Algorithmic Social Sciences Research Unit

ASSRU

Department of Economics
University of Trento
Via Inama 5
381 22 Trento Italy

DISCUSSION PAPER SERIES

6 – 2011/II

TOWARD AN AUTONOMOUS-AGENTS INSPIRED ECONOMIC ANALYSIS[♠]

Shu-Heng Chen & Tina Yu

JULY 2011

[♠] Text of the talk given at the ASSRU/Department of Economics Seminar, University of Trento, 18 May, 2011. Professor Shu-Heng Chen is a *Founding Honorary Associate* of ASSRU.

Toward an Autonomous-Agents Inspired Economic Analysis

Shu-Heng Chen
Department of Economics
National Chengchi University
Taipei, Taiwan 116
E-mail: chen.shuheng@gmail.com

Tina Yu
Department of Computer Science
Memorial University of Newfoundland
Canada
E-mail: gwoingyu@gmail.com

Abstract

This paper demonstrates the potential role of autonomous agents in economic theory. We first dispatch autonomous agents, built by genetic programming, to double auction markets. We then study the bargaining strategies discovered by them, and from there an autonomous-agent-inspired economic theory with regard to the optimal procrastination is derived.

Keywords: Agent-Based Double Auction Markets, Autonomous Agents, Genetic Programming, Bargaining Strategies, Monopsony, Procrastination Strategy

1 Motivation and Introduction

Economics is about the efficient use of resources, which very much relies on the ability of humans to discover chances and hidden patterns. However, what is lacking in current economic theory is a proper model of the chance-discovering agents. Recent developments in regard to autonomous agents have provided economists with an opportunity to fill this intellectual gap. This is particularly evident in the growing literature on agent-based computational economics (Tesfatsion and Judd, 2006). The massive use of the tools of intelligent agents has placed various kinds of autonomous agents in economic environments so that they can explore their surroundings and make decisions without too much external supervision (Chen, 2008a). Models built using these autonomous agents can, therefore, evolve on their own and changes are no longer placed exogenously, but generated endogenously.

In addition, by studying what kinds of chances or patterns are being discovered by these agents, we, as model-builders, can also better learn about the intricate structure of the models. In this way, autonomous agents not only learn by themselves, but also "instruct" model-builders to learn. Nevertheless, current studies on agent-based economic models largely focus only on the macroscopic level. The microscopic analysis has not been advanced enough to gain insights into the discovery behavior of autonomous agents. In this paper, we will use an agent-based double auction market for which autonomous agents are built by genetic programming to illustrate the challenges posed by

Market-1: Demand-Supply

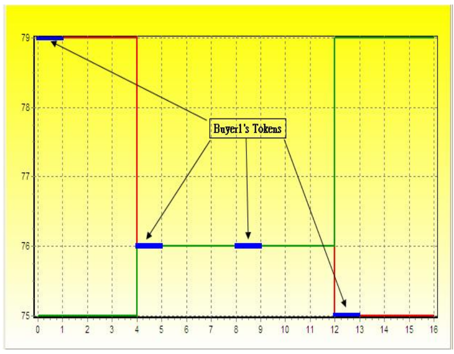
Demand-Supply Information

Demand Curve	28	26	26	25	23	23	22	17	14	14	13	12	12	9	7	6
Supply Curve	1	2	4	7	7	8	9	9	12	17	18	18	20	21	21	23

Token-Value Table

Buyer1	Buyer2	Buyer3	Buyer4	Seller1	Seller2	Seller3	Seller4
23	28	23	26	2	8	1	7
22	26	14	25	7	12	4	9
12	17	9	14	17	21	9	20
7	13	6	12	18	23	18	21

Market-2: Demand-Supply



Demand-Supply Information

Demand Curve	79	79	79	79	76	76	76	76	76	76	76	76	75	75	75	75
Supply Curve	75	75	75	75	76	76	76	76	76	76	76	76	79	79	79	79

Token-Value Table

Buyer1	**Buyer2**	**Buyer3**	**Buyer4**	**Seller1**	**Seller2**	**Seller3**	**Seller4**
79	79	79	79	75	75	75	75
76	76	76	76	76	76	76	76
76	76	76	76	76	76	76	76
75	75	75	75	79	79	79	79

Figure 1: Market 1 (Left) and Market 2 (Right)

The bottom panel is the token-value table, which specifies the reservation price of buyers and sellers for each additional token. The middle panel is a list of all reservation prices, which are arranged, from left to right, in descending order for buyers' reservation prices and ascending order for sellers' reservation prices. The corresponding demand and supply schedule is then given in the top panel.

knowledge discovery to the ecological market dynamics. In addition, through these familiar double auction markets, we shall see why this task can be difficult and will also see how interdisciplinary research can help to make a breakthrough.

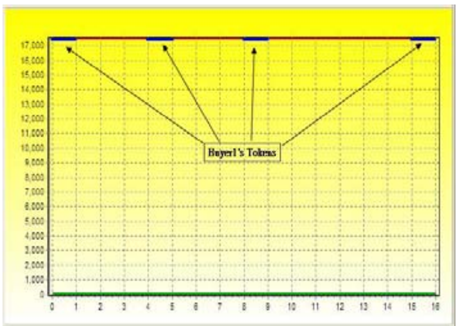
The rest of this paper is organized as follows. Section 2 presents the experimental designs of the paper. We adopt four market designs (four different demand and supply schedules), each with a different equilibrium (or equilibria), so as to generalize what we may be able to learn from our dispatched autonomous agents. Section 3 provides the simulation results. After a short summary, we analyze the best strategies found in each market scenario and try to discover the rationale behind them. A generalization of what we learn from these four cases motivates the theory of optimal procrastination proposed in Section 4, followed by concluding remarks in Section 5.

