

Dynamic Debt Swapping in Financial Networks

Henri Froese 

Goethe University Frankfurt, Germany

Martin Hoefer  

RWTH Aachen University, Germany

Lisa Wilhelmi  

RWTH Aachen University, Germany

Abstract

A *debt swap* is an elementary edge swap in a directed, weighted graph, where two edges with the same weight swap their targets. Debt swaps are a natural and appealing operation in financial networks, in which nodes are banks and edges represent debt contracts. They can improve the clearing payments and the stability of these networks. However, their algorithmic properties are not well-understood.

We analyze the computational complexity of debt swapping. Our main interest lies in *semi-positive* swaps, in which no creditor strictly suffers and at least one strictly profits. These swaps lead to a Pareto-improvement in the entire network. We consider network optimization via sequences of *v*-improving debt swaps from which a given bank *v* strictly profits. For ranking-based clearing, we show that every sequence of semi-positive *v*-improving swaps has polynomial length. In contrast, for arbitrary *v*-improving swaps, the problem of reaching a network configuration that allows no further swaps is PLS-complete. We identify cases in which short sequences of semi-positive swaps exist even without the *v*-improving property.

2012 ACM Subject Classification Theory of computation → Theory and algorithms for application domains; Networks → Network algorithms

Keywords and phrases Debt Swap, Financial Networks, Local Search

Digital Object Identifier 10.4230/LIPIcs.SAND.2025.2

Related Version *Full Version*: <https://arxiv.org/pdf/2302.11250>

Funding *Martin Hoefer*: Supported by DFG Research Unit ADYN under grant DFG 411362735 and ATHENE National Research Center for Applied Cybersecurity.

Lisa Wilhelmi: Supported by DFG Research Unit ADYN under grant DFG 411362735.

1 Introduction

The international financial system represents a complex and highly interconnected network. A prominent threat is that tight structural connections between banks, funds, and other financial institutions can yield devastating cascading effects. These systemic risks have been a major issue in the financial crisis of 2008 and several other crises since then. Mitigating these risks is an important problem, which has recently also received attention in computer science. The goal here is to understand the computational complexity of network-based interbank models for financial systems and potential interventions.

A fundamental aspect of modeling financial systems is *clearing*, a process of debt resolution by which solvency of banks in the system gets determined. In a financial network (where banks are nodes, and weighted, directed edges correspond to debt contracts) the clearing problem asks for a network flow that represents a consistent assignment of payments of debtor banks to their creditors. Based on the clearing, a bank might not receive the full amount of credit it gave to others. As a consequence, it might not have sufficient funds to clear all her own debt. Naturally, the solution to the clearing problem is of central importance to banks since it directly affects their assets.



© Henri Froese, Martin Hoefer, and Lisa Wilhelmi;
licensed under Creative Commons License CC-BY 4.0

4th Symposium on Algorithmic Foundations of Dynamic Networks (SAND 2025).

Editors: Kitty Meeks and Christian Scheideler; Article No. 2; pp. 2:1–2:16

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Banks have an obvious incentive to improve their clearing conditions. Moreover, governmental regulators, central banks, and other financial services share an interest in maintaining and improving the financial system. While allocation rules for clearing are often fixed, other interventions for improvement have the potential to significantly improve the clearing conditions. A particularly interesting variant of network improvement is *debt swapping* [15]: There are two debt contracts among two pairs of banks (u_1, v_1) and (u_2, v_2) . The contracts have the same value. In a debt swap, the creditor banks v_1 and v_2 change their role as recipients of these contracts. This leaves every bank with the same amounts of debt as before – only the network is affected by the exchange of the edges. A debt swap is a local edge-swap operation and represents a small change in the network structure. Since it involves only a small number of banks, it can be executed in a distributed fashion by the participating banks. Banks are more likely to agree to this operation if it does not deteriorate their network position, i.e., it does not decrease their assets in the clearing of the resulting network.

Debt swaps are a very natural and interesting swap operation with many conceptual advantages. For example, they are usually less demanding than the more prominent approach of portfolio compression [19, 22] (for a discussion of the relation between these two operations, see [15]). Moreover, seemingly different operations proposed by regulators (such as, e.g., claims trades) can be interpreted as debt swaps [6]. From a computational perspective, debt swaps are related to a substantial body of literature on edge swap operations in graphs. Numerous variants of such swaps have attracted a large interest in computer science over the decades (see related work discussed below). The computational aspects of debt swapping, however, are not well understood.

In this paper, we strive to understand debt swap dynamics and their effects on the network. Since a debt swap is a local improvement step in the network, we first focus on local optimization (e.g., from an initial network, can we reach a configuration that allows no more profitable debt swaps?). We discuss all our results consistently for networks in which banks use natural ranking-based clearing payments [2, 1, 9]. Indeed, many of our results can be extended to arbitrary monotone clearing. Throughout the paper we mention the extension explicitly whenever it applies.

Contribution and Techniques. After discussing preliminaries in Section 2, we classify debt swaps into positive (both creditors profit strictly in the clearing after the swap), semi-positive (one creditor profits strictly, the other one remains the same), or arbitrary ones. Our characterization results in Section 3 show that in every financial network, there can be no positive swap. Semi-positive swaps can exist, and, in fact, they always represent a *Pareto-improvement*: After the swap *no bank in the entire network suffers*, every bank obtains at least as many assets as before. As such, the *recovery rate* of every bank weakly increases, as well as the payments towards every debt contract. This makes semi-positive swaps very attractive. Similar results were shown by Papp and Wattenhofer [15] for proportional payments. Our paper generalizes these results substantially to *all monotone payments*.

We then concentrate on improvement dynamics with semi-positive debt swaps. Since they Pareto-improve the assets of all banks (and strictly for a creditor bank), every sequence of semi-positive swaps is inherently acyclic and terminates in a local optimum. A natural and challenging question that we study in Section 4 is whether there are short sequences that allow to reach a local optimum in polynomial time. We further refine the characterization of semi-positive swaps for ranking-based payments into *saturating*, and *extension* swaps, and further into *active*, *semi-* and *non-active* extension swaps. We show that there can be at

most a polynomial number of saturating, semi- and non-active extension swaps, so the core of the problem lies in the existence of short sequences of active extension swaps. When a single bank v strictly profits from every semi-positive swap, then we prove that every sequence of active extension swaps has indeed a length polynomial in the size of the network (and, thus, so has every sequence of semi-positive swaps). Instead, when we consider arbitrary swaps in which a given bank v strictly improves, then there are initial networks from which *every* sequence of such swaps has exponential length. More fundamentally, we prove that it is PLS-complete to construct any local optimum, i.e., any network where liabilities of each node are consistent with the initial network and that allows no (arbitrary) debt swap in which the assets of a given bank v strictly increase.

The problem without the property of strict improvement for a single bank v turns out to be extremely challenging. We also make some progress here – under a condition on the liabilities and the in-degree of partly paid edges of every bank, we can show existence of a short sequence of semi-positive swaps to a local optimum. The general problem remains as an interesting avenue for future work.

