Settlement Fund Circulation Problem*

Hitoshi Hayakawa[1], Toshimasa Ishii[2], Hirotaka Ono[3], and Yushi Uno[4]

1 **Graduate School of Economics, Hokkaido University, Sapporo, Japan**
hit.hayakawa@econ.hokudai.ac.jp
2 **Graduate School of Economics, Hokkaido University, Sapporo, Japan**
ishii@econ.hokudai.ac.jp
3 **Graduate School of Informatics, Nagoya University, Nagoya, Japan**
ono@nagoya-u.jp
4 **Graduate School of Engineering, Osaka Prefecture University, Sakai, Japan**
uno@cs.osakafu-u.ac.jp

Abstract

In the economic activities, the central bank has an important role to cover payments of banks, when they are short of funds to clear their debts. For this purpose, the central bank timely puts funds so that the economic activities go smooth. Since payments in this mechanism are processed sequentially, the total amount of funds put by the central bank critically depends on the order of the payments. Then an interest goes to the amount to prepare if the order of the payments can be controlled by the central bank, or if it is determined under the worst case scenario. This motivates us to introduce a brand-new problem, which we call the settlement fund circulation problem. The problems are formulated as follows: Let $G = (V, A)$ be a directed multigraph with a vertex set V and an arc set A. Each arc $a \in A$ is endowed debt $d(a) \geq 0$, and the debts are settled sequentially under a sequence π of arcs. Each vertex $v \in V$ is put fund in the amount of $p_\pi(v) \geq 0$ under the sequence. The minimum/maximum settlement fund circulation problem (Min-SFC/Max-SFC) in a given graph G with debts $d : A \to \mathbb{R}_+ \cup \{0\}$ asks to find a bijection $\pi : A \to \{1, 2, \ldots, |A|\}$ that minimizes/maximizes the total funds $\sum_{v \in V} p_\pi(v)$. In this paper, we show that both Min-SFC and Max-SFC are NP-hard; in particular, Min-SFC is (I) strongly NP-hard even if G is (i) a multigraph with $|V| = 2$ or (ii) a simple graph with treewidth at most two, and is (II) (not necessarily strongly) NP-hard for simple trees of diameter four, while it is solvable in polynomial time for stars. Also, we identify several polynomial time solvable cases for both problems.

1998 ACM Subject Classification F.2 Analysis of Algorithms and Problem Complexity, G.2 Discrete Mathematics

Keywords and phrases Fund settlement, Algorithm, Digraph, Scheduling

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2017.46

1 Introduction

Background

In the economic activities, when a company borrows money, it owes a debt and the debt is not cleared until the debtor pays its amount. If the debtor fails to prepare cash for the

* This work is partly supported by KAKENHI 15H02965, 16K00001, 17H01698, 17K00017, 26241031, 26280001, 26540005 and by JST CREST Grant Number JPMJCR1402, Japan.

28th International Symposium on Algorithms and Computation (ISAAC 2017).
Editors: Yoshio Okamoto and Takeshi Tokuyama; Article No. 46; pp. 46:1–46:13
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

payment until the deadline, it will go bankrupt. Such bankruptcy should be avoided when it could cause significant damage to the economy, and it is particularly true for the case of banks since their debts are highly interconnected each other and bankruptcy of a bank may cause chain reaction of bankruptcy. It is one of the reasons that debts among banks are cleared in a special system, called *interbank settlement system*, in which the central bank supports cash management of the banks.

In the system, cash held by the central bank is used as the fund for the payments. When a bank does not have enough funds for clearing its debts, the central bank will lend the necessary amount. Suppose, for example, that there are three banks, say A, B, and C, and they form debts such that A owes 50 to B, and B owes 30 to C, and A and B currently have 10 each on its own. Now if A pays for its debt, then A is short of 40. Therefore, the central bank is requested to put 40 in order to fill the shortage. Once 40 is put on A, it can clear its debt 50 to B, and then B can also clear its debt 30 by using its own funds 10 and a part of the received funds 50. Note that we assume each debt has to be cleared independently and "sequentially", that is, it is not allowed to cancel out payments; A pays 30 directly to C, and the rest 20 to B, for example.[1]

Objective

Now, suppose that B pays before A does. Then, the central bank has to put 20 to B, and in addition, 40 to A. This illustrates, in general, that the total amount of funds put to clear all debts depends on the order of the payments. Since funds in an interbank settlement system is scarce resource in the public interest, the efficient usage is socially desirable. Accordingly, one of the important roles of the central bank is to minimize the total funds put to clear the debts. Then we can consider a problem that finds the minimum total funds put to clear all debts by deciding a sequence of the payments, which we formulate as MIN-SFC.

In a different perspective, another role of the central bank is to prepare for the worst case scenario such that it could hardly control the sequence of the payments. It is typical at the time of financial disruption and is crucially important. These observations again motivate us to define a corresponding maximization version of the problem, that is, to estimate the maximum funds that have to be put to clear all given debts, which we formulate as MAX-SFC. It is quite significant to obtain insights concerning the desirable sequence of the payments in order to argue relevant policies.

Technically, both problems are formulated as optimization problems on networks. However, the nature of our problems is essentially different from the classic flow problems in the sense that the amount of each "debt" (flow) cannot be split at the time of the payment. On the contrary, such unsplittable flows come to have a feature that once some debt is cleared, then the transferred funds are accumulated in the bank's "account" and they can be split arbitrarily for the subsequent payments.

History and Perspective in Economics

Historically, we can find a primitive concern of fund circulation in the renowned Quesnay's "Economic Table" [10]. Only recently, Rotemberg explicitly discusses the amount of required funds in the context of interbank settlements [12], though he does not give its general formulation. A general formulation to derive each of the minimum and maximum amount of

[1] Sequential clearing is standard in the modern interbank settlement systems, as World Bank documents that 116 of 139 surveyed countries have adopted sequential clearing based systems up to 2010 [14].

required funds is then given by Hayakawa [5] for the purpose of economic analysis.[2] This paper now gives, from the computational aspect, detailed mathematical formulations for these problems as Min-SFC and Max-SFC, and presents a series of algorithmic or complexity results based on solid observations for the first time.

