Exploring Market Behaviors with Evolutionary Mixed-Games Learning Model

Yu Du[1*], Yingsai Dong[1], Zengchang Qin[1,2†], and Tao Wan[2]

[1]Intelligent Computing and Machine Learning Lab
School of Automation Science and Electrical Engineering
Beihang University, Beijing, China
[2]Robotics Institute, Carnegie Mellon University, USA
[*]ydu1989@gmail.com, [†]zcqin@andrew.cmu.edu

Abstract. The minority game (MG) is a simple model for understanding collective behavior of agents competing for a limited resource. Ma *et. al* [7] assumed that collective data can be generated from combination of behaviors of variant groups of agents and proposed the minority game data mining (MGDM) model. In this paper, to further explore collective behaviors, we propose a new behavior learning model based on evolutionary optimization of mixed-games, that assumes there are variant groups of agents playing majority games [3, 4] as well as the minority games. Genetic algorithms then are used to optimize group parameters to approximate the decomposition of the original system and use them to predict the outcomes of the next round. In experimental studies, we apply the EMGL model to real-world time-series data analysis by testing on a few stocks from Chinese stock market and the USD-RMB exchange rate. The results suggest that the EMGL model can predict statistically better than the MGDM model for most of the cases and both models perform significantly better than a random guess.

1 Introduction

Agent-based experimental games have attracted much attention of scientists from different research areas to explore complex systems such as financial markets [8]. New research themes such as experimental economics [2], financial market modeling [5] and market mechanism designs [9] have been flourished in recent years. In financial market modeling, an economic market is regarded as a complex adaptive system (CAS), and people try to analyze the real market system of which agents with similar capability compete for limited resources. Every agent knows the history data of the market and decides how to trade based on global information. The minority game (MG) has been widely used to model the interactions among agents as a simplified version of a financial market [1].

In previous work [6], we assumed the existence of one "intelligent agent" who can take advantages of the game by learning from all other agents' behaviors in minority games. In reality, it is always infeasible to obtain all records of agents' choices in each round of the game. If we assume that the collective data are generated from the combination of variant groups of agents' behaviors - which is intuitively true, how can we decompose the collective data into the combinations of micro-level data is what hope to explore in the Minority Game Data Mining (MGDM) [7]. Genetic algorithms (GA) are used to optimize the agent group parameters to yeild the best approximation of the original dynamic system. The MGDM model was applied to the real-world time-series data analysis by testing on its effectiveness in stock market predictions [7].

However, there are some weaknesses of using the MG model in real-world market data analysis. First, since all agents have the same memory length, the diversity of agents is limited. Second, in the real-world markets, some agents play the minority game, which are referred to as "foundation traders" who hope to maximize their profits; while others are just "trend chasers" who choose what the majority do (i.e. majority game). In order to establish an agent-based model which more closely approximate the real market, Gou [3, 4] modifies the MG model by dividing agents into two groups: one group play the minority game and the other group play the majority game, thus this system is referred to as a 'mixed-game' model. Inspired by the 'mixed-game' model, we propose the Evolutionary Mixed-game Learning (EMGL) model. We divide the agents in the game into three diverse groups: (1) the agents who make random decisions; (2) agents who play minority games and (3) agents who play majority games. By applying genetic algorithms, we can model the behaviors of above three types of agents, thus analyze and estimate the resource-constrained environment parameters to maximize the approximation of the system outputs to the real-world test data. That is a new way to understand the relationship between micro-behaviors and macro-behaviors in complex dynamic systems.

This paper is structured as follows: Section 2 introduces the mixed-game model. In section 3, we propose the EMGL model that uses genetic algorithms to optimize the mixed-game model to discover the composition of agents and predict the macro-behaviors in the resource-constrained environment. In section 4, we apply the EMGL model to predict financial time-series in the stock market. We also compare the results of the EMGL model with the previous MGDM model and verify the effectiveness of this learning mechanism. Conclusions and discussions are given in the end.

2 Mixed-Games Model

The Minority Game (MG)[1] was originated from the El Farol Bar problem and formulated to analyze decision-making. In the MG, there are an odd number of players and each must choose one of two choices independently at each round of the game, winners are those on the minority side at last. There is no prior communication among players; the only information available is numbers of players corresponding to two choices of the last round. In this section, we will set a resource-constrained environment populated by three diverse types of agents: agents who make random decisions (random traders), agents who play minority game, and agents who play majority game, representing so-called "trend chasers". This variety of agents is for simulating a more realistic real market [3].