2 Experimental Designs

2.1 Environment

In this paper, we consider four different schedules of demand and supply, as shown in Figures 1 and 2. Each market has four buyers and four sellers. They are numbered from Buyer 1 to Buyer 4 and Seller 1 to Seller 4. The commodity traded in this market is called the *token*. Buyers value these tokens and their *maximum willingness to pay* (the *reservation price* of buyers) for each token is specified in the *token-value table*. The willingness to pay

Market-3: Demand-Supply



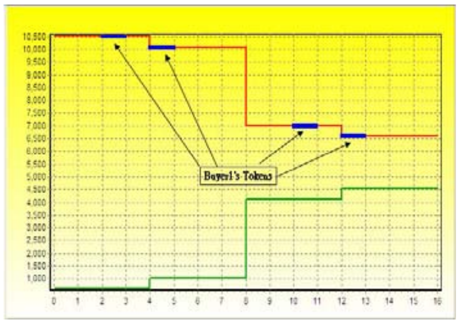
Demand-Supply Information

Demand Curve	17473	17473	17473	17473	17471	17471	17470	17470
Supply Curve	33	34	34	34	34	34	34	34
	17465	17465	17465	17465	17465	17465	17465	17464
	39	39	40	40	42	42	42	42

Token-Value Table

Buyer1	Buyer2	Buyer3	Buyer4	Seller1	Seller2	Seller3	Seller4
17473	17473	17473	17473	33	34	34	34
17471	17470	17470	17471	34	34	34	34
17465	17465	17465	17465	40	39	39	40
17464	17465	17465	17465	42	42	42	42

Market-4: Demand-Supply



Demand-Supply Information

Demand Curve	10521	10519	10518	10516	10073	10072	10071	10071
Supply Curve	618	619	622	622	1010	1013	1014	1016
	6987	6985	6984	6981	6593	6593	6590	6589
	4100	4100	4101	4102	4545	4547	4548	4550

Token-Value Table

Buyer1	Buyer2	Buyer3	Buyer4	Seller1	Seller2	Seller3	Seller4
10518	10519	10516	10521	622	622	618	619
10073	10072	10071	10071	1013	1010	1014	1016
6984	6981	6985	6987	4102	4101	4100	4100
6593	6593	6589	6590	4547	4548	4545	4550

Figure 2: Market 3 (Left) and Market 4 (Right)

The bottom panel is the token-value table, which specifies the reservation price of buyers and sellers for each additional token. The middle panel is a list of all reservation prices, which are arranged, from left to right, in descending order for buyers' reservation prices and ascending order for sellers' reservation prices. The corresponding demand and supply schedule is then given in the top panel.

is non-increasing with the number of tokens already owned. For example, in Figure 1, Market 1, for Buyer 1, the maximum willingness to pay for the first token is 23, then 22 for the second, 12 for the third, and 7 for the fourth. A similar structure holds for other buyers. On the other hand, sellers would like to provide these tokens and the *minimum acceptable price* (the *reservation price* of the seller) for each token is also specified in the token-value table. As the opposite of the maximum willingness to pay, the minimum acceptable price is non-decreasing with the number of tokens already sold. Let us use Seller 1 in Market 1 (Figure 1, Left) as an example. The minimum acceptable price starts from 2 for the first token, then 7 for the second, 17 for the third, and 18 for the fourth.

This structure of the token-value table is generated in light of the familiar behavior of marginal utility and marginal cost and hence it fits well with the law of demand and supply. If we pool all the maximum willingness to pay and the minimum acceptable price together, and arrange then in descending order and ascending order separately (as shown in the middle panel of each figure), then we can draw a downward-sloping demand schedule and upward-sloping supply schedule, as shown in the top panel of each figure.

Since different demand and supply schedules may have different effects on the behavior of bargaining, we try to make our demand-supply schedules different so as to explore this sensitivity. The first two cases (Figure 1) have demand and supply cross so that the equilibrium price is unique. Their difference lies in the multiplicity of the equilibrium quantity. For the first case, this is also unique, whereas for the second it has multiple equilibria. The last two cases (Figure 2) differ from the first two in that they never cross, so their equilibrium price is not unique, but has an equilibrium quantity. Between the two, differences exist in the distance between demand and supply curves. For Market 3, the demand and supply schedules are parallel to each other from the left end to the right end, and the distance between the two changes little, whereas for Market 4, the demand curve is piecewisely step-down and the supply curve is piecewisely step-up, and the distance between the two, therefore, gets smaller. It is not quite clear how these different topologies may impact the bargaining strategies. At the outset, we simply wonder and let our simulation results enlighten us.

One autonomous agent will be placed in this environment, and it will play the role of Buyer 1 (see Section 2.4 for more).[1] *The trading behavior of the autonomous agents are the focus of this study*. In particular, we ask how genetic programming enables these agents to discover profitable trading strategies and what they discover, a question already raised in Chen (2008).[2] To better understand the nature of this question, consider the case of one autonomous agent. The four reservation prices (the maximum willingness to pay) of Buyer 1 are highlighted in each figure. From this mark, we can immediately see the original competitive advantage of Buyer 1 relative to other competitors (buyers and sellers). The question is what the optimal bargaining strategy is given this competitive position, be it superior or inferior, and how genetic programming can help the autonomous agent to discover this strategy.

2.2 Bargaining Strategies

By a bargaining strategy, we mean an *algorithm* to indicate the kind of information required for making a decision (the inputs), and how the decision is made based upon the information received (the outputs), or, alternatively, a *process* to connect inputs to outputs. Of course, that would in turn depends on the inputs available for the artificial agents. In this paper, we follow the experimental design of the *Santa Fe Double Auction Tournament* (Rust, Miller, and Palmer, 1993, 1994) and make the information as summarized in Table 1 available for traders. Basically, three types of information are available for our autonomous agents: price or quote information (indexes 1–9 and 16–17), time information (indexes 10–11), and private information (indexes 12–14).

This information is rather concise compared to what human subjects will normally have from laboratory experiments, where they may get access to all information on the previous trading "days" and "rounds". What we have here is only partial information on the previous day and previous round. However, this condensed information may be sufficient given what we learned from Rust, Miller, and Palmer (1993, 1994). In fact, some later developed artificial double auction systems are also built upon this "minimal" information set (Andrews and Prager, 1994).