All missing proofs are deferred to the full version of this paper. Moreover, in the full version we also consider aspects of global optimization (i.e., maximize the assets of a single bank via a sequence of debt swaps) and reachability problems (i.e., decide if a target network structure can be reached using a sequence of debt swaps).

Related Work. There has been a recent surge of interest in the algorithmic and game-theoretic aspects of financial networks. Most works adopt and extend the clearing model by Eisenberg and Noe [4] (discussed in more detail below), where banks are nodes, edges are debt contracts, and clearing is governed by proportional payments. Most closely related to our work is recent work by Papp and Wattenhofer [15], who introduce and study the concept of debt swaps for proportional payments. They show that positive swaps do not exist, semi-positive swaps lead to Pareto-improvement, discuss shock models and hardness of a number of optimization problems. Another network-altering operation was recently introduced by Kanellopoulos et al. [11]. A debt transfer describes the process of a bank transferring a debt claim to another creditor. Formally, this operation exchanges a path of length two by one single edge. The authors consider optimization problems and a game-theoretic model where the banks strategically decide whether or not to perform a debt transfer.

More generally, algorithmic aspects of (extensions of) the Eisenberg-Noe model have been subject to substantial recent interest, including, e.g., the computation of clearing states in networks with credit-default swaps (CDS) [20, 9, 8]. Some clearing models (e.g., using CDS'es) suffer from default ambiguity [21], and several strategies for resolving this problem have been proposed [16]. Changes of the instance in terms of adding or deleting single liabilities, donations to other banks, lending, manipulation of external assets, allocation of stimulus checks, or bailouts were analyzed in [14, 10, 3, 13]. While most works assume the clearing state to manifest as a fixed point, the impact of *sequential* updates is studied in [17]. Further temporal aspects are investigated in [5], where liabilities are associated with individual maturity intervals. The works investigate the complexity of allocating integral and fractional payments in order to minimize defaults.

There is a growing interest in clearing problems beyond the restriction to proportional payments, using more general classes of monotone functions to model the clearing process [2]. This extension poses many interesting algorithmic and, in particular, game-theoretic problems that are very recently starting to get explored [1, 12, 7].

2 Model and Preliminaries

Financial Networks, Clearing, and Ranking. A *financial network* $\mathcal{F} = (G, \text{de}, \text{cr}, \mathbf{l}, \mathbf{a}^x, \mathbf{f})$ is given by a weighted, directed multi-graph $G = (V, E)$ with $|V| = n$ nodes and $|E| = m$ edges. Each node $v \in V$ resembles a financial institution or *bank*. Each edge $e \in E$ represents a debt contract, where a debtor bank $\text{de}(e) \in V$ owes an amount of $l_e \in \mathbb{N}_{>0}$ to a creditor bank $\text{cr}(e)$. Formally, e is a directed edge $(\text{de}(e), \text{cr}(e))$ with weight l_e . We assume that there are no loops, i.e., $\text{de}(e) \neq \text{cr}(e)$. The set of incoming and outgoing edges of v are denoted by $E^-(v) = \{e \in E \mid \text{cr}(e) = v\}$ and $E^+(v) = \{e \in E \mid \text{de}(e) = v\}$. The *total liabilities* of a bank v are defined as $L_v = \sum_{e \in E^+(v)} l_e$. Further, every bank holds *external assets* $a_v^x \in \mathbb{N}$, i.e., assets of v that are independent of the network. The payments that a bank receives from its debtors within network are called *internal assets* and, together with external assets, form the available assets of a bank.

An *allocation rule* is a collection $\mathbf{f} = (\mathbf{f}_v)_{v \in V}$, where for bank v $\mathbf{f}_v = (f_e)_{e \in E^+(v)}$ determines a distribution of the available assets $b_v \in \mathbb{R}_{\geq 0}$ towards her liabilities. $f_e : \mathbb{R}_{\geq 0} \rightarrow [0, l_e]$ is the *payment function* for edge $e \in E^+(v)$. For every bank v , the payment functions satisfy the following properties: (1) f_e is continuous¹ and non-decreasing in b_v (monotonicity), (2) each bank pays at most the respective liability, i.e., $f_e(b_v) \leq l_e$ (edge capacity), and (3) every bank spends all assets until all liabilities are paid off, i.e., $\sum_{e \in E^+(v)} f_e(b_v) = \min\{b_v, L_v\}$ (limited flow conservation). The available assets of v are the sum of external and internal assets $b_v = a_v^x + \sum_{e \in E^-(v)} f_e(b_{\text{de}(e)}) \geq 0$.

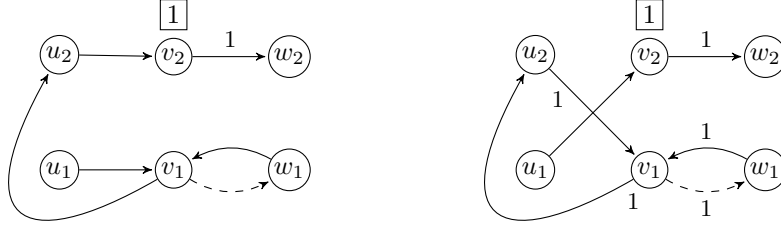
These constraints give rise to a fixed-point problem. A vector of assets $(b_v)_{v \in V}$ that fulfills all constraints is called *feasible*. For every allocation rule \mathbf{f} , the set of feasible vectors forms a complete lattice w.r.t. coordinate-wise comparison [1, 2]. Consequently, for every allocation rule \mathbf{f} there exists a supremum $(a_v)_{v \in V}$ of feasible assets and resulting payments $\mathbf{p} = (p_e)_{e \in E}$ with $p_e = f_e(a_{\text{de}(e)})$ that pointwise maximize the assets of each bank and the payments on each edge. The clearing settles on these maximal payments, which we term the *clearing state*. In the clearing state, bank v receives *total incoming assets* of $\sum_{e \in E^-(v)} p_e$ and holds *total assets* of $a_v = a_v^x + \sum_{e \in E^-(v)} p_e$.

In this paper, we will discuss all our results consistently in the context of a natural class of *edge-ranking* rules [1, 2, 9]: For each bank v , her outgoing edges $E^+(v)$ are ordered by a permutation π_v . First, v directs all her payments towards edge $\pi_v(1)$ until the edge is completely paid off or v has no remaining assets. Then, she pays off edge $\pi_v(2)$ until the edge is completely paid off or v has no remaining assets. This process continues until either all debt is settled or v runs out of funds.

Apart from being natural and well-motivated in applications, edge-ranking rules also have appealing computational properties. Given a financial network with edge-ranking rules, a maximal clearing state can be computed in strongly polynomial time [1]. Moreover, these favorable properties extend to general monotone integral [2, 1] allocation rules in explicit representation by using edge-rankings over a suitable set of auxiliary edges [1]. In this way, *all our statements generalize* to unit-ranking rules with edge-ranking representation.

Other classes of payments have also been of interest in the literature (e.g., *proportional payments*, where every bank allocates her total assets in proportion to her total liabilities, or combinations of ranking and proportional). Indeed, many of our results extend even to arbitrary monotone allocation rules.

