Physica A 605 (2022) 128029

Contents lists available at ScienceDirect

Physica A

journal homepage: www.elsevier.com/locate/physa

Reducing systemic risk in a multi-layer network using reinforcement learning

Richard Le, Hyejin Ku *

Department of Mathematics and Statistics, York University, 4700 Keele Street, Toronto, ON M3J 1P3, Canada

ARTICLE INFO

Article history:
Received 31 May 2022
Received in revised form 5 July 2022
Available online 11 August 2022

Keywords:
Systemic risk
Reinforcement learning
Constraint DDPG
Multi-layer network
Network reorganization
DebtRank

ABSTRACT

This paper introduces a novel framework to assess and manage systemic risk in a multi-layer financial network by taking advantage of reinforcement learning (RL). The reduction of systemic risk in the financial network is achieved by applying the deep deterministic policy gradient algorithm (DDPG) to reorganize the interbank lending structure of the network into an orientation that better mitigates the spread of contagion. The reorganization procedure itself was constrained in order to preserve the balance sheet of every bank. To achieve this, we develop a constraint DDPG model consisting of a safety layer coupled with a linear mapping to satisfy the total borrowing and lending amounts of each bank. Moreover, we propose a new multi-layer DebtRank (DR) algorithm taking into account how contagion spreads from one layer to another. Testing against networks of varying size and depth, our DDPG agent was able to reduce systemic risk levels by significant amounts, suggesting the feasibility and utility of employing RL in managing systemic risk through aiding regulatory policy design. We observe an increase in sparsity and an increase in network dissimilarity between the different layers of the network after optimization.

1. Introduction

A key property of a financial network is its interconnectedness. This interconnectedness, however, is a mechanism for amplifying shocks and distress, leading to contagion and potentially resulting in catastrophic failure of the network. The risk of financial collapse due to the failure of some portion of the financial network leading to economic decline is called systemic risk. The importance of financial stability and systemic risk in the financial sector has been underlined after the global financial crisis, and the monitoring and regulation of systemic risk have become a major concern for regulators, governments, and financial institutions. The insights gained from the crisis include the importance of interconnectedness among financial institutions and markets and the necessity of adopting a system-wide view of stability and risk. One can get useful insights from analogous problems related to the large-scale (in)stability of systems with many interconnected components and feedback loops in other disciplines. It is important to understand the mechanisms underlying systemic risk and financial instability, metrics for identifying sources of systemic risk, and tools for monitoring these sources in practice.

A high level of systemic risk can have several effects on members of the financial system. These effects include impacts on sovereign credit ratings [1–3], impacts on hedge fund returns [4], and reducing the benefits of diversification in

* Corresponding author.
E-mail addresses: ler3@yorku.ca (R. Le), hku@yorku.ca (H. Ku).

https://doi.org/10.1016/j.physa.2022.128029
0378-4371/

portfolios [5,6]. In this paper, we explore how RL can be used to reorganize the connections of a multi-layer financial network in order to reduce the overall systemic risk in the network.

Complex networks provide a convenient representation of the financial system. Typically, the nodes in the network represent the banks or financial institutions, while the edges connecting the nodes represent the relationship between the financial actors. Such a natural representation of the financial system has spurred the development of models adopting this framework to study systemic risk in financial systems [7–13]. The structure of the network has been intimately tied to the levels of systemic risk present in the network [14–17]. This idea has introduced a line of study investigating how we can capitalize on the connection between systemic risk and the network structure to reduce systemic risk. Poledna and Thurner [18], Poledna et al. [19] implement a systemic risk tax incentive using agent-based modelling and observe a reduction in systemic risk in the self-organized network. Diem et al. [20] reduce systemic risk in a direct-exposure network using mixed-integer linear programming to reorganize the network. They also observe that the reorganization of the direct exposure network can yield lower levels of systemic risk when compared to Basel III-like equity increases. They highlight some network characteristics expressing low levels of systemic risk, suggesting that these characteristics should be taken into account when designing policies for tackling financial market stability. Another study by Pichler et al. [21] reorganizes an overlapping bond portfolio network, represented by a bipartite network, in a similar manner by framing the reorganization problem as an optimization problem.

In order to manage systemic risk, we need to be able to measure systemic risk. In a network setting, systemic risk can be measured by using network-based measures [7,22,23]. The seminal work by Eisenberg and Noe [7] introduces a clearing algorithm while also providing an estimate of the systemic risk based on the number of "waves" of defaults required for a firm to fail in the algorithm. The approach taken by Furfine [22] uses interbank payment data to simulate the knock-on effects of the failure of a single firm. These measures do not consider how distress prior to default can lead to contagion. The popular DR measure by Battiston et al. [24] tackles this problem by taking into account the build-up and propagation of distress and its effect on the equity of banks in the network. The DR algorithm measures the systemic risk of financial institutions by propagating their initial distress through the network and calculating the induced loss as a result. This algorithm was extended by Bardoscia et al. [25] to allow banks to propagate distress more than once and by Silva et al. [26] to consider a feedback mechanism between the real and financial sector.

In reality, there are many different types of financial products and contracts. The failure of an institution to honour one type of contract is not always felt in isolation. In fact, Montagna and Kok [27], Poledna et al. [28], Cuba et al. [29] find that only considering the systemic risk in single-layer networks severely underestimates the total systemic risk of a financial system. Poledna et al. [28] extend the DR measure to the multi-layer case, allowing for the comparison of systemic risk between layers as well as the systemic risk of the combined network of projected layers. Poledna et al. [30] modify the DR algorithm to account for overlapping portfolios in bipartite networks, and hence account for indirect exposures. Cao et al. [31] extend the DR algorithm to the multi-layer case, accounting for investments of debt and equity between financial institutions.

Recently, there have been efforts to apply machine learning techniques to improve systemic risk assessment. Li et al. [32] use support vector machine to predict systemic risk in the Chinese banking system. Cerchiello et al. [33] use financial twitter and market data in predicting when shocks to the financial system might occur. Using algorithmic text analysis, Nyman et al. [34] use financial reports and news articles to measure relative sentiment shifts based off excitement and anxiety summary statistics, finding potential in predicting increases in distress in the financial system. Most recently, So et al. [35] proposed the use of Latent Dirichlet Allocation on financial news article data, allowing real-time prediction of systemic risk. For a more detailed review of machine learning applications in systemic risk, see the survey by Kou et al. [36].

Although there are a number of studies on machine learning applications in systemic risk, currently, only a handful of studies have adapted RL techniques to a financial network setting. Liu et al. [37] make use of a multi-agent model based on temporal difference RL to replicate lending and borrowing dynamics. In particular, RL here is used to help decide each banks' counterparties. Their work demonstrates how the risk preferences of individual banks can assemble networks that are less at risk for contagion. Although not modifying the interbank relationships themselves, Petrone et al. [38] proposes a framework in which an RL agent provides capital investments for different banks in the network, replicating the capital injections given by the government to increase the resilience of banks and minimize losses in the network.

In our study, we take a different approach to reorganizing the financial network by using RL. The main goal of this paper is to construct a RL framework to minimize systemic risk in a multi-layer financial network. In pursuing this goal, we made the following contributions. First of all, we develop the constraint DDPG algorithm to reorganize the interbank lending structure of a multi-layer network by modifying the classical DDPG algorithm, proposed by Lillicrap et al. [39]. To minimize the effects of network reorganization on the balance sheets of each bank in the network, we incorporate a safety layer inspired by Dalal et al. [40]. The flexibility that is offered by RL allows us to easily extend the optimization procedure from the single layer case to the multi-layer case circumventing the technical optimization challenges noted by Diem et al. [20]. Second, we propose a new multi-layer DR to measure systemic risk in our networks. Both optimization procedures in Diem et al. [20] and Pichler et al. [21] were done by minimizing the total direct impact, an approximation of the DR. Using RL we directly incorporate the DR measure into the model's objective via the reward function. The types of assets used in this paper will have different maturities. Therefore, to account for how contagion might spread in a multi-layer network of loans with differing maturities, we propose a modified DR measure. We further highlight the versatility of using RL by

considering preferential reduction in DR through the modification of the reward function to account for highly leveraged banks.

The remainder of this paper is organized into 6 sections. In Section 2 we outline how we model our multi-layer complex network and present both the conventional DR and our proposed multi-layer DR measures. In Section 3 we detail the implementation of our RL agent in the context of reducing systemic risk in a multi-layer complex network environment. In Section 4 we outline how to constrain the action of the RL agent to preserve specific properties of the complex network and also present the experimental details for the single-layer and multi-layer case along with the parameters and hyperparameters used in our model. In Section 5 we present the results and discussion. Finally, in Section 6 we conclude the study and present some possible extensions to our work.

2. Model

2.1. Multi-layer complex networks

We modelled the interbank liability network as a multi-layer weighted directed graph with $\mathcal{M} = \{G, Y\}$ where $G = \{(V, E_\alpha) \mid \alpha \in \{1, 2, \ldots, M\}\}$ is the set of graphs in the multi-layer network and $Y = \{\alpha \mid \alpha \in \{1, 2, \ldots, M\}\}$ is the index set for the different layers of the multi-layer network. The set of nodes in the multi-layer network is denoted by $V = \{i \mid i \in \{1, 2, \ldots, N\}\}$. $E_\alpha = \{(i, j) \mid i, j \in V, i \neq j\}$ denotes the set of edges connecting nodes V in layer α. Note that each layer contains the same set of nodes and the only difference between the layers is the topology of the edges.

In the context of financial networks, each node in the graph will represent a bank. The directed edge from bank i to j in layer α represents the loan of bank i to bank j in layer α. This lending amount is denoted by L_{ij}^α. In the case that bank j defaults, L_{ij}^α also represents the impact of bank j on bank i as this is the amount lost by bank j. We define the adjacency matrix of the graph as $\mathbf{Q}$ and denote its elements using

$$Q_{ij}^\alpha = \begin{cases} 1 & \text{if bank } i \text{ lends to bank } j \text{ in layer } \alpha \\ 0 & \text{Otherwise} \end{cases} \tag{1}$$

Using the methods described in Li et al. [10] and Maeno et al. [41], we can simulate a liability interbank network $\mathbf{L}^\alpha$ with parameters $N, A, \theta^\alpha, r, \beta$, and γ representing the total number of banks, the total asset in the network, the interbank loan ratio, the networks' degree heterogeneity, the cash deposit ratio, and the equity capital ratio, respectively. Then the lending amounts that appear on the balance sheet are calculated as

$$L_{ij}^\alpha = \frac{Q_{ij}^\alpha (k_{\text{out},i}^\alpha k_{\text{in},i}^\alpha)^r}{\sum_{i=1}^N \sum_{j=1}^N Q_{ij}^\alpha (k_{\text{out},i}^\alpha k_{\text{in},i}^\alpha)^r} \theta^\alpha A \tag{2}$$

where $k_{\text{out},i}^\alpha$ and $k_{\text{in},i}^\alpha$ are the outgoing degree and incoming degree of the ith bank in the α layer, respectively. Once the lending amounts are defined for every bank, we can then calculate the rest of the balance sheet. To calculate the balance sheets of the banks, we define the following, let $\theta = \sum_{\alpha=1}^M \theta^\alpha$, $l_i = \sum_{\alpha=1}^M \sum_{j=1}^N L_{ij}^\alpha$, $b_i = \sum_{\alpha=1}^M \sum_{i=1}^N L_{ij}^\alpha$, and $\text{TL} = \sum_{i=1}^N l_i$ where θ is the total proportion of the assets used for lending, l_i is the total lending amount of bank i, b_i is the total borrowing of bank i, and TL is the total amount used for lending in the network respectively. The balance sheet can then be calculated using the following set of equations

$$o_i = \max(b_i - l_i, 0) + \left[[(1-\theta) - \beta(1-\gamma) + \beta\theta]A - \sum_{i=1}^N \max(b_i - l_i, 0)\right] l_i/\text{TL} \tag{3}$$

$$e_i = \frac{\gamma(l_i + o_i) - \beta\gamma b_i}{1 + \beta\gamma - \beta} \tag{4}$$

$$d_i = \frac{(1-\gamma)(l_i + o_i) - \beta b_i}{1 + \beta\gamma - \beta} \tag{5}$$

$$c_i = \frac{\beta(1-\gamma)(l_i + o_i) - \beta b_i}{1 + \beta\gamma - \beta} \tag{6}$$

where o_i, e_i, d_i, and c_i are the other assets, equity, deposits, and cash of bank i. Then the total assets on bank i's balance sheet is $a_i = c_i + l_i + o_i$ and by basic accounting principles $a_i = d_i + b_i + e_i$. The simulated balance sheet for each bank in the network can be found in Fig. 1.

