PHYS 371 - Project Final Report - Agent Based Market Simulations With Levy-Levy-Solomon Model

Ozan Cem Baş

Department of Physics, Bilkent University, 06800 Ankara, Turkey

September 26, 2024

Abstract

Agent based simulations have been a large area in many research areas, from physics to sociology. Especially in economy, many different simulation methods, including agent based models, have been tried to have predictive power over the markets. In the area of agent based simulations, Levy-Levy-Solomon (LLS) model has had great influence as one of the pioneering works in this area [3]. However, besides providing an insight on market booming and crashing mechanisms, the results of the model cannot replicate real world markets [5]. In this project, I recreated the LLS model and tested it for several cases. Then, I assessed the agents' success and its dependence on agents' parameters and market conditions.

1 Introduction

1.1 Problem Statement

Agent based models and their simulations are computational tools where the dynamics of a group, population or an environment are evaluated by examining the interaction of the individual agents, which make up the population, with each other and with several other environmental factors. Contrary to the traditional methods, agent-based models allow for heterogeneous agents and stochastic events taking place. These properties of agent-based models makes studying the complex events taking place in the system possible. Therefore, they are widely employed by several fields including physics, biology, computational sociology, and economics. Especially in economics, they are quite powerful tools and are used for their predictive power and examining the emergence of patterns. In this project, I recreated the agent based model that was proposed by Moshe Levy, Haim Levy, and Sorin Solomon in their 1994 paper with Python [4]. I will examine the effects of different initial conditions and the interaction, the final wealth distribution in the presence of heterogeneous investors.

1.2 Importance of The Problem

Being able to make predictive decisions and figuring out the future value of an asset or a financial market is a major problem that is worth billions of dollars. Being able to predict the value of an asset accurately, even a second later from the initial time, gives the investor an arbitrage that can be exploited in order to make exponentially increasing profits. There are several corporations built just for this purpose, like hedge funds. However this problem is quite hard to tackle because of the extremely complex dynamics taking place in a financial market. Therefore, building and testing simulations are a big part of the risk evaluation of an asset.

1.3 Literature Review

Several models for simulating the macroscopic behavior of a financial market has been proposed by several authors with different approaches. One of the earlier model suggested by G. Kim and H. Markowitz [2]. Their model consists of two types of investors: rebalancers and portfolio insurers. While rebalancers will keep a constant fraction of their wealth constantly in stocks, portfolio insurers keep the value of their assets over a minimum insured value until a specified amount of time. The price of the stock is determined by supply and demand. Every investor evaluated their portfolio at a random time. If there are no buyers, price decreases, and if there are no sellers price increases. And anything in between if both exists. Although the model was pioneering, it failed to comply with the empirical facts of the real markets [5].

Another one of the most influential models were proposed by S. Solomon, M. Levy and H [4]. Levy. The LLS model is also an agent based microscopic simulation of the stock market. The main working mechanism is that each investor has a memory property. They look back at the returns of the stocks as long as their memory every time step and determine how much of their portfolio they want to keep in stocks and bonds. The model was successfully in producing booms, market bubbles, and crashes, as it is intended to, and could produce complex patterns [5]. However, the model still lacked the ability to produce the empirical statistics that real financial markets produced [7].

A percolation approach was provided by Cont-Bouchaud model [6]. In this model, the investors are represented by points on a lattice. Each point on the grid is assigned on or off, meaning the existence of an investor, with probability p. As a result, clusters of investors are formed, and the investors in a cluster buy or sell, with an equal probabilities and proportional to the cluster size. This model was proven to be a lot simpler and was capable of producing the return volatility of actual markets [5].

For my project, I decided to focus on building, experimenting with, and possibly improving on the Levy-Levy-Solomon model due to the abundance of explaining documents and ease of implementing in Python.

1.4 Previous Achievements

Besides microscopic simulations, implementation of the concepts of complex systems to financial markets are proven to be effective. One of the most recent and successful models is the The Johansen-Ledoit-Sornette (JLS) model that was used to predict the bubble and crash of Nikkei [1]. This model took a different approach by using log-periodic power law to predict a market crash.

