Waseda University Doctoral Dissertation

Study on Bidding Strategies using Genetic Network Programming

Chuan YUE

Graduate School of Information, Production and Systems
Waseda University

July 2012

Acknowledgements

Here I am going to say thanks to many people from whom I get a lot of help and support. This thesis could not be finished without the help and support of many people who are gratefully acknowledged here.

First and foremost, I would like to give my deepest appreciation and respect to my supervisor Professor Hirasawa for his constant encouragement and patient guidance, especially his kindness and care. During the research of my thesis, he taught me a lot about how to raise a new point, how to analyze and solve problems and how to write papers. Without his invaluable guidance, constructive advise, great support and patience, I may never be able to complete this thesis. His attitude toward work, brilliant thoughts toward research, healthy habits toward life and affability toward everyone impressed me deeply. He has shown us a respectable illustrious model of life, which I think is the most precious.

Secondly, I would like to express my great respect and gratitude to Prof. Hu, Prof. Yoshie and Prof. Fujimura, who make efforts to provide me lots of helpful comments and advise as the co-examiners.

Thirdly, I would like to thank Doctor Mabu, who help me a lot during my research. His invaluable suggestions and kindness are very important for the completion of my master courses. His help and support means a lot to me.

Thanks also goes to my friends and all fellow classmates, who always support and help me kindly. Special thanks should go to Dr. Lu Yu, Dr. Huiyu Zhou, Dr. Shanqing Yu, Mr. Jun Huang, Mr. Kira7, Dr. Deng Zhang, Dr. Feng Wen and Dr. Fengming Ye. They gave me plenty helpful suggestion and information in many aspects of my study and daily life. They are always by my side with sunshine smile. Without them, I couldn't have so beautiful memories in the past 5 years. I also owe my sincere gratitude to my best old friends, who are almost like my family and gave me their help, time and deep concern, especially Dr. Fang Cong, Mr. Miao Zhang and Ms. Chen Liu.

Moreover, I am most grateful my beloved Mum and Dad for their loving considerations and great confidence in me all through these years. They support me in spirit to go through the hard time during the difficult part of the thesis.

Last, but not least, I give my most respect to my grandparents sincerely. I love you.

Abstract

Due to the explosive development of global network structure, electronic commerce is increasingly playing an important role in many organizations and individual consumer's daily life. It offers opportunities to significantly improve the way for businesses interactions between both customers and suppliers. More and more large scale and decentralized e-commerce mechanisms have emerged in industrial and commercial domains in a wide range.

In particular, among all these applications, online auctions, which are flexible pricing mechanisms over internet, make the physical limitations of traditional auctions disappear. They gain their extra popularity in the daily life and attract globally dispersed users due to having the characteristics that ”bargaining” and ”negotiation” besides all of the convenience. Thus, online auctions become one of the most widely studied and employed negotiation mechanisms today. Traditionally, in most current online auction applications, the traders are generally humans who operate all the behaviors to make transactions. These behaviors may involve observing the auctions, analyzing the auction information, and bidding the suitable price for the items. However, facing the increasingly demanding requirements and complexity of online trading, this kind of manual operation does not reveal the full potential of this new mode of commerce. Thus, in order to relieve the users and be more effective, exploring possible types and automating the behaviors in the online auction attract high interest.

Now, in many studies, the agent-oriented auction mechanism, with its emphasis on autonomous actions and flexible interactions, arises as an effective and robust model for the dynamic and sensitive commerce environment. In such systems, the agent acts flexibly on behalf of its owner and is capable of local decision-making based on the environment information and pre-knowledge about the system.

Among many different types of online auction, two of the most popular and studied types are Multiple Round English Auctions (MREA), which is single side auction, and Continuous Double Auction (CDA), which is double side auction. These auctions are newly emerged in e-commerce era based on the traditional auction types. They allow multiple agents to participate and one agent can deal with several auctions continuously or simultaneously, which are effective auction types to save time and relieve the users. Towards to these types, because there is no centralized system-wide control, the major challenge for automatic bidding strategies is to improve the degree of automation and optimize the agent's bidding behavior in order to maximize the owner's profit. Most of the related researches have been conducted by using heuristic methods and fixed mathematical functions to compute the final optimal bidding price for the items or to compute how much should bid at each time step. Nevertheless, because auction environments are complicated and highly dynamic due to have many factors affecting each other, these approaches are not flexible enough for the dynamic environment, and there is no dominant strategy.

Against this background, this thesis is concerned with developing the intelligence of autonomous agent's bidding strategy in order to make the agent to be more efficient and competitive for agent-based online auction mechanisms, especially in MREA and CDA. In order to be more flexible and better exploit the market information, Genetic Network Programming (GNP) is firstly employed to the agent's bidding strategy since its applicability and efficiency have been clarified in complex and dynamic problems in many other fields. GNP is one of the evolutionary optimization techniques developed as an extension of Genetic Algorithm (GA) and Genetic Programming (GP), which uses compact directed graph structures as solutions. Basically speaking, in the proposed method, the GNP population represents the group of potential bidding strategies, and each individual uses the as-if/then decision-making functions to judge the auction information and guides the agent to take the suitable actions under different situations. Thus, it could be flexible and capable to adaptive to various auction situations. During the evolution, the GNP structure will be systematically organized, and finally, the individual which can obtain the highest profit is selected as the optimal bidding strategy at the end of training phase.

In chapter 2, we introduced the conception of MREA and CDA in detail, which are the study environments in this thesis. The related researches are also introduced.

In chapter 3, focusing on MREA, the bidding strategy for the auction agents in MREA is proposed using GNP. The performance of GNP-based agents is evaluated and studied in two situations: MREA is no time limit (NTL), and MREA is time limit (TL). Furthermore, according to the amount of the money each agent has, each situation is divided into 2 cases: general case and poorest case. All the participating agents in the simulations use GNP strategy. This chapter aims to study and analyze the capability and effectiveness of GNP for guiding bidding actions through the phenomenon of the simulations. The simulation results reveal that the agents using GNP strategy can understand various environments well through experiences and become smarter through evolution.

In chapter 4, as an extension of the bidding strategy in chapter 3, in order to improving the agent's intelligence and sensitivity, an enhanced bidding strategy for MREA is developed using GNP. Firstly, the GNP structure is modified to be able to judge more kinds of information and more situations at a time. Secondly, the strategy is improved to be able to consider the bidder's attitude towards to each good, which makes the strategy to be more personalized for each bidder and could make the bidder more satisfied with the auction result and profit. The proposed strategy is compared with the previous GNP strategy and the other conventional strategies in the simulations. The simulation results demonstrated that the proposed method can outperform the previous one and is more competitive than the agents based on mathematical functions.

In chapter 5, focusing on CDA, GNP with rectify nodes (GNP-RN) has been applied for CDA bidding strategy combined with proposed heuristic rules, which are derived based on the common believes for assisting agent's bidding behavior. GNP-RN is developed aiming to guide the agent to be competitive under different CDA environments, and maximize the agent's profit without losing chances for trading. Rectify Node (RN) is a newly proposed kind of nodes, which is used for bringing more flexible and various options for bidding action choices. 4 groups of simulations are designed to compare GNP-RN with conventional GNP and other strategies in CDA. In each simulation, the kinds of opponent agents are different in order to fully analyze the agents' performance. The simulation results show that the proposed method can outperform all the other strategies and achieve high success rate as well as high profit even when the situation is highly competitive.

In chapter 6, as an extension of GNP-RN, GNP with adjusting parameters (GNP-AP) for developing bidding strategy in large-scale CDAs is proposed and studied. In large-scale CDAs, much more history information can be obtained than small-scale CDAs. In order to enhance the sensitivity for large-scale CDAs and the capability of judging abundant information, the parameters used by GNP-AP decision-making functions are adjusted during the evolution instead of being fixed in GNP-RN. Moreover, the structure of GNP-AP is designed to be more comprehensive that the number of branches of some kinds of nodes is increased to adapt to the complicated environment situations. The simulation results show that GNP-AP can obtain a good guidance for the large-scale CDAs and could be very efficient for the markets.

In chapter 7, after giving the objectives and motivation of each research in this thesis, some conclusions about the proposed algorithms are described based on the simulation results.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Due to the explosive development of the global network structure, besides the usual way to do the business at the retail level, electronic commerce, which offers opportunities to significantly improve the way for businesses interactions between both customers and suppliers [1], is playing an important role increasingly in many organizations and individual consumer's daily life. It is obvious that the trades conducted electrically happen much more frequently, and the amount and categories of the trades have grown extraordinarily. Global retail e-commerce market is expected to reach 1.6 trillion in 2012 [2], and in China, in the first half year of 2011, retail e-commerce reaches 5540 billion dollar, with a year-on-year increase of 74.6%, especially with a growth of 268.8% on holidays [3]. Without doubt, E-commerce is a revolutionary model of the traditional commerce, and brings infinite possible modality for trading goods and services[4; 5; 6].

More and more large scale and decentralized e-commerce mechanisms have emerged in industrial and commercial domains in a wide range. In particular, among all these applications, online auctions, which are flexible pricing mechanisms, are high efficient methods for decentralised resource allocation and make the physical limitations of traditional auctions disappear [7]. Due to the characteristics of the "bargaining" and "negotiation" besides all the other convenience [8], the bidders could buy or sell things with more or less uncertain market value, so the online auction gains their extra popularity in the daily life and attracts globally dispersed users. Some of the well-known auction houses include eBay, Amazon.com, YahooAuction, Priceline, UBid and many others.

Online auctions are able to bring cheaper cost, offer more personalized options, be more convenient over time and space, be more sensitive for price and obtain various kinds of real-time business data which could benefit both seller and buyer side, theoretically. Thus, online auctions become one of the most widely studied and employed negotiation mechanisms today [9].

Traditionally, in most current online auction applications, the traders are generally humans who operate all the behaviors to make transactions. These behaviors may involve observing the auctions, analyzing the auction information, and bidding the suitable price for the items. However, facing the increasingly demanding requirements and complexity of online trading, this kind of manual operation does not reveal the full potential of this new mode of commerce in the e-commerce era. Online auctions could be more efficient, sensitive and personalized. For example, an auction agent could monitor all the auctions provided the desired good, bid among these auctions, and finally win the good with profitable price; or could be able to bid a combination of related items within its budget [10; 11; 12]; or could be able to bid in a sequential auctions for several goods; or consider several attributes for auctions such as different quality and quantity to the same kind of good [13; 14; 15] .Having considered the capability of online auctions, exploring possible types of online auctions and automating the behaviors in online auctions attract high interest in order to relieve the users and be more effective.

Consequently, in many studies, the agent-oriented auction mechanisms, with its emphasis on autonomous actions and flexible interactions [16], arise as an effective and robust model for the dynamic and sensitive commerce environments. In complex settings, it is believed that the agents are likely to be more effective than human traders. Das et al. show that agents can outperform their human counterparters in laboratory experiments [17]. In such agent-oriented auction systems, instead of human negotiations, the agent engage in similar processes to achieve the same end that satisfying its owners' particular objectives [18]. With the goal that trading goods and earning profit, The agent acts flexibly on behalf of its owner and is capable of local decision-making based on the environment information and pre-knowledge about the system [19; 20]. In one auction, one or more seller agents and one or more buyer agents could compete with each other for the opportunity for making a trade at profitable prices. The auction protocols defines the ”rules” for the agents in auctions. They choose bidding strategies and take bidding actions according to the protocols, and the deal occurs between one seller and one buyer when the transaction condition is satisfied [21].

<Auction Types>

There are many different types of single-sided auction [22], such as English auction, Dutch auction, and Vickey (second-price sealed-bid) auction [23; 24; 25]. In English auction, the auctioneer begins with the lowest acceptable price and bidders are free to raise their bids successively until there are no more offers to raise the bid, and the information on how the auction is going on is open to every bidders participated. The winning bidder is the one with the highest bid. Dutch auction is the converse of English one; the auctioneer calls for an initial high price, which is then lowered progressively until there is no offer from a bidder to claim the good. In Vickrey auction, each bidder submits his offer for the good independently without any knowledge of the other bids, where the highest bidder gets the good but he pays a price equal to the second highest bid.

Based on English auction, Multiple round English auctions (MREA) is a newly developed auction type specially in order to meet the requirement that the buyers want to bid in multiple auctions to get a group of desired goods efficiently [26; 27; 28; 29; 30]. In MREA, the buyer agent is able to bid in several rounds of auctions on behalf its owner in order to procure the best deals for the desired various goods.

On the other hand, the most common type of double-sided auction is the continuous double auction (CDA)[31]. CDA is valued as a significant e-commerce market mechanism because it can reflect and reserve the very basis of economy, where the real-time interactions occur between sellers and buyers, thus CDA is a free and highly responsive system, which can exploits the dynamics of the market and balance demand and supply efficiently and expediently [32]. NASDAQ and stock exchange and the major foreign exchanges use variants of CDAs. Other significant applications of CDAs are in market-based control [33], such as pricing decision, allocation of air pollution permits [33], air-conditioning systems [34], resource exchange in smart grid [35; 36] and other complex resource allocation problems[37]. There are variety of different kinds of CDA models have been conducted [38; 39; 40], but all these CDAs' protocols allow traders to make trade at any moment during the auction period once there is a pair of seller and buyer satisfy each other. The auction agents are able to continuously update their bidding prices on behalf buyers and sellers at any time in the trading period for several auction rounds until there is no good needed to be traded.

Both MREA and CDA allow multiple agents to participate and one agent can deal with several auctions continuously and simultaneously, which are effective auction types to allocate resources, save time and relieve the users.

1.2 Contents of this Research

1.2.1 Objective and Motivation

In MREA and CDA, the major challenge for automatic bidding strategies is to improve the degree of automation and optimize the agent's bidding behavior in order to maximize the owner's profits because there is no centralized system-wide control. Concerned about the planning, distributed constraint optimization and scheduling algorithms[41; 42], most of the related researches have been conducted by using heuristic methods and fixed mathematical functions to compute the final optimal bidding price for the items or to compute how much the bid should be done at each time step[26; 43; 44; 45; 46; 47; 48]. Nevertheless, because auction environments are complicated and highly dynamic due to many factors affecting each other, these approaches are not flexible enough for the dynamic environments, and there is no known dominant strategy[31].

Against the background introduced, this thesis is concerned with developing the intelligence of autonomous agent's bidding strategy in order to make the agent to be more efficient and competitive for agent-based online auction mechanisms, especially in Multiple Round English Auctions (MREA), Continuous Double Auction (CDA) and large-scale CDAs. So, the main research aims are as follows:

- to design a more efficient bidding strategy for MREA
- to design more efficient bidding strategies for CDA and large-scale CDA
- to design suitable simulations to properly evaluate and analyze strategies' performance, and to demonstrate the proposed strategies' efficiency

Specifically, in MREAs and CDAs, the objective of the auction agents is to monitor all the auction rounds, and make the right bidding price at each time step to ensure that they can make a trade at profitable prices under their preference. As described, MREA and CDA allow multiple agents compete with one another, which contains lots of interactions. These interactions are influenced by many factors, such as time, current highest price, current lowest price, personal limit price and so on, that is why the auction environment is highly dynamic and complex. So, by using the various available information on the environment, responding quickly and suitably to the changing market at each time step is the essential part of a good bidding strategy. Thus, in order to develop effectiveness and efficiency bidding strategies, we focus on the following four points:

- ensure the bidding strategies' flexibility and adaptability to various auction environments
- enhance the agent's capability of observing and judging the current auction information as well as history information
- design suitable and competitive bidding action options for the agents
- guide the agent to choose the proper bidding action at each time step

1.2.2 Contribution

Given the research aims outlined above, in order to be more flexible and better exploit the market information, Genetic Network Programming (GNP) is firstly employed to the agent's bidding strategy since its applicability and efficiency have been clarified in complex and dynamic problems in many other fields. Since the past studies suggested that the expression ability in each evolutionary computation is potentially linked to the complexity of the application problems, GNP is developed as an extension of Genetic Algorithm (GA)[49] and Genetic Programming (GP)[50], which is one of the evolutionary optimization techniques using compact directed graph structures as solutions [51; 52]. GNP is composed of a number of nodes and each node connects with each other using one or more directed branches, so the nodes are reusable and node transition usually won't end until the task is completed. GNP has been studied extensively in [51] and applied successfully to complex and dynamic problems in many other fields, where the search space is large and not well understood, and also GNP has been proved to be more efficient and effected than GA and GP because of the compact structure and reusable nodes. The applications include Stock Market Prediction [53], Double-Deck Elevator Group Control [54], Network Intrusion Detection [55] and Traffic Volume Forecast [56].

Basically speaking, in the proposed method, the GNP population represents the group of potential bidding strategies. Each individual uses the if/then decision-making function to judge the auction information and guides the agent to take the suitable actions under different situations, which is flexible and adaptive to various auction situations. During the evolution, the GNP structure will be systematically organized, and finally, the individual which can obtain the highest profit is selected as the optimal bidding strategy at the end of training phase.

The main contribution of my research is as follows:

- **GNP is firstly applied to the agent's bidding strategy.** In order to successfully employ the basic GNP algorithm into auction applications, the auction concept is embedded into GNP structure with proper logic. Each GNP individual represents one potential bidding strategy. Different node functions including judgment functions and action functions are proposed depending on the features of the auction, and the nodes transition describes and models the agents' bidding rules under certain auction situations.

- As the foundation of my research, **the capability and effectiveness of GNP for guiding bidding actions are studied and analyze through the simulations in MREA.** The performance of GNP-based agents is evaluated and studied in two situations: MREA with no time limit (NTL), and MREA with time limit (TL). Furthermore, according to the amount of money each agent has, each situation is divided into 2 cases: general case and poorest case. All the participating agents in the simulations use the GNP strategy. The simulation results reveal that the agents using the GNP strategy can understand various environments well through experiences and become smarter through evolution.

- **A bidding strategy for MREA is proposed.** It is able to judge several kinds of information and situations at a time, and also is able to consider the bidder's attitude towards to each good, which makes the strategy to be personalized for each bidder and could make the bidder satisfied with the auction results and profits. The bidding strategy for MREA is studied and compared with the other conventional strategies in the simulations. The simulation results demonstrated that the proposed method is more competitive than the agents based on mathematical functions.

- **GNP with rectify nodes (GNP-RN) has been applied to CDA bidding strategy combined with proposed heuristic rules**, which are derived based on the common believes. GNP-RN is developed aiming to guide the agent to be competitive under different CDA environments, and to maximize the agent's profit without losing chances for trading. **Rectify Nodes (RN) is newly proposed**, which is used for bringing more flexible and various options to bidding action choices. 4 groups of simulations are designed to compare GNP-RN with conventional GNP and other strategies in CDA. In each simulation, the kinds of opponent agents are different in order to fully analyze the agents' performance. The simulation results show that the proposed method can outperform all the other strategies and achieve high success rate as well as high profit even when the situation is highly competitive.

- As an extension of GNP-RN, **GNP with adjusting parameters (GNP-AP) for large-scale CDAs is proposed and studied**. In large-scale CDAs, much more history information can be obtained than small-scale CDAs. In order to enhance the sensitivity for large-scale CDAs and the capability of judging abundant information, the parameters used for GNP-AP decision-making functions are adjusted during the evolution instead of being fixed in GNP-RN. Moreover, the structure of GNP-AP is designed comprehensively so that the number of branches of some kinds of nodes is increased to adapt to the complicated environment situations. The simulation results show that GNP-AP can obtain a good guidance for the large-scale CDAs and could be very efficient for the markets.

1.3 Thesis Structure

In this thesis, there are four research topics to be studied based on the aforementioned objectives.

In Chapter 2, we introduced the concept of MREA and CDA in detail, which is mainly studied in this thesis. The auction protocol, which is a set of interaction rules and a set of clearing and pricing rules, is described according to MREA and CDA, respectively. The related researches are also introduced.

In Chapter 3, bidding strategy for the auction agents using GNP in MREA is proposed. This chapter aims to study and analyze the capability and effectiveness of GNP for guiding bidding actions through the simulations. The simulation results reveal that the agents using GNP strategy can understand various environments well through experiences and become smarter through evolution.

In Chapter 4, in order to improving the agent's intelligence and sensitivity, an enhanced bidding strategy for MREA using GNP is developed. The proposed strategy is compared with the previous GNP strategy and the other conventional strategies in the simulations.

In Chapter 5, GNP with rectify nodes (GNP-RN), which has been applied to CDA bidding strategy combined with proposed heuristic rules, is described. The simulation results show that the proposed method can outperform all the other strategies and achieve high success rate as well as high profit even when the situation is highly competitive.

In Chapter 6, as an extension of GNP-RN, the bidding strategy using GNP with adjusting parameters (GNP-AP) for large-scale CDAs is proposed and studied. The simulation results show that GNP-AP can obtain a good guidance for the large-scale CDAs and could be very efficient for the markets.

In Chapter 7, after giving the objectives and motivation of each research in this thesis, some conclusions about the proposed algorithms are described based on the simulation results.

Chapter 2

Auctions to be studied

Online auction is one of the most popular types of price mechanisms that allow selfish and profit-motivated agents to buy and/or sell resources. One or more seller agents and one or more buyer agents compete with each other for the opportunity for making a trade at profitable prices [5; 57].

Each type of online auction is determined by the market protocol which is a set of interaction rules and a set of clearing and pricing rules. The interaction rules define how participants interact through their actions. There are usually many interaction rules in one kind of auction, ranging from specifying whether a trader can be a buyer and seller to specifying that a bid or an ask that can be submitted in the market must have a particular format. The clearing and pricing rules only determine when and at what price a transaction occurs [58].

In this chapter, first of all, the basic general preliminaries of auctions are introduced. Then, the specific concept and protocol of MREA and CDA used in our research are described in detail, respectively. The related researches are also introduced.

2.1 Basic preliminaries

The basic preliminaries for online auctions are introduced in this section. There are agents (sellers) submitting *asks* to sell goods and agents (buyers) submitting *bids* to buy goods in online auctions. Usually, each auction has its own pre-assigned length of time steps, and there is only one single good to trade at any time step during one auction.

- *basic price*(P^B): P^B is the starting price for the agent toward to the ongoing good in the auction.

- one *time step* is the time period when each agent submits a bid(/ask) to the auction till the auction updates the current highest bid (/lowest ask). After the updating, the next time step starts.

- *private price*(P^P): Each bidder has a distinct private limit price (P^P) for each good he wants to trade. For a buyer agent, P^P means the highest value it is willing to pay for the good. If it submits the bid at a higher price than P^P, it will lose profit. For a seller agent, P^P means the lowest value it is willing to sell for the good. If it submits the ask at a lower price than P^P, it will lose profit.

- V_{step}: The smallest valid bidding step is denoted as V_{step}.

- *outstanding ask*(oa): oa is the current lowest ask in the market. Any following ask not lower than the current $oa - V_{step}$ is invalid and discarded by the auction. Sellers usually submit initial asks at a high price and decrease oa gradually.

- *outstanding bid*(ob): ob is the current highest bid in the market. Any following bid not higher than the current $ob + V_{step}$ is invalid and discarded by the auction. Buyers usually submit initial bids at a low price and increase ob gradually.

- *final price*(P^F): P^F is the final traded price for the ongoing good.

- one *auction round* is the time period of the bidding process for one good, which is from the beginning of the auction for the good until the transaction for the good to take place or there is no new asks or bids submitted in a pre-determined time. In one single auction round, only one good can be traded. Only one seller can sell it and only one buyer can get it from that seller. When the ongoing good is traded or there is no new asks or bids submitted in a pre-determined time, the current round ends and the next round starts.

- one *whole auction process* could contain one or more auction rounds, which means the time period from the beginning of the first auction round until the end of the last auction round.

In summary, in one auction round, each agent participating in the current auction submits a bid (/ask) to the auction at each time step of the auction according to his bidding strategy, and the auction chooses the highest bid or lowest ask to update the *ob* or *oa*, then the next step starts. The process of each time step will be repeated until the auction ends, and the buyer who submits the highest bid wins the good or the seller who submits the lowest ask sells the good. If there are multiple round auctions, the next auction round won't start unless the previous one is completed.

2.2 Multiple Round English Auction (MREA)

As introduced, in recent years, online auctions attract globally dispersed users and become one of the most popular and effective ways for trading goods over the Internet. The number of on-line auction systems keeps increasing, and more kinds of goods are traded on the auctions. Consequently, facing the fact that some bidders start to try to bid several different goods during a period of time for convenience and efficiency, multiple round auctions is emerged in the study.

2.2.1 Overview of MREA

Besides the basic preliminaries introduced in section 2.1, this section introduces the specific concept of MREA and gives a description of the protocol used in this thesis.

Multiple round auctions, which is a common auction to deal with multiple goods, usually consists of a number of auction rounds that run consecutively or concurrently [45]. This thesis only considers the English auction protocols, because English auction is the most common auction type.

MREA allows multiple buyer agents submitting bids to buy one or more goods during the trading period. Each agent has its own private price for each good it wants. In other words, the different agents have different private prices for the same good.

Each auction round is carried out for only one good. So, as shown in Fig.2.1, in one MREA process, there are usually several goods waiting to be traded, which leads to several auction rounds. Because each auction round is independent from each other, so the auctions are considered to be carried out good by good orderly in this thesis. Every agent will only attend the auctions for the goods it wants. If the agent doesn't want to bid for the ongoing good of the auction, then it won't attend the auction.

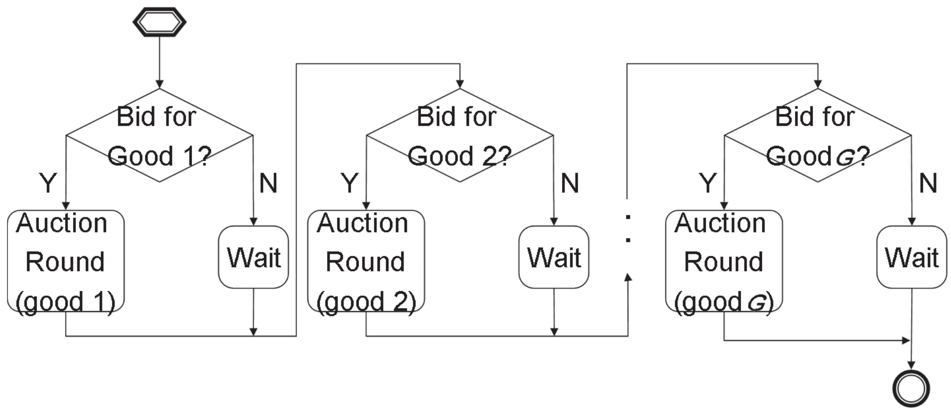


Figure 2.1: MREA process for one agent

In detail, for example, as shown in Fig.2.1, the process starts from the auction for good 1. All the agents who want to trade good 1 will attend the auction. Meanwhile, the other agents who don't want the current good won't attend the auction and wait to bid for the next round. When auction for good 1 ends, the auction for good 2 will start. Then, the agents who want to trade good 2 will attend the auction. This procedure will repeat until the last auction round. When the last round ends, one MREA process is finished.

As Fig.2.2 shows, in one auction round, if the current auction round is closed, then, no one can buy the good on the auction. If the auction round is not closed, the buyers submit bids not smaller than the current *ob* at each time step during the auction round period by judging the information of the current auction environment. Otherwise, the bids are invalid. If the agent has a wait-and-see attitude and doesn't want to submit a new bid price at the current time step, then the agent is regarded as submitting the same bid price as the current price. After all the agents submit their new bids, the auction takes the highest bid to update *ob*. Then, the auction turns to the next time step. If the agent quits from the bidding in the middle of an auction, it won't submit any bid price to that auction, but it can still attend the following auction round which it wants to bid in.

Moreover, in order to better study the bidding strategies, two kinds of MREA time situations are considered in this thesis: No time limit and Time limit. No time limit situation means that for each auction, there is no pre-assigned length of time steps,

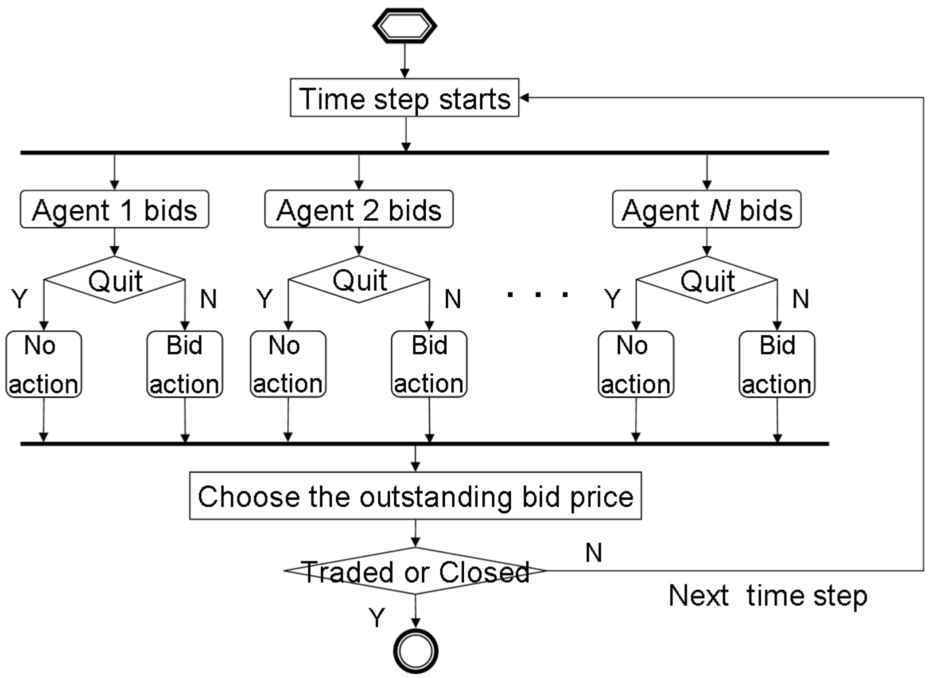


Figure 2.2: Auction round for one agent

which is like the infinite length of time steps, so an auction ends when no one is willing to submit a new price for a pre-specified time (i.e., continuous 3 time steps). Time limit situation means that each auction has the given length of time steps, so an auction ends when it reaches the closing time step.

2.2.2 Related works of MREA

There have been several attempts to design sophisticated and efficient bidding strategies for agents participating in online auctions. For example, Faratin [29] is broadly similar to the mechanism defined in this article. However, there are several important differences between one-to-one negotiations and multiple auctions. Chief amongst these, are the type of the tactics that are considered relevant.

An extension of Faratins model is given by Matos, Sierra and Jennings [59] who analyzed the evolution of the negotiation strategies using GAs, and determined which of them are appropriate in which situations. The aim of this work was to perform an evaluation of the range of negotiation strategies by analyzing their relative success, and to study how these strategies evolve over time to become a fitter population. The main difference from my work is in the domain dealing with (multiple auctions versus bilateral negotiations).

BiddingBot is a multi-agent system that supports users in attending, monitoring and bidding in multiple auctions through a process called co-operative bidding [30]. This approach demonstrates how agents can cooperate and work together to do the bidding process in multiple auctions. It consists of one leader and several bidder agents, where the leader agent acts as the coordinator and the facilitator of the whole bidding process. Bidding is done by exchanging messages between the users, the leader agent and the bidder agents. However, the main problem with this approach is that the agents do not actually make the bidding decision. This decision is left to the users. Thus, the agents do not have full autonomy and the decision-making process is slow since the agent needs to interact with the users from time to time.

Preist[27] proposed an algorithm for agents that participate in multiple simultaneous English auctions. The algorithm proposes a coordination mechanism to be used in an environment where all the auctions terminate simultaneously, and a learning method to tackle auctions that terminate at different times. Byde [60]also considers this environment, but utilizes stochastic dynamic programming to derive formal methods for the optimal algorithm specification that can be used by an agent when participating in simultaneous auctions for a single private-value good. Both of these works are designed specifically for purchasing items in the multiple English auctions and their algorithms are not applicable in a heterogeneous protocol context. Byde[60] presented another decision theoretic framework that an autonomous agent can use to bid effectively across multiple auctions with various protocols (English, Dutch, first price sealed bid and Vickrey auctions). In order to come up with the best bid value that guarantees the delivery of the item, an agent must always speculate about future events. To do this, Byde presented an approximation function that provides an estimate of the expected utility of participating in the set of future auctions.

In order to study the flexibility and efficiency of bidding strategies, some handcraft strategies based on P. Anthony and N. R. Jennings' research [26; 45] are used for comparing with the proposed MREA bidding strategy in chapter 3, which are remarkable in the multiple round auction research and the basis of many other researches [60; 61; 62]. Although these strategies are heuristic methods using mathematical functions, but they are the same as GNP-based strategy in terms that they help the agent to make bid decisions at every time step and collect the current auction information, such as the current bid price for the good and the total number of time steps of the current auction round to make bid decisions, only except that they use limited auction information. So, it is reasonable to compare the proposed GNP-based strategy with these non GNP-based strategies in the MREA.

