

Multi-Objective Optimisation using the Bees Algorithm

A thesis submitted to Cardiff University
in candidature for the degree of

Doctor of Philosophy

by

Ji Young Lee, B.Eng., M. Eng.

Manufacturing Engineering Centre
School of Engineering
Cardiff University
United Kingdom

December 2010

UMI Number: U585417

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



UMI U585417

Published by ProQuest LLC 2013. Copyright in the Dissertation held by the Author.
Microform Edition © ProQuest LLC.

All rights reserved. This work is protected against
unauthorized copying under Title 17, United States Code.



ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

ABSTRACT

In the real world, there are many problems requiring the best solution to satisfy numerous objectives and therefore a need for suitable Multi-Objective Optimisation methods.

Various Multi-Objective solvers have been developed recently. The classical method is easily implemented but requires repetitive program runs and does not generate a true “Pareto” optimal set. Intelligent methods are increasingly employed, especially population-based optimisation methods to generate the Pareto front in a single run.

The Bees Algorithm is a newly developed population-based optimisation algorithm which has been verified in many fields. However, it is limited to solving single optimisation problems. To apply the Bees Algorithm to a Multi-Objective Optimisation Problem, either the problem is converted to single objective optimisation or the Bees Algorithm modified to function as a Multi-Objective solver.

To make a problem into a single objective one, the weighted sum method is employed. However, due to failings of this classical method, a new approach is developed to generate a true Pareto front by a single run.

This work also introduces an enhanced Bees Algorithm. A new dynamic selection procedure improves the Bees Algorithm by reducing the number of parameters and new neighbourhood search methods are adopted to optimise the Pareto front.

The enhanced algorithm has been tested on Multi-Objective benchmark functions and the classical Environmental/Economic power Dispatch Problem (EEDP). The results obtained compare well with those produced by other population-based algorithms.

Due to recent trends in renewable energy systems, it is necessary to have a new model of the EEDP. Therefore, the EEDP was amended in conjunction with the Bees Algorithm to identify the best design in terms of energy performance and carbon emission reduction by adopting zero and low carbon technologies. This computer-based tool supports the decision making process in the design of a Low-Carbon City.

Dedicated to my loving parents

ACKNOWLEDGEMENTS

I would like to thank my supervisor Professor **Duc Truong Pham** for his excellent supervision, continuous encouragement and support. He deserves more thanks than I can express for the amount of time he gave me, and for making that time the most rewarding hours of my academic life.

I would also like to thank the members and staff of the MEC, especially Dr **Michael Packianather** and **Celia Rees** for their support throughout my research work. My thanks are also due to all my **Cardiff Bay Bees** colleagues, especially Dr **Marco Castellani** and Dr **Ahmed Haj Darwish**, who contributed to my work through their valuable help, advice and technical support.

Furthermore, special thanks also go to Professor **Jin-Seok Oh** and the members of the **E2E Lab.** at Korea Maritime University for their support. I also want to thank Dr **Jae-Min Kim** and **Roger Judd** for their valuable help.

Finally, I reserve my deepest gratitude for my parents, **Sang-Min Lee** and **Mi-Ok Oh** who gave and have given me continuous support and encouragement throughout my studies. Many thanks also to my sisters **Mi-Jung Lee** and **Hey-Jung Lee**, my brother **Neung-Hui Lee** and special thanks goes to my sister **Sun-Young Lee**.

DECLARATION

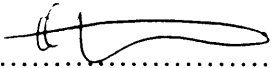
This work has not previously been accepted in substance for any degree and is not concurrently submitted in candidature for any degree.

Signed  (Ji Young Lee)

Date.....21/12/2010

Statement 1

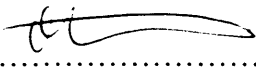
This thesis is being submitted in partial fulfilment of the requirements for the degree of PhD.

Signed  (Ji Young Lee)

Date.....21/12/2010

Statement 2

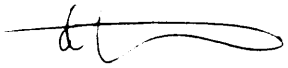
This thesis is the result of my own independent work/investigations, except where otherwise stated. Other sources are acknowledged by explicit references. A bibliography is appended.

Signed  (Ji Young Lee)

Date.....21/12/2010

Statement 3

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed  (Ji Young Lee)

Date.....21/12/2010

CONTENTS

ABSTRACT.....	i
ACKNOWLEDGEMENT.....	iv
DECLARATION	v
CONTENTS.....	vi
LIST OF FIGURES.....	xi
LIST OF TABLES.....	xvi
ABBREVIATIONS.....	xix
NOMENCLATURE.....	xxiii
1 INTRODUCTION	1
1.1 Motivation.....	1
1.2 Research Aim and Objectives	3
1.3 Outline of the Thesis	5
2 MULTI-OBJECTIVE OPTIMISATION USING SWARM-BASED OPTIMISATION ALGORITHMS.....	7
2.1 Multi-Objective Optimisation (MOO).....	7
2.2 Multi-Objective solvers.....	10
2.2.1 Classical methods	10
2.2.1.1 Weighted sum method.....	13
2.2.2 Intelligent methods	14

2.2.2.1	Population-based Multi-Objective solvers	14
2.2.2.1.1	Genetic Algorithms (GAs).....	17
2.2.2.1.2	Ant algorithm.....	26
2.2.2.1.3	Particle Swarm Optimisation (PSO).....	31
2.2.2.1.4	Artificial Immune System (AIS).....	36
2.2.2.1.5	Bacteria Foraging Optimisation Algorithm (BFOA).....	41
2.2.2.1.6	Bee-inspired Algorithms.....	45
2.3	Summary.....	58

3 THE PARETO-BASED BEES ALGORITHM WITH MEMORISED

SOLUTIONS FOR MULTI-OBJECTIVE PROBLEMS..... 59

3.1	Preliminaries	59
3.2	Applications	60
3.2.1	The Environmental/Economic Dispatch Problem (EEDP)	60
3.2.1.1	Objective functions.....	64
3.2.1.2	Constraints.....	65
3.2.1.3	Multi-Objective formulation	65
3.2.1.4	System parameters.....	66
3.2.2	Benchmark mathematical Multi-Objective functions	66
3.3	The Pareto-based Bees Algorithm with memorised solutions	69
3.4	Results.....	73
3.4.1	Results of the EEDP	73
3.4.1.1	Comparison of Multi-Objective Optimisation Algorithms ..	74
3.4.1.2	Effect of patch size	81
3.4.1.3	Effect of number of memorised solutions	85
3.4.1.4	Comparison of the number of solutions and Pareto fronts.....	88
3.4.2	Results of benchmark mathematical functions.....	92

3.5	Summary	98
-----	---------------	----

4 AN ENHANCED BEES ALGORITHM FOR MULTI-OBJECTIVE

PROBLEMS..... 99

4.1	Preliminaries	99
-----	---------------------	----

4.2	Applications	100
-----	--------------------	-----

4.2.1	The Environmental/Economic Dispatch Problem (EEDP)	100
-------	--	-----

4.2.2	Benchmark mathematical Multi-Objective functions	100
-------	--	-----

4.3	An enhanced Bees Algorithm for solving Multi-Objective Optimisation Problems.....	109
-----	--	-----

4.3.1	Enhancements.....	109
-------	-------------------	-----

4.3.1.1	Global enhancements	109
---------	---------------------------	-----

4.3.1.2	Local enhancements	110
---------	--------------------------	-----

4.3.1.2.1	Basic neighbourhood search (basicNGH) method.....	111
-----------	---	-----

4.3.1.2.2	Random and weighted sum neighbourhood search (randomNGH and wsNGH) methods.....	115
-----------	--	-----

4.3.2	The proposed algorithms	119
-------	-------------------------------	-----

4.3.2.1	The Single-Cycled Bees Algorithm (SCBA)	119
---------	---	-----

4.3.2.2	The Multi-Cycled Bees Algorithm (MCBA)	122
---------	--	-----

4.4	Results.....	125
-----	--------------	-----

4.4.1	Results of the EEDP	125
-------	---------------------------	-----

4.4.1.1	Comparison of two extreme Pareto solutions.....	128
---------	---	-----

4.4.1.1.1	Comparison of Multi-Objective Optimisation Algorithms.....	128
-----------	---	-----

4.4.1.1.2	Effect of neighbourhood search methods	140
-----------	--	-----

4.4.1.1.3	Effect of settings	145
-----------	--------------------------	-----

4.4.1.2	Additional comparisons.....	151
---------	-----------------------------	-----

4.4.1.3	Comparison between WSBA- <i>m</i> and MCBA-wsNGH methods.....	162
4.4.2	Results of benchmark mathematical functions.....	167
4.5	Summary	175

5 OPTIMISATION OF MULTI-FUEL ENERGY SYSTEMS FOR A

LOW-CARBON CITY USING THE BEES ALGORITHM 177

5.1	Preliminaries	177
5.2	The proposed system model.....	183
5.3	Proposed techniques.....	189
5.3.1	The Simultaneous Technique (ST).....	189
5.3.2	The Sequential Technique with Multi-Optimisation (STMO).....	197
5.3.2.1	Equations for the Pareto (a) front	198
5.3.2.2	Equations for the Pareto (d) front	202
5.3.3	The Sequential Technique with Single Optimisation (STSO).....	203
5.3.4	System parameters.....	204
5.4	Results.....	209
5.4.1	Comparison of two extreme solutions.....	209
5.4.1.1	Effects of techniques	212
5.4.1.1.1	Setting1	212
5.4.1.1.2	Setting2	213
5.4.1.2	Effect of settings.....	213
5.4.2	Comparison of Pareto fronts.....	218
5.4.3	Additional comparisons.....	232
5.5	Summary	237

6	CONCLUSION	239
6.1	Contributions.....	239
6.2	Conclusions.....	240
6.3	Suggestions for future research.....	242
	APPENDIX	243
	REFERENCES.....	268

LIST OF FIGURES

Figure 2.1 Illustration of the Pareto concept for a bi-objective optimisation problem.....	11
Figure 2.2 Taxonomy of Multi-Objective solvers.....	12
Figure 2.3 Pareto front generation using population-based techniques	16
Figure 2.4 Simplified flowchart of a Genetic Algorithm	19
Figure 2.5 Genetic Algorithm for Multi-Objective Optimisation Problems.....	21
Figure 2.6 Flowchart of Ant Colony Optimisation algorithm.....	30
Figure 2.7 Simplified flowchart of Particle Swarm Optimisation algorithm	33
Figure 2.8 Pseudo code of Immune System model (fitness scoring)	39
Figure 2.9 Simplified flowchart of Bacterial Foraging Optimisation Algorithm	44
Figure 2.10 Bee behaviour and algorithms	47
Figure 2.11 Flowchart of food processing procedure of honey bees in nature for one season.....	52
Figure 2.12 Flowchart of the Bees Algorithm.....	53
Figure 3.1 Single-line diagram of IEEE 30-bus test system	63
Figure 3.2 Illustration of the concept for the Bees Algorithm with memorised solutions.....	71
Figure 3.3 Flowchart of the proposed Pareto-based Bees Algorithm with memorised solutions.....	72
Figure 3.4 The result of the EEDP	90

Figure 3.5 The results by WSBA(20).....	95
Figure 3.6 The Pareto fronts by WSBA- e (20)	96
Figure 3.7 The Pareto fronts by WSBA- m (20)	97
Figure 4.1 Flowchart for the basicNGH method.....	113
Figure 4.2 The concept of basicNGH method.....	114
Figure 4.3 The concept for the both randomNGH and wsNGH methods.....	117
Figure 4.4 The flowchart for the both randomNGH and wsNGH methods	118
Figure 4.5 The flowchart of the Single-Cycled Bees Algorithm (SCBA)	121
Figure 4.6 The concept of the Multi-Cycled Bees Algorithm (MCBA)	123
Figure 4.7 The flowchart of the Multi-Cycled Bees Algorithm (MCBA)	124
Figure 4.8 Box plot of the best cost (f_c) regarding neighbourhood search	141
Figure 4.9 Box plot of the best emissions (f_e) regarding neighbourhood search methods	143
Figure 4.10 Box plot of the best cost (f_c) regarding settings	148
Figure 4.11 Box plot of the best emission (f_e) regarding settings	149
Figure 4.12 Box plots of the number of evaluations	154
Figure 4.13 Box plots of program running times	156
Figure 4.14 Box plots of the number of Pareto solutions.....	158
Figure 4.15 Pareto fronts regarding neighbourhood search methods.....	160
Figure 4.16 Comparison of Pareto fronts from wsNGH in setting1	161

Figure 4.17 Box plots of the best cost between the WSBA- $m(22)$ and the MCBA-wsNGH.....	164
Figure 4.18 Comparison between the WSBA- $m(22)$ and the MCBA-wsNGH	165
Figure 4.19 Comparison of the Pareto fronts between the WSBA- $m(22)$ (or m -Pareto(22)) and the MCBA-wsNGH.....	166
Figure 4.20 Pareto fronts of convex functions without constraints.....	171
Figure 4.21 Pareto front of non-convex function without constraints	172
Figure 4.22 Pareto front of three-objective function without constraints	172
Figure 4.23 Pareto fronts of functions with constraints	173
Figure 4.24 Comparison of Pareto fronts between the WSBA- $m(20)$ (or m -Pareto(20)) and the SCBA-randomNGH	174
Figure 5.1 Outline plan for ASAN new town	186
Figure 5.2 Overview of the procedure for this work.....	187
Figure 5.3 Multi-fuel energy system configuration.....	188
Figure 5.4 System structure of Simultaneous Technique (ST)	191
Figure 5.5 Flowchart of the Simultaneous Technique (ST) using the Bees Algorithm	192
Figure 5.6 Structure of the Sequential Technique with Multi-Optimisation (STMO)	199
Figure 5.7 Concept of the Sequential Technique with Multi-Optimisation (STMO)	200

Figure 5.8 Flowchart of the Sequential Technique with Multi-Optimisation (STMO) using the Bees Algorithm	201
Figure 5.9 Structure of the Sequential Technique with Single Optimisation (STSO).....	205
Figure 5.10 Concept of the Sequential Technique with Single Optimisation (STSO).....	206
Figure 5.11 Flowchart of the Sequential Technique with Single Optimisation (STSO) using the Bees Algorithm.....	207
Figure 5.12 Box plot of the best cost (f_c) regarding techniques in setting2	216
Figure 5.13 Pareto fronts of the Simultaneous Technique (ST).....	220
Figure 5.14 Individual proportions of the energy system and CO ₂ reduction (%) with the Simultaneous Technique (ST) setting1	221
Figure 5.15 Individual proportions of the energy system and CO ₂ reduction (%) with the Simultaneous Technique (ST) setting2	222
Figure 5.16 Pareto fronts of the Sequential Technique with Multi-Optimisation (STMO)	223
Figure 5.17 Individual proportions of the energy system and CO ₂ reduction (%) with the STMO setting1	224
Figure 5.18 Individual proportions of the energy system and CO ₂ reduction (%) with the STMO setting2	225
Figure 5.19 Pareto fronts of Sequential Technique with the Single Optimisation (STSO).....	226

Figure 5.20 Individual proportions of the energy system and CO₂ reduction (%)
with the STSO setting1..... 227

Figure 5.21 Individual proportions of the energy system and CO₂ reduction (%)
with the STSO setting2..... 228

Figure 5.22 Comparison of Pareto fronts of three techniques in setting1 229

Figure 5.23 Individual proportions of the energy system with three techniques
..... 230

Figure 5.24 Comparison of Pareto fronts between the ST and partial Pareto front
of thermal systems which exceeds electrical demand 231

Figure 5.25 Box plots of results with setting1 235

Figure 5.26 Box plots of results with setting2..... 236

LIST OF TABLES

Table 2.1 Applications of the Bees Algorithm.....	48
Table 3.1 Coefficients for the test system	67
Table 3.2 Equations for Multi-Objective mathematical function optimisation...	68
Table 3.3 Parameters for the Environmental/Economic Dispatch Problem.....	75
Table 3.4 Test settings.....	76
Table 3.5 Symbols used in Table 3.6 to 3.12	78
Table 3.6 Comparison of results between MOPSO and the proposed Bees Algorithm for finding the best fuel cost (f_c)	79
Table 3.7 Comparison of results between MOPSO and the proposed Bees Algorithm for finding the best NOx emissions (f_e)	80
Table 3.8 Comparison of results between NSGA- II , SPEA and the proposed Bees Algorithms ($itr=10$) for the best cost (f_c).....	82
Table 3.9 Comparison of results regarding neighbourhood size for the best cost (f_c).....	83
Table 3.10 Comparison of results regarding neighbourhood size for the best emissions (f_e)	84
Table 3.11 Comparison of results regarding memorised solutions for the best cost (f_c).....	86

Table 3.12 Comparison results regarding memorised solutions for the best emissions (f_e)	87
Table 3.13 The average number of solutions for each 20 independent runs	89
Table 3.14 Parameters of the Bees Algorithm for benchmark mathematical functions	93
Table 3.15 Results for benchmark mathematical functions	94
Table 4.1 Convex functions without constraints	104
Table 4.2 Non-convex functions without constraints.....	105
Table 4.3 Functions with constraints.....	107
Table 4.4 Three-functions optimisation	108
Table 4.5 Parameter settings for the SCBA-basicNGH and the SCBA-randomNGH	126
Table 4.6 Parameter settings for the SCBA-wsNGH and the MCBA-wsNGH	127
Table 4.7 Comparison between the SCBA-basicNGH, the SCBA-randomNGH and MOPSO	130
Table 4.8 Comparison between the SCBA-basicNGH, the SCBA-randomNGH and NSGA- II	131
Table 4.9 Comparison between the SCBA-basicNGH, the SCBA-randomNGH and SPEA	133
Table 4.10 Comparison between the SCBA-wsNGH and MOPSO.....	134
Table 4.11 Comparison between the SCBA-wsNGH, the MSBA-wsNGH and MOPSO	135

Table 4.12 Comparison between NSGA- II , SPEA and the SCBA-wsNGH ...	138
Table 4.13 Comparison between NSGA- II , SPEA, the SCBA-wsNGH and the MSBA-wsNGH	139
Table 4.14 The result of statistic analyses regarding settings by SPSS.....	147
Table 4.15 The average number of evaluations, program running times and number of Pareto solutions.....	153
Table 4.16 The result of statistical analyse of two extreme Pareto solutions ...	164
Table 4.17 Parameters for the Bees Algorithm	168
Table 4.18 The result of the number of evaluation and Pareto solutions.....	169
Table 5.1 Peak demands and total roof areas	188
Table 5.2 Limit of each system	195
Table 5.3 Parameters of the proposed multi-fuel energy system	208
Table 5.4 Parameter setting for the Bees Algorithm	210
Table 5.5 The average of two extreme solutions from 10 independent runs ...	211
Table 5.6 Statistical result in setting1	214
Table 5.7 Statistical results in setting2.....	215
Table 5.8 Statistical result from <i>t-test</i> regarding setting	217
Table 5.9 The average values from 10 independent program runs	234

ABBREVIATIONS

ABC	Artificial Bee Colony algorithm
A-BCO	Autonomous Bee Colony Optimisation algorithm
ACO	Ant Colony Optimisation
AI	Artificial Intelligence
AIS	Artificial Immune System
AIWF	Adaptive Inertia Weight Factor
ANOVA	Analysis of variance
ART	Automated Red Teaming
AS	The Ant System
AWS	The Adaptive Weighted-Sum method
BA	The Bees Algorithm
BAMOP	The Bees Algorithm-assisted Multi objective Optimisation Problems
BCMOA	Bacterial Chemotaxis Multi-Objective Optimisation Algorithm
BCO	Bee Colony Optimization algorithm
BFOA	Bacteria Foraging Optimisation Algorithm
BS	Bees System
BSO	Bee Swarm Optimisation
CHP	Combined Heat and Power
CLS	Chaotic Local Search
CO ₂	Carbon dioxide
COP	Coefficient Of Performance
DM	Decision Maker
EA/EAs	Evolutionary Algorithm/Evolutionary Algorithms

EDAS	Entropy-based Density Assessment Scheme
EEDP	The Environmental/Economic Dispatch Problem
EMOIA	Evolutionary Multi-Objective Immune Algorithm
EMOO	Evolutionary Multi-Objective Optimisation
ESRU	Energy Systems Research Unit
FACTS	Flexible AC Transmission System devices
FCM	Fuzzy C-Means clustering
FON	Fonseca and Fleming's study
GA/GAs	Genetic Algorithm/Genetic Algorithms
HBF	Honey Bee Foraging algorithm
HBMO	Honey-Bees Mating Optimisation algorithm
IEEE	Institute of Electrical and Electronics Engineers
KRW	Korea currency (won)
KUR	Kursawe's study
MACS	Multiple Ant Colony System
MBA	Multiobjective Bees Algorithm
MBO	Marriage Process in Honey Bees algorithm
MCBA-wsNGH	The Multi-Cycled Bees Algorithm with weighted sum neighbourhood search method
MG	Micro-Grid
MIMO MC-CDMA	Multiple Input Multiple Output Multi Carrier Code Division Multiple Access system
MO	Multi-Objective
MOAIS	Multi-Objective Artificial Immune System
MOBA	The Multi-Objective Bees Algorithm
MOBCO	Multi-Objective Bee Colony Optimisation algorithm
MODEST	Model for Optimisation of Dynamic Energy System with Time-Dependent Components and Boundary Conditions

MOEA/MOEAS	Multi-Objective Evolutionary Algorithm/Multi-Objective Evolutionary Algorithms
ε -MOEA	ε -Multi-Objective Evolutionary Algorithm
MOGA	Multi-Objective Genetic Algorithm
MOO	Multi-Objective Optimisation
MOOP/MOOPs	Multi-Objective Optimisation Problem/Multi-Objective Optimisation Problems
MOPSO	Multi-Objective Particle Swarm Optimisation
NPGA	Niched-Pareto Genetic Algorithm
NO _x	Generic term for mono-nitrogen oxides (NO and NO ₂)
NSGA	Non-dominated Sorting Genetic Algorithm
NSGA- II	Non-dominated Sorting Genetic Algorithm- II
PACO	Population-based Ant Colony Optimisation
PAES	Pareto Archived Evolution Strategy algorithm
PCB	Printed Circuit Board
POL	Poloni's study
PSO	Particle Swarm Optimisation
PV	Photovoltaic
QMOO	Queuing Multi-Objective Optimiser
RE	Renewable Energy
RES	Renewable Energy Sources
RT	The product of the molar gas constant R and the temperature T (The SI units for RT are joules per mole (J/mol))
SCBA- basicNGH	The Single-Cycled Bees Algorithm with basic neighbourhood search method
SCBA- randomNGH	The Single-Cycled Bees Algorithm with random neighbourhood search method
SCBA-wsNGH	The Single-Cycled Bees Algorithm with weighted sum

	neighbourhood search method
SCH	Schaffer's study
SO	Single-Objective
SOOP/SOOPs	Single Objective Optimisation Problem/Single Objective Optimisation Problems
SPEA	Strength Pareto Evolutionary Algorithm
SPEA 2	Strength Pareto Evolutionary Algorithm 2
SPSS	Statistical Package for the Social Sciences
ST	The Simultaneous Technique
STMO	The Sequential Technique with Multi-Optimisation
STSO	The Sequential Technique with Single Optimisation
TNK	Tanaka et. al.'s study
TSP	Travelling Salesman Problem
VBA	Virtual Bee Algorithm
VEABC	Vector Evaluated Artificial Bee Colony algorithm
VEGA	Vector-Evaluated Genetic Algorithm
VEPSO	Vector Evaluated Particle Swarm Optimisation
WBMOAIS	Weight-Based Multi-Objective Artificial Immune System
WSBA	Weighted Sum Bees Algorithm
WSBA- <i>e</i>	Weighted Sum Bees Algorithm with <i>e</i> sites
WSBA- <i>m</i>	Weighted Sum Bees Algorithm with <i>m</i> sites
ZEH	Zero Energy House
ZDT1	Zitzler et. al.'s study
ZDT2	Zitzler et. al.'s study
ZDT3	Zitzler et. al.'s study

NOMENCLATURE

Chapter 5

Symbol	Unit or value	Description
i	-	Index of electric power generator ($i = 1, \dots, N_G$)
j	-	Index of thermal generator ($j = 1, \dots, N_T$)
k	-	Index of heat & power generator ($k = 1, \dots, N_{GT}$)
p	-	Index of Pareto optimal solution ($p=1, \dots, S_{PO}$)
S_{PO}	-	The number of Pareto optimal solutions
N	-	The number of generator <ul style="list-style-type: none"> • N_G: number of electric power systems ($N_G = 2$: G1 and G2) • N_T: number of thermal system ($N_T = 4$: T1, T2, T3 and T4) • N_{GT}: number of heat & power system ($N_{GT} = 2$: GT1 and GT2)
f_{CALL}	KRW	The total capital cost of whole systems
UC_{Gi}	KRW/KW	The unit cost of electric power system i
C_{Gi}	MW _e	The capacity of electric power system i <ul style="list-style-type: none"> • C_{G1}: capacity of grid power • C_{G2}: capacity of PV
UC_{Tj}	KRW/KW	The unit cost of thermal system j
C_{Tj}	MW _e	The capacity of thermal system j <ul style="list-style-type: none"> • C_{T1}: capacity of solar thermal collector • C_{T2}: capacity of boiler • C_{T3}: capacity of geothermal heat pump

		<ul style="list-style-type: none"> • c_{r2}: capacity of air source heat pump
UC_{Gk}	KRW/KW	The unit cost of heat & power system k
C_{Gk}	MW _e	The capacity of heat & power system k <ul style="list-style-type: none"> • c_{gr1}: capacity of CHP • c_{gr2}: capacity of fuel cell
f_{eALL}	Ton	Total CO ₂ emission of whole systems
CO_2F	-	CO ₂ factor <ul style="list-style-type: none"> • CO_2F_{Gi}: CO₂ factor of electric power system i • CO_2F_{Tj}: CO₂ factor of thermal system j • CO_2F_{Gk}: CO₂ factor of heat & power system k
T_{Tj}	MW _{th}	Produced thermal from thermal generator j
T_{Gk}	MW _{th}	Produced thermal from heat & power system k
T_{DT}	MW _{th}	Peak thermal demand
T_{LT}	MW _{th}	Total thermal transmission losses
β_{Tj}	0 ~ 1	Thermal efficiency of thermal generator j <ul style="list-style-type: none"> • β_{r1}: efficiency of solar thermal collector • β_{r2}: efficiency of boiler • β_{r3}: efficiency of geothermal heat pump • β_{r4}: efficiency of air source heat pump
COP	-	The coefficient of performance <ul style="list-style-type: none"> • COP_{r3}: coefficient of performance of geothermal heat pump • COP_{r4}: The coefficient of performance of air source heat pump
T_{Gk}	MW _{th}	Produced thermal from heat & power system k
β_{Gk}	0 ~ 1	Thermal efficiency of heat & power system k

		<ul style="list-style-type: none"> • β_{GT_1} : thermal efficiency of CHP • β_{GT_2} : thermal efficiency of fuel cell
PHR_{GTh}	-	Power/heat ratio of heat & power system k <ul style="list-style-type: none"> • PHR_{GT_1} : power/heat ratio of CHP • PHR_{GT_2} : power/heat ratio of fuel cell
P_{Gi}	MW_e	Generated electricity from electric generator i
P_{GTh}	MW_e	Generated electricity from heat & power system k
P_{DE}	MW_e	Peak electricity demand
RP_{T3}	MW_e	Required electricity for geothermal heat pump input
RP_{T4}	MW_e	Required electricity for air source heat pump input
P_{LE}	MW_e	Total electricity transmission losses
α_{Gi}	0 ~ 1	Electric efficiency of electric generator i <ul style="list-style-type: none"> • α_{G1} : electric efficiency of grid power • α_{G2} : electric efficiency of PV
α_{GTh}	0 ~ 1	Electric efficiency of heat & power system k <ul style="list-style-type: none"> • α_{GT_1} : electric efficiency of CHP • α_{GT_2} : electric efficiency of fuel cell
C_{Tjmin}	MW_{th}	Minimum capacity boundary of thermal generator j <ul style="list-style-type: none"> • C_{T1min} : minimum capacity boundary of solar thermal collector • C_{T2min} : minimum capacity boundary of boiler • C_{T3min} : minimum capacity boundary of geothermal heat pump • C_{T4min} : minimum capacity boundary of air source heat pump
C_{Tjmax}	MW_{th}	Maximum capacity boundary of thermal generator j

		<ul style="list-style-type: none"> • $C_{T1\max}$: maximum capacity boundary of solar thermal collector • $C_{T2\max}$: maximum capacity boundary of boiler • $C_{T3\max}$: maximum capacity boundary of geothermal heat pump • $C_{T4\max}$: maximum capacity boundary of air source heat pump
tba	m^2	Total buildings' roof area
rua_{stc}	m^2/KW	Required unit areas for installation of solar thermal collectors
rua_{pv}	m^2/KW	Required unit areas for installation of PV panels
$allowance$	$0 \sim 1$	Allowance of installation of renewable systems from maximum boundary capacity <ul style="list-style-type: none"> • $allowance_{e_{T1}}$: allowance of solar thermal collector • $allowance_{e_{T3}}$: allowance of geothermal heat pump • $allowance_{e_{T4}}$: allowance of air source heat pump • $allowance_{e_{GT2}}$: allowance of fuel cell • $allowance_{e_{G2}}$: allowance of PV
$C_{GTk\min}$	MW_{th}	Minimum capacity boundary of heat & power system k <ul style="list-style-type: none"> • $C_{GT1\min}$: minimum capacity boundary of CHP • $C_{GT2\min}$: minimum capacity boundary of fuel cell
$C_{GTk\max}$	MW_{th}	Maximum capacity boundary of heat & power system k <ul style="list-style-type: none"> • $C_{GT1\max}$: maximum capacity boundary of CHP • $C_{GT2\max}$: maximum capacity boundary of fuel cell

C_{Gimin}	MW _e	Minimum capacity boundary of electric generator i <ul style="list-style-type: none"> • C_{G1min} : minimum capacity boundary of grid power • C_{G2min} : minimum capacity boundary of PV
C_{Gimax}	MW _e	Maximum capacity boundary of electric generator i <ul style="list-style-type: none"> • C_{G1max} : maximum capacity boundary of grid power • C_{G2max} : maximum capacity boundary of PV
RP_{T3max}	MW _e	Maximum required electricity for geothermal heat pump input
RP_{T4max}	MW _e	Maximum required electricity for air source heat pump input
$CO2RR_p$	%	Percentage of CO ₂ reduction of Pareto optimal solution p
E_{allp}	Ton	Total CO ₂ emission of Pareto optimal solution p
$ACO2$	kg	Total absolute CO ₂ quantity for conventional systems (boiler and grid power)
f_{cTH}	KRW	Total capital cost of thermal and heat & power systems
f_{eTH}	Ton	Total CO ₂ emission of thermal and heat & power systems

1 INTRODUCTION

1.1 Motivation

In the real world, there are many problems requiring the best solution to satisfy numerous objectives where single-objective optimisation is not ideal, hence the need for methods such as Multi-Objective Optimisation (MOO) to solve these problems. MOO (also called multi-criteria optimisation, multi-performance or vector optimisation) can then be defined according to Coello et al. (Coello et al. 2007) as the search for:

a vector of decision variables which satisfies constraints and optimises a vector function whose elements represent the objective functions. These functions form a mathematical description of performance criteria which are usually in conflict with each other. Hence, the term “optimise” means finding such a solution which would give the values of all the objective functions acceptable to the decision maker (Coello et al. 2007, p. 5).

The solution to Multi-Objective Optimisation Problems (MOOPs) has been a challenge to researchers for a long time. Despite the considerable variety of techniques developed in Operations Research (OR) and other disciplines to tackle this problem, the complexities of its solution calls for alternative

approaches such as the population-based Genetic Algorithm (GA). This algorithm has been motivated mainly to solve a Multi-Objective Optimisation Problem (MOOP) because it allows the generation of several elements of the Pareto optimal set (i.e., the set of solutions that are Pareto “efficient”, in other word, solutions for which the value for no objective function could be improved without causing those of the remaining objective function to worsen) in a single run. The complexity of some MOOPs (i.e., very large search spaces, uncertainty, noise, disjointed Pareto curves, etc.) may prevent the use (or application) of traditional OR MOOP-solution techniques.

Recently, other population-based algorithms such as Particle Swarm Optimisation (PSO), Ant Colony Optimisation (ACO) and Artificial Immune System (AIS) have been employed to solve MOOPs. While PSO is becoming popular, ACO and AIS have not yet seen many applications. More recently, the Bacteria Foraging Algorithm (BFOA) and the Bees Algorithm have received attention as possible new tools for Multi-Objective Optimisation.

The Bees Algorithm (Pham et al. 2005) is an intelligent optimisation tool which is inspired by the natural foraging behaviour of honey bees. The algorithm employs a combination of global exploration and local exploitation. However, the Bees Algorithm was basically developed for solving a Single-Objective

Optimisation Problem (SOOP) and it requires a large number of parameters to be correctly set before it can be run.

This work introduces a number of ways to solve Multi-Objective problems using the Bees Algorithm. It also proposes different improvements to the Bees Algorithm as a Multi-Objective solver such as reducing the number of parameters needed to run the algorithm. Additional neighbourhood search techniques are also developed in order to enhance the Pareto optimal set.

1.2 Research Aim and Objectives

The aim of this research is to evaluate and validate the Bees Algorithm as a Multi-Objective solver. The specific objectives are to:

- Survey current population-based Multi-Objective solvers
- Develop new forms of the Bees Algorithm
 - To make the algorithm suitable for a MOOP
 - To enhance the Bees Algorithm by reducing the number of parameters necessary to run it
 - To enhance the Pareto optimal set

- Apply the proposed optimisation algorithms to different categories of continuous optimisation problems and compare the results obtained with other optimisation methods
- Design a multi-fuel energy system for a Low-Carbon City and optimise it using the proposed Bees Algorithm

To achieve these objectives, the following methodologies were adopted.

Literature review: the most relevant population-based optimisation algorithms will be reviewed in both the basic and Multi-Objective versions. Their advantages and disadvantages will be discussed, helping to lay the groundwork for the research.

Experiments: The performance of the new versions of the algorithm will be evaluated by computer simulation to solve a number of Multi-Objective problems. In each case, performance measures will be computed to assess the effectiveness of the new methods and comparisons with traditional methods will also be carried out.

1.3 Outline of the Thesis

The thesis is organised as follows.

Chapter 2 begins by laying out the definitions of Multi-Objective problems and Pareto optimality. MOOP solvers are enumerated as classical methods and intelligent methods. The chapter then provides an overview of the population-based Multi-Objective algorithms currently available such as the most popular Multi-Objective Evolutionary Algorithms (MOEAs) in Pareto ranking, ACO, PSO, AIS, BFOA and the Bees Algorithm. Their advantages and disadvantages are briefly discussed.

Chapter 3 describes a Multi-Objective version of the Bees Algorithm adopting the weighted sum method. Three different versions of the Bees Algorithm are introduced. The proposed algorithms are tested on two different applications: nonlinear power system optimisation (the Environmental/Economical Dispatch Problem: EEDP) and optimisation of benchmark mathematical functions. Their results are compared with those of other optimisation techniques.

Chapter 4 presents a globally and locally enhanced Multi-Objective version of the Bees Algorithm with a reduction in the number of parameters needed to run the algorithm. In order to enhance the Pareto optimal set by local enhancement of

the Bees Algorithm, three different types of neighbourhood search methods are introduced. The proposed algorithms are also applied to both the EEDP and optimisation of benchmark mathematical functions for their validation. Their results are also compared with those of other optimisation techniques.

Chapter 5 presents the application of an enhanced version of the Bees Algorithm to the design of a ‘Low-Carbon City’. The algorithm is modified for three different techniques to satisfy constraints on the balance of power and heat.

Chapter 6 presents the main contributions of this research and suggestions for future work in this field.

2 MULTI-OBJECTIVE OPTIMISATION USING SWARM-BASED OPTIMISATION ALGORITHMS

2.1 Multi-Objective Optimisation (MOO)

The majority of optimisation problems require the simultaneous optimisation of more than one objective function and it is unlikely that the different objectives would be optimised by adopting the same set of parameters. The goal of Multi-Objective Optimisation (MOO) algorithms is to generate trade-offs between objectives. Exploring all these trade-offs is particularly important because it provides the system designer/operator with the ability to understand and evaluate the different choices available to them.

The general Multi-Objective (MO) problem requiring the optimisation of N objectives may be formulated as follows (Lee and El-Sharkawi 2008; Ngatchou et al. 2005):

Minimise

$$\bar{y} = \bar{F}(\bar{x}) = [f_1(\bar{x}), f_2(\bar{x}), \dots, f_N(\bar{x})]^T \quad \text{Eq. 2.1}$$

subject to $g_j(\bar{x}) \leq 0, j = 1, 2, \dots, M,$

where

$$\vec{x} = [x_1, x_2, \dots, x_n]^T \in \Omega.$$

\vec{y} : objective vector

g_j : constraints

\vec{x} : P-dimensional vector representing the decision variables within a parameter space Ω

Ω : parameter space

The utopian solution is the solution that is optimal for all objectives and it is formulated as follows:

$$\vec{x}_0^* \in \Omega : \forall \vec{x} \in \Omega, f_i(\vec{x}_0^*) \leq f_i(\vec{x}) \quad \text{Eq. 2.2}$$

for $i \in \{1, 2, \dots, N\}$.

- $N=1$: Single-Objective (SO) problem, utopian solution = global optimum
(always exists, even if it cannot be found)
- $N>1$: The utopian solution does not generally exist rather, there are non-dominated solutions

To compare candidate solutions to MO problems, the concept of Pareto dominance and Pareto optimality are commonly used. Pareto (Pareto 1906), cited in (Kim and de Weck 2005), introduced the concept of non-inferior solutions in the context of economics and Stadler (Stadler 1979, 1984), cited in (Kim and de Weck 2005), began to apply the notion of Pareto optimality to the field of

engineering and science in the 1970s. A solution belongs to the Pareto set if there is no other solution that can improve at least one of the objectives without degrading any other objective. Formally, a decision vector $\bar{u} = [u_1, u_2, \dots, u_P]^T$ is said to be a Pareto-dominated vector $\bar{v} = [v_1, v_2, \dots, v_P]^T$, in a minimisation context, if and only if:

$$\forall i \in \{1, \dots, N\}, f_i(\bar{u}) \leq f_i(\bar{v}), \text{ and } \exists j \in \{1, \dots, N\}, f_j(\bar{u}) < f_j(\bar{v}). \quad \text{Eq. 2.3}$$

In the context of MOO, Pareto dominance is used to compare and rank decision vectors: \bar{u} dominating \bar{v} means in the Pareto sense that $\bar{F}(\bar{u})$ is either better than or the same as $\bar{F}(\bar{v})$ for all objectives, and there is at least one objective function for which $\bar{F}(\bar{u})$ is strictly better than $\bar{F}(\bar{v})$.

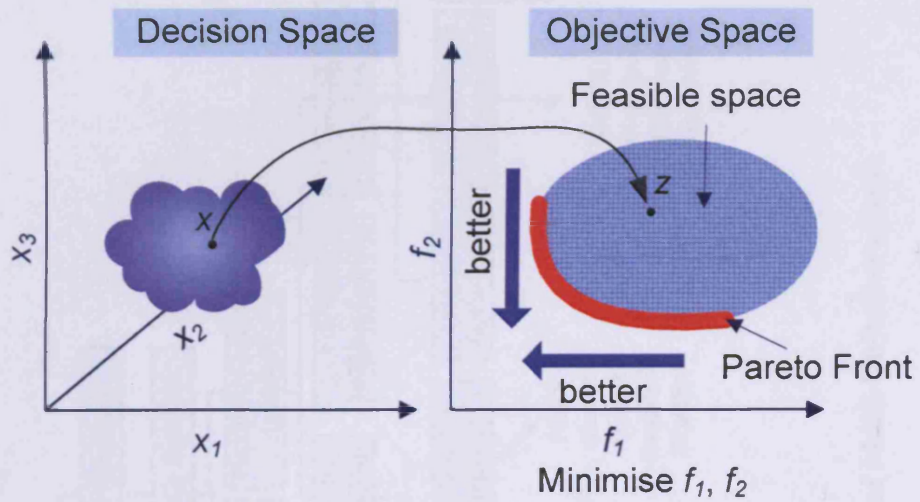
A solution \bar{a} is said to be Pareto optimal if and only if there does not exist another solution that dominates it. The corresponding objective vector $\bar{F}(\bar{a})$ is called a Pareto dominant vector, or non-inferior or non-dominated vector. All the Pareto optimal solutions are known as the Pareto optimal set. The corresponding objective vectors are said to be on the Pareto front. Figure 2.1 depicts the Pareto concept for a bi-objective minimisation problem.

2.2 Multi-Objective solvers

Solving MO problems has evolved over time. Initial classical approaches essentially consisted of converting the MO problem into a SO problem which can be solved using traditional scalar optimisation techniques. Population-based intelligent methods have more recently been used, which allows direct generation of trade-off curves in a single run. Taxonomy of MO solvers is shown in Figure 2.2.

2.2.1 Classical methods

Classical approaches have been used to convert an MO problem into an SO problem by either aggregating the objective functions or optimising the most important objective and treating the others as constraints which are geared towards finding a single solution. In order to generate a trade-off curve, the solution procedure has to be reapplied after modifying the aggregation modalities or design criteria. Although there are many classical methods, the weighted sum method is the simplest approach and probably the most widely used. Therefore only the weighted sum method is reviewed here.



- Objective space: spanned by the objective vectors
- Feasible space: the subspace of the objective vectors that satisfies the constraints

Figure 2.1 Illustration of the Pareto concept for a bi-objective optimisation problem

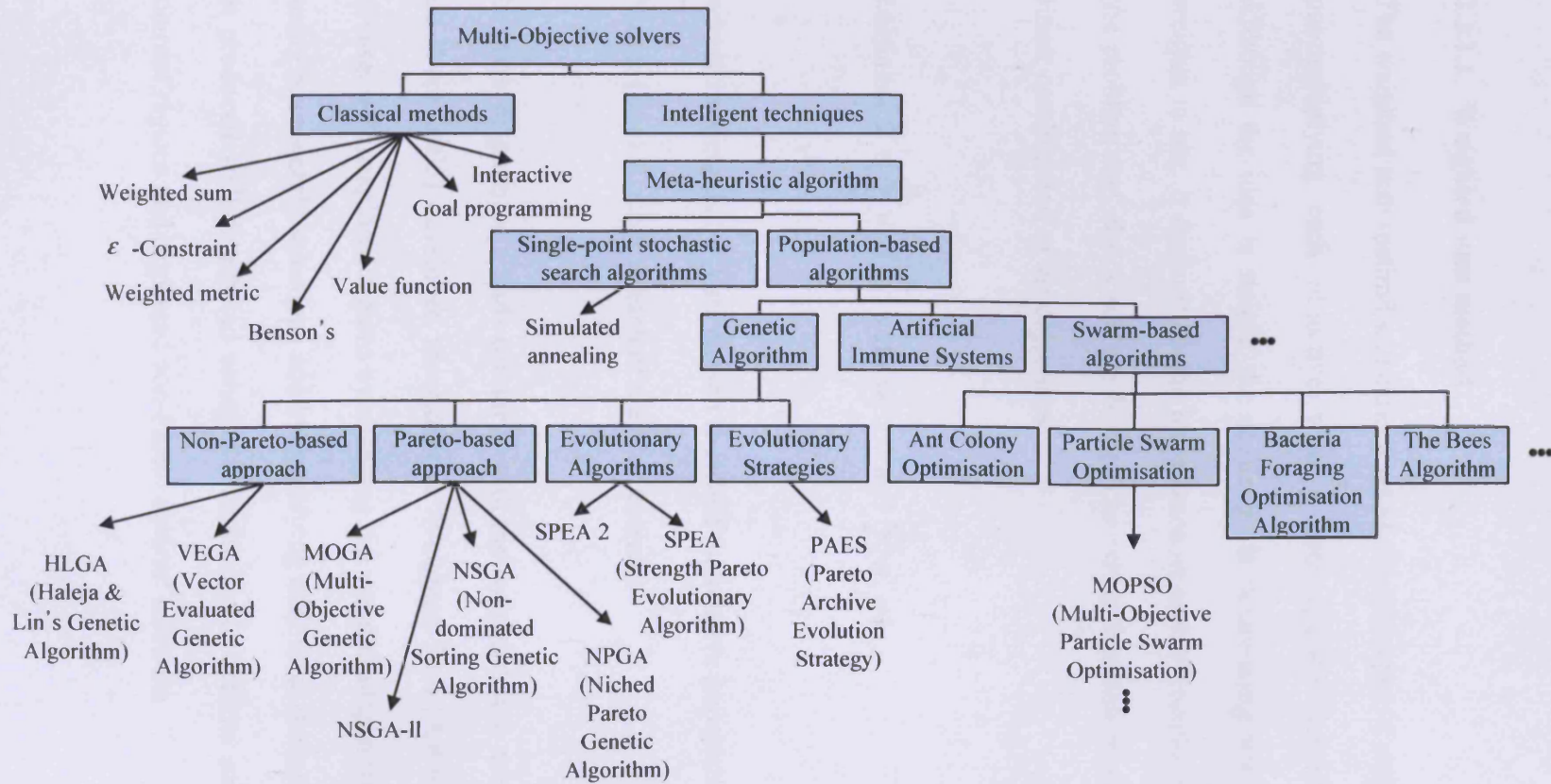


Figure 2.2 Taxonomy of Multi-Objective solvers

2.2.1.1 Weighted sum method

The weighted sum method scalarizes a set of objectives into a single objective by pre-multiplying each objective with a user-supplied weight (Deb 2001). Although the idea is simple, the challenge is determining what values of the weights to use. It depends on the importance of each objective in the context of the problem and also a scaling factor. The utility function of this method is a linear combination of the objectives:

$$\text{Minimise } Z = \sum_{j=1}^N w_j f_j(\vec{x}) \text{ with } w_j \geq 0 \text{ and } \sum_{j=1}^N w_j = 1, \quad \text{Eq. 2.4}$$

where the weights (w_j) are chosen to reflect the relative importance the Decision Maker (DM) attaches to each of the N objectives.

In order to generate a trade-off curve with the weighted sum method, (Kim and de Weck 2005) developed the Adaptive Weighted-Sum (AWS) method. This focuses on unexplored regions by changing the weights adaptively rather than by using prior weight selections and by specifying additional inequality constraints. It produced well-distributed solutions, found Pareto optimal solutions in non-convex regions and neglected non-Pareto optimal solutions.

2.2.2 Intelligent methods

Due to the complexity of real-world optimisation problems such as ill-defined functions and non-differentiability, it is virtually impossible to find exact algorithms. For these types of problems, meta-heuristics are a practical way to generate acceptable solutions, even though they cannot guarantee optimality. Several variants of meta-heuristic algorithms exist including single-point stochastic search algorithms (e.g., simulated annealing (Kirkpatrick et al. 1983; Pham and Karaboga 1999)) and population-based algorithms. However, only population-based algorithms are reviewed here.

2.2.2.1 Population-based Multi-Objective solvers

This section presents some popular population-based MO algorithms. Recent studies on Evolutionary Algorithms have shown that population-based algorithms are potential candidates to solve MOOPs. They can be efficiently used to eliminate most of the difficulties with classical single objective methods such as sensitivity to the shape of the Pareto front and the necessity of multiple runs to find multiple Pareto solutions. Although the majority are based on Evolutionary Algorithms (EAs), the most recent are inspired from swarm intelligence such as Ant Colony Optimisation (ACO) (Dorigo et al. 1996), Particle Swarm Optimisation (PSO) (Eberhart and Kennedy 1995; Kennedy and Eberhart 1995), Artificial Immune Systems (AIS) (Forrest et al. 1994), Bacteria

Foraging Optimisation Algorithm (BFOA) (Passino 2002) and the Bees Algorithm (Pham et al. 2005).

The general structure of population-based MO solvers is similar to those used for SO solvers. However, three different steps are adopted to solve MOOPs.

- 1) *fitness assignment* controls convergence, of which there are three methods: aggregation-based, criterion-based and Pareto-based. Of these the Pareto-based fitness assignment is the most popular and efficient technique. Figure 2.3 shows how to generate a Pareto front using population-based techniques.
- 2) *diversity mechanisms* such as niching are included to determine an individual's fitness in order to prevent premature convergence to a region of the front.
- 3) a form of *elitism* is applied to prevent the problem deteriorating to the point where dominant solutions may disappear from one generation to the next.

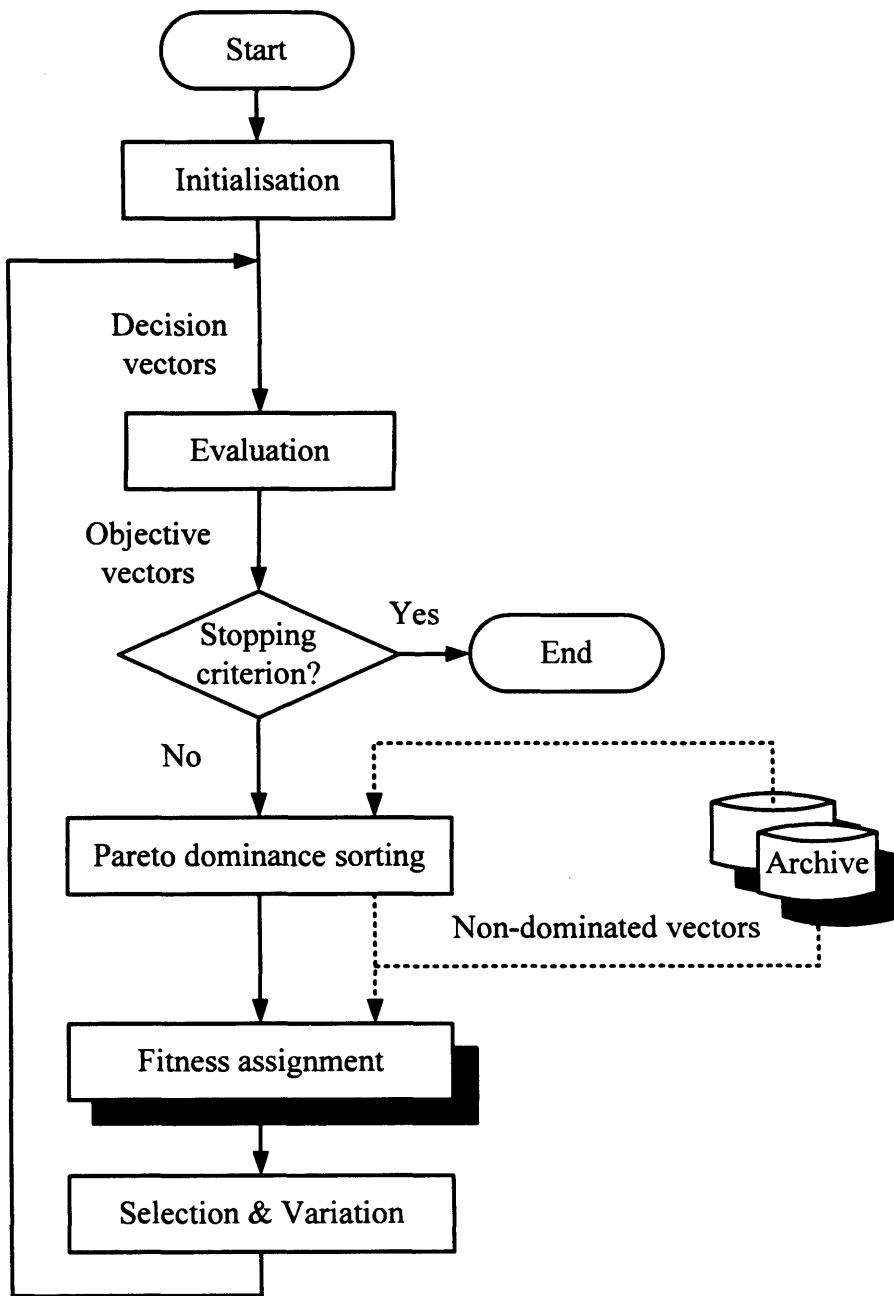


Figure 2.3 Pareto front generation using population-based techniques

2.2.2.1.1 Genetic Algorithms (GAs)

The concept of GA's developed by (Goldberg 1989; Holland 1992) in the 1970s was inspired by the Darwin's evolutionist theory explaining the origin of species. In nature, while weak and unhealthy species are faced with extinction by natural selection, the strong or healthier have a greater opportunity to pass their genes on to future generations via reproduction. In the long run, species carrying the correct combination in their genes become dominant in their population. Sometimes, during the slow evolutionary process, random changes may occur in genes. If these changes provide additional advantages in the challenge for survival, new species evolve from the old ones, but unsuccessful changes are eliminated by natural selection. In GA terminology, a solution vector is called a *chromosome* which consists of discrete units called *genes* and a collection of chromosomes which is called a *population*. In the original implementation of GA by Holland, genes are assumed to be binary digits. However, in later implementations, more varied gene types have been introduced. Normally, a chromosome corresponds to a unique solution in the solution space and it requires a mapping mechanism called an encoding mechanism between the solution space and the chromosomes. Figure 2.4 illustrates a simplified flowchart of GA in which the population is normally randomly initialised. As the search evolves, the population eventually converges, which means that it is dominated by a single solution. GA uses two operators to generate new solutions from existing ones:

- 1) **Crossover** is the most important operator in GA. Generally, two chromosomes, called *parents*, are combined together to form a new chromosome called *offspring*. The parents are selected from existing chromosomes in the population with a preference towards fitness so that their offspring are expected to inherit good genes. By iteratively applying the crossover operator, genes of good chromosomes are expected to appear more frequently in the population, eventually leading to convergence to an overall good solution.

- 2) **Mutation** introduces random changes into the characteristics of chromosomes, which are generally applied at the gene level. In typical GA implementations, the mutation rate (the probability of changing the properties of a gene) is very small, and depends on the length of the chromosome. Therefore the new chromosome produced by mutation will not be very different from the original one. However, it plays a critical role in GA because it reintroduces genetic diversity back into the population and assists the search to escape from local optima.

Reproduction involves the selection of chromosome for the next generation. There are different selection procedures in GA depending on how the fitness values are used. Proportional, ranking and tournament selection are the three most popular selection procedures.

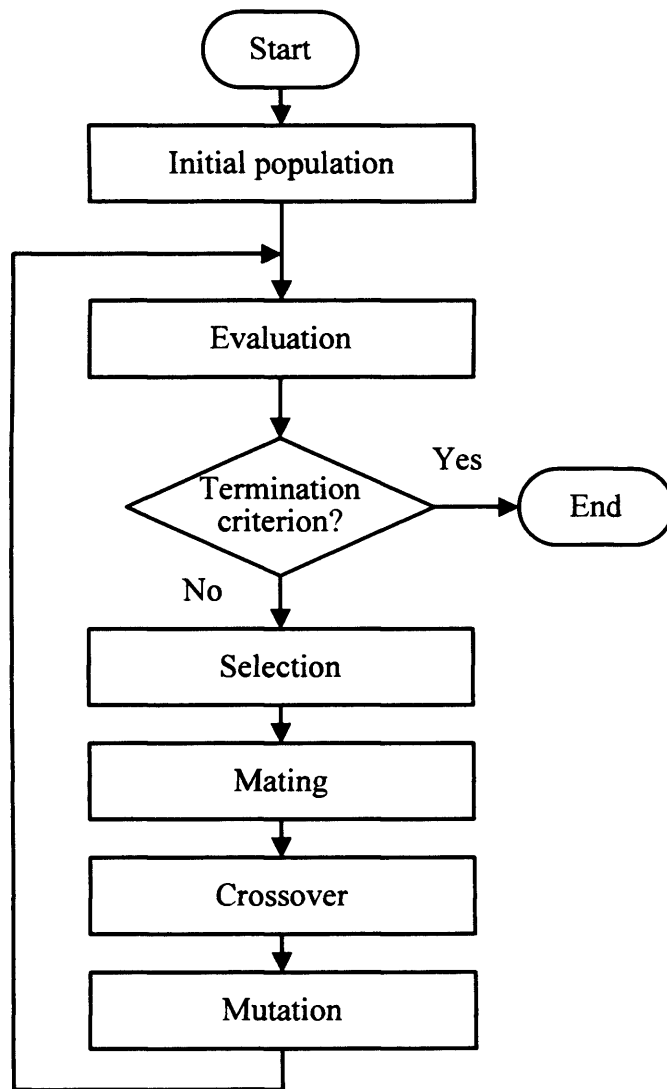


Figure 2.4 Simplified flowchart of a Genetic Algorithm

GA for Multi-Objective Optimisation Problems

GA is well suited to solve MOOPs as a population-based approach. The crossover operator in GA may exploit the structure of good solutions with respect to different objectives to create new non-dominated solutions in unexpected parts of the Pareto front. A GA does not require the user to prioritise, scale or weigh objectives, therefore it has become the most popular heuristic approach to Multi-Objective design and optimisation problems. (Jones et al. 2002) reported that 90% of the approaches to Multi-Objective optimisation aimed to approximate the true Pareto front for the underlying problem. A majority of these used a meta-heuristic technique and 70% of all meta-heuristics approaches were based on evolutionary approaches. (Deb 2001) and (Coello et al. 2007) are well documented regarding various versions of Multi-Objective GAs and their applications. In addition, (Guliashki et al. 2009; Konak et al. 2006; Lee and El-Sharkawi 2008) aptly summarised them. Figure 2.5 shows a GA for MOOPs according to their work.

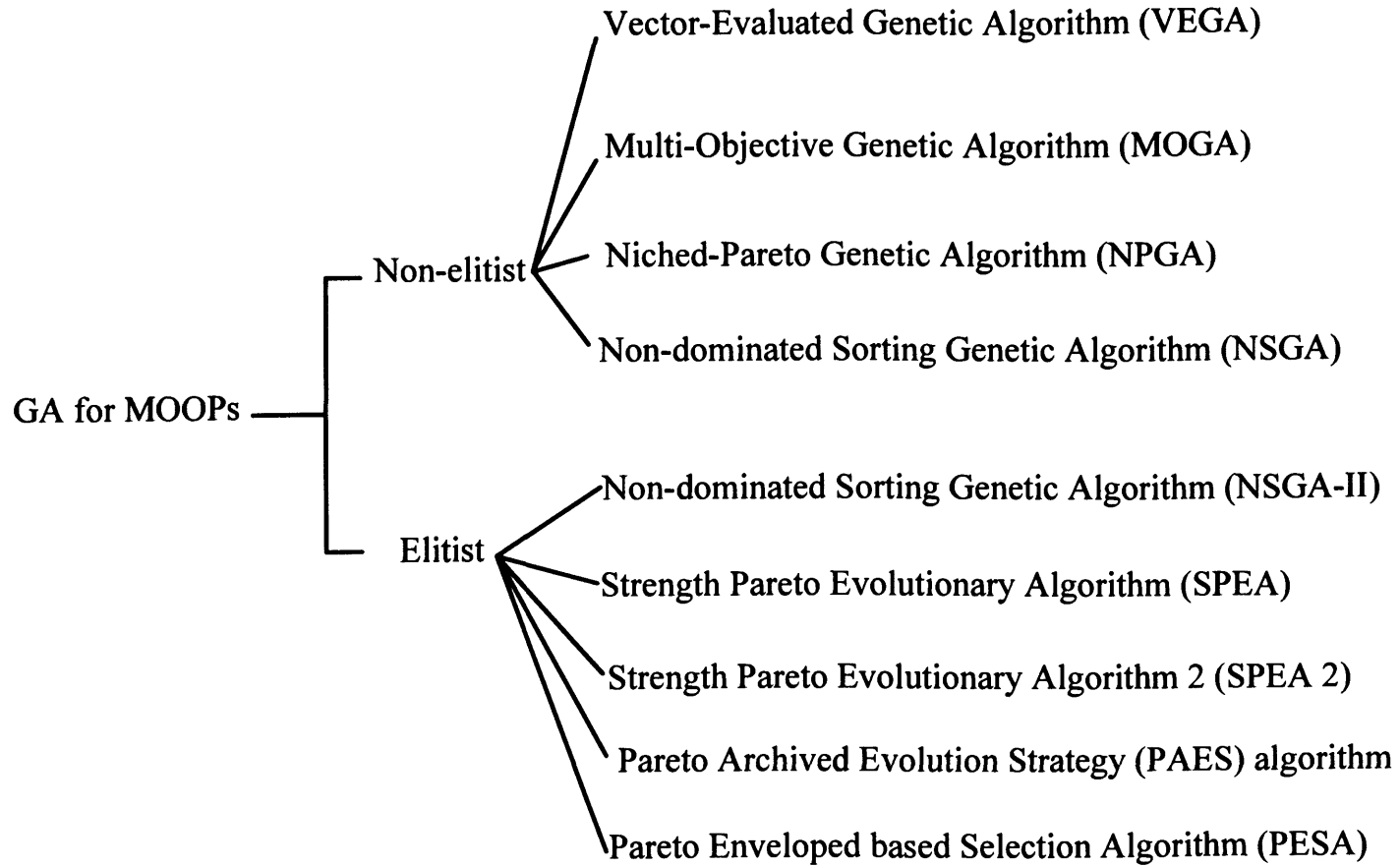


Figure 2.5 Genetic Algorithm for Multi-Objective Optimisation Problems

- Non-elitist Multi-Objective Evolutionary Algorithms

- Vector-Evaluated Genetic Algorithm (VEGA): (Schaffer 1984) suggested VEGA as the first implementation of a real Multi-Objective Evolutionary Algorithm in 1984. He separated the selection method for each individual objective to fill up a portion of the mating pool, which meant that each subpopulation is evaluated with respect to a different objective. Then, the entire population is thoroughly shuffled to apply crossover and mutation operators.
- Multi-Objective Genetic Algorithm (MOGA): (Fonseca and Fleming 1993) proposed MOGA in 1993, which is the simple extension of a single objective GA. The main concept is that all non-dominated individuals are assigned the same highest possible fitness value, while dominated ones are penalised according to the population density of the corresponding region.
- Niche-Pareto Genetic Algorithm (NPGA): (Horn et al. 1994) propounded NPGA which uses a tournament selection scheme based on Pareto dominance. The basic idea of NPGA is: two individuals are randomly chosen and compared against a comparison subset (typically around 10% of the population) from the entire population. When one candidate is dominated by the set and the other is not, the latter is selected. If neither or both candidates are dominated, fitness sharing is used to decide selection.

- Non-dominated Sorting Genetic Algorithm (NSGA): (Srinivas and Deb 1994) postulated NSGA based on several layers of classifications of the individuals. Before selection is performed, the population is ranked on the basis of non-domination. However, it is not very efficient because Pareto ranking has to be repeated over and over again.
- Elitist Multi-Objective Evolutionary Algorithms: The elitist Evolutionary Multi-Objective Optimisation (EMOO) methodologies include an elite-preservation mechanism in their procedures. The non-elitist EMOO algorithms do not use such a mechanism and usually perform worse than the elitist algorithms. In the context of Multi-Objective optimisation, elitism usually, (although not necessarily), refers to the use of an external population (also called secondary population) to retain the non-dominated individuals found along the evolutionary process. Following the theory offered by (Zitzler and Thiele 1999), most researchers began to incorporate external populations in their EMOO algorithms and the use of this mechanism became common practice.
 - Non-dominated Sorting Genetic Algorithm-II (NSGA-II): (Deb et al. 2002) introduced NSGA-II as an improved version of NSGA. It adopts a Crowding Distance mechanism for diversity. Its elitist mechanism combines the best parents with the best offspring obtained, instead of using an external memory.

- Strength Pareto Evolutionary Algorithm (SPEA): (Zitzler and Thiele 1999) introduced SPEA which uses an archive containing non-dominated solutions previously found (the so-called external non-dominated set). At each generation, non-dominated individuals are copied to the external non-dominated set. For each individual in this external set, a strength value is computed. The fitness of each member of the current population is computed according to the strengths of all external non-dominated solutions that dominate it. Although this approach does not require a niche radius, its effectiveness relies on the size of the external non-dominated set. If its size grows too large, it slows down the search procedure.
- Strength Pareto Evolutionary Algorithm 2 (SPEA 2): This was also developed by (Zitzler and Thiele 1999) to address some of the major drawbacks of SPEA. It has three main differences with SPEA:
 - 1) It incorporates a fine-grained fitness assignment strategy which takes into account for each individual the number of individuals that dominate it and the number of individuals by which it is dominated.
 - 2) It uses a nearest neighbour density estimation technique which guides the search more efficiently.
 - 3) It has an enhanced archive truncation method that guarantees the preservation of boundary solutions.

- Pareto Archived Evolution Strategy (PAES) algorithm: (Knowles and Corne 2000) introduced PAES consists of a 1+1 evolution strategy (i.e., a single parent that generates a single offspring) in combination with a historical archive that records the non-dominated solutions previously found. The role of the external archive is limited to storing the non-dominated solutions and provides a source of comparison for ranking candidate solutions. However, archive members are not involved in the mutation and crossover procedures. The interesting feature of this algorithm is in the adaptive grid at the heart of its diversity and niching mechanisms. The objective space is divided in a recursive manner, thus creating a multi-dimensional co-ordinate system, or grid, over the objective space. A crowding-based fitness sharing mechanism is applied by determining the location of solutions within this grid and estimating the density of solutions per cell. Individuals corresponding to solutions in the less crowded cells have higher fitness.
- Pareto Envelope based Selection Algorithm (PESA): (Corne et al. 2000) suggested PESA which combines the good aspects of SPEA and PAES. Like SPEA, PESA carries two populations (a smaller EA population and a larger archive population). Non-domination and the PAES crowding concept are used to update the archive with the newly created child solutions.

