

**RANDOM WEIGHT-BASED ANT
COLONY OPTIMIZATION
ALGORITHM FOR THE
MULTI-OBJECTIVE
OPTIMIZATION PROBLEMS**

By

ILANDARI DEWAGE

IRESHA DILHANI ARIYASINGHA

M.Phil.

2016

**RANDOM WEIGHT-BASED ANT
COLONY OPTIMIZATION
ALGORITHM FOR THE
MULTI-OBJECTIVE
OPTIMIZATION PROBLEMS**



By

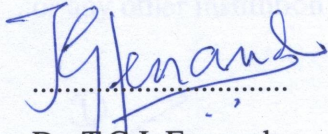
ILANDARI DEWAGE

IRESHA DILHANI ARIYASINGHA

Thesis submitted to the University of Sri Jayewardenepura -
Sri Lanka for the award of the Degree of Master of
Philosophy in Computer Science on 27.01.2016

Certification of the Supervisor

I certify that the candidate has incorporate all corrections, additions, and amendments recommended by the examiners to the final version of the M. Phil. thesis.

A handwritten signature in blue ink, appearing to read 'T. Fernando', written over a dotted line.

Dr. T.G.I. Fernando

Department of Computer Science

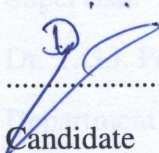
Faculty of Applied Sciences

University of Sri Jayewardenepura

Sri Lanka

Declaration

The work described in this thesis was carried out by me under the supervision of Dr. T.G.I. Fernando and a report on this has not been submitted in whole or in part to any university or any other institution for another Degree/Diploma.

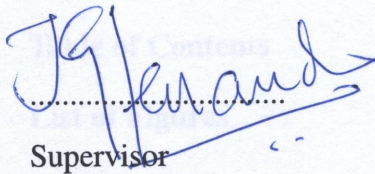


.....

Candidate

I.D.I.D. Ariyasingha

I certify that the above statement made by the candidate is true and that this thesis is suitable for submission to the university for the purpose of evaluation.



Supervisor

Dr. T.G.I. Fernando

Department of Computer Science

Faculty of Applied Sciences

University of Sri Jayewardenepura

Sri Lanka

1.1	Background	1
1.2	Research Motivation and Scope of the Thesis	1
1.3	The Aim and the Objectives of the Thesis	3
1.4	Outline of the Thesis	4
2	Literature Review	6
2.1	Combinatorial Optimization Problems	6
2.1.1	Computational Complexity	7
2.1.2	Traveling Salesman Problem (TSP)	9
2.1.2.1	Multi-objective Traveling Salesman Problem (MoTSP)	10
2.1.2.2	Previous Approaches to the Traveling Salesman Problem	11
2.1.3	Job Shop Scheduling Problem (JSSP)	14
2.1.3.1	Constraints for the Job Shop Scheduling Problem	15
2.1.3.2	Representation of the Job Shop Scheduling Problem	16
2.1.3.3	Notations Used in Job Shop Scheduling Problem	17
2.1.3.4	Previous Approaches to the Job Shop Scheduling Problem	18
2.1.4	Quadratic Assignment Problem (QAP)	20
2.1.4.1	Multi-objective Quadratic Assignment Problem (moQAP)	22
2.1.4.2	Previous Approaches to the Quadratic Assignment Problem	23
2.2	Multi-Objective Optimization	26
2.2.1	Multi-Objective Optimization Problem	28
2.2.1.1	Concept of Domination	29
2.2.1.2	Non-dominated Set	30
2.2.2	Illustrative Representation of Non-dominated Solutions	31
2.2.2.1	Scatter Plot Matrix Method	31
2.2.2.2	Value Path Method	32
2.2.2.3	Bar Chart Method	33
2.2.3	Performance Indicators	34

