SIMULATING AN ARTIFICIAL STOCK MARKET WITH GENETIC PROGRAMMING INDUCED TRADING STRATEGIES

Diana MARICA, PhD*

Abstract

The artificial stock market which is the subject of the hereto paper is simulated in Altreva Adaptive Modeler multi-agent simulation software application. The simulation model incorporates 2,000 heterogeneous agents which trade amongst each other on an artificial stock market which uses a call auction trading mechanism. Within the evolutionary agent-based model, the population of agents is continuously adapting and evolving by using genetic programming in order to obtain new agents with better trading strategies generated from combining the trading strategies of the best performing agents and thus replacing the agents which have the worst performing trading strategies.

Keywords: agent-based modeling, heterogeneous agents, technical trading.

JEL Classification: C63, G17

1. Introduction

The aim of the hereto research is to describe an adaptive agent-based model of the stock market, incorporating 2,000 heterogeneous agents which trade within an artificial stock market. In order to achieve our research aim, we use the Adaptive Modeler [1] software to simulate an adaptive agent-based model for artificial stock market generation. Thus, heterogeneous agents trade a stock floated on the stock exchange market, placing orders depending on their budget constraints and trading rules, where the artificial market is simulated as a call auction market.

The population of agents is constantly adapting to the new market conditions by using evolutionary computing, namely the Strongly Typed Genetic Programming [2]. This breeding process generates new agents with better trading strategies obtained from

* *Postdoctoral researcher, Romanian Academy*

recombining the trading strategies of the best performing agents and thus replacing the agents which have the worst performing trading strategies. This technique allows for adaptive, evolving and self-learning market modelling and forecasting solutions.

To explore and understand the complexity of the financial markets and trading behavior, models using agent-based modeling techniques have been successfully implemented, offering explanation for observed stylized facts and being able to reproduce many of them [3], [4]. Arthur et al. [5] from Santa Fe Institute, Ca., USA, developed an artificial stock market which allowed for testing of agent-based models with heterogeneous agents. Rust [6] and Phelps et al. [7] have used in their experiments heterogeneous agents which change their strategies during the learning process, as follows: the unprofitable strategies are being replaced with the more profitable ones, thus developing adaptive models which use genetic algorithms to evolve. Walia [8] has studied the development of the agent-based models which use genetic programming, allowing for more flexibility and effectiveness in finding optimal solutions, programs being encoded as tree structures, thus crossover and mutation operators being applied easier. The later is similar with the learning process used in the hereto paper.

An agent-based model represents a computational model for simulating the actions and interactions among agents in a multi-agent system in order to analyze the effects on a complex system as a whole, being a powerful tool in the understanding of markets and trading behavior. An agent-based model of a stock market consists of a population of agents (representing investors) and a price discovery and clearing mechanism (representing an artificial stock market). The complex dynamics of these heterogeneous investors and the resulting price formation process require a simulation model of a multi-agent system and an artificial market.

The evolutionary agent-based model referred to in this paper is simulated in Adaptive Modeler software, which supports up to 2,000 agents and 20,000 simulation periods for each epoch of simulations. The agents are autonomous and heterogeneous entities representing the traders of the stock market, each having their own *wealth* and their own trading strategy called the *genome*.

The agent-based model is simulated under different computational techniques compared to other scientific papers in this field, due to the fact that in Strongly Typed Genetic Programming (STGP) the process of estimating the agents' fitness function does not include re-execution of the trading rules based on historical data, therefore there is no over-fitting. This is possible due to the fact that the model evolves in a time-incremental way, and it does not optimize on historical data, thus avoiding over-fitting of the data which seems to represent one of the biggest forecasting pitfalls. This model uses a high number of artificial agents, namely 2,000 agents and 2,000 continuously adapting trading rules, which increases model stability and reduces sensitivity to random factors. Trading signals given by the model are based on the interaction of all artificial agents, and not just on a single trading rule. The agent-based model is dynamic, constantly evolving and adapting to market conditions.

The remainder of this paper is structured as follows. Section 2 presents the artificial stock market trading mechanism, Section 3 describes the cycle of the evolutionary agent-based model. Section 4 the specifications of the adaptive agent-based model used in the simulations, the paper ending with the conclusions and directions for future work.