2.1 Strategies of Agents

Suppose an odd number of N agents decide between two possible options, say to attend Room A or B at each round of the game. Formally, at round $t(t = 1, 2, \ldots, T)$: each agent takes an action $a_i(t)$ for $i = 1, 2, \ldots, N$ to choose, i.e :

$$a_i(t) = \begin{cases} A & \text{Agent } i \text{ choose room } A \\ B & \text{Agent } i \text{ choose room } B \end{cases} \tag{1}$$

At each round t, agents belonging to the minority group win. The winning outcome can be represented by binary code function $w(t)$. If A is the minority side, i.e. the number of agents choosing Room A is no greater than $(N-1)/2$, we define the winning outcome $w(t) = 0$; otherwise, $w(t) = 1$. In this paper, the winning outcomes are known to public, formally represented by:

$$w(t) = \begin{cases} 0 & \text{if: } \sum_{i=1}^{N} \Delta(a_i(t) = A) \leq (N-1)/2 \\ 1 & \text{otherwise} \end{cases} \tag{2}$$

where $\Delta(\alpha)$ is the truth function: if α is true, then $\Delta(\alpha)$ is 1; otherwise, $\Delta(\alpha)$ is 0. We assume that agents make choices based on the most recent m winning outcomes $h(t)$, which is called history memory and m is the memory length, formally: $h(t) = [w(t-m), \ldots, w(t-2), w(t-1)]$.

In the MG, we usually assume that each agent's reaction towards the previous data is governed by a "strategy" [1]. Each strategy is based on the past m-bit memory, described as a binary sequence, then there are 2^{2^m} possible strategies in the strategy space. Each agent looks into the most recent history for the same pattern of m bit string and predicts the outcome. Given history memory $h(t)$, we denoted Agent i's choice guided by strategy S as $S(h(t))$. Table 1 shows one possible strategy S with $m = 4$. For example, $h(t) = [0000]$ represents that if the winning outcomes of the latest 4 rounds are all 0, the next round (at round t) choice for this agent will be $S([0000]) = A$. Thus a strategy can be regarded as a particular set of decisions on the permutations of previous winning outcomes.

Table 1. One possible strategy S with the memory length $m = 4$.

$h(t)$	0000	0001	0010	0011	0100	0101	0110	0111
$S(h(t))$	A	A	A	B	B	A	A	A
$h(t)$	1000	1001	1010	1011	1100	1101	1110	1111
$S(h(t))$	A	B	B	B	B	A	A	B

2.2 Mixed-Games with Heterogeneous Agents

In order to obtain a better approximation of the collective behaviors in the real-world market, Gou [3, 4] modifies the MG model and proposes the 'mixed-game model', in which agents are divided into two groups: each group has different memory length, Group G_N plays minority game with the same strategy, while Group G_J plays majority game with the same strategy. Comparing to the MG, the most significant part of mixed-game is that it has an additional group of "trend chasers", therefore be more realistic to simulate a real-world market.

Given a training time range, all agents in G_N choose the best strategy with which they can predict the minority side most correctly, while all agents in G_J choose the best strategy with which they can predict the majority side most correctly. N_1 represents the number of agents in G_N and N_2 represents the number of agents in G_J. We use m_1 and m_2, respectively, to describe the memory length of these two groups of agents. As each agent's reaction is based on a strategy corresponding a response to past memories, there are $2^{2^{(m_1)}}$ and $2^{2^{(m_2)}}$ possible strategies for G_N or G_J, respectively.

However, the mixed-game model is still not a reasonable prediction tool in real-world market with the following reasons: in real-life scenarios, it is unrealistic to assume all agents playing minority game or majority game hold the same strategy and follow the same rule: some agents make random decisions and different subgroups hold different strategies; if all agents act in the same way, they will all lose. We assume the completeness of marketing world is embodied in existence of variant groups of agents using their own strategies. Therefore, in this paper, we improve the mixed-game by dividing the agents into three diverse types of agents: agents who make random decisions (denoted by G_R), agents of Group G_N (playing the minority game) with different strategies, agents of Group G_J (playing the majority game) with different strategies.

3 Evolutionary Mixed-Games Learning Model

As we mentioned above, we propose a framework based on the assumption that the macro-behavior of the market is an aggregation of three groups of agents:

- Group G_N: Agents who play minority game.
- Group G_J: Agents who play majority game.
- Group G_R: Agents who make random decisions.

