Optimizing Aggressiveness in Stock Trading Simulations

Zheyuan Su¹⊠ and Mirsad Hadzikadic²⊠

Complex Systems Institute, College of Computing and Informatics University of North Carolina at Charlotte, Charlotte, NC, 28223 ¹ zsu2@uncc.edu, ²mirsad@uncc.edu

Abstract. Learning plays a key role in stock investment. It enables people to mimic the best performers' strategies in the real-world market. Aggressiveness (or eagerness) in learning determines the degree of learning activity, or the degree to which one trader decides to mimic another during every opportunity to learn. In our agent-based model we built a stock-trading model that issues a daily stock trading signal. This paper introduces an agent-based model for finding the optimal aggressiveness in learning, while optimizing the stock trading returns. The system has been evaluated in the context of Bank of America stock performance in the period of 1987–2014. The model significantly outperformed both S&P 500 and buy-and-hold strategy for the Bank of America stock.

1 Introduction

Learning is an inseparable part of our daily lives. It is an act of acquiring new or modifying existing knowledge, behaviors, and skills. It's a skill that is key to achieving one's full potential. Learning builds upon previous knowledge and does not happen all at once. It is a lifelong process. Learning enables person's ability to adapt to the changes in the environment.

Similarly, learning plays an important role in investment. A simple way of learning in investment is to mimic behaviors of top market performers, copying their philosophy of stock trading. By doing so investors become aware of the latest market changes and possibly benefit from the current state of the market (Linn, 2007). The progress of learning over time is captured in what is frequently referred to as a learning curve, which is a graphical representation of the increase of learning with experience. The slope of learning reflects how aggressive (eager, motivated, or capable) the learner is in attempting to become a better performer (investor in this case) over time. A more aggressive learner tends to copy more from the market best performers, while a less aggressive ones tend to only use best performers' behaviors as a minor correction in updating its own trading rules.

Many of the real world activities have been simulated by computerized analytical techniques (Teixeira and DeOliveira, 2010), although with limited success. Recently, complex adaptive systems-inspired methods, primarily using agent-based modeling techniques, have been tried as a way to simulate traders' behaviors and capture the

intricacies of stock trading (Kodia et al., 2010). This paper introduces an agent-based model for finding the optimal value of aggressiveness in learning when maximizing stock trading returns. The model is a derivative of our multi-sectors trading model (Su and Hadzikadic, 2014). The system has been evaluated in the context of Bank of America stock performance in the period of 1987–2014. The model significantly outperformed the buy-and-hold strategy on both S&P 500 and Bank of America stock.

2 Background

Investment strategies are usually classified as either passive or active portfolio management strategies (Barnes, 2003). Passive portfolio management only involves limited buying and selling. Passive investors typically purchase and hold investments, anticipating long-term capital appreciation and limited portfolio maintenance. Consequently, only active portfolio management strategies can bring investors extra profits simply because they bring about the possibility of covering a wide range and frequency of stock price movements. An active equity portfolio management (Grinold and Kahn, 1995) requires periodic forecasts of economic conditions and portfolio rebalancing based on forecasted conditions. A degree of trust in markets by investors impacts the stock trading activities and risk control strategies (Asgharian et al., 2014). Risk control methods make it possible for a dynamic portfolio management strategy to outperform the market (Browne, 2000). A simple momentum-and-relative-strength strategy outperformed the buy-and-hold strategy 70% since the 1920s (Faber, 2010). Performance can also be improved by considering a simple trend before taking positions.

However, these methods simply provide a retroactive, aggregate simulation technique. Since the market consists of trading individuals, taking into consideration interactions among agents may provides investors with an important insight, potentially leading to a significant improvement in portfolio performance. The Complex Adaptive Systems (CAS) methodology offers a natural framework for augmenting portfolio management strategies with simulation of individual agent interactions in the market place and their surroundings. A further advantage of CAS stems from its ability to evaluate many different rules and parameters for agent interactions, as well as test different trust thresholds among agents, thus enabling the system designer and end users to uncover agent interactions that actually improve portfolio performance.