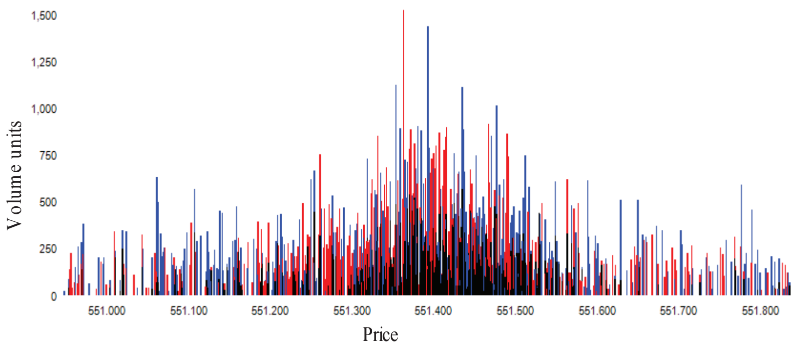
2. Artificial stock market trading mechanism

The artificial stock market trading mechanism is based on a call auction which represents an order driven facility which batches multiple limit orders together for simultaneous execution in a multilateral trade, at a single clearing price, at a predetermined point in time.

A limit order is a price-quantity pair which expresses an offer to buy or sell a specific quantity at a specific price, while a market order specifies a quantity but not a price, limit order price being the maximum allowable bid or minimum allowable ask which allows the order to be executed. A single trader may submit a single order per batch interval, which are not visible to other agents during the batch interval, the auction being a sealed bid.

Figure 1

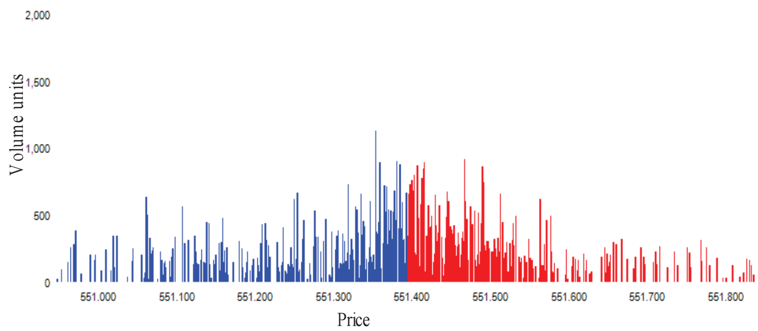
Order book example from the agent-based model simulation



Source: *Agent-based model simulations with Adaptive Modeler software application*
Note: Blue bars represent bid volume orders, red bars represent ask volume orders, black bars represent bid and ask volume orders at equal prices, before market clearing.

Figure 2

Order book example from the agent-based model simulation



Source*: Agent-based model simulations with Adaptive Modeler software application*
Note: Blue bars represent bid volume from unexecuted orders, red bars represent ask volume from unexecuted orders, after market clearing.

The trading mechanism used by the Adaptive Modeler software application to simulate the artificial stock market is set as call auctions mainly because many stock markets use this mechanism. In the call auction markets, agents introduce bid or ask orders, each order consisting of a price and quantity. The bids and asks orders received are put in the order book and an attempt is made to match them. The price of the trades arranged must lie in the bid-ask spread (interval between the selling price and buying price).

3. The cycle of the evolutionary agent-based model

The genetic programming evolutionary cycle of each simulation period can be summarized in the following points:

- **Receive new quote bar**.
- **Agents evaluate trading rules and place orders**: Agents receive access to historical prices and evaluate the evolution of prices according to the technical analysis generated by their trading rules found in the genomes, resulting in a desired position as a percentage of wealth limited by the budget constraints, and a limit price. Agents are two-way traders during the simulations, meaning that they are allowed to both sell and buy during multiple simulation periods, and they are one-way traders during a single simulation period (in this case a day) corresponding to an auction, as they are able to submit only one order per auction, either buy or sell. The position is generated in a random manner, while the limit price is generated after a technical analysis has been performed, according to the genome structure which represents trading functions.
- **Artificial stock market clearing and forecast generation**: The artificial stock market determines the clearing price in the call auction, which is a discrete time double-sided auction mechanism in which the artificial stock market collects all bids (buying orders) and asks (selling orders) submitted by the agents and then clears the market at a price where the supply quantity equals the demanded quantity. The clearing price is the price for which the highest trading volume from limit orders can be matched, thus all agents establish their final positions and cash at the same time. In case the same highest trading volume can be matched at multiple prices, then the clearing price will be the average of the lowest and the highest of those prices. The artificial stock market executes all executable orders and forecasts the price for the next simulation period. The forecasted price is set equal to the clearing price.