2.2. DebtRank

We will be measuring the systemic risk contribution of banks in the complex network in terms of their DR. The algorithm used to calculate the DR was first introduced by Battiston et al. [24] and was extended to multi-layer networks by Poledna et al. [28]. It should be noted however that Poledna et al. [28] do not take into account how distress might propagate between the different layers. In our study, we introduce our own mechanism for contagion to spread between

Assets	Liabilities & Equity
Interbank loans, $\sum_{\alpha=1}^{M} l_i^{\alpha}$	Interbank borrowings, $\sum_{\alpha=1}^{M} b_i^{\alpha}$
Cash, c_i	Deposits, d_i
Other Assets, o_i	Equity, e_i

Fig. 1. Balance sheet of the i-th bank for all layers of the multi-layer network.

the different layers of the multi-layer network. For the completeness of the paper, we first provide a brief introduction of the conventional DR for a single-layer network. The impact of bank i on bank j can be defined by

$$W_{ij} = \min\left[1, \frac{L_{ji}}{e_j}\right] \tag{7}$$

where L_{ji} is the lending amount from bank j to bank i and e_j is the equity of bank j. If $L_{ji} < e_j$ then the impact of bank i on bank j is L_{ji}/e_j. Therefore, given an adequate level of e_j, the impact of bank i on bank j can be mitigated by the buffer e_j. Given a sufficiently low e_j, the impact of bank i on bank j could lead to the default of bank j. For each bank, we define two state variables. Let $h_i \in [0, 1]$ represent the level of distress of bank i and $s_i \in \{U, D, I\}$ be a discrete variable taking three possible states U, D, and I, representing the undistressed, distressed, and inactive states, respectively. The dynamics of h_i follows

$$h_i(t) = \min\left\{1, h_i(t-1) + \sum_{j|s_j(t-1)=D} W_{ji} h_j(t-1)\right\} \tag{8}$$

$$s_i(t) = \begin{cases} D & \text{if } h_i(t) > 0; s_i(t-1) \neq I \\ I & \text{if } s_i(t-1) = D \\ s_i(t-1) & \text{otherwise} \end{cases}$$

where h_i is calculated for all i at each time step. The DR of a bank i is calculated after some finite time T has passed or once all the banks are in state U or I. The DR R_i of a bank i can be calculated as

$$R_i = \sum_j h_j(T) v_j - h_i(1) v_i \tag{9}$$

where v_i is the relative economic value of each bank defined as

$$v_i = \frac{l_i}{\sum_{k=1}^{N} l_k} \quad \forall i \in V \tag{10}$$

The relative economic value of each bank is the contribution of bank i's lending relative to the entire interbank network. When a bank is in distress, some or all of its value is lost (the bank is considered in default if all of its value is lost). Therefore, the DR can be interpreted as the relative economic value of the network that is potentially lost due to the distress caused by bank i propagating through the network. Given that the DR is dependent only on L_{ij} and e_j, we define $R_i(\mathbf{L}, e) = R_i$ for a given liability network $\mathbf{L}$ and vector e whose entries are the equities of the respective banks.

2.3. DebtRank accounting for differing maturity of loans

In order to take into account how distress in one layer propagates to other layers, we use the index set $Y = \{\alpha \mid \alpha \in \{1, 2, 3, \ldots, M\}\}$ and let the first layer, $\alpha = 1$, represent the interbank liability network of loans with the shortest maturities and the last layer, $\alpha = M$, represents the interbank liability network of loans with the longest maturities. Therefore L_{ij}^1 represents the interbank liability matrix of the loans with the shortest maturities and L_{ij}^M represents the interbank liability matrix of the loans with the longest maturities. It is assumed that any distress experienced impacts the short-term liability before the long-term liability. Then the impact matrix for layer $\alpha > 1$ is given by

$$W_{ij}^{\alpha} = \begin{cases} \dfrac{L_{ji}^{\alpha}}{\max\left(L_{ji}^{\alpha},\ e_j - \sum_{\kappa=1}^{\alpha-1} \sum_{p=1}^{N} L_{jp}^{\kappa} h_p^{\kappa}(T)\right)} & \text{if } L_{ji}^{\alpha} > 0 \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

and for $\alpha = 1$, the impact matrix is given by

$$W_{ij}^{1} = \min\left(1, \frac{L_{ji}^{1}}{e_j}\right) \tag{12}$$

where $h_i^{\kappa}(T)$ is the distress that bank i experiences in layer κ at time T. For $\alpha > 1$ the equity is reduced by the lending amount affected by the distress in the previous layers. The DR of the first layer is calculated using the usual initial conditions of the conventional DR as shown in Eq. (12), and the dynamics for layers $\alpha > 1$ follows similarly to the conventional DR algorithm. In other words, we let $h_i^1 \in [0, 1]$ represent the level of distress of bank i resulting from the initial distress in the first layer and $s_i^1 \in \{U, D, I\}$ be the discrete state variable where U, D, and I represent the undistressed, distressed, and inactive state, respectively. Then, for each layer, the dynamics of h_i^{α} follows

$$h_i^{\alpha}(t) = \min\left\{1, h_i^{\alpha}(t-1) + \sum_{j|s_j^{\alpha}(t-1)=D} W_{ji}^{\alpha} h_j^{\alpha}(t-1)\right\} \tag{13}$$

$$s_i^{\alpha}(t) = \begin{cases} D & \text{if } h_i^{\alpha}(t) > 0; s_i^{\alpha}(t-1) \neq I \\ I & \text{if } s_i^{\alpha}(t-1) = D \\ s_i^{\alpha}(t-1) & \text{otherwise} \end{cases}$$

where h_i^{α} is calculated for all i at every time step. The calculation for layer α is stopped as soon as all the banks in layer α is in the state U or I after some finite time T has passed. The initial distress and state for all nodes i in layers $\alpha > 1$ are set according to the following equations

$$h_i^{\alpha}(0) = h_i^{\alpha-1}(T) \tag{14}$$

$$s_i^{\alpha}(0) = \begin{cases} D & \text{if } h_i^{\alpha-1}(T) > 0 \\ U & \text{otherwise.} \end{cases} \tag{15}$$

Therefore, any node that was distressed in the previous layer will maintain the same levels of distress at time 0 in the next layer.[1] Furthermore, any nodes that become inactive after becoming distressed will have their state set to distressed or undistressed and will be able to propagate distress again in subsequent layers.

The DR of each layer α is calculated after some finite time T has passed or once all banks are in state U or I. Again, we define the total amount loaned by a bank i in layer α as $l_i^{\alpha} = \sum_{j=1}^{N} L_{ij}^{\alpha}$. Then DR of bank i is calculated as

$$R_i^{\alpha} = \begin{cases} \sum_j h_j^{\alpha}(T) v_j^{\alpha} - h_i^{\alpha}(1) v_i^{\alpha} & \text{if } \alpha = 1 \\ \sum_j h_j^{\alpha}(T) v_j^{\alpha} & \text{if } \alpha > 1 \end{cases} \tag{16}$$

where v_i^{α} is the relative economic value of each node i in layer α

$$v_i^{\alpha} = \frac{l_i^{\alpha}}{\sum_{k=1}^{N} l_k^{\alpha}} \quad \forall i \in V. \tag{17}$$

Note that the DR of each layer takes into account the distress from the previous layers, and therefore the DR of the multi-layered complex network will be greater than the conventional DR measure. Given that the DR is dependent only on L_{ij}^{α} and e_j, we define $R_i(\mathbf{L}^{\alpha}, e) = R_i^{\alpha}$ for a given liability network $\mathbf{L}^{\alpha}$ in layer α and vector e whose entries are the equities of the respective banks. The total DR of the multi-layer network is then calculated by

$$R(\mathbf{L}, e) = \sum_{\alpha=1}^{M} \sum_{i=1}^{N} v^{\alpha} R_i(\mathbf{L}^{\alpha}, e) \tag{18}$$

where

$$v^{\alpha} = \frac{\sum_{i=1}^{N} l_i^{\alpha}}{\sum_{\alpha=1}^{M} \sum_{i=1}^{N} l_i^{\alpha}}. \tag{19}$$

2.4. DebtRank weighted by leverage and credit risk

The DR measures the economic value lost due to the spread of the distress from a single bank. This measure does not provide insight into how likely a bank is to default. Credit risk is the measure of the likelihood that a bank will default on a debt obligation. One might be more concerned about a bank with high DR and high credit risk than a bank with a low DR but high credit risk, as the impact on the network due to the failure of the first bank is far greater than impact on the network due to the failure of the second bank. After generating the balance sheet of the banks in the complex network, we can use the leverage ratio as a proxy of the bank's credit risk. There are several different leverage ratios that are commonly used in finance. We will use debt-to-assets ratio k_i for a bank i as defined below

$$k_i = \frac{d_i + b_i}{c_i + l_i + o_i}. \tag{20}$$

[1] Eq. (14) can be modified to include a recovery rate to allow banks to reduce the level of distress that is transferred to the next layer.

This measure will be used to modify the objective of the RL agent to preferentially reduce the systemic risk of higher leveraged banks. To implement this desired behaviour we weight the individual DR of the banks by their respective level of credit risk using a credit weight function $w(k_i)$ dependent on the leverage ratio of the bank. The credit weighted DR is then

$$R^w(\mathbf{L}, e, k) = \sum_{\alpha=1}^{M}\sum_{i=1}^{N} w(k_i)v^\alpha R_i(\mathbf{L}^\alpha, e) \tag{21}$$

where k is the vector of the leverage ratios of each bank. The degree of importance placed on the level of credit risk when reducing systemic risk can be changed by modifying the form of $w(k_i)$. An alternative estimate of the credit risk can also be used instead of the defined value of k_i in Eq. (20).

3. Reinforcement learning

In our study, we take a RL approach to reorganizing the financial network. Since the DR reduction process can be constantly changed, we require an off-policy agent that maps a high dimensional state space to a high dimensional continuous action space. So we will adopt the DDPG algorithm. DDPG, proposed in Lillicrap et al. [39], is an actor–critic based deep RL algorithm that has made remarkable achievements in the financial perspective. It uses a neural network as a Q-function approximator. To address the relatively unstable learned action function, they propose the use of a replay buffer and soft target updates to improve convergence to the optimal policy.

The classical DDPG algorithm has been developed by considering a Markov decision process with a state space $\mathcal{S}$, action space $\mathcal{A}$, transition dynamics $p(s_{t+1} \mid s_t, a_t)$, and reward function r. The return from a state is defined as the sum of discounted future reward

$$\mathrm{R}_t = \sum_{i=t}^{T} \gamma^{(i-t)} r(s_i, a_i) \tag{22}$$

where $\gamma \in [0, 1]$ is the discount factor, $s_i \in \mathcal{S}$ and $a_i \in \mathcal{A}$ are the observation and the agent's action, respectively. The state–action value is defined by

$$Q^\mu(s, a) = \mathbb{E}_{r_{i\geq t}, s_{i>t}\sim E, a_{i>t}\sim\mu}[\mathrm{R}_t | s_t, a_t] \tag{23}$$

and we use the recursive Bellman equation

$$Q^\mu(s_t, a_t) = \mathbb{E}_{r_t, s_{t+1}\sim E}[r(s_t, a_t) + \gamma Q^\mu(s_{t+1}, \mu(s_{t+1}))] \tag{24}$$

where $r_t, s_{t+1} \sim E$ indicates that the current reward and the future state are sampled from the environment. The parametrized actor function $\mu(s|\theta^\mu)$ maps the states $\mathcal{S}$ to action $\mathcal{A}$. The Q-function will be approximated by the critic by minimizing the loss

$$L(\theta^Q) = \mathbb{E}_{(s_t, a_t, r_t, s_{t+1})\sim\mathcal{D}}[(Q(s_t, a_t \mid \theta^Q) - y_t)^2] \tag{25}$$

$$y_t = r(s_t, a_t) + \gamma Q'(s_{t+1}, \mu'(s_{t+1} \mid \theta^{\mu'}) \mid \theta^{Q'}) \tag{26}$$

where $\mathcal{D}$ is the replay buffer that stores the transitions of the DDPG agent and $\mu'(s \mid \theta^{\mu'})$ and $Q'(s, a \mid \theta^{Q'})$ are the target actor and critic networks, respectively. The weights of the target networks are slowly updated using the learned networks' weights. The purpose of the target networks is to improve the stability of learning. The policy $\mu : \mathcal{S} \to \mathcal{A}$ of the agent is learned using the actor network. We train the actor network by maximizing the expected return J with respect to θ^μ

$$J = \mathbb{E}_{s_t\sim\mathcal{D}}[Q(s, a \mid \theta^Q) \mid_{s=s_t, a=\mu(s_t|\theta^\mu)}]. \tag{27}$$

The actor network is updated with the policy gradient using the results from Silver et al. [42]

$$\nabla_{\theta^\mu} J = \mathbb{E}_{s_t\sim\mathcal{D}}[\nabla_a Q(s, a \mid \theta^Q) \mid_{s=s_t, a=\mu(s_t)} \nabla_{\theta^\mu}\mu(s \mid \theta^\mu) \mid_{s=s_t}]. \tag{28}$$

In the context of a complex network of banks, we have a single agent that assigns different amounts of lending to each bank in the network. In our problem setting we wish to reward the agent every time a network configuration results in a lower overall DR. In the following section, we will express our problem setting in an RL framework.