The strategies are composed by a combination of *The Desperateness Tactic (DE)*, *The Desire for Bargain Tactic (DB)* and *The Remaining Time Tactic (RT)*. These tactics indicates different bidding policies depending on the agents' intentions. *The Desperateness Tactic* implies that gaining the good is most important, while *The Desire for Bargain Tactic* implies that the agent prefers to earn a bargain with some profits. *The Remaining Time Tactic* implies that besides the attitudes, the time factor should be also considered during the bidding process.

According to the agents' three different attitudes to the goods, the non-GNP-based agents consider the bidding tactic *(DE and DB)* or combine it with *The Remaining Time Tactic*. If the agent's attitude toward to the good is *desperate for the good*, then the agent will use *DE* or the combination tactic of *DE and RT*, which means that the agent also considers the time factor's effect; if the attitude is *both desperate for the good and looking for a bargain*, then the agent could use the tactic which is a combination of *DE and DB* or use the tactic which is a combination of *DE, DB and RT* when the agent considers the time factor. In sum, there are two bidding policies for the agent to choose: only considering the attitude or also considering the time factor. The agent decides which policy to use randomly. If the agent uses the combination bidding tactics, a weight ω is allocated to each of them to denote their relative [44].

The functions employed by the tactics are parameterized by two key values, k and β. In all of the tactics, k is a constant that determines the value of the starting bid price and β defines the shape of the bidding curve, so different values of k and β can reflect the different attitudes of agent as shown in the following .

The non-GNP-based agents calculate their bid prices at each time step t by using their tactics mentioned above. The tactics are expressed in more details as follows:

The Desperateness Tactic: The agent bids b_{de} close to P^P at $t = 0$ by setting k_{de} at 1.0.

$$b_{de} = t/T * P + \alpha_{de}(t)(P^P - t/T * P), \tag{2.1}$$

$$where, \alpha_{de}(t) = k_{de} + (1 - k_{de})(t/T)^{1/\beta_{de}}, \tag{2.2}$$

The Desire for Bargain Tactic: The agent keeps his bid price b_{db} to a minimum as the auction goes on from $t = 0$ to T by setting k_{db} and β_{db} appropriately.

$$b_{db} = t/T * P + \alpha_{db}(t)(P^P - t/T * P), \tag{2.3}$$

$$where, \alpha_{db}(t) = k_{db} + (1 - k_{db})(t/T)^{1/\beta_{db}}, \tag{2.4}$$

The Remaining Time Tactic: The agent bids b_{rt} closer to P^P as t approaches T.

$$b_{rt} = \alpha_{rt}(t) * P^P, \tag{2.5}$$

$$where, \alpha_{rt}(t) = k_{rt} + (1 - k_{rt})(t/T)^{1/\beta_{rt}}, \tag{2.6}$$

The Combination Tactic: Consider both effects of attitude and time.

$$b_{cb} = \omega_{db} * b_{db} + \omega_{de} * b_{de} + \omega_{rt} * b_{rt}, \tag{2.7}$$

where,

- b: calculated bid price that the agent should bid at the current time step
- t: the current time step
- T: the total number of the time steps
- P: the current bid price for the good
- P^P: private price of the good
- k and β: control parameters
- ω: weight for each tactic

Each tactic can present the relevant attitude by setting k and β at different values. As an example, when $k \geq 0.5$ and $\beta \geq 1$, the agent demonstrates a reasonable degree of desperateness and starts the bidding from a value close to P^P and reaches P^P quickly. Actually, *The Desperateness Tactic* is employed when the agent is desperate to get the item, and the value of k_{de} is set at a high value since a desperate agent starts the bidding from a value near to P^P.

2.3 Continuous Double Auction (CDA)

Continuous double auction (CDA), where multiple sellers and buyers updates their bidding prices continuously to trade goods or services, is the most common forms of marketplaces due to its operational simplicity and efficiency in both of the two trading sides [31]. In CDA, there are usually several goods to trade, and the trade for one good can be made throughout the trading period once an ask and a bid are satisfied each other, where the trade is executed between the buyer who submits the bid and the seller who submits the ask.

CDA has emerged as the dominant financial institution, and provides a dynamic and efficient approach to the decentralized allocation of resources in the commerce market. CDA is valued as a significant e-commerce market mechanism because it can reflect and reserve the very basis of economy, where the real-time interactions occur between sellers and buyers, thus it is a free and highly responsive system, which can exploits the dynamics of the market and balance the demand and supply efficiently and expediently. The majors exchanges like the NASDAQ, the New York Stock Exchange and the concept of trading power in Smart Grid use variants of the CDA institution [63]. These examples of the CDA are highly domain specific and difficult to generalize. Thus, most research in this area has generally been structured around the market protocol initially proposed by Smith [64].

2.3.1 Overview of CDA

Besides the basic preliminaries introduced in section 2.1, this section introduces the specific concept of CDA and gives a description of the protocol used in this thesis.

One CDA environment can be described as:

$$E_{CDA} =< g, S, B, V_{step} >$$

where,

- g is the kind of goods to be traded in the CDAs.
- $S = \{s_1, s_2, ..., s_M\}$ is the finite set of identifiers of sellers, where M is the number of sellers. Each seller s_i has an attributes set that $\{c_{i1}, c_{i2}, ..., c_{iA}\}$, where c_{in} is the private limit price for the nth good g_n he wants to sell and A is the number of goods he wants to sell.
- $B = \{b_1, b_2, ..., b_N\}$ is the finite set of identifiers of buyers, where N is the number of buyers. Each buyer b_i has an attributes set that $\{v_{i1}, v_{i2}, ..., v_{iB}\}$, where v_{in} is the private limit price for the nth good g_n he wants to sell and B is the number of goods he wants to buy.
- V_{step} is the smallest valid bidding step

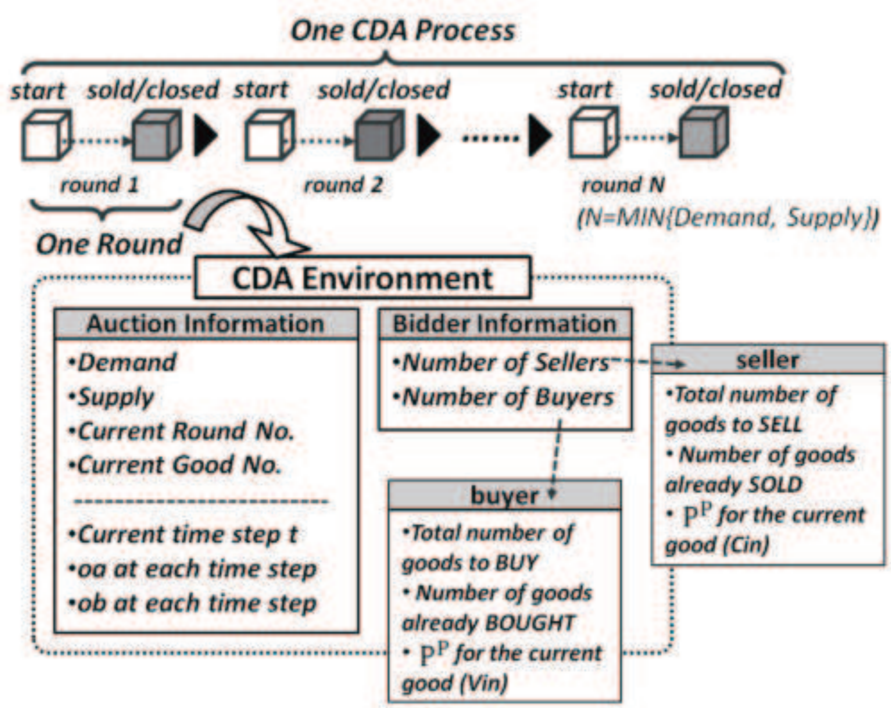


Figure 2.3: Information attributes of CDA

As implied by the name, in CDA as shown in Fig.2.3, usually goods of one kind are considered and there are more than two goods to trade good by good in the market, so one CDA process usually contains several auction rounds, and the succeeding auctions can refer to the information of the previous successful auctions due to the similarity of the goods. Besides, there is only one single good to trade at any time step, both sides of the bidders, i.e., sellers (s) and buyers (b), exist and the numbers of bidders on each side are greater than three. Toward each goods, each bidder has distinct private limit price(P^P). At time step t of round r, every seller submits ask and every buyer submits *bid* to bid for the goods. ask a^{it} is the bidding price at which the seller i is willing to sell the good at time step t. *bid* b^{it} is the bidding price at which the buyer i is willing to buy the good at time step t. The auction chooses the highest bid and lowest ask to update the current ob and oa, then the next time step starts. The process repeats until $ob \geq oa$, then the transaction occurs between the seller who submitted the oa and the buyer who submitted the ob. CDA terminates when either side of the bidders (seller side or buyer side) has no good to trade any more [47].

Additionally, in CDA, the total number of goods that all the seller want to sell is denoted as $supply$, and the total number of goods that all the buyer want to buy is denoted as *demand*. Fig. 2.3 shows the available information attributes of a CDA environment. As there are many uncertainties and many factors can influence the auctions, the CDA environments are complex and dynamic [46].

2.3.2 Related works

There are many related research about automatic bidding strategies, which are designed for guiding agents' buying and selling behavior in CDAs, such as ZI-U, ZI-C, ZIP, CP, GD, strategies developed by GP, fuzzy-based strategy, strategy using reinforcement learning and so on. [46; 47; 48].

Zero Intelligence (ZI) is proposed by Gode and Sunder [47], which includes ZI-U traders and ZI-C traders. These two traders are considered as randomly bidding traders. Their bidding prices are distributed independently and uniformly over the entire range of trading prices. ZI-U traders make bid decisions at a random price in the valid range of the market without considering the traders' private limit prices. ZI-C traders also make bid decisions at a random price, but consider their private limit prices. So, ZI-C and ZI-U traders can trade frequently, because they bid at random price without trying to make some profit margin, which may lead to the situation where they will accept the random price offered by the traders on the opposite side even if the price is unreasonable, that is why ZI-U traders can often get negative profit.

Cliff and Bruten [65] developed the Zero Intelligence Plus (ZIP)strategy. Different from the ZI-C and ZI-D agent, each ZI-P agent has a profit margin, which means the difference between its private limit price and the bidding price to be submitted. The profit margin is updated following some common believes: if there was a transaction at the last auction and the winner is the ZIP agent itself, the agent would increase the profit margin for aiming to gain more profit; if there was no transaction at the last auction or there was a transaction but the winner is not the ZIP agent itself, the agent would decrease the profit margin in order to increase the chance to trade.

Chris Presit [48] built up the CP strategy, which is based on the ZIP strategy. Besides the basic idea of ZIP strategy, CP strategy consists of a small number of heuristics and a learning rule. The CP agent does not jump to the target price directly, but moves toward to the target price with a learning rate little by little. CP and ZIP focus on adaptability only using the last auction information.

GD strategy is conducted by Gjerstand and Dickhaut [46]. GD agents memorize all the asks(bids) in the bidding history of the last several auctions. Using these information, GD agents compute the probability of an ask(bid) being accepted. Then, the probability is multiplied by the theoretical profits, which can give the expected utility of this ask(bid). For example, a GD buyer agent submits a bid b, which maximizes $\pi_b(v-b)$, where π_b is the belief function of a bid that is accepted, and v is the valuation of the good. The ask(bid) with the highest expected utility will be submitted by the agent. GD strategy is a highly history information based strategy.

Tesauro and Das [66] and Tesauro and Bredin [67] proposed some improvements to the GD algorithm. In order to solve excessive volatility of the original GD algorithm, the highest and the lowest transaction prices ever happened in the history are recorded. Also, the forecast of the changes of beliefs are used together with the basic belief function of GD. ZI-C, ZI-U and GD pay no attention to adaptability.

GP also could be used in resource allocation problem, such as auction based scheduling [68]. In CDA, strategies developed by GP are also conducted. In Phelps's method [69], GP is used to optimise pricing rules for agents. However, only bid price and ask price are considered in terminal set, which is limited information, and functions are composed of standard arithmetic functions. Richter [70] also uses GP to develop bidding rules, but use more information such as max/min/average ask/bid price in terminal set. Also, in his GP-automaton, there are 4 states for various action choices combined with GP-decider part to determine the bidding price. It seems to be more intelligence than the previous two, but their rules are basically equations rather than logic rules.

Other strategies have also been developed. FL strategy which use fuzzy logic and a group of updating rules to decide the value for a bid or ask to submit [71] (Fuzzy constraint-based framework has also been used to guide the agent's behavior [59; 72; 73; 74] in bilateral negotiation [75], which is usually concerned with multiattribute contracts, such as quality and delivery date, instead of the usual auction form that N to N negotiation[7]). The modified Roth-Frev strategy is based on reinforcement learning algorithm which encourage the agent to submit more profitable bidding price for a special type of CDA where buyers can reselling the goods they bought, and the updating rules are for auction rounds [76]. (Reinforcement learning also has been used to explore and analyze bidding patterns in auctions.[77])

All of the above strategies are proposed for CDAs. However, we will only consider ZI-C, ZI-U, CP and GP when benchmarking GNP strategy. This is because they are the most widely used benchmarking strategies in the literature, and uses these 4 different kinds of strategies together as opponents can mostly generalize the dynamic auction situations.

Chapter 3

Bidding Strategy Acquisition for MREA using GNP

3.1 Introduction

In this chapter, the bidding strategy for Multiple Round English Auction (MREA) based on Genetic Network Programming (GNP) is proposed and studied [78; 79].

3.1.1 Motivation

As more kinds of goods being traded on the auctions, some bidders start to try to bid several different desired goods during a period of time for convenience and efficiency. Consequently, MREA is emerged as a new auction form to deal with multiple goods [26; 43; 45]. If the bidder chooses to participate in the auctions himself, maybe this way is not really easy. Attending more than one auction in order to get different goods, not only makes the bidding strategy complex, but also make the bidders get tired easily, and the whole process is time consuming, usually several days. So, facing the attractive auctions of multifarious kinds of goods, the customers also face the problems such as how to handle various auctions for different kinds of goods in a period of time and how to bid the appropriate price at each time step to ensure that they can get the desired goods within their money limitation. As a result, the importance of the agent technology which can assist customers in the problems is increasing in these years. On the other hand, because multiple round auctions' environments are complicated and dynamic, and many factors should be considered, there are plenty of potential bidding strategies for the agent and it is hard to well understand them.

So, in order to make online auctions more efficient and more intelligent, this chapter develops and analyzes a bidding strategy based on Genetic Network Programming (GNP), which can help customers to deal with multiple round auctions automatically.

3.1.2 Major points

- The bidding strategy for Multiple Round English Auction (MREA) based on Genetic Network Programming (GNP) is proposed. It uses its directed graph structures to collect and judge the information on the ongoing auctions, then makes bid decisions according to the judgment results. With the evolutionary features, it is allowed that the agent can find the general optimal strategy from a large numbers of potential ones generation by generation.

- 25 kinds of judgment nodes and 4 kinds of processing nodes are proposed for GNP structure used by MREA bidding strategy.

- In order to study the effectiveness and capability of the proposed method for guiding auction agents, and study the adaptiveness of the proposed method to various situations, two kinds of MREA are considered: NTL MREA and TLM MREA. Further more, for each kind, two cases are studied: general case and poorest case.

Several conclusions have been obtained from simulations both in the no time limit model and time limit model. For example, firstly, the agent made by GNP can find appropriate bidding strategies for various situations, where the number of goods and the set of goods each agent wants are changed. Secondly, the agent could become smarter through evolution when he competes with other agents. Thirdly, poor agents can also get goods by evolution, although the number of goods obtained is a bit smaller than rich agents.

In section 3.2, the GNP structure and specific kinds of judgment nodes (JNs) and processing nodes (PNs) are described. Section 3.3 gives the simulations, and the results are analyzed. Section 3.4 concludes this chapter.

3.2 Bidding strategy for MREA using GNP

3.2.1 GNP structre

The strategy developed by GNP helps the agent to decide how much to bid at every time step. For more specific, the population is composed of many GNP individuals representing bidding strategies. According to the auction conditions and limitations introduced in section 2.2, we consider the useful kinds of judgment functions to judge the relevant environment's information from different aspects of the auction, such as the current time step, the current bid price, other agents' behaviors, bidding action history and so on. These kinds of judgment functions are assigned into different kinds of JNs, and the potential bid actions are assigned into different kinds of PNs. As Fig. 3.1 shows, when an auction round starts, the node transition starts from the start node, JNs judge the current auction situation and use the judgment result to take the corresponding branch to move to the next node until reaching a PN, and the bid action assigned to this PN decides how much the agent should bid at the current time step. When the next time step starts, the transition continues from the last PN and pauses when it moves to another PN. The process iterates until the auction round ends. All the individual will do the above process and be evaluated by the fitness function. The selected better ones will undergo the mutation and crossover to change their GNP structures for generating offspring for the next generation. After evolving for enough generations, the final optimal individual can be obtained, which has a well organized GNP structure to deal with the various situations and changes of the auction environments.

3.2.2 Kinds of nodes

Kinds of Processing Nodes (PN):

- PN1: Make a large bid, i.e., add 10 to the Current Bid Price.
- PN2: Make a small bid, i.e., add 1 to the Current Bid Price.
- PN3: Choose to stay, i.e., take no action.
- PN4: Quit from the current auction.

Kinds of Judgment Nodes (JN):

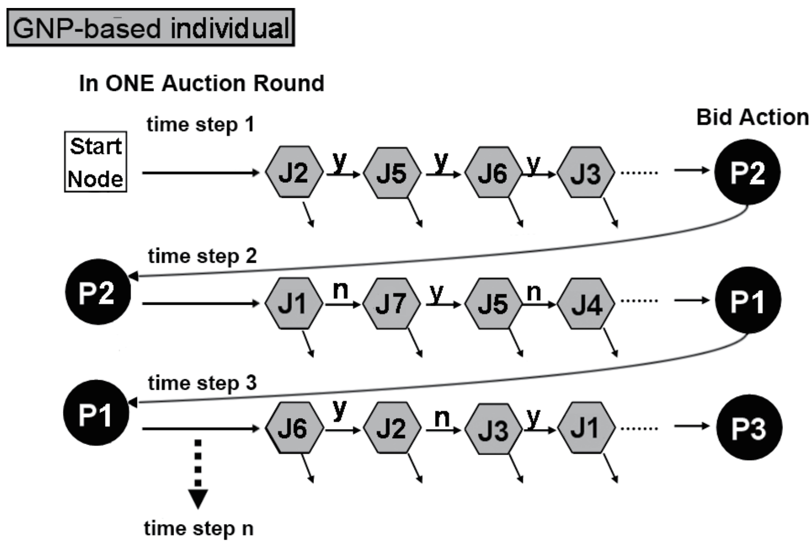


Figure 3.1: Structure of GNP individual

- JN1: The agent himself is the last bidder?
- JN2/3/4: The bid action of the agent in the last time step is large/small/no bid?
- JN5: Every agent stayed for 2 time steps?
- JN6: The current price is larger than the agent's private price? If Yes, then the agent quit quickly.
- JN7/8/9: The good is the first/second/third good the agent wants? i.e., if the agent wants good 1, good 3 and good 4, then, the second good the agent wants is good 3.
- JN10/11/12: The bid is stay action after the last large/small/stay bid action?
- JN13/14/15: The bid is large bid action after the last large/small/stay bid action?
- JN16/17/18: The current time step is in the first third/second third/last third steps of the whole given time steps?
- JN19: The current time step is the closing time step?

- JN20: No one bids for 5 successive time steps?
- JN21: The current bid price is smaller than 1/3 of the agent's private price?
- JN22: The current bid price is larger than 2/3 and smaller than 4/3 of the agent's private price?
- JN23: The current bid price is larger than 4/3 of the agent's private price?
- JN24: In the latest bidding history, the bids were offered for 5 successive time steps?
- JN25: In the latest bidding history, the bids were offered for larger than 5 successive time steps?

Where, JN10-JN15 are only used in the no time limit model. Because no one can know when the auction will end and what the others are thinking about, each bid action might have a big effect on the bidders' psychologies in the no time limit auction, which means that the current agent may have totally different bidding reaction in the next time step depending on the other agents' different bids. Each agent uses these judgment nodes to judge the current circumstance through studying the bid action history.

For the same reason explained above, JN16-JN20 are only used in the time limit model.

3.2.3 Fitness function

The fitness function of GNP individuals of agent i is defined as follows:

$$Fitness(i) = \sum_{g \in G_i} (P_g^P(i) - P_g^F(i) + 0.5P_g^C), \tag{3.1}$$

where,

- G_i: set of the suffixes of the goods agent i wanted and gained.
- $P_g^P(i)$: private price of good g by agent i.
- $P_g^F(i)$: final buy price of good g by agent i.

- P_g^C: common price of good g.

Fitness(i) is the price difference between Private Price $P_g^P(i)$ and Final Buy Price $P_g^F(i)$, which also considers a positive additional term, i.e., the common price of the good, which indicates that the goods the agent wants were obtained. So it is easy to see that the smaller the final buy price is, the higher the fitness is. Also, the fitness function encourages the agents to buy as many goods as they can, even if they need to pay a higher price, but if the final buy price is too high, the fitness may be negative.

3.3 Simulations

The two models, No Time Limit Model (NTLM) and Time Limit Model (TLM), were studied in this chapter. In this section, the basic evolutionary ability of the GNP-based agent is studied and it is studied how the GNP-based agent would behave under various situations. And also, the agent's performances are analyzed.

Both of the NTLM and the TLM have 4 cases of simulations, where 2 for training and 2 for testing.

3.3.1 Simulation settings

First of all, the simulations in both models have the same initialization, i.e., 10 goods and 7 agents. 50 different environments are used for training GNP individuals, where the common prices of different goods are different.

In the initialization of the simulations, the good number each agent wants to bid and the goods' common prices are randomly generated, and depending on the common price, the private price of each good is also randomly generated. All these values are fixed during the whole bid. In more details, the common prices of the goods are set from 100 to 500 in the simulations. Each agent's private price of the good is set at between 70% and 130% of its common price. So, there won't be a big difference among the agents' private prices for the same good. For example, if the common price of good g is 100, then each agent's private price for good g is distributed in the range from 70 to 130. All the prices are randomly generated for each environment. Table. 3.1 is an example of the setting of prices.

Table 3.1: Setting of prices

Good No.	1	2	3	4	5	6	7	8	9	10
P_g^C	201	379	434	182	294	146	360	202	452	433
agent 1		379	490		270	150	406	193	452	
agent 2	237	341			346	140	298		483	
agent 3			468	152	282			232	402	
agent 4		397		149	255	141		208	510	
agent 5	207	379	386	147	279		342		515	
agent 6							396			
agent 7			481			167			452	

In Table. 3.1, P_g^C means the common price of good g, and the goods numbers agent 1 wants are 2, 3, 5, 6, 7, 8, 9 and his private price for good 2 is 379, and there is no agent who wants to buy good 10 in this case.

All the agents have all kinds of JNs and evolve for various number of generations, except only agent 1 continues to evolve until 1000th generation in the evolution phase. To be more clear, as Fig. 3.2 shows, the auction process starts from the first generation for evolution. In each generation, firstly all the GNP individuals of the agent 1 do the auction procedure for all goods he wants, after that, the population of agent 1 is evolved with other populations being fixed. Then, the agent 2 does the same auction procedure for evolution. This procedure is called overall auction process. When all the agents carry out the overall auction process, the next generation begins.

In the testing (generalization) phase, the best individuals of each agent in the training phase compete with each other in 10 new different environments, which are different from 50 environments in the training phase. Here, in the new environments, the common prices are distributed between $P_g^C - 50$ and $P_g^C + 50$, where P_g^C is the good g's common price in the training phase.

The parameters used in the simulation are shown in Table. 3.2. And, all the results are the average results over different environments.

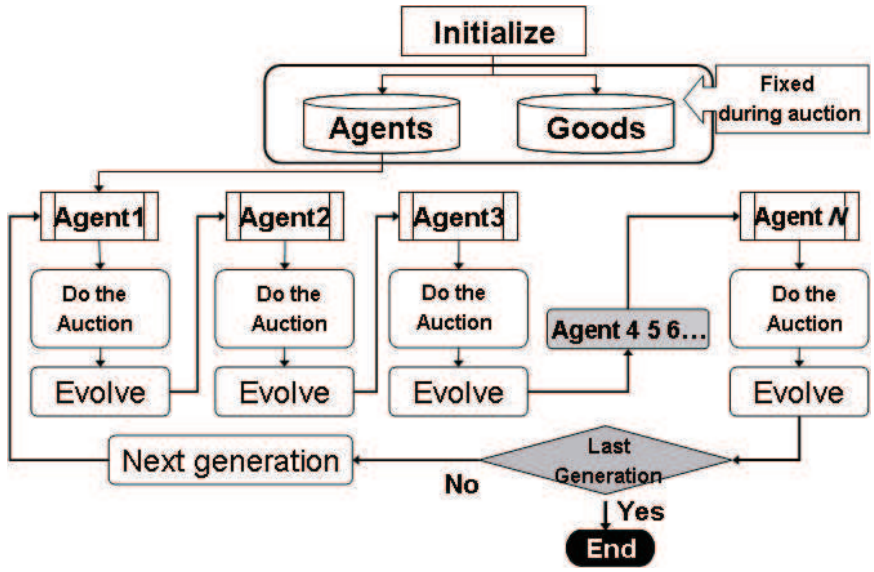


Figure 3.2: Evolutionary structure

3.3.2 No Time Limit Model (NTLM)

3.3.2.1 Training Simulations

Agent 1 and the other agents will do the co-evolution procedure for various numbers of generations in the training phase, then the other agents will use their best individuals to compete with agent 1, while agent 1 continues evolving to 1000 generations to try to find the individual with the optimal strategy.

Because all of the agents use the same strategy using GNP, if all of them evolve for the same number of generations, they will get their optimal strategies which have almost the same performances.

From the above, we can't analyze the evolvability of GNP-based auction model. That is why we did the following simulations, where other agents evolve for 1, 5, 10 and 100 generations, while agent 1 evolves to 1000 generations.

⟨*General Case*⟩

The fitness values, final buy prices and the number of goods obtained are studied when the number of generations of other agents is changed in the general case where the private prices are set randomly.

Table 3.2: Parameters Setting

Goods Number	10
Goods Price	100-500 (randomly set)
Agents Number	7
Population Size	200
Selection Rate	0.3
Crossover Rate	0.1
Mutation Rate	0.3
Elite Keeping Number	10
Offspring by Crossover	80
Offspring by Mutation	80
Offspring Randomly Generated	30
Number of Processing Nodes	15
Number of Judgment Nodes	75

Table. 3.3 shows the simulation results under the various numbers of generations of other agents. Here L means the number of goods agent 1 wants. *Buy Price*, which indicates the agent ability of earning a profit, shows the average ratio of the final buy prices of the goods agent 1 bought to their private prices. $Number\ of\ Goods$, which indicates the agent ability of success in obtaining the goods, means the average number of goods agent 1 bought. $Fitness$ value means the combination of the two above factors.

From Table. 3.3, we can see that the fitness value of agent 1 becomes lower when the number of generations of other agents is larger, and also the average buy price of agent 1 becomes higher. This is because agent 1 wants to buy goods as many as possible, at last, he should pay more than 100% of the private prices. In addition, agent 1 gets fewer goods if the other agents evolve for larger generations.

Fig. 3.3 shows the average fitness values of the best individual of agent 1 over 50 environments until 1000th generation. The result was obtained under the condition that the other agents evolve for 100 generations, which means the other agents gained smart strategies through evolution.

From Fig. 3.3, it can be seen that GNP can help agent 1 to get higher fitness through evolution, which means even when the other agents are much smarter than they were in the initialization, agent 1 still can find his better strategy. However, it is found from

Table 3.3: Study on Fitness, Buy Prices and Number of Goods (NTLM Training, General Case)

	Generations	1	5	10	100
L=3	Fitness	1167	876	683	124
	Buy Price	23%	76%	90%	117%
	Number of Goods	2.8	2.2	2.1	0.9
L=4	Fitness	986	788	655	71
	Buy Price	47%	64%	96%	125%
	Number of Goods	3.8	2.8	2.1	1.0
L=7	Fitness	2449	993	795	220
	Buy Price	21%	59%	90%	104%
	Number of Goods	6.7	3.4	3.1	1.9

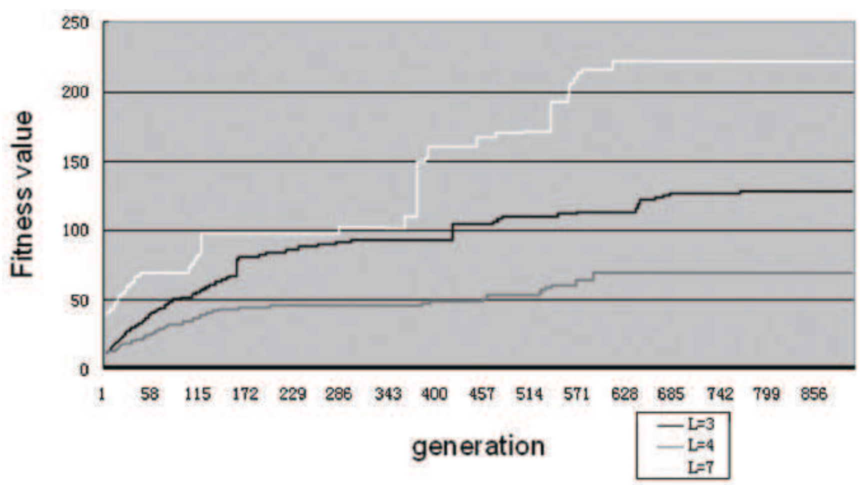


Figure 3.3: Fitness value of agent 1 in the case of NTLM

Table. 3.3 that the fitness values are less than the ones when the others are dull. In other words, when the other agents evolve for larger generations, then they can get smarter strategies and agent 1 can't perform well compared when they are dull.

Additionally, in Table. 3.3, the fitness value in the case of L=4 is quite lower compared to the other two cases, which is because that the fintness values depend on the two factors, i.e., *Buy Price* and *Number of Goods* as well as the different environments randomly generated simulation by simulation.

In the simulations, L=3, L=4 and L=7 are totally different cases. The goods agent 1 wants when L=3 are different from the goods agent 1 wants when L=4. In Table. 3, in the case that the number of generations of other agents is equal to one, agent 1 can get 2.8 goods when L=3 and get 3.8 goods when L=4, but buy price is 23% when L=3 and 47% when L=4, which means although agent 1 can get more goods when L=4, he has to pay higher to buy the goods. So, it is reasonable for the fitness values to show different values between different Ls as shown in Table. 3.3, because the number of goods agent 1 wants is changed from 3 to 4 and simulations are done under 50 different environments. For the above reason, it has a meaning to compare the fitness value in the training phase with the one having the same L in the testing phase, but it is meaningless to compare the fitness values with different Ls.

⟨*Poorest Case*⟩

In this simulation, agent 1's private prices are set at the lowest for each good under the same simulation conditions as the former simulation.

Table 3.4: Study on Fitness, Buy Prices and Number of Goods (NTLM Training, Poorest Case)

	Generations	1	5	10	100
L=3	Fitness	957	145	134	23
	Buy Price	45%	95%	110%	114%
	Number of Goods	2.8	1.2	1.1	0.7
L=4	Fitness	1223	641	583	53
	Buy Price	47%	84%	92%	146%
	Number of Goods	3.8	2.1	2.1	1.0
L=7	Fitness	1311	518	337	133
	Buy Price	63%	97%	115%	141%
	Number of Goods	4.0	2.0	1.7	0.9

Table. 3.4 shows the simulation results. We can see that the fitness value of agent 1 is getting lower when the number of generations of other agents is getting larger, and also the average buy price of agent 1 is becoming higher. Even if agent 1 has less money, he can buy more goods at lower prices when the number of generations of other agents is small. When the other agents evolve for larger generations, it is hard for agent 1 to get goods. At last, agent 1 has to pay a very high buy price which is very close to the highest limit price to get goods, i.e., nearly 150% of the common price. In other words, when GNP individuals evolve for enough generations, even the 'Poorest' agent can have a chance to win the goods. Compared to the former simulation, there is a tendency that the fitness becomes small, the buy price becomes large and the number of goods becomes small.

3.3.2.2 Testing Simulations

In the testing part, it is studied how the evolved GNPs can acquire the generalization ability. The following two simulations correspond to the two NTLM training simulations, where the same simulation conditions are used.

