

EVOLUTIONARY WINNER DETERMINATION IN ADVANCED
COMBINATORIAL REVERSE AUCTIONS

A Thesis

Submitted to the Faculty of Graduate Studies and Research

In Partial Fulfillment of the Requirements

For the Degree of

Doctor of Philosophy

in

Computer Science

University of Regina

By

Shubhashis Kumar Shil

Regina, Saskatchewan

September 2017

Copyright © 2017: S. K. Shil

UNIVERSITY OF REGINA
FACULTY OF GRADUATE STUDIES AND RESEARCH
SUPERVISORY AND EXAMINING COMMITTEE

Shubhashis Kumar Shil, candidate for the degree of Doctor of Philosophy in Computer Science, has presented a thesis titled, ***Evolutionary Winner Determination in Advanced Combinatorial Reverse Auctions***, in an oral examination held on September 5, 2017. The following committee members have found the thesis acceptable in form and content, and that the candidate demonstrated satisfactory knowledge of the subject material.

External Examiner: *Dr. Reda Alhajj, University of Calgary

Supervisor: Dr. Samira Sadaoui, Department of Computer Science

Committee Member: Dr. Lisa Fan, Department of Computer Science

Committee Member: Dr. Malek Mouhoub, Department of Computer Science

Committee Member: Dr. Mohamed El-Darieby,
Faculty of Engineering and Applied Science

Chair of Defense: Dr. Andrei Volodin, Department of Mathematics & Statistics

*via SKYPE

Abstract

Traditional Combinatorial Reverse Auctions (CRAs) are already very hard problems to solve. By considering the multiplicity of units, attributes and objectives of items; the complexity of CRAs increases. Winner Determination (WD) is one of the main challenges of CRAs, and which has been shown to be NP-complete. Researchers limit the auction features because of the issue of time efficiency.

We address several types of advanced CRAs by varying the auction parameters. We tackle the combinatorial procurement problem instances using Genetic Algorithms (GAs). In the first instance, we consider multiple units, single attribute and objective. To this end, we propose a modified crossover operator and two routines. Our proposed WD method is capable of finding the winner(s) with a minimum procurement cost and an efficient processing time. To validate our GA-based method, we conduct several experiments and the results clearly demonstrate the good-time performance and the high quality of the solution. In the second problem, we tackle CRAs with two attributes, single objective and multiple units and rounds. We make this problem more interesting by considering all-units discounts and the availability of sellers' stock. To evaluate our proposed method, we conduct several experiments. In the next problem instance, we include additional constraints, such as seller stocks and discount rate. In this part, we perform a comparative study of several exact and evolutionary techniques addressing different types of CRAs. In particular, we show that our technique based on GAs outperforms some other methods in terms of time efficiency. Lastly, we address CRAs with multiple units, attributes, objectives and rounds. We define optimization approach

based on GAs that we integrate with our own variants of diversity and elitism strategies to greatly improve the solution quality. We conduct a case study as well as simulated testing to illustrate the importance of the diversity and elitism schemes. We also validate the proposed WD method through simulated experiments by generating large instances of our CRA problem.

Moreover, we apply our WD approach to a real-life electricity application based on renewable energy sources. The option for public utilities to organize electricity CRAs to purchase the needed electricity from other power suppliers is a new concept. For this purpose, we develop a constrained CRA to procure power from diverse sources including residents and plants. In our CRA, subject to various trading constraints, an item denotes a time slot that has two conflicting attributes, energy volume and price. To secure electricity, we design our auction with two bidding rounds: the first one is for variable energy suppliers and the second one for other sources, like controllable load and other renewable technologies. Our CRA leads to a complex WD problem. We view this problem as a resource allocation optimization that we solve with multi-objective GAs in order to find the best trade-off solution that lowers the price and increases the energy. This solution consists of multiple winning suppliers, their prices, power volumes and schedules. We validate our WD approach based on simulated data by generating large instances of our multi-objective constrained auction problem. The goal of the experiments is to assess the time-efficiency of our WD method and its significant superiority to well-known heuristic and exact WD techniques. Furthermore, we implement two exact algorithms to solve our complex procurement problem in order to evaluate the accuracy of our WD method i.e. how near to optimal is the solution.

Acknowledgments

My first debt of gratitude goes to my supervisor Dr. Samira Sadaoui for providing me with constant support during my PhD studies. I would like to express my sincere thanks for her valuable guidance, financial assistance and constant encouragement. Her enthusiasm, patience and diverse knowledge helped and enlightened me on many occasions. She not only provided me with fruitful ideas but also gave me the freedom to think independently. I feel proud to be her research student and cannot imagine a better supervisor.

I acknowledge the Faculty of Graduate Studies and Research for providing me with the financial means, in the form of scholarships, to complete this study.

I also thank the committee members who read my thesis and providing suggestions and useful comments for improvement.

I would like to take this opportunity to thank every member of the Department of Computer Science who has helped me throughout my studies.

Dedication

I would like to dedicate my thesis and extend my deepest gratitude to my beloved parents and wife for their unconditional love and support throughout my entire life.

Table of Contents

Abstract	i
Acknowledgments	iii
Dedication	iv
Table of Contents	v
List of Tables	x
List of Figures	xii
Chapter 1 Introduction	1
1.1 Context.....	1
1.2 Problem Formulation and Motivations.....	3
1.2.1 Auction Mechanism.....	3
1.2.2 Winner Determination.....	4
1.3 Solution Approach.....	8
1.4 System Validation.....	12
1.5 A Summary.....	12
1.6 Contributions.....	13
1.7 Thesis Organization.....	16
Chapter 2 Related Work	17
2.1 Features of Auctions	17
2.2 Preference and Constraint Elicitation.....	21
2.3 Electricity CRAs.....	24
2.3.1 Traditional CRAs.....	24

2.3.2	Advanced CRAs.....	25
Chapter 3	MU-SA-SO-SR CRA	27
3.1	Summary.....	27
3.2	Introduction.....	27
3.3	Proposed Method.....	30
3.4	Experimentation.....	39
3.5	Conclusion.....	46
Chapter 4	MU-TA-SO-MR CRA Part-1	48
4.1	Summary.....	48
4.2	Introduction.....	48
4.3	Problem Statement.....	51
4.4	Proposed Method.....	54
4.5	Experimentation.....	63
4.5.1	Experiment 1: Parameter Tuning.....	63
4.5.2	Experiment 2: Statistical Analysis.....	64
4.5.3	Experiment 3: Number of Generations.....	65
4.5.4	Experiment 4: Number of Sellers and Items.....	65
4.6	Conclusion.....	68
Chapter 5	MU-TA-SO-MR CRA Part-2	70
5.1	Summary.....	70
5.2	Introduction.....	71
5.3	GAMICRA for Winner Determination.....	74
5.4	Experimentation.....	77

5.4.1	Experiment 1: Comparison with IAC, AC and EAB.....	78
5.4.2	Experiment 2: Comparison with Branch and Bound.....	78
5.4.3	Experiment 3: Comparison with the Exact Algorithm.....	79
5.5	Conclusion.....	81
Chapter 6	MU-MA-MO-MR CRA	83
6.1	Summary.....	83
6.2	Introduction.....	84
6.2.1	Research Scope.....	84
6.2.2	Multi-Objective Combinatorial Reverse Auctions.....	85
6.2.3	Winner Determination.....	86
6.3	State-Of-The-Art Combinatorial Reverse Auctions.....	87
6.4	Design of Multi-Objective CRAs.....	90
6.4.1	Auction Features.....	90
6.4.2	Trading Requirements.....	91
6.4.3	Valid Bids.....	92
6.4.4	Winning Solution.....	93
6.4.5	Buyer's Gain.....	94
6.5	Winner Determination Approach.....	95
6.5.1	Multi-Objective Optimization Problem.....	95
6.5.2	Solution Representation and Fitness Assignment.....	96
6.5.3	Solution Production.....	98
6.5.4	Winner Determination Algorithm.....	101
6.5.4.1	Winner Determination.....	103

6.5.4.2	Winner Determination with Diversity.....	103
6.5.4.3	Winner Determination with Diversity and Elitism.....	104
6.5.4.4	Optimization Process.....	105
6.6	A Case Study.....	105
6.6.1	Requirements Elicitation and Bid Submission.....	105
6.6.2	First Generation.....	108
6.6.3	Last Generation.....	112
6.7	Performance Evaluation.....	115
6.7.1	Trade-Off between Convergence and Diversity.....	116
6.7.2	Solution Quality Assessment.....	116
6.7.3	Computational Expense of WDDE.....	118
6.7.4	Statistical Analysis of WDDE.....	120
6.7.5	Feature Comparison of WDDE with Heuristic WD Techniques	121
6.7.6	Computational Expense Comparison of WDDE with Heuristic WD Techniques.....	122
6.7.7	Computational Expense Comparison of WDDE with an Exact WD Technique.....	123
6.8	Conclusion.....	124
Chapter 7	SU-TA-TO-TR CRA	126
7.1	Summary.....	126
7.2	An Overview.....	127
7.2.1	Problem.....	127
7.2.2	Auction Features.....	128

7.2.3	Winner Determination.....	130
7.2.4	Auction Validation.....	131
7.3	Auctioning Electricity from Diverse Sources.....	131
7.3.1	Auction Demand.....	132
7.3.2	Supplier Registration and Constraints.....	132
7.3.3	Supplier Bids with Two Rounds.....	133
7.3.4	Winner Determination.....	133
7.3.5	Trade Settlement	137
7.4	A Case Study.....	137
7.5	Validation and Comparison.....	142
7.6	Related Work.....	151
7.6.1	Traditional Auctions.....	152
7.6.2	Advanced Auctions.....	153
7.7	Conclusion.....	155
Chapter 8	Conclusion and Future Work	158
8.1	Conclusion.....	158
8.2	Future Work.....	160
References		162
Appendix A	Figures	176
Appendix B	Graphical User Interfaces	179
Appendix C	Algorithms	181
Appendix D	Research Papers	183

List of Tables

1.1	Types of CRAs.....	4
3.1	Bid Price Generation.....	32
3.2	Initial Chromosomes.....	32
4.1	Sellers' Minimum Price (MP) and Delivery Rate (DR).....	53
4.2	Sellers' All-Units Discount on Bid Price.....	53
4.3	Sellers' All-Units Discount on Delivery Rate.....	54
4.4	Bid Price Generation.....	57
4.5	Initial Chromosomes.....	58
4.6	Bid Result.....	62
5.1	Problem Formulation for GAMICRA.....	75
5.2	The Comparison of GAMICRA, IAC, AC and EAB.....	78
5.3	The Comparison of GAMICRA and Branch and Bound.....	79
5.4	Comparison of GAMICRA and EA when Varying the Number of Sellers	80
5.5	Comparison of GAMICRA and EA when Varying the Total Number Items Instances.....	80
6.1	Fitness Assignment Functions.....	97
6.2	Description of Genetic Algorithm Operations.....	99
6.3	Description of Diversity Mechanism.....	100
6.4	Description of Elitism Mechanism.....	101
6.5	Buyer and Seller Requirements.....	106
6.6	Valid offers.....	108

6.7	Initial Solutions.....	110
6.8	Potential Solutions of WD, WDD and WDDE.....	111
6.9	Winning Solutions of WD, WDD and WDDE.....	112
6.10	Winning Solutions at First Round.....	113
6.11	Breakdown of Winning Solution of WDDE.....	114
6.12	Item Description.....	115
6.13	Configuration of Items and Sellers.....	118
6.14	Feature Comparison.....	122
6.15	Computational Expense Comparison.....	123
6.16	Computational Expense Comparison of WDDE with Branch & Bound....	123
7.1	Buyer Requirements.....	138
7.2	Constraints of Wind and Solar Suppliers.....	139
7.3	Valid Bids of Wind and Solar Suppliers.....	139
7.4	Candidate and Winning Solutions for First Round.....	140
7.5	Constraints and Valid Bids of Hydro and Battery.....	141
7.6	Simulated Datasets.....	142
7.7	Computational Expense Comparison.....	149
7.8	Comparison with Exact Algorithms.....	150

List of Figures

1.1	General WD Scenario for CRAs.....	11
1.2	Thesis Summary	13
3.1	An Example in CRAs.....	33
3.2	Algorithm for GAMICRA.....	35
3.3	RemoveRedundancy Pseudo Code.....	36
3.4	RemoveEmptiness Pseudo Code.....	36
3.5	Modified Two-Point Crossover and Mutation Operation.....	39
3.6(a)	Running Time and Procurement Cost Performances (processing time vs number of generations).....	41
3.6(b)	Running Time and Procurement Cost Performances (bid price vs number of generations).....	42
3.6(c)	Running Time and Procurement Cost Performances (processing time vs number of sellers).....	43
3.6(d)	Running Time and Procurement Cost Performances (processing time vs number of items).....	44
3.6(e)	Running Time and Procurement Cost Performances (processing time vs number of instances).....	45
3.6(f)	Running Time and Procurement Cost Performances (bid price vs number of generations).....	46
4.1	A Sample Scenario in CRAs.....	52
4.2	Proposed Solving Algorithm.....	55

4.3	BidGenerator Function.....	59
4.4	ChromosomeGenerator Function.....	60
4.5	NewChromosomeGenerator Function.....	60
4.6	Modified-Two-Point Crossover and Mutation.....	62
4.7	Statistical Analysis of the Proposed Method.....	64
4.8(a)	Bid Price vs Number of Generations.....	66
4.8(b)	Computation Time vs Number of Generations.....	67
4.9(a)	Computation Time vs Number of Sellers.....	67
4.9(b)	Computation Time vs Number of Items.....	68
5.1	EA and GAMICRA pseudo code.....	77
6.1	Multi-Objective CRAs.....	94
6.2	Generic Winner Determination.....	102
6.3	WDD.....	103
6.4	WDDE.....	104
6.5	Schematic of Winner Determination Approach.....	107
6.6	Solution Representation.....	108
6.7	Modified Two-Point Crossover.....	109
6.8	Swap Mutation.....	110
6.9	Convergence and Diversity Trade-Off.....	117
6.10	Solution Quality Comparison.....	118
6.11	Computational Expense of WDDE by Varying Number of Attributes and Total Number of Units.....	119
6.12	Computational Expense of WDDE by Varying Number of Sellers.....	120

6.13	Statistical Analysis of WDDE.....	121
7.1	WD Algorithm for Electricity CRA.....	135
7.2(a)	Statistical Analysis of WD (Round 1).....	145
7.2(b)	Statistical Analysis of WD (Round 2).....	145
7.3(a)	Computational Time of WD by Varying Number of Items (Round 1).....	146
7.3(b)	Computational Time of WD by Varying Number of Items (Round 2).....	147
7.4(a)	Computational Time of WD by Varying Number of Sellers (Round 1).....	147
7.4(b)	Computational Time of WD by Varying Number of Sellers (Round 2).....	148
A1	Convergence and Diversity of WD method integrated with Diversity and Elitism Techniques.....	176
A2	Convergence WD method without Diversity and Elitism Techniques.....	177
A3	Diversity of WD method integrated with Diversity Technique.....	177
A4	3-tier Software Architecture.....	178
B1	Buyer Requirement GUI.....	179
B2	Case Study GUI.....	180
C1	Brute Force and Branch and Bound pseudo code.....	181

Chapter 1

Introduction

1.1 Context

A. Combinatorial Reverse Auctions

An auction is a market place where bidders compete for item(s). In traditional auctions, each item is auctioned separately, which leads to an inefficient resource allocation and trading time [6, 58]. To improve the efficiency of bid allocation, combinatorial auctions have been proposed by allowing participants to bid on multiple items [53, 58]. This type of auctions provides a combinatorial allocation that minimizes the procurement cost and processing time [6, 58, 91]. Combinatorial auctions have been widely adopted to solve many real-world applications [21], such as supply chain management [84], resource allocation with real-time constraints [6], computer grids [19] and sensor management [53]. A combinatorial auction problem has been demonstrated as a winner determination problem [29]. Determining the winners is still one of the main challenges of combinatorial auctions [6].

In our research, we are particularly interested in Combinatorial Reverse Auctions (CRAs) that can be of several types according to the multiplicity of units, attributes, objectives and rounds, such as (1) single unit and single attribute of each item, or (2) multiple units and multiple attributes of each item. Unlike forward combinatorial auctions, fewer research works were dedicated to CRAs.

B. Applications of CRAs

Traditional CRAs (multiple items, single unit and single attribute) have been effectively adopted in various practical applications, including government contract auctions, corporate procurement systems, bus and truckload transportation routes, airport runway arrival and departure slots, wireless communication services, and nation-wide food supplying contracts [7, 89]. As mentioned in the article [24], these types of auctions have been also successful in fulfilling the requirements of fast trading and large, price sensitive services with economic efficiency. For instance, CRAs have been used in Chile to provide million of low-income school children with meals [24]. The government utilized CRAs by establishing contracts with food supplying companies. In fact, CRAs were successful not only to improve the meal quality and coverage but saved a significant amount of money, about \$40 million.

Lately, more advanced CRAs have been introduced in the literature. As an example, in the study [62], CRA was defined with multiple attributes of items but with single unit. The authors restricted each supplier to a certain number of items to win. They established a bi-objective programming model that minimizes the price and maximizes the scores of non-price attributes. They extended the evolutionary technique Ant Colony (AC) to solve their particular CRA by using two strategies: the max-min pheromone to

prevent the premature convergence and improve the optimization capability, and the dynamic transition strategy to enhance the searching process of the global optimization. They compared their Improved AC (called IAC) with the standard AC and also with Enumeration Algorithm with Backtracking (EAB), and showed that it outperforms both algorithms in terms of computation time. In the article [13], the WD problem of CRAs is carrying out with multiple units. The authors tackled the WD in the context of combinatorial transportation procurement auctions. They took two objectives into account: minimizing the price and maximizing the service-quality. To solve this combinatorial problem, they proposed a GA-based algorithm. They constructed eight variants of the solution method, and performed several tests to compare the time efficiency of these evolutionary methods. They also compared these GA-based methods with Branch and Bound-based algorithm, and showed that evolutionary methods perform better, especially for large problem instances. In another article [12], the authors addressed CRAs with multiple units and two objectives in the WD procedure: minimizing the transport costs and maximizing the transport quality. They developed a heuristic Pareto Neighbourhood Search method, inspired by the ideas of two meta-heuristics: Greedy Randomized Adaptive Search Procedure and Adaptive Large Neighbourhood Search. They conducted several tests to evaluate the performance in time of their WD method, and showed that it outperforms some other heuristics in terms of the approximation set quality.

1.2 Problem Formulation and Motivations

1.2.1 Auction Mechanism

We consider several types of auction mechanisms. For each type, there are multiple items, multiple sources and partial/full bidding. Depending on the variations of other parameters, such as number of units, number of attributes, number of objectives and number of rounds, in this thesis we address four types of auction mechanisms that are given in Table 1.1.

Table 1.1: Types of CRAs

Types	Different Features				Chapters
	Unit	Attribute	Objective	Round	
SU-TA-TO-TR	Single	Two	Two	Two	Chapter 7
MU-SA-SO-SR	Multi	Single	Single	Single	Chapter 3
MU-TA-SO-MR	Multi	Two	Single	Multi	Chapters 4 and 5
MU-MA-MO-MR	Multi	Multi	Multi	Multi	Chapter 6

1.2.2 Winner Determination

Winner determination in combinatorial auctions is computationally expensive and has been classified as a NP-complete problem [58, 91]. In practice, the following three approaches have been adopted to tackle this problem [91]: the first one is to limit the allowable bids (number of bids or combination of items), the second one is to address the unrestricted problems using search techniques (generate approximate solutions), and the last one is to find a sub-optimal allocation (consider partial solution space). Winner determination is based on two parts [21]: satisfiability and optimization. The main target of satisfiability is to find a solution to a given problem respecting a set of required services. On the other hand, the basic task of optimization is to find the

winner(s) based on multiple criteria [21], such as the reduced procurement cost and processing time. The required services identified to solve a task at the optimum measurement are the winners. The relationship between services can be of three types, such as co-operation, benefit co-operation and no co-operation. The order in which these services are combined in all the three relationships according to the criteria used for optimization is important [21]. Searching for the best solution (a set of winning sellers) in traditional CRAs (i.e. multiple items, single unit and single attribute) is already difficult to solve due to its computational complexity [62]. Previous studies adopted exact algorithms to look for the optimal solution in CRAs but endured an exponential time cost [28], which is unpractical in real-life auctions. To address this issue, researchers introduced evolutionary optimization techniques, such as Genetic Algorithms (GAs) based method, which produce high-quality solutions with an excellent time-efficiency [73]. Furthermore, dealing with conflicting attributes makes it even more difficult and time consuming to solve. That is why Multi-Objective Optimization (MOO) methods have been introduced to find the best trade-off solution that minimizes the cost and time.

In our case, we consider multiple units, multiple attributes, multiple objectives and multiple rounds. Moreover, we allow the bidders to place partial bids. Addressing all these along with the buyer's requirements and sellers' constraints, it makes our advanced CRAs more complex.

A. Evolutionary Algorithms

We use Genetic Algorithms (GAs) with diversity and elitism techniques. GAs have been applied successfully to solve many combinatorial optimization problems [58], such

as quadratic assignment problem [78], job scheduling [66] and travelling salesman problem [85]. Nevertheless, not much work has been done by using GAs to solve the winner determination problem in the context of CRAs. For example, [58], [68] and [69] used GAs to solve winner determination in the context of several types of problem instances of CRAs.

GAs are powerful search techniques consisting of selection, crossover and mutation methods that follow the Darwinian principle of survival: “Survival is the fittest” [27]. They are considered as approximate search algorithms driven by genetics and natural selection [56]. GAs model the sexual reproduction [52]. They are stochastic and polynomial rather than exponential [6] and are often used as an alternative approach for solving hard problems. The crossover operator builds a child with the combined characteristics of its parents. In contrast, mutation is a unary operator that needs only one input. During the process, the mutation operator produces a child by selecting some bad genes from the parent and replacing them with the good genes. Crossover and mutation both follow the characteristic of GAs in that the next generation is expected to perform better than the current one. GAs can terminate anytime as required and the current best chromosome can be the best solution within the required running time; that is why it is called anytime algorithm [21]. We may note that the problem representation is one of the most challenging tasks to the success of GAs [21]. Determining an appropriate fitness function for a specific problem is another crucial part of GAs in which the qualities of chromosomes are assessed, and very often this task requires most CPU intensive part of GAs [91]. GAs perform iterative multi-directional non systematic searches by maintaining a constant size population of individuals and encouraging

information generation and exchange between these directions [50]. Each iteration is called a generation and it undergoes some changes through crossover and mutation operators [50]. At each generation, good solutions are expected to be produced and bad solutions die. It is the role of the fitness function to distinguish the goodness of the solution [50].

In order to avoid premature convergence to local optimum, and to reach faster to the global optimum, we need to incorporate the diversity and elitism mechanisms to solve these two issues respectively.

B. Priori Approach

It has been an ongoing research to solve multi-objective decision problems with evolutionary algorithms because the latter are regarded as the most suitable methods for this kind of problems. Finding the set of near-to-optimal solutions and selecting the best one are both crucial. Researchers came up with two ways: priori and posteriori approaches [20]. The former first elicits the user preferences and then performs the optimization (via a composite function) to produce a single best solution. This approach heavily depends on the individual's preferences and the scalarization method, such as weighted-sum [16]. If the preference elicitation process is well defined, this subjective approach is a good choice. The posteriori optimization method first generates a set of trade-off solutions and then focuses on the decision task. Nevertheless, in case of a large number of conflicting objectives (more than five), numerous candidate solutions are returned to the user. So the main difficulty for the user lies in finding the best solution. The posteriori model may return more reliable solutions but suffers from a hard decision

making task. Another serious drawback is the computation time, especially for large-dimensional problems since this method searches all trade-off solutions.

C. Exact Algorithms

We have developed two exact procedures (see appendix c) to solve our constrained combinatorial auction problems: Brute Force that guarantees the solution optimality because it checks the entire solution space and Branch and Bound that is the most time-performing exact algorithm. The goal here is to compare the time-efficiency and accuracy (how near is the solution to the optimal) of our proposed evolutionary WD method with the performance of these two exact algorithms.

1.3 Solution Approach

Our WD procedures are subjective because they depend on the personal preferences of the buyer on the procurement. Besides, for subjective applications, it is desirable to determine only one best solution [16]. In multi-objective CRAs, it is very challenging for the buyer to evaluate a set of trade-off solutions to find the one that is the closest to his needs. By taking into account the buyer requirements, the large solution search space may be narrowed down. This will really help our WD techniques to produce the solutions very quickly. For all these reasons, we follow the priori approach. GAs have been applied successfully to many multi-objective optimization problems [41]. In our research, we propose WD approaches based on GAs that we integrate with diversity and elitism mechanisms in order to improve the quality of the solutions. These two mechanisms are essential for the effectiveness of the GAs [20, 33]. We define our own variants of the diversity and elitism strategies; other variants can be found in the papers

[20, 33]. We also consider the buyer's pay-off at each auction round, and based on the achieved gain, our WD methods will be able to increase the solution optimality.

Generally speaking, a MOO problem is defined with a set of objective functions, a set of decision variables and a set of constraints. In our CRA, the objective function corresponds to the maximization or minimization of a given attribute. A decision variable denotes the solution i.e. the set of winning sellers. Since we follow the priori approach, our combinatorial problem is solved with a single objective function, which is the fitness function. Assume X is a candidate solution; the target of our optimization is to maximize the fitness value of X . However, we have a mixture of maximization and minimization objectives. So to make it all maximization, we propose two versions of derivation of the fitness function. Consequently, it does not matter anymore what the objective of an attribute is since our WD approaches convert it into a maximization objective. In our CRA, the solution X should consist of eligible sellers only. When generating a potential solution, a seller is assigned randomly to a certain unit of an item. But there is a chance that this seller has been wrongly chosen since he did not submit any bid for that item. Our WD methods always ensure the feasibility of the solutions by checking the eligibility of each seller for each unit.

Our WD approaches employ three genetic operators: Gambling Wheel Disk method [29], Modified Two-Point Crossover that we introduced in [68], and Swap Mutation [23]. Gambling Wheel Disk and Swap Mutation are both popular methods and therefore suitable for our CRA problems.

Gambling Wheel Disk: We reproduce the chromosomes with the best fitness values based on their scopes. The chromosome with the largest scope has the highest fitness value, and therefore a better chance to be chosen.

Modified Two-Point Crossover: Modified Two-Point Crossover differs from the basic two-point crossover in the direction of taking portions from the parents. The first child is created in the same way as in the basic version, but the second child takes the portions in the reverse order. This modified version is of great significance. Even if the parents are the same, it is able to produce new child where the basic version just clones the parents.

Swap Mutation: In this mutation, sellers exchange positions (units or items) among themselves. Two positions in a chromosome are first selected randomly, and then sellers are swapped accordingly. Thus, this operation creates new chromosomes.

We also incorporate diversity mechanism (Crowding Distance) and elitism technique (Elitism with External Population).

Crowding Distance: We adopt crowding distance method and update it to our necessity. We propose the relative fitness function to calculate the crowding distance. The chromosomes are ranked. w. r. t their relative fitness values and the top chromosomes are then selected. The target is to prevent the population from having many similar solutions by not allowing premature convergence to local optima.

Elitism with External Population: Elite solutions are the best solutions found so far. We store these solutions in an external population. After each generation, we inject certain number of elite chromosomes in the participant chromosomes for the next

generation. The target is to avoid losing good solutions and help converging to the global optima.

In Figure 1.1, we show the general scenario of our WD method.

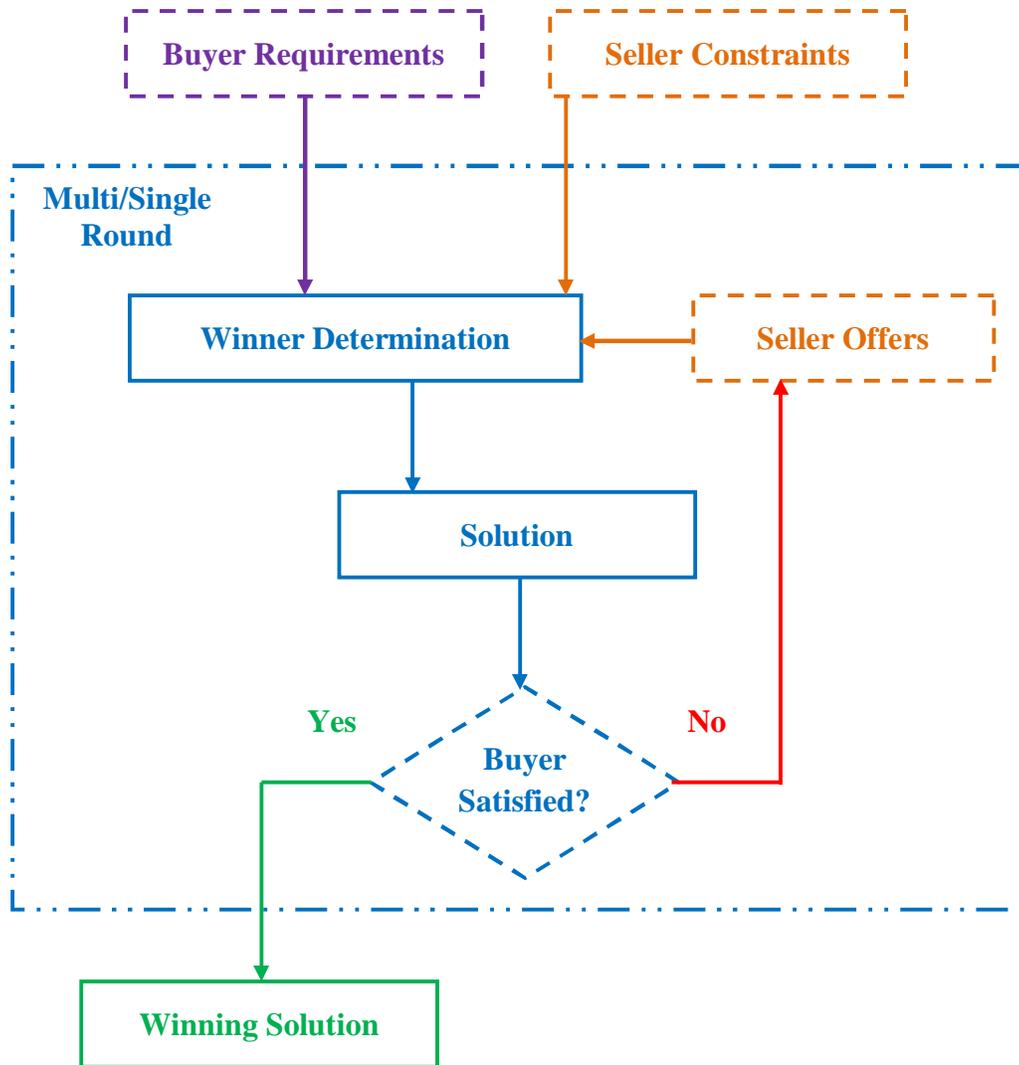


Figure 1.1: General WD scenario for CRAs

1.4 System Validation

We validate our WD approaches by conducting real case studies, performing simulations, statistical analysis, comparisons with exact and heuristics, and finally addressing real-life applications.

We conduct real case studies to show our WD methods step by step to make them understand better. We perform simulations to show the importance of diversity and elitism techniques with our WD methods. We conduct statistical analysis to discuss the consistency characteristic of WD approaches. We perform several experiments to compare the efficiency of our WD methods to well-known heuristic and exact WD algorithms that have been implemented for much simpler CRAs.

We also consider real-life application to validate our WD approaches. We address electricity combinatorial reverse auctions based on diverse renewable power sources.

1.5 A Summary

In Figure 1.2, we depict our thesis outline. We can easily divide our thesis into three domains, such as CRAs, WD methods and validation of WD approaches in the context of CRAs.

The first domain consists of the four types of CRAs we consider in our thesis. We define and formulate CRAs. The second domain, WD includes GAs as evolutionary algorithm, multi-objective optimization, and diversity and elitism mechanisms. We develop WD protocols based on GAs with diversity and elitism techniques to solve the multi-objective optimization problems in CRAs. The last domain validates our WD

approaches by conducting real case studies, simulating CRAs and performing statistical analysis, comparisons and real life applications.



Figure 1.2: Thesis summary

1.6 Contributions

In this section, we discuss our contributions.

- We address MU-SA-SO-SR CRAs. In this part of our research, we present CRAs having multiple units of each item and tackle the problem instance using GAs. Using a modification of the crossover operator as well as two routines for consistency checking, our proposed method is capable of finding the winner(s) with a minimum procurement cost and in an efficient processing time. In order to assess the performance of our GA-based method, we conduct several experiments on generated instances. The results clearly demonstrate the good time performance of our method as well as the quality of the solution returned.
- We consider MU-TA-SO-MR CRAs. In this research work we solve WD in the context of CRAs considering multiple attributes. In this part, we make the problem instance more interesting by considering all-units discounts on attributes and solving it using GAs. We also consider the availability of instances of items in sellers' stock. In order to evaluate the performance of our proposed method, we conduct several experiments on randomly generated instances.
- We present our research based on MU-TA-SO-MR. In this research work we consider CRAs with multiple attributes together with additional constraints, such as sellers' stocks and discount rate. In this part, we conduct a comparative study of several exact and evolutionary techniques that have been proposed to solve various instances of the combinatorial reverse auction problem. In particular, we show that a recent method based on genetic algorithms outperforms some other methods in terms of time efficiency while returning a near to optimal solution in most of the cases.

- We consider MU-MA-MO-MR CRAs. To address the WD problem for CRAs, we propose an optimization approach based on genetic algorithms that we integrate with our variants of diversity and elitism strategies to improve the solution quality. We conduct a case study as well as simulated testing to illustrate the importance of the diversity and elitism schemes. We also validate the proposed WD method through simulated experiments by generating large instances of our CRA problem.
- We address a real-life application. This part of research presents SU-TA-TO-TR. In this research work, we develop a CRA to procure power from diverse sources including residents and plants. Our CRA leads to a complex WD problem. We view this problem as a resource allocation optimization that we solve with multi-objective genetic algorithms in order to find the trade-off solution that best lowers the price and increases the energy. Moreover, we validate our WD approach based on simulated data by generating large instances of our multi-objective constrained auction problem. The goal of the experiments is to assess the time-efficiency of our WD method and its significant superiority to well-known heuristic and exact WD techniques. Moreover, we implement two exact algorithms to solve our complex problem in order to evaluate the accuracy of our WD method i.e. how near-to-optimal is the solution.

During PhD studies, we have published several journal articles, book chapters, conference papers, and abstracts [see Appendix D]. It is worthy to mention that we receive the *best student paper award* for the excellent research work in IEEE ICTAI 2016.

1.7 Thesis Organization

We organize our thesis into several chapters. The rest of this thesis is organized as follows:

Chapter 2 focuses on the literature related to CRAs, studies and research works carried out in this context. First we present some related features of auctions. Then we depict preference and constraint elicitation in the context of CRAs. Finally we discuss some state-of-the-arts based on electricity auctions.

Chapter 3 discusses our research work based on MU-SA-SO-SR CRAs. Chapter 4 presents MU-TA-SO-MR CRAs. In chapter 5, we present our research based on MU-TA-SO-MR. In chapter 6 we discuss MU-MA-MO-MR. In chapter 7 we present a real-life application. This chapter presents SU-TA-TO-TR.

Finally, chapter 8 presents concluding remarks as well as possible directions for future research work.

Chapter 2

Related Work

This chapter is divided into three sections. Section 2.1 includes features of auctions. Section 2.2 presents a brief overview of preference and constraint elicitation. Section 2.3 consists of Combinatorial Reverse Auctions (CRAs) including both traditional and advanced CRAs based on electricity market.