2.2.2.1.2 Ant algorithm

The Ant System (AS) developed by (Dorigo 1992) was the original ant algorithm. Since then several improvements to the Ant System have been devised. It was inspired by colonies of real ants that deposit a chemical substance on the ground called *pheromone*. This substance influences the behaviour of the ants as they tend to take those paths where there is a larger amount of *pheromone*. *Pheromone* trails can thus be seen as an indirect communication mechanism among ants. Three main ideas from actual ant colonies that have been adopted are:

- Indirect communication through *pheromone* trails
- Shortest paths tend to have a greater *pheromone* growth rate
- Ants have a higher preference (with a certain probability) for paths that have a larger quantity of *pheromone*

Ant Colony Optimisation (ACO) was introduced by (Dorigo et al. 1996) as a novel nature-inspired metaheuristic for solving combinatorial optimisation problems such as the Travelling Salesman Problem (TSP) which entails the cost function to be minimised. The simple ACO can be stated as given by (Bonabeau et al. 1999; Camazine et al. 2003; Dorigo and Stutzle 2004; Engelbrecht 2006): A combinatorial optimisation problem entails the cost function to be minimised. A candidate solution is defined as a sequence of parameters visualised as a path through several nodes. Each node corresponds to one of the solution's

parameters. Moving from one node (i) to another (j) is carried out by the probability (Eq. 2.5).

$$p_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha(t)\eta_{ij}^\beta(t)}{\sum_{u \in N_i^k(t)} \tau_{iu}^\alpha(t)\eta_{iu}^\beta(t)} & \text{if } j \in N_i^k(t) \\ 0 & \text{if } j \notin N_i^k(t) \end{cases} \quad \text{Eq. 2.5}$$

where

$p_{ij}^k(t)$: transition probability

τ_{ij} : posterior effectiveness of the move from node i to node j , as expressed in the *pheromone* intensity of the corresponding link (i, j)

η_{ij} : prior effectiveness of the move from i to j (i.e., the attractiveness or desirability of the move) and it is computed using some heuristic

α : parameter to control the influence of τ_{ij}

β : parameter to control the influence of η_{ij}

N_i^k : set of feasible nodes for ant k when located on node i

The *pheromone* concentrations, τ_{ij} , indicate how profitable it has been in the past to make a move from i to j , serving as a memory of previous best moves. *Pheromone* evaporation is implemented as given in Eq. 2.6. After completion of a path by each ant, the *pheromone* on each link is updated according to Eq. 2.7.

$$\tau_{ij}(t) \leftarrow (1 - \rho)\tau_{ij}(t) \quad \text{with } \rho \in [0,1] \quad \text{Eq. 2.6}$$

The constant, ρ , specifies the rate at which *pheromone* evaporates, causing ants to ‘forget’ previous decisions controlling the influence of search history. For example, if ρ is a large value, *pheromone* will evaporate rapidly and vice versa. The greater the evaporation, the more random the search becomes which means that it is facilitating a better exploration. When $\rho = 1$, the search is completely random.

$$\tau_{ij}(t+1) = \tau_{ij}(t) + \Delta\tau_{ij}(t) \quad \text{Eq. 2.7}$$

$$\Delta\tau_{ij}(t) = \sum_{k=1}^{n_k} \Delta\tau_{ij}^k(t) \quad \text{Eq. 2.8}$$

where

n_k : number of ants

$\Delta\tau_{ij}^k(t)$: amount of *pheromone* deposited by ant k on link (i, j) at time step t .

Figure 2.6 illustrates the flowchart of ACO for solving combinatorial problems.

ACO for Multi-Objective Optimisation Problems

The first implementations of ACO algorithms utilised only one colony of ants to construct solutions. However, multiple colony algorithms have been adapted to use multiple colonies, with the main objectives of mitigating stagnation and

minimising the chances of being trapped in local minima. Multiple colonies have also been used to solve MOOPs and it was one of the first applications of multiple colony ACO. MOOPs are solved by assigning to each colony the responsibility of optimising one of the objectives. If nc objectives need to be optimised, a total of nc colonies are used. Colonies cooperate to find a solution that optimises all objectives by sharing information about the solutions found by each colony. (Tippachon and Rerkpreedapong 2009) adopted this Multiple Ant Colony System (MACS) for the optimal placement of switches and protective devices in electrical power distribution systems. (Colson et al. 2009) also employed this algorithm for solving a Micro-Grid (MG) Multi-Objective power management problem.

Another version of ACO for solving MOOPs is Population-based ACO (PACO) which was introduced by (Guntsch and Middendorf 2002). (Angus 2007) applied this algorithm to Multi-Objective function optimisation. The defining difference between PACO and canonical ACO algorithms is the method used for storing solutions. Most traditional ACO algorithms store solution information from an (artificial) ant in a *pheromone* matrix only, PACO stores solutions in a population and then uses this population to make adjustments to the *pheromone* matrix. However, PACO still uses the core principles of ACO which include stepwise construction (solutions are constructed one piece at a time) and the use of global information in constructing solutions.

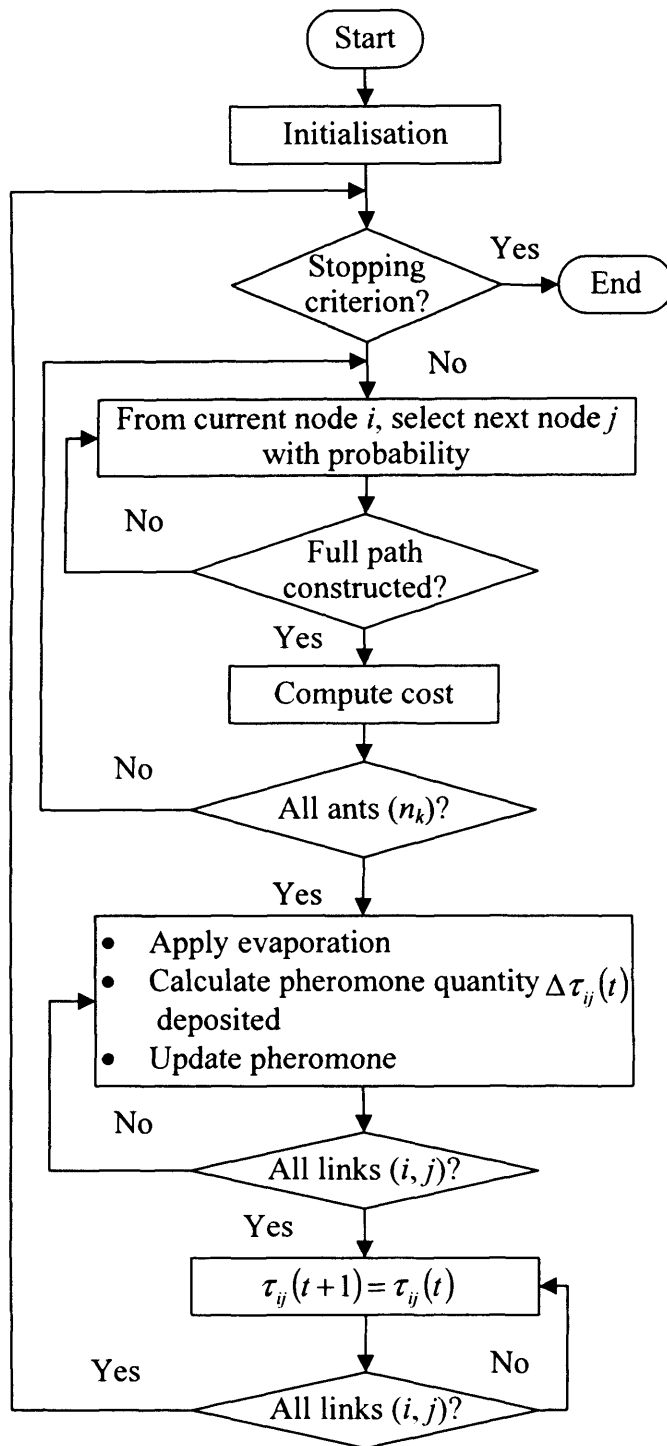


Figure 2.6 Flowchart of Ant Colony Optimisation algorithm

2.2.2.1.3 Particle Swarm Optimisation (PSO)

Kennedy and Eberhart proposed a population-based stochastic optimisation called ‘Particle Swarm Optimisation (PSO)’ in 1995 (Eberhart and Kennedy 1995; Kennedy and Eberhart 1995) inspired by the collective behaviour of social animals such as bird flocking or fish schooling (Engelbrecht 2006; Kennedy et al. 2001; Van den Bergh 2006). A PSO algorithm maintains a swarm of particles and an individual particle, which represents a potential solution, moving through a multi-dimensional search space to approach the optima. Figure 2.7 illustrates the flowchart for PSO. Initially, the number of particles are randomly created and set into motion through the search space (‘Initial population’ in Figure 2.7). A particle has its own position and flight velocity, which is constantly adjusted during the optimisation process. For a better position (fitness), each particle adjusts its position based on its own experiences (personal best: ‘pbest’) as well as that of the entire swarm (global best: ‘gbest’) at each generation. Eq. 2.9 represents the formula to change the position of the particle by adding a velocity, $V_i(t+1)$, to the current position ($X_i(t)$). The velocity formula is shown in Eq. 2.10.

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad \text{Eq. 2.9}$$

where

i ($i = 1, 2, \dots, n$): index of individual particles in the swarm

n : total number of particles in the swarm

t : discrete time steps

$X_i(t+1)$: updated position of particle i at next step $(t+1)$ which is calculated using the current position with updated velocity

$X_i(t)$: current position of particle i in the search space at step t

$V_i(t+1)$: updated velocity of particle i at next step $(t+1)$

$$v_{ij}(t+1) = wv_{ij}(t) + c_1r_{1j}(t)[pbest_j(t) - x_{ij}(t)] + c_2r_{2j}(t)[gbest_j(t) - x_{ij}(t)] \quad \text{Eq. 2.10}$$

where

j ($j = 1, 2, \dots, m$): index of dimensional search space

m : number of dimensional search space

$v_{ij}(t+1)$: velocity of particle i in dimension j at step $(t+1)$ calculated by using the current velocity and the distance from $pbest_i$ to $gbest_i$

w : inertia weight that controls the impact of previous velocity

$v_{ij}(t)$: velocity of particle i in dimension j at step t which is confined to the range $[v_{\min}, v_{\max}]$ to control excessive roaming of particles outside the search space

$x_{ij}(t)$: position of particle i in dimension j at step t

c_1 and c_2 : positive acceleration constants used to scale the contribution of the cognitive and social components respectively

$r_{1j}(t)$ and $r_{2j}(t)$: random values in the range $[0,1]$ sampled from a uniform distribution

$pbest_{ij}(t)$: personal best position of particle i at step t since the first step

$gbest_j(t)$: global best position at step t , which is the best position discovered by any of the particles to date

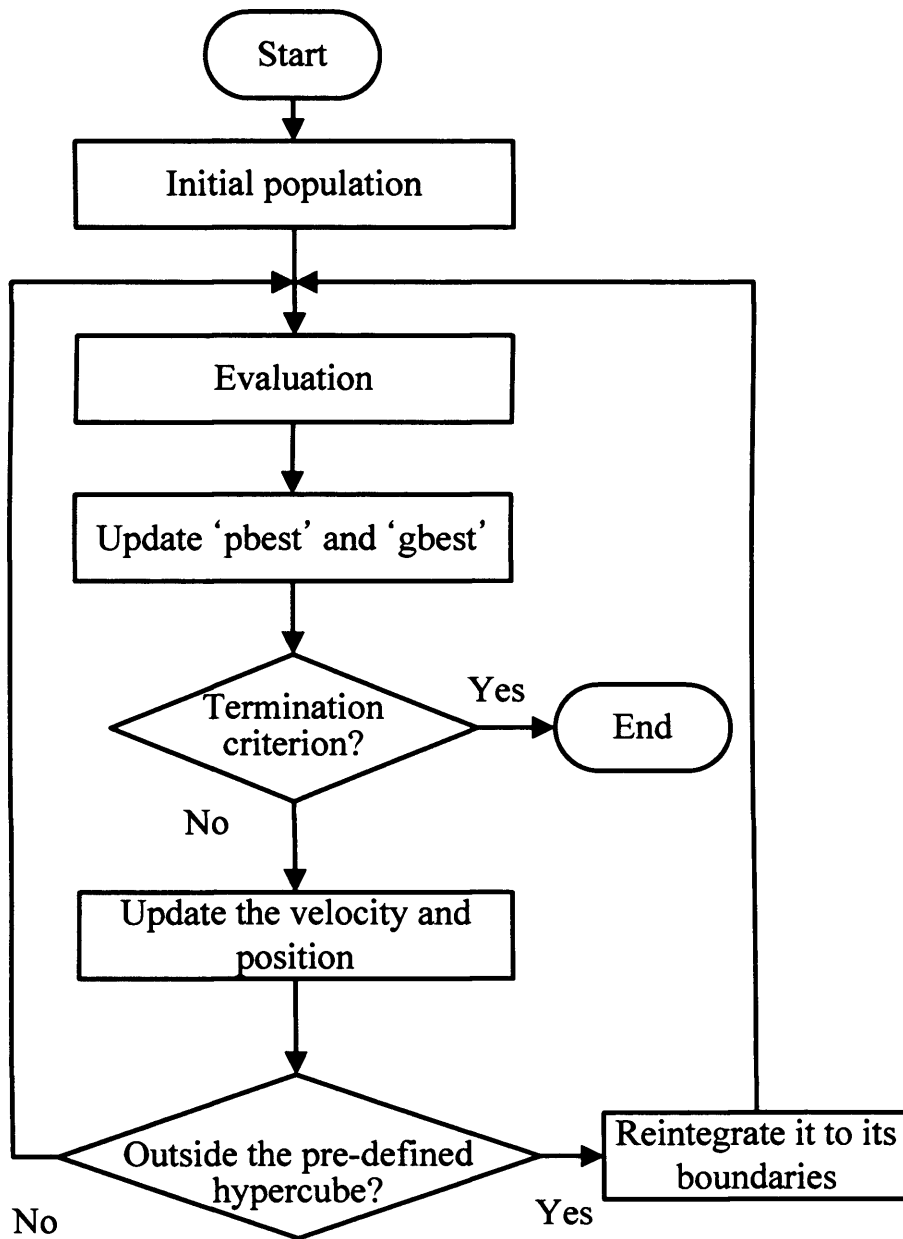


Figure 2.7 Simplified flowchart of Particle Swarm Optimisation algorithm

Under the guidance of these two updating rules (Eq. 2.9 and 2.10), the particles will proceed toward the best position found so far. This driving force makes the PSO algorithm search to find the optimal solutions.

PSO for Multi-Objective Optimisation Problems

The basic PSO cannot be applied directly to solve MOOPs (Engelbrecht 2006). This is due to the velocity update equation where the social component causes all particles to converge on one point. Changing conventional single objective PSO to a Multi-Objective PSO (MOPSO) requires redefining the global best because there is no absolute global best in the Multi-Objective PSO procedure, but rather a set of non-dominated solutions. Therefore choosing the global best to guide the swarm becomes a nontrivial task in the Multi-Objective domain (Abido 2009). Many researchers have proposed several PSO-based MO solvers which are generally categorised by three methods:

- **Aggregation-based methods:** One of the simplest approaches to deal with MOOPs, is to define an aggregate objective function as a weighted sum of the objectives, because uni-objective optimisation algorithms can be applied without any changes to the algorithm to find optimum solutions. (Parsopoulos and Vrahatis 2002a, b) applied this aggregation method to a number of standard benchmarking functions, but there are numerous problems. For example, the

algorithm has to be applied repeatedly to find different solutions. Even for repeated applications, there is no guarantee that different solutions will be found. It is also difficult to determine the best weight values because these depend totally on the particular problem (Engelbrecht 2006).

- **Criterion-based methods:** (Parsopoulos and Vrahatis 2002a, b) also developed a criterion-based MOPSO for MOOPs with two objectives. With criterion-based methods, different stages of the optimisation process use different objectives separately instead of considering all objectives simultaneously.
- **Dominance-based methods:** These make use of Pareto preference to find a set of non-dominated solutions. Apart from finding solutions with the best trade-offs, MOO algorithms should also maintain already found solutions (to ensure that they are not lost over time) whilst maintaining diversity of the estimated Pareto front. To facilitate diversity and memorisation of located non-dominated solutions, dominance-based methods usually use an archive (or repository) to store all located non-dominated solutions. These methods are popular among researchers therefore there are many versions of MOPSO with various applications. For example, some researchers divide the process into multi-phases to generate the Pareto front. (Mostaghim and Teich 2004) divided the process into two phases and they applied it to test functions and antenna design problems. In phase 1, the goal is to obtain a good approximation of the Pareto

front, and in the second, sub-swarms are generated to discover the Pareto front. (Kitamura et al. 2005) also sub-divided the Multi-Objective problem into partial problems and each partial problem is solved by using MOPSO for Multi-Objective energy management systems. Recently MOPSO has been applied to solve EEDPs. For example, (Wang and Singh 2006, 2007) developed an enhanced PSO which chooses the global best from the fuzzified region using binary tournament selection. (Abido 2007, 2009) also proposed MOPSO which redefined the global and local best individuals and developed mechanisms for the selection of these individuals for the EEDP. In addition, (Cai et al. 2009) developed a chaotic MOPSO using 1) an Adaptive Inertia Weight Factor (AIWF) to perform global exploration and 2) a Chaotic Local Search (CLS) to perform locally oriented search (exploitation).

The greatest advantages of a PSO are its simplicity (both conceptually and at the implementation level), its ease of use and its high convergence rate. The disadvantages are mainly related to the apparent difficulties to control diversity (Coello et al. 2007).

2.2.2.1.4 Artificial Immune System (AIS)

The Artificial Immune System (AIS) is a computational system inspired by the principles and processes of the vertebrate immune system. Our immune system protects organisms from bacteria, viruses and other foreign pathogens. Its main task is to recognise all cells within the body and characterise them into self and

foreign cells (or antigens). The immune system further characterises foreign cells and develops defensive mechanisms against them (i. e., antibodies). If a foreign pathogen (i. e., an antigen) enters the body, then the immune system can launch a specific response against it. Specialised B cells must interact with Helper T cells (specialised white blood cells) to initiate antibody production. Antibodies are specific to only one type of antigen, which they immobilise, thus preventing infections.

Computationally speaking, the immune system is a parallel intelligent system that is able to learn and retrieve previous knowledge (i.e., it has “memory”) in order to solve recognition and classification tasks. Due to these interesting features, several researchers have developed computational models of the immune system to solve various computational problems for mathematics, engineering and information technology. AIS began in the mid 1980s with (Farmer et al. 1986) and (Bersini and Varela 1991) on immune networks. However, it was only in the mid 1990s that AIS become a subject area in its own right. (Forrest et al. 1994) and (Kephart 1994) published their first papers on AIS in 1994 and (DasGupta 1998) conducted extensive studies on Negative Selection Algorithms. Figure 2.8 shows the pseudo code of AIS. The main idea of this approach is to construct a population of antigens and a population of antibodies. Antibodies are then matched with antigens and a fitness value assigned to each antibody (i.e., maximise matching between antigens and antibodies). Finally, a

conventional Genetic Algorithm is used to replicate the antibodies that best match the antigens present.

AIS for Multi-Objective Optimisation Problems

The first direct use of the immune system to solve MOOPs reported in the literature is the work of (Yoo and Hajela 1999). This approach uses a linear aggregating function to combine objective function and constraint information into a scalar value that is used as the fitness function of a GA. The best designs according to this value are defined as antigens and the rest of the population as a pool of antibodies. (Freschi et al. 2009) gave good reviews regarding Multi-Objective Optimisation with Artificial Immune Systems (MOAIS). (Coello et al. 2007) are also well documented regarding MOAIS. More recently, (Tan et al. 2008) developed an Evolutionary Multi-Objective Immune Algorithm (EMOIA) to solve several benchmark problems. In order to design an algorithm that is capable of exploiting the complementary features of EA and AIS, a few features such as archive, an Entropy-based Density Assessment Scheme (EDAS), clonal selection and genetic operators are incorporated into the EMOIA.

(Park et al. 2009) also developed a GA-based Immune System to apply to several benchmark problems. The main objective of their work is to locate the non-dominated Pareto solutions utilising both GA and AIS in the context of a Multi-Objective GA. In their design of population, dominated solutions are considered

Repeat

- 1) Select an antigen A from PA (PA : population of antigens)
- 2) Take (randomly) R antibodies from PS (PS : population of antibodies)
- 3) For each antibody $r \in R$, match it against the selected antigen A

 Compute its match score

- 4) Find the antibody with the highest match score

 Break ties at random

- 5) Add match score of winning antibody to its fitness

Until maximum number of cycles is reached

Figure 2.8 Pseudo code of Immune System model (fitness scoring)

as antigens, whereas non-dominated solutions are assigned to antibodies. Affinity measures are used to evaluate how many antigens and antibodies are similarly matched. A GA-based immune simulation corresponding with a selection or reproduction operation used in a conventional GA is also conducted to discover new antibodies with a higher degree of non-domination. This results in a positive contribution to the identification of Pareto solutions.

(Gao and Wang 2010) developed a Weight-Based Multi-Objective Artificial Immune System (WBMOAIS) based on opt-aiNET to several benchmark problems. The opt-aiNET algorithm is structured into two nested levels. The inner level (exploitation of the fitness landscape) takes into account computing the affinity relationship, and stimulating the most promising cells, while the outer level (exploration of the fitness landscape) manages the network of cells of the system, deleting similar ones and preserving the most promising ones into memory.

AIS are relatively simple algorithms with fine properties that make them naturally suitable for pattern recognition and classification tasks. Researchers have realised that this heuristic can also be successfully adopted for optimisation tasks. However, not many researchers have explored this area, since most of the current work on Artificial Immune Systems is focused on architecture, models and applications not related to optimisation. Although extending an Artificial

Immune System can deal with a MOOP, their main weakness is normally the additional parameters required (some of which may be difficult to fine tune for an arbitrary problem). In fact, some of these parameters may not be obvious at first sight (e.g., the proportion of antigens and antibodies in the population, the number of clones to be produced, etc.), and can become cumbersome when trying to use them (Coello et al. 2007).

2.2.2.1.5 Bacteria Foraging Optimisation Algorithm (BFOA)

BFOA inspired by the social foraging behaviour of bacteria (such as *Escherichia Coli* or *E. Coli*), has been widely accepted as a global optimisation algorithm. Bacteria have the tendency to gather in a nutrient-rich area by an activity called *chemotaxis*. *Chemotaxis* is cell movement in response to gradients of chemical concentrations present in the environment. This movement and the chemical substances involved are used by bacteria as a survival strategy that allows them to search for nutrients and avoid noxious environments. The chemotactical behaviour of bacteria as an optimisation process was modelled for the first time by (Bremermann 1974) in the early 1970s. In 2002, a new optimisation algorithm based on foraging behaviour of bacteria was introduced by (Passino 2002). This novel algorithm which is known as the Bacterial Foraging Optimisation Algorithm (BFOA), considers not only the chemotactical strategy but also other stages of bacterial foraging behaviour such as swarming, reproduction and elimination and dispersal. Bacterial communication acquires great influence on

the entire process, getting closer to the concept that foraging is a phenomenon of a bacterial colony rather than individual behaviour. E. Coli has 8-10 flagella placed randomly on a cell body. When all flagella rotate counter clockwise, they form a compact, propelling the cell along a helical trajectory, which is called run. When the flagella rotate clockwise, they pull on the bacterium in different directions, which cause the bacteria to tumble. The three prime steps in BFOA are:

Chemotaxis: This process simulates the movement of an E. Coli cell through swimming and tumbling via flagella (Liu and Passino 2002). It can alternate between these two modes of operation for its entire lifetime. The movement of the bacterium in computational *chemotaxis* is presented by Eq. 2.11.

$$\theta^i(j+1, k, l) = \theta^i(j, k, l) + C(i) \frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \quad \text{Eq. 2.11}$$

where

Δ : indicates a vector in the random direction whose elements lie in $[-1,1]$

$\theta^i(j, k, l)$: represents i^{th} bacterium at j^{th} chemotactic, k^{th} reproductive step
and l^{th} elimination-dispersal step

$C(i)$: size of the step taken in the random direction specified by the tumble
(run length unit).

If the cost at $\theta^i(j+1, k, l)$ is better than the cost at $\theta^i(j, k, l)$, then the bacterium takes another step of size $C(i)$ in that direction otherwise it is allowed to tumble. This process is continued until the number of steps taken is greater than the number of chemotactic loops.

Reproduction: After all chemotactic steps have been covered, a reproduction step takes place. The fitness values of the bacteria are sorted in ascending order. The lower half of the bacteria with the highest fitness die and the remaining healthier bacteria are allowed to split into two identical ones, which are replaced in the same location. This keeps the swarm size constant.

Elimination and Dispersal: Since bacteria may get stuck around the initial or local optima positions, it is necessary to diversify the bacteria either gradually or suddenly so that the possibility of being trapped in a local minimum is eliminated. Dispersion takes place after a certain number of reproduction processes. A bacterium is chosen according to a probability to be dispersed and moved to another position in the search space. These events may effectively prevent a local minima trap forming, but can have the undesirable consequence of disrupting the optimisation process. Figure 2.9 illustrates the simplified flowchart of BFOA.

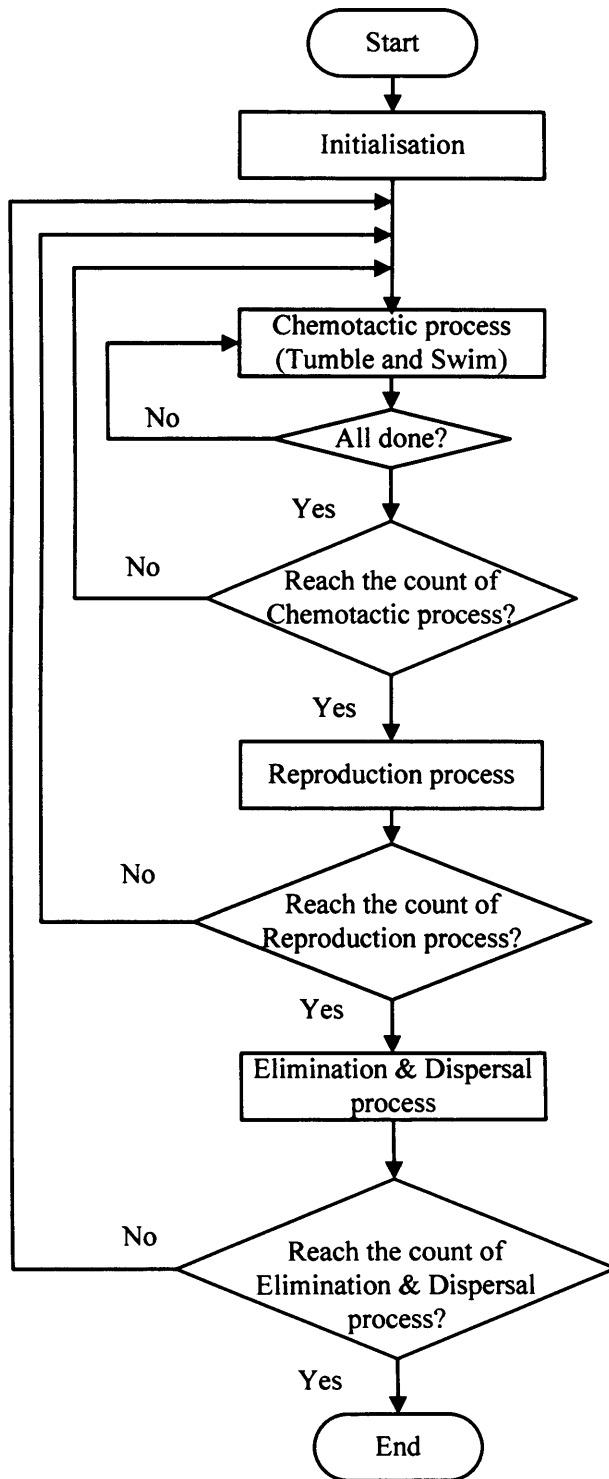


Figure 2.9 Simplified flowchart of Bacterial Foraging Optimisation

Algorithm

BFOA for Multi-Objective Optimisation Problems

(Guzman et al. 2010) recently developed a novel algorithm based on the *chemotaxis* process of *Echerichia Coli* to solve several benchmark problems. The algorithm used a fast non-dominated sorting procedure, via communication between the colony members and a simple chemotactical strategy to change the bacterial positions in order to explore the search space to find several optimal solutions.

2.2.2.1.6 Bee-inspired Algorithms

During the last decade, the intelligent behaviour of bee swarms has inspired researchers to develop new algorithms. Some of these have been utilised as algorithms and are illustrated in Figure 2.10. For example, the Marriage Process in Honey Bees (MBO) algorithm (Abbass 2001) and the Honey-Bees Mating Optimisation (HBMO) algorithm (Haddad et al. 2006) were inspired by the mating behaviour of bees. The BeeHive algorithm (Wedde et al. 2004; Xu et al. 2007) evolved from the communicative and evaluative methods and procedures of honey bees. Passino (Passino et al. 2008) was influenced by nest-site selection behaviour of honey bees and found numerous similarities in the functional organisation of vertebrate brains and honey bee swarms with decision making. However, foraging behaviour is the most interesting and numerous algorithms have developed from this behaviour such as the Bees System (BS) (Lucic and Teodorovic 2001; Sato and Hagiwara 1997), the Bee Colony Optimization

(BCO) algorithm (Teodorovic and Dell'Orco 2005), the Honey Bee Algorithm (Nakrani and Tovey 2004; Tovey 2004), BeeAdhoc (Wedde et al. 2005), Bee Swarm Optimisation (BSO) (Drias et al. 2005), the Artificial Bee Colony (ABC) algorithm (Karaboga 2005; Karaboga and Basturk 2008), the Virtual Bee Algorithm (VBA) (Yang 2005), the Honey bee social foraging algorithms (Quijano and Passino 2010), the Honey Bee Foraging (HBF) algorithm (Baig and Rashid 2007), the Honeybee search algorithm (Olague and Puente 2006) and the Bees Algorithm (Pham et al. 2005). More detailed reviews regarding bee-inspired algorithms can be found in (Karaboga and Akay 2009) and (Bitam et al. 2010).

In this section, only the Bees Algorithm and its applications are considered. Table 2.1 shows the list of its applications.

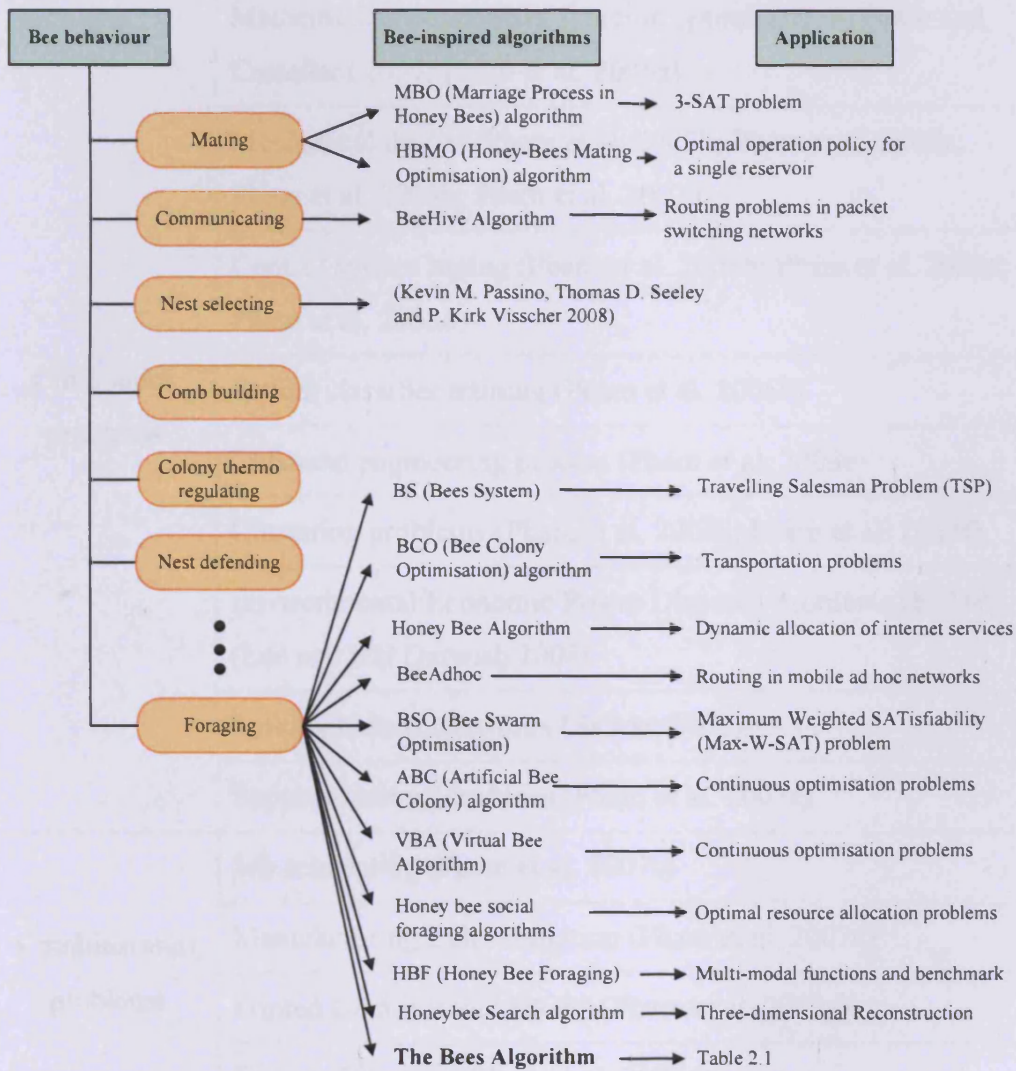


Figure 2.10 Bee behaviour and algorithms

	Applications
Continuous problems	Mathematical benchmark function optimisations (Pham and Castellani 2009; Pham et al. 2006a)
	Mechanical design (Pham et al. 2007b; Pham et al. 2008c; Pham et al. 2009b; Pham et al. 2007h)
	Control system tuning (Pham et al. 2008b; Pham et al. 2009a; Pham et al. 2008d)
	Pattern classifier training (Pham et al. 2006b)
	Chemical engineering process (Pham et al. 2008e)
	Clustering problems (Pham et al. 2008a; Pham et al. 2007f)
	Environmental/Economic Power Dispatch Problems (EEDP) (Lee and Haj Darwish 2008)
	Spiking Neural Networks (Sahran 2007)
	Support Vector Machines (Pham et al. 2007e)
Combinatorial problems	Job scheduling (Pham et al. 2007c)
	Manufacturing Cell Formation (Pham et al. 2007a)
	Printed Circuit Board (PCB) (Pham et al. 2007g)
	Feature Selection (Pham et al. 2007d)

Table 2.1 Applications of the Bees Algorithm

The Bees Algorithm is inspired by the honey bees' natural foraging behaviour (Camazine et al. 2003; Oldroyd and Wongsiri 2006) and it has been developed by (Pham et al. 2005, 2006a). Figure 2.11 illustrates the natural food processing procedures of honey bees. The food processing procedure in the colony consists mainly of two parts; one by middle-aged workers inside the hive (Figure 2.11) and the other by foraging bees outside the hive. In the hive, middle-aged workers receive the nectar from foragers, which is either distributed for immediate consumption or processed into honey and stored in special honey cells. The foraging process is conducted by scout bees which are sent to flower patches to search for a food source. When they return to the hive, they regurgitate the nectar to the middle-aged workers and then assume one of the three following roles: 1) foraging without advertisement of their found patches, 2) foraging with advertisement of their findings for recruitment or 3) abandoning their found nectar source to exploit new sources. To advertise their findings, bees conduct a '*waggle dance*' (Seeley 1996) which contains important information regarding a flower patch they have found and it guides unemployed foragers to go precisely to the found flower patches. Unemployed foragers have a greater attraction to the more lucrative patches thus more bees are sent to the most fruitful sites (Camazine et al. 2003). This enables the colony to gather food much quicker and more efficiently.

The Bees Algorithm only adopts the second and third roles. Role 2 is called a 'Neighbourhood search' (or 'Local search') and role 3 is called a 'Random search' (or Global search). A flowchart is shown in Figure 2.12. The algorithm requires a number of parameters to be set, namely:

- 1) number of scout bees: n ,
- 2) number of sites selected out of n visited sites: m ,
- 3) number of best (or elite) sites out of m selected sites: e ,
- 4) number of bees recruited for the best e sites: ne ,
- 5) number of bees recruited for the other $(m-e)$ selected sites: nm and
- 6) patch size which includes the site and its neighbourhood: ngh .

The algorithm starts with the n scout bees being placed randomly into the search space. The fitness of each site visited by the scout bees is evaluated and bees with the best fit are chosen as 'selected bees' and sites visited by them are picked for a 'Neighbourhood search' (or 'Local search') in Figure 2.12. Then, the algorithm conducts searches in the neighbourhood of the selected sites, assigning the most suitable bees to search in the vicinity of the best sites (e). The bees are selected directly according to the fitness associated with the sites they are visiting. Alternatively, the fitness values are used to determine the probability of the bees being selected. Searches in the neighbourhood of the best sites (e) which represent the more promising solutions are enhanced by recruiting more bees to

follow them than the other selected bees ($m-e$). Together with scouting, this differential recruitment is a key operation of the Bees Algorithm. However, for each patch, only the bee with the highest fitness will be selected to form the next bee population, although there is no such restriction in nature. This restriction is introduced here to reduce the number of points to be explored. The remaining bees in the population are assigned randomly around the search space scouting for new potential solutions, called 'Random search' (or 'Global search') in Figure 2.12. These procedures are repeated until a stopping criterion is met.

Refinements to enhance the performance of the Bees Algorithm include:

- (Ghanbarzadeh 2007) introduced the interpolation and extrapolation of mating of the unselected bees. He also proposed the shrinking method for neighbourhood size by introducing 'abandon sites without new information' when stuck in a local optimum.
- (Pham and Sholedolu 2008) proposed the hybrid PSO-Bees Algorithm to prevent premature convergence in the basic PSO algorithm. He employed an adaptive neighbourhood search (shrinking method) and also added random particles to conduct a global search.

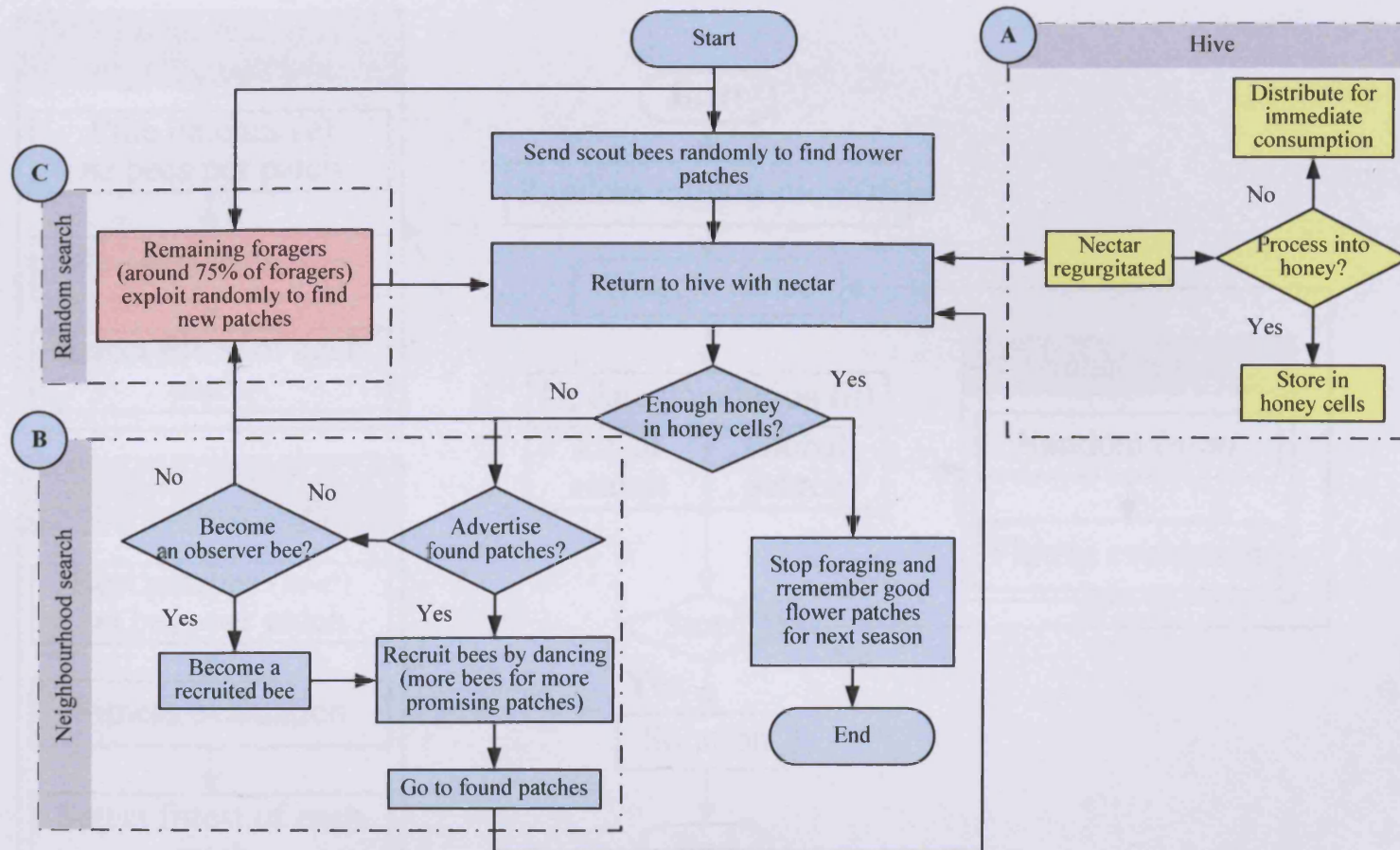


Figure 2.11 Flowchart of food processing procedure of honey bees in nature for one season

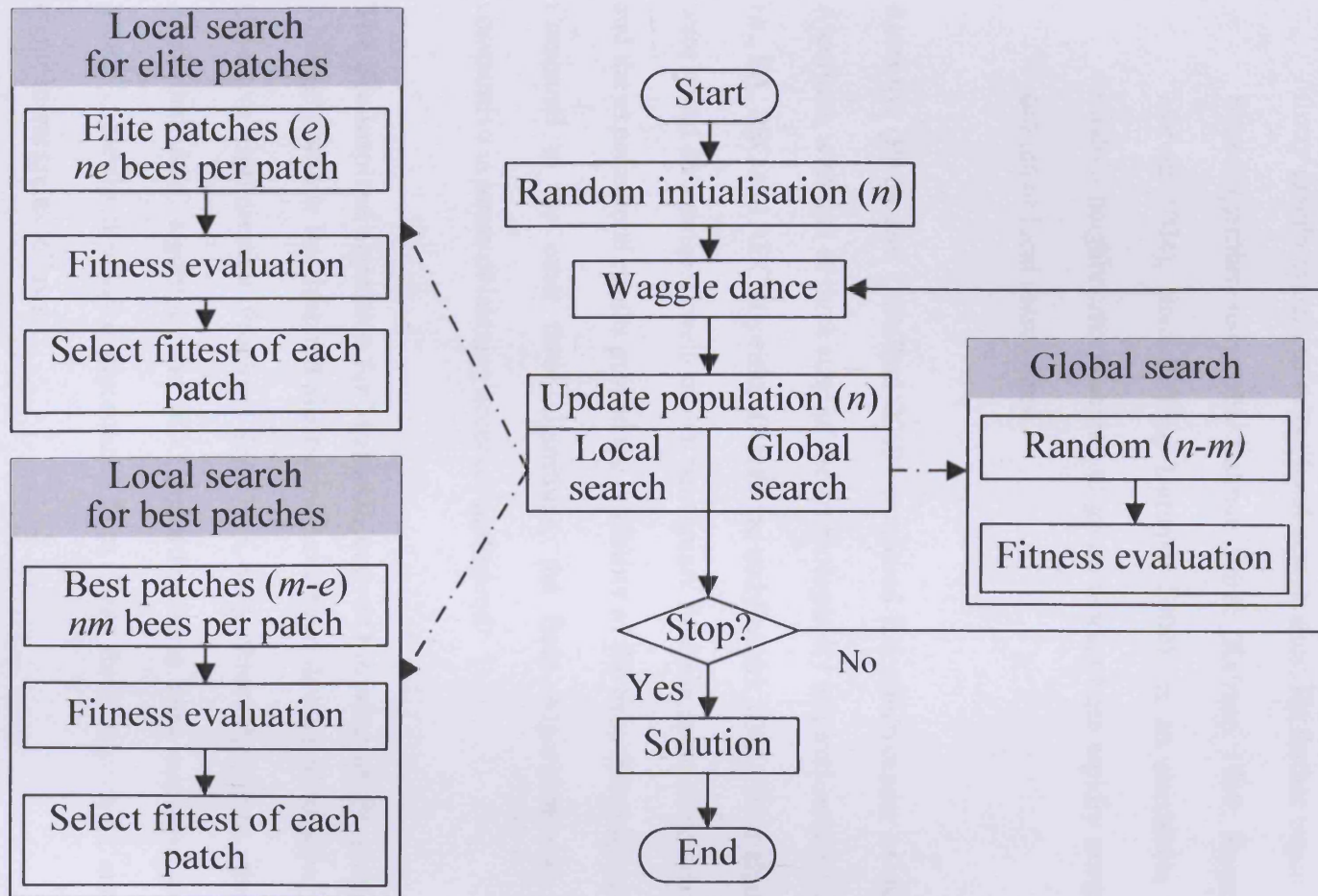


Figure 2.12 Flowchart of the Bees Algorithm

- (Haj Darwish 2009) developed an enhanced Bees Algorithm with Fuzzy Logic (Zadeh 1965) to reduce the number of parameters. It employed a fuzzy greedy system to select local search sites. He further improved the Bees Algorithm using the Kalman Filter (Kalman 1960; Russell and Norvig 2004), cited in (Haj Darwish 2009), as an alternative to the random neighbourhood search to guide worker bees rapidly towards the optima of local search sites.

Recently, (Pham and Castellani 2009) compared the effectiveness of the Bees Algorithm with that of three state of the art biologically inspired search methods, i.e., EA, PSO and ABC algorithm (Karaboga and Basturk 2008). Four algorithms were tested on a range of well-known benchmark function optimisation problems and the experimental results proved the reliability of the bees foraging metaphor. Compared to the other three algorithms, the Bees Algorithm was highly competitive in terms of learning accuracy and speed.

The Bee-inspired algorithm for Multi-Objective Optimisation Problems

Although various Bee-inspired algorithms have been developed to solve Single Objective Optimisation Problems (SOOPs), only three algorithms, the Bees Algorithm, ABC algorithm and BCO algorithm, have been employed to solve MOOPs. Newly developed algorithms from these three algorithms and their applications are as follows:

- **The Bees Algorithm**

- (Ghanbarzadeh 2007; Pham and Ghanbarzadeh 2007) introduced the Multi-Objective Bees Algorithm to solve the welded beam design problem which involves two nonlinear objective functions and seven constraints. In his work, the Bees Algorithm requires four parameters to be set such as n , m , nm and ngh . However, he added two strategies namely ‘shrinking neighbourhood size’ and ‘abandon sites without new information’ into the basic (or standard) Bees Algorithm, which means that it requires at least one additional parameter. For example, bees need to decide when they will abandon sites. For solving MOOPs, the m non-dominated sites are designated as ‘selected sites’ for the neighbourhood search. If there are more than m non-dominated sites in the population, the first m sites will be selected. Contrary to this, if there are less than m non-dominated sites, the others will be selected from those which have been dominated only once and this procedure is repeated until a sufficient number of sites are selected. When conducting a neighbourhood search, the representative bee in each selected patch will be the original one or it is displaced by a new dominant recruited bee, which is the same as that for the basic Bees Algorithm. One more step was

added to create a Pareto set in every iteration as for other swarm intelligent algorithms used as a MO solver.

- (Lee and Haj Darwish 2008) also applied the standard Bees Algorithm with the weighted sum method to solve the Environmental/Economic Dispatch Problem (EEDP) for power systems which involves two objective functions and two constraints. This method generates a unique solution (the fittest) at the end of the optimisation process. To approximate the Pareto front, the algorithm is run numerous times with different weights. Although their method obtained better results than other approaches in the literature, it is not guaranteed to converge to a true Pareto front. Since the number of solutions corresponds to the number of repeated runs, the size of the solution set is constrained by the algorithm running time.
- (Sayadi et al. 2009) proposed the Bees Algorithm-assisted Multi Objective Optimisation Problem (BAMOP), which is based on Pham (Pham and Ghanbarzadeh 2007), for a Multiple Input Multiple Output Multi Carrier-Code Division Multiple Access (MIMO MC-CDMA) system.
- (Mohamad Idris et al. 2009) also proposed the Multi-Objective Bees Algorithm (MOBA), which is also based on (Pham and Ghanbarzadeh 2007), for the optimal allocation of Flexible AC

Transmission System (FACTS) devices in restructured power systems.

- (Leeprechanon and Polratanasak 2010) also proposed the Multiobjective Bees Algorithm (MBA), which is based on the Bees Algorithm (Pham et al. 2006a), to solve the EEDP. Fuzzy C-Means (FCM) clustering (Dunn 1973; Matteucci) was employed to find m best solutions for neighbourhood search.

- **Artificial Bees Colony (ABC) algorithm**

- (Omkar et al. 2011) proposed Vector Evaluated Artificial Bee Colony (VEABC) algorithm for Multi-Objective design optimisation of composite structures. It is a Multi-Objective ABC method inspired by the concept and main ideas of VEGA (Schaffer 1985) algorithm and Vector Evaluated Particle Swarm Optimisation (VEPSO) (Omkar et al. 2008) algorithm. It is similar to two single objective functions being separately evaluated by separate artificial bees.

- **Bee Colony Optimisation (BCO) algorithm**

- (Low et al. 2009) proposed the Multi-Objective Bee Colony Optimisation (MOBCO) algorithm to solve ZDT (Zitzler et al. 2000) series of benchmark problems and Automated Red

Teaming (ART) which is a technique frequently used by the Military Operational Analysis community to uncover vulnerabilities in operational tactics.

- (Zeng et al. 2010) proposed the Autonomous Bee Colony Optimisation (A-BCO) algorithm for solving Multi-Objective numerical problems. In contrast with BCO algorithm, A-BCO utilizes a diversity-based performance metric to dynamically assess the archive set.

2.3 Summary

This chapter has reviewed background material relevant to the work presented in this thesis. The notations as well as the basic concepts of Multi-Objective Optimisation and Pareto optimality were surveyed. Multi-Objective solvers of both the classical and intelligent types, in particular population-based optimisation algorithms, were discussed. Specific attention was given to the Bees Algorithm and its theory, in order to provide general background information for the research in subsequent chapters of this thesis.

3 THE PARETO-BASED BEES ALGORITHM WITH MEMORISED SOLUTIONS FOR MULTI-OBJECTIVE PROBLEMS

3.1 Preliminaries

This chapter presents a newly developed Pareto-based Bees Algorithm for solving Multi-Objective Optimisation Problems (MOOPs). The algorithm is tested on two different applications:

- 1) nonlinear power system optimisation (the Environmental/Economic Dispatch Problem: EEDP) and
- 2) Multi-Objective optimisation of benchmark mathematical functions.

The bees in the new algorithm can ‘remember’ good locations discovered in the last harvest season. This memory is introduced to address shortcomings with existing techniques for Multi-Objective optimisation. Simulation results are compared to those obtained using other approaches to show the effectiveness of the proposed Bees Algorithm.

The Section 3.2 describes the two applications: the EEDP and benchmark mathematical functions. In Section 3.3, the proposed Pareto-based Bees

Algorithm with memorised solutions is described. Results are tabulated in Section 3.4 and the summary of this work is in Section 3.5.

3.2 Applications

3.2.1 The Environmental/Economic Dispatch Problem (EEDP)

Traditionally, solutions of the classical economic dispatch problem have focused on minimising the total fuel cost. However, due to rising public concerns about the environmental impact of fossil-fuelled electric power stations, a solution based only on the minimisation of the economic cost can no longer be considered acceptable. There are several strategies to reduce harmful atmospheric emissions such as installing pollution control equipment, switching to low-emission fuels, replacing aged fuel-burners and generators with more efficient ones, and emission dispatching (Abido 2006). The first three strategies can involve considerable costs. Moreover, due to the time-consuming procedures for the installation and modification of equipment, they can be regarded only as long-term solutions. For these reasons, the emission dispatching strategy has represented an attractive short-term alternative in recent years. In this case, the economic dispatch problem is re-stated, taking into account emissions as well as the fuel costs. As a result, the problem is treated as a Multi-Objective optimisation task with non-commensurable and conflicting goals.

The traditional way of solving a Multi-Objective optimisation problem consists of representing the goals via a single-objective (SO) function, and minimising this function whilst maintaining the physical constraints of the system (Lee and El-Sharkawi 2008). The evaluation result for the candidate solutions is thus expressed as a single value reflecting a compromise between the various conflicting goals. In the case of the economic dispatch problem, the SO function takes the form:

$$f(x) = w_1 \cdot f_c(x) + w_2 \cdot f_e(x) \quad \text{Eq. 3.1}$$

where $x = \{x_1, \dots, x_n\}$ is an n -dimensional vector of decision variables representing a feasible solution. f_c and f_e are respectively the fuel cost and emissions and w_1 and w_2 are user-defined weights. Although this technique is relatively simple to implement, it is not suitable for Multi-Objective applications because engineers may need to evaluate the detailed fitness of a solution for all the optimisation goals (f_c and f_e in the economic dispatch problem). In general, the set of solutions of a Multi-Objective optimisation problem lies on a hyper-surface formed by the best trade-offs (or Pareto front) which satisfy the multiple conflicting goals. In contrast to aggregation-based methods, intelligent techniques such as population-based algorithms are naturally suited to the direct generation of a Pareto front. This is because these methods work simultaneously to achieve solutions for the individual search objectives, and also evaluate multiple potential solutions in a single iteration. Therefore this generated Pareto

front can help a Decision Maker (DM)/Engineer to know all possible solutions throughout the various ranges. However, the challenge for intelligent techniques is to guide the search towards the Pareto optimal set whilst maintaining population diversity to prevent premature convergence.

Abido (Abido 2003a, b, c, 2006, 2007, 2009) pioneered research into applying intelligent Multi-Objective search techniques to the EEDP. He optimised the standard IEEE 30-bus system using the Non-dominated Sorting Genetic Algorithm- Π (NSGA- Π) (Ah King et al. 2005, 2006), Strength Pareto Evolutionary Algorithm (SPEA) (Abido 2003a), and Multi-Objective Particle Swarm Optimisation (MOPSO) (Abido 2007, 2009). All the above meta-heuristic techniques successfully address the limitations of classical approaches because they allow concurrent explorations of different points of the Pareto front, and also generate multiple solutions in a single run. Their main drawback is performance degradation as the number of objectives increases, since there are no computationally efficient methods to perform Pareto ranking. Furthermore, additional parameters (such as the 'sharing factor' and the number of Pareto samples) need to be introduced and tailored to suit (Lee and El-Sharkawi 2008).

In this chapter, test runs, utilising the proposed Bees Algorithm, were performed on the standard IEEE 30-bus 6-generator test system using the proposed Bees Algorithm. Figure 3.1 shows the test system layout.

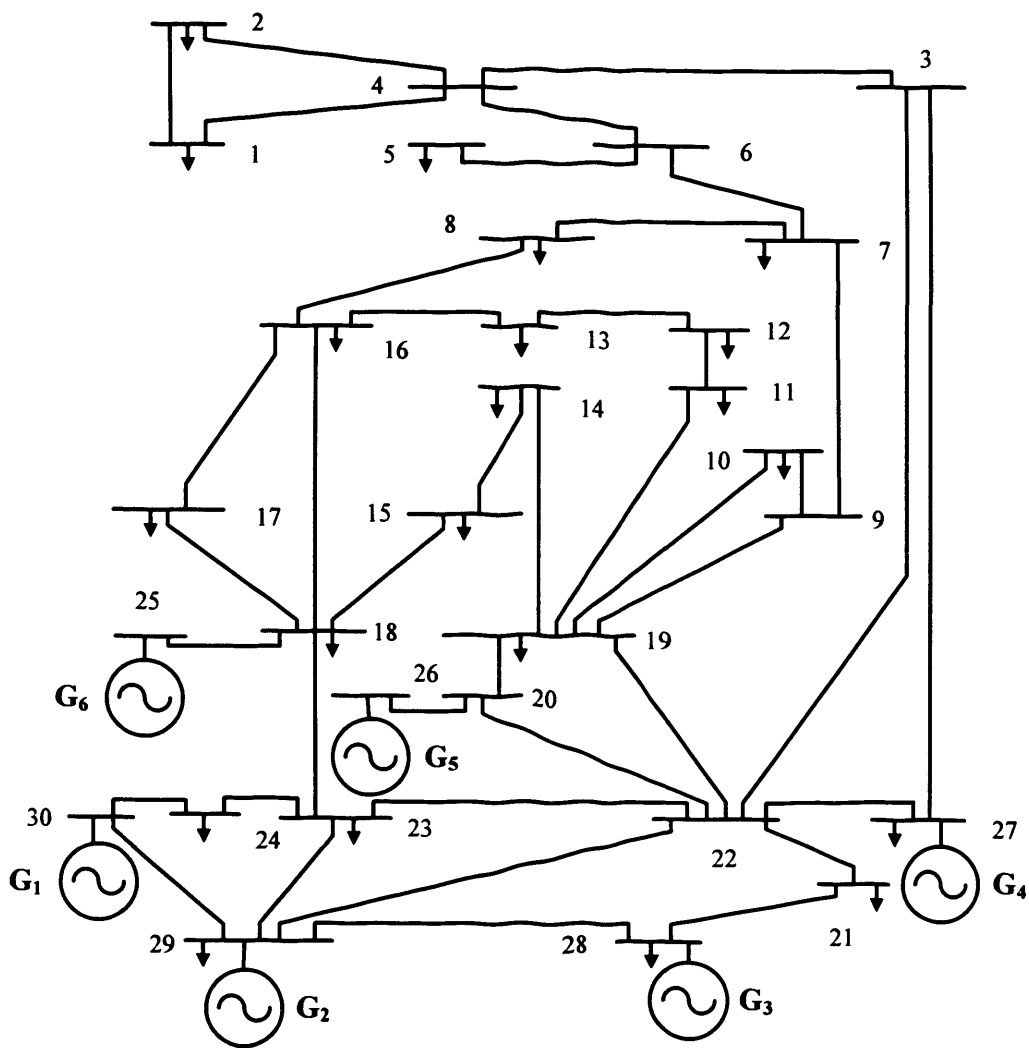


Figure 3.1 Single-line diagram of IEEE 30-bus test system

3.2.1.1 Objective functions

The EEDP for power generation requires the simultaneous achievement of two objectives:

1) Fuel cost objective

The requirement is to minimise the total fuel cost while satisfying the total demand. The equation used to determine the optimal combination for this problem is (Ah King et al. 2005):

$$f_c = \sum_{i=1}^n (a_i + b_i \times P_{Gi} + c_i \times P_{Gi}^2) \quad \text{Eq. 3.2}$$

where

f_c : total fuel cost (\$/hr),

a_i, b_i, c_i : fuel cost coefficients of generator i ,

P_{Gi} : power output by generator i , and

n : number of generators.

2) NOx emission objective

This objective is necessary to minimise the total NOx emission. The total NOx emission created by burning fossil fuels is expressed as:

$$f_e = \sum_{i=1}^n (a_{iN} + b_{iN} \times P_{Gi} + c_{iN} \times P_{Gi}^2 + d_{iN} \times \exp(e_{iN} \times P_{Gi})) \quad \text{Eq. 3.3}$$

where

f_e : total NOx emission (ton/hr),

$a_{iN}, b_{iN}, c_{iN}, d_{iN}$, and e_{iN} are NOx coefficients of the i^{th} generator emission characteristics.

3.2.1.2 Constraints

Two constraints need to be satisfied:

1) Power balance constraint

The total power generated must supply the total load demand plus the transmission losses expressed as:

$$\sum_{i=1}^n P_{Gi} - P_D - P_L = 0 \quad \text{Eq. 3.4}$$

where

P_D : total load demand and

P_L : transmission losses.

For this work, P_L was assumed to be 0.

2) Generation limits constraint

The power generated P_{Gi} by each generator is constrained between its minimum and maximum limits stated as:

$$P_{Gi\min} \leq P_{Gi} \leq P_{Gi\max} \quad \text{Eq. 3.5}$$

where

$P_{Gi\min}$: minimum power generated by generator i , and

$P_{Gi\max}$: maximum power generated by generator i .

3.2.1.3 Multi-Objective formulation

The Multi-Objective Environmental/Economic Dispatch optimisation problem is therefore formulated as:

$$\text{Minimise}[f_c, f_e] \quad \text{Eq. 3.6}$$

subject to: $\sum_{i=1}^n P_{Gi} - P_D = 0$ (power balance), and

$$P_{Gi\min} \leq P_{Gi} \leq P_{Gi\max} \text{ (generation limits).}$$

3.2.1.4 System parameters

The power system is interconnected by 41 transmission lines and the total system demand for the 21 load buses is 2.834(p.u.). Fuel cost and NOx emission coefficients for this system are given in Tables 3.1 (Yokoyama et al. 1988).

3.2.2 Benchmark mathematical Multi-Objective functions

In order to test the sensitivity of the shape of the Pareto front, the proposed algorithm was applied to three different function optimisation problems (Deb et al. 2002; Guzman et al. 2010). The equations of Multi-Objective function optimisation are shown in Table 3.2. While Schaffer's function (SCH) is a convex function, Fleming's study (FON) is concave. In addition, both SCH and FON have no constraints, while Tanaka et. al.'s study (TNK) is conducted with constraints.