3 Complex Adaptive System in Investment Management

Since Complex Adaptive System simulations have the capability to capture the essence of distribution, self-organization, and nonlinear social and natural phenomena, characterized by feedback loops and emergent properties, they offer a brand new way of modeling inherently complex systems such as a stock market. Interaction patterns and global regularities are important features in the financial markets (Cappiello et al., 2006). It is possible to utilize agent-based modeling (ABM) techniques to model

financial markets as a dynamic system of agents. There already have been successful implementations of ABM models in many theaters of human endeavor, including economics, government, military, sociology, healthcare, architecture, city planning, policy, and biology (Johnson et al., 2013, Hadzikadic et al., 2010). In financial market simulations a large number of agents engage repeatedly in local interactions, giving rise to global markets (Roberto et al., 2001, Bonabeau, 2002). This dynamic can be readily captured by a well designed CAS-enabled ABM simulation.

In this paper we describe an ABM system that seeks to optimize learning aggressiveness at the level of an individual agent, based on the single stock-trading model presented by Su and Hadzikadic (Su and Hadzikadic, 2015). This model issues a stock-trading signal (buy, sell, or hold) for a stock (Bank of America in our example) on a daily basis. Agents will trade the Bank of America stock based on the publicly available, adjusted, daily data from January 2, 1987 to December 31, 2014. In addition, agents have the knowledge of the current status of the stock market, either bull or bear, based on the recession data available from the National Bureau of Economic Research (NBER). Here, a bull market indicates a financial market of a group of securities in which prices are rising or expected to rise. Bear market denotes the opposite scenario in financial market terms. Agents use this information to select their trading rules. As the market has been in the bull market in the post-1987 period, the simulation for a single stock will only track the bull market trading decision rules.

3.1 A CAS Stock Trading Model

In our CAS Agent-Based Model we built a stock-trading model that issues a daily stock trading signal at the end of each trading day. The current closing price of Bank of America is used by agents to determine if and when they want to purchase or sell the Bank of America stock. The closing price is adjusted to eliminate the effect of stock dividends. The interest will be distributed based on the agent's cash on hand on a daily basis in the same timeframe. The interest rates were retrieved from the Federal Reserve Web site. In addition, the transaction cost is set to be \$10 for each transaction, thus letting the agents evaluate the trade-off between trading and holding stock positions. Also, the transaction cost prevents agents from making profit in slight price changes, which in turn may boost market fluctuation in the real world market.

Agents

A collection of agents constitutes the "trading world" in this ABM simulation. In order to reduce the complexity of the problem and minimize the size of the exploration space of the model as much as possible, we decided to investigate the agents on individual level only. Although one of the major components of the financial markets is a group known as institutional investors, our intent was to find the optimal degree of aggressiveness in learning among individual agents only, in order to learn how to maximize trading profits given the initially randomized proposed trading strategies. Agents are initialized with a certain amount of money. Their transactions are triggered by their trading decision rules, their disposable capital, and the momentum of the market (the momentum adjusts agents' decision rules temporarily). Given this knowledge and the current market status, agents choose the trading rules for the current tick. Table 1 describes the trading rules assigned to individual agents.

	Table 1. Trading rules assigned to individual agents
Buy-Threshold	Minimum price change required for taking a long position
Buy-Period	Time window agents observe before evaluating the Buy-
	Threshold
Sell-Threshold	Minimum price change required for taking a short position
Sell-period	Time window agents observe before evaluating the Sell-
	Threshold
Aggressiveness	Degree in learning while copying the trading styles from top
	performers
Self-Confidence	Degree of trust in the trading rules born with agents

Trading Rules

The following formulas describe agents' basic separated decision rules in detail.

Buy Rule:
- Price-Change > Buy-threshold $* (1 - (1 - \text{self-confidence}) * momentum of$
<i>buying</i>) in past Z days
 Agents will take long positions

For example, if the values for *buy-threshold*, *buy-period*, *self-confidence* and *market momentum* for an agent are 0.3, 20, 0.8, and 0.6 respectively, then the buying rule for this agent is:

IF the stock price goes up 0.3 * (1 - (1 - 0.8) * 0.6) * 100% = 26.4% in the past 20 trading days,

THEN agents will take a long position.

0 1	1	D	1		
N 0		· 12 ·	11	$\boldsymbol{\rho}$	٠
001	ш.	1/	սյ	ιu	

Price-Change < Sell-threshold * (1 – (1 – self-confidence) * momentum of selling) in past Z days

- Agents will take short positions

Similarly, if the values for *sell-threshold*, *sell-period*, self-confidence and market momentum are 0.1, 30, 0.3 and 0.7 respectively, then the selling rule for this agent is:

IF the stock price goes up less than 0.1 * (1 - (1 - 0.3) * 0.7) * 100% = 5.1% in the last 30 trading days,

THEN agents will take a short position.