¹ Monotonicity, edge capacity, and limited flow conservation together imply continuity. We postulate it explicitly for convenience.



■ **Figure 1** The figures visualize a financial network \mathcal{F} before (left) and \mathcal{F}^σ after a debt swap (right). All edges have unit weight and edge labels indicate payments. For v_1 , the solid edge is the preferred outgoing edge. v_2 has external assets of 1 as shown in the node label.

Debt Swaps. For a given financial network \mathcal{F} , a debt swap σ is a structural change to the network. For two edges e_1, e_2 among four distinct banks with the same liability (i.e., $l_{e_1} = l_{e_2} =: \ell$) we exchange their creditors. The result can be described by an adjustment in the creditor function by swapping the entries of e_1 and e_2 . Note that the allocation rule \mathbf{f} of the banks remains unaffected by the swap, since the debtor function remains unchanged. However, due to the structural change, a debt swap may change the payments in the clearing state. We denote the resulting financial network, creditor function, clearing state, and total assets of a bank v after the swap by \mathcal{F}^σ , \mathbf{cr}^σ , \mathbf{p}^σ and a_v^σ , respectively.

► **Definition 1 (Debt Swap).** Consider a financial network \mathcal{F} with edges $e_1, e_2 \in E$. Throughout we use the short notation $v_1 = \mathbf{cr}(e_1)$, $v_2 = \mathbf{cr}(e_2)$, $u_1 = \mathbf{de}(e_1)$ and $u_2 = \mathbf{de}(e_2)$. Suppose that the liabilities $l_{e_1} = l_{e_2}$ and u_1, u_2, v_1, v_2 are pairwise distinct nodes. A debt swap σ of e_1 and e_2 creates a new network $\mathcal{F}^\sigma = (G, \mathbf{de}, \mathbf{cr}^\sigma, \mathbf{l}, \mathbf{a}^x, \mathbf{f})$ where $\mathbf{cr}^\sigma(e_1) = v_2$, $\mathbf{cr}^\sigma(e_2) = v_1$, and $\mathbf{cr}^\sigma(e) = \mathbf{cr}(e)$ for all $e \in E \setminus \{e_1, e_2\}$.

Observe that a debt swap may introduce multi-edges even in initially simple graphs. We proceed with a small example that shows some of the interesting effects of debt swaps.

► **Example 2.** Consider the initial financial network in Fig. 1 (left) with edge-ranking allocation rules. Bank v_1 first pays off all debt to u_2 before directing payments to w_1 , i.e., $\pi_{v_1}(1) = (v_1, u_2)$, $\pi_{v_1}(2) = (v_1, w_1)$. All other banks pay towards their unique outgoing edge. Consider the clearing state. For bank v_2 the external assets of 1 equal her total liabilities L_{v_2} . Thus, v_2 can settle all her debt towards w_2 . For the other edges, suppose there are positive payments on edge (u_2, v_2) . Then, by definition, u_2 has to have these assets available, so v_1 must pay this amount to u_2 . By the same argument, u_1 must have positive total assets and pay them to v_1 . However, u_1 obviously has no assets, and there are no payments on these edges. Due to the allocation rule of v_1 , there are no payments in the cycle of v_1 and w_1 , unless v_1 fully pays off (v_1, u_2) first (for which she needs assets from u_1). The clearing state yields a payment of 1 on (v_2, w_2) . The total assets are 1 for v_2 and w_2 , and 0 for all others.

Now consider the financial network in Fig. 1 (right) after the swap of edges (u_1, v_1) and (u_2, v_2) to (u_1, v_2) and (u_2, v_1) . Obviously, v_2 can still pay off her single outgoing edge. After the swap, there is a cycle including v_1 and u_2 over prioritized edges. When the payments on both edges are raised to 1, the fixed-point properties remain satisfied. Since v_1 pays off all debt on the edge with highest priority, she will direct any additional assets towards the edge of her second priority to w_1 . Here a new cycle with w_1 over prioritized edges arises, and the fixed-point conditions allow to further raise payments along that cycle to 1.

The additional payments after the swap result from the emergence of the cycle between u_2 and v_1 . The total assets of v_1, u_2 and w_1 are increased to $a_{v_1} = 2$, $a_{u_2} = 1$ and $a_{w_1} = 1$ after the swap. For the total assets of the other banks the swap is inconsequential. ┘

A debt swap of two edges e_1 and e_2 maintains the local network properties of the debtors u_1 and u_2 . It has no effect on the set of liability values. Moreover, it does not impact their incoming edges. As such, we are interested in the gains and losses in total assets of the creditors v_1 and v_2 , since (arguably) they are the key actors to implement a debt swap. The following classification of debt swaps slightly extends the one in [15]. Observe that the debt swap explained in Example 2 is semi-positive and Pareto-improving.

► **Definition 3 (Swap Types).** *A debt swap σ of edges e_1 and e_2 is called (1) positive if $a_{v_1}^\sigma > a_{v_1}$ and $a_{v_2}^\sigma > a_{v_2}$, (2) semi-positive if $a_{v_1}^\sigma \geq a_{v_1}$ and $a_{v_2}^\sigma \geq a_{v_2}$ and exactly one inequality is strict, (3) Pareto-improving if $a_v^\sigma \geq a_v$ for all $v \in V$ and at least one inequality is strict.*

3 Characterization

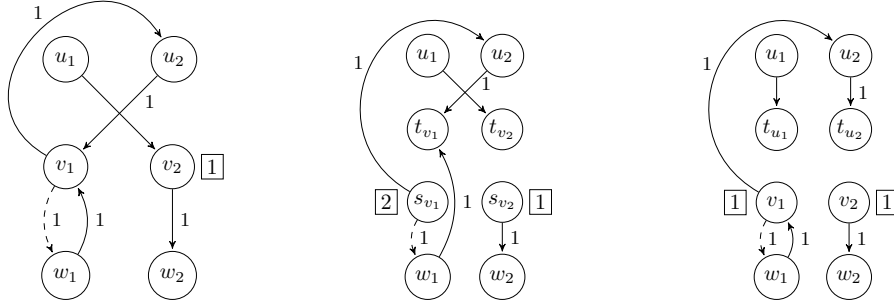
To reason about debt swaps, we first provide a structural analysis of the clearing payments before and after debt swaps. We obtain a characterization of semi-positive and positive swaps. The analysis substantially generalizes techniques from [15] and yields two major insights. First, there are no positive swaps. Second, every semi-positive debt swap yields a Pareto-improvement in the *entire network*, i.e., no bank in the financial network is harmed while at least one bank strictly profits. Thus, semi-positive swaps can only improve the recovery rate of each bank (i.e., the ratio of available assets over liabilities). Consequently, the payment assigned to each debt contract does not decrease. Interestingly, the converse also holds: Every Pareto-improving debt swap is semi-positive.