TABLE OF CONTENTS

Table of Contents	i
List of Figures	vi
List of Tables	ix
Abbreviations	xi
Acknowledgements	xiii
Abstract	xiv
1 Introduction	1
1.1 Background	1
1.2 Research Motivation and Scope of the Thesis	1
1.3 The Aim and the Objectives of the Thesis	3
1.4 Outline of the Thesis	4
2 Literature Review	6
2.1 Combinatorial Optimization Problems	6
2.1.1 Computational Complexity	7
2.1.2 Traveling Salesman Problem (TSP)	9
2.1.2.1 Multi-objective Traveling Salesman Problem (MOTSP)	10
2.1.2.2 Previous Approaches to the Traveling Salesman Problem	11
2.1.3 Job Shop Scheduling Problem (JSSP)	14
2.1.3.1 Constraints for the Job Shop Scheduling Problem . . .	15
2.1.3.2 Representation of the Job Shop Scheduling Problem . .	16
2.1.3.3 Notations Used in Job Shop Scheduling Problem	17
2.1.3.4 Previous Approaches to the Job Shop Scheduling Problem	18
2.1.4 Quadratic Assignment Problem (QAP)	20
2.1.4.1 Multi-objective Quadratic Assignment Problem (mQAP)	22
2.1.4.2 Previous Approaches to the Quadratic Assignment Problem	23
2.2 Multi-Objective Optimization	26
2.2.1 Multi-Objective Optimization Problem	28
2.2.1.1 Concept of Domination	29
2.2.1.2 Non-dominated Set	30
2.2.2 Illustrative Representation of Non-dominated Solutions	31
2.2.2.1 Scatter Plot Matrix Method	31
2.2.2.2 Value Path Method	32
2.2.2.3 Bar Chart Method	33
2.2.3 Performance Indicators	34

2.2.3.1	Pseudo-optimal Pareto Front	35
2.2.3.2	Overall Non-dominated Vector Generation (ONVG) . .	36
2.2.3.3	Overall True Non-dominated Vector Generation (OT- NVG)	36
2.2.3.4	Overall True Non-dominated Vector Generation Ratio (OTNVGR)	37
2.2.3.5	Error Ratio (ER)	37
2.2.3.6	Hypervolume Ratio (HVR)	38
2.2.3.7	Coverage Performance Indicator (C Performance Indi- cator)	38
2.2.3.8	Relative Percentage Deviation (RPD)	39
2.3	Ant Colony Optimization	40
2.3.1	What is a Meta-heuristic ?	41
2.3.2	Ant Colony Optimization Meta-heuristic	42
2.3.3	Applying ACO Algorithms to the Traveling Salesman Problem . .	44
2.3.3.1	Ant System (AS)	45
2.3.3.2	Ant Colony System (ACS)	47
2.3.4	Parameter Values	48
2.3.5	Algorithmic Components for MOACO Algorithms	49
2.3.6	Multi-objective Ant Colony Optimization Algorithms	51
2.3.6.1	An Efficient Ant Colony Optimization Algorithm for the Multi-objective Flow Shop Scheduling Problem (ACO- MOFS)	52
2.3.6.2	Performance Analysis of Elitism in Multi-objective Ant Colony Optimization Algorithms (AMPACOA)	56
2.3.6.3	Crowding Population-based Ant Colony Optimization (CPACO)	57
2.3.6.4	Ant Colony Optimization for Multi-objective Optimiza- tion Problems (m-ACO)	59
2.3.6.5	A Multi-objective Ant Colony System (MACS) for the Vehicle Routing Problem with Time Windows	63
2.3.6.6	Multi-Objective Optimization of Time, Cost, Quality and Quantity using Multi Colony Ant Algorithm (MCAA) . .	65
2.3.6.7	Pareto Strength Ant Colony Optimization (PSACO) . .	66
3	Methodology	69
3.1	The Proposed MOACO Algorithm - Random Weight-based Ant Colony Optimization (RWACO)	69
3.1.1	State Transition Rule	70
3.1.2	Local Updating Rule	71
3.1.3	Global Updating Rule	72
3.2	Adaptation of MOACO into the Traveling Salesman Problem	74
3.2.1	An Efficient Ant Colony Optimization Algorithm for the Multi- objective Flow Shop Scheduling Problem (ACOMOFs)	74