For G_N and G_J we assume that the overall effect can be decomposed into several small subgroups, while each subgroup of agents use a certain strategy. The decomposition of the collective behaviors involves a big set of parameters including the number of agents in each subgroup and the strategies they employ. We aim to use genetic algorithms to tune these parameters for these subgroups of agents to yield the collective behavior has the best approximation of the history data.

3.1 Chromosome Encoding

In our model, we use a parameter vector to represent the number of agents of each subgroup and the corresponding strategy they use, then we apply GA to explore the most likely combinations of subgroup behaviors that could generate the best approximated macro-level sequences. Given the history winning outcomes $w(t)$, the expected maximum number of subgroups using fixed strategies in G_N is K_N, and the expected maximum number of subgroups using fixed strategies in G_J is K_J. Thus agents of the whole system can be divided into $K_N + K_J + 1$ groups:

$$\{G_R, G(S_N^1), G(S_N^2), \ldots, G(S_N^{K_N}), G(S_J^1), G(S_J^2), \ldots, G(S_J^{K_J})\}$$

where G_R represents the group of random agents, $G(S_N^i)$ (for $i = 1, \ldots, K_N$) represents the subgroup agents holding strategy S_N^i in Group G_N (the group playing minority game). $G(S_J^k)$ (for $k = 1, \ldots, K_J$) represents the subgroup agents holding strategy S_J^k in Group G_J.

The chromosome for genetic algorithms $\mathbf{x}$ is encoded with the following parameters: $\mathbf{x} = \{P_R, P(S_N^1), S_N^1, \ldots, P(S_N^{K_N}), S_N^{K_N}, P(S_J^1), S_J^1, \ldots, P(S_J^{K_J}), S_J^{K_J}\}$

- P_R : the percentage of random agents among all agents (i.e. $P_R = \frac{G_R}{N}$)
- $P(S_N^i)$: the percentage of the number of agents in the minority game subgroup i ($i \in [1, 2, \ldots, K_N]$) with the fixed strategy S_N^i (i.e. $P(S_N^i) = \frac{|G(S_N^i)|}{N}$).
- S_N^i: Binary coding of the minority game strategy S_N^i.
- $P(S_J^k)$: the percentage of the number of agents in the majority game subgroup k ($k \in [1, 2, \ldots, K_J]$) with the fixed strategy S_J^k (i.e. $P(S_J^k) = \frac{|G(S_J^k)|}{N}$).
- S_J^k: Binary coding of the majority game strategy S_J^k.

Figure 1 illustrates that the collective behavior is a combination of choices from the above three types of agents. Given history sequence $h(t)$, the intelligent agent can use GA to explore all possible combinations of subgroups and compositions of the market, then use the information to make choice on the minority side.

3.2 Fitness Function

In the EMGL model, we aim to generate a system in which agents from both Group G_N and Group G_J can achieve their goals to the greatest extent, i.e., agents in Group G_N end up on the minority side while agents in Group G_J end up on the majority side. The final goal of EMGL model aims to obtain the best prediction of the market and make rational choice to maximum its profits. At round t, in order to evaluate the chromosome $\mathbf{x}_j (j = 1, 2, \ldots, J$ where J is the population size), we run the mixed-game with parameter setting decoded from $\mathbf{x}_j$ and get the prediction outcome. We choose the best chromosome by calculating the fitness function $f(\mathbf{x}_j)$ with the following three rules:

At round t, we consider collective data within the previous T steps: $(t - 1 - T, t - T, , \ldots, t - 2, t - 1)$

- *Rule 1*: For all agents in Group G_N, every time an agent predicts the correct outcome, i.e. chooses on the minority side, we add one point to $f(\mathbf{x}_j)$.

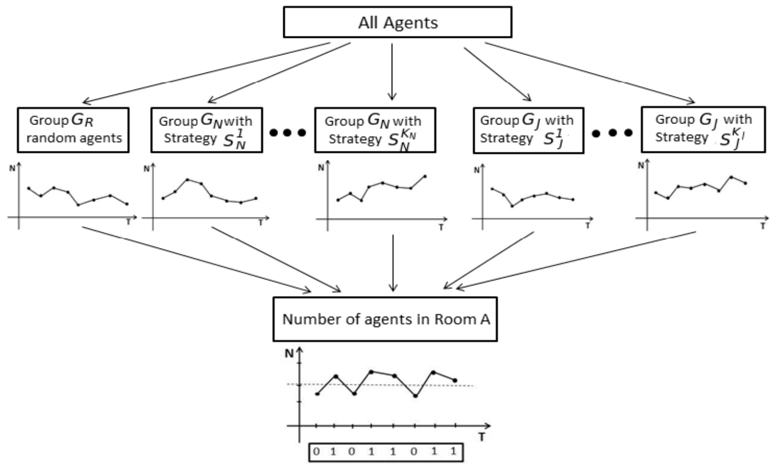

Fig. 1. The process of generating collective data. All agents can be divided into $K_N + K_J + 1$ groups where agents in the same subgroups act the same based on the strategy they follow. The collective data can be regarded as an aggregation of all agents' actions.