Similar to [15], we use an auxiliary network with the same payments as the resulting network after the debt swap. In this auxiliary network, we replace the swap edges with auxiliary sources and sinks with appropriate assets representing the payments on the swap edges. For a contradiction, we assume that the debt swap is positive. By tracing the payments in the auxiliary network, we show that payments in the original network before the swap did not correspond to the maximum clearing state. This contradiction shows that positive swaps cannot exist. As a rather direct consequence from this characterization, we obtain that semi-positive swaps lead to a Pareto-improvement in the entire network.

The main distinction from the proofs in [15] for proportional payments is that these payments have limited structure in the auxiliary network and, as such, are easier to handle. Instead, our proofs rely only on monotonicity of the allocation rule. By arguing about a broad range of possibilities and dependencies of potential payments, we can show that these results indeed apply to all financial networks with *any monotone allocation rule*.

3.1 Structural Properties

Monotonicity of the allocation rule implies that when the total assets of bank v increase, then payments on each $e \in E^+(v)$ do not decrease. Intuitively, therefore, every clearing state of a financial network can be decomposed into two clearing states of appropriate networks. We proceed in two steps: First, each bank v is assigned reduced external assets $\hat{a}_v^x \leq a_v^x$. This results in a new financial network, for which we get the first clearing state. In this state, v (partially) pays her liabilities. For the second step, we reduce each liability by the payment in the first step, and v receives external assets of $a_v^x - \hat{a}_v^x$. This again results in a new financial network, for which we get the second clearing state. The sum of the two clearing states is the clearing state of the original network.



■ **Figure 2** The network on the left corresponds to the network \mathcal{F}^σ in Example 2. Black edge labels indicate payments. The middle network results from applying source-sink equivalency to v_1 and v_2 , whereas the right network results from applying source-sink equivalency for edges (u_2, v_1) and (v_2, u_1) .

This separability condition can be used to show that increasing the external assets of one bank Pareto-improves the total assets of every bank.

► **Lemma 4 (Monotonicity).** *Suppose two financial networks \mathcal{F} and $\hat{\mathcal{F}}$ differ only in the external assets of a single bank v such that $a_v^x > \hat{a}_v^x$. For every bank $u \in V$, the total assets in the clearing state satisfy $a_u \geq \hat{a}_u$.*

The payments in a financial network emerge due to a combination of external assets and cyclic money flow. Clearly, a sink node t (with $E^+(t) = \emptyset$) is not included in any cycle. Thus, all incoming payments of a sink node originated from external assets of other banks.

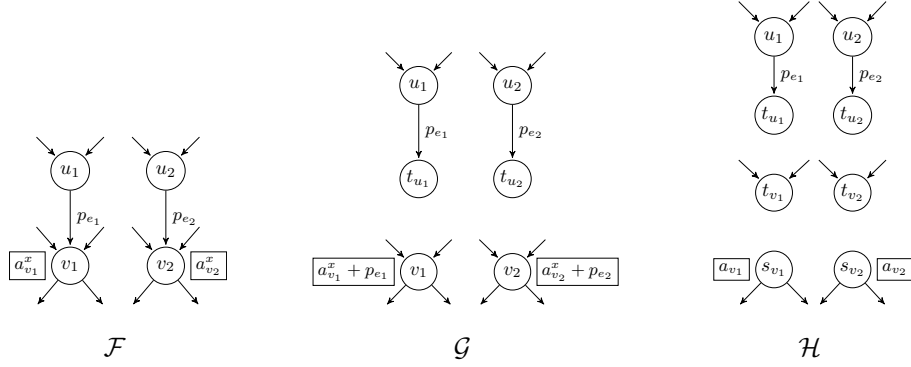
Indeed, we can show that the increase of $\Delta > 0$ in external assets of a source node s (with $E^-(s) = \emptyset$) raises the total assets of all sinks in sum by at most Δ .

► **Lemma 5 (Non-Expansivity).** *For a given financial network, let $T \subset V$ be the set of all sink nodes. If the external assets of a source $s \in V \setminus T$ are increased to $a_s'^x = a_s^x + \Delta$, then the total assets of all sinks increase by at most Δ , i.e., $\sum_{t \in T} a_t' \leq \Delta + \sum_{t \in T} a_t$.*

A similar property can be shown if s is not a source bank. The argument uses an auxiliary construction that externalizes the incoming payments of bank $s = v$ by splitting v into two separate nodes s_v and t_v . The sink t_v holds the same incoming edges as v in the original graph. Similarly, source s_v has the same outgoing edges as v . Setting the external assets of s_v to the total assets a_v of v in the clearing state ensures that the payments of s_v and v are equal. We can then verify that the modified network has the same clearing state as the original network, and, in particular, t_v has the same incoming payments as v . Clearly, the nodes representing v are not contained in any cycle.

► **Definition 6.** *For a given financial network \mathcal{F} with bank v , we define an adjusted network \mathcal{H} as follows. We replace v by a source node s_v and a sink node t_v . Sink t_v has external assets 0 and the same set of incoming edges as v , i.e., $cr_{\mathcal{H}}(e) = t_v$ for every $e \in E^-(v)$ and $E_{\mathcal{H}}^-(t_v) = E^-(v)$. Source s_v has external assets a_v and the same outgoing edges as v , i.e., $de_{\mathcal{H}}(e) = s_v$ for every $e \in E^+(v)$, and $E_{\mathcal{H}}^+(s_v) = E^+(v)$. Source s_v in \mathcal{H} uses the allocation rule of v in \mathcal{F} .*

An example construction can be seen in Fig. 2, where banks v_1 and v_2 are replaced by sources and sinks. The next lemma shows that the clearing state in the auxiliary network is equivalent to the one in the original network.



■ **Figure 3** Schematic illustration of networks \mathcal{G} and \mathcal{H} for a financial network \mathcal{F} .

► **Lemma 7** (Source-Sink-Equivalency). *For a given financial network \mathcal{F} and an adjusted network \mathcal{H} it holds that*

- (1) *the networks have equivalent clearing states, i.e., $p_e^{\mathcal{H}} = p_e$ for all $e \in E$.*
- (2) *the sink t_v has total assets $a_v - a_v^x$.*

A very similar construction can be used to externalize the payments on an edge. The right image in Fig. 2 depicts an example construction.

► **Corollary 8** (Source-Sink-Equivalency for Edges). *For a given financial network \mathcal{F} with edge e , let $u = de(e)$ and $v = cr(e)$. Add the sink t_v and set the target of e to $cr(e) = t_v$. Increase the external assets of v by p_e . Then, the resulting financial network \mathcal{G} has the same clearing state as \mathcal{F} , i.e., $p_e^{\mathcal{G}} = p_e$ for all $e \in E$.*

3.2 Existence of Debt Swaps

Example 2 shows that semi-positive and Pareto-improving debt swaps can exist, for both proportional and edge-ranking rules.

► **Observation 9.** *For proportional payments and edge-ranking rules, financial networks with semi-positive and Pareto-improving debt swaps exist.*

Our main result in this section shows that every financial network with a monotone allocation rule has no positive debt swap.

