

Human Agents and Intelligent Agents: An Experiment on the Internet

Jerome Yen
Department of Computer Science
The University of Hong Kong
Pok Fu Lam Road, Hong Kong
e-mail: jyen@cs.hku.hk

Abstract

To understand the strengths and limitations of software agents is very important. Many issues and limitations have been discussed in the literature, for example, software agents have limited learning capability. However, if we could see such limitations in a controlled environment and measure how such limitations affect the agents' performance. Such research can be extremely helpful to design better software agents. We developed an artificial market with networked computers to simulate the environment of Internet. This artificial market allows players, such as, humans or software agents, to compete in a simulated dynamic market.

We conducted two sets of experiments to study the limitations of software agents. The first set of five experiments were used to identify the knowledge or strategies that human subjects applied in competition. We classified the knowledge or strategies into four levels: perception, uses of simple model and heuristics, application of long-term strategies and psychological tactics, and prediction or forecasting. The second set of five experiments studied how much can humans transfer their knowledge to the software agents.

In all the experiments, subjects received monetary rewards. From the statistical analysis of the decisions of humans and software agents, we have identified several limitations to the software agents. For example, some subjects used very complicated strategies and psychological tactics in earning high profits. However, they could not transfer such knowledge to the software agents. From interviews and questionnaires we confirmed such findings. Based on these findings, we conclude that software agents are suitable for well-structured and repeating tasks and not suitable for critical and high-risk task, such as, investment.

1. Introduction

The advances in information technology, the growing complexity of work, the increasing dependence on the Internet, and the intensifying competition have pushed the demands for new tools or systems. The major purpose is to use such tools or systems to remove the burdens of humans from those tedious and repeating tasks so that humans can focus on higher value-added tasks. One of such applications is the software agents or if they have additional capabilities, such as, learning and

communication, they can be called intelligent agents. As stated by Marvin Minsky [9]:

"My on-line Random House dictionary includes many senses for agent:

1. a person or business authorized to act on another's behalf
2. a person who works for or manages an agency
3. a representative of a business firm, especially a traveling salesman

When you use the word "agent" in any of these ways, there is no implication that what the agent does is simple. They all suggest the agent is seen as having some specialized purpose. If you need help with making investments, you call a financial agent. If you're looking for a job, you call an employment agent. In the present-day jargon of computer interface agents, the word is used for a system that can serve as a go-between, because of possessing some specialized skills."

The agent approach to support intellectual work, collaboration, information searching and data analysis over the Internet has created many successful applications and gained significant attention over the past few years. Aparicio has identified eight major application areas that relate to agents [1]:

1. Agents that manage systems and networks
2. Agents that control and manage the network access
3. Agents that process mails and messages
4. Agents that assist in information access and management
5. Agents that collaborate with other agents
6. Agents that assist in workflow and administrative management
7. Agents that support electronic commerce
8. Agents that create user interface

These applications represent a fundamental shift in the human-computer interaction paradigm. As stated in a recent Science article: "...the agent approach transforms the computer user from a worker into a manager [12]. You delegate tasks to a set of agents who do things for you." An agent can repeatedly help searching for business counterparts, monitoring stock prices or currency exchange rate, or making recommendations for when to sell or to buy. The basic idea of the agent

research is to develop software systems which "engage" and "help" all types of end users [11].

Broadly defined, an "agent" is a program that can operate autonomously and accomplish unique tasks without direct human supervision (similar to human counterparts such as real estate agents, travel agents, etc.). Such agents might act as "spiders" on the Internet and look for relevant information [2], schedule meetings on behalf of executives based on their constraints or the patterns the agent learned previously, or filter news group articles based on the "induced" (or learned) users' profiles [8]. Agents receive assignments and instructions from the owners, then based on the models or knowledge, accomplish the assignments. Some agent researchers attempt to address the question: "How should agents interact with each other to form digital teamwork on Internet?" Other researchers are more concerned about designing agents which are "robust", "intelligent", and "human-like" [11].

However, to understand the limitations of software agents is also an important research. There are many issues and limitations have been discussed in early literature. For example, software agents have limited learning ability and they do not have emotion [9]. However, to see how such limitations affect the performance of agents in a controlled environment is even more important. For example, emotion is not an important issue if an agent is designed for searching home pages. However, if such software agent is designed to negotiate contract or teach children to learn a particular subject, emotion can be important. Sad, happy, curious and angry are the most direct responses to students' performance. Therefore, an understanding of such limitation and the nature of the target problem is extremely important to the development of a successful software agent.

In order to explore such dependence and the limitations of software agents, we developed an artificial market with networked computers to simulate the environment of Internet. This artificial market allows players, such as, human agents or software agents, to compete as sellers. The payoff to each player, either a software agent or a human, depends on the quality of decision and the ability to predict the behavior of the market and the competitors.

Importance research issues of software agents research will be discussed in Section 2. Definition and discussion about dynamic market will be discussed in section 3. Section 4 will be used to discuss the artificial market and experiment procedure. Data analysis will be provided in section 5. This paper is concluded with a discussion about the future research.