- *Rule 2*: For all agents in Group G_J, every time an agent predicts the correct outcome, i.e. chooses on the majority side, we add one point to $f(\mathbf{x}_j)$.
- *Rule 3*: If the prediction outcome $y_i(t)$ by the EMGL model is equal to the real-world macro outcome $w(t)$, we add a specific weight $W_{predict}$ to $f(\mathbf{x}_j)$.

Usually we set the weight value as a specific percentage of the total number of agents N: $W_{predict} = \beta N$ ($\beta \in [0, 1]$).

We calculate the fitness function $f(\mathbf{x}_j)$ for $t_0 = t-1-T, t-T, , \ldots, t-2, t-1$ and select the best chromosome $\mathbf{x}_j^*$ within the time range T.

$$\mathbf{x}^*(t) = \arg\max_j f(\mathbf{x}_j(t)) \quad for \quad j = 1, \ldots, J \tag{3}$$

Then we decode parameters from the best chromosome to obtain the best prediction of whole system and choose to be on the minority side.

4 Experiments on Real-World Markets

The EMGL model points a new way of using mixed-games model and evolutionary optimization in understanding the relationship between micro-data and macro-data. Given a sequence of history winning outcomes, we can use GA to explore the most likely combinations of single behaviors that could generate this sequence. Many real-world complex phenomena are caused by aggregations of agents' behaviors such as stock market and currency exchange rate, which are regarded as random and unpredictable in classical economics. In the following experiments, we apply the EMGL model to explore the compositions of the system using agents playing mixed-games. We can tune the parameters by training on the history data and use these estimated parameters to make future predictions.

Table 2. Comparisons of mean prediction accuracy of the EMGL and MGDM [7] models on 12 real-world financial time-series data including 11 stocks from Chinese market and the USD-RMB exchange rate.

# Data	Stock index	Start from (m/d/y)	MGDM in [7]	EMGL (4-3)	EMGL (5-3)	EMGL (6-3)
1. USD-RMB Exchange Rate	-	Jan 02 2001	58.59%	63.78%	64.13%	62.51%
2. SPD Bank Co.	600000	Jan 02 2001	55.99%	52.41%	50.17%	51.63%
3. Shandong Bohui Paper Co.	600966	Jun 08 2004	53.94%	54.45%	56.22%	54.54%
4. Shenergy Co.	600642	Jan 02 2001	51.78%	49.70%	49.61%	49.87%
5. China Minsheng Banking Co.	600016	Dec 19 2000	55.71%	52.63%	54.66%	52.44%
6. Qingdao Haier Co.	600690	Jan 02 2001	49.52%	54.01%	54.56%	54.02%
7. Huaneng Power Industrial Inc.	600011	Dec 06 2000	50.87%	51.23%	51.62%	51.21%
8. China United Network Comm.	600050	Oct 09 2002	51.34%	54.38%	53.83%	54.59%
9. CNTIC Trading Co.	600056	May 15 1997	52.99%	54.84%	55.09%	54.53%
10. Hisense Electric Co.	600060	Apr 22 1997	53.13%	56.93%	56.79%	57.68%
11. China Television Media Ltd	600088	Jun 16 1997	50.69%	52.11%	54.14%	53.29%
12. China Eastern Airlines Co.	600115	Nov 05 1997	55.62%	57.14%	56.69%	56.44%

4.1 Experiment Design

In the following experiments, We randomly select 11 stocks from the Chinese stock market through a downloadable software[1], and also the U.S.Dollar-RMB (Chinese Renminbi) exchange rate[2]. Compared with the validation method used in our previous work [7], we use a different validation benchmark in this paper: for each stock or currency exchange rate, we use the winning outcomes from $1-500$ trading days as training set to obtain relatively adaptable chromosomes, and then predict financial time-series of $501-800$ trading days. We compare the result of EMGL with MGDM to test the effectiveness of the new model.

