
Stochastic Differential Equation Models for Systemic Risk

Author:
Harry Li
CID: 02178113
Email: hxl21@ic.ac.uk

Supervisors:
Professor Grigorios A. Pavliotis
Dr Anastasia Borovykh

September 8, 2023

Declaration

The work contained in this thesis is my own work unless otherwise stated.

Moreover, opinions expressed in this thesis are solely my own and do not express the views or opinions of my employer, the Bank of England.

Acknowledgements

I would like to thank my supervisors at Imperial College London, Professor Grigorios A. Pavliotis and Dr Anastasia Borovykh for their guidance and generosity of time in supporting my dissertation. I have enjoyed working on this thesis and have learnt a lot from working with both of you over the last two years. I would also like to thank my personal tutor Dr Thibault Bertrand for his counsel.

My thanks also go to the Bank of England for their support in this Master's, as well as colleagues for their engagement in useful discussions surrounding my thesis and its applications.

Finally, I would like to thank colleagues, friends and family for their encouragement and support.

Abstract

In this thesis, we use continuous-time coupled stochastic diffusion models to model the interbank market for liquid reserves. The coupled dynamics allow us to capture the interconnectedness of banks in the banking system, which allows us to model feedback and amplification channels between banks in a continuous manner. We study the collective behaviour of banks and systemic risk, which is defined as the risk that many banks fail simultaneously, leading to a failure in the system as a whole. We find that even in our relatively simple model, small changes in the parameters of the model can have a material impact on the probability of a systemic event occurring.

Our contribution to the literature is to analyse the impact of different interbank network structures on systemic risk. We find that the optimal interbank network depends on the composition of the banks in the interbank market. We present a wide range of numerical results to illustrate the impact of model parameters, modelling features and network effects on the stability of the banking system. Finally, we develop deep learning methods for solving more analytically challenging problems, such as an interbank game, where banks can control their rate of borrowing and lending to a central bank, and a network inference problem, where we aim to infer the network structure of the financial system from observing the dynamics of the system. We find that deep learning methods are a promising approach for solving these high-dimensional problems.

Contents

1	Introduction	5
2	Interbank reserve model	8
2.1	Introduction	8
2.2	Common noise	12
2.3	Multiplicative noise	13
3	Bistable systemic risk model	17
3.1	Introduction	17
3.2	Impact of model parameters on systemic risk	19
3.3	Summary	21
4	Network structure model	22
4.1	Introduction	22
4.2	Star network	23
4.3	Erdős-Rényi model (random graphs)	24
4.4	Network homophily and heterophily	25
4.5	Exit system	26
4.6	Summary	30
5	Interbank game	31
5.1	Introduction	31
5.2	Methodology	32
5.3	Implementation	33
5.4	Results	35
5.5	Methodology caveats	36
6	Network inference using neural networks	38
6.1	Introduction to the problem	38
6.2	Methodology	39
6.3	Implementation	40
6.4	Results	42
6.5	Methodology caveats	42
7	Conclusion	44

7.1	Summary	44
7.2	Policy implications	45
7.3	Further work	45
A	Preliminaries	49
A.1	Stochastic processes	49
A.2	Graph theory	51
A.3	Neural networks	51
B	Additional material	56
B.1	Interbank model	56
B.2	Interbank game	56
B.3	Network inference using neural networks	57

Chapter 1

Introduction

Maintaining financial stability is an important objective of any central bank. A stable financial system is able to absorb economic shocks and prevent adverse events from disrupting the economy and spreading to other financial systems. Failures in the financial system can have a very negative impact on the economy and on the welfare of individuals. This was most notable in the 2007-2008 financial crisis, where a shock in the form of a collapsing housing bubble in the US resulted in the collapse of key financial institutions. Since the crisis, central banks have put increasing focus on understanding and reducing systemic risk in the financial system, and in particular, the banking sector. Recent failures in banks like Silicon Valley Bank in 2023 has shown that the banking system is still vulnerable to shocks, and that it is still important to implement policy to safeguard the financial system.

In 2013, Janet Yellen, Vice Chair of the Board of Governors of the Federal Reserve System gave a speech to highlight the importance of interconnectedness and systemic risk. In her speech, Yellen (2013) mentioned that the increased vulnerabilities in the financial system were consequences of the increasing complexity and interconnectedness of aspects of the financial system. She also cited many academic papers which study contagion and systemic risk in networks of interlinked financial institutions such as Cont et al. (2013), who develop a bank-based model for interbank lending and argue that capital requirements should not be uniformly applied across banks, but rather be targeted at systemically important institutions. Yellen's speech suggested that policymakers should carefully consider the role of the interconnectedness of banks when designing policy to safeguard the financial system.

The Bank of England is the central bank of the United Kingdom, and is responsible for maintaining financial stability in the UK. As part of maintaining financial stability, the Bank regularly stress tests UK Banks to assess their resilience to adverse economic shocks. In these stress tests, the Bank calibrates a severe macroeconomic scenario that represents a tail risk to the financial system, which is then run through a suite of models to assess its potential impact on the resilience of the banking system. The Bank uses the results of the stress tests to design policy to safeguard the financial system.

The models used in these stress tests are typically partial equilibrium models applied to individual banks, and the impact of feedback and amplification channels between banks are estimated

separately and layered on top of the initial model results. Feedback and amplification channels describe the impact of a change in the financial position of one bank on the financial position of other banks. For example, if a bank defaults on its obligations to other banks, this can cause other banks to default on their obligations to other banks, which can lead to a cascade of defaults in the financial system. This approach of separately estimating the impact of feedback and amplification channels, rather than incorporating it into the primary models, is mainly due to the complexity of building a granular, general equilibrium stress testing model that can incorporate these effects while modelling how banks' balance sheets evolve over time given a macroeconomic scenario.

We investigate the use of interacting particle models to model the banking system and systemic risk, which is the risk that many banks fail simultaneously, leading to a failure in the system as a whole. Interacting particle systems have seen use in various disciplines, such as the Ising model to model ferromagnetism in statistical physics, the voter model to model opinion formation in social sciences, and the Kuramoto model to model synchronisation. The advantage of using interacting particle systems is that they are simple enough to be tractable, yet complex enough to exhibit interesting dynamics. In general, it is difficult to obtain analytical results for interacting particle systems, so numerical methods are often used to analyse the dynamics of the system. Fortunately, it can be quite straightforward to simulate interacting particle systems to obtain numerical results.

In our model, the dynamics of bank reserves are modelled by continuous-time coupled stochastic differential equations. The coupling in the dynamics allows us to model the interconnectedness of banks, which allows us to model feedback and amplification channels between banks. We also consider the impact of several extensions to the model with policy implications in mind, such as analysing the impact of interbank network structures, aggregate uncertainty, or the inclusion of a central bank into the interbank market on the stability of the banking system.

Additionally, we consider the problem of inferring the network structure of an interacting particle system from observing its dynamics. We develop a deep learning method to estimate the network structure of the financial system, with policy applications in mind. For example, the Bank of England is interested in understanding the role of non-banks in the financial system, such as hedge funds and insurance companies. These companies are not required to report their balance sheets to the Bank, so the Bank has limited information on the financial position of these companies. Our method can help determine the underlying network structure of non-banks in the financial system.

Thesis outline

Our thesis is structured as follows. We conduct a literature review in chapters 2 and 3 of two important models for systemic risk that are presented in Fang et al. (2017) and Garnier et al. (2013). Chapter 2 introduces the simple model of systemic risk used in Fang et al. (2017) and reproduces and summarises the paper's key results. In this chapter, we also consider the impact of including common and multiplicative noise on the stability of the banking system. Chapter 3 reproduces and summarises the model presented in Garnier et al. (2013), which is a bistable model of systemic risk. In this model, banks can be in one of two states; a normal state and a failed state, and the banks transition between the two states. In chapter 4, we extend the simple model from chapter 2 to include different interbank network structures. We analyse the impact of common networks on systemic risk such as the star network and random networks. Additionally, we explore the impact of time-varying networks by providing numerical results on the impact of network homophily and

heterophily, as well as an exit system on the stability of the banking system. In chapter 5, we extend the model in chapter 2 to an interbank game as described in Carmona et al. (2015), where we include a central bank in the interbank market. Banks now control their rate of borrowing and lending to the central bank. We develop a deep learning method to estimate the optimal controls in this interbank game. We use this method to analyse whether the introduction of a central bank increases the stability of the financial system. Chapter 6 describes a network inference problem, where we aim to infer the network structure of the financial system from observing the dynamics of the system. We develop a deep learning method to estimate the network structure of the financial system. Finally, chapter 7 concludes the thesis and discusses the policy implications of our analysis, and future work. Appendix A covers some preliminary mathematical background such as the theory of stochastic processes, graph theory and neural networks. In the appendix, we included a link to our code for our algorithms that use neural networks. Appendix B covers some additional results that were not included in the main body of the thesis.

Chapter 2

Interbank reserve model

2.1 Introduction

We start by considering a simple interbank model of mean-field interacting diffusions to model systemic risk. Let N denote the number of banks and $X_t^i \in \mathbb{R}$ denote the log-monetary reserves of bank i at time $t \in [0, T]$. For $i = 1, \dots, N$, the dynamics of each bank solve the stochastic differential equation given in differential form by:

$$dX_t^i = \mu_i(\bar{X}_t - X_t^i) dt + \sigma_i dW_t^i, \quad X_0^i = 0, \quad t \in [0, T], \quad (2.1)$$

where W_t^i are independent, standard Brownian motions. We define \bar{X}_t to be the empirical mean of the log-monetary reserves in the system at time t , given by:

$$\bar{X}_t := \frac{1}{N} \sum_{i=1}^N X_t^i.$$

Equation (2.1) is one of the simplest continuous-time models for a banking system, yet the model is complex enough such that it exhibits interesting dynamics. The model is a system of N coupled stochastic differential equations, where each bank's dynamics are coupled through the empirical mean. In our model, we use the empirical mean as a measure of the overall health of the financial system. If the empirical mean is high, then the financial system is healthy and if the empirical mean is low, then the financial system is unhealthy. We use the empirical mean to measure systemic risk. If the empirical mean falls below a default level, then we say that the whole banking system has failed.

The dynamics given in eq. (2.1) imply that the monetary reserves for bank i are mean reverting. The economic interpretation of this is that if bank i has more reserves than the empirical mean, then reserves flow from bank i to other banks through interbank lending proportional to the difference $X_t^i - \bar{X}_t$, while if bank i has fewer reserves than the empirical mean, then reserves flow from other banks to bank i . The random Brownian motion term W_t^i represents exogenous shocks to each bank's monetary reserves that occur continuously. The parameter $\mu_i > 0$ represents the strength of the mean reversion, and σ_i controls the volatility for bank i 's level of reserves. The case where $\mu_i = \mu$ and $\sigma_i = \sigma$ corresponds to *homogeneous* case, where all banks have identical coefficients.

Let $\eta < 0$ denote the default level. We interpret the event that $X_t^i \leq \eta$ as bankruptcy for bank i at time $t \in [0, T]$, whereby the bank defaults. In our simple model, we assume that defaulted banks remain in the banking system, and continue to interact with other banks. This is a simplifying assumption, as in reality, defaulted banks are likely to be removed from the banking system. We explore relaxing this assumption in chapter 4.

To measure the stability of the financial system, we use Monte Carlo methods to estimate the *loss distribution*, which is the probability distribution of the number of banks that default within the time interval $[0, T]$ by:

$$\hat{\mathbb{P}}(N_{\text{def}} = n) = \frac{1}{N_{\text{sim}}} \sum_{j=1}^{N_{\text{sim}}} \mathbb{I}_{\{N_{\text{def}}^j = n\}},$$

where N_{def}^j denotes the number of banks that defaulted within the time interval $[0, T]$ for simulation j , and N_{sim} denotes the number of simulations used to estimate the probability distribution.

For a fixed default level η , we define A to be the default event, defined as the event that the banking system fails as a whole, given by:

$$A := \left\{ \min_{0 \leq t \leq T} \frac{1}{N} \sum_i^N X_t^i \leq \eta \right\}.$$

We estimate the probability of the default event using Monte Carlo methods by:

$$\hat{\mathbb{P}}(A) = \frac{1}{N_{\text{sim}}} \sum_{j=1}^{N_{\text{sim}}} \mathbb{I}_{\{A^j\}}.$$

Unless otherwise stated, we estimate loss distributions and the probability of the default event using $N_{\text{sim}} = 5000$ simulations.

Figure 2.1 shows an example of the simulated dynamics of the system. The left graph shows the trajectory of the banks' reserves over the time interval $[0, 1]$, simulated from the dynamics given in eq. (2.1) with $N = 10$ homogeneous banks with $\mu = 1$ and $\sigma = 1$. The red line is the default level $\eta = -0.7$, while the green line is the trajectory of the empirical mean, \bar{X}_t . For this particular simulation, the empirical mean falls the default level, so the banking system has failed. The right graph shows the estimated loss distribution. In the homogeneous case, the distribution is approximately normal, with the modal number of defaults being five banks. The probability of the banking system failing is estimated to be 3%.

Fang et al. (2017) study how the stability of the banking system, as measured by the loss distributions and the probability of the default event occurring are affected by the structure of the banking system. That is, how different combinations of the parameters (N, μ_i, σ_i) affect the stability of the banking system.

2.1.1 Heterogeneous banks - differing μ_i and σ_i

Fang et al. (2017) give examples of how the stability of the banking system can be affected in a large way depending on the structure of the banking system. They consider two banking structures

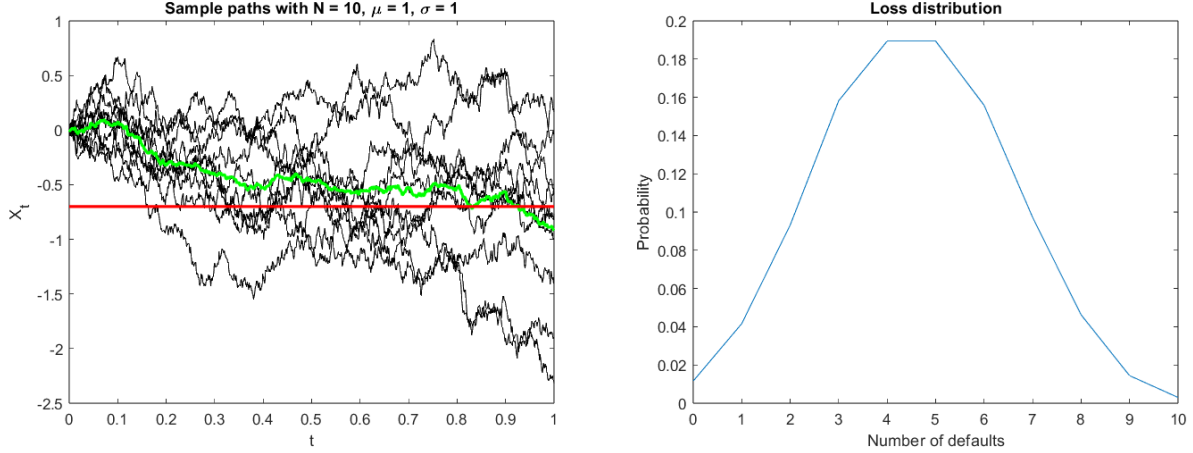


Figure 2.1: The left graph shows the trajectory of banks' reserves for the homogeneous case. The right graph shows the estimated loss distribution.

for the banking system with $N = 10$ banks, given by:

$$\begin{aligned} \text{Group A} : \{(\mu, \sigma)_{\{1,2\}}, (\mu, \sigma)_{\{3,4,5,6,7\}}, (\mu, \sigma)_{\{8,9,10\}}\} &= \{(1, 2), (10, 1), (100, 0.5)\} \\ \text{Group B} : \{(\mu, \sigma)_{\{1,2\}}, (\mu, \sigma)_{\{3,4,5,6,7\}}, (\mu, \sigma)_{\{8,9,10\}}\} &= \{(1, 0.5), (10, 1), (100, 2)\} \end{aligned} \quad (2.2)$$

Group A and B have the same parameters N and μ_i , but differing values for σ_i . In group A, banks with small coefficients for μ_i have large coefficients for σ_i , while in group B, banks with small coefficients for μ_i have small coefficients for σ_i . The ratio of banks with differing values of (μ_i, σ_i) is 2:5:3.

Figures 2.2 and 2.3 show the trajectory of the banks' reserves over the time interval $[0, 1]$ and the estimated loss distributions for group A and B. From the sample paths, we can see that some banks do not closely follow the mean. These correspond to the banks with small coefficients μ_i . For group A, these banks have a large volatility parameter σ , so large fluctuations in their reserves have a large impact on the empirical mean, which affects the reserve levels of other banks. As other banks with larger values for μ_i follow the average level of reserves more closely, these banks with small values for μ but large values for σ have a large impact on the stability of the banking system. The estimated loss distribution for group A is bimodal, with the most likely events being one bank or all ten banks defaulting. We estimate that the probability of the default event occurring is 30%, which is much higher than the homogeneous case. For group B, the banks with small values for μ also have small values for σ . This means that these banks have a smaller impact on the empirical mean and thereby the stability of the banking system. The estimated loss distribution for group B is much more positively skewed, with the modal number of bank defaults being zero. We estimate the probability of the banking system failing to be 2%. This shows that the composition of the banking structure has a large impact on the stability on the financial system.

Fang et al. (2017) also explore the impact of changing the ratio of banks in the groups given in eq. (2.2). They consider three cases, where the ratio of banks within groups A and B given in eq. (2.2) are 8:1:1, 1:8:1 and 1:1:8 instead of 2:5:3. Figures 2.4 and 2.5 show the estimated loss

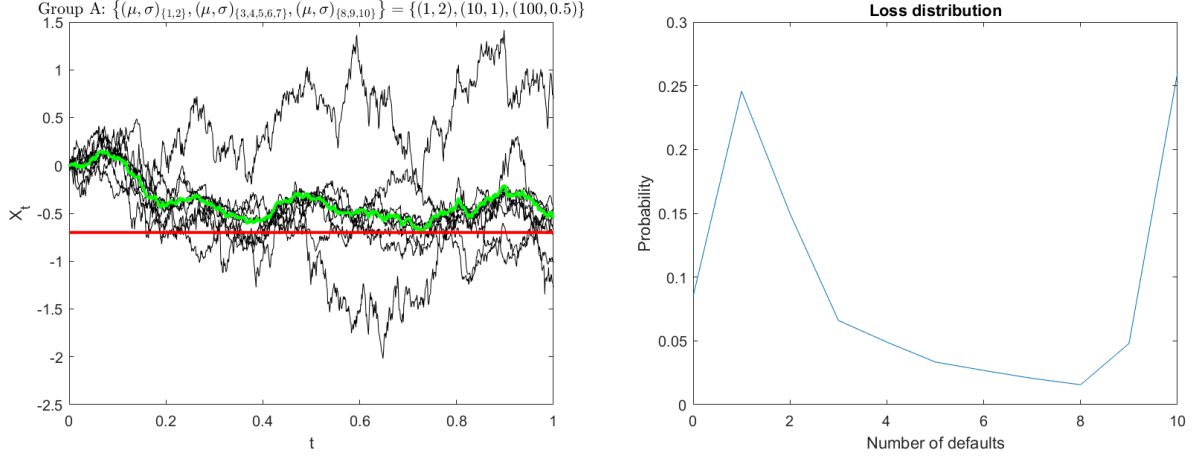


Figure 2.2: The trajectory of banks' reserves and estimated loss distribution for group A.

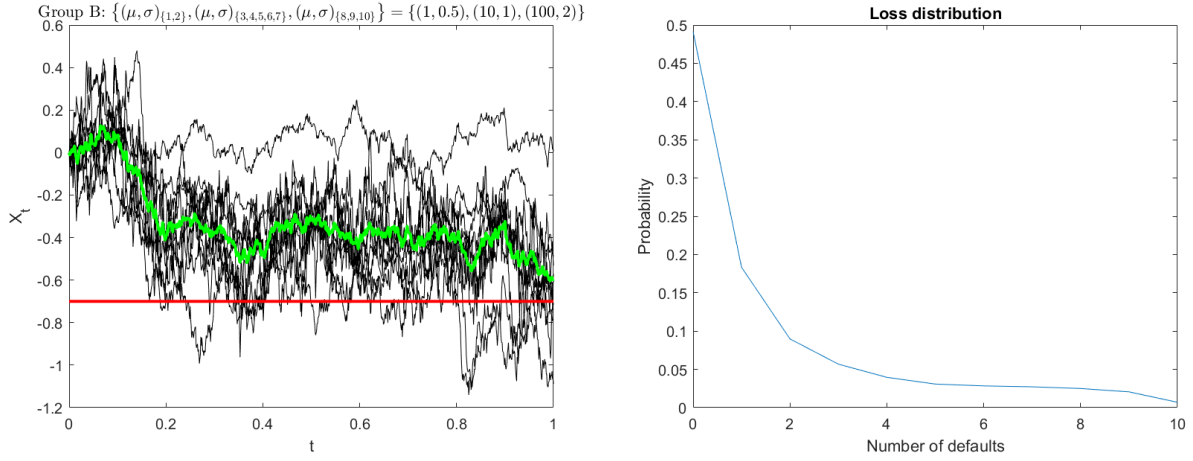


Figure 2.3: The trajectory of banks' reserves and estimated loss distribution for group B.

distributions for these three cases for groups A and B. We see that the estimated loss distributions are most severe when the ratio of agents is 1:1:8, which corresponds to the case where the majority of banks have large coefficients for μ_i . For group A, the probabilities of the banking system failing for the three cases are estimated to be 28%, 24% and 45%, respectively. For group B, the probabilities of the banking system failing for the three cases are estimated to be 0%, 2% and 12%, respectively. The probability of the banking system failing is highest when the number of banks with large coefficients for μ_i is large. These results imply that from a policy perspective, it may be desirable to prevent the formation of groups of larger banks that dominate the interbank market, as these banks have a large impact on the stability of the banking system and can become systemic banks, meaning that their failure can cause the failure of the whole banking system.