1.5 Contribution and Aim

Levy-Levy-Solomon (LLS) model is an influential and complex model. In my work, I aim to analyze the failure and success of the agents in different conditions and come up with robust strategies that could be implemented to real markets.

1.6 Expected Achievements

What I want to achieve in this project is to get more realistic results from the LLS model and, possibly, to explain mechanisms for bubble formations and crashes for complicated markets produced by this model.

2 Methodology

2.1 Model Description

The model that was proposed by Levy-Levy-Solomon is an agent based model for the price evolution over time for the stock market over time, where the agents are the individual investors that make up the market . All investors the investor will distribute their wealth in two assets: bonds and stocks. Bonds are risk-free assets that do not change in price and pay up a fixed amount of interest, r on the invested capital. On the other hand, stocks have a varying price, P_t and this price is determined by the demand of all the investors, therefore are risky assets.

In each time t, all investors evaluate their investments and determine how much of their money they want to keep in stocks and bonds. This ratio is determined by the memory length, utility function of the investor and the interest rate, dividend rate, history, and the initial price of the market.

Every investor has a positive integer memory that does not change throughout the simulation. The memory of the investor determines how many time steps back does the investor look at the return on investment does the stocks have. Say, an investor that has a memory of 10 days will only look at the last 10 days of the returns of stocks to determine how many stocks he wants to have in the next day. The ratio of how much return on investment did stocks had on a given time is called the history H_t and calculated in the following way,

$$H_t = \frac{P_t - P_{t-1} D_t}{P_t} \quad (1)$$

where P_t and P_{t-1} are the prices of the stocks at times t and $t-1$ correspondingly, and D_t is the dividend per stock the investor holds.

In each time step $t+1$, every investor tries to maximize their expected wealth at $t+1$, $W_h(i)$. To do so they find the optimum ratio of stocks and bonds in their portfolio. This preference is characterized by a utility function, $U(W)$ that is the same function for all the investors. To determine this ratio, every investor tries to maximize their utility $U(W)$. This common utility function is, taken as $U(W) = ln(W)$.

Now that the fundamental concepts are established, the simulation process will be explained.

2.2 Mathematical Formulations

At every time t, before the trading starts, the income gain generated by the interests on bonds and dividends is added to the initial wealth of each investor $W_t(i)$. Dividend is paid for each stock the investor i holds $N_t(i)$, and interest is given on the wealth that is not on stocks $(W_t(i) - N_t(i)P_t)$. So that, the wealth of i-th investor after the income gain $W_t(i)$ is given by,

$$W_t^*(i) = W_t(i) + (W_t(i) - N_t(i))r + N_t(i)D_t \quad (2)$$

where r is the interest rate, D_t is the dividend given at time t.

After the income gain is added, each investors decide on how many stocks they want to have at time $t+1$ for a hypothetical price at $t+1$, P_h. To do so, the hypothetical wealth W_h at a given future price P_h is calculated,

$$\begin{aligned} W_h(i) = W_t(i) + N_t(i)P_t + \\ (W_t(i) - N_t(i)P_t)r + (P_h - P_t)N_t(i) \end{aligned} \quad (3)$$

To find their estimated wealth in time $t+1$, each investor references the history of the stocks and the interest rate. Investors expect that each history from time t to $t-k+1$, where k is the memory length of the investor, will reoccur with equal probabilities. Therefore, each investors average utility is calculated,

$$\begin{aligned} E[U(W_h(i))] = \frac{1}{k} \sum_{n=t}^{t-k+1} ln[W_h(1 - X(i))(1-r) \\ + W_h(i)(1 + H_n)X(i)] \end{aligned} \quad (4)$$

where $X(i)$ is the ratio of wealth that the investor will keep in stocks. The $X(i)$ value that makes $EU(W_h(i))$ maximum is calculated numerically and detonated as $X_h^*(i)$. To account for the effect of random interactions in-between the investors, each investor is assigned a Gaussian random variable with mean 0 and standard deviation σ. This sigma value gives the investors some amount of difference in preference, and is chosen between 0.001 to 0.2 [3]. And this random variable $\epsilon(i)$ is added to this calculated X_h value. So that,

$$X_h(i) = X_h^*(i) + \epsilon(i) \quad (5)$$

I must be noted that $X_h < 0$ corresponds to taking short positions, and $X_h > 1$ corresponds to borrowing money. Both of these are prohibited in this model. Therefore the value of X_h is bound between 0 and 1.