Also, agents are permitted to short sell the stock at any time. This decision makes it possible for agents to capture profits in market downturns as well. An agent can sell short any amount of stock up to their available cash amount. In financial terms, a long position indicates the purchase of a security such as stocks, commodity or currency, with the expectation that value of the assets will rise in value over time. A short position denotes the opposite scenario. Short selling indicates the sale of a security that is not currently owned by the seller. However, in order to close the position in the future, agents have to buy the same amount of securities to cover their short selling positions placed earlier.

Market Momentum

Market momentum can be summarized as:

		•	FO	17
Momentum	ranges	1n	10	11

- Count how many people intend to buy/sell
- If no one is *buying/selling*, *momentum of buying/selling* will be 0
- If everyone is buying/selling, momentum of buying/selling will be 1

Rationality is not always the real basis for each agent transaction. Panics and frenzies are also driving the trend of the markets. Investors' greedy and panic-prone behaviors amplify regular ups and downs in the market, building up the bubble that inevitably ends in a market crash. Some of the most famous examples are the tulip craze, the south sea bubble, and the great depression. Also, recent market trends in Shanghai Composite Index, shown in Fig. 1, also validate this phenomenon.



Fig.1. Shanghai Composite Index (Source: Yahoo Finance)

Clearly, market momentum is an important factor that impacts agents' decisionmaking rules of the agents. Otherwise, there would be no bubbles. Heterogeneous agent models (Hommes, 2006) show that most of the behavioral models with bounded rational agents using different strategies may not be perfect, but they perform reasonably well. With the adaptive trading strategies, agents are able to seek more trading opportunities and boost their profits.

The inclusion of market momentum in the stock-trading model will potentially increase the return of investors, as it allows agents to adjust their trading rules temporarily according to the latest market changes (and anticipation of other traders' behavior). In the stock trading model, momentum was generated by the overall buy/sell behavior of agents, thus creating a bid-ask spread for the stock they are trading. Bid-ask spreads shows the amount by which the ask price exceeds the bid. In this way, the more agents buy stocks, the higher the bidding price. Likewise, the more agents sell stocks, the lower the stock prices, as agents are trying to liquidate their inventories.

Degree of Trust

Degree of Trust in our model is denoted as the variable called self-confidence, which is created to control the degree of trust in market momentum. The higher the selfconfidence, the stronger the agents believe in their own trading strategies, thus decreasing the impact of the trading environment around them.

Latest market information is available to all agents. They will conduct transaction based on the latest stock price, which is the price in the current tick. Agents can track the historical prices in any time period up to Jan. 2, 1987, which is the starting point of the simulation. Market momentum is also used to simulate the bandwagon effect in economics, which is characterized by the possibility of the individual adoption increasing with respect to agents who have already done so. This plays an important role in transactions. As a result, if there are a lot of agents who are buying stocks, then agents will increase their buy-threshold. On the other hand, if there are many agents who are shorting stocks, then a substantial number of agents will correspondingly decrease their sell-threshold, as they try to liquidate their assets as soon as possible. In some extreme scenarios, agents with the self-confidence of 1 will not listen to the latest market trends, while agents with the self-confidence of 0 will follow market trend makers completely.

However, only few agents will be in the abovementioned extreme scenarios. Therefore, self-confidence stands out to define the degree of trust in the market momentum by individual agents. The degree of trust affects the participation rate in stock trading (Guiso, 2008). The participation rate in stock trading impacts the bid-ask spread in return.

Self-confidence captures the agents' degree of trust among all other agents. Selfconfidence is randomly assigned at the initial stage, making it possible to explore the best degree of confidence one should adopt in order to outperform the market. Also, the simulation tracks the best performers' self-confidence.

Genetic Algorithm

The concept of "Survival of the fittest" proposed by Darwin is also applicable to the world of stock transactions. In artificial intelligence, genetic algorithms embody a search heuristic that mimics the process of natural selection. This heuristic mainly generates promising descendants that are more adaptive to the changes in the environment. A hatch-and-die concept in NetLogo was introduced in the model as a mechanism for likely regenerating the best performers and eliminating underperforming agents. If agents bankrupt in the simulation, newly initialized agents will replace them. This was introduced in order to keep constant the quantity of agents, thus ensuring an active trading environment. On the other hand, ruling out the bankrupt agents helps maintaining a faster simulation speed.