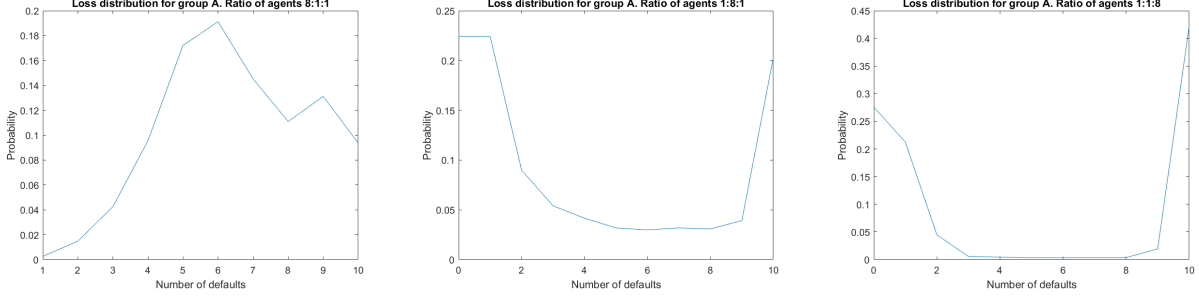


Figure 2.4: Loss distributions for group A when the ratio of banks is 8:1:1, 1:8:1 and 1:1:8.

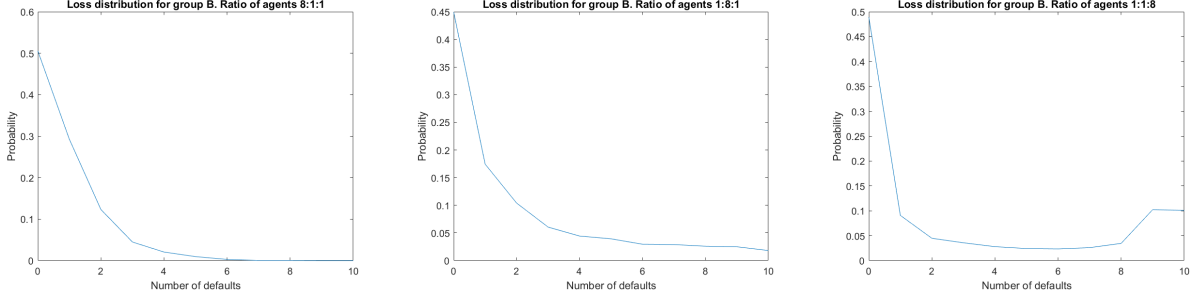


Figure 2.5: Loss distributions for group B when the ratio of banks is 8:1:1, 1:8:1 and 1:1:8.

2.1.2 Asymptotic probability of the default event

Finally, we investigate the probability of the default event occurring as the number of banks increases to infinity. Fang et al. (2017) use the theory of large deviations for Gaussian random variables to derive the asymptotic probability of the default event in the limit as $N \rightarrow \infty$, and show that the probability is of the order of $e^{-\eta^2 N / (2V_T^2)}$, where V_T^2 depends explicitly on μ_i and σ_i . We omit the derivations from Fang et al. (2017) and instead rely on Monte Carlo methods to estimate the asymptotic probability of the default event. We compute the probability of the default event occurring for increasing values of N , from $N = 10$ banks to $N = 300$ banks. For each value of N , we use $N_{\text{sim}} = 500$ simulations to estimate the probability of the default event. Figure 2.6 shows the estimated probabilities of defaulting for group A when the ratio of banks is 8:1:1 and 1:1:8 for different values of N . We see that the probability of the default event occurring decreases as N increases, which is consistent with the results from Fang et al. (2017). This implies that the banking system is more stable when the number of banks participating in the interbank market is high. From a policy perspective, it may then be desirable to ensure that the banking system is competitive with the banking structure consisting of many banks rather than a few banks. We also see that the asymptotic probability of the default event occurring is higher when the ratio of banks is 1:1:8, which is consistent with the results from figs. 2.4 and 2.5.

2.2 Common noise

One interesting extension of the model is to consider what happens when the noise is correlated. We extend the model from eq. (2.1) to include a common source of noise. The dynamics for the

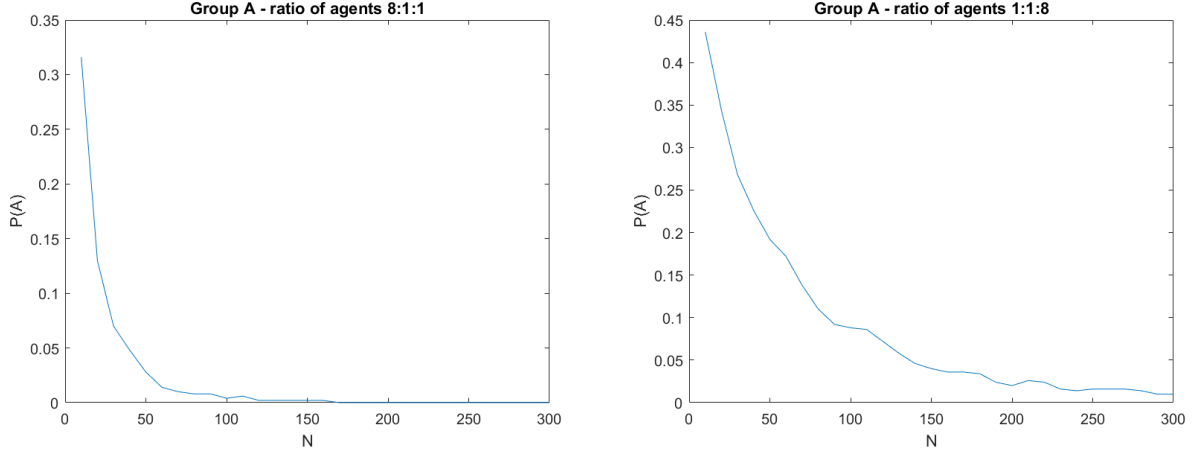


Figure 2.6: Probability of the default event occurring for group A when the ratio of banks is 8:1:1 and 1:1:8, for different values of N .

log-monetary reserve for each bank is given by:

$$dX_t^i = \mu_i(\bar{X}_t - X_t^i) dt + \sigma_i d\widetilde{W}_t^i, \quad X_0^i = 0, \quad t \in [0, T], \quad (2.3)$$

where $d\widetilde{W}_t^i := \rho dW_t^0 + \sqrt{1 - \rho^2} dW_t^i$, with W^i denoting the individual noises, W^0 the common noise and $0 \leq \rho \leq 1$. The parameter ρ describes the noise correlation between the banks. The economic interpretation of common noise is aggregate uncertainty in the interbank market, such as a deterioration in the macroeconomic environment, which can cause all banks to experience a shock to their reserves simultaneously.

We analyse the impact of including common noise on the stability of the banking system by varying the magnitude of the noise correlation coefficient ρ . We consider the homogeneous case with $\mu_i = \mu$ and $\sigma_i = \sigma$ and parameterise the model with $N = 10$ banks, $\mu = 1$ and $\sigma = 1$. Figure 2.7 shows the estimated loss distributions for the model with common noise for $\rho = \{0.2, 0.4, 0.6, 0.8\}$.

We find that increasing the value of ρ increases the probability of the default event occurring. The estimated probabilities of the default event occurring are 5%, 16%, 27% and 39% for $\rho = \{0.2, 0.4, 0.6, 0.8\}$, respectively. As ρ increases, the loss distributions become less normal. For large values of ρ such as $\rho = 0.8$, the loss distribution becomes bimodal, with the most likely events occurring being no banks defaulting or all banks defaulting. This is because when the noise correlation coefficient is high, the trajectory of banks' reserves become more correlated, so banks are more likely to default together, or not at all. These results suggest that including common noise has a large destabilising effect on the banking system. These results are also robust to changes in the parameters N , μ_i and σ_i .

2.3 Multiplicative noise

We consider a model of interbank lending with multiplicative noise similar to Giesecke et al. (2020) and Fouque and Sun (2013). We consider a dynamical system of the form:

$$dX_t^i = \mu(\bar{X}_t - X_t^i) dt + b(t, X_t^i) dW_t^i, \quad X_0^i = x, \quad t \in [0, T], \quad (2.4)$$

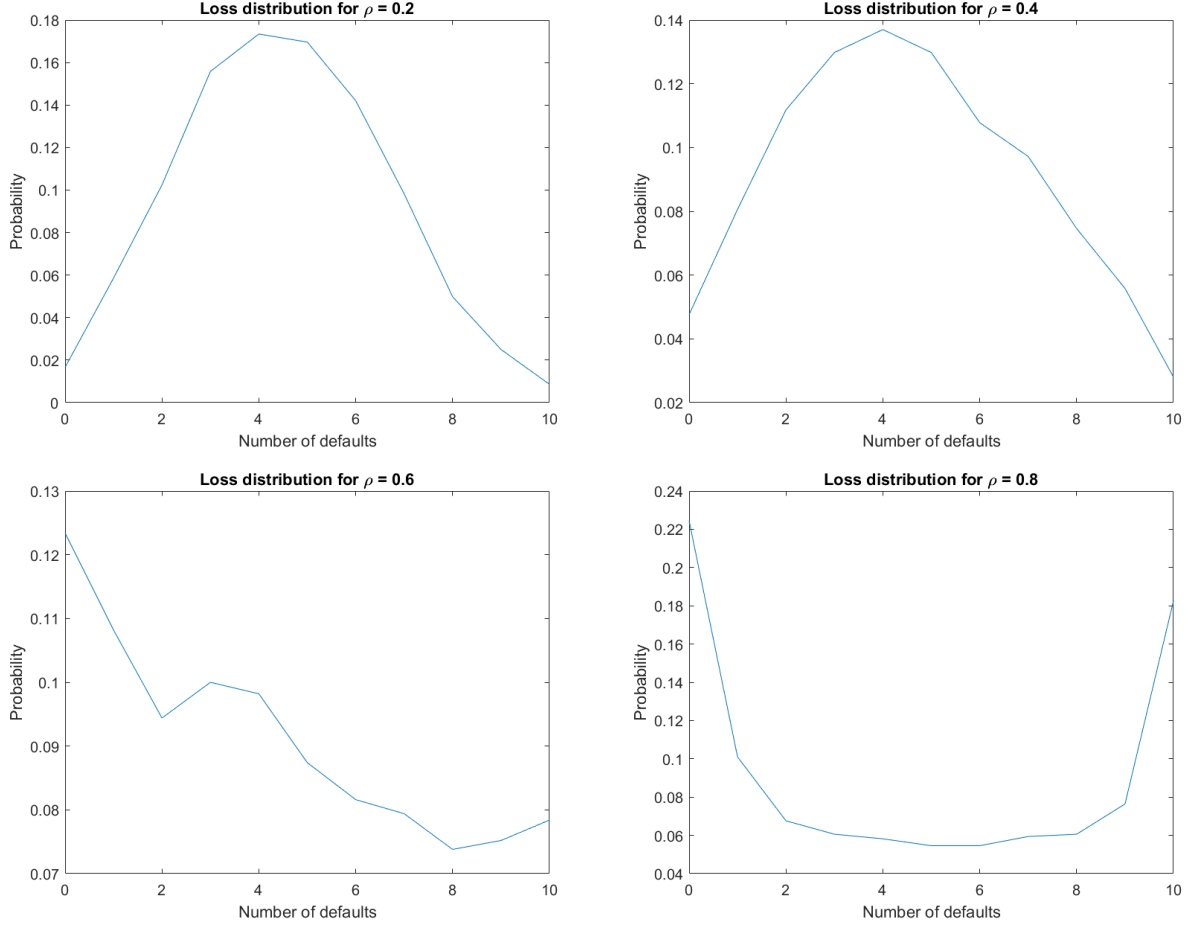


Figure 2.7: Loss distributions for the interbank model with common noise.

for some Borel function $b: [0, T] \times \mathbb{R}^+ \rightarrow \mathbb{R}^+$. In this model, $X_t^i > 0$ denotes the level of reserves of bank i . To start with, we consider the model with $b(t, X_t^i) = \sigma \sqrt{X_t^i}$, and compare the results to the model without multiplicative noise with $b(t, X_t^i) = \sigma$. We parameterise the model by setting $\mu = 1$, $\sigma = 10$, $X_0 = 100$ and we set the default level $\eta = 50$, which can be interpreted as a minimum reserve requirement set by the regulator. Fouque and Sun (2013) prove requirements on the parameters of the model such that the lower bound $X_t^i = 0$ is reached with probability zero over an infinite time horizon. We do not strictly follow these requirements, but set the parameters in such a way such that in practice, the lower bound on the level of reserves $X_t^i = 0$ is never reached when simulating over the finite time horizon $[0, 1]$.

Figure 2.8 shows the sample paths and the estimated loss distributions for the interbank model with and without multiplicative noise. We can see that the sample paths are more volatile for the model with multiplicative noise. Furthermore, the average number of banks that default is higher for the model with multiplicative noise, with the loss distribution for the model with multiplicative noise being approximately normal. The increase in the number of bank defaults is because with multiplicative noise, the noise term is proportional to the level of reserves of each bank, which has the effect of greater noise in the banking system. This increases the volatility of bank reserves,

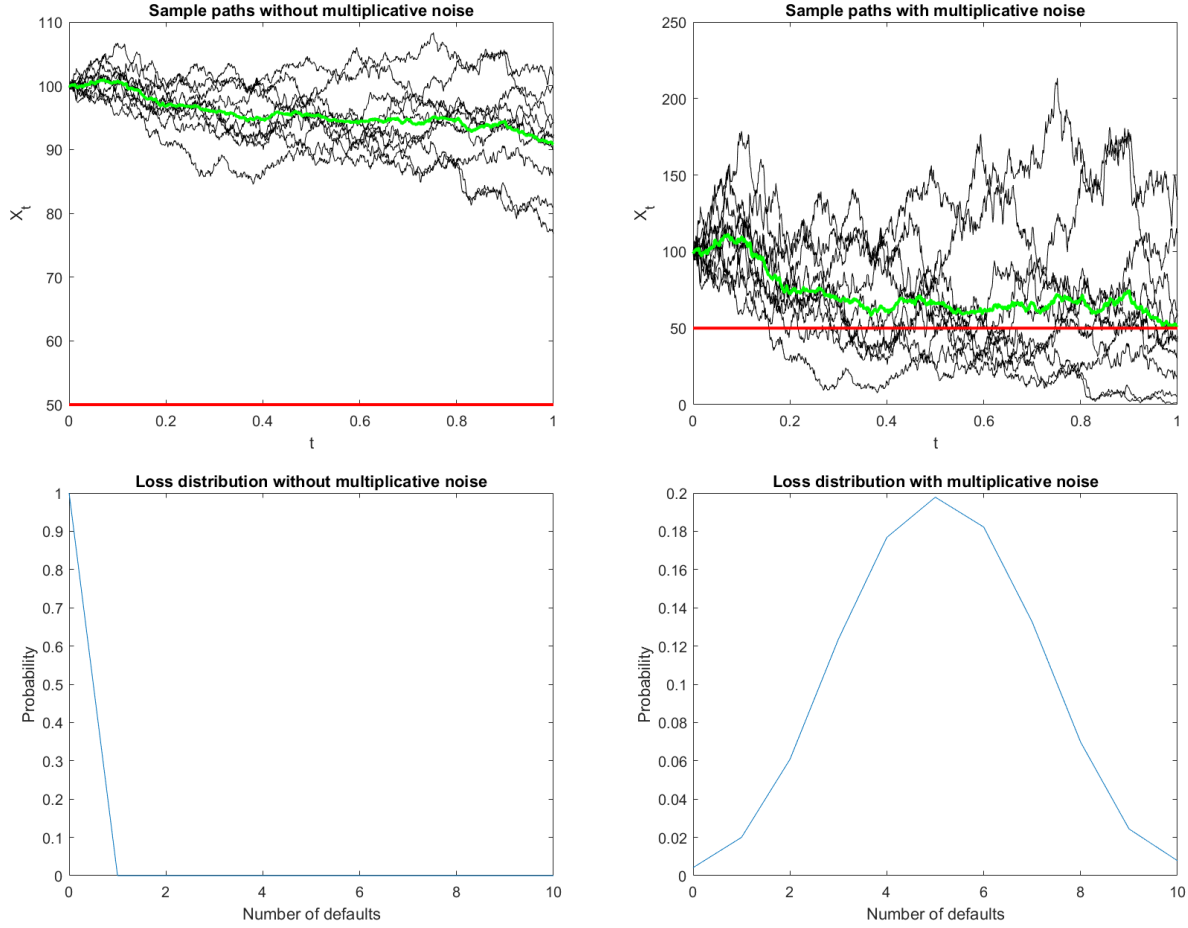


Figure 2.8: Loss distributions for the interbank model with and without multiplicative noise.

thereby making it more likely for banks to default. The estimated probabilities of the default event are 0% for the model without multiplicative noise, and 1% for the model with multiplicative noise. The difference between the probability of the default event is small, so the impact of multiplicative noise on the default event is small. This is because of the asymmetrical impact of multiplicative noise on the level of bank reserves. Banks with higher levels of reserves experience higher levels of noise, and the level of banks' reserves is unbounded from above. This has a positive impact on the empirical mean which counteracts the increased volatility in the banking system, so the probability of the default event occurring is not greatly affected.

Next, we consider the case where we also have multiplicative common noise. We consider a dynamical system of the form:

$$dX_t^i = \mu(\bar{X}_t - X_t^i) dt + \sigma \left(\rho(X_t^i)^\gamma dW_t^0 + \sqrt{1 - \rho^2} \sqrt{X_t^i} dW_t^i \right), \quad X_0^i = 100, \quad t \in [0, T], \quad (2.5)$$

where $0 \leq \rho \leq 1$ is a scalar and $0 \leq \gamma < 1$ is the degree of the multiplicative common noise. The upper bound $\gamma < 1$ is to ensure that X_t^i does not blow up in finite time. We find that varying the degree of the multiplicative common noise parameter can have large implications on the stability

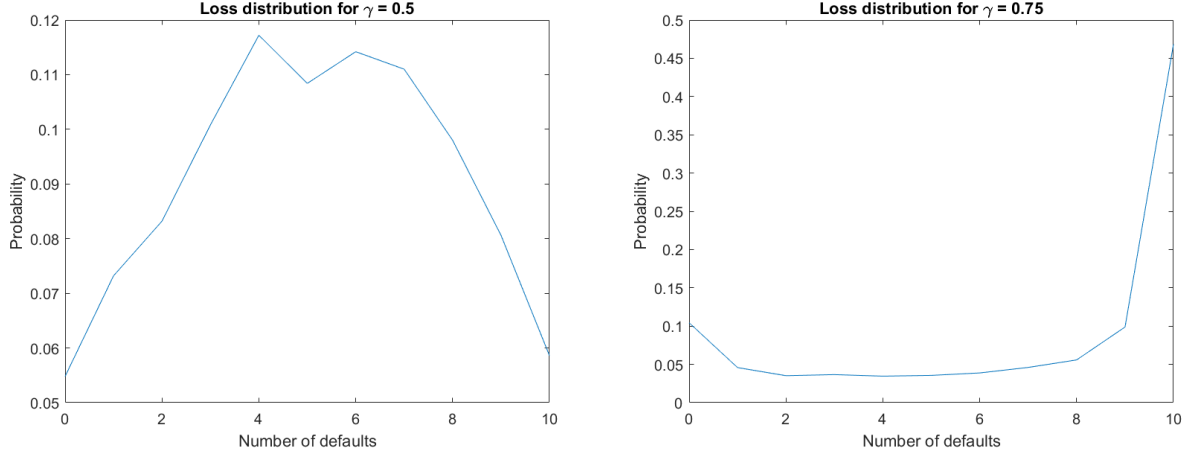


Figure 2.9: Loss distributions for the interbank model with $\gamma = 0.5$ and $\gamma = 0.75$ for the model with multiplicative common noise.

of the banking system. We consider two cases when $\gamma = 0.5$ and $\gamma = 0.75$. The rest of the model is parameterised as before, and we set the common noise parameter $\rho = 0.5$.

Figure 2.9 shows the loss distributions for the model with $\gamma = 0.5$ and $\gamma = 0.75$. We see that the loss distributions are much more severe for the model with the larger degree of multiplicative common noise. The estimated probabilities of the default event are 18% for the model with $\gamma = 0.5$ and 65% for the model with $\gamma = 0.75$. Increasing the degree of multiplicative common noise increases the volatility of the empirical mean. Combining this effect with the increased flocking behaviour of banks due to the common noise, where banks tend to follow the empirical mean more closely, this has the effect of increasing the probability of the default event occurring. These results suggest that the degree of multiplicative common noise can have a large impact on the stability of the banking system.