3.1. The observation space

The environment consists of N financial institutions or banks. We set M different types of lending relationships the banks can establish with one another. In our environment, each layer represents different maturity lengths of loans. Therefore, there exist $M(N^2 - N)$ lending relationships that the agent can assign to form the complex network. Some examples of different relationships include deposits and loans, security cross-holdings, derivatives, and foreign exchange, and loans with differing maturities [10,28]. Every bank in the environment is given a balance sheet as described in

Section 2. The objective of the agent is to find the network configuration with the least amount of systemic risk measured using Eq. (16). In our implementation, we let the observation of the agent consist of the vectorization of the interbank liability network. The vectorization of the liability matrix $\mathbf{L}$ is defined by

$$\text{vec}(\mathbf{L}) = (L_{11}, \ldots, L_{1N}, L_{21}, \ldots, L_{2N}, \ldots, L_{N1}, \ldots, L_{NN}). \tag{29}$$

Therefore the observation, s_t, of the DDPG agent is given by

$$s_t = \{\text{vec}(\mathbf{L}^\alpha(t)) | \alpha = 1, 2, \ldots, M\} \tag{30}$$

3.2. The action space

Our DDPG agent will interact with the environment by modifying lending amounts of each bank in the network based on the observation s_t. The interbank liability network at time t is denoted $\mathbf{L}(t)$. The action provided by the RL agent will be denoted by $\Delta\mathbf{L}(t)$. The quantity $\Delta\mathbf{L}^\alpha(t)$ is a modification to the current lending network. Through this action a new interbank lending network $\mathbf{L}^\alpha(t+1)$ will be constructed. Therefore, the new interbank lending network is given by

$$\mathbf{L}^\alpha(t+1) = \mathbf{L}^\alpha(t) + \Delta\mathbf{L}^\alpha(t) \tag{31}$$

After the agent acts on the environment and modifies the complex network, we wish to conserve the total borrowing and total lending amounts of each bank in the network. Additionally, we require the new lending amounts of each bank to be non-negative. Let the lending relationships at time t be given by $\mathbf{L}^\alpha(t)$. Then $l_i^\alpha(t) = \sum_{i=1}^N L_{ij}^\alpha(t)$ and $b_i^\alpha(t) = \sum_{j=1}^N L_{ji}^\alpha(t)$ is the total α-type lending and borrowing amount of the ith bank, respectively, at time t. Then we wish to find an $\mathbf{L}^\alpha(t)$ where the following constraints are satisfied

$$\sum_{j=1}^N L_{ij}^\alpha(t) = l_i^\alpha(0) \quad \forall i \in V, \alpha \in Y, t \geq 0 \tag{32}$$

$$\sum_{j=1}^N L_{ji}^\alpha(t) = b_i^\alpha(0) \quad \forall i \in V, \alpha \in Y, t \geq 0 \tag{33}$$

$$L_{ij}^\alpha(t) \geq 0 \quad \forall i \in V, \alpha \in Y, t \geq 0 \tag{34}$$

In order to construct the action of the RL agent, we will be using the framework outlined in Section 4.1 where we outline how to satisfy the lending and borrowing constraints by using a linear transformation, and in Section 4.2 where we outline how to satisfy the non-negativity constraints by introducing a safety layer using quadratic programming (QP). An overview of our constraint DDPG structure and its interaction with the safety layer can be found in Fig. 2.

3.3. Reward and episode termination

The objective of our problem is to minimize the systemic risk with respect to the multi-layer DR. To do so, we intend to reward the agent every episode when the DR is reduced. Here we define the reward function

$$r(s, a) = \max\left(1 - \lambda \frac{R(\mathbf{L}(t+1), e)}{R(\mathbf{L}(t), e)}, 0\right) \tag{35}$$

where $\lambda \in \mathbb{R}$. In this way, the agent is given a reward if the DR in the next step is lower than the previous step's DR. The factor λ can be used to set a threshold on how low the DR must be before the agent is given a positive reward. In our experiments we set $\lambda = 1$. The environment is also designed such that the episode ends if the DR achieved at time $t+1$ is higher than the DR at time t. Comparing the DR at time $t+1$ to the DR at time t instead of at time $t = 0$ has the added benefit that the DR measured at the end of an episode is the lowest DR achieved in that episode.

Incorporating the credit risk weighted DR into the reward function results in

$$r^w(s, a, k) = \max\left(1 - \lambda \frac{R^w(\mathbf{L}(t+1), e, k)}{R^w(\mathbf{L}(t), e, k)}, 0\right). \tag{36}$$

Therefore, the RL agent will be rewarded more when banks with a large leverage ratio (i.e., more risky in terms of credit risk) have their DR reduced. In this way, we incentivize reducing the DR of a bank with a high leverage ratio over reducing the DR of a bank with a lower leverage ratio.

4. Proposed approaches

The classical DDPG agent cannot be directly applied to our problem of reorganizing the multi-layer complex network as there are a number of properties that we wish to preserve.[2] To preserve the operational well-being of the banks in the

[2] The code (and data) in this paper is posted on https://github.com/PencilKit/Reducing-Systemic-Risk-with-DDPG.

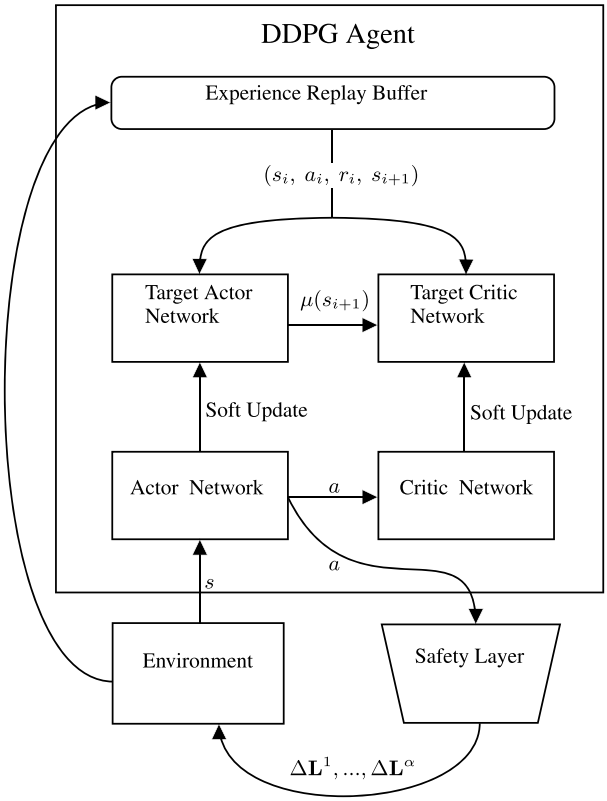


Fig. 2. A diagram of the constraint DDPG architecture interacting with the safety layer and environment.

Table 1
Parameter settings for the DDPG agent.

Hyperparameter	Value	Description
actor lr	3×10^{-5}	Actor learning rate
critic lr	3×10^{-4}	Critic learning rate
γ	0.80	Discount factor
τ	0.001	Factor for the soft update of target networks
N_{mini}	256	Batch size
$\mathcal{D}$	750	Replay buffer size
T	50	Maximum number of steps per episode

network, we require that the total lending and borrowing amounts on the stylized balance sheets to be conserved after reorganization. This idea is expressed through constraints (32) and (33). Additionally, after reorganization our model does not allow for negative lending. This idea is expressed through constraint (34). These constraints are achieved by using the transformation outlined in Section 4.1 and the safety layer presented in Section 4.2. The parameters and hyperparameters used in the construction of the complex multi-layer network and our DDPG agent, respectively, are outlined in Table 1.

4.1. Lending and borrowing constraints

This section will describe how we can modify the network without violating the lending and borrowing constraint. Note that the α and t in the notation are dropped in this section. This is because the framework outlined in this section is independent of the layer in the multi-complex network and the time when the DDPG agent acts in the environment. We define $\Delta\mathbf{L}$ as the change in the liability network and the new network configuration as $\mathbf{L}' = \mathbf{L} + \Delta\mathbf{L}$. We wish to find a mapping for the actions generated by the policy to $\Delta\mathbf{L}$. For a given liability matrix $\mathbf{L}$ we note that $L_{ij} = 0$ for $i = j$, therefore, we are only concerned with finding values of ΔL_{ij} where $i \neq j$, the off-diagonal elements of $\Delta\mathbf{L}$. For a matrix $\mathbf{A}$ of size $N \times N$ we define offdiag($\mathbf{A}$) to be the vector of size $N(N-1)$ containing the off-diagonal elements of $\mathbf{A}$.

In order to modify the interbank liability network while satisfying constraints (32) and (33), we require that $\Delta\mathbf{L}$ satisfy the following constraints

$$\sum_{j=1}^{N} \Delta L_{ij} = 0 \quad \forall i \in V \tag{37}$$

$$\sum_{j=1}^{N} \Delta L_{ji} = 0 \quad \forall i \in V. \tag{38}$$

To accomplish this, we solve the following homogeneous system of linear equations

$$\mathbf{Dx} = \mathbf{0} \tag{39}$$

where we define the solution vector by

$$\mathbf{x} = \text{offdiag}(\Delta\mathbf{L}) \tag{40}$$

$$\begin{aligned} =&(\Delta L_{12}, \Delta L_{13}, \ldots, \Delta L_{1N}, \\ &\Delta L_{21}, \Delta L_{23}, \ldots, \Delta L_{2N}, \ldots, \\ &\Delta L_{N1}, \Delta L_{N2}, \ldots, \Delta L_{N,N-1}) \end{aligned} \tag{41}$$

We then define a constraint matrix $\mathbf{D}$ of size $2N \times N(N-1)$ by

$$\mathbf{D} = \begin{bmatrix} \mathbf{C}_1 & \mathbf{C}_2 & \ldots & \mathbf{C}_N \\ \mathbf{J}_1 & \mathbf{J}_2 & \ldots & \mathbf{J}_{N-1} \end{bmatrix} \tag{42}$$

where $\mathbf{C}_n$ are the sub-matrices of size $N \times (N-1)$ for $1 \leq n \leq N$ whose entries are equal to 1 in the nth row and 0 in all other rows. $\mathbf{J}_n$ are the sub-matrices of size $N \times N$ defined by the following recursion

$$\mathbf{J}_1 = \mathbf{I}, \tag{43}$$

$$\mathbf{J}_{n+1} = \mathbf{E}_{(N-1),N} \ldots \mathbf{E}_{23}\mathbf{E}_{12}\mathbf{J}_n \tag{44}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{E}_{ij}$ is the elementary matrix corresponding to the column switching transformation between columns i and j. The sub-matrices $\mathbf{C_n}$ in this system constrain each row of $\Delta\mathbf{L}$ to sum to zero while the sub-matrices $\mathbf{J}_n$ constrains each column of $\Delta\mathbf{L}$. An example of the structure of the system for $N = 4$ is presented below

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta L_{12} \\ \Delta L_{13} \\ \Delta L_{14} \\ \Delta L_{21} \\ \Delta L_{23} \\ \Delta L_{24} \\ \Delta L_{31} \\ \Delta L_{32} \\ \Delta L_{34} \\ \Delta L_{41} \\ \Delta L_{42} \\ \Delta L_{43} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{45}$$

For $N \geq 3$, we have $2N \leq N(N-1)$ so the homogeneous system (39) has infinitely many solutions and $\mathbf{x} \in \text{null}(\mathbf{D})$. This null space is useful because the solutions here satisfy Eqs. (37) and (38).

In order for the DDPG agent to make a choice of $\Delta\mathbf{L}$, we will have it solve system (39). To accomplish this, we express the vector (40) as a linear combination of the basis vectors of the null space of $\mathbf{D}$. That is, let $\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_d$ be the basis vectors of null($\mathbf{D}$), then all solutions to system (39) are given by

$$\mathbf{Ku} = \mathbf{x} \tag{46}$$

where $\mathbf{K}$ is the basis matrix and $\mathbf{u} = (u_1, u_2, \ldots, u_d)$ is a vector of size d and $u_i \in \mathbb{R}$ for $i = 1, 2, \ldots, d$.

