Strong Approximations and Irrationality in Financial Networks with Derivatives

Stavros D. Ioannidis
Department of Informatics, King's College London, UK

Bart de Keijzer
Department of Informatics, King's College London, UK

Carmine Ventre
Department of Informatics, King's College London, UK

Abstract

Financial networks model a set of financial institutions (firms) interconnected by obligations. Recent work has introduced to this model a class of obligations called credit default swaps, a certain kind of financial derivatives. The main computational challenge for such systems is known as the clearing problem, which is to determine which firms are in default and to compute their exposure to systemic risk, technically known as their recovery rates. It is known that the recovery rates form the set of fixed points of a simple function, and that these fixed points can be irrational. Furthermore, Schuldenzucker et al. (2016) have shown that finding a weakly (or "almost") approximate (rational) fixed point is PPAD-complete.

We further study the clearing problem from the point of view of irrationality and approximation strength. Firstly, we observe that weakly approximate solutions may misrepresent the actual financial state of an institution. On this basis, we study the complexity of finding a strongly (or "near") approximate solution, and show FIXP-completeness. We then study the structural properties required for irrationality, and we give necessary conditions for irrational solutions to emerge: The presence of certain types of cycles in a financial network forces the recovery rates to take the form of roots of non-linear polynomials. In the absence of a large subclass of such cycles, we study the complexity of finding an exact fixed point, which we show to be a problem close to, albeit outside of, PPAD.

2012 ACM Subject Classification Theory of computation → Problems, reductions and completeness

Keywords and phrases FIXP, Financial Networks, Systemic Risk

Digital Object Identifier 10.4230/LIPIcs.ICALP.2022.76

Category Track A: Algorithms, Complexity and Games

Related Version *Full Version*: https://arxiv.org/abs/2109.06608 [18]

Funding *Carmine Ventre*: Partially supported by the UKRI Trustworthy Autonomous Systems Hub (EP/V00784X/1).

1 Introduction

The International Monetary Fund says that the global financial crisis (GFC) of 2007 has had long lasting consequences, including loss of growth, large public debt and even a decline of fertility rates, see [2]. Consequently, the need to assess the *systemic risk* of the financial network cannot be overstated. For example, if banks at risk of defaults could be easily identified in the complex network of financial obligations, then spread could be preemptively avoided with appropriate countermeasures such as bailouts from central banks or regulators.

In this context, the *clearing problem* introduced in [7] plays a central role. We are given a so-called financial network, that is, a graph where vertices are banks (or, more generally, financial institutions) and weighted arcs (u, v) model direct liabilities from bank u to bank v. Each bank has also some assets external to the network, that can be used to pay its liabilities.

49th International Colloquium on Automata, Languages, and Programming (ICALP 2022).
Editors: Mikołaj Bojańczyk, Emanuela Merelli, and David P. Woodruff;
Article No. 76; pp. 76:1–76:18
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

The question is to compute a *clearing recovery rate vector*, that is, the ratio between money available (coming from assets and payments from others) over liabilities for each bank. If this ratio is bigger than 1 for a bank, then it will be able pay its dues – in this case, we simply set its rate to 1. The banks that are in default have recovery rates smaller than 1. The problem of computing clearing recovery rates (which we will also refer to as the *clearing problem*) is well understood when there are only simple debt contracts in the network, then clearing recovery rate vectors always exist, are unique, and can be computed in polynomial time [7].

However, Eisenberg and Noe's model in [7] ignores the issue of financial derivatives that may be present in the system. The deregulation allowing banks to invest in these products is considered by many as one of the triggers of the GFC. The introduction of financial derivatives to financial networks is due to [22], where the focus is on a simple and yet widely used class of conditional obligations known as Credit Default Swaps (CDSes), the idea being to "swap" or offset a bank's credit risk with that of another institution. More specifically, a CDS has three entities: a creditor v, a debtor u and a reference bank z – u agrees to pay v a certain amount whenever z defaults. Whilst CDSes were conceived in the early 1990s as a way to protect v from the insolvency of z for direct liabilities (i.e., a (v, z)-arc in the network), they quickly became a speculative tool to bet against the creditworthiness of the reference entity and have in fact been widely used both as a hedging strategy against the infamous collateralised debt obligations, whose collapse contributed to the GFC, and pure speculation during the subsequent eurozone crisis. The clearing problem in the presence of these financial derivatives is somewhat less well-characterised: it is known that the clearing recovery rate vectors can be expressed as the fixed points of a certain function, and existence of solutions is then guaranteed via a fixed-point argument [22]. On the other hand, these fixed points can be irrational, and the computational problem is PPAD-complete [23] as long as one is interested in only a *weak* approximation of a recovery rate vector.

1.1 Our Contributions

In this paper we deepen the study of the clearing problem for financial networks with CDSes from two complementary viewpoints. Firstly, we argue that weak approximations can be misleading in this domain, as the objective under the weak approximation criterion is to find an "almost" fixed point (i.e., a point which is not too far removed from its image under the function). The risk estimate provided by this concept might be very far off the actual rate, thus changing the amount of bailout needed or even whether a bank needs rescue in the first place (see, e.g., our example in Figure 1b below). A more useful (but more difficult) objective is to obtain a strong approximation, that is, a point that is geometrically close to an actual fixed point of the function. Such a risk estimate would be actionable for a regulator, as the error could be measured in terms of irrelevant decimal places. Furthermore, the banks themselves would accept the rate when the strong approximation guarantee is negligible, whereas a weak approximation could significantly misrepresent their income and are subject to be challenged, legally or otherwise.

As our first contribution, we settle the computational complexity of computing strong approximations to the clearing problem in terms of FIXP [9], by showing that the clearing problem is complete for this class. In our reduction, we provide a series of financial network gadgets that are able to compute opportune arithmetic operations over recovery rates. Interestingly, not that many FIXP-complete problems are known, although there are a few important natural such problems (three-or-more-player Nash equilibria being a notable example). Hardness reductions for this class tend to be rather technically involved. The hardness reduction that we provide here indeed has some technical obstacles as well, although our reduction is quite natural at a high level, and could inspire further developments in the area. Our result complements the current state of the art and completes the picture about the computational complexity of the clearing problem with financial derivatives. It shows that computing strongly approximate fixed points is harder than computing weakly approximate fixed points, which holds due to PPAD being equal to the class Linear-FIXP, which is a restriction of FIXP, and this makes PPAD (indirectly) a subclass of FIXP.

▶ **Main Theorem 1** (Informal)**.** *Computing a strong approximation to the clearing recovery rates in a financial network with CDSes is FIXP-complete.*

The FIXP-hardness of the strong approximation problem indicates that there is an additional numerical aspect contributing to the hardness of the problem, which is not present in the weak approximation problem (where the hardness is of a combinatorial nature, due to the reducibility to the end-of-the-line problem which is canonical to PPAD). For the strong approximation problem, the nature of the underlying function for which we want to find the fixed points requires, in particular, the multiplication operation, which ultimately accounts for irrationality and super-polynomial numerical precision being necessary in order to derive whether a given point is a strong approximation to a clearing vector.