Chapter 3

Bistable systemic risk model

3.1 Introduction

Garnier et al. (2013) considers a model of interacting banks, where each bank can be in one of two states; a normal and failed, and banks transition between the two states. The model from eq. (2.1) is extended to include a bistable-state structure. Let $X_t^i \in \mathbb{R}$ denote the state of risk of bank i at time t . The dynamics of each bank are assumed to have an intrinsic stabilisation mechanism that keeps banks near a normal state. For $i = 1, \dots, N$, the dynamics of each bank are given by:

$$dX_t^i = (-hU(X_t^i) + \mu_i (\bar{X}_t - X_t^i)) dt + \sigma_i dW_t^i, \quad X_0^i = -1, \quad t \in [0, T], \quad (3.1)$$

The difference between the dynamics in eqs. (2.1) and (3.1) is the inclusion of the restoring force $U(x) = V'(x)$, and the initial condition. V is a potential which is assumed to have two stable states, which are approximately ± 1 , where -1 is the normal state and $+1$ the failed state. Banks are assumed to start in the normal state and can transition between the normal and failed state. The parameter $h > 0$ controls the level of intrinsic stabilisation. The economic interpretation of h is that it represents the cost of implementing risk management to counteract external risk. The parameter $\mu_i > 0$ represents the strength of the mean reversion, which can be interpreted as the rate of diversifying risk through cooperation with other banks. We assume that the cost to banks to reduce their risk by improving their risk management practices by increasing h is much higher than the cost of further diversifying their risk by increasing μ_i . Therefore, we assume that $h < \mu_i$. The noise term σdW_t^i represents exogenous shocks to each bank's risk state that occurs continuously. These shocks can cause banks to transition between the normal and failed state if the size of the shock is large enough.

We assume that $V(x) = \frac{1}{4}x^4 - \frac{1}{2}x^2$, so that V has two stable states at $x^* \pm 1$. Figure 3.1 shows the potential and restoring force. Notice that the restoring force is such that in the neighbourhood of each stable state, $-U(x) > 0$ for $x < x^*$ and $-U(x) < 0$ for $x > x^*$. This means that $x \rightarrow x^*$ in the neighbourhood of x^* , so x^* is a stable state.

We are interested in studying the stability of the banking system through the empirical mean \bar{X}_t , which is taken to be the measure of systemic risk. The parameters of the model are (N, h, μ, σ) . For the homogeneous case where $\mu_i = \mu$ and $\sigma_i = \sigma$, Garnier et al. (2013) find that there exists a

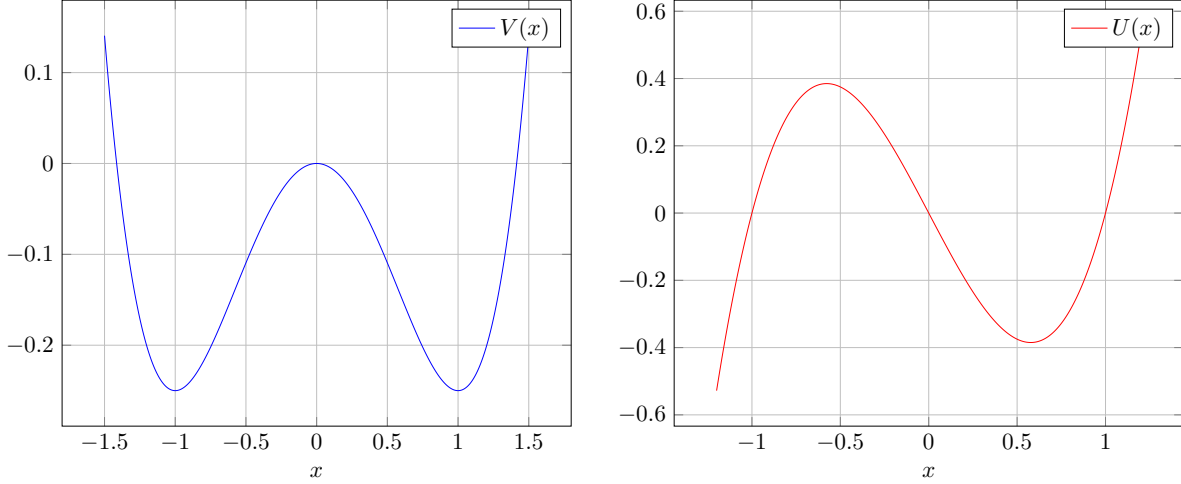


Figure 3.1: The potential $V(x) = \frac{1}{4}x^4 - \frac{1}{2}x^2$ with stable states at $x = \pm 1$ and the restoring force $U(x) = V'(x) = x^3 - x$.

critical value of the volatility parameter, $\sigma_c(h, \mu)$ such that for $\sigma \geq \sigma_c$, the empirical mean has only one stable state at $x^* = 0$. This is because when the volatility parameter is large, the dynamical system in eq. (3.1) behaves like N independent Brownian motions, so half of the banks are in the normal state and half of the banks are in the failed state, so the empirical is around zero. For $\sigma < \sigma_c$, the empirical mean has two stable states at around ± 1 . This is because when the volatility parameter is small, the dynamical system in eq. (3.1) behaves like N coupled Brownian motions, and the empirical mean is attracted to one of the stable states. Garnier et al. (2013) derive an approximation of the critical value of the volatility parameter for small h . The derivations are omitted here and instead, we simply note the existence of a critical value for the volatility parameter, and that the number of stable states depends on the relative value of the volatility parameter and the critical value. As we want to model systemic risk, we focus our attention on the case with two stable equilibrium states in order to analyse the rate of transitions of the empirical mean between the normal and failed states.

Figure 3.2 shows examples of the sample path of the empirical mean for the bistable model with $\sigma < \sigma_c$ and $\sigma \geq \sigma_c$. We parameterise both models $N = 100$ banks, $h = 0.1$ and $\mu = 6$. The dynamics in eq. (3.1) are simulated using the Euler-Maruyama scheme over the time interval $[0, 10000]$ with $\Delta t = 0.02$. The time horizon is longer to allow more time for the empirical mean to fluctuate between the states, as these phase shifts may occur infrequently. The critical value for the volatility parameter in this model is between one and two. For the model with $\sigma = 1$, we have $\sigma < \sigma_c$, so the model has two stable states and the empirical mean oscillates between the normal state and the failed state, as shown by the dashed red lines. The financial system is stable if the empirical mean stays in the normal state and the probability of transitioning to the failed state is low. For the model with $\sigma = 2$, we have $\sigma > \sigma_c$, so the model has one stable state at $x^* = 0$ as indicated by the dashed red line.

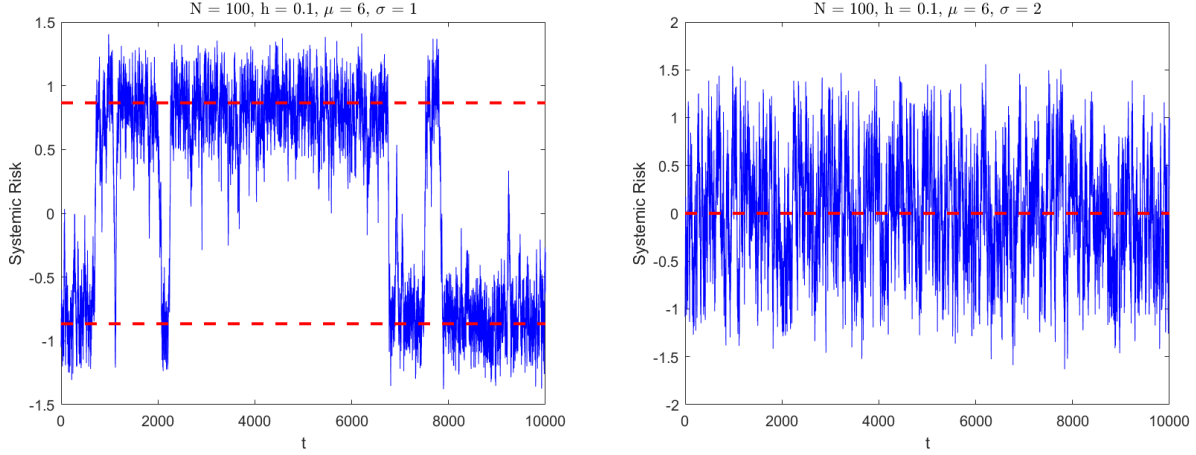


Figure 3.2: Sample paths of the empirical mean for the bistable model with $\sigma < \sigma_c$ and $\sigma \geq \sigma_c$.

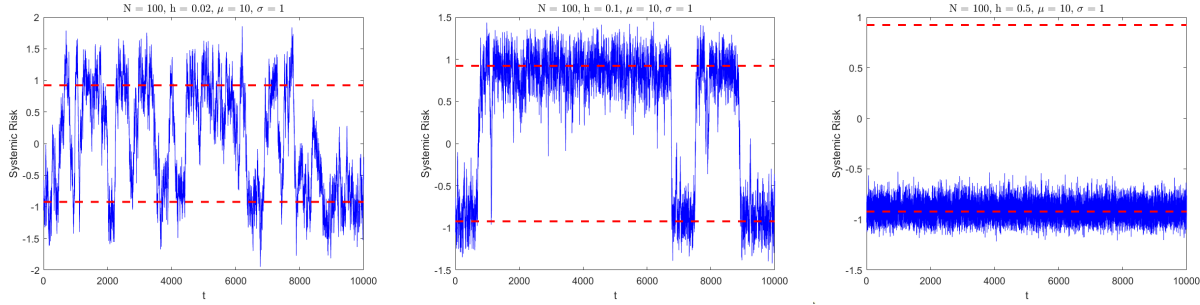


Figure 3.3: Sample paths of the empirical mean for the bistable model for different values of h .

3.2 Impact of model parameters on systemic risk

Garnier et al. (2013) explore the impact of alternative calibrations of the parameters h , μ , σ and N on systemic risk and the stability of the banking system. For the bistable interbank model, we define the banking system to be stable if the empirical mean does not fluctuate much between the normal and failed state. We reproduce and present some of these results and compare them with the model from chapter 2.

3.2.1 Varying h

Figure 3.3 shows the sample paths of the empirical mean for the bistable model for $h = \{0.02, 0.1, 0.5\}$. We can see that increasing h increases the stability of the banking system, as the empirical mean fluctuates less between the normal and failed states. This is intuitive as the parameter h controls the strength of the stabilisation mechanism, so increasing h increases the strength of the stabilisation mechanism, which increases the stability of the banking system. Note that while increasing h increases the stability of the banking system, it also increases the time taken for the empirical mean to transition between the normal and failed states. This means that the impact of a systemic event occurring can have a greater impact on the banking system as it spends more time in the failed state.

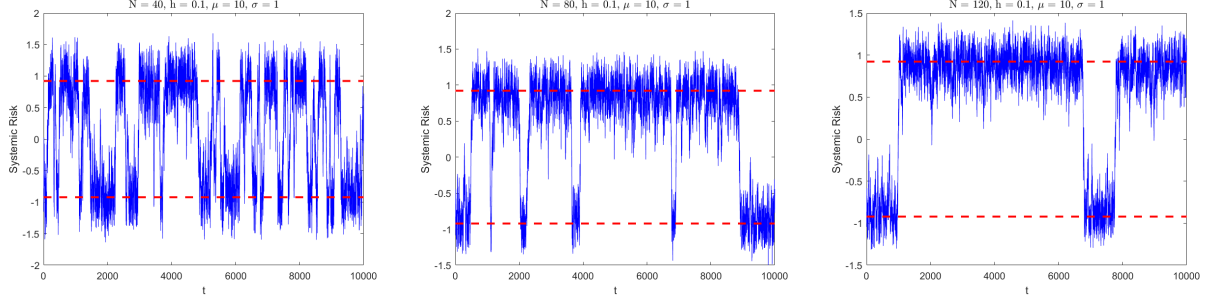


Figure 3.4: Sample paths of the empirical mean for the bistable model for different values of N .

3.2.2 Varying N

Figure 3.4 shows the sample paths of the empirical mean for the bistable model for $N = \{40, 80, 120\}$. Increasing N has a similar effect to increasing h in that it increases the stability of the banking system, as the empirical mean fluctuates less often between the normal and failed states. This is consistent with the results from chapter 2, where we found that increasing the number of banks increases the stability of the banking system.

3.2.3 Increasing diversity

The above results of increasing N or h resulting in an increase in the stability of the banking system also hold for different banking structures, namely banks having different values of the rate of mean-reversion μ_i . We now explore the impact of increasing the diversity of the banking structure. We allow banks to have different preferences on their rate of risk diversification, μ_i . We assume μ_i can take three distinct positive values from $M := \{M_1, M_2, M_3\}$, and we let ρ denote the proportion of banks with different mean-reversion parameters. We consider how increasing the heterogeneity of banks with different μ_i affects the stability of the banking system. Figure 3.5 shows the sample paths of the empirical mean for the bistable model for the three cases:

1. $M = 10$,
2. $M = \{6, 10, 14\}$ with $\rho = \{0.33, 0.34, 0.33\}$,
3. $M = \{2, 10, 18\}$ with $\rho = \{0.33, 0.34, 0.33\}$.

The mean value for μ remains the same for all three cases, but the diversity of cooperation in the banking structure increases as we move from case one to case three. We can see that increasing the diversity of cooperation in the banking structure decreases the stability of the banking system, as the empirical mean fluctuates more often between the normal and failed states.

We also consider how altering the ratio of banks with differing values of μ affects the stability of the banking system. Figure 3.6 shows the sample paths of the empirical mean for the bistable model for $M = \{5, 10, 15\}$, and we consider two cases where $\rho = \{0.1, 0.8, 0.1\}$ and $\rho = \{0.33, 0.34, 0.33\}$. The second case has the more diverse banking structure, but the banking system is less stable as the empirical mean fluctuates more often between the normal and failed states. We conclude that increasing diversity of cooperation in the banking structure decreases its stability. One rationale for this result is that banks with different preferences for risk diversification will diversify their risk

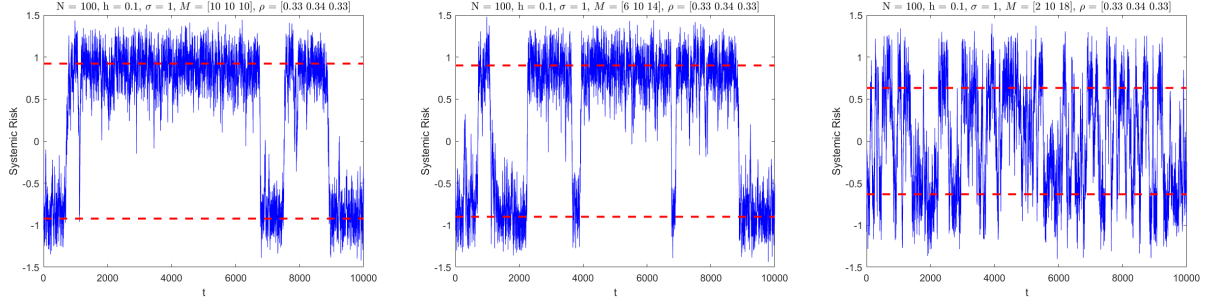


Figure 3.5: Sample paths of empirical mean for the bistable model for different values of M .

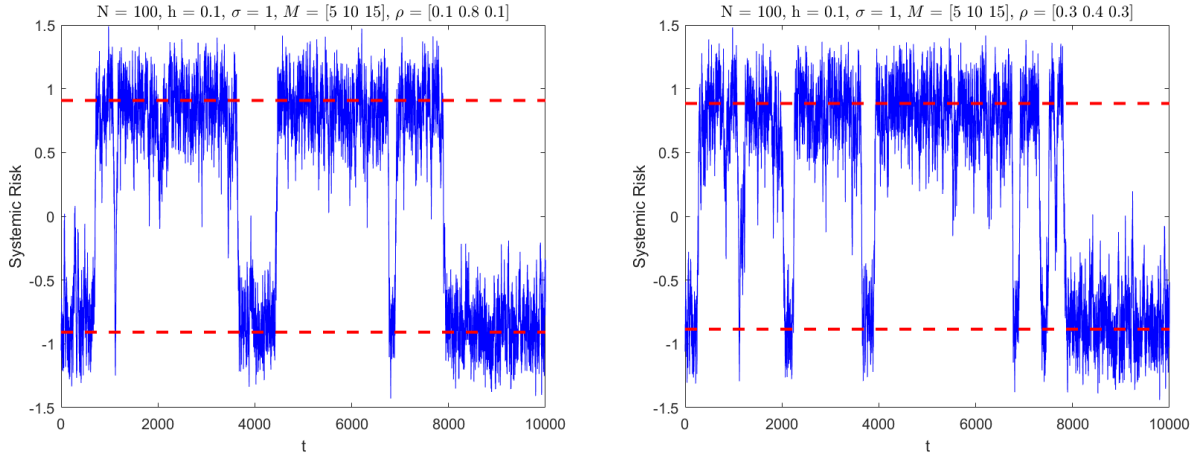


Figure 3.6: Sample paths of empirical mean for the bistable model for different values of ρ .

at different rates, which enhances the fluctuation of the empirical mean, resulting in a less stable banking system.

3.3 Summary

Garnier et al. (2013) considered an interbank model for systemic risk that extends from the model in chapter 2 by including a restoring force, which creates a bistable-state structure. The model is parameterised by (N, h, μ, σ) , where N is the number of banks, h is the strength of the stabilisation mechanism, μ is the rate of risk diversification and σ is the volatility parameter. When σ is small, the model has two stable states at around ± 1 , and the empirical mean is taken to be the measure of systemic risk. The model is stable if the empirical mean stays in the normal state and the probability of transitioning to the failed state is low. Garnier et al. (2013) found that increasing h or N increases the stability of the banking system, while increasing the diversity of cooperation in the banking structure decreases the stability of the banking system.

Chapter 4

Network structure model

4.1 Introduction

In the simple model given by eq. (2.1), the dynamics of the banks implied that we had all-to-all coupling between the banks. That is, each bank interacted with every other bank in the system. We showed that the dynamics of the banks imply that banks exhibit flocking behaviour, where the sample paths for the banks follow closely the mean behaviour of the system. In this chapter, we analyse how different interbank network topologies affect systemic risk. We model these network topologies with an unweighted graph, Γ . Let \mathbf{A} denote the adjacency matrix of Γ , with elements $a_{i,j}(\cdot) \in \{0, 1\}$ denoting the i th row and j th column of \mathbf{A} , for $i, j = 1, \dots, N$. Then, the model from eq. (2.1) can be extended such that each bank's interactions with other banks depends on whether they are connected in the network. The dynamics of each bank are given by:

$$dX_t^i = \frac{\mu_i}{N} \sum_{j=1}^N a_{i,j}(t, X_t)(X_t^j - X_t^i) dt + \sigma_i dW_t^i, \quad X_0^i = 0, \quad t \in [0, T], \quad (4.1)$$

The function $a_{i,j}(t, X_t)$ describes the lending preference from bank j to bank i . Notice that $a_{i,j}$ can be time dependent and depend on the monetary reserve levels of all banks. The deterministic cases where $(a_{i,j})_{1 \leq i, j \leq N} = 1$ corresponds to the model in eq. (2.1) with a complete graph, with all-to-all coupling of the dynamics. The case where $(a_{i,j})_{1 \leq i, j \leq N} = 0$ corresponds to the case where there are no interactions between the banks, and each bank's dynamics follow the path of an independent Brownian motion.

In this thesis, we study the impact on the stability of the banking system by incorporating some common graphs as described in Chiba et al. (2018), such as the star network and random graphs. Figure 4.1 shows an example of the star network, complete graph and a random graph. We also consider some time evolving graphs, such as the impact of network homophily, which is the theory that similar banks may be more likely to interact with each other than dissimilar ones, and network heterophily, which is the opposite of network homophily. Figure 4.2 shows an example of a homophilous and heterophilous graph, where similar banks are colour coded. Finally, we look at the impact of including an exit system in the banking system, where poorly performing banks are removed from the interbank network.

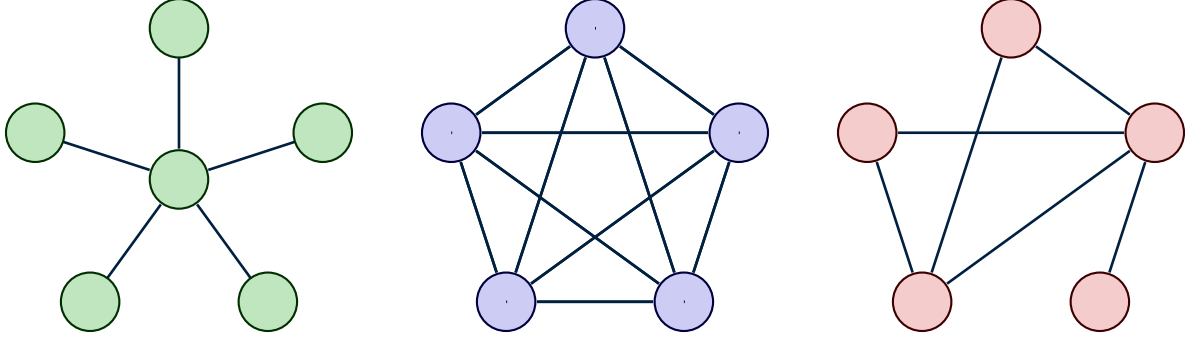


Figure 4.1: Graphs representing the star, complete and a random network topology. A vertex represents a bank, and an edge represents a link between two banks.