For each step in the environment the actor network will need to provide a set of vectors $\mathbf{u}$ for each layer. The action space is therefore $\mathcal{A} = \mathbb{R}^{Md}$ and the output of the actor network is then

$$\mu(s) = (u_1, u_2, \ldots, u_{(Md)}). \tag{47}$$

We can then partition the elements of vector (47) into M vectors of length d to be applied to the respective layers of the complex network. We will use

$$\mathbf{u}^{\alpha} = (u_{(\alpha-1)d+1}, u_{(\alpha-1)d+2}, \ldots, u_{\alpha d}) \tag{48}$$

to calculate the non-diagonal values of $\Delta\mathbf{L}^{\alpha}$ using Eq. (46).

Although a more intuitive approach might be to have the DDPG agent directly calculate $\Delta\mathbf{L}$, the action space would then be $\mathbb{R}^{M(N^2)}$. However, by using the basis matrix $\mathbf{K}$ as described above we can reduce the dimension of the action space.

Theorem 4.1.1. *Let* $\mathbf{D}$ *be the constraint matrix as described by Eq. (42) for a liability network of size* $N \times N$ *where* $N \in \mathbb{N}$ *such that* $N \geq 3$. *Then the dimension of the action space for a single layer network is reduced by* $2N-1$, *from* $N(N-1)$ *to* $N^2 - 3N + 1$.

Proof. Given a constraint matrix $\mathbf{D}$ as described by Eq. (42), let D_i be the ith row of matrix $\mathbf{D}$. We note that the first row can be written as the following linear combination

$$D_1 = \sum_{i=N+1}^{2N} D_i - \sum_{i=2}^{N} D_i \tag{49}$$

and so, D_1 is linearly dependent and can be made into a zero vector by subtracting the first row by Eq. (49), resulting in

$$\begin{bmatrix} \mathbf{0} & \mathbf{C}_2 & \dots & \mathbf{C}_N \\ \mathbf{J}_1 & \mathbf{J}_2 & \dots & \mathbf{J}_{N-1} \end{bmatrix} \tag{50}$$

where $\mathbf{0}$ is the $N \times (N-1)$ zero matrix. Second, swapping the first N rows with the last $N+1$ to $2N$ rows gives

$$\begin{bmatrix} \mathbf{J}_1 & \mathbf{J}_2 & \dots & \mathbf{J}_{N-1} \\ \mathbf{0} & \mathbf{C}_2 & \dots & \mathbf{C}_N \end{bmatrix} \tag{51}$$

Third, we shift row D_{N+1} down until we get

$$\begin{bmatrix} \mathbf{J}_1 & \mathbf{J}_2 & \dots & \mathbf{J}_{N-1} \\ \mathbf{0} & \mathbf{C}_1 & \dots & \mathbf{C}_{N-1} \end{bmatrix} \tag{52}$$

Finally adding $-D_N$ to D_{N+1} gives

$$\begin{bmatrix} \mathbf{J}_1 & \mathbf{J}_2 & \dots & \mathbf{J}_{N-1} \\ \mathbf{0} & \mathbf{C}_1 + \mathbf{Z} & \dots & \mathbf{C}_{N-1} + \mathbf{Z} \end{bmatrix} \tag{53}$$

where $\mathbf{Z}$ is a submatrix of size $N \times (N-1)$ with elements

$$Z_{ij} = \begin{cases} -1 & \text{if } i = 1, j = 1 \\ 0 & \text{otherwise} \end{cases} \tag{54}$$

the matrix $\mathbf{D}$ is now in reduced row echelon form and by inspection we find that the $\text{Rank}(\mathbf{D}) = 2N - 1$. Now by the rank-nullity theorem, we have

$$\text{Rank}(\mathbf{D}) + \text{Nullity}(\mathbf{D}) = N(N-1) \tag{55}$$

and the $\text{Nullity}(\mathbf{D})$ is then given by

$$\begin{aligned} \text{Nullity}(\mathbf{D}) &= N(N-1) - \text{Rank}(\mathbf{D}) && (56) \\ &= N(N-1) - (2N-1) && (57) \\ &= N^2 - 3N + 1. && (58) \end{aligned}$$

Therefore, the dimension of the action space for a single layer network is $N^2 - 3N + 1$. □

By Theorem 4.1.1 the exact size of the action space is reduced by $2N-1$ and, in fact, the dimension of the action space is $d = N^2 - 3N + 1$. Again we note that the matrix $\mathbf{K}$ is layer-independent and only needs to be calculated once. Therefore the action space of concern is denoted by $\mathcal{A} = \mathbb{R}^{M(N^2-3N+1)}$.

4.2. Safety layer: Non-negativity constraints

The framework described above conserves the total lending and borrowing amounts of each bank but still allows for the possibility of negative lending amounts after a single step through the environment. In order to maintain the non-negativity conditions on $\mathbf{L}^\alpha(t+1)$, we pass the action through a QP problem inspired by Dalal et al. [40]. This amounts to solving the following QP problem

$$\begin{aligned} \underset{\mathbf{x}^\alpha}{\arg\min} \quad & \tfrac{1}{2} \|\mathbf{K}\mathbf{x}^\alpha - \mathbf{K}\mathbf{u}^\alpha\|^2 \\ \text{subject to} \quad & \text{offdiag}(\mathbf{L}^\alpha(t)) + \mathbf{K}\mathbf{x}^\alpha \succeq \mathbf{0} \end{aligned} \tag{59}$$

where the inequality $\succeq$ represents an element-wise inequality. The constraints of problem (59) ensure that after a step $\Delta\mathbf{L}$ the elements of $\mathbf{L}(t+1)$ will be non-negative. The QP problem itself aims to perturb the off-diagonal elements of $\Delta\mathbf{L}$

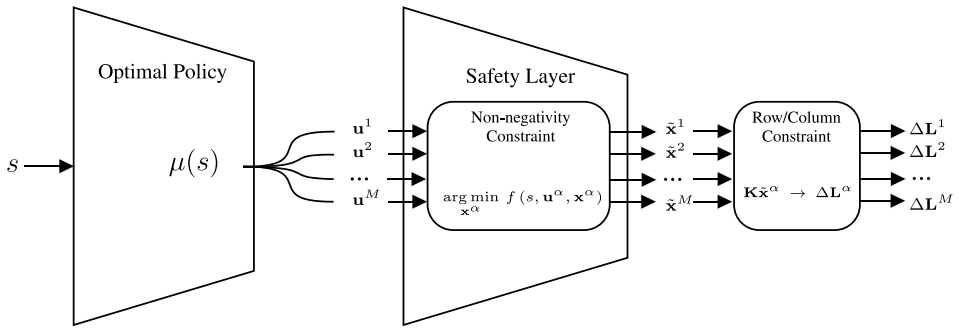


Fig. 3. A diagram describing how the optimal policy is modified using the safety layer. The optimal policy $\mu(s)$ is first partitioned into M policies for the respective layers of the multi-layer network. Each of these policies are then fed into the safety layer to operate on the policy to constrain the rows/columns and non-negativity condition of $\mathbf{L}(t+1)$ respectively. The constrained action $\mathbf{K}\tilde{\mathbf{x}}^\alpha$ contains the off-diagonal elements of a feasible $\Delta\mathbf{L}$.

in the Euclidean norm in order to satisfy constraint (34). Practically, we implement the CPLEX solver to solve problem (59).

By imposing these constraints, the actions from the agent will result in a liability network where the total lending and borrowing amounts appearing on their balance sheet are preserved. And after modifying the network structure, the non-negativity constraint will also be preserved. This will reduce the impact of the change in lending relationships on the banks' operations. We also note that by using the methods described in Section 4.1, we are able to avoid including constraints (32) and (33) to problem (59). A diagram detailing the flow of the policy through the safety layer can be found in Fig. 3.

4.3. Experiments

4.3.1. Initializing the complex network and DDPG agent

To generate our complex network we will start by using the R package *systemicrisk* to build the interbank liability matrix $\mathbf{L}^\alpha$. In doing so, we can forgo the use of Eq. (2). Following Diem [43], we begin by randomly sampling the row and column sum vectors $\hat{\mathbf{l}}^\alpha$ and $\hat{\mathbf{a}}^\alpha$ of $\mathbf{L}^\alpha$, respectively, where

$$\hat{\mathbf{l}}^\alpha = (\hat{l}_1^\alpha, \hat{l}_2^\alpha, \ldots, \hat{l}_N^\alpha)$$
$$\hat{\mathbf{a}}^\alpha = (\hat{a}_1^\alpha, \hat{a}_2^\alpha, \ldots, \hat{a}_N^\alpha).$$

In our experiments, we consider three different network sizes where $N \in \{10, 20, 30\}$. Let $b \in \hat{\mathcal{B}}$, $m \in \hat{\mathcal{M}}$, $s \in \hat{\mathcal{S}}$ be the set of indices denoting the big, medium, and small banks respectively. The elements of the row and column sum vectors are sampled from the following uniform distributions

$$\hat{l}_b^\alpha \sim U(6000, 10000), \hat{l}_m^\alpha \sim U(2000, 6000), \hat{l}_s^\alpha \sim U(500, 2000)$$

and

$$\hat{a}_b^\alpha \sim U(0, 2000), \hat{a}_m^\alpha \sim U(0, 700), \hat{a}_s^\alpha \sim U(0, 150)$$

where $\hat{\mathcal{B}} = \{1\}, \hat{\mathcal{M}} = \{2, 3\}, \hat{\mathcal{S}} = \{4, \ldots, N\}$ for $N = 10$ and $\hat{\mathcal{B}} = \{1, 2\}, \hat{\mathcal{M}} = \{3, 4, 5\}, \hat{\mathcal{S}} = \{6, \ldots, N\}$ for $N = 20, 30$. The *systemicrisk* package estimates interbank liability matrices satisfying $\hat{\mathbf{l}}^\alpha$ and $\hat{\mathbf{a}}^\alpha$ based on Bayesian methodology developed by Gandy and Veraart [44].

With the liability matrices given, we can set the total asset value of the entire network using $A = s\sum_\alpha \sum_{i,j} L_{ij}^\alpha$ where $s > 1$. The relative proportion of the network value that is used for lending can then be calculated as

$$\theta^\alpha = \frac{\sum_{i,j} L_{ij}^\alpha}{A}. \tag{60}$$

Finally, we can generate the balance sheet of each bank in the network using Eqs. (3) to (6). We set the cash deposit ratio to be $\beta = 0.18$ and the equity capital ratio for each bank i to be sampled from the following interval, $\gamma_i \in (0.07, 0.2)$. When modifying the reward function to incorporate credit risk, we set $\gamma_i = 0.2$ with $k_i \in (0.07, 0.12) \cup (0.85, 1.0)$. The important quantity to consider on the balance sheet is the equity given by Eq. (4) as this value is used in the calculation of the DR. Other values of the balance sheet may be used to calculate any other relevant financial variables, as required.

To test the effectiveness of applying RL in reducing systemic risk, we train and evaluate the RL agent on a number of different network structures. We also tested the flexibility of using RL by incorporating the notion of credit risk using the modified reward function, Eq. (36).

Given the simulated multi-layer complex network we can begin reducing the systemic risk using DDPG. We use the Tensorflow TF-Agents framework to accomplish this task. For both actor and critic networks, we use three-layer neural networks with node sizes (256, 256, 256). The parameter settings for the DDPG agent can be found in Table 1. Training was done for 8000 total iteration steps.

4.3.2. Single layer case

In the single layer case, we will be investigating the effectiveness of the RL agent in reducing the systemic risk of the network and the effect of modifying the reward function for preferential reduction in DR for particular banks across the single layer network. For the single-layer case experiments, we let $N \in \{10, 20, 30\}$ and $M = 1$. The reward function used in reducing the systemic risk is Eq. (35). To preferentially reduce systemic risk for highly leveraged banks, we consider the reward function (36) using four different weighting schemes. The DR distribution for the complex networks when exploring the effectiveness of the different weight functions is approximately uniform. This allows us to more easily judge the differences between the weight functions. In this way we can compare the different weight functions in a fair manner.

$$w_{\text{uniform}}(k_i) = 1.0 \tag{61}$$

$$w_{\text{linear}}(k_i) = k_i \tag{62}$$

$$w_{\exp,v}(k_i) = e^{vk_i} \tag{63}$$

The first and second weighting schemes use a constant weight of 1.0 and linear weights comprising of the leverage ratio defined by Eqs. (61), and (62) respectively. The third and fourth weighting schemes use an exponential weight dependent on the leverage ratio defined by Eq. (63) with parameter $v = 1.0$ and $v = 10.0$ respectively.