3.2.2	Ant Colony Optimization for Multi-objective Optimization Problems (mACO ₄)	75
3.2.3	A Multi Objective Ant Colony System for the Vehicle Routing Problem with Time Windows (MACS)	76
3.2.4	Multi-objective Optimization of Time Cost Quality Quantity using the Multi-colony Ant Algorithm (MCAA)	77
3.2.5	Pareto Strength Ant Colony Optimization (PSACO)	77
3.3	Adaptation of MOACO to the Job Shop Scheduling Problem	78
3.3.1	Random Weight-based Ant Colony Optimization Algorithm (RWACO)	78
3.3.2	An Efficient Ant Colony Optimization Algorithm for the Multi-objective Flow Shop Scheduling Problem (ACOMOFs)	80
3.3.3	Crowding Population-based Ant Colony Optimization (CPACO)	81
3.3.4	Ant Colony Optimization for Multi-objective Optimization Problems (mACO ₄)	81
3.3.5	A Multi-objective Ant Colony System (MACS) for the Vehicle Routing Problem with Time Windows	81
3.3.6	Multi-objective Optimization of Time Cost Quality Quantity using the Multi-colony Ant Algorithm (MCAA)	82
3.3.7	Pareto Strength Ant Colony Optimization (PSACO)	83
3.4	Adaptation of MOACO to the Quadratic Assignment Problem	83
3.4.1	Procedure to Calculate the Heuristic Information η_{ij} , for the Quadratic Assignment Problem	83
3.4.2	Random Weight-based Ant Colony Optimization Algorithm (RWACO)	85
3.4.3	An Efficient Ant Colony Optimization Algorithm for the Multi-objective Flow Shop Scheduling Problem (ACOMOFs)	87
3.4.4	Crowding Population-based Ant Colony Optimization (CPACO)	88
3.4.5	A Multi-objective Ant Colony System (MACS) for the Vehicle Routing Problem with Time Windows	88
3.5	Improve the PSACO Algorithm with the Random Weight-based Method	89
4	Results and Discussion	91
4.1	Results Obtained with Multi-objective Ant Colony Optimization Algorithms for the Traveling Salesman Problem	92
4.1.1	Experimental Setup	92
4.1.1.1	Problem instances and parameter setting	93
4.1.2	Analysis of Experimental Results and Discussion	94
4.1.2.1	Computation Times	95
4.1.2.2	Visual Representation of Non-dominated Solutions	99
4.1.2.3	Analysis of the Overall True Non-dominated Vector Generation Ratio (OTNVGR)	105
4.1.2.4	Analysis of the Coverage Performance Indicator	108
4.1.2.5	Discussion	110

4.2	Results Obtained with Multi-objective Ant Colony Optimization Algorithms for the Job Shop Scheduling Problem	113
4.2.1	Experimental Setup	113
4.2.1.1	Problem instances and parameter setting	114
4.2.2	Analysis of Experimental Results and Discussion	115
4.2.2.1	Discussion	123
4.3	Random Weight-based ACO Algorithm for the Traveling Salesman Problem	125
4.3.1	Experimental Setup	125
4.3.1.1	Problem instances and parameter setting	126
4.3.2	Analysis of Experimental Results and Discussion	127
4.3.2.1	Visual Representation of Non-dominated Solutions	127
4.3.2.2	Analysis of the Solutions with Overall True Non-dominated Vector Generation Ratio (OTNVGR)	128
4.3.2.3	Analysis of the C Performance Indicator	132
4.4	Random Weight-based ACO Algorithm for the Job Shop Scheduling Problem	133
4.4.1	Experimental Setup	133
4.4.1.1	Problem instances and parameter setting	134
4.4.2	Analysis of Experimental Results and Discussion	134
4.4.2.1	Discussion	137
4.5	Random Weight-based ACO Algorithm for the Quadratic Assignment Problem	138
4.5.1	Experimental Setup	138
4.5.1.1	Problem instances and parameter setting	138
4.5.2	Analysis of Experimental Results and Discussion	139
4.5.2.1	Visual Representation of Non-dominated Solutions	139
4.5.2.2	Analysis with OTNVGR	140
4.5.2.3	Analysis of the C Performance Indicator	144
4.5.2.4	Discussion	145
4.6	The PSACO Algorithm with Random Weight-based Method	146
4.6.1	Experimental Setup	146
4.6.1.1	Problem instances and parameter setting	147
4.6.2	Analysis of Experimental Results and Discussion	147
4.6.2.1	Visual Representation of Non-dominated Solutions	147
4.6.2.2	Analysis with OTNVGR	148
4.6.2.3	Analysis with C Performance Indicator	154
4.6.2.4	Discussion	155
5	Conclusions	156
5.1	Conclusions	156
5.1.1	Performances of the Multi-objective Ant Colony Optimization Algorithms	156
5.1.1.1	Performances of the MOACO for the TSP	157
5.1.1.2	Performances of the MOACO for the JSSP	158
5.1.1.3	General Conclusion	159