Unit i	a_i	b_i	c_i	$P_{Gi\min}$	$P_{Gi\max}$
1	10	200	100	0.05	0.50
2	10	150	120	0.05	0.60
3	20	180	40	0.05	1.00
4	10	100	60	0.05	1.20
5	20	180	40	0.05	1.00
6	10	150	100	0.05	0.60

(a) Fuel cost coefficients

Unit i	a_{iN}	b_{iN}	c_{iN}	d_{iN}	e_{iN}
1	4.091e-2	-5.554e-2	6.490e-2	2.0e-4	2.857
2	2.543e-2	-6.047e-2	5.638e-2	5.0e-4	3.333
3	4.258e-2	-5.094e-2	4.586e-2	1.0e-6	8.000
4	5.326e-2	-3.550e-2	3.380e-2	2.0e-3	2.000
5	4.258e-2	-5.094e-2	4.586e-2	1.0e-6	8.000
6	6.131e-2	-5.555e-2	5.151e-2	1.0e-5	6.667

(b) NOx emission coefficients

Table 3.1 Coefficients for the test system

Problem	n	Variable bounds	Objective functions	Optimal solutions/Constraints
SCH (Convex)	1	$-10^3 \leq x \leq 10^3$	Minimise $F = (f_1(x), f_2(x))$, where $f_1(x) = x^2$ $f_2(x) = (x-2)^2$	$x \in [0,2]$
FON (Non-convex)	3	$-4 \leq x_i \leq 4$ $i = 1,2,3$	Minimise $F = (f_1(X), f_2(X))$, where $f_1(X) = 1 - \exp\left(-\sum_{i=1}^n \left(x_i - \frac{1}{\sqrt{n}}\right)^2\right)$ $f_2(X) = 1 - \exp\left(-\sum_{i=1}^n \left(x_i + \frac{1}{\sqrt{n}}\right)^2\right)$	$x_1 = x_2 = x_3 \in [-1/\sqrt{3}, 1/\sqrt{3}]$
TNK (with constraints)	2	$x_i \in [0, \pi]$ $i = 1,2$	Minimise $F = (f_1(X), f_2(X))$, where $f_1(X) = x_1$ $f_2(X) = x_2$	Subjected to $g_1(X) = x_1^2 + x_2^2 - 1 - 0.1 \cos\left(16 \arctan \frac{x_1}{x_2}\right) \geq 0$ $g_2(X) = (x_1 - 0.5)^2 + (x_2 - 0.5)^2 \leq 0.5$

Table 3.2 Equations for Multi-Objective mathematical function optimisation

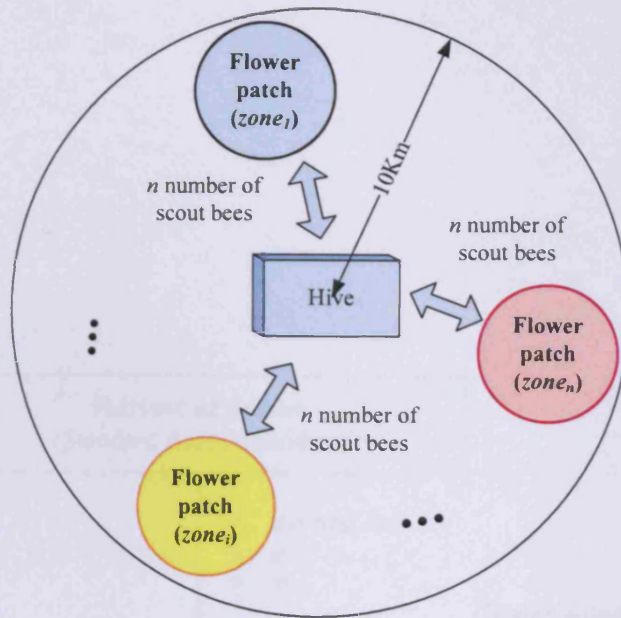
3.3 The Pareto-based Bees Algorithm with memorised solutions

The basic Bees Algorithm described in Chapter 2 is generally limited to SO applications. In order to apply the Bees Algorithm to a Multi-Objective problem, especially to generate a Pareto set, the complete food processing procedure of honey bees in the colony should be taken into account. The work by Lee and Darwish (Lee and Haj Darwish 2008) showed some shortcomings such as repetitive runs with different weights giving results which are not true Pareto solutions. Therefore a new parameter called ‘memorised solutions’ is introduced in this chapter. The new algorithm with memorised solutions consists of the following. A number of different combinations of weights w_1 and w_2 are defined for the SO function described in Eq. 3.1. For each combination of weights, the standard Bees Algorithm is run once. When one harvesting season is over (i.e., one run for a given pair of weights w_1 and w_2 is completed), the “bees” memorise the best locations (i.e., solutions). Each season, new locations will be memorised. This process of storing good solutions is reminiscent of the honey-making procedure which is carried out naturally by worker bees inside the hive. At the end of the last run of the standard Bees Algorithm, all memorised solutions will be used to generate the Pareto set. The number of locations that is memorised at

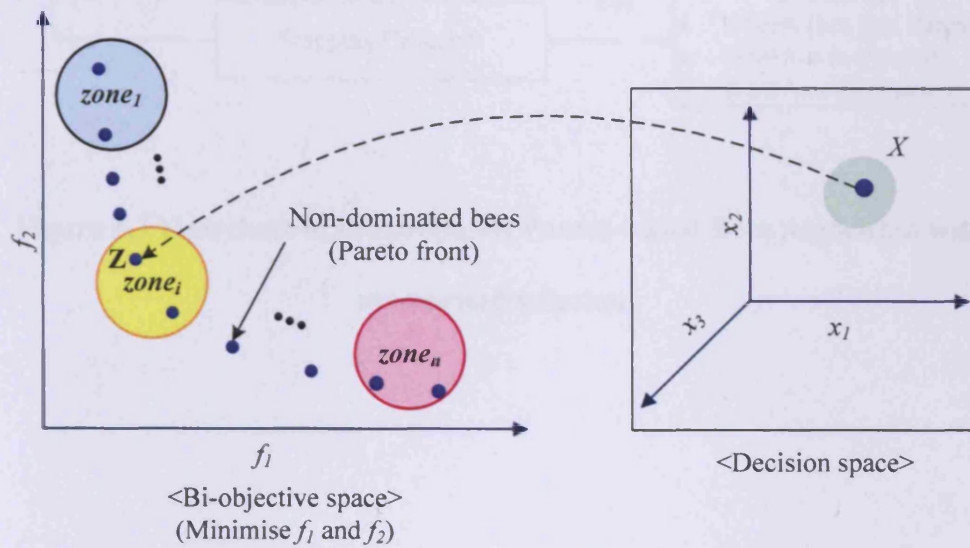
the end of each run is set by the user. The greater the number of memorised locations, the larger the Pareto set generated.

In this study, three ways to select the number of memorised locations will be evaluated, namely ‘the best one’, ‘ e ’ (corresponding to the number of elite sites) and ‘ m ’ (corresponding to the number of selected sites). Three different algorithms will be developed by utilising these three selection ways, namely ‘Weighted Sum Bees Algorithm (WSBA)’, ‘Weighted Sum Bees Algorithm with e sites (WSBA- e)’ and ‘Weighted Sum Bees Algorithm with m sites (WSBA- m)’. Figure 3.2 illustrates the concept for the Bees Algorithm with memorised solutions in the nature and objective space. Each ‘flower patch’ (from $zone_1$ to $zone_n$) represents a given pair of weights w_1 and w_2 and ‘the best one’, ‘ e ’ or ‘ m ’ selected sites are stored every iteration in each ‘flower patch’ to generate the Pareto set at the end.

Figure 3.3 shows the flowchart of the proposed Pareto-based Bees Algorithm with memorised solutions for MOOPs. The stopping criterion corresponds to the number of harvesting seasons (i.e., successive runs using different weighting combinations).



(a) Bee foraging in the nature



(b) Multi-Objective optimisation in the objective space

Figure 3.2 Illustration of the concept for the Bees Algorithm with memorised solutions

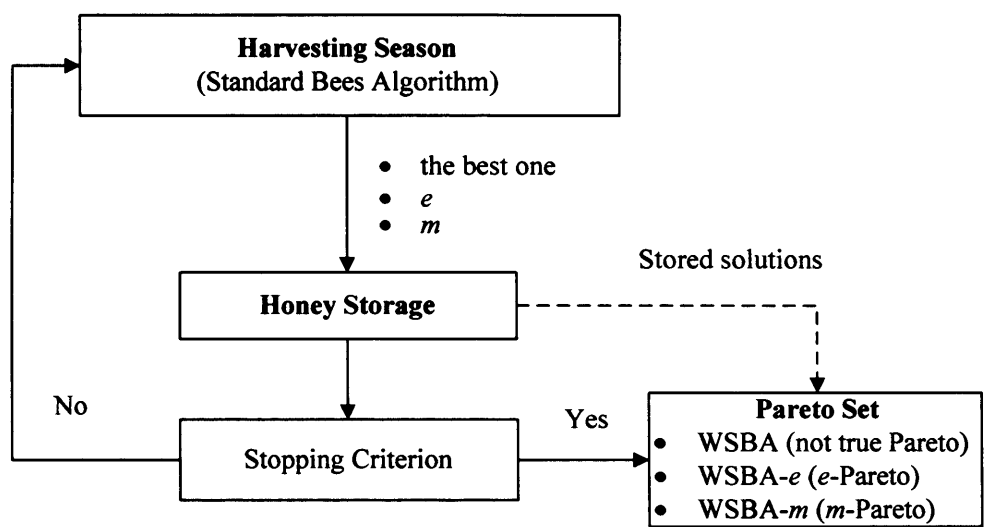


Figure 3.3 Flowchart of the proposed Pareto-based Bees Algorithm with memorised solutions

3.4 Results

3.4.1 Results of the EEDP

Table 3.3 shows the main parameter values used in this study for the proposed Pareto-based Bees Algorithm with memorised solutions. Two sets of respectively 11 and 22 parameters are given for the weight w_2 . To achieve good diversity, 11 of these are selected alternately. For the first set, the standard Bees Algorithm is iterated 11 times (harvesting seasons), and in the second it is iterated 22 times. The total generated power was required to satisfy the power balance constraint within an accuracy of ± 0.00001 (p.u.).

In the first set of tests, the effect of varying the initial patch size ngh (i.e., the breadth of the local search procedure (Abido 2006)) was evaluated. Three different combinations of ngh for the m selected and e best sites were tested and they are shown in Table 3.4 (a).

In the second set of tests, the effect of varying the number of memorised locations was studied. The five settings described in Table 3.4 (b) were evaluated. WSBA(22) (i.e., setting1) in Table 3.4 (b) is similar to (Lee and Haj Darwish 2008)'s work because the final optimal solutions are formed by each of the best

fit solutions from every weight combination instead of domination sorting. It is only included here to compare performance with the proposed Bees Algorithm such as WSBA-*e* and WSBA-*m*, although the final results cannot be claimed to be true Pareto solutions.

For each of the three combinations of patch sizes, four tests were carried out varying the duration (i.e., number of learning cycles) of one harvesting season (see Figure 3.3), and five tests were also carried out depending on the number of memorised solutions set. Thus, a total of sixty tests were undertaken. To obtain the average performance of the algorithm, 20 independent runs were carried out for each test to evaluate the statistics based on the average of two extreme solutions for minimum fuel cost and NOx emissions from all the non-dominated solutions of the Pareto set (Abido 2003b, c, 2007, 2009; Ah King et al. 2005).

3.4.1.1 Comparison of Multi-Objective Optimisation Algorithms

The performance of the proposed algorithm is compared to that of three state-of-the-art Multi-Objective optimisation methods, namely NSGA- II (Ah King et al. 2005), SPEA (Abido 2003a), and MOPSO (Abido 2007, 2009). Table 3.5 describes the shared symbols and values from Table 3.6 to Table 3.12.

<i>n</i> : the number of scout bees	20
<i>e</i> : the number of elite sites	3
<i>m</i> : the number of selected sites	8
<i>ne</i> : the number of recruited bees for elite sites	10
<i>nm</i> : the number of recruited bees for selected sites	5
<i>ngh</i> : patch size for neighbourhood search	0.01; 0.02
<i>itr</i> : the number of iterations	10; 15; 20; 25

(a) Parameters of the Bees Algorithm

set	w_1	1
1	w_2 (11)	1; 50; 100; 500; 900; 1,400; 3,000; 5,000; 10,000; 30,000; 70,000
2	w_2 (22)	1; 25; 50; 75; 100; 300; 500; 700; 900; 1,100; 1,400; 2,000; 3,000; 4,000; 5,000; 7,000; 10,000; 15,000; 30,000; 50,000; 70,000; 100,000

(b) Parameters of weighted sum method

Table 3.3 Parameters for the Environmental/Economic Dispatch Problem

<i>ngh</i> setting	elite sites (<i>e</i>)	selected sites (<i>m</i>)
1	0.02	0.01
2	0.01	0.02
3	0.02	0.02

(a) settings regarding patch size

setting	Algorithm (the number of weight combination)	Memorised solutions (per run)	Number of evolutions			
			<i>itr</i> =10	<i>itr</i> =15	<i>itr</i> =20	<i>itr</i> =25
1	WSBA(22)	The best one	15,180	22,550	29,920	37,290
2	WSBA- <i>e</i> (22)	<i>e</i>	15,180	22,550	29,920	37,290
3	WSBA- <i>e</i> (11)	<i>e</i>	7,590	11,275	14,960	18,645
4	WSBA- <i>m</i> (22)	<i>m</i>	15,180	22,550	29,920	37,290
5	WSBA- <i>m</i> (11)	<i>m</i>	7,590	11,275	14,960	18,645

(b) settings regarding the number of memorised solutions

Table 3.4 Test settings

Tables 3.6 and 3.7 show the average value of the best cost and emission results from 20 independent runs by the proposed Bees Algorithms. Data which correspond with MOPSO are denoted in bold type, whilst those which significantly outperform are highlighted in bold and underlined. All Data, (from 20 independent runs in 60 tests respectively), of f_c , $Corr. f_e$ and $Corr. f_c$ do not violate the assumption of normality, hence, a *One-Sample t-test* (Moore 2000) using *SPSS* program is carried out for a 5% alpha level of significance to compare the mean values with the results from MOPSO. All mean values of f_c are equal to or better than MOPSO's, but all mean values of $Corr. f_e$ are worse than MOPSO's, when the number of iterations is 15, 20 and 25. For 20 iterations, the proposed Bees Algorithm outperforms MOPSO in '*ngh setting2*', and only '*WSBA-e(22)*' and '*WSBA-m(22)*' outperform MOPSO in '*ngh setting3*'.

However, data of f_e violates the assumption of normality, therefore it cannot be compared to the mean value with other algorithms using a *One-Sample t-test*. Thus, if the average of f_e is exactly the same as other algorithms such as MOPSO, NSGA-II and SPEA which obtain the same best emission value of 0.1942 ton/h, it is highlighted and underlined in Table 3.7. When the number of iterations is 10, only '*WSBA(22)*', '*WSBA-e(22)*' and '*WSBA-m(22)*' in '*ngh setting1*' found the best emission value, although after 15 iterations, most of them found the best emission value which is exactly the same as the other algorithms.

In Tables 3.	Symbols	Definition
6, 8, 9 and 11	f_c	the average of the best fuel cost (US\$/h) from 20 independent runs
6, 9 and 11	$Corr. f_e$	the average of NOx emissions (ton/h) corresponding to f_c from 20 independent runs
7, 10 and 12	$Corr. f_c$	the average of fuel cost (US\$/h) corresponding to f_e from 20 independent runs
7, 10 and 12	f_e	the average of best NOx emissions (ton/h) from 20 independent runs
6 and 7	bold type	not significantly different to
6 and 7	bold and underlined	significantly better than
6 and 7	plain and underlined	the average value is equal to other algorithms such as MOPSO, NSGA- II and SPEA
8 to 12	√	significantly better (significantly lower mean or median value of $f_c, f_e, Corr. f_e,$ and $Corr. f_c$ from 20 independent runs)
8 to 12	-	not significantly different
8 to 12	×	significantly worse (significantly higher mean or median value of $f_c, f_e, Corr. f_e$ and $Corr. f_c$ from 20 independent runs)

(a) Symbols and their definitions

In Table 3.	Symbols	Values
6	f_c and $Corr. f_e$ by MOPSO	600.1200 (\$/h) and 0.22160 (ton/h) (Abido 2007)
7	$Corr. f_c$ and f_e by MOPSO	637.4200 (\$/h) and 0.19420 (ton/h) (Abido 2007)
8	The best of f_c by NSGA- II	600.1550 (\$/h) (Ah King et al. 2005)
8	The best of f_c by SPEA	600.1500 (\$/h) (Abido 2003a)

(b) Best cost and emission values from the literature

Table 3.5 Symbols used in Table 3.6 to 3.12

itr	Algorithm	ngh setting1		ngh setting2		ngh setting3	
		f_c	Corr. f_e	f_c	Corr. f_e	f_c	Corr. f_e
10	WSBA(22)	600.1317	0.22182	600.1989	0.22148	600.1351	0.22176
	WSBA-e(22)	600.1317	0.22182	600.1989	0.22148	600.1328	0.22170
	WSBA-e(11)	600.1377	0.22173	600.2157	0.22120	600.1413	0.22165
	WSBA-m(22)	600.1317	0.22182	600.1989	0.22148	600.1328	0.22170
	WSBA-m(11)	600.1377	0.22173	600.2157	0.22120	600.1413	0.22165
15	WSBA(22)	600.1216	0.22207	600.1202	0.22192	600.1210	0.22215
	WSBA-e(22)	600.1186	0.22204	600.1199	0.22197	600.1207	0.22222
	WSBA-e(11)	600.1217	0.22208	600.1238	0.22205	600.1208	0.22219
	WSBA-m(22)	600.1186	0.22204	600.1199	0.22197	600.1207	0.22222
	WSBA-m(11)	600.1217	0.22208	600.1238	0.22205	600.1208	0.22219
20	WSBA(22)	600.1216	0.22202	<u>600.1141</u>	0.22207	600.1193	0.22208
	WSBA-e(22)	600.1190	0.22207	<u>600.1137</u>	0.22209	<u>600.1179</u>	0.22203
	WSBA-e(11)	600.1201	0.22225	<u>600.1137</u>	0.22209	600.1184	0.22202
	WSBA-m(22)	600.1190	0.22207	<u>600.1137</u>	0.22209	<u>600.1179</u>	0.22205
	WSBA-m(11)	600.1201	0.22225	<u>600.1137</u>	0.22209	600.1184	0.22202
25	WSBA(22)	600.1186	0.22213	<u>600.1132</u>	0.22204	<u>600.1170</u>	0.22202
	WSBA-e(22)	<u>600.1168</u>	0.22208	<u>600.1123</u>	0.22207	<u>600.1161</u>	0.22201
	WSBA-e(11)	<u>600.1171</u>	0.22219	<u>600.1123</u>	0.22207	<u>600.1164</u>	0.22210
	WSBA-m(22)	<u>600.1168</u>	0.22208	<u>600.1123</u>	0.22207	<u>600.1161</u>	0.22201
	WSBA-m(11)	<u>600.1171</u>	0.22219	<u>600.1123</u>	0.22207	<u>600.1164</u>	0.22210

Table 3.6 Comparison of results between MOPSO and the proposed Bees Algorithm for finding the best fuel cost (f_c)

<i>itr</i>	Algorithm	<i>ngh</i> setting1		<i>ngh</i> setting2		<i>ngh</i> setting3	
		<i>Corr. f_c</i>	<i>f_e</i>	<i>Corr. f_c</i>	<i>f_e</i>	<i>Corr. f_c</i>	<i>f_e</i>
10	WSBA(22)	637.1675	<u>0.19420</u>	<u>636.4224</u>	0.19424	637.4160	0.19421
	WSBA-e(22)	637.3976	<u>0.19420</u>	<u>636.4224</u>	0.19424	637.6859	0.19421
	WSBA-e(11)	<u>636.5366</u>	0.19423	636.6893	0.19433	637.3291	0.19421
	WSBA-m(22)	637.3976	<u>0.19420</u>	<u>636.4224</u>	0.19424	637.6859	0.19421
	WSBA-m(11)	<u>636.5366</u>	0.19423	636.6893	0.19433	637.3291	0.19421
15	WSBA(22)	637.2134	<u>0.19420</u>	637.3715	<u>0.19420</u>	<u>637.1717</u>	<u>0.19420</u>
	WSBA-e(22)	637.5483	<u>0.19420</u>	637.5877	<u>0.19420</u>	637.6600	<u>0.19420</u>
	WSBA-e(11)	<u>637.0711</u>	<u>0.19420</u>	<u>636.6990</u>	0.19421	<u>637.1015</u>	<u>0.19420</u>
	WSBA-m(22)	637.5483	<u>0.19420</u>	637.5877	<u>0.19420</u>	637.6801	<u>0.19420</u>
	WSBA-m(11)	<u>637.0711</u>	<u>0.19420</u>	<u>636.6990</u>	0.19421	<u>637.1015</u>	<u>0.19420</u>
20	WSBA(22)	<u>637.0418</u>	<u>0.19420</u>	<u>637.2242</u>	<u>0.19420</u>	<u>637.1780</u>	<u>0.19420</u>
	WSBA-e(22)	637.4282	<u>0.19420</u>	637.4719	<u>0.19420</u>	637.6399	<u>0.19420</u>
	WSBA-e(11)	<u>636.8169</u>	<u>0.19420</u>	<u>636.9291</u>	<u>0.19420</u>	<u>637.0424</u>	<u>0.19420</u>
	WSBA-m(22)	637.4282	<u>0.19420</u>	637.4719	<u>0.19420</u>	637.6928	<u>0.19420</u>
	WSBA-m(11)	<u>636.8169</u>	<u>0.19420</u>	<u>636.9842</u>	<u>0.19420</u>	637.1621	<u>0.19420</u>
25	WSBA(22)	<u>637.1686</u>	<u>0.19420</u>	<u>637.1450</u>	<u>0.19420</u>	637.2277	<u>0.19420</u>
	WSBA-e(22)	637.5284	<u>0.19420</u>	637.3859	<u>0.19420</u>	637.6420	<u>0.19420</u>
	WSBA-e(11)	<u>636.9622</u>	<u>0.19420</u>	<u>637.0305</u>	<u>0.19420</u>	<u>637.1220</u>	<u>0.19420</u>
	WSBA-m(22)	637.5312	<u>0.19420</u>	637.3859	<u>0.19420</u>	637.6666	<u>0.19420</u>
	WSBA-m(11)	<u>636.9622</u>	<u>0.19420</u>	<u>637.1656</u>	<u>0.19420</u>	637.1494	<u>0.19420</u>

Table 3.7 Comparison of results between MOPSO and the proposed Bees Algorithm for finding the best NOx emissions (f_e)

Overall, the results after 15 iterations are equal to or better than MOPSO for both best cost and emission values. Therefore, only the result from 10 iterations is compared with other algorithms such as NSGA-II and SPEA, especially for the best cost (f_c), and Table 3.8 shows the comparison results. According to *One-Sample t-test*, the results obtained from the proposed algorithm are equal to or better than those from both NSGA-II and SPEA for the best cost (f_c), except with ‘*ngh setting2*’.

3.4.1.2 Effect of patch size

All the *ANOVA* (for normally distributed data with homogeneous variances) or *Kruskal-Wallis* (for non-parametric data) tests using the *SPSS* program are carried out for a 5% alpha level of significance to compare mean or median values from 20 independent runs.

In Table 3.9, when the number of iterations is 10, ‘*ngh setting1*’ and ‘*ngh setting3*’ outperform ‘*ngh setting2*’ for the best cost, but the results from 20 and 25 iterations are totally opposite to the results from 10 iterations. In 15 iterations, there are no significant differences regarding setting of neighbourhood size for the best cost.

For the best emissions, when the number of iterations is small (i.e., 10) in Table 3.10, ‘*ngh setting1*’ and ‘*ngh setting3*’ outperform ‘*ngh setting2*’. However, after

		Compared with NSGA- II			Compared with SPEA		
		<i>ngh</i> setting1	<i>ngh</i> setting2	<i>ngh</i> setting3	<i>ngh</i> setting1	<i>ngh</i> setting2	<i>ngh</i> setting3
<i>itr</i>	Algorithm	f_c	f_c	f_c	f_c	f_c	f_c
10	WSBA(22)	√	×	√	√	×	√
	WSBA- <i>e</i> (22)	√	×	√	√	×	√
	WSBA- <i>e</i> (11)	√	×	√	√	×	-
	WSBA- <i>m</i> (22)	√	×	√	√	×	√
	WSBA- <i>m</i> (11)	√	×	√	√	×	-

Table 3.8 Comparison of results between NSGA- II, SPEA and the proposed Bees Algorithms (*itr*=10) for the best cost (f_c)

<i>itr</i>	<i>ngh</i> setting	WSBA (22)		WSBA- <i>e</i> (22)		WSBA- <i>e</i> (11)		WSBA- <i>m</i> (22)		WSBA- <i>m</i> (11)	
		f_c	<i>Corr</i> $\cdot f_e$	f_c	<i>Corr</i> $\cdot f_e$	f_c	<i>Corr</i> $\cdot f_e$	f_c	<i>Corr</i> $\cdot f_e$	f_c	<i>Corr</i> $\cdot f_e$
10	1	√	-	√	-	√	-	√	-	√	-
	2	×	-	×	-	×	-	×	-	×	-
	3	√	-	√	-	√	-	√	-	√	-
15	1	-	-	-	-	-	-	-	-	-	-
	2	-	-	-	-	-	-	-	-	-	-
	3	-	-	-	-	-	-	-	-	-	-
20	1	×	-	×	-	×	-	×	-	×	-
	2	√	-	√	-	√	-	√	-	√	-
	3	×	-	×	-	×	-	×	-	×	-
25	1	×	-	×	-	×	-	×	-	×	-
	2	√	-	√	-	√	-	√	-	√	-
	3	×	-	×	-	×	-	×	-	×	-

Table 3.9 Comparison of results regarding neighbourhood size for the best cost (f_c)

<i>itr</i>	<i>ngh</i> setting	WSBA (22)		WSBA- <i>e</i> (22)		WSBA- <i>e</i> (11)		WSBA- <i>m</i> (22)		WSBA- <i>m</i> (11)	
		<i>Corr.</i> <i>f_c</i>	<i>f_e</i>	<i>Cor</i> <i>r.f_c</i>	<i>f_e</i>	<i>Corr</i> <i>.f_c</i>	<i>f_e</i>	<i>Corr</i> <i>.f_c</i>	<i>f_e</i>	<i>Corr</i> <i>.f_c</i>	<i>f_e</i>
10	1	-	√	-	√	-	√	-	√	-	√
	2	√	×	√	×	-	×	√	×	-	×
	3	×	√	×	√	-	√	×	√	-	√
15	1	-	-	-	-	-	-	-	-	-	-
	2	-	-	-	-	-	-	-	-	-	-
	3	-	-	-	-	-	-	-	-	-	-
20	1	-	-	-	-	-	-	-	-	-	-
	2	-	-	-	-	-	-	-	-	-	-
	3	-	-	-	-	-	-	-	-	-	-
25	1	-	-	-	-	-	-	-	-	-	-
	2	-	-	-	-	-	-	-	-	-	-
	3	-	-	-	-	-	-	-	-	-	-

Table 3.10 Comparison of results regarding neighbourhood size for the best emissions (f_e)

15 iterations, there are no significant differences regarding setting of neighbourhood size for the best emission.

In conclusion, when the number of iterations is small (i.e., 10), ‘*ngh* setting1’ and ‘*ngh* setting3’ outperform ‘*ngh* setting2’ for both the best cost and emission values but ‘*ngh* setting2’ outperforms the others in using larger (i.e., 20 and 25) numbers of iterations for only the best cost. In other words, if *ngh* for *e* sites is bigger than or equal to *m*’s *ngh*, it outperforms using a lower number of iterations for both the best cost and emissions, but if *ngh* for *e* is smaller than *m*’s *ngh*, it outperforms using a higher number of iterations for the best cost only.

3.4.1.3 Effect of number of memorised solutions

Tables 3.11 and 3.12 show comparison results regarding memorised solutions for the best cost and emissions. The results show there are no significant statistical differences regarding the use of *e* or *m* or different numbers of weights (22 or 11) to find the best cost in all iterations.

However, finding the best emission in ‘*ngh* setting1’ and ‘*ngh* setting2’ from 10 iterations, ‘WSBA(22)’, ‘WSBA-*e*(22)’ and ‘WSBA-*m*(22)’ outperform ‘WSBA-*e*(11)’ and ‘WSBA-*m*(11)’. After 15 iterations, there is no significant statistical difference regarding the use of *e* or *m* or different numbers of weights (22 or 11) to find the best emission.

<i>itr</i>	Algorithm	<i>ngh</i> setting1		<i>ngh</i> setting2		<i>ngh</i> setting3	
		f_c	<i>Corr. f_e</i>	f_c	<i>Corr. f_e</i>	f_c	<i>Corr. f_e</i>
10	WSBA(22)	-	-	-	-	-	-
	WSBA- <i>e</i> (22)	-	-	-	-	-	-
	WSBA- <i>e</i> (11)	-	-	-	-	-	-
	WSBA- <i>m</i> (22)	-	-	-	-	-	-
	WSBA- <i>m</i> (11)	-	-	-	-	-	-
15	WSBA(22)	-	-	-	-	-	-
	WSBA- <i>e</i> (22)	-	-	-	-	-	-
	WSBA- <i>e</i> (11)	-	-	-	-	-	-
	WSBA- <i>m</i> (22)	-	-	-	-	-	-
	WSBA- <i>m</i> (11)	-	-	-	-	-	-
20	WSBA(22)	-	-	-	-	-	-
	WSBA- <i>e</i> (22)	-	-	-	-	-	-
	WSBA- <i>e</i> (11)	-	-	-	-	-	-
	WSBA- <i>m</i> (22)	-	-	-	-	-	-
	WSBA- <i>m</i> (11)	-	-	-	-	-	-
25	WSBA(22)	-	-	-	-	-	-
	WSBA- <i>e</i> (22)	-	-	-	-	-	-
	WSBA- <i>e</i> (11)	-	-	-	-	-	-
	WSBA- <i>m</i> (22)	-	-	-	-	-	-
	WSBA- <i>m</i> (11)	-	-	-	-	-	-

Table 3.11 Comparison of results regarding memorised solutions for the best cost (f_c)

<i>itr</i>	Algorithm	<i>ngh</i> setting1		<i>ngh</i> setting2		<i>ngh</i> setting3	
		<i>Corr. f_c</i>	<i>f_e</i>	<i>Corr. f_c</i>	<i>f_e</i>	<i>Corr. f_c</i>	<i>f_e</i>
10	WSBA(22)	×	√	-	√	-	-
	WSBA- <i>e</i> (22)	×	√	-	√	-	-
	WSBA- <i>e</i> (11)	√	×	-	×	-	-
	WSBA- <i>m</i> (22)	×	√	-	√	-	-
	WSBA- <i>m</i> (11)	√	×	-	×	-	-
15	WSBA (22)	-	-	×	-	-	-
	WSBA- <i>e</i> (22)	×	-	×	-	×	-
	WSBA- <i>e</i> (11)	√	-	√	-	√	-
	WSBA- <i>m</i> (22)	×	-	×	-	×	-
	WSBA- <i>m</i> (11)	√	-	√	-	√	-
20	WSBA(22)	√	-	×	-	-	-
	WSBA- <i>e</i> (22)	×	-	×	-	×	-
	WSBA- <i>e</i> (11)	√	-	√	-	√	-
	WSBA- <i>m</i> (22)	×	-	×	-	×	-
	WSBA- <i>m</i> (11)	√	-	√	-	-	-
25	WSBA(22)	-	-	√		-	-
	WSBA- <i>e</i> (22)	×	-	×		×	-
	WSBA- <i>e</i> (11)	√	-	√		√	-
	WSBA- <i>m</i> (22)	×	-	×		×	-
	WSBA- <i>m</i> (11)	√	-	√		√	-

Table 3.12 Comparison results regarding memorised solutions for the best emissions (f_e)

In conclusion, the use of e or m or different numbers of weights (22 or 11) does not affect finding the best cost in all iterations and it also has no effect on finding the best emissions using 15, 20 and 25 iterations. However, it is noteworthy that the number of evaluations of 'WSBA- e (11)' and 'WSBA- m (11)' are only half the number of 'WSBA(22)', 'WSBA- e (22)' and 'WSBA- m (22)', which shows that the proposed algorithms (WSBA- e and WSBA- m) can find the best cost and emissions with less effort than by 'WSBA(22)' and the work by (Lee and Haj Darwish 2008).

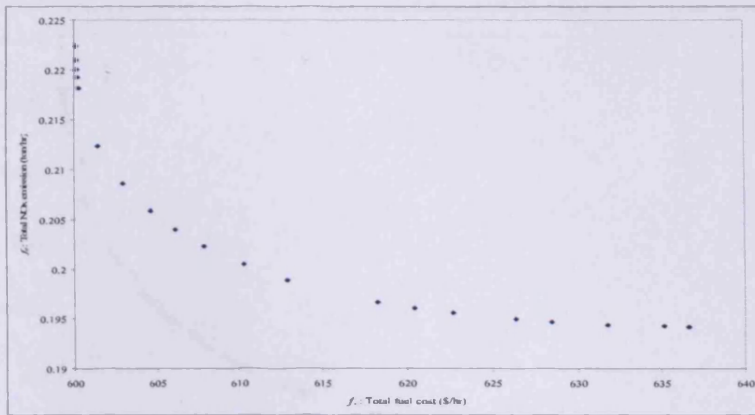
3.4.1.4 Comparison of the number of solutions and Pareto fronts

Table 3.13 shows the average number of solutions from each of the 20 independent runs. The number of solutions from 'WSBA(22)' is the same as the number of weight combinations because the final solution is formed from the best fit for every weight combination. The results show the number of solutions from m memorised solutions is always larger than e memorised solutions and using 22 weight combinations is almost double the number of solutions when using 11 weight combinations. However, the number of Pareto solutions from the proposed algorithm is less than MOPSO (Abido 2009) which obtains 147.

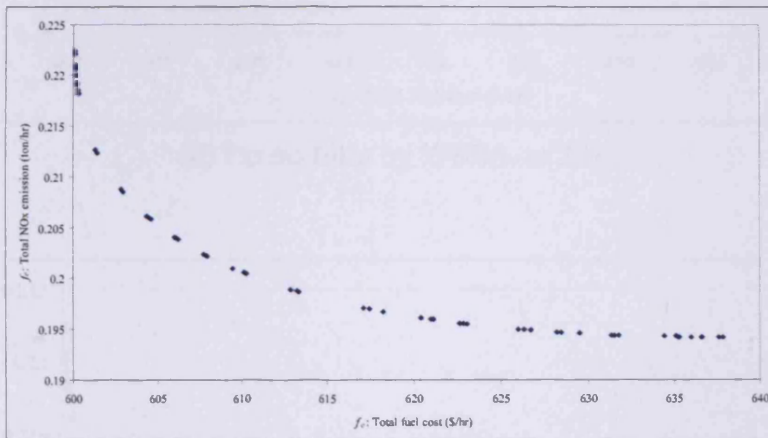
Figure 3.4 graphically shows the final result of solutions from each proposed approach in objective space. All data is taken from test number 3 of 20 runs in 'ngh setting3' of neighbourhood size when the number of iterations is 25.

<i>itr</i>	Algorithm	The average number of solutions		
		<i>ngh</i> setting1	<i>ngh</i> setting2	<i>ngh</i> setting3
10	WSBA(22)	22	22	22
	WSBA- <i>e</i> (22)	40	28	41
	WSBA- <i>e</i> (11)	23	18	24
	WSBA- <i>m</i> (22)	41	30	44
	WSBA- <i>m</i> (11)	25	22	27
15	WSBA(22)	22	22	22
	WSBA- <i>e</i> (22)	53	47	55
	WSBA- <i>e</i> (11)	28	26	29
	WSBA- <i>m</i> (22)	56	55	67
	WSBA- <i>m</i> (11)	34	35	40
20	WSBA(22)	22	22	22
	WSBA- <i>e</i> (22)	56	59	58
	WSBA- <i>e</i> (11)	29	30	30
	WSBA- <i>m</i> (22)	66	83	86
	WSBA- <i>m</i> (11)	41	48	48
25	WSBA(22)	22	22	22
	WSBA- <i>e</i> (22)	58	62	60
	WSBA- <i>e</i> (11)	30	31	31
	WSBA- <i>m</i> (22)	80	97	100
	WSBA- <i>m</i> (11)	48	54	55

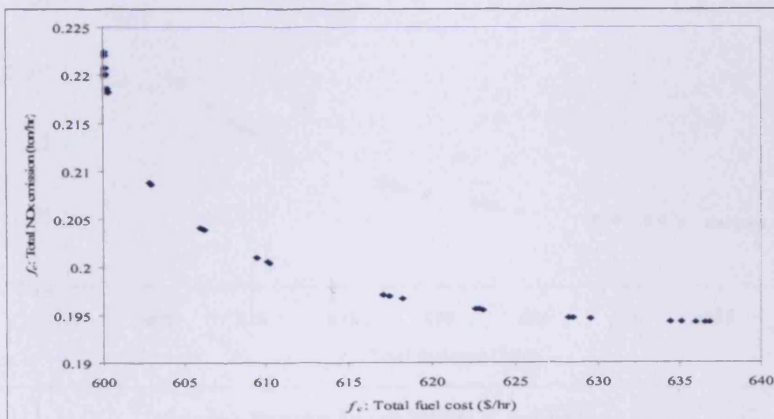
Table 3.13 The average number of solutions for each 20 independent runs



(a) Result by WSBA(22)

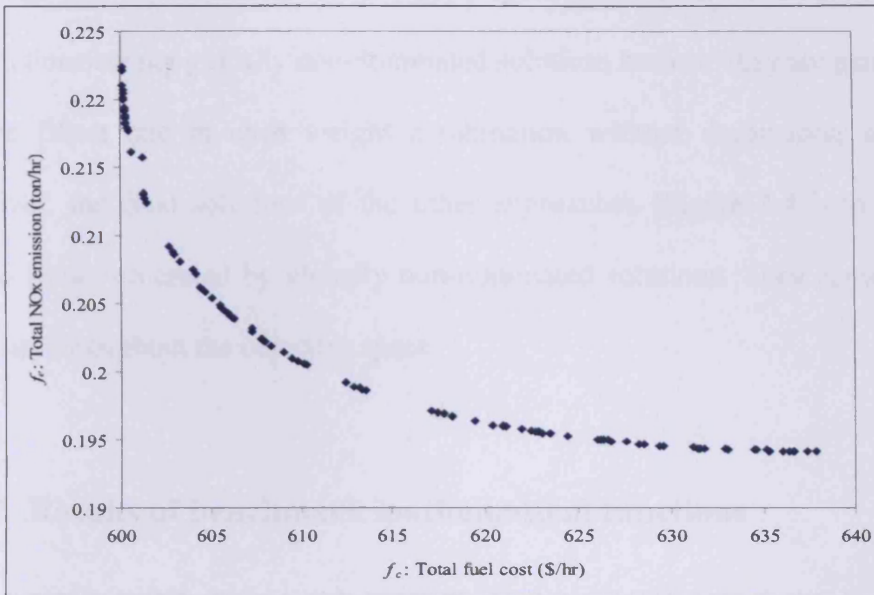


(b) Pareto front by WSBA-e(22)

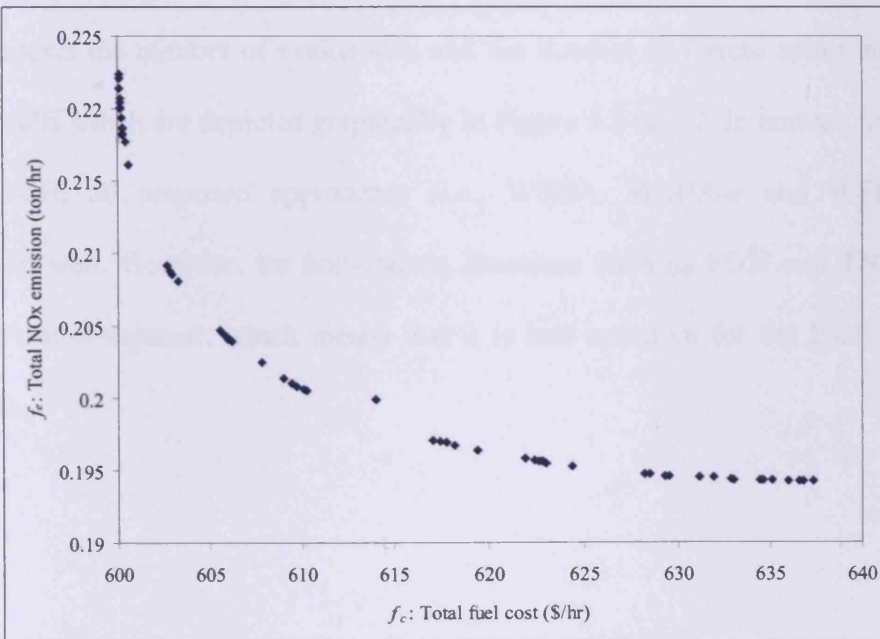


(c) Pareto front by WSBA-e(11)

Figure 3.4 The result of the EEDP



(d) Pareto front by WSBA-m(22)



(e) Pareto front WSBA-m(11)

Figure 3.4 continued

Although the final solution of WSBA(22) (Figure 3.4 a) resembles a Pareto front, all solutions are not globally non-dominated solutions because they are generated by the fittest one in each weight combination without domination sorting. However, the final solutions of the other approaches (Figure 3.4 b to e) are Pareto fronts generated by globally non-dominated solutions. They show good diversity throughout the objective space.

3.4.2 Results of benchmark mathematical functions

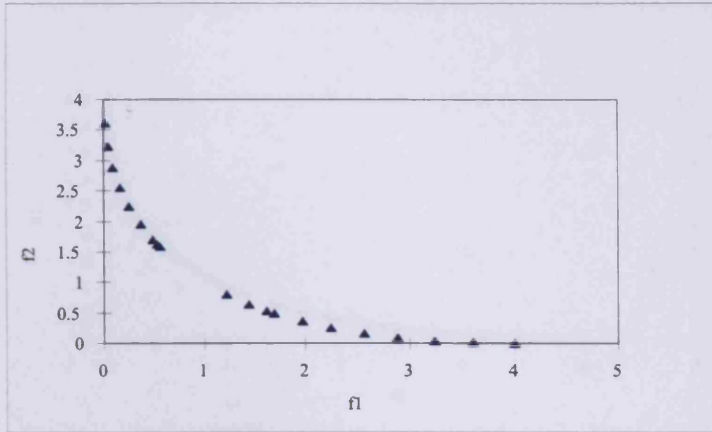
In order to evaluate the sensitivity to the shape of the Pareto front, three different functions are tested. Table 3.14 shows all parameters used in this work and Table 3.15 shows the number of evaluations and the number of Pareto solutions from the results which are depicted graphically in Figure 3.5 to 3.7. In convex function like SCH, all proposed approaches (i.e., WSBA, WSBA-*e* and WSBA-*m*) perform well. However, for non-convex functions such as FON and TNK, the WSBA-*m* is superior, which means that it is less sensitive for the Pareto front profile.

Parameters	SCH	FON and TNK
<i>n</i>	30	30
<i>e</i>	2	3
<i>m</i>	5	8
<i>ne</i>	10	10
<i>nm</i>	5	5
<i>ngh</i>	0.01	0.01
<i>itr</i>	30	30
<i>w₁</i>	1- <i>w₂</i>	
<i>w₂</i>	0.05; 0.1; 0.15; 0.2; 0.25; 0.3; 0.35; 0.4; 0.45; 0.5; 0.55; 0.6; 0.65; 0.7; 0.75; 0.8; 0.85; 0.9; 0.95; 1.0	

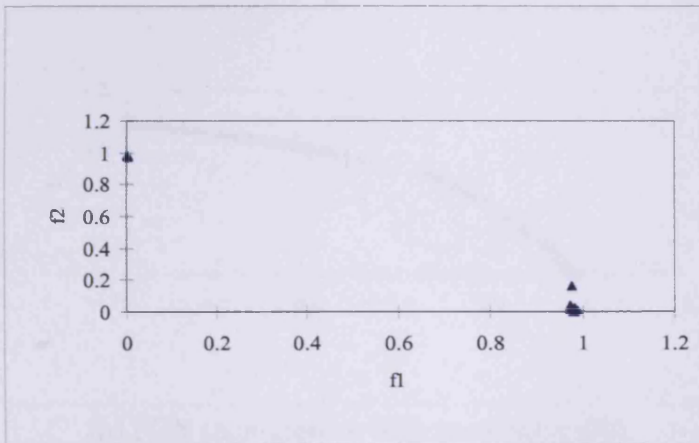
Table 3.14 Parameters of the Bees Algorithm for benchmark mathematical functions

Algorithm (the number of weight combination)	Memorised solutions (per run)	Number of evolutions			Number of Pareto solutions		
		SCH	FON	TNK	SCH	FON	TNK
WSBA(20)	The best one	36,600	46,800	46,800	20	20	20
WSBA- e (20)	e	36,600	46,800	46,800	801	173	138
WSBA- m (20)	m	36,600	46,800	46,800	2,210	196	275

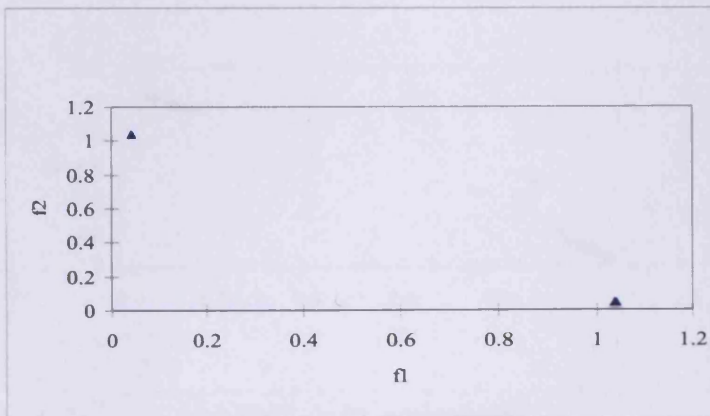
Table 3.15 Results for benchmark mathematical functions



(a) SCH (convex with no constraints)

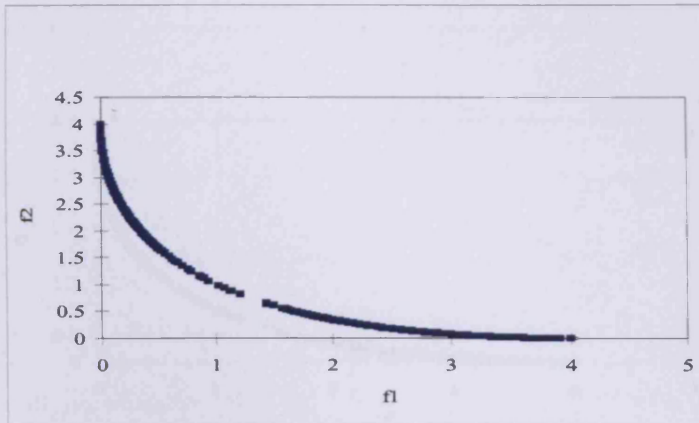


(b) FON (non-convex with no constraints)

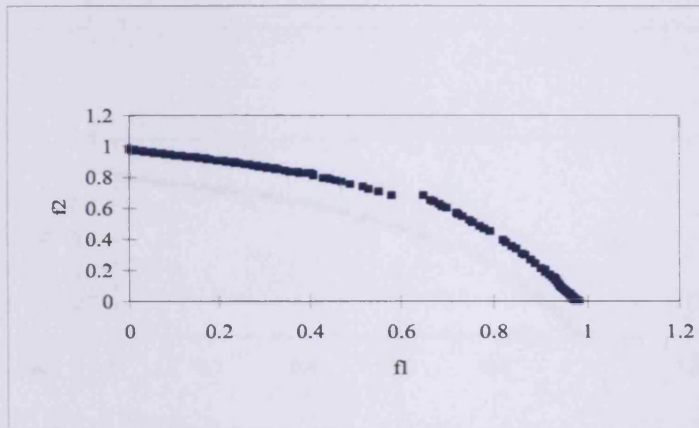


(c) TNK (with constraints)

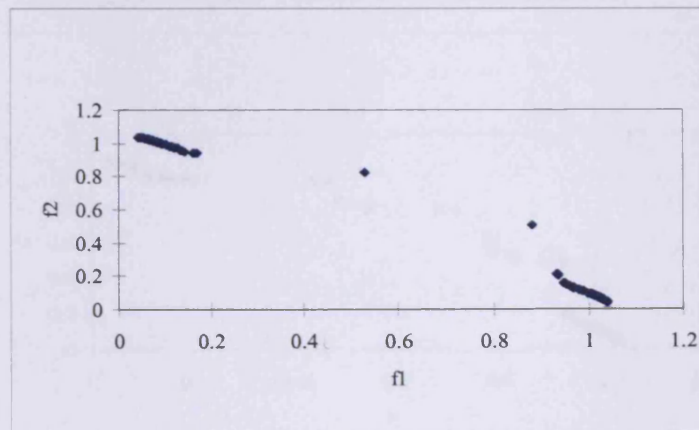
Figure 3.5 The results by WSBA(20)



(a) SCH (convex with no constraints)

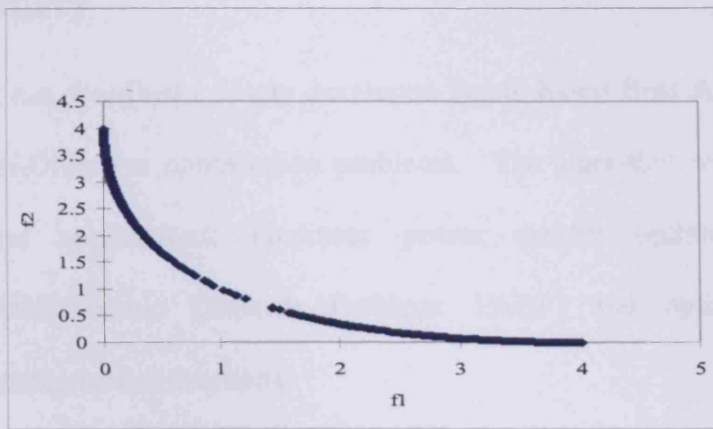


(b) FON (non-convex with no constraints)

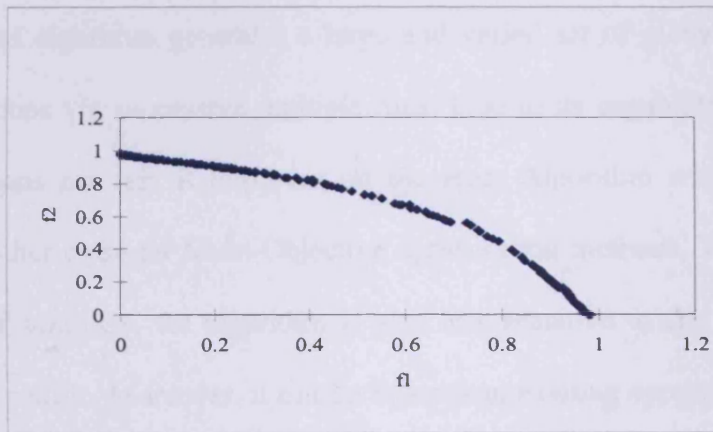


(c) TNK (with constraints)

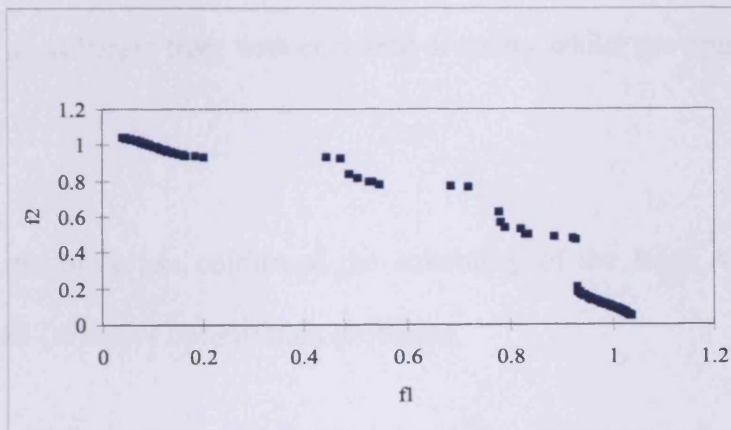
Figure 3.6 The Pareto fronts by WSBA- $e(20)$



(a) SCH (convex with no constraints)



(b) FON (non-convex with no constraints)



(c) TNK (with constraints)

Figure 3.7 The Pareto fronts by WSBA- $m(20)$

3.5 Summary

This chapter has described a newly developed Pareto-based Bees Algorithm for solving Multi-Objective optimisation problems. The algorithm was tested on two different applications: nonlinear power system optimisation (the Environmental/Economic Dispatch Problem: EEDP) and optimisation of benchmark mathematical functions.

The proposed algorithm generates a large and varied set of globally dominant Pareto solutions via successive multiple runs. Due to its capability to generate many solutions per run, it improves on the Bees Algorithm with memorised zones and other classical Multi-Objective optimisation methods. Thanks to the wider set of solutions, the algorithm is also less sensitive to the shape of the Pareto front profile. Moreover, it can be faster than existing approaches because it uses fewer evaluations. The results obtained show that the proposed algorithm delivered a good Pareto front with excellent diversity whilst proving to be stable and robust.

Therefore, this work has confirmed the suitability of the Bees Algorithm for solving Multi-Objective optimisation problems.

4 AN ENHANCED BEES ALGORITHM FOR MULTI-OBJECTIVE PROBLEMS

4.1 Preliminaries

This chapter presents a globally and locally enhanced Bees Algorithm for solving MOOPs. The algorithm is tested on the same two applications as in Chapter 3

- 1) nonlinear power system optimisation (EEDP) and
- 2) Multi-Objective optimisation of benchmark mathematical functions
(additional functions are also tested here than in the previous chapter).

The results of the proposed algorithm are compared with MOPSO (Abido 2009), NSGA-II (Ah King et al. 2005) and SPEA (Abido 2003a). They are also compared with the results of the Pareto-based Bees Algorithm with memorised solutions in Chapter 3.

This chapter is organised as follows: Section 4.2 presents a brief description of the two applications and the proposed enhanced Bees Algorithm is developed in Section 4.3. The results of test runs and their comparisons with other existing methods are carried out in Section 4.4. The final conclusions are presented in Section 4.5.

4.2 Applications

4.2.1 The Environmental/Economic Dispatch Problem (EEDP)

In order to compare results with the Pareto-based Bees Algorithm with memorised solutions from Chapter 3, the proposed algorithm in this chapter is tested on the same application which is the EEDP. All equations, system parameters and conditions are exactly the same as those used in Chapter 3.

4.2.2 Benchmark mathematical Multi-Objective functions

In the context of Multi-Objective optimisation, sufficient obstacles must be presented in test functions to impede the search for Pareto optimal solutions (Tan et al. 2008). Several characteristics have been identified by (Deb 1999) that may challenge an algorithm's ability to converge and maintain population diversity. Multi-modality, convexity, discontinuity and non-uniformity of the Pareto front may prevent population-based heuristics from finding a diverse set of solutions (Tan et al. 2008).

Test problems are employed from past studies in this area, which consists of 12 benchmark problems: SCH, FON, POL, KUR, ZDT1, ZDT2, ZDT3, CONSTR, SRN, TNK, WATER and the three-objective functions problem.

Firstly, seven problems without constraints were chosen. These seven problems can be categorised into two groups, which are convex and concave functions. Table 4.1 outlines convex problems and Table 4.2 outlines concave problems. Tables also show the number of variables, their bounds, the Pareto-optimal solutions and the nature of the Pareto optimal front for each problem. Although (Zitzler et al. 2000) proposed six test problems (ZDT1 to ZDT6), only three ZDT test problems, identified as ZDT1, ZDT2 and ZDT3, are employed in this chapter. The characteristics of each function are as follows:

- Convex functions
 - Schaffer's study (SCH) (Schaffer 1984, 1985)
 - ZDT1 has a convex Pareto front with a large number of variables to be optimised.
 - ZDT3 has a disconnected convex Pareto front with a large number of variables to be optimised.
- Concave functions
 - Fonseca and Fleming's study (FON) (Fonseca and Fleming 1998) challenges the algorithm's ability to find and maintain the entire trade-offs curve uniformly which is characterised by an optimal Pareto front that is non-convex and disconnected.
 - Poloni's study (POL) (Poloni 1995)
 - Kursawe's study (KUR) (Kursawe 1991) is also characterised by an optimal Pareto front that is non-convex and disconnected. It

contains three distinct disconnected regions in the final trade-offs and it is difficult to discover the decision variables which correspond to the global trade-offs because they are also disconnected in the decision variable space (Tan et al. 2008).

- ZDT2 has a non-convex Pareto front with a large number of variables to be optimised.

Secondly, four test problems with constraints that have been used in earlier studies were selected, and they are described in Table 4.3. Their characteristics are as follows:

- CONSTR: a part of the unconstrained Pareto optimal region is not feasible. Thus, the resulting constrained Pareto optimal region is a concatenation of the first constraint boundary and also with some part of the unconstrained Pareto optimal region.
- SRN was used in the original study of NSGA (Srinivas and Deb 1994) and the constrained Pareto optimal set is a subset of the unconstrained Pareto optimal set.
- TNK was suggested by (Tanaka et al. 1995) and has a discontinuous Pareto optimal region, falling entirely on the first constraint boundary.
- WATER is a five-objective with seven-constraints problem which (Ray et al. 2001) attempted to solve. With five objectives, it is too complex to

discuss the effect of the constraints on the unconstrained Pareto optimal region.

Finally, Table 4.4 outlines a three-objective function problem to find the global Pareto front in the three dimensional objective space (Seshadri 2006).

Many researchers still use these problems to validate their algorithms. For example,

- (Deb et al. 2002; Seshadri 2006) applied NSGA- II ,
- (Seshadri 2006) applied MOEA,
- (Tan et al. 2008) applied Evolutionary Artificial Immune System,
- (Park et al. 2009) applied GA and Artificial Immune System (AIS) and
- (Guzman et al. 2010) applied Bacterial Chemotaxis Multiobjective Optimisation Algorithm (BCMOA).

Therefore these problems should serve as good test samples for a fair comparison of different Multi-Objective algorithms.

Problem	n	Variable bounds	Objective functions	Optimal solutions	Comments
SCH	1	$-10^3 \leq x \leq 10^3$	Minimise $F = (f_1(x), f_2(x))$, where $f_1(x) = x^2$ $f_2(x) = (x - 2)^2$	$x \in [0, 2]$	Convex
ZDT1	30	$x_i \in \{0, 1\}$	Minimise $F = (f_1(X), f_2(X))$, where $f_1(X) = x_1$ $f_2(X, g) = g(X) \cdot (1 - \sqrt{f_1/g(X)})$ $g(X) = 1 + \frac{9}{n-1} \sum_{i=2}^n x_i$	$x_1 \in [0, 1]$ $x_i = 0,$ $i = 2, \dots, n$	Convex
ZDT3	30	$x_i \in \{0, 1\}$	Minimise $F = (f_1(X), f_2(X))$, where $f_1(X) = x_1$ $f_2(X, g) = g(X) \cdot \left(1 - \sqrt{\frac{f_1}{g(X)}} - \frac{f_1}{g(X)} \cdot \sin(10\pi f_1)\right)$ $g(X) = 1 + \frac{9}{n-1} \cdot \sum_{i=2}^n x_i$	$x_1 \in [0, 1]$ $x_i = 0,$ $i = 2, \dots, n$	Convex, disconnected

Table 4.1 Convex functions without constraints

Problem	n	Variable bounds	Objective functions	Optimal solutions	Comments
FON	3	$-4 \leq x_i \leq 4$ $i = 1, 2, 3$	<p>Minimise $F = (f_1(X), f_2(X))$, where</p> $f_1(X) = 1 - \exp\left(-\sum_{i=1}^n \left(x_i - \frac{1}{\sqrt{n}}\right)^2\right)$ $f_2(X) = 1 - \exp\left(-\sum_{i=1}^n \left(x_i + \frac{1}{\sqrt{n}}\right)^2\right)$	$x_1 = x_2 = x_3$ $\in [-1/\sqrt{3}, 1/\sqrt{3}]$	Non-convex
POL	2	$-\pi \leq x_1, x_2 \leq \pi$	<p>Minimise $F = (f_1(x_1, x_2), f_2(x_1, x_2))$, where</p> $f_1(x_1, x_2) = 1 + (A_1 - B_1)^2 + (A_2 - B_2)^2$ $f_2(x_1, x_2) = (x_1 + 3)^2 + (x_2 + 1)^2$ $A_1 = 0.5 \sin 1 - 2 \cos 1 + \sin 2 - 1.5 \cos 2,$ $A_2 = 1.5 \sin 1 - \cos 1 + 2 \sin 2 - 0.5 \cos 2,$ $B_1 = 0.5 \sin x_1 - 2 \cos x_1 + \sin x_2 - 1.5 \cos x_2,$ $B_2 = 1.5 \sin x_1 - \cos x_1 + 2 \sin x_2 - 0.5 \cos x_2$		Non-convex, disconnected (2 Pareto curves)

Table 4.2 Non-convex functions without constraints

Problem	n	Variable bounds	Objective functions	Optimal solutions	Comments
KUR	3	$-5 \leq x_i \leq 5,$ $i = 1, 2, 3$	Minimise $F = (f_1(X), f_2(X))$, where $f_1(X) = \sum_{i=1}^{n-1} \left(-10 \exp \left(-0.2 \times \sqrt{x_i^2 + x_{i+1}^2} \right) \right)$ $f_2(X) = \sum_{i=1}^n \left(x_i ^{0.8} + 5 \sin x_i^3 \right)$		Non-convex, disconnected (3 Pareto curves)
ZDT2	30	$x_i \in \{0, 1\}$	Minimise $F = (f_1(X), f_2(X))$, where $f_1(X) = x_1$ $f_2(X, g) = g(X) \cdot \left(1 - \left(\frac{f_1}{g(X)} \right)^2 \right)$ $g(X) = 1 + \frac{9}{n-1} \cdot \sum_{i=2}^n x_i$	$x_1 \in [0, 1]$ $x_i = 0,$ $i = 2, \dots, n$	Non-convex

Table 4.2 continued

Problem	n	Variable bounds	Objective functions	Constraints
CONSTR	2	$x_1 \in [0.1, 1.0]$ $x_2 \in [0, 5]$	Minimise $F = (f_1(X), f_2(X))$, where $f_1(X) = x_1$ $f_2(X) = (1 + x_2)/x_1$	Subjected to $g_1(X) = x_2 + 9x_1 \geq 6$ $g_2(X) = -x_2 + 9x_1 \geq 1$
SRN	2	$x_i \in [-20, 20]$ $i = 1, 2$	Minimise $F = (f_1(X), f_2(X))$, where $f_1(X) = (x_1 - 2)^2 + (x_2 - 1)^2 + 2$ $f_2(X) = 9x_1 - (x_2 - 1)^2$	Subjected to $g_1(X) = x_1^2 + x_2^2 \leq 225$ $g_2(X) = x_1 - 3x_2 \leq 10$
TNK	2	$x_i \in [0, \pi]$ $i = 1, 2$	Minimise $F = (f_1(X), f_2(X))$, where $f_1(X) = x_1$ $f_2(X) = x_2$	Subjected to $g_1(X) = x_1^2 + x_2^2 - 1 - 0.1 \cos\left(16 \arctan \frac{x_1}{x_2}\right) \geq 0$ $g_2(X) = (x_1 - 0.5)^2 + (x_2 - 0.5)^2 \leq 0.5$

Table 4.3 Functions with constraints

Problem	n	Variable bounds	Objective functions	Constraints
WATER	3	$0.01 \leq x_1 \leq 0.45$ $0.01 \leq x_2 \leq 0.10$ $0.01 \leq x_3 \leq 0.10$	Minimise $F = (f_1(X), f_2(X), f_3(X), f_4(X), f_5(X))$, where $f_1(X) = 106780.37(x_2 + x_3) + 61704.67$ $f_2(X) = 3000x_1$ $f_3(X) = \frac{(305700) \cdot 2289x_2}{(0.06 \times 2289)^{0.65}}$ $f_4(X) = (250) \cdot 2289 \exp(-39.75x_2 + 9.9x_3 + 2.74)$ $f_5(X) = 25 \cdot (1.39/(x_1 \cdot x_2) + 4940x_3 - 80)$	Subjected to $g_1(X) = 0.00139/(x_1 \cdot x_2) + 4.94x_3 - 0.08 \leq 1$ $g_2(X) = 0.000306/(x_1 \cdot x_2) + 1.082x_3 - 0.0986 < 1$ $g_3(X) = 12.307/(x_1 \cdot x_2) + 49408.24x_3 + 4051.02 \leq 50000$ $g_4(X) = 2.098/(x_1 \cdot x_2) + 8046.33x_3 - 696.71 \leq 16000$ $g_5(X) = 2.138/(x_1 \cdot x_2) + 7883.39x_3 - 705.04 \leq 10000$ $g_6(X) = 0.417 \cdot (x_1 \cdot x_2) + 1721.26x_3 - 136.54 \leq 2000$ $g_7(X) = 0.164/(x_1 \cdot x_2) + 631.13x_3 - 54.48 \leq 550$

Table 4.3 continued

Problem	Variable bounds	Objective functions
Three- functions	$0 \leq x_i \leq 1$, $i = 1, \dots, 12$	$f_1(x) = (1 + g(x)) \cos(0.5\pi x_1) \cos(0.5\pi x_2)$ $f_2(x) = (1 + g(x)) \cos(0.5\pi x_1) \sin(0.5\pi x_2)$ $f_3(x) = (1 + g(x)) \sin(0.5\pi x_1)$ $g_x(x) = \sum_{i=1}^{12} (x_i - 0.5)^2$

Table 4.4 Three-functions optimisation

4.3 An enhanced Bees Algorithm for solving Multi-Objective Optimisation Problems

4.3.1 Enhancements

The proposed Bees Algorithm is globally and locally enhanced for solving MOOPs with the enhancements described as follows.

4.3.1.1 Global enhancements

Two global enhancements are introduced here. The first global enhancement of the Bees Algorithm is the introduction of ‘dynamic procedure for site selection using domination sorting in the objective space’ instead of selecting a ‘static number of elected sites with the fittest sites in the search space’. The proposed selection method only accounts for non-dominated sites therefore it requires no effort to perform Pareto ranking which overcomes the main drawback of meta-heuristic techniques (Lee and El-Sharkawi 2008).

Also, according to (Pham and Ghanbarzadeh 2007)’s work, it needs extra efforts to find a static number of selected sites. The reason being that if there are insufficient numbers of selected sites (m), the candidate solutions from rank 1 which have been dominated once, are utilised until a sufficient number of selected sites necessary to conduct a neighbourhood search are found. The

proposed algorithm does not need to use this extra effort. The number of selected sites in the proposed algorithm is not static because it depends on the number of non-dominated sites.

The second global enhancement is the reduction in the parameters. While the basic Bees Algorithm uses 6 parameters (n , e , ne , m , nm , ngh), the proposed Bees Algorithm requires only 3 parameters to be set, namely:

- the number of scout bees n ,
- the number of recruited bees nm and
- patch size ngh .

In (Pham and Ghanbarzadeh 2007)'s work, there are 4 parameters (i.e., n , m , nm and ngh) and they did not use the e and ne parameters because elite sites are hard to identify especially in Multi-Objective space. Therefore, the proposed algorithm does not use them, and also due to introducing the 'dynamic procedure for site selection', the m parameter is no longer required to be set before the program runs.

4.3.1.2 Local enhancements

In this section, three different neighbourhood search methods are introduced:

- 1) basic neighbourhood search (basicNGH) method,

- 2) random neighbourhood search (randomNGH) method and
- 3) weighted sum neighbourhood search (wsNGH) method.

4.3.1.2.1 Basic neighbourhood search (basicNGH) method

When using the basic neighbourhood search method with the basic Bees Algorithm for solving single objective optimisation problems (SOOPs), a recruited bee is sent to a selected patch and its fitness evaluated. The recruited bee is then compared with the existing scout bee and the fittest one retains as the scout bee for this patch. This procedure is repeated until the last recruited bee is sent. The final retained bee will represent this flower patch to form a new population for the next iteration.