4.3.3. Multi-layer case

In the multi-layer case, we will be investigating the effectiveness of the RL agent in reducing the systemic risk of the network and present some observations on the network characteristics of multi-layer networks. For the multi-layer experiments we let $N \in \{10, 20, 30\}$ and $M \in \{1, 2, 3\}$. Therefore, the RL agent will be tasked with reducing systemic risk under nine different network sizes. Similar to the single-layer case, the reward function used in reducing the systemic risk is Eq. (35). The DR however, is calculated using the multi-layer DR algorithm outlined in Section 2.3.

Given the optimized complex networks, we can observe the different characteristics of a multi-layer complex network after it has been modified by our RL model. We present the density, Jaccard distance, clustering coefficient, and average-weighted neighbour degree. The density for layer α of the multi-layer complex network is given by

$$d = \frac{m^\alpha}{N(N-1)} \tag{64}$$

where m^α is the number of edges in layer α and N is the number of banks. This is the number of edges divided by the total number of possible edges. Therefore, the density can serve as a measure of sparsity.

To compare the differences between each layer of the network before and after optimization, we use the Jaccard distance measure. Given the two sets of edges E_α and E_κ we can calculate the Jaccard distance by

$$d_J(E_\alpha, E_\kappa) = 1 - J(E_\alpha, E_\kappa) \tag{65}$$

where $J(E_\alpha, E_\kappa)$ is the Jaccard similarity index between layers α and κ. The Jaccard similarity index is calculated as by

$$J(E_\alpha, E_\kappa) = \frac{|E_\alpha \cap E_\kappa|}{|E_\alpha \cup E_\kappa|} \tag{66}$$

The Jaccard distance measures the dissimilarity between the different layers by comparing the proportion of similar edge connections between the layers. The directed networks are converted to undirected networks before calculating the Jaccard distance.

Next we compute the clustering coefficient, treating each layer of the network as a directed unweighted graph. The clustering coefficient c_i for node i in layer α of the multi-layer complex network is given by

$$c_i = \frac{T_i}{2(k^\alpha_{\text{total},i}(k^\alpha_{\text{total},i} - 1) - 2k^\alpha_{\leftrightarrow,i})} \tag{67}$$

where T_i is the total number of directed triangles formed by node i and

$$k^\alpha_{\text{total},i} = k^\alpha_{\text{out},i} + k^\alpha_{\text{in},i} \tag{68}$$

is the total degree of node i in layer α and $k^\alpha_{\leftrightarrow,i}$ is the number of bilateral edges between node i and its neighbours. Eq. (67) measures the clustering coefficient for a directed unweighted graph. The clustering coefficient is the ratio between all directed triangles and the number of possible triangles which measures the tendency of the network to form tightly connected neighbourhoods [45].

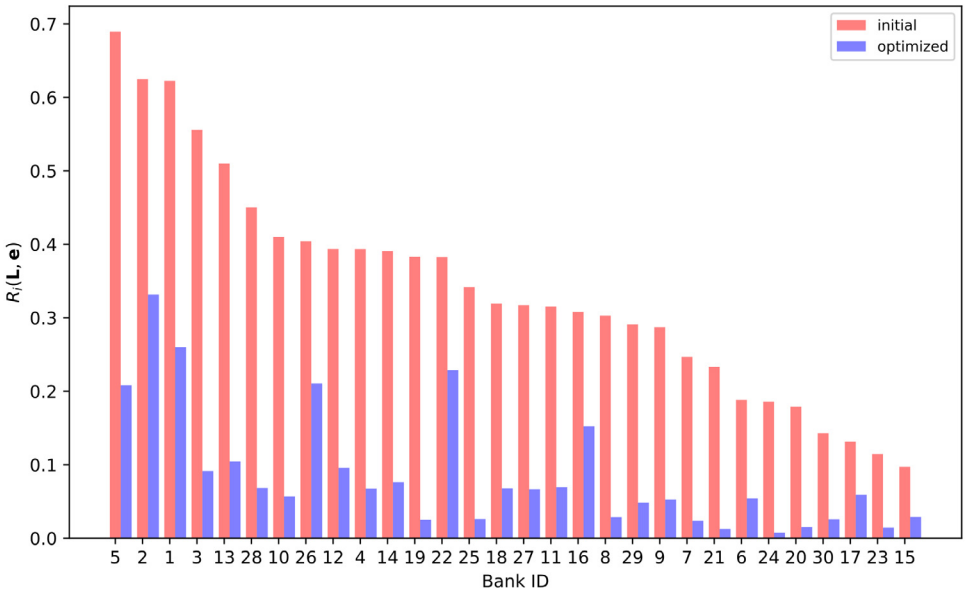


Fig. 4. The DR of single-layer network of size $N = 30$. The banks are ordered from largest to smallest based on their DR. The red bars represent the initial levels of DR while the blue bars represent the optimized levels of DR.

Finally, treating each layer of the network as a directed weighted graph, we compute the average-weighted neighbour degree [20]. The average-weighted neighbour degree $k^{\alpha}_{\text{nn},i}$ for node i in layer α of the multi-layer complex network is given by

$$k^{\alpha}_{\text{nn},i} = \frac{1}{s^{\alpha}_i} \sum_{j=1}^{N} (L^{\alpha}_{ij} + L^{\alpha}_{ji}) k^{\alpha}_{\text{total},j} \tag{69}$$

where

$$s^{\alpha}_i = \sum_{j=1}^{N} L^{\alpha}_{ij} + L^{\alpha}_{ji} \tag{70}$$

is the weighted node degree of bank i in layer α.

5. Results and discussion

We will be evaluating the effectiveness of our constraint DDPG model in reducing systemic risk for two cases: (1) the single-layer case and (2) the multi-layer cases. It should be noted that given the nature of RL, it cannot be guaranteed that the reduced DR is a global optimum. However, with this trade-off, we are granted the flexibility of RL allowing the DDPG agent to directly consider the recursive DR measure in its reward. Additionally, we introduce the idea of preferential systemic risk reduction to the DDPG agent by modifying the reward function whose results can be found in Section 5.1. Furthermore, to adapt the RL model to a multi-layer case we simply extend the DDPG agent's action to different layers and modify the reward function to include the DR of different layers whose results can be found in Section 5.2. The DDPG agent was tested on two types of networks. The first type consisting of a few number of large banks and the second type consisting of banks of similar sizes. In all cases we use the same set of hyperparameters described in Table 1. With more sophisticated hyperparameter tuning methods, it is suspected that the DR can be further reduced.

5.1. Single-layer complex network

The results in Figs. 4 and 5 were generated using Eq. (35) as the reward function with $N = 30$ and $M = 1$. The DR calculated in this section uses the conventional DR algorithm. For this particular network, the DR was reduced from 10.21 to 2.58. The DDPG agent achieved a reduction of 74.73%. From Fig. 4, it can be observed that every bank has had their DR significantly reduced after optimization. Although the degree of reduction in DR between each bank varies, we did not observe an increase in DR for any bank.

From Fig. 5 we can see how the network changes before and after optimization by the DDPG agent. An obvious increase in the sparsity can be observed. Using the policy that provides the greatest minimization of the DR we calculate average

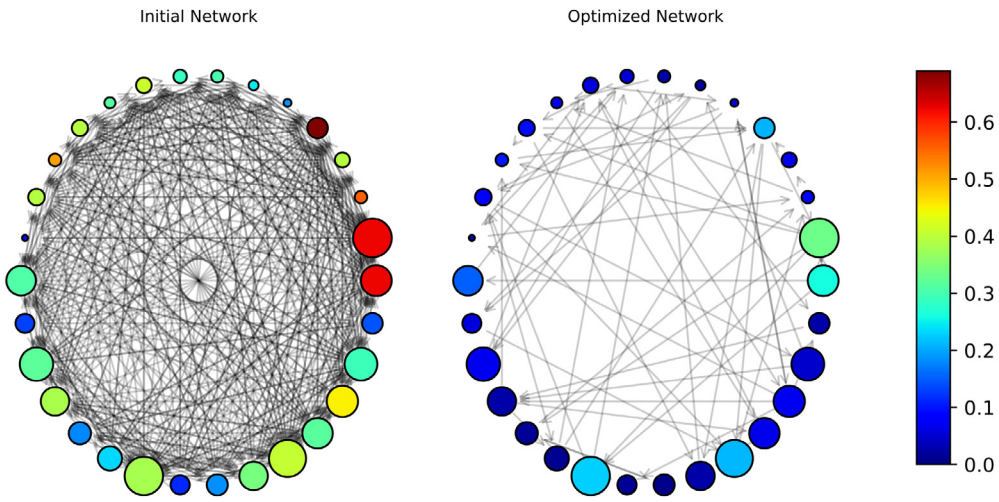


Fig. 5. The structure of the single-layer complex network of size $N = 30$, where red represents high DR and blue represents low DR. The size of the circle represents the initial equity of the banks. The directed edges represent the lending relationship from bank i to bank j.

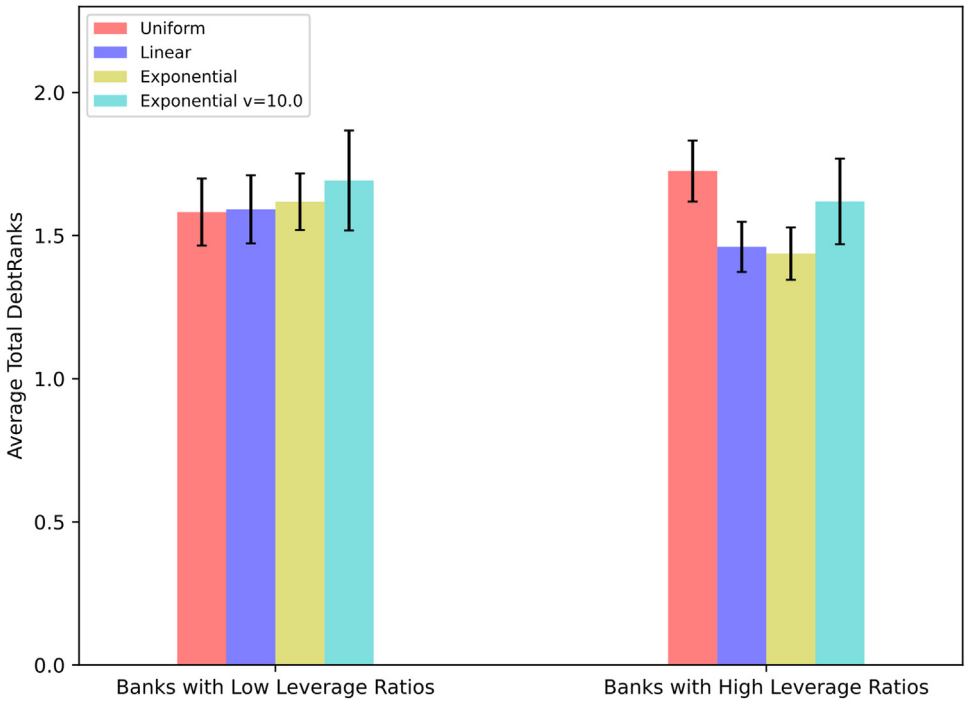


Fig. 6. The low and high leverage banks were separated into two groups and the average total optimized DR was plotted for a network of size $N = 30, M = 1$. The average and standard deviation (presented as error bars) was calculated using 100 episodes.

DR reduction across 100 episodes. From Table 2 we find that the DR was reduced significantly for all networks of size $N \in \{10, 20, 30\}$ and $M = 1$, the level of reduction achieved ranged from 70% to 75%.

The reward functions in this paper are designed to incentivize the DDPG agent to reduce the overall systemic risk of the network. By introducing the weight factors (61)–(63) and using reward function (36), we aim to incentivize the agent to reduce the overall systemic risk while also preferentially reducing the DR of highly leveraged banks. The level of reduction in DR using the uniform weight factor (61) will be treated as the benchmark (where there is effectively no weight factor on the DR of each bank). In Fig. 6, we compare the total DR for each leverage group. For the low leverage banks, we find that the total DR when using the linear and exponential weight functions is similar or greater compared to the benchmark total DR. The opposite observation is made for the high-leverage group. That is, when using linear and exponential weight functions, the total DR is lower compared to the benchmark total DR.

Table 2
The average initial DR, optimized DR, and % reduction under each weighting scheme (Uniform, Linear, and Exponential with $v = 1.0$ and $v = 10.0$). The results are generated using single layer networks of size $N \in \{10, 20, 30\}$ over 100 episodes. The standard deviations are presented in the brackets.