2.1 Features of Auctions

The auction reported in [75] as possibly one of the world's oldest commercial tools, is a market scenario in which bidders compete for an item. Auctions were used in Babylon as early as 500 BC and their usage was also found in Roman history [75]. Auctions have become a successful approach in buying and selling products such as properties, financial instruments and services. Moreover, several other markets are being introduced for auctions, for example, mobile-phone licenses, electricity, and pollution permits are some of them [40]. There are numerous applications of mechanism design in electronic market design, distributed planning, and combinatorial optimization problems [40] and reverse auction is one of the main branches of mechanism design. In a reverse

auction, there is one buyer and several bidders compete for an item. In a CRA, buyer can buy more than item at the same time. There are several features of an auction such as attribute, round and bidding protocol.

A. Single-Attribute vs Multi-Attribute Auctions

According to the negotiable aspects, auctions can be categorized as single-attribute or multi-attribute. Single-attribute auction in which the bidders can negotiate only on one attribute which is price. The mechanism of a single-attribute auction is ease for the bidders to understand. Single-attribute auctions limit the bidders to bid on one attribute to obtain the products. But, in many real world situations, competition and negotiation engage several dimensions rather than only price. This type of auction, which is an extension of the single-attribute auction, is called multi-attribute auction [11]. Multi-attribute auctions are auctions in which the bidders can submit bids on multiple attributes such as price, product quality, delivery type, seller's reputation, warranty, terms of transportation, and delivery time. With the arrival of the internet, multi-attribute reverse auctions that allow the buyer and sellers to engage in negotiation have appeared [34]. Bichler discussed the method of winner determination in multi-attribute reverse auctions and configurable offers which enable the suppliers to specify multiple values for each attribute in [10] and clarified that the bidders have chances to improve the bidding values at less cost in multi-attribute auctions in [8]. Bichler and Kalagnanam [10] presented a multi-attribute auction as a producer of high-gain because of its bidding flexibility. Teich et al. covered many other detailed reviews and design features of multi-attribute reverse auctions in [79]. The features of multiple attributes have become essential in e-commerce because many industries design their markets based on multiple

attributes. The advantage of a multi-attribute auction is that it allows the bidders to negotiate on multiple attributes, rather than limiting them to a single-attribute. This feature increases the efficiency of an auction. In this way, a multi-attribute auction offers bidding flexibility and it produces higher buyer-utility and seller-profit than that of a single attribute auction [9]. Needless to mention that a multi-attribute auction has difficulty in terms of making decisions as it deals with many attributes, but still it is a very useful method [76].

B. Single-Round vs Multi-Round Auctions

Based on the number of round, auctions can be of two types such as single or multiple rounds. In a single round auction, there is only one round. This limitation makes the bidding process a difficult task for the bidders. In this type of auction, the bidders have no chance to make their bids improved because there is no feedback from the previous round. Hence, the disadvantage of a single round auction is the lower potential to be the winner. On the contrary, a multi-round auction provides several rounds for the bidding process. In a multi-round auction the bidders submit their bids and the winner is announced at the end of each round which benefits the bidders to improve their bids in the next round. To determine the current winner, the bids are evaluated at the end of each round [55]. Thus, a multi-round auction provides the opportunity to the bidders to improve their bids in the next round [65] and eventually offers the higher potential to be the winner. It makes the auction more competitive. A multi-round auction continues until a termination criterion is met. The possible termination conditions are, for example, the maximum predefined number of rounds has been reached or the bids are not getting improved compared to the previous round. In a

multi-round auction, the bidder does not need to collect information of the bids of the competitors, because after each round all bids are shown to all bidders.

C. Open vs Sealed-bid Auctions

Depending on the visibility granted to the participants, auctions can be classified as open or sealed-bid. An open bid auction is a type of auction in which the bids are visible to all bidders. On the other hand, a sealed-bid auction is a type of auction in which the bids are private. In a sealed-bid auction, communication is not allowed between the bidders. By comparison to a open bid auction, having the bids invisible helps a sealed bid auction to reduce the potential of trust management issues and hence makes the process more secure. Sealed-bid auctions have been widely used in e-commerce for selling products and resource allocation and its reverse version has been used in task assignment scenarios [80]. A first price sealed-bid auction is a type of sealed-bid auction in which the bidding is private. In a first price sealed-bid auction, the bidders are allowed to place only one final bid and the bids are kept secret from all other bidders. This type of invisible bidding process prevents any type of communications among the bidders to use shared information. The bidder who submits the highest bid wins and he has to pay the bid he submitted. First price sealed-bid auctions are suitable for use in public procurement [18]. Vickrey auction is a type of sealed-bid auction presented by William Vickrey in 1961 in which the bidding process is private or closed. In a vickrey auction, the bidders bid without knowing the bids of other bidders. This feature of a vickrey auction makes the auction house secure. The bidder who submits the highest bid wins but he has to pay the second highest bid. This is the difference between a first price sealed-bid auction and a vickrey auction. Moreover, a vickrey auction differs from other

auction protocols by its closing time mechanism. In a vickrey auction protocol, the closing time is not determined and for that reason, the auction can be closed whenever the bid price matches the requested price [83]. For example, eBay uses a bidding strategy similar to vickrey protocol.

2.2 Preference and Constraint Elicitation

Preference elicitation plays an essential role in negotiation systems such as automated market places, bilateral bargaining, auctions and shopping websites. Maximizing the satisfaction of buyers may be achieved by considering precisely their preferences. [38] proposes a personalized matchmaking system that determines the best offer by evaluating and sorting the sellers' offerings according to the buyer's specific interests. This system is based on a neural-network technology, self-organizing map, and the economics model MultiNomial Logit. [67] discusses and evaluates different methods, such as Self Explicated Approach, Full Profile Conjoint, Hybrid Conjoint, and Analytic Hierarchy Process, to elicit and represent preferences in negotiation support systems. Based on a multi-attribute utility theoretic model of user preferences, [32] introduces an algorithm that learns an overall utility function with flexibility to accept several types of information, such as the attribute weights and utility functions. It combines evolutionary learning with the external knowledge and local search. By conducting some experiments, the authors show that their algorithm of utility elicitation in an agent-based negotiation system provides a good learning performance and thus can be used in a wide variety of applications. [25] exposes a general interactive tool to obtain users' preferences about concrete outcomes, and to learn utility functions automatically based on users' feedback. In [90], the authors propose a preference

elicitation framework as well as a linear programming model to infer the appropriate preference model w.r.t the user's preference statements. They implement the framework based on an agent intermediary architecture consisting of several components: the semantic analyzer, preference elicitation, bid evaluation, model base and database. [31] identifies five crucial factors in the context of real-world preference elicitation problems such as real-time, multi-attribute, low cognitive load, robust to noise and scalable. Based on these requirements, this paper suggests a framework that facilitates the efficient evaluation of value of information heuristics. This framework performs effectively with all the five factors for both real-world and synthetic datasets. In [31], preferences are expressed as additive value functions. Formalizing users' preferences accurately is also very important in most decision support systems [14]. These systems rely on preferences to produce effective user models. [39] develops a variant of the fully probabilistic design of decision-making strategies. In [39], the elicitation of preferences is based on quantitative data and the solving method on Bayes rules. In [60], the authors are concerned about the preference elicitation models of several domains such as decision support systems and recommender systems. They perform three experiments: the first one is to let the user express preferences through various models the second one is to analyze the trade-off between user's feedback and effort, and the third one is to study the influence of interfaces on users. This work uses ranking, ordering and navigational techniques to represent preferences. Furthermore, the acquisition of high quality users' preferences is significant in interactive web services [37]. The quality of the returned results depends on the capability of the services to acquire the preferences. [37] examines the adaptation and personalization strategies during the elicitation process for

web services. This paper describes preference elicitation techniques in ADVISOR SUITE, a domain-independent software tool used in e-commerce.

In the setting of auctions, very few papers take into account hard constraints. The auctioneer can consider constraints on different types of objects, such as attributes, bids, and auction termination. In [49], [80], constraints represent trivial information, such as the quantity of the auctioned item (e.g. quantity ≤ 2000). In [48], the authors propose a constraint-based negotiation framework where offers and counter-offers are based on constraints (such as budget and time constraints) and arguments. [10] defines constraints by linear inequalities in a multi-dimensional auction platform. Preferences and constraints can co-exist together in many domains [3], [61], and it is of great benefit to handle them together in many real-world applications. For instance, [26] employs MAUT to process quantitative preferences, and Conditional Preference networks (CP-nets) to formalize qualitative preferences. [61] describes constraints as a weighted Constraint Satisfaction Problem (CSP), conditional and qualitative preferences as CP-nets. [4] introduces an online shopping system that provides the buyers with the ability to specify in an interactive way constraints and preferences where the latter can be quantitative, qualitative or both. This work employs C-semiring to describe quantitative preferences, CP-nets for qualitative ones, CSP for constraints. This paper utilizes branch and bound method to provide the users with a list of outcomes. [3] introduces a new algorithm to determine the best outcome based on the arc consistency propagation technique. It performs several experiments to show that the proposed approach is able to save substantial amount of time to generate the optimal solutions.

2.3 Electricity CRAs

There are several studies regarding the mechanism design of electricity auctions. We first examine traditional (non-combinatorial) auctions and then more advanced (combinatorial) ones. Both auction types have several limitations, for example multiple attributes and objectives along with trading constraints are not considered. Generally speaking, WD approaches in e-auctions have been classified into three categories [81]: 1) exact approach that guarantees solution optimality but comes with huge computational cost, 2) approximated approach with low computational cost but solution optimality is not ensured, and 3) restricted approach with solution optimality and low computational cost but does not consider other features like partial bidding, multiple units and attributes, buyer and seller constraints. Researchers limit the auction features and parameters for the sake of simplicity.

2.3.1 Traditional CRAs

The traditional electricity auctions that we examine here are reverse protocols with single unit and most of them with single attribute (price). In [82], the authors addressed the management problem of energy resources of the future smart power grids in the context of multi-player negotiation. The proposed exact multi-agent approach is based on a knowledge-based modeling and a set of independent and competitive players. Due to the increased competition of players and the energy management environment being dynamic and complex, the legacy infrastructures are not suitable to accomplish the current need of energy management. The authors mentioned that artificial-intelligence based applications can tackle this situation by providing auction systems with

appropriate decision support. They also pointed it out that the usage of renewable energy sources has been increasing due to fossil fuels shortage and environmental concerns (i.e. low impact on greenhouse gas emissions). Lastly, this paper conducted a detailed case study on energy resource management, including storage, production and load response. [47] analyzed the electricity wholesale markets of most of the European countries by focusing on the Day Ahead Market, Intra Day Market and Balance Market. This paper compared the markets based on several bidding mechanisms, price formation and timing. The target of the auctions was to reduce the management cost. The authors mentioned that a good knowledge of the electricity markets is the key to gain profit from a single or a portfolio of power plants. They also analyzed the projects conducted by the European agencies in order to harmonize the wholesale electricity markets in the continent. Moreover, according to one of the agencies, several hundred million per year can be saved by integrating wholesale electricity markets.

2.3.2 Advanced CRAs

There are few electricity combinatorial auctions despite the fact that they match the demand and supply more efficiently and maximize the buyer payoff as well [59]. Even though these types of auctions are difficult to develop, they have been successfully adopted in other domains for both government and private sectors (free markets), like food supplying contracts, airport runway time slots and bus transportation routes [28]. [28] developed a Web-based CRA (single unit and single attribute) with user interfaces for electricity retail markets to minimize buyer expenditures. This auction allows consumers to open auctions, define hourly consumption amounts and choose suppliers with the cheapest energy acquisition. The authors claimed that this flexibility of

consumers creates more competition among suppliers, and ultimately increases the number and gain of the suppliers and consumers respectively. They employed the well-known optimizer named IBM CPLEX to determine the auction winners. They proved that their protocol produces efficient allocation of electricity usage because consumers purchase energy from several companies to minimize their expenditures. Nevertheless, this auction system was not tested yet with real markets.

In [77], it has been shown that evolutionary algorithms are suitable and perform well in the context of combinatorial auctions. [77] exposed an Evolutionary Iterative Random Search Algorithm defined for auctions with multiple units and multiple rounds. They showed that their protocol achieved Nash Equilibrium in the following way: assume the entire bidders offer the same price at the beginning; if any bidder bids higher the utility value of his bid will decrease; otherwise, the utility value will be zero. The same case holds for sellers. This situation indicates that the market is in Nash Equilibrium. The convergence of prices of buyer and bidders assures the market equilibrium. Another research [87] presented a GA-based optimal resource allocation approach for combinatorial auctions with multiple units and multiple rounds. This method, shown to be feasible and effective through simulation results, is able to maximize the total trading amounts of sellers and to reduce the processing time in the context of WD problem. [87] claimed that when the resource allocation problem has feasible solutions, then Nash equilibrium is always guaranteed.

Chapter 3

MU-SA-SO-SR CRA

3.1 Summary

Winner determination in combinatorial reverse auctions is very important in e-commerce especially when multiple instances of items are considered. However, this is a very challenging problem in terms of processing time and quality of the solution returned. In this part of our research, we tackle this problem using Genetic Algorithms (GAs). Using a modification of the crossover operator as well as two routines for consistency checking, our proposed method is capable of finding the winner(s) with a minimum procurement cost and in an efficient processing time. In order to assess the performance of our GA-based method, we conduct several experiments on generated instances. The results clearly demonstrate the good time performance of our method as well as the quality of the solution returned.

3.2 Introduction

The main purpose in combinatorial auctions is to increase the efficiency of bid allocation especially when bidding on multiple items [6], [58]. More precisely, the goal here is to minimize the procurement cost in an efficient running time [6], [58], [91]

when determining the winner [29]. Winner determination is still one of the prime challenges in combinatorial auctions [6] and is identified as an NP-complete problem [58], [91]. Nevertheless, applying combinatorial auctions to a procurement scenario [54], [63], such as transportation services, travel packages and sales of airport time slots, is cost saving [58]. Many algorithms have been introduced to tackle combinatorial auctions, e.g. Sitarz defined Ant algorithms and Hsieh and Tsai developed a Lagrangian heuristic method [58]. While GAs have been applied successfully to solve many combinatorial optimization problems [58], such as quadratic assignment problem [78], job scheduling [66] and travelling salesman problem [85], not much work has been done to solve the problem of winner determination in the context of Combinatorial Reverse Auctions (CRAs) with GAs.

In [69], we propose a GA based method, called GACRA, to solve winner determination specifically for CRAs by considering multiple items. In [69], we introduce two repairing methods, RemoveRedundancy and RemoveEmptiness, along with a modified two-point crossover operator. By using these two methods and an efficient crossover operator, we solve the problem of winner determination successfully and efficiently. The goal of these two methods is to repair the infeasible chromosomes and the main task of the crossover operator is to ensure a diversity of solutions. Indeed, increasing diversity allows bidders to get more chances to be selected. In [69], we have demonstrated that GACRA is better in terms of processing time and procurement cost when compared to another GA-based winner determination method [58]. Subsequently in [68], we conduct several statistical experiments and show that GACRA is a consistent

method since it is able to reduce the solution variations in different runs. Moreover, we describe the any-time behaviour of GACRA in [68].

In this part, we consider the procurement of a single unit of multiple instances of items and propose an improved GA-based method, that we call GAMICRA (**Genetic Algorithms for Multiple Instances of Items in Combinatorial Reverse Auctions**), by modifying the methods and operators of [68] and [69] as follows.

- We change the configuration of chromosomes to represent the bids in the case of multiple instances of items. We define the configuration of the chromosomes as well as the length of the chromosomes based on the number of sellers, items and instances of items.
- We modify the fitness function to consider multiple instances of items. The fitness function is kept simple enough in order not to perform a lot of calculation. This will help the processing time to remain reasonable.
- We also improve RemoveRedundancy and RemoveEmptiness methods to solve the winner determination problem associated with multiple instances of items. RemoveRundancy guarantees that the selected number of instances for each item does not exceed the buyer's requirements. On the other hand, RemoveEmptiness ensures that the selected number of instances for each item is not less than that of the buyer's requirements.

With these modifications and using the modified two-point crossover defined in [69], we solve efficiently the problem of winner determination for multiple instances of items in CRAs. In order to evaluate the performance of our method in terms of procurement cost and processing time, we perform several experiments. The results we report in this

study clearly demonstrate that our method is prominent in terms of time and cost and also show that it does not suffer from the inconsistency issue.

In many e-commerce and resource allocation problems, real-time response to a very large problem is expected [36]. When solutions are needed quickly and problem instances are large, exact algorithms that produce solutions at the expense of consuming extended required time are not only inadequate but also become infeasible as instances become larger. In real life, certain domains may require high-quality and approximate solutions within a suitable running time [36]. Sometimes it is unnecessary to spend a lot of time to slightly improve the quality of the solution. For these reasons, in this research we consider GAs to solve the winner determination problem as they have the ability to produce good solutions from the very first generations.

The rest of this topic is organized as follows. In Section 3.3, our GAMICRA method is presented and then explained through an example. In Section 3.4, the experimental study we conducted to evaluate the processing time, procurement cost and the consistency of our method is reported. In Section 3.5, concluding remarks with some future research directions are given.

3.3 Proposed Method

In a CRA, there is one buyer and several sellers who compete according to the buyer's requirements. First the buyer announces his demand (multiple instances of items) in the auction system. Then, the interested sellers register for that auction and bid on a combination of items. In this research, our goal is twofold: (1) to solve the winner determination problem for multiple instances of items in the context of CRAs, and (2) to find the winners in a reasonable processing time and reduced procurement cost. For this

purpose, we propose the method GAMICRA that utilizes two techniques, namely RemoveRedundancy and RemoveEmptiness, to repair infeasible chromosomes. In addition, we utilize the modified two-point crossover operator defined in [69] that provides a good distribution of the solutions and preventing a premature convergence. Figure 3.1 illustrates a possible scenario in CRAs. Here the buyer wants to purchase three instances of item A and five instances of item B. There are three sellers (S1, S2 and S3) who want to bid on items A and B. Therefore, the maximum instance between these two instances of items A (here 3) and B (here 5) is equal to 5. Hence, 3 bits are required to represent each item for each seller considering the following equation (3.1).

$$\max(\text{insItemA}, \text{insItemB}) = 2^{\text{reqBit}} \quad (3.1)$$

With 3 bits, we can generate $2^3 = 8$ combinations {000, ... , 111}. We assume for this example that the number of chromosomes = 4, and the length of chromosomes = $2 \times 3 \times 3 = 18$ according to the following equation (3.2).

$$\text{chromosomeLength} = m \times n \times \text{reqBit} \quad (3.2)$$

Figure 3.2 presents the pseudo code of our GAMICRA method (Algorithm 3.1) while RemoveRedundancy and RemoveEmptiness procedures are reported respectively in Figures 3.3 and 3.4. In Algorithm 3.1, the bid price of each item for each seller is generated randomly from the interval [100, 1000] (see Table 3.1). Table 3.2 shows the initial chromosomes that are produced randomly.

The next steps of Algorithm 3.1 are to convert infeasible chromosomes into feasible ones with the help of RemoveRedundancy and RemoveEmptiness methods. For our example, the conversion of chromosome X1 from infeasible to feasible one is described below.

Table 3.1: Bid price generation

	S1	S2	S3
Bid Price of Item A	951	868	525
Bid Price of Item B	826	920	696

Table 3.2: Initial chromosomes

Chromosome	S1		S2		S3	
	Item A	Item B	Item A	Item B	Item A	Item B
X1	010 (2)	011 (3)	110 (6)	001 (1)	001 (1)	101 (5)
X2	100 (4)	100 (4)	110 (6)	011 (3)	011 (3)	101 (5)
X3	010 (2)	010 (2)	010 (2)	101 (5)	000 (0)	010 (2)
X4	010 (2)	101 (5)	101 (5)	010 (2)	101 (5)	110 (6)

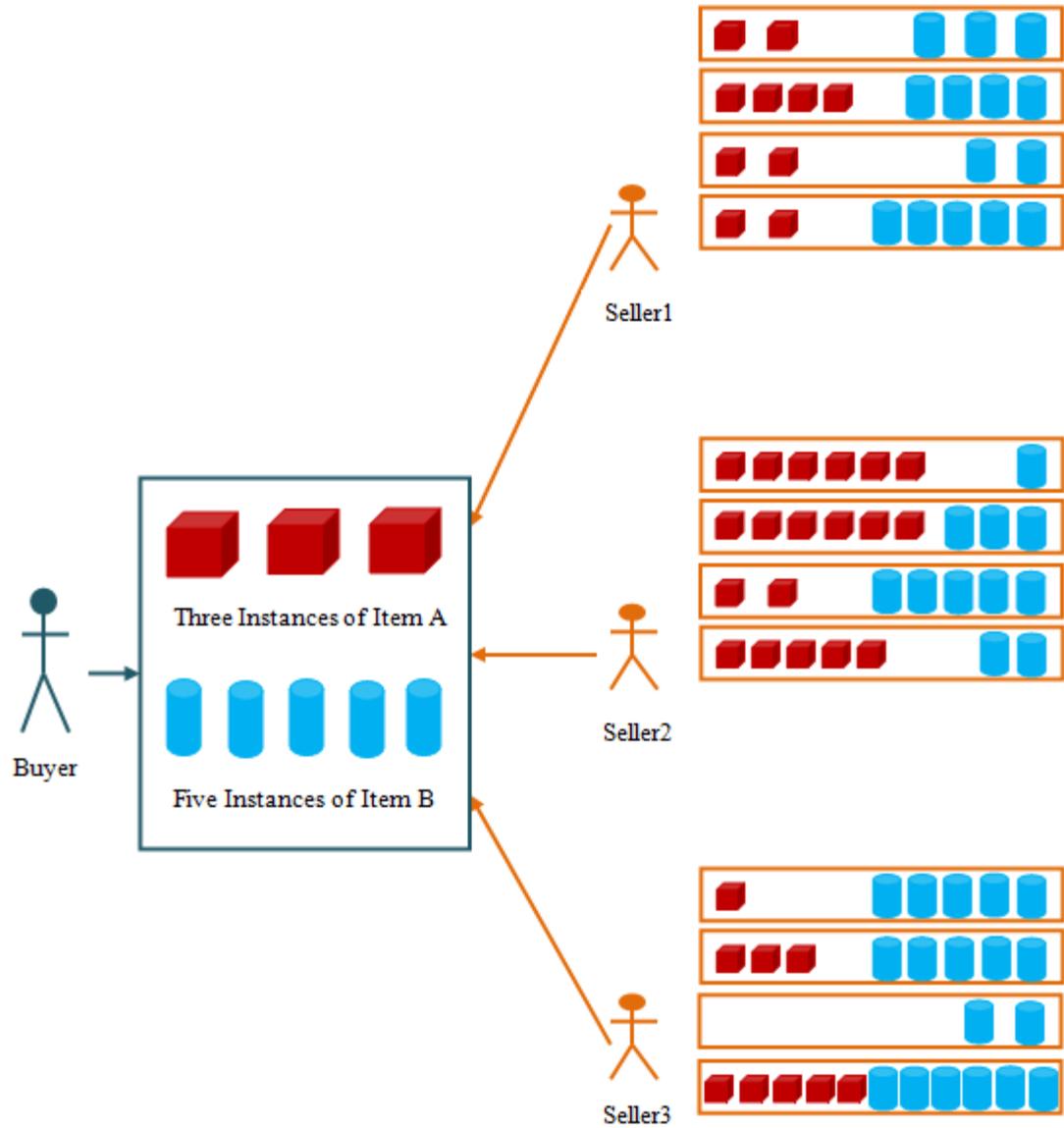


Figure 3.1: An example in CRAs

Initially, chromosome $X1$ is $010\ 011\ 110\ 001\ 001\ 101$. As shown above, the number of instances for Item A is 9 which is more than the buyer's requirement. The number of instances of Item B also exceeds the initial requirement. Hence, this chromosome is infeasible and needs to be repaired.

At first we will remove redundancy by using RemoveRedundancy method according to Algorithm 3.2 shown in Figure 3.3. After removing the redundancy, $X1$ is now equal to $000\ 000\ 000\ 000\ 000\ 101$. Although the number of instances of Item B is the same as the buyer's requirement, chromosome $X1$ is still infeasible as the number of instances of Item A is less than that of the buyer's requirement. In this case, we will apply the RemoveEmptiness method on this chromosome according to Algorithm 3.3 given in Figure 3.4. After removing the emptiness, $X1$ becomes $000\ 000\ 000\ 000\ 011\ 101$. Now, chromosome $X1$ is a feasible one. By applying the same process to all the chromosomes, the initial chromosomes after remove redundancy and remove emptiness are as follows.

$X1$: $000\ 000\ 000\ 000\ 011\ 101$

$X2$: $000\ 000\ 000\ 101\ 011\ 000$

$X3$: $000\ 000\ 010\ 101\ 001\ 000$

$X4$: $000\ 101\ 011\ 000\ 000\ 000$

Algorithm 3.1: GAMICRA (m : number of bid items, n : number of sellers, δ : number of generations, α : crossover rate, β : mutation rate)

1. $t = 1$;
2. bidGenerator();
//generates bid prices for each combination of bid items for each seller
3. chromosomeGAMICRA();
//generates initial chromosomes
4. RemoveRedundancy();
//removes redundant instances of items
5. RemoveEmptiness();

```

//adds instances of items as buyer's requirements

6. fitnessGAMICRA();

//computes fitness values of chromosomes

do{

7. selectionGAMICRA();

//selects chromosomes using gambling-wheel disk method

8. crossoverGAMICRA();

//generates child from parent chromosomes with modified two-point crossover

//considering rate  $\alpha$ 

9. mutationGAMICRA();

//mutates chromosomes considering mutation rate  $\beta$ 

10. RemoveRedundancy();

11. RemoveEmptiness();

12. fitnessGAMICRA();

13. newChromosomeGAMICRA();

//selects better chromosomes from both initial and new chromosomes of each

//generation

}while  $t \leq \delta$ ;

14. return winner(s);

//returns winner(s) with minimum bid price in optimal processing time

```

Figure 3.2: Algorithm for GAMICRA

Algorithm 3.2: RemoveRedundancy (X : chromosome, m : number of items, n : number of sellers)

```
1. for each X do
2.   { for each m do
3.     { if(redundantViolet())
        //item possesses redundant instances
4.       { generateRN();
        //generates random number between 1 and n
5.         convertZero();
        //converts bits for this seller to 0s
6.       }
7.     }
8.   }
```

Figure 3.3: RemoveRedundancy pseudo code

Algorithm 3.3: RemoveEmptiness (X : chromosome, m : number of items, n : number of sellers)

```
1. for each X do
2.   { for each m do
3.     { if(emptinessViolet())
        //item possesses less instances
4.       { while number of instances are not exactly same as the buyer wants
        to
```

```

buy do
5.      { generateRN();
6.          convertOne();
          //converts required bits for this seller to 1
7.      }
8.  }
9.  }
10. }

```

Figure 3.4: RemoveEmptiness pseudo code

For evaluating the fitness value of each chromosome, the following fitness function is applied.

$$F(x_i) = \frac{1}{\sum_{N=1}^n \sum_{M=1}^m b_{NM} \times l_{NM} \times x_{NM}(C)}$$

$$/x_{NM}(C) \in \{0,1\} \tag{3.3}$$

Where $F(X_i)$ is the fitness value of chromosome X_i ; b_{NM} is the bid price of item M submitted by seller N; l_{MN} is the number of instances of item M submitted by seller N; $x_{NM}(C)$ is 1 when the item combination C is selected for item M of seller N, and 0 otherwise.

Gambling Wheel Disk [29] is used in the selection operation. After selection, the following chromosomes are produced based on the fitness values calculated by using equation (3.3).

X1: 000 000 000 101 011 000 (Previous X2)

X2: 000 000 000 101 011 000 (Previous X2)

X3: 000 101 011 000 000 000 (Previous X4)

X4: 000 101 011 000 000 000 (Previous X4)

Then the modified two-point crossover is applied as shown in Figure 3.5(a). In our modified two-point crossover operation, the first child inherits the parent chromosomes between the crossover points in the forward direction as the basic two-point crossover, but the second child inherits it in reverse order to create a positive effect.

After crossover, the following chromosomes are generated.

X1: 000 000 000 101 011 000

X2: 011 000 000 101 000 000

X3: 000 101 011 000 000 000

X4: 000 101 011 000 000 000

Then the procedure will enter into the mutation operation as illustrated in Figure 3.5(b). For each chromosome, two unique random numbers are selected between 1 and n. Assume for this example, 1 and 3 are chosen, so the bit configurations of sellers S1 and S3 are interchanged. Algorithm 3.1 is repeated until the specified number of generations is completed. Thus, the winner(s) are determined based on the minimum bid price in an optimal processing time.

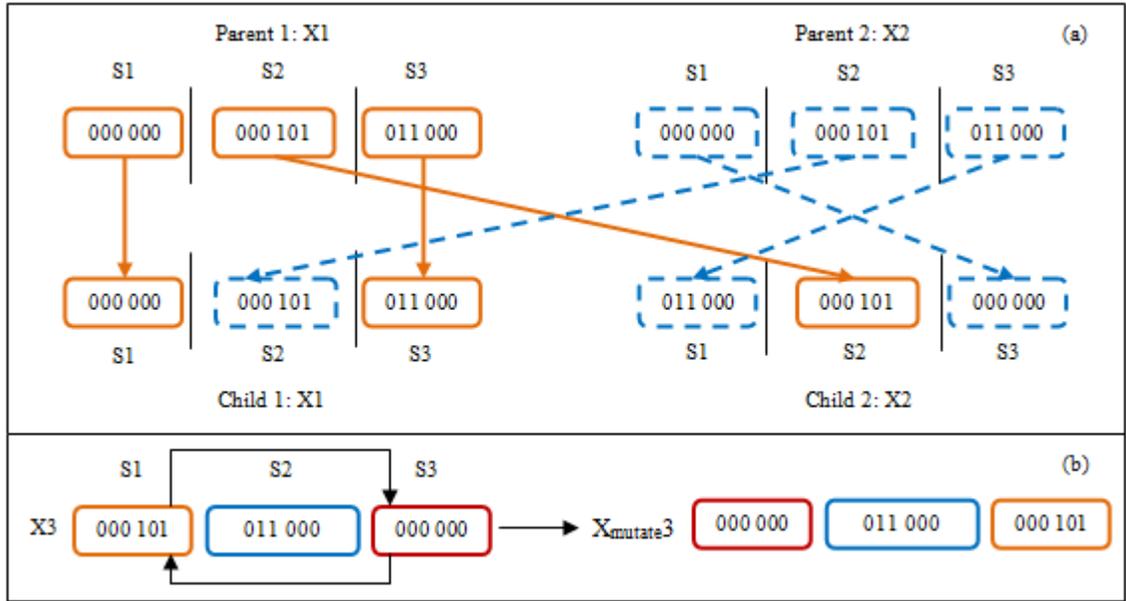


Figure 3.5: Modified two-point crossover and Mutation Operation

3.4 Experimentation

We have fully implemented GAMICRA in Java. This method is executed on an AMD Athlon (tm) 64 X2 Dual Core Processor 4400+ with 3.43 GB of RAM and 2.30 GHz of processor speed.

We use the following parameters and settings for all the experiments. The population size is 100 and we use the binary string for chromosomes encoding. The selection method is based on the Gambling Wheel Disk technique [29]. The crossover and mutation rates are 0.6 and 0.01 respectively. The running time in milliseconds and procurement cost (bid price) are averaged over 20 runs. The maximum number of generations is 50. The total number of sellers, items and instances are 500, 8 and 5 respectively unless indicated otherwise.

Figures 3.6(a), 3.6(b), 3.6(c), 3.6(d), 3.6(e) and 3.6(f) report the results of these experiments. Figure 3.6(a) shows how the running time varies with the increase of the number of generations. As we can notice our method can still run in a reasonable time (under the minute) even for a large number of iterations. Figure 3.6(b) shows how does the bid price improve with the increase of the number of generations, thanks to our modified two-point crossover initially defined in [63]. As we can easily see, this cost significantly improves in the first 30 iterations and stabilized afterwards. Figures 3.6(c) and 3.6(d) show how the running time increases when we augment the number of sellers as well as the number of items. In this regard our method is capable of producing the solutions in less than 30 seconds even for large number of sellers and a good number of items. Figure 3.6(e) shows the time performance of our method when varying the number of instances from 1 to 128. Here too, GAMICRA successfully returns the solutions in a reasonable running time (under than 2 minutes in the case of 128 instances). Finally Figure 3.6(f) demonstrates the consistency of GAMICRA. The bid price is indicated by a solid line with the maximum and minimum values represented by + and - respectively. The error bars are depicted with a confidence level of 95%. From the figure, it is clearly notable that the variations of bid price in a particular generation over different runs are quite small. This means that GAMICRA is able to reduce the solution variations. It is also notable that the solution quality increases steadily over the generations. After 5 and 10 generations, our method produces an average bid price of 6267.55 and 5731.3 respectively. It then continues producing better solutions generation after generation. Furthermore, the variability of the solution quality over multiple runs improves over generations. The variability of the solutions fluctuates till the 25th

generation and stabilizes afterwards. In addition, the minimum bid price remains constant after 25 generations. This suggests that the best solution found by GAMICRA might be the optimal solution.