We then turn our attention towards irrational solutions with the goal to determine the source of irrationality and understand when it is possible to compute the clearing recovery rate vector exactly in the form of rational numbers. We identify a structural property of cycles in an opportunely enriched network that leads to unique irrational solutions. This property exactly differentiates the CDSes that produce and propagate irrationality of the recovery rates, that we call "switched on", from those that do not, termed "switched off". We prove the following close-to-tight characterisation of irrationality:

▶ **Main Theorem 2** (Informal)**.** *If the financial network has only "switched on" CDSes in a cycle* and *the cycle cannot be shortcut with paths of length at most three then there exist rational values for debt and asset values for which the recovery rate vector is unique and irrational. Conversely, if every cycle of the financial network does not have any "switched on" CDSes then we can compute rational recovery rates in a polynomial number of operations, provided that we have oracle access to PPAD.*

Our proof of irrationality uses a graph "algebra" (i.e., a set of network fragments and an operation on them) that is able to generate all the possible cycle structures with the property above, which uncovers a connection between the network structure of the clearing problem and the roots of non-linear equations. For the opposite direction, we provide an algorithm that exploits the acyclic structure of financial networks with solely "switched off" CDSes. This algorithm iteratively computes the recovery rates of each strongly connected component of the network. We show that even for the simpler topologies of the financial system under consideration, the problem remains PPAD-hard, hence the need for the oracle access to PPAD.

Significance of Our Results. We see our results as important analytical tools that legislators can use to regulate financial derivatives. For example, our results contribute to the ongoing debate in the US and Europe about banning speculative uses of CDSes. In particular, they support, from a computational point of view, the call to ban so-called "naked" CDSes (as already done by the EU for sovereign debt in the wake of the Eurozone crisis, see [1]). A naked CDS is purely speculative since its creditor and debtor have no direct liabilities with the reference entity. It turns out that these CDSes add arcs between potentially unconnected nodes, thus possibly adding more of the cycles that lead to irrationality and, given that strong approximations are out of scope due to our FIXP-hardness, it is not only combinatorially but also numerically intractable to gain insight in the systemic risk of such financial networks. A mechanism to monitor the topology of a financial network might be useful to avoid the construction of cyclical structures that include CDSes.

Both of our main results introduce significant novel technical and conceptual innovations to the field. As mentioned above, our reduction for the FIXP-hardness is somewhat more direct than in previous work we are aware of. Our reduction is direct, in the sense that it starts from the algebraic circuit defined by an arbitrary problem in FIXP. The reduction employs two main steps: We firstly force the outputs of all gates in the circuit to be in the unit cube, by essentially borrowing arguments from [9], after which we produce a series of network gadgets that preserve gate-wise the computations of said circuit; this makes the reduction conceptually straightforward in its setup.

It is worth highlighting a specific technical challenge that we overcome in our proof, as we think it sheds further light on FIXP, and in particular, on the operator basis of the algebraic circuits that are used to define the class. It is known that the circuit of problems in FIXP can be restricted without loss of generality on the arithmetic basis $\{\max, +, *\}$ [9], whereas restricting the internal signals of the circuit to the unit cube (with the toolkit developed in [9]) needs some further operators, including /. For our optimisation problem to be in FIXP, we need the rather mild and realistic assumption that our instances are *non-degenerate* as defined in [23]. The function of which the fixed points define the recovery rates of non-degenerate instances is well defined, where the non-degeneracy is needed to avoid a division by 0. It turns out that non-degeneracy is incompatible with division being part of the FIXP operator basis, i.e., it seems difficult to build such a financial network that in any sense simulates a division of two signals in an algebraic circuit. To bypass this problem, our proof shows that it is possible to substitute / in the basis with the square root operator, $\sqrt{\cdot}$, whilst keeping the function well defined. This substitution can be used to simulate division with constant large powers of 2, and this turns out to be sufficient to omit the /-operator (i.e., arbitrary division). This novel observation might be useful for other problems where division is problematic to either define the fixed point function, or the reduction.

Our second result indirectly aims at characterising the "rational fragment" of FIXP: To the best of our knowledge this is the first study in this direction. A couple of observations can be drawn from our attempt. Firstly, our sufficiency conditions for irrationality suggest that any such characterisation needs to fully capture the connection between the fixed point condition and the rational root theorem; our proof currently exploits the cyclical structure of networks with "switched on" CDSes to define one particular quadratic equation with irrational roots. Whilst this captures a large class of instances, more work is needed to give a complete characterisation (see Section 6 for a discussion). Secondly, our sufficiency conditions for rational solutions highlight a potential issue with their representation. Due to the operations in the arithmetic basis, most notably multiplication, these solutions can grow exponentially large (even though each call to the PPAD oracle returns solutions of size polynomial in their input). This observation establishes a novel connection between the Blum-Shub-Smale computational model [5] (wherein the size to store any real number is assumed to be unitary and standard arithmetic operations are executed in one time unit), the rational part of FIXP, and PPAD. Our result paves way to further research on the subject.

Further Related Work. Systemic risk in financial networks has been studied extensively in the literature [3, 8, 13, 15, 16, 17, 6, 21, 19]. Game-theoretic perspectives of financial networks are studied in [4, 20]. Fixed point computations of total search problems, are studied in [9]. FIXP-complete problems are presented in [14, 10, 12, 11].

2 Model and Preliminaries

2.1 Financial Systems

Let $N = \{1, \ldots, n\}$ be a set of n banks. Each bank $i \in N$ has *external assets*, denoted by $e_i \in \mathbb{Q}_{\geq 0}$ and $e = (e_1, \ldots, e_n)$ is the external assets vector. We consider two types of liabilities among banks: *debt contracts* and *credit default swaps (CDSes)*. A debt contract requires one bank i (debtor) to pay another bank j (creditor) a certain amount $c_{i,j} \in \mathbb{Q}_{\geq 0}$. We denote by $\mathcal{DC}$ the set of all pairs of banks participating in a debt contract. A CDS requires a debtor i to pay a creditor j on condition that a third bank called the *reference bank* R is in default, meaning that R cannot fully pay its liabilities. Formally, we associate each bank i a variable $r_i \in [0, 1]$, called the *recovery rate*, that indicates the proportion of liabilities it can pay. Having $r_i = 1$, means bank i can fully pay its liabilities, while $r_i < 1$ indicates that i is in default. In case a reference bank R of a CDS is in default, the debtor i of that CDS pays the creditor j an amount of $(1 - r_R)c^R_{i,j}$, where $c^R_{i,j} \in \mathbb{Q}_{\geq 0}$ is the face value of the CDS. We denote by $\mathcal{CDS}$ the set of all triplets participating in a credit default swap. The value $c_{i,j}$ ($c^R_{i,j}$, resp.) of a debt contract (CDS, resp.) is referred to as the *notional* of the contract. Finally we let c be a three-dimensional $(n \times n \times n)$ matrix containing all contract notionals; we do not allow any bank to have a debt contract with itself, and assume that all three banks in any CDS are distinct.