5.1.2	Random Weight-based Ant Colony Optimization Algorithm . . .	160
5.1.2.1	RWACO algorithm for the TSP	160
5.1.2.2	RWACO algorithm for the JSSP	160
5.1.2.3	RWACO algorithm for the QAP	161
5.1.3	The PSACO Algorithm with Random Weight-based Method . . .	161
5.2	Future Work	162
References		163
Appendix A Publications		174
A.1	Journal Publications	174
A.2	Conference Publications	175
A.3	Abstracts	175
A.4	Paper Submissions	175

LIST OF FIGURES

2.1	The relationship between P, NP, NP-hard and NP-complete set of problems	8
2.2	Optimal solution returns for the TSP benchmark instance att532	10
2.3	Visual representation of FT03 benchmark instance	16
2.4	Procedure of the ideal multi-objective optimization	28
2.5	Different pareto fronts are represented in the objective space	30
2.6	The scatter plot matrix method	31
2.7	The value path method	32
2.8	The bar chart method	34
2.9	(A) Real ants travels between nest and food source. (B) An obstacle places on the path. (C) Ants choose whether to turn left or right with equal probability based on the pheromone which is deposited more on the shorter path. (D) All ants have followed the shorter path.	41
2.10	An ant in the city i chooses the next city within the cities that has not visited yet based on the function of heuristic η and pheromone trail τ	44
2.11	Integer ranks have been assigned for a set of solutions according to the dominance ranking procedure	58
4.1	Computation time returns by each algorithm for Kroab50 and Kroac50 TSP instances for model 1(10x100)	96
4.2	Computation time returns by each algorithm for Kroab100 and Kroabc50 TSP instances for model 1(10x100)	97
4.3	Computation time returns by each algorithm for Kroabc100 and Kroabcd50 TSP instances for model 1(10x 100)	97
4.4	Computation time returns by each algorithm for Kroab50 and Kroac50 TSP instances for model 2(20x100)	97
4.5	Computation time returns by each algorithm for Kroab100 and Kroabc50 TSP instances for model 2(20x100)	98
4.6	Computation time returns by each algorithm for Kroabc100 and Kroabcd50 TSP instances for model 2(20x 100)	98
4.7	Computation time returns by each algorithm for Kroab50 and Kroac50 TSP instances for model 3(20x50)	98
4.8	Computation time returns by each algorithm for Kroab100 and Kroabc50 TSP instances for model 3(20x50)	99
4.9	Computation time returns by each algorithm for Kroabc100 and Kroabcd50 TSP instances for model 3(20x 50)	99
4.10	Pareto fronts returns by each algorithm for Kroab50 TSP instance in model 1 (10 x 100)	100
4.11	Pareto fronts returns by each algorithm for Kroab50 TSP instance in model 2 (20 x 100)	101
4.12	Pareto fronts returns by each algorithm for Kroab50 TSP instance in model 3 (20 x 50)	101