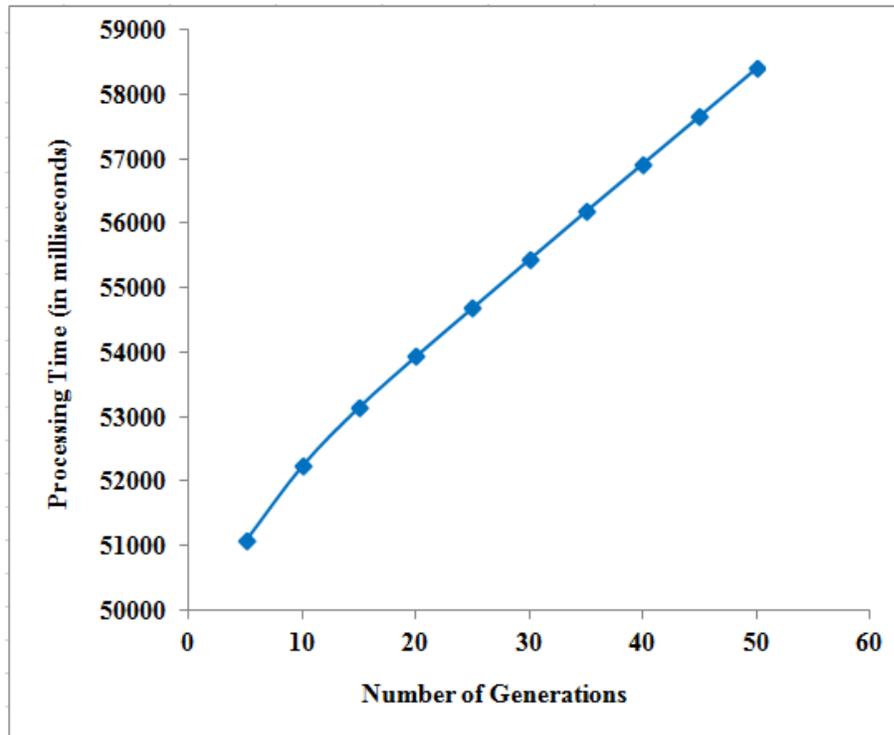


Figure 3.6(a): Running time and procurement cost performances (processing time vs number of generations)

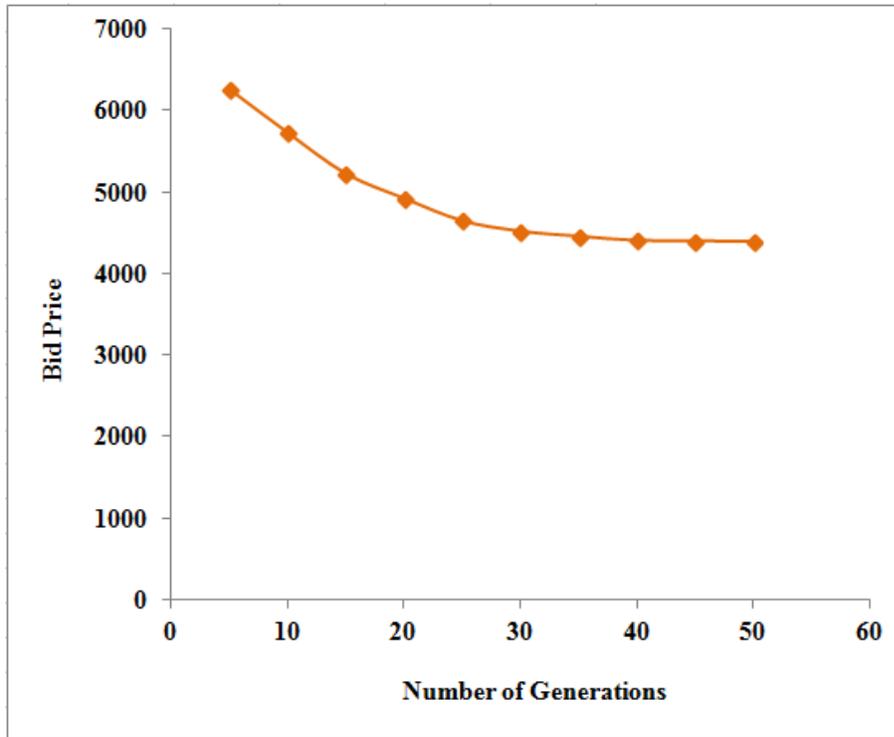


Figure 3.6(b): Running time and procurement cost performances (bid price vs number of generations)

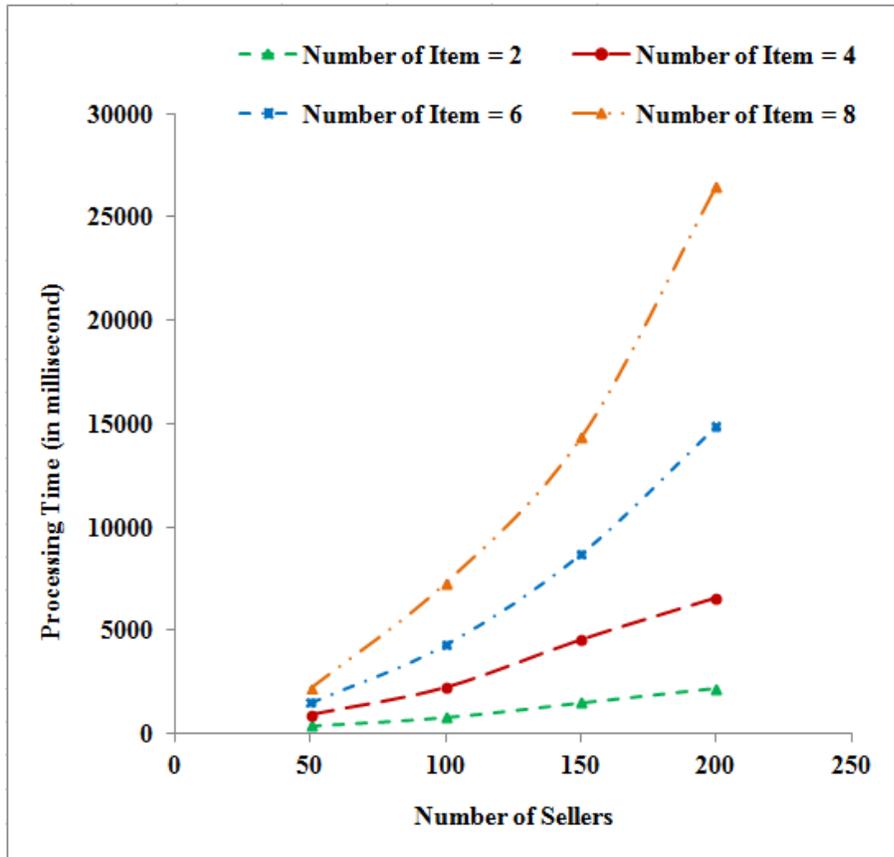


Figure 3.6(c): Running time and procurement cost performances (processing time vs number of sellers)

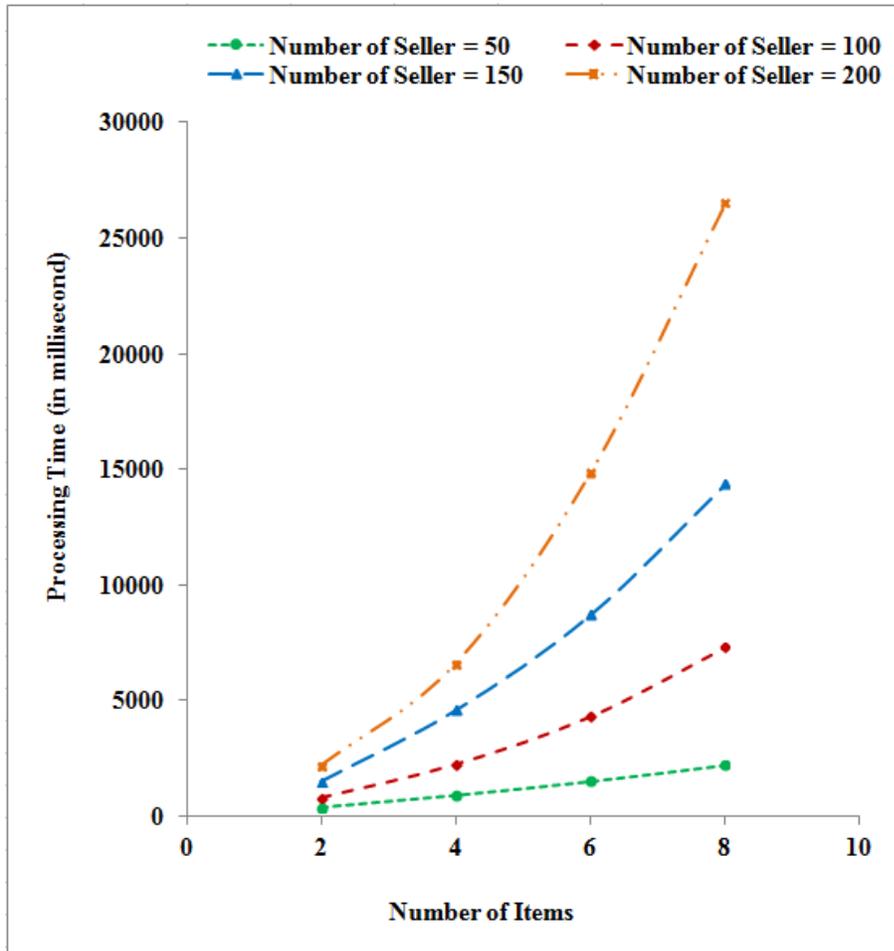


Figure 3.6(d): Running time and procurement cost performances (processing time vs number of items)

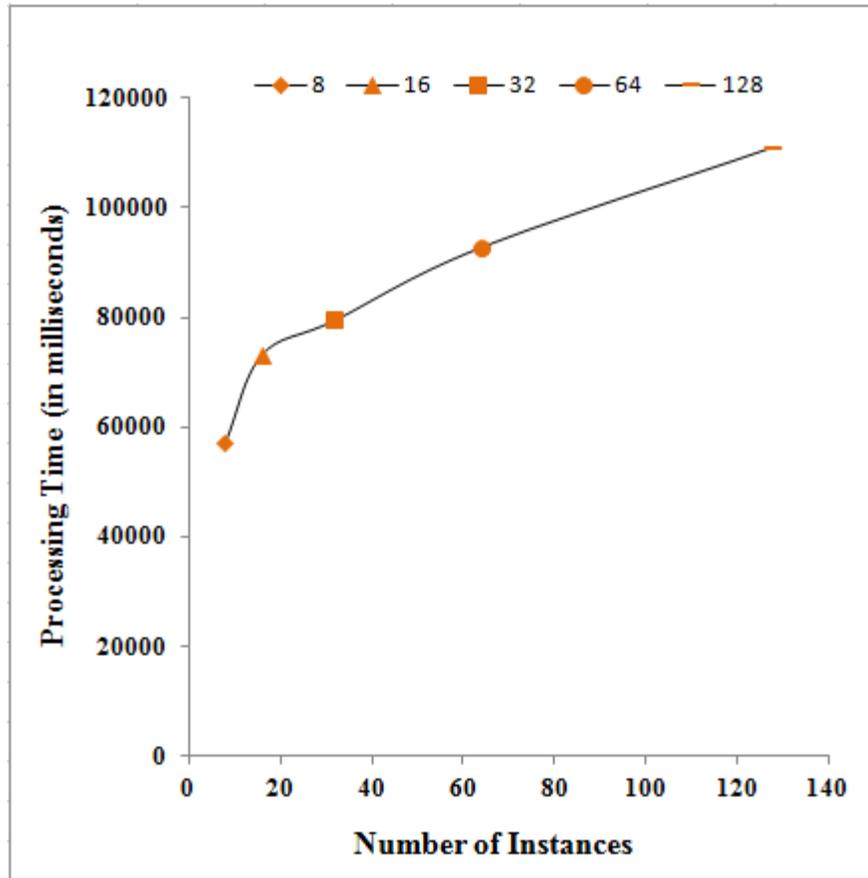


Figure 3.6(e): Running time and procurement cost performances (processing time vs number of instances)

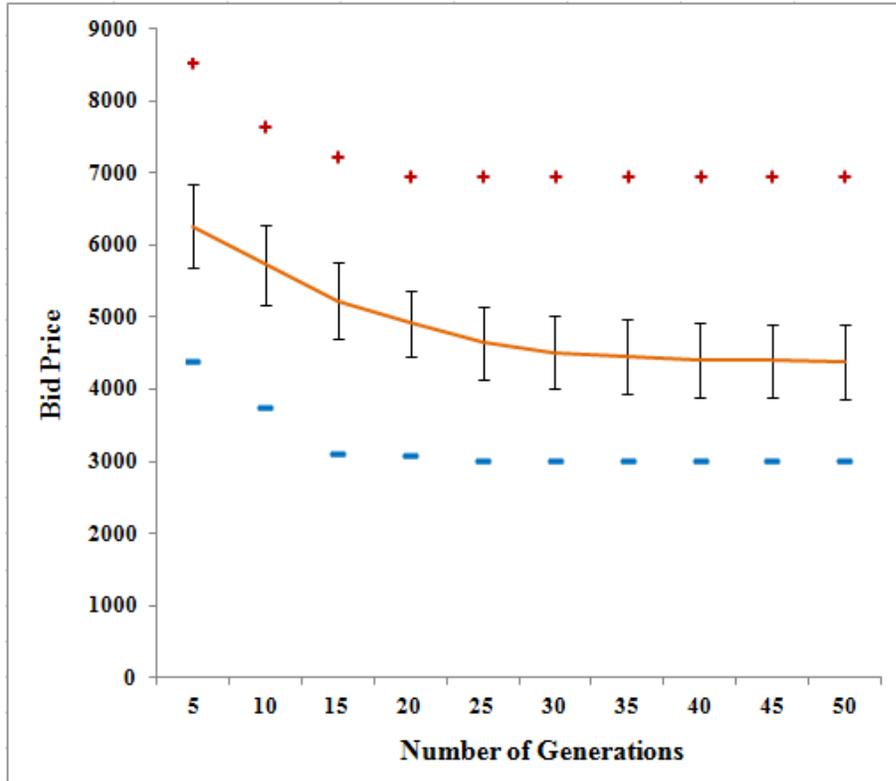


Figure 3.6(f): Running time and procurement cost performances (bid price vs number of generations)

3.5 Conclusion

We propose a GA based method for winner determination with multiple instances of items in CRAs. Thanks to the RemoveRedundancy and RemoveEmptiness functions as well as the modified two-point crossover operator, GAMICRA can produce optimal solutions in a reasonable processing time and optimal procurement cost. Moreover, GAMICRA is a consistent method as it is able to reduce the solution variations over different runs.

As it is obvious that parallel GAs are capable of providing the solutions in a better running time [1], the future target of this part of research is to design our proposed

method based on Parallel GAs (PGAs). Another future direction is to consider other bidding features besides 'Price', such as 'Seller Reputation', 'Delivery Time', 'Feedback from Other Buyers' and 'Warranty'. The winner will then be determined according to these criteria.

Chapter 4

MU-TA-SO-MR CRA Part-1

4.1 Summary

Winner(s) determination in Combinatorial Reverse Auctions (CRAs) is a very appealing application in e-commerce but very challenging especially when multiple attributes of multiple instances of items are considered. The difficulty here is to return the optimal solution to this hard optimization problem in a reasonable computation time. In this part of our research, we make this problem more interesting by considering all-units discounts on attributes and solving it using Genetic Algorithms (GAs). We also consider the availability of instances of items in sellers' stock. In order to evaluate the performance of our proposed method, we conduct several experiments on randomly generated instances. The results clearly demonstrate the efficiency of our method in determining the winner(s) with an optimal procurement cost in an efficient processing time.

4.2 Introduction

The main purpose in combinatorial auctions considering multiple items is to increase the efficiency of bid allocation which corresponds to minimizing the procurement cost in

a reasonable computation time [6, 58, 91] when determining the winner [29]. Winner determination is an NP-complete problem [58, 91]. It is one of the main challenges in combinatorial auctions [6]. Combinatorial auctions have been applied to a procurement scenario such as travel packages and transportation [54, 63]. Hsieh and Tsai developed a Lagrangian heuristic method to tackle combinatorial auctions [58]. Ant algorithms have been defined by Sitarz for the same purpose [58]. GAs have been used to solve combinatorial optimization problems [58], such as job scheduling [66], quadratic assignment problem [78] and travelling salesman problem [85]. In [69], we propose a GA based method, GACRA, and apply that method to solve winner determination in a CRA by considering multiple items. In a CRA, there is one buyer and several sellers. The buyer specifies his requirements and the sellers compete to win. In [69], we introduce two repairing methods, RemoveRedundancy and RemoveEmptiness, along with a modified two-point crossover operator. In that paper, we demonstrate that GACRA is better in terms of computation time and procurement cost when compared to another GA-based winner determination method [58]. In [68], we conduct several statistical experiments and the results show that GACRA is a consistent method. In [70], we solve CRA by another GA-based method, GAMICRA. In this latter research, we modify the configurations of chromosomes and the fitness function. In addition, we also improve RemoveRedundancy and RemoveEmptiness to consider multiple instances of items.

In this part, our target is twofold: (1) to solve the winner determination problem for multiple instances of items in the context of CRA, and (2) to find the winner(s) in a reasonable computation time and reduced procurement cost. In addition to the features

of CRA that have been addressed in [70], we have included some new interesting dimensions. First, we consider two attributes, price and delivery rate. We also consider all-units discount strategy [22] on both price and delivery rate. In our case, the problem is also multi-sourcing as in [70]. The buyer can buy different items from different sellers. He can also buy instances of the same item from different sellers. In addition, we consider various situations related to sellers' stock such as the number of available instances of items provided by a given seller is (a) greater than or (b) less than the buyer's requirement or (c) the seller is out of that item. Also, the maximum price constraint of the buyer as well as the minimum price constraint of the sellers for each item instance is taken into consideration. To tackle these additional features we define the chromosome representation based on the number of items and item instances. We also define the fitness function and keep it simple enough in order to maintain a reasonable computation time. In order to evaluate the time performance of our method to return the best procurement cost, we conduct several experiments on randomly generated instances. Before conducting these experiments, we tune the different parameters of our proposed method to their best. The results we report in this research clearly demonstrate that our method is a prominent one. After each round we rank the bids and show the prices to the buyer and the sellers' as well. We also present the breakdowns of the number of instances of the items the seller(s) will provide. We perform a statistical analysis to investigate the behavior of our proposed method and show that it does not suffer from the inconsistency issue.

CRA is a computationally complex problem and by considering more parameters we add extra dimensions to the complexity. For instance, if there are p items, m instances

and n sellers, then the search space is $p \times m \times n$. In addition, the solving procedure needs to satisfy both the buyer's and sellers' constraints which add to the complexity of the problem. Supplier selection is one of the most important in a multi-criteria decision problem [22, 88]. In addition, the problem becomes more complex when considering more than one attribute of items and the discount strategy [88].

The rest of this research is organized as follows. In Section 4.3, we state the problem we are tackling in details. In Section 4.4, our proposed method is presented and described through an example. Section 4.5 reports the experimental study we conducted to evaluate the computation time, procurement cost and the consistency of our method. Finally in Section 4.6, concluding remarks and future research directions are listed.

4.3 Problem Statement

We consider a CRA with multi-attributes of multiple instances of multiple items. To determine the winner we consider the items price along with the delivery rates and the respective discounts. Figure 4.1 illustrates a sample scenario with the buyer's request and sellers' stock. Here, the buyer requests two instances of item 1, three instances of item 2 and two instances of item 3. Then, he specifies the maximum buying price for each instance of item and terminating condition as follows. The maximum price for each instance of item1, item2 and item3 is 500, 700, and 200 respectively and the maximum number of round is 5.

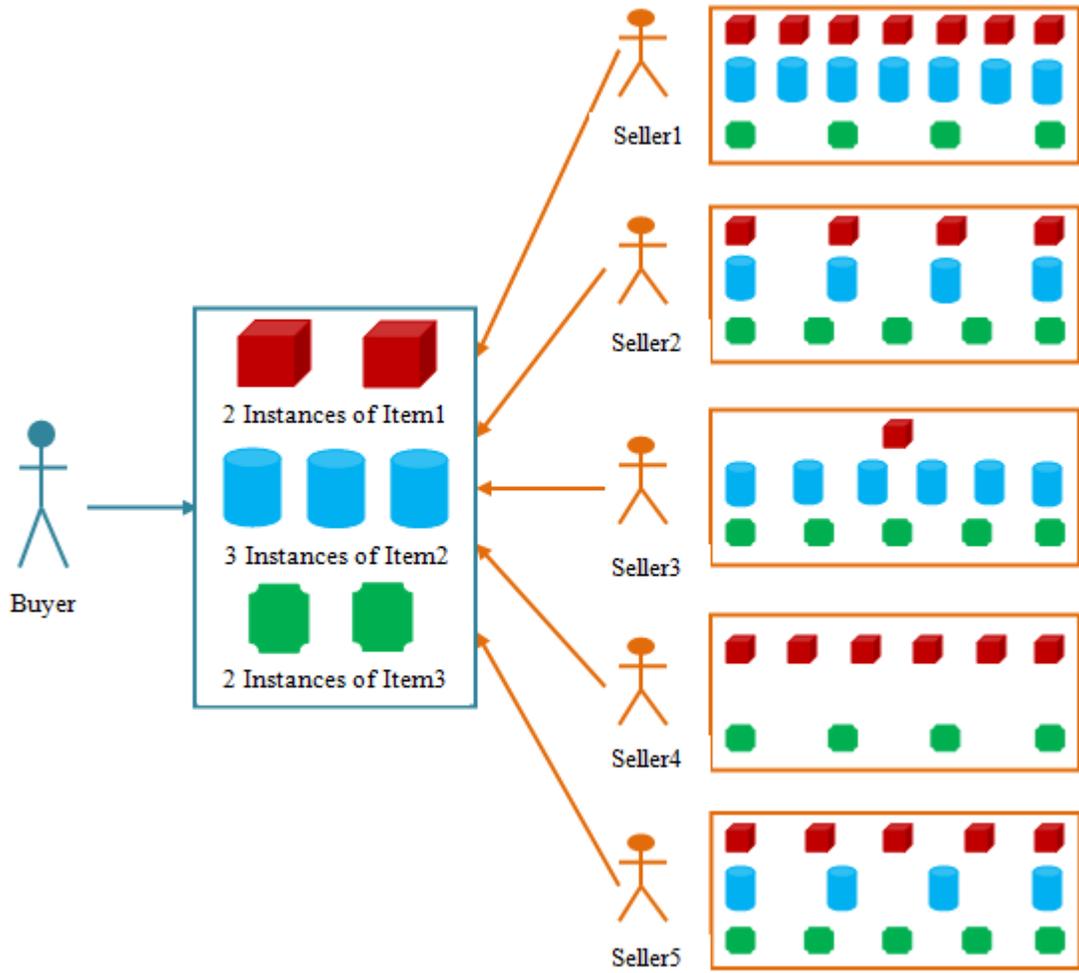


Figure 4.1: A sample scenario in CRAs

These are treated as the buyer’s constraints. We consider the minimum price the sellers can afford for each instance of items as shown in Table 4.1. Hence, there is an implicit constraint that the bid price should be greater than or equal to the Minimum Price (MP) of the seller and less than or equal to the buyer’s maximum price. Table 4.1 also shows the Delivery Rate (DR).

Table 4.1: Sellers' Minimum Price (MP) and Delivery Rate (DR)

	Item1		Item2		Item3	
	MP	DR	MP	DR	MP	DR
S1	400	40	600	60	120	10
S2	350	35	650	65	100	10
S3	380	35	580	55	150	15
S4	350	55	-	-	150	15
S5	400	35	650	65	120	10

Table 4.2: Sellers' all-units discount on bid price

	Discount Rate					
	Item1		Item2		Item3	
	>2	>5	>2	>5	>2	>5
S1	5%	10%	4%	9%	5%	10%
S2	5%	9%	5%	8%	5%	9%
S3	-	-	5%	9%	5%	10%
S4	5%	9%	-	-	4%	10%
S5	5%	10%	5%	10%	4%	9%

Table 4.3: Sellers’ all-units discount on delivery rate

	Discount Rate					
	Item1		Item2		Item3	
	>2	>5	>2	>5	>2	>5
S1	50%	100%	40%	90%	50%	100%
S2	50%	90%	50%	80%	50%	90%
S3	-	-	50%	90%	50%	100%
S4	50%	90%	-	-	40%	100%
S5	50%	100%	50%	100%	40%	90%

We use all-units discount strategy. The sellers provide discounts on price and delivery rate under some conditions. Tables 4.2 and 4.3 show the discount rates on bid price and delivery rate respectively. Given all these data, the goal is to determine the winner(s) with optimal price in a reasonable computation time.

4.4 Proposed Method

We discuss our proposed method using the sample scenario of Figure 4.1. We generate the required number of bits (rb) to represent each seller's item instance using the following rule:

$$n \leq 2^{rb} \tag{4.1}$$

where n is the number of sellers. Here $n = 5$ and $rb = 3$.

Algorithm 4.1: Proposed Method (p : number of items, m_p : number of instances of items p specified by the buyer, s_p : number of available instances of items p possessed by sellers, n : number of sellers, δ : number of generations, γ : number of rounds, α : crossover rate, β : mutation rate, ζ : number of chromosomes, $maxPrice$: set of maximum prices of instances of items specified by the buyer, $minPrice$: set of minimum prices of instances of items provided by sellers, d : set of delivery rate, discount rate on bid price and delivery rate)

1. $r = 1$;
do{
2. $t = 1$;
3. bidGenerator ($maxPrice, minPrice, p, n$);
//checks price constraints and generates bid prices, $bidPrice$
4. chromosomeGenerator(m_p, s_p, p, n, ζ);
//generates feasible initial chromosomes
5. fitnessOperation(X_ζ : set of initial chromosomes, $price$: calculated from $bidPrice$ and d);
//computes fitness values of chromosomes, fv
do{
6. selectionOperation(X_ζ, fv);
//selects chromosomes using gambling-wheel disk method
7. crossoverOperation(X_ζ^s : set of chromosomes after selection, fv);
//generates child chromosomes from parent chromosomes with modified two-
//point crossover considering crossover rate, α

```

8. mutationOperation( $X_{\zeta}^c$ : set of chromosomes after crossover,  $fv$ );
   //mutates chromosomes considering mutation rate,  $\beta$ 
9. newChromosomeGenerator( $X_{\zeta}$ ,  $NX_{\zeta}$ : set of chromosomes after mutation);
   //removes duplicity and selects better chromosomes from both initial and new
   //chromosomes of generation,  $\delta$ 
   }(while  $t \leq \delta$ );
   }(while  $r \leq \gamma$ );
10. return winner(s);
    //returns winner(s) with minimum bid price

```

Figure 4.2: Proposed solving algorithm

Assume p = the number of items and m_p = the number of instances of $item_p$ where $1 \leq P \leq p$, then the length of chromosome is calculated by the following equation:

$$lengthChromosome = \sum_{P=1}^p m_p \times rb \quad (4.2)$$

Figure 4.2 shows our proposed GA-based solving algorithm. The procedures bidGenerator, chromosomeGenerator and newChromosomeGenerator are presented in figures 4.3 - 4.5. Here, the number of chromosomes is equal to 4. Table 4.4 presents the first round bid price and table 4.5 shows the randomly generated initial chromosome representation. For each instance, a random number from $[1, n]$ is generated and its correspondence binary representation according to rb becomes the part of a chromosome.

Table 4.4: Bid price generation

Bid Price of Each Instance	S1	S2	S3	S4	S5
Item1	480	450	460	430	490
Item2	670	680	660	-	660
Item3	180	190	170	180	150

According to X1 in table 4.5, both seller 1 and seller 2 will supply a single instance of both item 1 and item 3. Seller 2 will supply all three instances of item 2. X1 and X4 are feasible but X2 and X3 are not. In X2, seller 3 is selected for two instances of item 1 whereas he has only one instance in his stock. In X3, seller 4 is selected for one instance of item 2 whereas he has no instance of item 2 in his stock. Assume the following valid chromosomes are generated after removing infeasibility as shown in Algorithm 4.3:

X1: 001 010 010 010 010 001 010

X2: 011 011 101 011 001 101 100

X3: 010 011 010 101 101 011 011

X4: 101 101 001 001 001 001 001

Table 4.5: Initial chromosomes

Chromosome	Item1		Item2			Item3	
	Ins1	Ins2	Ins1	Ins2	Ins3	Ins1	Ins2
X1	001	010	010	010	010	001	010
X2	011	011	101	011	001	101	100
X3	010	011	100	101	101	011	011
X4	101	101	001	001	001	001	001

To evaluate the fitness value of each chromosome, the following fitness function is applied.

$$F(X_i) = 1 / \sum_{N=1}^n \sum_{P=1}^p b_{NP} \times l_{NP} \quad (4.3)$$

where l_{NP} is the number of instances of item P for seller N and b_{NP} is the price (discounted bid price + discounted delivery rate) of item P submitted by seller N .

Gambling Wheel Disk [29] method is used as a selection strategy. After the selection process, the following chromosomes are selected based on the fitness values according to equation (4.3).

X1: 001 010 010 010 010 001 010 (previous X1)

X2: 011 011 101 011 001 101 100 (previous X2)

X3: 101 101 001 001 001 001 001 (previous X4)

X4: 101 101 001 001 001 001 001 (previous X4)

The modified two-point crossover [69] is then applied as shown in figure 4.6(a).

After crossover, the following chromosomes are generated.

X1: 001 010 101 011 001 001 010

X2: 101 100 010 010 010 011 011

X3: 101 101 001 001 001 001 001

X4: 101 101 001 001 001 001 001

```
Algorithm 4.2: bidGenerator (maxPrice, minPrice, p, n)  
bidPrice =  $+\infty$ ;  
//initial value before generating bid price in the first round  
  
1. for each n  
2. { for each p  
3.   { if(maxPrice > bidPrice)  
4.     maxPrice = bidPrice;  
5.     generateBid(maxPrice, minPrice);  
        //generates bid price less than or equal to maxPrice and greater than or  
        //equal to minPrice  
6.   }  
7. }
```

Figure 4.3: bidGenerator function

Algorithm 4.3: chromosomeGenerator (m_p, s_p, p, n, ζ)

```
1. for each  $\zeta$ 
2. { for each  $p$ 
3.   { for each  $m_p$ 
4.     { generateRN( $n$ );
           //generates random number between 1 and  $n$ 
5.     }
6.   }
7.   buildChromosome( $nm_p$ : set of random numbers);
8.   //generates chromosome
9.   checkFeasibility( $m_p, s_p$ );
10.  //makes the chromosome feasible
11. }
```

Figure 4.4: chromosomeGenerator function

Algorithm 4.4: newChromosomeGenerator (X_ζ, NX_ζ, ζ)

```
1.  $X_\zeta \cup NX_\zeta$ ;
2. uniqueCheck( $X_\zeta, NX_\zeta$ );
   //returns only the unique chromosomes and  $xn$ , the number of unique
   //chromosomes
3. if( $xn < \zeta$ )
4. { injectChromosome();
   //generates the remaining new chromosomes by invoking
```

```
//chromosomeGenerator()
}
5. if( $xn > \zeta$  and  $xn < 2\zeta$ )
6. { removeChromosome();
    //removes additional chromosomes according to worse fitness values
7. }
8. return  $\zeta$  chromosomes according to better fitness values
```

Figure 4.5: newChromosomeGenerator function

The mutation is then performed as shown in figure 4.6(b). Table 4.6 shows the result of the first generation for round 1. According to this result, seller 1 will provide two instances of item 1 and three instances of item 2 while seller 5 will provide two instances of item 3. Algorithm 4.1 is repeated until the specified number of generations and rounds are completed.

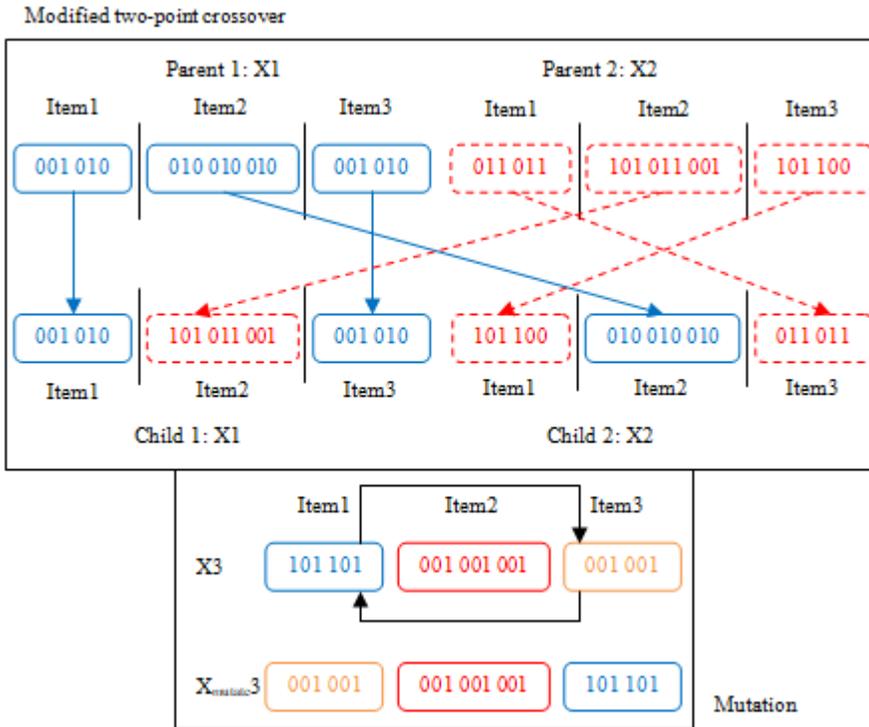


Figure 4.6: Modified two-point crossover and Mutation

Table 4.6: Bid result

Rank	Chromosome	Item1	Item2	Item3	Price
1	X1	S1(2)	S1(3)	S5(2)	3398
2	X2	S4(1), S5(1)	S2(3)	S3(2)	3416
3	X3	S5(2)	S1(3)	S1(2)	3468
4	X4	S1(1), S2(1)	S1(1), S3(1), S5(1)	S1(1), S2(1)	3565

4.5 Experimentation

Our proposed method has been implemented in Java and is executed on an Intel (R) Core (TM) i3-2330M CPU with 4 GB of RAM and 2.20 GHz of processor speed. The experiments are conducted on randomly generated instances. *maxPrice* is generated randomly from [100, 1000] whereas *minPrice* is generated from [50% of *maxPrice*, 75% of *maxPrice*]. For delivery rate we use a value from [10% of *minPrice*, 25% of *minPrice*]. 3% to 5% discount on price is considered if more than 10 instances of a given item are supplied by a seller and 6% to 10% when it is more than 25. We also consider 30% to 50% discounts on delivery rate when the seller supplies more than 10 instances of an item and 60% to 100% if it is more than 25. Values of s_p are generated from [0, 30].

4.5.1 Experiment 1: Parameter Tuning

The goal of this first experiment is to tune the parameters of our method to their best values for the specific instances we are using. We perform 27 tests by varying the number of chromosomes from 50 to 200, the crossover rate from 0.5 to 0.7 and the mutation rate from 0.01 to 0.1. For this experiment, we use the following parameters and settings: number of sellers = 100; number of items = 10; number of instances from [1-100]; number of generations = 50 and number of rounds = 5. From the results of this experiment that are the average of 20 runs, we have decided to use the configuration of test21 for the next experiments as with this configuration, the algorithm performs the best. The configuration of test21 is: number of chromosome = 200, crossover rate = 0.5 and mutation rate = 0.1.

4.5.2 Experiment 2: Statistical Analysis

Figure 4.7 illustrates the statistical analysis of the proposed method. It depicts the average price with maximum and minimum values of generations. It also shows the error bars with a confidence level of 95%. From this figure it is clearly notable that the method is able to control the solution variations and stabilizes after some generations. Notice that the minimum bid price remains constant after 70 generations which means that the best solution found by the method might be the optimal solution. The results shown in this experiment are the average values of 20 runs. We also perform the same experiment for 50 runs but find no major improvement. For this reason we continue the next experiments for 20 runs and analyze the results based on the average values.

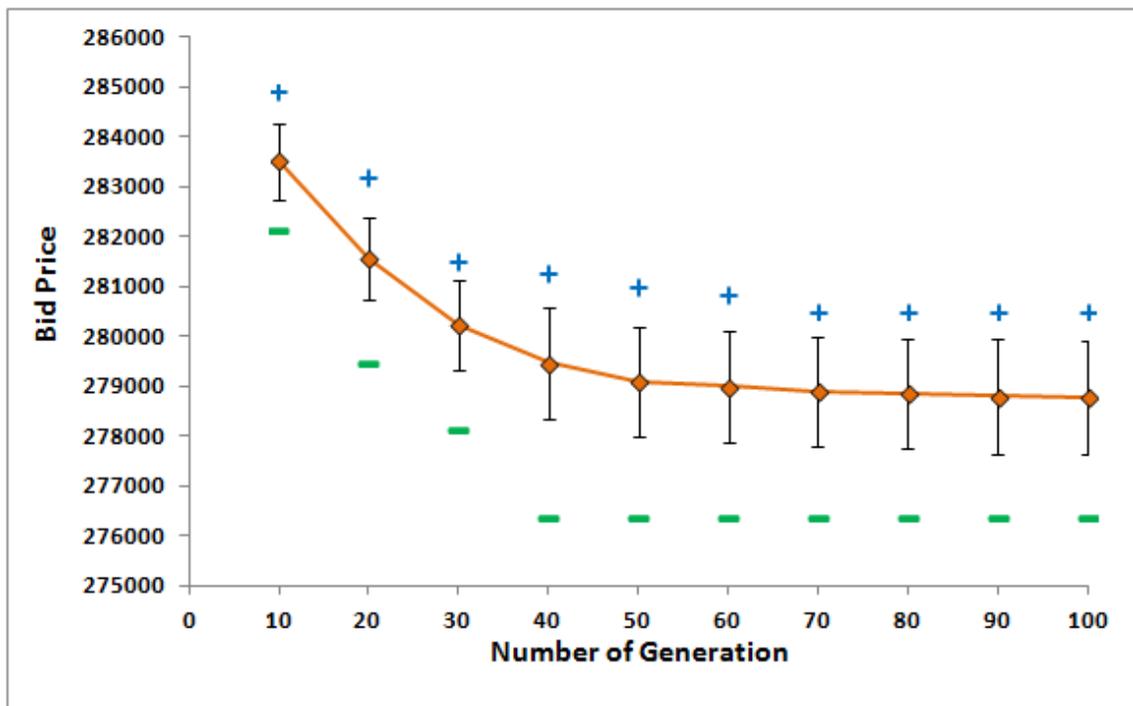


Figure 4.7: Statistical analysis of the proposed method

4.5.3 Experiment 3: Number of Generations

We use the following settings for experiments 3 and 4. The maximum number of generations and rounds are 50 and 5 respectively. The number of sellers and items are 100 and 10 respectively. The range of instances is from 1 to 100. Figure 4.8 reports the results of experiment 3. Figure 4.8.a shows how the bid price improves with the increase of the number of generations. As we can easily see the cost significantly improves in the first generations and stabilizes afterwards. Figure 4.8.b shows how the computation time varies with the increase of the number of generations.