► **Theorem 10.** *A financial network with monotone allocation rules allows no positive debt swap.*

To prove this result, we assume for a contradiction there exists a network \mathcal{F} with a positive swap of edges e_1 and e_2 . To analyze the payments in \mathcal{F} , source-sink-equivalency is applied to the swap edges. Let \mathcal{G} denote the resulting network, see Fig. 3 for an illustration. To accommodate for the reduced incoming payments $a_{v_1} - a_{v_1}^x - p_{e_1}$ of v_1 , the external assets are set to $a_{v_1}^x + p_{e_1}$. The external assets of v_2 are set analogously.

The network \mathcal{H} is obtained by additionally applying source-sink equivalency to v_1 and v_2 in \mathcal{G} . Then, t_{v_1} and t_{v_2} receive incoming payments of $a_{v_1} - a_{v_1}^x - p_{e_1}$ and $a_{v_2} - a_{v_2}^x - p_{e_2}$, respectively. This construction has no impact on the payments in the clearing state. Analogously, the networks \mathcal{G}^σ and \mathcal{H}^σ are constructed for the network \mathcal{F}^σ after the debt swap. The networks $\mathcal{G}, \mathcal{G}^\sigma$ and $\mathcal{H}, \mathcal{H}^\sigma$ have the same cycles but differ in external assets, resp.

The next technical lemma characterizes the effects of increasing external assets of s_{v_1} and s_{v_2} in \mathcal{H} .

Let \mathcal{H}_Δ be the resulting network when increasing the external assets of s_{v_1} and s_{v_2} in \mathcal{H} by $\delta_{s_{v_1}} \in [0, a_{v_1}^\sigma - a_{v_1}]$ and $\delta_{s_{v_2}} \in [0, a_{v_2}^\sigma - a_{v_2}]$, respectively. We use Δ_w to denote the increase in total assets of bank w , i.e., $\Delta_w = a_w^{\mathcal{H}_\Delta} - a_w$. We use \mathcal{F}_Δ to denote the network when reversing the source-sink equivalency both for t_{v_1}, s_{v_1} and t_{v_2}, s_{v_2} as well as for t_{u_1} and t_{u_2} .

► **Lemma 11.** *For \mathcal{H}_Δ and \mathcal{F}_Δ the following are true:*

- (1) $\delta_{s_{v_1}} + \delta_{s_{v_2}} = \Delta_{t_{v_1}} + \Delta_{t_{v_2}} + \Delta_{t_{u_1}} + \Delta_{t_{u_2}}$.
- (2) If $\delta_{s_{v_1}} > 0$ or $\delta_{s_{v_2}} > 0$, then $\delta_{s_{v_1}} \neq \Delta_{t_{v_1}} + \Delta_{t_{u_1}}$ and $\delta_{s_{v_2}} \neq \Delta_{t_{v_2}} + \Delta_{t_{u_2}}$.

Suppose there exists a positive debt swap for a financial network \mathcal{F} , in which the total assets of both creditor banks v_1 and v_2 strictly increase by $\delta_{v_1} > 0$ and $\delta_{v_2} > 0$. In \mathcal{H} the debt swap is simulated by increasing the total assets of source nodes s_{v_1} and s_{v_2} by δ_{v_1} and δ_{v_2} , respectively. We will show that there exists a suitable set of values for δ_{v_1} and δ_{v_2} such that v_1 receives δ_{v_1} additional incoming payments. Intuitively, then, the pairs u_1, v_1 and u_2, v_2 are each contained in the same cycle. Hence, payments could be raised along the cycles, which contradicts maximality of the clearing state. This idea can be used to complete the proof of Theorem 10.

The theorem shows that both creditors participating in a debt swap cannot simultaneously strictly increase their total assets. In the next part, we consider the payments on the edges e_1 and e_2 before and after a generic swap σ . The first proposition shows they cannot both be strictly increased.

► **Proposition 12.** *For every financial network, there is no debt swap σ such that $p_{e_1}^\sigma > p_{e_1}$ and $p_{e_2}^\sigma > p_{e_2}$.*

► **Proposition 13.** *For every semi-positive debt swap σ , we have either $p_{e_1}^\sigma > p_{e_1}$ and $p_{e_2}^\sigma \leq p_{e_2}$, or $p_{e_2}^\sigma > p_{e_2}$ and $p_{e_1}^\sigma \leq p_{e_1}$.*

Towards semi-positive swaps, the total assets of v_1 and v_2 Pareto-improve. Perhaps surprisingly, the second main result of this section shows that a swap σ is semi-positive if and only if it Pareto-improves the assets of every bank in the network. In particular, no bank in the financial network is harmed while at least one bank strictly profits. The proof of the theorem uses our previous results on source-sink-equivalency and an auxiliary network \mathcal{H} .

► **Theorem 14.** *For every financial network with monotone allocation rules, a debt swap is semi-positive if and only if it is Pareto-improving.*

4 Debt Swap Dynamics

When two creditors perform a debt swap, the structure of the underlying graph changes. This might enable further swaps which potentially further increase the assets of the involved banks. In this way, the swap operation naturally gives rise to *dynamics* where, in each step, a pair of creditors agree to perform a swap in the current network. Conceptually, semi-positive swaps are very attractive – they are equivalent to Pareto-improving swaps and improve the situation not only for the creditors but for the entire network.

In this section, we consider natural distributed dynamics of debt swaps. Given an initial financial network with monotone allocation rules, our goal is to find a small number of semi-positive debt swaps to reach a local optimum, i.e., a network structure without semi-positive

swaps. We denote this as the SEMISWAP problem. While all banks (weakly) profit from a semi-positive swap, we are interested in semi-positive swaps that are v -improving, i.e., *strictly increase the assets of a given bank v* . This is natural, e.g., when v is a creditor that profits strictly in every such swap, or v is a bank that is “too big to fail” and should strictly profit from network changes. We denote the problem with semi-positive v -improving swaps by SEMISWAP(v). Given a financial network, can we reach a local optimum by a sequence of polynomially many semi-positive v -improving swaps?

We consider this question for the broad class of edge-ranking rules. In a given network with m edges, there are at most $\binom{m}{2}$ edge swaps. Recall that we can compute the clearing state for edge-ranking rules in strongly polynomial time [1]. Hence, for a given network, we can also enumerate all semi-positive debt swaps in strongly polynomial time.

4.1 Sequences of Semi-Positive Swaps

Characterization. Consider a bank v with edge-ranking rule. v pays her outgoing edges in the order of a permutation π_v until she runs out of funds. Assume v cannot settle all debt. Let e denote the unique highest-ranked outgoing edge of v that is not fully paid. We call e the *active edge* of v . Clearly, all edges e' with higher priority than e are fully paid for, i.e., they are *saturated*. Additionally, all edges e' with lower priority than e receive no payments. Note that the set of all active edges is *acyclic* – we can strictly increase the feasible payments along any cycle, which is impossible when the clearing state has maximal feasible payments.

Since the active outgoing edge of every bank is unique, the subgraph of active edges is simple and never contains multi-edges. When analyzing this subgraph, we use the notation for simple graphs, i.e., $e = (u, v)$ if $\text{de}(e) = u$ and $\text{cr}(e) = v$, for $e \in E$. More generally, we will use the standard notation (u, v) whenever it unambiguously describes one edge.