The proposed method adopts this principle, but it differs slightly. The difference being is that the selected bee will be replaced by the dominant recruited bee in the objective space instead of the fittest bee in the search space. However, the other principle of sending bees one by one is exactly the same as the basic Bees Algorithm, thus the proposed neighbourhood search method is called the ‘basic neighbourhood search (basicNGH) method’.

Figure 4.1 depicts its flowchart, and Figure 4.2 graphically illustrates the concept. $Z (\bullet)$ in Figure 4.2 is one of the dominant solutions after finishing the arbitrary iteration and represents the selected site in which to conduct a neighbourhood

search. Figure 4.2 (a) shows the basic information before a neighbourhood search starts. Z (●) is surrounded by a neighbourhood zone according to patch size (ngh) and the zone can be sub-divided into four areas (1 to 4). It is assumed that the number of recruited bees (nm) is five and their sending orders are explained in Figure 4.2 (a). The whole procedure is illustrated in Figure 4.2 (b to g). In the basicNGH method, a dominant recruited bee replaces the in situ bee until all the recruited bees (here five bees) have been sent to the site, which means that only a recruited bee in area 3 can replace the selected bee as in (b and d). Otherwise, the existing dominant bee is retained as in (c, e and f). The reason is that the values of f_1 and f_2 from a non-dominated bee (▲) are simultaneously lower than Z 's, which means that Z is dominated by ▲. If a new recruited bee replaces the selected bee as in (b), the neighbourhood zone is also redefined by ngh size as shown in (c) and the next recruited bee (■) will be sent into this new neighbourhood zone. However, if the next recruited bee is sent to areas 1 (★), 2 (■) or 4 (⊕), it cannot replace neither the existing dominant bee nor change its neighbourhood zone. Finally, after finishing the neighbourhood search, the third recruited bee (◆) becomes the new representative bee in this flower patch as in (g) to form a new population for the next iteration.

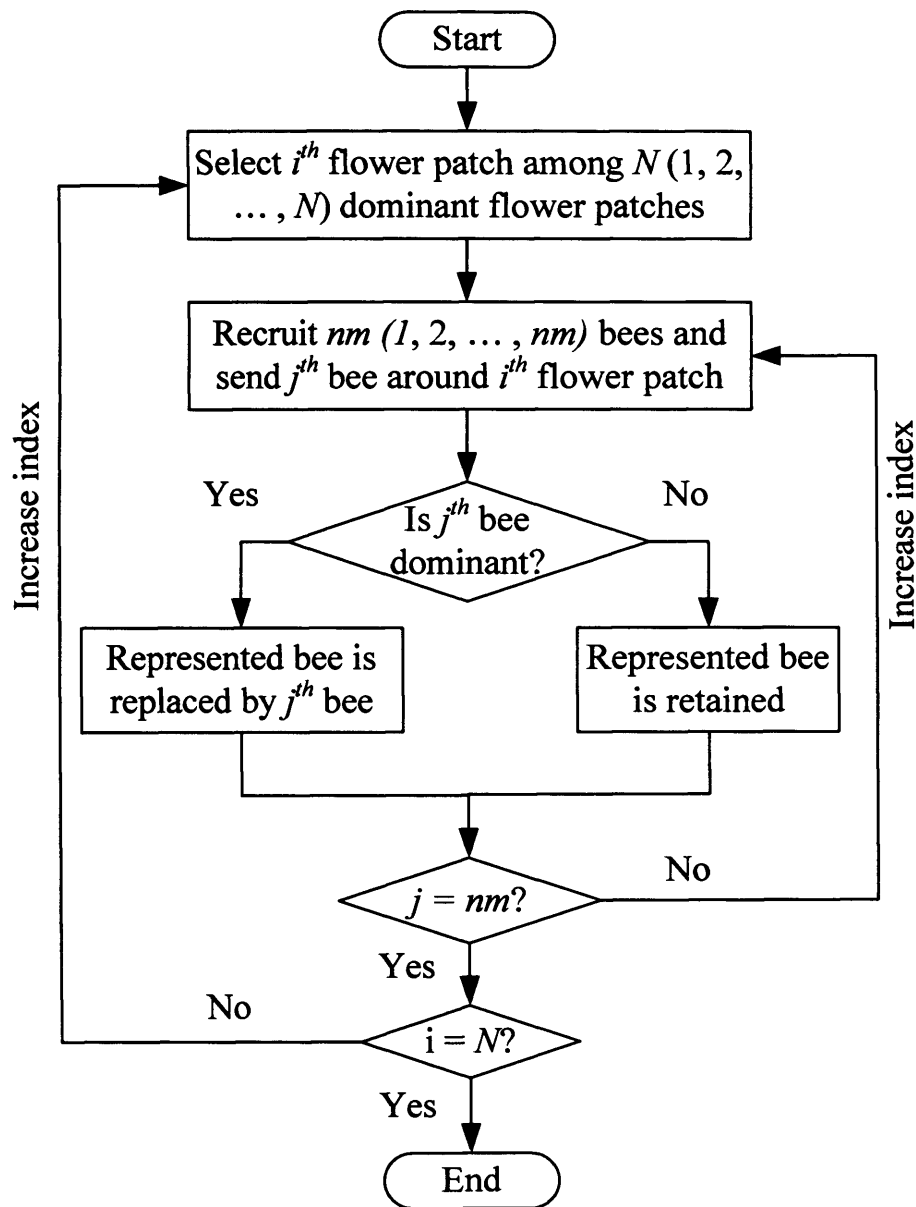


Figure 4.1 Flowchart for the basicNGH method

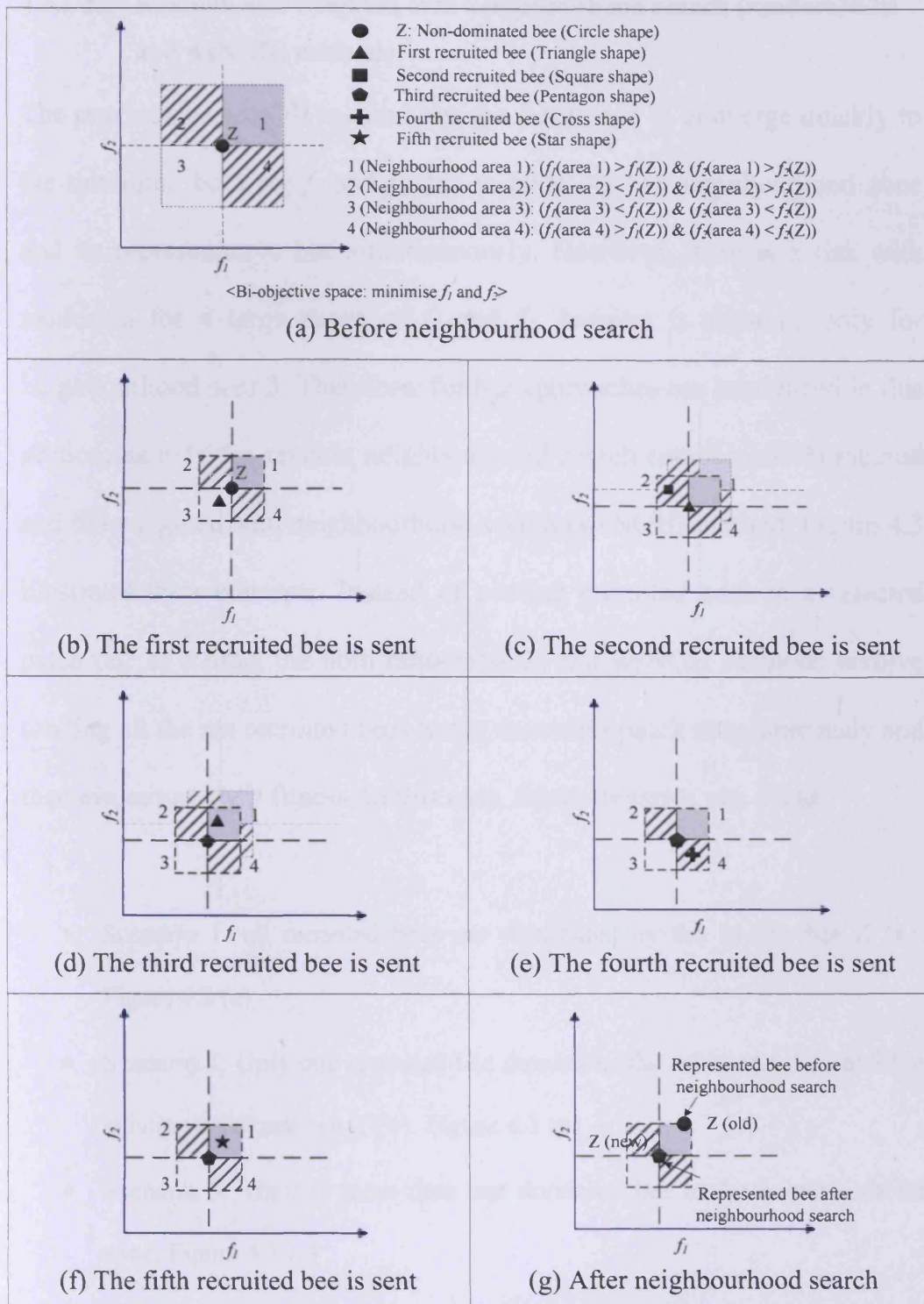


Figure 4.2 The concept of basicNGH method

4.3.1.2.2 Random and weighted sum neighbourhood search (randomNGH and wsNGH) methods

The previous basicNGH method has the advantage to converge quickly to the minimum both for f_1 and f_2 due to updating the neighbourhood zone and its representative bee simultaneously. However, there is a risk with exploring for a large range of f_1 and f_2 , because it accounts only for neighbourhood area 3. Therefore, further approaches are introduced in this section, namely the random neighbourhood search (randomNGH) method and the weighted sum neighbourhood search (wsNGH) method. Figure 4.3 illustrates their concepts. Instead of posting recruited bees to a selected patch one at a time, the both randomNGH and wsNGH methods involve sending all the nm recruited bees to the dominant patch simultaneously and then evaluating their fitness. In this case, three scenarios can occur.

- Scenario 1: all recruited bees are dominated by the in situ bee Z (●).
Figure 4.3 (a)
- Scenario 2: Only one recruited bee dominates the other recruited and the existing dominant bee Z (●). Figure 4.3 (c)
- Scenario 3: There is more than one dominant bee in the neighbourhood zone. Figure 4.3 (e)

Scenarios 1 and 2 are the same as those for the basicNGH method described previously. Scenario 3 represents a major advantage over the basicNGH method. In this scenario, where there is more than one dominant bee only one can be selected to become the representative in this flower patch as in Figure 4.3 (g; h). Two selection methods are introduced here:

- 1) the random selection method and
- 2) the weighted sum selection method.

If there are two or more dominant bees in the randomNGH method, one of them will be chosen at random to replace the existing scout bee. In the wsNGH method, where there are two or more dominant bees, a linear combination of the objective functions (Eq. 3.1) is used to decide which one of the dominant bees is to be selected. The latter method concentrates on a specific search area utilising a user-defined weight combination, therefore it quickly finds a better solution in this specific search area. Both of these methods have more chances to divert to various objective spaces when compared to the basicNGH method, because they consider more neighbourhood areas, e.g., areas 2 and 4. Thus, they should find more Pareto solutions through the large range. Figure 4.4 shows the flowchart for the both randomNGH and wsNGH methods.

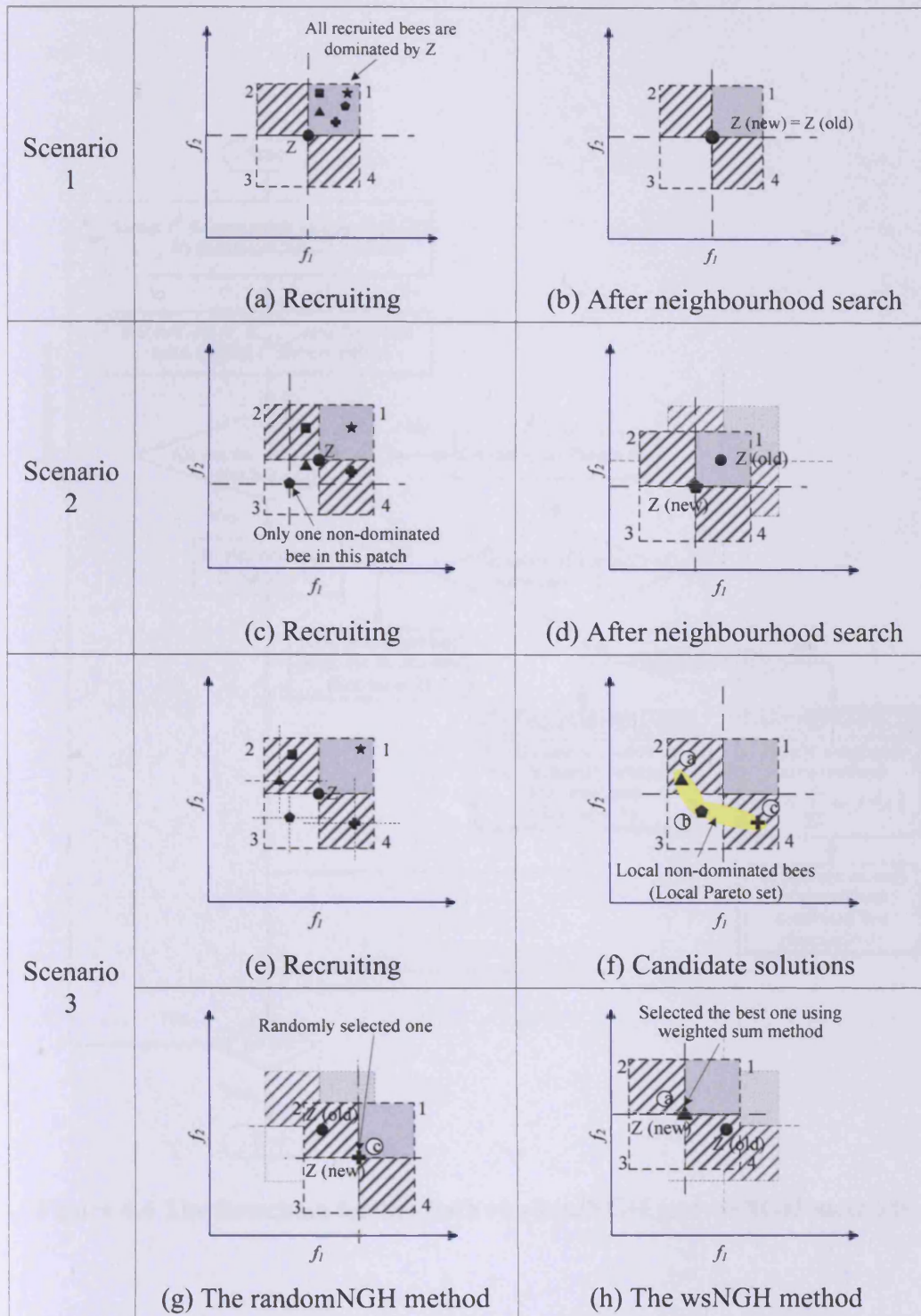


Figure 4.3 The concept for the both randomNGH and wsNGH methods

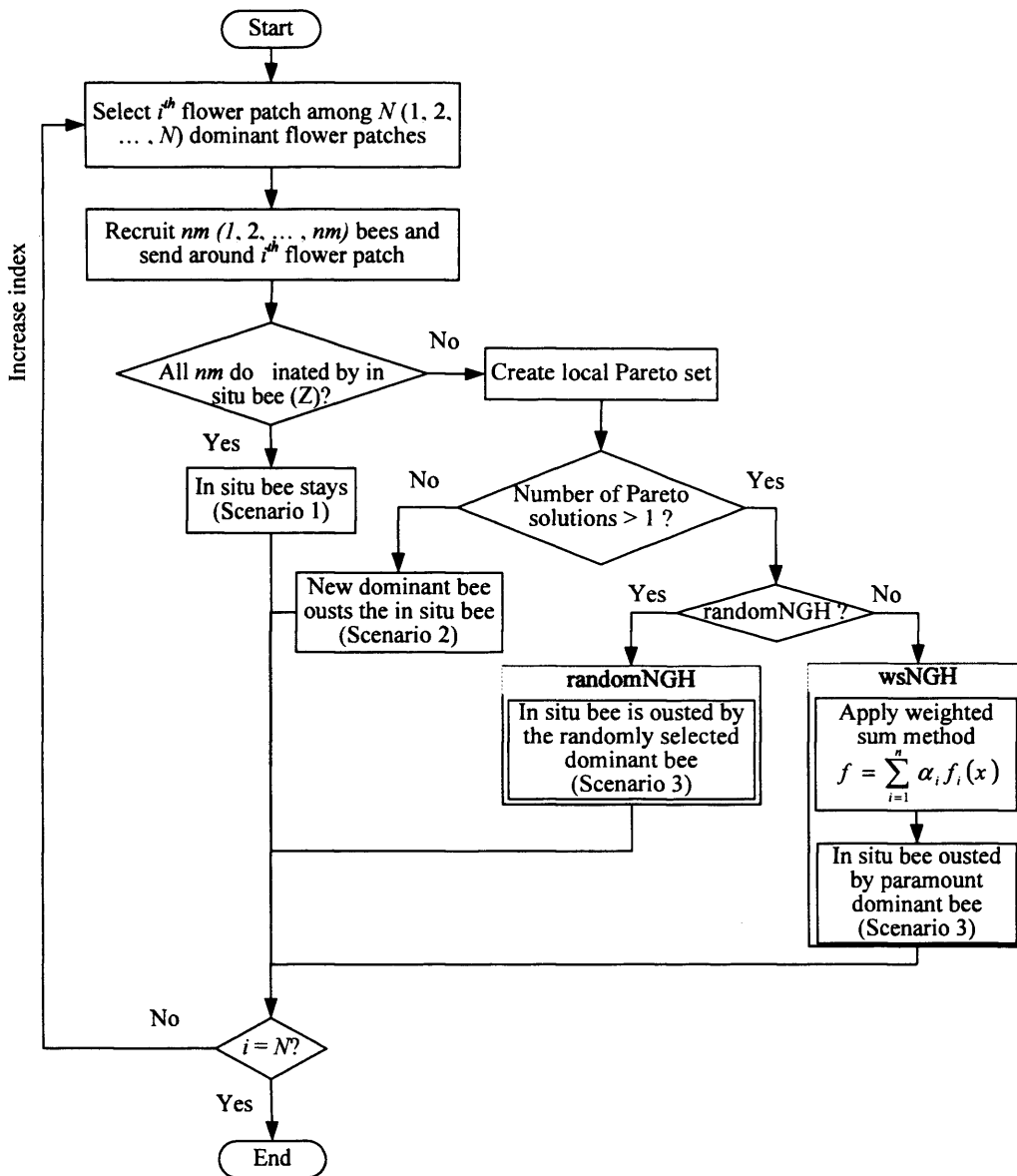


Figure 4.4 The flowchart for the both randomNGH and wsNGH methods

4.3.2 The proposed algorithms

The algorithms proposed in this chapter also take into account the complete food processing procedure of honey bees in the colony to generate the Pareto front. Two different types of Bees Algorithm are presented in this section: the Single-Cycled Bees Algorithm (SCBA) and the Multi-Cycled Bees Algorithm (MCBA).

4.3.2.1 The Single-Cycled Bees Algorithm (SCBA)

There are three different Single-Cycled Bees Algorithms regarding a neighbourhood search method, i.e., the basicNGH method, the randomNGH method and the wsNGH method. Figure 4.5 shows the SCBA flowchart. The main procedure is similar to the basic Bees Algorithm for solving SOOPs but the major differences use the newly developed neighbourhood search methods (box A Figure 4.5) and add an additional step (box B Figure 4.5) to create the Pareto front as a MO solver. The algorithm starts with n scout bees which are randomly sent to the search space. After fitness evaluation, those patches with dominant scout bees are designated “selected patches” for a further neighbourhood search to be conducted in them. With the basic Bees Algorithm, the number of selected patches ‘ m ’ and their locations are always static within a neighbourhood area for each iteration. However, the proposed algorithm is totally dynamic for both the number of selected patches and also their locations in each iteration. After this

neighbourhood search, only the one bee is elected by the basicNGH method, randomNGH or wsNGH methods to each of the selected patches to form the next bee population. The remaining bees in the population are randomly assigned in the search space for new potential solutions. The remaining number of bees is also dynamic as their total fluctuates with the number of dominant bees. These steps are repeated until a stopping criterion is met. The Pareto optimal set is created from all the non-dominated bees in each iteration. The proposed algorithm conducts only one cycle of the Bees Algorithm, therefore it is named the Single-Cycled Bees Algorithm (SCBA). Additionally, depending on the neighbourhood search method used, it is identified as the SCBA-basicNGH, the SCBA-randomNGH and the SCBA-wsNGH respectively.

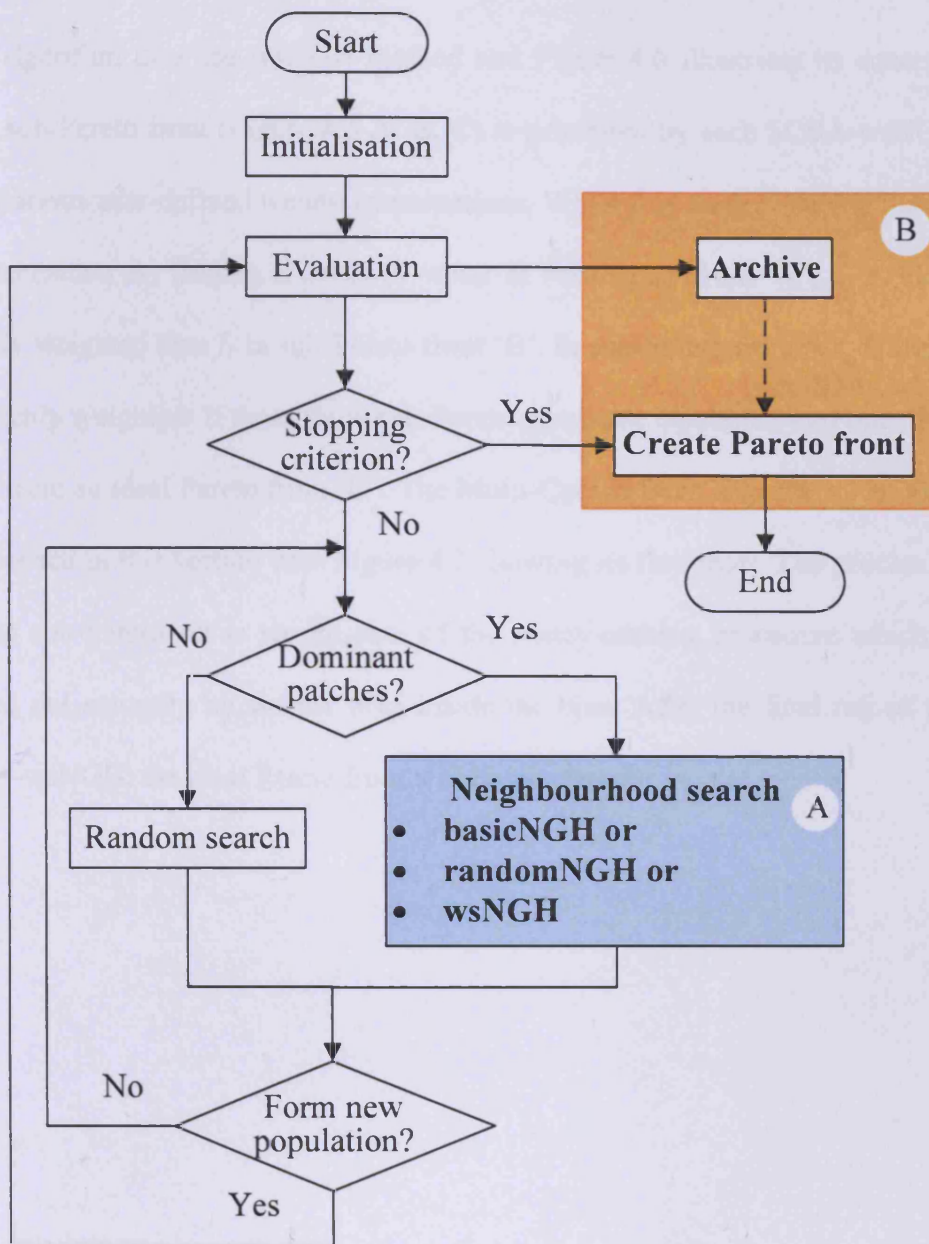


Figure 4.5 The flowchart of the Single-Cycled Bees Algorithm (SCBA)

4.3.2.2 The Multi-Cycled Bees Algorithm (MCBA)

This algorithm uses the wsNGH method and Figure 4.6 illustrates its concept. Each sub-Pareto front (Figure 4.6 A, B, C) is generated by each SCBA-wsNGH with various user-defined weight combinations. While f_1 is more weighted than f_2 to concentrate on finding a better f_1 value in sub-Pareto front 'A', f_2 is more heavily weighted than f_1 in sub-Pareto front 'B'. In sub-Pareto front 'C', f_1 and f_2 are evenly weighted. If these three sub-Pareto fronts are combined into one, they will create an ideal Pareto front 'D'. The Multi-Cycled Bees Algorithm (MCBA) is proposed in this section with Figure 4.7 showing its flowchart. The process of storing sub-Pareto set is reminiscent of the honey-making procedure which is carried out naturally by worker bees inside the hive. After the final run of the SCBA-wsNGH, the ideal Pareto front will be generated.

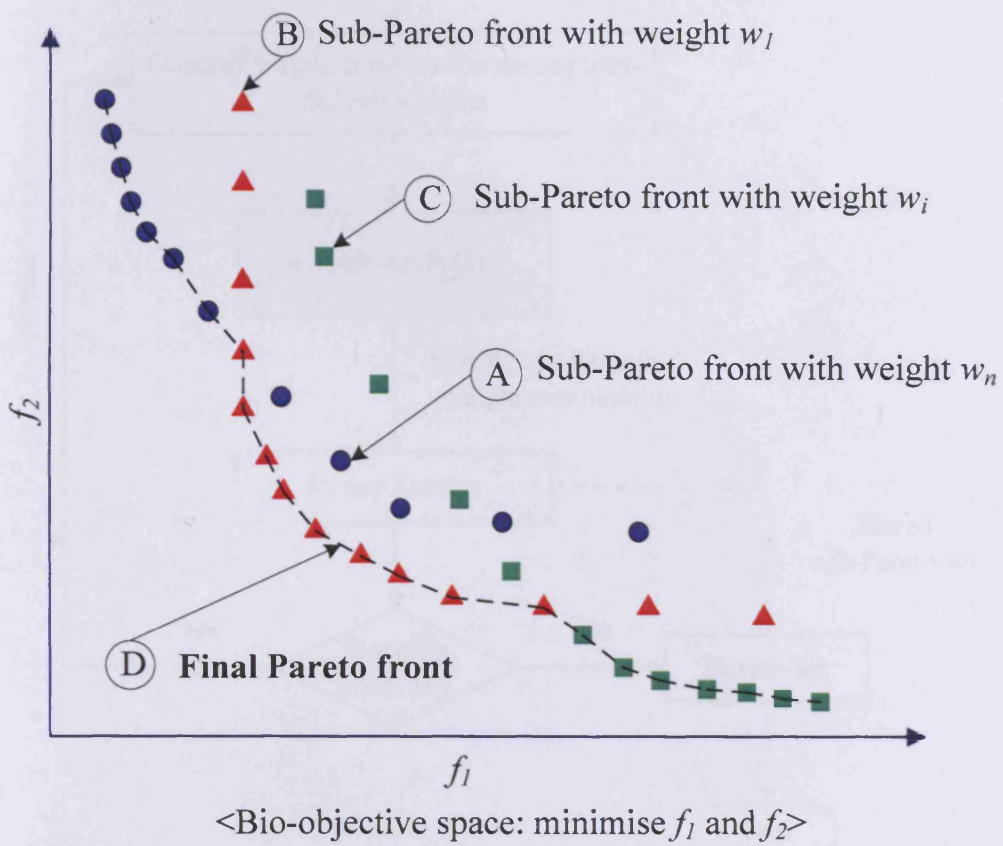


Figure 4.6 The concept of the Multi-Cycled Bees Algorithm (MCBA)

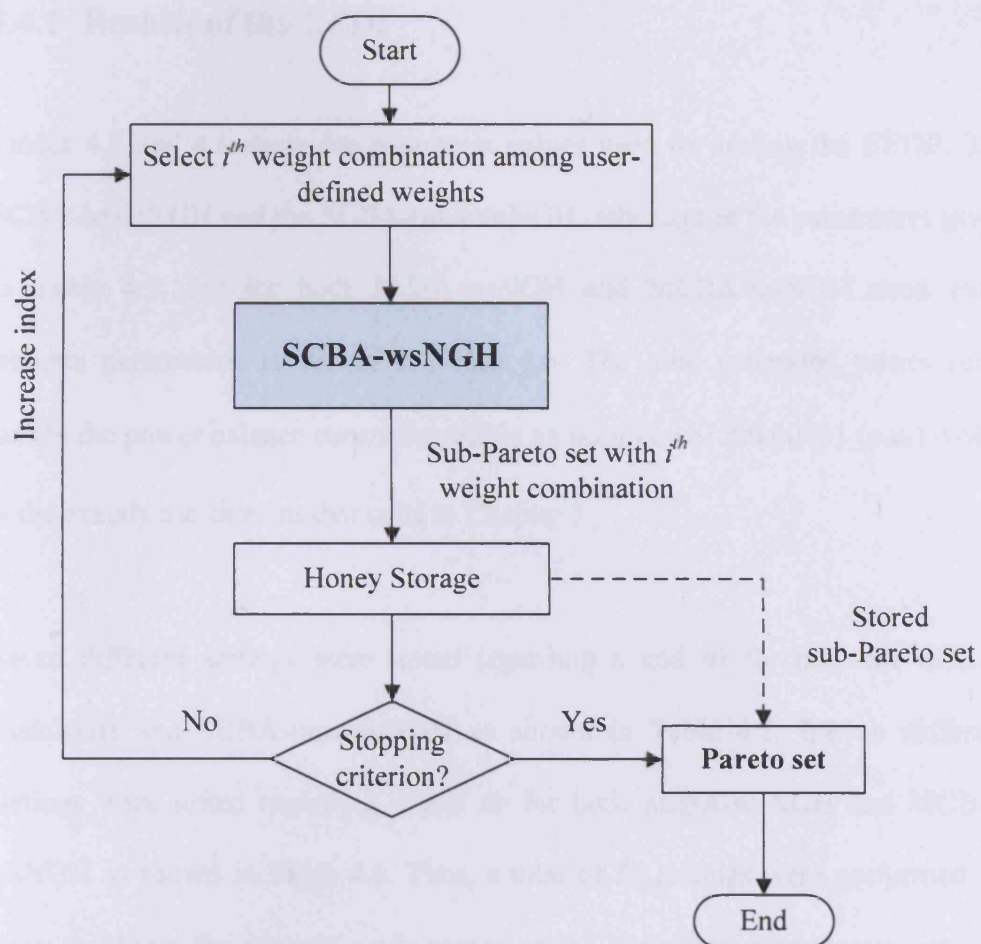


Figure 4.7 The flowchart of the Multi-Cycled Bees Algorithm (MCBA)

4.4 Results

4.4.1 Results of the EEDP

Tables 4.5 and 4.6 show the parameter values used for testing the EEDP. The SCBA-basicNGH and the SCBA-randomNGH only require the parameters given in Table 4.5, but for both SCBA-wsNGH and MCBA-wsNGH need extra weights parameters as shown in Table 4.6. The total generated power must satisfy the power balance constraint within an accuracy of ± 0.00001 (p.u.) which is the exactly the same to that used in Chapter 3.

Seven different settings were tested regarding n and itr for both the SCBA-basicNGH and SCBA-randomNGH as shown in Table 4.5. Eleven different settings were tested regarding n and itr for both SCBA-wsNGH and MCBA-wsNGH as shown in Table 4.6. Thus, a total of 58 settings were performed. In order to obtain the average performance of the algorithm, 20 independent runs were carried out in each setting. At the end of each run, the performance of the various algorithm configurations was evaluated taking two extreme solutions for minimum fuel cost and NOx emissions from the Pareto set. Performance was also evaluated taking into consideration the range and quality of the Pareto fronts plus the number of solutions and evaluations.

SCBA-basicNGH and SCBA-randomNGH				
setting	Parameters			
	<i>n</i>	<i>nm</i>	<i>ngh</i>	<i>itr</i>
1	100	10	0.01	100
2				150
3				200
4				300
5				400
6	200			200
7				400

n: number of scout bees

ngh: patch size

nm: number of bees allocated to dominant patches

itr: number of iterations

Table 4.5 Parameter settings for the SCBA-basicNGH and the SCBA-randomNGH

SCBA-wsNGH and MCBA-wsNGH						
setting	Parameters					
	<i>n</i>	<i>nm</i>	<i>ngh</i>	<i>itr</i>	<i>w₁</i>	<i>w₂</i>
1	25	10	0.01	25	1	1; 1,000; 100,000
2				50		
3	30			30		
4				40		
5	50			25		
6				30		
7				50		
8				100		
9	80			25		
10	100			25		
11				50		

w₁: weight for fuel cost
w₂: weight for NOx emissions

Table 4.6 Parameter settings for the SCBA-wsNGH and the MCBA-wsNGH

4.4.1.1 Comparison of two extreme Pareto solutions

To evaluate the performance of the various algorithm configurations, two extreme solutions for minimum fuel cost and NOx emissions from the Pareto set are compared. These two Pareto solutions from 20 independent runs were compared with the other existing Multi-Objective optimisation algorithms such as MOPSO (Abido 2007, 2009), NSGA-II (Ah King et al. 2005) and SPEA (Abido 2003a). They were compared with each other regarding the neighbourhood search methods proposed in this chapter.

4.4.1.1.1 Comparison of Multi-Objective Optimisation Algorithms

The Tables from 4.7 to 4.14 show the comparison of two extreme Pareto solutions. Most of the data sets for f_c , $Corr. f_e$ and $Corr. f_c$ from 20 independent runs do not violate the assumption of normality according to analysis by the SPSS program, therefore, a *One-Sample t-test* was conducted to compare the mean value with MOPSO, NSGA-II and SPEA. The symbols and values used in Table 4.7 to Table 4.14 are the same to those as shown in Table 3.5, except four additional symbols:

- 1) n/a (Tables 4.8, 4.9, 4.12 and 4.13) indicates that the data sets are non applicable to compare mean values using the *One-Sample t-test* because most of the data sets of f_e from 20 independent runs violate the assumption of normality. The average value itself is compared to the other algorithms instead of using the *One-Sample t-test*.

- 2) n/a^+ : the average value is higher (worse) than the other algorithms
- 3) n/a^* : the average value is the exact same to the other algorithms
- 4) n/a^- : the average value is lower (better) than the other algorithms

Table 4.7 shows the comparison of two extreme Pareto solutions between MOPSO (Abido 2007, 2009), the SCBA-basicNGH and the SCBA-randomNGH. According to the results, the average value of the best cost (f_c) from setting7 in the SCBA-randomNGH is not significantly different to MOPSO, but the others are significantly inferior to MOPSO. The average values of the best emission (f_e) from setting5 and setting7 in the SCBA-randomNGH is equal to MOPSO.

Table 4.8 shows the comparison between the NSGA- II (Ah King et al. 2005), the SCBA-basicNGH and the SCBA-randomNGH. The average value of the best cost (f_c) from setting7 in the SCBA-basicNGH correspond to the NSGA- II , however, the other settings are significantly worse than the NSGA- II . The best emission (f_e) from all settings in the SCBA-basicNGH is also worse than for the NSGA- II . The average value of the best cost (f_c) from settings 5, 6 and 7 in the SCBA-randomNGH is significantly better than those of the NSGA- II . The best emission values (f_e) from settings 5 and 7 in the SCBA-randomNGH are exactly the same as the NSGA- II .

Algorithm	setting	The best cost		The best emissions	
		f_c	$Corr. f_e$	$Corr. f_c$	f_e
SCBA- basicNGH	1	600.28222	<u>0.21892</u>	<u>633.60390</u>	0.19434
	2	600.26478	<u>0.21894</u>	<u>633.71762</u>	0.19433
	3	600.28184	<u>0.21873</u>	<u>634.16988</u>	0.19431
	4	600.21972	<u>0.21949</u>	<u>634.44370</u>	0.19428
	5	600.23763	<u>0.21907</u>	<u>634.86058</u>	0.19427
	6	600.22327	<u>0.21941</u>	<u>634.34729</u>	0.19429
	7	600.19393	<u>0.21989</u>	<u>635.08662</u>	0.19426
SCBA- randomNGH	1	600.29912	<u>0.22033</u>	<u>635.40831</u>	0.19430
	2	600.16638	<u>0.22100</u>	<u>636.63751</u>	0.19423
	3	600.14706	<u>0.22121</u>	<u>637.34353</u>	0.19422
	4	600.14455	<u>0.22104</u>	<u>637.49389</u>	0.19421
	5	600.12784	<u>0.22152</u>	<u>637.58761</u>	<u>0.19420</u>
	6	600.14084	<u>0.22109</u>	<u>637.03869</u>	0.19421
	7	600.12083	<u>0.22179</u>	637.73237	<u>0.19420</u>

Table 4.7 Comparison between the SCBA-basicNGH, the SCBA-randomNGH and MOPSO

Algorithm	setting	The best cost		The best emissions	
		f_c	$Corr. f_e$	$Corr. f_c$	f_e
SCBA- basicNGH	1	×	√	√	×
	2	×	√	√	×
	3	×	√	√	×
	4	×	√	√	×
	5	×	√	√	×
	6	×	√	√	n/a ⁺
	7	-	√	√	n/a ⁺
SCBA- randomNGH	1	×	√	√	n/a ⁺
	2	-	√	√	n/a ⁺
	3	-	√	√	n/a ⁺
	4	-	√	√	n/a ⁺
	5	√	√	√	n/a [*]
	6	√	√	√	n/a ⁺
	7	√	-	√	n/a [*]

Table 4.8 Comparison between the SCBA-basicNGH, the SCBA-randomNGH and NSGA- II

Table 4.9 shows the comparison between the SPEA (Abido 2003a), the SCBA-basicNGH and the SCBA-randomNGH. The average value of the best cost (f_c) from all settings in the SCBA-basicNGH is significantly inferior to the SPEA and the best emissions (f_e) from all settings is also worse than SPEA. However, the average value of the best cost (f_c) from settings 2, 3, 4 and 6 in the SCBA-randomNGH have no significant difference to the SPEA and those from settings 5 and 7 are significantly better than the SPEA. The best emissions (f_e) from settings 5 and 7 are exactly the same as the SPEA.

Tables 4.10 and 4.11 show the comparison of two extreme Pareto solutions between MOPSO (Abido 2007, 2009), the SCBA-wsNGH and the MSBA-wsNGH.

- SCBA-wsNGH ($w_2=1$)
 - the best cost (f_c): the average value from setting5 is not significantly different to MOPSO, but settings 2, 4, 6, 7, 8, 9 and 11, are significantly better than MOPSO. The average value of the best cost (f_c) from setting10 is lower (i.e., better) than for MOPSO
 - the best emission (f_e): the average value from all settings is higher (i.e., worse) than for MOPSO
- SCBA-wsNGH ($w_2=1,000$)
 - The average values from all settings for both the best cost (f_c) and the best emissions (f_e) are inferior to those for MOPSO

Algorithm	setting	The best cost		The best emissions	
		f_c	$Corr. f_e$	$Corr. f_c$	f_e
SCBA- basicNGH	1	×	√	√	×
	2	×	√	√	×
	3	×	√	√	×
	4	×	√	√	×
	5	×	√	√	×
	6	×	√	√	n/a ⁺
	7	×	√	√	n/a ⁺
SCBA- randomNGH	1	×	√	√	n/a ⁺
	2	-	√	√	n/a ⁺
	3	-	-	√	n/a ⁺
	4	-	√	√	n/a ⁺
	5	√	-	√	n/a [*]
	6	-	√	√	n/a ⁺
	7	√	×	√	n/a [*]

Table 4.9 Comparison between the SCBA-basicNGH, the SCBA-randomNGH and SPEA

Algorithm	setting	The best cost		The best emissions	
		f_c	$Corr. f_e$	$Corr. f_c$	f_e
SCBA- wsNGH ($w_2=1$)	1	600.16340	0.22197	637.19952	0.19527
	2	<u>600.11231</u>	0.22216	636.15354	0.19494
	3	600.12046	0.22192	637.08929	0.19497
	4	<u>600.11312</u>	0.22212	637.10332	0.19499
	5	600.12394	0.22219	636.66324	0.19489
	6	<u>600.11519</u>	0.22215	637.72327	0.19475
	7	<u>600.11247</u>	0.22211	636.47811	0.19467
	8	<u>600.11117</u>	0.22217	636.58825	0.19458
	9	<u>600.11586</u>	0.22222	638.20259	0.19474
	10	<u>600.11576</u>	0.22209	636.34602	0.19472
	11	<u>600.11215</u>	0.22217	637.07723	0.19460
SCBA- wsNGH ($w_2=$ 1,000)	1	601.04174	<u>0.22014</u>	<u>635.19579</u>	0.19508
	2	601.01857	0.22018	<u>635.14119</u>	0.19485
	3	601.14447	<u>0.21949</u>	636.42977	0.19474
	4	600.99740	<u>0.21958</u>	<u>634.83457</u>	0.19501
	5	601.01486	<u>0.21930</u>	636.05503	0.19484
	6	600.82951	<u>0.21930</u>	<u>635.91738</u>	0.19471
	7	600.67160	<u>0.22049</u>	636.67998	0.19464
	8	600.61981	0.22084	636.52948	0.19448
	9	600.78007	0.22194	636.97182	0.19477
	10	600.67790	<u>0.21988</u>	635.78992	0.19469
	11	600.44556	0.22122	636.37855	0.19455

Table 4.10 Comparison between the SCBA-wsNGH and MOPSO

Algorithm	setting	The best cost		The best emissions	
		f_c	$Corr. f_e$	$Corr. f_c$	f_e
SCBA- wsNGH ($w_2=$ 100,000)	1	601.55247	0.22148	637.54095	0.19423
	2	601.12214	0.22074	637.93431	<u>0.19420</u>
	3	601.06146	0.22077	637.48582	0.19421
	4	601.14147	0.22207	637.72784	<u>0.19420</u>
	5	600.87293	0.22101	637.29410	0.19422
	6	601.07212	0.22114	637.74397	<u>0.19420</u>
	7	600.67790	0.22107	637.92438	<u>0.19420</u>
	8	600.56921	0.22159	638.05104	<u>0.19420</u>
	9	600.85501	0.22115	637.38361	0.19421
	10	600.74773	0.22115	637.72211	<u>0.19420</u>
	11	600.60059	0.22167	638.00395	<u>0.19420</u>
MCBA- wsNGH	1	600.16340	0.22197	637.54095	0.19423
	2	<u>600.11231</u>	0.22216	637.93431	<u>0.19420</u>
	3	600.12046	0.22192	637.48582	0.19421
	4	<u>600.11312</u>	0.22212	637.72784	<u>0.19420</u>
	5	600.12394	0.22219	637.29410	0.19422
	6	<u>600.11519</u>	0.22215	637.74397	<u>0.19420</u>
	7	<u>600.11247</u>	0.22211	637.92438	<u>0.19420</u>
	8	<u>600.11117</u>	0.22217	638.05104	<u>0.19420</u>
	9	<u>600.11586</u>	0.22222	637.38361	0.19421
	10	<u>600.11576</u>	0.22209	637.72211	<u>0.19420</u>
	11	<u>600.11215</u>	0.22217	638.00395	<u>0.19420</u>

Table 4.11 Comparison between the SCBA-wsNGH, the MSBA-wsNGH and MOPSO

- SCBA-wsNGH ($w_2=100,000$)
 - the best cost (f_c): the average value from all settings is worse than for MOPSO
 - the best emissions (f_e): the average values from settings 2, 4, 6, 7, 8, 10 and 11 are exactly the same as those for MOPSO
- MCBA-wsNGH ($w_2=1; 1,000; 100,000$)
 - the best cost (f_c): the average value from setting5 does not differ significantly to MOPSO, but settings 2, 4, 6, 7, 8, 9 and 11 are significantly better than for MOPSO. The average value from setting10 is lower (i.e., better) than MOPSO
 - the best emissions (f_e): the average values from settings 2, 4, 6, 7, 8, 10 and 11 are exactly the same as those for MOPSO

Tables 4.12 and 4.13 show the comparison of two extreme Pareto solutions between the NSGA-II (Ah King et al. 2005), SPEA (Abido 2003a), the SCBA-wsNGH and the MSBA-wsNGH. The results of SPEA are identical to those for the NSGA-II.

- SCBA-wsNGH ($w_2=1$)
 - the best cost (f_c): the average value from setting1 is not significantly different to the NSGA-II, but settings 2, 4, 5, 6, 7, 8, 9 and 11 are

significantly better than for the NSGA- II . The average value from settings 3 and 10 is lower (i.e., better) than NSGA- II

- the best emissions (f_e): the average values from all settings are statistically worse than for the NSGA- II
- SCBA-wsNGH ($w_2=1,000$)
 - The average values from all settings for both the best cost (f_c) and the best emissions (f_e) are worse than those for the NSGA- II
- SCBA-wsNGH ($w_2=100,000$)
 - the best cost (f_c): the average values from all settings are worse than for the NSGA- II
 - the best emissions (f_e): the average values from settings 2, 4, 6, 7, 8, 10 and 11 are exactly the same as those for the NSGA- II
- MCBA-wsNGH ($w_2=1; 1,000; 100,000$)
 - the best cost (f_c): the average value from setting1 has no significant difference to those for the NSGA- II , but settings 2, 4, 5, 6, 7, 8, 9 and 11 are significantly better than for the NSGA- II . The average values from settings 3 and 10 are lower (i.e., better) than for the NSGA- II
 - the best emissions (f_e): the average values from settings 2, 4, 6, 7, 8, 10 and 11 are exactly the same as for the NSGA- II

Algorithm	setting	The best cost		The best emissions	
		f_c	$Corr. f_e$	$Corr. f_c$	f_e
SCBA- wsNGH ($w_2=1$)	1	-	-	-	×
	2	√	×	√	×
	3	n/a ⁻	-	-	×
	4	√	×	-	×
	5	√	×	-	×
	6	√	×	-	×
	7	√	×	√	×
	8	√	×	√	×
	9	√	×	-	×
	10	n/a ⁻	×	√	×
	11	√	×	√	×
SCBA- wsNGH ($w_2=1,000$)	1	×	√	√	×
	2	×	√	√	×
	3	×	√	√	×
	4	×	√	√	×
	5	×	√	-	×
	6	×	√	√	×
	7	×	√	-	×
	8	×	√	√	×
	9	×	-	-	×
	10	×	√	√	×
	11	×	-	√	×

Table 4.12 Comparison between NSGA-II, SPEA and the SCBA-wsNGH

Algorithm	setting	The best cost		The best emissions	
		f_c	$Corr. f_e$	$Corr. f_c$	f_e
SCBA- wsNGH ($w_2=100,000$)	1	×	-	√	n/a^+
	2	×	√	√	n/a^*
	3	×	-	n/a^-	n/a^+
	4	×	-	√	n/a^*
	5	×	-	√	n/a^+
	6	×	-	√	n/a^*
	7	×	-	√	n/a^*
	8	×	-	√	n/a^*
	9	×	-	√	n/a^+
	10	×	-	√	n/a^*
	11	×	-	√	n/a^*
MCBA- wsNGH	1	-	-	√	n/a^+
	2	√	×	√	n/a^*
	3	n/a^-	-	n/a^-	n/a^+
	4	√	×	√	n/a^*
	5	√	×	√	n/a^+
	6	√	×	√	n/a^*
	7	√	×	√	n/a^*
	8	√	×	√	n/a^*
	9	√	×	√	n/a^+
	10	n/a^-	×	√	n/a^*
	11	√	×	√	n/a^*

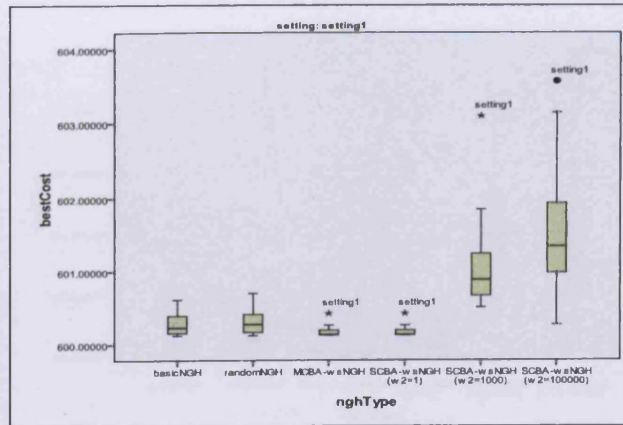
Table 4.13 Comparison between NSGA-II, SPEA, the SCBA-wsNGH and the MSBA-wsNGH

4.4.1.1.2 Effect of neighbourhood search methods

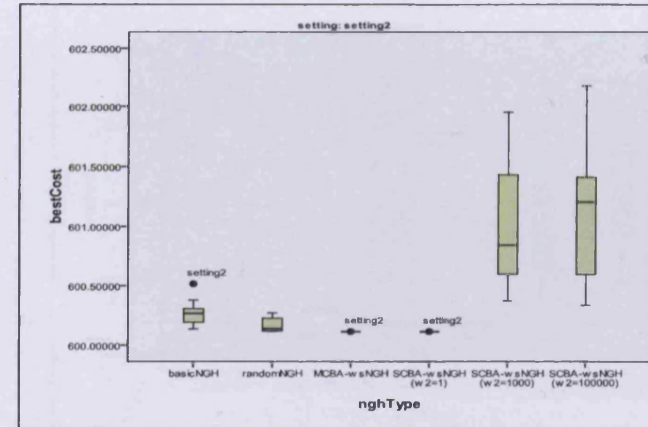
In order to evaluate the performance of the proposed neighbourhood search methods such as the SCBA-basicNGH, the SCBA-randomNGH, the SCBA-wsNGH and the MCBA-wsNGH, seven settings from each neighbourhood search method were taken. Although all data sets, except for f_e , do not violate the assumption of normality, they do violate the assumption of homogeneity of variances. Therefore the *Kruskal-Wallis* test was carried out to compare the median values regarding neighbourhood search methods. The results from the *Kruskal-Wallis* test show, there is a significant difference in all data sets. The box plot of the best cost and the best emissions for all settings regarding neighbourhood search methods was compiled as shown in Figures 4.8 and 4.9.

For the best cost (f_c), the SCBA-randomNGH for setting5 only (Figure 4.8 (e)) has no significant difference to both the MCBA-wsNGH and the SCBA-wsNGH ($w_2=1$). Other than this, both the MCBA-wsNGH and the SCBA-wsNGH ($w_2=1$) outperform the SCBA-basicNGH, the SCBA-randomNGH, the SCBA-wsNGH ($w_2=1,000$) and the SCBA-wsNGH ($w_2=100,000$) in all settings.

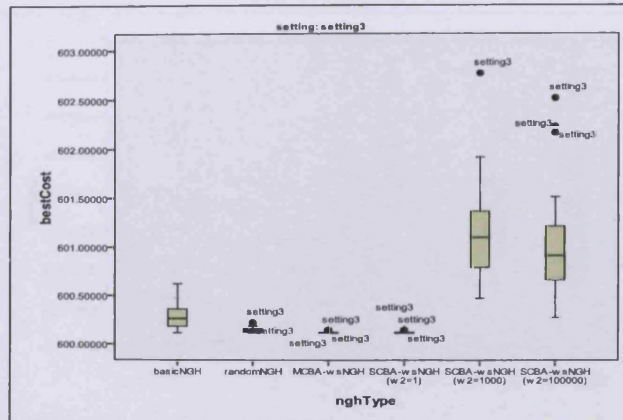
To determine the best emission value (f_e), the SCBA-randomNGH in settings 3, 4, 6 and 7 (Figure 4.9 c, d, f and g) have no significant differences to both, the MCBA-wsNGH and the SCBA-wsNGH ($w_2=100,000$). However, there are significant differences between the SCBA-randomNGH, the MCBA-wsNGH and



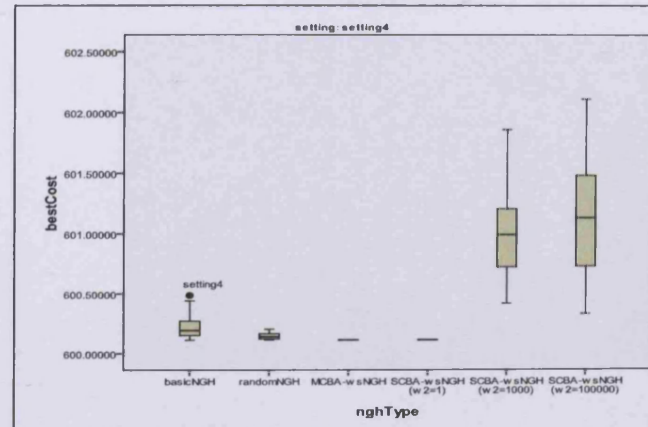
(a) setting1



(b) setting2

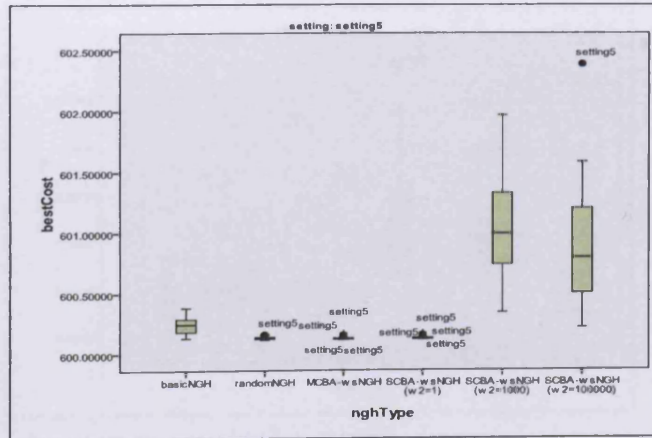


(c) setting3

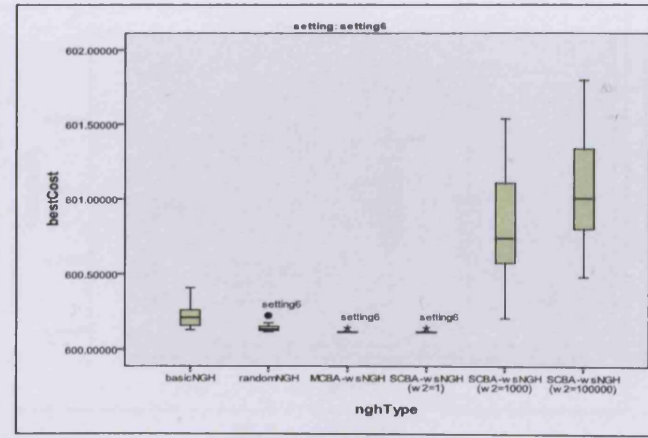


(d) setting4

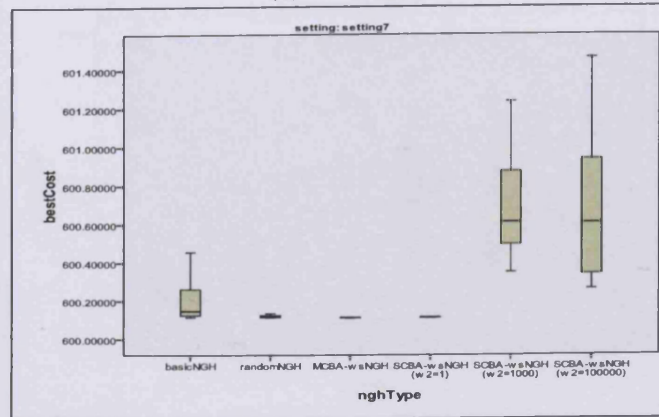
Figure 4.8 Box plot of the best cost (f_c) regarding neighbourhood search methods



(e) setting5

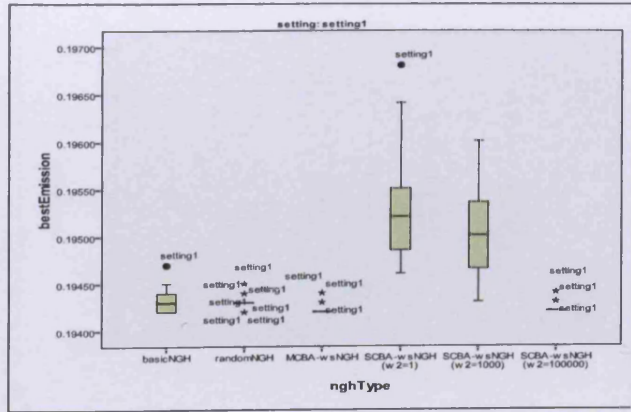


(f) setting6

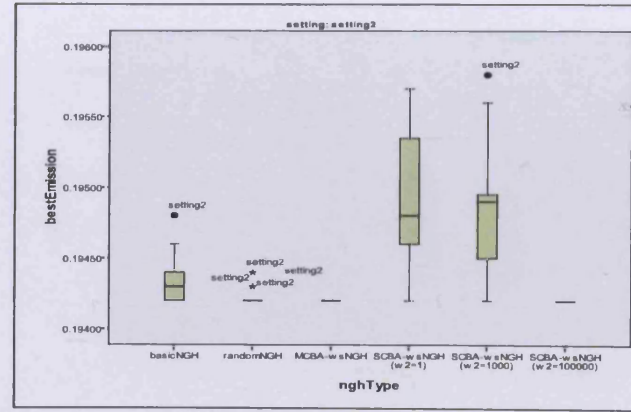


(g) setting7

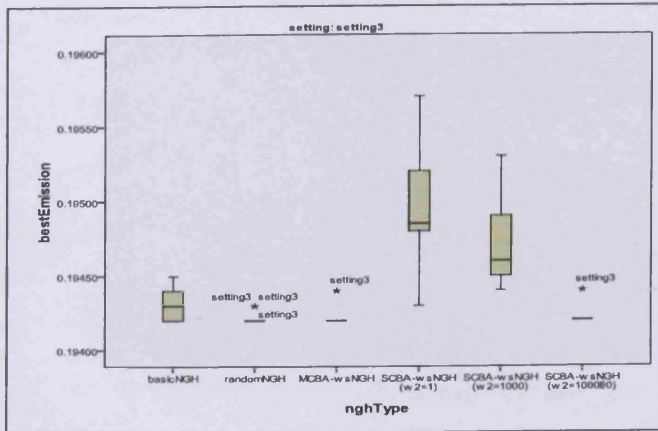
Figure 4.8 continued



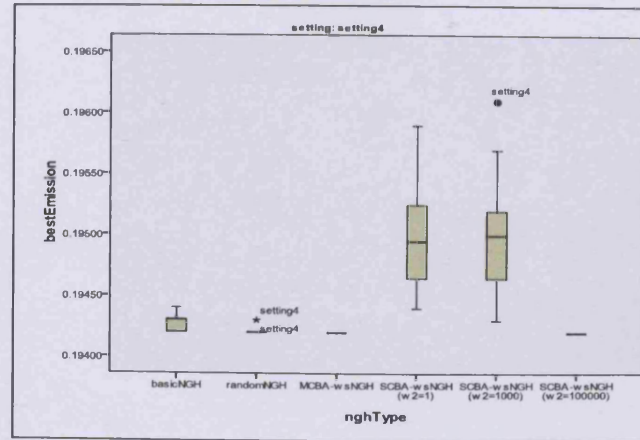
(a) setting1



(b) setting2

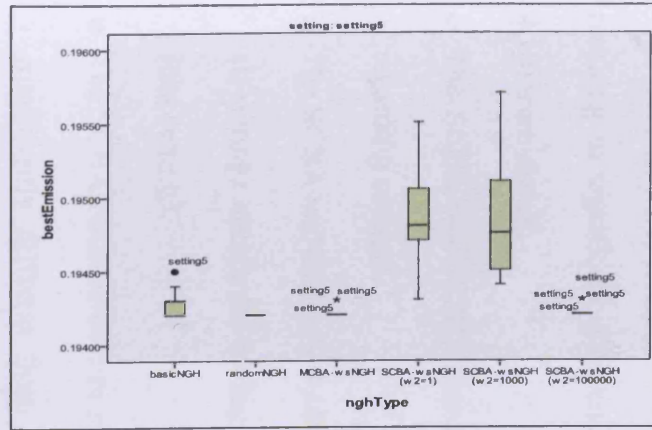


(c) setting3

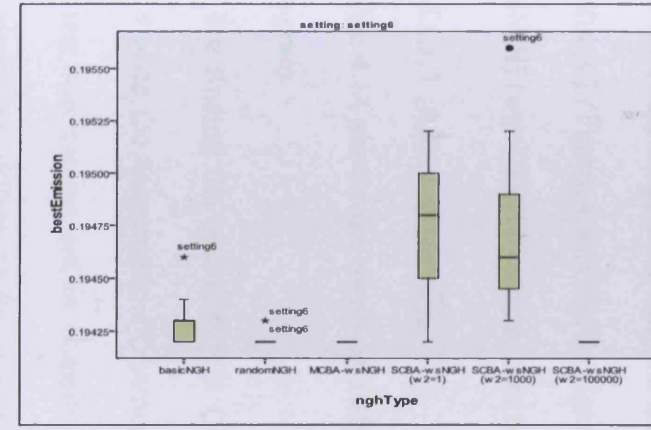


(d) setting4

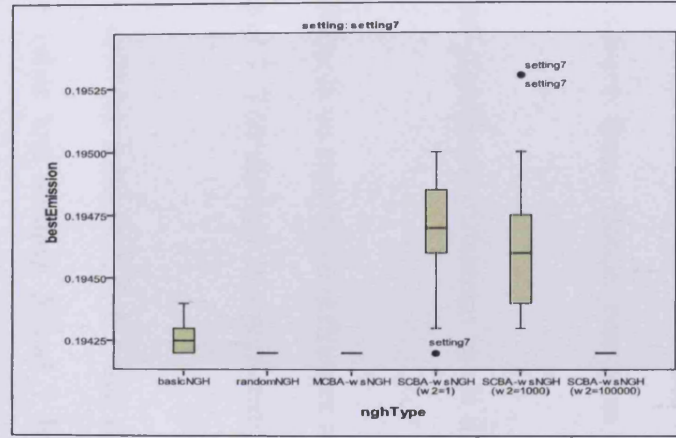
Figure 4.9 Box plot of the best emissions (f_e) regarding neighbourhood search methods



(e) setting5



(f) setting6



(g) setting7

Figure 4.9 continued

the SCBA-wsNGH ($w_2=100,000$). In setting5, Figure 4.9 (e), the SCBA-randomNGH outperforms all the other neighbourhood search methods. For both settings 1 (Figure 4.9 a) and 2 (Figure 4.9 b), the MCBA-wsNGH and the SCBA-wsNGH ($w_2=100,000$) outperform the other methods.

4.4.1.1.3 Effect of settings

Table 4.14 shows the result of statistic analyses regarding settings by the *SPSS* program.

- For finding the best cost (f_c): Only the SCBA-wsNGH ($w_2=1,000$) does not violate the assumption of homogeneity of variances, as a result an *ANOVA* test was carried out to compare the mean value. It can be seen there is a significant difference for at least one setting. Therefore a *Bonferroni* test was carried out. *Kruskal-Wallis* tests were conducted on all the other methods resulting in significant differences between them. Hence, box plots (Figure 4.10) were drawn:
 - The SCBA-basicNGH: There is no significant difference in all data sets regarding settings.
 - The SCBA-wsNGH ($w_2=1,000$): There is no significant difference amongst all settings except for settings 3 and 7. The setting7 is significantly better than setting3.
 - The SCBA-randomNGH in Figure 4.10 (a): settings 2, 3, 4, 5 and 6 are not significantly different from each other and settings 5 and 7 have no

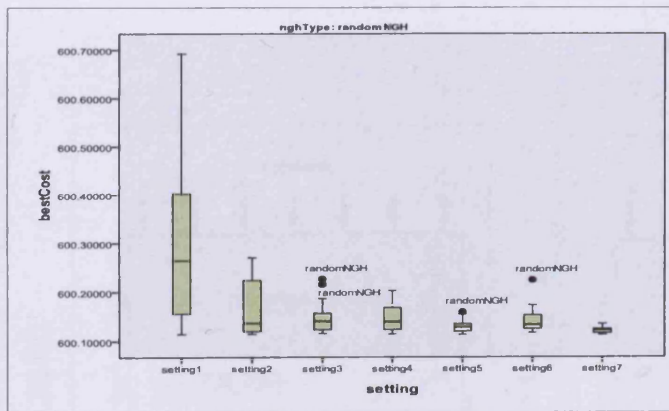
- significant differences between each other. Settings 2 to 7 have significant differences, with setting7 outperforms the others.
- The MCBA-wsNGH and the SCBA-wsNGH ($w_2=1$) Figure 4.10 (b and c): settings 2, 4 and 7 have no significant and they outperform the remaining settings.
 - The SCBA-wsNGH ($w_2=100,000$) Figure 4.10 (d): settings 5 and 7 have no significant difference and they outperform the remaining settings.
 - For finding the best emissions (f_2): All neighbourhood search methods violate the assumption of homogeneity of variances therefore they were tested by *Kruskal-Wallis*. There are significant differences hence box plots Figure 4.11 were drawn:
 - The SCBA-basicNGH: There is no significant difference in all data sets regarding settings.
 - The SCBA-randomNGH for settings 2, 3, 4, 5, 6 and 7 in Figure 4.11 (a) are not significantly different to each other and they all outperform setting1.
 - The MCBA-wsNGH and the SCBA-wsNGH ($w_2=100,000$) Figure 4.11 (b and e): settings 2, 4, 6 and 7 have no significant differences from each other and also they outperform the remaining settings.
 - The SCBA-wsNGH ($w_2=1$) and the SCBA-wsNGH ($w_2=1,000$) Figure 4.11 (c and d): settings 2, 3, 4, 5, 6 and 7 have no significant differences from each other and they outperform setting1.