▶ **Definition 1.** *A financial system is a triplet (N, e, c), where $N = \{1, .., n\}$ is a set of banks, $e = (e_1, .., e_n) \in \mathbb{Q}^n_{\geq 0}$ is the vector of external assets, and $c \in \mathbb{Q}^{n\times n\times n}_{\geq 0}$ is the three-dimensional matrix of all contract notionals.*

The *contract graph* of $I = (N, e, c)$ is defined as a directed multigraph $G_I = (V, A)$, where $V = N$ and $A = (\cup_{k\in N} A_k) \cup A_0$ where $A_0 = \{(i, j) \mid c_{i,j} \neq 0\}$ and $A_k = \{(i, j) \mid c^k_{i,j} \neq 0\}$. Each arc $(i, j) \in A_0$ is coloured blue and each $(i, j) \in A_k$ orange. For all $(i, j, R) \in \mathcal{CDS}$ we draw a dotted orange line from node R to arc $(i, j) \in A_R$, denoting that R is the reference bank of the CDS between i and j. Finally, we label each arc with the notional of the corresponding contract, and each node with the external assets of the corresponding bank.

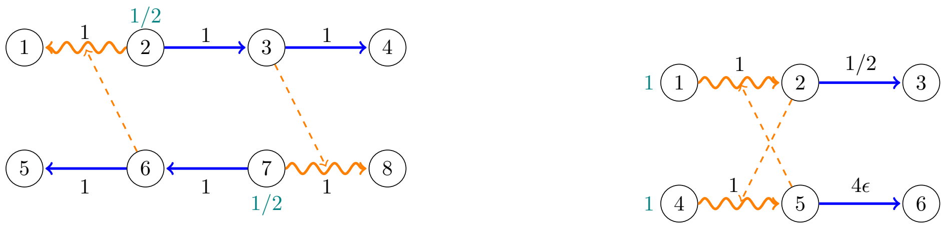
Given a recovery rate vector $r \in [0, 1]^n$, we define the *liabilities*, *payments*, and *assets* in a financial system as follows. The liability of a bank $i \in N$ to a bank $j \in N$ under r is denoted by $l_{i,j}(r) = c_{i,j} + \sum_{k\in N}(1 - r_k)c^k_{i,j}$. That is, we sum up the liabilities from the debt contract and all CDS contracts between i and j. We denote by $l_i(r)$ the total liabilities of i: $l_i(r) = \sum_{j\in N} l_{i,j}(r)$. The payment bank i makes to bank j under r is denoted by $p_{i,j}(r)$, where $p_{i,j}(r) = r_i \cdot l_{i,j}(r)$. The assets of a bank i under r are the total amount it possesses through its external assets and incoming payments made all by other banks, i.e., $a_i(r) = e_i + \sum_{j\in N} p_{j,i}(r)$. Our research focuses on *clearing recovery rate vectors* (CRRVs).

▶ **Definition 2.** *Given a financial system (N, e, c), a recovery rate vector r is called clearing if and only if for all banks $i \in N$, $r_i = \min\{1, a_i(r)/l_i(r)\}$, if $l_i(r) > 0$, and $r_i = 1$ if $l_i(r) = 0$.*

We set to 1 the recovery rate of nodes without liabilities, whereas [22] leaves these unconstrained. This is in line with the interpretation that these banks are not in default and only a cosmetic difference, as discussed in the full version of the paper [18].

We call CDS-CLEARING the problem of computing a CRRV in a given financial system with debt contracts and credit default swaps. For an instance $I \in$ CDS-CLEARING any clearing vector is a *solution* and the solution set is denoted by $\mathrm{Sol}(I)$. Let $I \in$ CDS-CLEARING and consider $f_I : [0, 1]^n \mapsto [0, 1]^n$ defined at each coordinate $i \in [n]$ by $f_I(r)_i = a_i(r)/(\max\{l_i(r), a_i(r)\})$. It is easy to see that $Sol(I)$ actually consists of the fixed point of function f_I. The existence of at least one fixed point of f_I for every $I \in$ CDS-CLEARING was proved in [22]. Unfortunately, there exist instances of CDS-CLEARING in which all clearing vectors have irrational values, one example given in [23]. We present another example, to also illustrate our contract graphs.

▶ **Example.** *Figure 1a consists of eight banks, $N = \{1, \ldots, 8\}$. They all have external assets 0 except for banks 2 and 7 ($e_j = 0$ for $j \neq 2, 7$, $e_2 = e_7 = 1/2$). The set of debt contracts is $\mathcal{DC} = \{(2,3),(3,4),(6,5),(7,6)\}$ and the set of CDSes is $\mathcal{CDS} = \{(2,1,6),(7,8,3)\}$. All contract notionals are set to 1. For a recovery rate vector r, node 2's liability is $l_2(r) =$*



(a) CRRVs can be irrational.

(b) Approximations of CRRVs.

Figure 1 On the left a financial system (a) with irrational solutions. On the right a financial system (b) where the weak approximate fixed point is far from the actual fixed point.

$l_{2,3}(r) + l_{2,1}(r) = c_{2,3} + (1 - r_6)c^6_{2,1} = 2 - r_6$. For node 3, it holds $l_3(r) = l_{3,4}(r) = c_{3,4} = 1$. The assets of node 2 are $a_2(r) = e_2 = 1/2$ whereas $a_3(r) = e_3 + p_{2,3}(r) = r_2 c_{2,3} = r_2$. Symmetrically, $l_7(r) = 2 - r_3, a_7(r) = 1/2$ and $l_6(r) = 1, a_6(r) = r_7$. For node 1 it holds that $a_1(r) = e_1 + p_{2,1}(r) = r_2(1 - r_6)c^6_{2,1}$ and for node 8 $a_8(r) = r_7(1 - r_3)c^3_{7,8}$. From the above computations and by applying the CRRV condition we get that any solution must satisfy: $r_2 = \min\{1, 1/(2(2 - r_6))\}$, $r_6 = r_7$, $r_7 = \min\{1, 1/(2(2 - r_3))\}$, and $r_3 = r_2$, implying that $2r_2^2 - 4r_2 + 1 = 0$ and then $r_2 = r_3 = r_6 = r_7 = 1 - \sqrt{2}/2$ and $r_1 = r_5 = r_4 = r_8 = 1$.