Search Space and Mutation

As agents have a lot of parameters, the search space covering all possibilities counts trillions of states. A mutation mechanism is introduced to use a small quantity of agents to simulate agents with all possible combination of parameters. With mutation, agents' parameters are initialized in a small range. However, some agents' parameters are generated in a much larger range. Mutation also takes place in hatch-and-die

activity of a NetLogo simulation. With the benefit of mutation, some agents' parameters are randomized far beyond the preset small ranges, thus making it possible to explore the whole parameter search space. At the same time, Monte Carlo simulation was applied in the model to achieve the best result with the minimal number of simulation rounds.

Learning and Interaction

In ABM implementations, agents have the ability to learn from each other. The process of learning offers agents the opportunity to refine their transaction decision rules, thus helping them to secure more profits in a complex market (Cui, 2012). Agent learn from the best performers within a certain radius in the NetLogo simulation. The neighborhood structure is introduced to enhance learning efficiency. Learning squeezes the search space from the size of trillions into a much smaller one.

Learning mechanism makes it possible to investigate alternative strategies that have not yet been discovered in the market (Outkin, 2012). In this implementation, to preserve computational time, there is a radius around agents. Agents can only see the other agents in the radius while they are moving around. The introduction of radius makes the learning more sustainable.

The variable *aggressiveness* indicates to what extent the agents want to adopt their neighbor's behavioral structure. Also, there is a period of time during which the learning process is prohibited in order to allow agents to evaluate their current trading strategies. Contrary to the market momentum-induced changes, according to which agents adjust their trading decision rules temporarily, the changes made through learning are permanent.

Benchmark Agents

Two benchmark agents are created to evaluate the performance overtime. Benchmark agents will use buy-and-hold strategy on the underlying assets, which are Bank of America and S&P 500 respectively.

Global Trading Environment

In the CAS stock-trading model, the world is two-dimensional. X-axis and Y-axis, ranging from -10 to +10, define the size of the world. There is a variable called *radius* defining how far agents can see each other and reach out to them for learning. Each agent will have the same value of radius, ensuring they have equal opportunity to reach out to the best performers within the radius range. Agents have the knowledge of where they are and who else is within their radius.

3.2 Implementation

This stock trading CAS model was implemented using the Netlogo 5.1.0 programmable modeling environment (Wilensky 2009). Netlogo offers a user-defined simulation grid and the possibility of defining agents with different properties.

In this model, the exploration space for all possible trading strategy combinations is in the size of trillions. As the combination is extreme large, a huge quantity of agents needs to be created to cover all possibilities. This is theoretically doable, but it has a huge impact on the computing speed of the simulation. It may take decades to get the result if trillions of agents are used in the model. In order to provide a trade-off between the computing speed and the space exploration, the number of agents in the model is capped at 1,000, while taking advantage of the mutation mechanism to explore the possibility of exploring the whole space of parameter dimesions. The settings of parameters are listed in the Table 2. All transaction decision rules are randomized within the [-0.4,0.4] range for required returns, and within [0,100] range for the trading periods. Aggressiveness is randomized between 0.1 and 0.0001, with step size 0.0001. The rationale behind this setting range is that there are 7,110 trading days in the whole simulation. This range will prevent all agents from learning too fast, and ensuring that all agents have identical trading rules. Self-confidence is randomized between 0 and 1, with the step of 0.01. The small range and large step sizes are used in the simulation as a mechanism for decreasing the size of the search space, thus boosting the coverage of each run. However, the full search range will be closed to the real world trading. Mutation is used to allow a subset of agents to mutate from [-0.4,0.4] to [-1,1] for required returns, and from [1,100] to [1,1000] for trading periods. The mutation rate is fixed at 0.1, which allows 10% of all agents to get buy/sell threshold and buy/sell period generated in [-1,1] and [1,1000] respectively. Agents are assigned the initial capital in the amount of \$50,000. The transaction cost is fixed at \$10 per transaction. Also, interest is distributed at the end of each tick, based on the amount cash held on hand and the current tick interest rate.