Each round of the game represents one trading day. Given macro-level data $w(t)$, the best chromosome $\mathbf{x}^*(t)$ is selected and the parameter information in $\mathbf{x}^*$ is used for predicting the winning choice in the next round. Suppose the opening price is V_b and the closing price is V_f. For each trading day t, fluctuation of the stock price or exchange rate can be transferred to $w(t)$ as follows: if $V_b > V_f$, then $w(t) = 1$; otherwise, $w(t) = 0$. By correctly predicting $w(t)$ using the learning model, we can capture the ups and downs of the market prices.

[1] Website: http://big5.newone.com.cn/download/new_zszq.exe
[2] Data obtained from: http://bbs.jjxj.org/thread-69632-1-7.html

In the following experiments with the MGDM and EMGL models, we set $K_N = K_J = 20$. Since almost all agents play with history memories of 6 or less in a typical MG [10], and m_N is usually larger than m_J when using mixed-game model to simulate real market [3], we set $m_N = 4, 5, 6$ and $m_J = 3$ to establish three configuration of EMGL models. We set $K = 20$ and $m = 3$ in the MGDM model. As for the GA, we set population size $J = 50$, crossover rate $P_c = 0.8$, mutation rate $P_m = 0.05$, the specific weight $\beta = 0.5$. We run the whole experiments for 30 times to reduce the influences of randomness in GAs.

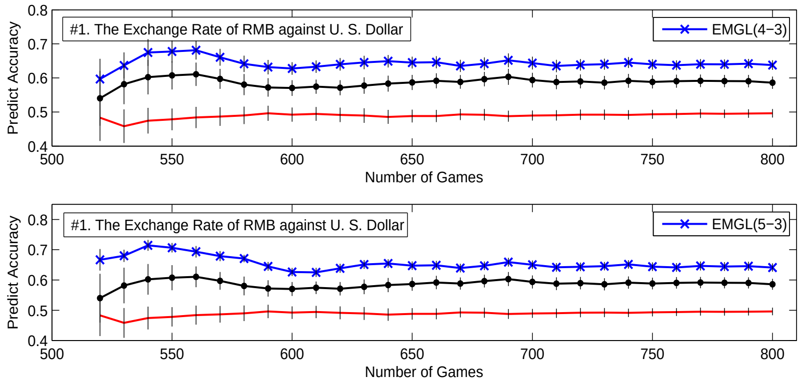

Fig. 2. Performance of the MGDM model and the EMGL model with different memory lengths on the USD-RMB exchange rate.

4.2 Data Analysis

Table 2 shows the prediction accuracy of 11 stocks and the US Dollar-RMB exchange rate within $501 - 800$ trading days, where EMGL(6-3) represents $m_N = 6$, $m_J = 3$, etc. We use different configurations of memory length ($m_N = 4, 5, 6; m_J = 3$) and calculate the mean prediction accuracy and its standard deviations. For most of the cases, the EMGL model performs statistically better than the MGDM model (for 8 of 11 stocks and U.S.Dollar-RMB exchange rate). By adding agents who play majority game, we can generate a more realistic market and predict the stock prices more accurately.

From the USD-RMB experiment shown in Figure 2, we can see both EMGL (starred curve) and MGDM (dotted curve) can predict with high accuracy (the mean accuracy is up to 58.6% for MGDM and 63.8% for EMGL (4-3)), indicating a strong existing pattern captured by the new models. In general, almost all results of MGDM and EMGL are statistically better than the random guess (the mean is around 50% with a small variance) plotted at the bottom.

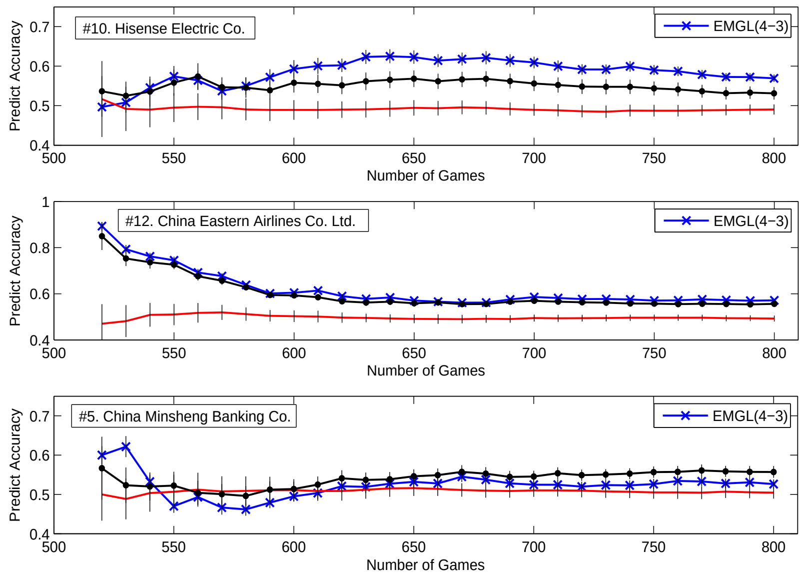

Fig. 3. Performance of the MGDM model and the EMGL model on three representational stocks # 10, 12 and 5.