4.13	Pareto fronts returns by each algorithm for Kroac50 TSP instance in model 1 (10 x 100)	102
4.14	Pareto fronts returns by each algorithm for Kroac50 TSP instance in model 2 (20 x 100)	102
4.15	Pareto fronts returns by each algorithm for Kroac50 TSP instance in model 3 (20 x 50)	103
4.16	Pareto fronts returns by each algorithm for Kroab100 TSP instance in model 1 (10 x 100)	103
4.17	Pareto fronts returns by each algorithm for Kroab100 TSP instance in model 2 (20 x 100)	104
4.18	Pareto fronts returns by each algorithm for Kroab100 TSP instance in model 3 (20 x 50)	104
4.19	Box-plots of the results obtained by the C performance indicator for MOACO algorithms of model 1	110
4.20	Box-plots of the results obtained by the C performance indicator for MOACO algorithms of model 2	111
4.21	Box-plots of the results obtained by the C performance indicator for MOACO algorithms of model 3	111
4.22	Pareto fronts return by each algorithm for Kroab50 TSP instance	128
4.23	Pareto fronts return by each algorithm for Kroac50 TSP instance	129
4.24	Pareto fronts return by each algorithm for Kroab100 TSP instance	129
4.25	Box-plots of the results obtained by the C performance indicator for MOACO algorithms on TSP instances	133
4.26	Non-dominated solutions returns by each algorithm for KC10-2fl-1uni QAP instance	140
4.27	Non-dominated solutions returns by each algorithm for KC10-2fl-2uni QAP instance	141
4.28	Non-dominated solutions returns by each algorithm for KC10-2fl-3uni QAP instance	141
4.29	Non-dominated solutions returns by each algorithm for KC10-2fl-1rl QAP instance	142
4.30	Non-dominated solutions returns by each algorithm for KC10-2fl-2rl QAP instance	142
4.31	Non-dominated solutions returns by each algorithm for KC20-2fl-1uni QAP instance	143
4.32	Box-plot of the results returns by C performance indicator for MOACO algorithms on QAP instances	146
4.33	Non-dominated solutions returns by PSACO and newPSACO algorithms for Kroab50 TSP instance	148
4.34	Non-dominated solutions returns by PSACO and newPSACO algorithms for Kroac50 TSP instance	149
4.35	Non-dominated solutions returns by PSACO and newPSACO algorithms for Kroab100 TSP instance	149
4.36	The value path method of newPSACO algorithm and PSACO algorithm for Kroabc50 TSP instance	150

4.37	The value path method of newPSACO algorithm and PSACO algorithm for Kroabc100 TSP instance	150
4.38	The value path method of newPSACO algorithm and PSACO algorithm for Kroabcd50 TSP instance	151
4.39	The bar chart method of newPSACO algorithm and PSACO algorithm for Kroabc50 TSP instance	151
4.40	The bar chart method of newPSACO algorithm and PSACO algorithm for Kroabc100 TSP instance	152
4.41	The bar chart method of newPSACO algorithm and PSACO algorithm for Kroabcd50 TSP instance	152
4.42	Box-plots of the results obtain for the C performance indicator with the six benchmark TSP instances	154

LIST OF TABLES

2.1	Methods Used for Solving the TSP	14
2.2	Methods Used for Solving the JSSP	18
2.3	Methods Used for Solving the QAP	27
2.4	MOACO algorithmic components and their values	50
2.5	A taxonomy for review papers on multi-objective ACO algorithms	53
2.6	Recent Multi-objective Ant Colony Optimization Algorithms	66
3.1	A taxonomy for multi-objective ACO algorithms	73
4.1	Three different models which MOACO algorithms were applied by changing both the number of ants and number of iterations	94
4.2	Total number of non-dominated solutions of y_{apr} for Kroab50, Kroac50 and Kroab100 TSP instances	106
4.3	Total number of non-dominated solutions of y_{apr} for Kroabc50, Kroabc100 and Kroabcd50 TSP instances	107
4.4	Comparison of solutions with y_{apr} of Kroab50 TSP instance for each model	107
4.5	Comparison of solutions with y_{apr} of Kroac50 TSP instance for each model	107
4.6	Comparison of solutions with y_{apr} of Kroab100 TSP instance for each model	107
4.7	Comparison of solutions with y_{apr} of Kroabc50 TSP instance for each model	108
4.8	Comparison of solutions with y_{apr} of Kroabc100 TSP instance for each model	108
4.9	Comparison of solutions with y_{apr} of Kroabcd50 TSP instance for each model	108
4.10	Objective functions considered in each JSSP instance	115
4.11	Parameter values considered for each MOACO algorithm	115
4.12	Results of each JSSP problem instance for two objectives, obtained with model 1 (20 ants are used)	116
4.13	Results of each JSSP problem instance for three objectives, obtained with model 1 (20 ants are used)	117
4.14	Results of each JSSP problem instance for four objectives, obtained with model 1 (20 ants are used)	118
4.15	Results of each JSSP problem instance for two objectives, obtained with model 2 (50 ants are used)	119
4.16	Results of each JSSP problem instance for three objectives, obtained with model 2 (50 ants are used)	120
4.17	Results of each JSSP problem instance for four objectives, obtained with model 2 (50 ants are used)	121
4.18	Parameter values considered by each MOACO algorithm	126
4.19	Total number of non-dominated solutions in Y_{apr} for TSP instances	131
4.20	Comparison of solutions with $ Y_{apr} $ for Kroab50 and Kroac50 TSP instances	131
4.21	Comparison of solutions with $ Y_{apr} $ for Kroab100 and Kroabc50 TSP instance	131