2. Are Intelligent Agents Intelligent? Some Issues to Agents Research

The agent research faces the same problems that artificial intelligence faced over the past. Especially when people started to call such systems intelligent agents. There are many definitions to the artificial

intelligence, here is a famous one: it is the study of ideas that enable computers to be "intelligent" [13]. Another way of defining artificial intelligence is that it is a subfield of computer science which aims to construct agents that exhibit aspects of intelligent behavior [14]. But, what is "intelligence"? Is it the ability to learn and to reason? Is it an ability to apply the knowledge it learned? It is an ability to do something that needs some judgment and evaluation? It is no doubt that these are all parts of what human beings think what intelligence supposed to be.

To create an "agent" that able to take over tasks from humans, which must be more powerful than an expert system or a decision support system, is the goal to artificial intelligence research. However the research in agent was never became an important research area until the late 1980's [6]. Since then, this area grew very fast and attracted many researchers. Agents or intelligent agents are now widely studied not only by computer scientists, but also by people who are in the related areas, such as, Internet and robotics. Both industry and academia are now taking agent research very seriously, such as, IBM[1].

However, there is a major concern to the agent research. Agent can be an oversold topic just like artificial neural network did in the 1960's. Main streams artificial intelligence research, which include expert systems and neural networks, has not been very successful and hope that use a new name could help them find a way to escape [5]. Similarly, many researchers who were not in this domain, just thought that agent is a hot topic and anything that labeled "agent" sells. However, most of applications that labeled "agent" were nothing more than an expert system or a "C" program. What was the basis that they claimed that the system they created are "intelligent agents"?

A weak notion of agency: perhaps the most general way in which the term "agent" is used to denote a hardware or software-based computer system that possess the following properties [14]:

1. Autonomy: agents should operate without the direct intervention of humans or others, and have some kind of control over their actions and states.
2. Social ability: agents interact with other agents (and possibly humans) via some kind of agent-communication language.
3. Reactivity: agents perceive the changes in the environment and respond in a timely fashion to the changes.
4. Pro-activeness: agents do not simply act in response to their environment, they are able to exhibit goal-directed behavior by taking the initiative.

A stronger notion of agency: For some researchers "agent" should have a stronger and more specific meaning than the above four properties. These researchers generally imply that an agent to be a system that, in addition to the properties identified above, should have more properties which only possessed by human. In our first set of experiment, we will show that

all the above properties can be found in the decisions of human subjects. However, the last property was difficult to find in the decisions of the software agents.

3. A Dynamic Market

There is no market in the world is a static one. There are new players join or leave the market every year. Some players are aggressive and some are not. Some changed marketing strategy every quarter and some never changed. In such market, forecast demand of the market, understand the behavior of the opponents, predict their movements, and make an appropriate decision are very difficult and complicated, however, important[3]. Especially when the interactions and competition among the players are significant as, for example, in the automobile and petroleum industries. Much information needs to be considered, for example, strategies and past history of the competitors, available resources, forecasting of the future, goals of the organization, uncertainties and risks, etc. [4]. Sometimes these factors, for example, the strategies of the competitors, are not easily to be learned. However, in order to be a winner, an understanding of all of the above factors is extremely important, especially the behavior and strategies of the competitors.

With computers and experience, some decision makers can do good jobs and help their companies earn high profits. However, can we develop "decision agents" to replace the decision makers in doing such tasks? If not, what are the limitations? In this study we wish to use networked computers to create an artificial market for search for the answer.

In order to see the interactions and the competition among the players, we chose an oligopoly model to control the behavior of the market. One important aspect of an oligopoly is the interactions among the producers are so significant that they cannot be ignored [10]. Also the supply-demand curve determines the relationship between productivity and price. If one player becomes very greedy, the others may suffer from having a low transaction price. How to punish or discourage such behavior is part of the strategies and rules that players need to develop.

As we see in the real world, some companies sacrificed the short-term profits for greater market share, for example, the Japanese automobile industry, the VLSI chip manufacturers in the U.S., etc., and some played psychological games with their competitors, for example, the competition between Windows NT and OS2. Such competitions are complicated and their strategies are difficult to be explained with rules or theories. However, through experiments, we could observe the development and use of tactics or strategies as used in the real world even in such a simple market.

The control rules of our dynamic market are very simple. The production cost is a linear function of production level. There is no time lag between producing and selling the products. No inventory or backlog is allowed. The profit at each trading period of

each player can be simply determined by the transaction price, production cost, and production level. However, according to the supply-demand relation the market price also depends on the total outputs from all the producers. If there are more products sent to the market, the price dropped. If transaction price is higher than the cost, players make profits.

4. Experiment Design

We designed and implemented an artificial market, which is an electronic laboratory that contains several IBM Compatible PCs connected by local area network to a RS-6000 server. Every computer in this artificial market can access to the Internet through a gateway. All these PCs run Microsoft Window which allow the user to use Netscape or Mosaic to browse WWW home pages or telnet to a remote site. The artificial market was developed with C language.