N	Uniform			Linear		
	Initial DR	Optimized DR	% reduction	Initial DR	Optimized DR	% reduction
10	7.85 (0.41)	2.31 (0.26)	70.52 (3.94)	7.84 (0.38)	2.61 (0.40)	66.62 (5.53)
20	11.37 (0.48)	2.69 (0.16)	76.27 (1.52)	11.36 (0.50)	3.14 (0.19)	72.34 (2.09)
30	13.28 (0.49)	3.31 (0.18)	75.02 (1.55)	13.22 (0.57)	3.05 (0.19)	76.87 (1.65)
	Exponential ($v = 1.0$)			Exponential ($v = 10.0$)		
10	7.86 (0.32)	2.36 (0.26)	69.98 (3.50)	7.83 (0.31)	2.60 (0.35)	66.75 (4.76)
20	11.44 (0.49)	3.13 (0.27)	72.62 (2.41)	11.44 (0.52)	3.07 (0.32)	73.07 (3.11)
30	13.28 (0.53)	3.08 (0.16)	76.82 (1.36)	13.23 (0.50)	3.34 (0.36)	74.76 (2.73)

Note that despite higher leveraged banks receiving a greater weight in magnitude when using the exponential weight functions (63) compared to linear weight functions (62), we find that this does not necessarily mean there is a greater prioritization for the reduction of DR with respect to credit risk. From Fig. 6, we observe that the linear weight provides a much more significant bias to reduce the DR of high leverage banks when compared to using the exponential weights with $v = 10.0$. This variability may be due to the stochastic nature of RL in training and the initial network structure to be optimized. Despite this observation, the modification to the reward function appears to encode the desired preferential reduction in systemic risk for highly leveraged banks when compared to not applying any weighting.

Additionally, regardless of the weight function used, the main goal of the DDPG agent was achieved where the DR of the networks is reduced overall. The change in the DRs can be found in Table 2. The DDPG agent achieved a reduction as low as 67% to as high as 77% in DR depending on the weight function used. Preferentially reducing systemic risk of highly leveraged banks is beneficial because although some banks might have high levels of systemic risk, their credit risk might also be lower. In this case, reducing the systemic risk of banks with higher credit risk can be prioritized by simply modifying the reward function.

5.2. Multi-layer complex network

All DRs in this section were reduced using Eq. (35) as the reward function for the DDPG agent. Figs. 7 and 8 were generated using a complex network of size $N = 30, M = 3$. The initial DR that was calculated before applying the DDPG agent was 11.10. After training and evaluation, we found that the DR was reduced to 6.53, a reduction of 41%. In Fig. 7, a notable increase in DR can be observed across layers. This is expected as the multi-layer DR algorithm accounts for the inter-layer contagion spreading through successive layers. As distress accumulates from one layer to another, the equity of the banks may not be sufficient to cover the default of loans in higher layers. The equity of the banks in distress are reduced by the lending amount proportional to the distress experienced in the previous layer described by Eq. (11). Despite the additional level of systemic risk accumulated in a multi-layer network, we can see that by modifying the previous layers' structure we can reduce the overall DR by some amount in the subsequent layers. This is further evident when we note that the initial DR of the first, second and third layer are 1.14, 3.21, and 6.74 respectively. After reduction, the DR in the first, second and third layers was 0.62, 2.05, and 3.85. Therefore, the reduction achieved across the layers was 46%, 36%, and 43%, respectively. We find the greatest reduction in systemic risk in the first layer, despite having a relatively low level of initial DR compared to the other layers. Considering that the distress from the first layer propagates to the following layers, targeting the layer where the distress originates for optimization may prove to be the most effective. In this example, the DDPG agent targets the first layer, but other network configurations and different shock propagation dynamics may present different nodes or layers to prioritize.

The average optimized DRs can be found in Table 3. For all network types in the multi-layer case, we found that the DDPG agent was able to achieve an average reduction ranging from 8% to 57%. The lowest reduction achieved was in the case of similar sized banks with $N = 30, M = 3$ with a reduction of 8%. This may be due to the already low average initial DR. The network structure consisting of similarly sized banks in a multi-layer setting may also present some difficulty in DR reduction for the DDPG agent. Bear in mind Glasserman and Young [9] report that contagion effects are more pronounced when node sizes are heterogeneous, suggesting that the performance of the DDPG agent under the "Few Large Banks" scenario should take higher precedence. By allowing contagion to accumulate across successive layers, we find that the increase in overall distress results in an overall lower level of systemic risk reduction achieved after optimization.

From Fig. 8, the reorganization of the multi-layer complex network results in an increase in sparsity of the network based on the density of the edges. The average density values for each layer of the corresponding networks can be found in Table 4. We observe that in every case of optimization, the average density was lowered. It appears that given the larger network size, a larger reduction in density is observed. Although the increase in sparsity is consistent with what has been observed in other studies, this observation is not necessarily unique to an optimized network with low DR, as networks with high DR after optimization have been observed as well [20].

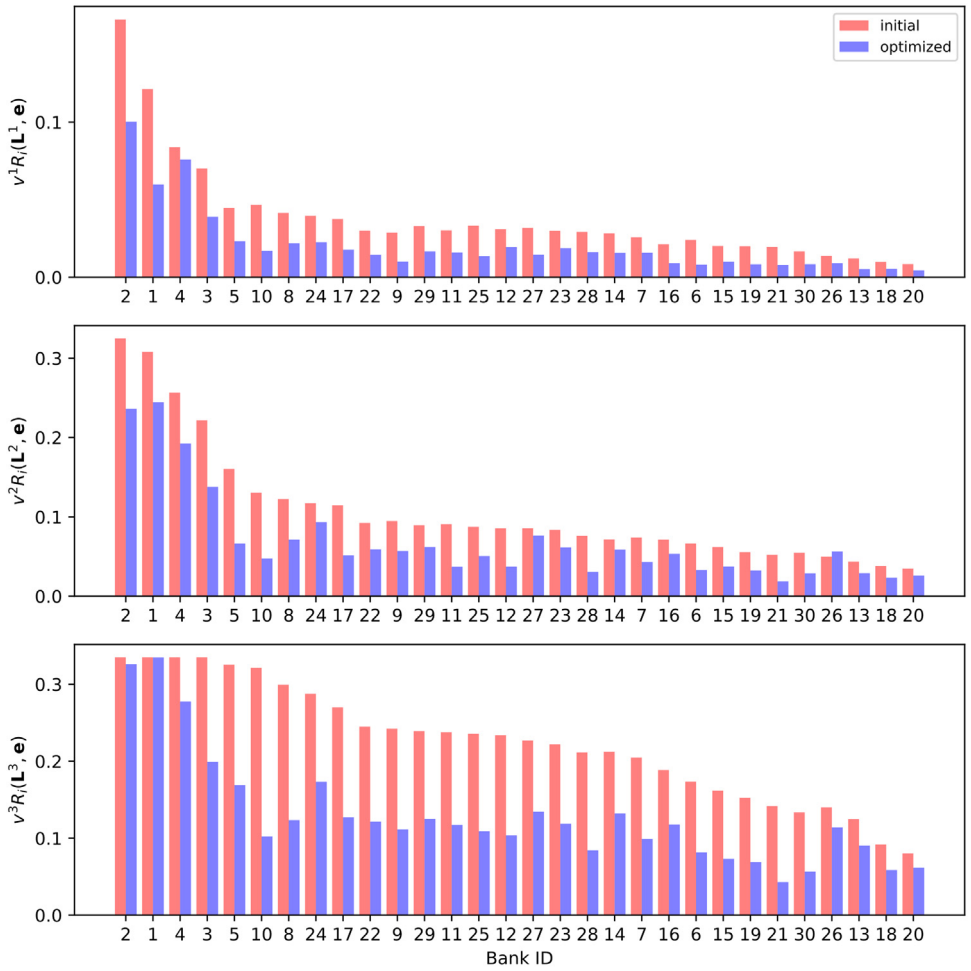


Fig. 7. The DR of multi-layer network of size $N = 30$. The DRs in this plot are weighted by each layer's total relative economic value, v^{α}. The banks are ordered from largest to smallest based on their DR. The red bars represent the initial levels of DR while the blue bars represent the reduced levels of DR.

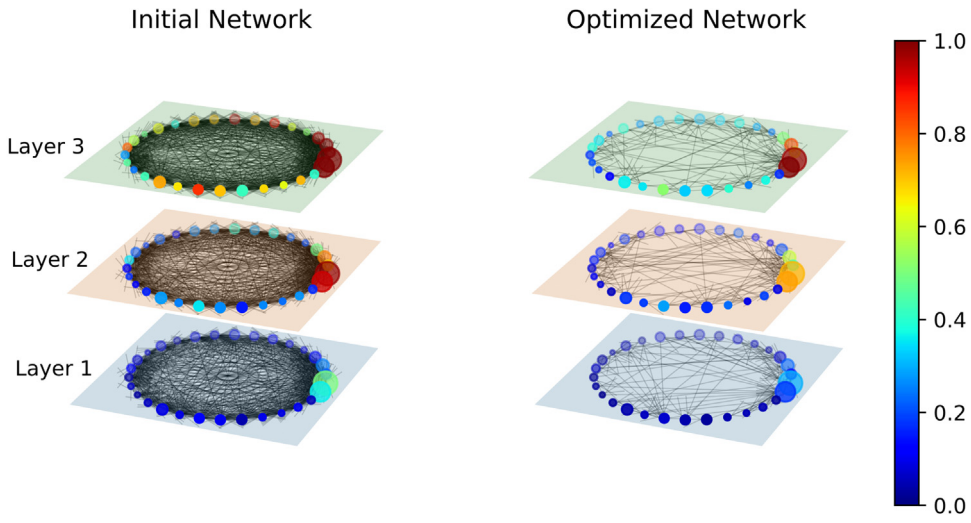


Fig. 8. The structure of the multi-layer complex network of size $N = 30$, where red represents high DR and blue represents low DR. The size of the circle represents the initial equity of the banks. The directed edges represent the lending relationship from bank i to bank j.

Table 3
The average DR of the initial and optimized networks along with the % reduction after optimization for multi-layer networks of size $N \in \{10, 20, 30\}$ and $M \in \{1, 2, 3\}$ over 100 episodes. The values in the brackets are the standard deviations. The DR values are calculated using equation (35). The ratio of large to medium to small banks under the "Few Large Banks" scenario is 1:2:8 for $N = 10$ and 2:3:(N-5) for $N = 20, 30$, while the distribution of bank sizes is uniform under the "Similar Sized Banks" scenario.

N	M	Few Large Banks			Similar Sized Banks		
		Init. DR	Opt. DR	% red.	Init. DR	Opt. DR	% red.
10	1	7.81 (0.30)	3.40 (0.19)	56.38 (2.73)	7.86 (0.33)	2.25 (0.24)	71.32 (3.23)
	2	7.00 (0.28)	4.41 (0.22)	36.91 (4.05)	5.53 (0.21)	3.32 (0.12)	39.89 (2.83)
	3	7.00 (0.24)	4.70 (0.12)	32.88 (2.79)	4.87 (0.10)	3.13 (0.11)	35.72 (2.32)
20	1	13.88 (0.52)	4.11 (0.33)	70.37 (2.71)	11.62 (0.48)	3.02 (0.12)	73.99 (1.52)
	2	11.98 (0.29)	5.54 (0.09)	53.75 (1.18)	6.19 (0.17)	4.24 (0.08)	31.52 (2.21)
	3	10.03 (0.20)	5.95 (0.30)	40.68 (3.33)	4.90 (0.08)	4.21 (0.05)	14.05 (1.62)
30	1	18.79 (0.64)	4.04 (0.38)	78.46 (2.10)	13.13 (0.49)	3.20 (0.24)	75.63 (1.87)
	2	13.38 (0.38)	5.72 (0.51)	57.21 (4.28)	6.14 (0.11)	4.15 (0.08)	32.38 (1.51)
	3	11.29 (0.21)	6.66 (0.13)	41.05 (1.48)	4.81 (0.05)	4.41 (0.03)	8.24 (0.71)

Table 4
The average initial and optimized density for multi-layer complex networks of size $N \in \{10, 20, 30\}$, and $M \in \{1, 2, 3\}$ for each respective layer over 100 episodes. The values in the brackets are the standard deviations. The initial and optimized density are the density before and after optimizing the network configuration with respect to the DR, respectively.