4.22	Comparison of solutions with $ Y_{apr} $ for Kroabc100 and Kroabcd50 TSP instance	131
4.23	Results of JSSP with RPDs and ARPDs of each MOACO algorithm for two objectives	136
4.24	Results of JSSP with RPDs and ARPDs of each MOACO algorithm for three objectives	136
4.25	Results of JSSP with RPDs and ARPDs of each MOACO algorithm for four objectives	137
4.26	Parameter values for each MOACO algorithm	138
4.27	Number of non-dominated solutions in y_{apr} for QAP benchmark instances	144
4.28	Comparison of solutions with ONVG, OTNVG and OTNVGR of KC10-2fl-1uni and KC10-2fl-2uni QAP instances	144
4.29	Comparison of solutions with ONVG, OTNVG and OTNVGR of KC10-2fl-3uni and KC10-2fl-1rl QAP instances	144
4.30	Comparison of solutions with ONVG, OTNVG and OTNVGR of KC10-2fl-2rl and KC20-2fl-1uni QAP instances	144
4.31	Number of non-dominated solutions in y_{apr} for TSP benchmark instances	153
4.32	Comparison of solutions with ONVG, OTNVG and OTNVGR of Kroab50 and Kroac50 TSP instances	153
4.33	Comparison of solutions with ONVG, OTNVG and OTNVGR of Kroabc50 and Kroabcd50 TSP instances	153
4.34	Comparison of solutions with ONVG, OTNVG and OTNVGR of Kroab100 and Kroabc100 TSP instances	153

ABBREVIATIONS

ACO	Ant Colony Optimization
ACOMOFS	Efficient Ant Colony Optimization for Multi Objective Flow Shop Scheduling
ACS	Ant Colony System
ARPD	Average Relative Percentage Deviation
AS	Ant System
CO	Combinatorial Optimization
CPACO	Crowding Population-based Ant Colony Optimization
ER	Error Ratio
GA	Genetic Algorithm
HAMC	Hybrid Algorithms for Multi Criterion
HVR	Hyper Volume Ratio
JSSP	Job Shop Scheduling Problem
MACS	Multiple Ant Colony System
MCAA	Multi Colony Ant Algorithm
MOACO	Multi Objective Ant Colony Optimization
MOACSA	Multi Objective Ant Colony System Algorithm
MOOP	Multi Objective Optimization Problem
MOTSP	Multi Objective Traveling Salesman Problem
mQAP	Multi-objective Quadratic Assignment Problem
NP	Non-deterministic Polynomial Time
ONVG	Overall Non-dominated Vector Generation
OTNVG	Overall True Non-dominated Vector Generation
OTNVGR	Overall True Non-dominated Vector Generation Ratio
P	Polynomial Time

PSACO	P areto S trength A nt C olony O ptimization
QAP	Q uadratic A ssignment P roblem
RPD	R elative P ercentage D eviation
RWACO	R andom W eight-based A nt C olony O ptimization
SA	S imulated A nneling
TS	T abu S earch
TSP	T raveling S alesman P roblem
VRPTW	V ehicle R outing P roblem with T ime W indows

ACKNOWLEDGEMENTS

I am especially grateful to Dr T.G.I. Fernando, for providing his valuable guidance on making this research a success. In addition to that, Professor Arjuna Parakrama deserves my sincere thanks for proofreading my papers, as do the professors and doctors of internal and external examiners.

