Project 5 - FYS3150 Computational Physics

Fredrik Østrem (`fredost`)
github.com/frxstrem/fys3150

Joseph Knutson (`josephkn`)
github.com/mathhat/Computational_Physics

December 9, 2016

Abstract

We are going to use four different models, each adding to the former, to simulate financial transactions between agents in a financial society. To simulate these transactions, we use Monte Carlo methods. Specifically, we will look at how to reduce wealth inequality, and we will see that having small transactions, and having no preference for the wealth of transaction partners, but high preference for previous partners, will do most to decrease inequality.

Contents

1 Introduction

The aim of this project is to simulate transactions of money between financial agents, people, using Monte Carlo methods. The final goal is to extract a distribution of income as function of the income m. From Paretos work (Pareto 1897), and from empirical studies, it is known that the higher end of the distribution of money, rich end, follows a power distribution.

$$\omega_m = m^{-1-\alpha}$$

with $\alpha \in [1, 2]$. We will follow the analysis made by Patriarca et al. (2004).

We assume we have N agents that exchange money in pairs (i, j). We assume also that all agents start with the same amount of money $m_0 > 0$. At a given "time step", we choose a pair of agents (i, j) and let a transaction take place. This means that agent i's money m_i changes to m_i' and similarly we have $m_j \to m_j'$. Money is conserved during a transaction:

$$m_i + m_j = m_i' + m_j'. \tag{1}$$

In order to decide which agent gets what, we draw a random number ϵ. The change is done via a random reassignement (a random number) ϵ, meaning that

$$m_i' = \epsilon(m_i + m_j), \tag{2}$$

leading to

$$m_j' = (1 - \epsilon)(m_i + m_j). \tag{3}$$

The number ϵ is extracted from a uniform distribution. In this simple model, no agents are left with a debt, that is $m \geq 0$. Due to the conservation law above, one can show that the system relaxes toward an equilibrium state given by a Gibbs distribution

$$w_m = \beta \exp\{(-\beta m)\},$$

with

$$\beta = \frac{1}{\langle m \rangle},$$

and $\langle m \rangle = \sum_i m_i/N = m_0$, the average money. It means that after equilibrium has been reached that the majority of agents is left with a small number of money, while the number of richest agents, those with m larger than a specific value m', exponentially decreases with m'.

In each simulation, we need a sufficiently large number of transactions, say 10^7. Our aim is find the final equilibrium distribution w_m. In order to do that we would need several runs of the above simulations, at least 10^3–10^4 experiments.

2 Theory of Models

Now there are multiple models to explore. Our first model picks two entirely random agents, i and j, and makes them trade a random amount decided by the factor ϵ (see equation (2) and (3)). The following three models each add another parameter to the previous resulting in more realistic distributions of money.

2.1 Second model: The Saving Factor λ

In our simplest model, nothing is stopping our agents from throwing away all their money. In order to make our agents a bit more rational, we will limit the amount of money they trade away.

The conservation law of (1) holds, but the money to be shared in a transaction between agent i and agent j is now $(1 - \lambda)(m_i + m_j)$. This means that the amount of money our previous model would transfer is reduced by $\lambda\%$, which can be written as

$$m_i' = m_i + \delta m$$

and

$$m_j' = m_j - \delta m,$$

with

$$\delta m = (1 - \lambda)(\epsilon m_j - (1 - \epsilon)m_i), \tag{4}$$

From (4), if $\lambda = 0.8$, our agents will never trade more than 20% of their total money. As λ goes to 1, our agents only engage in microtransactions. This reduces risk and you will soon see how helpful λ can be for the reduction of poverty.

2.2 Third model: The Preferential Factor α

Our next model is the same as the last, except our agents will not only have risk reduction through λ, but also a preferred trading partner through α. How our agents will come to prefer each other will be based on how equal they are economically.

$$P(i, j; \alpha) \propto |m(i) - m(j)|^{-\alpha} \tag{5}$$

We use the expression in (5) to produce a probability for agent i and j's transaction. We can't know what the proportionality constant is without calculating the probability of every pair of agents (which is costly). Instead we generate a random number r between 0 and S, such that S is larger than the probability expression some large fraction of the time. We discard a transaction between pair of agents if

r is less than the probability expression in (5). If S is too small, any pair where the probability expression in (5) is larger than S will always be accepted, causing a less accurate distribution, while if S is too large, too many pairs of agents will be discarded, so computation time will take longer.

2.3 Fourth model: Memory Factor γ

This time, we're adding another parameter to the way the agent's prefer trading with each other. In real life, people build trust with those they've traded with before. We're going to make the agent's remember each other. The goal is to make them prefer those they've traded with before. The more trades that a pair has comitted in past, the more likely it is for that pair to trade again in the future.

The new probability expression will contain a new factor:

$$P(i, j; \alpha, \gamma) \propto |m(i) - m(j)|^{-\alpha} (C_{ij} + 1)^{\gamma}$$

C is a matrix which stores the amount of times that agent i and j have traded. This means that each time agent i and j trades, C_{ij} grows with 1.

We see that for $\gamma = 0$, the probability of a transaction is the same as before, but for $\gamma > 0$ the probability of trade increases.

3 Methods

To model the trading between our agents, we use random sampling, the essence of Monte Carlo methods. To complicate things further, we added probabilistic if-expressions to create preferential bonds between agents based on money and past transactions, requiring even more sampling.

To see if our simulations ever reach a form of equilibrium, we observe the variance of the agents' money. Doing so over time could prove useful for pinpointing when the distribution of money has stabilized.

3.1 Monte Carlo

All our simulations ran 10 million transactions for each of the 1 thousand cycles. Each of the $10^3 \times 10^7$ transactions consist of drawing two random numbers, i and j, representing two agents. We have both used different random number generators to do this. Fredrik used the C++ random number generator, while Joseph used the C function `rand()`.

$P(i, j; \alpha)$ and $P(i, j; \alpha, \gamma)$ were later calculated and compared to random samples between 0 and S. This was in order to decide if agents i and j knew each other were economically equal from before.