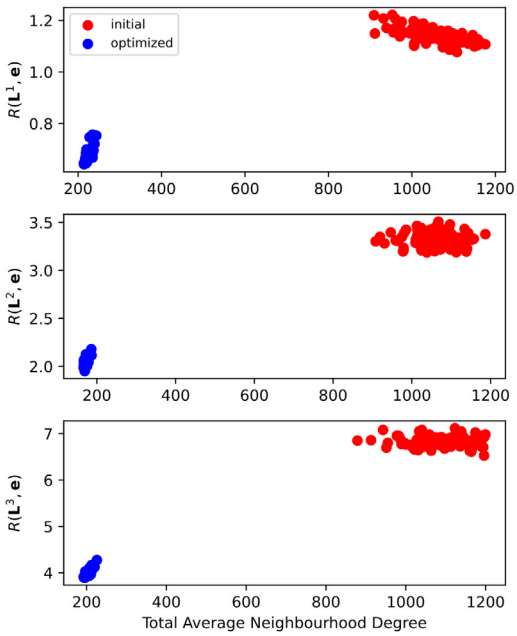
N		30		20		10	
M	α	Init. density	Opt. density	Init. density	Opt. density	Init. density	Opt. density
1	1	0.55 (0.05)	0.07 (0.00)	0.57 (0.05)	0.11 (0.00)	0.47 (0.06)	0.21 (0.00)
2	1	0.54 (0.04)	0.08 (0.01)	0.55 (0.06)	0.12 (0.01)	0.56 (0.07)	0.25 (0.01)
	2	0.55 (0.04)	0.09 (0.01)	0.57 (0.05)	0.10 (0.00)	0.46 (0.05)	0.22 (0.01)
3	1	0.52 (0.04)	0.07 (0.00)	0.57 (0.05)	0.11 (0.00)	0.59 (0.07)	0.22 (0.01)
	2	0.54 (0.04)	0.07 (0.00)	0.54 (0.06)	0.11 (0.01)	0.56 (0.08)	0.22 (0.01)
	3	0.55 (0.04)	0.07 (0.00)	0.56 (0.04)	0.11 (0.00)	0.46 (0.05)	0.21 (0.00)

Table 5
The matrices containing the average Jaccard dissimilarity between the layers of multi-layer complex networks of size $N \in \{10, 20, 30\}$, and $M = 3$ over 100 episodes. The values in the brackets are the standard deviations. The initial and optimized Jaccard dissimilarity are the Jaccard dissimilarity before and after optimizing the network configuration with respect to the DR, respectively.

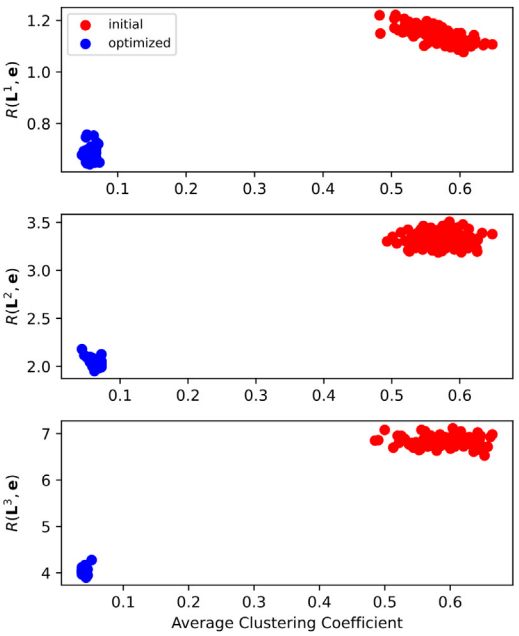
N	α	Initial			Optimized		
		1	2	3	1	2	3
10	1	0	0.33 (0.09)	0.41 (0.08)	0	0.63 (0.02)	0.64 (0.03)
	2	0.33 (0.09)	0	0.42 (0.08)	0.63 (0.02)	0	0.59 (0.03)
	3	0.41 (0.08)	0.42 (0.08)	0	0.64 (0.03)	0.59 (0.03)	0
20	1	0	0.35 (0.05)	0.34 (0.04)	0	0.80 (0.01)	0.83 (0.01)
	2	0.35 (0.05)	0	0.35 (0.05)	0.80 (0.01)	0	0.83 (0.01)
	3	0.34 (0.04)	0.35 (0.05)	0	0.83 (0.01)	0.83 (0.01)	0
30	1	0	0.37 (0.03)	0.37 (0.04)	0	0.90 (0.00)	0.90 (0.01)
	2	0.37 (0.03)	0	0.36 (0.04)	0.90 (0.00)	0	0.93 (0.00)
	3	0.37 (0.04)	0.36 (0.04)	0	0.90 (0.01)	0.93 (0.00)	0

Comparing the Jaccard distances in Table 5 across the different layers for the initial networks, we find that there is some similarity between all layers. After optimization any similarities between the layers are significantly reduced. That is, the topology of each layer becomes more dissimilar, suggesting that holding more dissimilar lending patterns across loans with differing maturities may produce complex networks more resilient against systemic risk. However, the benefits of diversification for the reduction of systemic risk are heavily debated and have been shown to be intimately related to systemic risk [46,47]. Acemoglu et al. [48] has shown that diversification protects well against small shocks but poorly against large shocks. Through empirical evidence [47] argues that large and medium sized banks contribute to systemic risk through diversification.

Fig. 9(a) depicts the total average neighbourhood degree of the banks in the multi-layer complex network. We find that there is a significant change in the network characteristics after optimization. After optimization, a large number of banks will have a reduced total average neighbourhood degree. In all layers, it can be observed that before optimization, networks with high DR will have a large total average neighbourhood degree. After optimization, the networks have a



(a) Total average neighbourhood degree



(b) Average clustering coefficient

Fig. 9. The total DR was plotted against (a) the total average neighbour degree $\sum_{i=1}^{N} k^{\alpha}_{nn,i}$ in layer α. (b) The average clustering coefficient $\frac{1}{N}\sum_{i=1}^{N} c_i$. Both figures were generated by plotting the respective measures across 100 episodes using multi-layer complex networks of size $N = 30$, and $M = 3$.

reduced total DR and lower total average neighbourhood degree. This is consistent with Teteryatnikova [49] who has shown through a tiered banking system that a negative correlation between neighbouring banks' degree can increase the resilience of the network. Fig. 9(b) shows the average clustering coefficient of the multi-layer network, we find that after optimization, the average clustering coefficient is reduced for the majority of the networks. This suggests that the banks in the network begin to form less complete subgraphs with their neighbours after optimization.

5.3. Feasibility and regulatory guidance

In our model we delegate the task of discovering lower systemic risk networks to a single RL agent. To accomplish this task, the RL agent is incentivized through the use of reward functions. At the same time, we attempt to mitigate the disruption to the operation of the banks by constraining the change to the total lending and borrowing amounts of a bank. In reality, there are many factors such as interest rates, credit worthiness, and liquidity that influence their decision making process. Explicitly modifying an actual interbank network presents a practicability issue as banks have several objectives and constraints to achieve and satisfy.

To clarify, we do not suggest imposing the network configuration designed by the RL agent on the participants of a real interbank network. Instead, the optimized networks from our model can serve as a benchmark to aid in designing regulatory policies when considering the multi-layer aspects of interbank networks, which is a use case that has been similarly suggested by Li et al. [10], Diem et al. [20], Pichler et al. [21]. To encourage reorganization of the real interbank network, Poledna and Thurner [18], Poledna et al. [19] propose to implement a systemic risk tax. This is a tax on transactions between any two counterparties to incentivize the formation of lower systemic risk networks. Their systemic risk tax is dependent on the change in expected systemic loss, a function of the DR of every bank in the network. In our model, the RL agent is guided by a relative change in DR of the network after every optimization pass. While the functional forms of the incentives are different, the marginal change to DR is a similar concept in both models.

An approach that may admit a more interpretable incentive mechanism would be to consider a multi-agent RL model such as the Multi-agent DDPG algorithm [50]. In this case, each bank can represent an agent and the environment can be designed to be competitive with respect to their own objectives or cooperative with respect to reducing overall systemic risk. The reward functions would then represent a direct incentive influencing the behaviour of each bank and hence, the evolution of the interbank network.

6. Conclusion

RL is an incredibly powerful tool that proves to be effective in the context of systemic risk management. In this paper, we introduce a systemic risk reduction framework that takes advantage of RL by modifying the classical DDPG algorithm. The model reorganizes the interbank lending relationships of banks into a configuration that better mitigates the effects of contagion. The asset composition of the multi-layer networks consisted of short-term and long-term debts. In our model, the repayment of long-term debts is dependent on the solvency of short-term debts.

To calculate the systemic risk of such a network, we propose a new measure of DR accounting for the contagion that may spread from one layer to another as well as accounting for the impact of previous defaults on the individual banks' ability to repay future debts. The behaviour of the RL agent is guided by the reward function, and as a result, our RL agent is capable of solving problems in assessing and managing systemic risk.

To the best of our knowledge, this cannot be solved by traditional optimization techniques, since a recursive algorithm can be challenging to incorporate into the objective function of the optimization problem. We propose the DR reduction learning algorithm, called constraint DDPG, to find a network structure with reduced systemic risk. In order to satisfy the borrowing and lending constraints and maintain non-negativity with respect to the individual banks' lending after applying the DDPG agent's action, we modify the actor output in two ways. The first is by proposing a homogeneous system of linear equations whose solutions satisfy the lending and borrowing constraint. The second is the safety layer, which satisfies the non-negativity constraint by solving a QP problem. The effectiveness of our model was tested on different single-layer and multi-layer network with varying sizes, layers, and distribution of assets.

The performance of the RL agent was evaluated based on the level of DR reduction achieved. In all cases, a reduction in DR was observed, suggesting that RL is indeed an efficient tool in producing network structures that have reduced systemic risk in terms of DR. In the single-layer case, a reduction as high as 75% was observed while in the multi-layer case a reduction as high as 57% was observed. We find that the optimization process results in considerably different network topologies. The density, average neighbourhood degree, and clustering coefficient were observed to decrease after optimization. The Jaccard distance increased between the layers after optimization.

Finally, we present some potential extensions of our work. While we only consider a multi-layer interbank lending network, there are many different transmission channels for systemic risk. Our model can be extended to consider lending, security cross-holdings, derivatives, and foreign exchange transactions using the multi-layer exposure network model presented in Poledna et al. [28]. The complex networks used in this study were simulated. It would be interesting to see the degree of optimization when considering real-world interbank network structures. Another extension could include a multi-action RL framework to consider equity levels or redistribution of wealth across different layers. At the moment we assign a single DDPG agent with the task to reduce the systemic risk of an entire multi-layer complex network, the alternative approach to this problem is to design a multi-agent RL framework and let every bank be its own RL agent and work cooperatively to reduce the systemic risk.

CRediT authorship contribution statement

Richard Le: Investigation, Methodology, Visualization, Writing – original draft. **Hyejin Ku:** Conceptualization, Methodology, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The code (and data) in this paper is posted on https://github.com/PencilKit/Reducing-Systemic-Risk-with-DDPG

Acknowledgement

This research was partly supported by Natural Sciences and Engineering Research Council of Canada, Discovery grant 504316.