This information set defines part of the algorithms which may be discovered by the autonomous agents, but it is just a collection of primitive inputs. To process this raw data,

[1] This fixed design is simple and make our analysis coming later easier.

[2] In an experimental setting, the same kind of question has been asked by Rust, Miller, and Palmer (1993) and Rust, Miller, and Palmer (1994). The difference is that they asked how *human agents* discover or design the trading strategies, but here we are addressing how *software agents* do that. For other subtle differences, the interested reader is referred to Chen and Tai (2010).

Table 1: Information Available for Traders (Terminal Set)

Index	*Terminal*	*Interpretation*
1	PMax	the highest transaction price on the previous day
2	PMin	the lowest transaction price on the previous day
3	PAvg	the average transaction price on the previous day
4	PMaxBid	the highest bidding price on the previous day
5	PMinBid	the lowest bidding price on the previous day
6	PAvgBid	the average bidding price on the previous day
7	PMaxAsk	the highest asking price on the previous day
8	PMinAsk	the lowest asking price on the previous day
9	PAvgAsk	the average asking price on the previous day
10	Time1	the number of auction rounds left for today
11	Time2	the number of auction rounds that have no transaction
12	HT	the highest token value
13	NT	the second highest token value
14	LT	the lowest token value
15	Pass	pass the current auction round
16	CASK	the lowest asking price in the previous auction round
17	CBID	the highest bidding price in the previous auction round
18	Constant	randomly generated constant number

Table 2: Logic and Mathematical Operators (Function Set)

		Function		
+	-	*	%	min
>	exp	abs	log	max
sin	cos	if-then-else	if-bigger-then-else	

some further operations are expected. Normally this can be done by allowing agents to obtain access to some logical and mathematical operators, and Table 2 provides a list of these options.

Given the information (Table 1) and the way to operate it (Table 2), various bargaining strategies can be formed. Two examples are given as follows:[3]

- `(Min PMinBid HT)`
- `(If_Bigger_Then_Else HT CASK CASK+1 Pass)`

In the first example, to decide how much to bid, the buyer simply looks at the minimum bid on the previous day (PMinBid) and his current reservation price (HT), and bids at the minimum of the two. In the second example, the buyer first checks whether his reservation price (HT) is bigger than the lowest ask (CASK) in the previous round. If this condition is met, he will bid by adding one dollar to the current ask; otherwise, he will simply pass. Not all bargaining strategies are that simple. A little knowledge of

[3]The generation of these examples is based on the grammar of the formal language, in particular, the *context-free grammar*. The famous *Backus-Naur Form* (BNF) is extensively used in the literature and is also applied here.

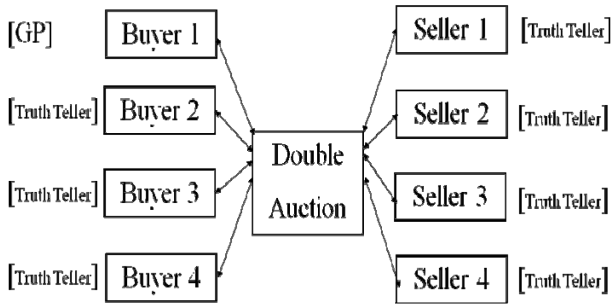


Figure 3: Composition of Market Participants

combinatorics or context-free grammar will lead us to see that the formed bargaining algorithm can potentially become complex like the next one.

- `((Min (If_Bigger_Then_Else PMinBid PAvgBid CASK PAvgBid)`
  `(If_Bigger_Then_Else HT PAvgBid PAvgBid CASK))`

2.3 Institutional Arrangements

One may question whether the syntax developed above can be *semantically* meaningful as well. That depends on the kind of institutional arrangements. In a double auction market, both buyers and sellers can submit bids and asks. This contrasts with only buyers shouting bids (as in an *English Auction*) or only sellers shouting asks (as in a *Dutch Auction*). There are several variations of DA markets. One example is the *clearinghouse* DA of the Santa Fe Token Exchange (SFTE) (Rust, Miller, and Palmer, 1993) on which this work is based.

On the SFTE platform, time is *discretized* into alternating *bid/ask* (BA) and *buy/sell* (BS) steps. Initially, the DA market opens with a BA step in which all traders are allowed to simultaneously post bids and asks for one token only. After the clearinghouse informs the traders of each others' bids and asks, the holders of the *highest bid* and *lowest ask* are matched and enter into a BS step. During the BS step, the two matched traders carry out the transaction using the *mid-point* between the *highest bid* and the *lowest ask* as the transaction price. Once the transaction is cleared, the market enters into a BA stage for the next auction round. The DA market operations are a series of alternating BA and BS steps.

Our introduced syntax of the bargaining strategy works well with the SFTE platform because for this platform the decision only involves a bid or an ask for a single unit of token. There is no involvement of *market time*, which certainly exists in the continuous-time double auction, and there is no involvement of *units* when each transaction allows for one unit as the maximum.

2.4 Opponents' Behavior

In this paper, to make our later analysis simple, we assume that all opponents are *truth tellers* except one autonomous agent (the first buyer) (see Figure 3). Being a truth teller,

Table 3: Experimental Designs

# of Autonomous Agents (Code)	Markets (Code)	Cognitive Capacity (Code)
One, Buyer 1 (B1)	1 (M1), 2 (M2), 3 (M3), 4 (M4)	10 (P10)
	1 (M1), 2 (M2), 3 (M3), 4 (M4)	50 (P50)

Inside the brackets is the code of the corresponding design. So, for example, Case B1M1-P10 refers to the design involving one GP buyer in Market one, and the population size of the GP buyer is set to 10. Case B1M4-P50 refers to the one involving one GP buyer in Market 4, and the population size is 50.

the trader simply bids or asks at his current reservation price (HT for buyers and LT for sellers). This simplification makes it easier for us to make sense of the behavior of autonomous agents and evaluate their novelty-discovery capability. We start with the simplest case, only one one autonomous agent (the left panel of Figure 3) to gain some basic understanding of the bargaining strategies discovered.[4]

The inquiry above is addressed based on multiple runs of four designs. The four designs differ according to the population size of GP being 10 or 50. Each of the designs is implemented with the four demand-supply schedules. A summary is given in Table 3. 90 runs are conducted for each market under each design. Hence, a total of 720 ($2 \times 2 \times 4 \times 90$) runs are completed. For the purpose of running GP, each run lasts for 300 generations (iterations).