3.2 Variance and Equilibrium

Variance shows the average squared difference between the people's actual wealth and their initial money, m_0. It doesn't tell if there is extreme poverty, but the larger it is the more inequality we will find. To calculate it, we simply iterate over all our agents' money:s

$$\sigma_m^2 = \frac{1}{N} \sum_i^N (m_i - m_0)^2 = \left\langle m^2 \right\rangle - \left\langle m \right\rangle^2$$

where N equals the number of agent and m_i is agent i's money. The variance between our agents' money is calculated between every transaction (see Results).

4 Result

In all our simulations, we run 1000 runs with 10^7 transactions each.

4.1 First model: No preference

We start the simulation with the simplest model: Initial amount $m_0 = 1$ with no saving ($\lambda = 0$) and no preference for transaction partners ($\alpha = 0, \gamma = 0$). In this case, we expect the Gibbs distribution

$$w_m = \beta \exp(-\beta m)$$

where $\beta = \langle m \rangle^{-1} = m_0^{-1}$.

In fact, when we run the simulations, we get the distribution shown in fig. 1, we see that the distribution matches almost exactly up with the distribution be would expect, where $\beta = m_0^{-1} = 1$. This is further comfirmed by plotting the y axis logarithmically, in which case we get fig. 2, which clearly shows this relation.

We know that the variance of the Gibbs distribution is $\sigma_m^2 = \beta^{-2} = m_0^2$, and we can use this to find out how many transactions we need to get equilibrium. Before any transactions, the variance is zero (as every agent has the same amount of money); after equilibrium is reached, we expect the variance to vary around the theoretical variance. In fig. 3 we see the calculated variance of the simulated distribution plotted against the number of transactions, for different number of agents. We can see that it takes only about 10^2–10^3 transactions to reach equilibrium, and that this number is roughly proportional to N.
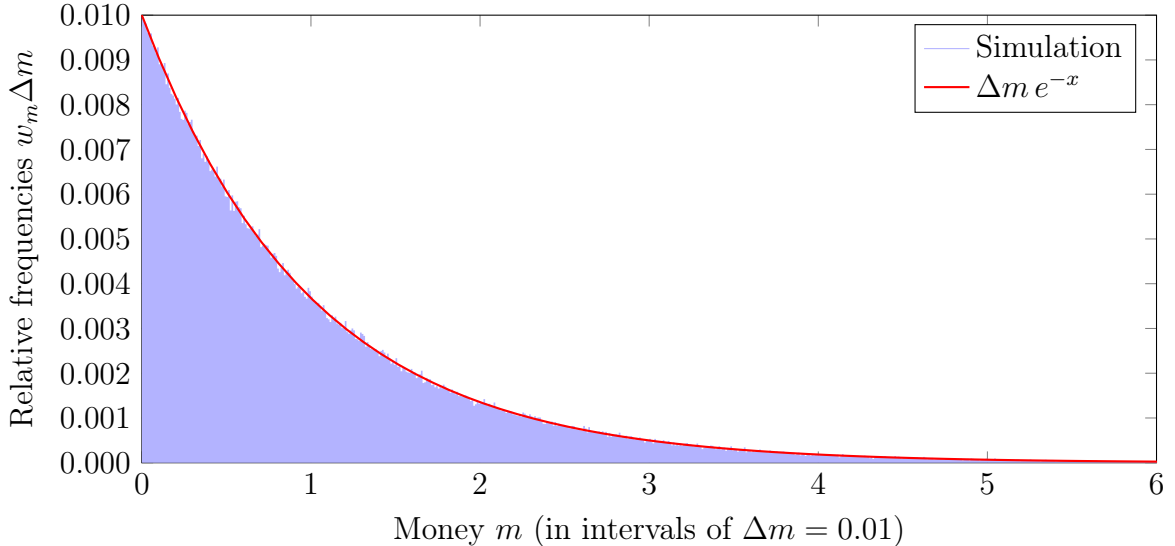

Figure 1: Histogram of the distribution of wealth after running the simple model ($\lambda = 0, \alpha = 0, \gamma = 0$) with 10^7 transactions 10^3 times, compared with the theoretical Gibbs distribution. Bin width is $\Delta m = 0.01$.
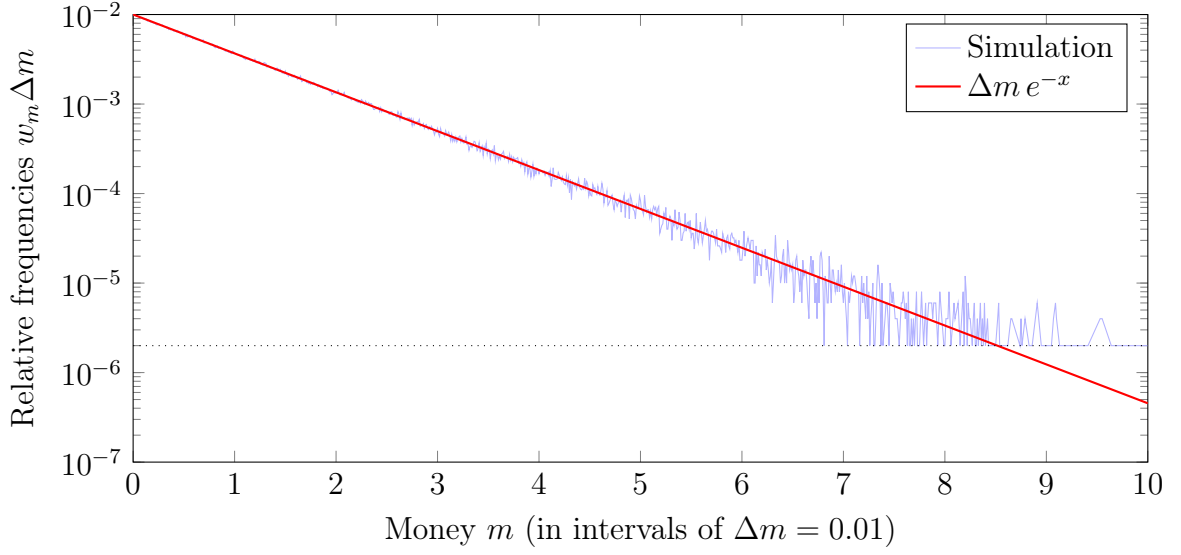
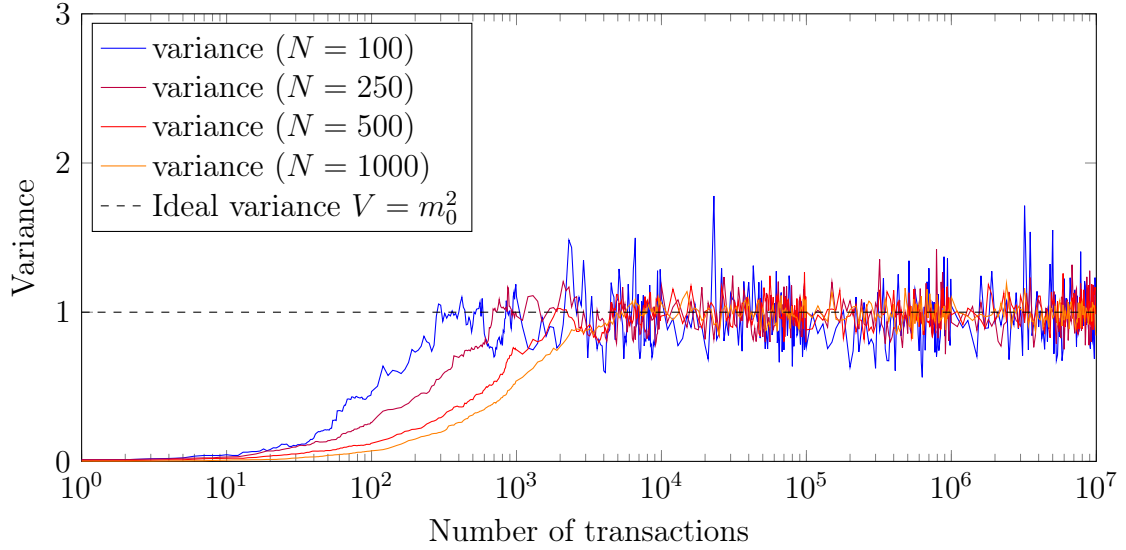Figure 2: The distribution from fig. 1, plotted on a logarithmic y axis.