Table 3.5: Study on Fitness, Buy Prices and Number of Goods (NTLM Testing, General Case)

	Generations	1	5	10	100
L=3	Fitness	998	670	450	61
	Buy Price	46%	81%	112%	125%
	Number of Goods	2.8	2.2	1.9	0.7
L=4	Fitness	884	764	502	62
	Buy Price	57%	79%	119%	137%
	Number of Goods	3.8	2.7	1.8	0.8
L=7	Fitness	2217	833	661	79
	Buy Price	34%	71%	101%	124%
	Number of Goods	6.7	3.2	2.7	1.1

Table. 3.5 and Table. 3.6 show the simulation results of the average performance of agent 1 individual over 10 new different environments.

Table 3.6: Study on Fitness, Buy Prices and Number of Goods (NTLM Testing Poorest Case)

	Generations	1	5	10	100
L=3	Fitness	938	110	74	17
	Buy Price	49%	102%	135%	133%
	Number of Goods	2.8	1.2	1.1	0.6
L=4	Fitness	1087	561	477	37
	Buy Price	64%	93%	124%	148%
	Number of Goods	3.8	2.1	2.1	0.9
L=7	Fitness	1241	450	306	64
	Buy Price	70%	113%	132%	147%
	Number of Goods	4.0	2.0	1.7	0.9

From Table. 3.5 and Table. 3.6, we can see that the results have the same trend as the training results and even in the testing the best individual still works very well in the new environments. Compared to the training results, agent 1 can get almost as many goods as the training, although the fitness is a little bit lower. In other words, the best GNP-based agent can get good generalized strategies for many different situations in the no time limit model.

3.3.3 Time Limit Model (TLM)

3.3.3.1 Training Simulations

⟨*General Case*⟩

In the time limit model, all the auctions are done in the same way as the no time limit model except the limited time steps of 100.

Table. 3.7 shows the simulation results changing the numbers of generations of other agents in the general case where the private prices are set randomly. Here, *Fitness*, *Buy Price* and *Number of Goods* have the same meaning as the no time limit model.

Fig. 3.4 shows the average fitness values of the best individual of agent 1 over 50 environments until 1000th generation. The result was obtained under the condition that the other agents evolve for 100 generations and it can be seen from Fig. 3.4 that

GNP can help agent 1 to get higher fitness through evolution as the former NTLM simulation. From Table. 3.7, it can be seen that the fitness value of agent 1 becomes lower when the number of generations of other agents is larger. In addition, agent 1 gets fewer goods if the other agents evolve for larger generations. The above phenomenon are the same as the results of NTLM.

Table 3.7: Study on Fitness, Buy Prices and Number of Goods (TLM Training, General Case)

	Generations	1	5	10	100
L=3	Fitness	1261	1198	1018	572
	Buy Price	22%	11%	4.1%	3.6%
	Number of Goods	3.0	2.7	2.0	1.4
L=4	Fitness	1693	1525	1319	1101
	Buy Price	19%	13%	3.9%	3.6%
	Number of Goods	4.0	3.6	3.1	2.2
L=7	Fitness	2318	2261	2094	1178
	Buy Price	37%	16%	5.5%	4.3%
	Number of Goods	6.1	6.0	5.5	3.1

But, we can see that the average buy price of agent 1 becomes quite lower than NTLM when the number of generations of other agents is larger, which means that agent 1 can buy the goods at very low price. This is the interesting difference resulted from the different conditions of selling goods.

In NTLM, there is no time limit in the auction, what's more, if no one wants to bid for successive 3 time steps, the auction will end. For this reason, if agent 1 wants to buy goods as many as possible, he keeps bidding to scare away the others, and pays more than 100% of the private prices. But in TLM, the allowed time steps of the auction is predetermined, as a result, even when there is no one willing to bid for quite long time steps, the auction won't end until the closing time step. Therefore, all the agents learn the fact that the less frequently the bid is done, the higher fitness is obtained through evolution. So, at last, in most situations, the agents usually do not bid until the last time step. But, in some situations, agent 1 still tries to bid at higher prices to compel the others to quit.

Fig. 3.5 shows the details of how the best individual of agent 1 bids at each step in

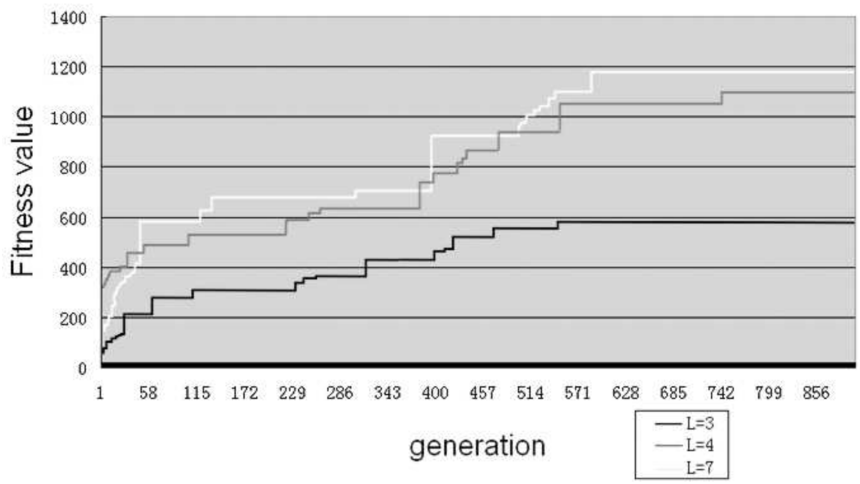


Figure 3.4: Fitness value of agent 1 in the case of TLM

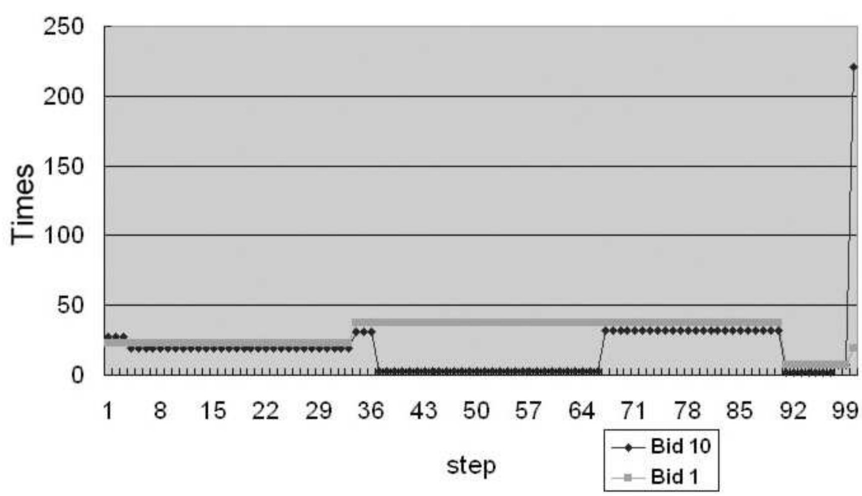


Figure 3.5: Bid actions of the best individual (TLM Training, General case)

50 training environments when the number of goods agent 1 wants is equal to 5. We can see from Fig. 3.5 that the best individual bids 10 at the last step quite frequently, and the bid 1 is done less frequently. This behavior shows that the agents become to know the importance of the last time step for winning the auction. And, we can find that the agents also try to bid at other time steps, even though it is not so frequent, which indicates the agent sometimes intends to force the others to quit.

⟨*Poorest Case*⟩

All the auctions are done in the same way as the former TLM simulation except that agent 1's private prices are set at the lowest values for each good.

Table. 3.8 shows the simulation results. We can see from Table. 3.8 that the fitness value of agent 1 is getting lower and also the average buy price of agent 1 is getting lower when the number of evolving generations of other agents becomes larger, which is the same as the former TLM simulation. Even if agent 1 has less money, he can buy more goods at low prices when the number of generations of other agents is small. But, when the other agents evolve for enough generations, they get smarter to choose the bid at the last time step. As a result, it becomes hard for agent 1 to get enough goods.

Table 3.8: Study on Fitness, Buy Prices and Number of Goods (TLM Training, Poorest Case)

	Generations	1	5	10	100
L=3	Fitness	988	869	543	175
	Buy Price	36%	18%	5.4%	4.5%
	Number of Goods	2.9	2.6	1.3	0.4
L=4	Fitness	1273	1069	809	357
	Buy Price	22%	18%	18%	4.5%
	Number of Goods	3.9	3.1	2.2	0.9
L=7	Fitness	1977	1761	1306	525
	Buy Price	39%	24%	13%	5.6%
	Number of Goods	5.7	5.1	3.7	1.2

Fig. 3.6 shows the details of how the best individual of agent 1 bids at each step in 50 training environments. From Fig. 3.6, we can see the same results as the former TLM simulation that the best individual bids 10 at the last time step quite often, and the best individual of agent 1 rarely bids at other steps. So, at last, agent 1 chooses

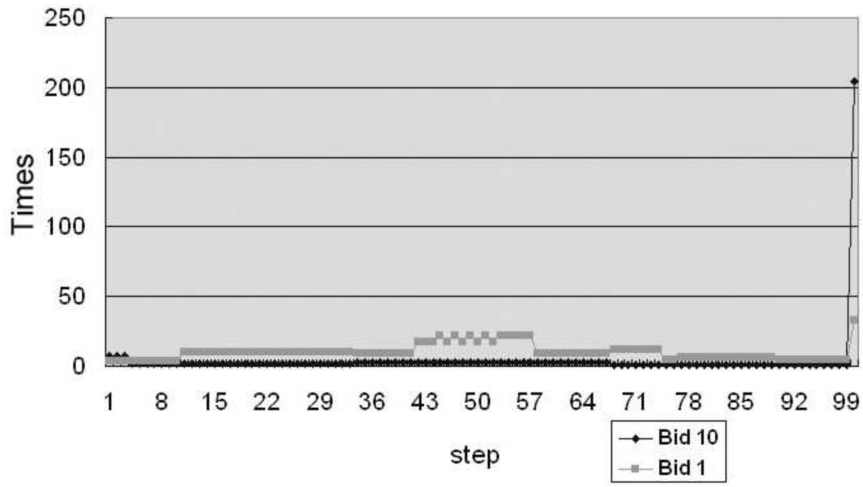


Figure 3.6: Bid actions of the best individual (TLM Training, Poorest case)

to bid as less frequently as possible to avoid the price rise. It is because there is less possibility for the agent with lowest private prices to scare the other agents than the former TLM simulation. In other words, when GNP individuals evolve for enough generations, even the 'Poorest' agent can have a chance to win the goods. Even though sometimes the poorest agent tries to bid at a very high price, the others won't quit, as a result, even if he wins finally, he may get a very low fitness.

3.3.3.2 Testing Simulations

In the testing part, for studying the generalization ability, the conditions are set in the same way as the no time limit model. The following two testing simulations correspond to the two TLM training simulations, where the same simulation conditions are used.

Table. 3.9 and Table. 3.10 show the simulation results of the average performance of agent 1 individual over 10 new different environments.

From Table. 3.9 and Table. 3.10, we also can see that the results have the same trend as the TLM training results, which are also the same as NTLM. In addition, even in the testing, the best individual still works very well in the new environments. Compared to the training results, agent 1 can get almost as many goods as the training phase, although the fitness is a little bit lower. The other observed phenomena are the same as the training part. In other words, the best GNP-based agent can get good

Table 3.9: Study on Fitness, Buy Prices and Number of Goods (TLM Testing, General Case)

	Generations	1	5	10	100
L=3	Fitness	1206	1084	914	492
	Buy Price	34%	17%	4.5%	3.8%
	Number of Goods	3.0	2.7	1.8	1.1
L=4	Fitness	1553	1377	1208	989
	Buy Price	41%	23%	8.1%	3.6%
	Number of Goods	3.9	3.3	2.9	2.0
L=7	Fitness	2017	1992	1787	1298
	Buy Price	54%	19%	8.1%	4.4%
	Number of Goods	5.5	5.3	4.8	3.1

Table 3.10: Study on Fitness, Buy Prices and Number of Goods (TLM Testing, Poorest Case)

	Generations	1	5	10	100
L=3	Fitness	945	817	466	119
	Buy Price	44%	27%	11%	7.9%
	Number of Goods	2.9	2.5	1.1	0.3
L=4	Fitness	1146	1002	683	211
	Buy Price	51%	33%	23%	8.1%
	Number of Goods	3.8	3.1	2.0	0.4
L=7	Fitness	1850	1572	1084	352
	Buy Price	44%	39%	19%	10%
	Number of Goods	5.5	4.6	3.0	0.8

generalized strategies for many different situations in the time limit model as well.

3.3.4 Comparison with Conventional Strategies

The comparison between the proposed intelligent agent and the conventional agents is done in this section. First of all, the auction model is initialized to 10 goods and 7 agents. In the real world, the conventional agents usually judge the environment and make decisions by using only a few information and less consider the opponents' behavior. So, in the simulations, agent 1 is treated as the intelligent agent, whose GNP individuals have all kinds of JNs (in NTLM, 19 kinds of JNs, in TLM, 20 kinds of JNs). On the other hand, the other agents are assumed to be the conventional auction agents, whose GNP individuals have only 5 kinds of JNs. The above is supposed because if the agent has more kinds of JNs, he can judge the auction situations more accurately, while if the agent has only 5 kinds of JNs, he can't judge the auction situations very well.

Table 3.11: Comparison with conventional agents

	Generations	NTLM(proposed)	TLM(proposed)
L=3	Fitness	641(998)	1087(1206)
	Buy Price	103%(46%)	37%(34%)
	Number of Goods	2.6(2.8)	2.8(3.0)
L=4	Fitness	987(884)	1203(1553)
	Buy Price	94%(57%)	48%(41%)
	Number of Goods	3.8(3.8)	3.8(3.9)
L=7	Fitness	1466(2217)	1679(2017)
	Buy Price	97%(34%)	33%(54%)
	Number of Goods	5.8(6.7)	5.9(5.5)

In summary, the conventional agents have 5 kinds of JNs and all of them evolve for 1000 generations in 50 different environments using GNP method as the proposed agent, i.e. agent 1. Then, the proposed agent and the conventional agents use their best individuals in the training to compete each other in 10 testing environments.

The price setting are the same as the general case described above. After evolution, 10 new environments also used for testing. The same parameters are used as Table. 3.2, except that the number of JNs for agent 2-7 is 25.

Table. 3.11 shows the average testing results obtained from 10 new testing environments. We can see that after all the agents evolved for 1000 generations, agent 1 can get almost all the goods he wants. Because the other agents have only 5 kinds of JNs, even if they evolve for 1000 generations, agent 1 can perform well. It is a great difference compared with Table. 3.5 and Table. 3.9. Also, we can see that Buy Prices are not very low, this is because the others can not judge the environments very well, so it is difficult for them to find the strategy that can help them to buy the goods at very low prices. As a result, agent 1 also has to pay a little higher prices to get the goods he wants.

3.4 Conclusions

We can see from the simulations of the proposed no time limit and time limit model that GNP can help the auction agents to understand various environments through experiences, then to find the generalized optimal strategies which suit for many environments.

The simulation results show that the GNP-based agents can understand the environments well and become smarter through evolution, and even the poorest agent can get goods when it evolves longer than the others. Testing simulation results indicate that the GNP-based agents can get better generalization ability and finally can find the general optimal strategies for different new environments, no matter in the no time limit model or time limit model. Compared with the conventional auction agents, it is also found that the agent based on GNP with the ability of judging various kinds of environments is more flexible for various auction situations due to the evolutionary feature.

Chapter 4

Enhancing Bidding Strategy for MREA using GNP

4.1 Introduction

After the ability of GNP for guiding auction agent has been proved in chapter 3, this chapter aims to enhancing the effectiveness and sensitivity of the bidding strategy for MREA using GNP [80; 81].

4.1.1 Motivation

In chapter 3, it has been found that GNP-based agent can participate in multiple round auctions and collect information from the ongoing auctions, then make bid decisions to get more goods without losing money. Furthermore, no matter the auction has the time limit or no time limit, the strategy developed by GNP can help the well evolved agents to find the suitable general strategy depending on the auction situations and to get almost all the desired goods they want. However, we believe its intelligence can be further improved. On the other hand, in chapter 3, only GNP-based agents are considered and participated in the auction environments. It is helpful for better evaluating and analyzing the bidding strategy developed by GNP to study how the GNP agent will behave when it competes with agents using different strategies.

4.1.2 Major points

- The GNP structure is modified, where the judgment functions of GNP can judge more kinds of situations at a time.

- The agent considers not only the auction's state and bidder's private price, but also the bidder's attitude towards the good. The aspiration is devided into three attitudes [44; 45]: *desperate for the good*, which means the agent wants the good desperately; *looking for a bargain*, which means the agent does not want the good very much and wants to buy it at a cheap price; *the combination of the above two*, which means the agent considers the balance of both aspects: saving money and gaining goods.

- the agent still uses the GNP structure as its bidding strategy, but uses more information on the current auction environments.

- In order to realize the above, new various judgments are proposed, such as judging the agent's attitude, the time step length of the auction, the current bid price, other agents' behaviors and bidding action history. More processing functions are also proposed.

- The aim of the improved agent is to make bidders more satisfied with the profit and the number of goods they obtain by considering more comprehensive factors. The improved GNP strategy is compared with the previous one, and also the NonGNP strategy which is introduced in chapter 2.2 in MREA with general case and poorest case.

In section 4.2, the improved GNP bidding strategy is introduced in detail. Section 4.3 gives the simulations for comparing and studying the proposed GNP strategy, and the results are analyzed. Section 4.4 concludes this chapter.

4.2 Enhanced MREA Bidding Strategy using GNP

4.2.1 Improvement

Based on the past research mentioned in chapter 3, the improved strategy developed by GNP has been proposed and compared with the related conventional strategies introduced in section 2.2.2.

- Firstly, the agents are enhanced to consider the bidders' intentions as well, which is more comprehensive for the agent to judge the auction situation and make bidding decisions.

- Additionally, the judgment nodes are modified to be more efficient. Each proposed JN can have not only two conditional branches, but also three or more branches in order to judge more complicated situations. Take a simple example, in the past, for judging the current price, we used *JN1* to judge whether the current price is low, while *JN2* and *JN3* judged whether it is middle or high, respectively; while in the modified judgment node, a JN can have three outgoing branches to judge whether the current price is low, middle, or high, respectively. This amelioration brings more compact structure, and reduces the redundant judgments during the node transition.

- Lastly, instead of increasing the current bidding price by only a small step or a large step in the previous research, the agent is improved to have various bid actions to increase bid prices.

4.2.2 Detailed Explanation

The improved structure is given in Fig. 4.1. The specific explanation for the GNP transition is as follows: The directed connection of JNs and PNs work as auction bidding rules. JNs judge the situations of the current auction and decide to which node to move next until reaching a PN, and the GNP-based agent carries out the bid action at the PN. While, other participating agents submit the bid by using its own strategy. Then, when the bid actions of all the agents are done, the next time step begins, and the next transition of the GNP-based agent begins from the last PN until it moves to another PN. This process iterates until the designated time steps finish.

In Fig. 4.1, the bold line represents a part of the nodes transition in GNP. At time step t, the transition reaches $PN3$ and it reaches $PN5$ at time step t+1, and it might reach the same $PN3$ and $PN5$ again several time steps later. Such a transition reuses PNs and JNs until the auction is done. For more specific, there is another basic example to explain how nodes transition goes on according to the auction situations.

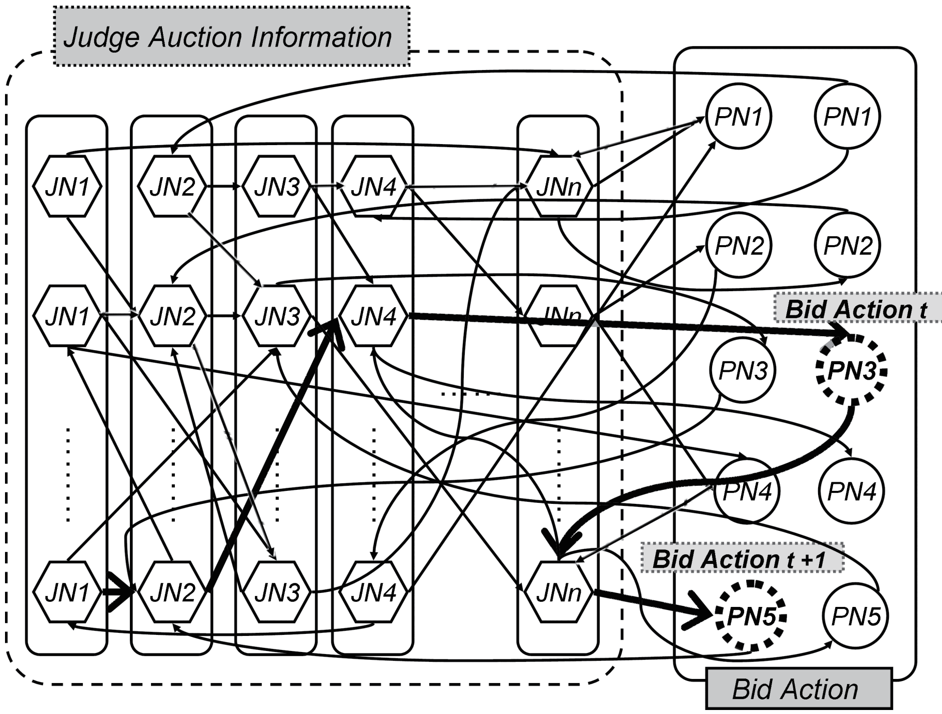


Figure 4.1: Structure of GNP individual

Table 4.1: 7 Kinds of Judgment Nodes for MREA

$JN1$:	the current good number on the current auction
$JN2$:	length of the time steps of the current auction, short/medial/long.
$JN3$:	price of the bid action P^L in the last time step.
	$P^L = 0/1 \leq P^L \leq 3/4 \leq P^L \leq 7/8 \leq P^L \leq 10$.
$JN4$:	agent's attitude to the current good, desperate/both/desire for bargain.
$JN5$:	position of the current time step in the whole time steps, early/medial/late.
$JN6$:	if the current time step is n, then the bid actions of the last time step and the time step before the last time step are denoted as b_{n-1} and b_{n-2}, respectively. Judge S, which is the amount of $b_{n-1} + b_{n-2}$.
	$S = 0/1 \leq S \leq 7/8 \leq S \leq 15/16 \leq S \leq 20$.
$JN7$:	judge which one is bigger, b_{n-1} or b_{n-2}

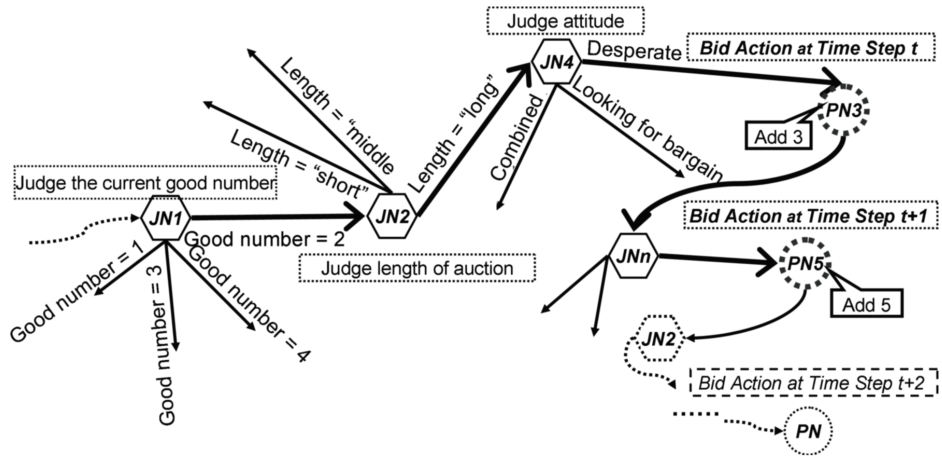


Figure 4.2: Nodes transition

Take the bold line shown in Fig. 4.1, we can see from Fig. 4.2 that when the node transition moves to a *JN*, the judgment function of *JN* judges the auction situation and choose one of the branches of it. When the node transition moves to a *PN*, the corresponding bid action of time step *t* will be done, then the node transition keeps on moving until the next *PN* for the bid action of time step *t+1*.

The basic evolutionary flowchart of GNP-based agent in MREA is shown in Fig. 4.3. In the training phase, GNP-based agent and other agents do the auction round processes in MREA in each generation like the *Do Auction* part in Fig. 4.3. After every individual has done the process, GNP-based agent evolves to the next generation for better GNP structure until the last generation. One run of MREA for each individual is shown as Fig. 4.4.

4.2.3 Kinds of Nodes

7 kinds of judgment nodes and 11 kinds of processing nodes are newly proposed for MREA. The *JN*s are proposed based on the auction situations needed to judge, while *PN*s correspond to the bid actions. 7 kinds of *JN*s judge the situations based on the auction information, such as the current good on the auction, the length of the current auction, the relation between the current price and agent's private price, the agent attitude to goods, the position of the current time step in the whole time steps, price increasing amount, price increase speed and so on. Meanwhile, 11 kinds of *PN*s represent 11 different bid actions that increase the current bid price by $0/1/2/\cdots/9/10$,

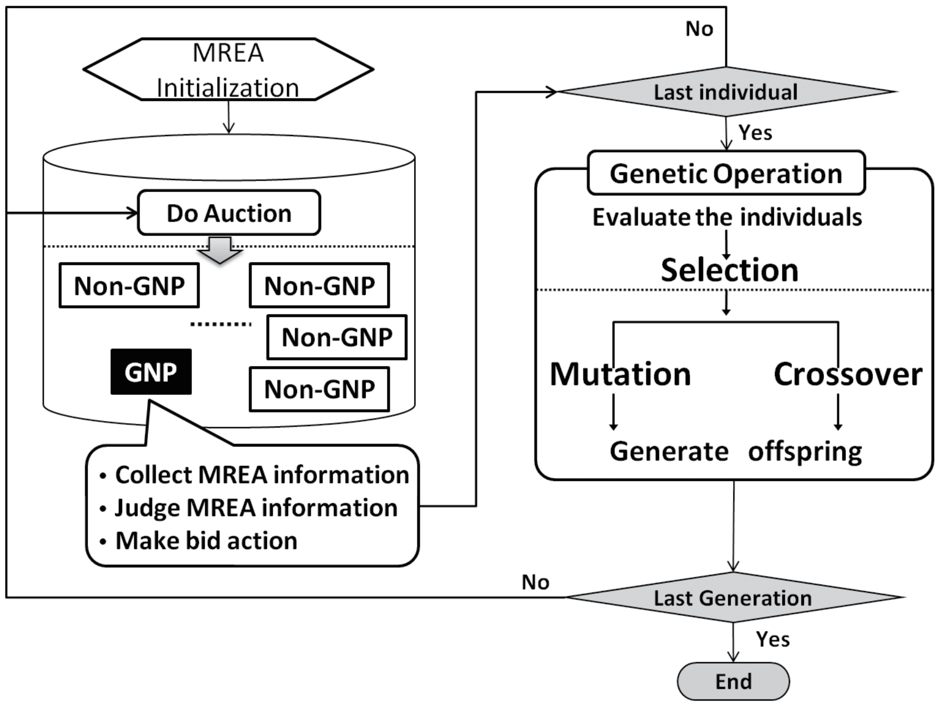


Figure 4.3: Basic evolutionary structure of GNP-based agent

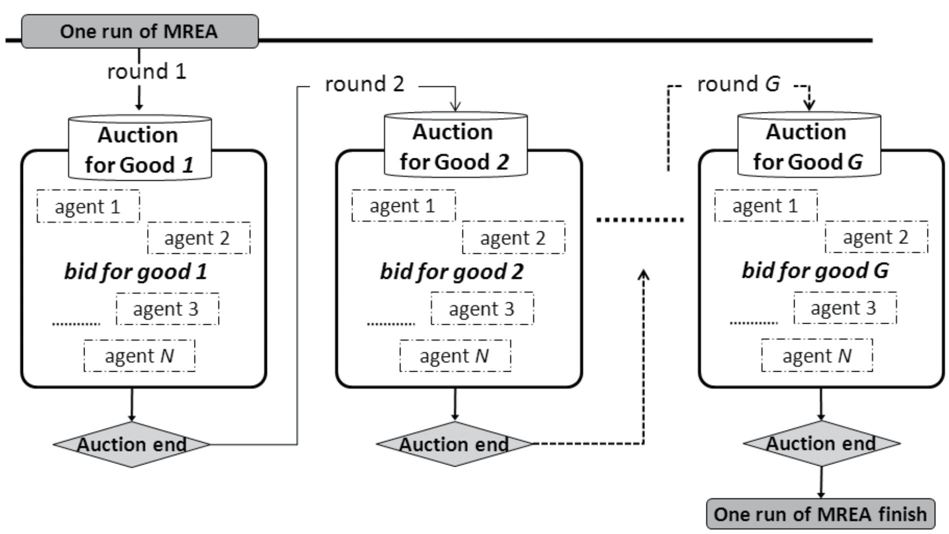


Figure 4.4: Procedure for one run of multiple round English auction

respectively. There are two special cases: if the agent does not want to submit a bid, which means it wants to stay at the current price, we treat this case as submitting a bid equal to 0 to the current price; if the agent quits from the current auction, it does not need to submit a bid to this auction any more.

The proposed Judgment Nodes (JNs) are shown in Table. 4.1.

4.2.4 Fitness function

The fitness function $Fitness(i)$ of individual i is composed of $f_1(i)$, $f_2(i)$ and $f_3(i)$, which are denoted as the fitness values of the goods obtained with three different attitudes. It is easily understood that P^P minus P^F indicates the profit the agent obtains, and we use an additional value ($\alpha * P_g^C$) to calculate the fitness. This is because we consider the cases where an agent can finally buy goods exactly at their private prices, which means the profit is 0, and an agent can buy no good, which means the profit is also 0. So, in order to distinguish the agents that could buy goods from the agents who failed to buy goods, we add the term of $\alpha * P_g^C$ to make sure that the agents winning goods have a higher fitness. Moreover, we set different weights (ω_1, ω_2 and ω_3) of the fitness for different attitudes. Because the goods with *desperate aspiration* attitude should be evaluated highly, so we set a higher weight for *desperate aspiration* attitude to encourage the agent to make most efforts for the goods. For the same reason, we set a lower weight for the *looking for a bargain* attitude.

$$Fitness(i) = \omega_1 \times f_1(i) + \omega_2 \times f_2(i) + \omega_3 \times f_3(i), \tag{4.1}$$

$$f_1(i) = \sum_{g \in G_1} (P_g^P(i) - P_g^F(i) + \alpha * P_g^C), \tag{4.2}$$

$$f_2(i) = \sum_{g \in G_2} (P_g^P(i) - P_g^F(i) + \alpha * P_g^C), \tag{4.3}$$

$$f_3(i) = \sum_{g \in G_3} (P_g^P(i) - P_g^F(i) + \alpha * P_g^C), \tag{4.4}$$

where,

- i: individual number of the GNP-based agent.

- G_1: set of suffixes of goods the GNP-based agent wanted with desperate aspiration and finally gained.

- G_2: set of suffixes of goods the GNP-based agent wanted with both desperate and looking for a bargain attitudes and finally gained.

- G_3: set of suffixes of goods the GNP-based agent wanted with looking for a bargain and finally gained.

- $P_g^P(i)$: private price of good g by agent i.

- $P_g^F(i)$: final buy price of good g by agent i.

- P_g^C: common price of good g.

- ω_1, ω_2 and ω_3: the weight corresponding to three attitudes, $\omega_1 \succ \omega_2 \succ \omega_3$.

- α: control factor.

In summary, the GNP-based agent uses GNP structure for determining his bidding strategy. After judging the relevant information, GNP-based agent takes the bid action according to its own GNP structure. After all auction rounds of MREA finish, GNP-based agent evaluates the performance of GNP individuals by using the fitness function, select the better ones and use them to generate new individuals for the next generation by mutation and crossover operations.

4.3 Simulations and Analysis

The simulations are done and extended based on [81].

4.3.1 Comparison of the Proposed GNP strategy with Conventional GNP strategy

In this subsection, the proposed new GNP strategy (ProGNP) is compared with the conventional GNP strategy (ConGNP)[78]. The features of the new strategy are explicitly revealed by this comparison.

$\langle Assumption\ for\ GOODs \rangle$

Table 4.2: Parameters Setting

Items	Value
Number of Goods	10
Number of Training environment	50
Number of Testing environment	10
Number of Agents	3-10, 3-7
Generation	500
Population Size	200
—Elite	10
—Crossover	80
—Mutation	80
—Randomly Generated	30
Selection	upper 30%
Crossover Rate	0.1
Mutation Rate	0.3
Node	
—Judgment Node	100
—Processing Node	55
—Start Node	1

There are 10 goods in one environment of MREA, so there are 10 English auctions which should be done good by good orderly. They have pre-assigned different length of time steps which are distributed from 30 to 100 time steps. If the auction reaches the closing time step, the agent which offers the highest price wins the good, and the auction for the next good starts. Also, each good has a different common price which is randomly generated from 50 to 200. After the initialization, all these values are fixed during the whole bidding procedure.