Then the personal demand of investor i is found as,

$$N_h(i, P_h) = \frac{W_h(i) X_h(i)}{P_h} \quad (6)$$

Adding the personal demand of all investors gives the collective demand of the investors for a given future price P_h,

$$N_h(P_h) = \sum_i N_h(i, P_h) \quad (7)$$

The price of the stocks at time $t+1$ is determined by the equilibrium of supply and demand. Since the number of stocks in the market does not change, the supply is constant for all prices. And the equilibrium is achieved when the collective demand is equal to the supply, which is the number of stocks in the market. And the price at which this equality happens is determined to be the new market price.

Finally, after the price is set as P_{t+1}, the wealth of each investor is updated, adding all the mentioned incomes, gains and losses.

$$W_{t+1}(i) = W_t(i) + N_t(i)D_t \\ +(W_t(i) - N_t(i)P_t)r + N_{t+1}(i, P_{t+1})P_{t+1} \quad (8)$$

By completing this, the trading at time $t+1$ is completed, which ends one cycle. This whole cycle is done for desired amount of times.

2.3 Relationship between Concepts

The major concept that are related in LLS model are the price, demand, and returns. Their relationship can be explained as follows. Returns are derived directly from the current price and the past price,

$$H_t = \frac{P_t - P_{t-1} + D_t}{P_t} \quad (9)$$

The demand does not depend strictly on price, but rather it depends on the returns, also referred as the history of the stocks. And the cumulative demand is needed to calculate the price at time $t+1$. These core element make up the simulation and their calculation simulated the model forward in this loop.

2.4 Algorithms

In the LLS model, there are two instances where a numerical calculation has to be implemented. When determining the demand at an initial price, each investor tries to find the optimal partition of their wealth that maximizes the estimated future wealth. This partition value $X_h(i)$ has to be calculated numerically. Since the interval is well defined, from 0 to 1, and the demand function generally does not have high frequency terms, I wrote a simple algorithm of my own. Initially, six points are selected to be 0.01, 0.2, 0.4, 0.6, 0.8, and 0.99 and the estimated wealth is calculated at these wealth partitions. Then, the index of two $X(i)$ values that return the largest two estimated wealth are recorded. Finally, value that brings the least wealth is found, and this value is value is replaced by the average of the two recorded $X(i)$ values. This iteration is repeated 5 times for good measure.

The more computationally demanding numerical calculation occurs when calculating the equilibrium price for the time $t+1$. To find this value, the price that makes the cumulative demand of every investor equal to the total number of stocks has to be found. I found most success in implementing Brent's method for root finding, where the function whose root has to be found is $f(P_h) = N_h(P_h) - S$, S being the number of stocks in the market and is a constant. Brent's method is a hybrid method of bisection, secant, and inverse quadratic interpolation methods. I found this method to be the fastest and most reliable for my usage. First the initial interval to find the equilibrium price P_{t+1} is taken as from 0 to P_t, the initial price. Then, the algorithm is used to check for roots. If there are not roots within the interval, the interval is updated to be from P_t to $P_t + 1000$. This process is repeated as many times as needed until there is a root.

2.5 Pseudo-code

Here I explained the simplified workings of the code:

1. An instance of market object is "Market" created with necessary parameters (initial price, initial history, etc.).

2. Desired number of agents with initial wealth and memory length is added to the market.

3. For the initial price and stock history, get the summed demand of all investors as N_h.

4. Run Brent's method for the interval for the function $N_h - S$ for the interval 0 to P_t

5. **if** the algorithm does not converge

   Go to step 4 with the interval P_t to $P_t + 1000$

6. The converged price is saved as $P_h = P_{t+1}$.

7. Update the wealth and number of stocks each investors according to their personal previously owned stocks and their demand.