2.5 Genetic Programming

Autonomous agents in this paper are programmed by genetic programming. Genetic programming is a population-based stochastic search algorithm. The population is composed of a number of chromosomes. In different applications, these chromosomes represent different things. In our case, each chromosome simply represents a bargaining strategy as described in Section 2.2. The grow method is used to generate an initial population of bargaining strategies. Each initial strategy can have a hierarchy with a depth of up to 5. The population size is set to 10 and 50. The population is time-variant; it evolves for a given duration (a given number of generations), which is 300 in our setting. To have effectively equal sampling for evaluating two populations with different sizes, each generation lasts $2 \times pop\_size$ (population size) days.

The evolving population is driven by a sequence of genetic operators, which introduce a few more parameters. The selection mechanism is the tournament selection with a tournament size of 5. To avoid disruption due to genetic variations, the elitism operator is triggered to automatically keep the best strategy discovered in the previous generation in the next. Finally, the new bargaining strategies are discovered through crossover (recombination) and mutation. The crossover rate is 100%. There are two styles of mutation,

[4]Co-evolution will complicate the situation when one more autonomous agent is added to the market. While one may be interested in knowing whether the market co-evolves toward Nash equilibria when there are two autonomous agents, given the infinite number of possible bargaining strategies, it would be very difficult to identify Nash equilibria in a strategy space.

Table 4: GP Parameters

Parameter	*Value*	*Parameter*	*Value*
Initialization method	grow	initial maximum tree depth	5
population size	10,50	no. of generations	300
number of days	2×pop_size	tournament selection size	5
elitism size	1	crossover rate	1.0
point mutation rate	0.045	subtree mutation rate	0.005

point mutation and subtree mutation. The mutation rate for each is 0.45 and 0.05, respectively. Table 4 gives the GP parameter values used to perform simulation runs.

3 Novelty-Discovering Agents

To "appreciate" what our GP buyers have found, we need to be reminded that these GP buyers are not much different from ants in the sense that they are almost "blind" without knowing what the market structure is, both on the demand side and the supply side. The only thing known to them is what are given in Table 1. Hence, they are placed in a much more disadvantageous situation human agents that were placed in the double auction market experiments. Yet, their opponents (the programmed agents), from the beginning to the end, following the same trading rules to trade, round to round, have created a very friendly environment as in the movie *Ground Hog Day* for these GP agents.[5] Novelty-discovering agents will not always be rewarded by their constant attempt to discover hidden patterns, but when these "hidden" patterns are just there it is just a matter of time for these agents to discover them. With this remark, we hope that the following description of what we have learned from GP agents will not be considered to be trivial due to human *hindsight bias*.

3.1 Data Processing and Analytical Procedure

As mentioned earlier, the analysis below is not based on any single run, but on 90 runs for each market with each design. This leaves a huge amount of data in front of us. For example, for each single run, depending on the population size, one can have 6000 ($300 \times 2 \times 10$) or 30,000 ($300 \times 2 \times 50$) strategies being observed in the auctions.[6] By multiplying them by 90, one would have a total of 540,000 or 2,700,000 strategies. This huge amount of data inevitably drives us to take some steps to make it tractable. What we do here is the following. We drop off the first 290 generations and focus only on the last 10 generations. The usual justification to have this convenience is something related to ergodicity, which we found quite applicable to our environment when the simulation has run for such a long time. Restricting our analysis to the last 10 generations will then reduce the set of observable strategies to "only" a size of 18,000 (pop_size=10) or 90,000 (pop_size=50) separately. This reduced sample is still large enough to provide a valid

[5] See Thaler (2000) for the use of this metaphor.

[6] As mentioned in Section 2.5, the number of iterations for each generation is composed of 2×pop_size (population size) trading days.

Table 5: Distribution of daily profit generated by 18,000 strategies (pop_size 10).

Profit	-0.5	0	8	10.5	12.5	14	14.5	17	18	18.5	21	Total
Count	18	429	346	5,666	72	8	3,829	16	4	335	7,277	18,000

Table 6: The 3 most frequently-used strategies and the associated information (pop_size 10).

Strategy	Profit	Count	Ratio (Count/18,000)
`PMinBid`	21	7,277	0.4043
`PMinBid`	10.5	5,666	0.3148
`HTV`	14.5	3,829	0.2127
Total		16,722	0.9318

answer to the question: *what did the autonomous agent learn and what is the effect of a larger population size*?

The second step which we take to deal with this large amount of data is to focus our attention on some most frequently-used strategies. An alternative is to first generate a profit distribution over the reduced dataset. For example, Table 5 gives the distribution of daily profit generated by the set of 18,000 strategies in the case of B1M1-P10. As shown in the table, the strategies that generated the profits 21, 10.5 and 14.5 were used for a total of 93% of the auctions. It is therefore reasonable to assume that they represent the GP buyer's trading strategies. Table 6 presents these 3 strategies and their associated information.

The most highly-used strategy is (`PMinBid`): the lowest bidding price on the previous day. The other highly-used strategy is (`HT`): the highest token value, which is just the truth-telling strategy. In this way we can then have a general idea of what the autonomous agent discovered, and why it is so; we as experimenters can then also learn (Chen, 2008).[7] Finally, the procedure suggested above will be carried out on both the cases of P-10 and P-50 so that a comparison between the results of the two will enable us analyze how the autonomous agents behave differently so as to see the effect of population size.