4.5.4 Experiment 4: Number of Sellers and Items

Figure 4.9 shows how the computation time increases when we increase the number of sellers as well as the number of items. In this regard our method is capable of producing the solutions in a very short time even for large number of sellers and items.

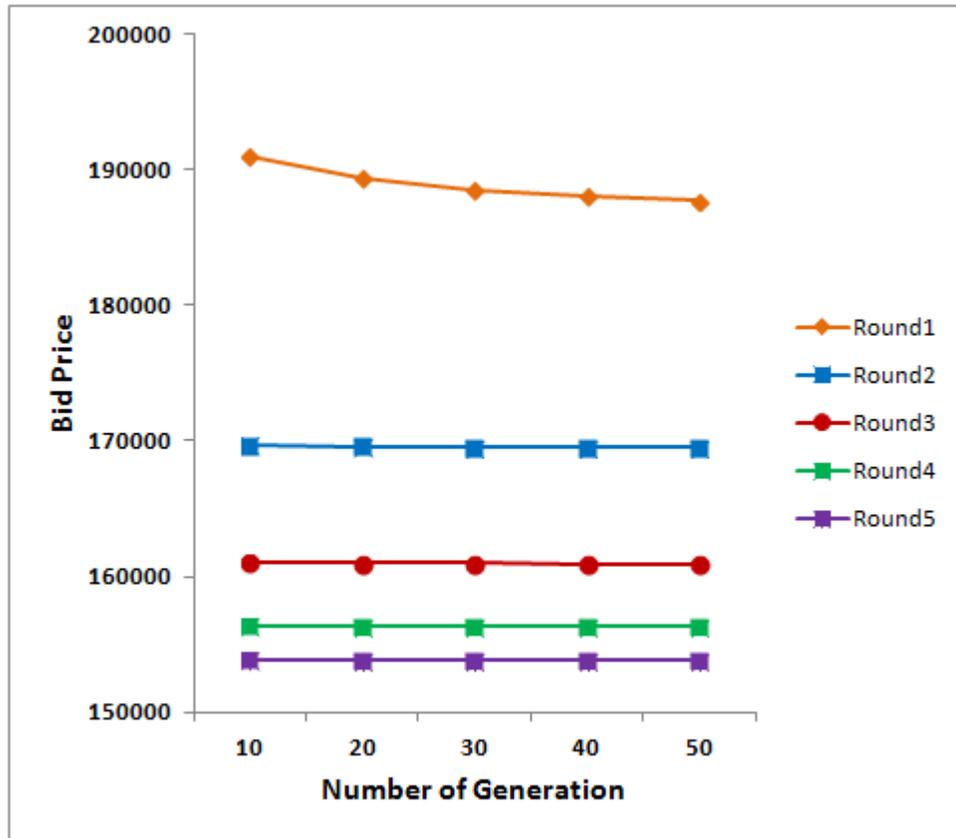


Figure 4.8(a): Bid price vs number of generations

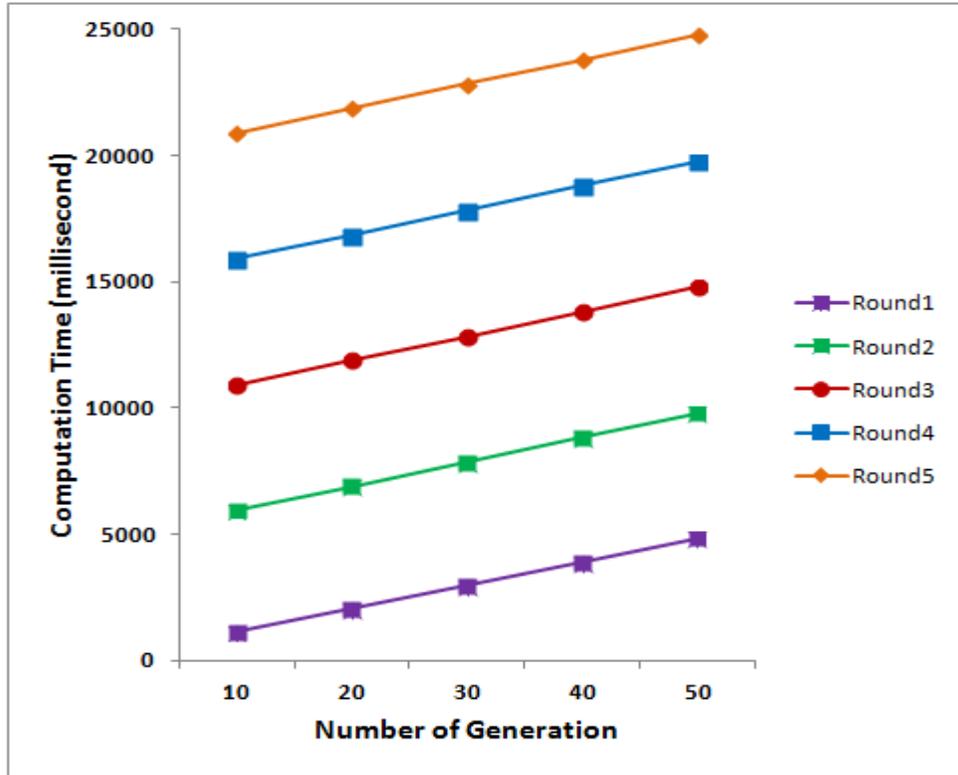


Figure 4.8(b): Computation time vs number of generations

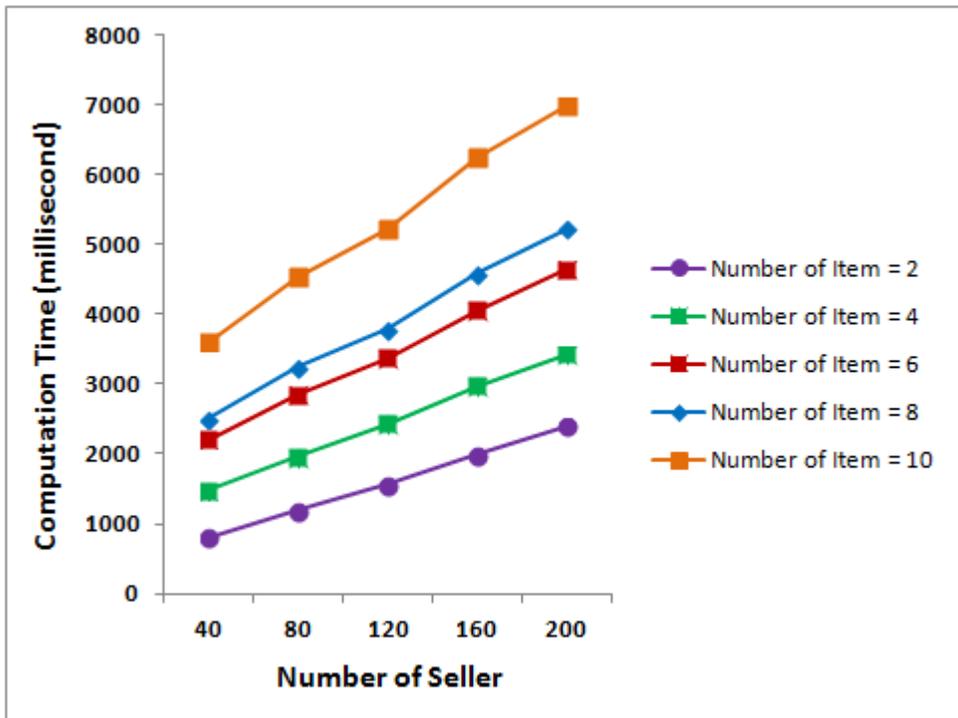


Figure 4.9(a): Computation time vs number of sellers

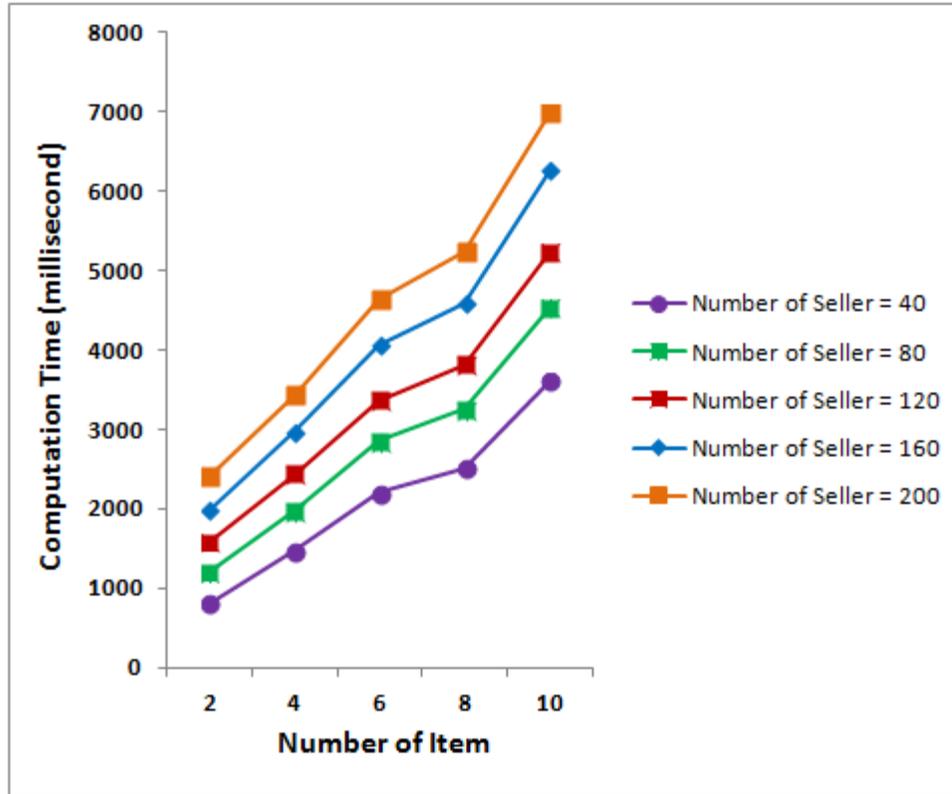


Figure 4.9(b): Computation time vs number of items

4.6 Conclusion

E-commerce and resource allocation problems need real-time response [36]. In these problems, the instances are large and the solutions are needed in a very short time. While the exact algorithms guarantee to return the optimal solution, they often suffer from their exponential time cost especially for larger instances [36]. A good alternative is the approximation methods that still produce high-quality solutions in practice with a reasonable computation time [36]. For this reason, we have opted for evolutionary techniques to solve the winner determination problem as they have the ability to produce good solutions from the very first generations. We propose a GA based method for winner determination with two attributes of multiple instances of items in CRAs

considering all-units discount on both of these attributes. We show that our proposed method can produce optimal solutions in a reasonable computation time. Moreover, it shows that the method is a consistent one as it is able to reduce the solution variations over different runs.

As future works of this part of our research, we will apply an exact method to get the optimal solution for the instances we used in the experiments. This will help us evaluate the quality of the solutions (procurement cost) returned by our proposed method. We will also compare the time performance of our method to the well known techniques used to solve related problems [28, 62]. In this regard, we have to acknowledge that we tackle unique instances that have not been solved before. Since parallel GAs are capable of providing the solutions in a better computation time especially when some variable ordering heuristics are used [1, 2, 51], one of the future works is to use Parallel GAs. Another promising direction is to consider, in addition to 'price' and 'delivery rate', more attributes such as 'delivery time', 'seller reputation', and 'warranty'. We will also investigate the applicability of other meta-heuristics together with GAs [57]. Finally we will study several diversity models while using GAs to find the most appropriate one for our problem [33].

Chapter 5

MU-TA-SO-MR CRA Part-2

5.1 Summary

Winner(s) determination in online reverse auctions is a very appealing e-commerce application. This is a combinatorial optimization problem where the goal is to find an optimal solution meeting a set of requirements and minimizing a given procurement cost. This problem is hard to tackle especially when multiple attributes of instances of items are considered together with additional constraints, such as seller's stocks and discount rate. The challenge here is to determine the optimal solution in a reasonable computation time. Solving this problem with a systematic method will guarantee the optimality of the returned solution but comes with an exponential time cost. On the other hand, approximation techniques such as evolutionary algorithms are faster but trade the quality of the solution returned for the running time. In this study, we conduct a comparative study of several exact and evolutionary techniques that have been proposed to solve various instances of the combinatorial reverse auction problem. In particular, we show that a recent method based on genetic algorithms outperforms some other methods in terms of time efficiency while returning a near to optimal solution in most of the cases.

5.2 Introduction

The ultimate purpose of combinatorial auctions considering multiple items is to increase the efficiency of bid allocation; the latter corresponds to minimizing the procurement cost in a reasonable computation time when selecting the winner [5, 6, 19, 29, 58, 91]. Winner determination is an NP-complete problem [58, 91] and can be even more challenging if we consider additional constraints, such as seller's stocks and discount rate. Solving this problem with a systematic method will guarantee the optimal solution but comes with an exponential time cost. On the other hand, approximation techniques such as evolutionary algorithms are faster but trade the quality of the solution returned for the running time. In this regard, several exact and approximation methods have been proposed in the past to tackle this hard to solve problem. In [54, 63] combinatorial auctions have been applied to procurement scenarios such as travel packages and transportation. Hsieh and Tsai developed a Lagrangian heuristic method to tackle combinatorial auctions [58]. Ant Colony algorithms have been defined by Sitarz for the same purpose [58].

Combinatorial Reverse Auctions (CRAs) are a particular case of combinatorial auctions that can be of three types based on the number of attributes, items and instances: (1) single attribute multiple items with single instance per item, (2) single attribute multiple items with multiple instances per item and (3) multi-attribute multiple items with multiple instances per item. Unlike combinatorial auctions, fewer research works were dedicated to CRAs. These contributions can be summarized as follows.

In [69], we tackle the first type of CRAs by proposing a GA based method called Genetic Algorithms for Combinatorial Reverse Auctions (GACRA) to solve the winner

determination problem. GACRA is based on Genetic Algorithms (GAs) and includes two repairing methods respectively called RemoveRedundancy and RemoveEmptiness as well as a modified two-point crossover operator. We conduct several experiments and based on the results we demonstrate that GACRA is better in terms of procurement cost and processing time comparing to another GA-based winner determination method [58]. Following the work reported in [69], we conduct several statistical experiments demonstrating that GACRA is a consistent method [68].

We address the second type of CRAs in [70] with a new GA-based method called Genetic Algorithms for Multiple Instances of Items in Combinatorial Reverse Auctions (GAMICRA). GAMICRA extends GACRA by improving the RemoveRedundancy and RemoveEmptiness methods. We update the chromosome representation as well as the definition of the fitness function to deal with multiple instances of items. Based on the results of several experiments on many CRAs instances, we show that GAMICRA is a consistent method which is capable of determining the winner in a very efficient computation time.

In [72], we address the third type of CRAs by considering two attributes, namely price and delivery rate, along with multiple instances of items. We also consider all-units discount strategy on both price and delivery rate which, undoubtedly, makes the problem even more challenging to solve as shown in [88]. In this latter paper, we treat the problem as multi-sourcing where supplier selection is an important task in a multi-criteria decision problem [22, 88]. Here, the buyer can order different items from different sellers. He can also purchase instances of the same item from different sellers. Moreover, we consider various situations which are related to the sellers' stock, such as

the number of available instances of items provided by a given seller is (a) greater than or (b) less than the buyer's requirement or (c) the seller is out of that item. Also, the maximum price constraints of the buyer as well as the minimum price constraint of the sellers for each item instance were taken into consideration. To tackle these additional features, we define the chromosome representation based on the number of items and item instances. We also define the fitness function and keep it simple enough in order to maintain a reasonable processing time. Here, we propose a GA-based method to solve the winner determination problem. In order to evaluate the time performance of this method to return the best procurement cost, we conduct several experiments on randomly generated instances after tuning the parameters to their best. The results of these experiments clearly show that the proposed method is efficient and consistent.

The third type of CRAs is computationally complex. For instance, if there are j items, i instances (where $I = \sum_{j=1}^j i_j$ and i_j is the number of instances of item J) and k sellers, then the search space is k^I (if $j = 10$, $I = 50$, and $k = 100$, the number of potential solution space is 100^{50}). Moreover, to determine the winner, the solving procedure needs to satisfy other bidding, buyer and sellers' constraints.

While we address this latter problem with the performance results reported in [72], comparison with the existing methods has not been conducted. Moreover there is no evidence that the solutions returned in the experiments are the optimal ones.

In this study, our goal is to address the above two issues. At first, we perform comparative experiments with a recent evolutionary technique proposed in [62] for solving a similar problem. This new method is an improved ant colony algorithm called Improved Ant Colony (IAC). IAC considers the Max-Min pheromone and dynamic

transition strategy. The problem tackled by IAC is, however, different from the one addressed by GAMICRA as discount strategy, multiple instances of items and sellers' stock are not considered. In [62], IAC has been compared to two other methods: Enumeration Algorithm with Backtracking (EAB) and the traditional Ant Colony (AC) algorithm. It has been shown that IAC outperforms those two methods. We show that GAMICRA is superior in running time than AC, EAB and IAC. Besides evolutionary algorithms, we compare GAMICRA with a branch and bound technique proposed in [28] for solving multi-unit combinatorial auctions. Here again, based on the results returned we show that GAMICRA outperforms the branch and bound technique. To tackle the second issue, we implement an exact algorithm and use it to evaluate the optimality of the returned solutions. From the result of the comparative experiments (for both the exact method and GAMICRA), we demonstrate that GAMICRA is able to produce very close to optimal solutions.

The rest of this study is organized as follows. In Section 2, we state the problem in details. We also present the problem formulation for GAMICRA and describe our algorithm. Section 3 reports the experimental study we conducted to address the above two issues. Finally in Section 4, concluding remarks and future research directions are depicted.

5.3 GAMICRA for Winner Determination

In a combinatorial reverse auction, there is one buyer and several sellers. The buyer specifies his requirements and the sellers compete to win. In [72], GAMICRA considers multiple attributes (price and delivery rate), multiple instances, multiple items and all-units discounts. It also considers the buyer's constraints such as maximum price of each

instance of item and terminating condition. Bidders' constraints (minimum price for each instance of item, available stock and discount rates) are also taken into account. Given all these data, the goal is to determine the winner(s) in a reasonable computation time. Based on [72], this optimization problem can be formulated as shown in Table 5.1.

In order to check the quality of the solution returned by GAMICRA we implemented an Exact Algorithm (EA) that we present in Figure 5.1 together with the pseudo code of GAMICRA.

Table 5.1: Problem formulation for GAMICRA

Variables	
$nb_sellers$	Number of sellers
nb_items	Number of items
$nb_instances_j$	Number of instances requested by the buyer for item j
$capacity_instances_{jk}$	Number of instances of item j , seller k has
$minPrice_{jk}$	The lowest price, the kth seller can offer for the jth item
$maxPrice_j$	The highest price, the buyer can pay for the jth item
$Bid(X_{ijk})$	Bid price, the kth seller bids for the ith instance of jth item
max_rounds	The maximum number of rounds used as a terminating condition
Constraints	
<ul style="list-style-type: none"> • X_{ijk}: 1 if the ith instance of the jth item of the kth seller is selected and 0 	

otherwise.

$$1 \leq i \leq \text{capacity_instance}_{jk}$$

$$1 \leq j \leq \text{nb_items}$$

$$1 \leq k \leq \text{nb_sellers}$$

- $\sum_{k=1}^{\text{nb_sellers}} \sum_{i=1}^{\text{capacity_instances}_{jk}} X_{ijk} = \text{nb_instances}_j$

$$1 \leq j \leq \text{nb_items}$$

- $\text{minPrice}_{jk} \leq \text{Bid}(X_{ijk}) \leq \text{maxPrice}_j$

Objective Function

$$\min \sum_{k=1}^{\text{nb_sellers}} \sum_{j=1}^{\text{nb_items}} \sum_{i=1}^{\text{capacity_instances}_{jk}} \text{Bid}(X_{ijk})$$

Algorithm 5.1: EA and GAMICRA

EA:

Begin:

generate solution space;

initialize feasible solutions by considering bidding, buyer and sellers' constraints;

evaluate feasible solutions;

return the winner(s);

```

End;

GAMICRA:

Begin:

    generation ← 0;

    generate bids;

    initialize chromosomes X(generation);

    evaluate X(generation);

    while (not maximum generation) do

        Begin:

            generation ← generation + 1;

            select X(generation) from X(generation – 1) by Gambling Wheel Disk
method

            [10];

            recombine X(generation) by modified two-point crossover and mutation;

            evaluate X(generation);

        End;

        Return the winner(s);

    End;

End;

```

Figure 5.1: EA and GAMICRA pseudo code

5.4 Experimentation

EA has been implemented in Java and executed on an Intel (R) Core (TM) i3-2330 M CPU with 4 GB of RAM and 2.20 GHz of processor speed. The experiments are

conducted on the same data used in [72]. For GAMICRA, all results returned are the average values of 20 runs.

5.4.1 Experiment 1: Comparison with IAC, AC and EAB

In this experiment, we compare the computation time of GAMICRA with IAC, AC and EAB algorithms [62]. Here, ζ and δ denote number of chromosomes and number of generations respectively.

Table 5.2 presents the comparative computation time of these methods. In this table, “-” indicates a non-existing value. The test results for IAC, AC and EAB are taken from [62]. As we can easily see from the table, GAMICRA outperforms all these evolutionary techniques.

Table 5.2: The Comparison of GAMICRA, IAC, AC and EAB

Algorithm	ζ	δ	<i>nb_sellers</i>	<i>nb_items</i>	Computation time (second)
EAB	-	-	100	30	3
AC	50	50	100	30	4
IAC	50	50	100	30	9
GAMICRA	50	50	100	30	0.83

5.4.2 Experiment 2: Comparison with Branch and Bound

In this experiment, we compare the computation time of GAMICRA with a method based on Branch and Bound [28]. The tests have been conducted on the same instances

but with a different computer (450 MHz of processor speed for Branch and Bound vs 2.20 GHz for GAMICRA). Despite the difference in processor speed, we see again here that GAMICRA is superior to the branch and bound technique (Table 5.3).

Table 5.3: The Comparison of GAMICRA and Branch and Bound

Algorithm	<i>nb_sellers</i>	<i>nb_items</i>	$\sum nb_instances_j$	Computation time (second)
Branch and Bound based Method	100	10	25	>100
GAMICRA	100	10	25	1.7

5.4.3 Experiment 3: Comparison with the Exact Algorithm

In order to assess the quality of the solution returned by GAMICRA, we compare it with the EA reported in Algorithm 5.2 above. Two types of tests are conducted and the results are respectively reported in Tables 5.4 and 5.5.

Table 5.4: Comparison of GAMICRA and EA when varying the number of sellers

Algorithm	Bid price				Computation time (millisecond)			
	Test 1	Test 2	Test 3	Test 4	Test 1	Test 2	Test 3	Test 4
					1	2	3	4
	Number of sellers				Number of sellers			
	10	20	30	40	10	20	30	40
EA	1472	1538	1412	915	60	392	4906	42137
GAMICRA	1474 (99.86%)	1555 (98.91%)	1462 (96.58%)	988 (92.61%)	39	46	55	62

Table 5.5: Comparison of GAMICRA and EA when varying the total number items instances

Algorithm	Bid price			Computation time (millisecond)		
	Test 1	Test 2	Test 3	Test 1	Test 2	Test 3
	Number of total instances			Number of total instances		
	2	3	4	2	3	4
EA	621	861	915	90	270	42137
GAMICRA	621 (100%)	865 (99.54%)	988 (92.61%)	59	61.5	62

In the first type of tests, we measure bid price, computation time and the quality of the solution returned by GAMICRA and EA when varying the number of sellers. In these tests we use the following parameters: number of items = 3, total number of instances = 4 (2 instances of item1, 1 instance of item2, and 1 instance of item3), and number of sellers = 10, 20, 30, and 40. For example, in Test 4 (Table 5.4), there are 40 sellers and 4 instances of items, hence there are 40^4 (2560000) potential solutions. To determine the winner, EA takes 42.137 s while GAMICRA takes only 0.062 s. On the other hand, EA returns the optimal bid price of 915 and GAMICRA returns 988 which is close to the optimal solution. Hence, the accuracy $((EA \text{ result}/GAMICRA \text{ result}) \times 100 \%)$ of GAMICRA is 92.61 %. In the second type of tests, we vary the total number of instances of items from 2 to 4, and fix the number of sellers to 40. From the results of this second experiment reported in Table 5, we can easily see that, in most of the time, GAMICRA returns near to optimal solution. It also proves that GAMICRA is not trapped in local optima.

5.5 Conclusion

In this study, we compare a very recent GA-based solution for CRAs with exact and evolutionary methods. Based on the results of the experimental comparative study, we conclude that GAMICRA outperforms all these methods. Moreover, to assess the quality of the solution returned by GAMICRA we compare this latter with an exact method that we have implemented. The results of this comparative experiment clearly show that GAMICRA always produce near to optimal solutions and in some cases the optimum is reached.

Since parallel GAs [52, 56] are capable of providing the solutions in a better computation time [1, 2]; we will design, in the near future, a parallel version of GAMICRA. Another promising direction is to consider more attributes such as ‘delivery time’, ‘seller reputation’, and ‘warranty’ as reported in [71]. We will also investigate the applicability of other meta-heuristics such as Simulated Annealing, Hill Climbing, and Late Acceptance together with GAs [57]. Finally we will study and apply several diversity models such as Crowding, Restricted Mating, and Ranked Space with GAs [33].

Chapter 6

MU-MA-MO-MR CRA

6.1 Summary

This study introduces an advanced Combinatorial Reverse Auction (CRA), multi-units, multi-attributes and multi-objective, which is subject to buyer and seller trading constraints. Conflicting objectives may occur since the buyer can maximize some attributes and minimize some others. To address the Winner Determination (WD) problem for this type of CRAs, we propose an optimization approach based on genetic algorithms that we integrate with our variants of diversity and elitism strategies to improve the solution quality. Moreover, by maximizing the buyer's revenue, our approach is able to return the best solution for our complex WD problem. We conduct a case study as well as simulated testing to illustrate the importance of the diversity and elitism schemes. We also validate the proposed WD method through simulated experiments by generating large instances of our CRA problem. The experimental results demonstrate on one hand the performance of our WD method in terms of several quality measures, like solution quality, run-time complexity and trade-off between convergence and diversity, and on the other hand, it's significant superiority to well-

known heuristic and exact WD techniques that have been implemented for much simpler CRAs.

6.2 Introduction

6.2.1 Research Scope

To increase their revenue, buyers can purchase several items all together by adopting Combinatorial Reverse Auctions (CRAs). It has been shown that combinatorial markets lead to great economic efficiency [44, 62]. Numerous studies in CRAs focused on items with multiple units but with single attribute [44] whereas some other studies considered items with multiple attributes but with single unit [62]. Multiple attributes along with multiple units have been ignored in the combinatorial setting. Researchers restricted the auction parameters for the sake of simplicity. Determining the winners in traditional CRAs is already a NP-hard problem [62], and because of the computational complexity issue, CRAs are difficult to implement. Some previous research in combinatorial auctions endorsed exact algorithms to search for the optimal winners but endured an exponential time cost. Besides, time increases exponentially with the size of the applications. Few studies applied evolutionary techniques to combinatorial auctions, which produce approximate solutions, still with high quality, in a very efficient processing time [12, 13, 62]. These techniques are necessary for large-scale applications and also for problems that require solutions in a very short time, such as resource allocation, flight scheduling, route planning and wireless communication. In the article [72], the winners are determined for a new CRA scenario: multi-items with multi-units and two quantitative attributes, price and delivery rate. The authors devised a Winner

Determination (WD) method based on Genetic Algorithms (GAs), powerful searching methods, to look for the best solution with the least procurement cost and time. As demonstrated in the study [73], this WD method outperforms other algorithms, exact and evolutionary, which have been deployed for simpler CRAs in terms of time-efficiency.

6.2.2 Multi-Objective Combinatorial Reverse Auctions

In this article, we tackle a more advanced combinatorial procurement auction that involves multiple units, multiple attributes and multiple objectives of heterogeneous items all together with the trading constraints of buyer and sellers. More precisely, our new CRA possesses valuable features, including:

- Items have different attributes.
- We take into account the individuated requirements of buyers, i.e. attribute ranking, objectives and constraints as well as constraints of suppliers.
- A seller competes on any combination of items. Indeed, we allow full and partial bidding. But if a seller bids on a certain item, we assume he has a full stock of that item.
- A buyer may maximize some attributes (like increasing the item quality) and minimize some others (like reducing the item price). So, we are in the presence of conflicting objectives. Consequently, we view our CRA as a Multi-Objective Optimization (MOO) problem for which we need to optimize several conflicting criteria simultaneously.
- We consider a multi-sourcing auction where the bid-taker may purchase several units of the same item from different sellers.

6.2.3 Winner Determination

It has been an ongoing research to solve multi-objective decision problems with evolutionary algorithms because the latter are regarded as the most suitable methods for this kind of problems. Finding the set of near-to-optimal solutions and selecting the best one are both crucial. Researchers came up with two ways: priori and posterior approaches [20]. The former first elicits the user preferences and then performs the optimization (via a composite function) to produce a single best solution. This approach heavily depends on the individual's preferences and the scalarization method, such as weighted-sum [16]. If the preference elicitation process is well defined, this subjective approach is a good choice. The posteriori optimization method first generates a set of trade-off solutions and then focuses on the decision task. Nevertheless, in case of a large number of conflicting objectives (more than five), numerous candidate solutions are returned to the user. So the main difficulty for the user lies in finding the best solution. The posteriori model may return more reliable solutions but suffers from a hard decision making task. Another serious drawback is the computation time, especially for large dimensional problems since this method searches all trade-off solutions.

Our WD procedure is subjective because it depends on the personal preferences of the buyer on the procurement. Besides, for subjective applications, it is desirable to determine only one best solution [16]. In multi-objective CRAs, it is very challenging for the buyer to evaluate a set of trade-off solutions to find the one that is the closest to his needs. By taking into account the buyer requirements, the large solution search space may be narrowed down. This will really help our WD technique to produce the solution very quickly. For all these reasons, we follow the priori approach. GAs have been

applied successfully to many multi-objective optimization problems [41]. In our research, we propose a WD approach based on GAs that we integrate with diversity and elitism mechanisms in order to improve the quality of the solutions. These two mechanisms are essential for the effectiveness of the GAs [20, 33]. We define our own variants of the diversity and elitism strategies; other variants can be found in the papers [20, 33]. We also consider the buyer's payoff at each auction round, and based on the achieved gain, our WD method will be able to increase the solution optimality.

Moreover, we conduct a real case study as well as simulated testing to show that our WD method is more prominent than without the diversity and elitism schemes. For the simulation part, we generate large instances of our CRA problem. The experimental results demonstrate on one hand the performance of our WD method in terms of several quality measures, like solution quality, trade-off between convergence and diversity, and run-time complexity, and on the other hand, the superiority of our WD to well-known heuristic and exact WD algorithms that have been implemented for much simpler CRAs.

6.3 State-Of-The-Art Combinatorial Reverse Auctions

Traditional CRAs (multiple items, single unit and single attribute) have been effectively adopted in various practical applications, including government contract auctions, corporate procurement systems, bus and truckload transportation routes, airport runway arrival and departure slots, wireless communication services, and nation-wide food supplying contracts [7, 89]. As mentioned in the article [24], these types of auctions have been also successful in fulfilling the requirements of fast trading and large, price sensitive services with economic efficiency. For instance, CRAs have been used in Chile to provide million of low-income school children with meals [24]. The government

utilized CRAs by establishing contracts with food supplying companies. In fact, CRAs were successful not only to improve the meal quality and coverage but saved a significant amount of money, about \$40 million.

Lately, more advanced CRAs have been introduced in the literature. As an example, in the study [62], CRA was defined with multiple attributes of items but with single unit. The authors restricted each supplier to a certain number of items to win. They established a bi-objective programming model that minimizes the price and maximizes the scores of non-price attributes. They extended the evolutionary technique Ant Colony (AC) to solve their particular CRA by using two strategies: the max-min pheromone to prevent the premature convergence and improve the optimization capability, and the dynamic transition strategy to enhance the searching process of the global optimization. They compared their Improved AC (called IAC) with the standard AC and also with Enumeration Algorithm with Backtracking (EAB), and showed that it outperforms both algorithms in terms of computation time. In the article [13], the WD problem of CRAs is carrying out with multiple units. The authors tackled the WD in the context of combinatorial transportation procurement auctions. They took two objectives into account: minimizing the price and maximizing the service-quality. To solve this combinatorial problem, they proposed a GA-based algorithm. They constructed eight variants of the solution method, and performed several tests to compare the time-efficiency of these evolutionary methods. They also compared these GA-based methods with Branch and Bound-based algorithm, and showed that evolutionary methods perform better, especially for large problem instances. In another article [12], the authors addressed CRAs with multiple units and two objectives in the WD procedure:

minimizing the transport costs and maximizing the transport quality. They developed a heuristic Pareto Neighbourhood Search method, inspired by the ideas of two meta-heuristics: Greedy Randomized Adaptive Search Procedure and Adaptive Large Neighbourhood Search. They conducted several tests to evaluate the performance in time of their WD method, and showed that it outperforms some other heuristics in terms of the approximation set quality.

More recently, in the article [44] the authors tackled CRAs with multiple units of items but with single attribute. They considered the specific context of iterative combinatorial auctions with feedback through Lagrange multipliers. The main contributions of this paper are to examine the Lagrangian relaxation for the WD problem and to introduce a novel technique to solve this problem. This technique may provide a good quality of solution in much less time when compared to CPLEX 12, a popular optimization studio package. This paper also contributed to the combinatorial auction test suite by generating data in the context of multi-item iterative combinatorial auctions. Lastly, it showed that the proposed method saves significant amount of time than with the traditional Lagrangian relaxation methods. The article [72] considered multiple units of items and two quantitative attributes, price and delivery rate. Regarding the constraints, the buyer specifies the upper bounds of both price and delivery rate, and on the other hand, each seller elicits the lower bounds. This paper took into account the all-unit discount scheme for both attributes, and several situations of the sellers' stock as well. It addressed the WD problem as multi-sourcing, and which is solved with a GA-based method, called GAMICRA. The latter generates optimal solutions via a single objective fitness function that minimizes all the quantitative attributes. With respect to

the results of numerous experiments, the authors showed that their WD technique is efficient and consistent. Later in the paper [73], they demonstrated that GAMICRA is superior to several other methods, such as IAC, EAB, and Branch & Bound, in terms of processing time. Furthermore, they developed an exact algorithm to solve the same WD problem. They calculated the accuracy of both GAMICRA and the exact algorithm, and proved that GAMICRA produces optimal solutions. In the paper [77], it has been shown that evolutionary algorithms are suitable and perform well in the context of auctions. The authors also depict the proposed evolutionary algorithm achieved a market equilibrium. In another paper [87], the authors pointed out that the combinatorial auctions achieved resource allocation that ultimately gained the Nash equilibrium. In this paper, they proposed a GA-based optimal resource allocation solution.

6.4 Design of Multi-Objective CRAs

Our goal is to elaborate a robust and efficient evolutionary WD model for our advanced combinatorial procurement auction. The target is to return one best solution in a very low computational time by respecting all the elicited trading constraints.

6.4.1 Auction Features

We design our CRA with the following useful characteristics:

A. Requirements Revelation

Buyer requirements are fully revealed to sellers prior to bidding to maximize his expected payoff. It has been shown that truthful revelation achieves the optimal payoff [86].

B. Sealed Bidding

The bidding process is sealed to benefit sellers of the bid privacy. The auction discloses only the final score (the fitness value) of the winning solution and the underlying winning bidders.