8. Update the stock's history and the price.

9. One time step of the simulation is completed. Go to step 3 for desired amount of simulation length.

2.6 Testing Procedures

Before modifying the model to implement the changes I proposed, I tested the model to make sure the model works as intended. Doing so was rather simple because the authors of the original paper and many other literature sources have shared their verified results for the fundamental cases of the model.

The simplest test case for the model is the case of homogeneous agents, with same memory length, with increasing and static dividends. As explained by the authors, these conditions are expected to result periodic, square wave like, price fluctuations with a periodicity of around two times the memory length [4]. This behavior should nod depend on initial price and the initial stock history. The main reasoning behind this behaviour can be explained. If the price increases initially, every investor determines that the price is going to increase the next time as well, therefore all of them buys. This makes the price boom up. However, after their memory length of time passes, they forget the initial price booming. Then, the small price fluctuations and low returns on investment deters the investors and everyone sells, resulting with a market crash.

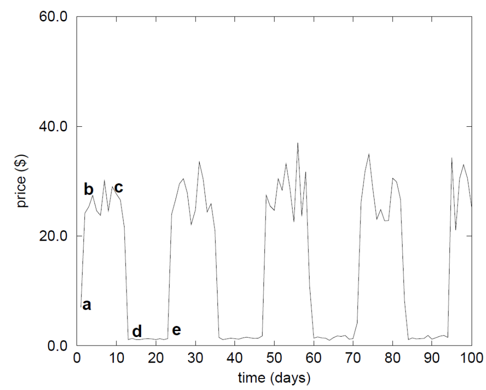

Figure 1: The result of the authors for 100 homogeneous investors with memory 10, and initial wealth 1000 [4]

Second test case I will use is once again provided by the authors. This case involves two types of investors with different memory lengths, m_0 and m_1, of equal numbers. Once again, this case also creates fully predictable results with the following patterns for a market with a period of $2m_0$ [4]:

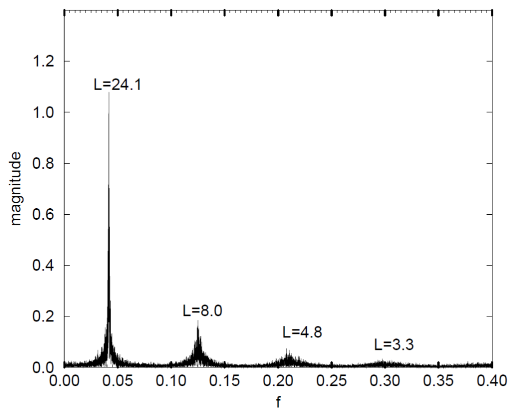

Figure 2: The Fourier transform of the results from the figure 1 [4]

1. $m_0 < m_1 < 2m_0$, m_1 is performing very poorly.
2. $2nm_0 < m_1 < (2n+1)m_0$, m_1 is doing relatively well.
3. $(2n+1)m_0 < m_1 < (2n+2)m_0$ for $n > 1$, better than case 1 but worse than case two.
4. $m_1 < m_0$, m_1 is doing extremely well.

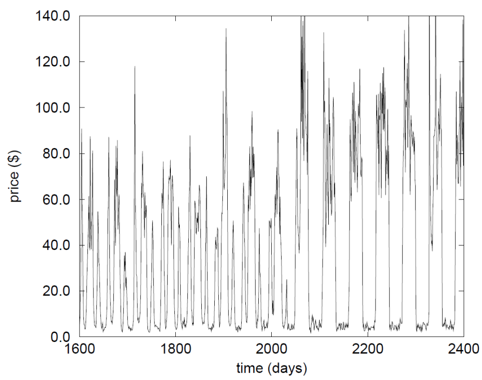

Figure 3: Stock price as a function of time in a market with two equal investor populations, memory spans 10 and 26 days[4]

The first test that were tries were the model verification experiments. The result I obtained for the homogeneous agent, with the same parameters as the authors' can be seen in figure 5