$\langle Assumption\ for\ AGENTs \rangle$

There are 3 different agents participating in the auctions to compete with each other. All of them want to bid for all the 10 goods. Agent's private prices for goods are distributed from 85% to 115% of the common prices of the goods. Also, the attitude of each agent, like desperate for the good, looking for a bargain and the combination of the two, is randomly generated for each good.

$\langle Assumption\ for\ PROCESS \rangle$

We only focus on the performance of Agent No. 1 of ConGNP and ProGNP in the following two simulations. $\langle A \rangle$ The ConGNP strategy is assigned to agent No. 1, and the other 2 agents also use ConGNP strategy; $\langle B \rangle$ The ProGNP strategy is assigned to agent No. 1, and the other 2 agents use ConGNP strategy.

In the training phase, in both simulations, these 3 agents do the co-evolution for 10 generations first, then only agent No. 1 using ConGNP evolves up to 1000 generations in simulation A, while agent No. 1 using ProGNP evolves up to 500 generations in simulation B.

Each environment has 10 auctions, and 50 different environments are used for training GNP individuals, which means there are 500 auction rounds. For all the environments, the common prices of the goods are different from each other. The specific parameters used in the simulations are shown in Table. 4.2, except the number of agents is only 3.

Table 4.3: Comparison of Averaged Number of Goods Obtained of ConGNP and ProGNP

	ALL	DE	Both	DB
$\langle A \rangle$ ConGNP	8.125	7.980	8.225	8.160
$\langle B \rangle$ ProGNP	8.475	9.040	8.260	7.970

In the testing, i.e., the generalization phase, the best individual of agent No. 1 in the training phase compete with the same other agents as the training phase in 10 new environments different from the 50 environments in the training phase. Each simulation runs 5 times and all the results are the average results over 5 runs in 10 environments.

The average *Number of Goods(NoG)* of agent No. 1 over 10 different testing auction environments in simulation A and B are compared in Table. 4.3. *ALL* means the NoG for all the wanted goods; *DE* means the NoG for the wanted goods with Desperate attitude, while *Both* and *DB* means the NoG for the wanted goods with other two attitudes, respectively.

It can be found from Table. 4.3 that the proposed new GNP strategy outperforms the conventional GNP strategy in terms of the number of goods obtained. Additionally, because the new GNP strategy considers the attitude factors, and the fitness function guides the agent to give more importance to the wanted goods with Desperate attitude, so the NoG of *Desperate* is the highest. On the other hand, the NoG of ConGNP does not reveal such a phenomenon as ProGNP because ConGNP does not consider the attitude factors.

4.3.2 Comparison of the Proposed GNP Strategy with Non GNP-based Strategy

In this subsection, the proposed GNP strategy is compared with Non GNP-based bidding strategy described in section 2.2.2 in order to make the advantages of the proposed method clear.

$\langle Assumption\ for\ GOODs \rangle$

It is the same as the assumption described in section 4.3.1.

$\langle Assumption\ for\ AGENTs \rangle$

It is also the same as the assumption described in section 4.3.1, except that one GNP-based agent, and other several different Non GNP-based agents participate in the auctions.

$\langle Assumption\ for\ PROCESS \rangle$

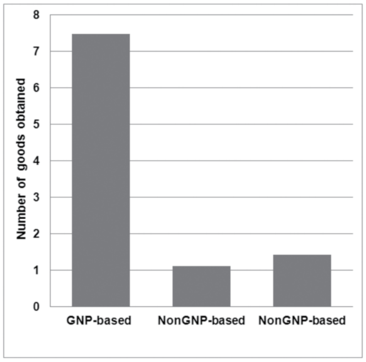
Only agent No. 1 has the GNP based evolving strategy. As we mentioned, after all the individuals finish the *Do Auction* part in Fig. 4.3, the GNP-based agent does the genetic operations and evolves to the next generation.

In the training phase, agent No. 1 based on the GNP strategy competes with other agents based on their handcraft strategies for 500 generations. Each MREA environment has 10 auctions, and 50 different MREA environments are used for training GNP individuals. In all the environments, the common prices of the goods are different from each other.

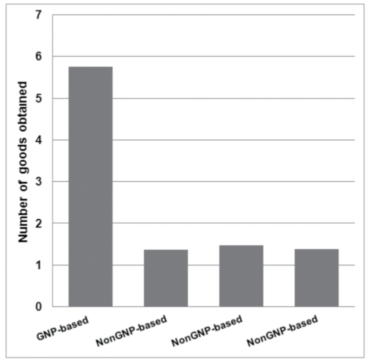
The specific parameters used in the simulations are shown in Table. 4.2. Moreover, the parameters in the fitness function are set at: $\omega_1 = 1.1$, $\omega_2 = 1.0$, $\omega_3 = 0.9$ and $\alpha = 0.2$.

In the testing, 10 new MREA environments different from the 50 environments in the training phase are used for testing the best individual of agent No. 1. Each simulation runs 5 times and all the results are the average results over 5 runs in 10 environments.

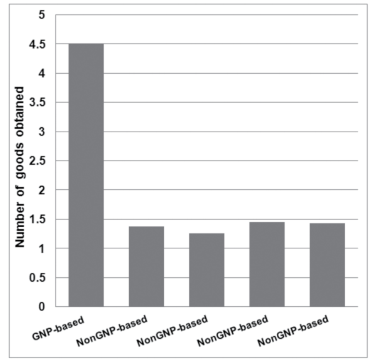
The parameters used for the *Non-GNP based strategy* are shown in Table 4.4.



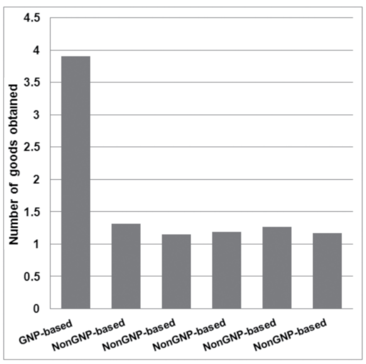
(a) General Case,3 agents



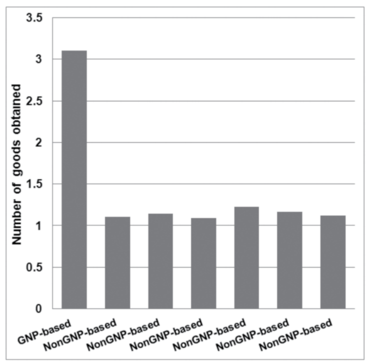
(b) General Case,4 agents



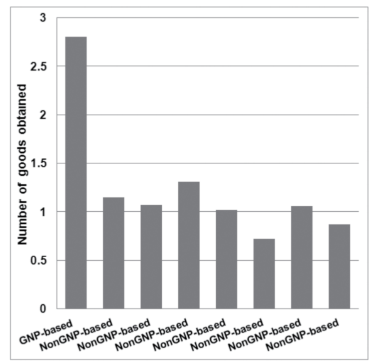
(c) General Case,5 agents



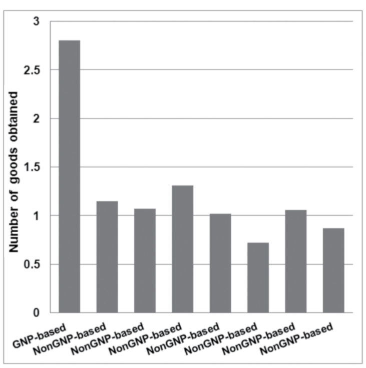
(d) General Case,6 agents



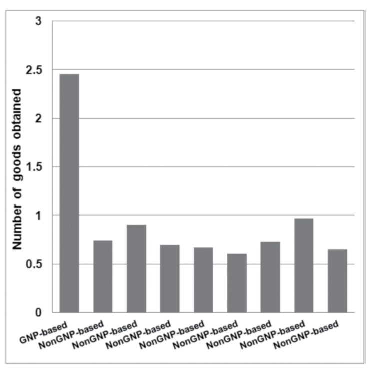
(e) General Case,7 agents



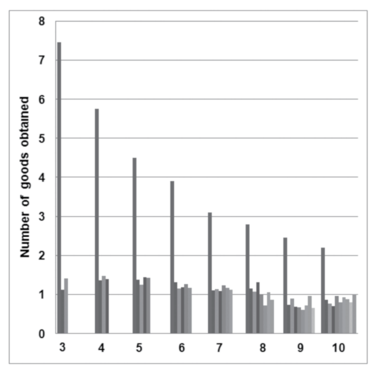
(f) General Case,8 agents



(g) General Case,9 agents

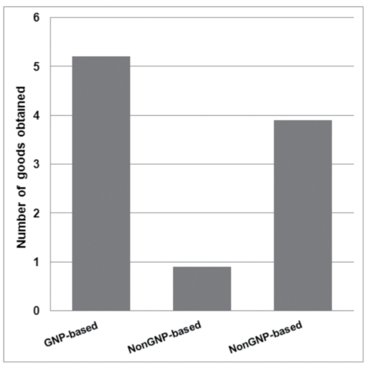


(h) General Case,10 agents

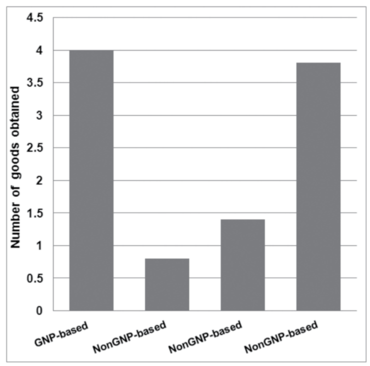


(i) General Case,Overview

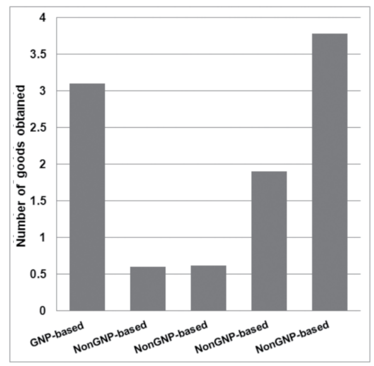
Figure 4.5: Number of goods obtained in general case



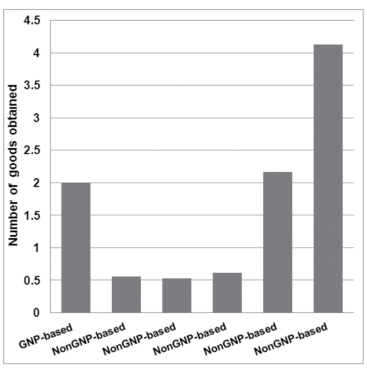
(a) Poorest Case,3 agents



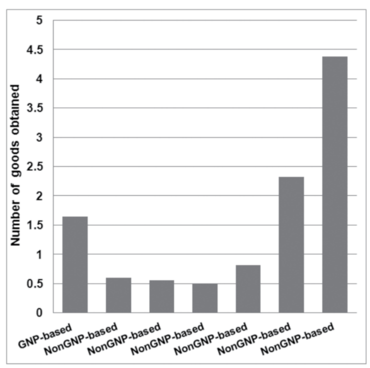
(b) Poorest Case,4 agents



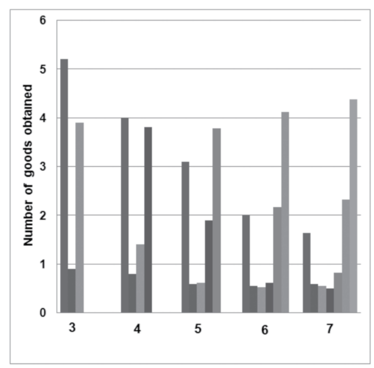
(c) Poorest Case,5 agents



(d) Poorest Case,6 agents



(e) Poorest Case,7 agents



(f) Poorest Case,Overview

Figure 4.6: Number of goods obtained in poorest case

Table 4.4: Parameters Setting for non-GNP-based Agents

Items	Value
k_{db}	0.3
β_{db}	0.3
k_{de}	0.7
β_{de}	5
k_{rt}	0.6
β_{rt}	4
2 tactics combination	ω_{db}=ω_{de}=ω_{rt}=0.5
3 tactinc combination	ω_{db}=ω_{de}=0.25, ω_{rt}=0.5

⟨*General Case*⟩

In this case, all the agents' private prices of the goods are set at between 85% and 115% of their common price. This setting allows each agent to have a higher or lower private price than others. The GNP-based agent, i.e., agent No. 1 and the handcraft strategy based agents are compared in Fig. 4.5, in terms of the average Number of Goods(*NoG*) obtained over 10 different testing auction environments under various numbers of participating agents.

It can be seen from Fig. 4.5 that when a small number of agents are participating in the auction, agent No. 1 can perform very well and achieve higher NoG than the agents using the strategy proposed in [44], where agent No. 1 is the GNP-based agent and the other agents are the Non GNP-based (handcraft) strategy based agents. This can be explained by the features of the GNP-based agent. Agent No. 1 can analyze the others' bidding strategies and evolve to find the winning strategy using the information on many situations of the auction. Meanwhile, we can also see from Fig. 4.5 that agent No. 1 can get more goods compared with other agents in many different environments. Furthermore, from each testing run, the average number of goods(*NoG*) obtained by Non-GNP strategy is compared with the number of goods obtianed by GNP strategy. As is shown in Table 4.5, the p values of the t-test (two-tailed) show there are statistically significant differences between the GNP-bidding strategy and Non-GNP strategy (at 5% significant level).

However, it can be found from Fig. 4.5(i) that the *NoG* of agent No. 1 decreases as more agents participate in the auctions. This is because if there is only a small number of Non GNP-based agents participating in the auctions, agent No. 1 can find the appropriate bidding price at each time step more easily. On the other hand, as more opponents participate in the auction, the time complexity and space complexity for evolution increase, and also the environments turn to be more complicated because agent No. 1 have to consider all the bids from others, which leads to worse performances of agent No. 1.

⟨*Poorest Case*⟩

Different from the general case that randomly generate the private prices, in this case, the private price of agent No. 1 is set at the lowest for each good, and the private prices of other agents are determined in such a way that: the P^Ps become higher as the agent *id* number becomes larger. For example, if there are 4 agents, then the private prices of No. 1 agent to No. 4 agent may be: 88, 93, 102 and 109 for the good with common price of 100.

Fig. 4.6 shows the average *NoG* of each agent over 10 different testing auction environments. We can see from these figures that when the number of participating agents is only 3, agent No. 1 can still achieve a good performance even if it has the least money. But, agent No. 1 can not get the highest NoG when the number of the participating agents is getting larger. When there are only 3 agents in the auction, the difference of the private prices between agent No. 1 and the last numbered agent is not so large, but when there are more agents in the auction, the difference between the lowest and highest private prices become larger. But, the performance of agent No. 1 is reasonably well under these unfair situations. Even though agent No. 1 can not

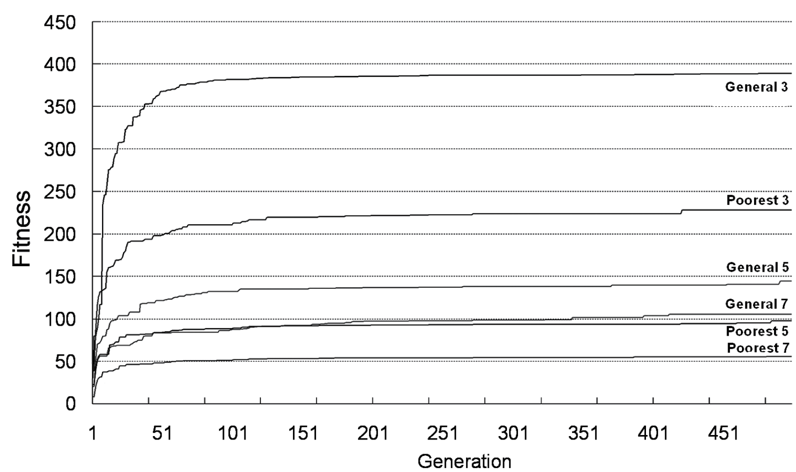


Figure 4.7: Fitness values of 6 simulations

be the winner, it can defeat some other agents except the last numbered agent with the most money. Also, Table 4.6 gives the p values of the t-test (two-tailed) between the GNP-bidding strategy and Non-GNP strategy. From each testing run, the averaged number of goods (NoG) which are obtained by Non-GNP strategy is compared with the number of goods obtained by GNP strategy. The p values show there are statistically significant differences between the GNP-bidding strategy and Non-GNP based bidding strategy (at 5% significant level).

Additionally, Fig. 4.7 gives the fitness curves of the GNP-based agent in the training phase of 6 selected simulations(General Case 3,5,7, and Poorest Case 3,5,7), which shows the GNP-based agent's ability of evolution.

4.4 Conclusions

The proposed method of applying GNP to auction agents does give a good guidance for the intelligent auction systems. GNP-based agent makes it possible to bid a price using its gene structure, which is more flexible for various auction situations compared to the non-GNP based strategies due to the evolutionary features of GNP.

It has been clarified that the GNP-based agent is more competent than the agents based on the mathematical functions when it is needed to decide what price to bid at each time step. Even when GNP-based agent has the least money, it can still perform fairly well.

Table 4.5: The t-test (p value) results of GNP and Non-GNP in general case

3 agents

	GNP	Non-GNP
average NoG (standard deviation)	7.460 (0.45)	1.270 (0.22)
t-test(p-value)	3.26×10^{-9}	

4 agents

	GNP	Non-GNP
average NoG (standard deviation)	5.750 (0.42)	1.416 (0.14)
t-test(p-value)	1.83×10^{-8}	

5 agents

	GNP	Non-GNP
average NoG (standard deviation)	4.508 (0.06)	1.376 (0.02)
t-test(p-value)	6.14×10^{-14}	

6 agents

	GNP	Non-GNP
average NoG (standard deviation)	3.900 (0.52)	1.220 (0.10)
t-test(p-value)	3.61×10^{-6}	

7 agents

	GNP	Non-GNP
average NoG (standard deviation)	3.180 (0.09)	1.136 (0.01)
t-test(p-value)	2.01×10^{-11}	

8 agents

	GNP	Non-GNP
average NoG (standard deviation)	2.800 (0.19)	1.028 (0.03)
t-test(p-value)	3.32×10^{-8}	

9 agents

	GNP	Non-GNP
average NoG (standard deviation)	2.449 (0.07)	0.943 (0.01)
t-test(p-value)	4.43×10^{-11}	

10 agents

	GNP	Non-GNP
average NoG (standard deviation)	2.201 (0.09)	0.866 (0.01)
t-test(p-value)	5.82×10^{-10}	

Table 4.6: The t-test (p value) results of GNP and Non-GNP in poorest case

3 agents

	GNP	Non-GNP
average NoG (standard deviation)	5.104 (0.30)	2.448 (0.15)
t-test(p-value)	9.73×10^{-8}	

4 agents

	GNP	Non-GNP
average NoG (standard deviation)	3.994 (0.20)	2.069 (0.18)
t-test(p-value)	2.32×10^{-7}	

5 agents

	GNP	Non-GNP
average NoG (standard deviation)	3.026 (0.19)	1.744 (0.05)
t-test(p-value)	4.73×10^{-7}	

6 agents

	GNP	Non-GNP
average NoG (standard deviation)	1.990 (0.11)	1.602 (0.02)
t-test(p-value)	6.89×10^{-5}	

7 agents

	GNP	Non-GNP
average NoG (standard deviation)	1.636 (0.09)	1.394 (0.02)
t-test(p-value)	3.99×10^{-4}	

Chapter 5

Bidding Strategy Acquisition with Heuristic Rules for CDA using GNP

5.1 Introduction

After the ability of GNP for guiding auction agent has been proved in MREA, this chapter aims to developing the bidding strategy for CDA using GNP [82; 83].

5.1.1 Motivation

As introduced in chapter 2.3, *Continuous Double Auction* (CDA) permits multiple sellers and multiple buyers to update their asks and bids through the trading period continuously. Its popularity is due to its operational simplicity and expediency in both of the two trading sides. It is valued as a significant e-commerce market mechanism because it can reflect and reserve the very basis of economy, where the real-time interactions occur between sellers and buyers, and it has the extraordinary expansibility in different domains. Thus, it is a free and highly responsive system, which can exploits the dynamics of the market and balance demand and supply efficiently.

Given its prominence and importance, and also the big improvement of autonomous software agents as well as e-commerce, the bidding strategies for agents in CDAs attract lots of attention. As introduced in chapter 2.3.3, although most of the existing approaches consider many factors of the auctions and use common belief and specific rules to guide the agents' bidding behaviors, they have some shortages. ZI-C, ZI-U and

GD pay no attention to adaptability. CP and ZIP focus on adaptability only using the last auction information. In order to be more intelligent and to overcome the shortages of the above strategies in CDA, Genetic Network Programming (GNP) with Rectify Nodes (RNs) has been applied and combined with the proposed heuristic rules for the CDA bidding strategy in this chapter (GNP-RN strategy). GNP-RN is developed aiming to guide the agent to be competitive under different environment conditions, and maximize the agent's profit without losing chances for trading. RNs are used for bringing more flexible and various options for bidding action choices.

5.1.2 Major points

- The basic judgments and processing functions, which are especially suited for the GNP-RN bidding strategy, are designed in the proposed method. This is because the specific judgment nodes and processing nodes should be designed according to the requirement of each task. The judgment contained in JNs should be able to judge the current environment information, and the processing actions contained in PNs should be able to represent the suitable actions for the current environment.

- The heuristic algorithms implemented in the GNP-RN bidding strategy have been also proposed referring to the related research in order to provide suitable and competitive potential options for bidding actions with reasonable number of processing nodes.

- RN is an extended node to the basic concept of GNP. RNs work together with PNs for more flexible and various bidding options avoiding too many number of processing nodes. Based on the heuristic knowledge, GNP-RN agent can use its structure to judge many kinds of information from the ongoing auctions, and make suitable ask and bid decisions according to the judgment results.

- From the simulation results, it can be found that when the environment contains CDAs with multiple buyers and sellers who want to trade different amount of goods, in other words, when the environment is complicated, GNP-RN can outperform other strategies. Without knowing which information is more important or what kinds of combinations of information are more useful, the GNP-RN agent can automatically find the most useful judgment functions and get the generalized best strategy, which can deal with the whole CDAs environment generation by generation. The simulation results also show that the agent can deal with various situations very well, since several situations of different supply and demand pairs are studied, and GNP-RN performed best in all the situations. It is possible for the agent to choose the pertinent solution for a certain situation as the GNP-RN structure is systematically built from the evolutionary process.

In section 5.2, the GNP-RN bidding strategy is introduced in detail. Section 5.3 gives the simulations for comparing and studying the proposed GNP-RN strategy with basic GNP strategy, ZI-C, ZI-U, CP and GD strategy, and the results are analyzed. Section 5.4 concludes this chapter.

5.2 GNP-RN: Bidding Strategy developed by GNP

5.2.1 Overview of GNP-RN bidding strategy for CDA

We studied applying GNP to develop bidding strategy for MREA, which revealed the effectiveness of GNP on guiding bidders' behaviors[78; 80; 84]. Based on these previous research, this chapter studies the GNP-RN bidding strategy for CDA using GNP with rectify nodes (RNs) and heuristic rules. The RNs are combined with PNs intending to provide more potential bidding options. The heuristics based on common believes are employed to help GNP-RN to make suitable and competitive bidding decisions with using history information.

5.2.1.1 Overview of GNP-RN structure

As Fig. 5.1 illustrates, the left part shows an overview of a CDA process, and the right part shows an example of GNP-RN bidding structure.

The GNP-RN auction agent has a population composed of many individuals representing potential bidding strategies. In GNP-RN structure, there are 4 kinds of nodes: a start node, JNs, PNs and RNs. The connections of JNs represent strategic logics of judging the auction environment, while PNs combined with RNs indicate different ask/bid actions.

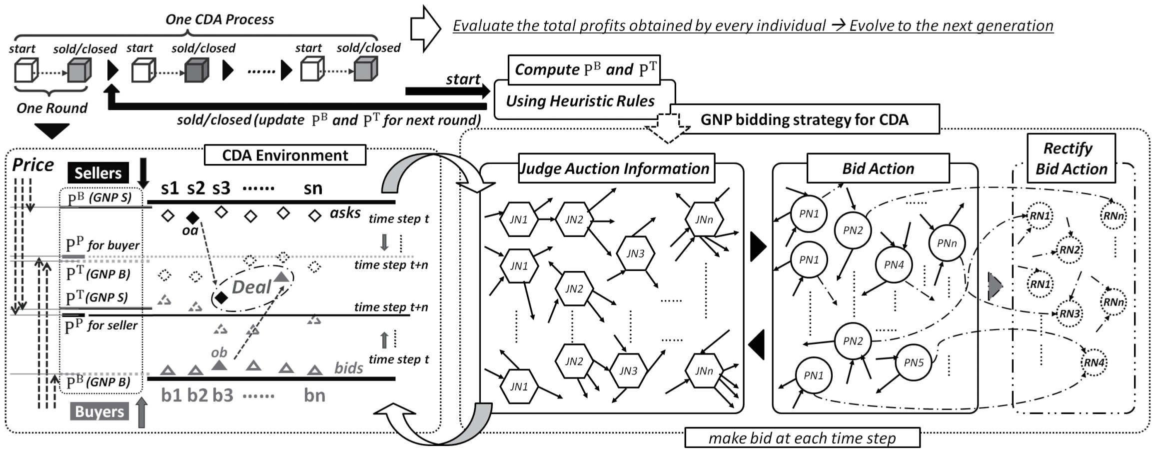


Figure 5.1: GNP structure for CDA

When an auction starts, firstly, the auction agent collects auction information and bidder information. Then, combined with the heuristic rules, GNP-RN uses these information to judge the current bidding situation. According to the judgment results, its corresponding branch is taken. When the transition from JNs reaches a PN, its corresponding action will be executed after it is adjusted by the connected RN. Therefore, different judgment results lead to different bidding decisions, so the agent can make the real-time responses to the changing auction environments.

Generally speaking, as Fig. 2.3 shows, considering the features of CDA, the following information will be collected and judged by GNP-RN:

- the agent's own P^Ps of the goods
- the total number of goods the agent wants to trade (NUM)
- the time steps of the current auction (t)
- the current oa and ob including the previous oas and obs of the past L^{ts} time steps, where L^{ts} is the number of time steps stored in the current round history
- the P^Fs of the past L_r successful transactions

Compared to the GNP structure in our previous research [78; 80], RNs are added. As shown in Fig. 5.1, each PN is connected to a RN. Although every PN contains a pre-decided heuristic function using P^P, P^B, P^T, P^F, oa and ob to determine the

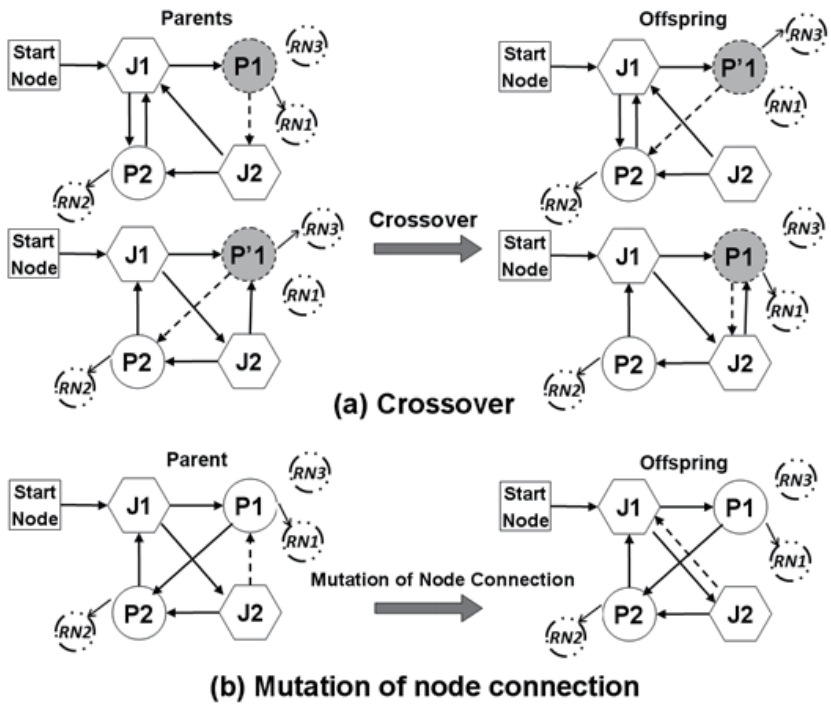


Figure 5.2: Genetic operators

ask or bid prices without increasing too many new kinds of *PNs*, the price calculated from the function in *PN* could be modified by adding a small price δ which is positive, negative or zero in *RN* to make more flexible bidding strategies. More specifically, there are several kinds of *RNs* containing different price of δ, so the *PNs* containing the same heuristic function have the possibility to connect to different kinds of *RNs*, which makes it possible to obtain different ask or bid prices even under the same heuristic function. Moreover, one *PN* connects to one *RN*, and each *RN* has no outgoing branches. After the bidding action of *PN* has been modified by *RN*, the node transition continues from the current *PN*. *RN* brings the variety and flexibility to the bidding actions without using a large number of *PNs*, and finally the most appropriate combination of *PN* and *RN* could be obtained through the evolution.

As described before, the connections among *JNs*, *PNs* and *RNs* are determined by the genetic operations of GNP evolution. The rank selection is used to select the better individuals in the current generation. Uniform crossover and uniform mutation are used to generate the offspring for the next generation. Fig. 5.2 gives a simple example of genetic operators for GNP-RN. In the crossover, two parents can exchange the corresponding parts with each other under a certain rate, while in the mutation, each parent can mutate a part of itself under a certain rate.

5.2.1.2 Overview of Heuristic Rules

The proposed heuristics are derived based on the common believes [48; 65; 71]. The outline of the proposed heuristics is as follows:

- If a seller does trade frequently, then it will submit a new *ask* a little bit higher than the previous P^F in order to gain more profit by selling at a higher price. On the opposite, if the seller does not do trade so frequently, then it is willing to submit a lower *ask* for the good even equal to its P^P to sell the good rather than no trade at all.

- Similarly, if a buyer does trade so frequently, then it will submit a new *bid* a little bit lower than the previous P^F in order to gain more profit by buying at a lower price. If the buyer does not do trade so frequently, then it is willing to submit a higher *bid* for the good even equal to its P^P to buy the good rather than no trade at all.

The above implemented in the bidding strategy developed by GNP is explained in more detail in section 3.3 and 3.4. By using the heuristics, the basic price (P^B) and the target price (P^T) for GNP-RN agents are determined. P^B and P^T are used for guiding the bidding price and making bidding at the competitive price quickly in each round.

Definition 4. *A* ***basic price*** *(*P^B*) is the starting price for the agent toward to the ongoing good in the current round.*

For a seller, $P^B = \alpha_1 \times P^P$, where $\alpha_1 \in (1, 1.5)$. For a buyer, $P^B = \alpha_2 \times P^P$, where $\alpha_2 \in (0, 1.0)$. P^Bs are set like this because a seller is willing to sell a good at a higher price than his own P^P at the very beginning of an auction, while a buyer is willing to buy a good at a lower price than his own P^P in order to make some profits. A simple example of the relation between P^B and P^P is shown in 5.1.

Definition 5. *A* ***target price*** *(*P^T*) is the expected price for the agent to make a transaction in the current round.*

Similarly, each agent has a P^T for the current good in each round by the relation between P^P and P^T shown in 5.1, which is relative to its P^P under the assumption that all the agents want to make some profit margins from the transactions [48]. Initially, for a seller, P^T is a little bit higher than his P^P, i.e., $P^T = (1 + \alpha_3) \times P^P$, where $\alpha_3 \in (0, 0.1)$, while for a buyer, P^T is a little bit lower than his P^P, i.e., $P^T = (1 - \alpha_4) \times P^P$, where $\alpha_4 \in (0, 0.1)$. Apparently, if an agent sets its profit margin at too low values, it may lose some possible profits. Nevertheless, if the agent sets its profit margin at too high values, it will lose the chances to do the trade. So, P^T will be modified through CDA period in the proposed strategy in order to do the appropriate trade with the maximal profit in the proposed method.

To sum up, the bidding strategy developed by GNP can be simply described as follows:

- Initialize the GNP population.
- At the beginning of each round, each individual firstly computes a basic price (P^B) and a target price (P^T) based on the knowledge and the relevant environment information introduced above.
- Then, at each time step, the node transition will finally turn to the PN according to the judgment results and make the corresponding bid action in PN.
- Repeat the above step until the current round ends.
- Update P^B of the next good based on P^P. Update P^T based on the relevant history information. Then, the next round starts.
- When the CDA process is finished, all the individuals' performances are evaluated by the gained profit, then genetic operations are done for the population of the next generation.

Additionally, the important terms used are summarized in Table. 5.1.