C. Partial/Full Bidding

Each seller bids on any bundle of items, and each bid should satisfy his constraints and buyer constraints as well. The system will check for the bid validity. The seller competes on an item if he has sufficient amount of that item.

D. First-Score Protocol

The highest solution score wins, which represents the combination of sellers that best optimizes all the objectives.

E. Multi-Round Bidding

The auction has multiple rounds to give sellers a chance to improve their bids by providing better configurations. Multi-rounds take place upon the request of the buyer to go for another round in case he is not satisfied yet with the gain attained at this stage.

6.4.2 Trading Requirements

To launch an auction, the buyer first chooses the items he would like to purchase, and for each item the attributes he is interested in and the quantity (number of units). He also provides a ranking of the chosen attributes according to their order of importance.

Additionally, he divides all the attributes into two categories: the first one comprises the attributes to be maximized (A_{max}), and the second one the attributes to be minimized (A_{min}). Next, the buyer specifies his constraints i.e., the maximum and minimum value for each attribute of each item. All these requirements (selection of items, units and attributes as well as ranking, objectives and constraints of attributes) are treated as hard constraints since the winning solution should satisfy them all. With respect to buyer requirements, each registered seller reveals his constraints on the attributes of the items he would like to bid on.

6.4.3 Valid Bids

A seller competes only on the items he is interested in. We represent a bid, a bundle of items, as shown below. Any bid value submitted by seller s for attribute a of item i (V_{sai}) must satisfy the constraints of his own and the buyer's (see eq. 6.1).

$$\begin{aligned}
 & bid_s = \{ \{V_{s11}, V_{s21}, \dots\}, \{V_{s12}, V_{s22}, \dots\}, \dots, \{V_{s1I}, V_{s2I}, \dots\} \} \\
 & \text{such that } \begin{aligned}
 & \max V_{ai} \geq V_{sai} \geq \min V_{sai} \quad \text{when } ai \in A_{min} \\
 & \max V_{sai} \geq V_{sai} \geq \min V_{ai} \quad \text{when } ai \in A_{max} \\
 & i \in [1, I] \quad \text{and } s \in [1, S]
 \end{aligned}
 \end{aligned} \tag{6.1}$$

where $\min V_{ai}$ and $\max V_{ai}$ are respectively the minimum and maximum values placed by the buyer for an attribute a of item i ; $\min V_{sai}$ and $\max V_{sai}$ are respectively the minimum and maximum values given by the supplier. Our system checks bid validity.

6.4.4 Winning Solution

The winning solution denotes which sellers have been selected to provide which units of which items according to the submitted bid values and trading constraints. We may note that the units of the same item may be provided from different sellers.

In the situation where there is more than one solution with the same highest score, we suggest to incorporate one or more non-negotiable attributes (e.g. feedback ratings, trust and cooperation of sellers) to break the tie. A non-negotiable attribute represents the reputation of the seller in the e-market, and we assume a third party provides its value. For each obtained solution, we sum the points achieved by the winning sellers for the non-negotiable attributes. The solution with the highest tally wins.

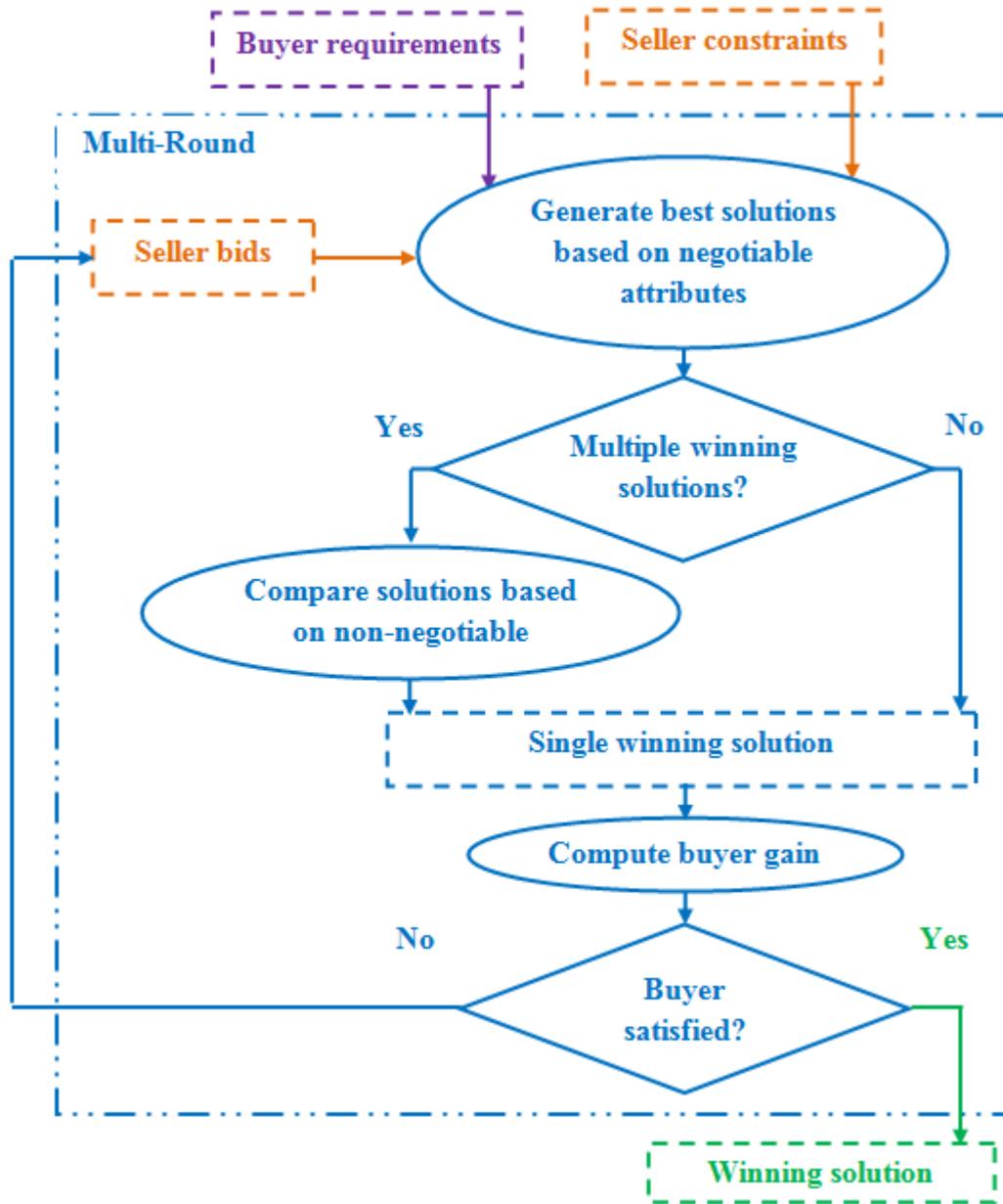


Figure 6.1: Multi-Objective CRAs

6.4.5 Buyer's Gain

At the end of each auction round, our system calculates the buyer's revenue to help him decide whether to go to the next round or not. We propose the following algorithm

to compute the gain by considering the bids and constraints of the winning bidders of the current round and constraints of the buyer as well.

$$\begin{aligned}
 gainBuyer_{total} &= \frac{\sum_{i=1}^I \sum_{ui=1}^{U_i} gainBuyer(ui)}{\sum U_i} \times 100\% \\
 \text{if } ai &\in A_{min} \\
 \text{then } gainBuyer(ui) &= \frac{\sum_{ai=1}^{ai} (\frac{maxV_{ai} - V_{sai}}{maxV_{ai} - minV_{sai}})}{\sum ai} \\
 \text{else } gainBuyer(ui) &= \frac{\sum_{ai=1}^{ai} (\frac{V_{sai} - minV_{ai}}{maxV_{sai} - minV_{ai}})}{\sum ai}
 \end{aligned} \tag{6.2}$$

6.5 Winner Determination Approach

This section first formalizes the optimization problem of our particular CRA. It then describes how the solutions in our CRAs are represented and generated through GAs operations and diversity and elitism techniques. Furthermore, this section introduces three WD methods that differ in their ways in producing the candidate solutions.

6.5.1 Multi-Objective Optimization Problem

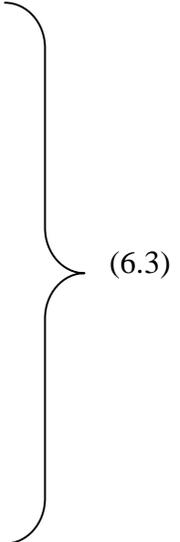
Generally speaking, a MOO problem is defined with a set of objective functions, a set of decision variables and a set of constraints. In our CRA, the objective function corresponds to the maximization or minimization of a given attribute. A decision variable denotes the solution i.e. the set of winning sellers. Since we follow the priori approach, our combinatorial problem is solved with a single objective function, which is the fitness function as defined in the subsequent section (see eq. 6.4). Assume x is a candidate solution, the target of our optimization is to maximize the fitness value of x .

However, we have a mixture of maximization and minimization objectives. So to make it all maximization, we propose two versions of derivation of the fitness function (see eq. 6.4). Consequently, it does not matter anymore what the objective of an attribute is since our WD approach converts it into a maximization objective as shown in eq. 6.4. In our CRA, the solution x should consist of eligible sellers only. When generating a potential solution, a seller is assigned randomly to a certain unit of an item. But there is a chance that this seller has been wrongly chosen since he did not submit any bid for that item. Our WD method always ensures the feasibility of the solution by checking the eligibility of each seller for each unit as shown in eq. 6.3.

```

maximizeFitness( $x$ )
such that isFeasible( $x$ ) = true
isFeasible( $x$ ):
     $ui = 1$  // first unit of item  $i$ 
    for all  $ui \in U$  do
        while doNotExist( $bid_{si}$ ) //  $bid_{si}$  is the bid of seller  $s$  for item  $i$ 
            do reSelect( $s, ui$ )
    return true

```



(6.3)

6.5.2 Solution Representation and Fitness Assignment

We encode a solution or more precisely a chromosome based on the number of items and units. To represent a seller, a random number is generated from 1 to S (number of

sellers), and its binary representation becomes one part of the chromosome. We employ the equation of $S < 2^{reqBits}$ to determine the required number of bits (*reqBits*) for each seller. Hence, a chromosome consists of binary bits, and for each *reqBits* is associated a seller who was selected to provide that unit of an item. Our WD algorithm first generates the initial chromosomes randomly, and then performs the fitness assignment of each chromosome as explained below. Table 6.1 exposes the fitness assignment functions.

Table 6.1: Fitness assignment functions

Function	Description
<i>weight(ai)</i>	Weight of attribute <i>a</i> of item <i>i</i> , which emphasizes its relative importance among other attributes based on the ranking given by the buyer.
<i>utility(sai)</i>	Utility value of attribute <i>a</i> of item <i>i</i> for seller <i>s</i> .
<i>totalUtility(si)</i>	Utility value of item <i>i</i> for seller <i>s</i> .
<i>unit(si)</i>	The number of units of item <i>i</i> to be provided by seller <i>s</i> .
<i>fitness(x)</i>	The fitness value of chromosome <i>x</i> represents its quality. It is a measurement to determine whether <i>x</i> will survive or not.
<i>totalFitness(p)</i>	Total fitness value of population of chromosomes, <i>p</i> .

We produce the fitness value of a chromosome with the following equations by taking into account both buyer and seller requirements (i.e. attribute ranking, objectives and constraints):

$$\begin{aligned}
fitness(x) &= \sum_{s=1}^S \sum_{i=1}^I totalUtility(si) \times unit(si) \\
totalUtility(si) &= \sum_{ai=1}^{al} weight(ai) \times utility(sai) \\
weight(ai) &= \frac{al-rank_{ai}+1}{\sum_{ai=1}^{al} (al-rank_{ai}+1)} \\
\text{such that } \sum_{ai=1}^{al} weight(ai) &= 1 \\
\text{if } ai \in A_{max} \text{ then } utility(sai) &= \frac{V_{sai}-minV_{ai}}{maxV_{sai}-minV_{sai}} \\
\text{else } utility(sai) &= \frac{maxV_{ai}-V_{sai}}{maxV_{sai}-minV_{sai}}
\end{aligned} \tag{6.4}$$

In the fitness assignment, we adopt the weighted sum method. We remodel the fitness assignment procedure based on the requirements of our particular problem. Since we are following the priori approach, deducing weights from the requirements is crucial. We apply the Rank Sum Weights method [30] to obtain the relative weight of each attribute w. r. t. buyer ranking. We also calculate the utility vectors according to the buyer and seller objectives and constraints. Thus, the returned fitness value reflects the optimization of the solution based on all attributes, hence on all objectives.

6.5.3 Solution Production

The chromosomes are re-produced by following the Darwinian statement of “survival of the fittest”. Our WD approach employs three genetic operators: Gambling Wheel Disk method [29], Modified Two-Point Crossover that we introduced in [69], and Swap Mutation [23] (see Table 6.2). Gambling Wheel Disk and Swap Mutation are both popular methods and therefore suitable for our CRA problem.

In order to avoid premature convergence to local optimum, and to reach faster to the global optimum, we incorporate the diversity and elitism mechanisms to solve these two issues respectively. We propose our own variants as shown in Tables 6.3 and 6.4. We may note that we maintain the same number of elite chromosomes as the participant chromosomes ($\psi = \zeta$). Equation (6.5) shows the formula of scope calculation of Gambling-Wheel Disk Selection. Equation (6.6) shows the formula of relative fitness value calculation.

$$scope(x) = \left[\frac{\sum_{x=1}^{\zeta-1} fitness(x)}{totalFitness(p)}, \frac{\sum_{x=1}^{\zeta} fitness(x)}{totalFitness(p)} \right] \in [0,1] \quad (6.5)$$

$$relFitness(x) = \frac{|fitness_{\zeta+1} - fitness_{\zeta-1}|}{maxFitness(p) - minFitness(p)} \in [0,1] \quad (6.6)$$

Table 6.2: Description of genetic algorithm operations

Operation	Description
Gambling-Wheel Disk Selection	We reproduce the chromosomes with the best fitness values based on their scopes. The chromosome with the largest scope has the highest fitness value, and therefore a better chance to be chosen.
Modified Two-Point Crossover	Modified-Two Point Crossover differs from the basic two-point crossover in the direction of taking portions from the parents. The first child is created in the same way as in the basic version, but the second child takes the portions in the reverse order. This modified version is of great significance. Even if the parents are the same, it is able to produce new child where the basic version just clones the parents.

Swap Mutation	In this mutation, sellers exchange positions (units or items) among themselves. Two positions in a chromosome are first selected randomly, and then sellers are swapped accordingly. Thus, this operation creates new chromosomes.
Parameter	Description
δ	Number of generations
ζ	Number of chromosomes
α	Crossover Rate
β	Mutation Rate

Table 6.3: Description of diversity mechanism

Method	Crowding Distance
Description	We adopt the crowding distance method and update it to our necessity. We propose the relative fitness function to calculate the crowding distance. The chromosomes are ranked w. r. t their relative fitness values, and the top ζ chromosomes are then selected.
Target	To prevent the population from having many similar solutions by not allowing premature convergence to local optima.
Function	Description
$relFitness(x)$	Relative fitness value of chromosome x , a measurement to

	determine how diverse the chromosome is from others.
$maxFitness(p)$	Maximum fitness value among chromosomes of population p .
$minFitness(p)$	Minimum fitness value among chromosomes of population p .

Table 6.4: Description of elitism mechanism

Method	External Population
Description	Elite solutions are the best solutions found so far. We store these solutions in an external population. After each generation, we inject $\psi \times \omega$ elite chromosomes in the participant chromosomes for the next generation.
Target	To avoid losing good solutions and help converging to the global optima.
Parameter	Description
ψ	Number of elite chromosomes.
ω	Elite chromosome rate.

6.5.4 Winner Determination Algorithm

Algorithm 6.1: Generic Winner Determination
<pre> { do { 1: sellers submit bids at each round 2: generation ← 1 3: initialize a population of chromosomes p such that each solution of p is feasible </pre>

```

4: assign fitness to  $p$ 

5: do
    { 5.1: produce  $p'$  from  $p$  with GA operators
      5.2: assign fitness to  $p'$ 
      5.3: produce  $p$  (participant population for the next generation) from  $p$  and
            $p'$  such that each solution of  $p$  is feasible
      5.4: generation  $\leftarrow$  generation + 1
    }

    while (maximum generation is not reached)

6: return round winning solution

}

while (buyer is not satisfied)

return final winning solution

}

```

Figure 6.2: Generic Winner Determination

The steps of the generic WD algorithm for our particular CRA are given in Algorithm 6.1. For each candidate solution x of population p , we check its feasibility. We propose three versions of the WD method as explained below. The step 5.3: “*produce p (participant population for the next generation) from p and p'* ” is defined differently with respect to the three versions.

6.5.4.1 Winner Determination

The first one, called WD, employs only the three genetic operators. For the next generation, WD selects ζ chromosomes among p and p' with the best fitness values.

6.5.4.2 Winner Determination with Diversity

The second one called WDD utilizes the diversity method based on the crowding distance (see eq. 6.6). WDD takes special care of maintaining good diversity among the chromosomes. Instead of populating new chromosomes for the next generation based on their fitness values, rather it uses relative fitness values as shown in Algorithm 6.2. We may note that the side effect of applying crown distance is that we may sacrifice chromosomes with high fitness values. But for the sake of maintaining diversity and gaining even better chromosomes in the long run, this technique is very effective. Therefore, this version does not suffer from early convergence.

Algorithm 6.2: WDD

```
{ 5.3.1: assign relative fitness values to  $p$  and  $p'$  according to eq. 6.6  
 5.3.2:  $p = p \cup p'$   
 5.3.3: sort  $p$  with respect to relative fitness values  
 5.3.4: update  $p$  by keeping only the best  $\zeta$  chromosomes (participant population for the  
        next generation)  
 5.3.5: return  $p$   
}
```

Figure 6.3: WDD

6.5.4.3 Winner Determination with Diversity and Elitism

Algorithm 6.3: WDDE

```
{  
  5.3.1°:  $e = p \cup e$   
  5.3.2°: assign fitness values to  $e$   
  5.3.3°: sort  $e$  with respect to fitness values  
  5.3.4°: update  $e$  by keeping only the best ( $\zeta = \psi$ ) chromosomes (new elite  
         chromosomes)  
  5.3.5°: re-produce  $p$  (participant population for the next generation) by taking  
          $\zeta - (\psi \times \omega)$  chromosomes of  $p$  returned by 5.3.5, and  $\psi \times \omega$  elite  
         chromosomes of  $e$   
  5.3.6°: return  $p$   
}
```

Figure 6.4: WDDE

The third one called WDDE uses both diversity and elitism schemes (see Algorithm 6.3). In each generation, we select the best (elite) chromosomes among the initial and final populations of chromosomes for the next generation. We inject $\psi \times \omega$ elite chromosomes into the participating chromosomes in the population of the next generation. By adopting this technique, we ensure the preservation of the best chromosomes throughout the generations.

6.5.4.4 Optimization Process

Figure 6.5 illustrates the entire optimization procedure where p_t is the population at the current generation t , p_{t+1} for the next generation $t + 1$, e_t elite chromosomes at the current generation t , e_{t-1} elite chromosomes from the previous generation $t - 1$.

6.6 A Case Study

The three proposed WD methods have been implemented in Java and executed on an Intel (R) core (TM) i3-2330M CPU with 2.20 GHz processor speed and 4 GB of RAM.

6.6.1 Requirements Elicitation and Bid Submission

For our combinatorial market, we consider here four electronic items, and the buyer requirements for these items are exposed in Table 6.5. We have eight optimisation criteria. The highest attribute ranking has a value of 1. As an example, we give the constraints of the first seller. We have in total 20 sellers, and in Table 6.6, we sketch examples of submitted valid bids for the first round. As we allow both full and partial participation, a seller may place bids for all the items or any subsets of items. For instance, Seller1 does not enter any bid for Laptop, same for Seller19 for Scanner and Printer.

Table 6.5: Buyer and seller requirements

Item	Attribute	Rank	Objective	Constraint	Seller1 Constraint
Television (2 units)	Price	1	Min	≤ 2000	≥ 1500
	Screen Size	2	Max	≥ 30	≤ 60
Scanner (1 unit)	Price	1	Min	≤ 300	≥ 120
Laptop (3 units)	Price	1	Min	≤ 3000	≥ 2000
	Hard Drive	2	Max	≥ 256	≤ 4096
	System Memory	3	Max	≥ 2	≤ 64
Printer (2 units)	Price	1	Min	≤ 300	≥ 150
	Printing Speed	2	Max	≥ 10	≤ 100

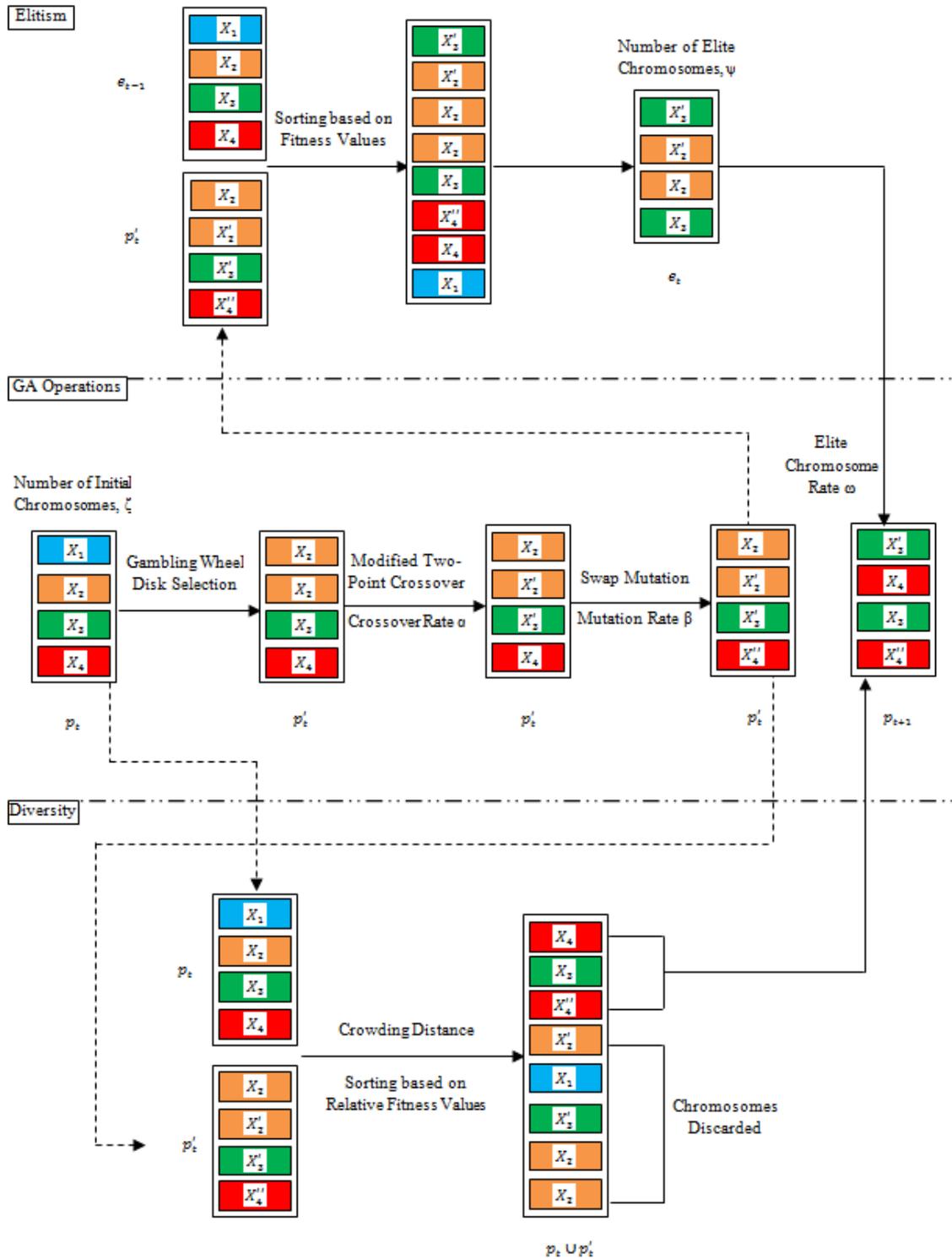


Figure 6.5: Schematic of winner determination approach

Table 6.6: Valid offers

Seller	Submitted Bid
S1	{{1800,40},{250},{},{280,20}}
S2	{{1900,42},{280},{2700,1024,4},{250,20}}
...	...
S19	{{1600,40},{},{2600,612,8},{}}
S20	{{1600,42},{250},{2500,612,16},{280,30}}

6.6.2 First Generation

Figure 6.6 shows how a chromosome is defined in line with our example. Since we have 20 sellers, therefore we need 5 bits to encode each unit. The first ten bits are for Television, the next five bits are for Scanner, and so on. For instance, seller S18 has been chosen at this stage to supply one unit of Television and one unit of Printer. Figures 6.7 and 6.8 illustrate the working process of the crossover and mutation operations.

Television		Scanner	Laptop			Printer	
Unit1	Unit2	Unit1	Unit1	Unit2	Unit3	Unit1	Unit2
10010	00110	00010	00011	10100	10001	00101	10010
S18	S6	S2	S3	S20	S17	S5	S18

Figure 6.6: Solution representation

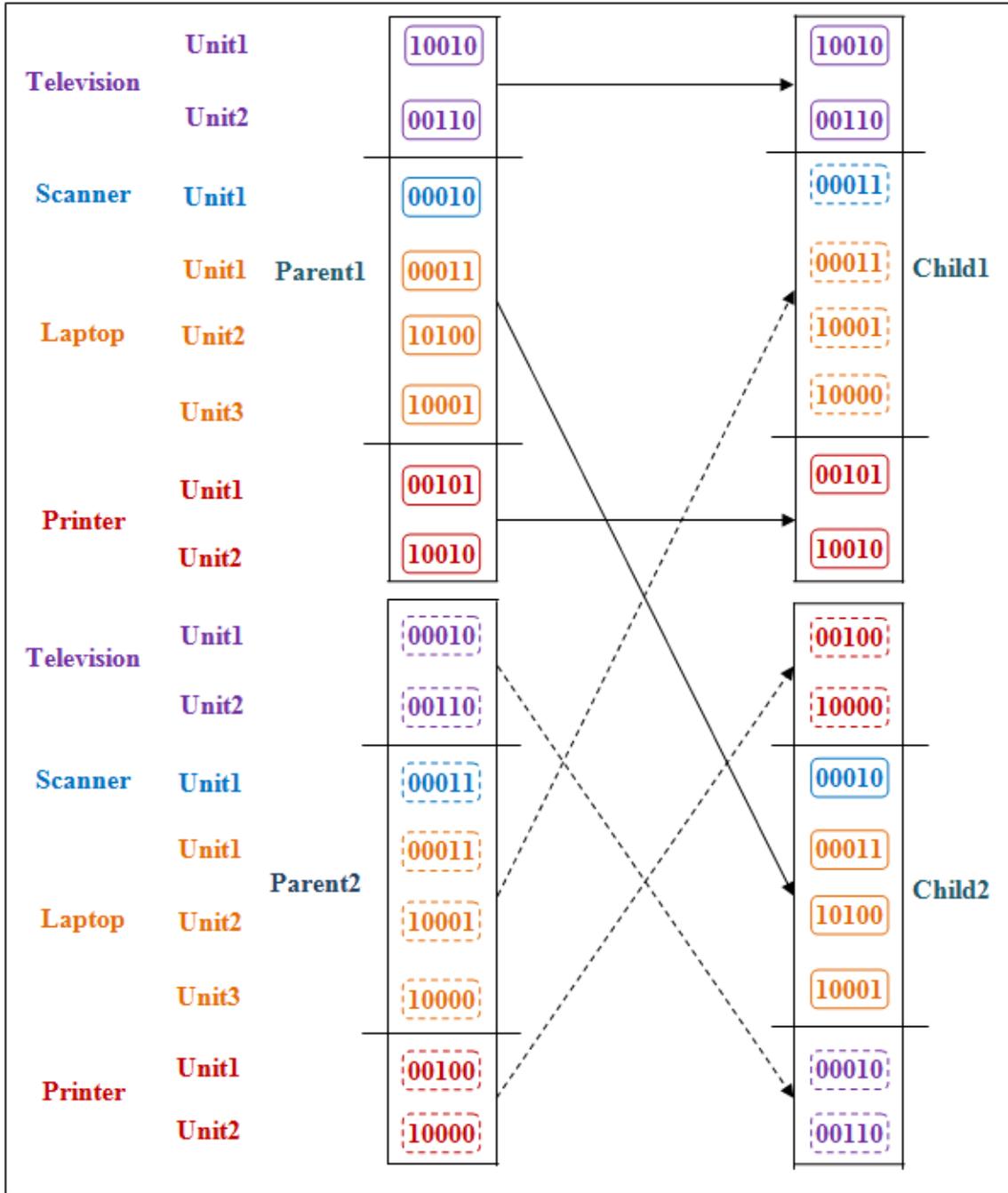


Figure 6.7: Modified two-point crossover

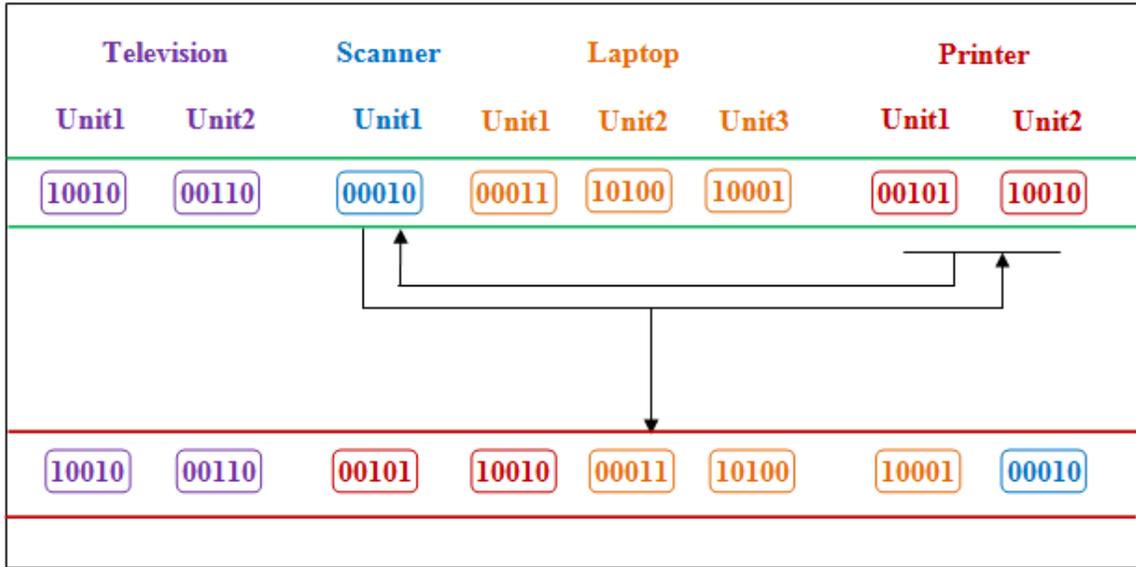


Figure 6.8: Swap mutation

In Table 6.7, in the first generation, the three WD algorithms produced 100 initial chromosomes (ranked w. r. t fitness values), and then re-produced new ones by applying selection, crossover and mutation operators. Table 6.8 presents the participating chromosomes for each WD algorithm for the next generation. To select those chromosomes, WD considers their fitness values, WDD their relative fitness values, and WDDE takes elite chromosomes into account along with their relative fitness values.

Table 6.7: Initial solutions

Initial Chromosome		Fitness Value
X1	0111000110010011000100011010000100100100	3.788
...
X100	0111000100010010111101110000110110001110	3.56
Chromosome after Selection, Crossover and Mutation		Fitness Value

Xg1	0010001011011100100000101010001000001001	3.851
...
Xg100	0001000111011000011000011011100101100111	3.686

Table 6.8: Potential solutions of WD, WDD and WDDE

Participant Chromosome			Fitness Value	Relative Fitness Value	Elite Chromosome
WD	Xp1	010010110100011100010001101 0000101110001	4.203	No	No
	
	Xp100	001100101000100010110011000 1010111000011	3.952		
WDD	Xp1	0101000010010100001010100100 0110100100101	4.386	1.0	No
	
	Xp100	010010110100011100010011100 0110101110001	4.084	0.084	
WDDE	Xp1	010100010101010000100100100 0110100110000	4.444	1.0	Yes
	
	Xp100	010100001010000100100101001 0100010100101	4.105	0.769	

In Table 6.8, we show that WDDE performs best among the three WD algorithms based on the fitness values.

6.6.3 Last Generation

The algorithms will be repeated 100 times (here $\delta = 100$). Table 6.9 exposes the final chromosomes returned by each algorithm.

Table 6.9: Winning solutions of WD, WDD and WDDE

Final Chromosome			Fitness Value	Relative Fitness Value	Elite Chromosome
WD	Xf1	01010000100101000101010010001 10100100101	4.386	No	No
		
	Xf100	01011001011000100011010000010 00111010011	4.003		
WDD	Xf1	01010001010010101001000110100 10001001010	4.726	1.0	No
	
	Xf100	01010000101000010010010100101 00010100101	4.105	0.269	
WDDE	Xf1	01010000100101010000100100101 00101000101	4.875	1.0	Yes
	
	Xf100	01011100011000001001000110100 10110100011	4.3	0.342	

At the end of the first round, we reveal the winning solution for each method. Table 6.10 depicts the breakdown of these winning solutions. Table 6.11 shows the winning bid values along with what the buyer expected and what the winning seller could have offer. This may be helpful for the buyer to decide to go to the next round or not. This case study demonstrates that WDDE produces the best solution quality i.e. the best combination of sellers and also the best buyer’s gain. We assume here that the buyer is satisfied with the current gain, and there is no need to go to the next round.

Table 6.10: Winning solutions at first round

Item	Unit	WD	WDD	WDDE
Television	Unit1	S10	S10	S10
	Unit2	S2	S5	S2
Scanner	Unit1	S10	S5	S10
Laptop	Unit1	S5	S9	S16
	Unit2	S9	S3	S18
	Unit3	S3	S9	S10
Printer	Unit1	S9	S2	S10
	Unit2	S5	S10	S5
Fitness Value		4.386	4.726	4.875
Buyer’s Gain		64.7%	77.3%	81.1%

Table 6.11: Breakdown of winning solution of WDDE

Item	Unit	Attribute	Buyer Value	Seller Value	Bid Value	
Television	Unit1	Price	2000	1400	1500	
		Screen Size	30	70	65	
	Unit2	Price	2000	1100	1200	
		Screen Size	30	42	40	
Scanner	Unit1	Price	300	240	250	
Laptop	Unit1	Price	3000	2200	2400	
		Hard Drive	256	1200	1024	
		System Memory	2	16	12	
	Unit2	Price	3000	2400	2500	
		Hard Drive	256	560	512	
		System Memory	2	8	6	
	Unit3	Price	3000	1400	1500	
		Hard Drive	256	560	512	
		System Memory	2	8	6	
	Printer	Unit1	Price	300	220	240
			Printing Speed	10	100	75

	Unit2	Price	300	100	110
		Printing Speed	10	210	200

6.7 Performance Evaluation

For the simulation, we randomly generate several instances of our WD problem. We consider 500 sellers, 10 items with a total of 20 units and 25 objectives. In Table 6.12, we give the quantity and number of attributes of each item. Note that since we are considering all the CRA parameters and constraints, it is obvious that if our algorithm performs well on simulated data, it undoubtedly will perform similarly well or better for real data. On the other hand, since GAs need very few knowledge about a specific problem, it is actually not a big deal whether the data is real or artificial.