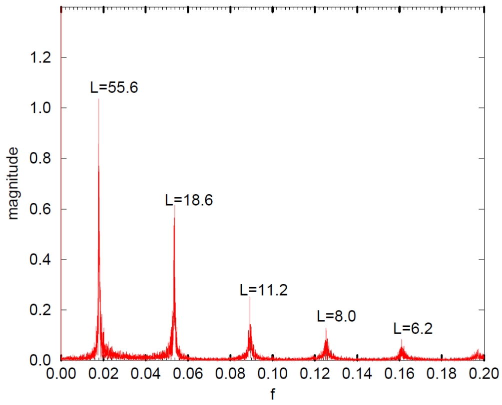

Figure 4: The Fourier transform of the results from the figure 3 [4]

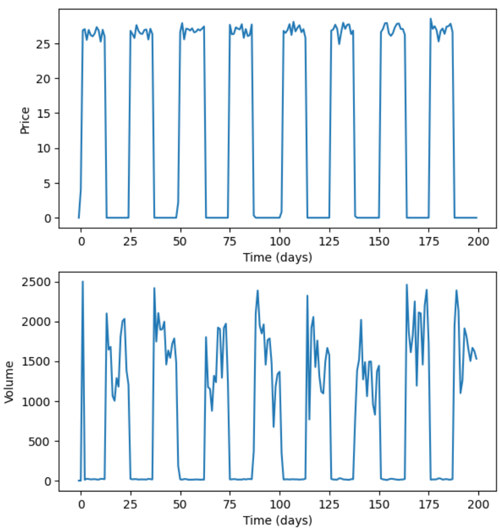

Figure 5: The result of the reproduced model 100 homogeneous investors with memory 10, and initial wealth 1000

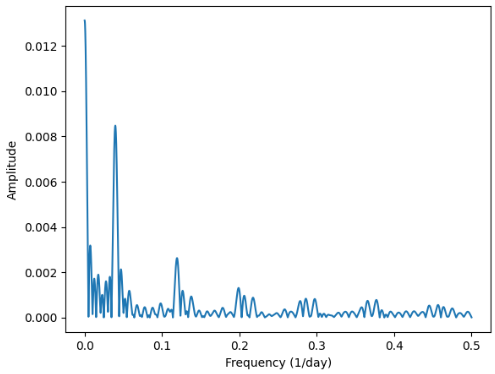

Figure 6: The Fourier transform of the results from the reproduced model 100 homogeneous investors with memory 10, and initial wealth 1000

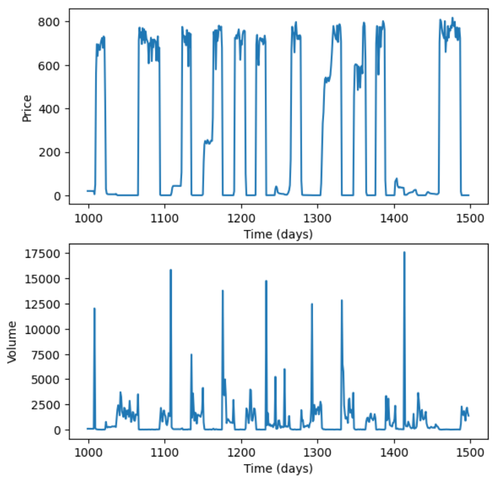

Figure 7: Recreated stock price as a function of time in a market with two equal investor populations, memory spans 10 and 26 days

As seen on figure 7, the Fourier transform of the reproduced results have a peak at around 0.0393. This frequency corresponds to a period of 25.4, which is in the expected range. For the second test scenario, I recreated the two agent case as well for a second test with the same parameters as the authors. The simulation results can be seen on figure 7.

As seen on figure 8, the Fourier transform has a peak at the frequency 0.0175, which corresponds to a period of 57 days. As this is really close the reported value of 55.6, I have verified that the model is functioning correctly with two tests.