- **Breeding**: During the breeding process, new agents are created from best performing agents in order to replace the worst performing agents, creating new genomes by recombining the parent genomes through a crossover operation, and creating unique genomes by mutating a part of the genome. The breeding process repeats at each bar, with the condition that the agents must have a minimum breeding age of 80 simulation periods, in order to be able to assess the agents' performance.

In order to obtain random seed, the Adaptive Modeler software uses the Mersenne Twister algorithm [9] to generate pseudo random number sequences for the initial creation of trading rules or genomes and for the crossover and mutation operators of the breeding process.

The genomes attached to the each agent uses a tree composed of genes which generates the trading strategies. The initial node in the genetic program tree combines the position desired in the security generated randomly, and the limit price value generated by a collection of functions working as a technical analysis on the historical prices, into a buy or a sell order advice. The desired position value ranges between -100% (short position, or selling position) and 100% (long position, or buying position) which is randomly generated from a uniform distribution. The limit price value is generated by a collection of functions which uses simple technical indicator initially generated in a random manner from the list of functions selected to be used in the model, which develop during the breeding process, in order to generate the limit price for the buy or sell order.

The buy or sell order is introduced in the market after comparing the desired position with the agent's current position and calculating the number of shares that need to be bought or sold, taking also in consideration the available cash. The trading rules of the model use historical price data as input from the artificial stock market, and return an advice consisting of a desired position, as a percentage of wealth, and an order limit price for buying or selling the security. Through evolution the trading rules are set to use the input data and functions (trading strategies) that have the most predictive value.

The agents' trading rules development is implemented in the software by using a special adaptive form of the Strongly Typed Genetic Programming (STGP) approach, and use the input data and functions that have the most predictive value in order for the agents with poor performance to be replaced by new agents whose trading rules are created by recombining and mutating the trading rules of the agents with good performance. In order to do this, a dynamic fitness function is used to evaluate the performance of the agents, and only the most recent simulation periods of the epoch are taken in consideration for computing the fitness function. As regards to the breeding process at each simulation period, the adaptive form of the STGP approach only takes in consideration a percentage of 5% of the total population of agents.

The STGP was introduced by Montana (2002) [2], with the scope of improving the genetic programming technique by introducing data types constraints for all the procedures, functions and variables, thus decreasing the search time and improving the generalization performance of the solution found. Therefore, the genomes (programs) represent the agents' trading rules and they contain genes (functions), thus agents trade the security on the artificial stock market based on their technical analysis of the real market historical price data.

During the breeding process, new offspring agents are created from some of the best performing agents to replace some of the worst performing agents. In order to achieve this, at every bar, agents with the highest value of the Fitness Return function are selected as parents, and the genomes (trading rules) of pairs of these parents are then recombined through genetic crossover to create new genomes that are given to new offspring agents. These new agents replace agents with the lowest value of the Fitness Return function. The fitness function is a metric of the agent's investment return over a certain period, therefore the Fitness Return function is computed as the wealth return over the last 80 analyzed quotes and represents the selection criterion for breeding.

The agents are endowed with wealth and a trading strategy which is called the genome, which is randomly created by taking in account the selected genes (which represent functions) using genetic programming. Broker fees are fixed at 10 points of value for each transaction. There is no market maker. All the parameters of the model are described in Table 1.