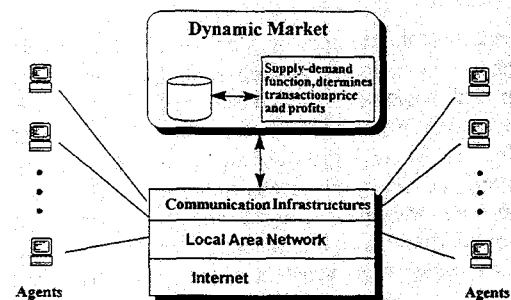


Figure 1. Architecture of dynamic market

A group of five subjects participated in each multi-period experiment as sellers. No buyers participated in these experiments, since the demand function governs the behavior of the market. The demand function determines the market price based on total productions from all the sellers.

First Set of Experiment

Subjects were undergraduate students. They were randomly recruited from campus class rooms and did not required to have a major in either economics or business. It was interesting to see that, from our earlier experiments, the students who majored in business administration or economics did not any significant advantage over the others. All the subjects were promised \$5.00 for showing up for an experiment, but only the first five students are selected to participate. Normally, a dynamic market experiment with fifty trading periods lasts two hours. In order to facilitate the calculations of production levels and the possible profits, subjects are allowed to use calculators. However, the environment for the second set of experiments allowed the subject to design rules and strategies to develop an software agent and therefore the calculators were not very important.

Once a group of five subjects is formed, instructions to the experiment shown on the screens. One or more experiment monitors are available to answer questions. In order to avoid the formation of coalitions, subjects

were told to remain silent and absolutely no communication was allowed throughout the entire experiment. Each of the first set of experiments has fifty trading periods, and the length of the experiment was not revealed to the subjects. Students were also told that they will be interviewed and have to answer a questionnaire at the end of the experiment.

Subjects were given an up-front capital payment. If the accumulated capital of any subject becomes negative, the subject is bankrupt and asked to leave the experiment. The first three periods were practice periods, and no monetary rewards given to the subjects. The experiment subjects sit in separate carrels. Each subject is fed the instructions for the experiment through the terminal. Individual production levels of the other subjects and the total production level as well as the market (transaction) price of the last trading period were presented to the subjects. However, the subjects still be able to trace the production levels and market prices of earlier trading periods. The decision on the production level, the only decision variable, was determined by the subject either based on calculation or judgment.

When all the players submitted their decisions, the host computer sums up the productions, determines the market price, calculates the profit of each player, and added the profits to the total capital to all the players. For all the experiments, the demand function P equals 80 minus the total production enters the market. Therefore, if all the players are aggressive and produce more products, the transaction price may become lower than the production cost C and everyone loses money. The demand function remains the same for all the trading periods.

At the end of each experiment, each subject was interviewed by one graduate student for the decision model or rules, how he or she predicts or estimates the market movement, and any special strategy, tricks or tactics been used. The subjects had to write down the decision model on a piece of paper. A questionnaire was designed to find out what subject think that he or she has done in the experiment. The first four questions were used to find out the background of the subjects (see Appendix I). The next seven questions asked the subjects what models, such as a simple Cournot model (which assumes all the competitors will not change their production levels) or a extrapolative model (which assume the decision is based on what happened in the past few time periods) have been used [10]. There were three possible choice: used in most trading periods, used in some trading periods, and seldom used it. The next five questions were to search for the uses of long-term strategies or special tactics, such as, be aggressive in the beginning to occupy greater market share or punish competitors.

Second Set of Experiment

Each of the first three experiments in the second set has fifty trading periods. They were divided into two parts: the first thirty trading periods played by the subjects and the last twenty played by software agents. When the

first part finished, each subject wrote down her/his strategy and instructed a graduate assistant to create a software agent. Assistants cannot inject any personal preference into the development of software agents. They simply created agents according to the rules or heuristics designed by the subjects. Then assistant entered the rules and heuristics in "C". In order to motivate the subjects to transmit their best strategy, the monetary payoffs for the second part were doubled.

System intentionally put a delay in each trading period to simulate the real-world situation. At the end of experiment, each subject was interviewed by one graduate students for how he or she felt about the differences between humans and software agents. It was also extremely important for the subjects to tell us what limitations they felt about software agents in making decisions for humans. We also asked the subjects to answer a simple questionnaire for their background, such as, their experience with computer as well as their opinion about human agents and software agents (See Appendix II).

During these three experiments, we have identified that subjects were able to constantly improve the rules or strategies. Most of these subjects were unhappy about the decision rules or heuristics they created for the software agents. They thought they could significantly made their software agents much more "intelligence" if we could extend the experiment for another twenty trading periods. When we looked at the rules or heuristics they wrote down on the questionnaires, they were much more complicated or complete than what the software agents had. Therefore, we decided to run two more experiments, which divided the experiment into three parts to further explore the learning behavior.

These two experiments had seventy trading periods and divided into three parts: thirty for the first part, and twenty for the second and third parts. Similar to the first three experiments, subjects had to quickly learn to develop their rules and strategies for the agents. During the second phase, each subject continued improving his or her decision rules or create new heuristics to make software agent more "intelligent". At the end of the second part, with the assistance from a graduate student, each subject quickly coded the new decision rules or heuristics into his or her software agent.

5 Data Analysis

5.1 Data Analysis of the First Set of Experiments

The data sets collected from the five experiments show there is wide range of variations in the subjects' behavior and knowledge in the market competition. Also, there is a significant gap between what subjects thought they did and what they really did. From the questionnaires, we knew that subjects were aware that they used some decision rules. However, it was very interesting to see that most subjects did more than what they could answer in the questionnaires. From statistical analysis, we have identified that some have developed prediction schemes or decision models, some generated good long-term strategies, and some were able to use

psychological tactics to threaten their competitors. However, some of the subjects did not mark them in the questionnaires or write them down in their decision models.