2.2 Approximation and Complexity

Let F be a continuous function that maps a compact convex set to itself and let $\epsilon > 0$ be a small constant. A *weak ϵ-approximate fixed point* of F is a point x such that $\|x - F(x)\|_\infty < \epsilon$. A *strong ϵ-approximate fixed point* of F is a point x s.t $\exists x' : F(x') = x' \wedge \|x' - x\|_\infty < \epsilon$. Moreover, under a mild condition on the fixed point problem under consideration, known as *polynomial continuity*, a strong approximation is also a weak approximation [9], which explains the use of the terms "strong" and "weak".

Formally a *fixed point problem* Π is defined as a search problem such that for every instance $I \in \Pi$ there is an associated continuous function $F_I : D_I \to D_I$ where $D_I \subseteq \mathbb{R}^n$ (for some $n \in \mathbb{N}$) is compact and convex, such that the solutions of I are the fixed points of F_I. The problem Π is said to be *polynomially computable* if there is a polynomial q such that (i.) D_I is a convex polytope described by a set of at most $q(|I|)$ linear inequalities, each with coefficients of a size at most $q(|I|)$, and (ii.) For each x in $D_I \cap \mathbb{Q}^n$, the value $F_I(x)$ can be computed in time $q(|I| + \text{size}(x))$. Here, the "size" of a rational number means the number of bits needed to represent the numerator and the denominator in binary. Furthermore Π is said to be *polynomially continuous* if there is a polynomial q such that for each $I \in \Pi$, and rational $\epsilon > 0$, there is a rational δ of size $q(|I| + \text{size}(\epsilon))$ satisfying the following: for all $x, y \in D_I$ with $\|x - y\|_\infty < \delta$ it holds that $\|F_I(x) - F_I(y)\|_\infty < \epsilon$.

With regard to CDS-CLEARING, it is straightforward to verify that CDS-CLEARING is polynomially computable. Furthermore, [23] establishes implicitly that CDS-CLEARING is polynomially continuous under a (very) mild assumption that the authors call *non-degeneracy*.

▶ **Definition 3.** *A financial system is* non-degenerate *if and only if the following two conditions hold. Every debtor in a CDS either has positive external assets or is the debtor in at least one debt contract with a positive notional. Every bank that acts as a reference bank in some CDS is the debtor of at least one debt contract with a positive notional.*

We define CDS-CLEARING to contain only non-degenerate financial networks, both for the sake of compatibility with [23] and for the analytical convenience that non-degeneracy provides us (note that a division by 0 never occurs in $f_I(r)_i$ for these instances). In [23], it is also shown that the weak approximation version of CDS-CLEARING is PPAD-hard. The polynomial continuity of CDS-CLEARING shows that the strong approximation version of CDS-CLEARING is at least as hard as its weak approximation version. As noted above, weakly approximate fixed points may contain misleading information about whether a bank is in default or not, as shown in the next example. This motivates the study of strong approximations.

▶ **Example.** *Consider the instance in Figure 1b. It is not hard to see that $r = (1, 1, 1, 1, 0, 1)$ is an exact fixed point; $r' = (1, 1 - 2\epsilon, 1, 1, 1/2 + \epsilon, 1)$ is instead a weakly ϵ-approximate fixed point since $f_2(r') = 1 - 2\epsilon$ and $f_5(r') = 1/2$ implying that $\|r' - f(r')\|_\infty \le \epsilon$. We observe that r is very far from r' and, in particular, as $r'_2 = 1 - 2\epsilon < 1$, we would conclude that 2 is in default whereas 2 can actually fully pay its liabilities since $r_2 = 1$.*

FIXP is the complexity class introduced to study the strong approximation and exact versions of fixed point problems [9].

▶ **Definition 4.** *The class FIXP consists of all fixed point problems Π that are polynomially computable, and for which for all $I \in \Pi$ the function $F_I : D_I \to D_I$ can be represented by an algebraic circuit C_I over the basis $\{+, -, *, \max, \min\}$, using rational constants, such that C_I computes F_I, and C_I can be constructed from I in time polynomial in $|I|$.*

The class FIXP$_a$ is defined as the class of search problems that are the strong approximation version of some fixed point problem that belongs to FIXP.

The class Linear-FIXP is defined analogously to FIXP, but under the smaller arithmetic basis where only the gates $\{+, -\max, \min\}$ and multiplication by rational constants are used.

The classes FIXP, Linear-FIXP, and FIXP$_a$ admit complete problems. Hardness of a search problem Π for FIXP (resp. Linear-FIXP and FIXP$_a$) is defined through the existence of a polynomial time computable function $\rho : \Pi' \to \Pi$, for all $\Pi \in$ FIXP (resp. FIXP$_a$), such that the solutions of I can be obtained from the solutions of $\rho(I)$ by applying a (polynomial-time computable) linear transformation on a subset of $\rho(I)$'s coordinates. This type of reduction is known as a *polynomial time SL-reduction.*

It is known that FIXP$_a \subseteq$ PSPACE and Linear-FIXP = PPAD [9]. Consequently, the solutions of instances in Linear-FIXP are always rationals of polynomial size. An informal understanding of how the hardness of FIXP compares to PPAD (or Linear-FIXP) is as follows. PPAD captures a type of computational hardness stemming from an essentially combinatorial source. The class FIXP introduces on top of that a type of numerical hardness that emerges from the introduction of multiplication and division operations: These operations give rise to irrationality in the exact solutions to these problems, and may independently also require the computation of rational numbers of very high precision or very high magnitude.

3 FIXP-Completeness of CDS-Clearing

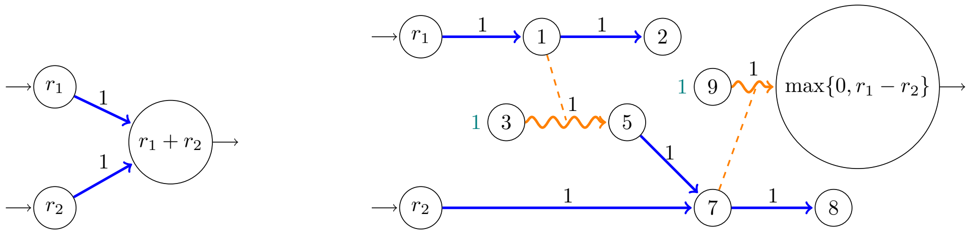
Our first main result shows that CDS-CLEARING and its strong approximation variant are FIXP$_{(a)}$ complete. We show that we can take an arbitrary algebraic circuit and encode it in a direct way in the form of a financial system. Hence, our polynomial time hardness reduction is implicitly defined to work from to any arbitrary fixed point problem in FIXP. The reduction is constructed by devising various financial network gadgets which enforce that certain banks in the system have recovery rates that are the result of applying one of the operators in FIXP's arithmetic base to the recovery rates of two other banks in the system: In other words, we can design our financial systems such that the interrelation between the recovery rates mimics a computation through an arbitrary algebraic circuit.