Figure 3: Simulated distribution variance as a function of number of transactions, for different numbers of agents. Dashed black line marks the "ideal" variance (theoretical variance at equilibrium).

4.2 Second model: The Saving Factor λ

Next, we introduce saving into the equation: every time a transaction happens, each of the agents save a fraction λ before each transaction (as described in section 2.1). We run three simulations with saving fractions $\lambda = 0.25, 0.50, 0.90$, and look at how these alter the distribution of money. In fig. 4 we see the distributions of these simulations. We see that having a high saving fractions lowers lowers wealth inequality, and at very high saving fractions ($\lambda = 0.90$), the money is distributed according to an approximate narrow Gaussian distribution centered at $m_0 = 1$.
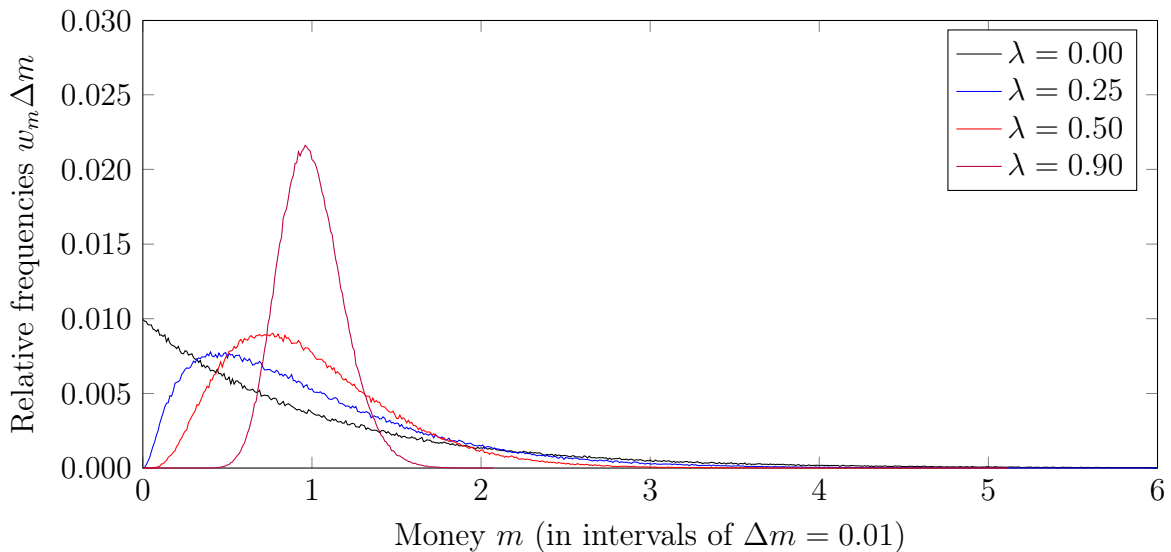

Figure 4: Simulated distributions after we introduce a saving fraction λ in the model.

4.3 Third model: The Preferential Factor α

We then introduce a new parameter α, that affects the agents preference for transaction partners with wealth close to their own (as described in section 2.2). We run eight new simulations with parameters $\lambda = 0.00, 0.25$ and $\alpha = 0.5, 1.0, 1.5, 2.0$ for $N = 500$ and $N = 1000$.

In figs. 5 to 7 we see the wealth distributions of these models. From these figures we can tell that a higher preferential factor α causes higher wealth inequality, since there are more poor people, and the richest people get richer, when we increase the α factor.

In fig. 7, we have plotted the same data for $N = 100$ with logarithmic axes. We expect this plot to be similar to figure 1 of Goswami and Sen (2014). Indeed, our plot is very similar, except that our plot does not reproduce the kink at $m \approx 2 \times 10^{-2}$ for the larger α values.