From the subjects' decisions, we used regression analysis to search for the strategies, models or rules. If the level of significance is lower than 0.05, the hypothesis that the subject used a particular strategy is accepted. If the level of significance is higher than 0.05 but lower than 0.1, then our conclusion is that the subject used the particular strategy in some trading periods. For the long-term strategies we could not rely on the statistical analysis. For example, if a subject used

a strategy to occupy greater market share in the beginning, we have to check whether the average production level is higher than his opponents or not. Many of the complicated strategies and decision rules were very difficult to infer from the results of the experiments using regression analysis or other methodologies such as statistical pattern recognition.

Two graduate students were asked individually to run the statistical analysis and judged whether a particular long-term strategy or tactic been used or not. The internal reliability was 0.82 which was higher than the 0.8 as recommended by Krippendorff [6].

Question No.	Constantly	Sometimes	Seldom	Match	Mismatch
Q5 (Use of data)	21	3	1		
Q6 (Able to predict)	11/7	13/9	1/9	12	13
Q7 (Use Cournot model)	7/9	4/7	14/9	13	12
Q8 (Use extrapolative model)	6/8	5/6	14/11	15	10
Q9 (Use two-level model)	4/5	5/6	16/14	16	9

Table 1. Data analysis for using rules and heuristics

Question	Used strategy	Did not use it	Match	Mismatch
Q10 Set up minimum productivity	17/16	8/9	22	3
Q11 Occupy market share	7/8	18/7	23	2
Q12 Punish opponents	6/5	19/20	22	3
Q13 Use psychological tactics	6/7	19/18	23	2
Q14 Send testing signals	7/6	18/19	22	3

Table 2. Data analysis for using strategies and tactics

Exp No-Subj. No	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14	Profits
1-1	Y	N	N	N	N	N	N	N	N	1221
1-2	Y	N	N	N	Y	N	Y	Y	N	2702
1-3	Y	N	N	N	N	N	N	N	N	1119
1-4	Y	N	N	N	N	N	N	N	N	1437
1-5	Y	N	N	N	N	N	N	N	Y	2000
2-1	Y	N	Y	N	Y	N	N	N	N	3066
2-2	Y	N	N	N	N	N	N	N	N	2541
2-3	Y	N	N	N	N	N	N	N	N	1193
2-4	N	N	N	N	N	N	N	N	N	669
2-5	Y	Y	Y	Y	Y	N	Y	Y	N	4139
3-1	Y	Y	Y	N	Y	N	N	N	Y	4086
3-2	Y	Y	Y	Y	Y	Y	Y	Y	N	5370
3-3	Y	Y	Y	Y	Y	N	N	N	N	4890
3-4	Y	Y	Y	Y	N	N	N	N	N	4692
3-5	Y	N	Y	Y	N	N	N	N	N	3110
4-1	Y	Y	N	N	N	Y	N	N	Y	4614
4-2	Y	N	Y	Y	N	N	N	N	N	3794
4-3	Y	Y	Y	Y	Y	Y	Y	Y	N	6350
4-4	Y	Y	Y	Y	Y	Y	Y	Y	Y	5344
4-5	Y	Y	Y	N	N	Y	N	N	Y	4977
5-1	Y	N	N	N	N	N	N	N	N	3768
5-2	Y	N	N	N	N	N	N	N	N	1532
5-3	Y	Y	Y	Y	N	N	N	N	N	4736
5-4	Y	N	N	N	N	N	N	N	N	2740
5-5	Y	Y	Y	N	N	Y	Y	Y	N	4468

Table 3. Summary of first set of experiments

We compared the results of the statistical analysis with the answers that the subjects wrote down on the questionnaires, we have the results in Table 1. For the second, third and fourth columns, the number on the left shows how many been identified from the statistical analysis and the number on the right is how many subject chose on the questionnaire. The fifth and sixth columns indicate how many matches and mismatches in the statistical analysis and subjects' selections.

Table 1 indicates that about fifty percent of the subjects failed in identifying some of the rules or models they used. Over eighty percent of the subjects constantly use data on the screen to make their decision. However, in predicting the movement of the market, majority of the subjects did not do as well as they expected. For questions 10 to 13, the uses of long-term strategies or particular tactics, more subjects were able to correctly identify what particular strategy they used as shown in Table 2.

What we learned from the first set of experiments are:

1. In order to win in the competitions, subjects need to develop and use models, rules, strategies and tactics.
2. For use of models, some subjects did not aware that they used certain models or prediction schemes in making decisions.
3. For long-term strategies and tactics, most subjects were able to tell whether they use them or not.

From the experiments, we also identified that those subjects who were able to quickly learn and develop prediction schemes or models, or be able to use long-term strategies and tactics earned higher profits than those who did not. The following is the table that shows what were the models or strategies been used by the subjects and their total profits. Profits were fictitious money and from which monetary rewards were determined. For data analysis, whenever the level of significance is lower than 0.1, we consider a particular strategy was used.