3.2 General Results

Since there are a total of eight scenarios to be discussed, it would be easier to have a general picture first, and then to get into some specific results later. Table 7 provides such a summary of the effect of a larger population size. The column "benchmark" gives the major bargaining strategies discovered by the autonomous agents when the population size is set to 10 (pop_size=10). These major strategies are identified based on the procedure given in Section 3.1. For example, as we have seen earlier, the key strategy discovered by the autonomous agent in Case B1M1 is (`PMinBid`). The next column "Differences" then shows the essential differences after the population size increased to 50 (pop_size=50).

As we summarize in this table, there are a number of things that we are looking at, namely, *profitability*, *stability* (or *robustness*) and *complexity*. They are our focuses because

[7]This is a response to the criticism that *agents learned but we do not*.

Table 7: A Summary of the Results

Case	Benchmarks (P-10)	Alternative (P-50)
B1M1	`(PMinBid)`	`(Min (HT PMinBid))`, `(P-22)` profitability ↑, stability ↑, complexity ↑
B1M2	`(NT)`	`(NT)` No Effect
B1M3	`(CASK)`, `(PMaxAsk)`, `(PMin)`	`(CASK)` ↑, `(PMaxAsk)` ↓, `(PMinn)` ↓ stability ↑
B1M4	`(CASK)`	`(CASK)` complexity ↑

the fundamental question to address is: *Does a larger population size lead to the discovery of better bargaining strategies, better in the sense of higher and stable (robust) profits?* If the answer is *yes*, we further ask: *what may cause this difference?* The answer to the second question hinges upon *complexity*.

A larger population size makes it easier for autonomous agents to find the profitable bargaining strategies which are complex and are normally beyond the availability of the agents with a smaller population size. Complex profitable bargaining strategies are observed in both Markets 1 and 4. For example, in Market 1, when the population size increases to 50, a new, *better*, and more complex strategy, `(Min (HT PMinBid))`, is discovered. In fact, this is not the only improvement being discovered. A class of new strategies, called `P-22`, is also discovered.[8] Nevertheless, some of the better but more complex strategies discovered in the course of evolution did not get stabilized and remain to the end. Market 4 has several such examples. Hence, the best surviving strategy in both cases P10 and P50 is the same, i.e., `(CASK)` (see Table 7).

Stability (robustness) can be an issue because bargaining strategies can be *context-dependent*, or, more specifically, *history-dependent*. As we can see from Table 1, autonomous agents relied very much on the historical data to develop their strategies. Hence, even though some strategies are good in one or a few runs, the use of these strategies may cause history to change and result in a new environment, which in turn leads to their deterioration. However, the stability issue is not only the privilege of complex strategies, for simple strategies can evoke the same problem. In fact, in Markets 1 and 3, we observe the possibility that larger population size can either enhance stability by discovering more complex strategies (Market 1), or by intensifying the use of robust, not necessarily complex, strategies (Market 3). This is reflected in a higher frequency of using `(CASK)`, which is more robust, and a lower frequency of using `(PMaxAsk)` and `(PMin)`, which is more history-dependent.

Among the four markets, only in Market 2, does the expansion of population size not have much effect on any features which we mentioned above. In both cases of P-10 and P-50, the best strategies discovered are both `(NT)`.

[8] `P-22` refers to a class of strategies which can gain a trading profit up to or greater than 22.

3.3 Analysis of the Best Strategy Found: What Do We Learn?

3.3.1 Market 1

One of the most competitive strategies found by the autonomous agent in Market 1 is (`PMinBid`). This strategy is very aggressive, and attempts to maximize the possible profits from trading. Buyer 1 first realized that in equilibrium the market can have a trading volume of 9. His advantageous positions, determined by the reservation prices, ranked him as the fifth (his first unit) and the seventh (his second unit) trade, which means that these two are surely sellable. The question is what would be the best strategy to sell them. From "experience", Buyer 1 also realized that it would not be wise to compete with those buyers with similar advantageous positions. Therefore, he decided to wait and make others with lower ranks trade first. He then gave concessions to Buyers 2 and 3 (Figure 4, left panel). While losing the opportunities of early trades with good offers, this strategy of *procrastination* allowed him to stand in a monopsony position after the first few trades, and hence enabled him to exploit the residual sellers much more when all those high bids were gone. This is what the strategy (`PMinBid`) did for him.

While (`PMinBid`) enabled a procrastination strategy for Buyer 1 to wield a "monopsony power", it also constantly led Buyer 1 to bid a price of 14, which was the minimum bid on the previous trading day. This bidding not only made Buyer 1 successfully sell the first two tokens, but also made him able to sell the third token, which nonetheless has a reservation price lower than 14 (Figure 4, left panel). Therefore, Buyer 1 suffered a loss from the trade of the last unit. This is equivalent to selling the first three tokens in a package, something like "42 for three". Buyer 1 then used the profits gained from the first two tokens to compensate for the loss of the last token. Now the question is whether we can separate the first two from the third and even make a bigger profit. The answer is this strategy, `(Min (HT PMinBid))`, which has a description length of 3 and hence is more complex than (`PMinBid`). This strategy will guide the GP buyer to bid the minimum of `PMinBid` and `HT`. So, after trading his first two tokens, the GP buyer will bid `HT` because it is the minimum of the two. Of course, from Figure 1, this bid will not get matched, but neither will it incur an economic loss (Figure 4, right panel). With this improvement, the profit of the GP buyer increases by one accordingly.

3.3.2 Market 2

In Market 2, the best strategy discovered by the GP buyer is (`NT`). With the help of Figure 1 (the right part), this strategy can again be interpreted as a procrastination strategy. The GP Buyer gave up the privilege of an early trade. In fact, in this case, his advantageous position was ranked as number one, but he made other participants trade first. When three other tradable tokens had been finished, he held the last possible tradable token, and used this monopsony power to fully exploit the producer's surplus from the residual seller (Seller 4). After this trade, because the demand curve overlaps the supply curve, trades are still possible, but making profits is infeasible. After the population size increases to 50, the GP buyer can not discover any better strategy. As we can see from this market (Figure 1), there exists no better strategy given that all participants are truth tellers.