Table 6.12: Item description

Item	Number of Unit	Number of Attribute
Item1	2	2
Item2	2	3
Item3	3	1
Item4	1	4
Item5	1	3
Item6	3	2
Item7	2	2
Item8	2	3
Item9	3	1
Item10	1	4

The ranking and objectives of the attributes are also determined randomly. The maximum value of each attribute is randomly produced from [100, 10000], and the minimum value from [10% of maximum value, 50% of maximum value]. Moreover, we perform numerous parameter tuning tests, and based on the results, we use the following best configuration, unless otherwise stated, $\zeta = 500$, $\delta = 100$, $\alpha = 0.6$, $\beta = 0.01$, $\psi = \zeta$ and $\omega = 0.2$. All the results returned by WD, WDD, and WDDE represent the average value of 20 runs.

6.7.1 Trade-Off between Convergence and Diversity

The goal behind this experiment is to understand the effect of diversity and storing the elite solutions (see Fig. 6.9). We initially generated 500 unique chromosomes for all the three methods, WD, WDD and WDDE. Because of the lack of diversity, WD suffers from early convergence. On the other hand, WDD and WDDE are able to continue searching for a better solution by compromising between convergence and diversity.

6.7.2 Solution Quality Assessment

Here we compare the solution quality in terms of the fitness value obtained by the three WD techniques as depicted in Fig. 6.10. Since WD suffers from premature convergence and lack of diversity, it produces comparatively poor results. Boosting with diversity, WDD provides better results. But since it does not store the elite solutions, it struggles with the issue of compromising between convergence and diversity. We can see that WDDE returns the best results thanks to the power of diversity and storage of elite solutions. WDDE is able to keep a good balance among the best solutions produced

in each generation. Hence, we drop WD and WDD, and continue only with WDDE in the next experiments.

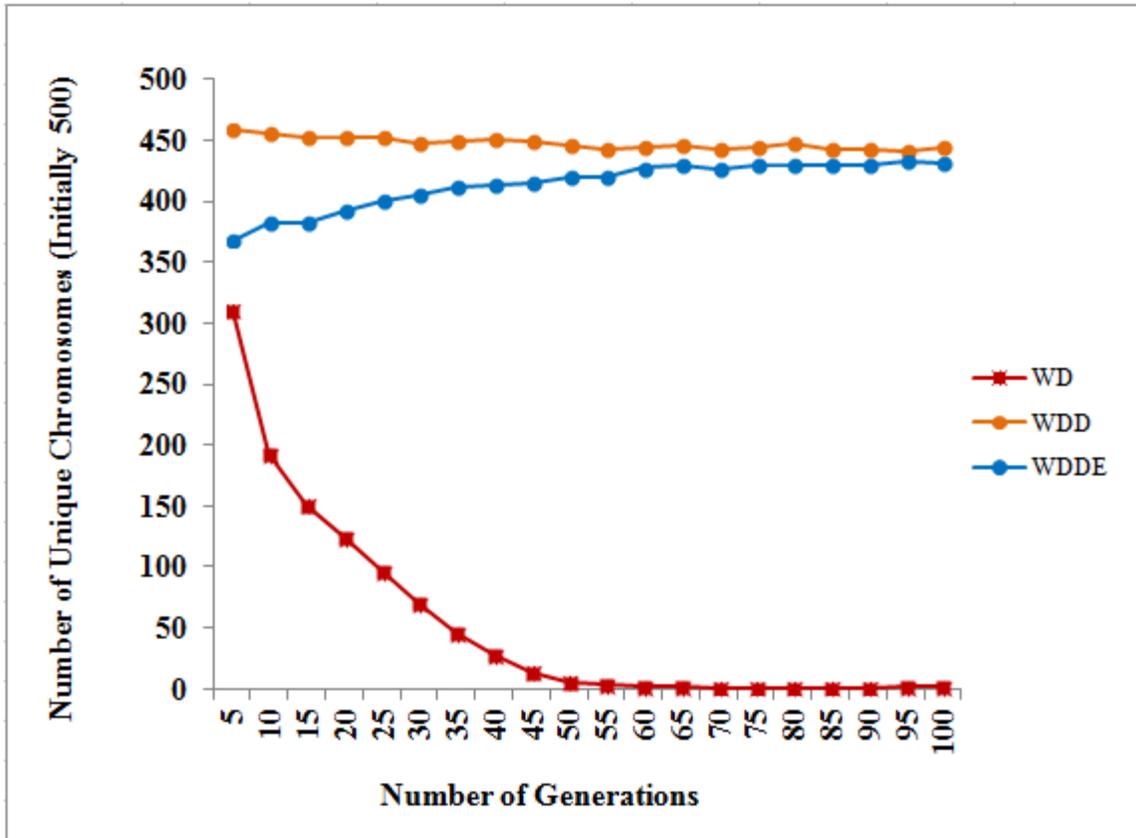


Figure 6.9: Convergence and diversity trade-off

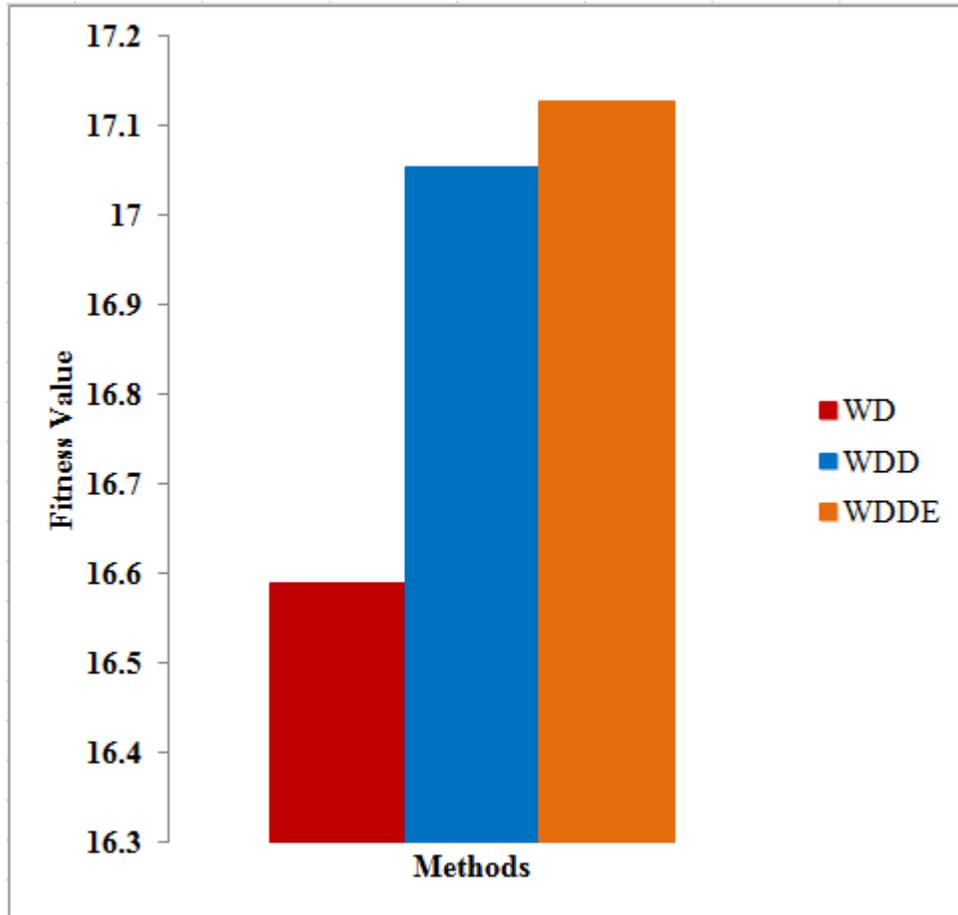


Figure 6.10: Solution quality comparison

6.7.3 Computational Expense of WDDE

Table 6.13: Configuration of items and sellers

Attribute	Number of Attributes (Number of Items)	5(2), 10(4), 15(6), 20(8), 25(10)
Unit	Number of Units (Number of Items)	4(2), 8(4), 12(6), 16(8), 20(10)
Seller	Number of Sellers	100, 200, 300, 400, 500, 600, 700, 800, 900, 1000

In this experiment, we use the configuration given in Table 6.13. We evaluate the time-efficiency of WDDE by varying the number of attributes and total number of units (see Fig. 6.11) and number of sellers (see Fig. 6.12). From the results returned by WDDE, we may say that the required computational expense is not exponential but rather polynomial. It is clear that the execution time increases linearly with the increase of attributes, units of items and sellers.

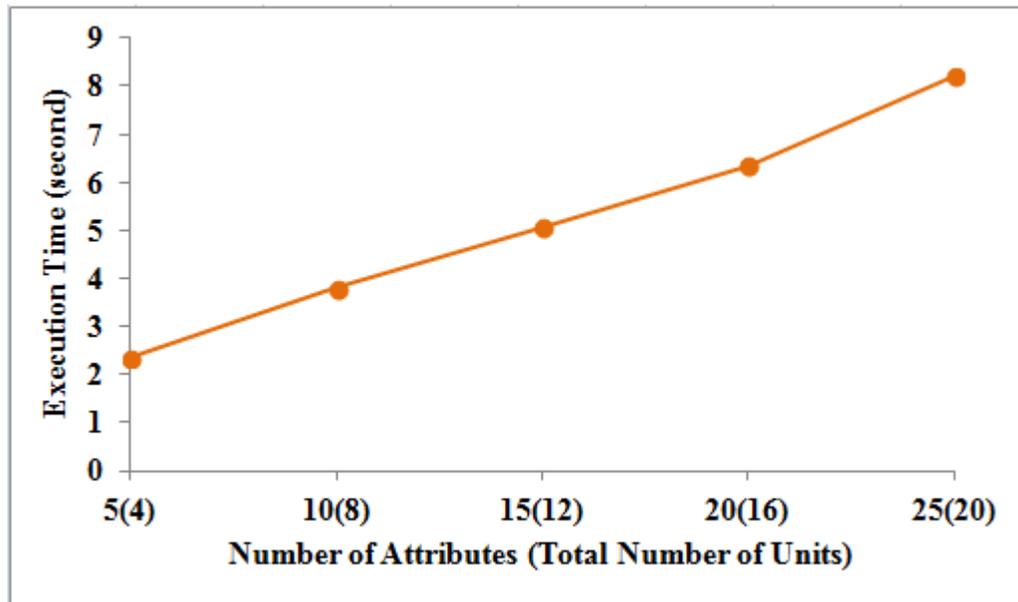


Figure 6.11: Computational expense of WDDE by varying number of attributes and total number of units

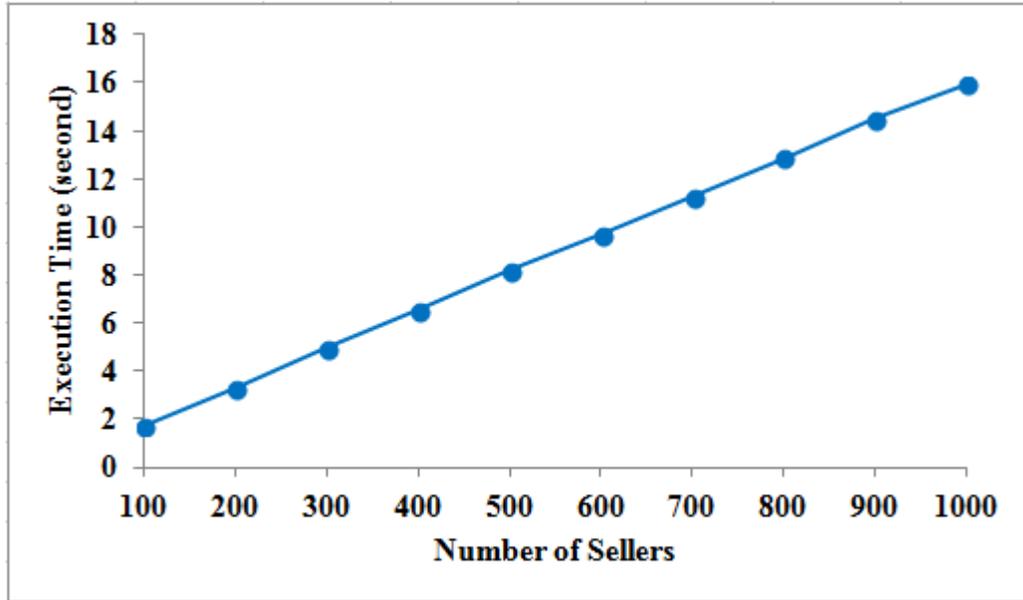


Figure 6.12: Computational expense of WDDE by varying number of sellers

6.7.4 Statistical Analysis of WDDE

We also examine statistically WDDE. Figure 6.13 presents the average fitness value with the maximum and minimum values of generations. It also depicts the error bars with a confidence level of 95%. It is noticeable that after a certain number of generations, the maximum fitness value remains constant. This means that the best solution found by WDDE might be the optimal solution. It is also obvious that WDDE is able to control the solution variations and their differences after a certain number of generations.

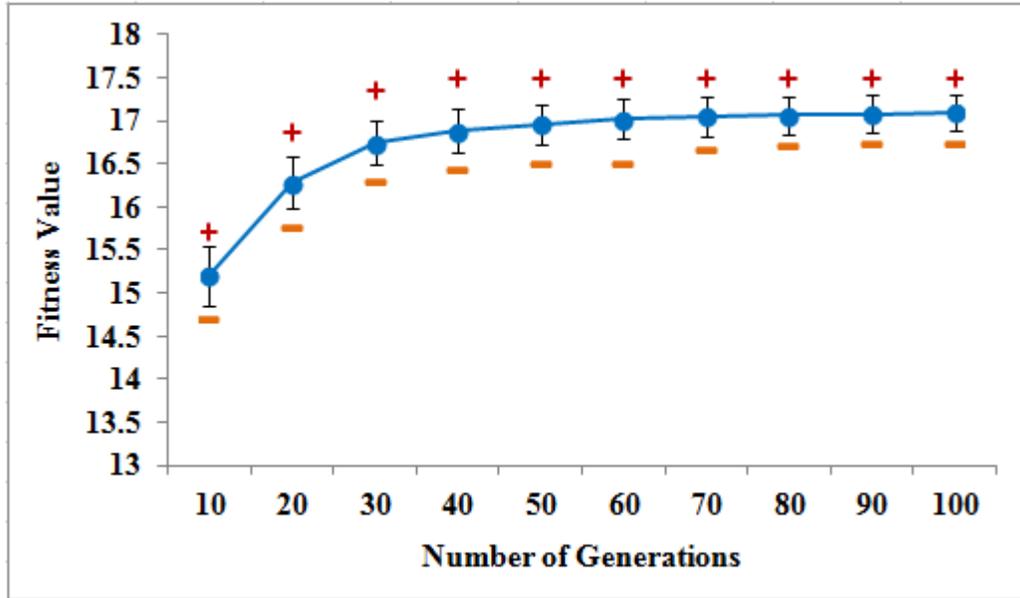


Figure 6.13: Statistical analysis of WDDE

6.7.5 Feature Comparison of WDDE with Heuristic WD Techniques

We compare the features of WDDE with three other heuristic WD methods: Improved Ant Colony (IAC), Enumeration Algorithm with Backtracking (EAB) and Genetic Algorithms for Multiple Instances of Items in Combinatorial Reverse Auctions (GAMICRA), and also with an exact WD method: Branch & Bound. Table 6.14 presents the features possessed by these methods, which all address multiple items of CRAs. Here, “-” indicates the feature is not supported. All these WD methods return one single best solution.

Table 6.14: Feature comparison

WD Algorithm	Features			
	Unit	Attribute	Objective	Constraint
IAC	Single	Multiple	Single	-
EAB	Single	Single	Single	-
GAMICRA	Multiple	Two	Single	Buyer's upper bound, sellers' lower bound
Branch & Bound	Multiple	Single	Single	-
WDDE	Multiple	Multiple	Multiple	Ranking, Objectives, Buyer's upper/lower bound, sellers' lower/upper bounds

6.7.6 Computational Expense Comparison of WDDE with Heuristic WD Techniques

We compare the computational expense of WDDE with the three other heuristic WD methods. Table 6.15 gives the processing time of IAC, EAB and GAMICRA; the first two were taken from the study [62] and the last one from the paper [73]. WDDE is significantly superior to all of them.

Table 6.15: Computational expense comparison

WD Algorithm	Number of Chromosomes	Number of Generations	Number of Sellers	Number of Items	Time (Second)
IAC	50	50	100	30	9
EAB	-	-	100	30	3
GAMICRA	50	50	100	30	0.83
WDDE	50	50	100	30	0.09

6.7.7 Computational Expense Comparison of WDDE with an Exact WD Technique

We compare the computational expense of WDDE with the exact WD method. Table 6.16 shows the results for the Branch & Bound-based method, taken from the paper [28]. Though the tests are performed on computers of different processor speeds (450 MHz for Branch & Bound, and 2.20 GHz for WDDE), it is clear that WDDE greatly outperforms Branch & Bound.

Table 6.16: Computational expense comparison of WDDE with branch & bound

WD Algorithm	Number of Sellers	Number of Items	Total Number of Units	Time (Second)
Branch & Bound	100	10	25	>100
WDDE	100	10	25	0.085

6.8 Conclusion

Determining the winners for CRAs is a challenging task because of its computational complexity. Our winner selection model is new as it addresses a new type of CRAs, multi-units, multi-attributes and multi-objectives, which is subject to various trading constraints. We have addressed the WD as an evolutionary MOO problem with mixed minimization and maximization criteria. We have solved this complex WD problem by using genetic algorithms that we have integrated with our variants of diversity and elitism strategies. Even though evolutionary algorithms are very powerful to address large-scale MOO applications, they however require advanced design mechanisms, and their implementation is difficult. Our WD approach returns a single best solution based on the known constraints of buyers and sellers. We have also proposed a strategy to maximize the buyer's revenue in CRAs. Furthermore, we have conducted a case study and simulated experiments and showed that our solving method is more prominent than without the diversity and elitism schemes. Additionally, we have verified experimentally the performance of our WD approach in terms of three quality metrics. We have showed that diversity and elitism strategies increased the solution optimality. Finally, we have demonstrated its significant superiority to other popular WD techniques implemented for simpler CRAs.

There are several promising directions of this research. In our study, it is up to the seller to negotiate on a particular item or not. But if he bids on an item, we assume he has sufficient stock of that item. In the future, even though the seller has insufficient stock of an item, he can still compete for that item. Speaking of evolutionary algorithms, we will certainly obtain better solution quality by combining our WD method with other

suitable algorithms, such as Simulated Annealing, Ant Colony and Tabu Search [57]. This is a prominent future work in the context of CRAs. Also, we would like to investigate the interactive optimization approach to incorporate the buyer's decisions and constraints during the iterations of optimization [20]. Though this is the most reliable model, it however suffers from the time-cost due to its complicated process. So, we will examine parallel genetic algorithms to greatly improve the time-efficiency as shown in different applications like in the article [2]. Moreover, we will include qualitative attributes along with quantitative ones [64], never handled before in the combinatorial context.

Chapter 7

SU-TA-TO-TR CRA

7.1 Summary

The option for public utilities to organize e-auctions to purchase the needed electricity from other power suppliers is a new concept. To this end, we develop a Combinatorial Reverse Auction (CRA) to procure power from diverse sources including residents and plants. In our CRA, subject to various trading constraints, an item denotes a time slot that has two conflicting attributes, energy volume and its price. To secure electricity, we design our auction with two bidding rounds: the first one is for variable-energy suppliers and the second one for other sources, like controllable load and other renewable technologies. Our CRA leads to a complex Winner Determination (WD) problem. We view this problem as a resource allocation optimization that we solve with multi-objective genetic algorithms in order to find the trade-off solution that best lowers the price and increases the energy. This solution consists of multiple winning suppliers, their prices, energy volumes and schedules. In this study, we validate our WD approach based on simulated data by generating large instances of our multi-objective constrained auction problem. The goal of the experiments is to assess the time-efficiency of our WD method and its significant superiority to well-known heuristic and exact WD techniques.

Moreover, we implement two exact algorithms to solve our complex procurement problem in order to evaluate the accuracy of our WD method i.e. how near-to-optimal is the solution.

7.2 An Overview

7.2.1 Problem

To meet the additional load during peak periods, public utility companies may organize online auctions to procure electricity from diverse sources simultaneously, such as variable energy (solar and wind), active controllable load (battery storage, electric vehicles and heat storage) and controllable renewable energy (hydroelectricity, biomass and geothermal heat). It has been shown that auctions in the electricity sector promote economic growth and foster competition among energy suppliers [15], [17], [35]. The deregulation of the electricity sector and privatization of power retailers are becoming common in many countries [17]. In this present study, we introduce efficient electricity Combinatorial Reverse Auction (CRA) for short-term contracts in the settings of Consumer-to-Business and Business-to-Business. With the help of e-auctions, grid companies would be able to obtain the needed energy with a good price since auctions are competitive environments. Each one, buyer and supplier, tries to maximize his own profit. It is worth mentioning that our auction is substantially different from what has been proposed and implemented in the electricity markets. Contracting electricity from diverse renewable energy sources (residents and plants) to offset the extra demand load is a new concept. So far, energy auctions have been defined for long-term (several years) or short-term (hourly) contracts [45], and can be technology specific (one or more

renewable sources) or neutral (any technology) [45]. Renewable energy has performed greatly in the market by decreasing the price of electricity and their participation is increasing rapidly, for instance 60% by 2013 in the Spanish market [15]. The entry of new power sources has shown to increase the market competition [15]. Countries and their governments mostly adopted renewable-energy auctions for long-term contracts (typically with one or two suppliers over several years) and their number has increased from 9% in 2009 to 44% by the beginning of 2013 [43]. Another very common type of electricity auctions is held yearly to contract renewable energy facilities to be built in a year ahead [17]. The target of these auctions is to attract new investors (small and large) [45]. Limited studies have been carried out for combinatorial electricity markets despite the fact that they match the demand and supply very efficiently (price-wise) and maximize buyer's revenue [62]. Previous combinatorial electricity auctions have some limitations, for example they do not consider multiple attributes, conflicting objectives and trading constraints as done in our procurement auction.

Our new electricity CRA will solve the serious issue of extra load and will greatly benefit grid companies and their end-consumers both environmentally and economically. Indeed, this new retail market will encourage the expansion of renewable energies as well as home-based technologies that are accessible to customers, such as solar panels and plug-in electric vehicles.

7.2.2 Auction Features

One of the challenges is to design our advanced electricity auction with relevant features and solving mechanisms that will lead to the best outcome in terms of solution quality and time-efficiency.

A. Reverse

The distribution company (buyer) purchases electricity from multiple green power suppliers (residents and plants). This energy is needed for a certain time period. [87] stated that in the future smart electricity markets, different scale suppliers will participate and small players will have an active role.

B. Combinatorial

Multiple items and each one represent a time slot of fifteen minutes within the demand period. In this way, small (residents) suppliers will be able to satisfy the energy demand of 15 minutes.

C. Two Conflicting Negotiable Attributes (Volume and price of energy)

These attributes are in conflict because the buyer objective is to lower the price and at the same time increase the energy volume.

D. Trading Constraints

The buyer needs to decide on some trading requirements, such as the energy delivery schedule, energy demand and set price. Suppliers also indicate their own requirements, such as the minimum price and operational constraints, which depend on the energy type and production costs. Electricity price may vary throughout the day in a free retail market.

E. Two Bidding Rounds

Utilities normally favor variable energy (wind and solar) because they are free resources. So suppliers of variable energy compete first. To secure energy for

any remaining demand from the first round, the other non-intermittent sources, like controllable load and renewable energy, participate in the second round. For instance, storage can accommodate the need of the power system any time.

F. Sealed Bidding

Suppliers are independent and compete privately for more than one time slot (a bundle of items). We allow partial bidding to increase the buyer pay-off. Sealed auctions are simple, foster competition among bidders and avoid their collusion [35], [46]. It has been shown in the literature that sealed auctions attract more participants than ascending open auctions.

7.2.3 Winner Determination

The above auction mechanism leads to a complex Winner Determination (WD) problem. Searching for the best solution (a set of winning sellers) in traditional CRAs (i.e. multiple items, single unit and single attribute) is already difficult to solve due to the computational complexity [62]. Previous studies adopted exact algorithms to look for the optimal solution in CRAs but endured an exponential time cost [28], which is unpractical in real-life auctions. To address this issue, researchers introduced evolutionary optimization techniques, such as Genetic Algorithms (GAs), which produce high-quality solutions with an excellent time-efficiency [73]. Furthermore, dealing with several conflicting attributes makes it even more difficult and time consuming to solve. That is why Multi-Objective Optimization (MOO) methods have been introduced to find the best trade-off solution that minimizes the cost and time. In this study, we customize the GA-based MOO algorithm proposed for CRAs with multiple units, multiple

attributes and conflicting objectives [74]. Our WD technique returns a solution that consists of several winning suppliers selected to provide electricity for multiple time slots. This solution should satisfy all the buyer requirements as well as the supplier constraints and offers. It represents the best combination of suppliers that lowers the price and increases the energy.

7.2.4 Auction Validation

In this present study, we conduct a real case study to illustrate our electricity procurement auction, WD method and generated solution. Afterwards, we validate our WD approach through simulated data by generating large instances of our advanced CRA problem (multiple items, two attributes and two conflicting objectives). The goal of the experiments is on one hand to assess the time-efficiency of our WD method, and on the other hand its significant superiority to well-known heuristic and exact WD techniques that have been proposed for much simpler CRAs. The execution time is not the only critical auction requirement but the quality of the winning solution is important too. Thus, we fully implement two well-known exact algorithms to solve our complex electricity trading problem in order to evaluate the accuracy of our WD method (how near to optimal is the generated solution).

7.3 Auctioning Electricity from Diverse Sources

Our electricity procurement auction is conducted with six major phases described in the following sections.

7.3.1 Auction Demand

A. Delivery Period

The time interval (peak time) of the needed energy, which is split into slots of fifteen minutes (called items). Usually this period is small (1 or 2 hours). There are few studies where one-hour term has been considered for electricity combinatorial auctions but with different purposes (i.e. long term contracts).

B. Energy Volume

The energy required for the demand period. It is defined with the minimum amount (to avoid a blackout) and a maximum amount (to avoid an excess).

C. Items and Constraints

Each item is described with three constraints: minimum and maximum energy and maximum allowable price. These constraints are necessary so that suppliers do not submit unrealistic prices and volumes.

7.3.2 Supplier Registration and Constraints

Potential green power suppliers (those already connected to smart meters) are then invited to the auction, and buyer requirements are fully disclosed to them. With smart metering, any power supplier would be able to transfer electricity to the power grid. Interested suppliers, residents and plants, register to the auction to provide electricity according to the auction demand. Each participant submits two constraints:

- The minimum price for each item.

- The operational constraint i.e. how long the supplier will be able to stay active during the trading period (after turning ON from the OFF status).

7.3.3 Supplier Bids with Two Rounds

Our auction is sealed-bid i.e. does not reveal any information about the competitor offers in order to protect their privacy. With a sealed-bid protocol, truth telling is the dominant bidder strategy [59]. Participants compete realistically on two item attributes: energy quantity and the corresponding price. If a seller bids for a certain item, he should realize the contract if he is the winner of that item. To reduce the uncertainty of electricity due to variable energy, we take into account other energy types by designing our auction with two rounds (each round lasts 20 minutes).

Round 1: This round is for the suppliers of variable energy. Only these sellers can submit partial bids because they might not be able to generate electricity all the time. When these suppliers bid for an item, we assume that they are able to allocate the energy according to the weather forecast. We may note that predicting wind for the short term is more accurate than for the long term.

Round 2: This round is for other energy sources. In case the solution from the first round is not complete, then other sources, like controllable renewable energy and active controllable load, can bid for all the remaining items: (1) items that do not have any placed bids, and/or (2) items that do not have a winner from the first round.

7.3.4 Winner Determination

Our WD algorithm searches efficiently for the best trade-off solution that satisfies all the buyer requirements as well as supplier constraints and valid bids. The solution

represents the best combination of suppliers that lowers the price and increases the energy. More precisely, it consists of a set of winning suppliers, their prices, volumes and schedules. We have several winners for the auction and one winner for each item. The submitted bids are accepted partially or fully depending on the constraint satisfaction. To produce the best trade-off solution, in [74], a very efficient GA-based MOO algorithm was introduced to solve CRAs with multiple items, units and objectives of attributes. We customize the algorithm of [74] specifically for our electricity auction: multiple items, single unit, two attributes and two conflicting objectives (increasing the energy amount and decreasing the price). For more details, refer to [74]. In Algorithm (Figure 7.1), we give an overview of our WD approach. All the placed offers for both rounds should be first validated (see equations 7.1 and 7.2) to make sure they respect all the trading requirements.

Algorithm 7.1: WD for Electricity CRA

Inputs: Requirements of buyer

Constraints of suppliers

Output: Best set of suppliers

1. Volume-price bid submission: partial or full bidding

2. Bid Validation:

$$Demand_{mini} \leq SupplyBid_{si} \leq Demand_{maxi} \quad (7.1)$$

$$Price_{minsi} \leq PriceBid_{si} \leq Price_{maxi} \quad (7.2)$$

3. Initial Solution Generation: randomly generate initial solutions based on a uniform distribution.

4. Winning Solution Generation:

- 4.1. Calculate fitness values of the solutions by using different utility functions for demand maximization and price minimization.
- 4.2. Improve the solutions with three GA operators (selection, crossover and mutation).
- 4.3. Apply diversity (crowding distance) to not end-up with similar solutions.
- 4.4. Apply elitism with an external population to keep the best solutions.
- 4.5. Among the two sets of solutions (GA-based and elite), select the best ones (w. r. t. their fitness values) as the participant solutions for the next generation.
- 4.6. After repeating steps 4.1 to 4.5 for a certain number of generations, return the best trade-off solution.
- 4.7. If any item remains, the auction goes to the second round (repeat steps 1 to 4).

Figure 7.1: WD algorithm for electricity CRA

The WD method first generates initial solutions randomly based on uniform distribution [74]. More precisely, it selects randomly a seller for each item, and then checks whether this selection is feasible or not by verifying two equations: the chosen seller has indeed bided for that item, and the supply of seller is possible because he is still active for that time slot since he started transferring electricity for the buyer. In case of an infeasible selection, the algorithm tries new sellers if possible.

$$\wedge SellerBid_{si} == true \quad (7.3)$$

$$ActiveDuration_s \geq CertainTime - StartingTime_s \quad (7.4)$$

where

- $SellerBid_{si}$ = true if seller s has placed a bid for item i ; false otherwise.

- *ActiveDuration_s* is the operational constraint of seller *s*.
- *CertainTime* is a certain time slot within the delivery period.
- *StartingTime_s* is the time when seller *s* turned ON from OFF status.

Afterwards, the method improves the initial solutions with three GA operators, selection (Gambling-Wheel Disk [29]), crossover (Modified Two-Point [69]) and mutation (Swap Mutation [23]), based on the quality measurement values (fitness values [74]). To prevent the population from having many similar solutions, the algorithm uses the diversity mechanism that is based on the updated crowding distance method introduced in [74]. In this variant, a relative fitness function is derived to calculate the distance between chromosomes (candidate solutions). By doing so, we prevent our WD algorithm from converging prematurely to local optima. Elite solutions are the best solutions found in each generation. Thus, we utilize the elitism technique to store these solutions in an external population [74]. The target here is to avoid losing good solutions and help our algorithm to converge to the global optima.

Among the two sets of solutions (GA-based and elite), the method selects the best ones as the participant solutions for the next generation. After repeating this optimization process for a certain number of times, it returns one single near-to-optimal solution (sometimes optimal). The way we build the winning solution ensures that the buyer gets energy for each time slot by respecting all the buyer requirements and seller constraints. Our method is able to return a high quality solution in a very efficient time as demonstrated in the experiment section. For instance, if 100 sellers compete for 8 items with two conflicting attributes, then the solution space is $2 \times 100^{2^8}$, which is a very large. Exact algorithms become infeasible (as shown in the experiment section)

because they deal with the entire solution space, and even heuristic ones take a certain amount of time. This is not practical in real-life applications like online auctions. Our method processes a subset of the solution pool and is able to improve the fitness quality in a very short time.

7.3.5 Trade Settlement

The utility allocates the required electricity w. r. t. the planned trading schedule and constraints. To conduct a successful delivery of energy, we consider the following assumptions:

- All suppliers are OFF at the beginning of the demand period.
- Switching every fifteen minutes among different suppliers is not an issue for the power grid.

7.4 A Case Study

We have implemented the proposed WD algorithm in Java using NetBeans IDE 8.0.2 and execute it on an Intel (R) core (TM) i3-2330M CPU with 2.20 GHz processor speed and 4 GB of RAM. Here we illustrate the proposed CRA with a small-scale electricity market (8 items and 5 sellers). The utility would like to secure energy for the period of 11am to 1pm with a minimum of 700 KW and a maximum of 850 KW. The buyer also specifies his needs for each item (see Table 7.1). Since we are dealing with two conflicting attributes (demand and price), the buyer needs to rank the attributes to be able to find a trade-off solution. He might prefer one attribute over another for all the items, or one for few items and another for the remaining ones. In our present scenario, demand has a higher importance than price for the 8 items.

Table 7.1: Buyer requirements

Item	Minimum Demand (KW)	Maximum Demand (KW)	Maximum Price (\$)
Item1 (11:00 - 11:15)	100	110	20
Item2 (11:15 - 11:30)	120	130	25
Item3 (11:30 - 11:45)	80	90	15
Item4 (11:45 - 12:00)	100	120	20
Item5 (12:00 - 12:15)	50	75	13
Item6 (12:15 - 12:30)	100	125	22
Item7 (12:30 - 12:45)	75	100	18
Item8 (12:45 - 13:00)	75	100	17
Total Demand	700	850	

Table 7.2: Constraints of wind and solar suppliers

Supplier	Minimum Price (\$) for items	ON (Hours)
S1 (Wind)	{18, 23, 14, -, -, -, -, -}	2
S2 (Solar)	{-, -, -, -, 10, 20, -, -}	1
S3 (Wind)	{17, 24, 13, 19, 12, 20, 17, -}	1

Let us assume we have five grid-connected power facilities (2 wind, 1 hydroelectricity, 1 battery storage and 1 solar-resident) that registered to this auction. The variable-energy suppliers compete first. Table 7.2 shows the minimum price of the 3 sellers for each of the 8 items, and how long they can stay active. For example, S1 might supply energy for Item1 at the minimum price of \$18 and after getting ON, S1 stays active for 2 hours. The symbol ‘-’ means during that time interval there would be no energy generation from the seller. Next, sellers submit their qualified bids consisting of their preferred supply and price for the items of their choice (see Table 7.3). For instance, S1 bided only for three items; for Item1, he can supply 105 KW for \$20. We can see there are no bids for Item8.

Table 7.3: Valid bids of wind and solar suppliers

Item Supplier	S1 (Wind)	S2 (Solar)	S3 (Wind)
Item1	{105, 20}	-	{110, 18}
Item2	{125, 24}	-	{122, 25}
Item3	{85, 14}	-	{85, 13}

Item4	-	-	{110, 20}
Item5	-	{72, 10}	{50, 12}
Item6	-	{110, 21}	{120, 22}
Item7	-	-	{95, 18}
Item8	-	-	-

Our WD algorithm solves the combinatorial problem above. Table 7.4 shows the breakdown of one of the candidate solutions. However, this solution is invalid since it does not satisfy the feasibility condition of equation (3): S3 has been selected for Item3 and again for Item7, which means that source must be active for 75 minutes but the active duration of S3 is only 1 hour. Also, this solution does not respect the feasibility condition of equation (7.4) because S1 has been chosen for Item 4 but did not bid for it. So the WD algorithm tries other sellers for Item4 and Item7 if any satisfies both conditions of feasibility. At the end of the first round, it returns the best solution given in Table 7.4, which is still not complete because there is no feasible solution found for Item7 and no placed bids for Item8.

Table 7.4: Candidate and winning solutions for first round

Item	Candidate Solution (infeasible)	Winning Solution (partial)
Item1	S1 (Wind)	S1 (Wind)
Item2	S1 (Wind)	S1 (Wind)
Item3	S3 (Wind)	S3 (Wind)
Item4	S1 (Wind)	S3 (Wind)

Item5	S2 (Solar)	S2 (Solar)
Item6	S2 (Solar)	S3 (Wind)
Item7	S3 (Wind)	-
Item8	-	-

For the next round, hydro and battery storage compete for the remaining two items. Table 7.5 exposes their constraints and valid bids. Supplier S5 and S4 are the winners of Item7 and Item8 respectively.