Figure 3 shows the performance on stock # 10, 12 and 5, which are three representational results in the experiments. Like the experimental results on USD-RMB exchange rate, the test on stock # 10 shows that the EMGL model outperform the GMDM model and both models outperform the random guess. 7 of 12 data (# 1, 3, 6, 8, 9, 10, 11) have similar performance. In the experiments on stock # 12 (and 7), the EMGL model and the MGDM model have similar accuracy (which are not statistically different from each other) and both models outperform the random guess. For stock # 5 (and 4), two prediction curves are overlapped, therefore we are not able to tell which model is statistically better. For stock # 2, the MGDM model outperform the EMGL model which means that the minority game modeling could be more appropriate than mixed-games in this case. The stock prices are driven by complex behaviors and influenced by many unknown factors, it is hard to tell what sort of micro-behavior could be more appropriate than others. However, empirical results on these data have shown that the proposed learning framework of collective data decomposition is effective in solving this difficult problem. Though the EMGL model performs statistically better than the MGDM model for most of the cases in our experiments, we still need to be cautious about choosing between the MGDM model and the EMGL model (as well as different configurations of memory lengths), the performance of these two models may vary with specific stocks when making predictions of the market. The computation time of 30 rounds of GAs on the given 12 dataset is about 5 hours using the Matlab code on an Intel Pentium dual-core PC.

5 Conclusions

In this paper, we proposed a novel learning framework of considering the collective market behavior is an aggregation of several subgroup of agents' behaviors based on the mixed-games model. By using GAs to explore all the possibilities of decomposition of the system, the new model is capable in predicting time-series data and make decisions to maximize its profits. We tested the EMGL model on a few real-world stock data and the USD-RMB exchange rate. For most of the cases, the EMGL model performs statistically better than the MGDM model and both models perform significantly better than a random guess. The future work will focus on obtaining the real returns on more stocks in market. We are also interested in analyzing the correlations between different memory length configurations of the EMGL model.

Acknowledgment

This work is partially funded by the NCET Program of MOE, China, the SRF for ROCS and the China Scholar Council.

References

1. D. Challet and Y. Zhang. Emergence of cooperation in an evolutionary game. *Physica A: Statistical and Theoretical Physics.* Volume 246, Issue 3-4, pp.407-418. (1997).
2. D. Gode and S. Sunder. Allocative efficiency of markets with zero-intelligence traders: Market as a partial substitute for individual rationality, *Journal of Political Economy*, 101(1), pp.119-137. (1993)
3. C. Gou. Dynamic Behaviors of Mix-game Models and Its Application. http://arxiv.org/abs/physics/0504001 (2005)
4. C. Gou, Agents Play Mix-game. Econophysics of Stock and Other Markets, Lecture Notes in Computer Science, Part II, pp.123-132. (2006)
5. N. Johnson, P. Jefferies, and P. Hui. *Financial Market Complexity*, Oxford University Press, Oxford. (2003)
6. G. Li, Y. Ma, Y. Dong, Z. Qin. Behavior learning in minority games, to appear in *Collaborative Agent-REsearch and Development International Workshop(CARE)* LNAI 6066. (2011)
7. Y. Ma, G. Li, Y. Dong, Z. Qin. Minority Game Data Mining for Stock Market Predictions. *The Proceedings of ADMI 2010*, LNCS, Volume 5980, pp.178-189 (2010)
8. R. Mantegna and H. Stanley. *An Introduction to Econophysics: Correlations and Complexity in Finance*, Cambridge University Press. (1999)
9. Z. Qin. Market mechanism designs with heterogeneous trading agents, *Proceedings of Fifth International Conference on Machine Learning and Applications (ICMLA)*, pp.69-74, Orlando, Florida, USA. (2006)
10. R. Savit, K. Koelle, W. Treynor, and R. Gonzalez. *Collectives and the Design of Complex System*, Edited by Tumer and Wolpert, Springer-Verlag, pp.199-212. (2004)