I also thank my beloved mother and husband and the academic and non-academic staff of the Department of Computer Science of the University of Sri Jayewardenepura who gave their fullest cooperation during the time of my M.Phil. research, and to my employer the Vice Chancellor, The Open University of Sri Lanka.

RANDOM WEIGHT-BASED ANT COLONY OPTIMIZATION ALGORITHM FOR THE MULTI-OBJECTIVE OPTIMIZATION PROBLEMS

I.D.I.D. Ariyasingha

ABSTRACT

This thesis concentrates on analysing, improving and developing ant colony optimization algorithms for solving multiple objectives. Initially, the thesis reviews the recently proposed ant colony optimization algorithms which have been introduced for optimizing multiple objectives simultaneously. Then, it analyses the performances of these multi-objective ant colony optimization (MOACO) algorithms when applied to some combinatorial optimization problems.

Therefore, the performances of the MOACO algorithms are analysed by applying them to several benchmark instances of the traveling salesman problem (TSP). It considers various objectives, such as two, three and four objectives, and changes the number of ants and number of iterations for understanding their effects on the performances of MOACO algorithms. The results of the detailed analysis have shown that some MOACO algorithms achieve better performances. Also, their performance slightly depends on the number of ants, the number of objectives and the number of iterations used in the colony.

Then, the thesis analyses the performance of some recent multi-objective ant colony optimization algorithms by applying them to sixteen benchmark problem instances of the job shop scheduling problem (JSSP) on up to $20 \text{ jobs} \times 5 \text{ machines}$. Also, it considers two, three, and four objectives and it optimizes four criteria – makespan, mean flow time, mean tardiness, and mean machine idle time – simultaneously. Furthermore, different numbers of ants are used in a colony, to see their effects on the performance of the algorithms. The results obtained have shown that the performance of some multi-objective ant colony optimization algorithms depend on the number of objectives and the number of ants.

This thesis proposes a new ant colony optimization (ACO) algorithm named the random weight-based ant colony optimization algorithm (RWACO) to solve multiple objectives simultaneously. The RWACO algorithm is based on the ant colony system (ACS) algorithm and uses randomly generated weights for each objective associated with heuristic information which is called the random weight-based method. Each weight finds solutions in the specific part of the Pareto front. RWACO algorithm uses different weights for each objective in each iteration. Finally, the randomly generated weights defined for heuristic information, obtain solutions by covering the whole Pareto-optimal front.

The performance of the newly proposed algorithm, RWACO, is evaluated by comparing it with more recent MOACO algorithms when applied to the three combinatorial optimization problems: the traveling salesman problem (TSP), the job shop scheduling problem (JSSP) and the quadratic assignment problem (QAP). According to the results obtained with these studies, it has been shown that the RWACO algorithm achieves better performances than all the other MOACO algorithms considered in these studies. Furthermore, the RWACO algorithm obtains well-distributed Pareto front than all the other MOACO algorithms considered in this thesis.

According to the results obtained in this thesis, the Pareto strength ant colony optimization (PSACO) algorithm obtains better performance for the TSP and the JSSP. However, it obtains the third best performance for the TSP and the JSSP and returns the solutions only in the central part of the Pareto front. Therefore, the random weight-based method, which has been introduced for the RWACO algorithm, is applied to the PSACO algorithm to examine its effects on the performance of the PSACO algorithm. The performance is evaluated by applying it to the traveling salesman problem. The experimental results have shown that the PSACO algorithm performs better when the random weight-based method is applied. Also, the new PSACO algorithm obtains the well-distributed Pareto front and the distributed solutions than the original PSACO algorithm.

Keywords: ant colony system, ant system, heuristic information, job shop scheduling problem, multi-objective ant colony optimization algorithm, multi-objective problem, non-dominated solution, pareto front, pheromone trail, quadratic assignment problem, random weight-based ant colony optimization algorithm, traveling salesman problem.