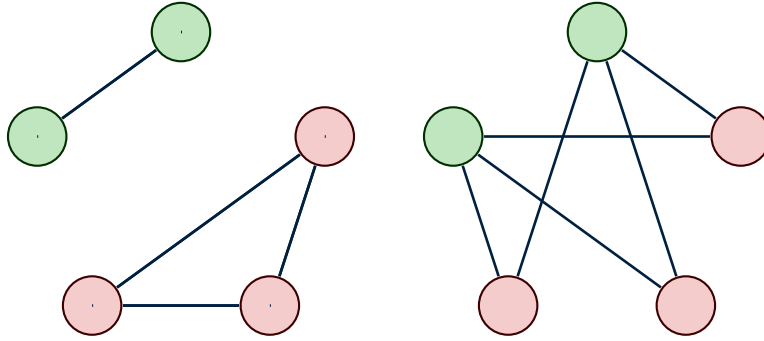


Figure 4.2: Homophilous and heterophilous graphs.

4.2 Star network

In a star network, there is one (or more) bank that is *central*, meaning it is connected to all other banks, and other banks are not connected to each other. Let c denote the index of any central banks. The adjacency matrix for the star network is modelled by:

$$a_{i,j}(\cdot) = \begin{cases} 1, & \text{if } i = c \text{ or } j = c \\ 0, & \text{if } i \neq c \text{ and } j \neq c \end{cases} \quad \text{for } i, j = 1, \dots, N.$$

We consider the banking structures given in eq. (2.2) with $N = 10$ banks and the ratio of banks being 1:8:1. We vary which banks are central according to their coefficients μ_i and σ_i and study the impact of imposing the star topology on the probability of the default event. That is, we take the central banks to be from the set $c \in \{\{1\}, \{2, \dots, 9\}, \{10\}\}$. For group B, we do not find any significant impact of imposing the star topology, with the probability of the default event remaining at 2% in all three cases. For group A, we find that the banking system is most stable when the central bank is the bank with the large coefficient μ_i and small coefficient σ_i . Figure 4.3 shows the estimated loss distributions for group A with the star network for $c \in \{\{1\}, \{2, \dots, 9\}, \{10\}\}$. We see that the loss distributions are least severe when the central bank is the bank with the large coefficient μ_i and small coefficient σ_i , as the right tail of the distribution is the thinnest. The estimated probability of the default event is 26% when $c = \{1\}$, 23% when $c = \{2, \dots, 9\}$ and 9% when $c = \{10\}$. When group A has no coupling in the dynamics, the probability of the

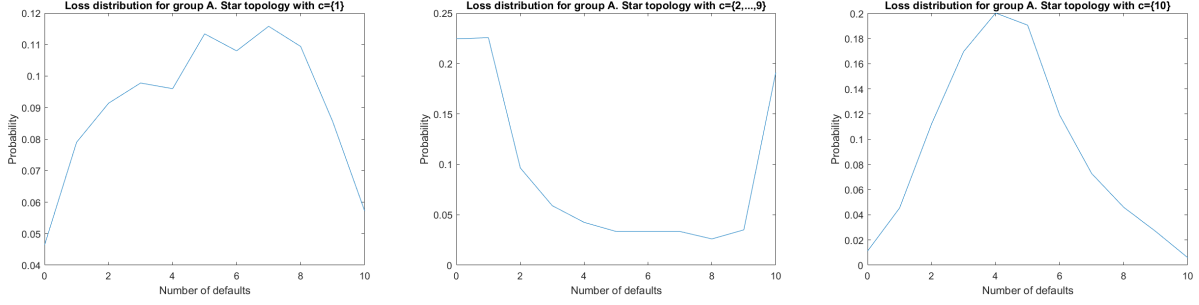


Figure 4.3: Loss distributions for group A with the star network. The ratio of banks is 1:8:1 and the index of the central banks vary.

default event is 5%. The reason why the probability of the default event is highest when the central bank is the bank with the small coefficient μ_i and large coefficient σ_i is because this bank is more likely to default first, and if it is central, it has a large influence on the empirical mean, and when it defaults, it causes other banks to default. When the central bank is the bank with the small coefficient μ_i and large coefficient σ_i , the probability of the default event is lower because there are fewer connections from the problematic bank that has the small coefficient μ_i and large coefficient σ_i to other banks. We conclude that the star network can have a material impact on the stability of the banking system when the banking structure is relatively unstable, similar to the group A structure.

4.3 Erdős-Rényi model (random graphs)

In a random graph, $(a_{i,j})_{1 \leq i,j \leq N}$ is a random matrix, which we still assumed to be symmetric. We denote the probability of an edge between any two banks by a constant p . The adjacency matrix for our random graphs are defined by:

$$a_{i,j}(\cdot) = \begin{cases} 1, & \text{with probability } p \\ 0, & \text{with probability } 1 - p \end{cases} \quad \text{for } i, j = 1, \dots, N \text{ with } i < j, \text{ and } p \in [0, 1].$$

For $i > j$, we assume $a_{i,j} = a_{j,i}$, so the adjacency matrix is symmetric. Note that when $i = j$, the dynamics in eq. (4.1) do not depend on $a_{i,i}$, so we can assume that it takes any value.

We analyse the impact of random graphs generated using the Erdős-Rényi model on the stability of the banking system. Figure 4.4 shows the loss distributions when the adjacency matrix is generated using the Erdős-Rényi model with $p = 0.25$, when the banking structure is given by groups A and B as in eq. (2.2). In comparison to the loss distributions in figs. 2.2 and 2.3, we find that the loss distributions for the random graphs are more normally distributed. The probability of the default event occurring decreases materially for group A, while the impact for group B is minimal, with the probability of the default event increasing marginally from 1% to 2%.

This is because the Erdős-Rényi model is a random graph where the probability of an edge between any two banks is $p = 0.25$, so the adjacency matrix is sparse. As a result, the network structure is similar to the case when there is no coupling in the dynamics and banks' reserves follow an independent Brownian path. For group A, we previously identified that the banks with a small

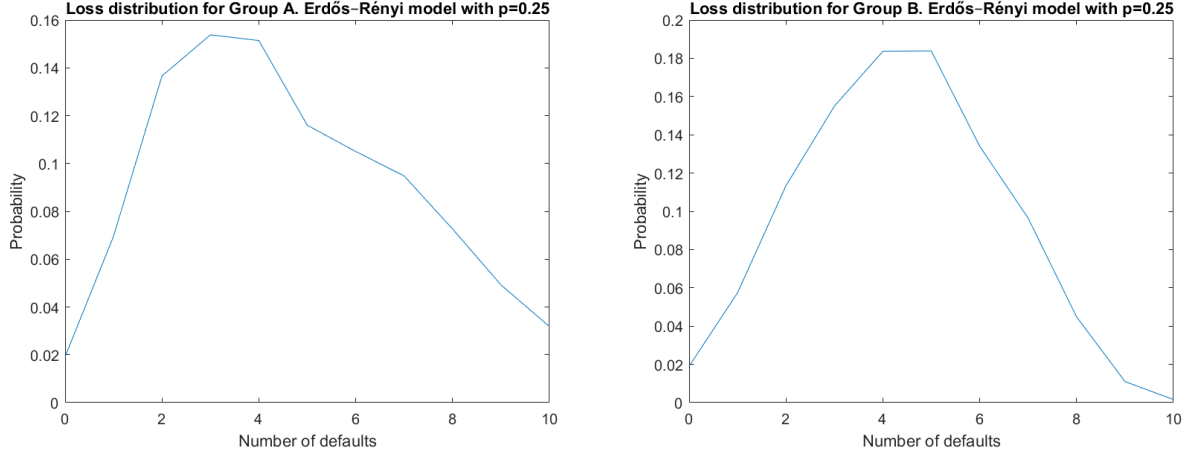


Figure 4.4: Loss distributions for the Erdős-Rényi models with $p = 0.25$ for groups A and B.

coefficient μ_i and a large coefficient σ_i play an important role in the stability of the banking system, as these banks have a large influence on the empirical mean and them defaulting often caused other banks to default. In the Erdős-Rényi model, the banks with a small coefficient μ_i and a large coefficient σ_i are less likely to be connected to other banks, so they have a smaller impact on the stability of the banking system. The banking structure in group B was already stable, so the impact of altering the network structure by using random graphs is minimal, and may even have a slight adverse impact on the stability of the banking system by reducing the number of connections between banks, making them more likely to revert away from the empirical mean and default.

4.4 Network homophily and heterophily

Network homophily refers to the theory that similar banks may be more likely to interact with each other than dissimilar ones. Network heterophily is the opposite of network homophily and refers to the theory that dissimilar banks may be more likely to interact with each other than similar banks. Network homophily and heterophily can be important in the study of social networks, as it can be used to explain the formation of social ties. For example, Motsch and Tadmor (2014) study a class of models for self-organised dynamics based on alignment and find that heterophilous dynamics rather than homophilous dynamics enhance consensus. We study the impact of network homophily and heterophily on the stability of the banking system.

We first consider the banking structures given by groups A and B in eq. (2.2), and define banks to be similar if they have the same coefficients for μ_i and σ_i . We then define the adjacency matrix for the homophilous and heterophilous graphs as:

$$\begin{aligned} \text{Homophily: } a_{i,j} &= \begin{cases} 1, & \text{if } \mu_i = \mu_j \text{ and } \sigma_i = \sigma_j \\ 0, & \text{else} \end{cases} \\ \text{Heterophily: } a_{i,j} &= \begin{cases} 0, & \text{if } \mu_i = \mu_j \text{ and } \sigma_i = \sigma_j \\ 1, & \text{else} \end{cases} \end{aligned}$$

Figure 4.5 shows the loss distributions for the homophilous and heterophilous graphs for groups A and B. We find that for group A, the loss distribution is less severe in the homophilous case as opposed to the heterophilous case. The estimated probability of the default event occurring is 5% for the homophilous case and 32% for the heterophilous case. Network homophily is preferred for group A because the banks with small coefficients μ_i and large coefficients σ_i are problematic banks that have a large impact on the empirical mean and are more likely to default first due to having a large σ , which can result in a cascade of other banks defaulting. In the homophilous case, these banks are not connected to other banks, so they have a smaller impact on the stability of the banking system. On the other hand, for group B, the loss distribution is less severe in the heterophilous case compared to the homophilous case. The estimated probability of the default event occurring is 9% for the homophilous case and 2% for the heterophilous case. Network heterophily is preferred for group B because when the banking system is inherently stable, increased interbank lending between banks means that the dynamics of the banks are more coupled, so banks' reserves are more likely to stay close to the empirical mean, which increases the stability of the banking system. This example shows that network homophily might be preferred when the banking system is inherently unstable, while network heterophily might be preferred when the banking system is inherently stable.

Alternatively, we could define banks to be similar if they have similar monetary reserve levels. We define the adjacency matrix for the homophilous graphs as:

$$a_{i,j}(t, X_t) = \begin{cases} 1, & \text{if } |X_t^i - X_t^j| \leq \tau \\ 0, & \text{if } |X_t^i - X_t^j| > \tau \end{cases} \quad \text{for } i, j = 1, \dots, N,$$

where τ denotes the radius of interactions, defined as the maximum difference in reserves allowed between X_t^i and X_t^j for them to continue interacting with each other.

Similarly, we model adjacency matrix for the heterophilous graphs as:

$$a_{i,j}(t, X_t) = \begin{cases} 1, & \text{if } |X_t^i - X_t^j| > \tau \\ 0, & \text{if } |X_t^i - X_t^j| \leq \tau \end{cases} \quad \text{for } i, j = 1, \dots, N,$$

where τ is now defined as the minimum difference in reserves allowed between X_t^i and X_t^j for them to continue interacting with each other.

We parameterise the model as in chapter 2 with $N = 10$ banks with $\mu = 1$, $\sigma = 1$, $\eta = -0.7$ and take $\tau = 0.5$. Figure 4.6 shows the loss distributions for the homophilous and heterophilous graphs for groups A and B under this new definition. We find similar results to the case when banks are similar if they have the same coefficients for μ_i and σ_i . For group A, the probability of the default event occurring is 6% for the homophilous case and 32% for the heterophilous case. For group B, the probability of the default event occurring is 9% for the homophilous case and 2% for the heterophilous case. The results are robust to the definition of similarity between banks, as banks with similar coefficients μ_i and σ_i are also likely to have similar monetary reserve levels.

4.5 Exit system

Until now, the models we have considered assumed that defaulted banks remain in the banking system and continue to interact with other banks through the empirical mean. However, in reality,

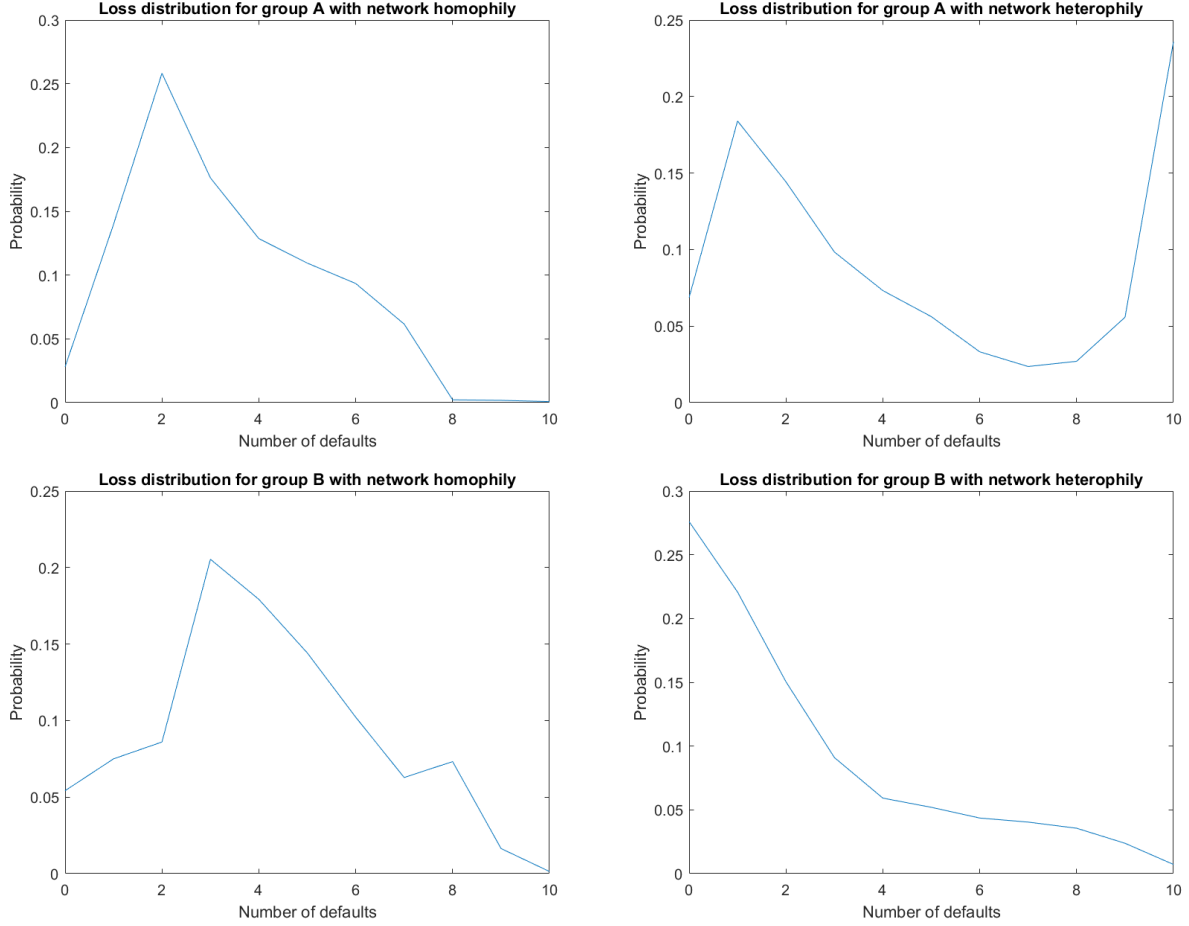


Figure 4.5: Loss distributions for the homophilous and heterophilous graphs for groups A and B. Banks are similar if they have the same coefficients for μ_i and σ_i .

banks that have defaulted, or are illiquid, may be removed from the banking system and no longer borrow or lend on the interbank market. We model this by introducing an exit system, where banks that hold monetary reserves less than a default threshold are removed from the interbank network. This means that the dynamics of an exited bank are given by an independent Brownian motion: $dX_t^i = \sigma dW_t^i$. We allow a bank to recover from default if their monetary reserves exceed the default threshold and recovered banks can borrow and lend in the interbank market as normal. We call bank i ‘weak’ at time t if $X_t^i \leq \eta$ and we remove it from the interbank network by setting $a_{i,j}(t, X_t) = a_{j,i}(t, X_t) = 0$ for $j = 1, \dots, N$.

The effect of removing weak banks from the interbank network on systemic risk is ambiguous. On one hand, removing weak banks from the interbank network may have a stabilising effect on the banking system, as banks with a low amount of monetary reserves borrow from other banks, reducing other banks’ reserve levels, which can cause them to default. This is known as the contagion channel, where one banking failure can cause other banks to fail. On the other hand, a weak bank is less likely to recover if it is removed from the interbank network, as it cannot borrow from other banks. This can have a destabilising effect on the banking system, as weak banks are

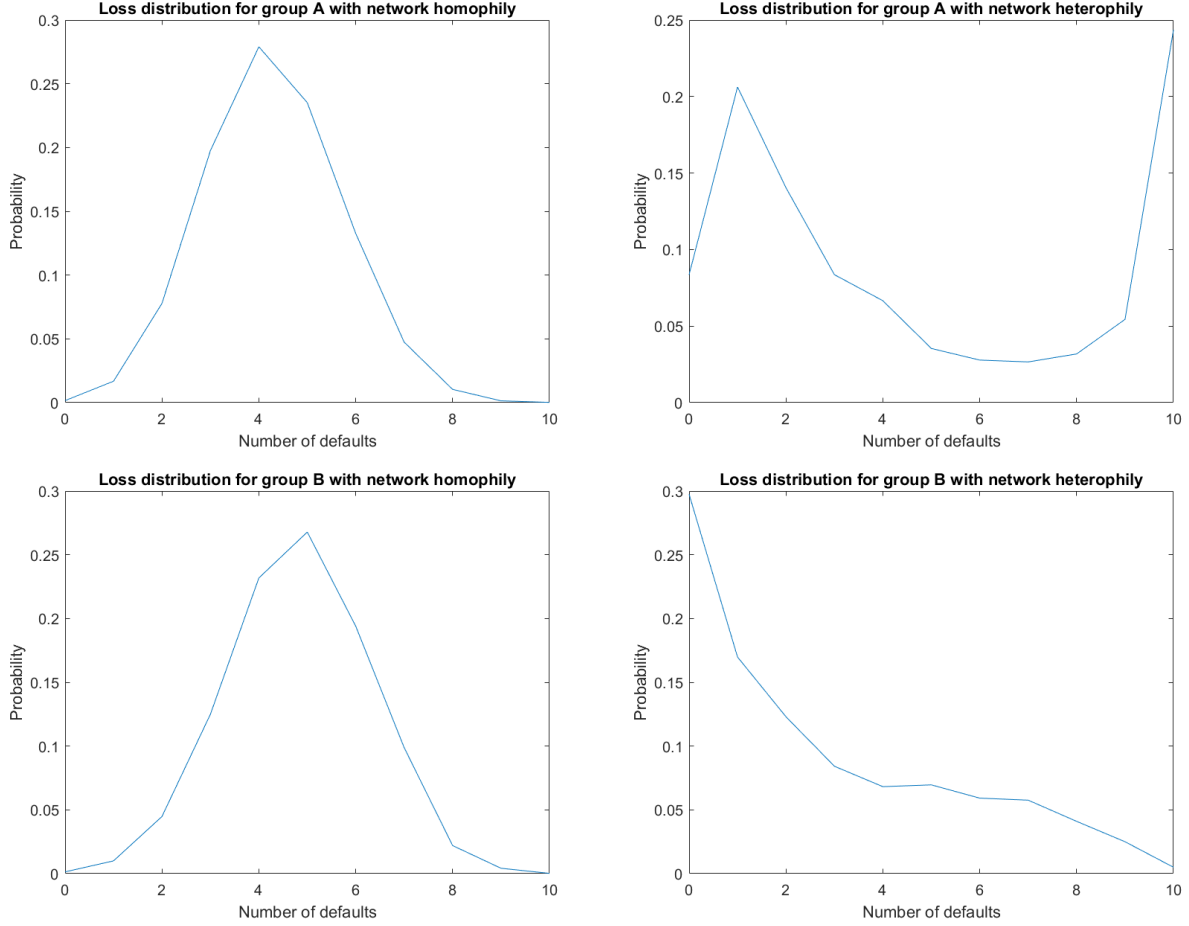


Figure 4.6: Loss distributions for the homophilous and heterophilous graphs for groups A and B. Banks are similar if they have similar monetary reserve levels.

more likely to stay defaulted, which has a negative impact on the empirical mean. We explore the impact of removing weak banks from the interbank network on systemic risk.