In the wake of the recent world-wide financial crisis, analyzing "dominos" of default comes to have critical importance. Seminal studies in the literature effectively assume "simultaneous" clearing that makes payments cancel out whenever possible, not only bilaterally but also multilaterally, though "sequential" clearing, which we assume, is standard in the modern interbank settlement systems. The assumption of simultaneous clearing lets the relevant analyses be highly tractable [1, 2], however, it could considerably underestimate the amount of funds required to prevent "dominos" of default. In the light of these, we believe that the study in this paper serves as fundamental tools of the estimation and suggests a new methodology in the analyses that is applicable to complex economic situations in reality.

This paper is organized as follows. In Section 2, after giving several terminologies, we formalize our problem of interests and show some examples. Sections 3 and 4 discuss the minimization version of the problem, and show tractable and intractable cases, respectively. Section 5 deals with the maximization version. Finally in Section 6, future work is described.

2 Preliminaries

2.1 Definitions and Terminology

For a positive integer n, let $[n] = \{1, 2, \ldots, n\}$. For a finite set V, a family $\mathcal{X}$ of subsets in V is a *partition of* V if $\bigcup_{X \in \mathcal{X}} X = V$ holds and every two distinct sets in $\mathcal{X}$ are disjoint.

A directed graph (digraph) D is an ordered pair of its vertex set $V(D)$ and arc set $A(D)$ and is denoted by $D = (V(D), A(D))$, or simply $D = (V, A)$. An arc, an element of $A(D)$, is an ordered pair of vertices, and is denoted by $a = (u, v)$; this is distinct from (v, u). For an arc $a = (u, v)$, u is its *start vertex* and v is *end vertex*; they are denoted by $s(a)$ and $t(a)$, respectively. A digraph D is *multiple* when $A(D)$ is a multiple set; otherwise it is *simple*.

The *underlying graph* of a digraph D is an undirected graph G_D whose vertex set is $V(D)$ and edge set $E(G_D)$ has an edge $\{u, v\}$ as its element if and only if $(u, v) \in A(D)$ or $(v, u) \in A(D)$. A digraph D is *weakly connected* if its underlying graph G_D is connected. We assume throughout the paper that all digraphs are weakly connected. We usually use n and m to denote the number of vertices and arcs (edges), respectively, of a graph.

The *degree* of v is the number of arcs incident on v. We use $\Delta(D)$ to denote the maximum degree of a digraph D. Let $N_D(v)$ denote the set of vertices u with $(u, v) \in A(D)$ or $(v, u) \in A(D)$. Let $D[V']$ (resp., $D[A']$) denote the subgraph of D induced by a subset $V' \subseteq V(D)$ of vertices (resp., a subset $A' \subseteq A(D)$ of arcs). For a digraph D and a subset $A' \subseteq A(D)$ of arcs, we denote by $D \setminus A'$ the subgraph of D obtained from D by deleting A'.

2.2 Models and Problem Description

In the paper, we describe our problem by a digraph whose nodes are banks and arcs are loan relationship from one bank to another together with debts as arc weights.

Given a digraph $D = (V, A)$, *debt* of arcs is a function $d : A \to \mathbb{R}_+ \cup \{0\}$. For convenience, we sometimes introduce a (virtual) arc $a = (u, v)$ with $d(a) = 0$ if $(v, u) \in A$ and $(u, v) \notin A$.

[2] The relevant chapter of the paper [5] is reorganized as an independent article [6] with additional results.

A debt function d is *uniform* if $d(a) = c$ (constant) for all $a \in A$, otherwise *non-uniform*; it is *unit* if d is uniform and $c = 1$. Debts on a vertex v is *balanced* if $\sum_{(u,v)\in A} d(u,v) = \sum_{(v,w)\in A} d(v,w)$, and debts on a pair of vertices u and v is *symmetric* if $d(u,v) = d(v,u)$.

In our model, debts on arcs are settled individually in a single installment and sequentially. We say that a debt on an arc is *cleared* when it is settled (or, simply *clear* the arc), and we *put* funds on vertices to clear debts on their out-going arcs. When an arc is cleared, the amount for it is accumulated on the end vertex of the arc and can be reused for subsequent settlements. The amount of funds existing on a vertex is called its *residual*. A sequence of arcs, which corresponds to the order of selecting arcs to be cleared, can be represented as a permutation $\pi : A \to \{1, 2, \ldots, |A|\}$. We sometimes refer to this permutation as a *sequence* of A. We denote by $p_\pi(u,i)$ the fund put on u and by $r_\pi(u,i)$ the residual of u, immediately before putting fund $p_\pi(u,i)$ to clear arc $\pi^{-1}(i)$ for all $u \in V$ and for $i = 1, \ldots, |A|$. Then we can clear the debt on arc (u,v) if $r_\pi(u,\pi(u,v)) + p_\pi(u,\pi(u,v)) \geq d(u,v)$. We assume that we always put the minimum amount of funds to clear an arc, that is, $p_\pi(s(a),\pi(a)) = \max\{0, d(a) - r_\pi(s(a),\pi(a))\}$.

Now we define the *minimum settlement fund circulation problem* (Min-SFC) and the corresponding maximization problem (Max-SFC), which are introduced in [5] in the context of the interbank fund settlement systems, as follows.

> Min-SFC (Max-SFC)
> Instance: a digraph $D = (V, A)$ and debt $d : A \to \mathbb{R}_+ \cup \{0\}$.
> Question: minimize (maximize) $\sum_{v\in V} p_\pi(v)$ ($\triangleq p_\pi(V)$)
> subject to permutation $\pi : A \to \{1, \ldots, |A|\}$
> and $p_\pi(s(a),\pi(a)) + r_\pi(s(a),\pi(a)) \geq d(a)$ for all $a \in A$, where
> $p_\pi(u,0) = 0, r_\pi(u,0) = 0$ for all $u \in V$,
> $$r_\pi(u,i) = \begin{cases} r_\pi(u,i-1), & \pi^{-1}(i-1) \text{ is not incident on } u,\\ r_\pi(u,i-1) + d(\pi^{-1}(i-1)), & \pi^{-1}(i-1) \text{ is incident to } u,\\ \max\{0, r_\pi(u,i-1) - d(\pi^{-1}(i-1))\}, & \pi^{-1}(i-1) \text{ is incident from } u.\end{cases}$$

Here, we define $p_\pi(v) = \sum_{i=1}^{|A|} p(v,i)$, and for notational convenience, we often let $p_\pi(X) = \sum_{v\in X} p_\pi(v)$ for a subset X of $V(D)$ in a digraph D and a sequence π of $A(D)$.