From Table 3, it is interesting to see that those people who have used models, strategies or tactics out performed those who did not, for example, subject 2 in experiment 3 and subject 4 in experiment 4.

As a summary, winners were those who were able to learn quickly and develop rules or heuristics to help them predict the movement of the market or behavior of competitors. Similarly, subjects were able to play some psychological tricks or developed long-term strategies also earned higher profits. The question and also the major challenge to our research is how much can subjects transfer their knowledge to their software agents and let agents make decision (play the games) for them.

From the first set of experiments, we identified that there are four levels of knowledge or capabilities been used by some subjects:

1. Perceive the changes in the market and the behavior of competitors.
2. Use simple decision rules or models.
3. Use long-term strategies, psychological tricks or tactics and punish opponents.
4. Predict or foresee the transitions of market

This confirms the four properties: autonomy, social ability, reactivity and pro-activeness, recommended by Wooldridge and Jennings [14] for software agents to have.

In the second set of experiments, we would like to study the differences between the human agents and software agents by identifying how much of these knowledge can be transferred to software agents.

5.2 Data Analysis of the Second Set of Experiments

As mentioned in the previous subsection that the abilities or knowledge needed for a winner in a dynamic market can be divided into four levels. The major purpose of the second set of experiments was to explore how much can humans transfer their knowledge to the software agents. These experiments asked the subjects to develop strategies or rules for competing in the dynamic market. Then, with the assistance from a graduate assistant, each subject developed a software agent to play for him. The monetary reward for the trading periods played by software agents was doubled to give subjects a stronger incentive to develop "intelligent" software agents.

Twenty five subjects were divided into two groups: fifteen for the first three experiments where each has fifty trading periods and ten for the last two experiments where each has seventy trading periods. We used the combination of statistical analysis of subjects' and software agents' decisions, questionnaires and interviews to identify the differences of behavior between the human subjects and software agents as well as the limits of software agents.

The First Three Experiments

In each experiment, the subjects played the first thirty trading periods. They had to quickly learn and develop the rules and heuristics for software agents in the next twenty periods. As mentioned earlier, the monetary reward for the last twenty time periods was doubled. The first step the subjects needed to do was to write down their strategies in terms of rules or heuristics, such as the following:

Estimate that each player (subject) produces 9 or 10 units, some of them may send more into the market. However, if there is any one who is too greedy, I would punish him by sending a surprisingly high production into the market. I expect there are 39-40 units from the other players. Based on the total production level in the previous time period, I set my production level at which I may have highest profit. I set my production level between 8 and 10 if the total productivity from my competitors is lower than 45. If the total productivity from the rest of the market is higher than 45, then I

reduce my production level to 5 - 7. However, if I sense that my competitors will reduce their production level, I would set my production level up to 10 or 11. Only when there is a competitor who has been greedy for some trading periods and refuse to give in, I will send a very high productivity, 15 or 16, to tell him or her that someone is unhappy. If the game becomes boring, I will either gradually increase my productivity or suddenly jump up by 3 or 4 units from my previous decision."

The next step, with assistance from a graduate assistant, is to code the rules or heuristics into a simple 'C' program. The graduate assistant spent another half of an hour or so to finish up the program, compiled it and made it ready for the competition. The input that the software agents received were six arrays of numbers: the past decisions of the competitors (four arrays), the total inputs to the market (one array), and the transaction price (one array). The only output from the software agent is the production level.

However, it was not always the case that all the rules or heuristics could be coded. For example, statement like "... if I sense that my competitors will reduce their production level..." was very subjective and judgmental, and was very difficult for the subjects and graduate assistants to transfer such knowledge into rules or heuristics. Let's look at another case, in which the subject moved the competition into psychological and strategic level:

"I started out at an "aggressive" number first, about 25 (which was very high and very risky), so as to find out how the other subjects react and set their production levels. Periodically, every other period or so, no matter what I believed the total production of my competitors to be, I bid a high production level even if I would not be able to make much profit. This might (I hope it would) cause my competitors to drop, because of their expecting me to put in a high number. However, if some one would like to challenge, I will not give in immediately and wait for what happens next. I watched what the production levels of my competitors have been consistently and then see what my decision would cause me to make the most

profit based on my guess of my competitors' decision. I never bid over 30 unless the market price was good in the previous period, because I would risk losing a great deal of profit.

Other than periodically tossing in a higher number (28-32) to set total production level off balance, I bid consistently based on my expectations and I wish no one knows that I am greedy. I do not like to see bloody fight, so that I would gain the most profit from this. With measuring the expectation of the competitors, be sure to be guessing modestly in your expectations of the competitors so as not to lose a lot if the competitors should bid erratically. Acquiescence, if pressed (or losses of profit) two time periods in a row."

It was not easy to write such strategy into rules and heuristics. It took almost an hour for the subject and the graduate student to develop the program for the software agent. However, as she said in the interview:

"I know it was not going to work. Because I have so much strategies and instincts that could not be explained or written in rules. It was even less possible to write them into a program. I just felt sad that the robot (agent) could not earn the same profits as I did by myself."