We first study the homogeneous case when all banks have identical coefficients $\mu_i = \mu$ and $\sigma_i = \sigma$. We parameterise the model as in chapter 2 with $N = 10$ banks with $\mu = 1$, $\sigma = 1$ and $\eta = -0.7$. Figure 4.7 shows the loss distributions for the model with and without the exit process. We find that the loss distribution with the exit process is similar to the loss distribution without the exit process. Furthermore, the probability of the default event remains unchanged at 3%. For this homogeneous banking system, we find that removing weak banks from the interbank network has little impact on the stability of the banking system, as the probability of the default event occurring remains unchanged. This is because when weak banks are removed from the interbank network, there is a trade-off between reducing the likelihood of other banks failing through the contagion channel, but it also reduces the likelihood of weak banks recovering. For this homogeneous banking system, both competing channels are roughly balanced, so there is not a significant change in the stability of the banking system.

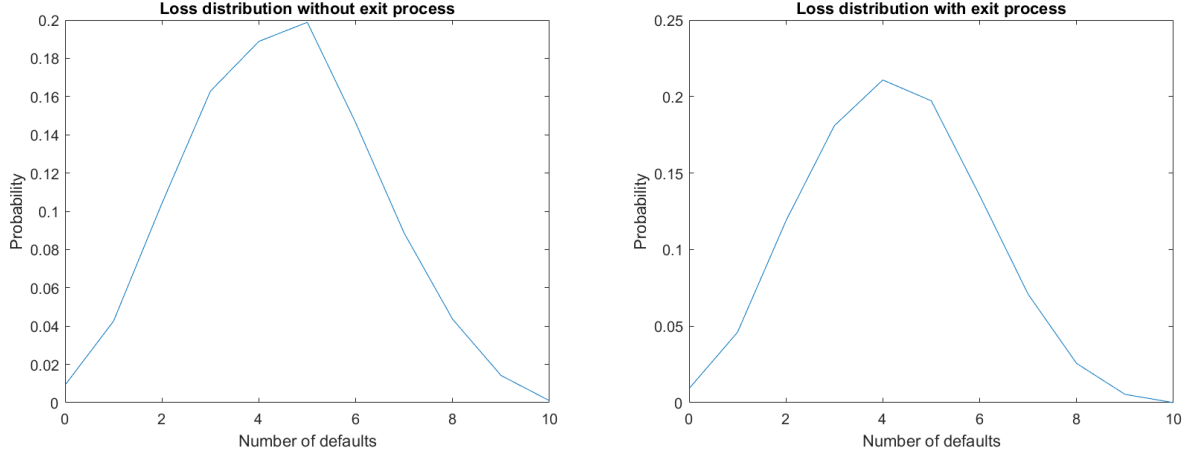


Figure 4.7: Loss distributions with and without the exit process.

Next, we study the case when the banking system is heterogeneous. We consider the banking structure given by group A as given in eq. (2.2). Figure 4.8 shows the loss distributions for group A when the ratio of banks is 8:1:1, 1:8:1 and 1:1:8, and weak banks are removed from the interbank network. We compare these results with the model without the exit process in fig. 2.4. The loss distributions with the exit process are more positively skewed than the loss distributions without the exit process, with the probability of all banks defaulting dropping to zero. This is because the exit process removes weak banks from the interbank network, which reduces the likelihood of other banks failing through the contagion channel. Additionally, we estimate the probabilities of the default event occurring for group A for the three cases on the ratio of banks with the exit process to be 17%, 5% and 8%, respectively. These probabilities are much lower than the model without the exit process, when the probabilities estimated were 28%, 24% and 45%, respectively. For this heterogeneous banking system, we find that removing weak banks from the interbank network has a significant stabilising effect on the banking system, as the probability of the default event occurring decreases significantly. This is because group A contains banks with small coefficients μ_i but high coefficients σ_i , who are systemic as they have a large impact on the stability of the banking system, as described in chapter 2. Removing these systemic banks from the interbank network reduces the probability of the default event occurring as other banks are less likely to fail through the contagion channel.

These results have important policy implications. A social planner who aims to minimise the probability of the default event must carefully consider the banking structure when assessing whether it is beneficial to remove weak banks from the interbank network. For a homogeneous banking system, removing weak banks from the interbank network has little impact on the stability of the banking system. However, for a heterogeneous banking system, removing weak banks from the interbank network can have a significant stabilising effect on the banking system. It is worth remembering that there is a trade-off on systemic risk when removing weak banks from the banking system, as these weak banks are less likely to recover. There could also be banking structures where removing weak banks from the interbank network has a destabilising effect on the banking system.

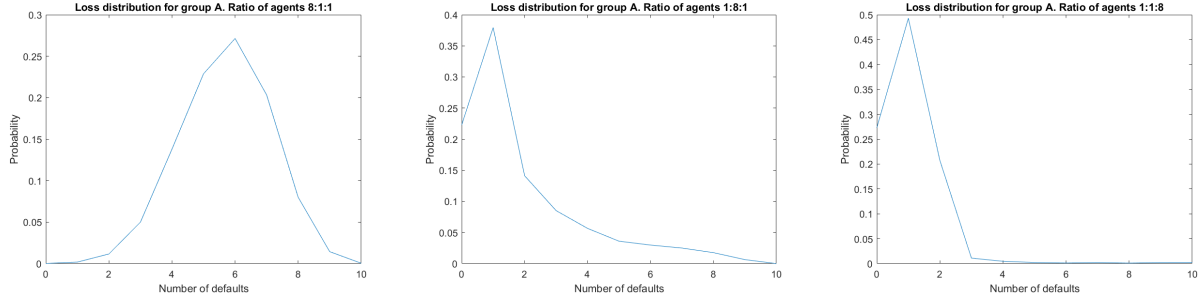


Figure 4.8: Loss distributions for group A when the ratio of banks is 8:1:1, 1:8:1 and 1:1:8, and weak banks are removed from the interbank network.

4.6 Summary

Overall, we found that the interbank network structure can have a material impact on systemic risk. Our numerical experiments suggest that some networks are preferred over others, depending on the banking structure. A central bank may be able to reduce the probability of the default event occurring by implementing policy that encourages banks to form a network structure that is preferred for the banking structure. For example, a less interconnected network may be preferred for a banking structure that is inherently unstable, while a more interconnected network may be preferred for a banking structure that is inherently stable.

Chapter 5

Interbank game

5.1 Introduction

In this chapter, we study the impact of including a central bank into the interbank market. Carmona et al. (2015) develop a stochastic differential game that models interbank borrowing and lending, where each bank controls its rate of borrowing or lending to a central bank. We assume that number of banks N is finite, and that the time horizon T is finite. Let X_t^i denote the log-monetary reserves of bank for $i = 1, \dots, N$ at time $t \in [0, T]$. The dynamical process is given by

$$dX_t^i = (\mu(\bar{X}_t - X_t^i) + \alpha_t^i) dt + \sigma \left(\rho dW_t^0 + \sqrt{1 - \rho^2} dW_t^i \right), \quad X_0^i = 0, \quad t \in [0, T], \quad (5.1)$$

where the control $\alpha^i := (\alpha_t^i)_{t \in [0, T]}$ is chosen to minimise the running quadratic cost f^i and terminal cost g^i

$$\alpha^i = \arg \min_{\alpha^i} J^i(\alpha) := \mathbb{E} \left[\int_0^T f^i(t, X_t, \alpha_t^i) dt + g^i(X_T) \right], \quad (5.2)$$

where $f^i : [0, T] \times \mathbb{R}^N \times \mathbb{R} \rightarrow \mathbb{R}$ and $g^i : \mathbb{R}^N \rightarrow \mathbb{R}$ are given by:

$$f^i(t, X_t, \alpha_t^i) = \frac{1}{2} (\alpha_t^i)^2 - q \alpha_t^i (\bar{X}_t - X_t^i) + \frac{\varepsilon}{2} (\bar{X}_t - X_t^i)^2, \quad \text{with } q^2 \leq \varepsilon, \quad (5.3)$$

$$g^i(X_T) = \frac{c}{2} (\bar{X}_T - X_T^i)^2, \quad (5.4)$$

where $X_t := (X_t^1, \dots, X_t^N)$ and $\alpha := (\alpha^1, \dots, \alpha^N)$.

The interpretation of this model is that banks can control their rate of borrowing or lending to the central bank at each time $t \in [0, T)$ through the control $\alpha_t^i \in \mathbb{R}$. Banks borrow from the central bank if $\alpha_t^i > 0$ and lend to the central bank if $\alpha_t^i < 0$. The introduction of the controls and the optimisation problem in eqs. (5.2) to (5.4) give banks an incentive to have reserves, in contrast to previous models. The running cost function given by eq. (5.3) consists of three terms. The first term $\frac{1}{2}(\alpha_t^i)^2$ is the cost of borrowing or lending from the central bank. The second term $q\alpha_t^i(\bar{X}_t - X_t^i)$ incentivises the bank to borrow if their monetary reserve level is smaller than the mean level, and lend if their monetary reserve level is larger than the mean level. The parameter q is the strength of this incentive. Finally, the quadratic terms $(\bar{X}_T - X_T^i)^2$ in the running cost and

terminal cost in eq. (5.3) and eq. (5.4) penalise the bank for having a monetary reserve level that is different from the mean level. The parameters ε and c are the strengths of this penalty. The assumption that $q^2 \leq \varepsilon$ ensures that f^i is convex in (X_t, α_t^i) . This ensures that the optimisation problem in eq. (5.2) has a unique solution.

5.2 Methodology

The interbank game given in eq. (5.1) falls into the class of N -player *Linear-Quadratic* (LQ) games due to the model's linear dynamics and quadratic cost function. We focus on the closed-loop Nash equilibrium of the game, where each bank's control at time $t \in [0, T)$ is a function of the state of the system, X_t . This is as opposed to the open-loop Nash equilibrium, where banks' optimal controls are deterministic functions α_t given at time $t = 0$. We focus on the closed-loop Nash equilibrium as we think the assumption that banks can observe or infer other banks reserve levels over time is more realistic, and the problem is more challenging.

For this particular game, Carmona et al. (2015) construct analytical solutions for the Nash Equilibrium of eq. (5.1) for both the open-loop and closed-loop Nash equilibria. However, the existence of the analytical solution to this game is dependent on its particular features; having linear dynamics, quadratic costs and interactions through the empirical mean. Most models will not have these specific features. For example, if we wish to extend the interbank game in eq. (5.1) to analyse the impact of the banking structure on financial stability by allowing the model parameters μ and σ to be bank dependent, or by incorporating different network structures, these games might no longer admit an analytical solution, or it will be difficult to derive. Therefore, we need to rely on numerical methods to construct approximate solutions to the games. Constructing numerical solutions to these games is often still challenging, due to the high-dimensional nature of the problem, which increases with N and T . Other modelling features such as common noise add additional complications. To overcome these issues, we use a deep learning method to construct an approximate solution to solve these games.

Recent developments in deep learning methods have made it possible to approximate solutions to high-dimensional problems such as the interbank game described in eqs. (5.1) to (5.4), which may not be feasible to solve using standard numerical methods due to the curse of dimensionality. Hu and Lauriere (2022) give a comprehensive overview of different machine learning methods for solving stochastic games. We adapt their global in time approach to use a single neural network to directly parameterise the banks' control functions in order to estimate the optimal controls. This approach trains the neural network using the whole time horizon at once.

We construct an approximate solution to the discrete time version of eqs. (5.1) to (5.4). We discretise the time interval $[0, T]$ into N_T equally spaced subintervals $(t_n)_{n=0}^{N_T}$, where $t_0 = 0$, $t_n = T$ and the width of a subinterval $t_n - t_{n-1}$ is equal to $\Delta t > 0$. We approximate the dynamics in eq. (5.1) by their discretised versions:

$$\check{X}_{t_{n+1}}^i = \check{X}_{t_n}^i + (\mu (\check{X}_{t_n} - \check{X}_{t_n}^i) + \alpha_{t_n}^i) \Delta t + \sigma \left(\rho \Delta \check{W}_{t_n}^0 + \sqrt{1 - \rho^2} \Delta \check{W}_{t_n}^i \right), \quad \check{X}_0^i = 0, \quad (5.5)$$

where $\Delta \check{W}^0$ and $\Delta \check{W}^i$ are i.i.d random variables with distribution $\mathcal{N}(0, \Delta t)$. The optimal controls

$(\alpha_{t_n})_{n=0}^{N_T-1}$ are chosen to minimise the discretised version of the total cost in eq. (5.2):

$$(\alpha_{t_n}^i)_{n=0}^{N_T-1} = \arg \min_{(\alpha^i)} \tilde{J}^i(\alpha) = \mathbb{E} \left[\sum_{n=0}^{N_T-1} f^i(t_n, \tilde{X}_{t_n}, \alpha_{t_n}^i) \Delta t + g^i(\tilde{X}_T) \right]. \quad (5.6)$$

We parameterise the control function at each time step with a neural network $\alpha_\theta(t_n, X_{t_n}): \mathbb{R}^{N+1} \rightarrow \mathbb{R}^N$, where $S := (t_n, X_{t_n}^1, \dots, X_{t_n}^N)$ is the $(N+1)$ -dimensional input for the neural network and θ are the parameters of the neural network. Given this parameterisation, the cost function in eq. (5.6) can be rewritten as:

$$\tilde{J}^i(\theta) = \mathbb{E} \left[\sum_{n=0}^{N_T-1} f^i(t_n, \tilde{X}_{t_n}^\theta, \alpha_\theta^i(t_n, \tilde{X}_{t_n}^\theta)) \Delta t + g^i(\tilde{X}_T^\theta) \right], \quad (5.7)$$

where $(\tilde{X}_{t_n}^\theta)_{n=1}^{N_T}$ are computed using eq. (5.5) using the parameterised version of the controls $\alpha_\theta(t_n, \tilde{X}_{t_n}^\theta)$.

We can estimate the expected value in eq. (5.7) using Monte Carlo simulations to generate sample paths $(\tilde{X}_{t_n}^{b,\theta})_{n=0}^{N_T}$ for $b = 1, \dots, N_B$ and then taking the average of the cost function for each sample:

$$L^i(\theta) = \frac{1}{N_B} \sum_{b=1}^{N_B} \left[\sum_{n=0}^{N_T-1} f^i(t_n, \tilde{X}_{t_n}^{b,\theta}, \alpha_\theta^i(t_n, \tilde{X}_{t_n}^{b,\theta})) \Delta t + g^i(\tilde{X}_T^{b,\theta}) \right]. \quad (5.8)$$

In practice, we use the antithetic variates method to reduce the variance of our Monte Carlo estimator, as explained in Kroese et al. (2013). For each Brownian path $(\Delta \tilde{W}_{t_n})_{n=1}^{N_T}$ used to simulate $(\tilde{X}_{t_n}^\theta)_{n=0}^{N_T}$ using eq. (5.5), we also simulate $(\tilde{X}_{t_n}^\theta)_{n=0}^{N_T}$ using the antithetic Brownian path $(-\Delta \tilde{W}_{t_n})_{n=1}^{N_T}$. In total, we generate $N_B/2$ antithetic pairs of Brownian paths, where an antithetic pair is given by $(\Delta \tilde{W}_{t_n}, -\Delta \tilde{W}_{t_n})_{n=1}^{N_T}$. This gives us N_B Brownian paths, which are used to generate the sample paths $((\tilde{X}_{t_n}^{b,\theta})_{n=0}^{N_T})_{b=1}^{N_B}$ that are used in eq. (5.8).

Finally, the loss function for training our neural network is given by the mean of the losses for each bank:

$$L(\theta) = \frac{1}{N} \sum_{i=1}^N L^i(\theta). \quad (5.9)$$

5.3 Implementation

We solve the discretised version of the interbank game given in eqs. (5.5) to (5.7) using the global in time direct parameterisation method described above. The parameters we choose for the interbank game are $N = 10$, $N_T = 50$, $N_B = 100$, $T = 1$, $\mu = 1$, $\sigma = 0.2$, $\rho = 0.2$, $q = 1$, $c = 1$ and $\varepsilon = 1.5$. We use a feedforward neural network to approximate the optimal controls in the interbank game. Figure 5.1 shows the architecture of our neural network. The hyperparameters for our neural network were chosen from performing a simple hyperparameter sweep, and we found that a neural architecture consisting of two hidden layers with 64 nodes and the ReLU activation function for each hidden layer, along with no activation function for the output layer works well to approximate the banks' optimal control. The neural network takes as input the $(N+1)$ -dimensional state vector

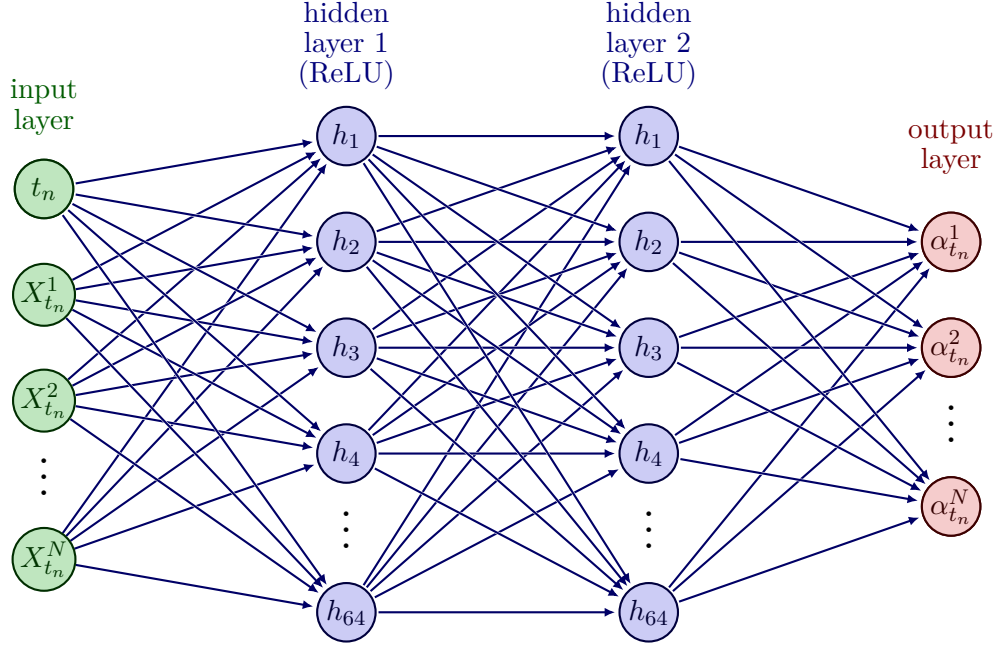


Figure 5.1: The feedforward neural network's architecture for approximating the optimal controls for the interbank game. Image adapted from Neutelings (2023).

Algorithm 1: Global in time direct parameterisation of the control function

Input: $N, T, \Delta t, X_0, \mu, \sigma, \rho$: Model parameters
 N_T : Number of subintervals on $[0, T]$
 N_B : Batch size
 N_{epochs} : Number of epochs

- 1 **Initialisation:**
- 2 Pre-compute $N_B/2$ Brownian paths $(\Delta \check{W}_{t_n}^0)_{n=0}^{N_T}$ and $(\Delta \check{W}_{t_n}^i)_{n=0}^{N_T}$ for $i = 1, \dots, N$, from the distribution $\Delta \check{W} \sim \mathcal{N}(0, \Delta t)$. Include the antithetic Brownian paths.
- 3 Define the architecture of the neural network, $\alpha_\theta(t_n, \check{X}_{t_n})$.
- 4 Initialize the weights of the neural network, θ using the Glorot uniform distribution.

//Training loop:

- 5 **for** epoch = 1 **to** N_{epochs} **do**
- 6 Simulate N_B sample paths $(\check{X}_{t_n})_{n=0}^{N_T-1}$ according to eq. (5.5), using the pre-computed Brownian increments and the neural network to parameterise the control;
 $\alpha_{t_n} = \alpha_\theta(t_n, \check{X}_{t_n})$.
- 7 Compute the loss $L(\theta)$ according to eqs. (5.8) to (5.9).
- 8 Compute $\nabla_\theta L$, the gradients of the loss with respect to the weights, θ .
- 9 Update the weights, θ using gradient descent.