We show examples of Min-SFC and Max-SFC in Figure 1, and see how debts are cleared in detail. In the sequence π_1, 20 funds are put on v_5 to clear the debt $d(v_5,v_6) = 20$ for the first arc $\pi_1^{-1}(1) = (v_5,v_6)$; $p_{\pi_1}(v_5,1) = 20$. The 20 funds are transferred to and accumulated in v_6; $r_{\pi_1}(v_6,2) = 20$. The second arc $\pi_1^{-1}(2) = (v_3,v_6)$ is cleared by putting 10 funds on v_3; $p_{\pi_1}(v_3,2) = 10$. The 10 funds are transferred to v_6, and it turns out that the residual on v_6 becomes $20 + 10 = 30$; $r_{\pi_1}(v_6,3) = 30$. Now, the residual are used for clearing the third arc $\pi_1^{-1}(3) = (v_6,v_1)$ and no additional fund needs to be put on v_6; $r_{\pi_1}(v_1,4) = 30$. We remark here again that a debt can only be cleared by a single installment. Also a residual can be split. Next, therefore, a part 20 of the residual 30 of v_1 is used for clearing $\pi_1^{-1}(4) = (v_1,v_2)$, where $d(v_1,v_2) = 20$, and so on.

2.3 Summary of the Results

The results of this paper are summarized in Table 1. To explain the table and for the use throughout the paper, we introduce some additional definitions. For a digraph D, if the underlying graph G_D of D belongs to some class $\mathcal{C}$ of graphs, then we may simply say that D belongs to $\mathcal{C}$ if no confusion occurs. A digraph is called *balanced* if debts on each vertex is

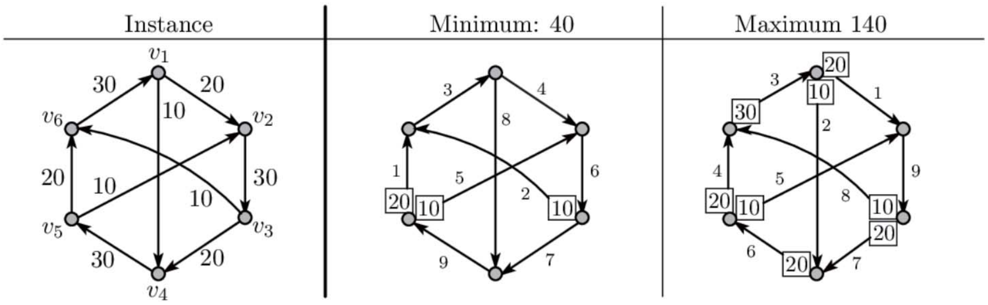

Figure 1 **[Left]** An instance of both MIN-SFC and MAX-SFC; a digraph $D = (V, A)$ with the debts $d(a)$ beside each arc a. **[Middle, Right]** Sequences π_1 and π_2 of A, respectively. The number beside each arc $a \in A$ indicates $\pi_i(a)$ and the number in the square attached to $s(a)$ indicates the amount of funds put on $s(a)$ for clearing the debt of a in π_i, i.e., $p_{\pi_i}(s(a), \pi_i(a))$ $(i = 1, 2)$. In fact, π_1 and π_2 are optimal solutions of MIN-SFC and MAX-SFC for D, with $p_{\pi_1}(V) = 40$ and $p_{\pi_2}(V) = 140$, respectively.

balanced. A digraph D is called *symmetric* if debts on each pair of vertices u and v with $\{u, v\} \in E(G_D)$ is symmetric. A digraph is called *uniform* if its debt function is uniform.

We emphasize here that all the results are new. Especially, we can see that those for general and simple graphs show sharp border with respect to the complexity in the sense that it is tractable for stars, but is intractable for trees.

3 Min-SFC: Intractable Cases

In the subsequent two sections (Sections 3 and 4), we discuss about MIN-SFC, which is our main interest in the context of analyzing settlements of debts. We first observe in this section that the problem is hard in general, but later in Section 4 we will see that it is tractable in some practical cases. Throughout these two sections, for an instance (D, d) of MIN-SFC, we denote by $opt(D, d)$ the minimum amount of funds put on $V(D)$ for clearing all arcs in D, i.e., $opt(D, d) = \min\{p_\pi(V(D)) \mid \pi \text{ is a sequence of } A(D)\}$.

Now let $D = (V, A)$ be a multiple digraph. We show that even if $|V| = 2$ or D is balanced, MIN-SFC with D is strongly NP-hard by a reduction from 3-PARTITION, which is known to be strongly NP-hard [4, p.224].

3-PARTITION
Instance: $(\{x_1, x_2, \ldots, x_{3m}\}, B)$: A set of $3m$ positive integers $x_1, x_2, \ldots, x_{3m}$ and an integer B such that $\sum_{i \in [3m]} x_i = mB$ and $B/4 < x_i < B/2$ for each $i \in [3m]$.
Question: Is there a partition $\{X_1, X_2, \ldots, X_m\}$ of $[3m]$ such that $\sum_{i \in X_j} x_i = B$ for each $j \in [m]$?

▶ **Theorem 1.** *For a multiple digraph D,* MIN-SFC *is strongly NP-hard even if $|V(D)| = 2$ or D is balanced.*

Proof. Take an instance $I_{\text{3PART}} = (\{x_1, x_2, \ldots, x_{3m}\}, B)$ of 3-PARTITION. From the I_{3PART}, we construct an instance $I_{\text{SFC}} = (D = (V, A), d)$ of MIN-SFC as follows. Let $V = \{u, v\}$ and A be the set of arcs consisting of $3m$ multiple arcs from u to v and m multiple arcs from v to u; denote an arc from u to v by a_i, $i \in [3m]$, and an arc from v to u by b_j, $j \in [m]$. Let $d(a_i) = x_i$ for $i \in [3m]$ and $d(b_j) = B$ for $j \in [m]$. Note that D is balanced.

Table 1 Summary of our results in this paper together with their corresponding theorem/lemma numbers; Linear and P stand for linear and polynomial time solvable, respectively, and T, C and L in brackets stand for Theorem, Corollary and Lemma, respectively.

arcs		graphs				
debt	multiplicity	dag	path	star	tree	larger classes
MIN-SFC						
uniform	multiple	Linear [T4]				
symmetric	simple	—	P [T7]			
balanced	multiple	—	strongly NP-hard for two vertices [T1]			
	simple	—	P	P	P [C8]	strongly NP-hard for bipartite or tw≤ 2 [T2]
general	simple	Linear [T6]	P [C12]	P [T10]	NP-hard [T3] FPT wrt. Δ [T11]	
	multiple	Linear [T6]				
MAX-SFC						
uniform		NP-hard [T17]				
general	multiple	Linear [T16]				

We claim that the instance $I_{3\text{PART}}$ is a yes-instance of 3-PARTITION if and only if there exists a sequence π of A with $p_\pi(V) \le B$. Notice that since I_{SFC} can be constructed from $I_{3\text{PART}}$ in polynomial time, this claim proves the theorem.