It is easy to see that this sophisticated strategy was beyond the technical level. It seems that to develop such an effective strategy, good psychological intuition is needed. However, such intuition or knowledge just could not be replicated in a software agents. We did the same analysis as we did for the first set of experiments. Again, we also ran statistical analysis to check if any strategy or model been used for the first thirty time periods and last twenty time periods. We set the level of significance to 0.1 for the use of rules or heuristics. For the uses of long-term strategies or special tactics, we use the same set of questions for the codes as they did in the first set of experiment. For the numbers in each column, the left is for the first thirty time periods and the right is for the last twenty periods.

Exp No-Subj. No	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14	Unit Profits	Total Profits
1-1	N/N	N/N	N/Y	N/N	N/N	N/N	N/N	N/N	N/N	81.8/81.65	2454/1633
1-2	N/N	N/N	N/Y	N/N	N/N	N/N	N/N	N/N	N/N	94.1/83.7	2823/1674
1-3	N/Y	Y/Y	Y/Y	Y/Y	Y/N	Y/Y	Y/N	N/N	Y/N	114.4/93.75	3432/1875
1-4	N/N	Y/N	Y/Y	Y/N	N/N	Y/N	Y/N	Y/N	N/N	107.8/87.8	3234/1756
1-5	N/N	N/N	N/N	N/N	Y/N	Y/N	N/N	N/N	N/N	84.8/71.9	2543/1439
2-1	N/Y	Y/Y	Y/Y	Y/Y	Y/Y	Y/N	Y/N	N/N	N/N	99.3/84.9	2978/1698
2-2	Y/Y	N/Y	N/Y	N/Y	N/N	N/N	N/N	N/N	N/N	104.8/92.9	3143/1859
2-3	N/Y	N/Y	N/N	N/Y	Y/N	N/N	N/N	N/N	N/N	62.5/67.5	1874/1348
2-4	N/N	N/N	Y/Y	N/Y	N/N	N/N	N/N	N/N	N/N	76.4/88.1	2293/1762
2-5	N/N	Y/Y	Y/Y	Y/Y	Y/Y	Y/N	Y/Y	Y/Y	Y/N	125.4/121.9	3762/2439
3-1	N/N	N/N	N/Y	N/Y	N/N	Y/N	N/N	N/N	N/N	116.2/92.4	3487/1847
3-2	Y/Y	N/Y	Y/Y	Y/Y	N/N	Y/Y	Y/N	Y/N	Y/N	95.8/123.4	2874/2467
3-3	N/N	N/N	Y/N	N/Y	N/N	Y/N	N/N	N/N	N/N	77.4/71.9	2322/1438
3-4	N/N	N/N	N/N	N/N	N/N	N/N	N/N	N/N	N/N	51.1/64.5	1534/1289
3-5	Y/Y	Y/Y	Y/Y	Y/Y	Y/N	Y/N	N/N	N/N	Y/N	115.7/119.2	3472/2382

Table 4. Summary of the first three experiments in the second set

From Table 4, we can see that most subjects were able to develop simple rules and heuristics. About one third of the subjects used long-term strategies or special tactics in the first thirty time periods. However, for the last twenty time periods, only five occasions (out of sixty) that both experimenters agreed that such strategies or special tactics been used by software agents and most of them (four out of five) were either used in occupying greater market share or punish the greedy opponents. From this analysis, it is clear that software agents were good in following direction or doing the repeating tasks. However, tasks that require judgment or experience, such as, playing tricks, sending test signals or punish the greedy ones, were difficult and complicated to the software agents.

From the questionnaires, it is even more clear that subjects felt that software agents had so many limitations. For question 8, 12 out of 15 subjects indicate that they could set up rules and heuristics after they completing the first thirty time periods. However, for question 9, only 4 subject felt that they have transferred all the heuristics or rules to the software agents. For question 10, 14 out of 15 said they could do better than the software agents. However, it is interesting to see that there still 5 subjects who received higher average profits in the last twenty periods. For question 11, 13 subjects agreed that some of the strategies or tactics could not be modeled or coded. For last question, the limitations of software agents, 14 subjects selected learning, 11 selected emotion, 12 selected prediction, and 12 selected exceptions handling.

As a summary, among the four types of knowledge, only the first two, perceive the changes and use simple models, been observed in the behavior of the software agents. However, set up strategies, use tricks, and predict or foresee transitions of market were rarely been observed.

The Last Two Experiments

In response to the requests from the subjects who participated in the first three experiments, we ran two more experiments which has more trading periods. Each experiment has seventy trading periods. Subjects played the first thirty trading periods. Similar to the first three

experiments, subjects had to quickly learn to develop their rules and strategies for the agents. The monetary reward for the next twenty trading periods was doubled. During the second phase, each subject continued improving his or her decision rules or create new heuristics to make software agent more "intelligent". At the end of the second part, with the assistance from a graduate student, each subject spent another half of an hour to cod or add the new decision rules or heuristics into the software agent.

From these two experiments, we have observed that the results for the first two phases were very similar to the first three experiments. However, the complexity of the strategies and rules of software agents was significantly increased. One measurement we used to measure the complexity of model is by the number of lines of the program. For the second phase, the average was 32.1 lines, and 46.7 lines for the third phase.