5.2.2 Bidding Strategy for Sellers

Based on the overview of GNP-RN, this section will introduce the proposed heuristic algorithms for the bidding strategy for sellers in detail. Suppose that CDA is in the r th round and the current time step is t. GNP-RN seller i, who wants to sell NUM goods and has already sold num goods, is willing to sell good n ($n = num + 1$), and P^P of this good for seller i is c_{in}.

Table 5.1: Important Terms

P^P	private price of a good the agent wants to trade	P^B	basic price for the agent to start from
c_{in}	P^P of good g_n seller i wants to sell	P^T	target price which gives a profit margin for the agent
v_{in}	P^P of good g_n buyer i wants to buy	L^{ts}	a number of time steps recorded for judgments
oa^t	outstanding (lowest) ask at time step t	L_r	a number of rounds recorded for updating P^T
ob^t	outstanding (highest) bid at time step t	R_J	a number of rounds recorded for judgments
oa_r	the last oa in rth round	$\alpha_1, \alpha_2, \alpha_3, \alpha_4$	four independent real numbers for calculating P^B and P^T
ob_r	the last ob in rth round	$\beta_1, \beta_2, \beta_3, \beta_4$	four independent small prices for updating P^T
$P_r^{F_r}$	the final traded price in the rth round	γ_s, γ_b	two small prices for bid actions
$P_g^{F_g}$	the final traded price of good g	δs	small prices contained in different RNs
V_{step}	the smallest valid bidding price		

First of all, as shown in Fig. 5.1, at the beginning of the r th round, the seller's P^B and P^T for the ongoing good should be calculated using the heuristic logics based on the common believes described in section 3.2.

When a round begins,

- P^B is given by:

$$P^B = \alpha_1 \times c_{in}, \tag{5.1}$$

  where, $\alpha_1 \in (1, 1.5)$ as introduced before.

- P^T for selling the good is given by:

  $\langle$*when* $r = 1\rangle$

$$P^T = (1 + \alpha_3) \times c_{in}, \tag{5.2}$$

  where, $\alpha_3 \in (0, 0.1)$ as introduced before.

  $\langle$*when* $r > 1\rangle$

  (A) if a transaction occurred in the r-1 th round,

– if the successful seller in the r-1 th round is seller i,

$$P^T = P^{F_r}_{r-1} + \beta_1 \times \frac{num}{NUM}, \tag{5.3}$$

where, $\beta_1 \in (10 \times V_{step}, 50 \times V_{step})$ and $P^{F_r}_{r-1}$ is the traded final price in r-1 th round. P^T is updated using this formula because if seller i can sell a good at the price of $P^{F_r}_{r-1}$, then he knows that the buyer side can accept this price and he will try to make more profit than the current round by increasing the price. The increased price is related to the number of the remaining goods to trade. If there are still many goods needed to trade, the seller would take a cautious attitude and add just a little bit to $P^{F_r}_{r-1}$ to ensure that he can still do the trade. On the other hand, if the seller has sold many goods and only a few goods are left, it means the seller traded frequently at a low price, so he can take an aggressive attitude and try to add a large value to $P^{F_r}_{r-1}$.

– else

$$P^T = max(\frac{1}{|L_r|} \sum_{rs \in L_r} P^{F_r}_{rs},\ (1 + \alpha_3) \times c_{in}), \tag{5.4}$$

where, $rs \in L_r$, L_r is the set of suffixes of rounds in which the successful transaction occurred in the past. This formula is set because of the following reason: If seller i is not the successful seller in the $r-1$ th round, he knows that the trade can be dealt with at the price of $P^{F_r}_{r-1}$, and he would rather do the trade, so seller i would not increase $P^{F_r}_{r-1}$, but refer to P^Fs in the past rs rounds and take the average of these prices as an optional choice. Because sellers always want to sell at a high price, so P^T is decided to be the larger value of the averaged price from the history information and the basic P^T derived from c_{in}.

(B) if there was no transaction in the r-1 th round,

$$P^T = max(oa_{r-1} - \beta_2 \times (1 - \frac{num}{NUM}),\ c_{in}), \tag{5.5}$$

where, $\beta_2 \in (10 \times V_{step}, 50 \times V_{step})$ and oa_{r-1} is the oa of the r-1 th round. P^T

is updated using this formula because if no one can sell a good, which means no buyer is willing to accept the price of *oa*, it shows that the price maybe too high to be accepted. Then, seller *i* will try to decrease P^T to earn the chance to sell a good by losing some potential profits. For the same reason as we mentioned in (A), the decreased price should be related to the number of the remaining goods to trade. If the seller still has many goods to trade, he will decrease P^T by a bit to be more competitive. Contrarily, if there is only a few goods left, seller *i* can save some profits by maintaining oa_{r-1} as a candidate of P^T. Because sellers always want to sell at a high price, so P^T is decided to be the larger value of the price obtained based on oa_{r-1} and the basic P^T derived from c_{in}.

Table 5.2: 10 Kinds of Judgment Nodes for GNP-RN Seller

$JN1_s$:	Seller *i* sold a good in the last round?
$JN2_s$:	$\frac{num}{NUM} = 1$ or ≥ 0.5 or < 0.5
$JN3_s$:	The *ask* seller *i* submitted is the *oa* in the last time step?
$JN4_s$:	$\frac{N^{oa}_{Si}}{L^{ts}} \geq 0.5$ or < 0.5
$JN5_s$:	$\frac{t}{T} \in (0, 1/3]$ or $(1/3, 2/3]$ or $(2/3, 1]$
$JN6_s$:	oa^t is closer to P^B, or closer to P^T but still lager than P^T, or smaller than P^T
$JN7_s$:	ob^t is smaller than c_{in}, or lager than c_{in} but closer to c_{in}, or lager than c_{in} but closer to P^T, or lager than P^T
$JN8_s$:	$oa^{t-2} - oa^{t-1} \geq (P^B - P^T)/2$, or $< (P^B - P^T)/2$
$JN9_s$:	$T_S \in [0, \frac{1}{3} \times R_J)$ or $[\frac{1}{3} \times R_J, < \frac{2}{3} \times R_J)$ or $[\frac{2}{3} \times R_J, 1]$.
$JN10_s$:	c_{in} is low, or middle, or high, according to the price range of the goods in the market.

where, the number of the time steps that seller *i* submitted *oa* in the past L^{ts} time steps is denoted by N^{oa}_{Si}, the *oa* of the current time step *t* is denoted by oa^t and the *ob* of the current time step *t* is denoted by ob^t, and in the past R_J rounds, the number of the agent's successful transactions is denoted by T_S

According to the features of CDA, 10 different kinds of judgment functions of JNs for GNP-RN seller are proposed. The items considered include: *oa*, *ob*, P^B, P^T, c_{in}, $P^F s$, the current time step *t*, the relation among the above prices and the relation between *num* and NUM. Suppose that CDA is in the *r th* round and the current time

Table 5.3: 7 Kinds of Processing Nodes for Seller

$PN1_s$:	ask the current *ob*
$PN2_s$:	ask the current $ob + \gamma_s$
$PN3_s$:	ask the last $P^F + \gamma_s$
$PN4_s$:	ask the current oa - γ_s
$PN5_s$:	stay. make no new ask.
$PN6_s$:	ask P^T.
$PN7_s$:	ask $P^T + \gamma_s$

Here, γ_s is a small price in the set of $\{V_{step}, 2 \times V_{step}, 3 \times V_{step}\}$. γ_s is set like this because GNP-RN is supposed to imitate the behavior in the real-life. In the real-life auction, if the bidders want to submit a price little higher than the current *ob* or little lower than the current *oa*, most of them might change the current *oa* or *ob* by the smallest valid bidding price. So, the value of γ_s is set according to the current private price range and the smallest valid bidding price.

step is t. GNP-RN seller i, who wants to sell NUM goods and has already sold *num* goods, is willing to sell good n ($n = num + 1$), and P^P of this good for seller i is c_{in}. So, the judgments the seller can use to judge the CDAs information are shown in Table. 5.2.

7 different kinds of bidding actions of PNs for GNP-RN seller are also proposed. So, the GNP-RN seller i can submit the ask according to the following 7 potential bidding actions in Table. 5.3 at each time step.

The final ask price submitted by seller i is obtained by the result of PN with the δ adjustment in the connected RN. Generally speaking, suppose there are *Rn* kinds of RNs, because each kind of PNs have the possibility to connect to each kind of RNs, so for one kind of PN which contains one kind of bidding action, the potential bidding options it can provide will be increased to *Rn* instead of 1 by connecting to *Rns* kinds of RNs. These 7 kinds of bidding actions provide various and flexible bidding options to the agent by using RNs, that is, the agent can bid toward to P^T gradually, and also can bid at *ob*, at the price close to *ob* or at P^T and so on.

7 different δ values, that is, $-3 \times V_{step}$, $-2 \times V_{step}$, $-1 \times V_{step}$, 0, V_{step}, $2 \times V_{step}$ and $3 \times V_{step}$ are assigned to 7 different kinds of RNs, respectively.

What kinds of information should be judged and which action should be taken are determined by the node transition of the GNP individual.

5.2.3 Bidding Strategy for Buyers

The heuristic algorithms for the bidding strategy for buyers are almost the same as the ones of sellers.

Similarly, suppose that CDA is in the *r th* round, and the current time step is t. GNP-based buyer i, who wants to buy NUM goods and has already bought *num* goods, is willing to buy good n.

For the first time step,

- P^B is given by:

$$P^B = \alpha_2 \times v_{in}, \tag{5.6}$$

  where, $\alpha_2 \in (0, 1.0)$.

- P^T for buying the good is given by:

  $\langle when\ r = 1 \rangle$

$$P^T = (1 - \alpha_4) \times v_{in}, \tag{5.7}$$

  where $\alpha_4 \in (0, 0.1)$.

  $\langle when\ r > 1 \rangle$

  (A) if a transaction occurred in the r-1 *th* round,

  – if the successful buyer in the r-1 *th* round is buyer i,

$$P^T = P^{F_r}_{r-1} - \beta_3 \times \frac{num}{NUM}, \tag{5.8}$$

  where, $\beta_3 \in (10 \times V_{step}, 50 \times V_{step})$ and $P^{F_r}_{r-1}$ is the traded final price in r-1 *th* round.

  – else

$$P^T = min(\frac{1}{|L_r|} \sum_{rs \in L_r} P^{F_r}_{rs},\ (1 - \alpha_4) \times v_{in}), \tag{5.9}$$

  The meaning of the parameters is the same as the ones in the seller part.

(B) if there was no transaction in the r-1 th round,

$$P^T = min(ob_{r-1} + \beta_4 \times (1 - \frac{num}{NUM}),\ v_{in}), \tag{5.10}$$

where, $\beta_4 \in (10 \times V_{step}, 50 \times V_{step})$ and ob^{r-1} is the ob of the r-1 th round.

Eq. (5.6), (5.7), (5.8), (5.9) and (5.10) correspond to Eq. (5.1), (5.2), (5.3), (5.4) and (5.5), respectively. The reason why these formula are used is similar to the seller side except the logics are in the opposite direction.

10 different kinds of judgment functions of JNs for GNP-RN buyer are also proposed. Suppose that CDA is in the r th round and the current time step is t. GNP-RN buyer i, who wants to buy NUM goods and has already bought num goods, is willing to buy good n (n = num+1), and P^P of this good for buyer i is v_{in}. So, the judgments the buyer can use to judge the CDAs information are shown in Table. 5.4.

Table 5.4: 10 Kinds of Judgment Nodes for GNP-RN Buyer

$JN1_b$:	Buyer i bought a good in the last round?
$JN2_b$:	$\frac{num}{NUM}$ = 1 or ≥ 0.5 or < 0.5
$JN3_b$:	The bid buyer i submitted is the ob in the last time step?
$JN4_b$:	$\frac{N^{ob}_{Bi}}{L^{ts}} \geq 0.5$ or < 0.5
$JN5_b$:	$\frac{t}{T} \in (0, 1/3]$ or $(1/3, 2/3]$ or $(2/3, 1]$.
$JN6_b$:	ob_t is closer to P^B, or closer to P^T but still lower than P^T, or larger than P^T.
$JN7_b$:	oa^t is higher than v_{in}, or lower than v_{in} but closer to v_{in}, or lower than v_{in} but closer to P^T, or lower than P^T
$JN8_b$:	$ob^{t-1} - ob^{t-2} \geq (P^T - P^B)/2$, or $< (P^T - P^B)/2$
$JN9_b$:	$T_S \in [0, \frac{1}{3} \times R_J)$ or $[\frac{1}{3} \times R_J, < \frac{2}{3} \times R_J)$ or $[\frac{2}{3} \times R_J, 1]$.
$JN10_b$:	v_{in} is low, or middle, or high, according to the price range of the goods in the market.

where, the number of the time steps that buyer i submitted ob in the past L^{ts} time steps is denoted by N^{ob}_{Bi}.

7 different kinds of bidding actions of PNs for GNP-RN buyer are also proposed as shown in Table. 5.5.

Table 5.5: 7 Kinds of Processing Nodes for Buyer

$PN1_b$:	bid the current oa
$PN2_b$:	bid the current oa - γ_b
$PN3_b$:	bid the last P^F - γ_b
$PN4_b$:	bid the current ob + γ_b
$PN5_b$:	stay. make no new *bid*.
$PN6_b$:	bid P^T.
$PN7_b$:	bid P^T - γ_b
Here, γ_b is a small price as γ_s.	

7 different kinds of *RN*s for GNP-RN buyer are set like the *RN*s for GNP-RN seller. In the same way as the seller case, the final *bid* price submitted by buyer i is obtained by the result of *PN* with the δ adjustment in the connected *RN*.

5.2.4 Fitness Function for Agents

When the CDA process is done, each GNP individual is evaluated by the fitness function. The fitness is calculated by the profit gained by the individual, which is the most common and classical way to evaluate the agent performance in the literature.

For seller individual i, the profit in CDA process is calculated by $\sum_{g \in G_i^s} (P_g^{F_g} - c_{ig})$, where, G_i^s is the set of suffixes of goods seller individual i sold and $P_g^{F_g}$ is the final price of good g. For buyer individual i, the profit in CDA process is calculated by $\sum_{g \in G_i^b} (v_{ig} - P_g^{F_g})$, where, G_i^b is the set of suffixes of goods buyer individual i bought. The way to calculate the profit as the fitness value is the most common and classical way to evaluate the agent performance in the literature. The individual with the highest profit will survive to the next generation, while the other weaker ones will have the genetic operations for creating new candidates for the optimal strategy.

5.3 Simulations

We compared the proposed GNP-RN method with ZI-U, ZI-C, GD and CP strategies, which are the most cited and commonly adopted strategies in the literature of CDAs.

Table 5.6: Parameters Setting

Items	Value
Number(N) of Goods(G)	20 to 80 (5×4 to 5×16)
N of Agents	5 sellers, 5 buyers
N of G Agents Wants to Trade	from 4 to 16
N of Training Environment	30
N of Testing Environment	10
N of Time Steps in One Round	100
P^p range for sellers	(1.0, 2.5)
P^p range for buyers	(2.0, 3.5)
V_{step},	0.01
$\alpha_1, \alpha_2, \alpha_3, \alpha_4$	1.5, 0.5, 0.05, 0.05
$\beta_1, \beta_2, \beta_3, \beta_4$	0.1, 0.3, 0.1, 0.3
L^{ts}, L_r, R_J	5, 3, 6
Generation	300
Population Size	200
—Elite (survived from last generation)	10
—Generated by Crossover	80
—Generated by Mutation	80
—Generated Randomly	30
Selection	upper 30%
Crossover Rate	0.1
Mutation Rate	0.3
Node	
—Judgment Node	50
—Processing Node	15
—Rectify Node	15
—Start Node	1

5.3.1 Basic Study

3 groups of simulations are carried out: 1) The first group is to compare the performance of GNP-RN agent and the other agents adopting other 4 strategies when all the agents on the other side use the same strategy; 2) In order to compare the performance of GNP-RN with conventional GNP in CDAs, the second group has the same setting as the first group, except the GNP-based agent uses the conventional GNP strategy without *RN*s. The profits obtained by GNP agent are compared to the profits obtained by GNP-RN agent; 3) The third group is also designed for comparing the performance of GNP-RN agent and the other agents adopting other 4 strategies, but the agents on the other side use different strategies. Through all the simulations, it is demonstrated that

GNP-RN agent performs more competitively and significantly than the agents using other strategies.

5.3.1.1 Simulations Setting

To evaluate the behaviors of 5 kinds of agents using different strategies, the following two cases are considered in each group of simulations:

Seller Case: One of the 5 strategies is used for one of the 5 seller agents. Each seller is assumed to have 4-16 units of goods to sell, while each buyer is assumed to want 10 units of goods to buy. So, the supply of the CDAs is from 20-80, and the demand is 50. The profits obtained by each seller are compared.

Buyer Case: Similarly, one of the 5 strategies is used for one of the 5 buyer agents. Each buyer is assumed to want 4-16 units of goods to buy, while each seller is assumed to have 10 units of goods to sell. So, the supply of the CDAs is 50, and the demand is from 20-80. The profits obtained by each buyer are compared.

For each pair of supply and demand, 30 runs are carried out. An agent's profit is calculated by the averaged value over the 30 runs. In addition, under this setting, in both seller case and buyer case, the CDAs markets can experience 3 conditions: supply equals to demand, supply larger than demand and supply smaller than demand, which is better for studying and comparing the performance of different kinds of agents. For all sellers, the range of P^P of each good is (1.0, 2.5), which is derived from the special normal distribution of $N(1.75, 1.00)$ with the data more than 2.5 and less than 1.0 being omitted. For all buyers, the range of P^P of each good is (2.0, 3.5), which is derived from the special normal distribution $N(2.75, 1.00)$ with the data more than 3.5 and less than 2.0 being omitted. These range setting for private prices is quite common in the literature for CDAs. The private price ranges used for experiments usually are in the range of $(0.5, 4.0)$. According to the ranges of P^P, the smallest bidding price is 0.01, so the price values δ contained in RNs are in the set of {-0.03, -0.02, -0.01, 0.00, 0.01, 0.02, 0.03}.

In each generation, GNP population has 200 individuals, and GNP-based agents will evolve for 300 generations. There are 30 environments for the training and 10 for testing in order to avoid the loss of generality. In each environment, P^P of each good for each seller and P^P of each good for each buyer are different. All the testing results for each kind of strategies are the total profits of the 10 environments. To make it more clear, for example, in seller case, when the current supply and demand pair is (55, 50), which means each seller wants to sell 11 units of goods and buyer wants to buy 10 units of goods, one CDA environment includes 50 rounds, and there are 30 continuous CDAs environments for an agent to participate in, which means at most $50 \times 30 = 1500$ rounds to participate for each run. Finally, in the last generation, the GNP individual that can handle these 30 CDAs best is chosen to do the testing, and in the testing, each kind of seller participates in 10 new and different CDAs, and the total profits obtained by each kind of seller over 10 CDAs are recorded. After 30 runs, under the same supply and demand condition, the averaged profit of each seller is regarded as this seller's profit.

The more specific parameters used in the simulations are shown in Table. 5.6. The values of α_1, α_2, α_3 and α_4 are 1.5, 0.5, 0.05 and 0.05, respectively, which are set based on the common believes. α_1 and α_2 are only used to determine the basic price, so they are just simply decided according to the experience in the real-life, for example, if a buyer private price for a good is \$100, it is reasonable to assume that the bidder bid \$50 at the first time step. α_3 and α_4 are only used to determine the target price in the first round of each CDA, so it is reasonable to assume that the percentage of the profit margin of each bidder is 5% to their private price. According to the ranges of P^P, β_1, β_2, β_3 and β_4 are set at 0.1, 0.3, 0.1 and 0.3, respectively. If the values are too small, there is little effect for updating the target price, while if the values are too large, the profit becomes unstable, even gets worse because invalid target prices are generated.

Table 5.7: Averaged number of rounds that agent makes the first trade when supply = demand = 50

	GNP	ZI-C	ZI-U	CP	GD
Seller side	26.84	7.71	4.32	22.1	22.87
Buyer side	33.84	5.85	1.16	22.2	23.52

L^{ts}, L_r and R_J are set to be 5, 3 and 6, respectively, because if they are smaller than these values, the simulation results become very unstable, while if they are larger than these values, the results show no obvious difference and even show worse performances if these values get much larger.

Table 5.8: Simulation 1: Averaged number of goods that agents can trade under different conditions.

	Seller Case (Demand=50)												
	4	5	6	7	8	9	10	11	12	13	14	15	16
GNP	4.00	5.00	6.00	7.00	8.00	9.00	9.99	10.51	11.81	12.67	13.33	14.07	14.84
ZI-C	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	11.99	12.95	13.97	14.85	14.27
ZI-U	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	12.00	13.00	14.00	15.00	16.00
CP	4.00	5.00	6.00	7.00	8.00	9.00	10.00	9.93	8.88	7.31	5.87	3.73	1.75
GD	4.00	5.00	6.00	7.00	8.00	9.00	9.72	7.40	5.30	4.07	2.90	2.35	2.57
	Buyer Case (Supply=50)												
	4	5	6	7	8	9	10	11	12	13	14	15	16
GNP	4.00	5.00	6.00	7.00	7.99	8.96	9.97	11.00	11.95	12.61	13.37	14.13	12.87
ZI-C	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	12.00	13.00	14.00	14.69	15.97
ZI-U	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	12.00	13.00	14.00	15.00	16.00
CP	4.00	5.00	6.00	7.00	8.00	9.00	10.00	10.83	11.71	9.97	7.72	4.51	4.95
GD	4.00	5.00	6.00	7.00	8.00	9.00	10.00	6.17	2.33	1.43	0.91	0.67	0.21

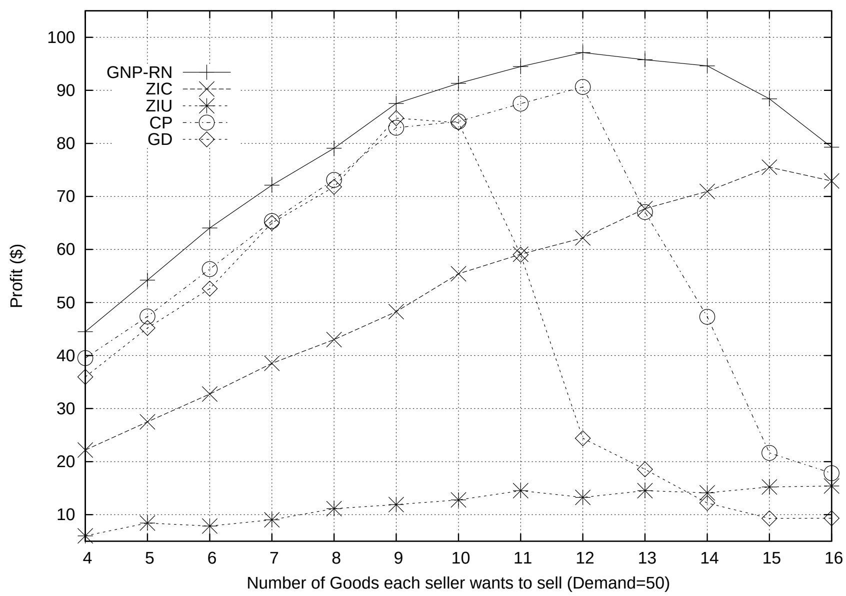
5.3.1.2 Simulation 1

Simulation group 1 is conducted to evaluate the performance of each kind of agents when all the agents on the other side use ZI-C strategy. So, the agents on the other side are randomly bidding bidders.

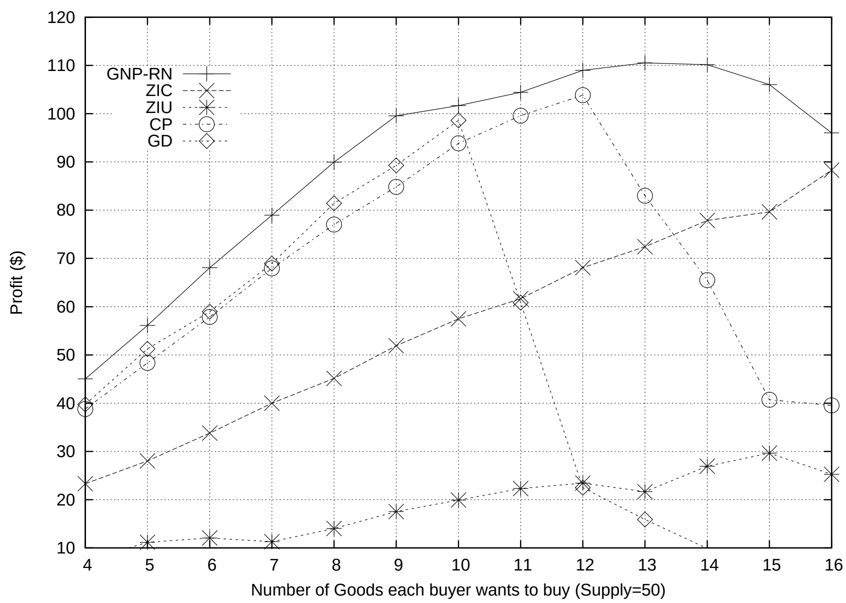
Fig. 5.3 shows the simulation results on how much profit each agent can get competing with each others under different market conditions. The results are the total profits obtained by agents from 10 testing environments, which is the averaged value over 30 runs. Table. 5.7 studies at which round each strategy can make the first trade, which shows the averaged round number under the condition that all of the traders want to trade 10 goods as an example. Table. 5.8 shows the average number of goods that agents can trade on the seller side and buyer side under different conditions.

From Fig. 5.3, it can be found that GNP-RN strategy outperforms other strategies under all the simulated market conditions, no matter the supply is more than, equals to, or less than demand.

Fig. 5.3(a) shows that when the demand is unchanged and more than supply, the profit that a seller can get increases as the number of good he wants to trade increases. This is because when the demand is more than supply, the traders can get enough chances to make the trade because the seller side is less competitive, so it is easier for a seller to trade all the goods with profit, thus the sellers' profit increases when the number of good he wants to sell increases. Specifically, for ZI-U and ZI-C trader, as Table. 5.7 and Table. 5.8 indicate, there is a high possibility that ZI-U and ZI-C



(a)Performance of 5 different sellers against ZI-C agents



(b)Performance of 5 different buyers against ZI-C agents

Figure 5.3: Against ZI-C agents

accept the price at a very unreasonable level, so they usually can trade good faster, but obtain less profit. CP trader has an updating rule to adapt to the environment, so it can outperform ZI-U and ZI-C, but it only use the last one round information. GD trader relies on the history information, and since the chance for trading is enough, GD can perform well by using the information of the successful transactions carried out by CP, GD and GNP-RN. For GNP-RN traders, it can perform best in all of these situations because the evolved compact directed structure learns the auctions adequately using the history information and heuristic algorithms. From the simulation data, it can be found that, when GNP-RN learns that the demand is larger than supply, in the other words, when GNP-RN agent believes that it is not difficult to trade all the goods it wants to trade, it becomes inclined to bid from the basic price and increase its bidding price slowly and slightly, even keep stay action to force the other side to decrease the bidding price, that is why GNP-RN can obtained more profit than the other strategies.

Contrarily, when the demand is unchanged and less than supply, the profit that a seller can get decrease as the number of good he wants to trade increases. This is because when the demand is less than supply, the number of trading chances in the market is limited, so the seller side becomes more competitive, and it becomes harder for a seller to make trades. From Fig. 5.3(a), it can be found GNP-RN is superior than other strategies, and keeps performing very well even where the supply become much more than demand. Specifically, for ZI-U and ZI-C trader, as explained above, they usually can trade good faster and obtain less profit. The trading chances of CP and GD are snatched by ZI-C and ZI-U, and there is less useful history information. For GNP-RN seller, once it learns the situation, it becomes inclined to accept the bid as soon as the bid price is higher than its target price to ensure it can make trade in the competitive situation, therefore, GNP-RN can trade more times than CP and GD, and obtain much higher profit than ZI-U and ZI-C although it seems that ZI-U and ZI-C can trade more goods. However, it is shown that along with the increasing intensity of the competition, the difference between GNP-RN and ZI-C is getting less due to the decreasing opportunity for trading goods. In order to make sure to trade goods, GNP-RN is inclined to accept a less profitable price, which behaves like ZI-C partly.

Similarly, the same phenomena are also found from the simulations for buyer case, which can be explained in the same way as the above. Under these kinds of situations of seller case and buyer case, GNP-RN bidding strategy can have a good performance by adapting to the changing environments of the supply and demand and limited trading chances compared to other strategies because of the flexible bidding choices and well understanding of the information. Because ZI-C and ZI-U use the random strategy and less consider the profit, the changes of the environment do not have much effect on their bidding actions.

As is shown in Table. 5.9, the *p* values of the t-test (two-tailed) show there are statistically significant differences between GNP-RN strategy and other strategies under different situations (at 5% significant level).

5.3.1.3 Simulation 2

Simulation Group 2 is done under the same conditions as simulation group 1, except the GNP-RN strategy is changed to the conventional GNP strategy, which has no *RN*s. The performance of GNP agent is compared to the performance of GNP-RN, which are obtained from the results of simulation group 1.

Fig. 5.4 shows that when the amount of goods needed to trade is more than the other side, which mean the agent is under a more competitive situation, GNP-RN can outperform conventional GNP by using *RN*s. GNP-RN agent has the ability to make more profit margins from the target price and keep the chance to trade goods.

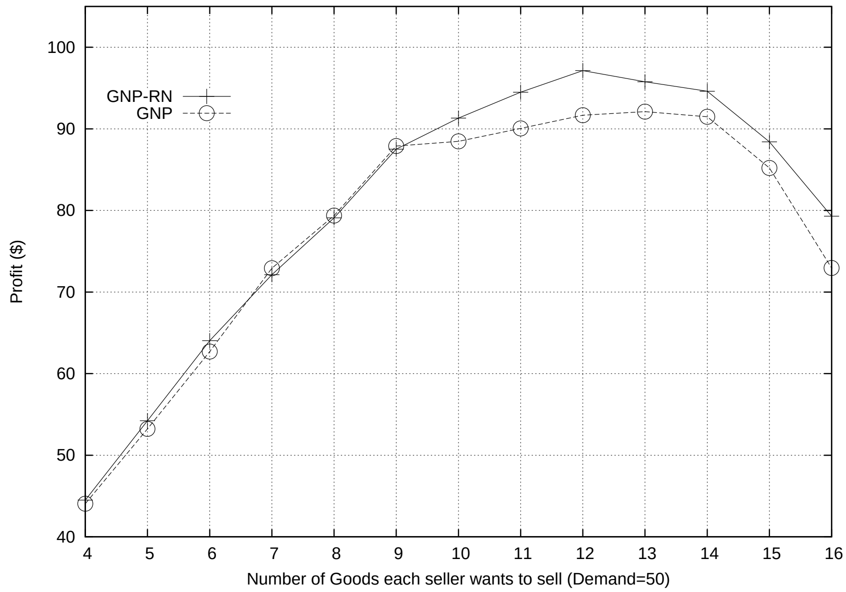
As is shown in Table. 5.10, the *p* values of the t-test (two-tailed) show there are statistically significant differences between the GNP-RN strategy and conventional GNP strategy when they are under the competitive situations (at 5% significant level).

5.3.1.4 Simulation 3

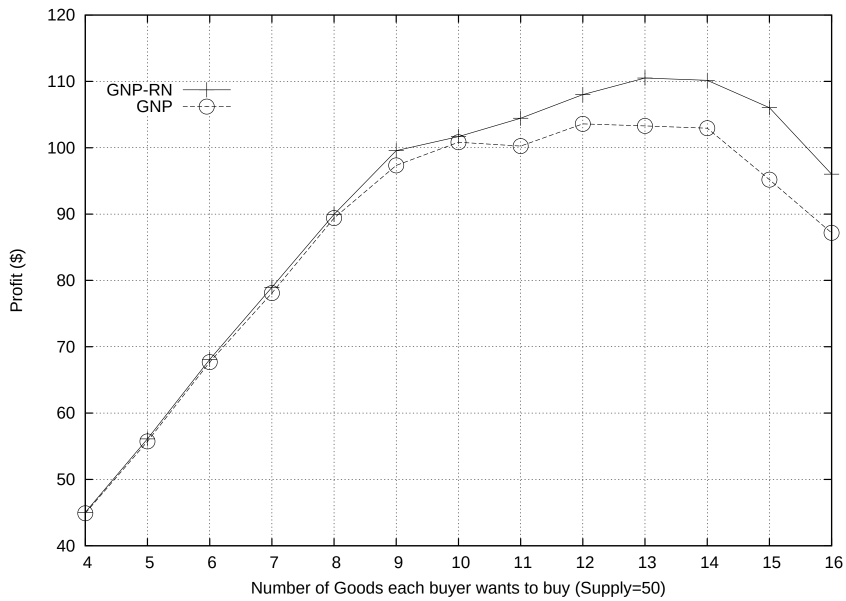
Simulation group 3 is conducted to evaluate the performance of each kind of agents when the agents on the other side use different strategies, and in order to study more comprehensively, it is composed of 2 parts: part *A* and part *B*, which are different from simulation group 1, where all the agents on the other side use ZI-C strategy,

- in part *A*, each agent on the other side randomly chooses a strategy from ZI-C, CP and GD. So, the agents on the other side are composed of not only randomly bidding bidders but also the bidders with heuristics, which enables to reach their private price gradually. The other setting is the same as simulation group 1.