First, we show "only-if" part. Assume that $I_{3\text{PART}}$ is a yes-instance of 3-PARTITION; there exists a partition $\{X_1, X_2, \ldots, X_m\}$ of $[3m]$ such that $\sum_{i \in X_j} x_i = B$ for each $j \in [m]$. Without loss of generality, let $X_j = \{3j-2, 3j-1, 3j\}$ for $j \in [m]$ (note that $|X_j| = 3$ holds since $B/4 < x_i < B/2$ for each $i \in [3m]$). Then the sequence π of A defined as (b_1, a_1, a_2, a_3, b_2, a_4, a_5, a_6, $b_3, \ldots,$ $b_m, a_{3m-2}, a_{3m-1}, a_{3m}$) satisfies $p_\pi(V) = B$. Notice that $p_\pi(v, 1) = B$, $p_\pi(v, \ell) = 0$ for all $\ell \ge 2$, and $p_\pi(u) = 0$.

Next we show "if" part. Assume that there exists a sequence π of A with $p_\pi(V) \le B$. Since $d(b_j) = B$, we have $opt(D, d) \ge B$ and hence $p_\pi(V) = B$. Without loss of generality, assume that $\pi(b_1) < \pi(b_2) < \cdots < \pi(b_m)$. For $j \in [m-1]$, let X_j be the set of indices $i \in [3m]$ such that $\pi(b_j) < \pi(a_i) < \pi(b_{j+1})$. Since we need funds with amount B for clearing b_1 and $p_\pi(V) = B$ holds, no additional fund is put on V when any arc $a' \in A$ with $\pi(a') > \pi(b_1)$ is cleared. Hence, the total debts of arcs cleared between b_j and b_{j+1} is exactly B, i.e., $\sum_{i \in X_j} x_i = B$ for each $j \in [m-1]$. Furthermore, since $B/4 < x_i < B/2$ for $i \in [3m]$, we have $|X_j| = 3$ for each $j \in [m-1]$. Let $X_m = [3m] \setminus (\bigcup_{i=1}^{m-1} X_j)$. Note that $|X_m| = 3m - \sum_{j=1}^{m-1} |X_j| = 3$ and $\sum_{i \in X_m} x_i = mB - \sum_{j=1}^{m-1} \sum_{i \in X_j} x_i = B$. Thus, the partition $\{X_1, X_2, \ldots, X_m\}$ of $[3m]$ shows that $I_{3\text{PART}}$ is a yes-instance of 3-PARTITION. ◀

Let D_1 be the graph obtained from the graph $D = (V, A)$ of I_{SFC} in the proof of Theorem 1 by introducing new vertices w_i, $i \in [3m]$, and w'_j, $j \in [m]$, replacing each arc a_i with two arcs (u, w_i) and (w_i, v) with $d(u, w_i) = d(w_i, v) = x_i$, and replacing each arc b_j with two arcs (v, w'_j) and (w'_j, u) with $d(v, w'_j) = d(w'_j, u) = B$. Notice that D_1 is a simple and balanced graph, and the underlying graph G_{D_1} of D_1 is bipartite and series-parallel. Also we can prove that $I_{3\text{PART}}$ is a yes-instance of 3-PARTITION if and only if there exists a sequence π of $A(D_1)$ with $p_\pi(V(D_1)) \le B$, in a similar way to the proof of Theorem 1.

▶ **Theorem 2.** *For a simple digraph D,* Min-Sfc *is strongly NP-hard even if D is balanced, G_D is bipartite, or series-parallel (i.e., the treewidth of G_D is at most two).*

Furthermore, we can show that the problem Min-SFC is NP-hard even in the case of trees, while it is open whether it is strongly NP-hard. The proof is given later in Section 4.3.

▶ **Theorem 3.** *For a simple digraph D,* Min-Sfc *is NP-hard even if G_D is a tree of diameter at most four.*

4 Min-SFC: Tractable Cases

In this section, we show that in some practical cases the problem Min-SFC becomes tractable. We assume in this section that D is a simple digraph, unless otherwise mentioned.

4.1 Uniform Digraphs, Acyclic Digraphs, and Symmetric Graphs

In the case of uniform debt, Min-SFC is equivalent to the problem which asks to partition a given graph into a minimum number of directed paths, which is known to be linearly solvable (e.g., see [3, Lemma 2]).

▶ **Theorem 4.** *If each debt is uniform,* Min-SFC *can be solved in linear time.*

Let $D = (V, A)$ be a digraph and $comp(D)$ be the number of components in D. Let A_δ denote the set of arcs a in A with $d(a) \le \delta$. We denote $\{d(a) \mid a \in A\}$ by $\{\delta_1, \delta_2, \dots, \delta_q\}$ with $\delta_1 < \delta_2 < \cdots < \delta_q$. Then, we have the following lemma about lower bounds on $opt(D, d)$.

▶ **Lemma 5.**

(i) *For a digraph $D = (V, A)$, $opt(D, d) \ge \sum_{v \in V} \max\{0, \sum_{a \in A^+_D(v)} d(a) - \sum_{a \in A^-_D(v)} d(a)\}$, where $A^+_D(v)$ (resp., $A^-_D(v)$) denotes the set of all arcs incident from v (resp., to v) in D.*

(ii) *For a digraph $D = (V, A)$, we have $opt(D, d) \ge \sum_{i=1}^{q} comp(D[A \setminus A_{\delta_{i-1}}])(\delta_i - \delta_{i-1})$, where we let $\delta_0 = 0$.*

Assume that D is an acyclic digraph and let $\tau : V \to [n]$ be a topological ordering of V. It is not difficult to see that a sequence π of A such that $\pi(a_1) < \pi(a_2)$ if and only if $\tau(s(a_1)) \le \tau(s(a_2))$ for each $a_1, a_2 \in A$ satisfies $p_\pi(V) = \sum_{v \in V} \max\{0, \sum_{a \in A^+_D(v)} d(a) - \sum_{a \in A^-_D(v)} d(a)\}$; Min-SFC is linearly solvable by Lemma 5(i).