A semi-positive swap Pareto-improves the total assets of all banks. It can only increase the number of saturated edges. If a semi-positive swap saturates at least one additional edge, we call it a *saturating swap*. As a second class, *extension swaps* are non-saturating swaps with an interesting structural property.

► **Definition 15 (Extension Swap).** Consider a semi-positive debt swap σ of edges e_1 and e_2 with $p_{e_1} < p_{e_2}$. The swap is called *extension swap* if (1) no additional edge gets saturated and (2) in \mathcal{F} there is an active path P , i.e., a path of active edges from v_1 to v_2 , (3) $l_e - p_e > p_{e_2} - p_{e_1}$ for every $e \in P$, (4) in the clearing state $p_{e_1} = p_{e_1}^\sigma$ and $p_{e_2} = p_{e_2}^\sigma$.

In an extension swap, creditors v_1 and v_2 swap two edges e_1 and e_2 with different payments, where w.l.o.g. we assume $p_{e_2} > p_{e_1}$. The swap does not create a cycle of active edges, this would yield a new saturated edge along this cycle. Therefore, v_1 receives in \mathcal{F}^σ additional incoming assets of $\delta_{v_1} = p_{e_2} - p_{e_1} > 0$. Since there exists a path P from v_1 to v_2 of active edges, each with residual liability of more than δ_{v_1} , all additional assets get routed to v_2 . The total assets of v_2 after the swap remain the same (see Fig. 4 for an example).

The next proposition proves an interesting and very useful characterization.

► **Proposition 16.** For a given financial network with edge-ranking rules, every semi-positive debt swap is either a saturating or an extension swap.

We further classify extension swaps based on the activation status of the involved edges.

► **Definition 17 (Classes of Extension Swaps).** Consider extension swap σ of edges e_1 and e_2 with $p_{e_1} < p_{e_2}$. In a *non-active swap* both edges are nonactive. In a *semi-active swap*, either e_1 or e_2 is active. In an *active swap*, both e_1 and e_2 are active.



■ **Figure 4** Minimal example of an extension swap where all edges have weight $M > k$ and edge labels indicate payments in the clearing state. The swap extends the path of u_2 's payments to reach v_2 , increasing v_1 's total assets of v_1 along the way.

Observe that in a non-active swap, the edges e_1 and e_2 are each either fully saturated or receive no payments. Since $p_{e_1} < p_{e_2}$, we have both $p_{e_1} = 0$ and $p_{e_2} = l_{e_2}$. In a semi-active swap, $p_{e_1} = 0$ or $p_{e_2} = l_{e_2}$. Finally, in an active swap, $0 \leq p_{e_1} < p_{e_2} < l_{e_2}$.

Periods, Phases, Active Extension Swaps. We now discuss constructing short sequences of swaps and start with saturating swaps.

► **Observation 18.** *Given an initial financial network with edge-ranking rules and m edges, every sequence of semi-positive debt swaps can contain at most m saturating ones.*

We define a *period* as a sequence of consecutive extension swaps. Every sequence of semi-positive swaps has at most $m + 1$ periods. We turn attention to non-active and semi-active extension swaps. Every swap has exactly one non-profiting creditor. We classify non-active and semi-active swaps in a period based on this bank.

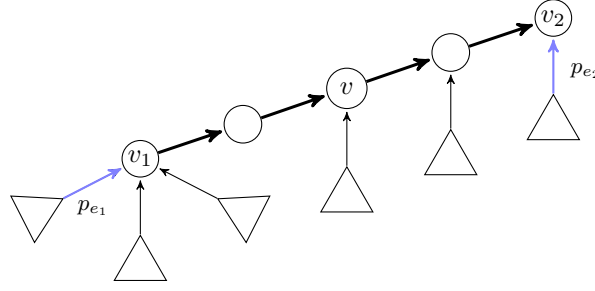
► **Lemma 19.** *For every $v \in V$, every single period can contain at most $2 \cdot |E^-(v)|$ many non-active or semi-active extension swaps such that v is the non-profiting creditor.*

Proof. For each semi-active swap of this kind, one of two possible events occurs: Either (1) an incoming non-active saturated edge of v is changed to an active (non-saturated) one, or (2) an incoming active edge of v with positive payment is changed to a non-active one with 0 payment. In a non-active swap, a sequential composition of both events occurs: An incoming non-active saturated edge of v is changed directly to a non-active one with 0 payments.

In a semi-positive swap, the payment on every edge can only weakly increase. No edge gets saturated (this would end the period). Hence, no active edge can become non-active (neither saturated, nor with 0 payments). Events (1) and (2) cannot re-occur for an incoming edge of v . There are at most $2 \cdot |E^-(v)|$ events for node v in the period, and at most $2 \cdot |E^-(v)|$ many non-active or semi-active extension swaps with v as the non-profiting creditor. ◀

Hence, from an initial network with m edges, there are at most $O(m^2)$ saturating, non-active, or semi-active extension swaps. The challenge lies in bounding the number of active extension swaps. We define a *phase* as a sequence of consecutive active extension swaps. For bounding the length of swap sequences to a polynomial, Observation 18 and Lemma 19 allow to restrict attention to the number of swaps in a single phase.

Recall that the set of active edges is cycle-free. Every node has at most one outgoing active edge. Consequently, the set of active edges forms a forest of in-trees, where all edges are directed towards the parent node. We denote the set of these in-trees by \mathcal{T} . In an active extension swap, the active path P runs from v_1 to v_2 . As such, we can only have active extension swaps where all four banks involved in the swap are in the same tree $T \in \mathcal{T}$, and the non-profiting creditor $v_2 \in T$ is an ancestor of all $v_1, u_1, u_2 \in T$.



■ **Figure 5** Schematic picture of a v -improving semi-positive swap of active edges where payments $p_{e_2} > p_{e_1}$. Triangles indicate subtrees. When the blue edges are swapped then payments on the thick black edges strictly increase.

We first focus on sequences of active extension swaps that strictly improve a given bank v . Afterwards, we consider the more general case of arbitrary active extension swaps, where we do not condition on a particular bank benefiting.

Improvement for a Given Bank. We analyze sequences of semi-positive swaps that are v -improving for a given bank v , i.e., in every swap v strictly profits. Note that v may not be involved in the swap as a creditor bank herself. Our main result is the following.

► **Theorem 20.** *For every bank v in a financial network with edge-ranking rules, every sequence of v -improving semi-positive swaps has polynomial length. $\text{SEMISWAP}(v)$ can be solved in polynomial time.*

Proof. It suffices to prove the result for active extension swaps in a single phase. Consider the tree T containing v . Suppose there is an active extension swap, then the set of banks that strictly profit lie on the active path P , with the exception of creditor v_2 . Hence, if such a swap is also v -improving, v is a node on the active path, the creditor v_1 is v or a descendant of v , and the creditor v_2 is an ancestor of v . As such, all active edges on the path P_v from v to the root of T cannot be swapped. In terms of payments, the edges whose payments are strictly increased by the swap are located in the subtree T_v rooted at v or on the path P_v . The swap keeps all other payments the same.