- in part *B*, each agent on the other side randomly chooses a strategy from CP and GD. So, the agents on the other side are only composed of heuristics-based

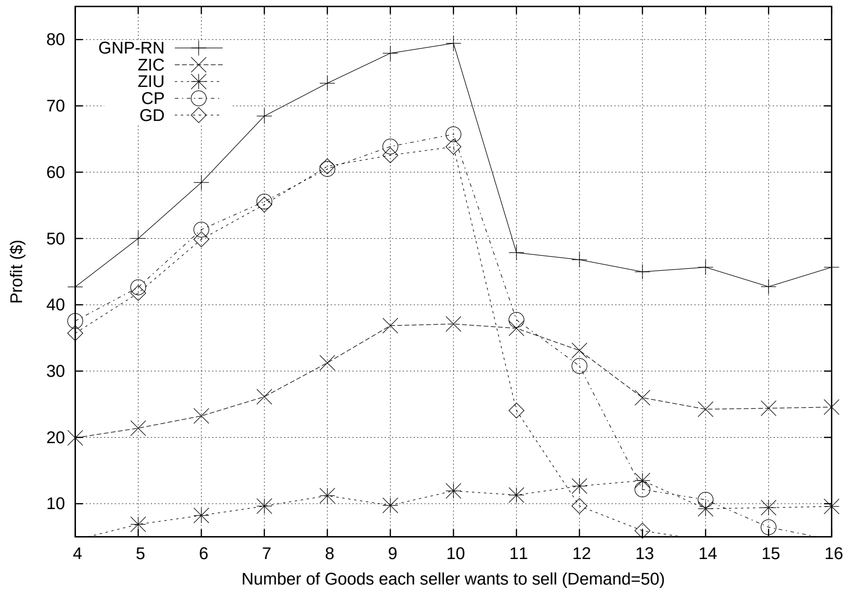


(a)Performance of GNP-RN and GNP sellers

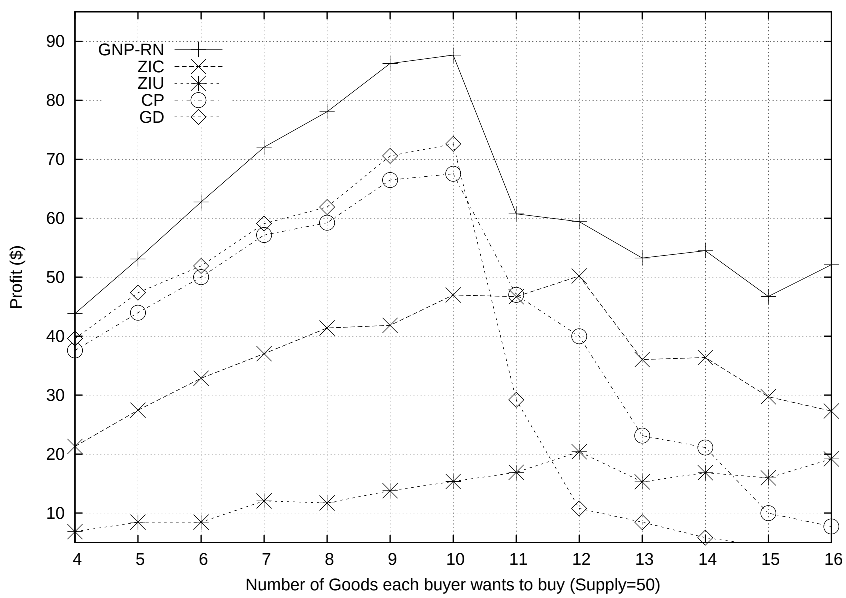


(b)Performance of GNP-RN and GNP buyers

Figure 5.4: Comparison between GNP-RN and GNP

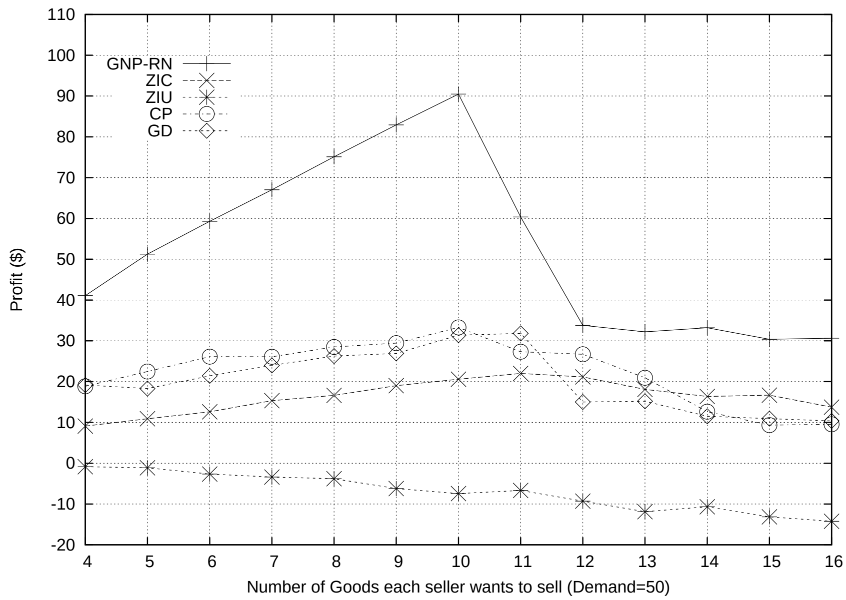


(a)Performance of 5 different sellers

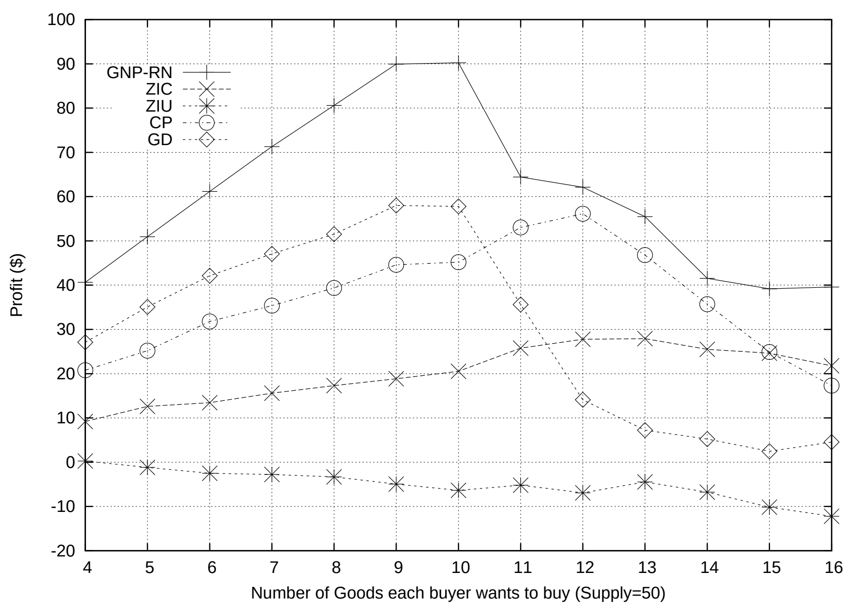


(b)Performance of 5 different buyers

Figure 5.5: Part *A*: Against bidders using ZI-C, CP and GD



(a)Performance of 5 different sellers



(b)Performance of 5 different buyers

Figure 5.6: Part *B*: Against bidders using CP and GD

bidders, which have no randomly bidding bidders. The other setting also is the same as simulation group 1.

Fig. 5.5 and Fig. 5.6 show the profit obtained by each agent under different market conditions. Table. 5.11 shows the average number of goods that each kind of agents can trade. The results clearly show that, in both seller case and buyer case, GNP-RN still give the best performance when the other side uses different strategies, and because the superiority of GNP-RN is obvious, the p values of the t-test of this section are not given. The results emphasize that GNP-RN can adapt to the various situations, even if the competition becomes more intensive, and GNP-RN can deal with the situation better than CP and GD and can obtain more profits. In addition, it is clear from Fig. 5.5 and Fig. 5.6 that, when CP and GD agents are the less competitive cases, they can perform much better than ZI-C and ZI-U agents, while when the competition become more intensive, the profits obtained by CP and GD agents decrease because of losing chances to trade, which is consistent with the phenomena observed in simulation group 1. Additionally, there can be found a sharp drop in profit of GNP-RN, CP and GD when they turn to the competitive situation. It can be explained by the change of the relation between the demand and supply. When the opponents include intelligent strategies, not just randomly bidding strategy, if the market turns to be more competitive, it is normal for the competitive side to get less profits by trading the same number of goods.

5.3.2 Extended Study

After studying and analyzing each strategies' performance in the 3 group of simulations, this section established the simulations, where there are more agents participating in the auction environment. The simulation settings are the same as the previous simulations except there are N agents on each side of the auction, where, N is larger than 5. In the same way, there are two cases considered:

Seller Case: There are N sellers and N buyers. One of the 5 strategies is used for one of the seller agents. All the other agents use CP strategy in order to avoid being non-intelligent. All the agents want to trade 4 units of goods. The profits obtained by each seller with different strategies are compared.

Buyer Case: Similarly, there are N sellers and N buyers. One of the 5 strategies is used for one of the buyer agents. All the other agents use CP strategy. All the

Table 5.9: The t-test (p value) results of GNP-RN and other strategies in different situations

[Simulation Group 1, Seller case]

	4		5		6		7		8	
	GNP-RN	CP	GNP-RN	CP	GNP-RN	CP	GNP-RN	CP	GNP-RN	CP
mean	44.50	39.53	54.22	47.36	64.05	56.30	72.12	65.36	79.10	73.13
standard deviation	0.1128	0.1034	0.1077	0.1782	0.1819	0.1844	0.2017	0.2539	0.2091	0.2302
t-test(p value)	5.01×10^{-25}	—	6.16×10^{-23}	—	1.39×10^{-23}	—	4.72×10^{-17}	—	1.11×10^{-16}	—
	9		10		11		12			
	GNP-RN	CP	GNP-RN	CP	GNP-RN	CP	GNP-RN	CP		
mean	87.51	82.95	91.31	84.14	94.49	87.51	97.14	90.64		
standard deviation	0.2264	0.3437	0.2151	0.3748	0.3434	0.7602	0.3579	0.8319		
t-test(p value)	2.84×10^{-5}	—	3.95×10^{-12}	—	2.33×10^{-5}	—	9.71×10^{-5}	—		

	4		5		6		7		8	
	GNP-RN	GD	GNP-RN	GD	GNP-RN	GD	GNP-RN	GD	GNP-RN	GD
mean	44.50	35.98	54.22	45.19	64.05	52.61	72.12	65.00	79.10	71.80
standard deviation	0.1128	0.1813	0.1077	0.3052	0.1819	0.3417	0.2017	0.3915	0.2091	0.4538
t-test(p value)	2.24×10^{-29}	—	7.68×10^{-17}	—	1.86×10^{-19}	—	8.25×10^{-12}	—	1.69×10^{-10}	—
	9		10							
	GNP-RN	GD	GNP-RN	GD						
mean	87.51	84.77	91.31	83.91						
standard deviation	0.2264	0.3840	0.2151	0.5194						
t-test(p value)	1.05×10^{-5}	—	3.34×10^{-7}	—						

[Simulation Group 1, Buyer case]

	4		5		6		7		8	
	GNP-RN	CP	GNP-RN	CP	GNP-RN	CP	GNP-RN	CP	GNP-RN	CP
mean	45.04	38.79	56.11	48.35	68.08	57.88	78.97	67.94	89.95	77.03
standard deviation	0.1145	0.1051	0.0907	0.1421	0.0952	0.2593	0.1361	0.2379	0.2260	0.1731
t-test(p value)	4.78×10^{-28}	—	5.36×10^{-29}	—	5.12×10^{-24}	—	1.73×10^{-28}	—	3.76×10^{-35}	—
	9		10		11		12			
	GNP-RN	CP	GNP-RN	CP	GNP-RN	CP	GNP-RN	CP		
mean	99.57	84.83	101.67	93.87	104.44	99.57	108.99	103.84		
standard deviation	0.2887	0.2329	0.2938	0.1892	0.3340	0.3688	0.3926	0.3910		
t-test(p value)	5.43×10^{-32}	—	3.64×10^{-18}	—	3.32×10^{-7}	—	1.10×10^{-7}	—		

	4		5		6		7		8	
	GNP-RN	GD	GNP-RN	GD	GNP-RN	GD	GNP-RN	GD	GNP-RN	GD
mean	45.04	39.75	56.11	51.25	68.08	58.97	78.97	68.95	89.95	81.43
standard deviation	0.1145	0.1815	0.0907	0.1857	0.0952	0.3094	0.1361	0.2330	0.2260	0.2097
t-test(p value)	2.06×10^{-18}	—	1.89×10^{-19}	—	2.96×10^{-18}	—	7.99×10^{-28}	—	1.69×10^{-27}	—
	9		10							
	GNP-RN	GD	GNP-RN	GD						
mean	99.57	89.30	101.67	98.62						
standard deviation	0.2887	0.3311	0.2938	0.2242						
t-test(p value)	2.95×10^{-22}	—	4.98×10^{-9}	—						

Table 5.10: The t-test (p value) results of GNP-RN and GNP in different situations

[Simulation Group 2, Seller case]

	11		12		13	
	GNP-RN	GNP	GNP-RN	GNP	GNP-RN	GNP
mean	94.49	90.05	97.14	91.67	95.77	92.11
standard deviation	0.3434	0.5441	0.3579	0.2967	0.3650	0.5518
t-test(p value)	8.36×10^{-4}	—	6.46×10^{-9}	—	2.32×10^{-3}	—
	14		15		16	
	GNP-RN	GNP	GNP-RN	GNP	GNP-RN	GNP
mean	94.61	91.48	88.41	85.20	79.30	72.95
standard deviation	0.3998	0.3094	0.4422	0.4967	0.4747	0.4913
t-test(p value)	2.04×10^{-3}	—	8.98×10^{-3}	—	4.41×10^{-6}	—

[Simulation Group 2, Buyer case]

	11		12		13	
	GNP-RN	GNP	GNP-RN	GNP	GNP-RN	GNP
mean	104.44	100.25	108.99	103.60	110.51	103.28
standard deviation	0.3340	0.3506	0.3926	0.3195	0.3548	0.8639
t-test(p value)	1.37×10^{-5}	—	1.54×10^{-6}	—	8.05×10^{-6}	—
	14		15		16	
	GNP-RN	GNP	GNP-RN	GNP	GNP-RN	GNP
mean	110.16	102.94	106.03	95.18	96.02	87.17
standard deviation	0.3181	0.9922	0.3598	0.5714	0.7309	0.5043
t-test(p value)	8.63×10^{-4}	—	1.46×10^{-11}	—	2.10×10^{-6}	—

agents want to trade 4 units of goods. The profits obtained by each buyer with different strategies are compared.

Here, N is considered to be 10 and 100 as typical situations. All the other settings are the same as introduced in section 5.3.1.

Table. 5.12 shows the averaged profits obtained by each strategy when there are more than 5 agents competing on each side of the auction.

5.4 Conclusions

The bidding strategy developed by GNP for CDA agents has been proposed in this chapter to obtain a good guidance for the intelligent auction systems. The GNP based agent can find the generalized optimal strategies which suit for many environments.

It is found from the simulations comparing with the conventional auction agents that the use of GNP to choose the suitable functions for the bidding is more flexible for various situations of auctions due to its evolutionary features and well organized structures.

Table 5.11: Simulation 3: Averaged number of goods that agents can trade under different conditions.

Part *A*

	Seller Case (Demand=50)												
	4	5	6	7	8	9	10	11	12	13	14	15	16
GNP	4.00	5.00	6.00	7.00	8.00	9.00	9.94	10.47	11.82	12.59	13.17	13.92	14.92
ZI-C	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	12.00	12.94	14.00	14.67	14.22
ZI-U	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	12.00	13.00	14.00	15.00	16.00
CP	4.00	5.00	6.00	7.00	8.00	9.00	10.00	10.45	9.11	7.19	5.15	3.69	1.48
GD	4.00	5.00	6.00	7.00	8.00	9.00	9.69	6.90	5.05	4.27	2.65	2.71	3.38

	Buyer Case (Supply=50)												
	4	5	6	7	8	9	10	11	12	13	14	15	16
GNP	4.00	5.00	6.00	7.00	7.97	8.96	9.97	9.79	11.64	12.31	13.20	13.84	13.36
ZI-C	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	12.00	13.00	13.99	15.00	15.84
ZI-U	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	12.00	13.00	14.00	15.00	16.00
CP	4.00	5.00	6.00	7.00	8.00	9.00	10.00	10.79	10.07	8.51	5.80	4.03	4.96
GD	4.00	5.00	6.00	7.00	8.00	9.00	10.00	7.43	4.29	3.18	3.01	2.13	1.04

Part *B*

	Seller Case (Demand=50)												
	4	5	6	7	8	9	10	11	12	13	14	15	16
GNP	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	10.42	10.79	12.97	13.31	13.67
ZI-C	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	9.92	9.08	7.87	7.78	6.82
ZI-U	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	12.00	13.00	14.00	15.00	16.00
CP	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	11.71	10.63	9.45	7.93	8.67
GD	4.00	5.00	6.00	7.00	8.00	9.00	10.00	6.00	5.95	6.5	5.71	5.98	4.84

	Buyer Case(Supply=50)												
	4	5	6	7	8	9	10	11	12	13	14	15	16
GNP	4.00	5.00	6.00	7.00	8.00	8.93	9.91	10.95	11.65	10.33	11.60	13.31	14.62
ZI-C	4.00	5.00	6.00	7.00	8.00	9.00	10.00	10.84	11.78	12.06	11.63	11.58	10.54
ZI-U	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	12.00	13.00	14.00	15.00	16.00
CP	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	11.99	12.87	11.10	9.39	7.44
GD	4.00	5.00	6.00	7.00	8.00	9.00	9.70	6.21	2.58	1.74	1.67	0.92	1.40

Table 5.12: Averaged profits obtained by each strategy when there are a large number of agents on each side

	GNP	ZI-C	ZI-U	CP	GD
Seller side, N=10	3.14	1.35	0.93	1.47	2.69
Seller side, N=100	2.80	1.28	1.17	1.66	2.14
Buyer side, N=10	2.77	1.21	0.98	1.61	2.62
Buyer side, N=100	2.76	1.22	1.01	2.10	2.35

Chapter 6

Bidding Strategy Acquisition with Heuristic Rules for large-scale CDA using GNP with Adjusting Parameters

6.1 Introduction

This chapter studied and discussed a bidding strategy developed by GNP with adjusting parameters for autonomous software agents in agent-based large-scale CDAs (GNP-AP). Based on GNP-RN introduced in chapter 5, the parameters for helping to select the right decision are adjusted during the evolution in order to get more profits for large-scale CDAs. In the experiments, we studied and discussed the performance of the proposed bidding strategies, and compared it with other classic bidding strategies and previous GNP-RN strategy in a large-scale CDA under different settings.

6.1.1 Motivation

As introduced, in the last few years, there has been an explosive development of electronic commerce both in industrial and commercial domains. The amount and categories of the trades conducted electrically have grown extraordinarily with widespread Internet usage. One supplier can sell 10,000 goods online in a single day is not rare anymore, in other words, online trade with a large amount of goods occurs much more frequently nowadays. When it turns to the large-scale CDA environment, much more history information can be obtained than when the trading amount is small. These history data can provide more sensitive and general information about the CDA situation. Thus, enhancing the capability of judging abundant information and using it better is significant for the bidding strategy to improve its effectiveness and efficiency in large-scale CDA.

In GNP-RN strategy [82], pre-designed fixed values, which are decided by the experience, are used as thresholds when GNP-RN agents judge the environment information to decide which action to take next. But, in large-scale CDA facing much more complex information, pre-designed parameters are no longer the most robust and efficient ones to get better performances, therefore better values should be explored. In order to be more flexible, intelligent and adaptive to the various situations in large-scale CDA, a bidding strategy using GNP with adjusting parameter (GNP-AP) is proposed.

6.1.2 Major points

In GNP-AP, as the name implies, the parameters are adjusted during the evolution process by evaluating the agents' performance instead of the fixed values. The parameter values finally obtained are regarded as the most effective thresholds for helping the agent to choose the following actions.

Compared to the previous GNP-RN bidding strategy, the proposed strategy has the following features:

- The environment is changed from small scale to large scale, which is better fitted to the characteristic of CDA, where the supply and demand are balanced by the related history information in the market.
- The parameters in GNP-AP are adjusted during the evolution for determining the best bidding price for a given situation of CDA.
- The structure of GNP-AP is designed to be more comprehensive, where the number of branches of judgment nodes is increased to adapt to the complicated environment situations.
- Since a lot of trading price data are handled in large-scale CDAs, new kinds of judgments are employed to analyze the movement of the trading prices.

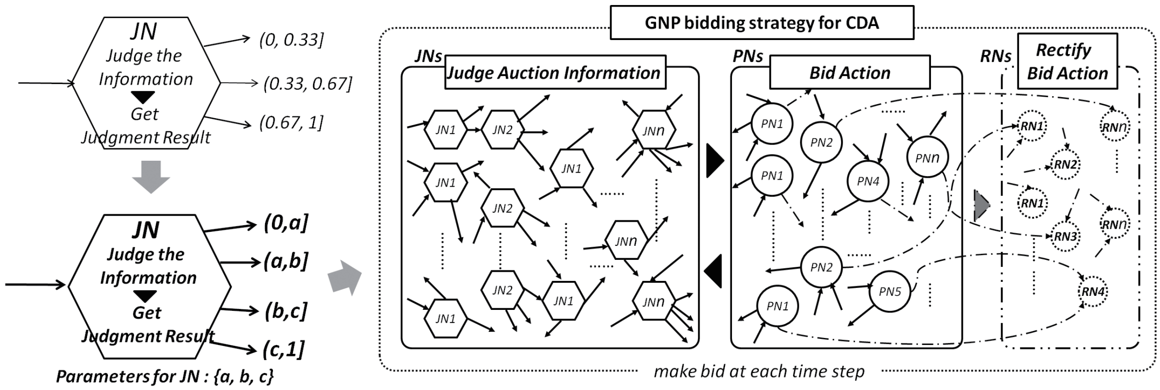


Figure 6.1: Structure of CDA using GNP-AP

Section 6.2 introduced detailed GNP-AP and the specific nodes' functions of the proposed bidding strategy. Simulation results of GNP-AP under different market conditions are shown in section 6.3. Section 6.4 concludes the studies.

6.2 Bidding strategy using GNP with adjusting parameter (GNP-AP)

6.2.1 Proposed GNP-AP structure

We've studied strategy developed by GNP in MREA and small-scale CDAs [78; 80; 82; 84]. These previous studies have revealed the effectiveness of GNP on guiding bidder's behaviors. Based on the previous research, this chapter aims to improve the flexibility and comprehensiveness of the bidding strategy, especially for large-scale CDAs by using GNP with adjusting parameters.

The same as GNP for MREA and GNP-RN, in GNP-AP, different node functions including judgment functions and action functions are proposed depending on the features of CDA, and the node transition describes the agents' bidding rules under certain auction situations. Similarly, the nodes of GNP-AP individuals also have the following 4 different kinds: (1) A Start Node (SN), (2) Judgment Node (JN), (3) Processing Node (PN) (4) Rectify Node (RN). Most of the nodes functions of PNs and RNs in GNP-AP are the same as the functions in GNP-RN. The main improvement is done for JNs: (1) the parameters are adjusted during the evolution, (2) new kinds of JNs are

Table 6.1: Heuristic rules for updating P^B and P^T

<table>
<tr><td></td><td>Heuristic Rules for GNP-based Seller</td><td>Heuristic Rules for GNP-based Buyer</td></tr>
<tr><td>first time step</td><td>$P^B = \alpha_1 \times c_{in}$</td><td>$P^B = \alpha_2 \times v_{in}$</td></tr>
<tr><td>when $r = 1$</td><td>$P^T = (1 + \alpha_3) \times c_{in}$</td><td>$P^T = (1 - \alpha_4) \times v_{in}$</td></tr>
<tr><td rowspan="6">when $r > 1$</td><td colspan="2">$Situation\ A$-1: if a transaction occurred in the r-1 th round, the successful agent is agent i</td></tr>
<tr><td>$P^T = P^F_{r-1} + \beta_1 \times \frac{num}{NUM}$</td><td>$P^T = P^F_{r-1} - \beta_3 \times \frac{num}{NUM}$</td></tr>
<tr><td colspan="2">$Situation\ A$-2: if a transaction occurred in the r-1 th round, the successful agent is not agent i</td></tr>
<tr><td>$P^T =$
$max(\frac{1}{|R_s|}\sum_{rs \in R_s} P^F_{rs},\ (1 + \alpha_3) \times c_{in})$</td><td>$P^T =$
$min(\frac{1}{|R_s|}\sum_{rs \in R_s} P^F_{rs},\ (1 - \alpha_4) \times v_{in})$</td></tr>
<tr><td colspan="2">$Situation\ B$: if there was no transaction in the r-1 th round</td></tr>
<tr><td>$P^T =$
$max(oa^{r-1} - \beta_2 \times (1 - \frac{num}{NUM}),\ c_{in})$</td><td>$P^T =$
$min(ob^{r-1} + \beta_4 \times (1 - \frac{num}{NUM}),\ v_{in})$</td></tr>
</table>

where, $\alpha_1 \in (1, 1.5)$, $\alpha_2 \in (0, 1.0)$, and $\alpha_3, \alpha_4 \in (0, 0.1)$. $\beta_1, \beta_2, \beta_3, \beta_4 \in (0.1, 0.5)$. P^F_{r-1} is the traded final price in r-1 *th* round. $rs \in R_s$, R_s is the set of suffixes of rounds in which the successful transactions occurred in the past. oa^{r-1} is the oa of the r-1 *th* round. ob^{r-1} is the ob of the r-1 *th* round.

designed appropriately for better judging the environments.

In detail, as Fig.6.1 shows, parameters (0.33, 0.67) divide the situations into 3 domains for choosing the branch and they are decided in advance and fixed during the evolution in GNP-RN. GNP-AP is proposed in order to be more adaptive to the dynamical CDA environment. Firstly, if the information with continuous values is judged, JNs usually have 3 outgoing branches in the GNP-RN method, while in GNP-AP, the number of branches is changed to 4 to divide the situations into more domains because the number of goods is increased, which strengthens the requirement of the strategy's correctness and adaptability to the various environments. So, there proposed 3 parameters, i.e., $\{a, b, c\}$ in each kind of JN. Secondly, the parameters used for JNs can be adjusted during the evolution to obtain the optimal values. Each kind of JN has different parameters, so if there are L kinds of JNs, then there are 3*L parameters in one individual. The values of $\{a, b, c\}$ are normalized into the range of (0, 1). For initialization of $\{a, b, c\}$, the parameters of one third of the GNP-AP population are set at $\{0.25, 0.50, 0.75\}$, while the parameters of the other two third are set at random values under the condition that $a \in (0.10, 0.40)$, $b \in (0.35, 0.65)$, $c \in (0.60, 0.90)$ and $a \prec b \prec c$.

GNP-AP uses 4 kinds of nodes to generate various potential bidding strategies. Aiming to be competitive for trading goods, JNs collect and judge the auction information and choose the suitable outgoing branch depending on the judge results, while PNs and RNs perform the suitable bidding actions at each time step. Individuals are evaluated by the fitness function in the evolution after the individuals in the GNP-AP population complete CDA. The elite individuals with higher fitness values can survive to the next generation and the other lower ones are replaced by the new ones generated by crossover and mutation. These genetic operations will be executed in every generation until the terminal condition meets. In GNP-AP, in order for the parameters in JNs to be adjusted during the evolution, both GNP structure and parameters will do the genetic operations for every individual. The genetic operations will be executed with a certain crossover rate and mutation rate to both of the structure and parameters. For the structure, both of crossover and mutation will be done for the nodes' connections; for the parameters, only mutation will be done for changing the value of each parameter, actually, the value of the parameters can be changed by 0.02, 0.01, −0.01 or −0.02, whose probability is 25%.

6.2.2 Bidding Strategy for CDA Bidders

Firstly, as Table.6.1 shows, the heuristics derived based on the common believes [48] [65] [71], which are proposed in GNP-RN, are also used in GNP-AP for updating P^B and P^T to guide the bidding actions.

The basic logic for the heuristic rules are briefly reviewed as follows [82]:

- Each GNP-AP bidder wants to trade goods as many as possible and gain profits as much as possible. The bidders will never bid over and under its P^P to avoid losing profits.

- At the very beginning of the auctions, as shown in Table.6.1, the bidder is willing to submit P^B which has a large difference from its P^P because of the expectation about earning money and also because there is no need to submit a very competitive price at the first time step.

Table 6.2: Information judged by 13 kinds of judgment nodes for GNP-AP agent

	GNP-AP seller	GNP-AP buyer
$JN1$, $\{a_1, b_1, c_1\}$, 4 branches:	$\frac{t}{T}$	$\frac{t}{T}$
$JN2$, $\{a_2, b_2, c_2\}$, 4 branches:	$\frac{N^{oa}_{Si}}{L^{ts}}$	$\frac{N^{ob}_{Bi}}{L_{ts}}$
$JN3$, $\{a_3, b_3, c_3\}$, 4 branches:	$\frac{num}{NUM}$	$\frac{num}{NUM}$
$JN4$, $\{a_4, b_4, c_4\}$, 4 branches:	$\frac{oa_{t-2}-oa_{t-1}}{P^B-c_{in}}$	$\frac{ob_{t-1}-ob_{t-2}}{v_{in}-P^B}$
$JN5$, $\{a_5, b_5, c_5\}$, 4 branches:	$\frac{P_{MAX}-c_{in}}{P_{MAX}-P_{MIN}}$	$\frac{v_{in}-P_{MIN}}{P_{MAX}-P_{MIN}}$
$JN6$, $\{a_6, b_6, c_6\}$, 4 branches:	$\frac{T_S}{R_J}$	$\frac{T_S}{R_J}$
$JN7$, 2 branches:	Seller i sold a good in the last round?	Buyer i bought a good in the last round?
$JN8$, 2 branches:	Seller i submitted oa in the last time step?	Buyer i submitted ob in the last time step?
$JN9$, 3 branches:	oa^t is closer to P^B, or closer to P^T but still lager than P^T, or smaller than P^T	ob_t is closer to P^B, or closer to P^T but still lower than P^T, or larger than P^T.
$JN10$, 4 branches:	ob^t is smaller than c_{in}, or lager than c_{in} but closer to c_{in}, or lager than c_{in} but closer to P^T, or lager than P^T	oa^t is higher than v_{in}, or lower than v_{in} but closer to v_{in}, or lower than v_{in} but closer to P^T, or lower than P^T.
$JN11$, 2 branches:	ob_t is smaller or larger than the last P^F	oa_t is smaller or larger than the last P^F
$JN12$, 2 branches:	last P^F is smaller or larger than the average of all the P^F in the history	
$JN13$, 2 branches:	$\sum_{r-j_r+1}^{r} \left\| P^F - \frac{1}{j_r} \times \sum_{r-j_r+1}^{r} P^F \right\|$ is smaller or larger than $\sum_{r-2*j_r+1}^{r-j_r} \left\| P^F - \frac{1}{j_r} \times \sum_{r-2*j_r+1}^{r-j_r} P^F \right\|$	

where, a_i, b_i and c_i are the parameters used to divide the situations into domains, P_{MAX} is the maximum valid price for the auction, P_{MIN} is the minimum valid price for the auction, the number of the time steps for seller i submitted oa in the past L^{ts} time steps is denoted by N^{oa}_{Si}, the number of the time steps for buyer i submitted ob in the past L^{ts} time steps is denoted by N^{ob}_{Bi}, oa of the time step t is denoted by oa^t and ob of the time step t is denoted by ob^t, and the number of the agent's successful transactions in the past R_J rounds is denoted by T_S, j_r in $JN13$ is a suitable number helping to judge.

Table 6.3: 7 kinds of processing nodes for GNP-AP

	GNP-AP seller	GNP-AP buyer
$PN1$:	ask the current *ob*	bid the current *oa*
$PN2$:	ask the current $ob + \gamma_s$	bid the current oa - γ_b
$PN3$:	ask the last $P^F + \gamma_s$	bid the last P^F - γ_b
$PN4$:	ask the current oa - γ_s	bid the current $ob + \gamma_b$
$PN5$:	stay. make no new *ask*.	stay. make no new *bid*.
$PN6$:	ask P^T.	bid P^T.
$PN7$:	ask $P^T + \gamma_s$	bid P^T - γ_b

Here, γ_s and γ_b are small prices from the set of $\{V_{step}, 2 \times V_{step}, 3 \times V_{step}\}$.