▶ **Theorem 6.** *For an acyclic digraph D,* Min-SFC *can be solved in linear time.*

Assume that D is a symmetric digraph. Then, we can show that a sequence π of A with $p_\pi(V) = \sum_{i=1}^{q} comp(D[A \setminus A_{\delta_{i-1}}])(\delta_i - \delta_{i-1})$ which composes an Eulerian cycle of D can be found in $O(m^2)$ time.

▶ **Theorem 7.** *For a symmetric digraph D,* Min-SFC *can be solved in $O(m^2)$ time.*

For a tree D, if debts on each vertex is balanced, then debts on each pair of two vertices u and v with $\{u, v\} \in E(G_D)$ become symmetric. Therefore, as a corollary of Theorem 7, we can show that Min-SFC with a tree D is polynomially solvable if D is balanced.

▶ **Corollary 8.** *For a balanced tree,* Min-Sfc *can be solved in $O(n^2)$ time.*

4.2 Stars with General Debts

We next consider the case where the underlying graph of $D = (V, A)$ is a star with arbitrary debts. We remark that the interbank network system in Japan was a kind of star structures before 1997 [7]. Throughout this subsection, we assume that for each pair of vertices v and v' in V with $\{v, v'\} \in E(G_D)$, both of (v, v') and (v', v) belong to A; otherwise (say, $(v, v') \notin A$), then we add an arc (v, v') with debt 0 to D and redenote the resulting graph by D (note that the existence of arcs with debt 0 does not affect to $opt(G, d)$).

Now let $D = (V, A)$ be a star with center u. Then, $E(G_D) = \{\{u, v\} \mid v \in V \setminus \{u\}\}$ holds. Let $V^+ = \{v \in V \setminus \{u\} \mid d(v, u) \geq d(u, v)\}$ and $V^- = \{v \in V \setminus \{u\} \mid d(v, u) < d(u, v)\}$. We have the following theorem about an optimal solution.

▶ **Theorem 9.** *Let $D = (V, A)$ be a star with center u. There exists an optimal sequence π of A for* MIN-SFC *satisfying the following* (i)–(iv)*:*
- **(i)** $\pi(u, v) = \pi(v, u) - 1$ *for all* $v \in V \setminus \{u\}$.
- **(ii)** $\pi(u, v) < \pi(u, v')$ *for all* $v \in V^+$ *and* $v' \in V^-$.
- **(iii)** $\pi(u, v) < \pi(u, v')$ *if and only if* $d(u, v) \leq d(u, v')$ *for all* $v, v' \in V^+$.
- **(iv)** $\pi(u, v) < \pi(u, v')$ *if and only if* $d(v, u) \geq d(v', u)$ *for all* $v, v' \in V^-$.

This theorem shows that we can obtain an optimal solution of MIN-SFC by the following algorithm MINSTAR(D, d).

Algorithm MINSTAR(D, d)
Input: A star $D = (V, A)$ with center u and a debt function d.
Output: A sequence π of A such that $p_\pi(V)$ is minimized.
Step 1: Order vertices of V^+ such that $d(u, v_1) \leq d(u, v_2) \leq \cdots \leq d(u, v_{|V^+|})$ and let $\pi(u, v_i) = 2i - 1$ and $\pi(v_i, u) = 2i$ for $i = 1, 2, \ldots, |V^+|$.
Step 2: Order vertices of V^- such that $d(v_{|V^+|+1}, u) \geq d(v_{|V^+|+2}, u) \geq \cdots \geq d(v_{|V^+|+|V^-|}, u)$ and let $\pi(u, v_i) = 2i - 1$ and $\pi(v_i, u) = 2i$ for $i = |V^+| + 1, |V^+| + 2, \ldots, |V^+| + |V^-|$.

It is fairly straightforward to see that the time complexity of this algorithm is $O(n \log n)$, since it is dominated by that of sorting $O(n)$ arcs.

▶ **Theorem 10.** *For a star,* MIN-SFC *can be solved in $O(n \log n)$ time.*

4.3 Trees with General Debts

We consider the case where the underlying graph of D is a tree. As shown in Theorem 3 and Corollary 8, the problem is NP-hard even for trees, while it is polynomially solvable if a given tree is balanced. In this subsection, we will show that MIN-SFC is fixed-parameter tractable with respect to the maximum degree Δ, and give a hardness proof of Theorem 3.

Throughout this subsection, we assume that for each pair of vertices v and v' in V with $\{v, v'\} \in E(G_D)$, both of (v, v') and (v', v) belong to A.

A Fixed-parameter Algorithm

We first show the following theorem.

▶ **Theorem 11.** *For a tree D,* MIN-SFC *can be solved in $O(2^{\Delta(D)} n \log n)$ time.*

As a corollary of this theorem, we can see that MIN-SFC with paths is polynomially solvable.

▶ **Corollary 12.** *For a path,* MIN-SFC *can be solved in $O(n \log n)$ time.*

Before proving Theorem 11, we prepare some auxiliary lemmas. For a digraph D, a vertex is called a *leaf* if its degree is one in G_D. For a leaf v, *splitting* v is to introduce a new vertex v' and to replace the arc $(u, v) \in A(D)$ incident to v with an arc (u, v') with debt $d(u, v)$. We denote the resulting digraph and its debt function by $D_{v,v'}$ and $d_{v,v'}$, respectively.

▶ **Lemma 13.** *Assume that a digraph $D = (V, A)$ has a leaf v; denote the two arcs incident on v by (u, v) and (v, u). Let π be a sequence of A with $\pi(u, v) > \pi(v, u)$ and π' be the sequence of $A(D_{v,v'})$ such that $\pi'(u, v') = \pi(u, v)$ and $\pi'(a) = \pi(a)$ for all other arcs $a \in A \setminus \{(u, v)\}$. Then, π' is an optimal sequence of* Min-SFC *for $D_{v,v'}$ if and only if π is an optimal sequence for D under the assumption that (v, u) is cleared before (u, v).*

For a vertex u in D, let D_0 be the star induced by $\{u\} \cup N_D(u)$, and $D_1, D_2, \ldots, D_q$ be subtrees in the graph obtained from D by deleting u, where $q = |N_D(u)|$. We denote two arcs connecting u and D_i by (u, v_i) and (v_i, u), where $v_i \in V(D_i)$. The following lemma shows that if we know in advance whether (u, v_i) is cleared after (v_i, u) or not for each $v_i \in N_D(u)$, then the minimum amount $opt(D, d)$ of funds for clearing $A(D)$ follows from optimal solutions for the star D_0, and either trees $D_i + u$ or $(D_i + u)_{u,u'}$ obtained from $D_i + u$ by splitting u, where for a subgraph D' of D and a vertex $u \in V \setminus V(D')$, we denote $(V(D') \cup \{u\}, A(D') \cup \bigcup_{v \in N_D(u) \cap V(D')} \{(u, v), (v, u)\})$ by $D' + u$.