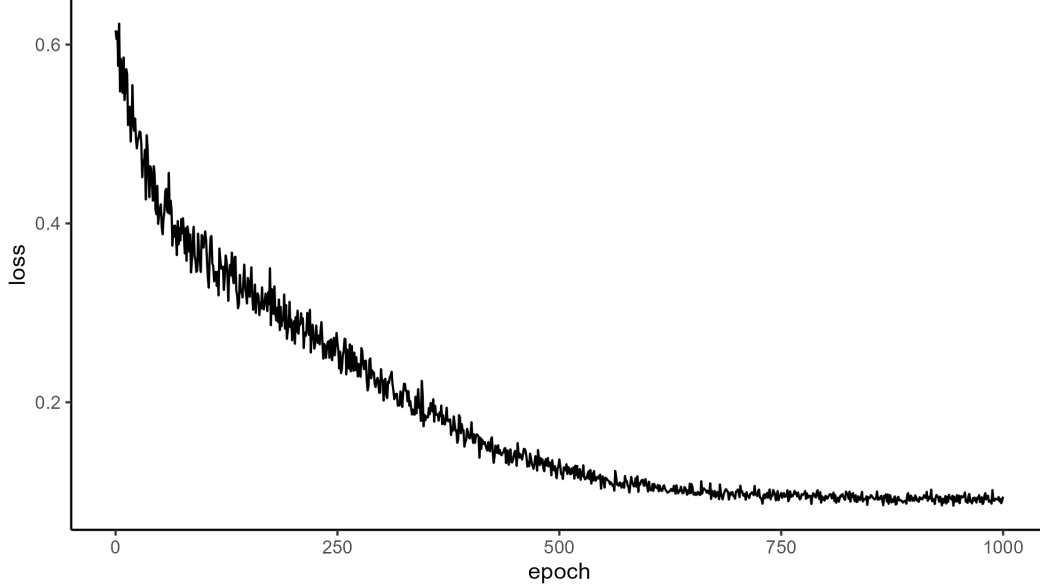


Figure 5.2: The neural network’s loss function during training for the interbank game.

$S = (t_n, X_{t_n}^1, \dots, X_{t_n}^N)$ and outputs the approximated controls for each bank at time step t_n given by $(\alpha_{t_n}^1, \dots, \alpha_{t_n}^N)$. For $N = 10$, this neural network has 5578 trainable parameters. We use the Adam optimiser with a learning rate of 0.0001 and train the neural network for 1000 epochs, after which the loss function has converged. Figure 5.2 shows the loss function for the neural network during training. The training time for this model was around six minutes on a regular laptop CPU.

5.4 Results

We find that the global in time direct parameterisation method is able to estimate the optimal controls and dynamics for the interbank game with a good degree of accuracy. Figure 5.3 compares the analytical and estimated dynamics and optimal controls for $N = 10$ banks over the time horizon $[0, 1]$, with $N_T = 50$ time steps, given the same simulated Brownian paths $(\Delta \check{W}_{t_n}^0, \Delta \check{W}_{t_n}^i)_{n=0}^{N_T}$ for $i = 1, \dots, N$. The estimation of the dynamics are very close to the analytical solution and the estimation of the optimal controls are also reasonably close to the analytical optimal controls. It is likely that we could improve the estimation of the optimal controls by increasing the batch size used for estimating the expectation in eq. (5.8) or by training the neural network longer, but this would increase the computational time. As the method is able to estimate the optimal controls and dynamics with a good degree of accuracy, we could use the method to study variations of the model given by eqs. (5.1) to (5.4) which do not admit an analytical solution, such as allowing for heterogeneity in the model parameters μ and σ , or incorporating different interbank network structures.

Additionally, we find that the introduction of a central bank into the interbank market increases the stability of the financial system. We estimate the loss distribution of the interbank game described in eqs. (5.5) to (5.7), with and without the controls. We define a bank to default if its log-monetary reserves fall below -0.15 at any time step $t_n \in [0, 1]$. Each loss distribution

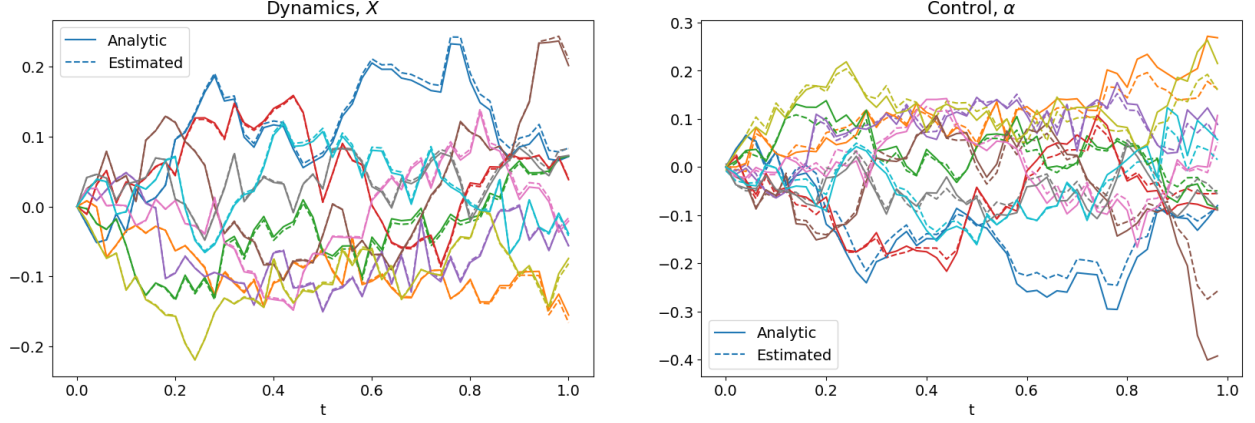


Figure 5.3: The analytical and estimated dynamics and optimal controls for the interbank game.

is estimated using 5000 sample paths; i.e. we simulate the sample paths $(\tilde{X}_{t_n})_{n=0}^{N_T}$ according to eq. (5.5) with and without controls 5000 times, and from this, compute the probability density function for the number of banks defaulting. We use the same simulated Brownian paths for each model, so the only difference between the two models is the controls.

Figure 5.4 compares the loss distribution for the model with the controls to the loss distribution for the model without the controls. The loss distribution of the model with the controls is more concentrated around zero, which indicates that the model with the controls is more stable than the model without the controls. This is because the controls incentivise a bank to borrow from the central bank if its log-monetary reserves falls below the mean level. The central bank is able to provide additional liquidity to the failing banks, which helps to prevent failing banks from defaulting. Interestingly, for this example, the right tail of the loss distribution of the model with the controls is not heavier than the right tail of the loss distribution of the model without the controls. We may have expected this to be the case given that the introduction of the control should increase the flocking behaviour of the banks, as banks are incentivised to borrow and lend to stay close to the mean level of monetary reserves in the financial system. This should increase the probability of many banks defaulting at the same time. However, this is not the case for this example, which suggests that the introduction of the control has strictly improved the stability of this financial system.

5.5 Methodology caveats

The advantage of this global in time direct parameterisation method is that we are able to use a single neural network to estimate the control functions for all banks. Moreover, the parameters of the neural network are shared across all time steps and banks, so all the parameters can be trained at the same time. In practice, we find that the global in time direct parameterisation method works well when the number of banks and the number of time steps are relatively small. This is because the computational complexity of the problem increases exponentially as the number of banks and the number of time steps increase, resulting in increasingly long training times. For example, we find that the model in eqs. (5.5) to (5.7) with $N = 10$ banks and $N_T = 20$ time steps

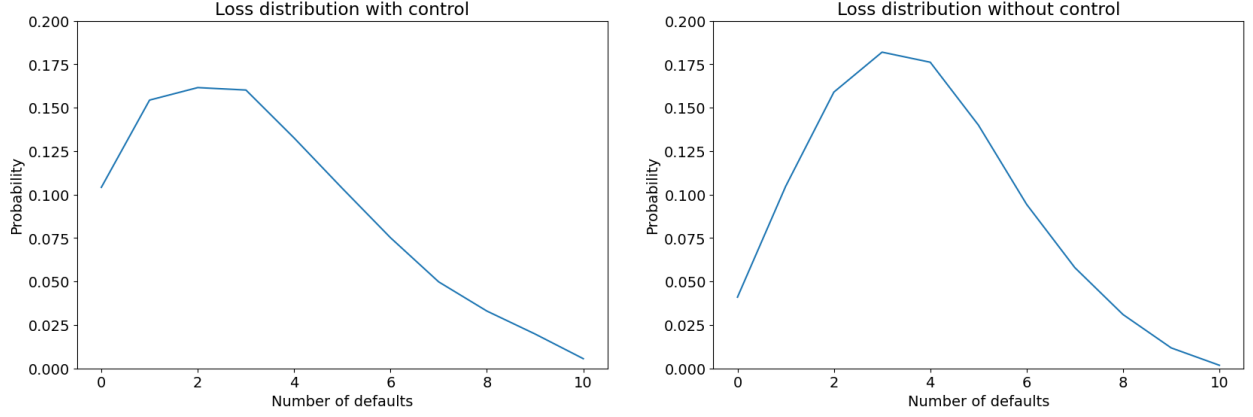


Figure 5.4: The estimated loss distributions for the model with and without controls.

takes approximately one minute to train, while the model with $N = 10$ banks and $N_T = 50$ time steps took approximately six minutes to train, on the same regular laptop CPU. This makes the method unsuitable for learning the control functions for the mean field game when the number of banks is very large, or when the time horizon is very long, due to the computation complexity. Hu and Lauriere (2022) suggest that other methods such as the *signature deep fictitious play* method might be more suitable for solving these types of mean-field games with common noise when the number of agents is large.

Chapter 6

Network inference using neural networks

6.1 Introduction to the problem

In this chapter, we solve the inverse problem of inferring the network given an observed time series of the banks' dynamics. We follow the methodology presented in Gaskin et al. (2023), who developed a methodology for inferring the network adjacency matrix given the observed time series of the agents' dynamics, using a neural network.

The goal is to use a neural network to learn the function that maps the observed time series to an adjacency matrix that is consistent with the observed dynamics. We assume that other than the adjacency matrix, the dynamical process is known, including the value of all parameters. In particular, we assume that the dynamical process is known and given by

$$dX_t^i = \frac{\mu}{N} \sum_{j=1}^N a_{i,j} (X_t^j - X_t^i) dt + \sigma dW_t^i, \quad X_0^i = x, \quad t \in [0, T], \quad (6.1)$$

and the parameters μ , σ and x are assumed to be known. We also assume that μ and σ are constant for each bank, although this assumption can be relaxed. Here, $a_{i,j}$ refer to the value of the i -th row and j -th column of the adjacency matrix \mathbf{A} .

This inverse problem may not have a unique solution. Suppose we observe a time series of X from eq. (6.1) with length L . If $L < N$, the problem is under-determined and requires additional constraints on \mathbf{A} in order to estimate it. If $L = N$, the problem is just-identified. However, symmetries in the data may mean that the number of linearly independent equations is less than N , so the problem may still be under-determined. Even with $L \gg N$, there may be multiple adjacency matrices that are consistent with the observed time series of the dynamical system.

Ordinary Least Squares (OLS) can be used to solve the inverse problem when the dynamics are linear in the adjacency matrix and the problem is just-determined. However, Gaskin et al. (2023) find that their neural approach to inferring the network performs better than OLS in the presence of noise. We employ their method to infer the network with the dynamical process given by eq. (6.1) in the presence of no noise ($\sigma = 0$) and additive noise ($\sigma > 0$).

6.2 Methodology

To test our methodology, we generate synthetic data. First, we generate a random adjacency matrix \mathbf{A} , where for simplicity, \mathbf{A} is assumed to be symmetric, the diagonal elements of \mathbf{A} are equal to 1, and other values of the matrix take values 1 with probability $p = 0.3$ and 0 with probability $1 - p$. The assumption that \mathbf{A} is symmetrical, and the diagonal elements are 1 are not necessary, but simplifies the problem of estimating the number of values of \mathbf{A} from N^2 to $\frac{N(N-1)}{2}$. This reduces the computational complexity of the problem, which is noticeable for large N . Given \mathbf{A} , we simulate eq. (6.1) using the Euler-Maruyama method to create a time series \mathbf{T}^{obs} , where

$$\mathbf{T}^{\text{obs}} = \begin{pmatrix} X_0 \\ X_1 \\ \vdots \\ X_L \end{pmatrix}, \quad X_t = (X_t^1, \dots, X_t^N), \quad L \gg N. \quad (6.2)$$

The goal is to estimate an adjacency matrix $\hat{\mathbf{A}}$ such that the simulated dynamics given the estimated network of banks, denoted $\hat{\mathbf{T}}^{\text{pred}}$, produces a good approximation of the observed time series given the same realisation of the Brownian paths. We adopt a reinforcement learning approach to train a neural network on the observed time series to accurately estimate $\hat{\mathbf{A}}$.

We batch the observed time series into rolling windows of length $q = 2$ to produce our training dataset \mathbf{T} , where

$$\mathbf{T} = \begin{pmatrix} X_0, X_1 \\ X_1, X_2 \\ \vdots \\ X_{L-1}, X_L \end{pmatrix}. \quad (6.3)$$

We define a neural network $m_\theta : \mathbb{R}^{Nq} \rightarrow \mathbb{R}^{N^2}$, where Nq is the dimension of an input, which is a flattened observation of the batched time series \mathbf{T} , and N^2 is the dimension of the outputted adjacency matrix $\hat{\mathbf{A}}$. In our algorithm, we directly impose that $\hat{\mathbf{A}}$ is symmetrical and that the diagonal values of $\hat{\mathbf{A}}$ are equal to one, so in practice, only $\frac{N(N-1)}{2}$ outputs are utilised. The assumption that $\text{diag}(\mathbf{A}) = 1$ is completely arbitrary as the dynamics in eq. (6.1) would be unchanged if the diagonal elements of \mathbf{A} were equal to any other constant.

Next, we simulate the dynamics given in eq. (6.1), but replacing $a_{i,j}$ with their estimated counterparts $\hat{a}_{i,j}$ to produce the estimated time series $\hat{\mathbf{T}}$, using the initial values of each row of \mathbf{T} , where

$$\hat{\mathbf{T}} = \begin{pmatrix} X_0, \hat{X}_1 \\ X_1, \hat{X}_2 \\ \vdots \\ X_{L-1}, \hat{X}_L \end{pmatrix}, \quad \hat{X}_{t+1} = (\hat{X}_{t+1}^1, \dots, \hat{X}_{t+1}^N),$$

and \hat{X}_{t+1}^i is computed using the Euler-Maruyama method:

$$\hat{X}_{t+1}^i = X_t^i + \frac{\mu}{N} \sum_{j=1}^N \hat{a}_{i,j} (X_t^j - X_t^i) \Delta t + \sigma \Delta W_t^i. \quad (6.4)$$

The loss function is chosen as

$$J(\mathbf{T}, \hat{\mathbf{T}}) = \frac{1}{2} \|\mathbf{T} - \hat{\mathbf{T}}\|_2^2 = \frac{1}{2} \sum_{t=1}^L (X_t - \hat{X}_t)^2, \quad (6.5)$$

which is proportional to the sum of squared differences between estimated and observed time series. The neural network's weights are then updated to minimise the loss.

6.3 Implementation

We solve for the network given two different calibrations of eq. (6.1). For both calibrations, we set $N = 9$, $T = 1$ and generate a random adjacency matrix \mathbf{A} that acts as the true network. For the numerical solver, we set $\Delta t = 0.001$, $X_0 = (-4, -3, -2, -1, 0, 1, 2, 3, 4)$ and simulate an observed time series over the time horizon $[0, T]$. The first half of the time series is used as training data for the models, and the second half is reserved for validating the model's performance by assessing its ability to forecast out of sample. We assume the model parameters for eq. (6.1) are known, and set $\mu = 1$ for both models. For the first model, we set $\sigma = 0$, so the dynamics are deterministic. For the second model, we set $\sigma = 0.1$, which adds a moderate amount of noise to the dynamical system.

We use a feedforward neural network to estimate the adjacency matrix of the interbank network. Figure 6.1 shows the architecture of our neural network. The hyperparameters for our network are chosen using heuristics and a simple hyperparameter sweep. We find that a deep neural network with two hidden layers and 20 nodes per layer works well to estimate an adjacency matrix that is consistent with the observed dynamics of the system. The input layer takes one row of \mathbf{T} , which is a vector of length Nq . The neural network outputs the adjacency matrix, which has N^2 elements. This is then reshaped to an $N \times N$ dimensional matrix to get $\hat{\mathbf{A}}$, and we then impose that $\hat{\mathbf{A}}$ is symmetrical and has diagonal elements equal to one. In each layer, we omit the bias and use the ReLU activation function except for the outer layer, where we use the sigmoid activation function to constrain the output to be between 0 and 1. For $N = 9$, this neural network has 2380 trainable parameters.

To train the neural network's weights, we use the Adam optimiser with a learning rate of 0.002. The neural network's weights are initialised such that the prior is the identity matrix, I_N . To initialise the weights, we define the loss function by:

$$L(\hat{\mathbf{A}}) = \frac{1}{2} \|\hat{\mathbf{A}} - I_N\|_2^2 = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (\hat{a}_{i,j} - \delta_{i,j})^2, \quad (6.6)$$

and train the neural network's output given the training data until $L(\hat{\mathbf{A}})$ is sufficiently small. After initialising the neural network's weights, we reset the learning rate and other parameters of the Adam optimiser. If one had prior knowledge of the true network, the training can be sped up by changing the prior of the adjacency matrix from I_N to a matrix closer to the true solution. For example, if we knew that the true network was more densely connected, it would be better to initialise the weights with a prior of a fully connected adjacency matrix.

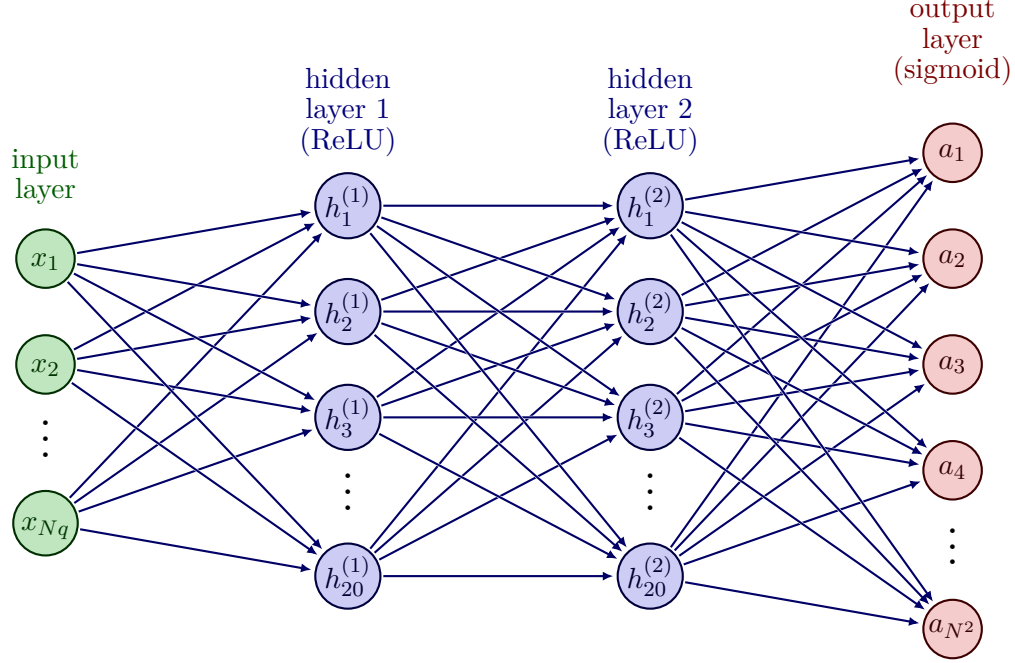


Figure 6.1: The feedforward neural network's architecture for estimating the interbank network.

Algorithm 2: Network inference using a neural network

Input: $N, T, \Delta t, X_0, \mu, \sigma, q$: length of time window, N_{epochs} : number of epochs

1 **Initialisation:**

 //Generate synthetic data

2 Generate a random, symmetrical adjacency matrix \mathbf{A} with $\text{diag}(\mathbf{A}) = 1$.

3 Simulate the dynamical process in eq. (6.1) using the numerical solver to get an observed time series \mathbf{T}^{obs} .

4 Batch \mathbf{T}^{obs} into rolling windows of length q according to eq. (6.3) to get the training dataset \mathbf{T} .

 //Initialise neural network

5 Define the architecture of the neural network, m_θ .

6 Initialize the weights of the neural network, θ , such that $m_\theta(\mathbf{T}) = I_N$, where I_N is the prior of the adjacency matrix.

 //Training loop:

7 **for** epoch = 1 **to** N_{epochs} **do**

8 Compute $\hat{\mathbf{A}} = \text{mean}(m_\theta(\mathbf{T}))$.

9 Impose additional assumptions on $\hat{\mathbf{A}}$ (symmetrical and $\text{diag}(\hat{\mathbf{A}}) = 1$).

10 Simulate the dynamical process according to eq. (6.4) using the numerical solver to get an estimated time series $\hat{\mathbf{T}}$.