	Buyer's Bid				Seller's Ask				Matched Trader		
Step	Bid1	Bid2	Bid3	Bid4	Ask1	Ask2	Ask3	Ask4	Buyer	Seller	Traded Price
1	14	28	23	26	2	8	1	7	2	3	14.5
2	14	26	23	26	2	8	4	7	2	1	14
3	14	17	23	26	7	8	4	7	4	3	15
4	14	17	23	25	7	8	9	7	4	1	16
5	14	17	23	14	17	8	9	7	3	4	15
6	14	17	14	14	17	8	9	9	2	2	12.5
7	14	13	14	14	17	12	9	9	1	3	11.5
8	14	13	14	14	17	12	18	9	1	4	11.5
9	14	13	14	14	17	12	18	20	1	2	13
10	14	13	14	14	17	21	18	20			
11	14	13	14	14	17	21	18	20			
12	14	13	14	14	17	21	18	20			
13	14	13	14	14	17	21	18	20			
14	14	13	14	14	17	21	18	20			
15	14	13	14	14	17	21	18	20			
16	14	13	14	14	17	21	18	20			
17	14	13	14	14	17	21	18	20			
18	14	13	14	14	17	21	18	20			
19	14	13	14	14	17	21	18	20			
20	14	13	14	14	17	21	18	20			
21	14	13	14	14	17	21	18	20			
22	14	13	14	14	17	21	18	20			
23	14	13	14	14	17	21	18	20			
24	14	13	14	14	17	21	18	20			
25	14	13	14	14	17	21	18	20			
Profit	**21**	**30**	**8**	**20**	**21**	**5.5**	**27**	**10.5**			

	Buyer's Bid				Seller's Ask				Matched Trader		
Step	Bid1	Bid2	Bid3	Bid4	Ask1	Ask2	Ask3	Ask4	Buyer	Seller	Traded Price
1	14	28	23	26	2	8	1	7	2	3	14.5
2	14	26	23	26	2	8	4	7	2	1	14
3	14	17	23	26	7	8	4	7	4	3	15
4	14	17	23	25	7	8	9	7	4	1	16
5	14	17	23	14	17	8	9	7	3	4	15
6	14	17	14	14	17	8	9	9	2	2	12.5
7	14	13	14	14	17	12	9	9	1	3	11.5
8	14	13	14	14	17	12	18	9	1	4	11.5
9	12	13	14	14	17	12	18	20	3	2	13
10	12	13	9	14	17	21	18	20			
11	12	13	9	14	17	21	18	20			
12	12	13	9	14	17	21	18	20			
13	12	13	9	14	17	21	18	20			
14	12	13	9	14	17	21	18	20			
15	12	13	9	14	17	21	18	20			
16	12	13	9	14	17	21	18	20			
17	12	13	9	14	17	21	18	20			
18	12	13	9	14	17	21	18	20			
19	12	13	9	14	17	21	18	20			
20	12	13	9	14	17	21	18	20			
21	12	13	9	14	17	21	18	20			
22	12	13	9	14	17	21	18	20			
23	12	13	9	14	17	21	18	20			
24	12	13	9	14	17	21	18	20			
25	12	13	9	14	17	21	18	20			
Profit	**22**	**30**	**9**	**20**	**21**	**5.5**	**27**	**10.5**			

Figure 4: Trading Processes of GP Buyer in Market 1

The two trading processes above correspond to two different trading strategies: (`PMinBid`) (left panel) and `(Min (HT PMinBid))` (right panel).

3.3.3 Market 3

The best strategies found in Market 3 are `(CASK)`, `(PMaxAsk)`, and `(PMin)`. Given what we have learned from the previous two cases, it is not surprising to see that all these three strategies are also kinds of procrastination strategies, which can also be seen from Figure 2 (the left part). Since the demand curve is parallel to the supply curve, all tokens are in principle tradable. The only question is how the created surplus should be divided. Using the procrastination strategy, the GP buyer simply waited for other participants to trade first, and when all 12 other tokens had been traded, he acquired the monopsony power and completely exploited all the remaining producer's surplus. As we can see from the trading processes exhibited in Figure 5, the three strategies led to the same trading pattern, and all four tokens of the GP buyers were traded at a price of 42 (zero producers' surplus). Nevertheless, it does not mean that the three strategies are the same. The subtle difference between `(CASK)` and the other two, `(PMaxAsk)` and `(PMin)`, is that the former is history-independent, whereas the latter are not. They will work only when on the previous trading day these same kinds of strategies were played; if not, the history may not be the same, and there is no guarantee that `(PMaxAsk)` or `(PMin)` will still be 42. This is why these two strategies are not that robust (stable) as compared to `(CASK)`.

When the population size increases to 50, there is no better strategy being discovered. However, something interesting still happens. The GP buyer increases his reliance on `(CASK)` and hence enhances the stability of his monopsony profits.