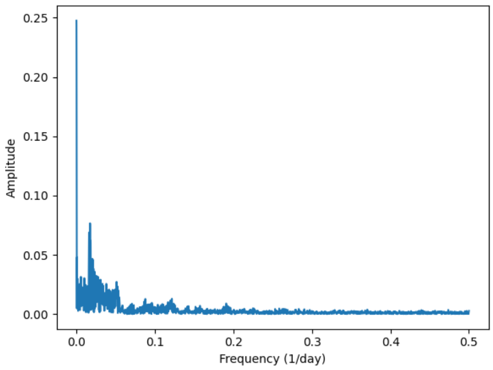

Figure 8: The Fourier transform of the results from the figure 7

3 Results

3.1 Parameter Testing

I tested the model for three three investors for different conditions. First I tried three investor types of memories 10, 141, and 256, 33 investors each, for stationary dividends. Dividends are a measure of the earnings of a company, and are proportional to the growth of the company. In the case this value is constant, the prices are expected to stay within a tighter interval. However I could not complete any of the simulations for this case for the simulations took too long. Most likely because of the algorithm I used for price calculation, for lower prices, calculating the equilibrium price takes longer than higher prices. Also, due to the relatively low prices and fluctuations, the results were not significant.

I tried the same simulation for increasing dividends over time. The results can be seen in figures 9 and 10.

The results indicate that the price fluctuates almost in a decaying exponential until it either booms or crashes. Measuring the period of the occurrence of these oscillating decay shows that they occur around every 250 to 270 days, which corresponds to the agents with memory 256. The oscillations taking place inside these funnels are most likely caused by the agents with memory length 10, since the period of the oscillations varies from 20 to 26 days. Due to the agents with 141 days of memory, I anticipated to observe an event with a period of 300 days. However this was not observed. This absence

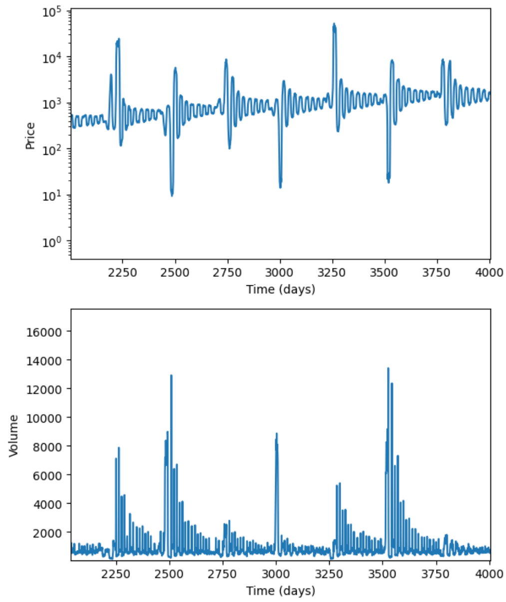

Figure 9: The simulation results for three different classes of investors of memories 10, 141, and 256

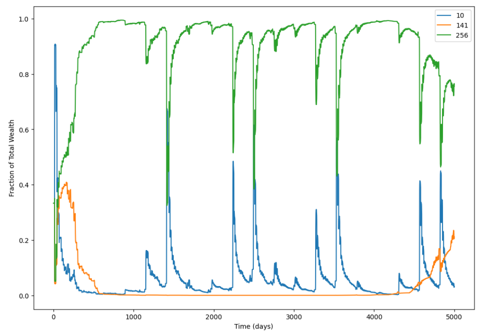

Figure 10: The wealth distribution of each investor class for all times for memories 10, 141, and 256

is clarified by observing that in those time periods, the agents with 141 days of memory owned a really small fraction of the total wealth of the market, therefore could not influence the prices that much.

I continued by changing the memory lengths of the investors. It can be seen that agents with 10 days of memory played an important role in dynamics and trend setting in the simulation mentioned in figure 9. For the next run, I set a market with agent memories 20, 141, and 256. I expected to see similar patterns with different periods, and indeed, that is what happened. In the figures 11 and 12, you can see the results.