▶ **Lemma 14.** *For a vertex u in a digraph $D = (V, A)$, let v_i, D_0, D_i, and $D_i + u$, $i = 1, 2, \ldots, q$ be defined as above. Let N_1 and N_2 be a partition of $N_D(u)$ (N_1 or N_2 may be empty). Let $opt(D, d, u, N_1, N_2)$ denote the minimum amount of funds put on V for clearing all arcs in A under the assumption that (v, u) is cleared before (u, v) for each $v \in N_1$ and (u, v) is cleared before (v, u) for each $v \in N_2$. Then,*

$$\begin{aligned} opt(D, d, u, N_1, N_2) \;=\; & opt((D_0)_{N_1,N_1'}, d_{N_1}) - \textstyle\sum_{v \in N_1} d(v, u) \\ & - \textstyle\sum_{v \in N_2} \max\{0, d(v, u) - d(u, v)\} \\ & + \textstyle\sum_{v_i \in N_1} (opt(D_i + u, d) - \max\{0, d(u, v_i) - d(v_i, u)\}) \\ & + \textstyle\sum_{v_i \in N_2} (opt((D_i + u)_{u,u'}, d_{u,u'}) - d(u, v_i)), \end{aligned}$$

where $(D_0)_{N_1,N_1'}$ denotes the star obtained from the star D_0 by splitting all vertices in N_1, N_1' denotes the set of vertices generated by these splitting operations, and d_{N_1} denotes the resulting debt function on $A((D_0)_{N_1,N_1'})$.

Proof. Let

$$\begin{aligned} f(D, d, u, N_1, N_2) \;=\; & opt((D_0)_{N_1,N_1'}, d_{N_1}) - \textstyle\sum_{v \in N_1} d(v, u) \\ & - \textstyle\sum_{v \in N_2} \max\{0, d(v, u) - d(u, v)\} \\ & + \textstyle\sum_{v_i \in N_1} (opt(D_i + u, d) - \max\{0, d(u, v_i) - d(v_i, u)\}) \\ & + \textstyle\sum_{v_i \in N_2} (opt((D_i + u)_{u,u'}, d_{u,u'}) - d(u, v_i)). \end{aligned}$$

Let π be an arbitrary sequence of A such that $\pi(v, u) < \pi(u, v)$ for each $v \in N_1$ and $\pi(u, v) < \pi(v, u)$ for each $v \in N_2$. First we show that $p_\pi(V) \geq f(D, d, u, N_1, N_2)$, from which $opt(D, d, u, N_1, N_2) \geq f(D, d, u, N_1, N_2)$. We will consider lower bounds L_1, L_2, and L_3 on $p_\pi(u)$, $\sum_{v_i \in N_1} p_\pi(V(D_i))$, and $\sum_{v_i \in N_2} p_\pi(V(D_i))$, respectively; $p_\pi(V) \geq L_1 + L_2 + L_3$.

Consider a lower bound on $p_\pi(u)$. Since how much funds need to be put on u depends only on debts of arcs incident from/to u, we consider the minimum amount p^* of funds for clearing all arcs in the star D_0 with center u. By the assumption that (v_i, u) is cleared before (u, v_i) for each $v_i \in N_1$ and (u, v_i) is cleared before (v_i, u) for each $v_i \in N_2$ and Lemma 13 for leaves $v_i \in N_1$ of D_0, we can see that $p^* = opt((D_0)_{N_1,N_1'}, d_{N_1})$. Now we can observe that any sequence π' of $A((D_0)_{N_1,N_1'})$ satisfies $p_{\pi'}(N_{(D_0)_{N_1,N_1'}}(u)) = \sum_{v \in N_1} d(v, u) +$

$\sum_{v \in N_2} \max\{0, d(v,u) - d(u,v)\}$. Hence, the amount $p_\pi(u)$ of funds put on u is at least $opt((D_0)_{N_1,N_1'}, d_{N_1}) - (\sum_{v \in N_1} d(v,u) + \sum_{v \in N_2} \max\{0, d(v,u) - d(u,v)\})$.

Consider a lower bound on $p_\pi(V(D_i))$ for $v_i \in N_1$. Note that how much funds need to be put on $V(D_i)$ depends on debts of $A(D_i) \cup \{(u,v_i),(v_i,u)\}$. By taking into account the assumption that (v_i,u) is cleared before (u,v_i), we can observe that $p_\pi(V(D_i))$ is at least the minimum amount of funds put on $V(D_i)$ among any funds for clearing $A(D_i+u)$ in D_i+u. Here notice that the amount of funds put on u in D_i+u is always $\max\{0, d(u,v_i) - d(v_i,u)\}$. It follows that $p_\pi(V(D_i)) \geq opt(D_i+u,d) - \max\{0, d(u,v_i) - d(v_i,u)\}$. Similarly, we can observe that for each $v_i \in N_2$, $p_\pi(V(D_i)) \geq opt((D_i+u)_{u,u'}, d_{u,u'}) - d(u,v_i)$ holds.

Thus, we can see that $p_\pi(V) \geq f(D,d,u,N_1,N_2)$. Finally, we show that some sequence π^* of A satisfies $p_{\pi^*}(V) = f(D,d,u,N_1,N_2)$; π^* is optimal and proves this lemma. Let π_0' be a sequence of $A((D_0)_{N_1,N_1'})$ obtained by applying Algorithm MinStar$((D_0)_{N_1,N_1'}, d_{N_1})$, and π_0 be the sequence of $A(D_0)$ obtained from π_0' by letting $\pi_0(u,v) = \pi_0'(u,v')$ for all $v \in N_1$ and $\pi_0(a) = \pi_0'(a)$ for all other arcs a incident on u. For a tree D_i with $v_i \in N_1$, let π_i be a sequence of $A(D_i+u)$ with $p_{\pi_i}(V(D_i+u)) = opt(D_i+u,d)$ with $\pi_i(v_i,u) < \pi_i(u,v_i)$. For a tree D_i with $v_i \in N_2$, let π_i be a sequence of $A((D_i+u)_{u,u'})$ with $p_{\pi_i}(V((D_i+u)_{u,u'})) = opt((D_i+u)_{u,u'}, d_{u,u'})$ with $\pi_i(v_i,u') > \pi_i(u,v_i)$. Note that such a π_i exists for each $v_i \in N_1 \cup N_2$. We can construct a sequence π^* of A with $p_{\pi^*}(V) = f(D,d,u,N_1,N_2)$ by combining π_0 and π_i, $i \in N_1 \cup N_2$. ◀

Let $D = (V,A)$ be a tree. Based on Lemma 14, we will give a dynamic programming algorithm for finding an optimal sequence of A in $O(2^\Delta n)$ time, which proves Theorem 11.