Appendix. Proposed DR reduction algorithm

Algorithm 1 Constraint DDPG

Initialize the multi-layer network
Randomly initialize critic network $Q(s, a \mid \theta^Q)$ and actor $\mu(s \mid \theta^\mu)$ with weights θ^Q and θ^μ
Initialize target network Q' and μ' with weights $\theta^{Q'} \longleftarrow \theta^Q, \theta^{\mu'} \longleftarrow \theta^\mu$
Initialize replay buffer $\mathcal{D}$
for episode = 1, N_{episode} **do**
  Initialize a random process $\mathcal{N}$ for action exploration
  Receive an initial random observation state s_1
  for $t = 1, T$ **do**
    Calculate $a_t = \mu(s_t \mid \theta^\mu) + \mathcal{N}$ according to the current policy and exploration noise
    for $\alpha = 1, M$ **do**
      Take partition $\mathbf{u}^\alpha$ from a_t and pass to the safety layer to find $\tilde{\mathbf{x}}^\alpha$
      Use $\tilde{\mathbf{x}}^\alpha$ to calculate $\Delta\mathbf{L}^\alpha(t)$ by Eq. (46)
      Calculate the new network by $\mathbf{L}^\alpha(t+1) = \mathbf{L}^\alpha(t) + \Delta\mathbf{L}^\alpha(t)$
    end for
    Calculate the reward r_t based on Eq. (35) or (36) and observe the new state s_{t+1}
    Store transition (s_t, a_t, r_t, s_{t+1}) in $\mathcal{D}$
    Sample a random minibatch of N_{mini} transitions (s_i, a_i, r_i, s_{i+1}) from $\mathcal{D}$
    Set $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1} \mid \theta^{\mu'}) \mid \theta^{Q'})$
    Update critic by minimizing the loss:

$$L_{\text{loss}} = \frac{1}{N_{\text{mini}}} \sum_i (y_i - Q(s_i, a_i \mid \theta^Q))^2$$

    Update the actor policy using the sampled policy gradient:

$$\nabla_{\theta^\mu} J \approx \frac{1}{N_{\text{mini}}} \sum_i \nabla_a Q(s, a \mid \theta^Q) \mid_{s=s_i, a=\mu(s_i)} \nabla_{\theta^\mu} \mu(s \mid \theta^\mu) \mid_{s_i}$$

    Update the target networks:

$$\theta^{Q'} \longleftarrow \tau\theta^Q + (1-\tau)\theta^{Q'}$$

$$\theta^{\mu'} \longleftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'}$$

    if $R(\mathbf{L}(t+1), e) \geq R(\mathbf{L}(t), e)$ **or** $t = T$ **then**
      End the episode
    end if
  end for
end for

References

[1] M.S. Pagano, J. Sedunov, A comprehensive approach to measuring the relation between systemic risk exposure and sovereign debt, J. Financ. Stab. 23 (2016) 62–78, http://dx.doi.org/10.1016/j.jfs.2016.02.001.
[2] S. Sehgal, S. Mathur, M. Arora, L. Gupta, Sovereign ratings: Determinants and policy implications for India, IIMB Manag. Rev. 30 (2) (2018) 140–159, http://dx.doi.org/10.1016/j.iimb.2018.01.006.
[3] Y.-L. Huang, C.-H. Shen, The sovereign effect on bank credit ratings, J. Financ. Serv. Res. 47 (3) (2015) 341–379, http://dx.doi.org/10.1007/s10693-014-0193-7.
[4] R. Bianchi, M. Drew, T.R. Wijeratne, Systemic Risk, the TED Spread and Hedge Fund Returns, Griffith University, Department of Accounting, Finance and Economics, 2010, Retrieved from https://EconPapers.repec.org/RePEc:gri:fpaper:finance:201004.
[5] M. Busse, M. Dacorogna, M. Kratz, The impact of systemic risk on the diversification benefits of a risk portfolio, Risks 2 (3) (2014) 260–276.
[6] S. Strobl, Stand-alone vs systemic risk-taking of financial institutions, J. Risk Financ. (2016).
[7] L. Eisenberg, T.H. Noe, Systemic risk in financial systems, Manage. Sci. 47 (2) (2001) 236–249.
[8] P. Gai, A. Haldane, S. Kapadia, Complexity, concentration and contagion, J. Monetary Econ. 58 (2011) 453–470, http://dx.doi.org/10.1016/j.jmoneco.2011.05.005.
[9] P. Glasserman, H.P. Young, How likely is contagion in financial networks? J. Bank. Financ. 50 (2015) 383–399, http://dx.doi.org/10.1016/j.jbankfin.2014.02.006.
[10] S. Li, M. Liu, L. Wang, K. Yang, Bank multiplex networks and systemic risk, Physica A 533 (2019) 122039, http://dx.doi.org/10.1016/j.physa.2019.122039.
[11] M. Bardoscia, P. Barucca, S. Battiston, F. Caccioli, G. Cimini, D. Garlaschelli, F. Saracco, T. Squartini, G. Caldarelli, The physics of financial networks, 2021, http://dx.doi.org/10.1038/s42254-021-00322-5.

[12] V. Macchiati, G. Brandi, T.D. Matteo, D. Paolotti, G. Caldarelli, G. Cimini, Systemic liquidity contagion in the European interbank market, J. Econ. Interact. Coord. (2021) http://dx.doi.org/10.1007/s11403-021-00338-1.
[13] M.O. Jackson, A. Pernoud, Systemic risk in financial networks: a survey, 2021, http://dx.doi.org/10.1146/annurev-economics.
[14] F. Allen, D. Gale, Financial contagion, J. Polit. Econ. 108 (2000) http://dx.doi.org/10.1086/262109.
[15] M. Boss, H. Elsinger, M. Summer, S. Thurner, Network topology of the interbank market, Quant. Finance 4 (2004) 677–684, http://dx.doi.org/10.1080/14697680400020325.
[16] E. Nier, J. Yang, T. Yorulmazer, A. Alentorn, Network models and financial stability, J. Econom. Dynam. Control 31 (2007) 2033–2060, http://dx.doi.org/10.1016/j.jedc.2007.01.014.
[17] P. Gai, S. Kapadia, Contagion in financial networks, Proc. R. Soc. A Math. Phys. Eng. Sci. 466 (2010) 2401–2423, http://dx.doi.org/10.1098/rspa.2009.0410.
[18] S. Poledna, S. Thurner, Elimination of systemic risk in financial networks by means of a systemic risk transaction tax, Quant. Finance 16 (10) (2016) 1599–1613, http://dx.doi.org/10.1080/14697688.2016.1156146.
[19] S. Poledna, O. Bochmann, S. Thurner, Basel III capital surcharges for G-SIBs are far less effective in managing systemic risk in comparison to network-based, systemic risk-dependent financial transaction taxes, J. Econom. Dynam. Control 77 (2017) 230–246, http://dx.doi.org/10.1016/j.jedc.2017.02.004.
[20] C. Diem, A. Pichler, S. Thurner, What is the minimal systemic risk in financial exposure networks? J. Econom. Dynam. Control 116 (2020) 103900, http://dx.doi.org/10.1016/j.jedc.2020.103900.
[21] A. Pichler, S. Poledna, S. Thurner, Systemic risk-efficient asset allocations: Minimization of systemic risk as a network optimization problem, J. Financ. Stab. 52 (2021) http://dx.doi.org/10.1016/j.jfs.2020.100809.
[22] C. Furfine, Interbank exposures: quantifying the risk of contagion, J. Money Credit Bank. 35 (2003) 111–128, http://dx.doi.org/10.1353/mcb.2003.0004.
[23] A.R. Neveu, A survey of network-based analysis and systemic risk measurement, J. Econ. Interact. Coord. 13 (2018) 241–281, http://dx.doi.org/10.1007/s11403-016-0182-z.
[24] S. Battiston, M. Puliga, R. Kaushik, P. Tasca, G. Caldarelli, Debtrank: Too central to fail? financial networks, the fed and systemic risk, Sci. Rep. 2 (2012) 541.
[25] M. Bardoscia, S. Battiston, F. Caccioli, G. Caldarelli, DebtRank: A microscopic foundation for shock propagation, PLoS One 10 (2015) http://dx.doi.org/10.1371/journal.pone.0130406.
[26] T.C. Silva, M.A. da Silva, B.M. Tabak, Systemic risk in financial systems: A feedback approach, J. Econ. Behav. Organ. 144 (2017) 97–120, http://dx.doi.org/10.1016/j.jebo.2017.09.013.
[27] M. Montagna, C. Kok, Multi-Layered Interbank Model for Assessing Systemic Risk, (Kiel Working Papers No. 1873), Kiel Institute for the World Economy (IfW Kiel), 2013, Retrieved from https://EconPapers.repec.org/RePEc:zbw:ifwkwp:1873.
[28] S. Poledna, J.L. Molina-Borboa, S. Martnez-Jaramillo, M. van der Leij, S. Thurner, The multi-layer network nature of systemic risk and its implications for the costs of financial crises, J. Financ. Stab. 20 (2015) 70–81, http://dx.doi.org/10.1016/j.jfs.2015.08.001.
[29] W. Cuba, A. Rodriguez-Martinez, D.A. Chavez, F. Caccioli, S. Martinez-Jaramillo, A network characterization of the interbank exposures in Peru, Latin Amer. J. Central Bank. 2 (2021) 100035, http://dx.doi.org/10.1016/j.latcb.2021.100035.
[30] S. Poledna, S. Martínez-Jaramillo, F. Caccioli, S. Thurner, Quantification of systemic risk from overlapping portfolios in the financial system, J. Financ. Stab. 52 (2021) 100808, http://dx.doi.org/10.1016/j.jfs.2020.100808, Network models and stress testing for financial stability: the conference.
[31] J. Cao, F. Wen, H.E. Stanley, X. Wang, Multilayer financial networks and systemic importance: Evidence from China, Int. Rev. Financ. Anal. 78 (C) (2021) http://dx.doi.org/10.1016/j.irfa.2021.10188.
[32] S. Li, M. Wang, J. He, Prediction of banking systemic risk based on support vector machine, Math. Probl. Eng. 2013 (2013) http://dx.doi.org/10.1155/2013/136030.
[33] P. Cerchiello, P. Giudici, G. Nicola, Big Data Models of Bank Risk Contagion, (DEM Working Papers Series No. 117), University of Pavia, Department of Economics and Management, 2016, Retrieved from https://EconPapers.repec.org/RePEc:pav:demwpp:demwp0117.
[34] R. Nyman, S. Kapadia, D. Tuckett, News and narratives in financial systems: Exploiting big data for systemic risk assessment, J. Econom. Dynam. Control 127 (2021) http://dx.doi.org/10.1016/j.jedc.2021.104119.
[35] M.K. So, A.S. Mak, A.M. Chu, Assessing systemic risk in financial markets using dynamic topic networks, Sci. Rep. 12 (2022) http://dx.doi.org/10.1038/s41598-022-06399-x.
[36] G. Kou, X. Chao, Y. Peng, F.E. Alsaadi, E. Herrera-Viedma, Machine learning methods for systemic risk analysis in financial sectors, Technol. Econ. Dev. Econ. 25 (2019) 716–742, http://dx.doi.org/10.3846/tede.2019.8740.
[37] A. Liu, C.Y.J. Mo, M.E. Paddrik, S.Y. Yang, An agent-based approach to interbank market lending decisions and risk implications, Information 9 (6) (2018) http://dx.doi.org/10.3390/info9060132.
[38] D. Petrone, N. Rodosthenous, V. Latora, Artificial intelligence applied to bailout decisions in financial systemic risk management, 2021, arXiv preprint arXiv:2102.02121.
[39] T.P. Lillicrap, J.J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, D. Wierstra, Continuous control with deep reinforcement learning, 2015, arXiv preprint arXiv:1509.02971.
[40] G. Dalal, K. Dvijotham, M. Vecerik, T. Hester, C. Paduraru, Y. Tassa, Safe exploration in continuous action spaces, 2018, arXiv preprint arXiv:1801.08757.
[41] Y. Maeno, S. Morinaga, K. Nishiguchi, H. Matsushima, Optimal portfolio for a robust financial system, in: 2013 IEEE Conference on Computational Intelligence for Financial Engineering & Economics, CIFEr, IEEE, 2013, http://dx.doi.org/10.1109/cifer.2013.6611695.
[42] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, M. Riedmiller, Deterministic policy gradient algorithms, in: E.P. Xing, T. Jebara (Eds.), Proceedings of the 31st International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 32, (1) PMLR, Bejing, China, 2014, pp. 387–395, Retrieved from https://proceedings.mlr.press/v32/silver14.html.
[43] C. Diem, Financial exposure network optimization via mixed integer linear programming, 2020, Retrieved from https://csh.ac.at/vis/code/network_optimization/.
[44] A. Gandy, L.A. Veraart, A Bayesian methodology for systemic risk assessment in financial networks, Manage. Sci. 63 (2017) 4428–4446, http://dx.doi.org/10.1287/mnsc.2016.2546.
[45] G. Fagiolo, Clustering in complex directed networks, Phys. Rev. E 76 (2007) 026107, http://dx.doi.org/10.1103/PhysRevE.76.026107.
[46] P. Gai, S. Kapadia, Networks and systemic risk in the financial system, Oxf. Rev. Economic Policy 35 (2019) http://dx.doi.org/10.1093/oxrep/grz023.
[47] H.F. Yang, C.L. Liu, R.Y. Chou, Bank diversification and systemic risk, Q. Rev. Econ. Finance 77 (2020) http://dx.doi.org/10.1016/j.qref.2019.11.003.
[48] D. Acemoglu, A. Ozdaglar, A. Tahbaz-Salehi, Systemic risk and stability in financial networks, Amer. Econ. Rev. 105 (2015) http://dx.doi.org/10.1257/aer.20130456.
[49] M. Teteryatnikova, Systemic risk in banking networks: Advantages of "tiered" banking systems, J. Econom. Dynam. Control 47 (2014) 186–210, http://dx.doi.org/10.1016/j.jedc.2014.08.007.
[50] R. Lowe, Y.I. Wu, A. Tamar, J. Harb, O.P. Abbeel, I. Mordatch, Multi-agent actor-critic for mixed cooperative-competitive environments, in: Advances in Neural Information Processing Systems, 2017, pp. 6379–6390.