▶ **Theorem 5.** *CDS-CLEARING is FIXP-complete, and its strong approximation version is* FIXP_a*-complete.*

Proof Sketch. The clearing vectors for an instance $I \in$ CDS-CLEARING are the fixed points of the function f_I defined above, which can be computed using a polynomial size algebraic circuit with only $\{\max, +, *\}$, and rational constants. Note that non-degeneracy of I prevents division by 0, so that the output of the circuit is well-defined for every $x \in [0,1]^n$. This shows that CDS-CLEARING is in FIXP and that its strong approximation version is in FIXP_a.

For the FIXP-hardness of the problem, let Π be an arbitrary problem in FIXP. We describe a polynomial-time reduction from Π to CDS-CLEARING. Let $I \in \Pi$ be an instance, let $F_I : [0,1]^n \to [0,1]^n$ be I's associated fixed point function, and let C_I be the algebraic circuit corresponding to F_I. As a pre-processing step, we convert C_I to an equivalent alternative circuit C'_I that satisfies that all the signals propagated by all gates in C'_I and all the used rational constants in C'_I are contained in the interval $[0,1]$. The transformed circuit C'_I may contain two additional type of gates: Division gates and gates that computes the absolute value of the difference of two operands. We will refer to the latter type of gate as an *absolute difference gate*. The circuit C'_I will not contain any subtraction gates, and will not contain max and min gates either. The transformation procedure for C_I follows the same approach of the transformation given in Theorem 4.3 of [9] where the 3-Player Nash equilibrium problem is proved FIXP-complete, and borrows some important ideas from there. Nonetheless, there are important differences in our transformation, starting with the fact that we use a different set of types of gates in our circuit. (Details can be found in the full proof in [18].)

For notational convenience, in the remainder of the proof we may treat C'_I as the function F_I, hence we may write $C'_I(x) = y$ to denote $F_I(x) = y$. Let ρ denote the reduction to CDS-CLEARING: We construct our instance $\rho(I)$ of CDS-CLEARING (i.e., a non-degenerate financial network) from the circuit C'_I. The instance $\rho(I)$ will have the property that its clearing vectors are in one-to-one correspondence with the fixed points of C'_I, and that banks $1, \ldots, n$ in our construction correspond to the input gates of C'_I. More precisely, our construction is such that for each fixed point x of C'_I, in the corresponding clearing vector r for $\rho(I)$ it holds that $(r_1, \ldots, r_n) = x$. Our reduction works through a set of financial system *gadgets*, of which we prove that their recovery rates (under the clearing condition) must replicate the behaviour of each type of arithmetic operation that can occur in the circuit C'_I. Each of our gadgets is non-degenerate, has one or two *input banks* that correspond to the input signals of one of the types of arithmetic gate, and there is an *output bank* that corresponds to the output signal of the gate. For each of the gadgets, it holds that the output bank must have a recovery rate that equals the result of applying the respective arithmetic operation on the recovery rates of the input banks, see examples in Figure **??** ($g_{\text{pos}-}$ is a building block for the absolute difference gadget). Besides gadgets for the necessary arithmetic operations, our reduction employs an additional *duplication gadget* g_{dup} that can be used to connect the output of a particular gadget to the input of more than one subsequent gadget. A technically involved step is needed for the division gates; we replace some of the divisions in the circuit C'_I by taking successive square-roots followed by successive squaring operations, where proper care has to be taken to ensure that the results of all these operations stay within the interval $[0, 1]$. Full definitions of our gadgets (including gadgets for squares and square roots) can be found in [18].



(a) Addition gadget g_+. **(b)** Positive subtraction gadget $g_{\text{pos}-}$ computes $\max\{0, r_1 - r_2\}$.

Figure 2 Exemplar gadgets from our reduction ρ.

In our financial system, these gadgets are then connected together according to the structure of the circuit C'_I: Output banks of (copies of) gadgets are connected to input banks of other gadgets through a single unit-cost debt contract, which mimics the propagation of a signal between two gates of the arithmetic circuit. This results in a financial system whose behaviour replicates the behaviour of the arithmetic circuit. The first layer of the financial system consists of n banks representing the n input nodes of the circuit, and the last layer of the financial system has n banks corresponding to the n output nodes of the circuit. As a final step in our reduction, the n output banks in the last layer are connected through a single unit-cost debt contract to the n input banks. This last step enforces the recovery rates of the input banks (i.e., banks $1, \ldots, n$) are equal to the recovery rates of the last layer, under the clearing requirement. Consequently, any vector of clearing recovery rates for $\rho(I)$ must then correspond to a fixed point of C'_I, where the recovery rates of the first n banks in the system equal those of the final n banks, so that $C'_I(r_1, \ldots, r_n) = (r_1, \ldots, r_n)$, i.e., $(r_1, \ldots, r_n)$ is a fixed point of C'_I. It is clear that $\rho(I)$ can be constructed in polynomial time from C'_I, and since C'_I can be constructed in polynomial time from I, the financial system $\rho(I)$ takes polynomial time to compute.

FIXP_a-completeness of strong approximations holds since any strong ϵ-approximation of the CRRV of $\rho(I)$ corresponds to a strong ϵ-approximate fixed point of C'_I. ◀

4 A Sufficient Structural Condition for Irrational Solutions

In this section we investigate the existence of irrational solutions in financial systems in more depth. Our starting point is the observation that irrational clearing recovery rates can only arise under certain structural conditions on the financial system (e.g., a system with no CDSes has a rational CRRV [7]). Which structural conditions must exactly hold in a financial system for irrational clearing vectors to potentially exist? In this section, we present a set of sufficient structural conditions that provides a partial answer to this question.

4.1 Switched Cycles

We define the *auxiliary graph* $G_{I,\text{aux}}$ of $I = (N, e, c)$ to be a tricoloured directed graph obtained from G_I by adding a red-coloured arc (R, i) for every $(i, j, R) \in \mathcal{CDS}$. The auxiliary graph corresponding to the instance in Figure 1b is given in Figure 3. We say that an instance I is *acyclic* if and only if its $G_{I,\text{aux}}$ contains no directed cycle. It is not hard to see that every acyclic financial system has only rational solutions. (We defer the proof to the full version [18].)

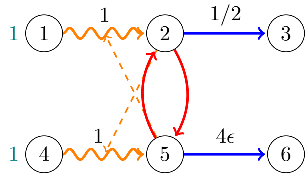


Figure 3 The auxiliary graph for the instance in Figure 1b.