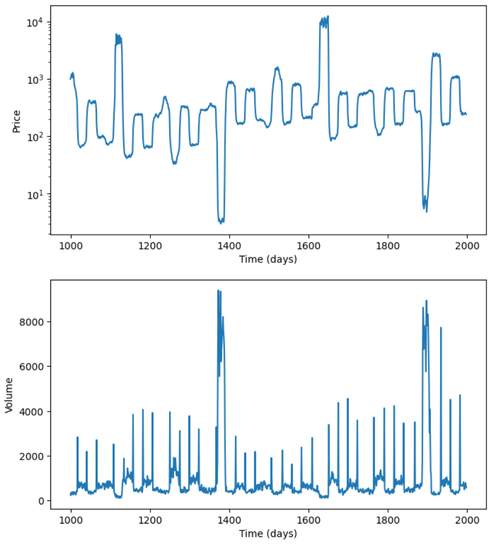

Figure 11: The simulation results for three different classes of investors of memories 20, 141, and 256

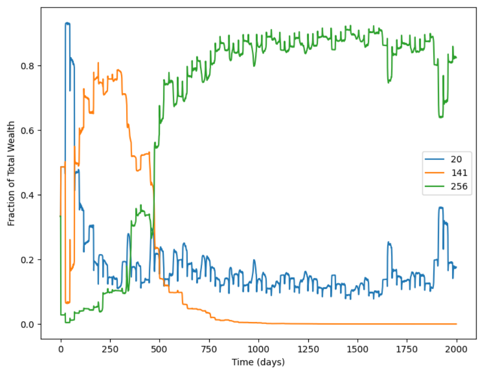

Figure 12: The wealth distribution of each investor class for all times for memories 20, 141, and 256

As expected, periodic booms and crashes occur with smaller oscillations occurring in between them. The smaller oscillations occur with a period of roughly 56 days, which confirms that they are caused by the agents with 20 days of memory.

Finally, I investigated the case of investors with memories of lengths 40, 141, 256. The results of this run can be seen in the figures 13 and 14.

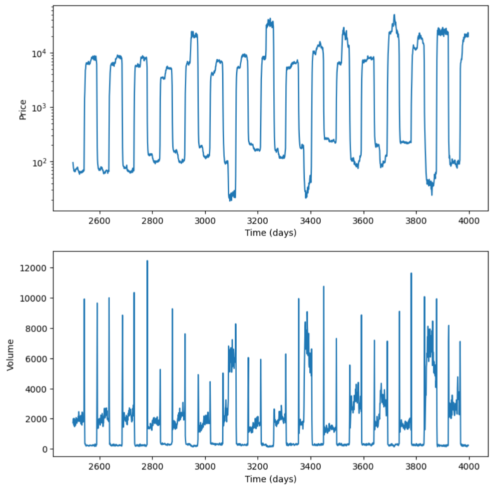

Figure 13: The simulation results for three different classes of investors of memories 20, 141, and 256

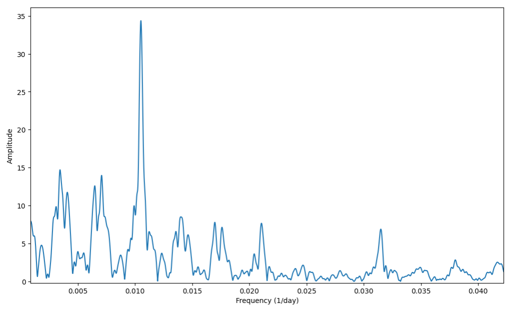

Figure 14: The Fourier transform of the prices for memories 40, 141, and 256

Once again, the same kind of oscillations occur, followed by a boom or a crash. However, judging by the Fourier transform, the smaller oscillations have a period of 100 days instead of the anticipated 80 days. Also, unlike the other simulations, the Fourier transform has much more obvious peaks that can be analysed. the highest peak corresponds to a period of 100 days. The two peaks located at frequencies 0.00344, 0.00707, and 0.01398 corresponds to periods of 290, 141, and 71 days. Therefore these peaks are likely due to the presence of the investors with 141 days of memory.

3.2 Literature Comparison

My findings differ from the results of the authors for the three investors case. They claimed that for three types of investors and above, there would not be any patterns and the results are expected to appear mostly random and unpredictable [4]. However, in my simulations, the results show clear patterns and predictable behavior.