Algorithm	The best cost		The best emissions	
	f_c	$Corr. f_e$	$Corr. f_c$	f_e
SCBA-basicNGH	≡	≡	≡	≡
SCBA-randomNGH	≠	≠	≠	≠
MCBA-wsNGH	≠	≡	≠	≠
SCBA-wsNGH ($w_2=1$)	≠	≡	≡	≠
SCBA-wsNGH ($w_2=1,000$)	≠	≡	≡	≠
SCBA-wsNGH ($w_2=100,000$)	≠	≡	≠	≠

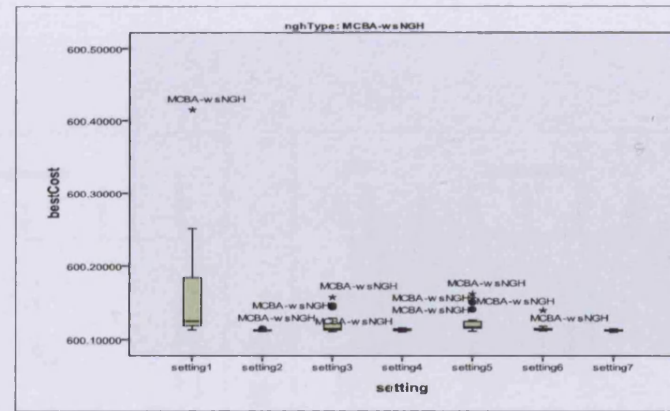
≡ not significantly different regarding settings

≠ significantly different at least one setting

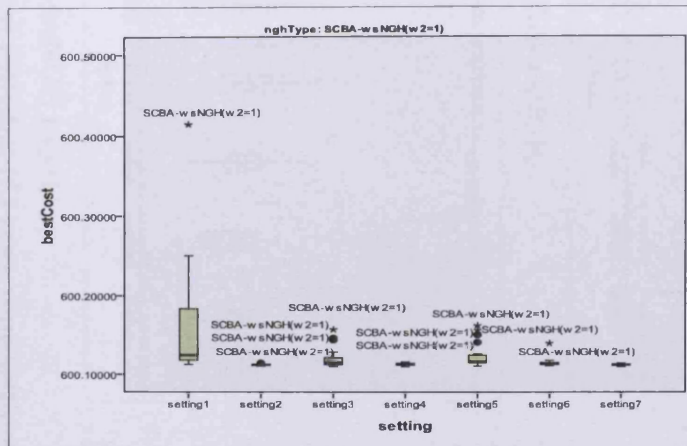
Table 4.14 The result of statistic analyses regarding settings by *SPSS*



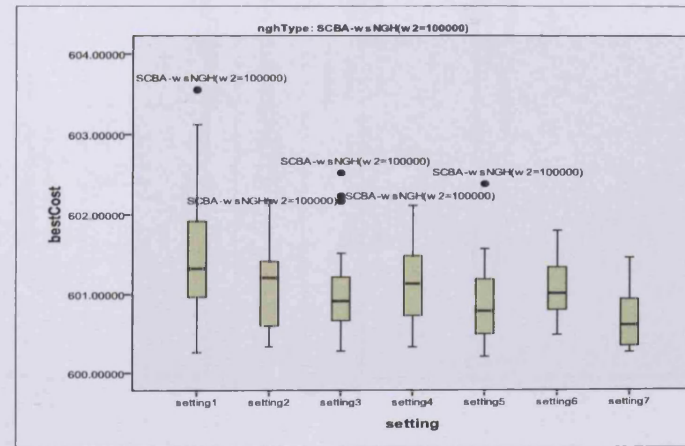
(a) SCBA-randomNGH



(b) MCBA-wsNGH

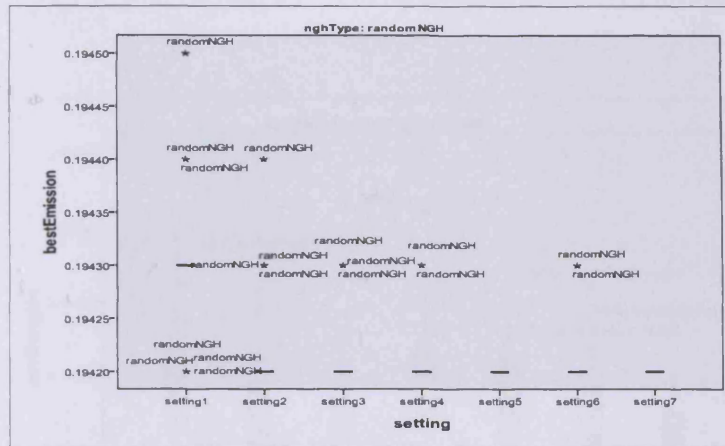


(c) SCBA-wsNGH ($w_2=1$)

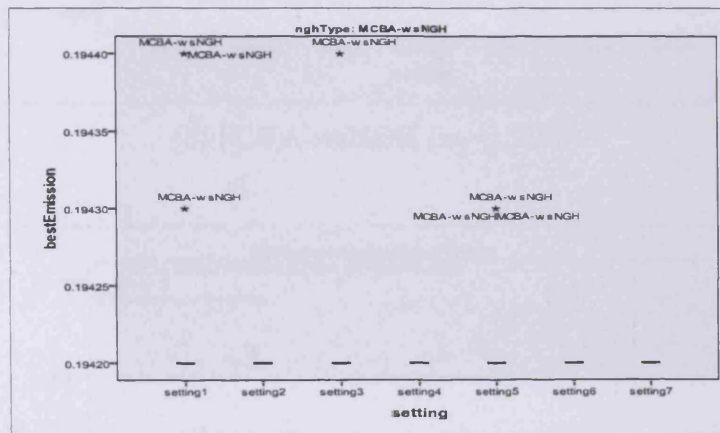


(d) SCBA-wsNGH ($w_2=100,000$)

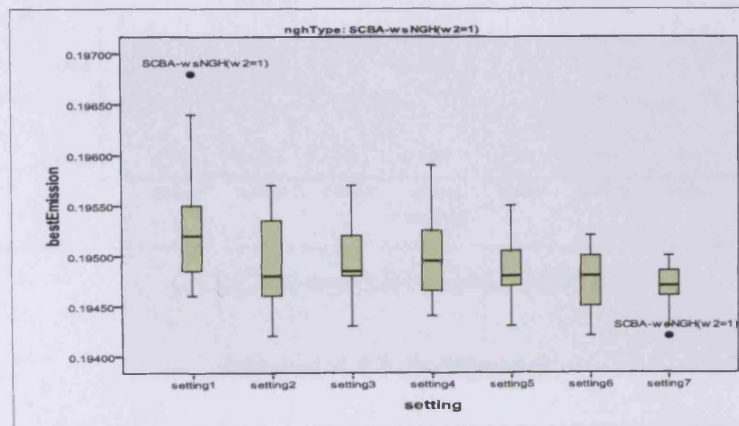
Figure 4.10 Box plot of the best cost (f_c) regarding settings



(a) SCBA-randomNGH

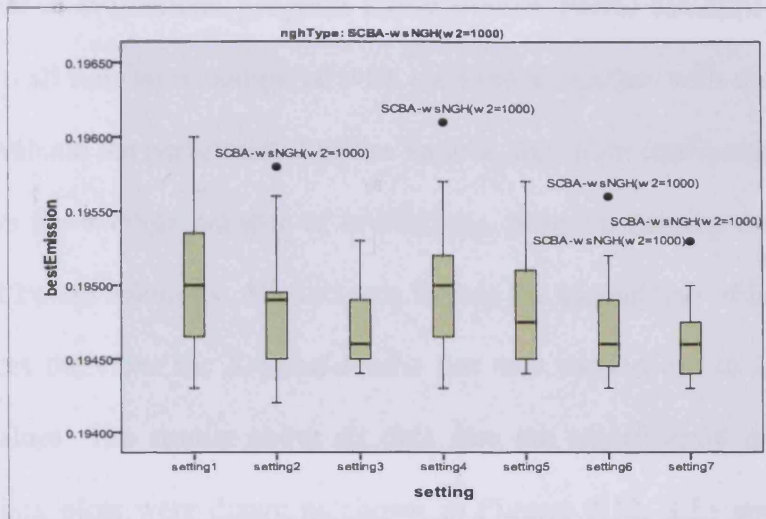


(b) MCBA-wsNGH

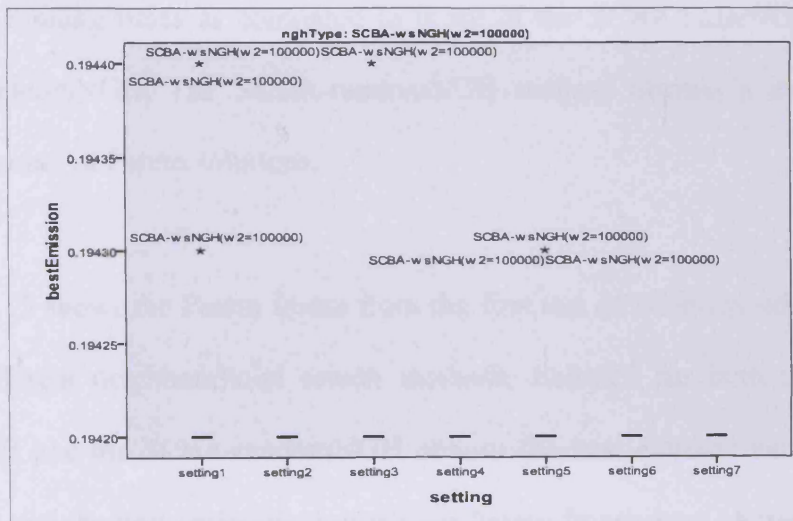


(c) SCBA-wsNGH ($w_2=1$)

Figure 4.11 Box plot of the best emission (f_e) regarding settings



(d) SCBA-wsNGH ($w_2=1,000$)



(e) SCBA-wsNGH ($w_2=100,000$)

Figure 4.11 continued

4.4.1.2 Additional comparisons

The number of evaluations, program running times, Pareto solutions and Pareto fronts from all runs were compared with each other together with the settings in order to evaluate the performance of the various algorithm configurations. Table 4.15 shows the average number of evaluations, program running times and the number of Pareto solutions. All data sets violate the assumption of homogeneity of variances therefore the *Kruskal-Wallis* test was carried out to compare the median values. The results show all data sets are significantly different and therefore box plots were drawn as shown in Figures 4.12, 4.13 and 4.14. The MCBA-wsNGH method requires a significantly lower number of evaluation and program running times as compared to those of the SCBA-basicNGH and the SCBA-randomNGH. The SCBA-randomNGH method obtains a significantly larger number of Pareto solutions.

Figure 4.15 shows the Pareto fronts from the first test of 20 independent runs of three different neighbourhood search methods. Setting7 for both the SCBA-baiscNGH and the SCBA-randomNGH obtains the best extreme values for the best cost and the best emissions hence their Pareto fronts were plotted. Setting8 for the MCBA-wsNGH also drew the same conclusion and its Pareto front is shown in Figure 4.15 (c).

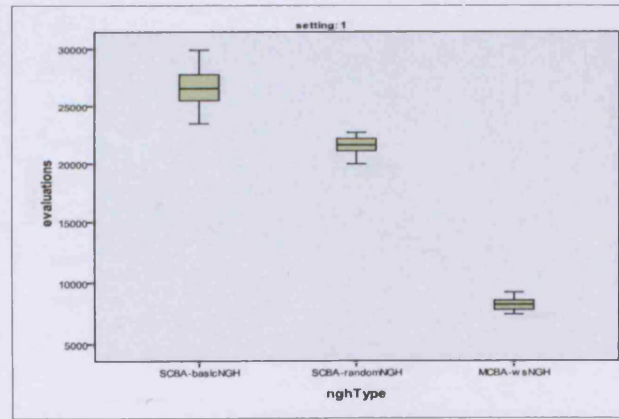
A good Multi-Objective Optimisation (MOO) algorithm will attempt to generate an approximate set with a uniform distribution of trade-off points which is that simultaneously covers the largest range in the objective space very close to the Pareto front. Therefore solving MOOPs is in itself a Multi-Objective problem (Lee and El-Sharkawi 2008).

Although all Pareto fronts generated by the proposed Bees Algorithms are very close to each other (Figures 4.15), the objective space range scoured by the random selection method and weighted sum neighbourhood search method is wider than for the basic neighbourhood search method. However, it is noticeable that the required number of evaluations in the MCBA-wsNGH is almost three or four times less than the others.

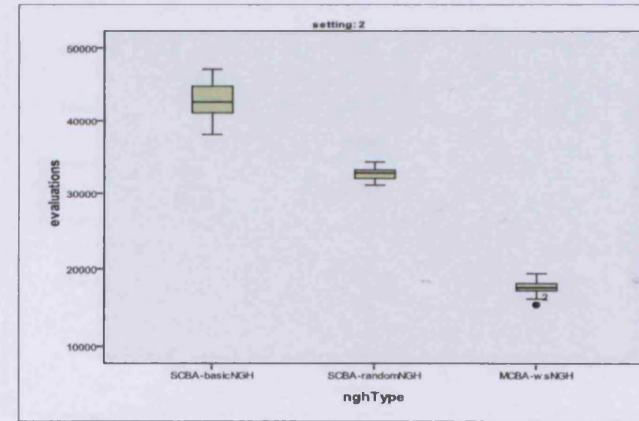
Figure 4.16 shows the comparison of Pareto fronts for the weighted sum neighbourhood search methods with regard to cycles. The SCBA-wsNGH ($w_2=1$) focuses on searching for the best cost Figure 4.16 (a) and the SCBA-wsNGH ($w_2=1,000$) focuses on searching for the middle of the Pareto front Figure 4.16 (b). The SCBA-wsNGH ($w_2=100,000$) focuses on searching for the best emissions Figure 4.16 (c), whilst the MCBA-wsNGH takes only dominant solutions to generate an enhanced Pareto front by combining the three Pareto fronts.

BA	<i>ngh</i> method	setting	Evaluations	Times (s)	Solutions
SCBA	basicNGH	1	26,776	3,555	22
		2	42,711	5,596	25
		3	59,390	7,002	27
		4	94,945	9,951	29
		5	130,545	13,081	31
		6	89,149	14,763	33
		7	194,882	27,837	40
	randomNGH	1	21,524	3,597	297
		2	32,703	5,552	403
		3	43,768	7,492	491
		4	65,025	11,035	600
		5	86,664	14,537	689
		6	67,515	15,111	525
		7	135,266	30,171	742
MCBA	wsNGH	1	8,209	591	96
		2	17,567	1,211	234
		3	11,016	876	120
		4	15,066	1,134	179
		5	11,677	1,499	117
		6	14,407	1,556	149
		7	24,645	2,649	275
		8	52,058	5,172	511
		9	15,239	2,522	131
		10	17,440	2,915	143
		11	36,377	5,557	329

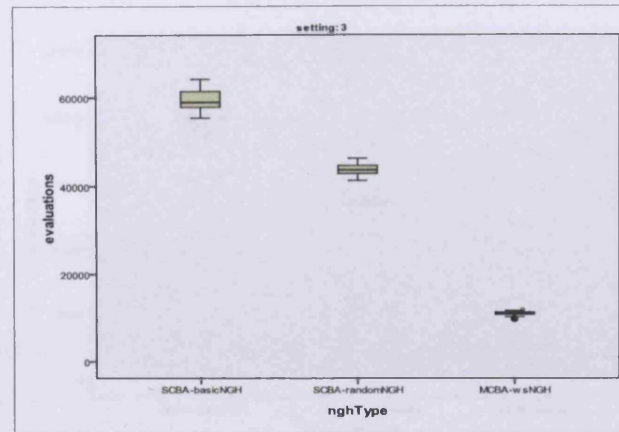
Table 4.15 The average number of evaluations, program running times and number of Pareto solutions



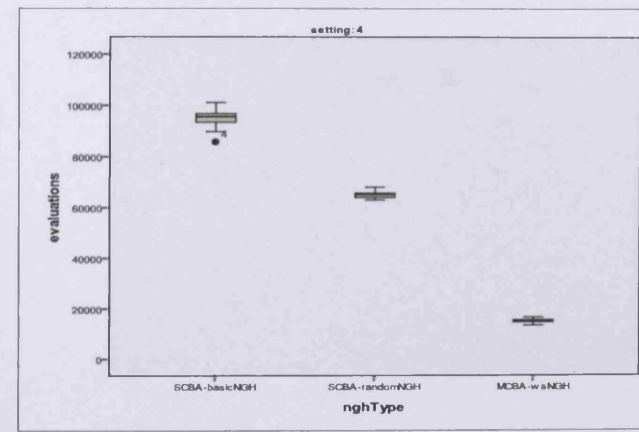
(a) setting1



(b) setting2

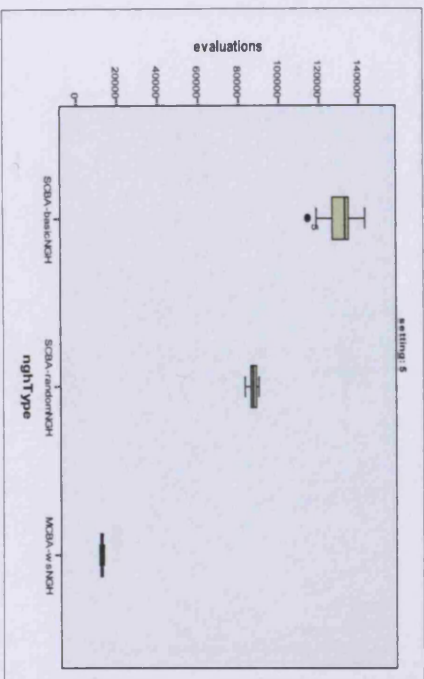


(c) setting3

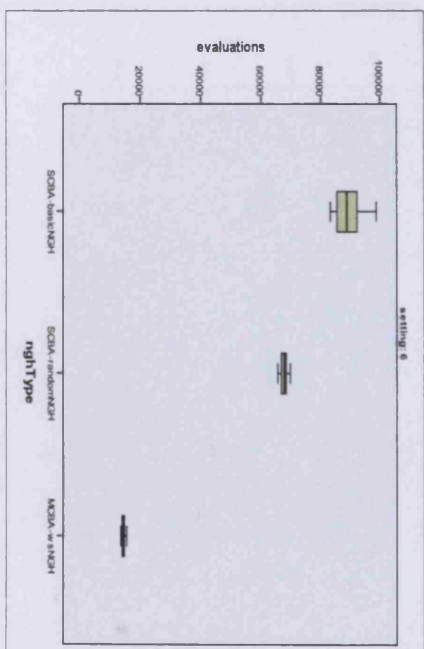


(d) setting4

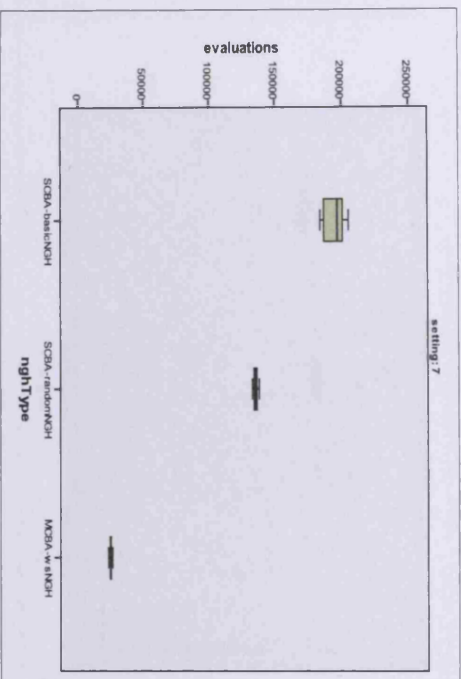
Figure 4.12 Box plots of the number of evaluations



(e) setting5

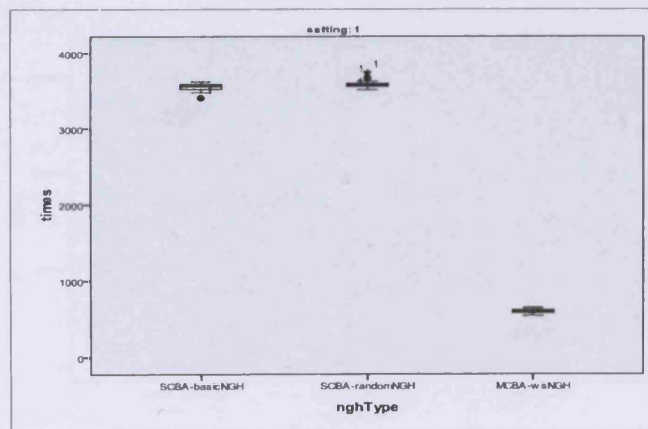


(f) setting6

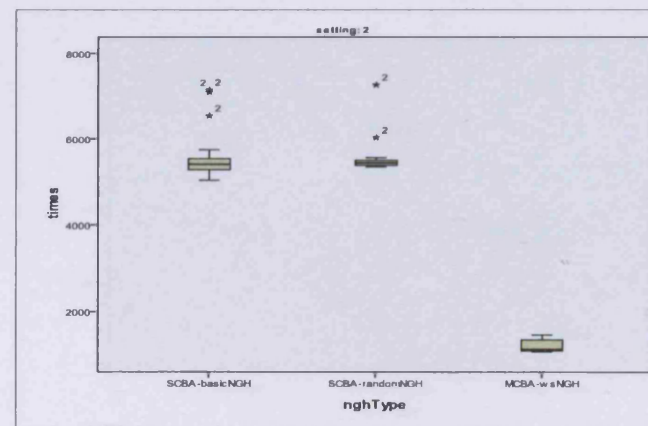


(g) setting7

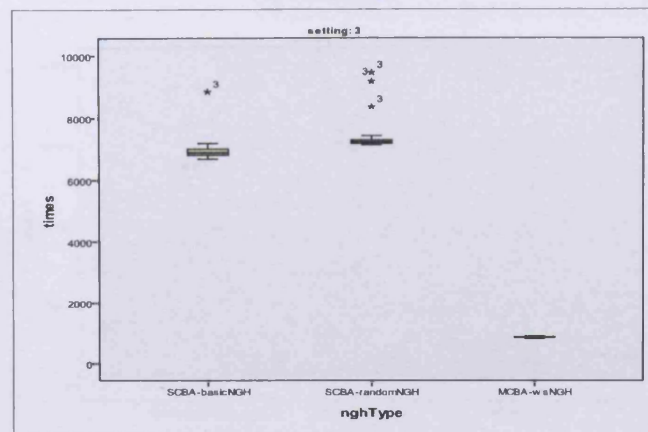
Figure 4.12 continued



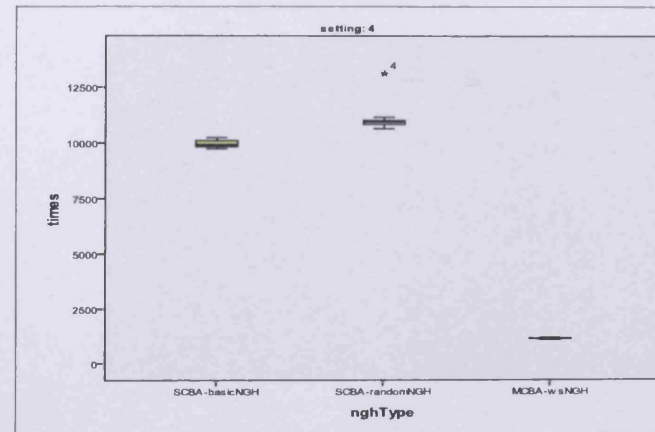
(a) setting1



(b) setting2

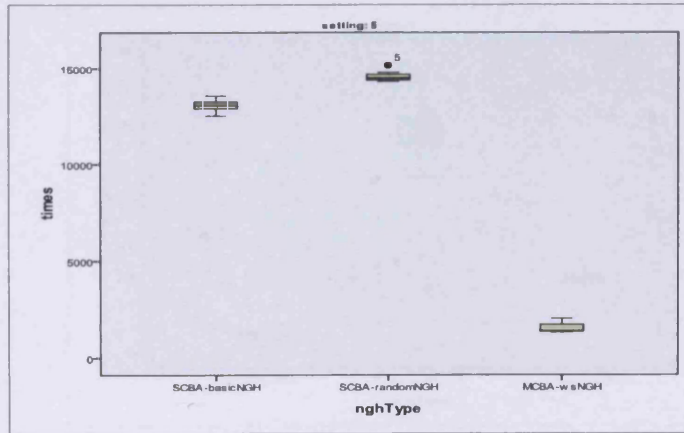


(c) setting3

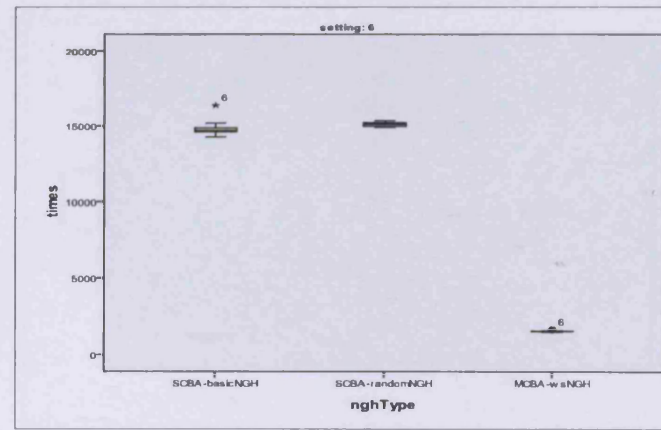


(d) setting4

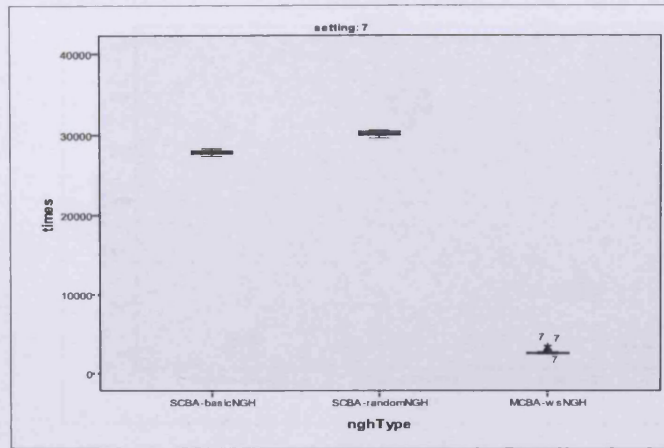
Figure 4.13 Box plots of program running times



(e) setting5

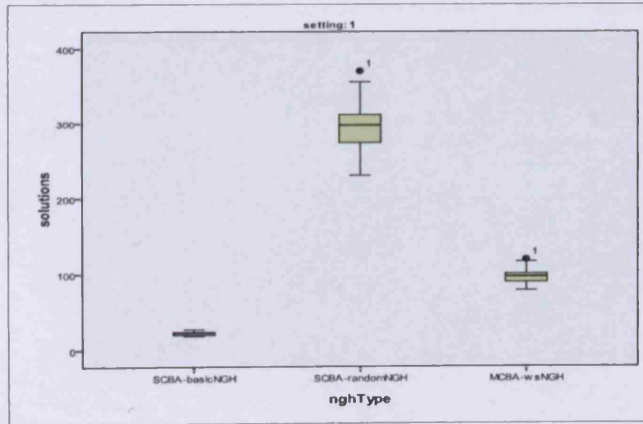


(f) setting6

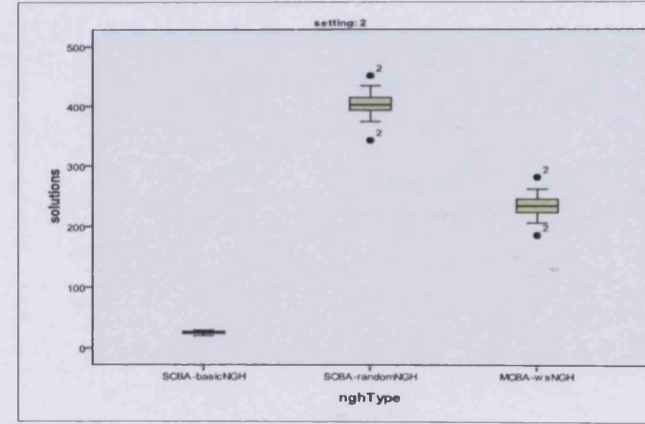


(g) setting7

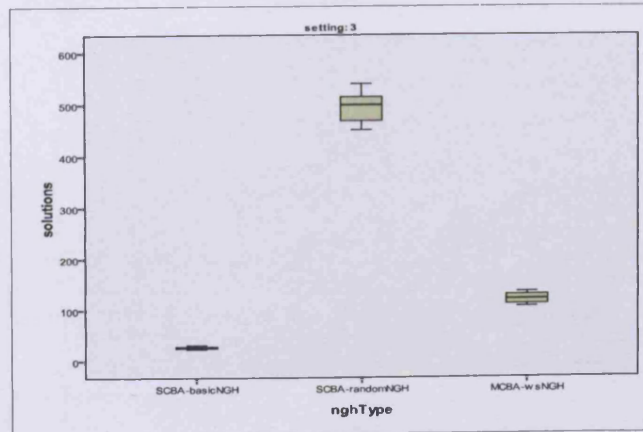
Figure 4.13 continued



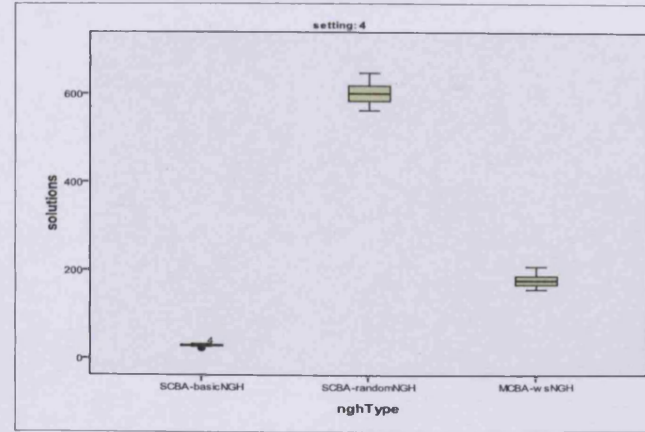
(a) setting1



(b) setting2

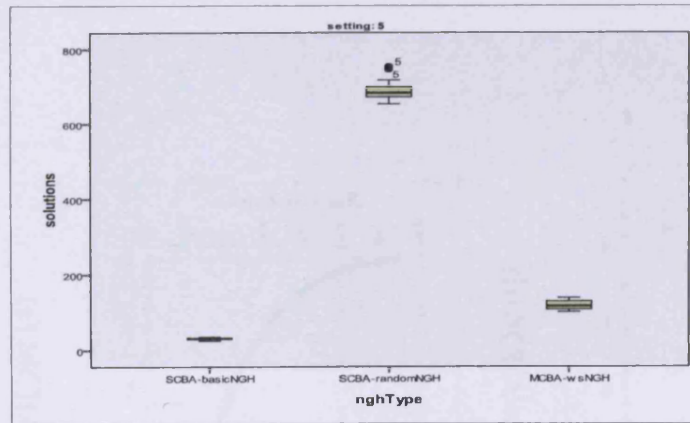


(c) setting3

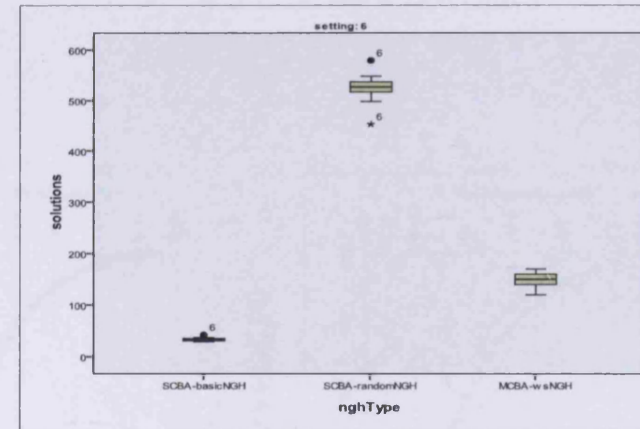


(d) setting4

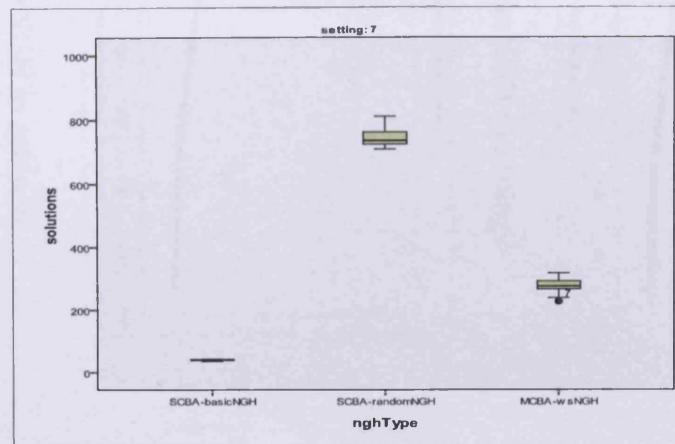
Figure 4.14 Box plots of the number of Pareto solutions



(e) setting5

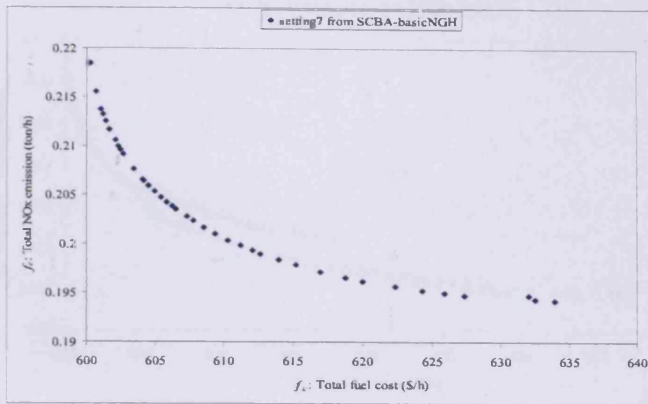


(f) setting6

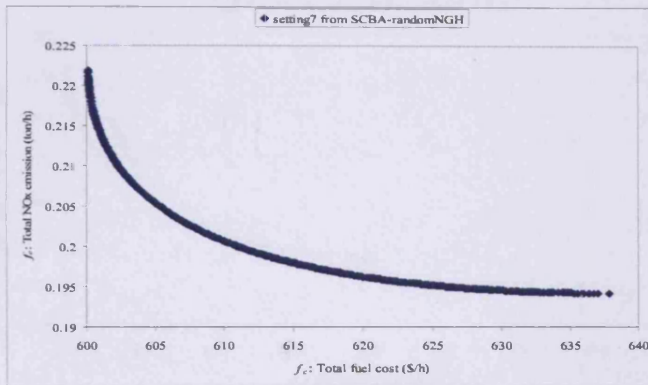


(g) setting7

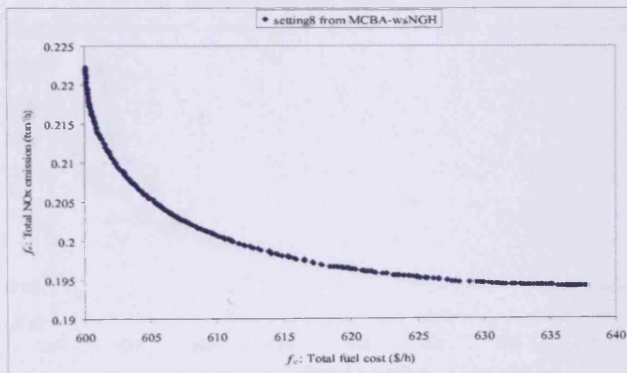
Figure 4.14 continued



(a) SCBA-basicNGH in setting7

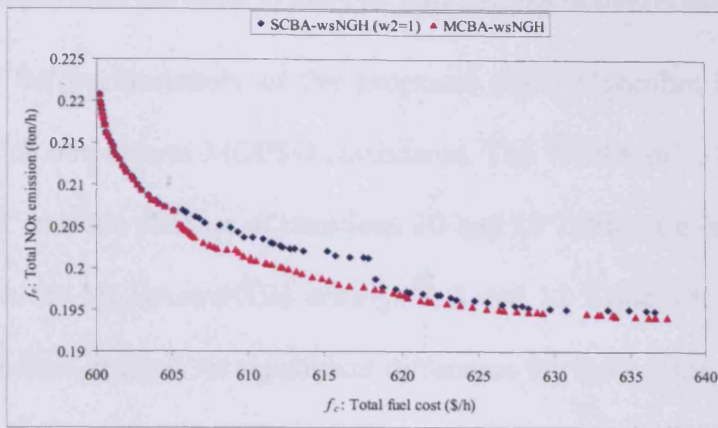


(b) SCBA-randomNGH in setting7

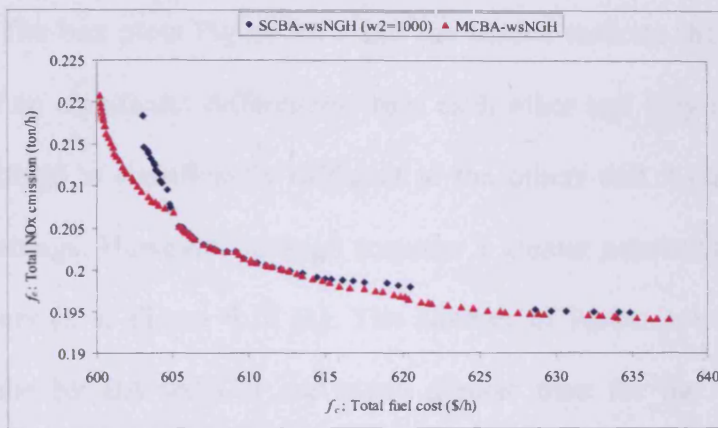


(c) MCBA-wsNGH in setting8

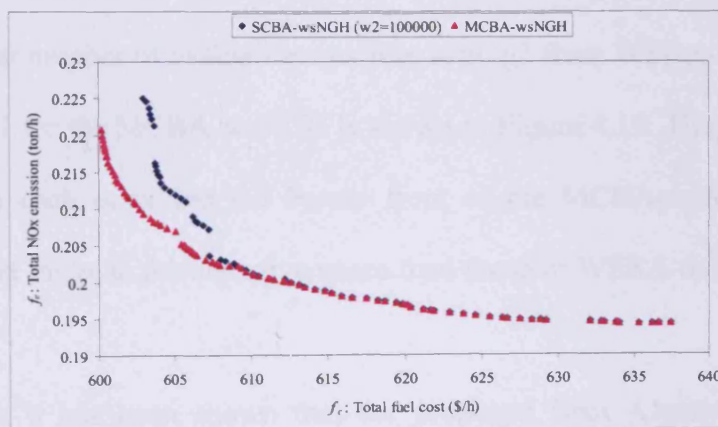
Figure 4.15 Pareto fronts regarding neighbourhood search methods



(a) SCBA-wsNGH ($w_2=1$) and MCBA-wsNGH



(b) SCBA-wsNGH ($w_2=1,000$) and MCBA-wsNGH



(c) SCBA-wsNGH ($w_2=100,000$) and MCBA-wsNGH

Figure 4.16 Comparison of Pareto fronts from wsNGH in setting1

4.4.1.3 Comparison between WSBA-*m* and MCBA-wsNGH methods

To compare the performances of the proposed Bees Algorithm in Chapter 3, data sets which outperform MOPSO considered. The WSBA-*m*(22) method with ‘*ngh* setting2’ and the number of iterations 20 and 25 Tables 3.4 was selected to compare with the MCBA-wsNGH settings 7, 8 and 11 Table 4.6. According to the SPSS outcome, there is no significant difference for finding the best emission value (f_e), but there is a significant difference when finding the best cost (f_c), Table 4.16. The box plots Figure 4.17 and the results indicate that settings 2, 7 and 11 have no significant differences from each other and they all outperform setting1. Setting8 is significantly different to the others and it outperforms the remaining settings. However, setting8 requires a greater number of evaluations than the others as in Figure 4.18 (a). The number of Pareto solutions from all settings in the MCBA-wsNGH are much greater than for the WSBA-*m*(22) method as shown in Figure 4.18 (b). A comparison between their Pareto fronts after a similar number of evaluations such as setting2 from WSBA-*m*(22) method and setting11 for the MCBA-wsNGH is shown in Figure 4.19. They approximate closely with each other but the Pareto front of the MCBA-wsNGH covers a slightly larger range in the objective space than those of WSBA-*m*(22).

In summary, it has been shown that the proposed Bees Algorithm is ideally suited to the EEDP in obtaining the Pareto front. Pareto optimality required just the one single program run to achieve a satisfactory number of solutions,

whereas the Bees Algorithm with weighted sum (Lee and Haj Darwish 2008) method needed numerous runs to obtain a suitable outcome. It also alleviates one of the drawbacks of intelligent techniques for the MOOP because it requires only half the number of parameters which need to be adjusted as those for the Bees Algorithm (Pham et al. 2005, 2006a) which needs to be adjusted. The three proposed new neighbourhood search methods (basicNGH, randomNGH and wsNGH) demonstrated good diversity. However, the randomNGH and the wsNGH covered the widest range in the objective space for Pareto optimality whilst achieving the greatest number of solutions in comparison with the basic neighbourhood search method (basicNGH).

The best cost (f_c)		The best emissions (f_e)	
<i>Kruskal-Wallis</i>	<i>ANOVA</i>	<i>ANOVA</i>	<i>Kruskal-Wallis</i>
f_c	<i>Corr. f_e</i>	<i>Corr. f_c</i>	f_e
\neq	\equiv	\neq	\equiv

\equiv not significantly different

\neq significantly different

Table 4.16 The result of statistical analyse of two extreme Pareto solutions

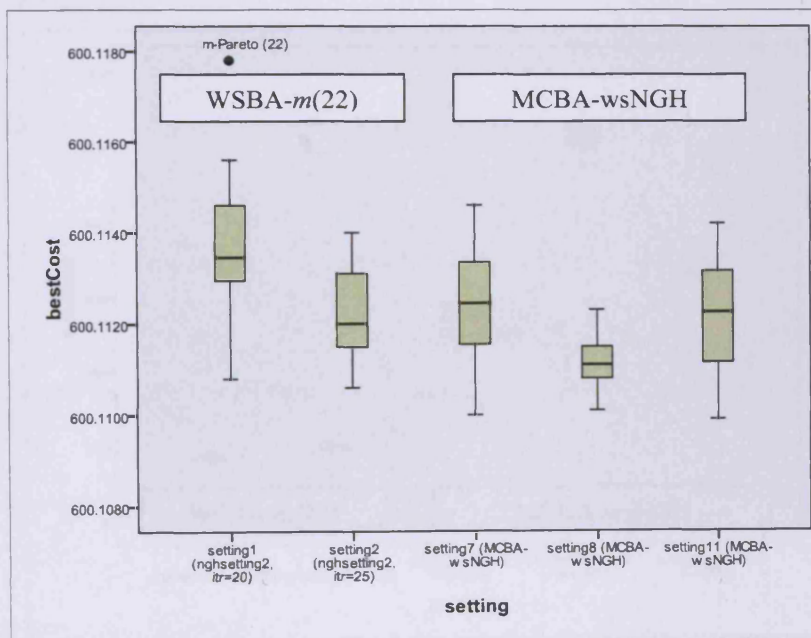
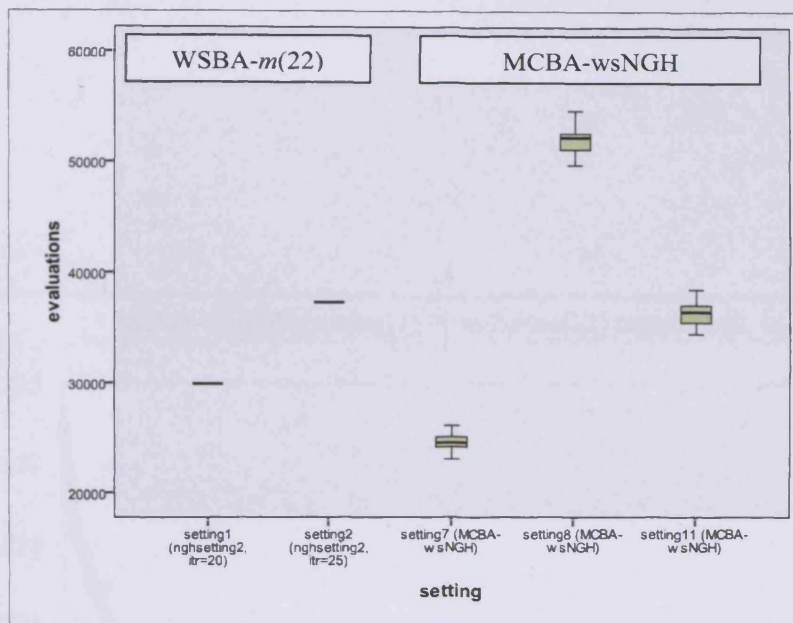
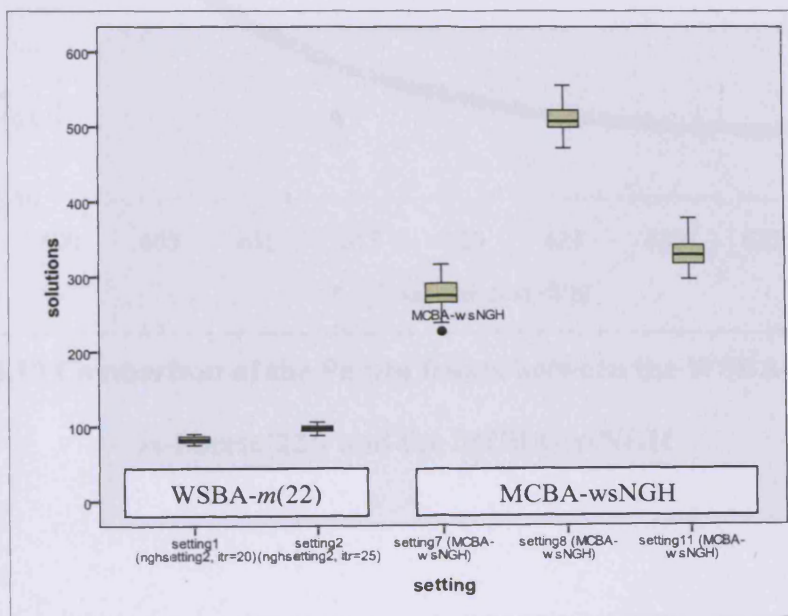


Figure 4.17 Box plots of the best cost between the WSBA- $m(22)$ and the MCBA-wsNGH



(a) Box plots of the number of evaluations



(b) Box plots of the number of Pareto solutions

Figure 4.18 Comparison between the WSBA- $m(22)$ and the MCBA-wsNGH

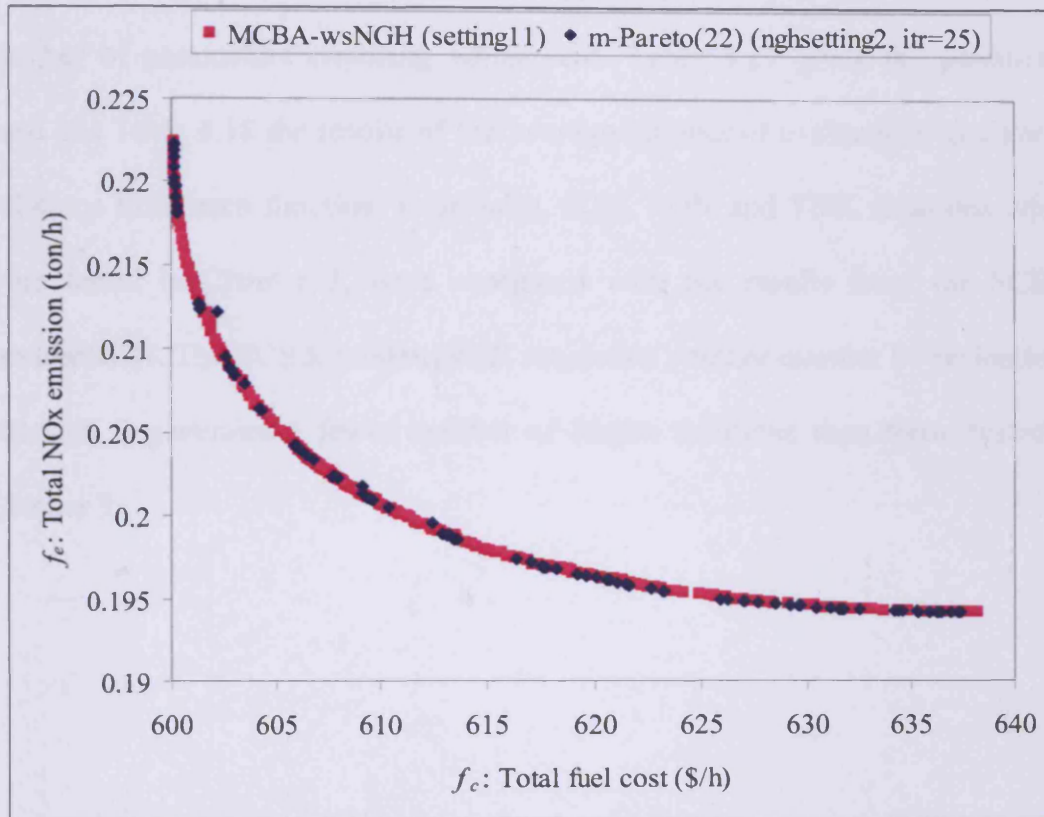


Figure 4.19 Comparison of the Pareto fronts between the WSBA- $m(22)$ (or m -Pareto(22)) and the MCBA-wsNGH

4.4.2 Results of benchmark mathematical functions

More Multi-Objective functions with various characteristics are tested and compared with those of Chapter 3. To evaluate the proposed Bees Algorithm, and in particular the SCBA-randomNGH was employed because it has a smaller number of parameters requiring adjustment. Table 4.17 gives the parameters used and Table 4.18 the results of the average number of evaluations and Pareto solutions from each function. Especially, SCH, FON and TNK functions which were tested in Chapter 3, were compared with the results from the SCBA-randomNGH. The SCBA-randomNGH requires a smaller number of evaluations, although it generates a fewer number of Pareto solutions than those tested in Chapter 3.

Function	<i>n</i>	<i>ngh</i>	<i>nm</i>	<i>itr</i>
SCH	50	1	5	100
ZDT1, ZDT3, FON, POL, CONSTR, SRN and TNK	50	0.1	5	50
KUR	200	0.1	5	200
ZDT2	100	0.01	10	50
WATER	50	0.01	5	200
3 functions	50	0.01	5	200

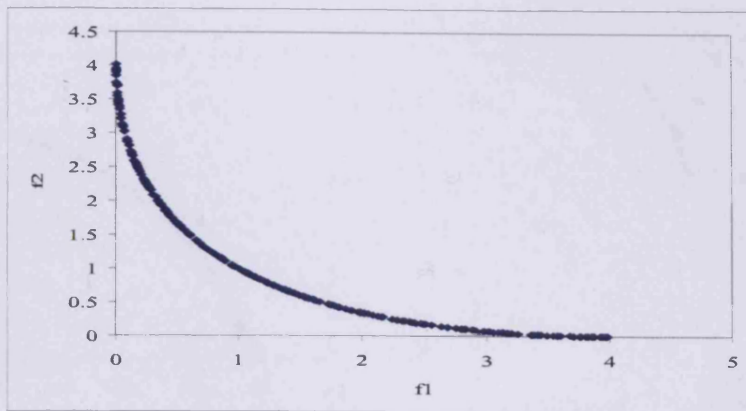
Table 4.17 Parameters for the Bees Algorithm

Function	The average number of	
	Evaluations	Solutions
SCH	6,161 (6 times less)*	234 (9 times less) *
ZDT1	12,494	2,440
ZDT3	6,799	890
FON	3,574 (13 times less) *	90 (2 times less) *
KUR	44,840	107
POL	4,734	202
ZDT2	50,100	4,991
CONSTR	4,950	185
SRN	10,670	963
TNK	5,122 (9 times less) *	90 (3 times less) *
WATER	44,958	2,967
3 functions	45,678	1,495

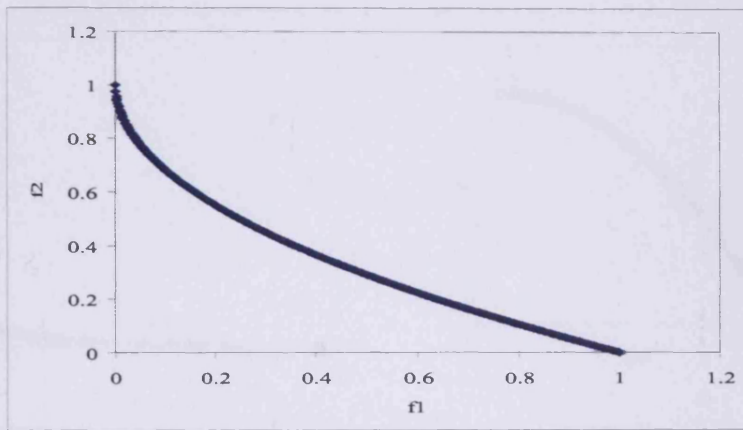
*: than WSBA- $m(20)$ in Table 3.15

Table 4.18 The result of the number of evaluation and Pareto solutions

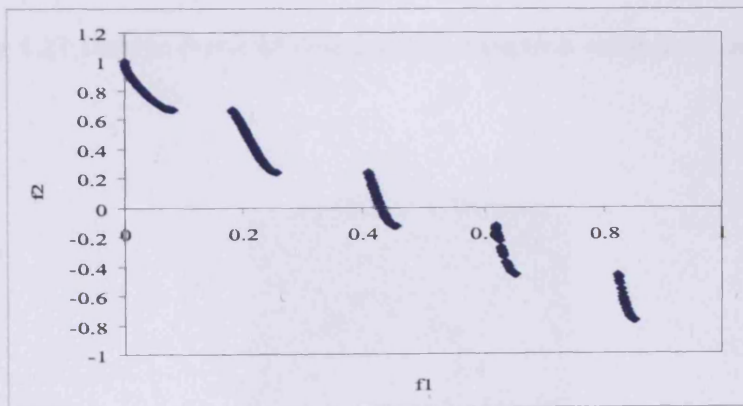
Figure 4.20 shows the Pareto fronts for convex functions without constraints. All the Pareto fronts are well defined and cover all ranges in the objective space. Figure 4.21 shows the Pareto fronts for non-convex functions without constraints. They also show good definition and cover all ranges in the objective space. Therefore, it can be stated that the SCBA-randomNGH is less sensitive to the shape of the Pareto front profile. Figure 4.22 shows the Pareto front of a three-objective function without constraints and Figure 4.23 illustrates the Pareto fronts of functions with constraints. They are also well shaped and cover all ranges in the objective space. Figure 4.24 depicts a comparison of the Pareto fronts generated from the WSBA- $m(20)$ and the SCBA-randomNGH. Both fronts deliver good shapes and cover all ranges in the objective space. It should be noted that the shape of the Pareto front from the SCBA-randomNGH has better definition and required 9 times less effort than those of the WSBA- $m(20)$ in TNK function Figure 4.24 (c).



(a) SCH

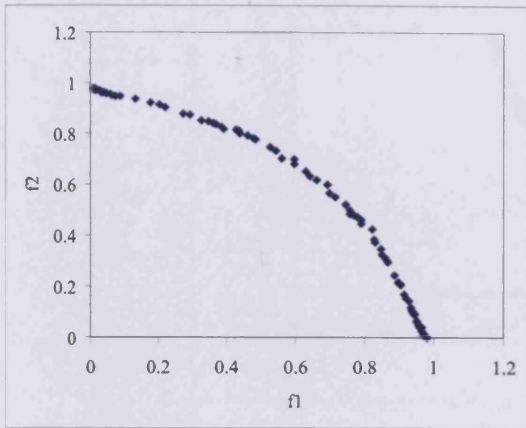


(b) ZDT1

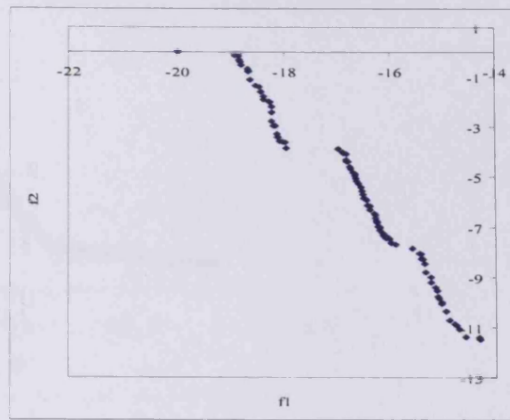


(c) ZDT3

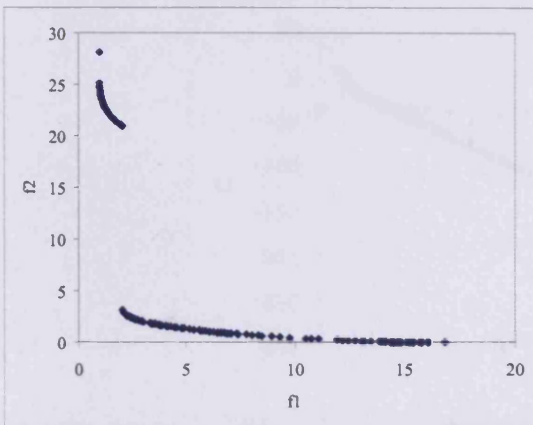
Figure 4.20 Pareto fronts of convex functions without constraints



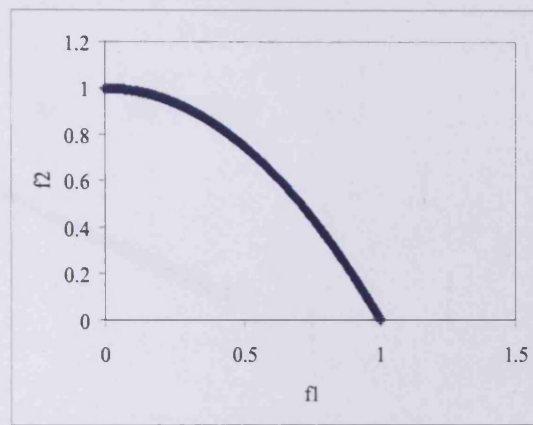
(a) FON



(b) KUR



(c) POL



(d) ZDT2

Figure 4.21 Pareto front of non-convex function without constraints

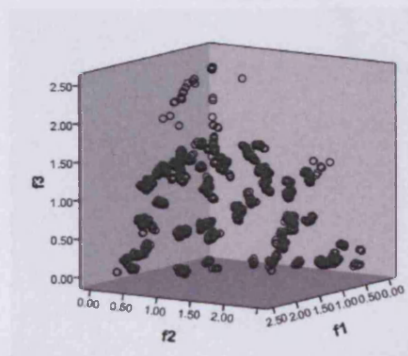
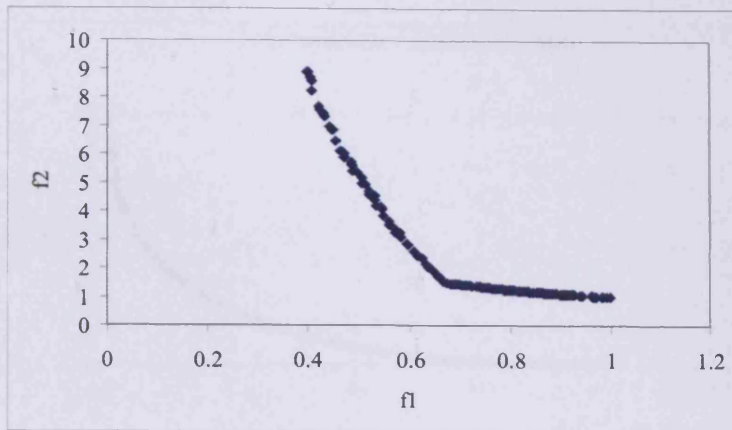
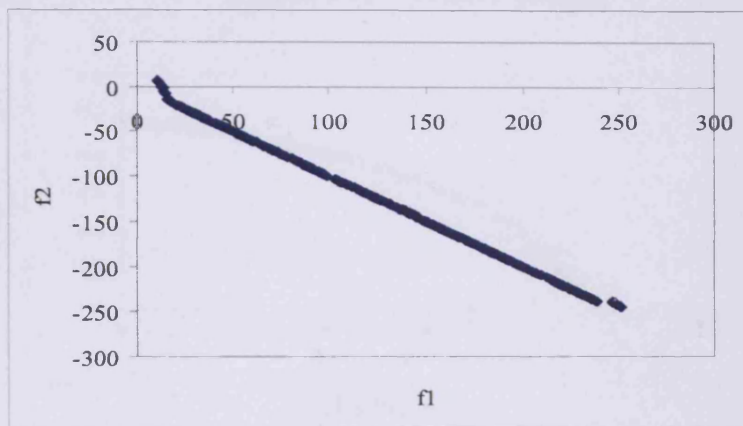


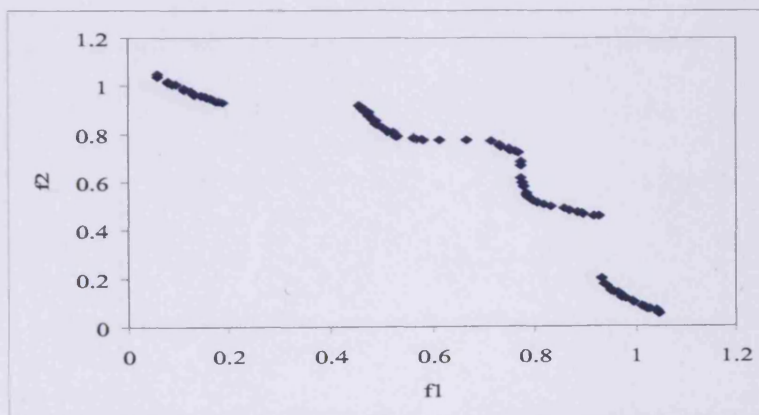
Figure 4.22 Pareto front of three-objective function without constraints



(a) CONSTR

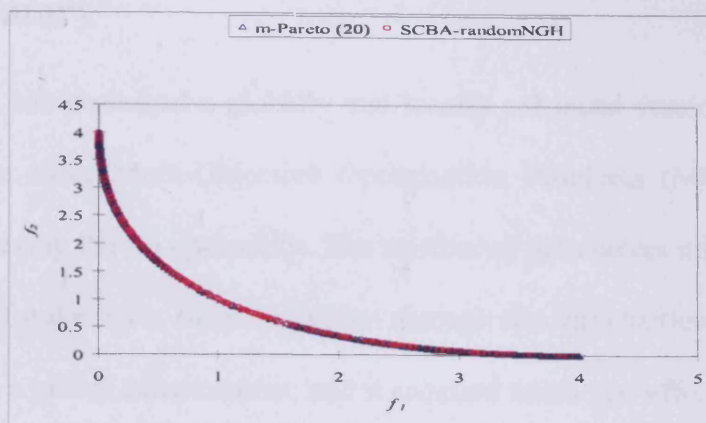


(b) SRN

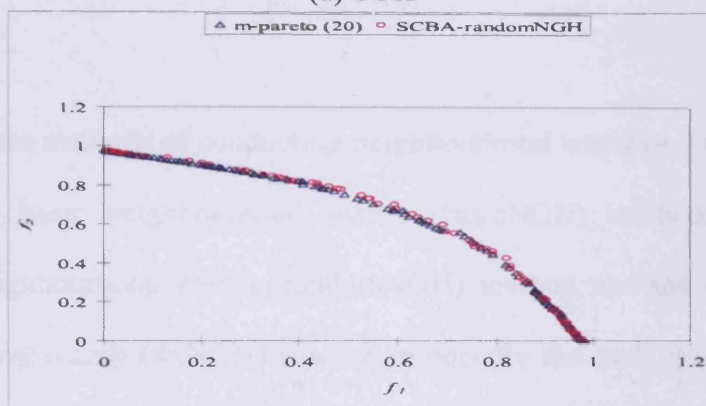


(c) TNK

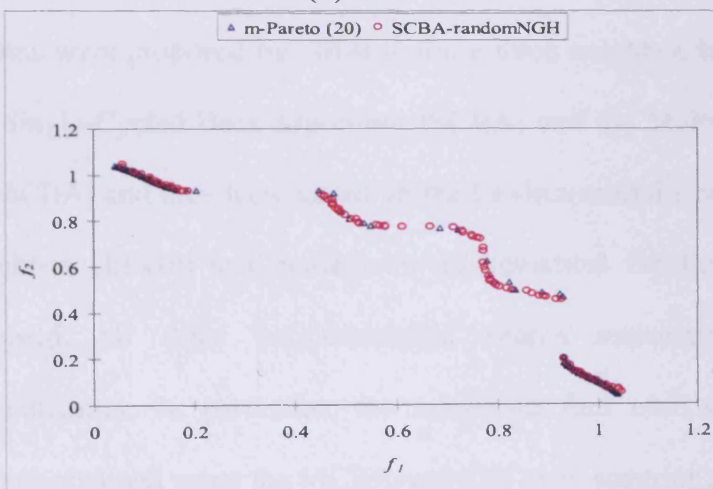
Figure 4.23 Pareto fronts of functions with constraints



(a) SCH



(b) FON



(c) TNK

Figure 4.24 Comparison of Pareto fronts between the WSBA- $m(20)$ (or m -Pareto(20)) and the SCBA-randomNGH

4.5 Summary

This chapter has described a globally and locally enhanced version of the Bees Algorithm to solve Multi-Objective Optimisation Problems (MOOPs) with a view to achieving Pareto optimality. The number of parameters utilised was half of that used for the basic Bees Algorithm through the introduction of a dynamic procedure as a global enhancement, and it required much less effort than existing algorithm.