Here, for a vertex $r \in V$ chosen arbitrarily, we regard D as a rooted tree with root r. For a vertex u in D, let $pa(u)$ be the parent of u if it exists, $Ch(u)$ be the children of u, and $D(u)$ be the subtree of D rooted at u. For a partition $\{N_1,N_2\}$ of $Ch(u)$, we define $opt_1(u,N_1,N_2)$ (resp., $opt_2(u,N_1,N_2)$) as the minimum amount of funds clearing $A(D(u)+pa(u))$ under the assumption that (v,u) is cleared before (u,v) for each $v \in N_1$ (resp., $v \in N_1 \cup \{pa(u)\}$) and (u,v) is cleared before (v,u) for each $v \in N_2 \cup \{pa(u)\}$ (resp., $v \in N_2$). Note that $opt_1(r,N_1,N_2) = opt_2(r,N_1,N_2)$. Let $opt_i^*(u) = \min\{opt_i(u,N_1,N_2) \mid N_1 \subseteq Ch(u)\}$ for $i = 1,2$. We here remark that $opt_1^*(u)$ (resp., $opt_2^*(u)$) is the minimum amount of funds for clearing $A(D(u)+pa(u))$ under the assumption that $(u,pa(u))$ (resp., $(pa(u),u)$) is cleared before $(pa(u),u)$ (resp., $(u,pa(u))$).

Our dynamic programming algorithm proceeds in a bottom-up manner in D, while computing these two values $opt_1^*(u)$ and $opt_2^*(u)$ for each vertex u in D. note that $opt_1^*(r) = opt_2^*(r) = opt(D,d)$. Lemma 14 indicates that $opt_1(u,N_1,N_2)$ and $opt_2(u,N_1,N_2)$ can be computed by using $opt_1^*(v)$ and $opt_2^*(v)$ for $v \in Ch(u)$. Namely, we have

$$\begin{aligned} opt_1(u,N_1,N_2) &= opt((D_0)_{N_1,N_1'}, d_{N_1}) - \textstyle\sum_{v \in N_1} d(v,u) \\ &\quad - \textstyle\sum_{v \in N_2} \max\{0, d(v,u) - d(u,v)\} \\ &\quad + \textstyle\sum_{v \in N_1} (opt_1^*(v) - \max\{0, d(u,v) - d(v,u)\}) \\ &\quad + \textstyle\sum_{v \in N_2} (opt_2^*(v) - d(u,v)), \end{aligned}$$

and

$$\begin{aligned} opt_2(u,N_1,N_2) &= opt((D_0)_{N_1 \cup \{pa(u)\}, N_1' \cup \{pa'\}}, d_{N_1 \cup \{pa(u)\}}) - \textstyle\sum_{v \in N_1} d(v,u) \\ &\quad - \textstyle\sum_{v \in N_2} \max\{0, d(v,u) - d(u,v)\} \\ &\quad + \textstyle\sum_{v \in N_1} (opt_1^*(v) - \max\{0, d(u,v) - d(v,u)\}) \\ &\quad + \textstyle\sum_{v \in N_2} (opt_2^*(v) - d(u,v)), \end{aligned}$$

where $D_0 = D[\{u, pa(u)\} \cup Ch(u)]$, pa' denotes the vertex generated by splitting $pa(u)$ in $(D_0)_{N_1,N_1'}$, and $d_{N_1 \cup \{pa(u)\}}$ denotes the debt function on $A((D_0)_{N_1 \cup \{pa(u)\}, N_1' \cup \{pa'\}})$.

Here we note that in these two equations, $opt_1^*(v) = opt(D(v) + u, d)$ and $opt_2^*(v) = opt((D(v) + u)_{u,u'}, d_{u,u'})$, by the assumption on N_1 and N_2. For stars $(D_0)_{N_1,N_1'}$ and $(D_0)_{N_1\cup\{pa(u)\},N_1'\cup\{pa'\}}$, we can compute $opt((D_0)_{N_1,N_1'}, d_{N_1})$ and $opt((D_0)_{N_1\cup\{pa(u)\},N_1'\cup\{pa'\}}, d_{N_1\cup\{pa(u)\}})$ in $O(|N_D(u)| \log |N_D(u)|)$ time by Theorem 10. Hence, if we know $opt_1^*(v)$ and $opt_2^*(v)$ for all $v \in Ch(u)$, then we can compute $opt_1^*(u)$ and $opt_2^*(u)$ in $O(2^{|N_D(u)|}|N_D(u)| \log |N_D(u)|)$ time by computing $opt_1(u, N_1, N_2)$ and $opt_2(u, N_1, N_2)$ for all possible N_1 and N_2. Thus, we can compute $opt(D, d) = opt_1^*(r) = opt_2^*(r)$ in $O(2^{\Delta} n \log n)$ time.

NP-hardness

Next, we give a proof of Theorem 3; we show the NP-hardness of Min-SFC with a tree. We will reduce from Partition, which is known to be NP-hard [8].

> Partition
> Instance: $\{x_1, x_2, \ldots, x_n\}$: A set of n positive integers $x_1, x_2, \ldots, x_n$.
> Question: Is there a partition $\{X_1, X_2\}$ of $[n]$ such that $\sum_{i\in X_1} x_i = \sum_{i\in X_2} x_i$?

Take an instance $I_{\text{PART}} = \{x_1, x_2, \ldots, x_n\}$ of Partition. From the I_{PART}, we construct an instance $I_{\text{SFC}} = (D = (V, A), d)$ of Min-SFC as follows. Let $V = \{r, u\} \cup \bigcup_{i=1}^n \{v_i, w_i\}$ and $E(G_D) = \{\{r, u\}\} \cup \bigcup_{i=1}^n \{\{u, v_i\}, \{v_i, w_i\}\}$. Let $x^* = \sum_{i\in[n]} x_i$, $d(r, u) = d(u, r) = x^*/2$, $d(u, v_i) = d(v_i, u) = d(v_i, w_i) = x_i$, and $d(w_i, v_i) = x_i/2$ for $i \in [n]$.