- When r=1, the bidder makes P^T close to P^P, but still has a small difference from P^P.

- When $r > 1$, if a seller does trade frequently considering its trading times, then it will submit a new *ask* a little bit higher than the previous P^F in order to gain more profits by selling at a higher price. Similarly, if a buyer does trade frequently, then it will submit a new *bid* a little bit lower than the previous P^F in order to gain more profits by buying at a lower price.

- When $r > 1$, if a seller does not trade so frequently, then it is willing to submit a lower *ask* for the good, which might be equal to its P^P to sell the good rather than no trade at all. Similarly, if a buyer does not trade so frequently, then it is willing to submit a higher *bid* for the good, which might be equal to its P^P to buy the good rather than no trade at all.

Secondly, the judgment functions are developed based on the following information of large-scale CDAs:

- the agent's own P^Ps of the goods

- the total number of goods the agent wants to trade (NUM) and the total number of goods the agent already traded (num)

- the current round number (r)

- the time step of the current auction (t)
- the current oa, ob, the previous oas and obs in the past L_{ts} time steps, where L_{ts} is the number of time steps stored in the current round history
- the P^Fs of the past L_r successful transactions

6.2.3 Kinds of nodes

Suppose that CDA is in the r th round and the current time step is t. GNP-based seller i, who wants to sell NUM goods and has already sold num goods, is willing to sell the n th good ($n = num + 1$), and P^P of this good for seller i is c_{in}. GNP-based buyer i, who wants to buy NUM goods and has already bought num goods, is willing to buy the n th good, and P^P of this good for buyer i is v_{in}.

13 different kinds of judgment functions of JNs for GNP-AP seller and GNP-AP buyer are proposed, respectively as shown in Table. 6.2. The items include: oa, ob, P^B, P^T, c_{in}, v_{in} and $P^F s$, the current time step t, the relation among the above prices and the relation between num and NUM. Especially, JN11, JN12 and JN13 are newly proposed for GNP-AP, which judges new kinds of information that are not included in GNP-RN.

7 different kinds of bidding actions of PNs for GNP-AP seller and GNP-AP buyer are shown in Table.6.3, respectively, which are the same as GNP-RN. GNP-AP agent i submits the ask or bid according to 7 main potential bidding actions in Table. 6.3 at each time step, respectively.

7 different kinds of RNs are assigned with 7 different δ values, that is, $-3 \times V_{step}$, $-2 \times V_{step}$, $-1 \times V_{step}$, 0, V_{step}, $2 \times V_{step}$ and $3 \times V_{step}$, respectively.

6.2.4 Fitness function for agents

The same fitness functions are used as section 5.2.4.

For seller individual i, the profit in the CDA process is calculated by

$$\sum_{g \in G_i^s} (P_g^{F_g} - c_{ig}),$$

where, G_i^s is the set of suffixes of goods seller individual i sold and $P_g^{F_g}$ is the final price of good g;

for buyer individual i, the profit in the CDA process is calculated by

$$\sum_{g \in G_i^b} (v_{ig} - P_g^{F_g}),$$

where, G_i^b is the set of suffixes of goods buyer individual i bought.

6.3 Simulations

We compared the proposed GNP-AP method with ZI-U, ZI-C, GD and CP [46] [47] [48] [82].

To evaluate the behaviors of each agent using GNP-AP, ZI-C, ZI-U, CP and GD, each trader intends to trade the same number of goods as the other agents on the seller side or buyer side to make fair comparison.

3 simulations are studied:

Simulation 1: Referring to the simulation settings in the previous work [82], the following 2 cases are studied in order to observe the basic ability of each strategy:

For seller side: One of the 5 strategies is used for one of the 5 seller agents. At the same time, the buyers are all ZI-C agents in order to make fair comparison for 5 sellers. Each seller is assumed to have 60-140 units of goods to sell, while each buyer is assumed to want 100 units of goods to buy. So, the supply of CDAs is 300-700, and the demand is 500. The profits obtained by each seller are compared.

For buyer side: Similarly, one of the 5 strategies is used for one of the 5 buyer agents. The sellers are all ZI-C agents in order to make fair comparison for 5 buyers. Each seller is assumed to have 100 units of goods to sell, while each buyer is assumed to want 60-140 units of goods to buy. So, the supply of CDAs is 500, and the demand is 300-700. The profits obtained by each buyer are compared.

Moreover, in order to fairly compare the GNP-AP and GNP-RN strategy, GNP-RN agent also performed the above experiments with ZI-C, ZI-C, CP and GD agents under the same situations as the above, and the profit gained by GNP-RN is compared with GNP-AP.

Simulation 2: One of the 5 strategies is used for one of the 5 seller agents and also

Table 6.4: Parameters setting

Items	Value
N of Goods	300 to 700 (5×60 to 5×140)
N of Agents	5 sellers, 5 buyers
N of Goods that Agents Wants to Trade (NUM)	from 60 to 140
N of Training Environments	30
N of Testing Environments	10
N of Time Steps in One Round	100
α_1, α_2, α_3, α_4	1.5, 0.5, 0.05, 0.05
$\beta_1, \beta_2, \beta_3, \beta_4$	0.1, 0.3, 0.1, 0.3
L_{ts}, R_s, R_J, j_r	5, $\frac{1}{6}NUM$, $\frac{1}{10}NUM$, $\frac{1}{10}NUM$
Total Generation	600
Generation for Structure	300
Generation for Parameters	300
Population Size	300
—Elite	60
—Crossover	120
—Mutation	120
Selection	upper 30%
Crossover Rate for Structure	0.1
Mutation Rate for Structure	0.3-
Mutation Rate for Parameters	$0.3 \times (1 - \frac{Current_{Generation}}{2MAX_{Generation}})$
Node	
—Judgment Node	65
—Processing Node	15
—Rectify Node	15
—Start Node	1

used for one of the 5 buyer agents. The seller and buyer adopting the same strategy are treated as a pair, so there are 5 pairs of agents, and the total profit gained by each pair is compared. The following 2 cases are studied:

When demand is fixed: Each seller is assumed to have 60-140 units of goods to sell, while each buyer is assumed to want 100 units of goods to buy.

When supply is fixed: Similarly, each seller is assumed to have 100 units of goods to sell, while each buyer is assumed to want 60-140 units of goods to buy.

Similarly, in order to fairly compare the GNP-AP and GNP-RN strategy, GNP-RN agent also performed the above experiments in stead of GNP-AP under the same situations as the above, and the profit gained by GNP-RN agent pair is compared with GNP-AP agent pair.

Simulation 3: All of the sellers and buyers use the same kind of strategy. This simulation is done for the 5 kinds of strategies, respectively. The efficiency of each kind of strategy is compared. The following 3 cases are studied:

$demand > supply$: Each seller is assumed to have 60 units of goods to sell, while each buyer is assumed to want 100 units of goods to buy.

$demand = supply$: Each seller is assumed to have 100 units of goods to sell, while each buyer is assumed to want 100 units of goods to buy.

$demand < supply$: Each seller is assumed to have 100 units of goods to sell, while each buyer is assumed to want 60 units of goods to buy.

P^P of each good for each seller is from the special normal distribution of $N(1.75, 1.00)$ with the data more than 2.5 and less than 1.0 being omitted. P^P of each good for each buyer is also from the special normal distribution $N(2.75, 1.00)$ with the data more than 3.5 and less than 2.0 being omitted. The smallest bidding step is 0.01. δ is from the set of {-0.03, -0.02, -0.01, 0.00, 0.01, 0.02, 0.03}, which is contained in RNs. The values of α_1, α_2, α_3 and α_4 are set based on the common believes. The values of β_1, β_2, β_3 and β_4 are set according to the value range of P^Ps of the goods for each trader. There are 30 environments for the training and 10 for testing in order to avoid the loss of generality. In each environment, P^P of each good for each seller and P^P of each good for each buyer are different. All the testing results are the total profits of the 10 environments. Each simulation runs for 30 times, and the result is the average result over 30 runs. The more specific parameters used in the simulations are shown in Table. 6.4.

Table 6.5: Comparison between GNP-AP and GNP-RN in simulation 1 (profit gained by the agent on one side)

Seller Case, Demand=500									
	60	70	80	90	100	110	120	130	140
GNP-AP	670.73	702.19	790.24	874.85	992.30	1035.23	1048.73	1013.14	1001.19
GNP-RN	648.32	686.28	770.46	848.14	973.02	1010.45	1006.99	985.87	970.98
Improvment	3.46%	2.32%	2.57%	3.15%	1.98%	2.45%	4.41%	2.28%	3.11%
Buyer Case, Supply=500									
	60	70	80	90	100	110	120	130	140
GNP-AP	653.06	782.29	891.19	892.58	1037.42	1055.32	1035.67	1054.10	1021.27
GNP-RN	635.33	757.37	843.04	952.96	1011.12	1024.98	1010.37	1029.33	990.46
Improvment	2.79 %	3.29%	5.71%	3.11%	2.60%	2.95%	2.51%	2.41%	3.10%

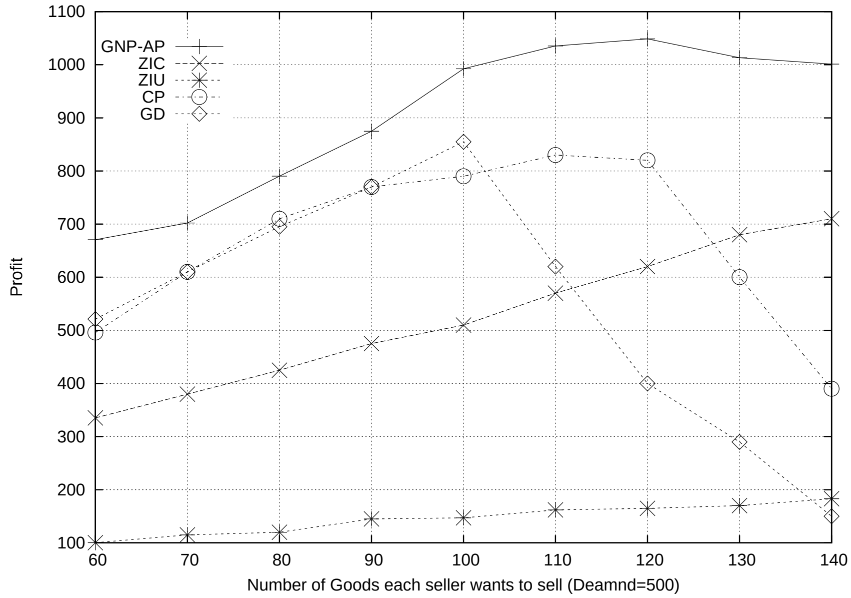
6.3.1 Simulation 1

Simulation group 1 is conducted to evaluate the performance of each kind of agents when all the agents on the other side use ZI-C strategy. From Fig. 6.2, it can be found that GNP-AP agents can get the highest profits under all the conditions, which is the same results as we obtained in [82]. Table. 6.5 shows the comparison between the profits obtained by GNP-AP agent and GNP-RN agents under the same condition. The results show there is an improvement when GNP-AP is adopted by the agent in large-scale CDAs.

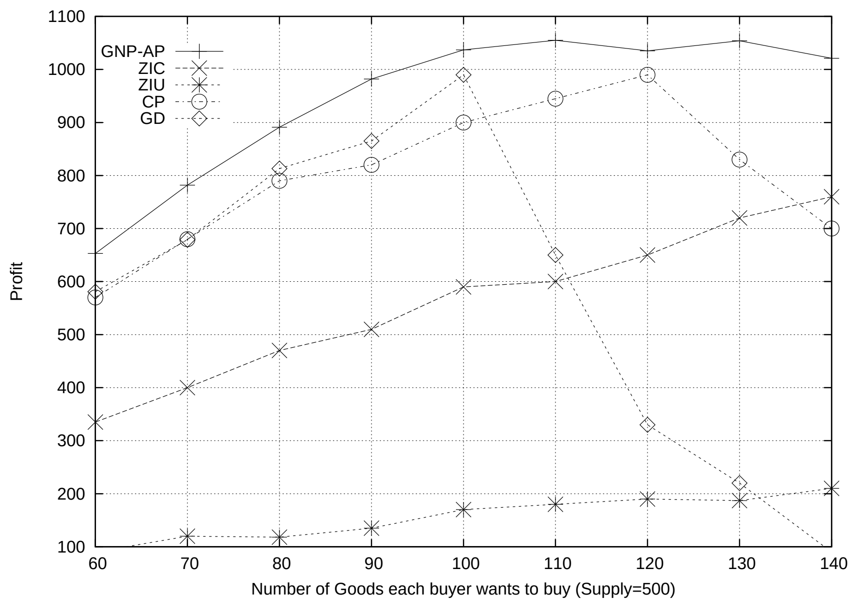
6.3.2 Simulation 2

Based on Simulation 1, Simulation 2 is conducted to evaluate the performance of each kind of strategy when it is used on both seller side and buyer side at the same time. In simulation 2, the total profit obtained by the seller and buyer using the same kind of strategy is evaluated instead of evaluating the profit obtained by the agent only on one side in simulation 1. Because each strategy is used by both the seller and buyer, the simulations do not focus on the seller side or buyer side, instead they consider both sides in the following two cases, i.e., (1). deamnd=500, and supply changes from 300 to 700; (2). supply=500, and demand changes from 300 to 700.

From Fig. 6.3, it can be found that GNP-AP agent pair can also get the highest profits among all the agent pairs under all the conditions. Furthermore, separating the

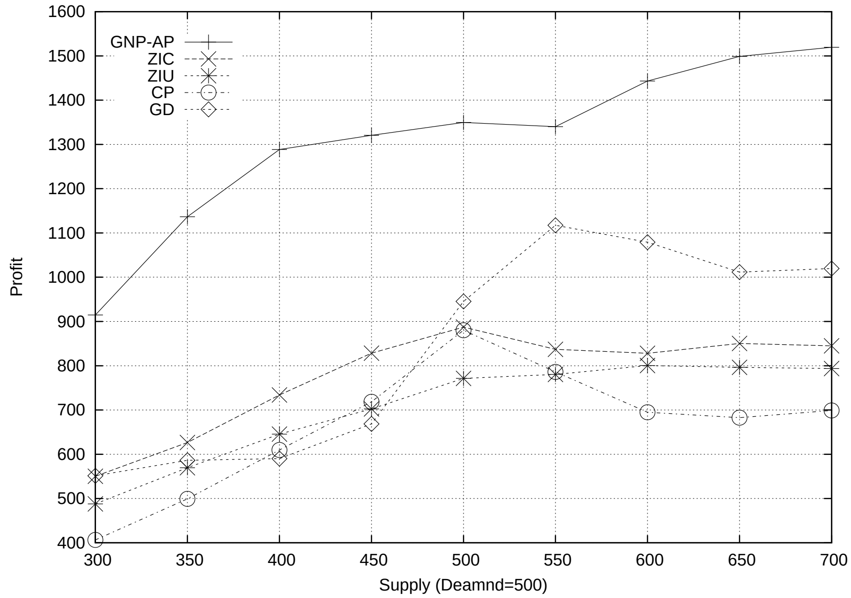


(a)Seller Case

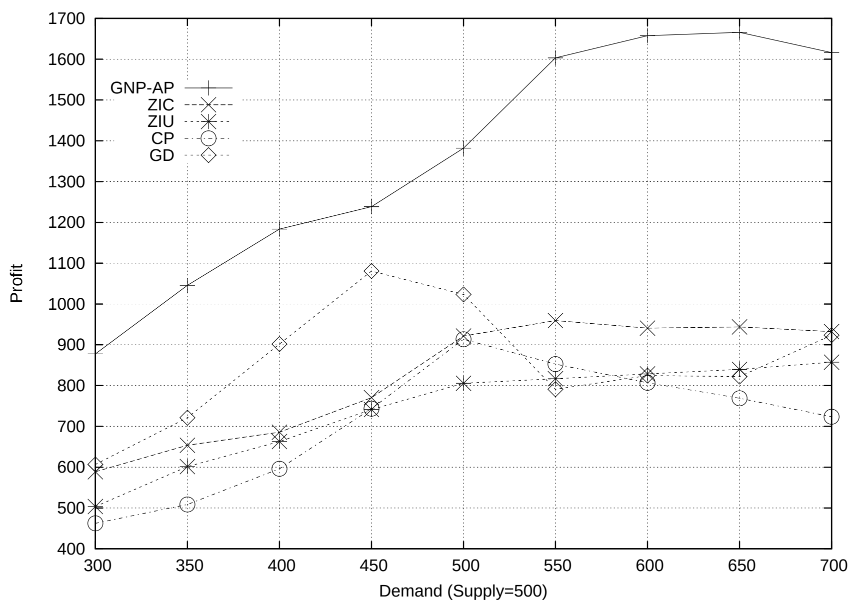


(b)Buyer Case

Figure 6.2: Comparison of the strategies when target strategies are only used by one side of auction agents



(a)Demand==500



(b)Supply=500

Figure 6.3: Comparison of the strategies when target strategies are used by both sides of auction agents

Table 6.6: Comparison between GNP-AP and GNP-RN in simulation 2 (profit gained by the agent pair on both sides)

	Demand=500								
	300	350	400	450	500	550	600	650	700
GNP-AP	877.58	1045.73	1183.72	1238.70	1382.08	1603.18	1657.77	1665.94	1616.30
GNP-RN	852.06	995.74	1151.19	1164.63	1321.8	1557.27	1611.37	1617.70	1552.58
Improvment	2.99 %	5.02%	2.82%	6.36%	4.56%	2.95%	2.88%	2.98%	4.10%
	Supply=500								
	300	350	400	450	500	550	600	650	700
GNP-AP	914.59	1136.67	1288.63	1320.66	1349.58	1340.02	1443.23	1498.87	1519.54
GNP-RN	888.55	1093.06	1238.95	1276.37	1307.98	1274.51	1401.61	1427.36	1455.08
Improvment	2.93%	3.99%	4.01%	3.47%	3.18%	5.14%	2.97%	5.01%	4.43%

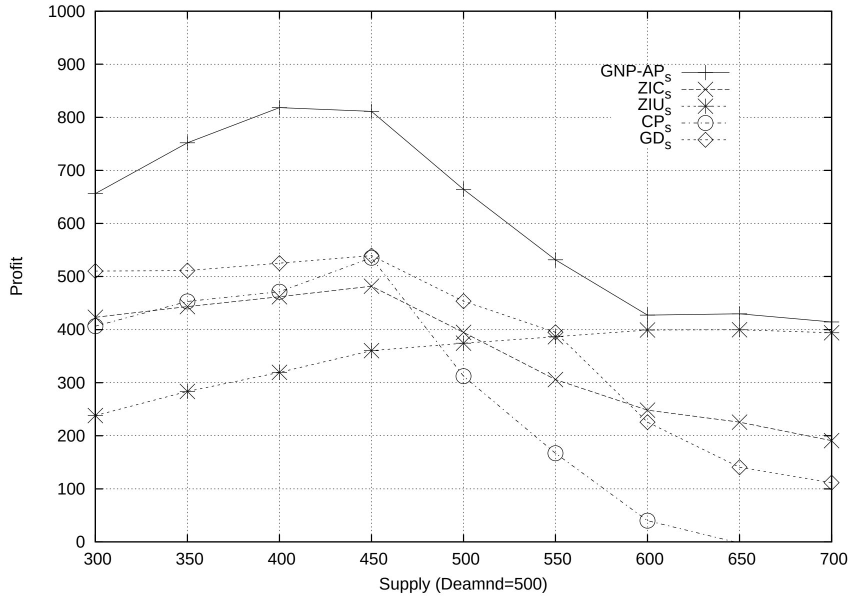
total profits gained by each kind of agent pair into seller side and buyer side, Fig. 6.4 and Fig. 6.5 give the profits gained by each seller and each buyer when demand==500 and supply==500, respectively.

Table. 6.6 shows the comparison between the profits obtained by GNP-AP agent pair and GNP-RN agent pair under the same condition. The results show there is also an improvement when GNP-AP is adopted. Besides the above observation, Fig. 6.4 and Fig. 6.5 show that not only GNP-AP agent pair perform best, but also GNP-AP agent can always get the highest profits no matter which single side is considered.

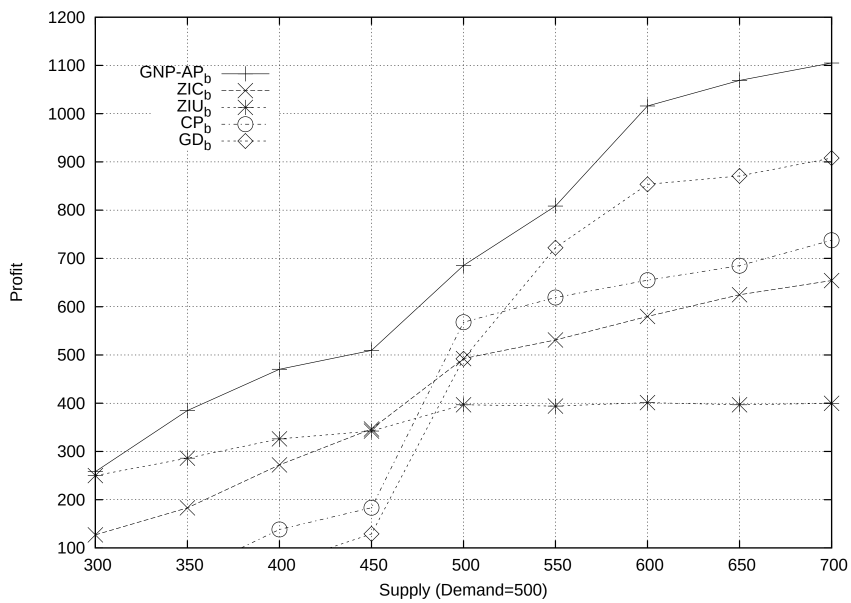
Moreover, Table. 6.7 gives the success rate of each strategy under different situations. The success rate means the percentage of the number of goods the agent actually trades to the number of goods the agent wants to trade. Table. 6.8 gives the average trading price of each kind of strategy. For the buyer side, the lower the trading price is, the better the strategy is, because it means the agent can buy the good at a profitable price. For the seller side, the higher the trading price is, the better the strategy is. Table. 6.8 does not show all the data of all the situations, but gives the results in the representative situations when the demand equals to supply, demand is smaller than supply and demand is larger than supply. From these results, it can be observed that GNP-AP agents can trade goods at very profitable prices, while ZI-U agent trades at very competitive prices, but has little profit due to its random bidding strategy. That explains why even if ZI-U agents have the success rate of 100%, it obtains very low profits.

Table 6.7: Trading success rate in simulation 2

	Demand=500								
	300	350	400	450	500	550	600	650	700
$GNP-AP_s$	99.98%	99.99%	99.03%	99.30%	98.51%	90.62%	90.01%	95.92%	92.59%
$GNP-RN_s$	99.99%	99.99%	99.01%	99.11%	98.37%	90.61%	87.13%	92.25%	92.62%
$ZI-C_s$	100.00%	100.00%	100.00%	100.00%	100.00%	63.93%	41.40%	34.64%	25.28%
$ZI-U_s$	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
CP_s	100.00%	100.00%	100.00%	100.00%	100.00%	99.99%	98.33%	88.57%	67.58%
GD_s	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	86.01%	64.45%	71.70%
$GNP-AP_b$	81.86%	96.64%	97.52%	100.00%	100.00%	100.00%	98.91%	98.64%	100.00%
$GNP-RN_b$	80.38%	94.59%	98.13%	99.59%	99.99%	99.89%	97.60%	98.92%	100.00%
$ZI-C_b$	24.93%	39.33%	60.89%	80.25%	100.00%	100.00%	100.00%	100.00%	100.00%
$ZI-U_b$	100%	100.00%	100%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
CP_b	85.70%	91.12%	99.43%	99.99%	100.00%	100.00%	100.00%	100.00%	100.00%
GD_b	7.50%	22.89%	41.38%	69.13%	98.51%	100.00%	100.00%	100.00%	100.00%
	Supply=500								
	300	350	400	450	500	550	600	650	700
$GNP-AP_s$	86.38%	96.52%	97.49%	90.12%	100.00%	100.00%	99.98%	99.87%	99.52%
$GNP-RN_s$	83.17%	96.14%	92.00%	92.59%	99.99%	99.70%	99.70%	98.87%	99.99%
$ZI-C_s$	23.57%	26.94%	35.93%	59.45%	100.00%	100.00%	100.00%	100.00%	100.00%
$ZI-U_s$	100.00%	100.00%	100%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
CP_s	54.64%	67.84%	91.93%	99.88%	100.00%	100.00%	100.00%	100.00%	100.00%
GD_s	35.06%	56.48%	74.08%	100.00%	99.33%	98.75%	100.00%	100.00%	100.00%
$GNP-AP_b$	99.42%	99.83%	99.05%	99.89%	100.00%	99.95%	100.00%	99.91%	98.63%
$GNP-RN_b$	99.31%	99.89%	97.93%	99.28%	99.71%	97.03%	94.77%	92.25%	94.10%
$ZI-C_b$	100.00%	100.00%	100.00%	100.00%	100.00%	79.03%	61.00%	50.61%	37.38%
$ZI-U_b$	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
CP_b	100.00%	100.00%	100.00%	100.00%	100.00%	99.66%	98.67%	97.10%	94.99%
GD_b	100.00%	100.00%	100.00%	99.49%	99.33%	74.77%	56.98%	36.89%	25.80%

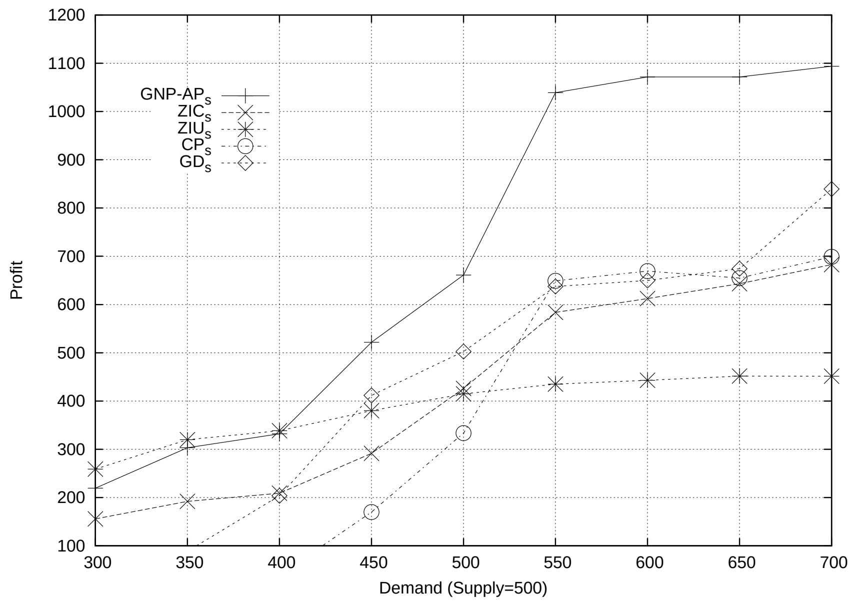


(a)Demand==500, Seller side

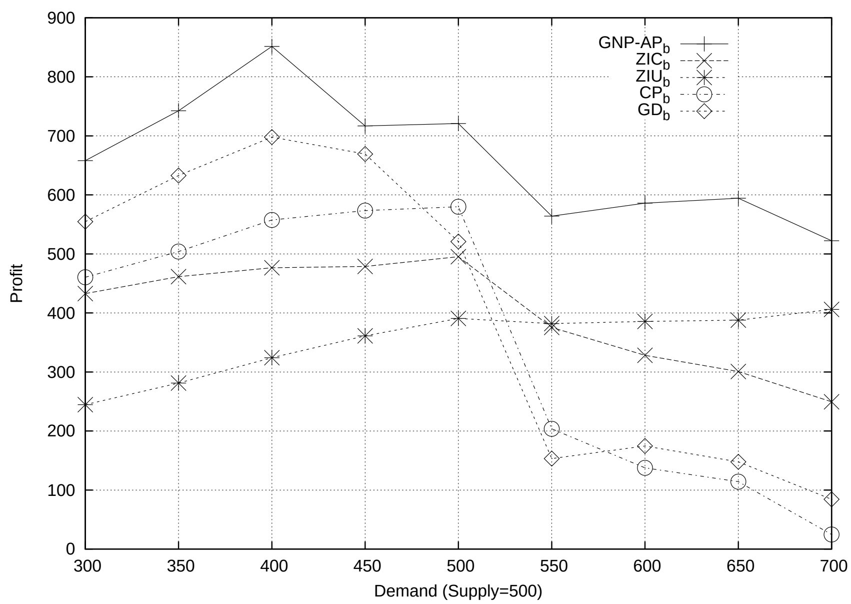


(b)Demand==500, Buyer side

Figure 6.4: Comparison of sellers and buyers when demand==500



(a)Supply==500, Seller side



(b)Supply==500, Buyer side

Figure 6.5: Comparison of sellers and buyers when supply==500

Table 6.8: Average trading price of each kind of agent in simulation 2

	Demand=500 Supply=300	Demand=Supply=500	Demand=700 Supply=500
$GNP-AP_s$	2.79	2.38	2.80
$ZI-C_s$	2.50	2.15	2.42
$ZI-U_s$	2.17	2.13	2.18
CP_s	2.42	2.06	2.45
GD_s	2.66	2.21	2.59
	Demand=300 Supply=500	Demand=Supply=500	Demand=500 Supply=700
$GNP-AP_b$	1.68	2.05	1.67
$ZI-C_b$	2.01	2.24	2.05
$ZI-U_b$	2.33	2.34	2.33
CP_b	1.92	2.13	1.98
GD_b	1.80	2.20	1.80

Table 6.9: Efficiency of each kind of agent in simulation 3

	Supply=500 Demand=300			Supply=100 Demand=100			Supply=300 Demand=500		
	buyer	seller	all	buyer	seller	all	buyer	seller	all
GNP-AP	1.037	0.9522	0.9946	0.9983	0.9988	0.9986	0.9663	1.0269	0.9966
ZI-C	0.9881	0.9815	0.9848	0.9747	0.9832	0.9790	0.9873	0.9768	0.9821
ZI-U	1.0111	0.9403	0.9757	0.9671	0.9783	0.9727	0.9780	0.9643	0.9711
CP	0.9921	0.9828	0.9875	0.9712	1.0213	0.9963	0.9887	0.9951	0.9919
GD	0.9354	1.0331	0.9843	0.9602	1.0230	0.9916	0.9535	1.0023	0.9779

6.3.3 Simulation 3

Extended from the above two simulations, simulation 3 studies the situation, where all of the agents use only one kind of strategy when the supply is larger than demand, supply equals to demand, supply is smaller than demand, respectively. Table.6.9 gives the market efficiencies of each strategy when it is used by all the agents under different conditions. The efficiency of the market means the percentage of the profit the agent actually gained to the profit could be gained if the agent trades the goods at the equilibrium price, at which the demand meets supply in the market. Here, we use $p^* = \frac{1}{H} \times \sum_{r=R-H}^{R} P_r^F$ to represent the equilibrium price of the market, where R is the latest round when a trade occurs and P_r^F is the final price of round r. Because the demand and supply are from 300-700 in this chapter, H is decided from the equation that $H = \frac{1}{50} \times min\{demand, supply\}$. It can be observed that when GNP-AP is used, the highest market efficiency can be obtained.

Particularly, when the demand is smaller than supply, the efficiency of the GNP-AP buyer is higher than seller, which means buyers have more profitable transactions, and are more successful in driving the market price. The opposite observation is obtained when the demand is larger than supply.

In summary, GNP-AP is observed to be the most efficient strategy and has the ability to gain most profits with very high success rates under various situations.

6.4 Conclusions

GNP-AP bidding strategy for CDA agents has been proposed in this chapter to obtain a good guidance for the intelligent auction systems, especially large-scale CDAs. The GNP based agent can find the general optimal strategy which suits for many environments and can be very efficient for the market.

It is found from the simulations comparing with the conventional auction agents that the use of GNP-AP to choose the suitable prices for the bidding is more flexible for the various situations of auctions due to its evolutionary features and well organized structures.

Chapter 7

Conclusions

In this research, some studies on developing bidding strategy using GNP for online auctions were done in order to guide auction agents bidding actions. The bidding strategies for MREA and CDA are developed based on GNP because GNP has been proved to be efficient and effective in complex and dynamic situations. The aim of the study is to better automate the online auction, and to facilitate the agent to be more efficient and competitive for making bidding decisions in order to maximize its owner's profit.

