis were discussed. The structure of the thesis was also discussed in this chapter. The next chapter will focus on the literature review of the research.



CHAPTER 2

LITERATURE REVIEW

2.1 Overview

In this chapter, the basic framework of the ant algorithm will be explained first. Then, the basic structure of the six ant algorithms used will be discussed. Next, the application of the ACO metaheuristic to solve dynamic optimization problems will be discussed. This will give a clear view of the current state-of-the-art of ACO in solving dynamic optimization problems. Finally, an explanation about the uncertainties in an environment is presented.

2.2 Swarm Intelligence

Swarm intelligence is known as the emergent collective intelligence of groups of simple agents. According to (Bonabeau *et al.*, 1999), any attempt to design algorithms or distributed problem-solving techniques that originally took its inspiration from the collective behaviour of social insects and other animals, is known as swarm intelligence. Some of the well known studies and application of swarm intelligence includes Particle Swarm Optimization (PSO) (Kennedy & Eberhart, 1995), ACO (Dorigo & Stützle, 2004), Nest Building Behaviour of Wasps and Termites (Bonabeau *et al.*, 1999), and Honey Bees (Karaboga and Akay, 2009) and many more. The focus on this research is on ACO.

2.3 Ant Colony

Ant colonies consist of individual ants that communicate with each other in performing difficult tasks (Cordon *et al.*, 2002). With this behavior, the ants are known as social insects. One of the most interesting behaviors of ants is the ability to find the shortest path to the food source exploring from the nest. To add to the fact that most ant species are visually impaired, which makes food searching based on visual clue is almost next to impossible (Dorigo & Stützle, 2004; Cordon *et al.*, 2002). This foraging behavior has been the key aspect of the Ant Colony Optimization study. So, how do the ants manage to always find the shortest path to the food source and back to the nest? The next part will describe how the real life ants manage to find the shortest path.

Chapter 1 had explained on how the ants were able to communicate with each



other. The ants coordinate through stigmergy communication. While the ants travel between their nest and food source, they deposit a chemical substance known as pheromone. This pheromone enables the ants to communicate with each other. As the ants leave behind the pheromone, this will leave trails of pheromone on the route. First, the ants will search randomly on the route for food source. For example, there are two routes to the food source that appear in front of the ants. One is shorter and the other is longer. This will give a clear view on how the foraging behavior of ants works. When choosing a path, ants tend to choose the trail that is higher in pheromone level. At the start of searching for food, ants will travel randomly on both paths. Both paths will be reinforced with pheromone as ants travel along them. A shorter path will get more pheromone reinforcement as the path will be regularly used by other ants compared to the longer path. The ants that travel on the shorter path will reach the food source earlier.

The ants will then choose the path that has the higher pheromone level, which is the shorter path as the ants on the longer path are still on the way to the food source. After some times, the shorter path will have a high pheromone concentration as more and more ants will use this path. This is due to the fact that ants traveling on the shorter path will reach the food source and bring it back to the nest earlier compared to the longer path. Eventually, this high pheromone concentration path is the shortest route to the food source found by the ants. All other ants will follow this path as the ants prefer the path that is marked with high pheromone concentration.

2.3.1 The artificial ants compared to real ants

Unlike real ants, an artificial ant is a simple agent that tries to solve problem based on the pheromone and heuristics information available (Dorigo & Stützle, 2004). There are some similarities and differences between the artificial ants and real ants. One of the most significant similarities is the usage of a colony of individuals that interact with each other to solve a given task (Dorigo & Stützle, 2004).

Other than that, both artificial and real ants share a common goal: to search for the shortest path from the nest to the food source (Dorigo & Stützle, 2004). Both ants share another common characteristic which is to deposit pheromones that will affect the environment. Real ants deposit an odourous substance called pheromone while artificial ants' pheromone will be in a numerical form.