3.3.4 Market 4

As opposed to the other three market scenarios, Market 4 is more intriguing. In one respect, it is very similar to Market 3 of which the demand and supply schedules never intersect. Hence, all 16 tokens in the market can in principle be sold. Under these circumstances, one may expect that the GP buyer will develop a strategy which is very similar

CASK

	Buyer's Bid				Seller's Ask				Matched Trader		
Step	Bid1	Bid2	Bid3	Bid4	Ask1	Ask2	Ask3	Ask4	Buyer	Seller	aded Price
1	-1	17473	17473	17473	33	34	34	34	2	1	8753
2	33	17470	17473	17473	34	34	34	34	3	1	8753.5
3	34	17470	17470	17473	40	34	34	34	4	2	8753.5
4	34	17470	17470	17471	40	34	34	34	4	2	8752.5
5	34	17470	17470	17465	40	39	34	34	2	3	8752
6	34	17465	17470	17465	40	39	34	34	3	3	8752
7	34	17465	17465	17465	40	39	39	34	2	4	8749.5
8	34	17465	17465	17465	40	39	39	34	2	4	8749.5
9	34		17465	17465	40	39	39	40	3	2	8752
10	39		17465	17465	40	42	39	40	3	3	8752
11	39			17465	40	42	42	40	4	1	8752.5
12	40			17465	42	42	42	40	4	4	8752.5
13	40				42	42	42	42			
14	42				42	42	42	42	1	1	42
15	42					42	42	42	1	2	42
16	42						42	42	1	3	42
17	42							42	1	4	42
18											
19											
20											
21											
22											
23											
24											
25											
Profit	69705	34869	34864	34863	26152	26151	26149	26144			

PMaxAsk

	Buyer's Bid				Seller's Ask				Matched Trader		
Step	Bid1	Bid2	Bid3	Bid4	Ask1	Ask2	Ask3	Ask4	Buyer	Seller	aded Price
1	42	17473	17473	17473	33	34	34	34	2	1	8753
2	42	17470	17473	17473	34	34	34	34	3	1	8753.5
3	42	17470	17470	17473	40	34	34	34	4	2	8753.5
4	42	17470	17470	17471	40	34	34	34	4	2	8752.5
5	42	17470	17470	17465	40	39	34	34	2	3	8752
6	42	17465	17470	17465	40	39	34	34	3	3	8752
7	42	17465	17465	17465	40	39	39	34	2	4	8749.5
8	42	17465	17465	17465	40	39	39	34	2	4	8749.5
9	42		17465	17465	40	39	39	40	3	2	8752
10	42		17465	17465	40	42	39	40	3	3	8752
11	42			17465	40	42	42	40	4	1	8752.5
12	42			17465	42	42	42	40	4	4	8752.5
13	42				42	42	42	42	1	1	42
14	42					42	42	42	1	2	42
15	42						42	42	1	3	42
16	42							42	1	4	42
17											
18											
19											
20											
21											
22											
23											
24											
25											
Profit	69705	34869	34864	34863	26152	26151	26149	26144			

PMin

	Buyer's Bid				Seller's Ask				Matched Trader		
Step	Bid1	Bid2	Bid3	Bid4	Ask1	Ask2	Ask3	Ask4	Buyer	Seller	aded Price
1	42	17473	17473	17473	33	34	34	34	2	1	8753
2	42	17470	17473	17473	34	34	34	34	3	1	8753.5
3	42	17470	17470	17473	40	34	34	34	4	2	8753.5
4	42	17470	17470	17471	40	34	34	34	4	2	8752.5
5	42	17470	17470	17465	40	39	34	34	2	3	8752
6	42	17465	17470	17465	40	39	34	34	3	3	8752
7	42	17465	17465	17465	40	39	39	34	2	4	8749.5
8	42	17465	17465	17465	40	39	39	34	2	4	8749.5
9	42		17465	17465	40	39	39	40	3	2	8752
10	42		17465	17465	40	42	39	40	3	3	8752
11	42			17465	40	42	42	40	4	1	8752.5
12	42			17465	42	42	42	40	4	4	8752.5
13	42				42	42	42	42	1	1	42
14	42					42	42	42	1	2	42
15	42						42	42	1	3	42
16	42							42	1	4	42
17											
18											
19											
20											
21											
22											
23											
24											
25											
Profit	69705	34869	34864	34863	26152	26151	26149	26144			

Figure 5: Trading Processes of GP Buyer in Market 3

The three trading processes above correspond to three different trading strategies: `(CASK)` (upper panel) `(PMaxAsk)` (middle panel), and `(PMin)` (lower panel).

to what the GP buyer used in Market 3. This is indeed the case. The most frequently seen strategy used by the GP buyer in this market is still `(CASK)`. Obviously, Buyer 1 already learned that all tokens are tradable, and he wanted to be patient to wait for the last few rounds so that he could completely exploit all of the remaining producers' surplus. However, Market 4 is different from Market 3. In Market 3, procrastination does not cause good trading opportunities to be missed since the supply curve is essentially flat; nevertheless, in Market 4, the supply curve is piecewisely step-up. The later the GP buyer gets into the market, the less likely he is to receive favorable offers. In other words, the cost of delayed trading becomes more significant in Market 4, and the pure procrastination strategy simply neglects this cost. Therefore, one may wonder whether the GP buyer can learn something even more intelligent, i.e., a strategy which can balance the gains from full monopsony against the loss due to missing favorable offers. The answer is *yes*.

In addition to `(CASK)`, our GP buyer also learned the following strategies:

- ((* NT (% PAvgBid PMaxBid)) (profit=15997)

- ((Min (If_Bigger_Then_Else PMinBid PAvgBid CASK PAvgBid) (If_Bigger_Then_Else HT PAvgBid PAvgBid CASK)) (profit = 16512)
- ((Min (If_Bigger_Then_Else PAvgBid HT CASK HT) PAvgBid) (profi= 16512)

None of these strategies will advise the GP buyer to wait until the very end of trading, but to use a more aggressive strategy (higher bidding) to compete with other opponents (other buyers) and to obtain the favorable offers from suppliers.

4 The Theory of Optimal Procrastination

The theory of *optimal procrastination* means that the agent attempts to delay his participation in the market transaction so as to avoid early competition and become a monopsonist in the later stage. Once getting there, he will then fully exercise the monopsony power by bidding with *third-degree price discrimination*. However, procrastination may also cause the agent to miss some good offers; therefore, there is an opportunity cost for procrastination and the agent will try to optimize the procrastination time by balancing his monopsony profits against these costs. Economic theory requires economists to have a good understanding of the structure of the problem and then to find a good solution to it. Sometimes, both of these tasks are demanding and, in this case, we simply dispatch autonomous agents to the "complex world" and see what they find and get inspiration from there. The theory of optimal procrastination presented here is a perfect illustration of what we mean by *autonomous-agents-inspired economic theory*.