If the distribution followed a tail power law, we would expect the plots in fig. 7 to make a straight line in the tail. Instead
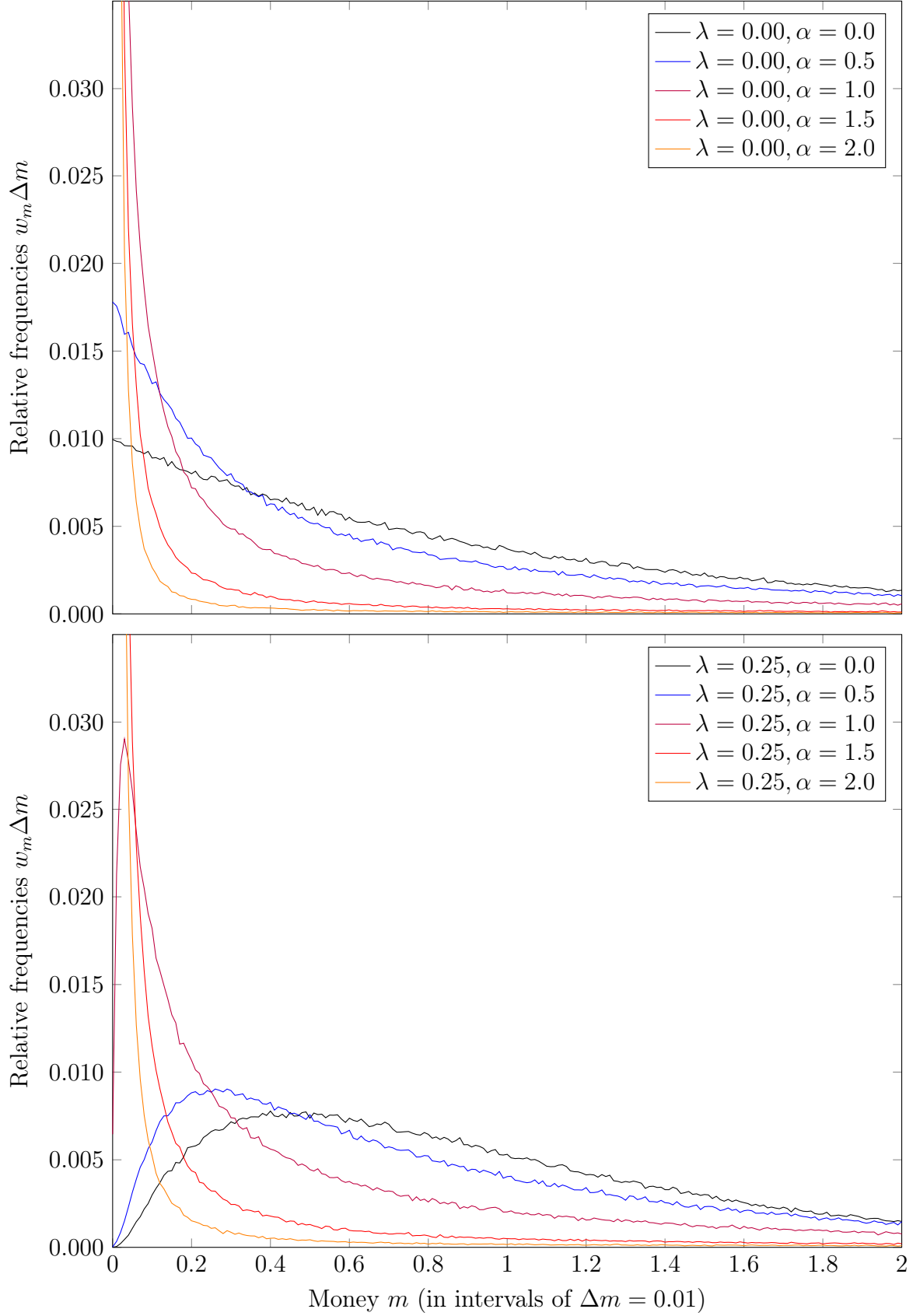
9