3.3 Causality Analysis

In the simulations, I expected to see no patterns in the case of agents with three or more different memory lengths. But apparently, there is still pattern formation and predictability in the simulation. During the simulations, I realized some patterns that are observed in the single and two memory length systems have carried on to the three memory system as well in the frequency domain. In figure 12, most of the peaks are just the frequencies of single agent system and their higher harmonics. One thing to be noted is the presence of the closely packed peaks. Although I do not have a direct explanation on them, they seem to be the product of the effect of the other investors with different memories. .

4 Discussion

4.1 Project Aim Recap

This project was aimed towards observing the complex behavior of the prices for the LLS model and to have a predictive power on what kind of investors succeed in the market.

4.2 Achievements Summary

According to the simulation results, there does not seem to be a single memory length that would result the highest gains in all situations. Though it is seen that having a memory of at least 3 times the main price fluctuation period allows for higher profits.

4.3 Importance of Results

Although the results from the LLS model simulation cannot be directly implemented to real world applications, they still imply methods of great importance. Having a long memory corresponds to being patient in real life. It must also be noted that the ones who set the trends in the market usually will come out at a loss. The model simulation results show that the most beneficial strategies to use are to either act faster than the trends, or to disregard the trend. In a real market, constant periods do not happen that often. Even when they do, they do not last to long. Therefore implementing the first strategy would not be possible, or sustainable at the very least. Second strategy on the other hand is already a widely used strategy in the form of moving averages and the idea of mean reversal behavior of the markets.

4.4 Future Directions

To summarize, LLS model was one of the pioneering works in the field of econophysics and it had some shortcomings. One suggestion to improve upon these is to try a large amount of market conditions, agent numbers, utility functions or to add more randomness. However, it was shown that for a very large number of parameters and initial conditions, the LLS model cannot create the characteristic behaviors of real markets, like scaling laws [5]. A more sensible approach would be to change the model to account for agent-to-agent interactions, and herding behavior. This could be done by introducing perculations between the agents on top of the already existing structure of the model.

References

[1] Anders Johansen and Didier Sornette. "FINANCIAL "ANTI-BUBBLES": LOG-PERIODICITY IN GOLD AND NIKKEI COLLAPSES". In: *International Journal of Modern Physics C* 10.04 (June 1999), pp. 563–575. DOI: 10.1142/s0129183199000437. URL: https://doi.org/10.1142/s0129183199000437.

[2] Gew-Rae Kim and Harry M. Markowitz. "Investment rules, margin, and market volatility". In: *The Journal of Portfolio Management* 16.1 (Oct. 1989), pp. 45–52. DOI: 10.3905/jpm.1989.409233. URL: https://doi.org/10.3905/jpm.1989.409233.

[3] Moshe Levy, Haim Levy, and Sorin Solomon. "A microscopic model of the stock market". en. In: *Econ. Lett.* 45.1 (May 1994), pp. 103–111.

[4] Moshe Levy, Nathan Persky, and Sorin Solomon. "The complex dynamics of a simple stock market model". In: *Int. J. High Speed Comput.* 08.01 (Mar. 1996), pp. 93–113.

[5] Eleni Samanidou et al. "Agent-based models of financial markets". In: *Reports on Progress in Physics* 70.3 (Feb. 2007), pp. 409–450. DOI: 10.1088/0034-4885/70/3/r03. URL: https://doi.org/10.1088/0034-4885/70/3/r03.

[6] Dietrich Stauffer. "PERCOLATION MODELS OF FINANCIAL MARKET DYNAMICS". In: *Advances in Complex Systems* 04.01 (Mar. 2001), pp. 19–27. DOI: 10.1142/s0219525901000061. URL: https://doi.org/10.1142/s0219525901000061.

[7] Elmar Zschischang and Thomas Lux. "Some new results on the Levy, Levy and Solomon microscopic stock market model". In: *Physica A: Statistical Mechanics and its Applications* 291.1-4 (Mar. 2001), pp. 563–573. DOI: 10.1016/s0378-4371(00)00609-9. URL: https://doi.org/10.1016/s0378-4371(00)00609-9.