11 Compute $J(\mathbf{T}, \hat{\mathbf{T}})$ according to eq. (6.5).

12 Compute $\nabla_\theta J$, the gradients of the loss with respect to the weights, θ .

13 Update the weights, θ using gradient descent.

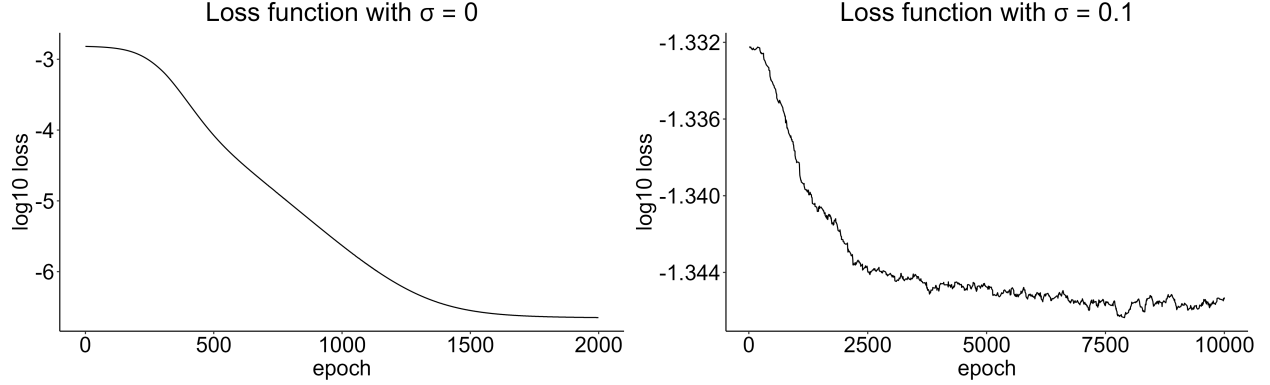


Figure 6.2: The \log_{10} losses for the models with $\sigma = 0$ and $\sigma = 0.1$. The chart on the right is smoothed using an exponential moving average with a smoothing parameter of 0.99.

6.4 Results

We train the neural networks for both calibrations of the model using batch gradient descent until the loss functions converge. Figure 6.2 shows the training progress for the neural networks for both models. The model with no noise smoothly converges after 2000 epochs, which takes less than one minute when trained on a regular laptop CPU. The model with noise takes around 10000 epochs to converge and takes around three minutes to train. The loss remains higher compared to the model without noise because even if the neural network perfectly predicts the network, the simulated paths for $\hat{\mathbf{T}}$ will have different realisations for the Brownian increments, so $J(\mathbf{T}, \hat{\mathbf{T}})$ is non-zero.

We find that the neural network does a good job of inferring the network structure for both calibrations of the model. Figures 6.3 and 6.4 show the simulated dynamics using the actual network and the predicted network for both calibrations of the model. The models are trained on the first half of the time series (up to $t = 0.5$), and the second half of the time series is used as a measure of the out of sample performance for the model.

For the model without noise, the dynamics are deterministic, and the model estimates a network that produces a time series that replicates the training data to machine precision, which can be seen on fig. 6.3. For the model with noise, the model is almost able to perfectly fit the training data and does a good job of out-of-sample forecasting too, which can be seen on fig. 6.4. This suggests that the neural network is able to learn the network structure from the observed time series of the dynamical system, despite there being a moderate amount of noise in the system with $\sigma = 0.1$.

6.5 Methodology caveats

We find that the method works well in predicting a network consistent with the observed time series of the dynamical system when the noise parameter σ is not too high relative to the parameter controlling the strength of interactions, μ . Furthermore, it is easier to predict the network when the banks are dissimilar. We achieved this in the above models by spreading out the initial starting positions of the banks. It becomes difficult to estimate the network when all the banks start at the same point, the noise parameter is high and the strength of bank interactions is weak.

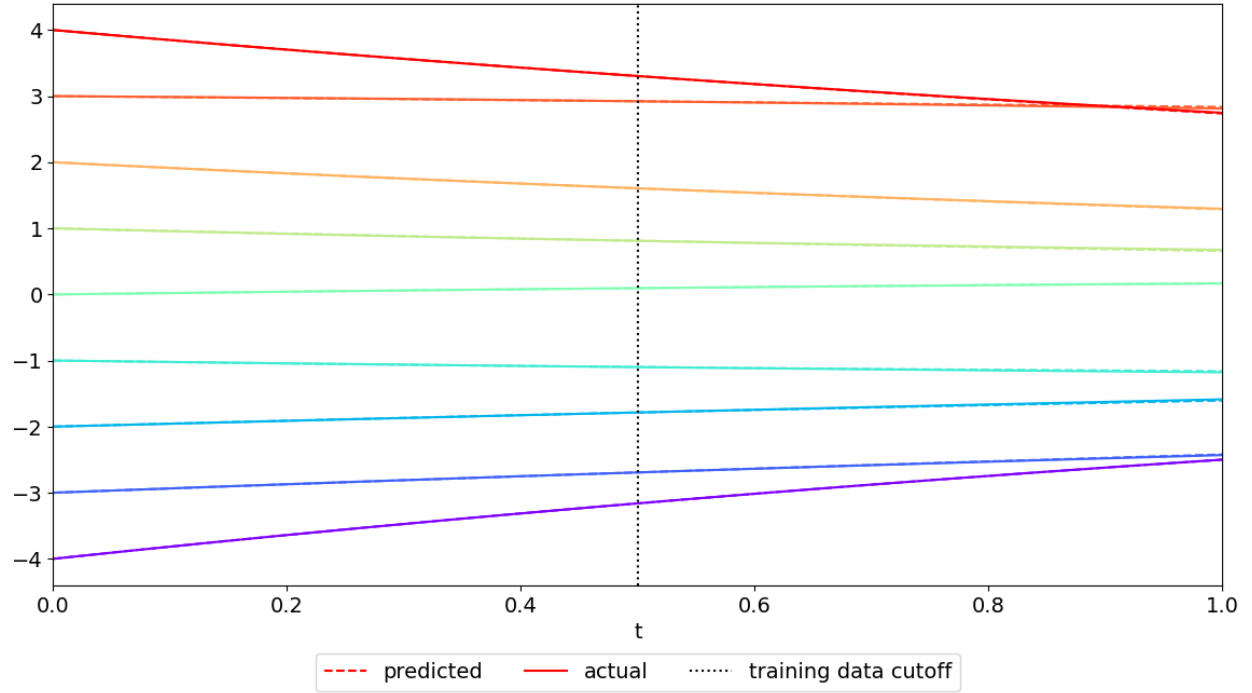


Figure 6.3: Simulated dynamics using the actual and predicted network for the model with $\sigma = 0$.

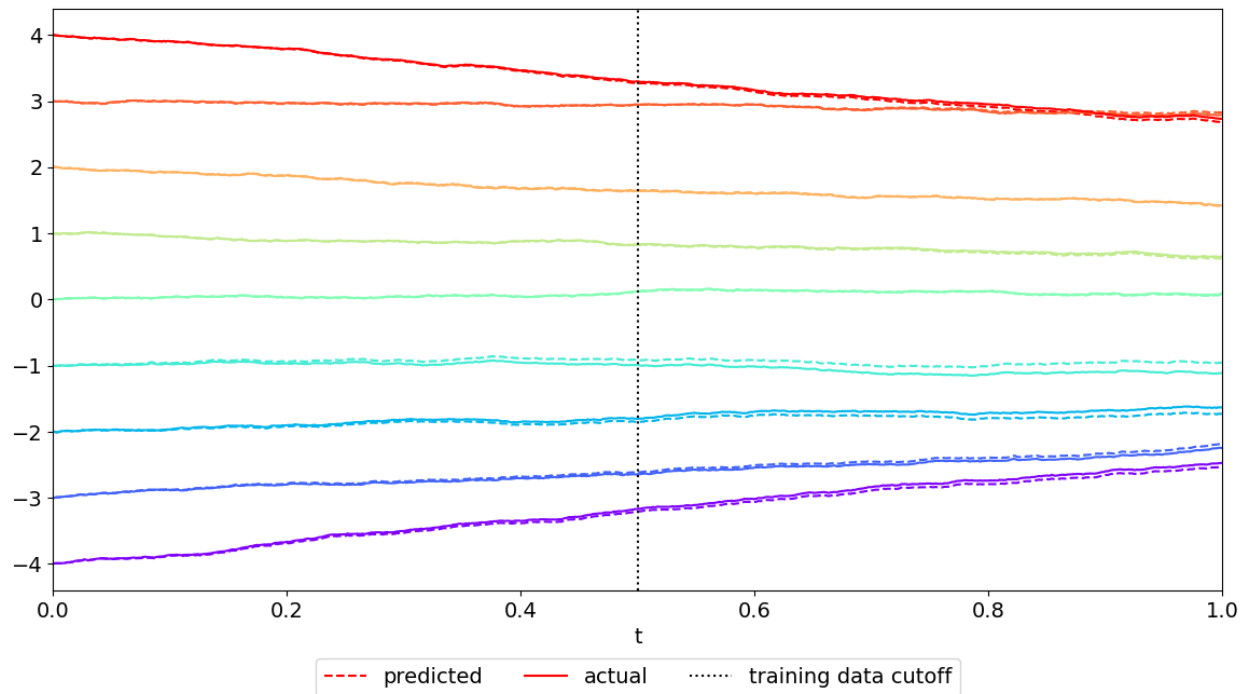


Figure 6.4: Simulated dynamics using the actual and predicted network for the model with $\sigma = 0.1$, under the same Brownian path.

Chapter 7

Conclusion

7.1 Summary

In this thesis, we analysed several models of the interbank market that used coupled stochastic differential equations to capture the interconnectedness of the interbank market. Our contribution to the literature is through numerical experiments and analysis of modelling features on the stability of the banking system.

To start with, we considered a simple model of the interbank market, where the banks are identical and interact with each other through the empirical mean. We found that even a simple model of the interbank market can have rich dynamics, and that different parameters for μ and σ can lead to different outcomes on the stability of the banking system.

We then extended the model to allow for heterogeneity in the model parameters. Our numerical experiments suggested that a heterogeneous banking system with predominantly intermediate sized banks can lead to a more stable banking system. We also analysed the impact of including common and multiplicative noise to model aggregate uncertainty in the banking system, and found that including a highly correlated noise process leads to a much less stable banking system.

Additionally, we analysed the impact of alternative network structures on the stability of the banking system, which to our knowledge, hasn't been done before for the coupled diffusion models that we consider. We found that the interbank network can have a significant impact on the stability of the banking system, and that the probability of the default event can be highly dependent on the network structure. In particular, we found that when there are problematic banks in the interbank network, which are banks with low coefficients μ_i but high coefficients σ_i , it is better for the banking system if these banks are not connected to other banks. This is because if the problematic banks are connected to other banks, they can cause a cascade of defaults in the banking system, which can lead to a systemic crisis. On the other hand, increased interconnectedness can improve the stability of the banking system if the banking structure is more healthy. This is because the banks can share liquidity with each other, which can help to prevent banks from defaulting.

Next, we studied the impact of including a central bank in the interbank market to create a game, where banks now have a control function where they can optimally choose the rate at which they borrow and lend from the central bank. We found that the introduction of a central bank into

the interbank market increases the stability of the financial system. In general, it is not possible to analytically derive the optimal controls for the banks in the interbank game, so we developed a deep learning method for approximating the solution of the interbank game. We found that our method is able to accurately approximate the optimal controls and dynamics for the banks in the interbank game.

Finally, we developed a deep learning methodology for inferring the network structure of the interbank market using neural networks. Our method is able to predict an interbank network that is consistent with the observed dynamics, even when the dynamical system has a moderate degree of noise.

7.2 Policy implications

This class of coupled stochastic diffusion models are currently not being used at the Bank of England. In this thesis, we have shown that these models can be useful for studying the stability of the banking system and has potential to be used for policy analysis. In particular, we have shown that the composition of banks in the interbank market, as well as the interbank network can have a significant impact on the stability of the banking system and the probability of the default event. This suggests that the Bank could consider how it could implement policy to achieve a more stable banking structure. For example, our numerical experiments suggest that the current banking system with a few large banks dominating the interbank market might be less stable than a banking system with more intermediate sized banks. These models also have application in stress testing, where the Bank could use this class of models to develop a model for bank liquidity, which has recently gained traction due to the failure of Silicon Valley Bank in 2023 due to inadequate liquidity.

Our development of a deep learning methodology for inferring the network structure of market has potential to be used for policy analysis. For example, the Bank is interested in studying the market for non-banks such as mutual funds, where the market structure is not as well understood given data limitations. The Bank could use our methodology to infer the network of non-banks from some observed time series of their dynamics, and use the inferred network for stability analysis and for forecasting.

7.3 Further work

Finally, we describe some possible extensions to the models and methodologies presented in this thesis that we think could be interesting to explore.

In this thesis, we generally limited the number of banks to be relatively small due to the computational complexity from simulating many coupled diffusion processes, but it would be interesting to study the dynamics of the model when the number of banks is very large, as well as deriving theoretical estimates for the mean-field limit of the probability of the default event for the interbank model with networks in chapter 4.

There are also several modelling features which we did not consider in our simple interbank model. For example, we could have considered a model with a Hawkes process, which is a self-exciting

process that could be used to model the impact of fire sales on the financial system, as in Borovykh et al. (2018).

In Chapter 5, we developed a deep learning method that accurately approximates the banks' controls in the interbank game. We used this method to study the effect of introducing a central bank into the interbank market, which increased the stability of the financial system. However, we only considered a simple model where all other banks were identical. It would be interesting to study the effect of introducing a central bank into a more complex banking system, where the banks are heterogeneous. For example, we could consider a model where the banks have different parameters for μ and σ , such as in chapter 2. It would also be interesting to study the effect of introducing a central bank into a model with a more complex network structure. For example, we could consider a model where the underlying network structure results in an unstable banking system, and see whether the introduction of the central bank can improve the stability of the system, or whether it would make the system less stable. From a numerical perspective, it would also be interesting to consider other algorithms for approximating the solution of the interbank game that scale better as the number of banks and the number of simulated time steps increase, such as those algorithms presented in Hu and Lauriere (2022).

For the network inference methodology presented in Chapter 6, it would be interesting to see if our method is robust when the number of banks is very large, as the size of the adjacency matrix increases quadratically with the number of banks. It would also be interesting to see if our method is robust when the dynamical system has common or multiplicative noise.

Finally, we could calibrate our models to fit the structure of the UK banking system, such as calibrating banks' initial reserve levels, their model parameters as well as the interbank network, and then try to improve the stability of the banking system by implementing policy that could alter modelling features such as the network structure. This would require a more detailed analysis of the UK banking system and would require the use of confidential banking data and is beyond the scope of this thesis.

Bibliography

- Abadi, Martín, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng (2015), “Tensorflow: Large-scale machine learning on heterogeneous systems.” URL <https://www.tensorflow.org/>. Software available from tensorflow.org.
- Borovykh, Anastasia, Andrea Pascucci, and Stefano La Rovere (2018), “Systemic risk in a mean-field model of interbank lending with self-exciting shocks.” *IIE transactions*, 50, 806–819, URL <https://www.tandfonline.com/doi/abs/10.1080/24725854.2018.1448491>.
- Carmona, Rene A., Jean Pierre Fouque, and Li Hsien Sun (2015), “Mean field games and systemic risk.” *Communications in Mathematical Sciences*, 13, 911–933. Publisher Copyright: © 2015 International Press.
- Chiba, Hayato, Georgi S. Medvedev, and Matthew S. Mizuhara (2018), “Bifurcations in the kuramoto model on graphs.” *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 28, 073109, URL <https://doi.org/10.1063%2F1.5039609>.
- Chollet, François et al. (2015), “Keras.” <https://keras.io>.
- Cont, Rama, Amal Moussa, and Edson B. Santos (2013), *Network Structure and Systemic Risk in Banking Systems*, 327–368. Cambridge University Press.
- Fang, Fei, Yiwei Sun, and Konstantinos Spiliopoulos (2017), “On the effect of heterogeneity on flocking behavior and systemic risk.” *Statistics & Risk Modeling*, 34, 141–155, URL <https://doi.org/10.1515/strm-2016-0013>.
- Fouque, Jean-Pierre and Li-Hsien Sun (2013), *Systemic Risk Illustrated*, 444–452. Handbook on Systemic Risk, Cambridge University Press, URL <http://dx.doi.org/10.1017/CB09781139151184.023>.
- Garnier, Josselin, George Papanicolaou, and Tzu-Wei Yang (2013), “Large deviations for a mean field model of systemic risk.” *SIAM journal on financial mathematics*, 4, 151–184, URL <https://search.proquest.com/docview/1314766305>.

- Gaskin, Thomas, Grigorios A. Pavliotis, and Mark Girolami (2023), “Neural parameter calibration for large-scale multiagent models.”
- Giesecke, Kay, Gustavo Schwenkler, and Justin A. Sirignano (2020), “Inference for large financial systems.” *Mathematical finance*, 30, 3–46, URL <https://onlinelibrary.wiley.com/doi/abs/10.1111/mafi.12222>.
- Glorot, Xavier and Yoshua Bengio (2010), “Understanding the difficulty of training deep feedforward neural networks.” In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics* (Yee Whye Teh and Mike Titterton, eds.), volume 9 of *Proceedings of Machine Learning Research*, 249–256, PMLR, Chia Laguna Resort, Sardinia, Italy, URL <https://proceedings.mlr.press/v9/glorot10a.html>.
- Goodfellow, Ian J., Yoshua Bengio, and Aaron Courville (2016), *Deep Learning*. MIT Press, Cambridge, MA, USA. <http://www.deeplearningbook.org>.
- Higham, D. J. (2001), “An algorithmic introduction to numerical simulation of stochastic differential equations.” *SIAM Review*, 43, 525–546, URL <http://dx.doi.org/10.1137/S0036144500378302>.
- Horchler, Andrew (2023), “Sdetools.” URL <https://github.com/horchler/SDETools>.
- Hornik, Kurt, Maxwell Stinchcombe, and Halbert White (1989), “Multilayer feedforward networks are universal approximators.” *Neural Networks*, 2, 359–366, URL <https://www.sciencedirect.com/science/article/pii/0893608089900208>.
- Hu, Ruimeng and Matthieu Lauriere (2022), “Recent developments in machine learning methods for stochastic control and games.” URL <http://dx.doi.org/10.2139/ssrn.4096569>.
- Kingma, Diederik P. and Jimmy Ba (2015), “Adam: A method for stochastic optimization.” In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (Yoshua Bengio and Yann LeCun, eds.), URL <http://arxiv.org/abs/1412.6980>.
- Kroese, D.P., T. Taimre, and Z.I. Botev (2013), *Handbook of Monte Carlo Methods*. Wiley Series in Probability and Statistics, Wiley.
- Motsch, Sebastien and Eitan Tadmor (2014), “Heterophilous dynamics enhances consensus.” *SIAM review*, 56, 577–621, URL <https://www.jstor.org/stable/24244329>.
- Neutelings, Izaak (2023), “Neural networks.” URL https://tikz.net/neural_networks/.
- Pavliotis, Grigorios A. (2014), *Stochastic Processes and Applications: Diffusion Processes, the Fokker-Planck and Langevin Equations*, 2014 edition, volume 60. Springer, New York, NY.
- Yellen, Janet L. (2013), “Interconnectedness and Systemic Risk: Lessons from the Financial Crisis and Policy Implications: a speech at the American Economic Association/American Finance Association Joint Luncheon, San Diego.” Speech 631, Board of Governors of the Federal Reserve System (U.S.), URL <https://ideas.repec.org/p/fip/fedgsq/631.html>.

Appendix A

Preliminaries

A.1 Stochastic processes

We give a brief overview of some technical details regarding stochastic processes. The reader is referred to Pavliotis (2014) for a more detailed treatment of the topics presented in this section. We assume that the reader is already familiar with the fundamentals of probability theory, although the reader can refer to Appendix B of Pavliotis (2014) for a review of the topic.

A stochastic process is a collection $X = (X_t)_{t \in \mathbb{T}}$ of \mathcal{F} -measurable random variables, where \mathbb{T} is an ordered set and $\mathbf{F} = (\mathcal{F}_t)_{t \in \mathbb{T}}$ is the natural filtration with respect to X .

A.1.1 Brownian Motion

A stochastic process $W = \{W_t\}_{t \geq 0}$ is a standard Brownian motion if it satisfies the three conditions:

- $W_0 = 0$;
- W_t is almost surely continuous;
- for every $0 \leq s \leq t$, $W_t - W_s$ is independent and normally distributed with mean 0 and variance $t - s$.

In this thesis, we assume that bank's monetary reserves is a stochastic process, which fluctuates randomly over time and that these fluctuations are independent of each other. The properties of Brownian motion make it suitable to model such a process. Although Brownian motion is almost surely continuous, it is nowhere differentiable, so standard methods from calculus are not applicable. Itô calculus extends the standard methods of calculus to stochastic processes.

Thus far, we have taken for granted that a stochastic process satisfying the above properties exists and that there exists a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ supporting a countable sequence of independent Brownian motions. The proof for the existence of Brownian motion is given in Pavliotis (2014). For our thesis, we omit such technicalities and always assume that an underlying probability space $(\Omega, \mathcal{F}, \mathbb{P})$ supporting our stochastic process X exists.

A.1.2 Stochastic differential equation

A stochastic differential equation (SDE) is a differential equation in which one or more of the terms is a stochastic process. We consider stochastic differential equations of the form

$$X_t = x + \int_0^t a(u, X_u) \, du + \int_0^t b(u, X_u) \, dW_u \quad (\text{A.1})$$

or equivalently in differential equation form,

$$dX_t = a(t, X_t) \, dt + b(t, X_t) \, dW_t, \quad X_0 = x, \quad 0 \leq t \leq T, \quad (\text{A.2})$$

where a and b are Borel-measurable functions. The initial condition $X_0 = x$ can either be deterministic or a random variable with a known distribution.