Three different methods of conducting neighbourhood searches were developed, namely, the basic neighbourhood search (basicNGH) method, the random selection neighbourhood search (randomNGH) method and the weighted sum neighbourhood search (wsNGH) method, especially the both randomNGH and wsNGH were developed as a local enhancement to enhance the Pareto front. Two algorithms were proposed by utilising these three neighbourhood methods, namely, the Single-Cycled Bees Algorithm (SCBA) and the Multi-Cycled Bees Algorithm (MCBA) and they were tested on the Environmental/Economic power Dispatch Problem (EEDP) and benchmark mathematical functions. From the results obtained, all three neighbourhood search approaches delivered satisfactory outcomes. In particular, the minimum fuel cost and minimum emission values obtained using the MCBA-wsNGH were superior to those of the other existing approaches. Moreover, the proposed Bees Algorithm also gave a

Pareto frontier with an excellent diversity for achieving Pareto optimality whilst proving to be both stable and robust.

In summary this work has confirmed the suitability of the Bees Algorithm for solving Multi-Objective optimisation problems.

5 OPTIMISATION OF MULTI-FUEL ENERGY SYSTEMS FOR A LOW-CARBON CITY USING THE BEES ALGORITHM

5.1 Preliminaries

Issues of global warming and greenhouse gas emissions are increasingly becoming some of the major technological as well as important societal and political challenges (Klimes and Pierucci 2008). These challenges are related to energy generation and usage and all nations face them (Karlsson et al. 2009). Cities are responsible for nearly 75% of the world's energy consumption and about 90% of future growth will occur in urban areas (Bhatt et al. 2010). In regions with a high population increase due to rapid economic growth (e.g., China and India) a requirement for new cities and urban areas will arise in the future (Bouvy and Lucas 2007; Day et al. 2009). One crucial factor for the optimal development of such areas is to satisfy the total electrical, thermal and cooling energy demands. Energy usage in the building sector is increasing rapidly, and there is the necessity to reduce fossil fuel consumption. Fossil fuel usage for building applications results in dwindling energy resources and deteriorating environmental conditions. Two approaches can be used to mitigate these two effects: enhancing the energy saving features of buildings (e.g., Zero

Energy House: ZEH) and using renewable energy sources like solar, wind and biomass (Zhu et al. 2009).

A system designed to allow the operator to choose between multiple energy sources is referred to as a multi-energy system (or hybrid energy system). These systems are currently receiving increasing attention as they represent a valuable means of exploiting renewable energy sources and options for facility companies. There are various types of multi-energy systems, using different combinations of thermal and electric equipment such as co-generators, electric chillers, engine-driven chillers, gas or steam absorption chillers, fuel cells, traditional boilers, wood burning appliances, thermal solar collectors, photovoltaic collectors, thermal and photovoltaic collectors, etc. A multi-energy system is therefore fed by a combination of various energy sources, both renewable and non-renewable, to meet the thermal and electric loads of a building with the maximum efficiency. Several examples of multi-energy source building systems can be found in the literature (Dalton et al. 2009a, b; Hawkes and Leach 2005; Ooka and Komamura 2009). Recently researchers are more interested in large scale energy systems such as an island (Koroneos et al. 2004) and urban (Aki et al. 2003; Bhatt et al. 2010; Sugihara et al. 2004; Zhu et al. 2008) and remote communities (Favrat and Pelet 2007; Henning et al. 2006; Pelet et al. 2005). (Bhatt et al. 2010) received good reviews regarding integrated energy and environmental systems analysis methodology to facilitate a Low-Carbon City.

The design of multi-energy systems involves resolving problems such as: the correct sizing and the efficiency of the different systems and the cost and availability of different energy wares. Another important design and operational problem concerns the mismatch between the energy demand (the load) and the energy supply (both renewable and conventional energy sources) which is usually addressed through the integration of storage and/or a back-up energy source and connection to the power network. Since there are many adoptable configurations, it is important to carry out an optimisation study between the energy demand, the energy supply, the converters, the storage and the back-up sources characteristics when designing and operating a multi-energy system (Fabrizio et al. 2010). The conflicting criteria regarding the cost and CO₂ emissions leads to a large number of structural and operational variants to be taken into consideration (Østergaard 2009). Moreover traditional research has been more concerned with operating cost rather than capital cost, but the initial capital cost of renewable systems is considerably more expensive than conventional power systems, and the Decision Maker (DM) or an engineer cannot easily design their installation and capacity. Hence the design process for the planning engineer is complex and time consuming (Hobbs 1995). Thus computer based tools in the design phase are essential (Bouvy and Lucas 2007).

Consequently many researchers developed or employed design models such as MODEST (Henning et al. 2006) and refined them using various optimisation

algorithms such as GA (Ooka and Komamura 2009) and the Queuing Multi-Objective Optimiser (QMOO) (Leyland 2002; Molyneaux 2002) or tools such as HOMER (Dalton et al. 2009a, b). Further reforms include:

- (Hernandez-Aramburo et al. 2005) and (Mohamed 2008; Mohamed and Koivo 2007a; Mohamed and Koivo 2007b) introduced the Micro-Grid (MG) system which combined a variety of power sources.
 - The MG system in (Hernandez-Aramburo et al. 2005) consists of two reciprocating gas engines, a Combined Heat and Power (CHP) plant, a photovoltaic (PV) array and a wind generator. The optimisation is aimed at minimising the fuel consumption of the system and constraining it to fulfil the local electrical and thermal energy demands whilst providing a certain minimum reserve power. A penalty is applied for any excess heat produced. It was considered only as an economic dispatch problem as it did not take any environmental factors into consideration.
 - The MG system in (Mohamed and Koivo 2007a; Mohamed and Koivo 2007b) was applied as an environmental/economic problem because the model considered the operational and maintenance costs as well as the emissions, NO_x, SO₂ and the reduction of CO₂. It consists of a wind turbine, a micro turbine, a diesel generator, a PV array and fuel cell. In this model the only constraint considered was total load demand.

- (Henning 1997) introduced a Model for Optimisation of Dynamic Energy Systems with Time-Dependent Components and Boundary Conditions (MODEST) which uses linear programming to minimise the capital and operational costs of energy supply and demand-side management. However, the capital cost can be no longer the sole consideration due to rising environmental restrictions.
- MERIT (Born 2001; Born et al. 2001a; Born et al. 2001b; Clarke et al. 2007) has been developed as a quantitative appraisal tool for assisting decision-makers to design and plan renewable and low energy systems using detailed hourly demand and supply data. One key feature of this tool is its matching facility, which uses a statistical method to evaluate how closely the hourly demand and supply profiles can be matched. Two coefficients, an inequality coefficient and a correlation coefficient, are employed to analyse the differences in magnitude and change of phase between demand and supply profiles. The correlation coefficient establishes the phase matching between demand and supply, while the inequality coefficient ascertains the matching magnitude. A detailed description of this methodology can be found elsewhere (Scheaffer and McClave 1982; Williamson 1995). Normally, a user manually tries to find a good match rate which will not be an optimal solution in terms of capital cost and CO₂ emission.
- The Renewable Energy System (RES) software HOMER (Lambert et al. 2004) was developed by The National Renewable Energy Laboratory in the

US and is utilised as an assessment tool with modelling performed from hourly load data input. It considered capital cost and CO₂ emissions separately. The selection criteria are frequently in conflict so different objective functions lead to differing results, especially for economy versus energy or environmental objective functions. Hence this problem needs to be handled as a Multi-Objective problem.

The aim of this work is to design an optimisation tool to provide an optimal approximate capacity for a multi-fuel energy system in conjunction with a MERIT program for a Low-Carbon City. To design the optimisation tool as a Multi-Objective solver, the Bees Algorithm is applied to minimising total capital cost and total CO₂ emissions simultaneously. It can also provide Pareto solutions which obtain total feasible solutions satisfying the peak thermal and power demands.

The chapter is organised as follows: Section 5.2 describes the design of a multi-fuel energy system. The three proposed techniques are developed in Section 5.3. Test runs using these techniques and comparisons with each other are carried out in Section 5.4. The final conclusions are presented in Section 5.5.

5.2 The proposed system model

This study is to know which of the renewable energy systems are best suited and which combinations of technologies are cost-effective in the new city context. It is also to establish an economically viable strategy of distributed energy systems for each district. Since Renewable Energy (RE) systems are intermittent and weather-dependent supply resources, it is crucial to identify the realistic efficiency of these systems. The aim of the project was to identify the best design and technical options in terms of energy performance and carbon emission reduction whilst adopting zero and low carbon technologies.

The study comprised the following:

- Implementing an analysis in context of the local climate and site of the city.
- Establishing the databases of the hourly demand profiles (i.e., heating, hot water and electricity) and supply systems specifications (e.g., PV cell, solar collector, wind turbines, heat pumps, CHP systems, etc.).
- Developing analysis tools for performance assessment and optimisation of zero and low carbon energy systems.
- Testing the performance of RE systems in the local climate context.
- Optimising analysis of zero and low carbon energy systems in terms of the capital cost and the carbon emission.

Figure 5.1 shows the outline plan for the new town ASAN in Korea, which is sub-divided into ten districts labelled 'A' to 'J', regarding the district's layout. It is assumed that each district installs an independent multi-fuel energy system which does not interfere with any other district and Figure 5.2 depicts the overview. It is first necessary to determine what the multi-fuel energy system should consist of. Efficient renewable energy systems should be considered in conjunction with the weather profile and the geographic location of ASAN. Due to the intermittency and high cost of renewable energy systems, fuel-based energy systems are dependable energy supply systems and able to meet the large scale energy demands of a city, therefore they should not be ignored. The feasibility of each renewable system's performance in this region is studied by MERIT using an hourly-based annual weather profile and more detail is shown in the Appendix. This feasibility study led to the selection of the multi-fuel system shown in Figure 5.3. The annual average efficiency of a wind turbine in this region is approximately 3% which is insufficient, therefore it was eliminated. The renewable systems consist of solar thermal collector (T1), geothermal heat pump (T2), air source heat pump (T3), fuel cells (GT2) and PV cell (G1), whilst the fuel-based energy systems consist of grid power (G2), natural gas boiler (T4) and natural gas CHP (GT1). It is assumed that the electricity input for T2 and T3 are supplied by G2, therefore their CO₂ factors are taken into consideration in G2 and their input is taken as a further electricity demand in G2. All system input parameters for the optimisation tool such as unit cost, CO₂ factor, Coefficient of

Performance (COP) and power/heat ratio, are surveyed and researched worldwide with their details shown in the Appendix. The CO₂ reduction ratio will be calculated for a conventional system such as a boiler for peak thermal demand and grid power for peak electrical demand. Unfeasible solutions will be discarded regarding the CO₂ reduction requirement and budget limitations.

Only 'H' district is considered in this work, and Table 5.1 shows its peak thermal and electrical demands which operate as constraints for the system. These also affect the maximum capacity for each system except for the PV cell and the solar thermal collector because it is assumed that they are installed on the roof. Therefore total roof areas were calculated to determine the maximum capacities of the PV cell and solar thermal collector.



Figure 5.1 Outline plan for ASAN new town

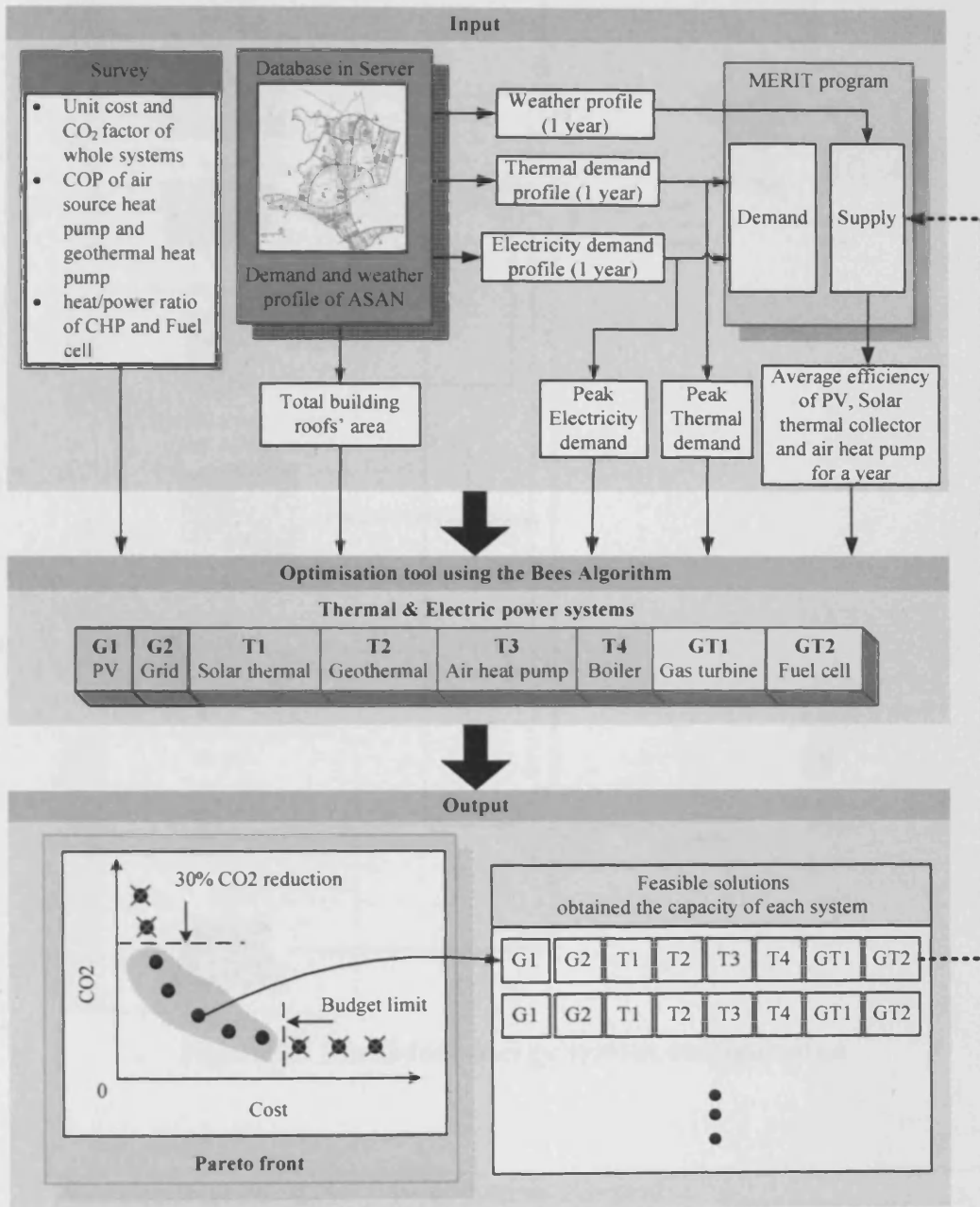


Figure 5.2 Overview of the procedure for this work

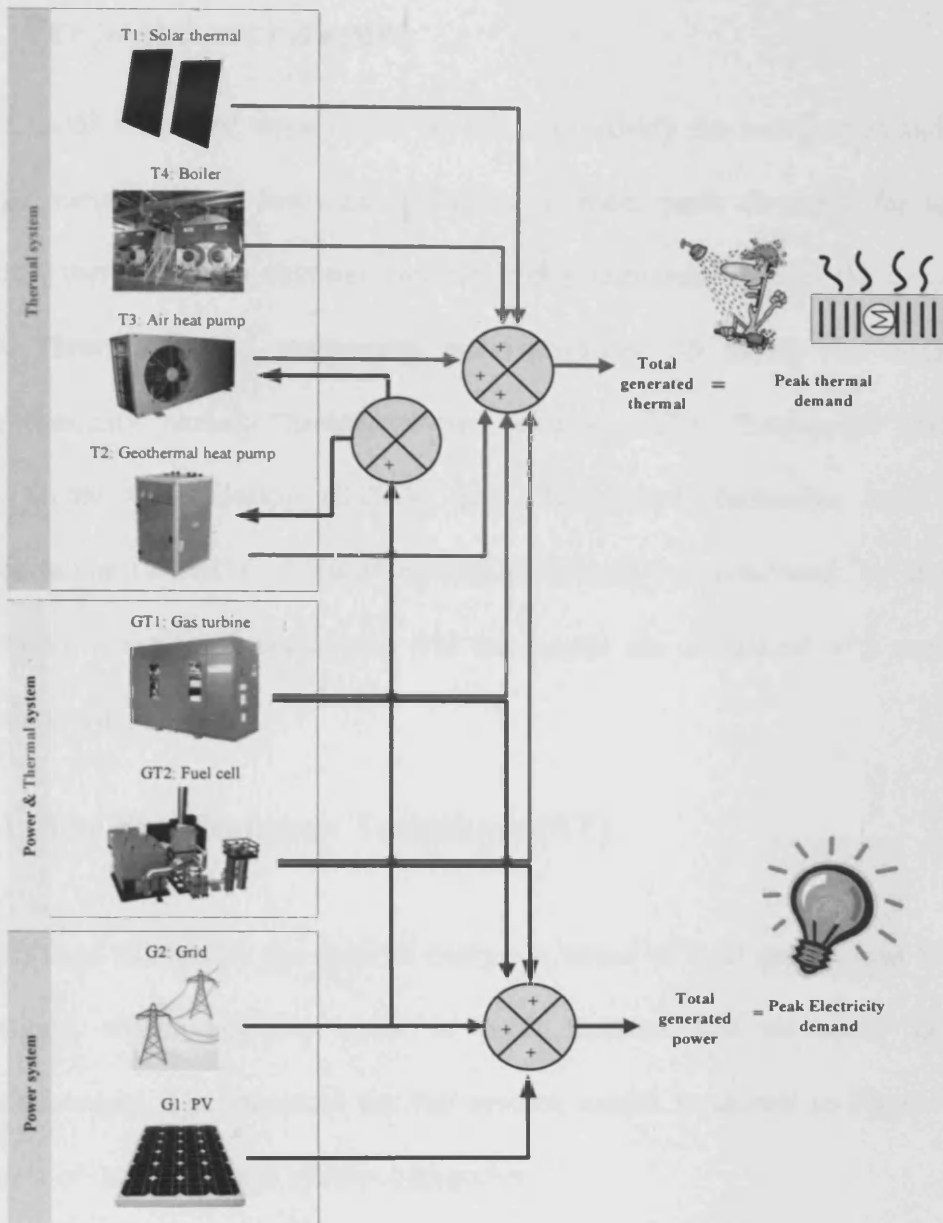


Figure 5.3 Multi-fuel energy system configuration

Peak electrical demand (kW/hr)	Peak thermal demand (kW/hr)	Total roof areas (m ²)
51,000	129,000	652,848

Table 5.1 Peak demands and total roof areas

5.3 Proposed techniques

One crucial factor for the optimal design is to satisfy the energy demands. The energy supply system must be sufficient to meet peak demands for security reasons, therefore peak thermal and electricity demands become the constraints used. Three proposed techniques are introduced to tackle the constraints simultaneously, namely ‘Simultaneous Technique (ST)’, ‘Sequential Technique with Single Optimisation (STSO)’ and ‘Sequential Technique with Multi-Optimisation (STMO)’. Pareto optimal solutions are generated by the Bees Algorithm using these techniques and the results are compared with each other and analysed.

5.3.1 The Simultaneous Technique (ST)

This is used to identify the optimal design in terms of total capital cost and CO₂ emissions, when they are equal to peak thermal and electricity demands simultaneously. The structure for the system model is shown in Figure 5.4. It consists of three different system categories:

- power systems: PV cell (G1) and grid power (G2)
- thermal systems: solar thermal collector (T1), geothermal heat pump (T2), air source heat pump (T3) and natural gas boiler (T4)
- combined heat & power systems: natural gas CHP (GT1) and fuel cell (GT2)

G1, G2, GT1 and GT2 generate sufficient electric power to meet the power balance constraint, whilst T1, T2, T3, T4, GT1 and GT2 produce thermal energy satisfying the thermal balance constraint. Electric power input for T2 and T3 is supplied from G2 and their CO₂ factors are considered with G2.

Figure 5.5 shows the flowchart of this approach using the Bees Algorithm. First of all, n numbers of scout bees are initialised, their fitness evaluated and then dominant bees are chosen as selected patch sites. The recruited bees (nm) are sent to these dominant sites to conduct neighbourhood searches around them and the remaining bees continue to conduct random searches until the criterion is met. All dominant sites are stored after every iteration in order to generate the Pareto front. Equations (5.1) to (5.19) show all relevant equations used for this technique.

The total capital cost objective is given in Eq. (5.1) and the total CO₂ emission objective by Eq. (5.2).

$$f_{cALL} = \sum_{i=1}^{N_G} (C_{Gi} \times UC_{Gi}) + \sum_{j=1}^{N_T} (C_{Tj} \times UC_{Tj}) + \sum_{k=1}^{N_{GT}} (C_{GTk} \times UC_{GTk}) \text{ (KRW)} \quad \text{Eq. (5.1)}$$

$$f_{eALL} = \sum_{i=1}^{N_G} (C_{Gi} \times CO2F_{Gi}) + \sum_{j=1}^{N_T} (C_{Tj} \times CO2F_{Tj}) + \sum_{k=1}^{N_{GT}} (C_{GTk} \times CO2F_{GTk}) \text{ (ton)} \quad \text{Eq. (5.2)}$$

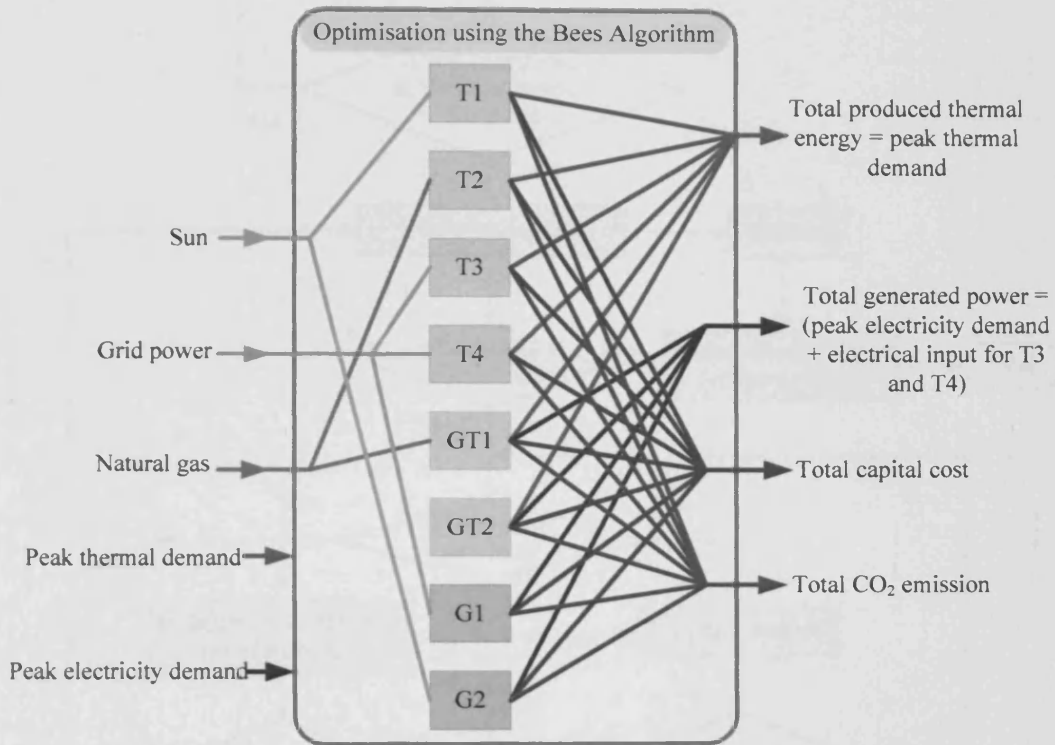


Figure 5.4 System structure of Simultaneous Technique (ST)

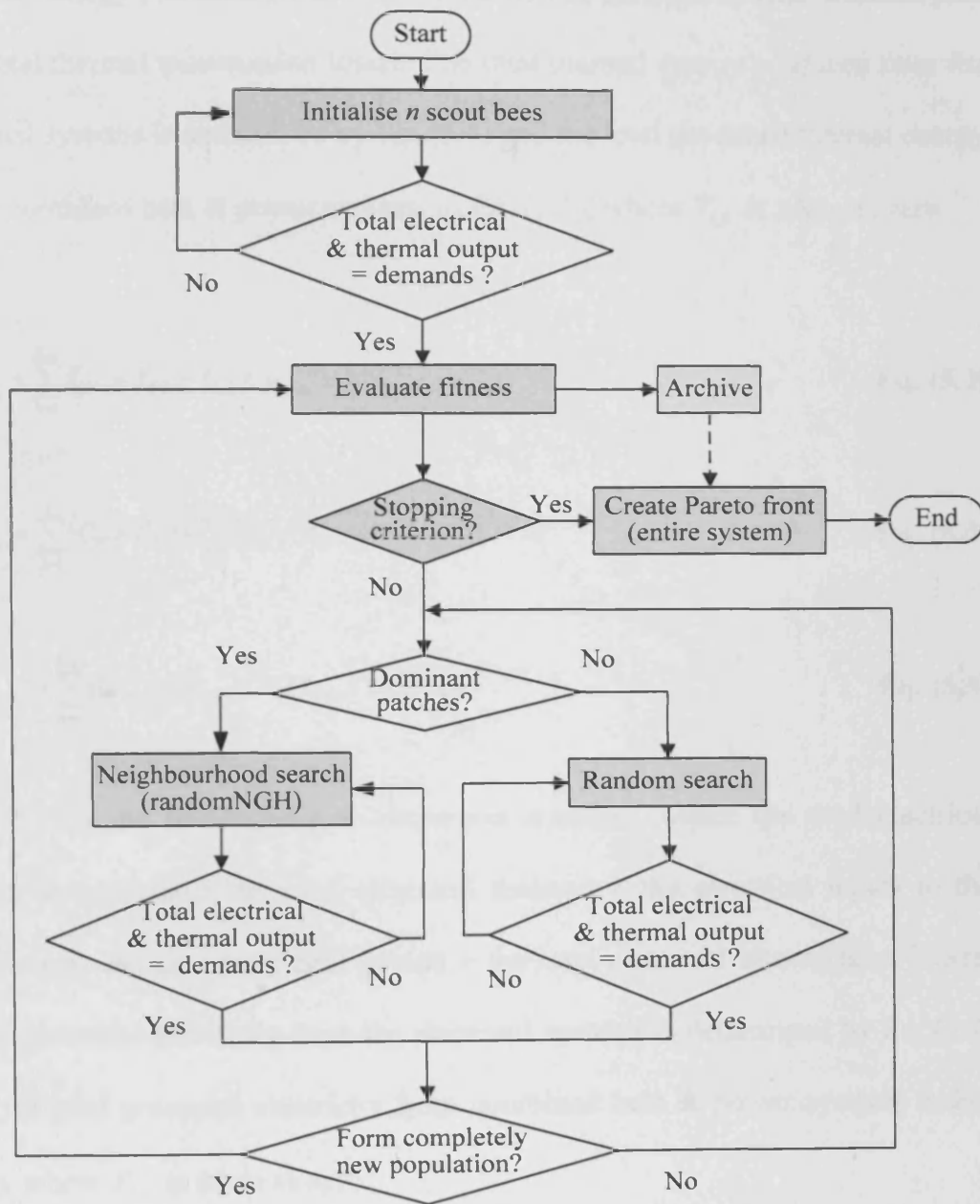


Figure 5.5 Flowchart of the Simultaneous Technique (ST) using the Bees Algorithm

Eq. (5.3) is the *thermal balance constraint* equation, where the total produced thermal energy produced should match the sum of the peak thermal demand plus the total thermal transmission losses. The total thermal energy produced from the thermal systems is determined by Eq. (5.4) and the total produced thermal energy from combined heat & power systems in Eq. (5.5), where T_{LT} is taken as zero.

$$\sum_{j=1}^{N_T} T_{Tj} + \sum_{k=1}^{N_{GTk}} T_{GTk} - T_{DT} - T_{LT} = 0 \quad \text{Eq. (5.3)}$$

$$\sum_{j=1}^{N_T} T_{Tj} = \sum_{j=1}^2 (C_{Tj} \times \beta_{Tj}) + \sum_{j=3}^{N_T} (C_{Tj} \times \beta_{Tj} \times COP_{Tj}) \quad (\text{MW}_{th}) \quad \text{Eq. (5.4)}$$

$$\sum_{k=1}^{N_{GTk}} T_{GTk} = \sum_{k=1}^{N_{GTk}} ((C_{GTk} \times \beta_{GTk}) / PHR_{GTk}) \quad (\text{MW}_{th}) \quad \text{Eq. (5.5)}$$

Eq. (5.6) is the *power balance constraint* equation, where the total electrical output should match the peak electrical demand + the electrical inputs to the geothermal and air source heat pumps + the total electrical transmission losses. Total generated electricity from the electrical systems is determined by Eq. (5.7) and the total generated electricity from combined heat & power systems in Eq. (5.8), where P_{LE} is taken as zero.

$$\sum_{i=1}^{N_G} P_{Gi} + \sum_{k=1}^{N_{GTk}} P_{GTk} - (P_{DE} + RP_{T3} + RP_{T4}) - P_{LE} = 0 \quad \text{Eq. (5.6)}$$

$$\sum_{i=1}^{N_G} P_{Gi} = \sum_{i=1}^{N_G} (C_{Gi} \times \alpha_{Gi}) \text{ (MW}_e\text{)} \quad \text{Eq. (5.7)}$$

$$\sum_{k=1}^{N_{GT}} P_{GTk} = \sum_{k=0}^{N_{GT}} (C_{GTk} \times \alpha_{GTk}) \text{ (MW}_e\text{)} \quad \text{Eq. (5.8)}$$

Those maximum and minimum limits for each system are shown in Table 5.2. The maximum capacity boundary of the thermal systems, excepting the solar collector, is determined by the peak thermal demand. For the electrical systems the boundary is determined by the peak electrical demand with the exception of the PV cell. The peak thermal demand in Korea is typically much greater than the peak electrical demand, thus peak thermal demand is given higher priority here. Determination of the maximum capacity boundary for the CHP and fuel cell is dependant on the peak thermal demand.

System	Limits	Equations of minimum and maximum capacity boundary	
Solar thermal collector (T1)	$C_{T1\min} \leq C_{T1} \leq C_{T1\max}$ (MW _{th})	$C_{T1\min} = 0$ $C_{T1\max} = \frac{tba}{rua_{sic}} \times allowance_{T1}$	Eq. (5.9)
Boiler (T2)	$C_{T2\min} \leq C_{T2} \leq C_{T2\max}$ (MW _{th})	$C_{T2\min} = 0$ $C_{T2\max} = \frac{T_{DT}}{\beta_{T2}}$	Eq. (5.10)
Geothermal heat pump (T3)	$C_{T3\min} \leq C_{T3} \leq C_{T3\max}$ (MW _{th})	$C_{T3\min} = 0$ $C_{T3\max} = \frac{T_{DT}}{(\beta_{T3} \times COP_{T3})} \times allowance_{T3}$	Eq. (5.11)
Air source heat pump (T4)	$C_{T4\min} \leq C_{T4} \leq C_{T4\max}$ (MW _{th})	$C_{T4\min} = 0$ $C_{T4\max} = \frac{T_{DT}}{(\beta_{T4} \times COP_{T4})} \times allowance_{T4}$	Eq. (5.12)
CHP (GT1)	$C_{GT1\min} \leq C_{GT1} \leq C_{GT1\max}$ (MW _{th})	$C_{GT1\min} = 0$ $C_{GT1\max} = \frac{(T_{DT} \times PHR_{GT1})}{\beta_{GT1}}$	Eq. (5.13)
Fuel cell (GT2)	$C_{GT2\min} \leq C_{GT2} \leq C_{GT2\max}$ (MW _{th})	$C_{GT2\min} = 0$ $C_{GT2\max} = \frac{(T_{DT} \times PHR_{GT2})}{\beta_{GT2}} \times allowance_{GT2}$	Eq. (5.14)
Grid (G1)	$C_{G1\min} \leq C_{G1} \leq C_{G1\max}$ (MW _e)	$C_{G1\min} = \frac{RP_{T3} + RP_{T4}}{\alpha_{G1}}$ $C_{G1\max} = \frac{(P_{DE} + RP_{T3\max} + RP_{T4\max})}{\alpha_{G1}}$	Eq. (5.15)
PV (G2)	$C_{G2\min} \leq C_{G2} \leq C_{G2\max}$ (MW _e)	$C_{G2\min} = 0$ $C_{G2\max} = \frac{tba}{rua_{pv}} \times allowance_{G2}$	Eq. (5.16)

Table 5.2 Limit of each system

The Multi-Objective equation for the multi-fuel system is formulated as in Eq.

(5.17).

$$\text{Minimise} [f_{cALL}, f_{eALL}] \quad \text{Eq. (5.17)}$$

subject to

$$\text{Thermal balance: } \sum_{j=1}^{N_T} T_{Tj} + \sum_{k=1}^{N_{GT}} T_{GTk} - T_{DT} - T_{LT} = 0$$

$$\text{Power balance: } \sum_{i=1}^{N_G} P_{Gi} + \sum_{k=1}^{N_{GT}} P_{GTk} - (P_{DE} + RP_{T3} + RP_{T4}) - P_{LE} = 0$$

Limit:

$$C_{Gi \min} \leq C_{Gi} \leq C_{Gi \max} \quad (i = 1, \dots, N_G)$$

$$C_{Tj \min} \leq C_{Tj} \leq C_{Tj \max} \quad (j = 1, \dots, N_T)$$

$$C_{GTk \min} \leq C_{GTk} \leq C_{GTk \max} \quad (k = 1, \dots, N_{GT})$$

To calculate the percentage of CO₂ reduction (Eq. 5.18) for each Pareto solution (p) from the proposed multi-fuel energy system, the absolute CO₂ quantity Eq. (5.19) is calculated for the conventional systems, i.e., a natural gas boiler provides peak thermal demand and grid power peak electrical demand.

$$CO2RR_p = 100 - \left(100 \times \frac{E_{allp}}{ACO2} \right) (\%) \quad \text{Eq. (5.18)}$$

$$ACO2 = \left(\frac{T_{DT}}{\beta_{T2}} \times CO2F_{T2} \right) + \left(\frac{P_{DE}}{\alpha_{G1}} \times CO2F_{G1} \right) \quad (\text{ton/hr}) \quad \text{Eq. (5.19)}$$

5.3.2 The Sequential Technique with Multi-Optimisation (STMO)

This technique solves the problem by handling the thermal balance and power balance constraint sequentially. Figures 5.6 and 5.7 show its structure and concept, and share the same identifiers ① → ④ and (a) → (d). The Pareto (a) front of thermal systems is generated first and this can be categorised into two groups: partial Pareto (a₁) and (a₂). The (a₁) group of solutions do not need any additional electrical systems because they are satisfying the two demands simultaneously. However, the (a₂) group of solutions are not able to meet the electrical demand, thus they need additional electrical suppliers. While the thermal and combined heat & power systems from the (a₂) group of solutions hold their capacities Figure 5.6 (3), the additional electrical capacity necessary is determined by the total of the electrical demand plus the geothermal and air source heat pump electrical inputs. When all the individual components of the system have been identified its fitness in terms of the total capital cost and CO₂ emissions are calculated. Only then are dominant patch sites selected for neighbourhood searches and the remaining bees conduct random searches until the criterion is met. To reduce the number of evaluations, the second criterion is one-tenth of the first one. Each individual solution of system (3) is optimised by the Bees Algorithm respectively, therefore multi-optimisation is required and

multi-Pareto fronts of entire system are also generated as in Figure 5.7 (iii). At the end, the Pareto © front is generated from the systems (Figure 5.7 iv). Finally, the Pareto ④ front is generated by combining the partial Pareto ① front and the Pareto © front. Figure 5.8 illustrates its flowchart and Eqs. (5.20 to 5.24) show all relevant equations used in this technique.

5.3.2.1 Equations for the Pareto ② front

The Pareto ② front of the thermal systems was evaluated by using Eqs. (5.20 to 5.23). Its objective of total capital cost is Eq. (5.20) and its CO₂ emissions objective is Eq. (5.21). Although grid power (G1) is one of the electrical systems, the assumption is made that the electrical input for the geothermal and air source heat pumps is supplied from G1. Consequently, Eqs. (5.20 to 5.22) include G1. The thermal balance constraint is the same as that of the ST Eqs. (5.3 → 5.5), and the limitations for the thermal and heat & power systems is also the same as the ST Eqs. (5.9 → 5.14). However, the capacity of G1 is dependant on the electrical inputs required for both the geothermal and air source heat pumps (Eq. 5.22) and Eq. (5.23) is the Multi-Objective equation.

$$f_{cTH} = (C_{G1} \times UC_{G1}) + \sum_{j=1}^{N_T} (C_{Tj} \times UC_{Tj}) + \sum_{k=1}^{N_{GT}} (C_{GTk} \times UC_{GTk}) \text{ (KRW)} \quad \text{Eq. (5.20)}$$

$$f_{eTH} = (C_{G1} \times CO2F_{G1}) + \sum_{j=1}^{N_T} (C_{Tj} \times CO2F_{Tj}) + \sum_{k=1}^{N_{GT}} (C_{GTk} \times CO2F_{GTk}) \text{ (ton/hr)} \quad \text{Eq. (5.21)}$$

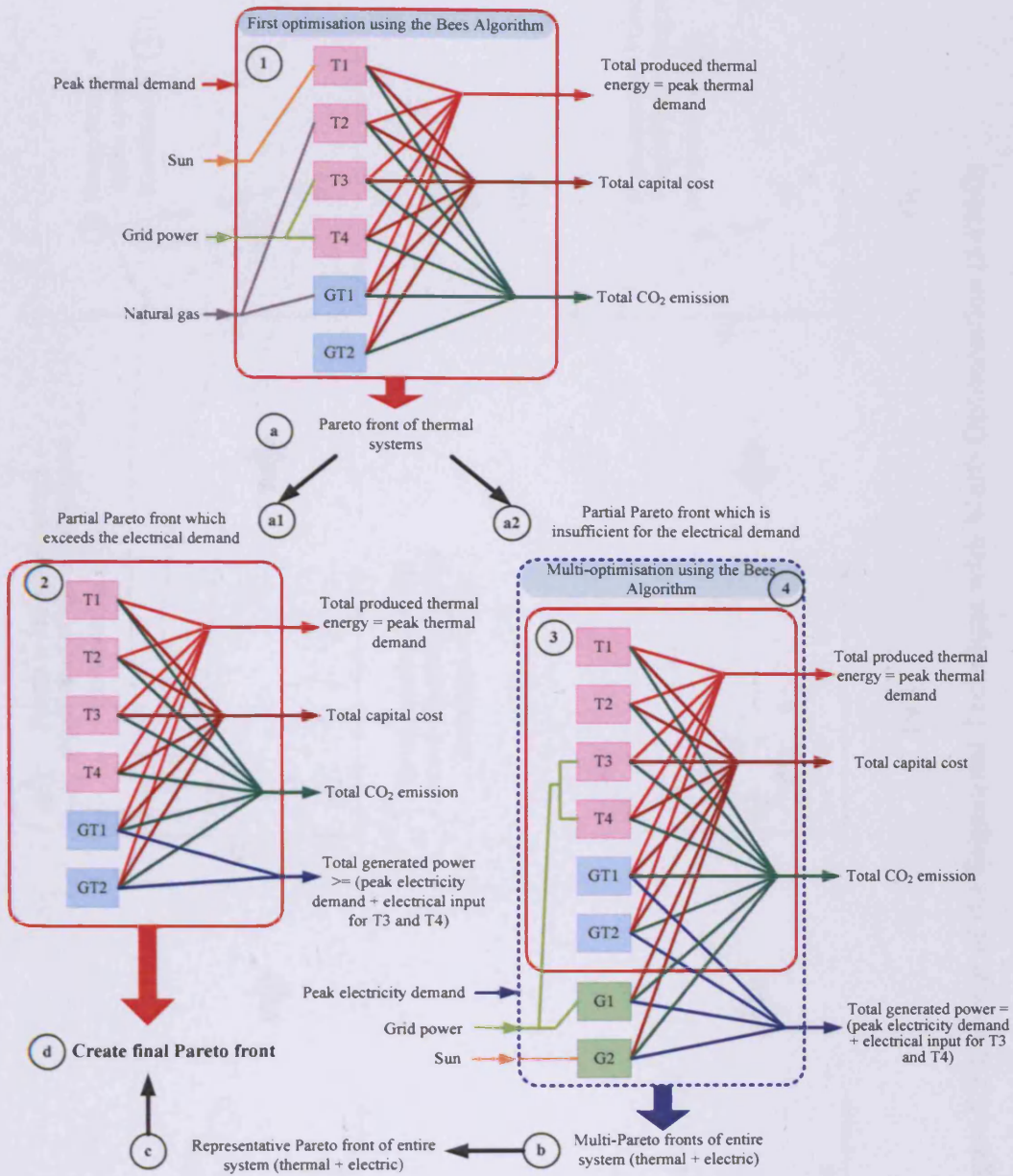


Figure 5.6 Structure of the Sequential Technique with Multi-Optimisation (STMO)

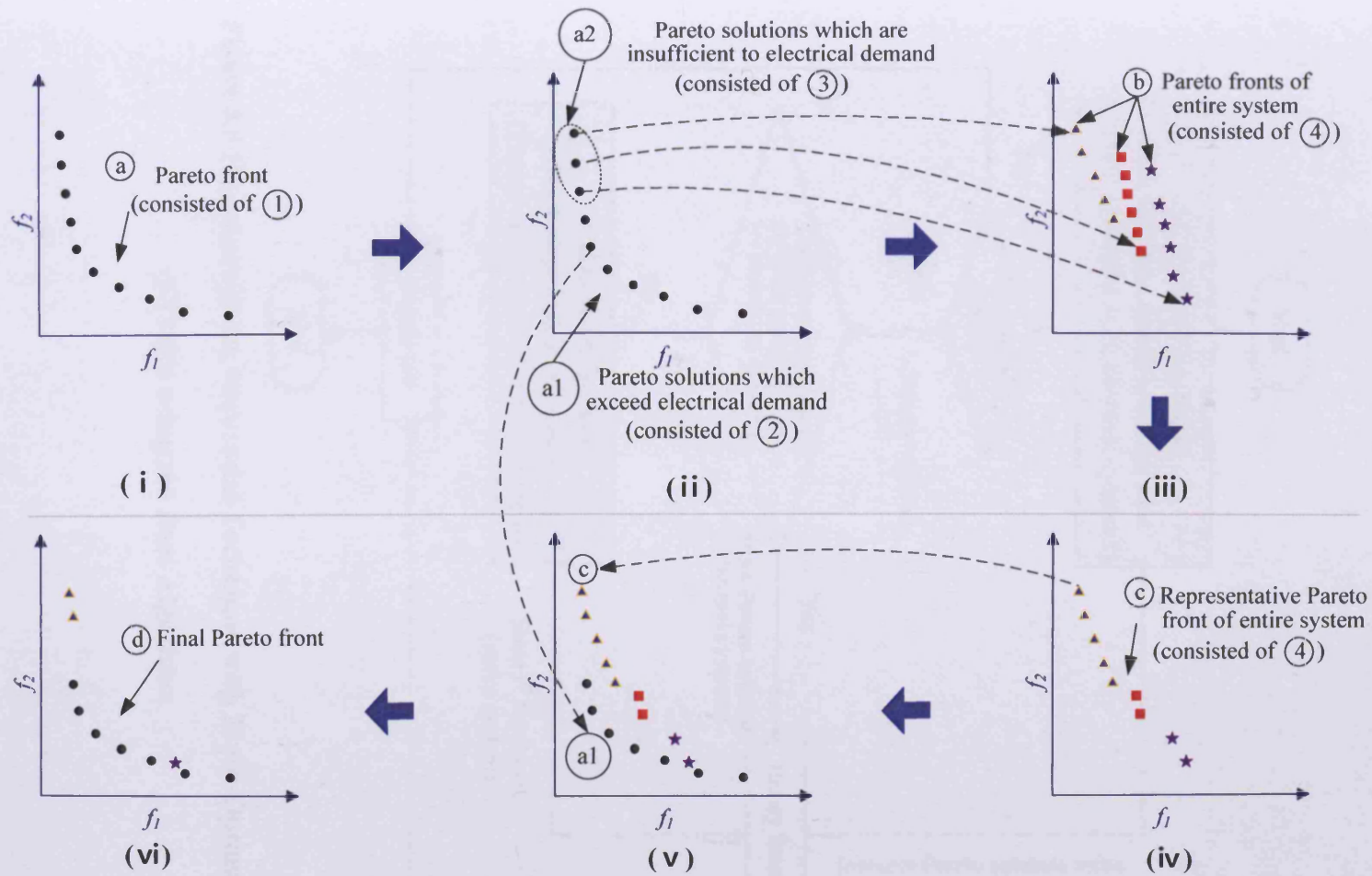


Figure 5.7 Concept of the Sequential Technique with Multi-Optimisation (STMO)

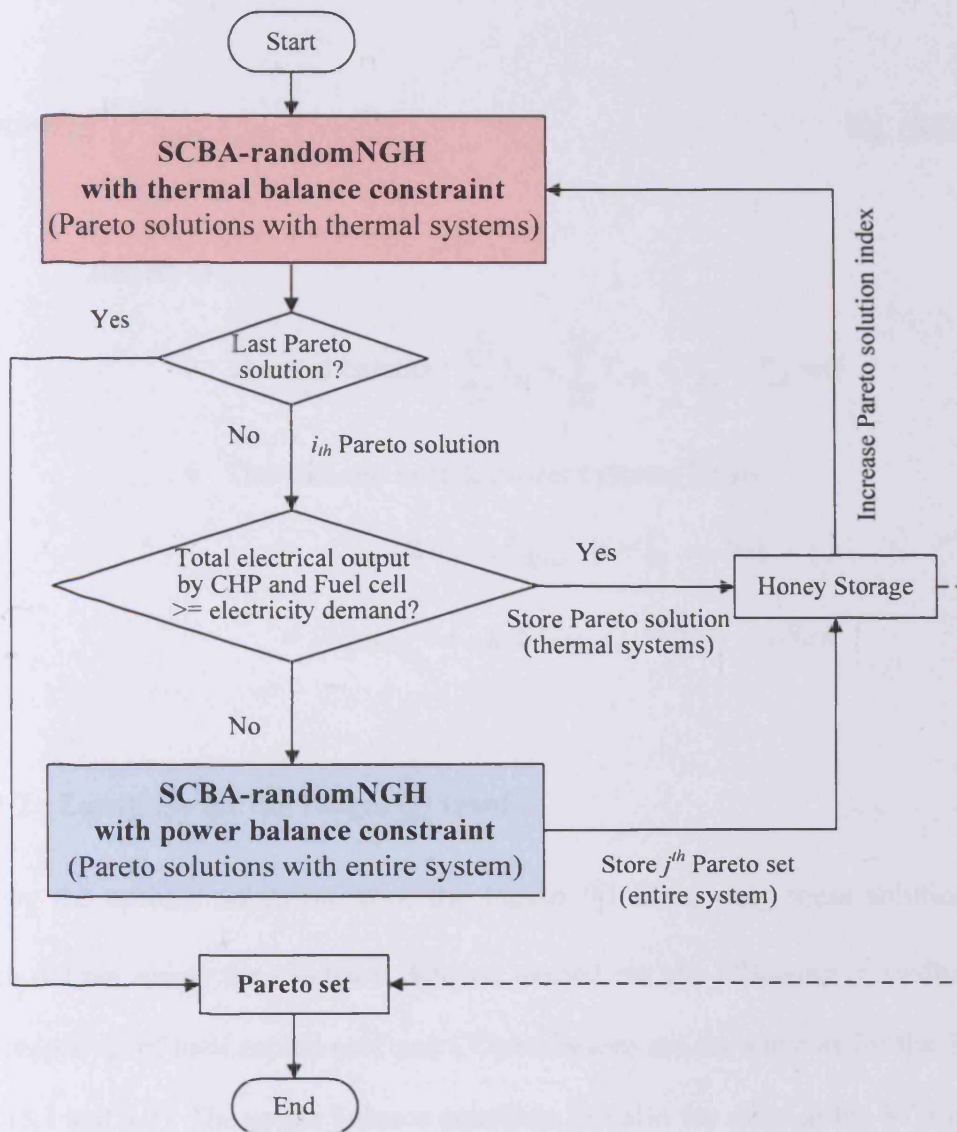


Figure 5.8 Flowchart of the Sequential Technique with Multi-Optimisation (STMO) using the Bees Algorithm

$$C_{G1\min} = \frac{RP_{T3} + RP_{T4}}{\alpha_{G1}} \quad \text{Eq. (5.22)}$$

$$\text{Minimise } [f_{cTH}, f_{eTH}] \quad \text{Eq. (5.23)}$$

subject to

- Thermal balance: $\sum_{j=1}^{N_T} T_{Tj} + \sum_{k=1}^{N_{GT}} T_{GTk} - T_{DT} - T_{LT} = 0$
- Thermal and heat & power systems limits
 - $C_{Tj\min} \leq C_{Tj} \leq C_{Tj\max} \quad (j = 1, \dots, N_T)$
 - $C_{GTk\min} \leq C_{GTk} \leq C_{GTk\max} \quad (k = 1, \dots, N_{GT})$

5.3.2.2 Equations for the Pareto ④ front

Among the optimal solutions from the Pareto ① front, only these solutions which did not satisfy the electrical demand carried out the following procedure. The objectives of total capital cost and CO₂ emissions are the same as for the ST Eqs. (5.1 and 5.2). The power balance equations are also the same as the ST Eqs. (5.6 → 5.8) and the electrical systems limits are also the same as the ST Eqs. (5.15 and 5.16). As the thermal and heat & power systems satisfy the thermal balance constraint, and their capacities are defined from the Pareto ① front, there is no need to check the thermal balance constraint and limitations again. Hence,

the Multi-Objective form of this technique is formulated as Eq. (5.24). The equation for the CO₂ reduction is also the same as that of the ST Eq. (5.18).

$$\text{Minimise}[f_{cALL}, f_{eALL}] \quad \text{Eq. (5.24)}$$

subject to

- Power balance: $\sum_{i=1}^{N_G} P_{Gi} + \sum_{k=1}^{N_{GT}} P_{GTk} - (P_{DE} + RP_{T3} + RP_{T4}) - P_{LE} = 0$
- Limitation:
 - $C_{Gi\min} \leq C_{Gi} \leq C_{Gi\max} \quad (i = 1, \dots, N_G)$

5.3.3 The Sequential Technique with Single Optimisation (STSO)

This technique also solves the problem sequentially just as the previous one (section 5.3.2), but this technique optimises the designed system only once, and solely for thermal systems only. Figures 5.9 and 5.10 show the system structure and concept of this technique. Most of the procedure is similar to the previous technique with the exception of (iii) in Figure 5.10. Instead of generating individual Pareto fronts as in STMO, this technique determines the capacity of the electrical systems within their limits to meet peak demand. This capacity is determined from the difference between the peak requirement plus both heat

pump inputs, minus the total of the outputs of the other systems, i.e., CHP and Fuel cells. The number of candidate solutions (b) in Figures 5.9 and 5.10, depends on the number of Pareto (a) solutions with lower electrical demand values. For example, there are three Pareto (a) solutions with lower electricity demand in Figure 5.10 (ii). Therefore only three candidate solutions can be generated Figure 5.10 (iii). The final Pareto (c) front is identified by combining partial Pareto (a¹) front and (b) candidate solutions. Figure 5.11 shows its flowchart.

All equations for thermal systems Pareto front of and entire systems are exactly the same as sections 5.3.2.1 and 5.3.2.2 with the exception of the Multi-Objective form in section 5.3.2.2. In this technique, optimisation for entire systems (system (4) in Figure 5.9) is not required, thus the Multi-Objective equation is redundant.

5.3.4 System parameters

Table 5.3 shows the parameters regarding the multi-fuel energy systems with further details shown in the Appendix.

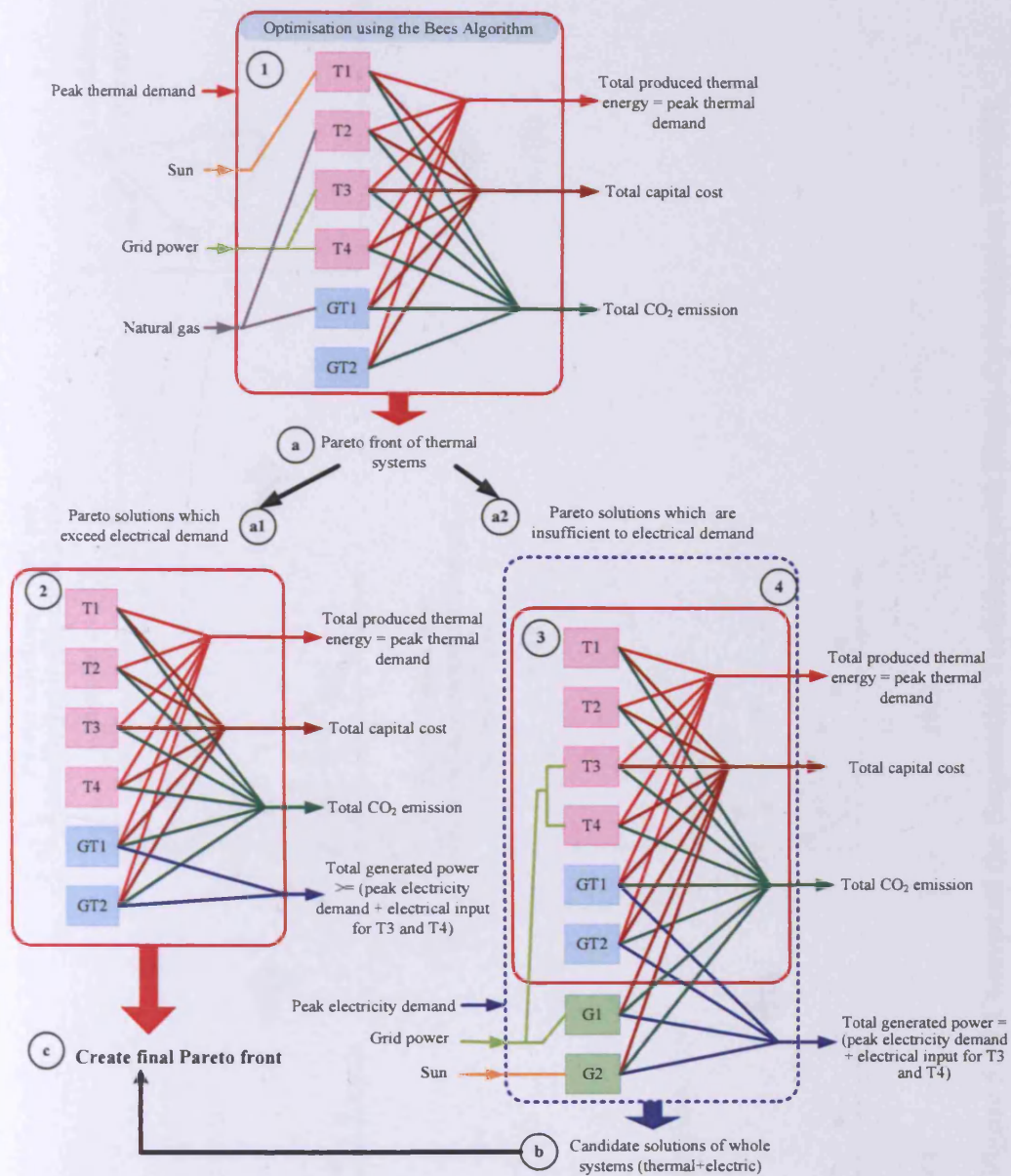


Figure 5.9 Structure of the Sequential Technique with Single Optimisation (STSO)

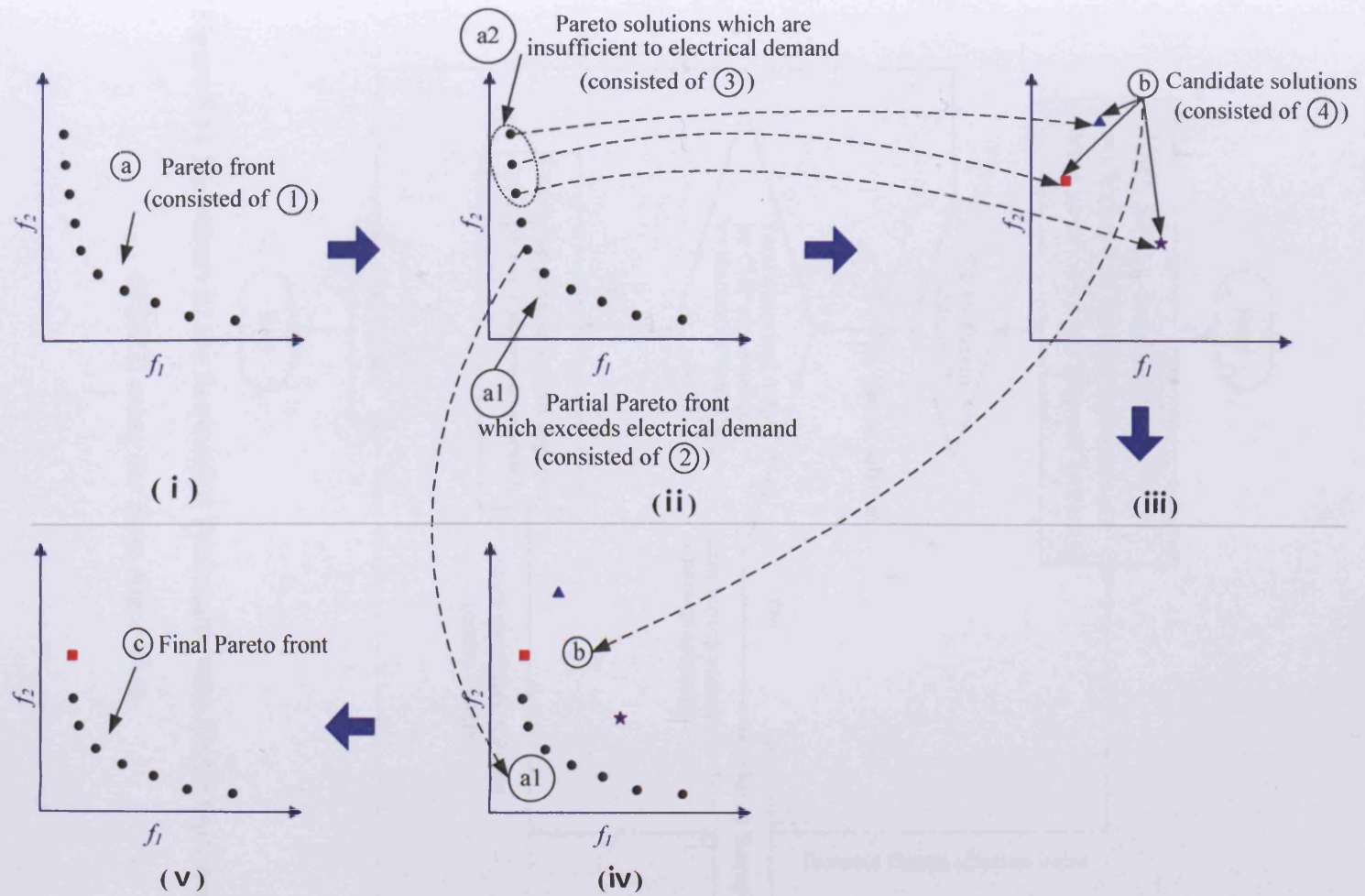


Figure 5.10 Concept of the Sequential Technique with Single Optimisation (STSO)

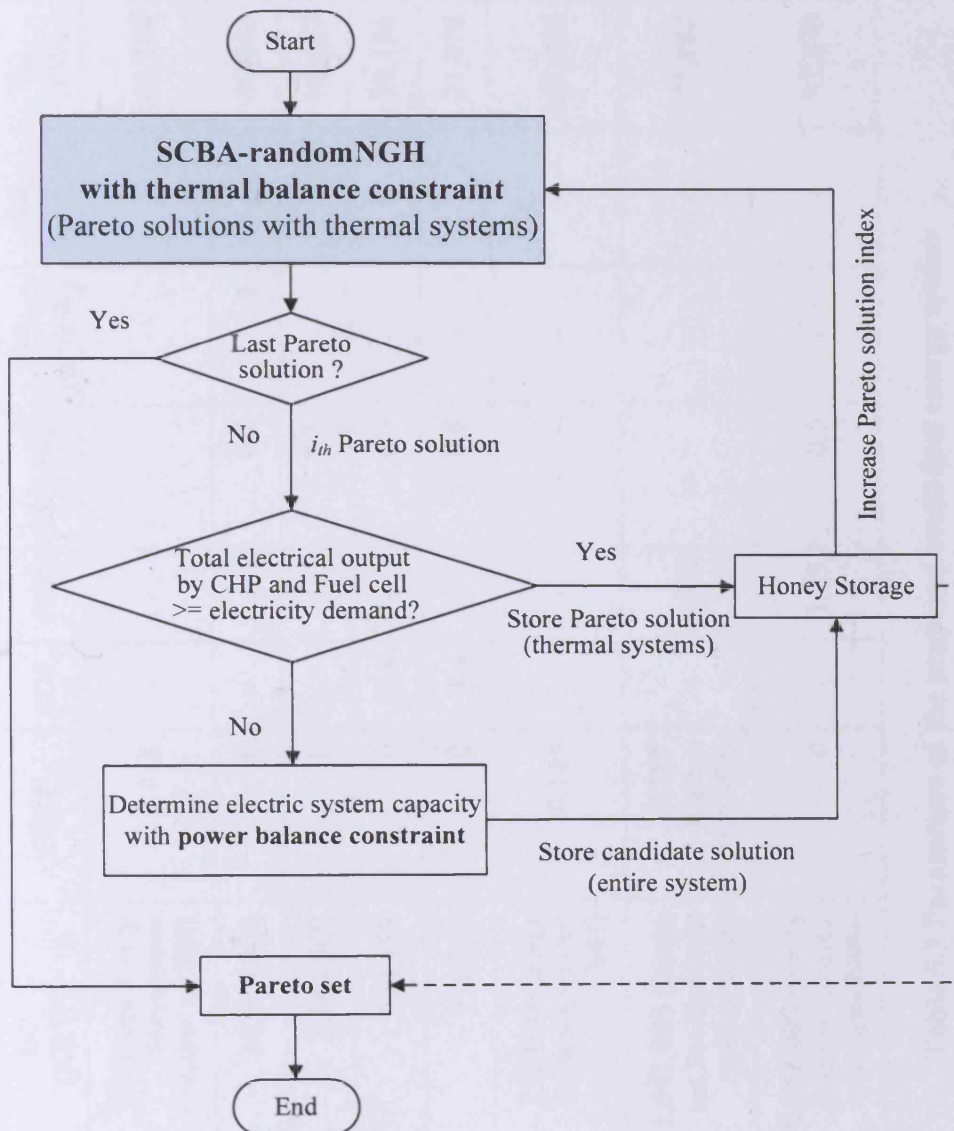


Figure 5.11 Flowchart of the Sequential Technique with Single Optimisation (STSO) using the Bees Algorithm

	β	α	UC (KRW)	CO_2F	COP	PHR	$allownace$	rua_{pv} (m ² /kW)	rua_{stc} (m ² /kW)	C_{max} (kW)	ACO_2 (ton/hr)
G1	-	1.0	72 (Korea Energy Management Corporation 2009)	0.448	-	-	-	-	-	101,264	50.92
G2	-	0.15	10,578,000	0	-	-	0.5	8	-	40,803	
T1	0.12	-	3,144,500	0	-	-	0.5	-	3.31	98,617	
T2	0.65	-	1,270,158	0	3.4	-	0.5	-	-	29,186	
T3	0.9	-	916,090	0	3.4	-	0.5	-	-	21,078	
T4	0.85	-	150,000 (Energy Technologies Unit 2007)	0.185	-	-	-	-	-	151,765	
GT1	0.85	1.0	1,643,095 (Energy and Environmental Analysis 2008b)	0.185 (Carbon Trust 2008)	-	0.5	-	-	-	75,882	
GT2	0.98	1.0	6,852,000 (Energy and Environmental Analysis 2008a)	0	-	1.25	0.5	-	-	82,270	

Table 5.3 Parameters of the proposed multi-fuel energy system

5.4 Results

The optimal capacities of the power and thermal systems were optimised by the Bees Algorithm with the randomNGH method. For each of the three techniques, two different settings were tested, regarding the number of scout bees and iterations as in Table 5.4. Thus, a total of six tests were performed. To obtain the average performance of the algorithm, 10 independent runs were carried out. At the end of each test, the performances of the various algorithm configurations were evaluated, using the two extreme solutions for minimum capital cost and CO₂ emissions from the Pareto set. The Pareto fronts, the number of solutions, number of evaluations and program running time are compared with each other.