Each experiment took about three hours and over one and a half hours were spent in the development of software agents. We conducted a statistical analysis for the decisions made in all the three stages the same as we did for the first three experiments. The results are shown in Table 5. In each column, the data on the left is for the first phase, data in the middle is for the second phase, and the data on the right is for the last twenty trading periods.

It is interesting to see that the average profit per trading period for phase 1 is 89.55, for phase 2 is 88.11, and for phase 3 is 95.14. It seemed that software agents played better than human subjects. However, if we look at the standard deviations in all the three phases: 27.4, 23.1 and 22.8, as well as the highest and lowest unit profit, we know that the differences in the performance are smaller when the game was played by the software agents. It seemed that some of the good tactics or strategies developed by subjects disappeared when the software agents took over. However, for subjects who did not play well in the first phase, as long as they come out with reasonably well strategies or rules, they could significantly improved their earnings. The conclusion is that use of software agents make competition less competitive and more mechanical.

Exp-Subj	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14	Unit Profits	Total Profits
1-1	Y/Y/Y	Y/Y/Y	Y/N/N	Y/N/Y	N/Y/Y	N/N/N	N/N/N	N/N/N	N/N/N	98.2/74.9/80.8	2946/1498/1615
1-2	Y/Y/Y	Y/Y/Y	N/N/N	N/N/Y	N/N/N	N/N/N	N/N/N	N/N/N	N/N/N	58.8/61.9/68.8	1765/1238/1376
1-3	Y/Y/Y	Y/N/Y	N/N/N	N/N/N	N/N/N	Y/N/N	Y/N/N	Y/Y/N	N/N/N	77.6/73.6/87.2	2327/1472/1743
1-4	N/Y/N	N/N/N	Y/Y/Y	Y/Y/Y	Y/N/N	Y/N/N	N/N/N	Y/N/N	Y/N/N	117.4/125.6/118.6	3521/2512/2372
1-5	Y/Y/N	Y/Y/Y	Y/N/Y	Y/N/N	Y/Y/Y	Y/N/N	N/N/N	N/N/N	N/N/N	92.8/93.7/91.5	2783/1873/1829
2-1	Y/Y/Y	Y/N/Y	Y/Y/Y	Y/Y/Y	Y/Y/Y	Y/N/Y	Y/Y/Y	N/N/N	Y/N/Y	113.8/106.4/117.9	3415/2127/2358
2-2	N/N/N	N/N/N	N/Y/Y	N/Y/Y	N/N/Y	N/N/N	N/N/N	N/N/N	N/N/N	91.4/96.9/110.7	2743/1937/2213
2-3	Y/Y/Y	Y/Y/Y	N/N/N	N/N/N	N/N/N	N/N/N	N/N/N	N/N/N	N/N/N	48.1/61.9/69.5	1443/1239/1389
2-4	Y/Y/Y	Y/N/N	N/N/N	N/N/N	N/N/Y	N/N/N	N/N/N	N/N/N	N/N/N	64.6/69.2/74.9	1937/1383/1498
2-5	Y/Y/Y	N/N/N	Y/N/Y	Y/Y/Y	Y/Y/Y	N/N/N	Y/N/N	Y/N/N	Y/N/N	132.9/117.1/131.7	3987/2342/2634

Table 5. Summary of the last two experiments in the second set

The questionnaires gave us similar results as in the first three experiments. For question 8, 9 out of 10 subjects indicate that they could set up rules and heuristics after they completing the first thirty time periods. However, for question 9, 5 subject felt that they have transferred all the heuristics or rules to the software agents. For question 10, 8 out of 10 said they could do better than the software agents. For question 11, 8 subjects agreed that some strategies or tactics could not be modeled or coded. For question 12, the limitations of software agents, 8 subjects selected learning, 7 selected emotion, 8 selected prediction, and 9 selected exceptions handling.

We added two additional questions to identify the improvement of software agents from phase 2 to phase 3. There were 9 subjects felt that they either strongly agree or agree that they have improved the model of their software agents. However, only 7 subjects strongly agree or agree that their systems did better in phase 3.

Conclusions and Future Research

In this research we used an experimental approach to explore the differences of the behavior between human subjects and software agents. In the first set of experiments, we have identified four different types of properties or capabilities from the decisions made by human subjects in a simulated market competition. They are perception, follow simple models, use psychological tricks, and forecasting or prediction. However, in the second set of experiments, we found that the first two were not easy to be transferred to the software agents. With strong incentive from monetary reward, the subjects did tried their best to create "intelligent" software agents. However, most subjects still felt software agents will never do as good as humans in making decisions which require the last two properties or abilities.

The results confirmed the statements made by the earlier researchers, such as, software agents have limited learning capabilities, software agents have difficulties in handling exceptions, software agents do not have emotion or the emotion. Even though the rules of the simulated dynamic market are simple. However, the behavior of human subjects, for example, some were very aggressive, some were conservative, and some were tricky, made the game very complicated and unstructured.

It is possible to guess the movement of your competitors. However, just like a stock market, in most cases your guess was wrong. We do not think, at this moment or in the near future, that software agents will replace humans in making investment decisions. This was the major reason why we paid the subjects to participate in these experiments. Top subjects could easily earn 50 to 60 US dollars in two hours. The best subject in the last two experiments of the second set earned about 80 dollars. Such incentive made these experiments extremely exciting and subjects worked extremely hard to design the best software agents.