Figure 5: Simulated distributions after a "wealth preference" parameter α is introduced into the model. Number of agents is $N = 500$.
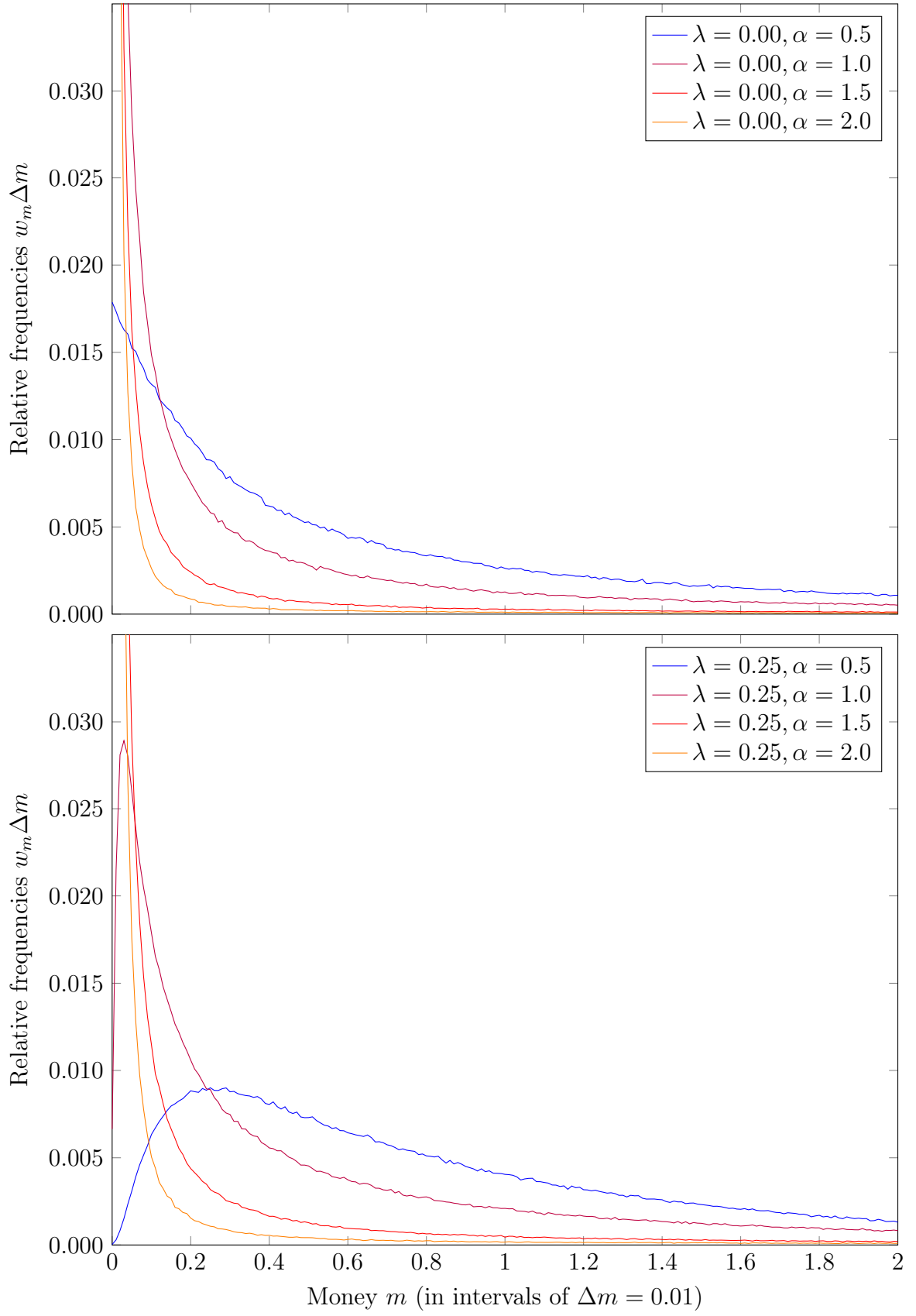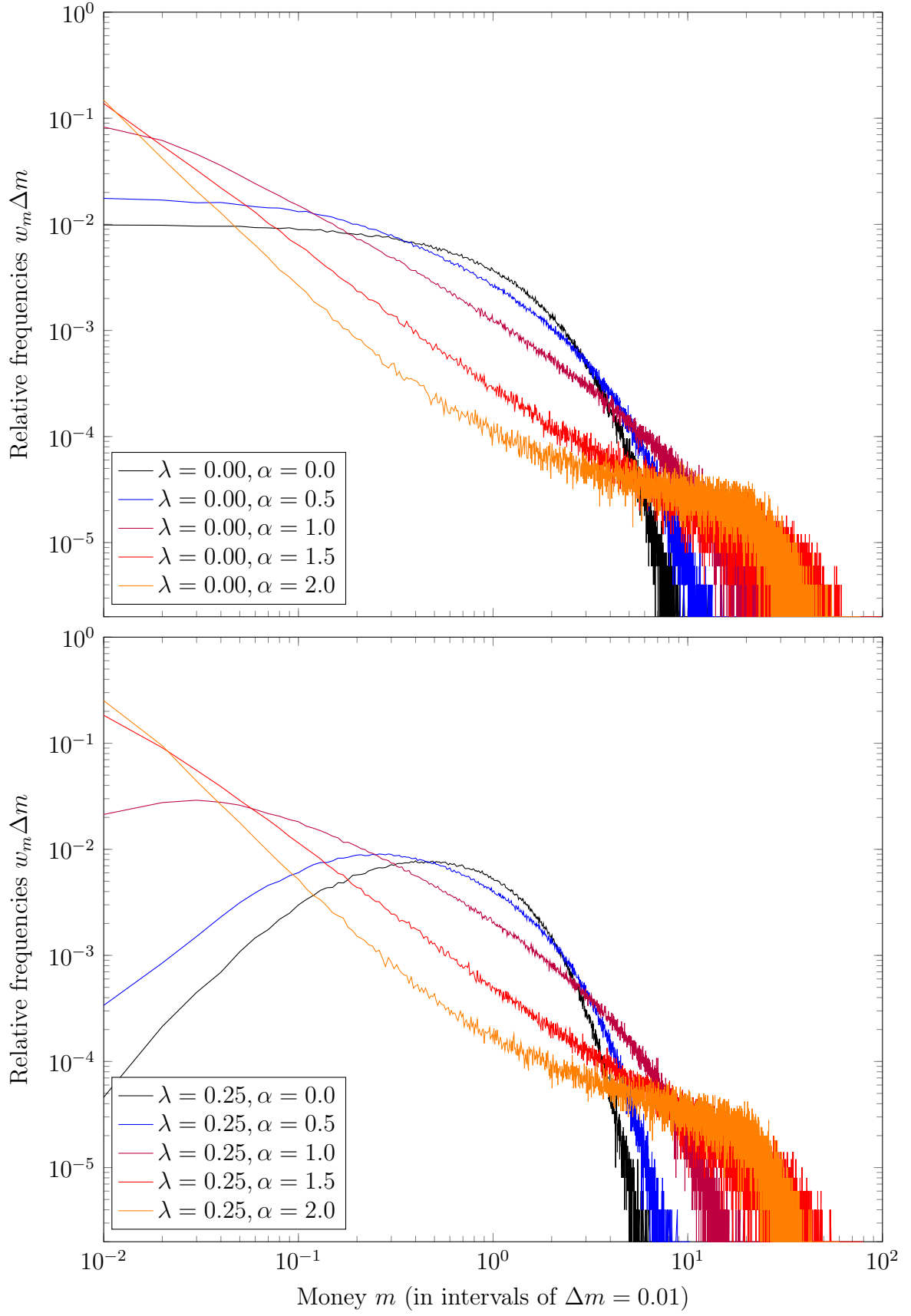
Figure 6: Same as fig. 5, but with $N = 1000$ agents.

Figure 7: Same data as in fig. 5, plotted with logarithmic axes.

4.4 Fourth model: Memory Factor γ

Finally, we introduce the parameter γ that affects the agents preference for previous transaction partners (as described in section 2.3). We run 20 simulations with parameters $\lambda = 0.00, 0.25$, $\alpha = 1.0, 2.0$ and $\gamma = 0.0, 1.0, 2.0, 3.0, 4.0$, with $N = 1000$ agents.