REFERENCES

- Angus, D. 2006. The Current State of Ant Colony Optimisation Applied to Dynamic Problems. http://www.itee.uq.edu.au/~uqdangus/research.html. Retrieved 20 June 2006.
- Bentley, J. L. 1992. Fast Algorithm for Geometric Traveling Salesman Problem. ORSA Journal on Computing. Volume 4. No. 4. 387-411.
- Blum, C. 2005. Ant colony optimization: Introduction and recent trends. Physics of Life Reviews. Volume 2(4). 353-373.
- Bullnheimer, B., Hartl, R. F., and Strauss, C. 1999. Applying the Ant System to the vehicle routing problem. Voss, S., Martello, S., Osman, I. H., and Roucairol, C., Eds. *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*. Dordrecht: Kluwer Academic Publishers.
- Bonabeau, E., Dorigo, M., and Theraulaz, G. 1999. *Swarm Intelligence: From Natural to Artificial System*. New York: Oxford University Press.
- Colorni, A., Dorigo, M., Maniezzo, V., and Trubian, M. 1994. Ant System for job-shop scheduling. *JORBEL—Belgian Journal of Operations Research*. Statistics and Computer Science. Volume **34**(1): 39–53.
- Cordon, O., Fernandez de Viana, I., Herrera, F., and Moreno, L. 2000. A new ACO model integrating evolutionary computation concepts: The best-worst Ant System. In proceedings of ANTS2000 – From Ant Colonies to Artificial Ants: A Series of International Workshops on Ant Algorithms. University Libre de Bruxelles.
- Cordon, O., Herrera, F., and Stützle, T. 2002. A Review on Ant Colony Optimization Metaheuristic: Basis, Models and New Trends. *Mathware and Soft Computing*, Volume 9(2-3). 141-175.
- Deneubourg, J. L., Aron, S., Goss, S., and Pasteels, J. M. 1990. The self-organizing exploratory pattern of the Argentine ant. *Journal of Insect Behavior*. Volume 3. 159-168.
- Di Caro, G., and Dorigo, M. 1997. AntNet: A mobile agents approach to adaptive routing. Technical report IRIDIA/97-12. University Libre de Bruxelles: IRIDIA.
- Donati, A. V., Montemanni, R., Casagrande, N., Rizzoli, A. E., and Gambardella, L. M. 2008. Time dependent vehicle routing problem with a multi ant colony system. *European Journal of Operational Research*. Volume **185**. Issue 3. 1174-1191.
- Dorigo, M., Maniezzo, V., and Colorni, A. 1991. Positive feedback as a search strategy. *Technical Report 91-016*. Politecnico di Milano.
- Dorigo, M., and Gambardella, L. M. 1997. Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem. *IEEE Transactions on Evolutionary Computation.* Volume **1**. 53-66.
- Dorigo, M., and Di Caro, G. 1999. The Ant Colony Optimization meta-heuristic. In Corne, D., Dorigo, M., and Glover, F., Eds. *New Ideas in Optimization*. London: McGraw Hill.



- Dorigo, M., Di Caro, G., and Gambardella, L. M. 1999. Ant algorithms for discrete optimization. Artificial Life. Volume 5(2). 137-172.
- Dorigo, M., and Stützle, T. 2001. The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances. *In Handbook of Metaheuristics*. Kluwer Academic Publishers.
- Dorigo, M., and Stützle, T. 2003. The ant colony optimization metaheuristic: Algorithms, applications and advances. *International Series in Operations Research & Management Science*. Volume **57**. 250-285.
- Dorigo, M., and Stützle, T. 2004. Ant Colony optimization. Cambridge: MIT Press.
- Dorigo, M., and Blum, C. 2005. Ant colony optimization theory: A survey. Theoretical Computer Science. Volume 344, Issues 2-3. 243 – 278.
- Eyckelhof, C. J., and Snoek, M. 2002. Ant systems for a dynamic TSP. In ANTS '02: Proceedings of the Third International Workshop on Ant Algorithms. London: Springer-Verlag. 88–99.
- Gambardella, L. M., and Dorigo, M. 1995. Ant-Q: A Reinforcement Learning Approach to the Traveling Salesman Problem, *In Proceedings of ML-95*. Twelfth International Conference on Machine Learning. Tahoe City. 252-260.
- Gambardella, L. M., and Dorigo, M. 1997. HAS-SOP: An hybrid Ant System for the sequential ordering problem. *Technical report IDSIA-11-97*. Lugano: IDSIA.
- Gambardella, L. M., Taillard, E. D., and Dorigo, M. 1999. Ant Colonies for the Quadratic Assignment Problem. *Journal of the Operational Research Society*. 167-176.
- Glover, F., and Kochenberger, G. 2002. Handbook of metaheuristics. Norwell: Kluwer Academic Publishers.
- Goss, S., Aron, S., Deneubourg, J. L., and Pasteels, J. M. 1989. Self-organized short-cuts in Argentine ant. *Naturwissenschaften*. Volume 76. 579-581.
- Guntsch, M., and Middendorf, M. 2001. Pheromone modification strategies for ant algorithms applied to dynamic TSP. In Lecture Notes in Computer Science. Volume 2037/2001. Como: EvoWorkshops on Application of Evolutionary Computing. 213-222.
- Guntsch, M., Middendorf, M., and Schmeck, H. 2001. An Ant Colony Optimization Approach to Dynamic TSP. In Proceedings of the Genetic and Evolutionary Computation Conference. San Francisco: Genetic and Evolutionary Computation Conference. 860-867.
- Hani, Y., Amodeo, L., Yalaoui, F., and Chen, H. 2007. Ant colony optimization for solving an industrial layout problem. *European Journal of Operational Research*. Volume 183. Issue 2. 633-642.
- Jin, Y., and Branke, J. 2005. Evolutionary Optimization in Uncertain Environments. IEEE Transactions on Evolutionary Computation. Volume 9(3). 303 – 317.
- Johnson, D. S., and McGeoch, L. A. 1997. The Traveling Salesmen Problem: A Case Study in Local Optimization in Local Search in Combinatorial Optimization. Aarts, E. H. L., and Lenstra, J. K., Eds. New York: Wiley.