Note that for the swap, $p_{e_1} < p_{e_2}$, i.e., the edge e_2 with higher payment is incoming to the ancestor v_2 of v and gets swapped to $v_1 = v$ or a descendant v_1 of v . For v_2 , the only incoming edge that increases in payment is the one on P_v . All other incoming edges of v_2 only keep their payments (when not being swapped) or are replaced by ones with strictly decreased payments (when being swapped). In contrast, for each edge, payments are non-decreasing over time. This implies that if edge e_2 is swapped away from v_2 for some edge e_1 , edge e_2 cannot re-appear at v_2 via a swap with e_1 or any edges swapped with e_1 . For a schematic illustration, see Fig. 5.

Hence, a single phase contains at most m^2 many v -improving active extension swaps. ◀

This result is potentially surprising, since for other allocation rules there are indeed extremely long sequences of such swaps. Indeed, with proportional rules there are exponentially long sequences of v -improving semi-positive swaps.

► **Proposition 21.** *There is a financial network with n banks, n edges, and proportional allocation rule, from which there is a sequence of $\Omega(2^n)$ many v -improving semi-positive swaps.*

Beyond Improvement for a Given Bank. In the remainder of the section, we concentrate on arbitrary active extension swaps (without the property that they are v -improving). Bounding the length of such sequences represents a very challenging problem. We make some initial progress on this problem: For a non-trivial class of instances, we construct a *single* short sequence of such swaps in each phase.

For the remainder of the section, we focus on instances with two network conditions. First, suppose that throughout every node has at most d incoming active edges, for some constant d . For example, $d = 1$ means that every network created in the sequence has at most one incoming active edge for each $v \in V$. This is fulfilled, in particular, when every node has an *indegree* of at most d .

► **Proposition 22.** *Consider an initial financial network with edge-ranking rules. Every sequence of semi-positive swaps, during which every node has at most $d = 1$ active incoming edges, has polynomial length.*

Proof. In an active extension swap, the non-profiting creditor v_2 has two incoming active edges. Thus, if $d = 1$, there can be no active extension swap. All swaps are saturating, semi-active or non-active swaps. Their number is bounded by $O(m^2)$. ◀

When $d \geq 2$, we consider a second condition – the network should have sufficient residuals.

► **Definition 23** (Sufficient Residuals). *In a financial network \mathcal{F} , a potential active swap is a pair of active edges $e_1 = (u_1, v_1)$ and $e_2 = (u_2, v_2)$ such that $l_{e_1} = l_{e_2}$, $p_{e_1} < p_{e_2}$, and there is a path of active edges from v_1 to v_2 . The network \mathcal{F} has sufficient residuals if for every potential active swap, $l_e - p_e > p_{e_2} - p_{e_1}$ for all $e \in P$, i.e., the potential active swap is an active extension swap.*

Intuitively, in a potential active swap all conditions for an extension swap are fulfilled, with the exception of the residual liabilities on the path P of active edges. If a network has *sufficient residuals*, then a potential active swap is indeed an active extension swap, since there is sufficient residual liability on P to forward the marginal increase of total assets for v_1 towards v_2 . For example, if the liabilities of active edges are sufficiently high compared to clearing payments, then the increases in payments will not “exhaust” the liabilities, and we are guaranteed to maintain the property of sufficient residuals throughout the swap sequence.

For networks with sufficient residuals and a constant number of active incoming edges, we can construct one sequence of swaps of polynomial length. The general case is a very interesting and challenging open problem.

► **Theorem 24.** *The SEMISWAP problem for financial networks with edge-ranking rules can be solved in polynomial time if, for all financial networks encountered during the sequence, the network has sufficient residuals and every node has at most a constant number of active incoming edges.*

► **Open Problem.** *Can the SEMISWAP problem for financial networks with edge-ranking rules be solved in polynomial time?*

4.2 Sequences of Arbitrary Swaps

In this section, we consider sequences of arbitrary swaps. In contrast to semi-positive ones, these swaps have no natural criterion of progress. We again focus on swaps in which a single bank v has to improve strictly. Unfortunately, sequences of such arbitrary v -improving swaps

can be very long, for every monotone allocation rule (including edge-ranking and proportional rules). Our results show that there are initial networks from which *every* sequence of v -improving swaps to a local optimum (in which no such swaps exist) is exponentially long.

Indeed, we obtain a more general statement. Consider a maximization problem for the total assets of a given bank v defined as follows. In an instance of the MAXASSETS_v problem, we are given a set of banks V . For each bank $v \in V$ we are given a non-negative value for the external assets as well as a number of incoming and outgoing *half-edges* or *stubs*. For each such stub i of v , we have a positive weight $l_i > 0$. Clearly, these stubs shall be *consistent* with at least one set of edges E among V , i.e., there must be a feasible underlying multi-graph $G = (V, E)$ in the sense that the stubs can be matched to form G . In particular, each incoming stub of weight l_i of each node v can be matched to some outgoing stub of weight l_i of some node $u \neq v$ to form an edge $e \in E$ with $\text{de}(e) = u$ and $\text{cr}(e) = v$. Consistency of stubs with *at least one* set E can be tested in polynomial time by solving a straightforward bipartite matching problem. For the remainder, we assume that it holds.

An instance of MAXASSETS_v has an edge-ranking rule \mathbf{f} that specifies the order in which banks pay their outgoing stubs. The rule is given implicitly by a *clearing oracle* \mathcal{O} . A clearing oracle for rule \mathbf{f} receives as input the remaining financial network $(G, \text{de}, \text{cr}, \mathbf{l}, \mathbf{a}^x)$ and outputs the clearing state. Note that a clearing oracle \mathcal{O} can be computed in polynomial time for edge-ranking rules [1].

Observe that any set of edges E that is consistent with the stubs gives rise to a financial network and a clearing state (where the latter can be computed using \mathcal{O}). The goal is to find such a consistent set to maximize the total assets of a given bank v in the resulting clearing state. The objective function in the instance are the total assets for a given bank v .

Finally, for edge swaps we define a neighborhood over the set of financial networks whose edges are consistent with the stubs in the input. For a given financial network \mathcal{F} , the *swap* neighborhood $\mathcal{N}(\mathcal{F})$ contains all networks \mathcal{F}' that can be obtained by performing an arbitrary debt swap in \mathcal{F} . A debt swap exchanges the matching of two pairs of stubs with pairwise disjoint incident nodes and with the same weights and thereby maintains consistency.

► **Proposition 25.** *MAXASSETS_v with swap neighborhood is in the complexity class PLS whenever the clearing oracle can be computed in polynomial time.*

Clearly, as noted above, we can obtain an initial network by computing a bipartite matching (indeed, *every* network consistent with the stubs represents a bipartite matching of the stubs and vice versa). The *swap* neighborhood of a network contains $O(m^2)$ other networks. For each of these networks, the corresponding clearing state can be determined using oracle \mathcal{O} . Hence, the *swap* neighborhood $\mathcal{N}(\mathcal{F})$ for a given network \mathcal{F} can be searched efficiently.