In figs. 8 to 11, wave have again plotted the distributions of these simulations with linear and logarithmic axes. By looking at these figures, we see that increasing the γ parameter "evens out" the distribution again. If we compare this to the distributions of differing α values in fig. 6, we see that γ parameters in effect "negate" the effects of the α parameter.

If we compare the figures without saving (fig. 10) with figures 5 and 6 of Goswami and Sen (2014), we see that for $\alpha = 1.0$ the plots have a similar shape, while for $\alpha = 2.0$ the plots are drastically different. However, in both cases we still don't reproduce the kink that Gowsami and Sen have in their figures.

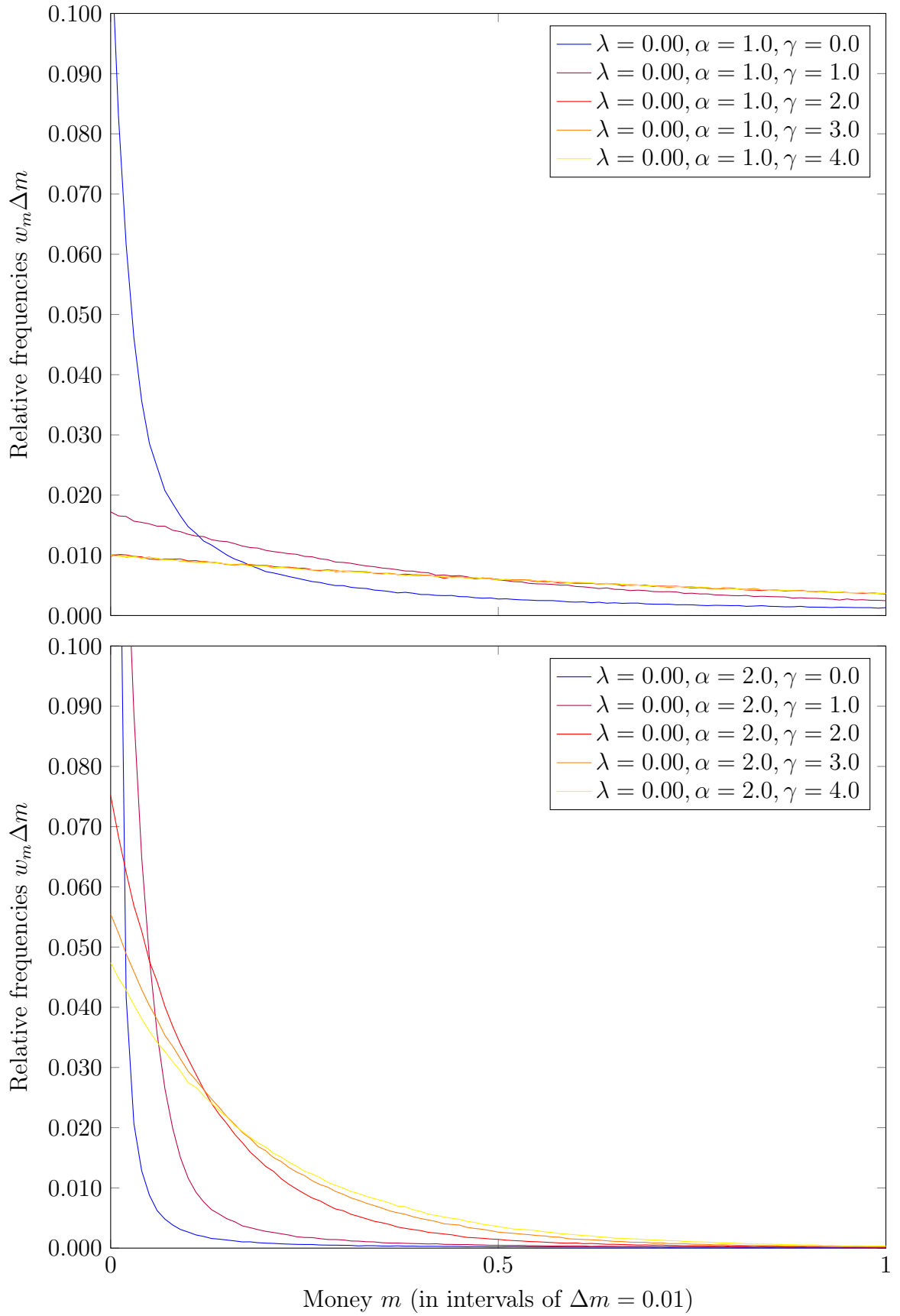For m between approximately 2 and 20, we see that the distribution stays approximately uniform.