A weak solution to the stochastic differential equation is a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and a stochastic process X_t that solves the integral eq. (A.1).

In this thesis, we omit many technical details and implicitly assume that all regularity conditions are satisfied such that there exists a weak solution to our stochastic differential equation. We rely on numerical methods to solve our stochastic differential equations.

A.1.3 Numerical simulation of stochastic differential equations

There are numerous numerical methods to approximate the numerical solution of stochastic differential equations of the form eq. (A.2).

Higham (2001) gives an introduction to numeric simulation methods for stochastic differential equations, such as the Euler-Maruyama and Milstein's method. These two methods are equivalent when the noise is additive; when the function b in eq. (A.2) is a function of t only. As most of our simulations only have additive noise, we use the Euler-Maruyama method as the primary method for simulating the solution to our stochastic differential equations. We give an overview of the Euler-Maruyama method for simulating the numerical solution of a stochastic differential equation of the form given in eq. (A.2) over the time interval $[0, T]$, where $T > 0$ is assumed to be finite.

A.1.4 Euler-Maruyama method

We discretise the time interval $[0, T]$ into equally spaced subintervals $[t_0, t_1, \dots, t_N]$, where $t_0 = 0$, $t_N = T$ and the width of a subinterval $t_n - t_{n-1}$ is equal to $\Delta t > 0$. We approximate the continuous process X_t by their discretised versions, denoted \check{X}_t :

$$\check{X}_{t_{n+1}} = \check{X}_{t_n} + a(t_n, \check{X}_{t_n})\Delta t + b(t_n, \check{X}_{t_n})\Delta \check{W}_{t_n},$$

where $\Delta \check{W} \sim \mathcal{N}(0, \Delta t)$ are discretised Brownian increments that are normally distributed with zero mean and variance Δt .

A.1.5 SDETools

In this thesis, we make extensive use of the MATLAB toolbox SDETools to simulate the trajectories of our stochastic processes, developed by Horchler (2023). This package has a fast implementation of the Euler-Maruyama method, with options for including common and multiplicative noise.

We primarily use the function `sde_euler` to simulate the numerical solution for our stochastic differential equations, and `sdeplot` to plot the time series output of our numerical solution.

A.2 Graph theory

We give a brief introduction to the basics of network theory. We define networks as graphs, where the nodes (or vertices) of the graph represent individual banks, and a line (or edge) connecting two nodes represent a link between two banks. Throughout this thesis, we assume that for any two nodes, we have no more than one edge connecting them.

Graphs may be *directed* or *undirected*. In a directed graph, the edges have a direction, whereas in an undirected graph, the edges do not have a direction. Furthermore, graphs may be *weighted* or *unweighted*. In a weighted graph, each edge has a weight associated with it, whereas in an unweighted graph, each edge has a weight of one.

In this thesis, we focus only on undirected, unweighted graphs. The interpretation of an edge between two banks represents an interaction between the two banks. Banks that interact with each other influence the reserve levels of each other through their borrowing or lending on the interbank market. We assume that the influence of bank i on bank j is symmetric, so the graph is undirected. The economic interpretation of this is if bank i lends to bank j , then bank j borrows from bank i .

We can represent a graph as an *adjacency matrix*, \mathbf{A} , where the i -th row and j -th column of \mathbf{A} is equal to 1 if there is an edge from node i to node j , and 0 otherwise. For an undirected graph, the adjacency matrix is symmetric. An example of a random undirected graph and its corresponding adjacency matrix is given below. The rows of the adjacency matrix correspond to the labels of the nodes in the graph.

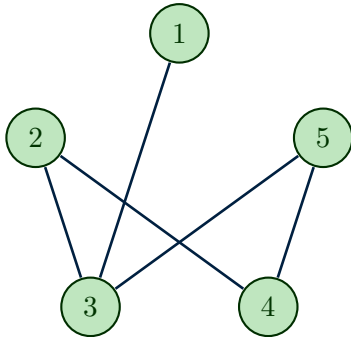


Figure A.1: An undirected graph

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Figure A.2: Adjacency matrix for the graph

A.3 Neural networks

A.3.1 Feedforward neural network

In this thesis, we use deep feedforward neural networks to approximate banks' optimal controls in the interbank game given in chapter 5, as well as in chapter 6 where we solve the inverse problem of inferring the network topology given the observed dynamics of the banks. In both problems, the feedforward neural network is used to learn some function that maps the input to

the output. We use feedforward neural networks because they work well in practice, particularly in approximating high-dimensional functions where traditional grid-based methods suffer from the curse of dimensionality. Feedforward neural networks also have the theoretical property of being universal function approximators. This comes from the universal approximation theorem for neural networks, proved in Hornik et al. (1989), which states that a multilayer feedforward neural network with at least one hidden layer can approximate any Borel measurable function to a desired degree of accuracy, provided that sufficiently many hidden units are available.

We give a brief overview of these networks, although the reader is referred to Goodfellow et al. (2016) for a more comprehensive overview of deep learning.

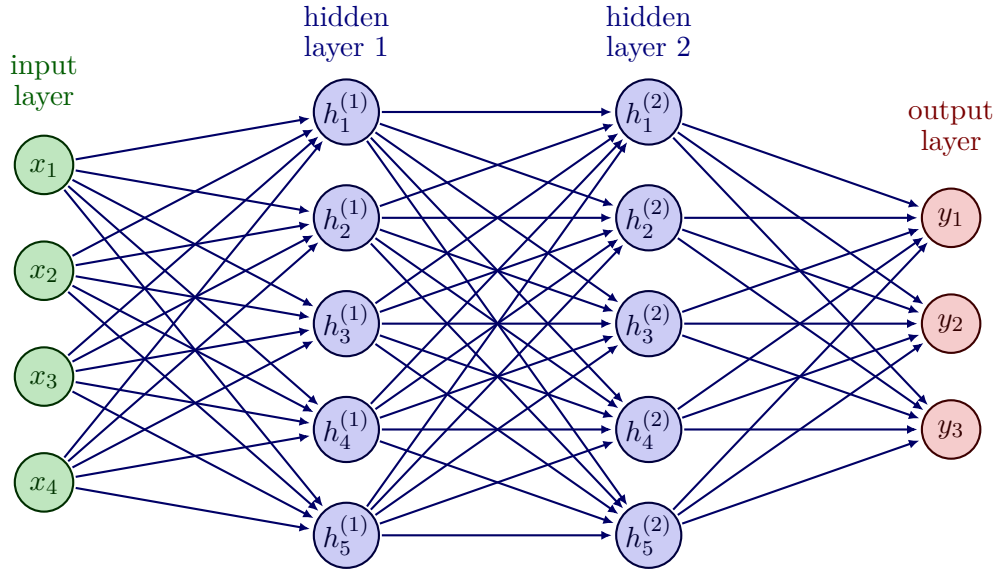


Figure A.3: A feedforward neural network with an input layer, two fully-connected hidden layers with five neurons per hidden layer, and an output layer. In a feedforward neural network, the information flow is unidirectional, which means that information flows from the input layer to the output layer without any feedback connections. Image adapted from Neutelings (2023).

A.3.2 Definition

Each layer of the neural network is a function $\phi : \mathbb{R}^{d_1} \rightarrow \mathbb{R}^{d_2}$ defined element-wise for $i = \{1, \dots, d_2\}$ by $\phi(x)_i = \sigma \left(b_i + \sum_{j=1}^{d_1} w_{i,j} x_j \right)$, where $\mathbf{w} \in \mathbb{R}^{d_2 \times d_1}$ and $\mathbf{b} \in \mathbb{R}^{d_2}$ are the trainable parameters for the neural network, called the weights and biases, respectively. σ is an *activation function* that transforms the summed weighted inputs.

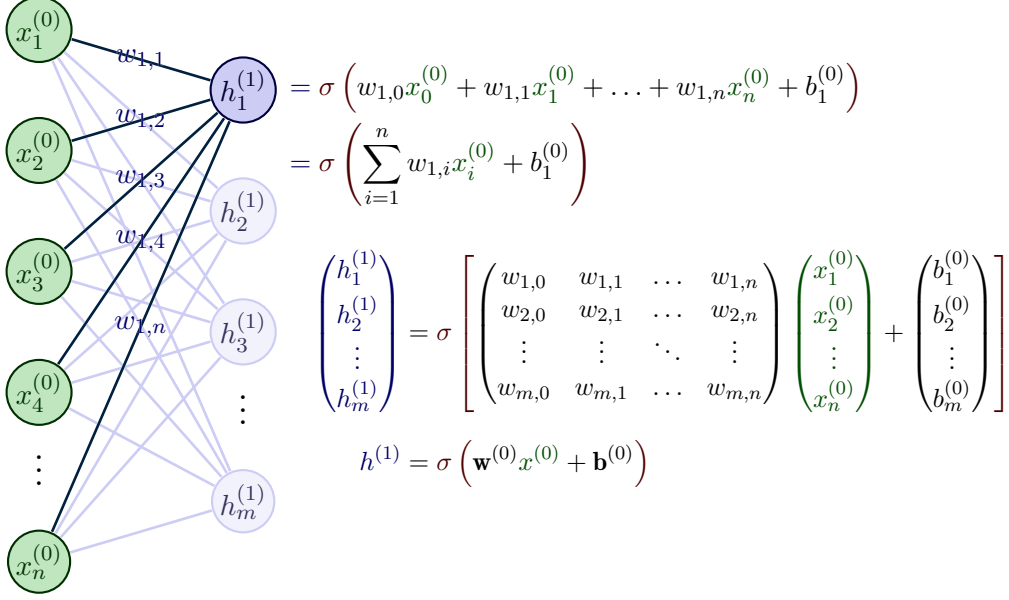


Figure A.4: An example of how the first hidden layer is computed as an activation function applied to the weighted sum of the inputs. Image adapted from Neutelings (2023).

A feedforward neural network is then defined by

$$m_\theta = \phi_l \circ \phi_{l-1} \circ \dots \circ \phi_0,$$

where \circ denotes the composition of functions, l is the number of layers and $\theta = (\mathbf{b}^0, \mathbf{w}^0, \dots, \mathbf{b}^l, \mathbf{w}^l)$ are the parameters of the neural network.

A.3.3 Activation functions

In this thesis, we rely mainly on the rectified linear unit (ReLU) activation function for the hidden layers. The ReLU function is defined as

$$\text{ReLU}(x) = \max(0, x).$$

The advantage of using ReLU is that it is computationally fast. In particular, it is fast to train the neural network using backpropagation as many gradients are 0. The ReLU activation function also works well empirically and it is often the default activation function used.

In this thesis, we also used the sigmoid activation function in the output layer, which is defined as

$$\text{sigmoid}(x) = \frac{1}{1 + \exp(-x)}.$$

The range of the sigmoid activation function is $[0, 1]$, which makes it a useful activation function for computing the weights of the adjacency matrix \mathbf{A} .

A.3.4 Backpropagation and stochastic gradient descent

We refer to the process of updating the parameters of the neural network to minimise a loss function as *training* the neural network. This is done by *backpropagation*, which describes the process where the input is passed through the network’s hidden layers and the output layer. In the output layer, the derivative of the loss function is computed with respect to the weights of the neural network. These weights are then updated using a gradient descent algorithm in order to minimise the loss function. This process is repeated for a fixed number of iterations, often stopping when the loss function has converged.

Algorithm 3: Training algorithm using gradient descent

Input: θ_0 : An initial parameterisation of the neural network
 N_{iter} : Number of iterations
 $(\beta_i)_{i=0,\dots,N_{\text{iter}}-1}$: Learning rates

```
1 for  $i = 1$  to  $N_{\text{iter}}$  do
2   | Compute  $\nabla_{\theta^i} J$ , the gradients of the loss with respect to the parameters,  $\theta^i$ 
3   | Update  $\theta^{i+1} = \theta^i - \beta_i \nabla_{\theta^i} J$ 
4 return  $\theta^{N_{\text{iter}}}$ 
```

There are various gradient descent methods that are used in practice. Batch gradient descent computes the gradient using the whole training data before updating the parameters. Stochastic gradient descent (SGD) computes the gradient for a single sample of the training data and updates the parameters. Mini-batch stochastic gradient partitions the training data into mini-batches, and computes the gradient for each mini-batch. We refer to an *epoch* as one complete pass through of the training data.

Stochastic gradient descent is a popular algorithm as it converges faster than batch gradient descent. This is because the gradient of a single sample is easier to calculate than the gradient of the training data. However, the convergence path is noisier due to the randomness of the impact of a single sample on the loss. Mini-batch stochastic gradient descent provides a middle-ground and is the preferred training method used in this thesis.

A.3.5 Initialisation of weights

The initialisation of the weights of the neural network can have a large impact on the convergence of the network. Unless otherwise specified, we use the Glorot uniform initialiser as described in Glorot and Bengio (2010), a popular initialisation method which helps control the stability of the gradients during the initial training steps.

A.3.6 Adam optimiser

In practice, the choice of the learning rate plays an important role in the convergence of the parameters of the neural network. The Adam optimisation algorithm is an extension of stochastic gradient descent introduced by Kingma and Ba (2015) that “computes individual adaptive learning rates for different parameters from estimates of first and second moments of the gradients. The

name Adam is derived from adaptive moment estimation”. Adam works well in practice and is the preferred optimisation method used in this thesis.

A.3.7 TensorFlow

We implement all our neural network algorithms in TensorFlow, a framework developed by Abadi et al. (2015) for training deep neural networks. We use TensorFlow version 2.10 and Keras, a Python interface for TensorFlow developed by Chollet et al. (2015). TensorFlow allows us to easily develop low-level algorithms for training our neural networks. In particular, the gradients of the loss function with respect to the weights are automatically computed, which makes the implementation of training the neural network straightforward. We also take advantage of TensorFlow’s XLA (Accelerated Linear Algebra) compiler, which results in large improvements in the speed of training. The Python codes that I developed for solving the interbank game and the network inference problem using TensorFlow are available on my GitHub at <https://github.com/HarryLi98/SDE-DeepRL>.

Appendix B

Additional material

B.1 Interbank model

B.1.1 Common noise

To see that the ρ describes the noise correlation between the banks, we can rewrite the dynamics in eq. (2.3) in matrix form:

$$dX_t = a(\bar{X}_t - X_t) dt + \Sigma dW_t,$$

where $X_t = (X_t^1, \dots, X_t^N)^T \in \mathbb{R}^N$, $W_t = (W_t^0, W_t^1, \dots, W_t^N)^T \in \mathbb{R}^{N+1}$ and

$$\Sigma = \begin{bmatrix} \sigma\rho & \sigma\sqrt{1-\rho^2} & 0 & \dots & 0 \\ \sigma\rho & 0 & \sigma\sqrt{1-\rho^2} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sigma\rho & 0 & 0 & \dots & \sigma\sqrt{1-\rho^2} \end{bmatrix} \in \mathbb{R}^{N \times (N+1)}.$$

The covariance matrix of the system is given by:

$$\Sigma\Sigma^T = \sigma^2 \begin{bmatrix} 1 & \rho^2 & \dots & \rho^2 \\ \rho^2 & 1 & \dots & \rho^2 \\ \vdots & \vdots & \ddots & \vdots \\ \rho^2 & \rho^2 & \dots & 1 \end{bmatrix}.$$

Since $\text{diag}(\Sigma\Sigma^T) = 1$, the correlation matrix coincides with the covariance matrix.

B.2 Interbank game

B.2.1 Analytical Solution

Carmona et al. (2015) derive the analytical solution of both the open-loop and closed-loop Nash equilibrium of the Linear-Quadratic interbank game. The details of the derivation are omitted

here, and we simply present the analytical solution of the closed-loop Nash equilibrium, which is given by:

$$\alpha_t^i = \left[q + \left(1 - \frac{1}{N} \right) \eta_t \right] (\bar{X}_t - X_t^i),$$

and the forward dynamics by:

$$dX_t^i = \left[a + q + \left(1 - \frac{1}{N} \right) \eta_t \right] (\bar{X}_t - X_t^i) dt + \sigma \left(\sqrt{1 - \rho^2} dW_t^i + \rho dW_t^0 \right)$$

with

$$\eta_t = \frac{- (\epsilon - q^2) \left(e^{(\delta^+ - \delta^-)(T-t)} - 1 \right) - c \left(\delta^+ e^{(\delta^+ - \delta^-)(T-t)} - \delta^- \right)}{(\delta^- e^{(\delta^+ - \delta^-)(T-t)} - \delta^+) - c \left(1 - \frac{1}{N^2} \right) \left(e^{(\delta^+ - \delta^-)(T-t)} - 1 \right)},$$

where we used the notation

$$\delta^\pm = -(a + q) \pm \sqrt{R},$$

with

$$R := (a + q)^2 + \left(1 - \frac{1}{N^2} \right) (\epsilon - q^2) > 0.$$

B.3 Network inference using neural networks

B.3.1 OLS Estimator

We can estimate the network topology using the ordinary least squares (OLS) estimator. Given L observations for the time series data for bank i , we obtain L equations with N unknowns, given in matrix form by

$$\mathbf{X}_i = \mathbf{G}_i \mathbf{A}_i + \boldsymbol{\varepsilon}_i,$$

where $\mathbf{X}_i \in \mathbb{R}^L$ are the L observations for bank i , $\mathbf{G}_i \in \mathbb{R}^{L \times N}$ is the matrix of observations for the N banks given by $\mathbf{G}_i = \mathbf{X} - \mathbf{X}^i$, where $\mathbf{X} = (\mathbf{X}_1, \dots, \mathbf{X}_N) \in \mathbb{R}^{L \times N}$, $\mathbf{A}_i \in \mathbb{R}^N$ is the vector of unknowns that correspond to the i th row of the adjacency matrix \mathbf{A} , and $\boldsymbol{\varepsilon}_i \in \mathbb{R}^L$ is the error term. We can estimate \mathbf{A}_i using the OLS estimator:

$$\hat{\mathbf{A}}_i = (\mathbf{G}_i^\top \mathbf{G}_i)^{-1} \mathbf{G}_i^\top \mathbf{X}_i,$$

provided that $\mathbf{G}_i^\top \mathbf{G}_i$ is invertible. If $L < N$, then $\mathbf{G}_i^\top \mathbf{G}_i$ is not invertible, and we cannot use the OLS estimator.

B.3.2 Adjacency matrices

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}$$

Figure B.1: The randomly generated adjacency matrix that represents the true network. \mathbf{A} is symmetric, and the diagonal elements are one.

$$\hat{\mathbf{A}} = \begin{bmatrix} 1 & 0.38 & 0.15 & 0.44 & 0.43 & 0.61 & 0.66 & 0.1 & 0.7 \\ 0.38 & 1 & 0.09 & 0.66 & 0 & 0.06 & 0.51 & 0.09 & 0.22 \\ 0.15 & 0.09 & 1 & 0.6 & 0.03 & 0.23 & 0.16 & 0 & 0.43 \\ 0.44 & 0.66 & 0.6 & 1 & 0 & 0.03 & 0.02 & 0 & 0.18 \\ 0.43 & 0 & 0.03 & 0 & 1 & 0.02 & 0.79 & 0.2 & 0.38 \\ 0.61 & 0.06 & 0.23 & 0.03 & 0.02 & 1 & 0.19 & 0 & 0.04 \\ 0.66 & 0.51 & 0.16 & 0.02 & 0.79 & 0.19 & 1 & 0.01 & 0.53 \\ 0.1 & 0.09 & 0 & 0 & 0.2 & 0 & 0.01 & 1 & 0.56 \\ 0.7 & 0.22 & 0.43 & 0.18 & 0.38 & 0.04 & 0.53 & 0.56 & 1 \end{bmatrix}$$

Figure B.2: The estimated network for the model with $\sigma = 0$, rounded to two decimal places. While the estimated network is different from the true network, it still produces a time series that is consistent with the observed time series. Symmetries in the data may mean that there is more than one solution to the problem of inferring the network.

$$\hat{\mathbf{A}} = \begin{bmatrix} 1 & 0.02 & 0.23 & 0.35 & 0.58 & 0.02 & 0.76 & 0.17 & 0.79 \\ 0.02 & 1 & 0.9 & 0.22 & 0.03 & 0.48 & 0.29 & 0 & 0.26 \\ 0.23 & 0.9 & 1 & 0.67 & 0.01 & 0.6 & 0.14 & 0.01 & 0.24 \\ 0.35 & 0.22 & 0.67 & 1 & 0 & 0.18 & 0.47 & 0.09 & 0.55 \\ 0.58 & 0.03 & 0.01 & 0 & 1 & 0.89 & 0.47 & 0.09 & 0.55 \\ 0.02 & 0.48 & 0.6 & 0.18 & 0.89 & 1 & 0.25 & 0 & 0.29 \\ 0.76 & 0.29 & 0.14 & 0.47 & 0.47 & 0.25 & 1 & 0.02 & 0.04 \\ 0.17 & 0 & 0.01 & 0.09 & 0.09 & 0 & 0.02 & 1 & 0.03 \\ 0.79 & 0.26 & 0.24 & 0.55 & 0.55 & 0.29 & 0.04 & 0.03 & 1 \end{bmatrix}$$

Figure B.3: The estimated network for the model with $\sigma = 0.1$, rounded to two decimal places.