Table 7.5: Constraints and Valid Bids of Hydro and Battery

Supplier	Minimum Price (\$) for Item7 & Item8	ON (Hours)	Valid Bid
S4 (Hydro)	{15, 15}	2	{98, 16}, {100, 15}
S5 (Battery)	{17, 15}	2	{99, 17}, {95, 16}

In conclusion, the bid-taker will receive in total 816 KW with a price of \$141:

- From first round: 617 KW (545 from Wind and 72 from Solar) with a price of \$109.
- From second round: 199 KW (99 from Battery and 100 from Hydro) with a price of \$32.

7.5 Validation and Comparison

We analyze the auction outcome in terms of two quality metrics: solution optimality and time-complexity. We perform three experiments with simulated data. We generate randomly five instances of our CRA problem (8 items, 2 attributes and 2 conflicting objectives). In Table 7.6, we give the details of the five artificial datasets by varying the number of sellers, items and generations.

Table 7.6: Simulated datasets

Dataset #1			
Constant		Variable	
Number of Sellers	500	Number of Generations	1-500
Number of Items	Round1: 15, Round2: 5		
Dataset #2			
Constant		Variable	
Number of Sellers	500	Number of Items	Round1: 3, 6, 9, 12, 15 Round2: 1, 2, 3, 4, 5
Number of Generations	100		
Dataset #3			

Constant		Variable	
Number of Items	Round1: 15, Round2: 5	Number of Sellers	100, 200, 300, 400, 500, 600, 700, 800, 900, 1000
Number of Generations	100		
Dataset #4			
Constant		Variable	
Number of Sellers	Round1: 60, Round2: 40	N. A.	
Number of Items	Round1: 24, Round2: 8		
Number of Generations	100		
Dataset #5			
Constant		Variable	
Number of Sellers	Round1: 600, Round2: 400	N. A.	
Number of Items	Round1: 15, Round2: 5		
Number of Generations	100		

The ranking and objectives of the attributes are also generated randomly. The maximum value of each attribute is randomly produced from [100, 1000], and the minimum value from [10% of maximum value, 50% of maximum value]. Furthermore, we perform several parameter-tuning tests, and based on the results, we use the following best configuration, unless otherwise stated: number of solutions = 500, crossover rate = 0.6, mutation rate = 0.01, number of elite solutions = number of participant solutions and elite solution rate = 0.2. All the results returned by WD method represent the average values of 20 runs.

Experiment 1: Statistical Analysis. We examine statistically the WD algorithm based on dataset #1. Fig. 7.1 presents the average quality measurement value with maximum and minimum values of generations. It also depicts the error bars with a confidence level of 95%. It is noticeable that after a certain number of generations, the maximum fitness value remains constant. This means that the best solution found by WD might be the optimal one. It is also obvious that WD is able to control the solution variations and their differences after a certain number of generations.

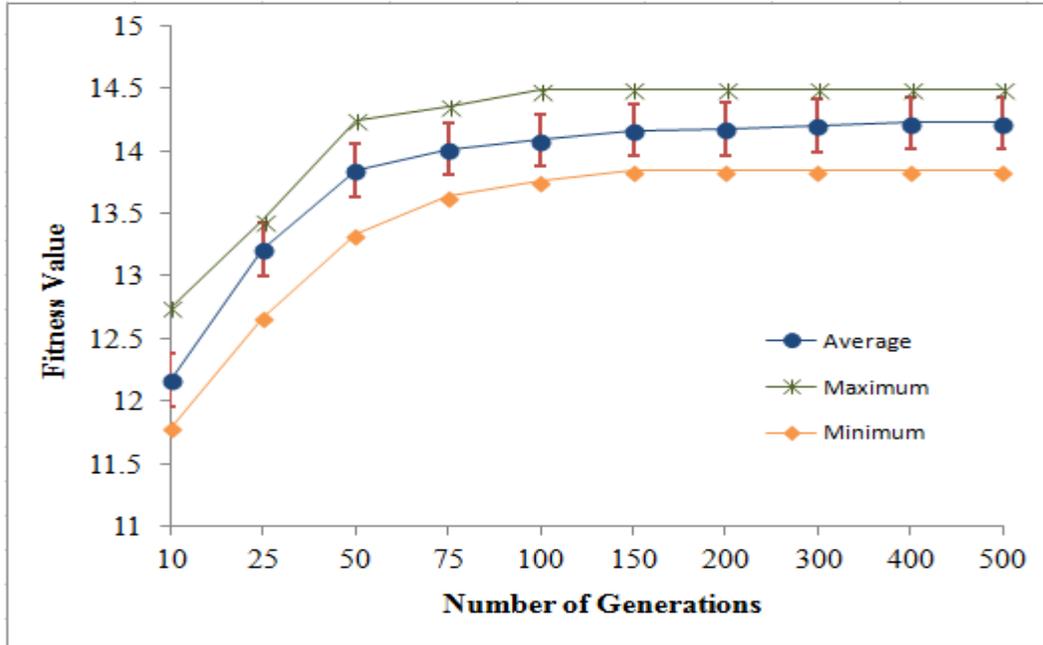


Figure 7.2(a): Statistical analysis of WD (Round1)

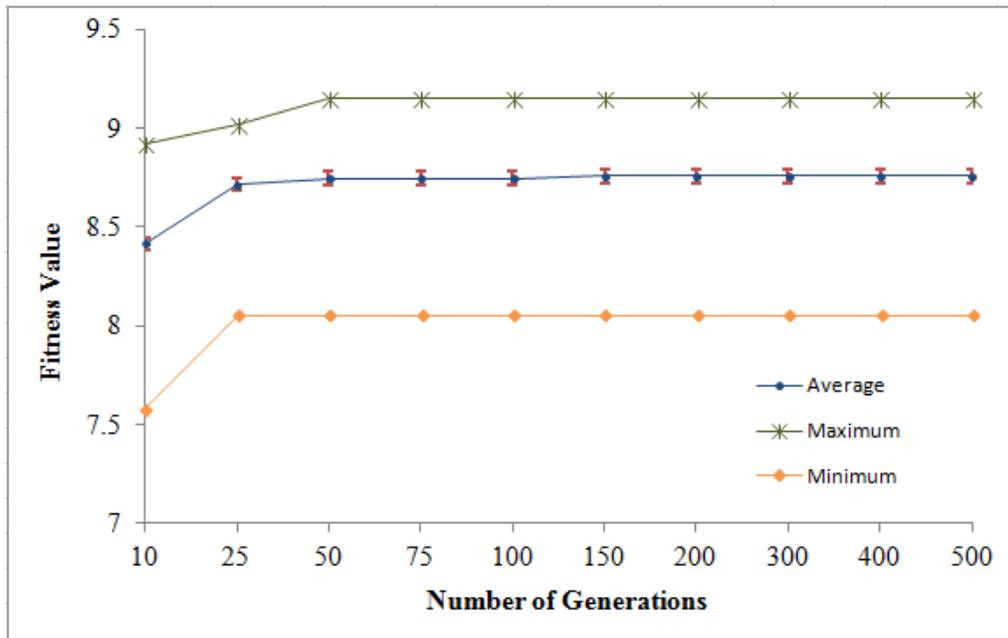


Figure 7.2(b): Statistical analysis of WD (Round2)

Experiment 2: Computational Time. The goal here is to assess the time-efficiency of our WD algorithm. We utilize dataset #2 by varying the number of items, and dataset #3 the number of sellers. From the results displayed in figures 7.2 and 7.3, we may say that the required computational expense is not exponential but rather polynomial. It is clear that the execution time increases linearly with the increase of items and sellers.

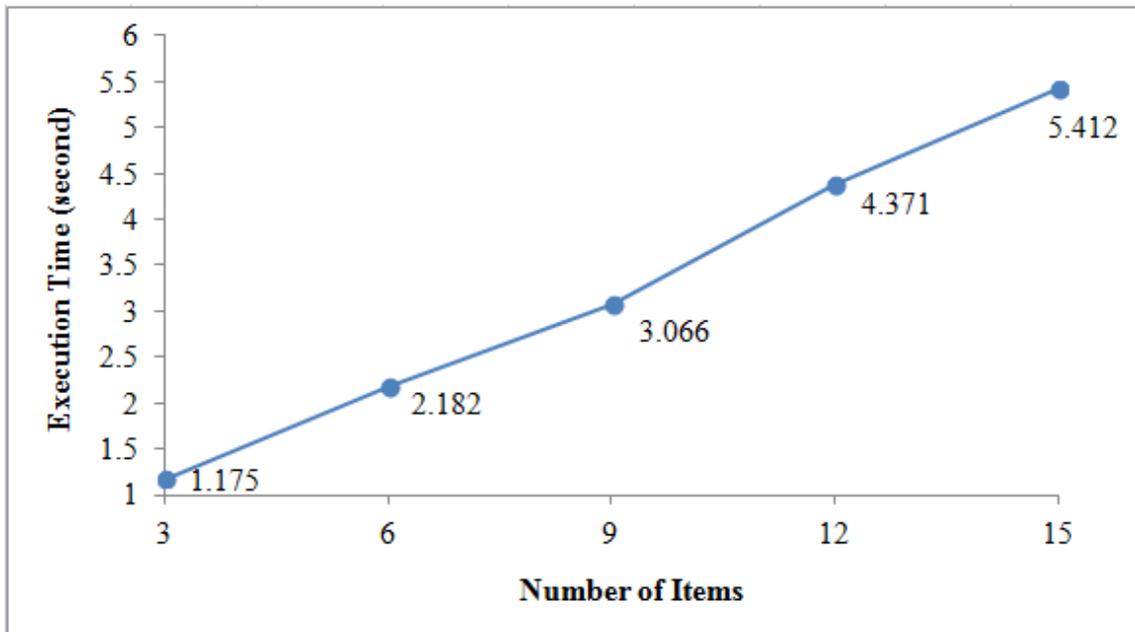


Figure 7.3(a): Computational time of WD by varying number of items (Round1)

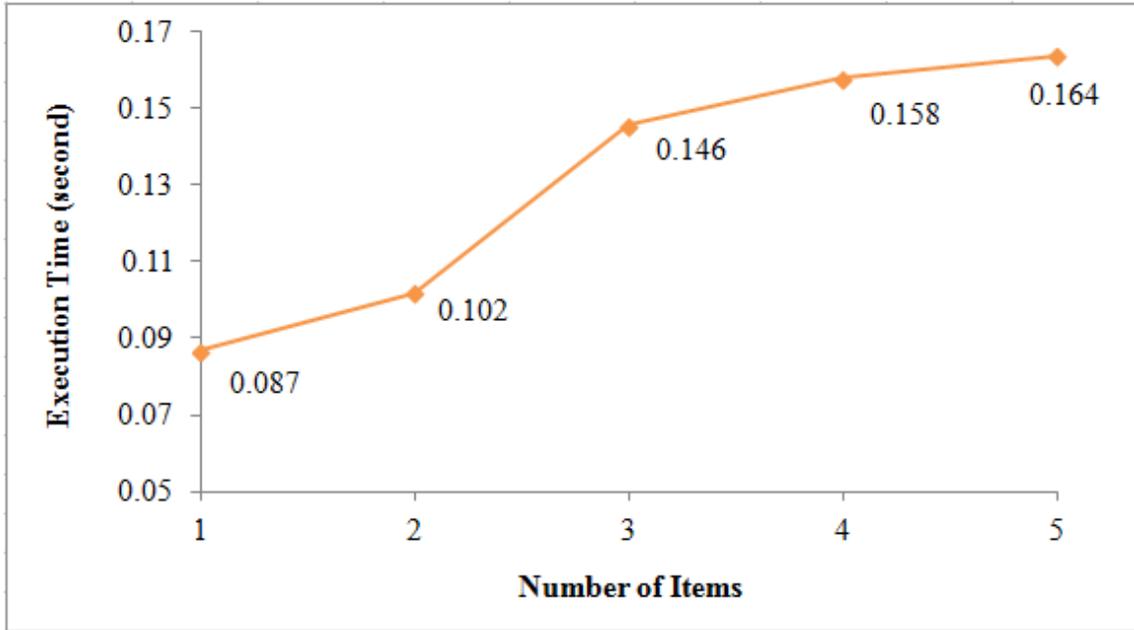


Figure 7.3(b): Computational time of WD by varying number of items (Round2)

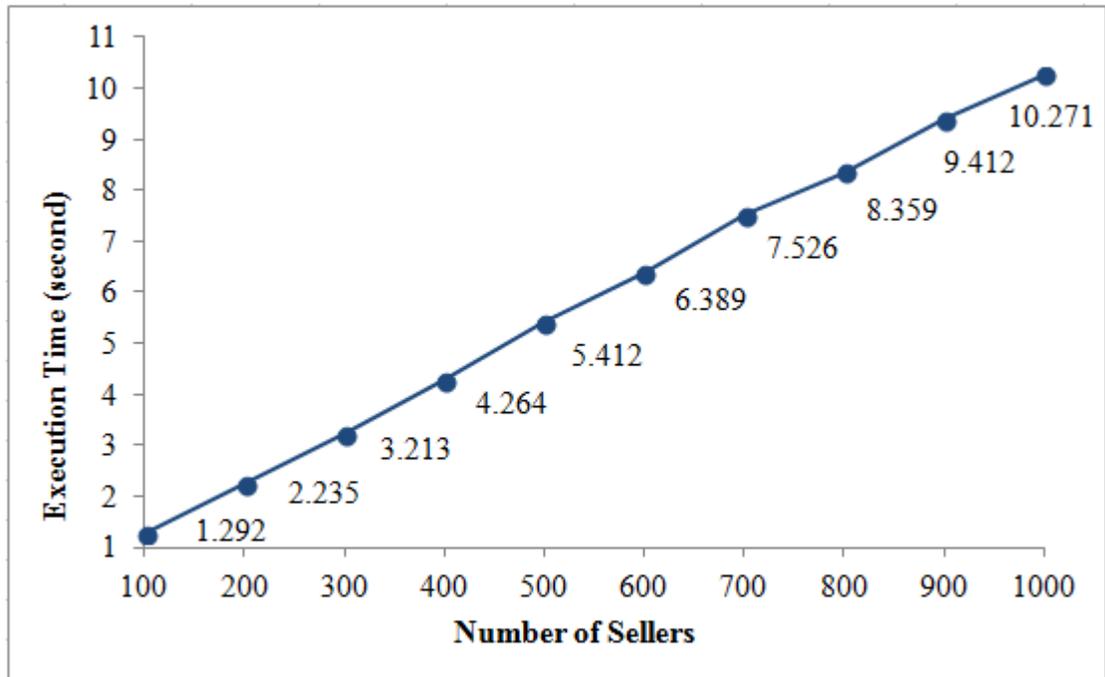


Figure 7.4(a): Computational time of WD by varying number of sellers (Round1)

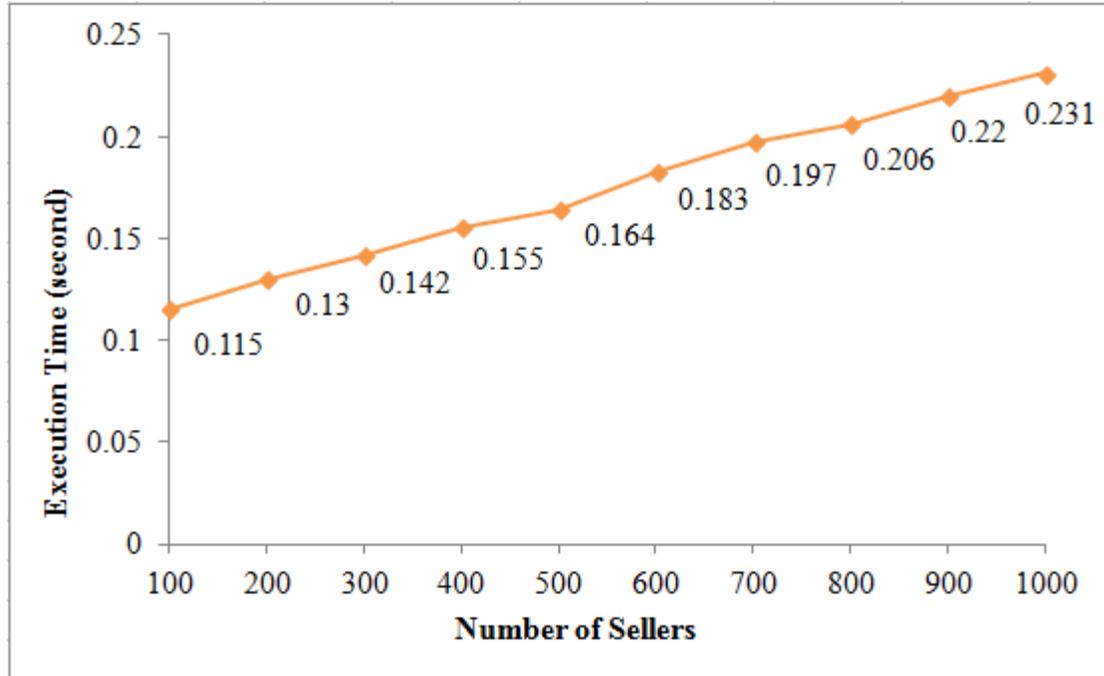


Figure 7.4(b): Computational time of WD by varying number of sellers (Round2)

Experiment 3a & 3b: Computational Time Comparison with Heuristic Algorithms. We compare the computational expense of WD with three other well-know heuristic WD methods: Improved Ant Colony (IAC), Enumeration Algorithm with Backtracking (EAB), and Genetic Algorithms for Multiple Instances of Items in Combinatorial Reverse Auctions (GAMICRA). All these WD methods return one single best solution. This comparison is based on dataset #4. Table 7.7 provides the processing time of IAC, EAB and GAMICRA for much simpler combinatorial auctions. The first two were taken from [62] and the last one from [73]. As we can see WD is significantly superior to all of them.

Table 7.7: Computational expense comparison

	WD Algorithm	Computation Time (second)	
Heuristics	IAC	9	
	EAB	3	
	GAMICRA	0.83	
	WD		0.226 (Round1),
			0.019 (Round2)

Experiment 4: Accuracy and Time-efficiency Comparison with Exact Algorithms. We have fully implemented in Java two exact procedures to solve our constrained combinatorial auction problem: Brute Force that guarantees the solution optimality because it checks the entire solution space and Branch and Bound that is the most time-performing exact algorithm. The goal here is to compare the time-efficiency and accuracy (how near is the solution to the optimal) of our proposed WD with the performance of these two exact algorithms. We measure the *accuracy* as $\frac{WD}{Exact} \times 100\%$. We use datasets #4 and #5 for this experiment. The former set (relatively smaller) is for Brute Force and the latter set is for Branch and Bound.

After applying Brute Force (BF) on the first set, we obtain the following results. In round 1, 60 sellers compete for 24 items and the solution space is 60^{24} . In round 2, 40 sellers compete for 8 items, and the search space is 40^8 . In round 1, the accuracy of WD is 87.3%; and in round 2, 93.3% where the accuracy of BF is 100% in both rounds. So we can conclude that our WD method is able to return near to optimal solutions. On

the other hand, our method returns the solution in 0.182 seconds in round 1 and 0.063 seconds in round 2 whereas BF takes more than 1 day in round 1 and almost 1 day in round 2. So in total WD takes 0.245 seconds where BF takes more than 2 days.

Now we will discuss the second exact technique. In round 1, 600 sellers bid for 15 items. The accuracy of WD is 91.32% and WD returns a solution in 10.271 seconds whereas Branch and Bound (BB) takes 39.687 minutes with 100% accuracy. In round 2, 400 sellers compete for 5 items. The accuracy of WD is 95.62% and returns a solution in 0.231 seconds whereas BB takes 33.375 seconds with 100% accuracy. Therefore, in total WD takes 10.502 seconds where BB takes 40.243 minutes. All the results are summarized in Table 7.8. It is clear that exact algorithms become unreasonable with the increased values of the input auction parameters. In conclusion, we have demonstrated that our WD method not only produces solution in a very efficient processing time but also generates near-to-optimal solutions.

Table 7.8: Comparison with exact algorithms

Comparison with Brute Force			
Method	Round	Solution Quality: Fitness value (accuracy)	Required Time
Brute Force	Round1	23.0126 (100%)	> 1 day
	Round2	7.9878 (100%)	~ 1 day
WD	Round1	20.0891 (87.3%)	0.182 seconds
	Round2	7.4526 (93.3%)	0.063 seconds
Comparison with Branch & Bound			

Method	Round	Solution Quality: Fitness value (accuracy)	Required Time
Branch & Bound	Round1	13.12 (100%)	39.687 minutes
	Round2	4.978 (100%)	33.375 seconds
WD	Round1	11.98 (91.32%)	10.271 seconds
	Round2	4.76 (95.62%)	0.231 seconds

7.6 Related Work

There are several studies regarding the mechanism design of electricity auctions. We first examine traditional (non-combinatorial) auctions and then more advanced (combinatorial) ones. Both auction types have several limitations, for example multiple attributes and objectives along with trading constraints are not considered. Generally speaking, WD approaches in e-auctions have been classified into three categories [81]: 1) exact approach that guarantees solution optimality but comes with huge computational cost, 2) approximated approach with low computational cost but solution optimality is not ensured, and 3) restricted approach with solution optimality and low computational cost but does not consider other features like partial bidding, multiple units and attributes, buyer and seller constraints. Researchers limit the auction features and parameters for the sake of simplicity.

7.6.1 Traditional Auctions

The non-combinatorial electricity auctions that we examine here are reverse protocols with single unit and most of them with single attribute (price). In [82], the authors addressed the management problem of energy resources of the future smart power grids in the context of multi-player negotiation. The proposed exact multi-agent approach is based on a knowledge-based modeling and a set of independent and competitive players. Due to the increased competition of players and the energy management environment being dynamic and complex, the legacy infrastructures are not suitable to accomplish the current need of energy management. The authors mentioned that artificial-intelligence based applications can tackle this situation by providing auction systems with appropriate decision support. They also pointed it out that the usage of renewable energy sources has been increasing due to fossil fuels shortage and environmental concerns (i.e. low impact on greenhouse gas emissions). Lastly, this paper conducted a detailed case study on energy resource management, including storage, production and load response.

Another research [35] introduced the Probability Bidding Mechanism (PBM), a new auction protocol that takes into account several constraints to maximize the trading quantities and total profits and on the other hand minimize the trading price. In this work, the authors considered single unit of each item and multiple attributes, such as the electricity price, transmission cost, network congestion and technology constraints. They implemented an agent-based system to develop and validate PBM by comparing it with the High-Low Matching bidding mechanism and performing empirical analysis. They claimed that PBM significantly optimizes electricity procurement auctions and promotes

economic growth. However, they stated that PBM is just an ideal model and has not been yet applied to real electricity auction markets. They also stated that making an appropriate auction bidding protocol in electricity auctions is the most challenging issue.

Lastly, [47] analyzed the electricity wholesale markets of most of the European countries by focusing on the Day Ahead Market, Intra Day Market and Balance Market. This paper compared the markets based on several bidding mechanisms, price formation and timing. The target of the auctions was to reduce the management cost. The authors mentioned that a good knowledge of the electricity markets is the key to gain profit from a single or a portfolio of power plants. They also analyzed the projects conducted by the European agencies in order to harmonize the wholesale electricity markets in the continent. Moreover, according to one of the agencies, several hundred million per year can be saved by integrating wholesale electricity markets.

7.6.2 Advanced Auctions

There are few electricity combinatorial auctions despite the fact that they match the demand and supply more efficiently and maximize the buyer payoff as well [59]. Even though these types of auctions are difficult to develop, they have been successfully adopted in other domains for both government and private sectors (free markets), like food supplying contracts, airport runway time slots and bus transportation routes [28].

In [59], the authors tackled power auctions with multiple units for both single and multiple items. The main advantage of combinatorial auctions is that they provide very performing resource allocation and maximise the revenue for the auctioneer. On the other hand, the disadvantage is the complexity of determining the winners. Researchers narrowed down the complexity by allowing only atomic bidding (accept only full bids),

which limits the profit of the auctioneer. The authors allowed partial bidding to maximize the auctioneer profit. They exposed two exact WD algorithms with constrained bidding for the electricity retail market: optimal single-item and optimal multiple-item. Daily auctions are held where the utility companies compete for 24 items (each item is 1 hour). The authors utilized Vickrey-Clarke-Groves bidding strategy since in this protocol the dominant strategy of bidders is to submit their true valuation of the items. They validated the proposed methods by comparing them with an optimal exact WD algorithm.

More recently, [28] developed a Web-based CRA (single unit and single attribute) with user interfaces for electricity retail markets to minimize buyer expenditures. This auction allows consumers to open auctions, define hourly consumption amounts and choose suppliers with the cheapest energy acquisition. The authors claimed that this flexibility of consumers creates more competition among suppliers, and ultimately increases the number and gain of the suppliers and consumers respectively. They employed the well-known optimizer named IBM CPLEX to determine the auction winners. They proved that their protocol produces efficient allocation of electricity usage because consumers purchase energy from several companies to minimize their expenditures. Nevertheless, this auction system was not tested yet with real markets.

In [77], it has been shown that evolutionary algorithms are suitable and perform well in the context of combinatorial auctions. [77] exposed an Evolutionary Iterative Random Search Algorithm defined for auctions with multiple units and multiple rounds. They showed that their protocol achieved Nash Equilibrium in the following way: assume all the bidders offer the same price at the beginning; if any bidder bids higher the utility

value of his bid will decrease; otherwise, the utility value will be zero. The same case holds for sellers. This situation indicates that the market is in Nash Equilibrium. The convergence of prices of buyer and bidders assures the market equilibrium.

Another research [87] presented a GA-based optimal resource allocation approach for combinatorial auctions with multiple units and multiple rounds. This method, shown to be feasible and effective through simulation results, is able to maximize the total trading amounts of sellers and to reduce the processing time in the context of WD problem. [87] claimed that when the resource allocation problem has feasible solutions, then Nash equilibrium is always guaranteed.

Moreover, in [81], the authors proposed a Nash Equilibrium Search Approach (NESA) that consists of a local search procedure and an evolutionary algorithm. They solved the WD problem in the context of standard combinatorial auctions. They validated the proposed procedure by measuring the performance of their Nash-Equilibrium solution based on revenue performance, anytime performance and optimal solution comparison. They showed that NESA performs pretty well for large (2000 bidders and 200 items) and small (1000 bidders and 100 items) scale settings. They also discussed the stability of Nash Equilibrium to solve the WD problem in combinatorial auctions (after a certain time, the equilibrium is established and then stays in the equilibrium position). NESA is able to produce near to optimal solutions.

7.7 Conclusion

Electricity consumption is increasing rapidly due to the growth of population, economy and infrastructures. To avoid any energy outage during peak load periods, with the help of online auctions, grid companies can contract electricity from diverse

renewable-energy sources (in aggregate). We have designed a combinatorial reverse electricity auction specifically for short-term resource allocation. The auction first gives priority to variable-energy suppliers and then to storage and controllable renewable energy. We have solved our constrained combinatorial procurement problem (multiple items, two attributes and two conflicting objectives) with a multi-objective evolutionary optimization technique to be able to find the best trade-off solution i.e. the best combination of sellers that lowers the cost and increases the energy quantity. We have demonstrated that our WD method not only determines the winners in a very efficient processing time but also generates near-to-optimal solutions. We view our electricity allocation as a set of multiple optimal allocations, each one corresponds to an item for which the constraints of the auction demand are met and a more profitable allocation is not possible. In Experiment 4 (accuracy comparison), we have proved that our WD method is able to produce near-to-optimal solutions, which means near-to-optimal resource allocation is achieved. [87] pointed out that combinatorial auctions that achieve the resource allocation (because of feasible solutions) ultimately fulfil the Nash Equilibrium (one type of market equilibrium). Therefore, we can claim that our WD algorithm also satisfies the market equilibrium. The proposed electricity auction will promote the expansion of renewable and sustainable energy as well as home-based generation with new technologies accessible to residents (electric vehicles and solar panels).

To adopt our new electricity auction, public utilities and their decision makers, which have deep knowledge about reformatting the electricity markets, need to establish policies for contracting electricity with private companies and individuals. These

policies include the auction fees, registration of qualified power suppliers, bid submission, service allocation, and payment and transmission costs. Additionally, utilities should analyze in practice the economical efficiency of the proposed market.

Chapter 8

Conclusion and Future Work

This chapter is organized into two sections. The first section provides the nut-shells of our research and the second section suggests some promising issues for future research.

8.1 Conclusion

Determining the winners for Combinatorial Reverse Auctions (CRAs) is a challenging task because of its computational complexity. Our Winner Determination (WD) models address several types of advanced CRAs dealing with various dimension of multiplicity of units, attributes, objectives and rounds.

We can conclude our study by the following remarks:

- We have addressed the WD as an evolutionary Multi-Objective Optimization (MOO) problem with mixed minimization and maximization criteria. We have solved these complex WD problems by using genetic algorithms that we have integrated with our variants of diversity and elitism strategies.
- Even though evolutionary algorithms are very powerful to address large-scale MOO applications, they however require advanced design mechanisms, and their

implementation is difficult. Our WD approaches return single best solutions based on the known constraints of buyers and sellers. We have also proposed a strategy to maximize the buyer's revenue in CRAs.

- Furthermore, we have conducted case studies and simulated experiments and showed that our solving methods are more prominent than without the diversity and elitism schemes. Additionally, we have verified experimentally the performances of our WD approaches in terms of three quality metrics. We have showed that diversity and elitism strategies increased the solution optimality. Finally, we have demonstrated the significant superiority of our WD protocols to other popular WD techniques implemented for simpler CRAs.
- Speaking of electricity CRAs, electricity consumption is increasing rapidly due to the growth of population, economy and infrastructures. To avoid any energy outage during peak load periods, with the help of online auctions, grid companies can contract electricity from diverse renewable-energy sources (in aggregate). We have designed electricity CRA specifically for short-term resource allocation. The auction first gives priority to variable-energy suppliers and then to storage and controllable renewable energy. We have solved our constrained combinatorial procurement problem (multiple items, two attributes and two conflicting objectives) with a multi-objective evolutionary optimization technique to be able to find the best trade-off solution i.e. the best combination of sellers that lowers the cost and increases the energy quantity. We have demonstrated that our WD method not only determines the winners in a very

efficient processing time but also generates near-to-optimal solutions. We also show that our WD algorithm satisfies the market equilibrium.

8.2 Future Work

There are some promising issues of our proposed WD protocols that can pave the ways for future research. By addressing these issues the WD methods can be more efficient and well-established. Some future work directions are discussed in the following.

- In our study, it is up to the seller to negotiate on a particular item or not. But if he bids on an item, we assume he has sufficient stock of that item. In the future, even though the seller has insufficient stock of an item, he can still compete for that item. Speaking of evolutionary algorithms, we will certainly obtain better solution quality by combining our WD method with other suitable algorithms, such as Simulated Annealing, Ant Colony and Tabu Search. This is a prominent future work in the context of CRAs.
- Though this is the most reliable model, we will examine parallel genetic algorithms to improve the time-efficiency. Moreover, we will include qualitative attributes along with quantitative ones, never handled before in the combinatorial context. We would also like to investigate the interactive optimization approach to incorporate the buyer's decisions and constraints during the iterations of optimization.
- Nevertheless, for our electricity CRAs, utilities should analyze in practice the economical efficiency of the proposed market. To adopt our new electricity

auction, public utilities and their decision makers, which have deep knowledge about reformatting the electricity markets, need to establish policies for contracting electricity with private companies and individuals. These policies include the auction fees, registration of qualified power suppliers, bid submission, service allocation, payment, and transmission costs. The proposed electricity auction will promote the expansion of renewable and sustainable energy as well as home-based generation with new technologies accessible to residents (electric vehicles and solar panels).

References

- [1] R. Abbasian and M. Mouhoub, *An efficient hierarchical parallel genetic algorithm for graph coloring problem*, In Proceedings of 13th Annual Genetic and Evolutionary Computation Conference (GECCO), pp. 521–528, 2011.
- [2] R. Abbasian and M. Mouhoub, *A hierarchical parallel genetic approach for the graph coloring problem*, Applied Intelligence, 39(3), pp. 510-528, 2013.
- [3] E. Alanazi and M. Mouhoub, *Managing qualitative preferences with constraints*, In Neural Information Processing, Springer Berlin Heidelberg, pp. 653-662, 2012.
- [4] E. Alanazi, M. Mouhoub and B. Mohammed, *A preference-aware interactive system for online shopping*, Computer and Information Science, 5(6), pp. 33-42, 2012.
- [5] V. Avasarala, T. Mullen, D. L. Hall, and A. Garga, *MASM: A market architecture or sensor management in distributed sensor networks*, In Proceedings of SPIE Defense and Security Symposium, pp. 5813–5830, 2005.
- [6] V. Avasarala, H. Polavarapu and T. Mullen, *An approximate algorithm for resource allocation using combinatorial auctions*, In Proceedings of the International Conference on Intelligent Agent Technology, pp. 571-578, 2006.

- [7] H. Bayindir, H. Kilic, and M. Rehan, *An agent-based trading infrastructure for combinatorial reverse auctions*, In Proceedings of IEEE Symposium on Intelligent Agents (IA), pp. 38-44, 2014.
- [8] M. Bichler, *An experimental analysis of multi-attribute auctions*, Decision Support Systems, 29(3), pp. 249-268, 2000.
- [9] M. Bichler, L. Juhnyoung, S. L. Ho, and J. Y. Chung, *ABSolute: an intelligent decision making framework for e-sourcing*, In Proceedings of the Third International Work-shop on Advanced Issues of E-Commerce and Web-Based Information Systems (WECWIS), pp. 195-201, 2001.
- [10] M. Bichler and J. Kalagnanam, *Configurable offers and winner determination in multi-attribute auctions*, European Journal of Operational Research, 160(2), pp. 380-394, 2005.
- [11] M. Bichler, M. Kaukal, and A. Segev, *Multi-attribute auctions for electronic procurement*, In Proceedings of the First IBM IAC Workshop on Internet Based Negotiation Technologies, pp. 154-165, 1999.
- [12] T. Buer and H. Kopfer, *A Pareto-metaheuristic for a bi-objective winner determination problem in a combinatorial reverse auction*, Computers & Operations Research, 41, pp. 208-220, 2014.
- [13] T. Buer and G. Pankratz, *Solving a bi-objective winner determination problem in a transportation procurement auction*, Logistics Research, 2(2), pp. 65-78, 2010.

- [14] T. S. Chandrashekar, Y. Narahari, C. H. Rosa, D. M. Kulkarni, J. D. Tew, and P. Dayama, *Auction-based mechanisms for electronic procurement*, IEEE Transactions on Automation Science and Engineering, 4(3), pp. 297-321, 2007.
- [15] A. Ciarreta, M. P. Espinosa, and C. Pizarro-Irizar, *Has renewable energy induced competitive behavior in the Spanish electricity market?* Energy Policy, 104, pp. 171-182, 2017.
- [16] C. A. Coello Coello, *Evolutionary multi-objective optimization: a historical view of the field*, IEEE Computational Intelligence Magazine, 1(1), pp. 28-36, 2006.
- [17] B. Colak, M. A. Gokmen, and H. KOIOc, *A web-based auction platform for electricity retail markets*, In Proceedings of the IEEE 14th International Conference on Machine Learning and Applications (ICMLA), pp. 1148-1152, 2015.
- [18] P. Cramton, Y. Shoham, and R. Steinberg, *An overview of combinatorial auctions*, SIGecom Exchanges, 7(1), pp. 3-14, 2007.
- [19] A. Das and D. Grosu, *Combinatorial auction-based protocols for resource allocation in grids*, In Proceedings of the 19th IEEE International on Parallel and Distributed Processing Symposium, pp. 1-8, 2005.
- [20] K. Deb, *Multi-objective optimization*, Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques, pp. 403-449, 2014.