Table 2. Parameter Ranges

Tuble 2.1 arameter Ranges				
Non-Mutation	Buy-Threshold	[-0.4, 0.4] with step size 0.2		
	Buy-Period	[0, 100] with step size 20		
	Sell-Threshold	[-0.4, 0.4] with step size 0.2		
	Sell-period	[0, 100] with step size 20		
Mutation	Buy-Threshold	[-1, 1] with step size 0.2		
	Buy-Period	[0, 1000] with step size 20		
	Sell-Threshold	[-1, 1] with step size 0.2		
	Sell-period	[0, 1000] with step size 20		
Mutation Rate	•	0.1		
Self-Confidence	e	[0,1] with step size 0.01		
Aggressiveness		[0.1, 0.0001] with step size 0.0001		
Initial Capital		\$50,000		
Transaction Cos	st	\$10		

Learning from other agents is disabled in the first 1,000 days, which leaves sufficient time for the agents to evaluate their initial trading strategies. After that, agents learn until the end of the simulation. This mechanism allows us to test the durability and profitability of agents' initial trading rules throughout the volatile market movements.

4 Results

In the stock trading model, S&P 500 and Bank of America (BAC) buy-and-hold strategies were used as the performance benchmarks. The S&P 500 index increased from 246.45 to 2059.9 during the period of 01/02/87 to 12/31/13. The Bank of America stock price increased from 2.34 to 17.89 in the same period. If investors used the buy-and-hold strategy, they would earn 735.82% and 664.53% of their initial investment respectively. Figure 2 shows the cumulative return for different agents in this simulation. The best performer in the simulation achieved 1,363.36% profit, which is 1.85 times better than the SP500 benchmark. Also, the top 10% performers achieved 697.94% profit, which is slightly better than the BAC benchmark. The main reason for this is that the S&P500 is a weighted index including stocks from all industries. The main cause of the last financial crisis is related to banking industries. Therefore, the banking industry suffered more downturn than the S&P 500. This resulted in BAC stock price being still 60% below its historical high, while SP500 was surpassing its historical highs again and again in the past few years. Table 3 shows the summary of the results.



From Monte Carlo simulations, two typical experiments results are selected for demonstration of model performance with different final aggressiveness results. Experiment 1 represents the one of our best results while the experiment 2 has below average performance. It's obvious that the best level of learning aggressiveness was at

the level of 0.07. It indicates that the best agent tended to accept 7% of the difference between its and the best performer's (within the radius) decision making parameters. After that, agents move around and seek others that are better off still. As the aggressiveness was randomized between 0.1 and 0.0001, 0.07 indicates a significantly fast learning speed. Comparing the results between these two experiments, higher aggressiveness tends to bring higher results. The preference for high speed of learning indicates that agents tend to mimic the best performer's trading behavior and learn as quickly as possible during the whole simulation process, without actually committing to early to mimic the first few more successful agents that they actually see. Because of the permanent effect of learning, the best agents are able to mimic all the best rules within a shorter period of time with faster learning speed.

The best decision rule set is described as follows.

- If the stock price goes down 27% in last 33 trading days, take a long position.
- If the stock price goes up less than 8% in last 9 trading days, take a short position.

However, these rules are subject to change with respect to the degree trust in the market momentum. As the market momentum changes everyday, agent change their decision rules correspondingly. The following shows that a particular trading rules set was actually put into practices on Dec. 31 2014.

- If the stock price goes down 14.18% in last 234 trading days, take a long position.
- If the stock price goes up less than 4.5% in last 42 trading days, take a short position.

The narrative for the trading rule set basically states that an investor should track the past performance for a particular stock and then decide whether to get in. For the exit strategy, if the stock's performance deviates from the expectations, investors should clear their positions immediately.

5 Discussions, Conclusion, and Future Work

Computer simulations allow us to see behind-the-scene actions of agents, and to make forecasts on the possible futures of the markets. Compared with the return of the S&P 500, the CAS model with optimized learning aggressiveness presented here has achieved a much higher return, around 1.8 times, than the S&P in the same timeframe.

Best agent and agents with the top 10% performance show a very aggressive learning style, with 0.07 and 0.05 derived values respectively. With a fast learning speed, agents can use the trial-and-error method to get trading rules that works best in the current market environment. However, one problem we face is the inability to run the simulation with a large amount of agents. If we could put more agents into the simulation, then we would be able to supply more randomly initialized decision rules into the pool. With a larger pool, agents could mimic the best set of rules immediately instead of waiting for a long time to get the best rules available to them.