We here claim that there exists a partition $\{X_1, X_2\}$ of $[n]$ such that $\sum_{i\in X_1} x_i = \sum_{i\in X_2} x_i$ if and only if there exists a sequence π of A with $p_\pi(V) \le 3x^*/4$. Notice that since I_{SFC} can be constructed from I_{PART} in polynomial time, this claim proves Theorem 3.

▶ **Claim 15.** *There exists a partition $\{X_1, X_2\}$ of $[n]$ such that $\sum_{i\in X_1} x_i = \sum_{i\in X_2} x_i$ if and only if there exists a sequence π of A with $p_\pi(V) \le 3x^*/4$.*

5 Max-SFC

5.1 Tractable Case

Assume that D is an acyclic digraph and let $\tau : V \to [n]$ be a topological ordering of V. It is not difficult to see that a sequence π of A such that $\pi(a_1) > \pi(a_2)$ if and only if $\tau(t(a_1)) \le \tau(t(a_2))$ for each $a_1, a_2 \in A$ satisfies $p_\pi(V) = \sum_{a\in A} d(a)$. Thus, Max-SFC is linearly solvable, since the summation of all debts is an upper bound on the optimal value.

▶ **Theorem 16.** *For an acyclic digraph D,* Max-SFC *can be solved in linear time.*

5.2 NP-hardness

Below, we show that Max-SFC is NP-hard, even in the case where each debt is unit or a given graph is bipartite.

▶ **Theorem 17.** *For a digraph D,* Max-SFC *is NP-hard even if each debt of an arc in $A(D)$ is unit or D is bipartite.*

We prove this theorem by reducing from Vertex Cover, which is known to be NP-hard [8]. For an undirected graph $G = (V, E)$, a set $V' \subseteq V$ of vertices is called a *vertex cover* if every edge $e = \{u, v\} \in E$ satisfies $\{u, v\} \cap V' \neq \emptyset$.

VERTEX COVER
Instance: An undirected graph $G = (V, E)$ and an integer k, that is, $(G = (V, E), k)$.
Question: Is there a vertex cover X with $|X| \leq k$ in G?

Take an instance $I_{\mathrm{VC}} = (G = (V, E), k)$ of VERTEX COVER. From the I_{VC}, we construct an instance $I_{\mathrm{SFC}} = (D = (V', A), d)$ of MAX-SFC as follows. For each vertex $v_i \in V$, we introduce two copies v_i^1 and v_i^2 of v_i and an arc (v_i^1, v_i^2), and let $V' = \bigcup_{v_i \in V}\{v_i^1, v_i^2\}$ and $A_1 = \bigcup_{v_i \in V}(v_i^1, v_i^2)$. For each edge $\{v_i, v_j\} \in E$, we introduce two arcs (v_i^2, v_j^1) and (v_j^2, v_i^1), and let $A_2 = \bigcup_{\{v_i,v_j\}\in E}\{(v_i^2, v_j^1), (v_j^2, v_i^1)\}$. Let $A = A_1 \cup A_2$ and $d(u, v) = 1$ for all $(u, v) \in A$. Note that D is bipartite. The following lemma completes the proof of Theorem 17.

▶ **Lemma 18.** *G has a vertex cover with cardinality at most k if and only if there exists a sequence π of A such that $p_\pi(V') \geq |A| - k$ in D.*

6 Future work

One of the most important future work is to deal with more appropriate graphs classes that reflects well the debts relationship among banks in our real economic activities. As we mentioned in Section 4.2, it is known that the interbank network system in Japan was a kind of star structures before 1997 [7]. On the contrary, Imakubo and Soejima [7] also showed that in the year of 2005 it had changed and turned to be a core-periphery structure, which is a certain kind of classic hub-authority biclique model [9] and thus one of the so-called complex networks. In the model, banks are classified into either one of the two categories, *core banks* or *periphery* banks, such that payments among the core banks are more densely connected among them compared to those among the periphery banks. Recent research observed similar facts in some other countries, e.g., in the US in 2004 [13], in the Netherlands in 2006 [11], and so on. In view of these recent observations, it would extremely be important to consider our problem on this realistic model and develop efficient algorithms for it.

References

1. Franklin Allen and Douglas Gale. Financial contagion. *Journal of Political Economy*, 108(1):1–33, february 2000.
2. Larry Eisenberg and Thomas H. Noe. Systemic risk in financial systems. *Management Science*, 47(2):236–249, February 2001.
3. John Gallant, David Maier, and James Astorer. On finding minimal length superstrings. *Journal of Computer and System Sciences*, 20(1):50 – 58, 1980.
4. Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman & Co., New York, NY, USA, 1979.
5. Hitoshi Hayakawa. Theoretical analyses on settlement system: Chapter 1. complexity of payment network (doctoral dissertation). *Retrieved from UTokyo Repository (No. A30011)*, January 2014.
6. Hitoshi Hayakawa. Characterization of lower bound and upper bound of required settlement fund under real-time gross settlement. *Available at SSRN: http://ssrn.com/abstract=2659975 [Accessed June 1, 2017]*, February 2016.
7. Kei Imakubo and Yutka Soejima. The transaction network in japan's interbank money markets. *Monetary and Economic Studies*, 28:107–150, November 2010.
8. Richard M. Karp. Reducibility among combinatorial problems. In R. Miller and J. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.

9 Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

10 François Quesnay. *Tableau économique.* Versailles, 1758.

11 Kirsten Bonde Rordam and Morten L. Bech. The topology of danish interbank money flows. *Finance Research Unit No. 2009/01*, 2009.

12 Julio J. Rotemberg. Minimal settlement assets in economies with interconnected financial obligations. *Journal of Money, Credit, and Banking*, 43(1):81–108, February 2011.

13 Kimmo Soramäki, Morten L. Bech, Jeffrey Arnold, Robert J. Glass, and Walter E. Beyeler. The topology of interbank payment flows. *Physica A: Statistical Mechanics and its Applications*, 379(1,i1):317–333, June 2007.

14 World Bank. *Global Financial Development Report 2013 : Rethinking the Role of the State in Finance.* World Bank, Washington, D.C., 2013.