Figure 8: Simulated distributions after a "agent preference" parameter γ is introduced into the model, with no savings ($\lambda = 0$). Number of agents is $N = 1000$.
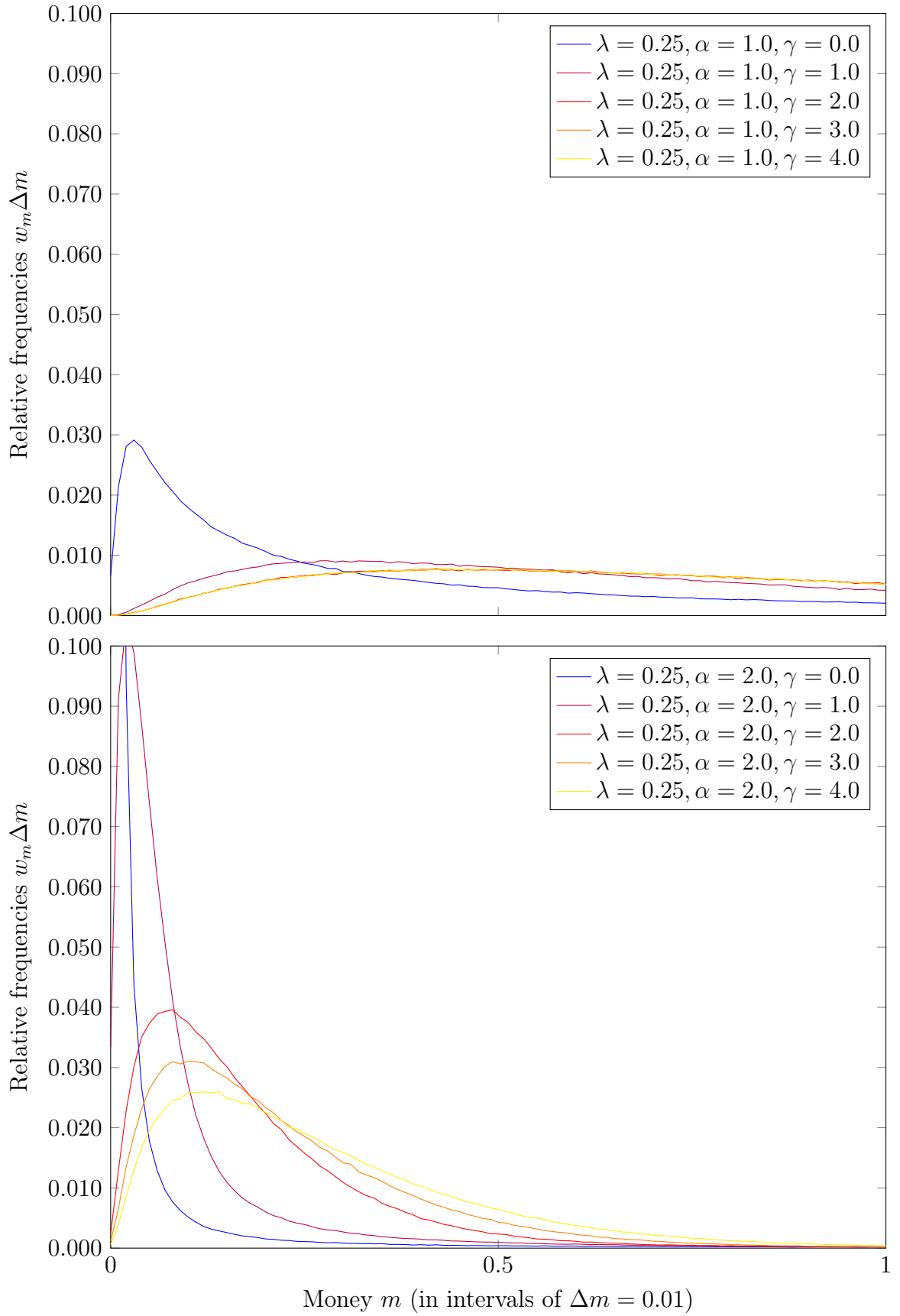
Figure 9: Simulated distributions after a "agent preference" parameter γ is introduced into the model, with savings ($\lambda = 0.25$). Number of agents is $N = 1000$.
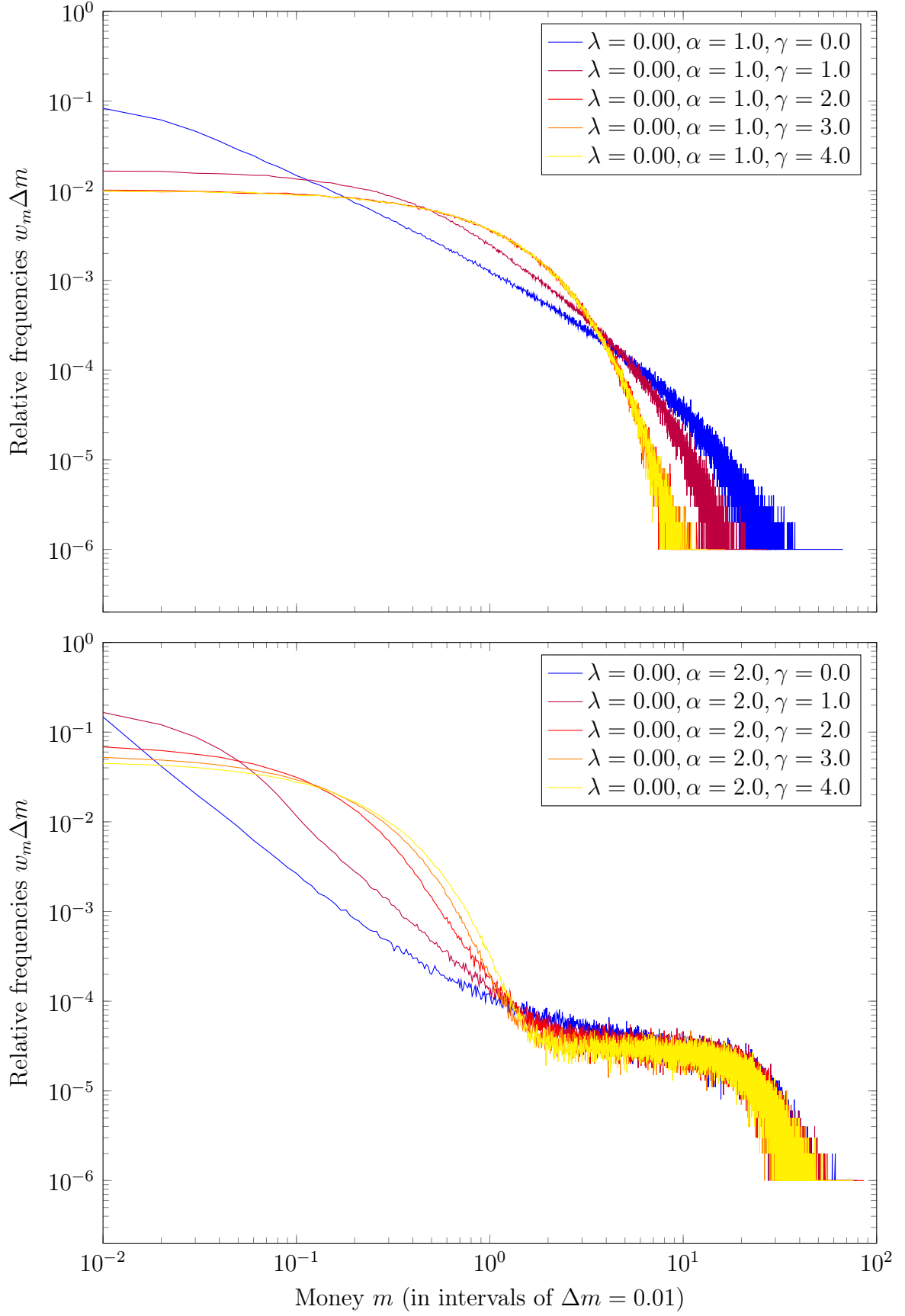
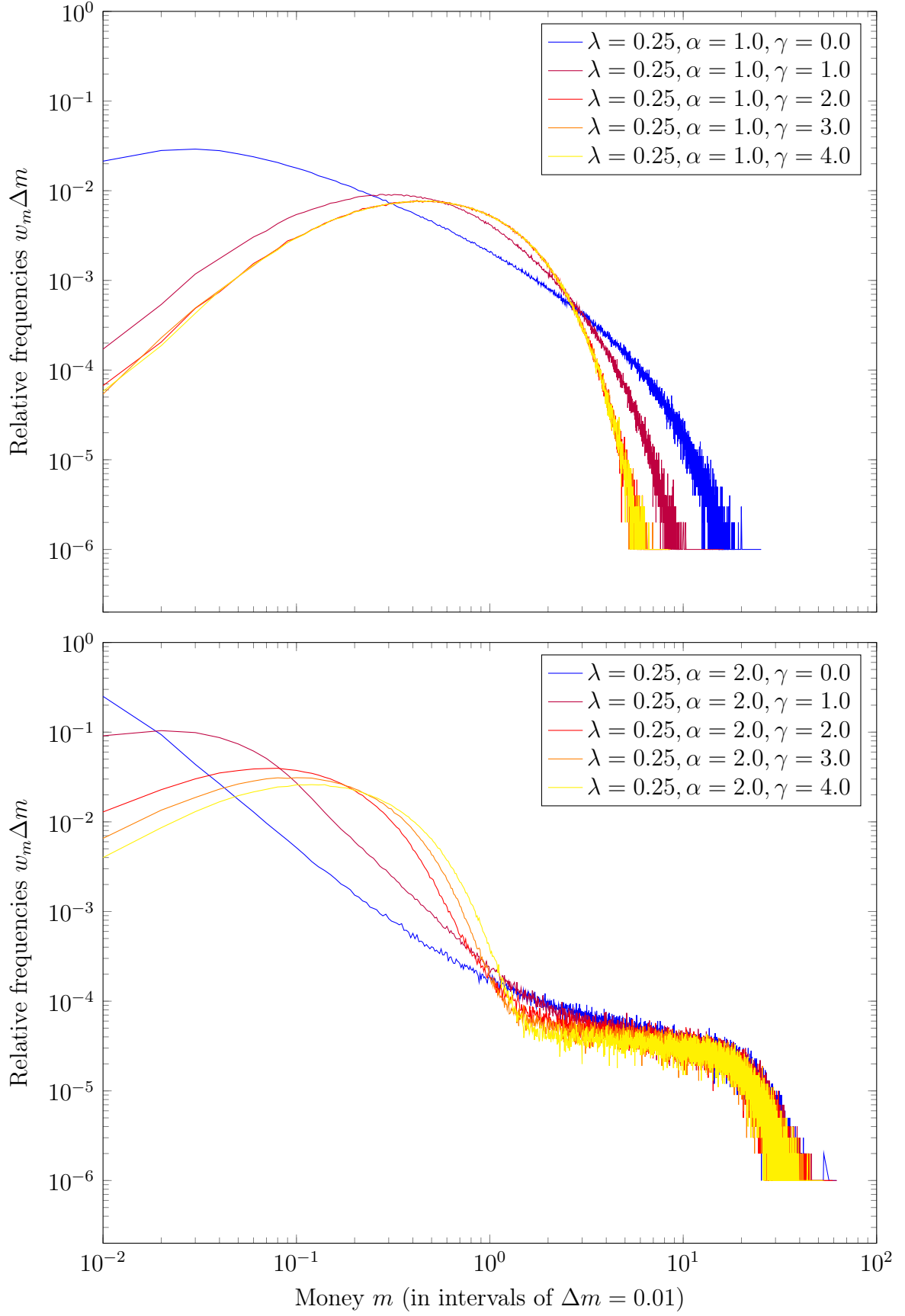Figure 10: Same as fig. 8 (without saving), plotted with logarithmic axes.