5.4.1 Comparison of two extreme solutions

To compare the two extreme solutions, each average value of 10 independent runs is taken as shown in Table 5.5. For statistical analysis, the *SPSS* program is also used in this chapter and the *ANOVA* (for data with normal distribution and homogeneous variances), the *Kruskal-Wallis* (for non-parametric data) test and the *t-test* are all carried out for a 5% alpha level of significance to compare the mean or median values. Two extreme solutions will be compared, the first regarding techniques and the second with regard to settings.

Name of parameters	setting1	setting2
<i>n</i> : number of scout bees	50	100
<i>nm</i> : number of recruited bees allocated to non-dominated patch sites	10	10
<i>ngh</i> : patch size	50	50
<i>itr</i> : number of iterations	50	100
Accuracy (MW)	0.1	0.1

Table 5.4 Parameter setting for the Bees Algorithm

Technique	setting	f_c	$Corr.f_e$	$Corr.f_c$	f_e	$CO2RR_c$	$CO2RR_e$
ST	setting1	89,940	41.39	726,258	13.72	18.72	73.05
	setting2	67,072	42.56	662,906	13.19	16.43	74.10
STMO	setting1	62,562	40.53	719,262	6.95	20.42	86.35
	setting2	48,079	44.85	843,956	6.97	11.94	86.31
STSO	setting1	88,774	38.60	792,577	7.27	24.21	85.73
	setting2	81,324	39.88	830,427	7.14	21.69	85.97

f_c : the average of the best cost (MKRW)

$Corr.f_e$: the average of emissions corresponding to f_c (ton)

$Corr.f_c$: the average of cost corresponding to f_e (MKRW)

f_e : the average of the best emissions (ton)

$CO2RR_c$: the average of CO₂ reduction rate for f_c (%)

$CO2RR_e$: the average of CO₂ reduction rate for f_e (%)

Table 5.5 The average of two extreme solutions from 10 independent runs

5.4.1.1 Effects of techniques

Before comparing the mean or the median values, all required assumptions (such as normal distribution and homogeneous variances) of data sets were checked. The *SPSS* results show all data sets (f_c : best cost, $Corr.f_e$: emission corresponding to f_c , $Corr.f_c$: cost corresponding to f_e and f_e : best emission) from 6 different tests are satisfied with the assumption of normal distribution. They are also satisfied with the assumption of homogeneous variances except for the f_c data set in setting2. Therefore, a *Kruskal-Wallis* test was carried out for this f_c data set to compare its median values and the other data sets were tested by *ANOVA* test to compare their mean values.

5.4.1.1.1 Setting1

According to the *ANOVA* test as shown in Table 5.6 (a), f_c and f_e are significantly different, but $Corr.f_e$ and $Corr.f_c$ have no significant difference. A *Bonferroni* test was carried out to identify the most appropriate technique for the best mean values of f_c and f_e as shown in Table 5.6 (b). There is no significant difference between technique1 (ST) and technique3 (STSO) in finding the best cost (f_c). However, technique2 (STMO) is significantly different to the other two and the mean values from technique2 are much lower than for the others. Technique2 outperforms techniques 1 and 3 in finding the best cost. The best emissions (f_e) from technique1 is significantly different to techniques 2 and 3, but they are not significantly different to each other. The mean values from techniques 2 and 3

are much lower than for technique1, therefore they outperform technique1 regarding finding the best emission.

5.4.1.1.2 Setting2

Two different tests were carried out in setting2. First of all, an *ANOVA* test was carried out to compare the mean values of $Corr.f_e$, $Corr.f_c$ and f_e . The results in Table 5.7 (a) show the mean values of $Corr.f_e$ have no significant differences between the techniques, but the mean values of $Corr.f_c$ and f_e do have significant differences. To find out which one is significantly different, a *Bonferroni* test was also carried out and Table 5.7 (b) shows the results. For finding $Corr.f_c$, technique1 outperforms techniques 2 and 3, but for f_e , techniques 2 and 3 outperform technique1 in finding the best emission. For the best cost (f_c), there is a significant difference between the techniques according to the *Kruskal-Wallis* test shown in Table 5.7 (a). Therefore a box plot was drawn as Figure 5.12. Technique3 is shown to be significantly different to techniques 1 and 2, but there is no significant difference between techniques 1 and 2 which both outperform technique3.

5.4.1.2 Effect of settings

To check the effect of a setting, a *t-test* was carried out and Table 5.8 shows there is no significant difference for all data sets (f_c , $Corr.f_e$, $Corr.f_c$ and f_e), with ST. For the best cost (f_c) in both sequential techniques, there is also no significant differences, but for the best emissions (f_e), there is a significant difference with

setting	f_c	$Corr.f_e$	$Corr.f_c$	f_e
setting1	\neq	\equiv	\equiv	\neq

(a) Result by ANOVA test

Dependent Variable	(I) technique	(J) technique	Mean Difference (I-J)
f_c	1 (ST)	2	27378.600*
		3	1166.800
	2 (STMO)	1	-27378.600*
		3	-26211.800*
	3 (STSO)	1	-1166.800
		2	26211.800*
f_e	1 (ST)	2	6.77100*
		3	6.45300*
	2 (STMO)	1	-6.77100*
		3	-.31800
	3 (STSO)	1	-6.45300*
		2	.31800

(b) Result by Bonferroni test

\equiv not significantly different

\neq significantly different

*. The mean difference is significant at the 0.05 level.

Table 5.6 Statistical result in setting1

setting	f_c	$Corr.f_e$	$Corr.f_c$	f_e
setting2	\neq	\equiv	\neq	\neq

(a) Result by ANOVA and Kruskal-Wallis tests

Dependent Variable	(I) technique	(J) technique	Mean Difference (I-J)
$Corr.f_c$	1 (ST)	2 (STMO)	-181049.400*
		3 (STSO)	-167520.700*
	2 (STMO)	1 (ST)	181049.400*
		3 (STSO)	13528.700
	3 (STSO)	1 (ST)	167520.700*
		2 (STMO)	-13528.700
f_e	1 (ST)	2 (STMO)	6.22000*
		3 (STSO)	6.04700*
	2 (STMO)	1 (ST)	-6.22000*
		3 (STSO)	-.17300
	3 (STSO)	1 (ST)	-6.04700*
		2 (STMO)	.17300

(b) Result by Bonferroni test

\equiv not significantly different

\neq significantly different

*. The mean difference is significant at the 0.05 level.

Table 5.7 Statistical results in setting2

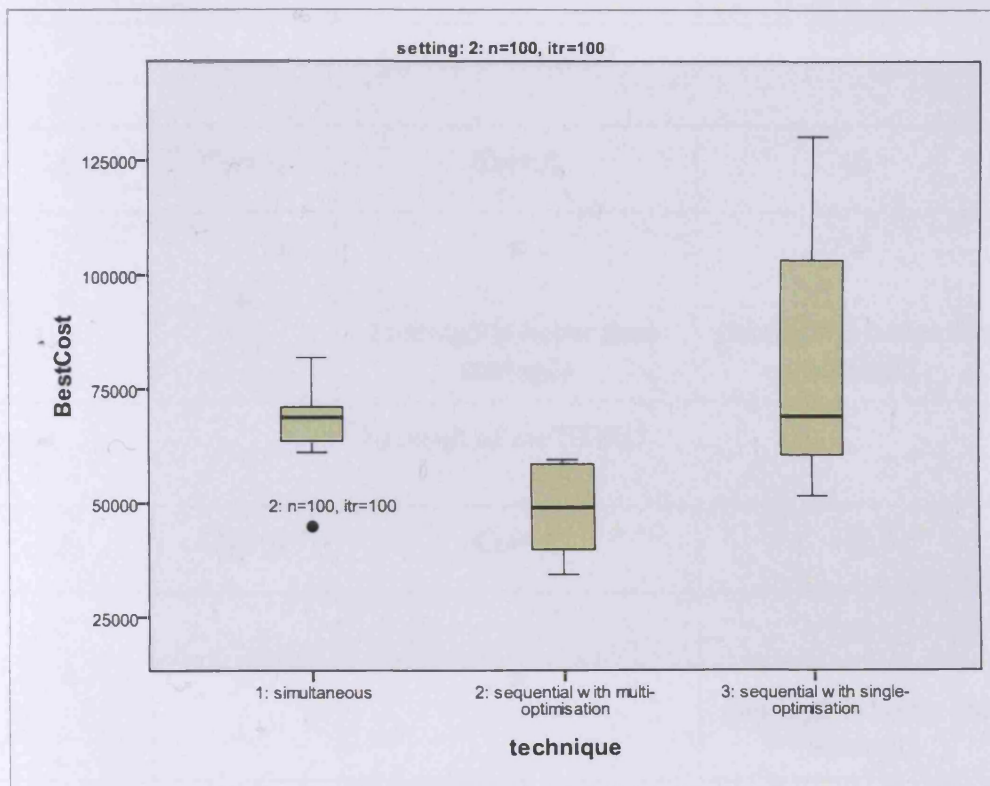


Figure 5.12 Box plot of the best cost (f_c) regarding techniques in setting2

f_c	$Corr.f_e$	$Corr.f_c$	f_e
≡	≡	≡	≡

(a) result of the ST

f_c	$Corr.f_e$	$Corr.f_c$	f_e
≡	≡	≠ (setting1 is better than setting2)	≠ (setting1 is better than setting2)

(b) result of the STMO

f_c	$Corr.f_e$	$Corr.f_c$	f_e
≡	≡	≡	≠ (setting2 is better than setting1)

(c) result of the STSO

≡ not significantly different
 ≠ significantly different

Table 5.8 Statistical result from *t-test* regarding setting

regard to setting. Setting1 has a better result than setting2 for the STMO, but setting2 result is better than setting1 in the STSO.

5.4.2 Comparison of Pareto fronts

Among 10 independent program runs from each test, one test result is arbitrarily selected to compare Pareto fronts. Figure 5.13 shows two Pareto fronts from setting1 (◆) and setting2 (■) and most of the Pareto solutions of setting2 dominate those of setting1. This confirms that setting2 outperforms setting1. Solutions near each end of the Pareto front and their near turning points as shown in Figure 5.13 were taken to construct the charts, Figures 5.14 and 5.15 depicting the individual proportions of the energy system making up the multi-fuel energy system with the ST. When the CO₂ reduction is low Figures 5.14 and 5.15 (a), the boiler and grid power provide the bulk of the energy (over 75%) to the system where both their inputs exceed 50MW. When the CO₂ reduction is around 60%, as Figures 5.14 and 5.15 (b) the CHP is the major energy provider. However, when the CO₂ reduction exceeds 70% as Figures 5.14 and 5.15 (c), the solar thermal collector entirely contributes almost as much as the others combined (i.e., \cong 50%).

Figure 5.16 shows two Pareto fronts from setting1 (●) and setting2 (▲) with the STMO and their shapes resemble the letter 'L'. They look similar but the Pareto

front of setting2 is better, especially the left side from the tuning point. Figures 5.17 and 5.18 show the individual fuels comprising the multi-fuel energy system with the STMO. When the CO₂ reduction is lower than 75% Figures 5.17 and 5.18 (a; b), its pattern is similar to that Figures 5.14 and 5.15 (a; b) in the ST. However, when the CO₂ reduction is high (around 85%) as Figures 5.14 and 5.15 (c), the capacity of the fuel cell is much greater, being more than double the capacity of the ST.

Figure 5.19 depicts two Pareto fronts with the STSO from setting1 (●) and setting2 (▲) respectively which have similar profiles. The energy inputs of each individual energy system are illustrated in Figures 5.20 and 5.21 which is similar to the STMO. They show the proportions of the energy system regarding CO₂ reduction. Figure 5.22 shows Pareto fronts from all three techniques and Figure 5.23 the pie charts of each system using a cluster of each technique (arrowed). System combinations from all three techniques show similar patterns.

Figure 5.24 shows a comparison between the Pareto front from the ST and the partial Pareto front of the thermal systems exceeding electrical demand. The former consists of entire systems where their total output and the peak demand match within accuracy limits. In the latter case thermal output matches thermal demand within acceptable limits, but generates a surplus of electricity, despite which, its Pareto front prevails especially on the right hand from the turning

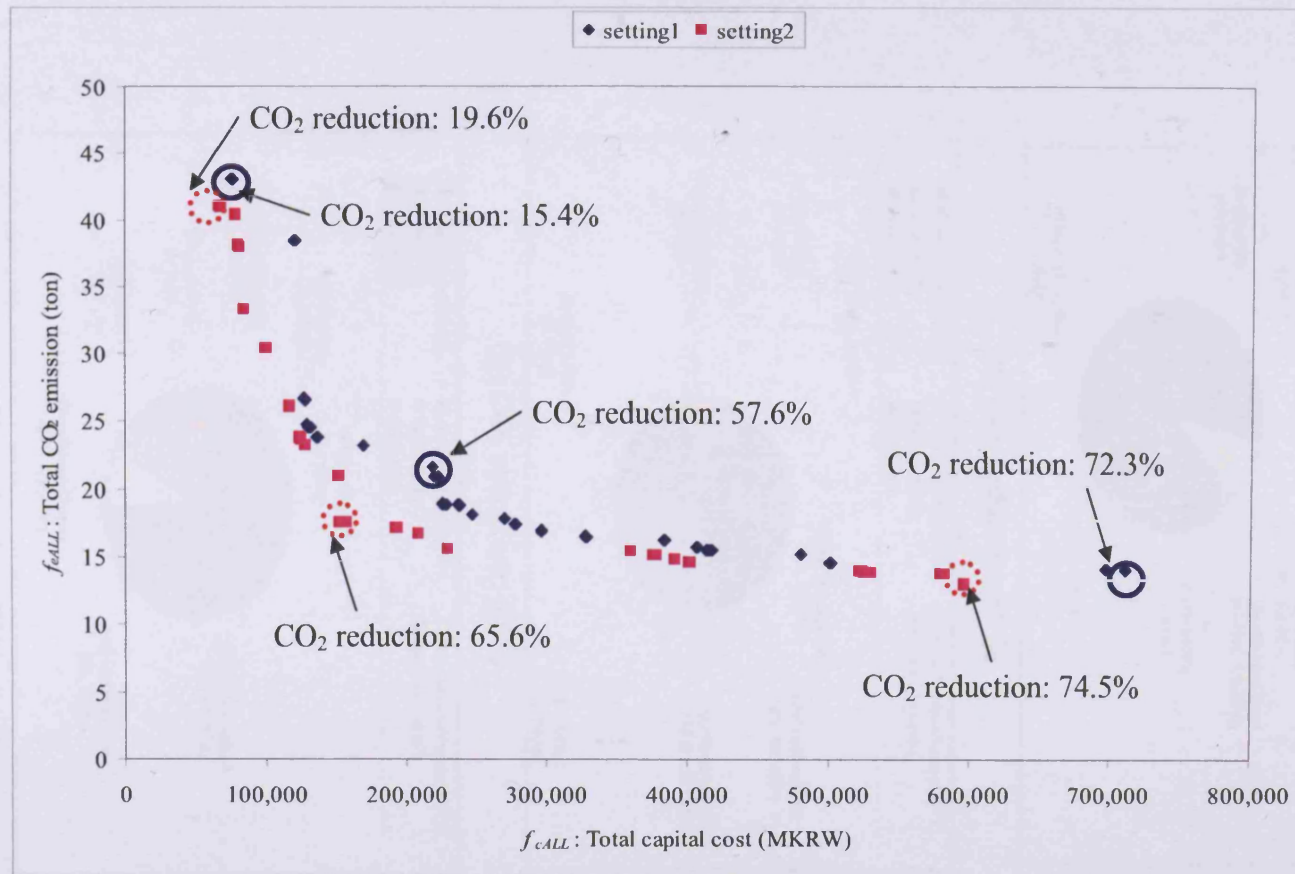
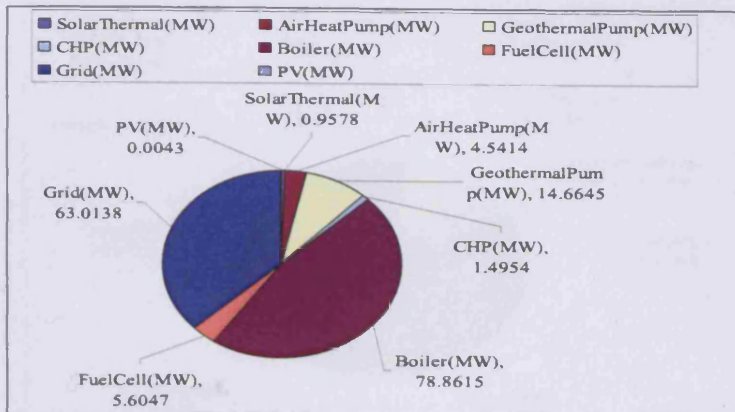
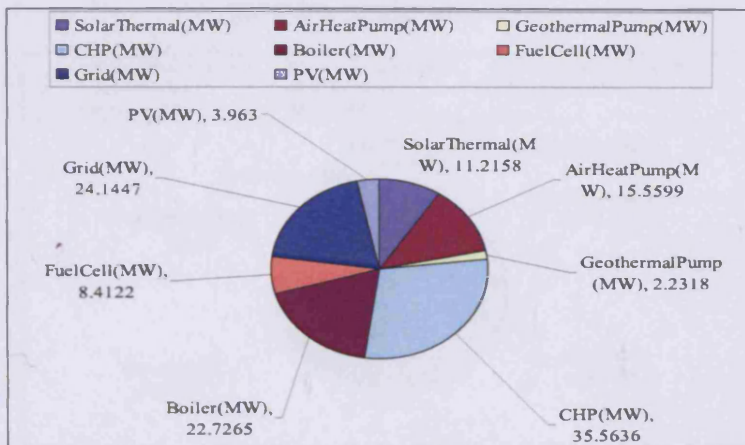


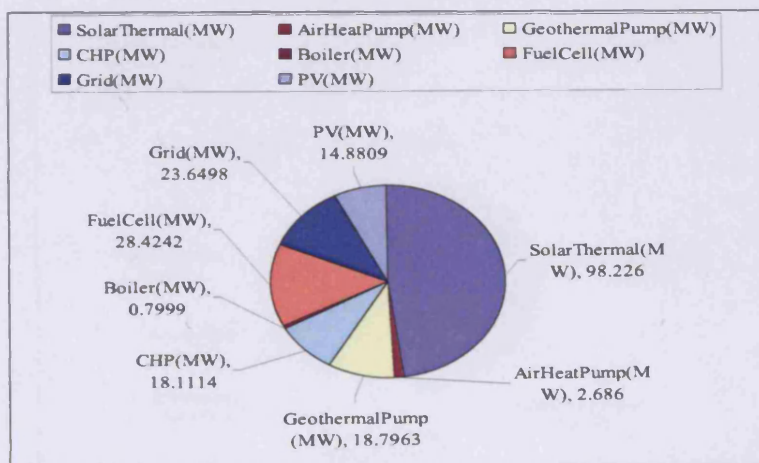
Figure 5.13 Pareto fronts of the Simultaneous Technique (ST)



(a) CO₂ reduction: 15 %

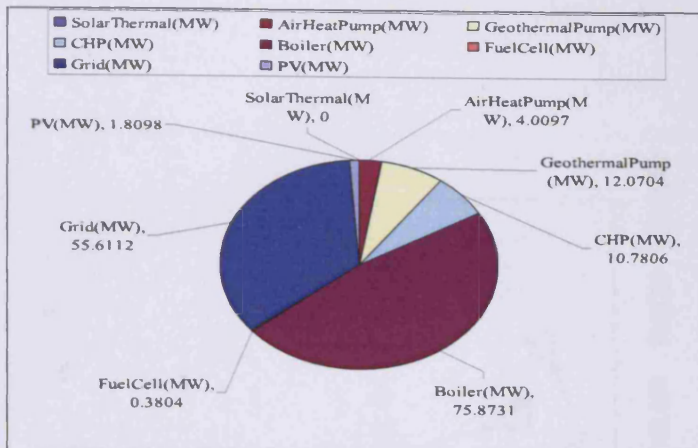


(b) CO₂ reduction: 57 %

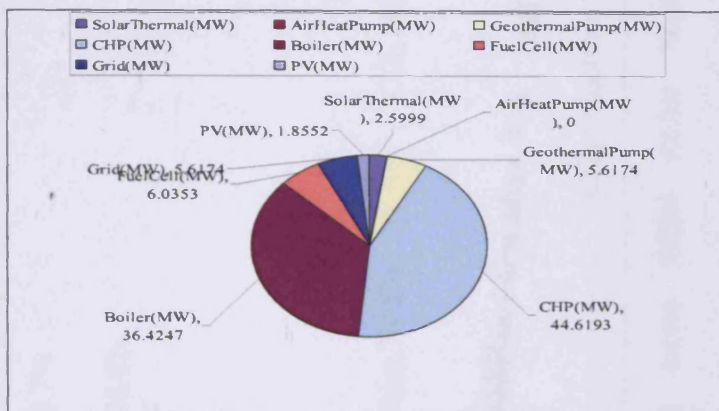


(c) CO₂ reduction: 72%

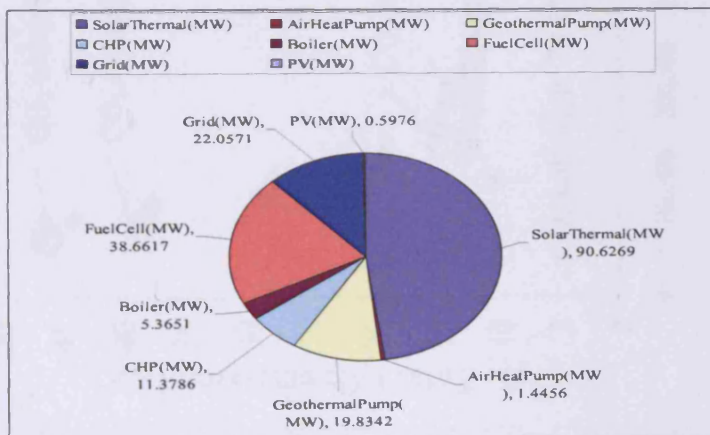
Figure 5.14 Individual proportions of the energy system and CO₂ reduction (%) with the Simultaneous Technique (ST) setting1



(a) CO₂ reduction: 19%



(b) CO₂ reduction: 65%



(c) CO₂ reduction: 74%

Figure 5.15 Individual proportions of the energy system and CO₂ reduction (%) with the Simultaneous Technique (ST) setting2

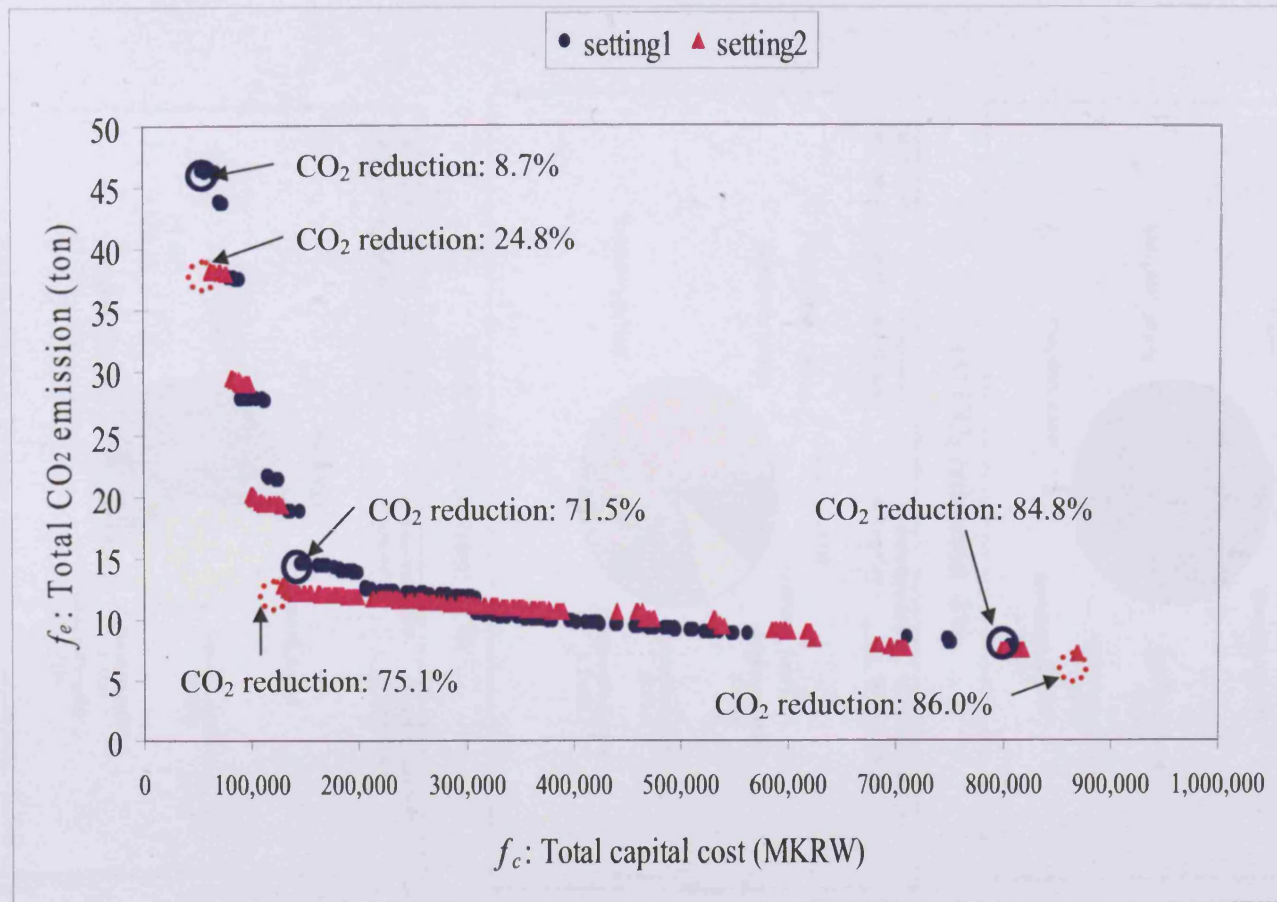
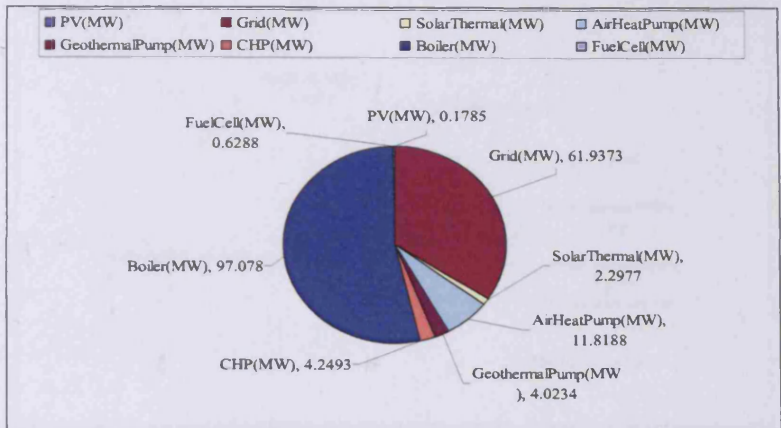
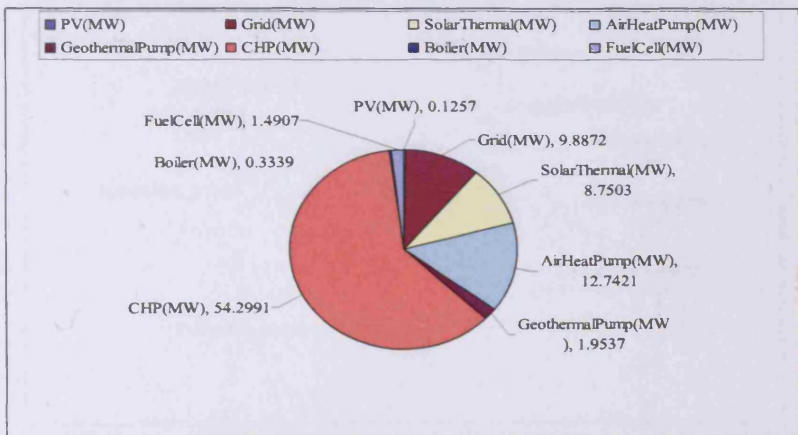


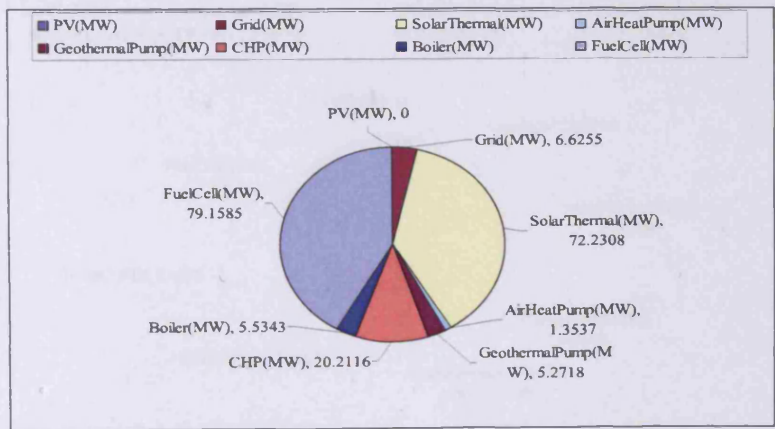
Figure 5.16 Pareto fronts of the Sequential Technique with Multi-Optimisation (STMO)



(a) CO₂ reduction: 8%

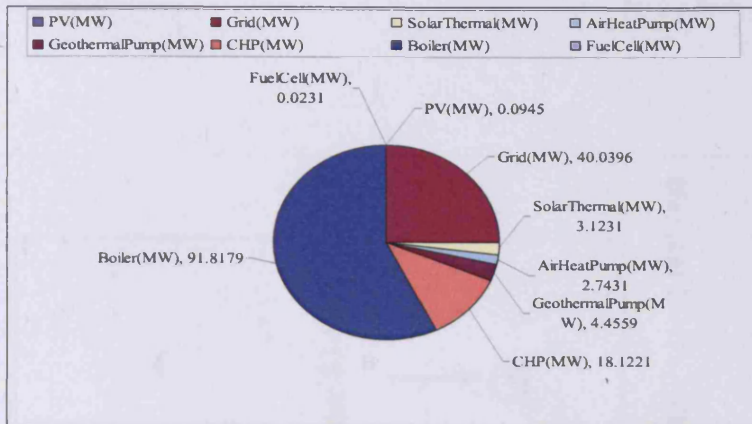


(b) CO₂ reduction: 71%

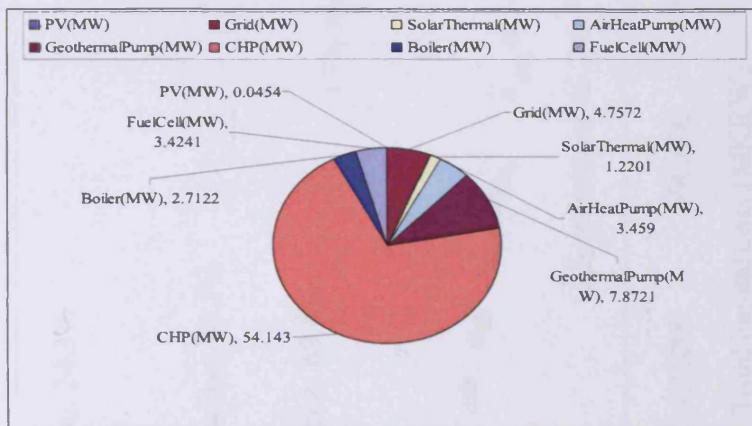


(c) CO₂ reduction: 84%

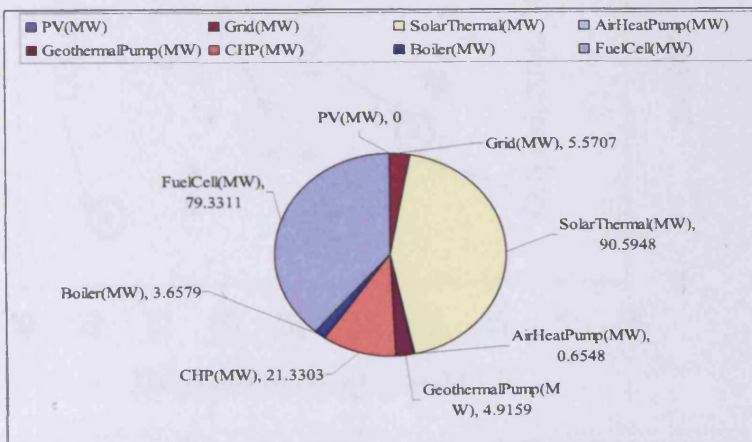
Figure 5.17 Individual proportions of the energy system and CO₂ reduction (%) with the STMO setting1



(a) CO₂ reduction: 24%



(b) CO₂ reduction: 75%



(c) CO₂ reduction: 86%

Figure 5.18 Individual proportions of the energy system and CO₂ reduction (%) with the STMO setting2

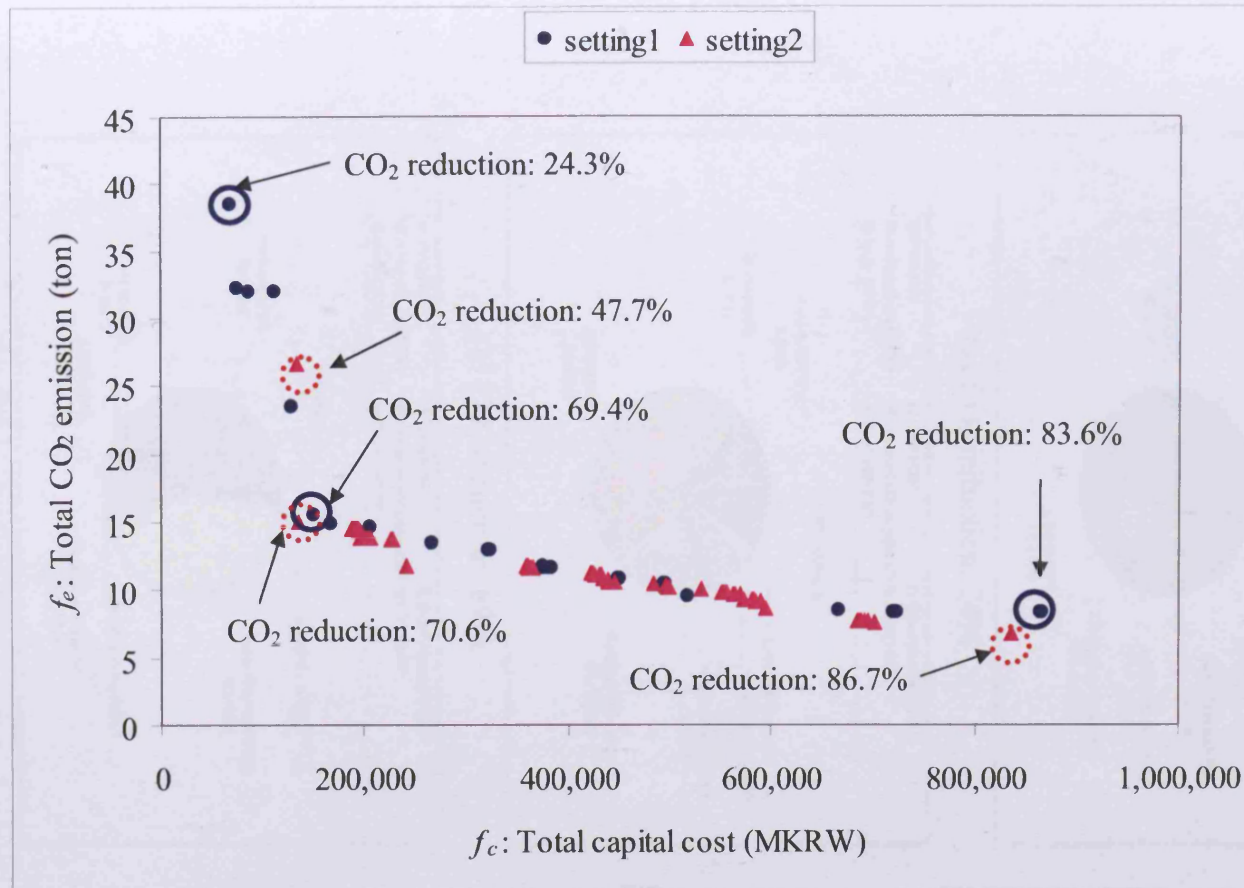
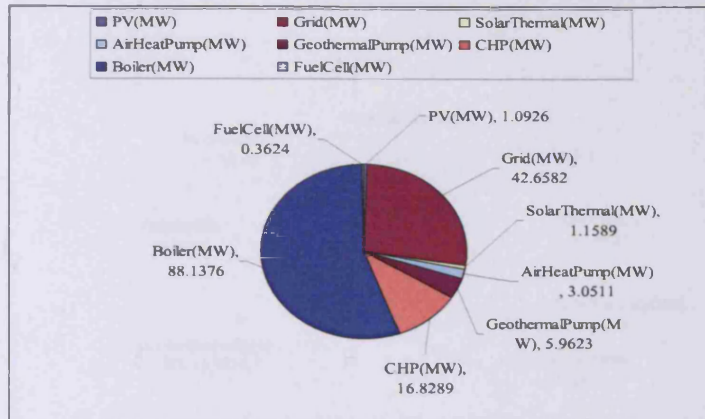
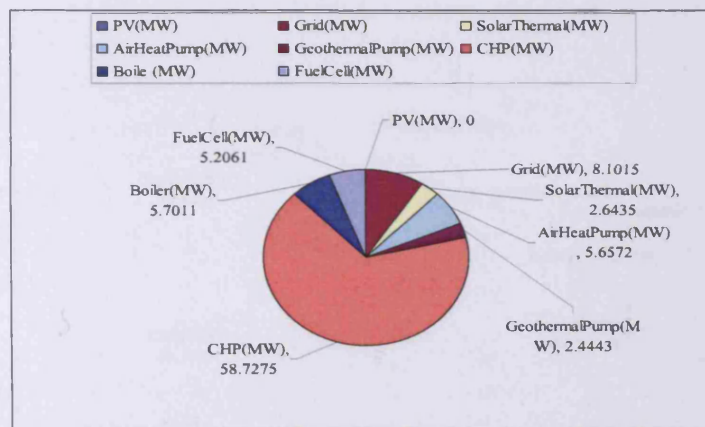


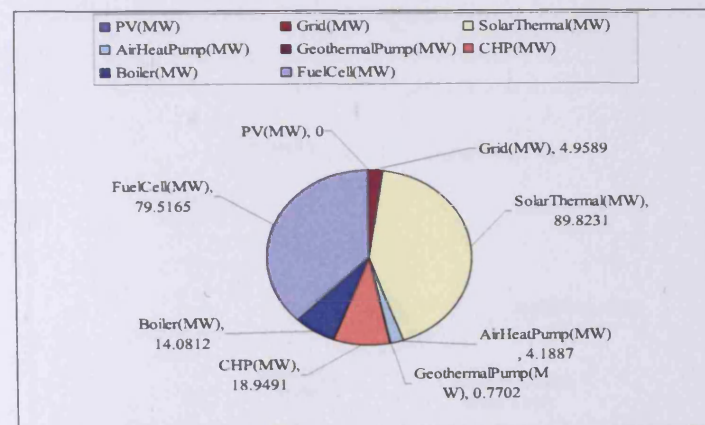
Figure 5.19 Pareto fronts of Sequential Technique with the Single Optimisation (STSO)



(a) CO₂ reduction: 24%

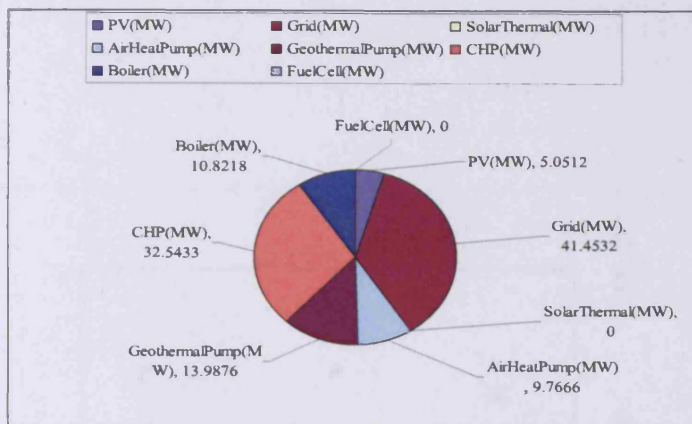


(b) CO₂ reduction: 69%

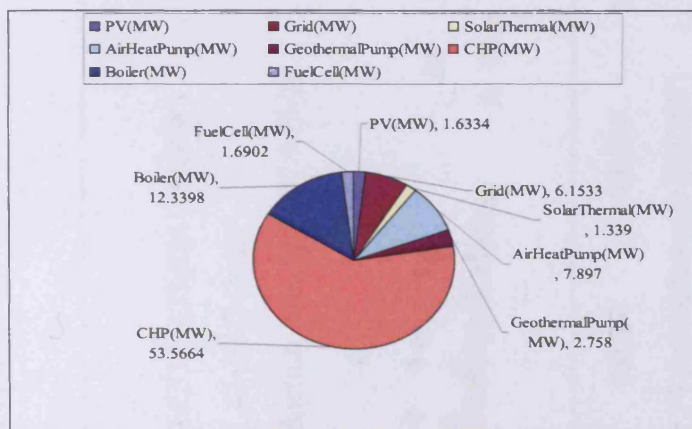


(c) CO₂ reduction: 83%

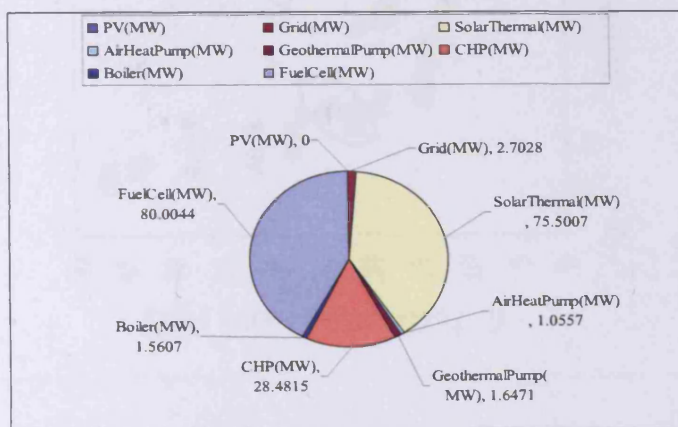
Figure 5.20 Individual proportions of the energy system and CO₂ reduction (%) with the STSO setting 1



(a) CO₂ reduction: 47%



(b) CO₂ reduction: 70%



(c) CO₂ reduction: 86%

Figure 5.21 Individual proportions of the energy system and CO₂ reduction (%) with the STSO setting2

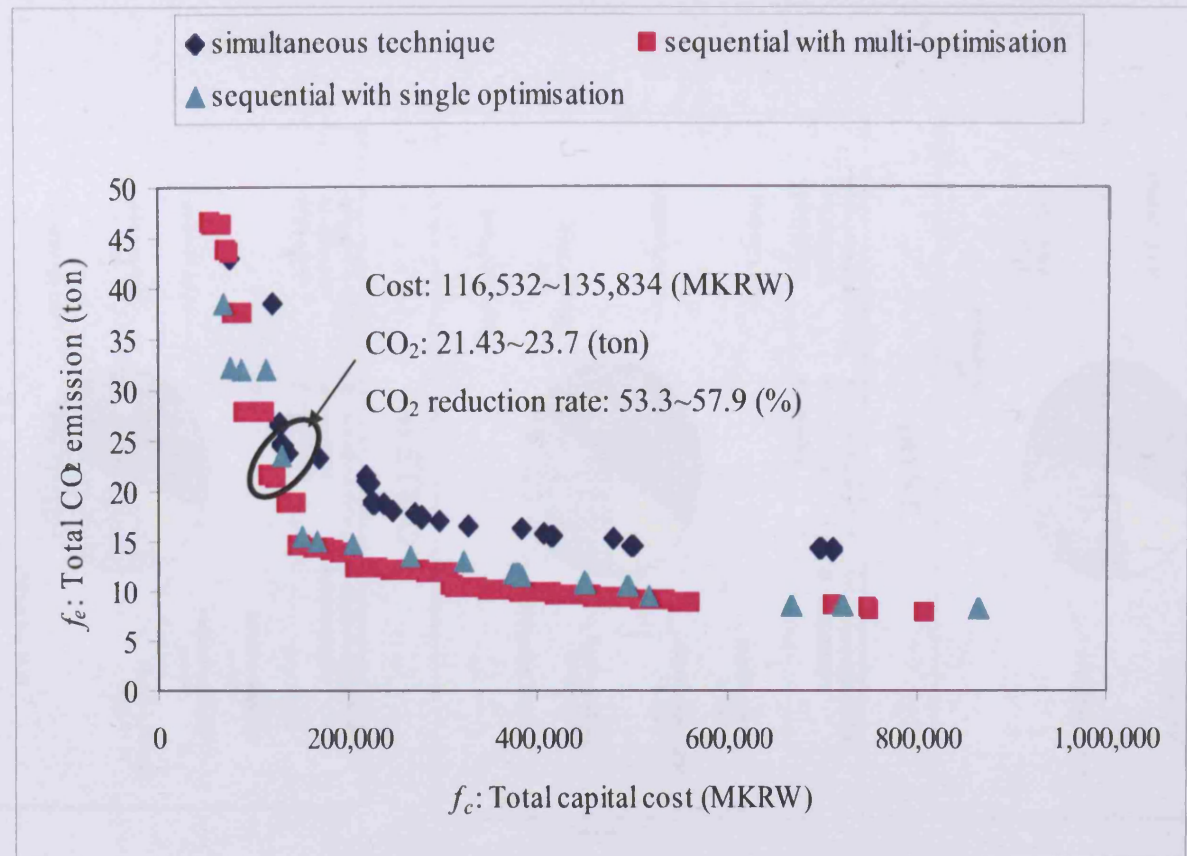
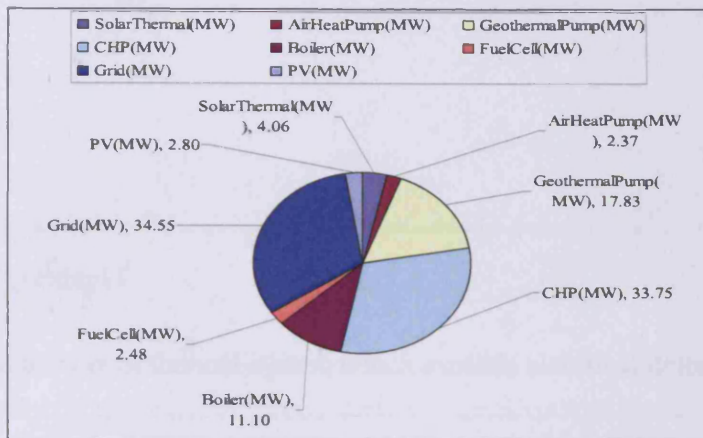
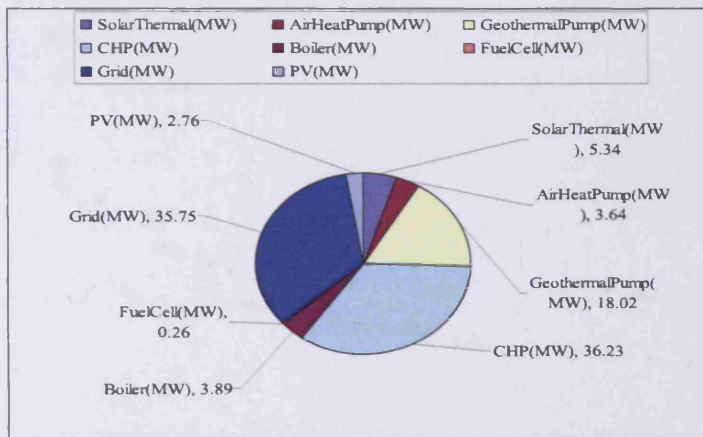


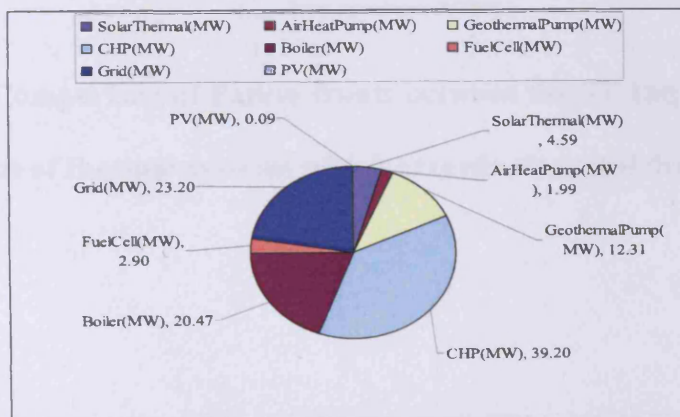
Figure 5.22 Comparison of Pareto fronts of three techniques in setting1



(a) ST



(b) STMO



(c) STSO

Figure 5.23 Individual proportions of the energy system with three techniques

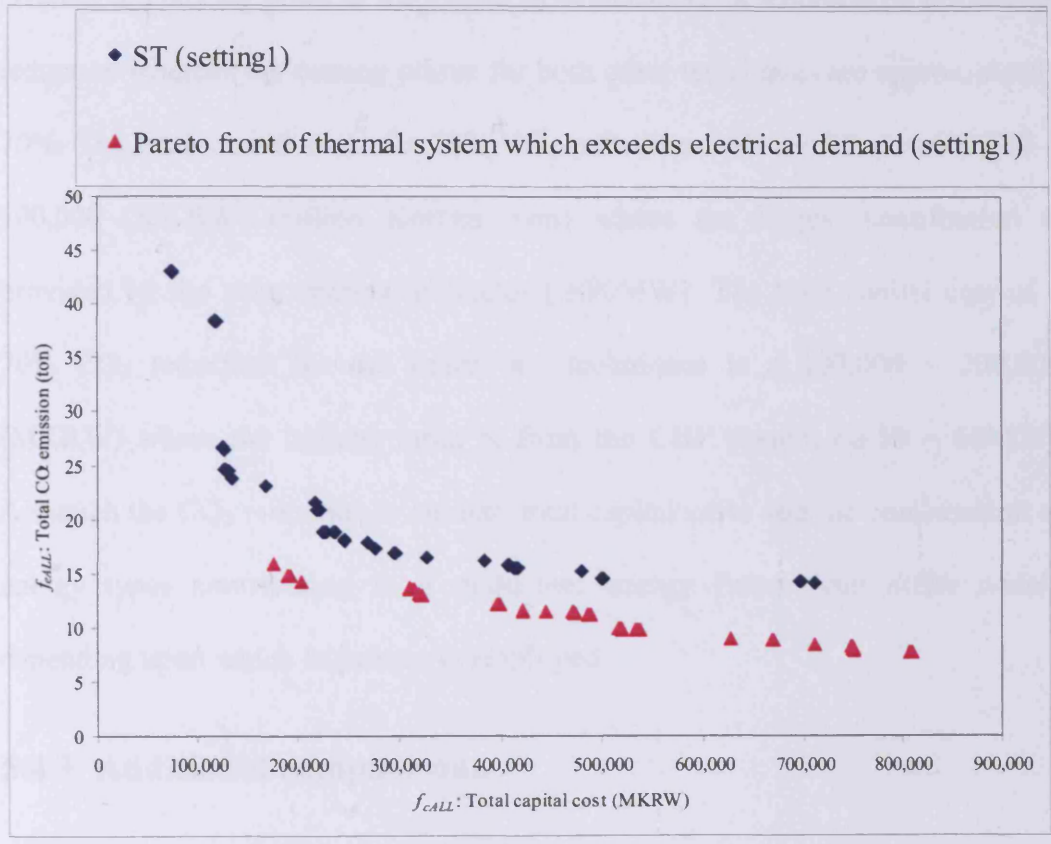


Figure 5.24 Comparison of Pareto fronts between the ST and partial Pareto front of thermal systems which exceeds electrical demand

point. When this surplus generated electricity is returned to the grid, it will enhance profits.

Overall, the turning point of the Pareto front in the ST is around 60% of the CO₂ reduction whereas the turning points for both other techniques are approximately 70%. The total capital cost of a 70% CO₂ reduction with the ST is \cong 600,000 ~ 700,000 (MKRW: million Korean won) where the largest contribution is provided by the solar thermal collector (\cong 90MW). The total capital cost of a 70% CO₂ reduction for the other two techniques is \cong 100,000 ~ 200,000 (MKRW) where the highest input is from the CHP system (\cong 50 ~ 60MW). Although the CO₂ reduction is similar, total capital costs and the combination of energy types contributing to a multi-fuel energy system can differ widely depending upon which technique is employed.

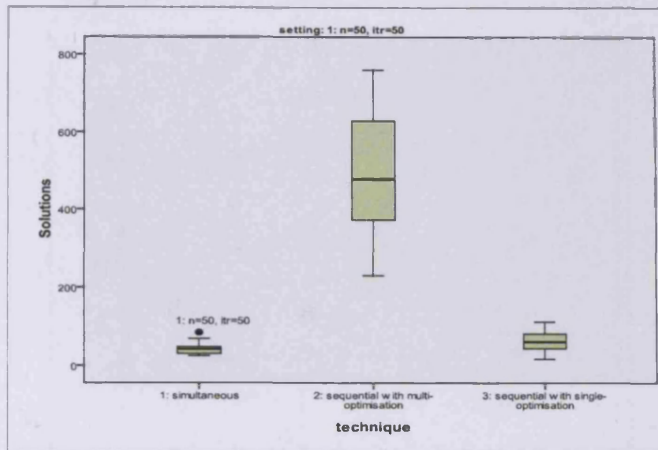
5.4.3 Additional comparisons

Final comparisons are made of the number of Pareto solutions, evaluations and program running times. Table 5.9 shows the average values from 10 independent program runs. All data sets (solutions, evaluations and times) violate the assumption of homogeneous variances according to the result from *SPSS* program, therefore a *Kruskal-Wallis* test was carried out to compare their median values. The results show there are significant

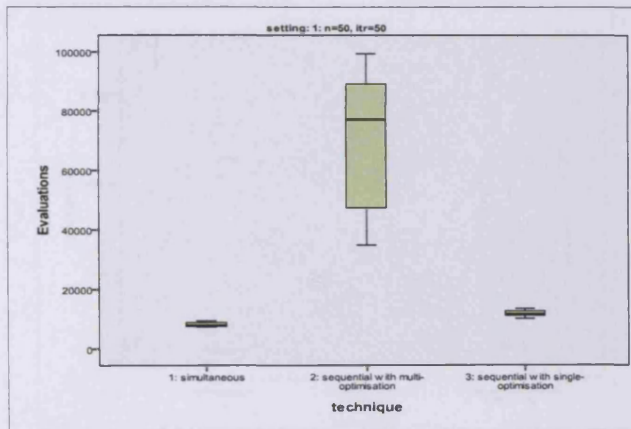
differences in all data sets regarding techniques, hence box plots were drawn as shown in Figures 5.25 and 5.26. While the number of Pareto solutions and evaluations from the STMO are noticeably higher than for the ST and the STSO, the program running time is significantly shorter than that of the ST. The number of Pareto solutions and evaluations from the STSO is marginally higher than for the ST, but the program running time is much shorter. Their Pareto fronts were compared in Figure 5.22 and the Pareto front of the STSO was superior to those of the ST. Therefore it can be claimed that the STSO is superior to the ST with a similar number of evaluations.

Technique	setting	Number of Pareto solutions	Number of evaluations	Program running times (second)
ST	setting1	45	8,077	24,543
	setting2	60	23,636	109,667
STMO	setting1	488	71,488	32
	setting2	1,346	424,476	188
STSO	setting1	63	11,699	30
	setting2	146	34,621	142

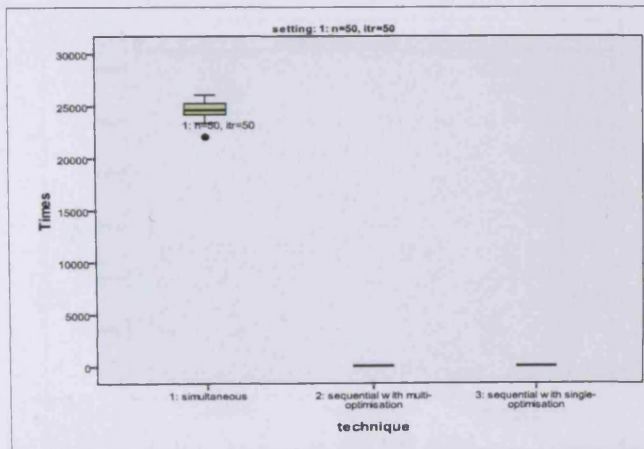
Table 5.9 The average values from 10 independent program runs



(a) Number of Pareto solutions

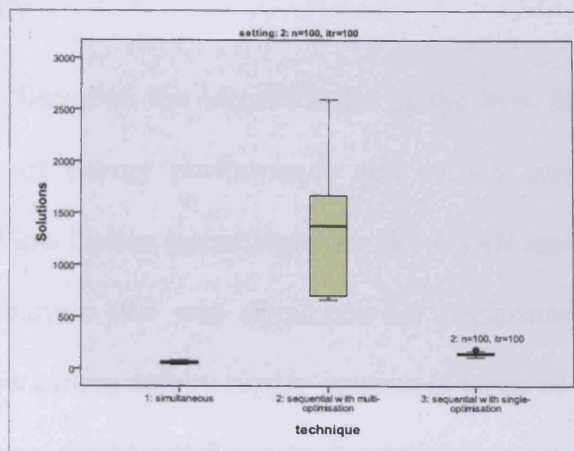


(b) Number of evaluations

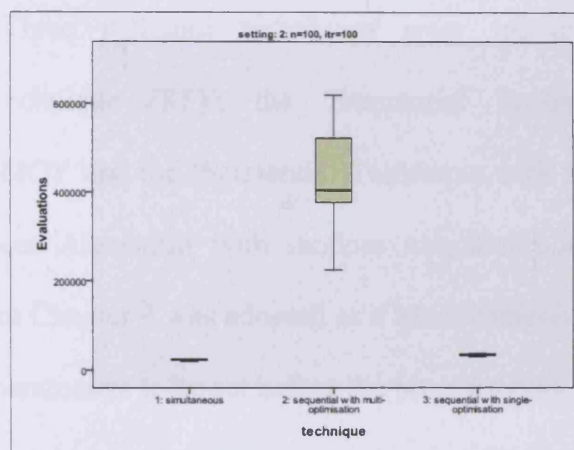


(c) Program running times

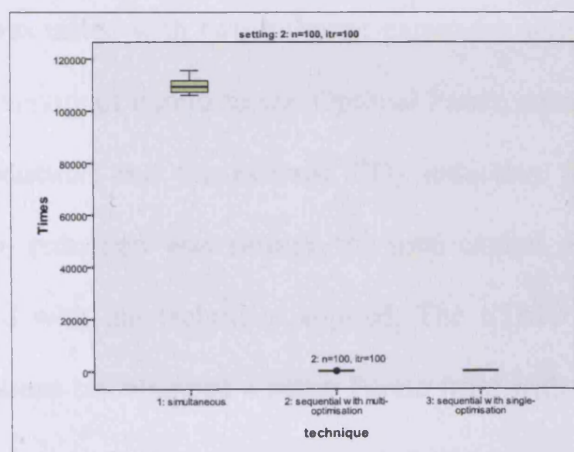
Figure 5.25 Box plots of results with setting1



(a) Number of Pareto solutions



(b) Number of evaluations



(c) Program running times

Figure 5.26 Box plots of results with setting2

5.5 Summary

This chapter has discussed the identification of the best design and technical options in terms of energy performance and carbon emission reduction in adopting zero and low carbon technologies for the ASAN new town in Korea. To enable this, an analysis tool was developed for performance assessment and optimisation of low carbon energy supply systems in terms of the capital cost and the carbon emissions, whilst satisfying the thermal and power balance constraints simultaneously. Three different techniques were introduced, namely the ‘Simultaneous Technique (ST)’, the ‘Sequential Technique with Multi-Optimisation (STMO)’ and the ‘Sequential Technique with Single Optimisation (STSO)’. The Bees Algorithm with random neighbourhood search (SCBA-randomNGH) from Chapter 4 was adopted as a Multi-Objective solver due to the small number of parameters to be set before the program runs.

Each technique was tested with two different parameter settings, and all results showed a good diversity of Pareto fronts. Optimal Pareto solutions were obtained for their CO₂ reduction and the average CO₂ reduction from 18% to 90%. Although the CO₂ reduction was similar, the total capital cost and the system proportions varied with the technique applied. The STMO required a greater number of evaluations but obtained a better Pareto front with a higher number of

Pareto solutions with less running time. Overall, it outperformed other algorithms in finding the best cost and the best emissions.

Therefore, this work has confirmed the suitability of the Bees Algorithm for designing a Low-Carbon City, and achieving financial savings.

6 CONCLUSION

6.1 Contributions

The research introduced a number of developments to the Bees Algorithm for application to Multi-Objective Optimisation Problems (MOOPs). It has also enhanced the Bees Algorithm in terms of its ease of use, robustness and speed.

The main contributions are:

- Developing new Multi-Objective versions of the Bees Algorithm (WSBA, WSBA-*e* and WSBA-*m*) to run in conjunction with the weighted sum method to generate Pareto optimality, whilst being less sensitive to the Pareto front profile.
- Developing a dynamic Multi-Objective Bees Algorithm with a reduced number of parameters.
- Proposing new ways (basicNGH, randomNGH and wsNGH) of performing local search in the Bees Algorithms enhancing the sensitivity and diversity of the Pareto front.
- Obtaining optimal solutions to the Environmental/Economic power Dispatch Problem (EEDP).

- Designing a multi-fuel hybrid energy system for a Low-Carbon City.
- Developing new multi-constraint optimisation techniques (ST, STMO and STSO) for identifying the optimum energy permutations of a multi-fuel system.

6.2 Conclusions

The feasibility of utilising the Bees Algorithm to solve Multi-Objective optimisation problems such as the EEDP, optimising benchmark mathematical functions, and in the design of a Low-Carbon City formed the basis for the research presented in this thesis. The proposed Bees Algorithm versions which have been presented improve upon the current state of the art of the Bees Algorithm. The key conclusions for each topic analysed are:

- The Bees Algorithm employing the weighted sum method (WSBA, WSBA-*e* and WSBA-*m*) generates a good Pareto front from a single program run. The EEDP results proved the robustness of the algorithm. The Pareto fronts from WSBA-*e* and WSBA-*m* were more diverse and less sensitive to the Pareto front profile.
- The number of parameters was reduced from six to three in the SCBA-basicNGH and the SCBA-randomNGH, and four in the SCBA-wsNGH and the MSBA-wsNGH. The EEDP results were superior for the

proposed algorithm. The new neighbourhood search methods enhanced the Pareto fronts.

- Applying the proposed Bees Algorithm (SCBA-randomNGH) to Multi-Objective function optimisation showed that it performed very well even for a disconnected Pareto front and non-convex set.
- Both the simultaneous and sequential approaches (ST, STMO and STSO) employing the proposed Bees Algorithm (SCBA-randomNGH) generated a good diversity of Pareto fronts. However, they proved to have contrasting advantages and disadvantages permitting the decision maker to choose technique according to specific requirements.
- A CO₂ reduction from 20% to 90% was achieved by the designed multi-fuel energy system. The results indicate that a Low-Carbon City with cheaper capital cost could be designed by the Bees Algorithm.

Therefore this work confirms the suitability of the Bees Algorithm to Multi-Objective problems.

6.3 Suggestions for future research

The following are possible extensions that can be made to this work:

- A diversity mechanism could be developed to obtain a more uniform spread of solutions along the Pareto front.
- A mechanism to calculate the distance between the approximation set and the true Pareto front could be introduced.
- A procedure to select the best compromise Pareto solution could be developed to support the decision-making process.
- Due to insufficient data, only one linear system was tested in the continuous search space. There are many energy conversion systems which may not have linear capital cost and CO₂ emissions. More data regarding types of energy converters is necessary in developing the Bees Algorithm in a discontinuous search space.
- For the Low-Carbon City, only capital cost was considered, which is more expensive than conventional systems. The running costs of renewable systems are almost zero. Therefore at the design stage, economic analysis should be considered over a long period.