- [21] A. M. Easwaran and J. Pitt, *An agent service brokering algorithm for winner determination in combinatorial auctions*, In Proceedings of the 14th European Conference on Artificial Intelligence, pp. 286-290, 2000.
- [22] R. M. Ebrahim, J. Razmi, and H. Haleh, *Scatter search algorithm for supplier selection and order lot sizing under multiple price discount environment*, Advances in Engineering Software, 40(9), pp. 766-776, 2009.
- [23] A. E. Eiben and E. S. James, *Introduction to evolutionary computing*, Heidelberg: Springer, 53, 2003.
- [24] R. Epstein, L. Henriquez, J. Catalan, G. Y. Weintraub, C. Martinez, and F. Espejo, *A combinatorial auction improves school meals in Chile: a case of OR in developing countries*, International Transactions in Operational Research, 11(6), pp. 593-612, 2004.
- [25] A. M. Geoffrion and R. Krishnan, *Prospects for operations research in the e-business era*, Interfaces, 31(2), pp. 6-36, 2001.
- [26] F. Ghavamifar, S. Sadaoui and M. Mouhoub, *Preference elicitation and winner determination in multi-attribute auctions*, In Proceedings of the International Florida Artificial Intelligence Research Society Conference (FLAIRS), AAAI, pp. 93-94, 2011.
- [27] D. E. Goldberg and K. Deb, *A comparative analysis of selection schemes used in genetic algorithms*, Foundations of Genetic Algorithms, pp. 69-93, 1991.

- [28] R. Gonen and D. Lehmann, *Optimal solutions for multi-unit combinatorial auctions: branch and bound heuristics*, In Proceedings of the 2nd ACM conference on Electronic commerce, pp. 13-20, 2000.
- [29] J. Gong, J. Qi, G. Xiong, H. Chen, and W. Huang, *A GA based combinatorial auction algorithm for multi-robot cooperative hunting*, In Proceedings of the International Conference on Computational Intelligence and Security, pp. 137-141, 2007.
- [30] S. Grafakos, A. Flamos, and E. M. Ensenado, *Preferences matter: a constructive approach to incorporating local stakeholders' preferences in the sustainability evaluation of energy technologies*, Sustainability, 7(8), pp. 10922-10960, 2015.
- [31] S. Guo and S. Sanner, *Real-time multiattribute bayesian preference elicitation with pairwise comparison queries*, In Proceedings of the International Conference on Artificial Intelligence and Statistics, pp. 289-296, 2010.
- [32] Y. Guo, J. P. Muller, and C. Weinhardt, *Learning user preferences for multi-attribute negotiation: an evolutionary approach*, Multi-Agent Systems and Applications III, 2691, pp. 303-313, 2003.
- [33] D. Gupta and S. Ghafir, *An overview of methods maintaining diversity in genetic algorithms*, International Journal of Emerging Technology and Advanced Engineering, 2(5), pp. 56-60, 2012.

- [34] K. L. Gwebu, M. Y. Hu, and M. S. Shanker, *An experimental investigation into the effects of information revelation in multi-attribute reverse auctions*, Behaviour & Information Technology, 31(6), pp. 631-644, 2012.
- [35] D. Han and M. Sun, *The design of a probability bidding mechanism in electricity auctions by considering trading constraints*, SIMULATION, 0037549715605618, 2015.
- [36] H. H. Hoos and C. Boutilier, *Solving combinatorial auctions using stochastic local search*, In Proceedings of the 17th National Conference on Artificial Intelligence, pp. 22–29, 2000.
- [37] D. Jannach and G. Kreutler, *Personalized user preference elicitation for e-services*, In Proceedings of the IEEE International Conference on e-Technology, e-Commerce and e-Service, pp. 604-611, 2005.
- [38] W. Jiang and S. Sadaoui, *Evaluating and ranking semantic offers according to users' interests*, Journal of Electronic Commerce Research, 13(1), pp. 1-22, 2012.
- [39] M. Karny and T. V. Guy, *Preference elicitation in fully probabilistic design of decision strategies*, In Proceedings of the 49th IEEE Conference on Decision and Control, pp. 5327-5332, 2010.
- [40] P. Klemperer, *Auctions: theory and practice*, Economics Papers, Economics Group, Nuffield College, University of Oxford, 2004.

- [41] A. Konak, D. W. Coit, and A. E. Smith, *Multi-objective optimization using genetic algorithms: a tutorial*, Reliability Engineering & System Safety, 91(9), pp. 992-1007, 2006.
- [42] D. S. Knysh and V. M. Kureichik, *Parallel genetic algorithms: a survey and problem state of the art*, Journal of Computer and Systems Sciences International, 49(4), pp. 579-589, 2010.
- [43] H. Lucas, R. Ferroukhi, and D. Hawila, *Renewable energy auctions in developing countries*, International Renewable Energy Agency (IRENA), 2013.
- [44] B. Mansouri and E. Hassini, *A Lagrangian approach to the winner determination problem in iterative combinatorial reverse auctions*, European Journal of Operational Research 244(2), pp. 565-575, 2015.
- [45] R. Marambio and H. Rudnick, *A novel inclusion of intermittent generation resources in long term energy auctions*, Energy Policy, 100, pp. 29-40, 2017.
- [46] L. T. A. Maurer and L. Barroso, *Electricity auctions: an overview of efficient practices*, A World Bank Study, Washington, DC: World Bank. 2011.
- [47] N. Mazzi, A. Lorenzoni, S. Rech, and A. Lazzaretto, *Electricity auctions: a European view on markets and practices*, In Proceedings of the 12th IEEE International Conference on the European Energy Market (EEM), pp. 1-5, 2015.

- [48] M. Mbarki, J. Bentahar, B. Moulin, and A. Moazin, *Constraints-based negotiation using argumentation*, In Proceedings of the 13th International Workshop on Non-Monotonic Reasoning, pp. 1-9, 2010.
- [49] M. Mia, S. P. Mudur, and T. Radhakrishnan, *An interactive system for negotiation in e-commerce with incremental user knowledge*, In Proceedings of Conference of the Centre for Advanced Studies on Collaborative Research, pp. 170-184, 2005.
- [50] M. Mouhoub, *Systematic versus local search and GA techniques for incremental SAT*, International Journal of Computational Intelligence and Applications, 7(1), pp. 77-96, 2008.
- [51] M. Mouhoub and B. Jafari, *Heuristic techniques for variable and value ordering in CSPs*, In Proceedings of the 13th annual conference on Genetic and evolutionary computation, pp. 457-464, 2011.
- [52] H. Muhlenbein, *Evolution in time and space - The Parallel Genetic Algorithm*, Foundations of Genetic Algorithms, pp. 316-337, 1991.
- [53] T. Mullen, V. Avasarala, and D. L. Hall, *Customer-driven sensor management*, IEEE Intelligent Systems, 21(2), pp. 41-49, 2006.
- [54] Y. Narahari and P. Dayama, *Combinatorial auctions for electronic business*, Sadhana 30 (pt. 2 & 3), pp. 179–211, 2005.
- [55] L. Nie, X. Xu, and D. Zhan, *Coevolutionary computation based iterative multi-attribute auctions*, Enterprise Interoperability III, Springer London, pp. 461-469, 2008.

- [56] M. Nowostawski and R. Poli, *Parallel genetic algorithm taxonomy*, In Proceedings of the 3rd International Conference on Knowledge-Based Intelligent Information Engineering Systems, pp. 88-92, 1999.
- [57] J. Ostler and P. Wilke, *Improvement by combination how to increase the performance of optimisation algorithms by combining them*, In Proceedings of the 10th international conference of the practice and theory of automated time tabling, p. 359-365, 2014.
- [58] P. Patodi, A. K. Ray, and M. Jenamani, *GA based winner determination in combinatorial reverse auction*, In Proceedings of 2nd International Conference on Emerging Applications of Information Technology (EAIT), pp. 361-364, 2011.
- [59] Y. K. Peña and N. R. Jennings, *Combinatorial markets for efficient energy management*, In Proceedings of the IEEE International Conference on Intelligent Agent Technology, pp. 626-632, 2005.
- [60] A. Pommeranz, J. Broekens, P. Wiggers, W. P. Brinkman, and C. M. Jonker, *Designing interfaces for explicit preference elicitation: a user-centered investigation of preference representation and elicitation process*, User Modeling and User-Adapted Interaction, 22(4-5), pp. 357-397, 2012.
- [61] S. Prestwich, F. Rossi, K. B. Venable, and T. Walsh, *Constraint-based preferential optimization*, AAAI, 5, pp. 461-466, 2005.

- [62] X. Qian, M. Huang, T. Gao, and X. Wang, *An improved ant colony algorithm for winner determination in multi-attribute combinatorial reverse auction*, In Proceedings of the IEEE Congress on Evolutionary Computation (CEC), pp. 1917-1921, 2014.
- [63] S. J. Rassenti, V. L. Smith, and R. L. Bulfin, *A combinatorial auction mechanism for airport time slot allocation*, The Bell Journal of Economics 13, pp. 402–417, 1982.
- [64] S. Sadaoui and S. K. Shil, *A multi-attribute auction mechanism based on conditional constraints and conditional qualitative requirements*, Journal of Theoretical and Applied Electronic Commerce Research (JTAER), 11(1), pp. 1-25, 2016.
- [65] A. R. Sadeghi, M. Schunter, and S. Steinbrecher, *Private auctions with multiple rounds and multiple items*, In Proceedings of the 13th International Workshop on Database and Expert Systems Applications, pp. 423-427, 2002.
- [66] P. Senthilkumar and P. Shahabudeen, *GA based heuristic for the open job scheduling problem*, International Journal of Advanced Manufacturing Technology, 30, pp. 297-301, 2006.
- [67] D. Shao-bin, F. Yu qiang, L. Ke xing, and S. Xun cheng, *Preference elicitation of the NSS in the e-commerce*, In Proceedings of the International Conference on Management Science and Engineering, pp. 135-140, 2007.
- [68] S. K. Shil, M. Mouhoub, and S. Sadaoui, *An approach to solve winner determination in combinatorial reverse auctions using genetic algorithms*, In Proceedings of the 15th Annual Genetic and Evolutionary Computation Conference (GECCO), pp. 75–76, 2013.

- [69] S. K. Shil, M. Mouhoub, and S. Sadaoui, *Winner determination in combinatorial reverse auctions*, Contemporary Challenges & Solutions in Applied AI, 489, pp. 35-40, 2013.
- [70] S. K. Shil and M. Mouhoub, *Considering multiple instances of items in combinatorial reverse auctions*, In Proceedings of the 27th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE), Lecture Notes in Computer Science, Modern Advances in Applied Intelligence, 8482, pp. 487-496, 2014.
- [71] S. K. Shil and S. Sadaoui, *Constraint and qualitative preference elicitation in multi-attribute reverse auctions*, In Proceedings of the 27th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE), Lecture Notes in Computer Science, Modern Advances in Applied Intelligence, 8482, pp. 497-506, 2014.
- [72] S. K. Shil, M. Mouhoub and S. Sadaoui, *Evolutionary technique for combinatorial reverse auctions*, In Proceedings of the 28th International Florida Artificial Intelligence Research Society Conference (FLAIRS), pp. 79-84, 2015.
- [73] S. K. Shil, M. Mouhoub, and S. Sadaoui, *Winner determination in multi-attribute combinatorial reverse auctions*, In Proceedings of the 22nd International Conference on Neural Information Processing (ICONIP), pp. 645-652, 2015.
- [74] S. K. Shil and S. Sadaoui, *Winner determination in multi-objective combinatorial reverse auctions*, In Proceedings of the 28th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 714-721, 2016.

- [75] L. R. Smeltzer and A. Carr, *Reverse auctions in industrial marketing and buying*, Business Horizons, 45(2), pp. 47-52, 2002.
- [76] T. R. Srinath, M. P. Singh, and A. R. Pais, *Anonymity and verifiability in multi-attribute reverse auction*, International Journal of Information Technology Convergence and Services, 1(4), pp. 1-8, 2011.
- [77] S. Tabandeh and H. Michalska, *An evolutionary random search algorithm for double auction markets*, In Proceedings of the IEEE Congress on Evolutionary Computation, pp. 2948-2955, 2009.
- [78] D. M. Tate, A. E. Smith, *A genetic approach to the quadratic assignment problem*, Computers and Operations Research, 22(1), pp. 73–83, 1995.
- [79] J. E. Teich, H. Wallenius, J. Wallenius, and O. R. Koppius, *Emerging multiple issue e-Auctions*, European Journal of Operational Research, 159(1), pp. 1-16, 2004.
- [80] J. E. Teich, H. Wallenius, J. Wallenius, and A. Zaitsev, *A multi-attribute e-auction mechanism for procurement: theoretical foundations*, European Journal of Operational Research, 175(1), pp. 90-100, 2006.
- [81] C. Tsung, H. Ho, and S. Lee, *An equilibrium-based approach for determining winners in combinatorial auctions*, In Proceedings of the IEEE 9th International Symposium on Parallel and Distributed Processing with Applications (ISPA), pp: 47-51, 2011.

- [82] Z. A. Vale, H. Morais, and H. Khodr, *Intelligent multi-player smart grid management considering distributed energy resources and demand response*, In Power and Energy Society General Meeting, IEEE, pp: 1-7, 2010.
- [83] W. Vickrey, *Counterspeculation, auctions, and competitive sealed tenders*, The Journal of Finance, 16(1), pp. 8-37, 1961.
- [84] W. E. Walsh, M. Wellman, and F. Ygge, *Combinatorial auctions for supply chain formation*, In Proceedings of the ACM Conference on Electronic Commerce, pp. 260-269, 2000.
- [85] H. Watabe and T. Kawaoka, *Application of multi-step GA to the traveling salesman problem*, In Proceedings of the 4th International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies, 2, pp. 510-513, 2000.
- [86] S. Wu and G. E. Kersten, *Information revelation in multi-attribute reverse auctions: an experimental examination*, In Proceedings of the 46th IEEE Hawaii International Conference on System Sciences (HICSS), pp. 528-537, 2013.
- [87] W. Xing-Wei, W. Xue-yi, and H. Min, *A resource allocation method based on the limited english combinatorial auction under cloud computing environment*, In Proceedings of the 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), pp. 905-909, 2012.
- [88] W. Xia and Z. Wu, *Supplier selection with multiple criteria in volume discount environments*, Omega 35(5), pp. 494-504, 2007.

- [89] C. Xu, L. Song, Z. Han, Q. Zhao, X. Wang, X. Cheng, and B. Jiao, *Efficiency resource allocation for device-to-device underlay communication systems: a reverse iterative combinatorial auction based approach*, IEEE Journal on Selected Areas in Communications, 31(9), pp. 348-358, 2013.
- [90] N. Yang, X. Liao, and W. W. Huang, *Decision support for preference elicitation in multi-attribute electronic procurement auctions through an agent-based intermediary*, Decision Support Systems, 57, pp. 127-138, 2014.
- [91] L. Zhang and R. Zhang, *The winner determination approach of combinatorial auctions based on double layer orthogonal multi-agent genetic algorithm*, In Proceedings of the 2nd IEEE Conference on Industrial Electronics and Applications, pp. 2382-2386, 2007.

Appendix A

Figures

In this appendix we include some figures of our advanced Combinatorial Reverse Auctions (CRAs). Figures A1, A2, and A3 show the convergence and diversity characteristics and figure A4 depicts the architecture of our Winner Determination (WD) protocol.

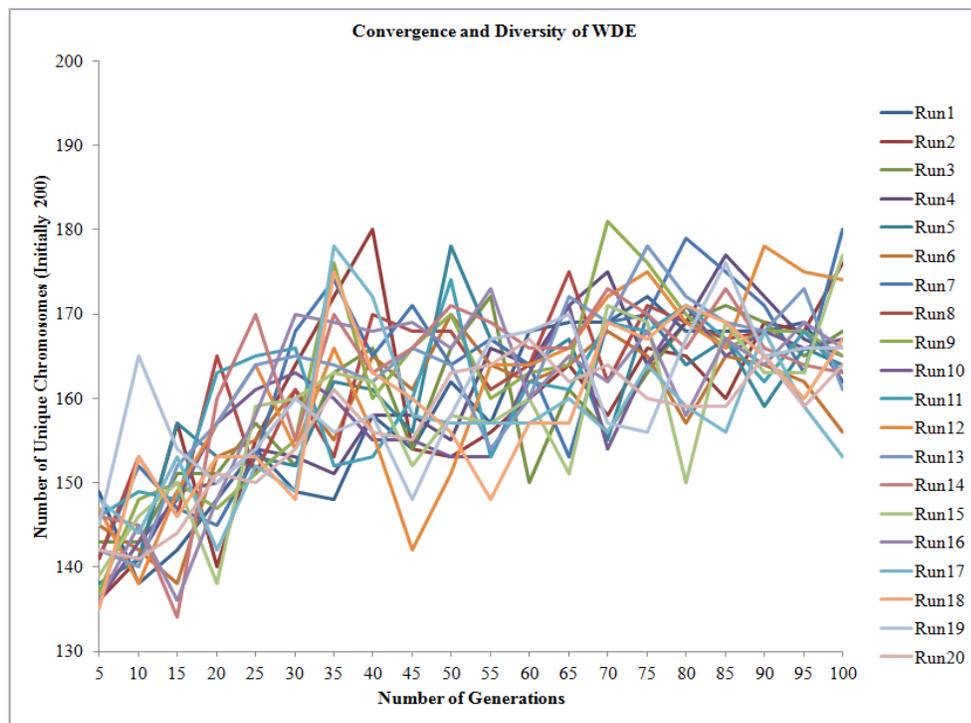


Figure A1: Convergence and Diversity of WD method integrated with Diversity and Elitism Techniques

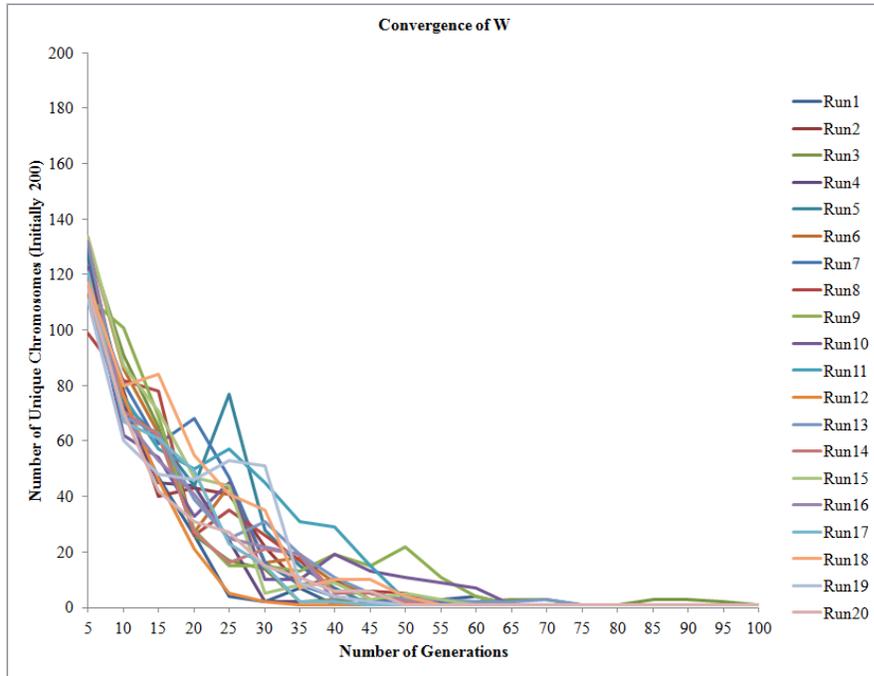


Figure A2: Convergence WD method without Diversity and Elitism Techniques

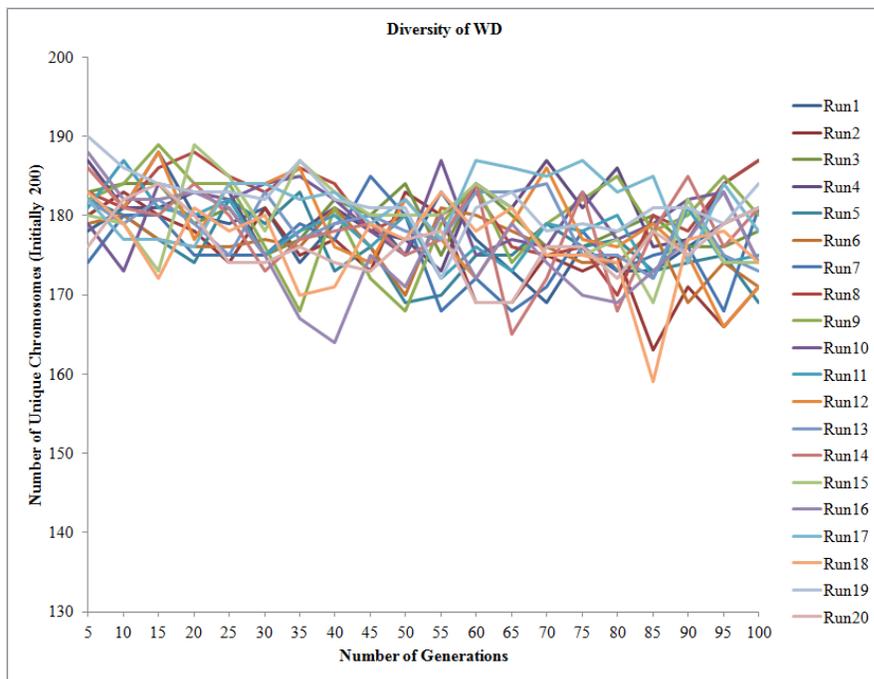


Figure A3: Diversity of WD method integrated with Diversity Technique

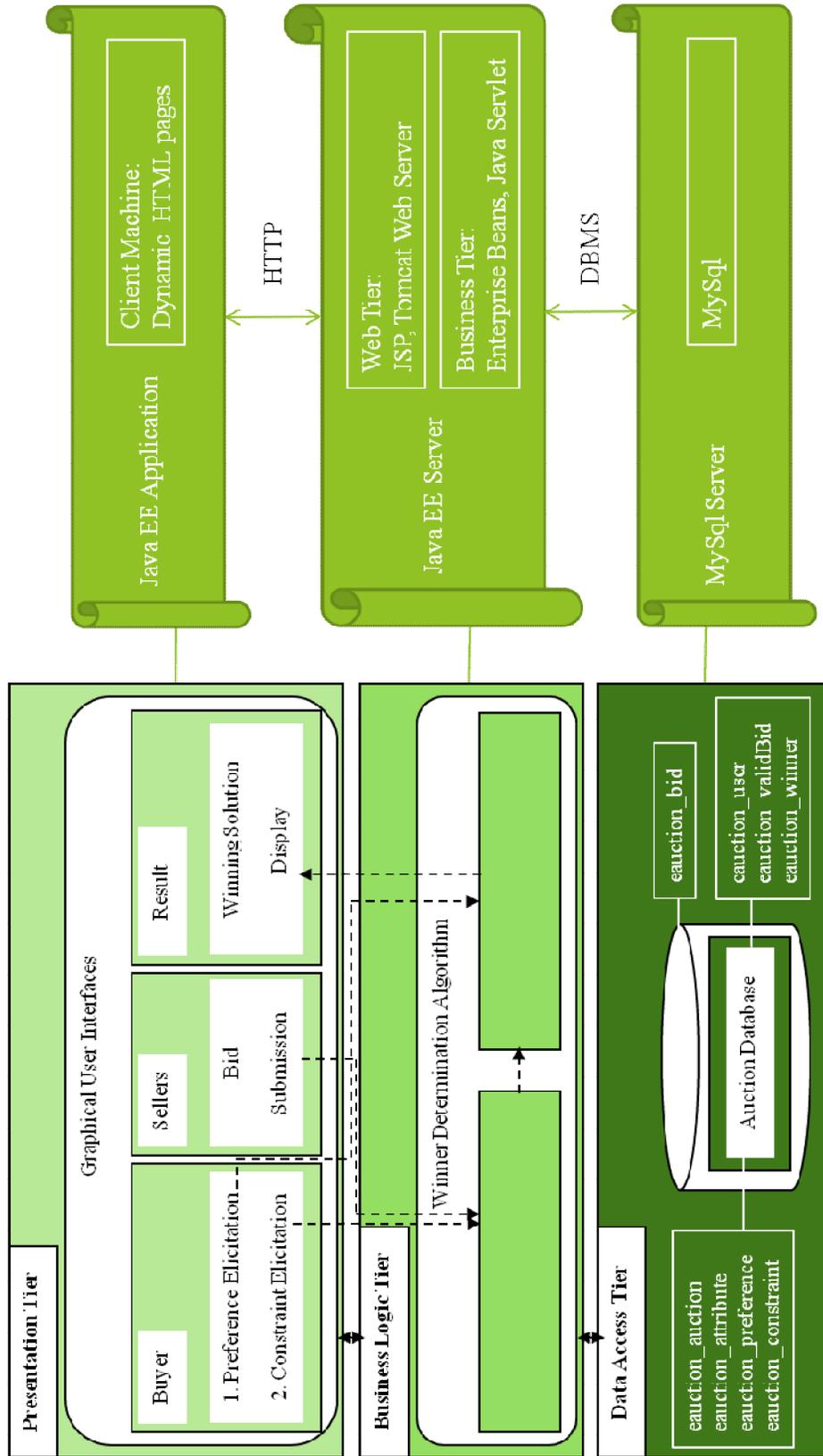


Figure A4: 3-tier Software Architecture

Appendix B

Graphical User Interfaces

In this appendix we include some Graphical User Interfaces (GUIs) of our advanced CRAs. We implement these GUIs by using Java.

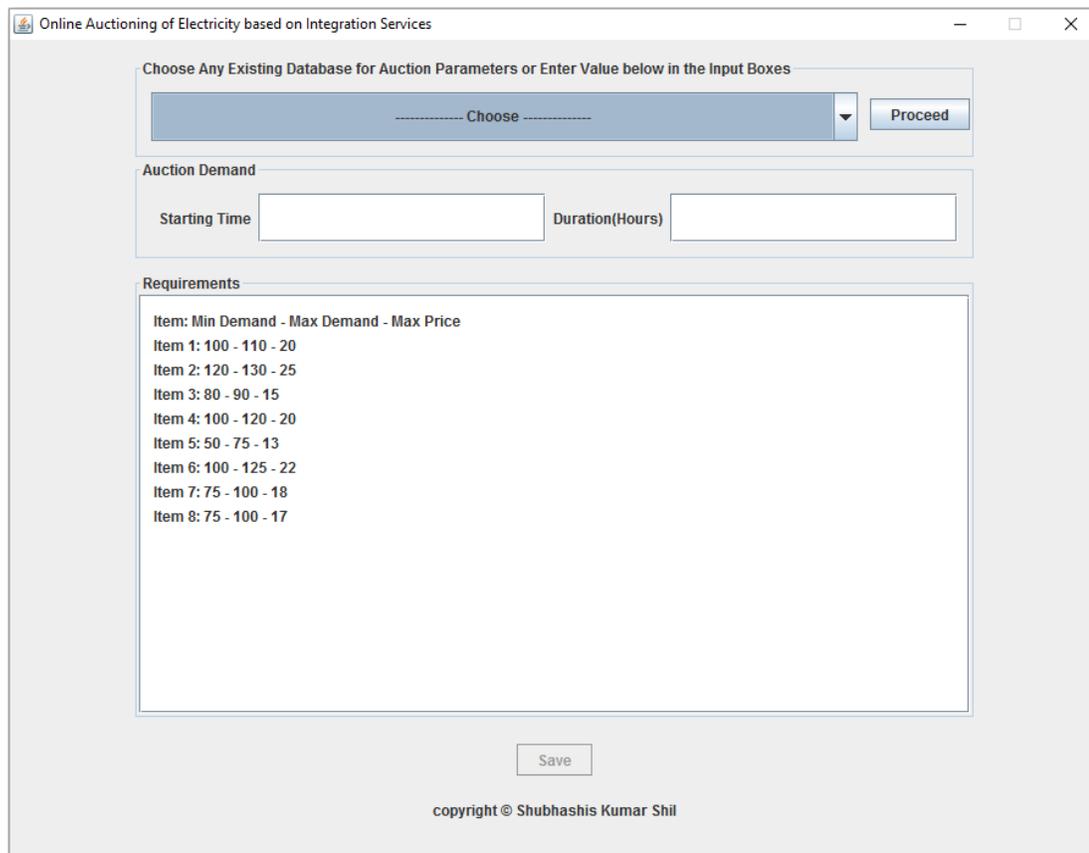


Figure B1: Buyer Requirement GUI

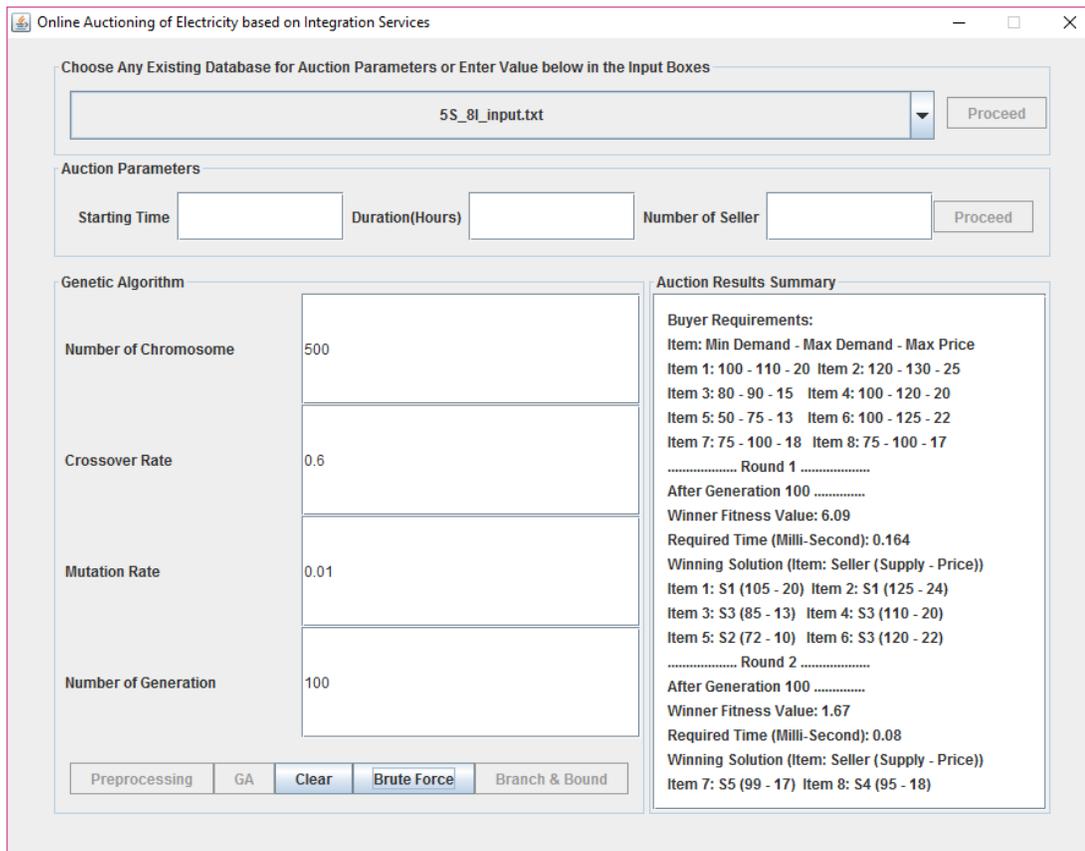


Figure B2: Case Study GUI

Appendix C

Algorithms

In this appendix we include two exact algorithms we developed, first one is Brute Force and second one is Branch and Bound WD methods.

Algorithm C1: Brute Force and Branch and Bound

Brute Force:

Begin:

generate entire solution space;

winner solution \leftarrow *first solution*;

while(not all solutions)

 Begin:

 evaluate solution based on fitness value;

 check feasibility of the solution;

 if(feasible)

 update winner solution;

 End;

return the winner;

End;

Branch and Bound:

Begin:

$activeSpace \leftarrow \{\emptyset\};$

$bestValue \leftarrow NULL;$

$incumbent \leftarrow NULL;$

while($activeSpace \neq empty$)

 Begin:

 choose branching solution, $S \in activeSpace$;

 remove S from $activeSpace$;

 generate children solutions, c_S of S and their optimistic bounds, ob_S

 while(not all children solution)

 Begin:

 if ($ob_S < bestValue$) delete child solution;

 else if update $bestValue$ and $incumbent$ by child

 solution;

 else add child solution to $activeSpace$;

 End;

 End;

 return the winner ($incumbent$);

End;

Figure C1: Brute Force and Branch and Bound pseudo code

Appendix D

Research Papers

During my PhD studies, we publish several research works in the context of different types of advanced CRAs to solve WD problems. Refereed abstracts, conference papers, book chapters, and journals are given below.

A. Abstracts

1. **S. K. Shil** and M. Mouhoub, *Evolutionary Techniques for Multiple Instances of Items in Combinatorial Reverse Auctions*, In Proceedings of the 27th International Florida Artificial Intelligence Research Society Conference (FLAIRS), pp. 525, 2014.
2. S. Sadaoui and **S. K. Shil**, *Qualitative Preference Elicitation in a MAUT-Based Reverse Auction System*, In Proceedings of the 27th International Florida Artificial Intelligence Research Society Conference (FLAIRS), pp. 524, 2014.

B. Conference Papers

1. **S. K. Shil** and S. Sadaoui, *Combinatorial Reverse Electricity Auctions*. In Proceedings of the 30th Canadian Conference on Artificial Intelligence (Canadian AI), pp. 162-168, 2017.

2. **S. K. Shil** and S. Sadaoui, *Winner Determination in Multi-Objective Combinatorial Reverse Auctions*, In Proceedings of the 28th IEEE International Conference on Tools with Artificial Intelligence (ICTAI), pp. 714-721, 2016.
3. **S. K. Shil**, M. Mouhoub and S. Sadaoui, *Evolutionary Technique for Combinatorial Reverse Auctions*, In Proceedings of the 28th International Florida Artificial Intelligence Research Society Conference (FLAIRS), pp. 79-84, 2015.

C. Book Chapters

1. **S. K. Shil** and S. Sadaoui, *Winner Determination in Multi-Attribute Combinatorial Reverse Auctions*, In Proceedings of the 22nd International Conference on Neural Information Processing (ICONIP), Lecture Notes in Computer Science, Springer International Publishing, 9491, pp. 645-652, 2015.
2. **S. K. Shil** and M. Mouhoub, *Considering Multiple Instances of Items in Combinatorial Reverse Auctions*, In Proceedings of the 27th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE), Lecture Notes in Computer Science, Modern Advances in Applied Intelligence, 8482, pp. 487-496, 2014.
3. S. Sadaoui and **S. K. Shil**, *Constraint and Qualitative Preference Elicitation in Multi-Attribute Reverse Auctions*, In Proceedings of the 27th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE), Lecture Notes in Computer Science, Modern Advances in Applied Intelligence, 8482, pp. 497-506, 2014.

D. Journals

1. **S. K. Shil** and S. Sadaoui, *Meeting Peak Electricity Demand through the Combinatorial Reverse Auctioning of Renewable Energy*, Journal of Modern Power System and Clean Energy (MPCE), 2017, [Accepted].
2. **S. K. Shil** and S. Sadaoui, *Multi-Objective Optimization in Multi-Attribute and Multi-Unit Combinatorial Reverse Auctions*, International Journal on Artificial Intelligence Tools (IJAIT), 2017, [to be published].
3. S. Sadaoui and **S. K. Shil**, *A Multi-Attribute Auction Mechanism based on Conditional Constraints and Conditional Qualitative Preferences*, Journal of Theoretical and Applied Electronic Commerce Research (JTAER), 11(1), pp. 1-25, 2016.

Moreover, just before starting PhD studies, we published another journal article of advanced CRAs.

S. K. Shil, S. Sadaoui, and M. Mouhoub, *Evolutionary Techniques for Reverse Auctions*, Intelligent Control and Automation, 4(4), pp. 371-378, 2013.