When locally maximizing MAXASSETS_v we do not necessarily restrict attention to a sequence of v -improving swaps from a given initial network to a local optimum. Instead, when given an initial network, we obtain an instance of MAXASSETS_v by breaking all edges into stubs. The goal is to find *any* local optimum network consistent with the stubs derived from the initial network. We show that this problem is PLS-complete.

► **Theorem 26.** *There is a tight PLS-reduction from MAX-2-SAT with flip neighborhood to MAXASSETS_v with swap neighborhood.*

This shows that MAXASSETS_v with *swap* neighborhood is PLS-complete. Moreover, the reduction implies that there exist initial states from which every execution of the standard local search algorithm requires exponentially many steps to reach a local optimum [18]. The standard local search algorithm for MAXASSETS_v yields a sequence of debt swaps.

► **Corollary 27.** *It is PLS-complete to compute a local optimum for MAXASSETS_v with swap neighborhood. There is an instance of MAXASSETS_v and an initial network \mathcal{F} such that every sequence of arbitrary v -improving swaps from \mathcal{F} to a local optimum has exponential length.*

In the reduction of Theorem 26, each possible network has the property that a bank has either 0 assets or enough assets to fully pay all outgoing stubs. The clearing oracle in these networks is trivial, and all payments are completely independent of the allocation rule.

► **Corollary 28.** *The results of Corollary 27 generalize to MAXASSETS_v with the swap neighborhood for any monotone allocation rule with efficient clearing oracle.*

References

- 1 Nils Bertschinger, Martin Hoefer, and Daniel Schmand. Flow allocation games. *Math. Oper. Res.*, 50(1):68–89, 2025. doi:10.1287/MOOR.2022.0355.
- 2 Péter Csóka and P Jean-Jacques Herings. Decentralized clearing in financial networks. *Manag. Sci.*, 64(10):4681–4699, 2018. doi:10.1287/MNSC.2017.2847.
- 3 Béni Egressy, Andreas Plesner, and Roger Wattenhofer. What is the price for lending in financial networks? In *Proc. 25th Int. Conf. Princ. Pract. Multi-Agent Syst. (PRIMA)*, pages 120–135, 2024. doi:10.1007/978-3-031-77367-9_11.
- 4 Larry Eisenberg and Thomas Noe. Systemic risk in financial systems. *Manag. Sci.*, 47(2):236–249, 2001. doi:10.1287/MNSC.47.2.236.9835.
- 5 Tom Friedetzky, David Kutner, George Mertzios, Iain Stewart, and Amitabh Trehan. Payment scheduling in the interval debt model. *Theor. Comput. Sci.*, 1028:115028, 2025. doi:10.1016/J.TCS.2024.115028.
- 6 Martin Hoefer, Carmine Ventre, and Lisa Wilhelmi. Algorithms for claims trades. In *Proc. 41st Symp. Theoret. Aspects Comput. Sci. (STACS)*, pages 42:1–42:17, 2024.
- 7 Martin Hoefer and Lisa Wilhelmi. Seniorities and minimal clearing in financial network games. In *Proc. 15th Symp. Algorithmic Game Theory (SAGT)*, pages 187–204, 2022. doi:10.1007/978-3-031-15714-1_11.
- 8 Stavros Ioannidis, Bart De Keijzer, and Carmine Ventre. Strong approximations and irrationality in financial networks with derivatives. In *Proc. 49th Int. Colloq. Autom. Lang. Programming (ICALP)*, pages 76:1–76:18, 2022. doi:10.4230/LIPICS.ICALP.2022.76.
- 9 Stavros Ioannidis, Bart De Keijzer, and Carmine Ventre. Financial networks with singleton liability priorities. *Theor. Comput. Sci.*, 963:113965, 2023. doi:10.1016/J.TCS.2023.113965.
- 10 Panagiotis Kanellopoulos, Maria Kyropoulou, and Hao Zhou. Forgiving debt in financial network games. In *Proc. 21st Conf. Auton. Agents and Multi-Agent Syst. (AAMAS)*, pages 1651–1653, 2022. doi:10.5555/3535850.3536065.
- 11 Panagiotis Kanellopoulos, Maria Kyropoulou, and Hao Zhou. Debt transfers in financial networks: Complexity and equilibria. In *Proc. 22th Conf. Auton. Agents and Multi-Agent Syst. (AAMAS)*, pages 260–268, 2023. doi:10.5555/3545946.3598645.
- 12 Panagiotis Kanellopoulos, Maria Kyropoulou, and Hao Zhou. On priority-proportional payments in financial networks. *Theor. Comput. Sci.*, 1014:114767, 2024. doi:10.1016/J.TCS.2024.114767.
- 13 Marios Papachristou and Jon Kleinberg. Allocating stimulus checks in times of crisis. In *Proc. 31th World Wide Web Conf. (WWW)*, pages 16–26, 2022. doi:10.1145/3485447.3512047.
- 14 Pál András Papp and Roger Wattenhofer. Network-aware strategies in financial systems. In *Proc. 47th Int. Colloq. Autom. Lang. Programming (ICALP)*, page 91, 2020.
- 15 Pál András Papp and Roger Wattenhofer. Debt swapping for risk mitigation in financial networks. In *Proc. 22nd Conf. Econ. Comput. (EC)*, pages 765–784, 2021. doi:10.1145/3465456.3467638.

- 16 Pál András Papp and Roger Wattenhofer. Default ambiguity: Finding the best solution to the clearing problem. In *Proc. 17th Conf. Web and Internet Econ. (WINE)*, pages 391–409, 2021. doi:10.1007/978-3-030-94676-0_22.
- 17 Pál András Papp and Roger Wattenhofer. Sequential defaulting in financial networks. In *Proc. 12th Symp. Innov. Theoret. Comput. Sci. (ITCS)*, pages 52:1–52:20, 2021. doi:10.4230/LIPICS.ITCS.2021.52.
- 18 Alejandro Schäffer. Simple local search problems that are hard to solve. *SIAM J. Comput.*, 20(1):56–87, 1991. doi:10.1137/0220004.
- 19 Steffen Schuldenzucker and Sven Seuken. Portfolio compression in financial networks: Incentives and systemic risk. In *Proc. 21st Conf. Econ. Comput. (EC)*, page 79, 2020. doi:10.1145/3391403.3399538.
- 20 Steffen Schuldenzucker, Sven Seuken, and Stefano Battiston. Finding clearing payments in financial networks with credit default swaps is PPAD-complete. In *Proc. 8th Symp. Innov. Theoret. Comput. Sci. (ITCS)*, pages 32:1–32:20, 2017. doi:10.4230/LIPICS.ITCS.2017.32.
- 21 Steffen Schuldenzucker, Sven Seuken, and Stefano Battiston. Default ambiguity: Credit default swaps create new systemic risks in financial networks. *Manag. Sci.*, 66(5):1981–1998, 2020. doi:10.1287/MNSC.2019.3304.
- 22 Luitgard Veraart. When does portfolio compression reduce systemic risk? *Mathematical Finance*, 32(3):727–778, 2019.