Table 1

General settings of the models. Market and agents' parameters configuration in the simulations

Parameter Type	Parameter Name	Parameter Value
Market Parameters	No. of simulation periods	Max 20,000
	No. of agents	2,000
	Minimum price increment	0.01
	Average bid/ask spread	0.01%
	Fixed Broker fee	10
Agent Parameters	Wealth Distribution	100,000 initial wealth for each agent
	Min. position unit	5%
	Max. genome size	1,000
	Max. genome depth	20
	Min. initial genome depth	2
	Max. initial genome depth	5
	Genes	CurPos, RndPos, LevUnit, Rmarket, Cash, Bar, PndPos, IsMon, IsTue, IsWed, IsThu, IsFri, close, bid, ask, average, min, max, >, change, +, dir, isupbar, upbars, pos, lim, Advice, and, or, not, if
	Breeding Cycle Length	1 simulation period
	Minimum breeding age	80 simulation periods
	Initial selection: randomly select	100% of agents of minimum breeding age or older
	Parent selection	5% agents of initial selection will breed
	Mutation probability	10% per offspring

4. Conclusions

The hereto paper brings to light the importance of the agent-based modeling of multi-agent systems, used to model the stock market. The results of the academic studies of artificial stock markets using genetic programming are still in contradiction, mainly due to the variety of models which make it hard to classify and compare. Their complexity, lack of transparency and high number of degrees of freedom make understanding and further development very hard.

Due to the high diversity of financial data, the fitting of a general model for all the data has become an impossible mission, but adaptive models such as the one described in this paper represent a step further when it comes to stock market simulations techniques, stressing the importance of using an evolutionary model, a model that evolves and adapts to the new market conditions.

The main improvements brought by this type of models are the following: the fitness function is computed as the actual return of the artificial agent after trading in the artificial stock market; there is no over-fitting the historical data during the learning process due to the fact that the model evolves in a time-incremental way, and it does not optimize on historical data. This model uses a high number of artificial agents, namely 2,000 agents and 2,000 continuously adapting trading rules, which increases model stability and reduces sensitivity to random factors. Trading signals given by the model are based on the interaction of all artificial agents, and not just on a single trading rule. The agent-based model is dynamic, constantly evolving and adapting to market conditions.

Further research will focus on simulations of the model and testing whether trading strategies developed in the model could be used for hedging purposes.

Acknowledgement

This work was financially supported through the project "Routes of academic excellence in doctoral and post-doctoral research - READ" co-financed through the European Social Fund, by Sectoral Operational Programme Human Resources Development 2007-2013, contract no POSDRU/159/1.5/S/137926.

References

1. Witkam, J. (2003) „Altreva Adaptive Modeler”, [Online], Available: http://www.altreva.com/technology.html [2 February 2015] .
2. Montana, D. (2002) “Strongly Typed Genetic Programming”, Evolutionary Computation, vol. 3, no. 2, 199-230.
3. Daniel, G. (2006) “Asynchronous simulations of a limit order book”, University of Manchester, Ph.D. Thesis.
4. Aloud, M., Fasli, M. and Tsang, E. (2013) “Modelling the High-Frequency FX Market: an Agent-Based Approach”, University of Essex, Technical Reports, Colchester.
5. Arthur, B., Holland, J., LeBaron, B. (1997) “Asset Pricing Under Endogenous Expectations in an Artificial Stock Market”, Published

In: Arthur, W.B., Durlauf, S.N., Lane, D. The economy as an evolving, complex system II, Redwood City, CA., Addison Wesley, pp. 15-44.

6. Rust, J., Palmer, R., Miller, J. (1989) “A Double Auction Market for Computerized Traders”, Santa Fe Institute, New Mexico.
7. Phelps, S., Marcinkiewicz, M., Parsons, S. (2005) “Using population-based search and evolutionary game theory to acquire better-response strategies for the double-auction market”. In Proceedings of IJCAI-05 Workshop on Trading Agent Design and Analysis (TADA-05).
8. Walia, V., Byde, A., Cliff, D. (2003) “Evolving Market Design in Zero-Intelligence Trader Markets”. In IEEE International Conference on E-Commerce (IEEE-CEC03), Newport Beach, CA., USA.
9. Matsumoto, M., Nishimura, T. (1998) “Mersenne Twister: A 623-dimensionally equidistributed uniform pseudorandom number generator”. ACM Trans. on Modeling and Computer Simulation, vol. 8, no. 1, pp. 3-30.