We say that a node $i \in N$ is *switched off* iff it has only one incoming red arc and no outgoing blue arcs. A node $i \in N$ is *switched on* iff its incoming red arcs exceeds 1 or its incoming red arcs equals 1 and there is at least one outgoing blue arc. Note that switched on and switched off nodes are not complements of each other. A node that is not a debtor in any CDS is neither switched on nor switched off. These notions are illustrated in Figure 4.

▶ **Definition 6** (Switched Cycles). *A cycle is* red *iff it has at least one red arc. A cycle is* weakly switched *iff it is red, and for at least one red arc (u, v) in C, v is switched on. A cycle is* strongly switched *iff it is red, and for each red arc (u, v) in C, v is switched on.*

We will prove that when a non-degenerate financial system I has a strongly switched cycle (and a certain additional technical condition holds), there exist coefficients for the financial system under which all CRRVs of I are irrational. Our proof introduces a framework for formulating strongly switched cycles and consists of three main steps. Firstly, we define a set of primitive financial systems without notionals and external assets, called *fragments*. Each of these fragments has a designated start and end node. A binary concatenation operation is also introduced so to obtain financial systems that are obtainable by "stringing" together fragments. We refer to graphs obtainable through this operation as *fragment strings* or *cycles* (when the end node is linked back to the initial start node). Secondly, we equip each fragment with particular choices of rational coefficients to define *arithmetic fragments*; these allow to conveniently rewrite fragment strings, given that the objective is to preserve recovery rates at the end nodes. We prove that each of the resulting arithmetic fragment cycles has irrational CRRVs. Finally, we show that a particular class of strongly switched cycles are constructible from these fragments and for each instance I with these cycles there exist rational coefficients also for nodes and arcs not in the cycle such that all the CRRVs of I are irrational.

4.2 Fragments

We denote by $\mathcal{G}$ the set of all fragments that we will use. Few representatives of $\mathcal{G}$ are defined in Figure 5 (left), presented in our tricoloured graphical notation. Start and end nodes are indicated by short incoming and outgoing black arrows, respectively. The full description of

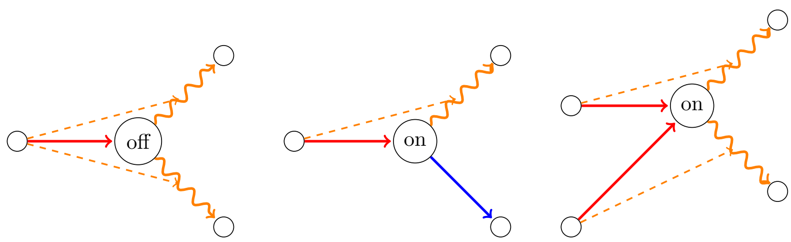


Figure 4 One switched off and two switched on nodes.

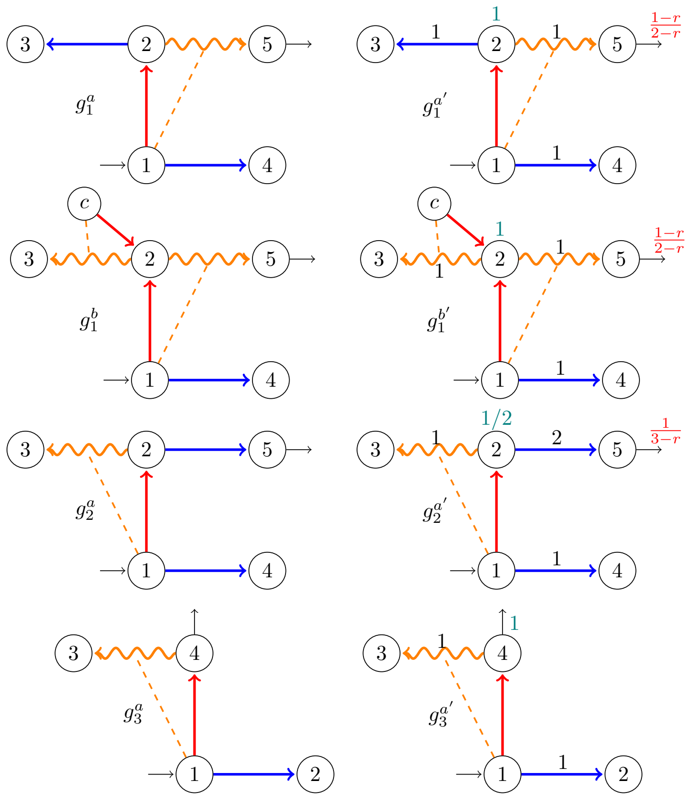


Figure 5 Some fragments in $\mathcal{G}$ (left) with their arithmetic version (right). Each fragment is labeled with its name.

$\mathcal{G}$ can be found in the full paper [18]; the additional fragments are either variants of those shown here (substituting direct liabilities with CDSes as in g_1^b vis-a-vis g_1^a) – called g_i^j, with $j \in \{a, b, c, d\}$ when $i \in [2]$ and $j \in \{a, b\}$ for $i = 3$ – or simpler configurations that just copy the recovery rate from start to end node, called d_1 and d_2.

We define a binary merging operation on ordered pairs of fragments (a, b), where every pair (a, b) is mapped to a graph obtained by taking disjoint copies of a and b, and connecting the two copies together by identifying the end node of a with the start node of b. The new start node and end node of the resulting system is the start node of the copy of a and the end node of the copy of b, respectively. We denote the result of the merge operation on fragments a and b symbolically by the notation ab. A *fragment string* is a fragment obtainable from fragments in $\mathcal{G}$ using any number of sequential applications of the merge operation. We let $\mathcal{GS}$ be the set of fragment strings (i.e., the closure of $\mathcal{G}$ under the *merge* operation). By identifying the start node with the end node of a fragment string we obtain a *fragment cycle*. The induced fragment cycle is denoted $\dot{x}g_s\dot{x}$, where $x \in \mathcal{G}$ and $g_s \in \mathcal{GS}$. Let $\mathcal{GC}$ to be the set of fragment cycles.

A fragment with fixed coefficients is called an *arithmetic fragment*, see Figure 5 (right) where we omit to show 0 external assets for some nodes. We denote by x' or x'' the arithmetic version of $x \in \mathcal{G}$. The difference between x' and x'' are minimal; for the fragments in Figure 5, the only difference between x' and x'' is that the notional for the bottom right liability (e.g., arc from node 1 to node 4 in g_1^a) is valued 2 rather than 1. The red labels at the end of an arithmetic fragment indicate the assets of the end node under any clearing vector as a function of the recovery rate r of the start node. Importantly, both x' and x'' have the same recovery rate at the end node. The *merge* operation and notation used for fragments apply to arithmetic fragments as well. The following observation can be derived by inspection.