More formally, this theory can be stated as follows. Without losing generality, let us use the simulated buyer (the autonomous agent) in this paper as an example. It is assumed that, at time t, the buyer can be a current holder, i.e., he can offer the highest bid, which is also greater than the current ask. Then the deal is made, and the transaction price P_t, by the Aurora rule, is the average of the two.

$$P_t = \frac{bid_t + ask_t}{2}, \;\; if \;\; bid_t \geq ask_t. \tag{1}$$

Now, in order to gain better terms and conditions, the buyer chooses to trade at a delayed time, say $t + \Delta t$. With this delay, we assume that he can reduce his bid by the amount $\Delta bid_{t+\Delta t}$ ($\Delta bid_{t+\Delta t} > 0$); nevertheless, with this delay, the more favorable offers have gone, and the alternative ask may require an additional amount, say $\Delta ask_{t+\Delta t}$ ($\Delta ask_{t+\Delta t} > 0$). Hence, the consequence of this delay is to pay a price, $P_{t+\Delta t}$.

$$P_{t+\Delta t} = \frac{(bid_t - \Delta bid_{t+\Delta t}) + (ask_t + \Delta ask_{t+\Delta t})}{2}, \;\; if \;\; bid_t - \Delta bid_{t+\Delta t} \geq ask_t + \Delta ask_{t+\Delta t}. \tag{2}$$

The gain, G, with this delay is, therefore, the difference between the two prices paid at different times, and that depends on the difference between $\Delta bid_{t+\Delta t}$ and $\Delta ask_{t+\Delta t}$,

$$G_{t,\Delta_t} = \mid \Delta bid_{t+\Delta t} - \Delta ask_{t+\Delta t} \mid, \tag{3}$$

which is further determined by the shape of the demand schedule and the supply schedule. When the demand schedule and supply schedule are flat and horizontal to each other, such as in Market III, $\Delta ask_{t+\Delta t}$ is zero, so it pays the buyer to wait until he is the sole buyer in the market. However, when the demand and supply schedules are step functions, things can get complicated, such as in Market I and Market IV, but our GP buyer can still figure out a good time to get into the market.

5 Concluding Remarks

In this paper, through a large computer simulation, we simulate the evolution of the bargaining strategies of autonomous agents (buyers) in a competitive environment. These autonomous agents, by design, are purported to search for better deals from which to gain. In the very foundation of economics, we do need these agents to discover, exploit and eventually destroy any hidden patterns and opportunities. The purpose of our simulation is then to have a clear picture of what the autonomous agents discover and what these discoveries mean, and from that to see whether we can also learn and construct a theory in this light. *The theory optimal of procrastination* found in this paper demonstrates what we can glean from the behavior of our autonomous agents. At the end of the paper, we would like to point out several directions for further study.

First, the choice of a highly static environment is not necessary, but it does make it easier for us to comprehend and make sense of the behavior of these chance-discovery agents. This learning can then help us to have a better idea of what these agents are doing or attempting to do when they are placed in a much more complex situation, such as the one defined in Chen and Tai (2010), where autonomous agents are placed in surroundings filled with SFI-style programmed agents. It may not surprise us to see that these autonomous agents eventually beat all these programmed agents, but it is difficult to perform an in-depth analysis of the discovered bargaining strategies given these complex surroundings. Maybe a challenging task for the future would be to introduce novel data mining or text mining techniques to this large database so as to know more of the "mental process" of these autonomous agents.

Second, this paper and Chen and Tai (2010) only consider a single autonomous agent. It would thus be interesting and challenging to see whether we can develop *a theory of multi-agent competition*. The immediate next step is to expand the current single-agent version into a two-agent version, so that in the latter we can have a co-evolutionary game-theoretic situation, and the monopsony result observed in this paper can become that of a duopsony, as would be expected.

Third, it is always interesting to know whether human agents can also learn the intelligent trading strategies discovered by the autonomous agents, for example, the optimal procrastination strategy. We are now designing market experiments to see whether the trading patterns realized by our autonomous agents can also be replicated by human agents.

Acknowledgement

The NSC grants NSC 98-2911-I-004-007 and NSC 98-2410-H-004-045-MY3 are gratefully acknowledged.

References

Andrews M, Prage R (1994) Genetic programming for the acquisition of double auction market strategies. In: Kinnear K Jr (ed), Advances in Genetic Programming. MIT Press, Cambridge, MA, 355–368.

Chen S-H (2008a) Computational intelligence in agent-based computational economics. In: Fulcher J, Jain L (eds.), Computational Intelligence: A Compendium, Springer. 517–594.

Chen S-H (2008b) Genetic programming and agent-based computational economics: From autonomous agents to product innovation. In Terano T, Kita H, Takahashi S, Deguchi H (eds), Agent-Based Approaches in Economic and Social Complex Systems, 6: 3-14, Springer.

Chen S-H, Tai C-C (2010) The agent-based double auction markets: 15 Years On. In: Takadama K, Cioffi-Revilla C, and Deffuant G (eds), Simulating Interacting Agents and Social Phenomena: The Second World Congress, Springer.

Rust J, Miller J, Palmer R (1993) Behavior of trading automata in a computerized double auction market, in Friedman D, Rust J (eds) Double Auction Markets: Theory, Institutions, and Laboratory Evidence, Redwood City, CA: Addison Wesley.

Rust J, Miller J, Palmer R (1994) Characterizing effective trading strategies: Insights from a computerized double auction tournament. Journal of Economic Dynamics and Control 18: 61–96.

Tesfatsion L, Judd K (2006) Handbook of Computational Economics, Volume 2: Agent-Based Computational Economics. North Holland.

Thaler R (2000) From Homo economicus to Homo sapiens. Journal of Economic Perspectives 14(1): 133–141.