Figure 11: Same as fig. 9 (with saving), plotted with logarithmic axes.

5 Conclusion

How realistic our models are can be discussed, but from these models we get a sense of how the different effects affect the distribution of wealth in society. High saving factors λ cause less risk-taking and smaller transactions, which gives us a narrower distribution and less wealth inequality. A high preferential factor α, on the other hand, causes more people to be poor and rich people to be wealthier, which means that there is a higher wealth inequality. Finally, a high memory factor γ means that the agents end up almost only trading with each other in small groups: this causes a limited flow of money and reduces the wealth inequality for most people.

Based on these models, we can tell that in an ideal financial society, agents should trade small amounts of money at a time, with no regards to their transactional partners' wealth, but with a high preference for previous transactional partners. This reduces the income equality the most.

A   Appendix

For this project we both wrote our own code for the simulations, but for all of the results we used the program written by Fredrik.

The code for Fredrik's programs and data files can be found here:
github.com/frxstrem/fys3150/tree/master/project5/code
github.com/frxstrem/fys3150/tree/master/project5/data

The code for Joseph's programs can be found here:
github.com/mathhat/Computational_Physics/tree/master/prosjekt5

B   Bibliography

Goswami, Sanchari and Parongama Sen (2014). "Agent based models for wealth distirbution with preference in interaction". In: *Physica A* 415, pp. 514–524. URL: http://www.sciencedirect.com/science/article/pii/S0378437114006967.

Pareto, Vilfredo (1897). *Cours d'Économie Politique Professé a l'Université de Lausanne*.

Patriarca, Marco, Anirban Chakraborti, and Kimmo Kaski (2004). "Gibbs versus non-Gibbs distributions in money dynamics". In: *Physica A* 340, pp. 334–339. URL: http://www.sciencedirect.com/science/article/pii/S0378437104004327.