Our conclusion for this research are:

1. Software agents are suitable for tasks which are well-structured and repeating.
2. Software agents are not ready for tasks which are judgmental, critical or high-risk, such as, making recommendation for investment.

References

- [1] Aparicio, G., "The Role of Intelligent Agents in the Information Infrastructure", IBM, USA, 1995. <http://activist.gpl.ibm.com:81/WhitePaper/ptc2.htm>
- [2] Etzioni, O and Weld, D., "A Softbot-based Interface to the Internet", *Communications of ACM*, Vol. 37, No. 7, pp. 72-79, July 1994.
- [3] Friedman, J. W., *The Theory of Games and Oligopoly*. North Holland Publ. Co., Amsterdam/ New York/Oxford, 1977.
- [4] Friedman, J. W., "Oligopoly Theory", Chapter 11 in K. J. Arrow and M. D. Intriligator, editors, *Handbook of Mathematical Economics*, North Holland Publ. Co., Amsterdam, pp. 491-534, 1981.
- [5] Hermans, B., "Intelligent Software Agents on the Internet", Tilberg University, The Netherlands, 1996.
- [6] Krippendorff, K., *Content Analysis: An Introduction to its Methodology*, Sage Press, Newbury Park, CA., 1980.
- [7] Maes, P., "The Dynamics of Action Selection". In *Proceedings of IJCAI-89*, pp. 991-997, 1989.
- [8] Maes, P., "Agents that Reduce Work and Information Overload", *Communications of ACM*, Vol. 37, No. 7, pp. 30-40, July 1994.
- [9] Minsky, M., "A Conversation with Marvin Minsky about Agents", *Communications of ACM*, Vol. 37, No. 7, pp. 23-29, July 1994.
- [10] Okuguchi, K. and Szidarovszky, F., *The Theory of Oligopoly with Multi-Product Firms*, Springer-Verlag, Berlin/Heidelberg/ New York, 1990.
- [11] Riecken, D., "Intelligent Agents", *Communications of ACM*, Vol. 37, No. 7, pp. 18-21, July 1994.
- [12] Waldrop, M. M., "Software Agents Prepare to Sift the Riches of Cyberspace", *Science*, Vol. 265, pp. 882-883, August 1994.
- [13] Winston, P. H., *Artificial Intelligence*, Addison Wesley, 1984.
- [14] Woodridge, M. J. and Jennings, N. R., "Agent Theories, Architectures, and Languages: A Survey". In M. J. Woodridge and N. R. Jennings, editors, *Intelligent Agents*. Springer-Verlag, Berlin, 1995.

Appendix I: Questionnaire I

1. What is your major (marketing/accounting/finance/ economics/MIS/others)?
2. Are you a (freshman/sophomore/junior/senior)?
3. Have you take: (microeconomics/game theory/ artificial intelligence)?
4. What is the major purpose for your attending this experiment (monetary reward/education and learning/have fun)?
5. Were your decisions based on the data presented on the screen (most of the trading periods/some of the trading periods/never watched them)?
6. Were you be able to accurately predict the movement of the market price (most of the trading periods/some of the trading periods/never was able)?
7. Have you used a simple Cournot prediction scheme, which assumes the other people will not change their decisions from the earlier time period (most of the trading periods/some of the trading periods/never used it)?
8. Have you used any extropolative prediction scheme, in which you developed a simple model that used the data in the past few time periods (most of the trading periods/some of the trading periods/never used it)?
9. Have you used a simple two-level model which depend on the transaction price of the previous time period (most of the trading periods/some of the trading periods/never used it)?
10. Have you set up a minimum production level (yes/no)?
11. Did you try to occupy greater market share in the beginning (yes/no)?
12. Did you try to punish your opponents (yes/no)?
13. Did you used psychological tactics (yes/no)?
14. Did you try send signals to test the responses of your opponents (yes/no)?

Appendix II: Questionnaire 2

1. What is your major (marketing/accounting, finance/economics/MIS/others)?
2. Are you a (freshman/sophomore/junior or senior)?
3. Have you taken any of the following courses: (microeconomics/game theory/artificial intelligence)
4. What is the major purpose you attended this experiment (monetary reward/education and learning/have fun) ?
5. How many years have you used computer (one/two/three/more)?
6. How often do you use Internet and WWW (everyday/ twice per week/ once per week / never)?
7. Have you heard of "agent", "intelligent agent" or "software agent" (yes/no)?
8. You were able to set up heuristics or rules after completed the first part of experiment (agree/disagree)?
9. You were able to transfer all the heuristics or rules to the software agent (agree/disagree)?
10. You could do better by playing the game by yourself (agree/disagree)?
11. You have some tactics or strategies could not be modeled in rules or heuristics (agree/ disagree)?
12. What are the major limitations to software agents (learning/ emotion/ prediction/ handle exceptions/ others)?
13. I have improved the models or rules in my software agent from phase 2 to phase 3 (strongly agree/ agree/ disagree)?
14. I feel that the software agent performed better in phase 3 (strongly agree/agree/disagree)?

Note: Questions 13 and 14 only used in the last two experiments in second set