In the proposed method, the GNP population represents the group of potential bidding strategies. Each individual uses the if/then decision-making functions to judge the auction information and to guide the agent to take the suitable actions under different situations. Thus, the proposed method is flexible and adaptive to various auction situations. During the evolution, the GNP structure is systematically organized, and finally, the individual which can obtain the highest profit is selected as the optimal bidding strategy. The contents of judgment nodes and processing nodes are designed properly according to the characteristic of MREA and CDA.

In chapter 3, a bidding strategy using Genetic Network Programming for MREA has been proposed. The proposed method can help the agent to make bidding decisions efficiently at every time step. Its effectiveness has been studied in the general case and poorest case of NTLM MREA and TLM MREA. The results showed that the agents based on the proposed bidding strategy can become smarter during the evolution. The proposed method is able to observe and judge the auction information very well, and make the suitable bidding decision at each time step. In all the different situations studied, GNP agent can obtain considerably good performance. In this stage, for better studying and analyzing GNP's characteristic, all the agents in the simulations use GNP strategy.

After the ability of GNP for guiding bidding actions are proved, the study in chapter 4 is devoted to improve the performance of GNP strategy. The judgment nodes in GNP are improved to have more outgoing branches, which permits to judge more auction situations at a time. The number of the kinds of processing nodes is increased to offer more bidding options for the agent. Moreover, the GNP agents are able to consider the owner's attitude towards to each good, which make the strategy more personalized and better satisfy the owner's objective. The enhanced GNP strategy is studied and compared to the previous GNP strategy and the non-GNP strategies under different situations. The simulation results show the superior performance of the enhanced GNP bidding strategy.

In chapter 5, GNP-RN bidding strategy have been developed to guide the agent's buying and selling behavior in CDA. According to its own structure, GNP-RN uses heuristic rules to decide what bids or asks. RN is proposed for bringing more flexible and various options for bidding action choices. We benchmarked the performance of GNP-RN strategy against other 4 prominent alternatives available in the literature under several situations. All the results reveal the effectiveness of GNP-RN.

Further improvement of GNP-RN is done in chapter 6. Aiming to enhance the sensitivity of the bidding strategy for large-scale CDAs, GNP with adjusting parameters (GNP-AP) is proposed. The parameters used by GNP-AP decision-making functions are adjusted during the evolution instead of being fixed in GNP-RN, which improves the ability of the bidding strategy for judging auction information. From the study and analysis, GNP-AP is confirmed that it can give a good guidance for the agents in large-scale CDAs and could be very efficient for the markets.

Appendix

Genetic Network Programming(GNP)

GNP is an extended method of GA[49] and GP[50; 85]. GA evolves strings and it is mainly applied to optimization problems. GA can find suboptimal solutions of the problems quickly, so it has been widely studied and applied to many real problems. GP was devised later in order to expand the expression ability of GA by using tree structures. This structural change of solutions brought the progress on the evolutionary computation and made GP applicable to more complex problems. But, it is generally said that GP is sometimes difficult to search for the optimum solution because the searching space of solutions becomes enormous due to its bloat, that is, the searching efficiency of GP is not so high in some cases.

Genetic Network Programming (GNP), whose genome structure is a directed graph, is proposed by K. Hirasawa in 2000 to overcome the problems of GP[51; 52; 54]. It is an extension of GA and GP, and unlike the expression of string information of Genetic Algorithm and the tree structure of Genetic Programming, GNP expresses itself in a directed graph network consist of nodes. The original idea is based on the more general representation ability of graphs than that of trees. The aim of developing GNP is to deal with dynamic environments efficiently by using the higher expression ability of graph structures than that of trees, and the inherently equipped functions in it. Each node of GNP is to be a minimum unit which executes the judgment or processing for the agents, and the transition rule of GNP is totally different from GP system.

On the other hand, GNP boots from the start node and never returns to it during the execution, then a series of node transitions generate the solutions of GNP. Therefore, these node transitions act like an implicit memory function in GNP. It is also possible to apply GNP to Partially Observable Markov Decision Processes using only the specific functional nodes needed for the current state of the problem. Thus, the aim of GNP is to construct an efficient graph based programming having implicit memory functions and applicable to even the Partially Observable Markov Decision Process environments. In other words, GNP is a new evolutionary method to construct generalized discrete event systems by combining program modules. GNP aims to be more applicable to many problems by separating the judgment nodes and processing nodes structurally so that the network can be easily evolved.

<Basic Concept of GNP>

Fig.1 shows the basic structure of GNP. The directed graph structure is used to represent individuals. GNP program is composed of one start node and plural judgment nodes and processing nodes. Start node has no function and no conditional branch. When GNP begins to boot, this node is executed at first. Judgment nodes have various decision functions dealing with the specific inputs from the environments such as sensor information and measured data. It returns a judgment result and determines the next node to be executed. Processing nodes work as action functions. After the start node, the current node is transferred according to the node connections and judgment results. In processing nodes, actions are conducted to environments. All movements of the agents are decided by the function of judgment nodes and processing nodes of GNP. The labels of all kinds of judgment and processing functions(Judgment node: 1, 2, ..., J, Processing node: 1, 2, ..., P) are set up in the libraries, which are prepared by the designers. The node transition begins from a start node, and there is no terminal node.

The connection is branched off by the judgement results, which are predefined by judgement functions in judgement nodes. Accordingly, if there are a lot of judgement results, then the number of branches increase and the network structure become complicated. And processing nodes have just one branch in order to carry out the next judgement. Actually, GNP can use the fixed number of nodes, in other words, GNP can adopt evolving the genotypes with variable number of nodes, but in this thesis paper, GNP evolves only the networks with the predefined number of nodes. It would be better to say that GNP here evolves the genotypes with fixed number of nodes. We set the number of each node in GNP, e.g., $J_1 \times 3, J_2 \times 3, \cdots, P_1 \times 3, P_2 \times 3$, and so on.

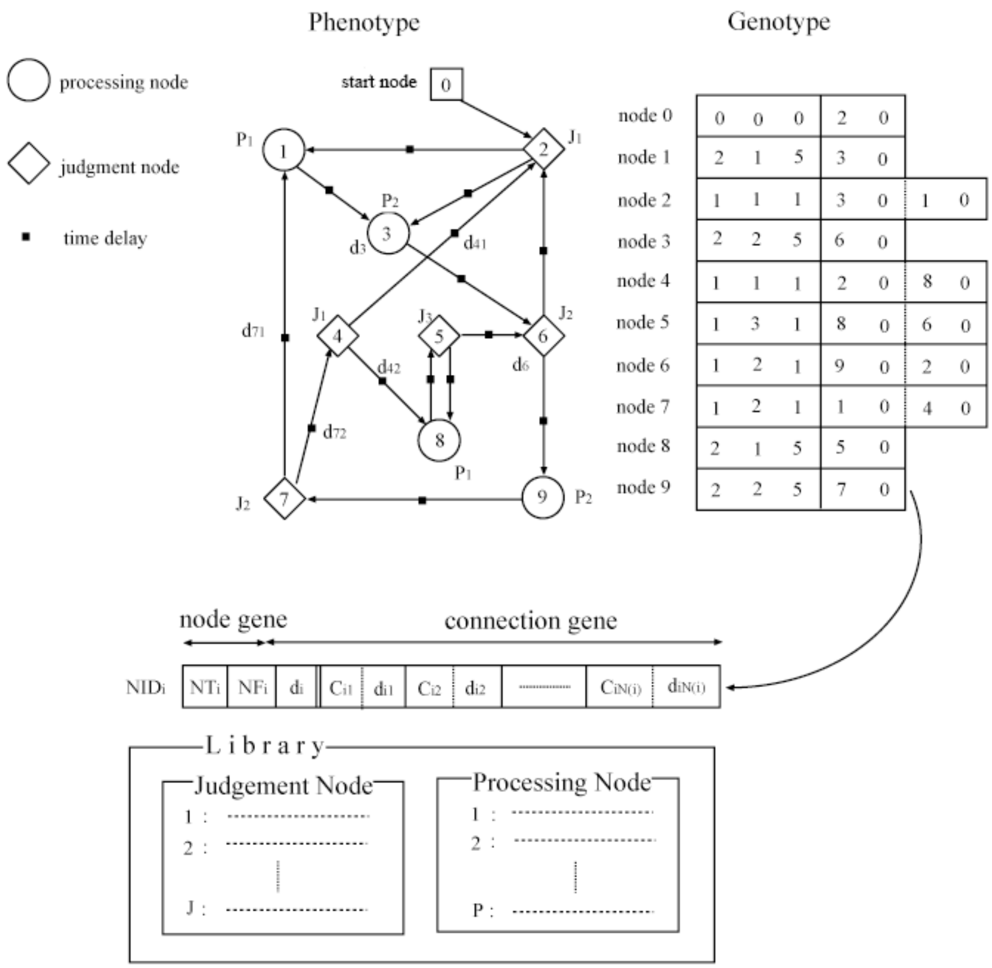
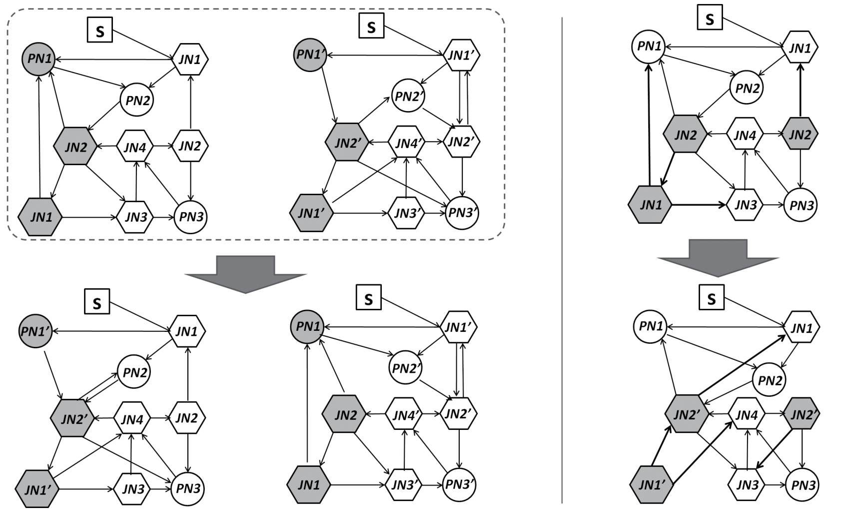


Figure 1: Basic structure of GNP



Crossover for GNP structure

Mutation for GNP structure

Figure 2: Genetic Operators

Once GNP is booted up, the execution starts from the start node, then the next node to be executed is determined according to the connection from the current activated node. If the activated node is judgement node, the next node is determined by the judgement results. When processing node is executed, the next node is uniquely determined by the single connection from processing node.

<Genetic Operators>

In general, the exploitation of the accumulated information resulting from the evolutionary search is done by the selection mechanism, while the exploration of new regions in the search space is accounted by genetic operators, to balance the exploration and exploitation.

The genetic operators mimic the process of heredity of genes to create new offspring in each generation. The operators are used to alter the genetic composition of individuals during evolution. In essence, the operators perform a random search, and cannot guarantee an improved offspring. There are three common genetic operators: selection, crossover and mutation. Fig. 2 shows a simple example.

<Summary of GNP>

Using the nodes transitions as solutions of the problem, GNP help the individual understand the conditions of the problem very well. Also, due to the various judgment nodes, it is also possible for GNP to respond to the changes of the environments quickly. which makes GNP as an effective method mainly for dynamic problems. Due to its gene structure, it has the implicit memory function and reuses the nodes, which leads to the compact structure of GNP.

With the evolutionary ability and network programming structure, the optimal solutions of many problems can be easily obtained by using GNP.

References

[1] R. Murch and T. Johnson, “Intelligent Software Agents”, Upper Saddle River, N.J.: Prentice Hall PTR, chapter 14, pp. 131-137, 1999

[2] “E-Commerce: A Global Outlook”, Global Industry Analysts, Inc., 2012.

[3] “Report of Chinese E-Commerce Market of The First Half of 2011”, E-Commerce Research Center of China, 2011.

[4] M. Subramani and E. Walden, “Economic Returns to Firms from Business-to-Business Electronic Commerce Initiatives: An Empirical Examination”, In Proc. of the 21st Intl Conf. Information Systems, pp. 229-241, 2000.

[5] R. P. McAfee and J. McMillan, “Auctions and bidding”, Economic Perspectives, vol. 2, pp. 699–738, 1987.

[6] P. R. Wurman, M. Wellman and W. Walsh, “The Michigan Internet Auctionbot: A Configurable Auction Server for Human and Software Agents”, In Proc. Second Int’l Conf. Autonomous Agents, pp. 301-308, 1998.

[7] M. He, N. R. Jennings and H. Leung, “On agent-mediated electronic commerce”, IEEE Trans on Knowledge and Data Engineering, vol. 15, no. 4, pp. 985–1003, 2003.

[8] C. Beam and A. Segev “Automated Negotiation: A Survey of the State of the Art”, CITM Working Paper 96-WP-1022, 1996.

[9] D. C. Parkes, L. H. Ungar and D. P. Foster, “Accounting for Cognitive Costs in On-Line Auction Design”, Agent Mediated Electronic Commerce, P. Noriega and C. Sierra, eds., pp. 25-40, 1998.

[10] Y. Fujishima, K. Leyton-Brown and Y. Shoham, “Taming the Computational Complexity of Combinatorial Auctions: Optimal and Approximate Approaches”, In Proc. 16th Int’l Joint Conf. Artificial Intelligence, pp. 548-553, 1999.

[11] R. Karp, “Reducibility among Combinatorial Problems”, Complexity of Computer Computations, R. Miller and J. Thatcher, eds., Cambridge, Mass.: MIT Press, pp. 85-103, 1972.

[12] S. Rassenti, V. Smith and R. Bulfin, “A Combinatorial Auction Mechanism for Airport Time Slot Allocation”, Bell J. Economics, vol. 13, no. 2, pp. 402-417, 1982.

[13] Y.-K. Che, “Design Competition through Multidimensional Auctions”, RAND J. Economics, vol. 24, no. 4, pp. 668-680, 1993.
[14] M. Bichler, M. Kaukal and A. Segev, “Multi-Attribute Auctions for Electronic Procurement”, In Proc. First IBM IAC Workshop Internet Based Negotiation Technologies, 1999.
[15] J. Teich, H. Wallenius and J. Wallenius, “Multiple-Issue Auction and Market Algorithms for the World Wide Web”, Decision Support Systems, vol. 26, pp. 49-66, 1999.
[16] M. Wooldridge and N. R. Jennings, “Intelligent Agents: Theory and Practice”, Knowledge Eng. Rev., vol. 10, no. 2, pp. 115-152, 1995.
[17] R. Das, J. Hanson, J. Kephart and G. Tesauro, “Agent-Human Interactions in the Continuous Double Auction”, In Proc. 17th Int’l Joint Conf. Artificial Intelligence, B. Nebel, ed., vol. 2, pp. 1169-1176, 2001.
[18] N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, C. Sierra and M. Wooldridge, “Automated Negotiation: Prospects, Methods and Challenges”, Int. J. Group Decision and Negotiation, vol. 10, no. 2, pp. 199-215, 2001
[19] N. R. Jennings, “An agent-based approach for building complex software systems”, Comm. ACM, vol. 44, no. 4, pp. 35–41, 2001.
[20] J. K. Goeree, C. R. Plott and J. Wooders, “Bidders’ choice auctions: Raising revenues through the right to choose”, Advanced Computational Intelligence and Intelligent Informatics, vol. 2, pp. 504–515, 2004.
[21] J. S. Rosenschein and G. Zlotkin, “Rules of Encounter”, MIT Press, 1994.
[22] V. Krishna, “Auction Theory”, Academic Press, 2002
[23] W. Vickrey, “Counter speculation, Auctions and Competitive Sealed Tenders”, J. Finance, vol. 16, pp. 8-37, 1961.
[24] T. Sandholm, “Distributed Rational Decision Making”, Multiagent Systems, G. Weiss, ed., pp. 201-258, MIT Press, 1999.
[25] P. Milgrom, “Auctions and Bidding: A Primer”, J. Economic Perspectives, pp. 3-22, 1989.
[26] P. Anthony and N. R. Jennings, “Developing a bidding agent for multiple heterogeneous auctions”, ACM Transaction on Internet Technology, vol. 3, pp. 185–217, 2003.
[27] C. Preist, C. Bartolini and I. Phillips, “Algorithm Design for AgentsWhich Participate in Multiple Simultaneous Auctions”, Agent Mediated Electronic Commerce III, F. Dignum and U. Cortes, eds., Springer-Verlag, pp. 139–154, 2001.
[28] C. Preist, A. Byde and C. Bartolini, “Economic Dynamics of Agents in Multiple Auctions”, In Proc. of the Fifth Intl Conf. Autonomous Agents, pp. 545–551, 2001.
[29] P. Faratin, C. Sierra and N. R. Jennings, “Negotiation decision functions for autonomous agents”, Robot. Autonom. Syst., vol. 24, pp. 159–182, 1998.

[30] T. Ito, N. Fukuta, T. Shintani and K. Sycara, "Biddingbot: A multiagent support system for cooperative bidding in multiple auctions", In Proc. of the 4th International Conference on Multi Agent Systems, pp. 399–400, 2000.

[31] D. Friedman and J. Rust, "The Double Auction Market: Institutions, Theories and Evidence", Addison-Wesley, New York, NY, USA, 1992.

[32] D. D. Davis and C. A. Holt, "Experimental Economics", Princeton University Press, Princeton, NJ, 1993.

[33] S. H. Clearwater (Ed.), "Market-Based Control—A Paradigm for Distributed Resource Allocation", World Scientific, 1996.

[34] D. B. Marron and C. W. Bartels, "The use of computer-assisted auctions for allocating tradeable pollution permits", in: S.H. Clearwater (Ed.), Market-Based Control: A Paradigm for Distributed Resource Allocation, pp. 274–299, 1996.

[35] J. C. Fuller, K. P. Schneider and D. Chassin, "Analysis of Residential Demand Response and double-auction markets", In Proc. IEEE Power and Energy Society General Meeting, 2011.

[36] P. Vytelingum, S. D. Ramchurn, T. D. Voice, A. Rogers and N. R. Jennings, "Trading agents for the smart electricity grid", In Proc. Ninth International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2010), pp. 897-904, Toronto, Canada, 2010.

[37] R. K. Dash, P. Vytelingum, A. Rogers, E. David and N. R. Jennings, "A market-based task allocation mechanism for limited capacity suppliers", IEEE Trans. on System, Man and Cybernetics (Part A), vol. 37, no. 3, pp. 391–405, 2007.

[38] D. Easley and J. Ledyard, "Theories of Price Formation and Exchange in Double Oral Auctions", The Double Auction Market: Institutions, Theories, and Evidence, D. Friedman and J. Rust, eds., pp. 63-97, 1993.

[39] D. Friedman, "The Double Auction Market Institution: A Survey", The Double Auction Market: Institutions, Theories, and Evidence, D. Friedman and J. Rust, eds., pp. 3-25, Reading, Mass: Addison-Wesley, 1993.

[40] A. Sadrieh, "The Alternating Double Auction Market: A Game Theoretic and Experimental Investigation", Springer, 1998.

[41] M. Yokoo, E. H. Durfee, T. Ishida and Kuwabara, "Distributed constraint satisfaction problems: Formalization and algorithms", IEEE Trans. Knowl. Data Eng., vol. 10, no. 5, pp. 673–685, 1998.

[42] C. Boutilier, "Multiagent systems: Challenges and opportunities for decision theoretic planning", AI Mag., vol. 20, no. 4, pp. 35–43, 1999.

[43] A. Greenwald and J. Boyan, "Autonomous agents for participating in multiple online auctions", Autonomous Agents and Multiagent Systems, vol. 10, pp. 67– 89, 2005.

[44] P. Anthony, W. Hall, V. D. Dang and N. R. Jennings, "Bidding algorithms for simultaneous auctions: A case study", IJCAI Workshop on E-Business and Intelligent Web, pp. 54–64, 2001.

[45] P. Anthony and N. R. Jennings, “Evolving bidding strategies for multiple auctions”, In Proc. 15th European Conference on Artificial Intelligence, pp. 178–182, 2002

[46] S. Gjerstad and J. Dickhaut, “Price formation in double auction”, Games and Economic Behavior, vol. 22, pp. 1–29, 1998.

[47] D. K. Gode and S. Sunder, “Allocative efficiency of markets with zero-intelligence traders: Market as a partial substitute for individual rationality”, J. Polit. Econ., vol. 101, pp. 119–137, 1993.

[48] C. Preist, “Commodity trading using an agent-based iterated double auction”, In Proc. of the Third Int’l Conf. Autonomous Agents, pp. 131–138, 1999.

[49] J. H. Holland, “Adaptation in natural and artificial systems”, University of Michigan Press, 1975.

[50] J. R. Koza, “Genetic programming ii, automatic discovery of reusable programs”, MIT Press, 1994.

[51] S. Mabu, K. Hirasawa and J. Hu, “A graph-based evolutionary algorithm: Genetic network programming (gnp) and its extension using reinforcement learning”, Evolutionary Computation, MIT Press, vol. 15, no. 3, pp.369-398, 2007

[52] K. Hirasawa, M. Okubo, H. Katagiri, J. Hu and J. Murata, “Comparison between genetic network programming (GNP) and genetic programming (GP)”, Evolutionary Computation 2001, vol. 2, pp. 1276 – 1282, Seoul , South Korea, 2001.

[53] Y. Chen, S. Mabu, K. Shimada and K. Hirasawa, “Real time updating genetic network programming for adapting to the change of stock prices”, IEEJ Trans. EIS, vol. 192, no. 2. pp. 344-354, 2009.

[54] K. Hirasawa, T. Eguchi, J. Zhou, L. Yu and S. Markon, “A double-deck elevator group supervisory control system using genetic network programming”, IEEE Trans. on Systems, Man and Cybernetics, Part C, vol. 38, no. 4. pp.535-550, 2008.

[55] S. Mabu, C. Chen, N. N. Lu, K. Shimada and K. Hirawasa, “An intrusion-detection model based on fuzzy class-association-rule mining using genetic network programming”, IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, vol. 41, no. 1, pp.130-139, 2011.

[56] H. Zhou, S. Mabu, W. Wei, K. Shimada and K. Hirasawa, “Time related class association rule mining and its application to traffic prediction”, IEEJ Transactions on Electronics, Information and Systems, vol. 130, no. 2. pp. 289-301, 2010.

[57] P. R.Wurman, “Dynamic pricing in the virtual marketplace”, IEEE Internet Computing, vol. 5, pp. 36–42, 2001.

[58] P. Vytelingum, D. Cliff and N. R. Jennings, “Strategic bidding in continuous double auctions”, Artificial Intelligence Journal, vol. 172, no. 14, pp. 1700–1729, 2008.

[59] N. Matos, C. Sierra and N. R. Jennings, “Successful negotiation strategies: An evolutionary approach”, In Proc. of the 3rd International Conference on Multi Agent Systems, pp. 182–189, Paris, France, 2001.
[60] A. Byde, “Programming model for algorithm design in simultaneous auctions”, In Proc. of the 2nd International Workshop on Electronic Commerce, pp. 152–163, New York, 2001.
[61] M. Goyal, J. Lu and G. Zhang, “A Novel Fuzzy Attitude Based Bidding Strategy for Multi-attribute Auction”, In. Proc of the 2006 IEEE/WIC/ACM international conference on Web Intelligence and Intelligent Agent Technology, pp. 535–539, 2006.
[62] X. Sui and H. F. Leung, “An Adaptive Bidding Strategy in Multi-Round Combinatorial Auctions For Resource Allocation”, In. Proc of the 20th IEEE International Conference on Tools with Artificial Intelligence, pp. 423–430, 2008.
[63] R. K. Dash, P. Vytelingum, A. Rogers, E. David and N.R. Jennings, “A market-based task allocation mechanism for limited capacity suppliers”, IEEE Trans. on System, Man and Cybernetics (Part A), vol. 37, no. 3, pp. 391–405, 2007.
[64] V. L. Smith, “An experimental study of competitive market behaviour”, Journal of Political Economy, vol. 70, pp. 111–137, 1962.
[65] D. Cliff and J. Bruten, Hewlett-Packard Labs., Bristol, UK, HP Labs. Tech. “Minimal intelligence agents for bargaining behaviors in market-based environments”, Rep. HPL 97-91, pp. 91–97, 1997.
[66] G. Tesauro and R. Das, “High-performance bidding agents for the continuous double auction”, EC01: In Proc. of the Third ACM Conference on Electronic Commerce, ACM Press, pp. 206–209, New York, NY, USA, 2001.
[67] G. Tesauro and J. L. Bredin, “Strategic sequential bidding in auctions using dynamic programming”, AAMAS02: In Proc. of the First International Joint Conference on Autonomous Agents and Multiagent Systems, ACM Press, pp. 591–598, New York, NY, USA, 2002.
[68] M. Bader-El-Den and S. Fatima, “Genetic Programming for Auction Based Scheduling”, In Proc. 13th European conference on Genetic Programming, Springer-Verlag Berlin, pp. 256-267, Heidelberg, Germany, 2010.
[69] S. Phelps, S. Parsons, E. Sklar and P. McBurney, “Using genetic programming to optimise pricing rules for a double-auction market”, In Workshop on Agents for Electronic Commerce, 2003.
[70] C.W. Richter,. Jr., G. B. Sheble, D. Ashlock, “Comprehensive bidding strategies with genetic programming/finite state automata”, IEEE Transactions on Power Systems, vol. 14, no. 4, pp. 1207- 1212, 1999.

[71] M. He and N. R. Jennings, “A fuzzy-logic based bidding strategy for autonomous agents in continuous double auctions”, IEEE Transactions on Knowledge and Data Engineering, vol. 15, no. 6, pp. 1345–1363, 2003.

[72] R. Kowalczyk, “On Negotiation as a Distributed Fuzzy Constraint Satisfaction Problem”, In Proc. DEXA E-Negotiation Workshop, pp. 631-637, 2000.

[73] R. Kowalczyk and V. Bui, “On Fuzzy E-Negotiation Agents: Autonomous Negotiation with Incomplete and Imprecise Information”, In Proc. DEXA e-Negotiation Workshop, 2000.

[74] X. Luo, N. R. Jennings, N. Shadbolt, H. F. Leung and J. H. M. Lee, “A Fuzzy Constraint Based Model for Bilateral, Multi-Issue Negotiation in Semi-Competitive Environments”, Artificial Intelligence, vol. 148, no. 1-2, 2003.

[75] N. R. Jennings, T. J. Norman, P. Faratin, P. O’Brien and B. Odgers, “Autonomous Agents for Business Process Management”, Applied Artificial Intelligence, vol. 14, no. 2, pp. 145-189, 2000.

[76] J. Nicolaisen, V. Petrov and L. Tesfatsion, “Market power and efficiency in a computational electricity market with discriminatory double-auction pricing”, IEEE Transactions on Evolutionary Computation, vol. 5, no. 5, pp. 504–523, 2001.

[77] A. E. Roth, “The economist as engineer: game theory, experimentation, and computation as tools for design economics”, Econometrica, vol. 70, no. 4, pp. 1341–1378, 2002.

[78] C. Yue, S. Mabu, Y. Wang and K. Hirasawa, “Multiple Round English Auction agent based on Genetic Network Programming”, IEEJ Transactions on Electrical and Electronic Engineering, vol. 5, no. 3, pp.450-458, 2010.

[79] C. Yue, S. Mabu, D. Yu, Y. Wang and K. Hirasawa, “A Bidding Strategy of Multiple Round Auctions based on Genetic Network Programming”, In Proc. of the IEEE World Congress on Computational Intelligence 2010, pp. 33-40, Barcelona, Spain, 2010.

[80] C. Yue, S. Mabu and K. Hirasawa, “Enhancing Bidding Strategy Using Genetic Network Programming in Agent-Based Multiple Round English Auction”, Journal of Advanced Computational Intelligence and Intelligent Informatics, vol. 15, no. 7, pp.747-758, 2011.

[81] C. Yue, S. Mabu, Y. Chen, Y. Wang and K. Hirasawa, “Agent bidding strategy of multiple round English Auction based on genetic network programming”, In Proc. of the SICE International Annual Conference (SICE2009), pp.3857-3862, Fukuoka, Japan, 2009.

[82] C. Yue, S. Mabu and K. Hirasawa, “A Bidding Strategy with Heuristic Rules for Continuous Double Auctions based on Genetic Network Programming”, Transaction of the Japanese Society for Evolutionary Computation, vol. 2, no. 2, pp.53-67, 2011.

[83] C. Yue, S. Mabu, D. Yu, Y. Wang and K. Hirasawa, “Bidding Strategy based on Genetic Network Programming in Continuous Double Auctions”, In Proc. of the SICE Annual

International Conference (SICE2010), pp. 2620-2625, Taibei, Taiwan, 2010.

[84] S. Mabu, D. Yu, C. Yue and K. Hirasawa, “No Time Limit and Time Limit Model of Multiple Round Dutch Auction based on Genetic Network Programming”, Journal of Advanced Computational Intelligence and Intelligent Informatics, vol. 15, no. 1, pp.3-12, 2011.

[85] J. R. Koza, “Adaptation in natural and artificial systems,” MIT Press, 1992.

Research Achievements

Journal Paper

(1) C. Yue, S. Mabu and K. Hirasawa, "A Bidding Strategy with Heuristic Rules for Continuous Double Auctions based on Genetic Network Programming", Transaction of the Japanese Society for Evolutionary Computation, Vol. 2, No. 2, pp.53-67, 2011/11.

(2) C. Yue, S. Mabu and K. Hirasawa, "Enhancing Bidding Strategy Using Genetic Network Programming in Agent-Based Multiple Round English Auction", Journal of Advanced Computational Intelligence and Intelligent Informatics, Vol. 15, No. 7, pp.747-758, 2011/09.

(3) S. Mabu, D. Yu, C. Yue and K. Hirasawa, "No Time Limit and Time Limit Model of Multiple Round Dutch Auction based on Genetic Network Programming", Journal of Advanced Computational Intelligence and Intelligent Informatics, Vol. 15, No. 1, pp.3-12, 2011/01.

(4) C. Yue, S. Mabu, Y. Wang and K. Hirasawa, "Multiple Round English Auction Agent based on Genetic Network Programming", IEEJ Transactions on Electrical and Electronic Engineering, Vol. 5, No. 3, pp.450-458, 2010/06.

International Conference Paper

(1) C. Yue, S. Mabu and K. Hirasawa "A bidding strategy for Continuous Double Auctions based on Genetic Network Programming with generalized judgments", In Proc. of the 2011 IEEE International Conference on Systems, Man, and Cybernetics (SMC) , pp. 144-151, Anchorage, AK, USA, 2011/10.

(2) C. Yue, S. Mabu, D. Yu, Y. Wang and K. Hirasawa “Bidding Strategy based on Genetic Network Programming in Continuous Double Auctions”, In Proc. of the SICE International Annual Conference (SICE2010), pp. 2620-2625, Taibei, Taiwan, 2010/08.

(3) C. Yue, S. Mabu, D. Yu, Y. Wang and K. Hirasawa “A Bidding Strategy of Multiple Round Auctions based on Genetic Network Programming”, In Proc. of the IEEE World Congress on Computational Intelligence 2010 (CEC2010), pp. 33-40, Barcelona, Spain, 2010/07.

(4) C. Yue, S. Mabu, Y. Chen, Y. Wang and K. Hirasawa, “Agent Bidding Strategy of Multiple Round English Auction based on Genetic Network Programming”, In Proc. of the ICROS-SICE International Conference, pp. 3857-3862, Fukuoka, Japan, 2009/08.

(5) Y. Chen, C. Yue, S. Mabu and K. Hirasawa, “A Genetic Relation Algorithm with Guided Mutation for the Large-Scale Portfolio Optimization”, In Proc. of the ICROS-SICE International Conference, pp. 2579-2584, Fukuoka, Japan, 2009/08.

(6) C. Chen, S. Mabu, C. Yue, K. Shimada and K. Hirasawa, “Analysis of Fuzzy Class Association Rule Mining based on Genetic Network Programming”, In Proc. of the ICROS-SICE International Conference, pp. 3480-3484, Fukuoka, Japan, 2009/08.

(7) C. Chen, S. Mabu, C. Yue, K. Shimada and K. Hirasawa, “Network Intrusion Detection using Fuzzy Class Association Rule Mining Based on Genetic Network Programming”, In Proc. of the IEEE International Conference on Systems, Man and Cybernetics, pp. 60-67, Texas, USA, 2009/10.

Japanese Domestic Conference Paper

(1) C. Yue, Y. Wang, S. Mabu and K. Hirasawa, “Multiple Round English Auction based on Genetic Network Programming”, 第 18 回インテリジェント・システム・シンポジウム, pp.229-234, 広島, 日本, 2008/10.