We are currently working on several other strategies to improve performance of the model. One is to change the distribution of aggressiveness in the initialization stage. Currently, aggressiveness is uniformly distributed. We are attempting to create derivative versions utilizing normal distribution and power law distribution. Another one is to extract the real time tweets from Tweeter and to run a sentiment analysis on those tweets. Then the signals from Twitter will be attached to the current momentum component. We are also planning to extend the trading of a single stock to multiple stocks, which are selected from different sectors. In this way, the learning will be more effective as it makes agents more adaptive, while conducting transaction over multiple stocks from different industries.

References

Su, Z., Hadzikadic, M. (2014), 5th World Congress on Social Simulation, Application of Complex Adaptive Systems in Portfolio Management, WCSS 2014 Proceedings,

Su, Z., Hadzikadic, M. (2015), Simultech, An Agent-based System for Issuing Stock Trading Signals, Simultech 2015 Proceedings,

Holland, J., (1992). Daedalus. Vol.121. No.1. A New Era in Computation, pp. 17-30.

Johnson, L., Hadzikadic, M., Whitmeyer, J., (2013). International Journal of Humanities and Social Science. *The Future Engaging Complexity and Policy: Afghanistan Citizen Allegiance Model*. Vol.3. No.10.

Wilensky U: NetLogo. (1999), http://ccl.northwestern.edu/netlogo. Center for Connected Learning and Computer-Based Modeling. Northwestern University, Evanston, IL.

Fishman, G. (1995). Springer. Monte Carlo: Concepts, Algorithms, and Applications.

Faber, M., (2010). Cambria. Relative Strength Strategies for Investing.

Barnes, J. (2003). Active vs. Passive Investing. CFA Magazine. Jan 2003: 28-30.

Grinold, R., & Kahn, R. (1995). Active portfolio management (pp. 367-8). Probus Publ.

Cappiello, L., Engle, R. F., & Sheppard, K. (2006). Asymmetric dynamics in the correlations of global equity and bond returns. Journal of Financial econometrics, 4(4), 537-572.

Raberto, M., Cincotti, S., Focardi, S. M., & Marchesi, M. (2001). Agent-based simulation of a financial market. Physica A: Statistical Mechanics and its Applications, 299(1), 319-327.

Bonabeau, E. (2002). ABM: Methods and techniques for simulating human systems. *Proc of the National Academy of Sciences of the United States of America*, 99 (Suppl 3), 7280-7287.

Hadzikadic, M., Carmichael, T., & Curtin, C. (2010). *Complex adaptive systems and game theory: An unlikely union*. Complexity, 16(1), 34-42.

Farmer, J. D., Patelli, P., & Zovko, I. I. (2005). *The predictive power of zero intelligence in financial markets*. Proceedings of the National Academy of Sciences of the United States of America, 102(6), 2254-2259.

Hommes, C. H. (2006). *Heterogeneous agent models in economics and finance*. Handbook of computational economics, 2, 1109-1186.

Outkin, A. V. (2012, Sep). An agent-based model of the NASDAQ stock market: Historic validation and future directions. In CSSSA Annual Conference, Santa Fe, New Mexico (pp. 18-21).

Cui, B., Wang, H., Ye, K., &Yan, J. (2012). *Intelligent agent-assisted adaptive order simulation system in the artificial stock market*. Expert Systems with Applications, 39(10), 8890-8898.

Linn, S.C., & Tay, N.S.(2007). Complexity and the character of stock returns: Empirical evidence and a model of asset prices based on complex investor learning. Management Science, 53(7), 1165-1180.

Darley, V., & Outkin, A. V. (2007). NASDAQ market simulation: insights on a major market from the science of complex adaptive systems. World Scientific Publishing Co., Inc.

Browne, S. (2000). *Risk-constrained dynamic active portfolio management*. Management Science, 46(9), 1188-1199.

Asgharian, H., Liu, L., & Lundtofte, F. (2014). *Institutional Quality, Trust and Stock-Market Participation: Learning to Forget*. Trust and Stock-Market Participation: Learning to Forget.

Guiso, L. (2008). Trusting the stock market. Journal of Finance, 63(6), 2557-2600.

Teixeira, L. A., & De Oliveira, A. L. I. (2010). A method for automatic stock trading combining technical analysis and nearest neighbor classification. Expert systems with applications, 37(10), 6885-6890.

Kodia, Z., Said, L. B., & Ghedira, K. (2010). *A study of stock market trading behavior and social interactions through a multi agent based simulation*. In Agent and Multi-Agent Systems: Technologies and Applications (pp. 302-311). Springer Berlin Heidelberg.