APPENDIX

Details of the multi-fuel energy systems discussed in

Chapter 5

1. Solar thermal collector

a. Efficiency

Model	Manufacturer	Ground conditions	No./100m ²	IDEAL generation (MWh)
Msc-32	American Energy Technologies, Inc.	Normal	33	703.392
SX-600	Solarmax Inc.	Normal	33	724.0444

Table 1 Specifications of selected model of solar thermal collector in MERIT

		South		East		North		West	
	Model	Energy generated	Efficiency	Energy generated	Efficiency	Energy generated	Efficiency	Energy generated	Efficiency
Tilt	Msc-32	114.30	16.25%	114.30	16.25%	114.30	16.25%	114.30	16.25%
@ 0	SX-600	91.61	12.65%	91.61	12.65%	91.61	12.65%	91.61	12.65%
Tilt	Msc-32	125.82	17.89%	87.20	12.40%	78.96	11.23%	113.97	16.20%
@30	SX-600	101.66	14.04%	68.39	9.45%	59.64	8.24%	92.36	12.76%
Tilt	Msc-32	121.42	17.26%	75.73	10.77%	62.84	8.93%	107.23	15.24%
@40	SX-600	97.89	13.52%	58.41	8.07%	44.98	6.21%	86.47	11.94%
Tilt	Msc-32	108.53	15.43%	55.01	7.82%	35.19	5.00%	88.51	12.58%
@ 60	SX-600	83.78	11.57%	40.53	5.60%	19.73	2.72%	69.59	9.61%
Total average of solar thermal collector efficiency								11.88%	

Table 2 Efficiency of solar thermal collector using weather profile in MERIT

b. Required unit area

Model	Collector length (m)	Collector width(m)
MSC-32-no	2.491	1.221
MSC-32-no1	2.491	1.221
SX-600-1	2.494	1.232
SX-600-2	2.491	1.323
Average	2.492	1.249
Average area (m ²)		3.11
Required solar thermal collector unit area (m ² /kW)		3.31

Table 3 Required solar thermal collector unit areas in MERIT

c. Unit cost

Type	Capacity (m ²)	Installation unite cost (KKRW/m ²)
Flat plate solar collector	1	930
Single vacuum tube solar collector	1	980
Double vacuum tube solar collector	1	940
Average of unit cost (KKRW/m ²)		950
Unit cost (KRW/kW)		3,144,500

Table 4 Unit cost of solar thermal collector from Korea Energy Management Corporation (KEMCO)

2. Boiler

a. Efficiency and unit cost

Capacity(kWth)	Cost (KRW)	Unit cost (KRW/kWth)	Efficiency (%)
20,000	2,000,000	100,000	85
1,000	150,000	150,000	85
250	50,000	200,000	85
Average of unit cost (KRW/kW)		150,000	

Table 5 Unit cost and efficiency of boiler from UK Biomass Strategy 2007
(Energy Technologies Unit 2007)

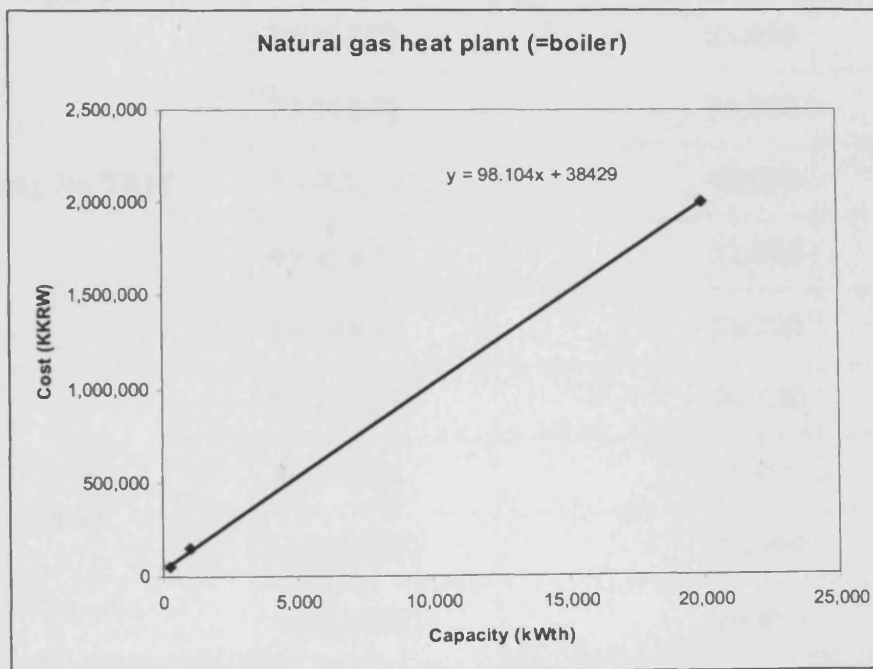


Figure 1 Graph of cost vs. capacity of boiler from UK Biomass Strategy 2007

3. Geothermal heat pump

a. Unit cost

Company and type		Capacity (kW)	Total cost (KKRW)	Unit cost (KKRW/Kw)
KEMCO	Vertical tight	1	1,140	1,140
	Open type	1	1,010	1,010
KongkanKorea		10(10RT)	14,300	1,430
		20(20RT)	22,000	1,100
		40(40RT)	36,300	908
		50(50RT)	41,800	836
Kyung Jin TRM		20(20RT)	23,650	1,183
		29(29RT)	34,650	1,195
		40(40RT)	40,040	1,001
		45(45RT)	52,800	1,173
		54(54RT)	58,300	1,080
Topsol		55(15RT)	20,350	370
		70(20RT)	26,400	377
		100(30RT)	36,300	363
		140(40RT)	50,600	361
Average of unit cost (KKRW/kW)				902

Table 6 Unit cost of geothermal heat pump in Korea

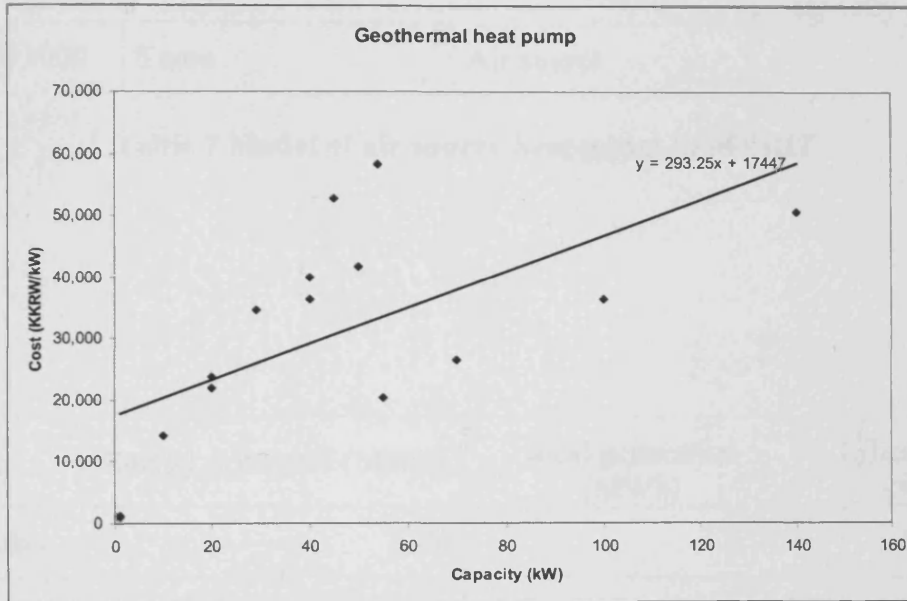


Figure 2 Graph of cost vs. capacity of geothermal heat pump in Korea

4. Air source heat pump

a. Efficiency

Model	Manufacturer	Type	Rated heating capacity (kW)
trane 1000	Trane	Air source	7.678

Table 7 Model of air source heat pump in MERIT

	Energy generated (MWh)	Ideal generation (MWh)	Efficiency (%)
January	3.76	5.71	65.82%
February	4.10	5.16	79.46%
March	5.32	5.71	93.13%
November	5.11	5.53	92.44%
December	4.36	5.71	76.32%

Table 8 Efficiency of air source heat pump using weather profile in MERIT

b. Unit cost

Fine Enertec		
Capacity (kW)	Cost (KKRW)	Unit cost (KKRW/Kw)
12	20,350	1,696
17	24,734	1,455
23	30,855	1,342
29	36,075	1,244
36	41,129	1,142
47	55,440	1,180
61	64,680	1,060
71	74,030	1,043
Average of unit cost (KKRW/Kw)		1,270

Table 9 Unit cost of air source heat pump in Korea

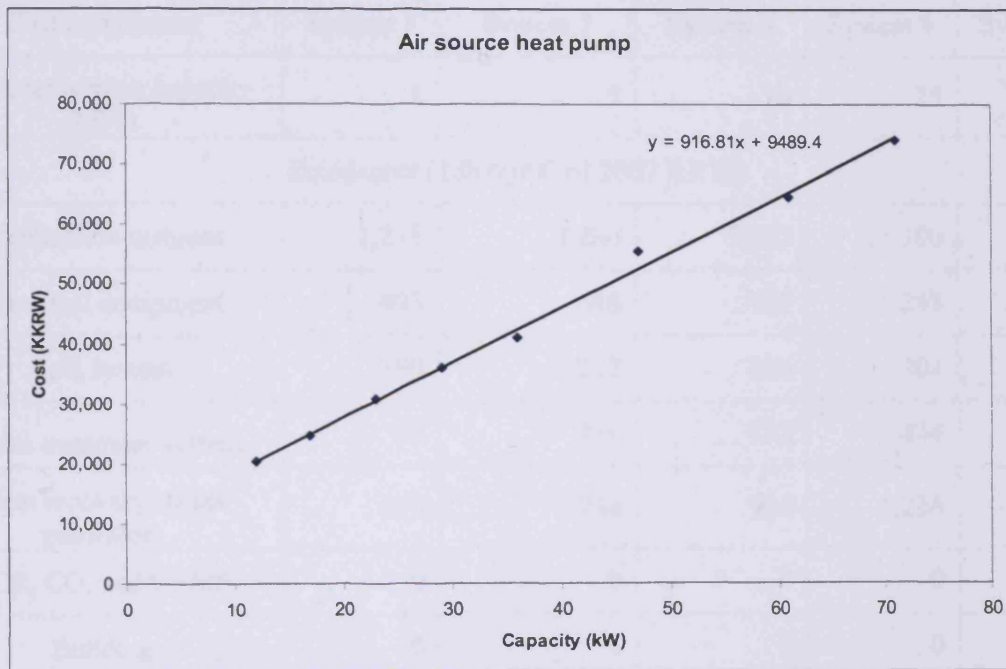


Figure 3 Graph of cost vs. capacity of air source heat pump in Korea

5. CHP

a. Unit cost

Cost component	System 1	System 2	System 3	System 4	System 5
Nominal turbine capacity (MW)	1	5	10	25	40
Equipment (1,000,000 of 2007 KRW)					
Combustion turbines	1,218	3,280	7,322	15,300	28,440
Electrical equipment	493	648	784	1,248	1,890
Fuel system	199	212	226	301	430
Water treatment system	89	216	352	444	499
Heat recovery steam generators	610	738	935	1,236	1,489
SCR, CO, and CEMS	0	0	0	0	0
Building	0	0	0	0	0
Total equipment	2,608	5,095	9,618	18,528	32,748
Construction	923	1,682	3,082	5,936	10,493
Total process capital	3,530	6,778	12,700	24,464	43,241

Table 10 Specific cost of CHP from Energy and Environmental Analysis (an ICF International Company) (Energy and Environmental Analysis 2008b; US Environmental Protection Agency 2008)

Nominal turbine capacity (MW)	Capital cost (MKRW)	Unit cost (MKRW/MW)
1	3,530	3,530
5	6,778	1,356
10	12,700	1,270
25	24,464	979
40	43,241	1,081
Average of unit cost(MKRW/MW)		1,643

Table 11 Unit cost of CHP

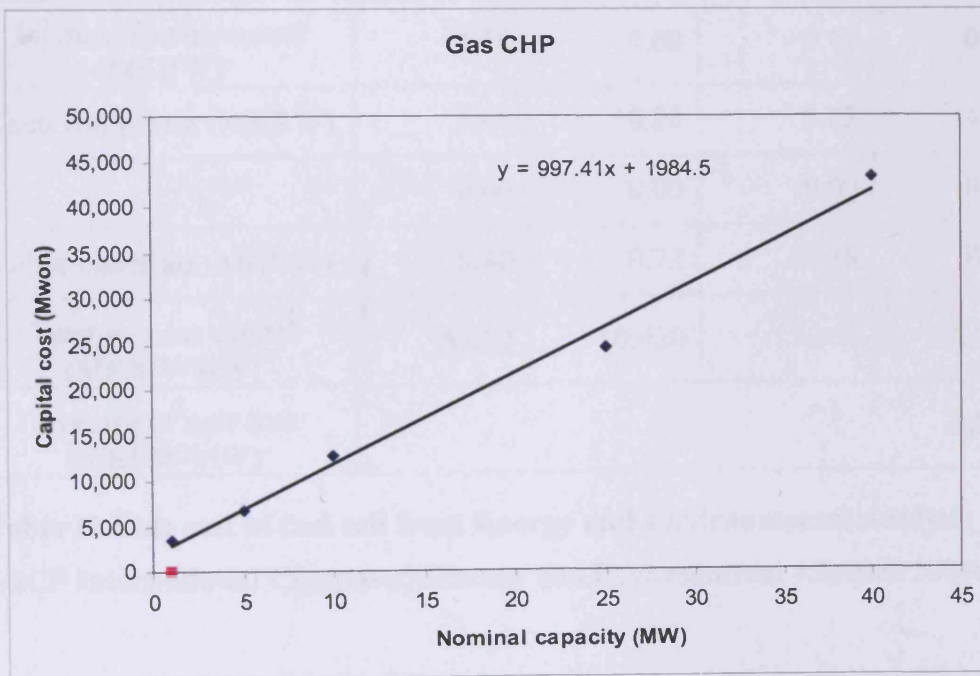


Figure 4 Graph of cost vs. capacity of CHP

6. Fuel cell

a. Unit cost

Installed cost components	System 1	System 2	System 4	System 5
Fuel cell type	PAFC	PEM	MCFC	MCFC
Nominal capacity (MW)	200	10	300	1,200
Equipment				
Fuel cell package (MKRW)	5.40	9.60	4.80	4.64
Heat recovery and other equipment (MKRW)	0.10	0.00	0.07	0.04
Interconnect/electrical (MKRW)	0.18	0.60	0.14	0.05
Total equipment (MKRW)	5.68	10.20	5.02	4.72
	0.00	0.00	0.00	0.00
Labor/materials (MKRW)	0.40	0.72	0.35	0.34
Total process capital (MKRW/MW)	6,072	10,920	5,364	5,052
Average of unit cost (MKRW/MW)	6,852			

Table 12 Unit cost of fuel cell from Energy and Environmental Analysis (an ICF International Company) (Energy and Environmental Analysis 2008a)

7. PV

a. Efficiency

Model	Manufacturer	Capacity(W)	Type	No/100m2	Ideal (MWh)
PV1	Yingli	110	Poly	100	96.36
PV2	Yingli	110	Mono	115	110.814
PV3	BP	130	Poly	90	102.492
PV5	Yingli	160	Poly	78	109.3248
PV6	PSE	160	Mono	79	110.7264
PV7	SE	189	Mono	80	132.4512
PV8	SE	240	Poly	62	130.3488
PV9	SE	250	Poly	62	135.78

Table 13 Specification of PV model in MERIT

		Annual generation (MWh)							
		South		East		North		West	
	Model	Energy generated	Efficiency	Energy generated	Efficiency	Energy generated	Efficiency	Energy generated	Efficiency
Tilt @ 0	PV1	12.65	13.1%	12.73	13.2%	12.73	13.2%	12.82	
	PV2	11.53	10.4%	11.75	10.6%	11.74	10.6%	12.02	10.8%
	PV3	12.81	12.5%	12.92	12.6%	12.91	12.6%	13.04	12.7%
	PV5	14.32	13.1%	14.4	13.2%	14.4	13.2%	14.5	13.3%
	PV6	14.23	12.9%	14.32	12.9%	14.31	12.9%	14.42	13.0%
	PV7	17.55	13.3%	17.65	13.3%	17.64	13.3%	17.75	13.4%
	PV8	17.24	13.2%	17.33	13.3%	17.33	13.3%	17.44	13.4%
	PV9	18.05	13.3%	18.15	13.4%	18.14	13.4%	18.26	13.4%
Tilt @30	PV1	18.03	18.7%	15.74	16.3%	14.52	15.1%	17.56	18.2%
	PV2	14.17	12.8%	13.51	12.2%	12.69	11.5%	14.86	13.4%
	PV3	17.85	17.4%	15.79	15.4%	14.6	14.2%	17.57	17.1%
	PV5	20.49	18.7%	17.85	16.3%	16.44	15.0%	19.91	18.2%
	PV6	20.33	18.4%	17.73	16.0%	16.34	14.8%	19.77	17.9%
	PV7	25.3	19.1%	21.94	16.6%	20.2	15.3%	24.5	18.5%
	PV8	24.76	19.0%	21.52	16.5%	19.82	15.2%	24.01	18.4%
	PV9	26	19.1%	22.56	16.6%	20.77	15.3%	25.19	18.6%

Table 14 Efficiency of PV using weather profile in MERIT

		Annual generation (MWh)							
		South		East		North		West	
	Model	Energy generated	Efficiency	Energy generated	Efficiency	Energy generated	Efficiency	Energy generated	Efficiency
Tilt @40	PV1	18.11	18.8%	15.38	16.0%	13.89	14.4%	17.47	18.1%
	PV2	13.96	12.6%	13.2	11.9%	12.3	11.1%	14.64	13.2%
	PV3	17.88	17.4%	15.42	15.0%	13.99	13.6%	17.45	17.0%
	PV5	20.59	18.8%	17.43	15.9%	15.72	14.4%	19.81	18.1%
	PV6	20.42	18.4%	17.31	15.6%	15.62	14.1%	19.67	17.8%
	PV7	25.44	19.2%	21.43	16.2%	19.3	14.6%	24.39	18.4%
	PV8	24.89	19.1%	21.02	16.1%	18.94	14.5%	23.9	18.3%
	PV9	26.15	19.3%	22.03	16.2%	19.85	14.6%	25.07	18.5%
	Tilt @ 60	PV1	17.19	17.8%	14.17	14.7%	12.4	12.9%	16.45
PV2		13.03	11.8%	12.24	11.0%	11.31	10.2%	13.7	12.4%
PV3		16.93	16.5%	14.22	13.9%	12.56	12.3%	16.42	16.0%
PV5		19.55	17.9%	16.05	14.7%	14.03	12.8%	18.66	17.1%
PV6		19.39	17.5%	15.94	14.4%	13.95	12.6%	18.53	16.7%
PV7		24.17	18.2%	19.73	14.9%	17.2	13.0%	22.97	17.3%
PV8		23.64	18.1%	19.35	14.8%	16.89	13.0%	22.51	17.3%
PV9		24.84	18.3%	20.28	14.9%	17.68	13.0%	23.62	17.4%
Total average of PV efficiency									15.1%

Table 14 Continued

b. Required unit area

PV Panel			
Name	Nominal power (kW)	Panel height(m)	Panel width(m)
100W Ploy	100	1.32	0.66
110W Mono	110	1.32	0.66
115W MIT Poly	115	1.5	0.67
120W MIT Poly	120	1.5	0.67
125W MIT Poly	125	1.5	0.67
130W BP Mono	130	1.38	0.81
130W MIT Poly	130	1.5	0.67
130W Mono	130	1.38	0.81
165W SP Poly	165	1.58	0.83
167W SP Poly	167	1.58	0.83
30W Poly	32.5	0.83	0.5
50W Poly	51.2	0.96	0.5
60W Poly	60	1.11	0.5
Average value	110.44	1.343	0.675
Panel area (m2)			0.907
Required PV unit area (m2/kW)			8

Table 15 Required unit areas for PV cell in MERIT

c. Unit cost

Type		Capacity (kW)	Installation unit cost (KKRW)
General building	Fixed PV	1	9,240
	Tracking PV	1	10,900
	Building integrated photovoltaic (BIPV)	1	14,960
Solar house		1	7,210
Average of unit cost			10,578

**Table 16 Unit cost of PV in Korea Energy Management Corporation
(KEMCO)**

8. Wind power generator

a. Efficiency

Model	Manufacturer	Capacity (kW)	Type	Hub height (m)	Ideal generation (MWh)
Abacus 5kW	Abacus Korea	5	Vertical	2.5	43.80
Airdophin 1kW	Semplice Energy	1	Horizontal	10	8.76
Proven 15kW	Proven	15	Horizontal	15	131.40
Nordex 800kW	Nordex	800	Horizontal	46	7,008.00
REPower 5MW	REpower System	5,000	Horizontal	117	43,800.00

Table 17 Specifications of wind power generator model in MERIT

Surrounding surface type	
type 1	Water Surface
type 2	Open terrain with smooth surface
type 3	Open agriculture with little shelter
type 4	Open agriculture well sheltered
type 5	Very tough and uneven terrain
type 6	Village / small town
type 7	Large city with tall buildings
type 8	Very large city with tall buildings and skyscrapers

Table 18 Surrounding surface type of wind power generator

Surrounding surface type	Model	Min (16m)		Average(40m)		Max(118m)	
		Energy (MWh)	Efficiency	Energy (MWh)	Efficiency	Energy (MWh)	Efficiency
type 1	Abacus 5kW	1.03	2.35%	1.29	2.95%	1.64	3.74%
	Airdophin 1kW	0.28	3.24%	0.33	3.76%	0.40	4.60%
	Proven 15kW	7.27	5.53%	8.17	6.22%	9.67	7.36%
	Nordex 800kW	102.32	1.46%	114.10	1.63%	136.66	1.95%
	REPower 5MW	1,080.00	2.47%	1,120.00	2.56%	1,250.00	2.85%
type 2	Abacus 5kW	0.92	2.10%	1.24	2.83%	1.65	3.77%
	Airdophin 1kW	0.27	3.06%	0.32	3.64%	0.41	4.63%
	Proven 15kW	6.91	5.26%	7.99	6.08%	9.75	7.42%
	Nordex 800kW	99.50	1.42%	113.27	1.62%	142.47	2.03%
	REPower 5MW	1,100.00	2.51%	1,150.00	2.63%	1,330.00	3.04%
type 3	Abacus 5kW	0.82	1.88%	1.16	2.65%	1.66	3.79%
	Airdophin 1kW	0.25	2.81%	0.31	3.50%	0.41	4.68%
	Proven 15kW	6.48	4.93%	7.80	5.94%	9.87	7.51%
	Nordex 800kW	94.61	1.35%	112.41	1.60%	145.85	2.08%
	REPower 5MW	1,110.00	2.53%	1,190.00	2.72%	1,380.00	3.15%

Table 19 Efficiency of wind power generator using weather profile in MERIT

Surrounding surface type	Model	Min (16m)		Average(40m)		Max(118m)	
		Energy (MWh)	Efficiency	Energy (MWh)	Efficiency	Energy (MWh)	Efficiency
type 4	Abacus 5kW	0.46	1.06%	0.91	2.09%	1.76	4.02%
	Airdophin 1kW	0.17	1.93%	0.26	2.97%	0.43	4.90%
	Proven 15kW	4.88	3.71%	6.88	5.24%	10.39	7.91%
	Nordex 800kW	81.41	1.16%	105.17	1.50%	165.57	2.36%
	REPower 5MW	1,210.00	2.76%	1,340.00	3.06%	1,690.00	3.86%
type 5	Abacus 5kW	0.46	1.06%	0.91	2.09%	1.76	4.02%
	Airdophin 1kW	0.17	1.93%	0.26	2.97%	0.43	4.90%
	Proven 15kW	4.88	3.71%	6.88	5.24%	10.39	7.91%
	Nordex 800kW	81.41	1.16%	105.17	1.50%	165.57	2.36%
	REPower 5MW	1,210.00	2.76%	1,340.00	3.06%	1,690.00	3.86%
type 6	Abacus 5kW	0.46	1.06%	0.91	2.09%	1.76	4.02%
	Airdophin 1kW	0.17	1.93%	0.26	2.97%	0.43	4.90%
	Proven 15kW	4.88	3.71%	6.88	5.24%	10.39	7.91%
	Nordex 800kW	81.41	1.16%	105.17	1.50%	165.57	2.36%
	REPower 5MW	1,210.00	2.76%	1,340.00	3.06%	1,690.00	3.86%

Table 19 continued

Surrounding surface type	Model	Min (16m)		Average(40m)		Max(118m)	
		Energy (MWh)	Efficiency	Energy (MWh)	Efficiency	Energy (MWh)	Efficiency
type 7	Abacus 5kW	0.37	0.85%	0.87	1.98%	1.77	4.04%
	Airdophin 1kW	0.15	1.76%	0.25	2.83%	0.43	4.95%
	Proven 15kW	4.52	3.44%	6.65	5.06%	10.53	8.01%
	Nordex 800kW	78.45	1.12%	104.17	1.49%	170.66	2.44%
	REPower 5MW	1,230.00	2.81%	1,370.00	3.13%	1,780.00	4.06%
type 8	Abacus 5kW	0.30	0.69%	0.80	1.82%	1.79	4.09%
	Airdophin 1kW	0.14	1.54%	0.24	2.73%	0.44	5.06%
	Proven 15kW	4.08	3.11%	6.34	4.82%	10.66	8.11%
	Nordex 800kW	73.27	1.05%	103.08	1.47%	176.59	2.52%
	REPower 5MW	1,250.00	2.85%	1,420.00	3.24%	1,870.00	4.27%
Total average efficiency of wind generator							3.31%

Table 19 continued

b. Unit cost

Capacity (kW)	Cost (KKRW)	Unit cost (KKRW/Kw)
3	18,120	6,040
5	25,120	5,024
6	36,240	6,040
8	43,240	5,405
9	54,360	6,040
10	50,240	5,024
12	72,480	6,040
15	75,360	5,024
16	86,480	5,405
18	108,720	6,040
20	100,480	5,024
24	129,720	5,405
25	125,600	5,024
27	163,080	6,040
30	150,720	5,024
50	251,200	5,024
80	432,400	5,405
Average of unit cost (KKR/kW)		5,472

Table 20 Unit cost of wind power generator in Korea

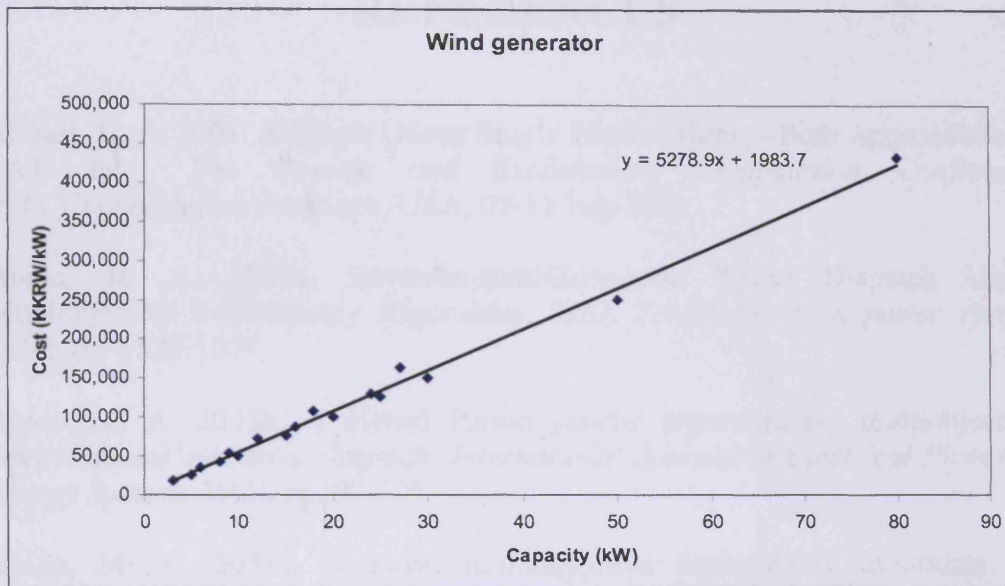


Figure 5 Graph of cost vs. capacity of wind power generator

10. Carbon

Conversion to CO ₂ (gross CV basis)		
Energy source	Units	Kg CO ₂ per unit
Natural gas	kWh	0.184

Table 21 Carbon factor from Carbon Trust (Carbon Trust 2008)

REFERENCES

Abbass, H. A. 2001. A Single Queen Single Worker Honey–Bees Approach to 3-SAT. In: *The Genetic and Evolutionary Computation Conference (GECCO2001)*. San Francisco, USA, 07-11 July 2001.

Abido, M. A. 2003a. Environmental/Economic Power Dispatch Using Multiobjective Evolutionary Algorithms. *IEEE Transactions on power system* 18(4), pp. 1529-1537.

Abido, M. A. 2003b. A niched Pareto genetic algorithm for multiobjective environmental/economic dispatch. *International Journal of Electrical Power & Energy Systems* 25(2), pp. 97-105.

Abido, M. A. 2003c. A novel multiobjective evolutionary algorithm for environmental/economic power dispatch. *Electric Power Systems Research* 65(1), pp. 71-81.

Abido, M. A. 2006. Multiobjective Evolutionary Algorithms for Electric Power Dispatch Problem. *Evolutionary computation on IEEE* 10(3), pp. 315-329.

Abido, M. A. 2007. *Multiobjective Particle Swarm for Environmental/Economic Dispatch Problem*. Proceedings of the 8th International Power Engineering Conference (IPEC 2007). Meritus Mandarin Singapore, 03-06 December 2007 pp. 1385-1390.

Abido, M. A. 2009. Multiobjective particle swarm optimization for environmental/economic dispatch problem. *Electric Power Systems Research* 79(7), pp. 1105-1113.

Ah King, R. T. F. and Rughooputh, H. C. S. and Deb, K. 2005. *Evolutionary Multi-objective Environmental/Economic Dispatch: Stochastic Versus Deterministic Approaches*. Proceedings of Evolutionary Multi-Criterion Optimization. Guanajuato, Mexico, 09-11 March 2005. Springer-Verlag, pp. 677-691.

Ah King, R. T. F. and Rughooputh, H. C. S. and Deb, K. 2006. *Stochastic Evolutionary Multiobjective Environmental/Economic Dispatch*. Proceedings of IEEE Congress on Evolutionary Computation 2006 (CEC 2006). Palo Alto, California, USA, 26-29 June 2006. pp. 946-953.

Aki, H. and Oyama, T. and Tsuji, K. 2003. Analysis of energy pricing in urban energy service systems considering a multi-objective problem of environmental and economic impact. *IEEE transactions on Power systems* 18(4), pp. 1275-1282.

Angus, D. 2007. *Population-based ant colony optimisation for multi-objective function optimisation*. Proceedings of the 3rd Australian conference on Progress in artificial life. Gold Coast, Australia, 04-06 December 2007. Springer-Verlag, pp. 232-244.

Baig, A. R. and Rashid, M. 2007. Honey bee foraging algorithm for multimodal and dynamic optimization problems. In: *Proceedings of the 9th annual conference on Genetic and evolutionary computation*. London, England. ACM

Bersini, H. and Varela, F. L. 1991. Hints for Adaptive Problem Solving Gleaned from Immune Networks. In: *The 1st Workshop on Parallel Problem Solving from Nature*. Dortmund, Germany, 01-03 October 1991. Springer-Verlag

Bhatt, V. and Friley, P. and Lee, J. 2010. Integrated energy and environmental systems analysis methodology for achieving low carbon cities. *Renewable and Sustainable energy* 2(031012), pp. 1-19.

Bitam, S. and Batouche, M. and Talbi, E.-g. 2010. *A survey on bee colony algorithms*. 2010 IEEE International Symposium on Parallel & Distributed Processing, Workshops and PhD Forum (IPDPSW). Atlanta, GA, USA. IEEE, pp. 1-8.

Bonabeau, E. and Dorigo, M. and Theraulaz, G. 1999. *Swarm Intelligence: From Natural to Artificial Systems*. New York: Oxford University Press, p. 320.

Born, F. J. 2001. *Aiding renewable energy integration through complementary demand-supply matching*. University of Strathclyde.

Born, F. J. and Clarke, J. A. and Johnstone, C. M. 2001a. Development and demonstration of a renewable energy based energy demand/supply decision support tool for the building design profession. In: *The 7th International IBPSA Conference*. Rio de Janeiro, Brazil, 13-15 August 2001.

Born, F. J. and Clarke, J. A. and Johnstone, C. M. and Smith, N. A. V. 2001b. Merit - An evaluation tool for 100% renewable energy provision. [Online]. Available at: <http://eprints.cdlr.strath.ac.uk/5926/>

Bouvy, C. and Lucas, K. 2007. Multicriterial optimisation of communal energy supply concepts. *Energy conversion and management* 48, pp. 2827-2835.

Bremermann, H. 1974. Chemotaxis and optimization. *Journal of the Franklin Institute* 297(5), pp. 397-404.

Cai, J. and Ma, X. and Li, Q. and Li, L. and Peng, H. 2009. A multi-objective chaotic particle swarm optimization for environmental/economic dispatch. *Energy Conversion and Management* 50, pp. 1318-1325.

Camazine, S. and Deneubourg, J. L. and Franks, N. and Sneyd, J. and Theraula, G. and Bonabeau, E. 2003. *Self-Organization in Biological Systems: (Princeton Studies in Complexity)*. Princeton: Princeton University Press, p. 560.

Carbon Trust. 2008. *Greenhouse Gas Conversion Factors* [Online]. Available at: http://www.carbontrust.co.uk/resource/conversion_factors/default.htm [Accessed: 01 July 2009].

Clarke, J. A. and Hong, J. and Johnstone, C. M. and Kelly, N. J. 2007. *The role of DSM+C to facilitate the integration of renewable energy and low carbon energy technologies within buildings*. Proceedings of Building Simulation 2007. Beijing, China, 03-06 September 2007. pp. 1653-1660.

Coello, C. A. and Lamont, G. L. and Van Veldhuizen, D. A. 2007. *Evolutionary Algorithms for Solving Multi-Objective Problems*. 2nd ed. Berlin, Heidelberg: Springer.

Colson, C. M. and Nehrir, M. H. and Wang, C. 2009. Ant Colony Optimization for Microgrid Multi-Objective Power Management. *IEEE*.

Corne, D. and Knowles, J. and Oates, M. 2000. The Pareto Envelope-Based Selection Algorithm for Multiobjective Optimization. *Parallel Problem Solving from Nature (PPSN VI)*. pp. 839-848.

Dalton, G. J. and Lockington, D. A. and Baldock, T. E. 2009a. Case study feasibility analysis of renewable energy supply options for small to medium-sized tourist accommodations. *Renewable Energy* 34(4), pp. 1134-1144.

Dalton, G. J. and Lockington, D. A. and Baldock, T. E. 2009b. Feasibility analysis of renewable energy supply options for a grid-connected large hotel. *Renewable Energy* 34(4), pp. 955-964.

DasGupta, D. 1998. *Artificial Immune Systems and Their Applications*. Springer-Verlag New York, Inc., p. 300.

- Day, A. R. and Ogumka, P. and Jones, P. G. and Dunsdon, A. 2009. The use of the planning system to encourage low carbon energy technologies in buildings. *Renewable Energy* 34(9), pp. 2016-2021.
- Deb, K. 1999. Multi-objective genetic algorithms: Problem difficulties and construction of test problems. *Evolutionary Computation* 7(3), pp. 205-230.
- Deb, K. 2001. *Multi-Objective Optimization using Evolutionary Algorithms*. West Sussex: John Wiley & Sons, Ltd.
- Deb, K. and Pratap, A. and Agarwal, S. and Meyarivan, T. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6, pp. 182-197.
- Dorigo, M. 1992. *Optimization, Learning and Natural Algorithms*. Politecnico di Milano, Italy.
- Dorigo, M. and Maniezzo, V. and Colorni, A. 1996. Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)* 26(1), pp. 29-41.
- Dorigo, M. and Stutzle, T. 2004. *Ant Colony Optimization*. Cambridge: MIT Press.
- Drias, H. and Sadeg, S. and Yahi, S. 2005. Cooperative Bees Swarm for Solving the Maximum Weighted Satisfiability Problem. In: Cabestany, J. et al. eds. *Computational Intelligence and Bioinspired Systems*. Vol. 3512. Springer Berlin / Heidelberg, pp. 318-325.
- Dunn, J. C. 1973. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics* 3(3), pp. 32-57.
- Eberhart, R. and Kennedy, J. 1995. *A new optimizer using particle swarm theory*. Proceedings of the 6th International Symposium on Micro Machine and Human Science 1995 (MHS '95). Nagoya, Japan, 04-06 October 1995. pp. 39-43.
- Energy and Environmental Analysis 2008a. *Technology Characterization: Fuel Cells*. Washington DC: Environmental Protection Agency.
- Energy and Environmental Analysis 2008b. *Technology Characterization: Gas Turbines*. Washington DC: Environmental Protection Agency.

Energy Technologies Unit 2007. *UK Biomass Strategy 2007 Working Paper 1 - Economic analysis of biomass energy*. Energy Technologies Unit.

Engelbrecht, A. P. 2006. *Fundamentals of Computational Swarm Intelligence*. John Wiley & Sons.

Fabrizio, E. and Corrado, V. and Filippi, M. 2010. A model to design and optimize multi-energy systems in buildings at the design concept stage. *Renewable Energy* 35(3), pp. 644-655.

Farmer, J. D. and Packard, N. H. and Perelson, A. S. 1986. The immune system, adaptation, and machine learning. *Phys. D* 2(1-3), pp. 187-204.

Favrat, D. and Pelet, X. 2007. Multiobjective Optimisation of integrated energy systems for remote communities considering economic and LCA factors. In: *International Conference Engineering for Sustainable Energy in developing countries*. Rio de Janeiro, Brazil, 15-18 August 2007.

Fonseca, C. M. and Fleming, P. J. 1993. *Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization*. Proceedings of the 5th International Conference on Genetic Algorithms. San Mateo, California, 17-22 July 1993. Morgan Kouffman Publishers, pp. 416-423.

Fonseca, C. M. and Fleming, P. J. 1998. Multiobjective optimization and multiple constraint handling with evolutionary algorithms-Part II: Application example. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on* 28(1), pp. 38-47.

Forrest, S. and Perelson, A. S. and Allen, L. and Cherukuri, R. 1994. *Self-nonsel discrimination in a computer*. Proceedings of IEEE Computer Society Symposium on Research in Security and Privacy 1994. 16-18 May 1994. pp. 202-212.

Freschi, F. and Coello, C. A. and Repetto, M. 2009. Multiobjective Optimization and Artificial Immune Systems: A Review. In: Mo, H. ed. *Handbook of Research on Artificial Immune Systems and Natural Computing: Applying Complex Adaptive Technologies*. Medical Information Science Reference, pp. 1-21.

Gao, J. and Wang, J. 2010. WBMOAIS: A novel artificial immune system for multiobjective optimization. *Computers & Operations Research* 37(1), pp. 50-61.

- Ghanbarzadeh, A. 2007. *THE BEES ALGORITHM A Novel Optimisation Tool*. Cardiff University.
- Goldberg, D. E. 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., p. 372.
- Guliashki, V. and Toshev, H. and Korsemov, C. 2009. Survey of Evolutionary Algorithms Used in Multiobjective Optimization. *Problems of Engineering Cybernetics and Robotics* 60, pp. 42-54.
- Guntsch, M. and Middendorf, M. 2002. Applying Population Based ACO to Dynamic Optimization Problems. In: Dorigo, M. et al. eds. *Ant Algorithms*. Vol. 2463. Springer Berlin / Heidelberg, pp. 97-104.
- Guzman, M. A. and Delgado, A. and Carvalho, J. D. 2010. A novel multiobjective optimization algorithm based on bacterial chemotaxis. *Eng. Appl. Artif. Intell.* 23(3), pp. 292-301.
- Haddad, O. and Afshar, A. and Marino, M. 2006. Honey-Bees Mating Optimization (HBMO) Algorithm: A New Heuristic Approach for Water Resources Optimization. *Water Resources Management* 20(5), pp. 661-680.
- Haj Darwish, A. 2009. *Enhanced Bees Algorithm with Fuzzy Logic and Kalman Filtering*. Cardiff University.
- Hawkes, A. and Leach, M. 2005. Impacts of temporal precision in optimisation modelling of micro-Combined Heat and Power. *Energy* 30(10), pp. 1759-1779.
- Henning, D. 1997. MODEST - An energy-system optimisation model applicable to local utilities and countries. *Energy* 22(12), pp. 1135 - 1150.
- Henning, D. and Amiri, S. and Holmgren, K. 2006. Modelling and optimisation of electricity, steam and district heating production for a local Swedish utility. *European Journal of Operational Research* 175(2), pp. 1224-1247.
- Hernandez-Aramburo, C. A. and Green, T. C. and Mugniot, N. 2005. Fuel consumption minimization of a microgrid. *Industry Applications, IEEE Transactions on* 41(3), pp. 673-681.
- Hobbs, B. F. 1995. Optimization methods for electric utility resource planning. *European Journal of Operational Research* 83(1), pp. 1-20.

Holland, J. H. 1992. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. The MIT Press.

Horn, J. and Nafpliotis, N. and Goldberg, D. E. 1994. *A niched Pareto genetic algorithm for multiobjective optimization*. Proceedings of the 1st IEEE Conference on Evolutionary Computation. 27-29 June 1994. pp. 82-87.

Jones, D. F. and Mirrazavi, S. K. and Tamiz, M. 2002. Multi-objective meta-heuristics: An overview of the current state-of-the-art. *European Journal of Operational Research* 137(1), pp. 1-9.

Kalman, R. E. 1960. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME-Journal of Basic Engineering* (82 (Series D)), pp. 35-45.

Karaboga, D. 2005. *An Idea Based on Honey Bee Swarm for Numerical Optimization*. Turkey: Erciyes University.

Karaboga, D. and Akay, B. 2009. A survey: algorithms simulating bee swarm intelligence. *Artificial Intelligence Review* 31(1), pp. 61-85.

Karaboga, D. and Basturk, B. 2008. On the performance of artificial bee colony (ABC) algorithm. *Applied Soft Computing* 8(1), pp. 687-697.

Karlsson, M. and Gebremedhin, A. and Klugman, S. and Henning, D. and Moshfegh, B. 2009. Regional energy system optimization - Potential for a regional heat market. *Applied Energy* 86(4), pp. 441-451.

Kennedy, J. and Eberhart, R. 1995. *Particle swarm optimization*. Proceedings of IEEE International Conference on Neural Networks. Perth, Australia, 27 November-01 December 1995. IEEE, pp. 1942-1948.

Kennedy, J. and Eberhart, R. C. and Shi, Y. 2001. *Swarm Intelligence*. San Francisco: Morgan Kaufmann.

Kephart, J. O. 1994. *A Biologically Inspired Immune System for Computers*. Artificial Life IV: Proceeding of the 4th International Workshop on the Synthesis and Simulation of Living Systems. Boston, USA, 06-08 July 1994. MIT Press, pp. 130-139.

Kim, I. Y. and de Weck, O. L. 2005. Adaptive weighted-sum method for bi-objective optimization: Pareto front generation. *Structural and Multidisciplinary Optimization* 29(2), pp. 149-158.

Kirkpatrick, S. and Gelatt, C. D., Jr. and Vecchi, M. P. 1983. Optimization by Simulated Annealing. *Science* 220(4598), pp. 671-680.

Kitamura, S. and Mori, K. and Shindo, S. and Izui, Y. and Ozaki, Y. 2005. *Multiobjective energy management system using modified MOPSO*. Proceedings of IEEE International Conference on Systems, Man and Cybernetics. Waikoloa, Hawaii, USA, 10-12 October 2005. pp. 3497-3503.

Klimes, J. and Pierucci, S. 2008. PRES 2007: Carbon footprint and emission minimisation, integration and management of energy sources, industrial application and case studies. *Energy* 33(10), pp. 1477-1479.

Knowles, J. D. and Corne, D. W. 2000. Approximating the Nondominated Front Using the Pareto Archived Evolution Strategy. *Evolutionary Computation* 8(2), pp. 149-172.

Konak, A. and Coit, D. and Smith, A. 2006. Multi-objective optimization using genetic algorithms: A tutorial. *Special Issue - Genetic Algorithms and Reliability* 91(9), pp. 992-1007.

Korea Energy Management Corporation. 2009. *Electric Rates Table* [Online]. Available at: <http://www.kepco.co.kr/eng/> [Accessed: 01 July 2009].

Koroneos, C. and Michailidis, M. and Moussiopoulos, N. 2004. Multi-objective optimization in energy systems: the case study of Lesvos Island, Greece. *Renewable and Sustainable Energy Reviews* 8(1), pp. 91-100.

Kursawe, F. 1991. A Variant of Evolution Strategies for Vector Optimization. In: *Proceedings of the 1st Workshop on Parallel Problem Solving from Nature*. Dortmund, Germany, 01-03 October 1991. Springer-Verlag

Lambert, T. and Gilman, P. and Lilienthal, P. 2004. *Micropower system modeling with HOMER* [Online]. Available at: <http://www.homerenergy.com/documentation.html> [Accessed: 10 January 2009].

Lee, J. Y. and Haj Darwish, A. 2008. *Multi-objective Environmental/Economic Dispatch Using the Bees Algorithm with Weighted Sum*. Proceedings of the EU-Korea Conference on Science and Technology (EKC2008). Heidelberg, Germany, 28-31 August 2008. Springer Berlin Heidelberg, pp. 267-274.

- Lee, K. Y. and El-Sharkawi, M. A. 2008. *Modern heuristic optimization techniques : theory and applications to power systems*. Hoboken, New Jersey: Wiley-IEEE Press.
- Leeprechanon, N. and Polratanasak, P. 2010. *Multiobjective Bees Algorithm with clustering technique for environmental/economic dispatch*. International Conference on Electrical Engineering/Electronics Computer Telecommunications and Information Technology (ECTI-CON 2010). Chiang Mai, Thailand, 19-21 May 2010. pp. 621-625.
- Leyland, G. 2002. *Multi-objective optimisation applied to industrial energy problems*. EPFL.
- Liu, Y. and Passino, K. M. 2002. Biomimicry of Social Foraging Bacteria for Distributed Optimization: Models, Principles, and Emergent Behaviors. *Journal of Optimization Theory and Applications* 115(3), pp. 603-628.
- Low, M. Y. H. and Chandramohan, M. and Chwee Seng, C. 2009. *Application of multi-objective bee colony optimization algorithm to Automated Red Teaming*. Proceedings of the Winter Simulation Conference (WSC). Austin, TX, USA, 13-16 December 2009. pp. 1798-1808.
- Lucic, P. and Teodorovic, D. 2001. *Bee system: Modeling Combinatorial Optimization Transportation Engineering Problems by Swarm Intelligence*. Triennial Symposium on Transportation Analysis (TRISTAN IV). Azores Islands, Portugal, 13-19 June 2001. pp. 441-445.
- Matteucci, M. *A Tutorial on Clustering Algorithms-Fuzzy C-Means Clustering* [Online]. Available at: http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/cmeans.html [Accessed: 12 October 2010].
- Mohamad Idris, R. and Khairuddin, A. and Mustafa, M. W. 2009. *A multi-objective Bees Algorithm for optimum allocation of FACTS devices for restructured power system*. Proceedings of TENCON 2009 - 2009 IEEE Region 10 Conference. Singapore, 23-26 November 2009. pp. 1-6.
- Mohamed, F. A. 2008. *Microgrid modelling and online management*. Helsinki University of Technology.
- Mohamed, F. A. and Koivo, H. N. 2007a. *MicroGrid Online Management and Balancing Using Multiobjective Optimization*. Proceedings of Power Tech 2007. Lausanne, Switzerland, 01-05 July 2007. IEEE, pp. 639-644.

Mohamed, F. A. and Koivo, H. N. 2007b. *System Modelling and Online Optimal Management of MicroGrid Using Multiobjective Optimization*. Proceedings of International Conference on Clean Electrical Power 2007 (ICCEP '07). Capri, Italy, 21-23 May 2007. IEEE, pp. 148-153.

Molyneaux, A. 2002. *A practical evolutionary method for the multi-objective optimisation of complex integrated energy systems including vehicle drivetrains*. EPFL.

Moore, D. S. 2000. *The Basic Practice of Statistics*. New York: W. H. Freeman and Company.

Mostaghim, S. and Teich, J. 2004. *Covering Pareto-optimal fronts by subswarms in multi-objective particle swarm optimization*. Proceedings of Congress on Evolutionary Computation 2004 (CEC2004). Portland, Oregon, USA, 20-23 June 2004. pp. 1404-1411.

Nakrani, S. and Tovey, C. 2004. *On Honey Bees and Dynamic Allocation in an Internet Server Colony*. The 2nd International Workshop on the Mathematics and Algorithms of Social Insects. Georgia, USA. pp. 115-122.

Ngatchou, P. and Anahita, Z. and El-Sharkawi, M. A. 2005. *Pareto Multi Objective Optimization*. Proceedings of the 13th International Conference on Intelligent Systems Application to Power Systems. 06-10 November 2005. pp. 84-91.

Olague, G. and Puente, C. 2006. Parisian evolution with honeybees for three-dimensional reconstruction. In: *Proceedings of the 8th annual conference on Genetic and evolutionary computation*. Seattle, Washington, USA. ACM

Oldroyd, B. P. and Wongsiri, S. 2006. *Asian Honey Bees: Biology, Conservation, and Human Interactions*. Cambridge, Massachusetts and London, England: Harvard University Press p. 360.

Omkar, S. N. and Mudigere, D. and Naik, G. N. and Gopalakrishnan, S. 2008. Vector evaluated particle swarm optimization (VEPSO) for multi-objective design optimization of composite structures. *Computers & Structures* 86(1-2), pp. 1-14.

Omkar, S. N. and Senthilnath, J. and Rahul, K. and Naik, G. N. and Gopalakrishnan, S. 2011. Artificial Bee Colony (ABC) for multi-objective design optimization of composite structures. *Applied Soft Computing* 11(1), pp. 489-499.

Ooka, R. and Komamura, K. 2009. Optimal design method for building energy systems using genetic algorithms. *Building and Environment* 44(7), pp. 1538-1544.

Østergaard, P. A. 2009. Reviewing optimisation criteria for energy systems analyses of renewable energy integration. *Energy* 34(9), pp. 1236-1245.

Pareto, V. 1906. *Manuale di Economia Politica*. Milano, Italy: Societa Editrice Libreria.

Park, H. and Kwak, N. S. and Lee, J. 2009. A method of multiobjective optimization using a genetic algorithm and an artificial immune system. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* 223(5), pp. 1243-1252.

Parsopoulos, K. E. and Vrahatis, M. N. 2002a. *Particle swarm optimization method in multiobjective problems*. Proceedings of the 2002 ACM symposium on Applied computing. Madrid, Spain, 11-14 March 2002. ACM, pp. 603-607.

Parsopoulos, K. E. and Vrahatis, M. N. 2002b. Recent approaches to global optimization problems through Particle Swarm Optimization. *Natural Computing: an international journal* 1(2-3), pp. 235-306.

Passino, K. 2002. Biomimicry of bacterial foraging for distributed optimization and control. *IEEE Control Systems Magazine* 22(3), pp. 52-67.

Passino, K. M. and Seeley, T. D. and Visscher, P. K. 2008. Swarm cognition in honey bees. *Behav Ecol Sociobiol* 62(3), pp. 401-414.

Pelet, X. and Favrat, D. and Leyland, G. 2005. Multiobjective optimisation of integrated energy systems for remote communities considering economics and CO2 emissions. *International Journal of Thermal Sciences* 44(12), pp. 1180-1189.

Pham, D. T. and Afify, A. and Koc, E. 2007a. *Manufacturing cell formation using the bees algorithm*. The 3rd Virtual International Conference on Innovative Production Machines and Systems (IPROMS 2007). Whittles, Dunbeath, Scotland, pp. 523-528.

Pham, D. T. and Al-Jabbouli, H. and Mahmuddin, M. and Otri, S. and Darwish, A. H. 2008a. *Application of the Bees Algorithm to Fuzzy Clustering*. The 4th Virtual International Conference on Innovative Production Machines and Systems (IPROMS2008). Whittles, Dunbeath, Scotland, pp. 404-408.

Pham, D. T. and Castellani, M. 2009. The Bees Algorithm: modelling foraging behaviour to solve continuous optimization problems. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* 223(12), pp. 2919-2938.

Pham, D. T. and Castellani, M. and Fahmy, A. A. 2008b. *Learning the inverse kinematics of a robot manipulator using the Bees Algorithm*. Industrial Informatics, 2008. INDIN 2008. 6th IEEE International Conference on. Daejeon, Korea, 13-16 July 2008. pp. 493-498.

Pham, D. T. and Castellani, M. and Ghanbarzadeh, A. 2007b. *Preliminary design using the bees algorithm*. The 8th international conference on laser metrology, CMM and machine tool performance (LAM DAMAP). Cardiff, UK, 25-28 June 2007. Euspen, pp. 420– 429.

Pham, D. T. and Castellani, M. and Sholehdol, M. and Ghanbarzadeh, A. 2008c. The Bees Algorithm and Mechanical Design Optimisation. In: *The 5th International Conference on Informatics in Control, Automation and Robotics (ICINCO)*. Funchal, Portugal, 11-15 May 2008.

Pham, D. T. and Darwish, A. H. and Eldukhri, E. E. 2009a. Optimisation of a fuzzy logic controller using the Bees Algorithm. *International Journal of Computer Aided Engineering and Technology* 1(2), pp. 250 - 264.

Pham, D. T. and Ghanbarzadeh, A. 2007. *Multi-Objective Optimisation using the Bees Algorithm*. The 3rd Virtual International Conference on Innovative Production Machines and Systems (IPROMS2007). 02-13 July 2007. Whittles, pp. 529-533.

Pham, D. T. and Ghanbarzadeh, A. and Koç, E. and Otri, S. and Rahim, S. and Zaidi, M. 2005. *The Bees Algorithm - Technical Report*. Cardiff: Manufacturing Engineering Centre, Cardiff University.

Pham, D. T. and Ghanbarzadeh, A. and Koç, E. and Otri, S. and Rahim, S. and Zaidi, M. 2006a. *The Bees Algorithm - A Novel Tool for Complex Optimisation Problems*. Proceedings of the 2nd Virtual International Conference on Intelligent Production Machines and Systems (IPROMS2006). 03-14 July 2006. Elsevier, pp. 454-459.

Pham, D. T. and Ghanbarzadeh, A. and Otri, S. and Koç, E. 2009b. Optimal design of mechanical components using the Bees Algorithm. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* 223(5), pp. 1051-1056.

Pham, D. T. and Karaboga, D. 1999. *Intelligent Optimisation Techniques: Genetic Algorithms, Tabu Search, Simulated Annealing and Neural Networks*. London, Heidelberg and New York: Springer Verlag.

Pham, D. T. and Koc, E. and Kalyoncu, M. and Tinkir, M. 2008d. *A Hierarchical PID Controller Design for a Flexible Link Robot Manipulator Using the Bees Algorithm*. The 6th International Symposium on Intelligent and Manufacturing Systems. Sakarya, Turkey. pp. 757-765.

Pham, D. T. and Koc, E. and Lee, J. Y. and Phrueksanant, J. 2007c. *Using the Bees Algorithm to schedule jobs for a machine*. The 8th International Conference on Laser Metrology, CMM and Machine tool performance (LAM DAMAP). Cardiff, UK. Euspen, pp. 430-439.

Pham, D. T. and Mahmuddin, M. and Otri, S. and Al-Jabbouli, H. 2007d. *Application of the Bees Algorithm to the Selection Features for Manufacturing Data*. The 3rd Virtual International Conference on Innovative Production Machines and Systems (IPROMS2007). pp. 517-522.

Pham, D. T. and Muhamad, Z. and Mahmuddin, M. and Ghanbarzadeh, A. and Koc, E. and Otri, S. 2007e. *Using the Bees Algorithm to optimise a Support Vector Machine for Wood Defect Classification*. The 3rd Virtual International Conference on Innovative Production Machines and Systems (IPROMS2007). Whittles, Dunbeath, Scotland, pp. 540-545.

Pham, D. T. and Otri, S. and Afify, A. and Mahmuddin, M. and Al-Jabbouli, H. 2007f. *Data Clustering using the Bees Algorithm*. In: *The 40th CIRP International Manufacturing Systems Seminar*. Liverpool, UK, 30 May-1 June 2007.

Pham, D. T. and Otri, S. and Darwish, A. H. 2007g. *Application of the Bees Algorithm to PCB assembly optimisation*. The 3rd Virtual International Conference on Innovative Production Machines and Systems (IPROMS2007). Whittles, Dunbeath, Scotland, pp. 511-516.

Pham, D. T. and Pham, Q. T. and Ghanbarzadeh, A. and Castellani, M. 2008e. *Dynamic Optimisation of Chemical Engineering Processes Using the Bees Algorithm*. the 17th World Congress The International Federation of Automatic Control. Seoul, Korea, 06-11 July 2008. pp. 6100-6105.

Pham, D. T. and Sholedolu, M. 2008. *Using a Hybrid PSO-Bees Algorithm to train Neural Networks for Wood Defect Classification*. The 4th Virtual International Conference on Innovative Production Machines and Systems (IPROMS2008). 01-14 July 2008. Whittles, pp. 385-390.

Pham, D. T. and Soroka, A. J. and Ghanbarzadeh, A. and Koc, E. and Otri, S. and Packianather, M. 2006b. *Optimising Neural Networks for Identification of Wood Defects Using the Bees Algorithm*. IEEE International Conference on Industrial Informatics. Singapore, 16-18 August 2006. pp. 1346-1351.

Pham, D. T. and Soroka, A. J. and Koc, E. and Ghanbarzadeh, A. and Otri, S. 2007h. *Some applications of the bees algorithm in engineering design and manufacture*. International conference on manufacturing automation (ICMA 2007). Singapore, 28-30 May 2007. pp. 782-794.

Poloni, C. 1995. *Hybrid GA for multiobjective aerodynamic shape optimisation*. Proceedings of Genetic Algorithms in Engineering and Computer Science. Chichester, U.K. Wiley, pp. 397-414.

Quijano, N. and Passino, K. M. 2010. Honey bee social foraging algorithms for resource allocation: Theory and application. *Engineering Applications of Artificial Intelligence* 23(6), pp. 845-861.

Ray, T. and Tai, K. and Seow, C. 2001. An evolutionary algorithm for multiobjective optimisation. *Eng. Optim.* 33(3), pp. 399-424.

Russell, S. and Norvig, P. 2004. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, p. 1132.

Sahran, S. 2007. *Application of Spiking Neural Networks and the Bees Algorithm to control chart pattern recognition*. Cardiff University.

Sato, T. and Hagiwara, M. 1997. *Bee System: finding solution by a concentrated search*. Systems, Man, and Cybernetics, 1997. 'Computational Cybernetics and Simulation', 1997 IEEE International Conference on. pp. 3954-3959 vol.3954.

Sayadi, F. and Ismail, M. and Misran, N. and Jumari, K. 2009. Multi-Objective Optimization Using the Bees Algorithm in Time-Varying Channel for MIMO MC-CDMA Systems. *European Journal of Scientific Research* 33(3), pp. 411-428.

Schaffer, J. D. 1984. *Some experiments in machine learning using vector evaluated genetic algorithms (artificial intelligence, optimization, adaptation, pattern recognition)*. Vanderbilt University.

Schaffer, J. D. 1985. Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. In: *The 1st International Conference on Genetic Algorithms*. Pittsburgh, PA, USA, July 1985. Lawrence Erlbaum Associates

Scheaffer, R. L. and McClave, J. T. 1982. *Statistics for Engineers*. PWS Publishers.

Seeley, T. D. 1996. *The Wisdom of the Hive: The Social Physiology of Honey Bee Colonies*. Cambridge: Massachusetts: Harvard University Press, p. 318.

Seshadri, A. 2006. *A fast elitist multiobjective genetic algorithm: NSGA-II* [Online]. Available at: <http://www.mathworks.com/matlabcentral/fileexchange/10429-nsga-ii-a-multi-objective-optimization-algorithm> [Accessed: 01 April 2008].

Seshadri, A. 2006 *Multi-objective optimisation using evolutionary algorithms (MOEA)* [Online]. Available at: <http://www.mathworks.com/matlabcentral/fileexchange/10351> [Accessed: 01 April 2008].

Srinivas, N. and Deb, K. 1994. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation* 2, pp. 221-248.

Stadler, W. 1979. A survey of multicriteria optimization or the vector maximum problem, part I: 1776–1960. *Journal of Optimization Theory and Applications* 29(1), pp. 1-52.

Stadler, W. 1984. Multiple Criteria Decision Making - Past Decade and Future Trends. In: Zeleny, M. ed. *Applications of Multicriteria Optimization in Engineering and the Sciences (A Survey)*. Greenwich.

Sugihara, H. and Komoto, J. and Tsuji, K. 2004. A Multi-objective Optimization Model for Determining Urban Energy Systems under Integrated Energy Service in a Specific Area. *Electrical Engineering in Japan* 147(3), pp. 20-31.

Tan, K. C. and Goh, C. K. and Mamun, A. A. and Ei, E. Z. 2008. An evolutionary artificial immune system for multi-objective optimization. *European Journal of Operational Research* 187(2), pp. 371-392.

Tanaka, M. and Watanabe, H. and Furukawa, Y. and Tanino, T. 1995. *GA-based decision support system for multicriteria optimization*. Proceedings of IEEE International Conference on Systems, Man and Cybernetics: Intelligent Systems for the 21st Century. 22-25 October 1995. pp. 1556-1561.

Teodorovic, D. and Dell'Orco, M. 2005. Bee colony optimization - A cooperative learning approach to complex transportation problems. *Advanced OR and AI Methods in Transportation*, pp. 51-60.

Tippachon, W. and Rerkpreedapong, D. 2009. Multiobjective optimal placement of switches and protective devices in electric power distribution systems using ant colony optimization. *Electric Power Systems Research* 79(7), pp. 1171-1178.

Tovey, C. A. 2004. *The Honey Bee Algorithm: A Biologically Inspired Approach to Internet Server Optimization*. Engineering Enterprise Magazine. pp. 13-15.

US Environmental Protection Agency 2008. *Catalog of CHP Technologies*.

Van den Bergh, F. 2006. *An analysis of Particle Swarm Optimizers*. University of Pretoria.

Wang, L. and Singh, C. 2006. *Reserve-Constrained Multiarea Environmental/Economic Dispatch Using Enhanced Particle Swarm Optimization*. Proceedings of Systems and Information Engineering Design Symposium. Virginia, USA, 28 April 2006. IEEE, pp. 96-100.

Wang, L. and Singh, C. 2007. Environmental/economic power dispatch using a fuzzified multi-objective particle swarm optimization algorithm. *Electric Power Systems Research* 77(12), pp. 1654-1664.

Wedde, H. and Farooq, M. and Zhang, Y. 2004. BeeHive: An Efficient Fault-Tolerant Routing Algorithm Inspired by Honey Bee Behavior. *Ant Colony, Optimization and Swarm Intelligence*. pp. 83-94.

Wedde, H. F. and Farooq, M. and Pannenbaecker, T. and Vogel, B. and Mueller, C. and Meth, J. and Jeruschkat, R. 2005. *BeeAdHoc: an energy efficient routing algorithm for mobile ad hoc networks inspired by bee behavior*. Proceedings of the 2005 conference on Genetic and evolutionary computation (GECCO'05). Washington DC, USA, 25-29 June 2005. ACM, pp. 153-160.

- Williamson, T. J. 1995. *A Confirmation Technique for Thermal Performance Simulation Models*. Proceedings of Building Simulation Conference and Exhibition 1995. Madison, Wisconsin, USA, 14-16 August 1995. IBPSA: International Building Performance Simulation Association, pp. 268-275.
- Xu, M. and Zhan, Y. and Cao, J. and Liu, Y. and Wang, X. and Liang, G. and Huang, M. 2007. A Beehive Algorithm Based QoS Unicast Routing Scheme with ABC Supported. *Advanced Parallel Processing Technologies*. Vol. 4847. Springer Berlin / Heidelberg, pp. 450-459.
- Yang, X.-S. 2005. Engineering Optimizations via Nature-Inspired Virtual Bee Algorithms. In: Mira, J. and Alvarez, J. eds. *Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach*. Vol. 3562. Springer Berlin / Heidelberg, pp. 317-323.
- Yokoyama, R. and Bae, S. H. and Morita, T. and Sasaki, H. 1988. Multiobjective optimal generation dispatch based on probability security criteria. *Power Systems, IEEE Transactions on* 3(1), pp. 317-324.
- Yoo, J. and Hajela, P. 1999. Immune network simulations in multicriterion design. *Structural and Multidisciplinary Optimization* 18(2), pp. 85-94.
- Zadeh, L. A. 1965. Fuzzy sets. *Information and Control* 8(3), pp. 338-353.
- Zeng, F. and Decraene, J. and Low, M. Y. H. and Hingston, P. and Cai, W. and Suiping, Z. and Chandramohan, M. 2010. Autonomous Bee Colony Optimization for Multiobjective Function. In: *2010 IEEE World Congress on Computational Intelligence (WCCI2010)*. Barcelona, Spain, 18-23 July.
- Zhu, J. and Wang, X. and Shi, T. and Wu, Y. 2008. Low-carbon City Construction Based on Large-scale Thermoelectric Cogeneration. In: *44th ISOCARP Congress 2008: Low-carbon City Construction*. Dalian, China.
- Zhu, L. and Hurt, R. and Correa, D. and Boehm, R. 2009. Comprehensive energy and economic analyses on a zero energy house versus a conventional house. *Energy* 34(9), pp. 1043-1053.
- Zitzler, E. and Deb, K. and Thiele, L. 2000. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. *Evol. Comput.* 8(2), pp. 173-195.
- Zitzler, E. and Thiele, L. 1999. Multiobjective Evolutionary Algorithms: A comparative case study and the Strength Pareto Approach. *IEEE Transactions on Evolutionary Computation* 3(4), pp. 257-271.