▶ **Observation 7.** *Let x'_1 and x'_2 be any two consecutive arithmetic fragments in a string or cycle C of arithmetic fragments. Let r be the recovery rate of the start node of x'_1 under a clearing vector of C. It holds that:*

- *If $x'_1 \in \{g_i^{j'}, g_i^{j''} : i \in [2], j \in \{a,b,c,d\}\}$ and $x'_2 \in \{g_i^{j'} : i \in [2], j \in \{a,b,c,d\}\} \cup \{d'_1, d'_2\}$, then the recovery rate of the end node of x'_1, is $(1-r)/(2-r)$ or $1/(3-r)$.*
- *If $x'_1 \in \{g_3^{j'}, g_3^{j''} : j \in \{a,b\}\}$ and $x'_2 \in \{g_i^{j''} : i \in [3], j \in \{a,b,c,d\}\}$, then the recovery rate of the end node of x'_1 is $1/(3-r)$.*

We now give a notion of equivalence between arithmetic fragment strings.

▶ **Definition 8.** *Let x_s^1, x_s^2 be two arithmetic fragment strings. We say that x_s^1 and x_s^2 are* equivalent *iff the recovery rate of the end node of x_s^1 equals the recovery rate of the end node of x_s^2 for all possible choices $r \in [0,1]$ of the recovery rate of the input nodes of x_1^s and x_2^s.*

Equivalence enables us to simplify big fragment string and cycles to simpler ones while preserving the recovery rate of the end node. This is achieved by a set of rewriting rules.

Rule 0: Replace an occurrence of a fragment $g_i^{j'}$ ($g_i^{j''}$, respectively), where $i \in [3]$ and $j \in \{a,b,c,d\}$, with the fragment $g_i^{a'}$ ($g_i^{a''}$, respectively).

Rule 1: Replace an occurrence of a fragment $g_2^{a'}$ (respectively $g_2^{a''}$) by $g_1^{a'} g_1^{a'}$ respectively $g_1^{a''} g_1^{a'}$ if the fragment $g_2^{a'}$ (or respectively $g_2^{a''}$) is followed by one of the fragments in $\{g_1^{a'}, g_2^{a'}, g_3^{a'}, d'_1, d'_2\}$.

Rule 2: Replace an occurrence of a consecutive pair of fragments $g'_3 g_i^{a''}$, where $g'_3 \in \{g_3^{a'}, g_3^{a''}\}$, and $i \in [3]$, by the fragments $g_2^{a'} g_i^{a'}$. By Observation 7, the recovery rates of the end nodes of g'_3 and $g_2^{a'}$ are identical under this substitution, under any clearing vector, so that the two fragment strings are equivalent.

Rule 3: Remove an occurrence of d'_1 or d'_2. This substitution is straightforward from the fact that both d'_1 and d'_2 just transfer the recovery rate from the start to the end node.

4.3 Irrationality of Strongly Switched Cycles

Consider any instance $I = (N, e, c)$ with auxiliary graph $G_{I,\text{aux}}$. If I has a strongly switched cycle, then this cycle is composed entirely of the fragments in $\mathcal{G}$. This is formalised as follows.

▶ **Definition 9.** *Let G' be a fragment cycle, and let C' be the unique directed cycle in G'. The fragment cycle G' is said to* agree with a cycle C of $G_{I,\text{aux}}$ *iff there is a mapping $\xi : V(G') \to V(G_{I,aux})$ with the following properties:*

- *For all $(v,w) \in E(G')$, $(\xi(v), \xi(w))$ is in $E(G_{I,aux})$ and has the same color as (v,w).*
- *ξ restricted to the domain $V(C')$ defines a bijection between $V(C')$ and $V(C)$.*
- *For each CDS (i,j,R) in G', $(\xi(i), \xi(j), \xi(R))$ is a CDS of G.*

Note that the above points imply that ξ restricted to $V(C')$ defines an arc-color-preserving isomorphism between C' and C. However, this isomorphism does not necessarily extend to node sets larger than C': nodes in $V(G') \setminus V(C')$ may be mapped by ξ to the same vertex of $G_{I,aux}$. We then define the fragment cycle $G' = G'_n \cup G'_l$ to simply agree *with a cycle C of $G_{I,aux}$, if G' agrees with C of $G_{I,aux}$ through a mapping ξ for which it additionally holds that*

- *all nodes outside C' are mapped to vertices outside C,*
- *For every pair of nodes $\{u,v\} \subseteq V(G')$, where $v \in G'_n$ and $u \in G'_l$, $\xi(u) \neq \xi(v)$, and*
- *for every node $u \in G'_l$, $\xi(u)$ has an outgoing arc pointing towards a node not in C',*

where G'_n is the set of nodes in the fragment cycle G' labelled with a number (as $1, \ldots, 5$ in Figure 5) and G'_l is the set of fragment nodes labelled with a letter (as c in Figure 5).

The notion of *simple* agreement informally requires that the neighbouring nodes of C' are sufficiently "independent" from each other and from the cycle C, under the mapping ξ. This brings us to the definition of a *simple* strongly switched cycle (which makes precise our condition on off-cycle paths between the nodes in the cycles in the informal statement of our second main theorem in the introduction).

▶ **Definition 10.** *A cycle C of $G_{I,aux}$ is a* simple *strongly switched cycle iff C is strongly switched, and for each red arc (u, v) of C there are non-red arcs (u, u') and (v, v') such that $u', v' \notin C$. Furthermore, if (u, u') or (v, v') is orange, then the reference bank R of the corresponding CDS is not in C and R has an outgoing non-red arc pointing to a node not in C.*

The fragments in $\mathcal{G}$, can represent any strongly switched cycle: If $G_{I,\text{aux}}$ has a strongly switched cycle C, then there is a fragment cycle G' consisting of fragments in $\mathcal{G}$ s.t G' agrees with C of $G_{I,\text{aux}}$. Similarly, if $G_{I,\text{aux}}$ has a simple strongly switched cycle C, then there is a fragment cycle G' consisting of fragments in $\mathcal{G}$ s.t G' simply agrees with C of $G_{I,\text{aux}}$. All switched on nodes of C correspond to the 2-labeled nodes of a g_2^j or g_1^j fragment, for some $j \in \{a, b, c, d\}$. The next lemmas show that we can set the coefficients in any strongly switched fragment cycle s.t the fragment cycle admits only irrational clearing recovery rates.

▶ **Lemma 11.** *For all fragment cycles $C \in \mathcal{GC}$ consisting of only fragments in $\{g_1^j : j \in \{a, b, c, d\}\}$, there exist coefficients s.t. the clearing recovery rate vector of C is irrational.*

Proof Sketch. Consider a fragment cycle consisting exclusively of only fragments in $\{g_1^j : j \in \{a, b, c, d\}\}$. For all $j \in \{a, b, c, d\}$, fix the coefficients of all g_1^j fragments in the cycle to obtain the arithmetic version $g_1^{j'}$. Use rewriting Rule 0 to replace all $g_1^{j'}$ occurrences by $g_1^{a'}$. The resulting arithmetic fragment cycle consists of a number of consecutive copies of $g_1^{a'}$, say k of them. Consider now any clearing vector r for the fragment cycle. We can prove by induction (details in the full paper [18]) that the end node of the ith fragment has recovery rate equal to $(f_i - rf_{i-2})/(f_{i+2} - rf_i)$, where f_i is the ith Fibonacci number, with $f_0 = 0$.