- Karaboga, D., and Akay, B. 2009. A survey: algorithms simulating bee swarm intelligence. Artificial Intelligence Review. Volume 31. 61-85.
- Kennedy, J., and Eberhart, R. 1995. Particle Swarm Optimization. Proceedings of IEEE International Conference on Neural Networks. Volume IV. 1942-1948
- Kong, M., Tian, P., and Kao, Y. 2008. A new ant colony optimization algorithm for the multidimensional Knapsack problem. *Computers & Operations Research*. Volume 35. Issue 8. 2672-2683.
- Lorpunmanee, S., Sap, M. N., Abdullah, A. H., and Chompoo-inwai, C. 2007. An Ant Colony Optimization for Dynamic Job Scheduling in Grid Environment. *International Journal of Computer and Information Science and Engineering*. Volume 1(4). 1307-4161.
- Maniezzo, V., Colorni, A., and Dorigo, M. 1994. The Ant System applied to quadratic assignment problem. *Technical report IRIDIA/94-28*. Brussels: IRIDIA.
- Musa, R., and Chen, F. F. 2008. Simulated annealing and ant colony optimization algorithms for the dynamic throughput maximization problem. *The International Journal of Advanced Manufacturing Technology*. Volume **37**(7). 837-850.
- Pu, L. M., Yu, H. J., and Chen, D. Z. 2008. Continuous Multi Ant-Colony Optimization and its Application in Dynamic Optimization Problems of Chemical Processes. *Journal* of Chemical Engineering of Chinese Universities. Volume 22(5). 871-876.
- Reinelt, G. 1995. Tsplib95. http://www.iwr.uniheidelberg.de/groups/comopt/software/TSPLIB95/. Retrieved 15 January 2006.
- Schoonderwoerd, R., Holland, O., Bruten, J., and Rothkrantz, L. 1996. Ant-based load balancing in telecommunications networks. *Adaptive Behavior*. Volume 5(2). 169–207.
- Stützle, T., and Hoos, H. 1997. Improvement on the Ant System: Introducing MAX (MIN) Ant System. In Proceedings of the International Conference on Artificial Neural Networks and Genetic Algorithms. International Conference on Artificial Neural Networks and Genetic Algorithms. Norwich. 245-249.
- Tfaili, W., and Siarry, P. 2008. A new charged ant colony algorithm for continuous dynamic optimization. *Applied Mathematics and Computation*. Volume **197**. Issue 2. 604-613.
- UPS Reports Results for 4th Quarter, Full Year. http://www.ups.com/content/my/en/about/news/press_releases/4th_qtr_asia.ht ml?srch_page=1&srch_phr=financial+report. Retrieved on 3 August 2010.
- Xiang, W., and Lee, H. P. 2008. Ant colony intelligence in multi-agent dynamic manufacturing scheduling. *Engineering Applications of Artificial Intelligence*. Volume 21. Issue 1. 73-85.
- Zhou, R., Nee, A. Y. C., and Lee, H. P. 2009. Performance of an ant colony optimisation algorithm in dynamic job shop scheduling problems. *International Journal of Production Research*. Volume 47. Issue 11, 2903-2920.