We know that the end node of the last fragment in the fragment cycle has a recovery rate that coincides with the recovery rate r of the start node of the first fragment. Therefore, in a clearing vector of recovery rates, it holds that $r = (f_n - rf_{k-2})/(f_{k+2} - rf_k)$ which is equivalent to solving the equation $r^2 f_k - (f_{k+2} + f_{k-2})r + f_k = 0$. Since $f_{k+2} + f_{k-2} = f_{k+1} + f_k + f_{k-2} = 2f_k + f_{k-1} + f_{k-2} = 3f_k$, computing the recovery rate of the initial node 1 comes down to solving the quadratic equation $r^2 - 3r + 1 = 0$. Solving this equation we obtain that the only solution in $[0, 1]$ is $r = (3 - \sqrt{5})/2$ which is irrational, thus the CRRV of the strongly switched arithmetic fragment cycle is irrational and is unique. ◀

The next lemma (proof omitted) extends the above to a larger class of arithmetic fragments.

▶ **Lemma 12.** *For all fragment cycles composed of fragments $\mathcal{G}$ in which every occurrence of a fragment in $\{g_3^j : j \in \{a, b\}\}$ is followed by a fragment in $\{g_i^j : i \in [2], j \in \{a, b, c, d\}\}$, there exist coefficients s.t the clearing recovery rate vector of C is irrational.*

▶ **Theorem 13.** *Let I be a non-degenerate financial system such that $G_{I,aux}$ has a simple strongly switched cycle. Then there exist rational coefficients for I such that all clearing vectors of I are irrational.*

Proof Sketch. Let C be a strongly switched cycle of $G_{I,\text{aux}}$ and let G' be a fragment cycle that simply agrees with C through a mapping ξ satisfying the conditions stated in Definition 9. By Lemma 12, there are coefficients for G' such that all clearing vectors of G' are irrational.

In $G_{I,\mathrm{aux}}$, we can now set the notionals and external assets on the vertices and arcs through the mapping ξ. This assignment of coefficients is well-defined by the properties of ξ stated in Definition 9 (i.e., there are no two arcs or vertices that get assigned multiple conflicting coefficients this way). We set the remaining coefficients of $G_{I,\mathrm{aux}}$ (i.e., the coefficients on the arcs and vertices outside the image of ξ) as follows: external assets to 0; notionals of (v, w) to 1, if $\xi^{-1}(v)$ is a node labeled with a letter, and w is not in the image of ξ, or 0 viceversa.

Let G'' denote the subgraph of G formed by the image of ξ. Note that no payments flow from G'' to any node outside G'' under any clearing vector. It then follows by Lemma 12 and the simple agreement properties, that under this setting of the coefficient of $G_{I,\mathrm{aux}}$, every clearing vector is irrational (and in particular these irrational recovery rates emerge in the nodes of G''). This establishes our claim. ◀

5 Financial Systems with Guaranteed Rational Solutions

In the previous section, we identified a sufficient structural condition for the ability of a financial system to have irrational clearing vectors. In this section we investigate how close these conditions are to a characterisation, by attempting to answer the opposite question: Under which structural conditions are rational clearing vectors guaranteed to exist in a financial system? The answer to this relates again to the notion of switched cycles: We will show that if a given non-degenerate financial system does not possess any weakly switched cycle, then there must exist clearing vectors of the system that are rational. We investigate furthermore the computational complexity of finding a clearing vector in this case: Solutions can, informally stated, be computed by solving a linear number of PPAD-complete problems. This latter result is achieved through identifying a natural class of financial systems for which the problem of computing an exact fixed point is PPAD-complete.

The results in this section indicate that the structural conditions for irrationality formulated in the previous section do close in on a characterisation, although there is still a "gray area" left: For those instances of financial systems that do have weakly switched cycles, but do not have any simple strongly switched cycles, we are not yet able to determine by the structural interrelationships of the financial contracts whether these systems are likely to possess rational or irrational solutions. This forms an interesting remaining problem that we leave open. The main result we will prove in this section is thus the following.

▶ **Theorem 14.** *Let I be a non-degenerate financial system. If $G_{I,aux}$ does not have any weakly switched cycles, then all clearing vectors of I are rational.*

We start by showing that for a particular subclass of financial systems without weakly switched cycles, the clearing vector computation problem lies in Linear-FIXP, which is equal to PPAD, and thus the clearing vectors of such financial system must have polynomial size rational coefficients.

▶ **Definition 15.** *An instance $I = (N, e, c)$ of a financial system is said to have the* dedicated CDS debtor property *iff for every node $i \in N$ that is a debtor of at least one CDS of I, the following holds: There are no debt contracts (with a non-zero notional) in which i is the debtor, and all CDSes (with a non-zero notional) for which i is the debtor share the same reference bank.*

▶ **Lemma 16.** *(The exact computation version of)* CDS-CLEARING *restricted to non-degenerate financial systems with the dedicated CDS debtor property is PPAD-complete.*

The proof, which is omitted, works by showing that for this special case of the problem, one can rewrite function f into a function f' where multiplication is omitted. This is done by disregarding the recovery rates of those nodes that are debtors of CDSes and instead expressing their individual CDS payments in a way that does not require multiplication. Furthermore, the remaining nodes do not need multiplication under our original formulation of f. Secondly, PPAD-hardness is established from minor modifications of the proof of the main theorem in [23].

The above PPAD-completeness result (and more precisely the PPAD-membership part of the result), shows that non-degenerate instances with the dedicated CDS debtor property must have polynomial size rational solutions. We use this fact to prove Theorem 14.

Proof sketch of Theorem 14. Consider the graph D that has as its nodes the strongly connected components (SCCs) of $G_{I,\text{aux}}$, and has an arc from a node S to a node T if and only if there exists an arc in $G_{I,\text{aux}}$ that runs from a node in S to a node in T. It is clear that D is a directed acyclic graph.

We may show that we can find a rational clearing vector for $G_{I,\text{aux}}$ by finding rational clearing vectors of the separate SCCs of the system. However, both the assets and the liabilities of the nodes in a given SCC might depend on the contracts from outside the SCC that point into the SCC. Similarly, the liabilities of the nodes in the SCC might depend on arcs pointing from the SCC to external nodes. We may overcome this problem by including the outward-pointing arcs of an SCC into the subinstances that we aim to solve for, and by iterating over the SCCs according to the topological order of D: That is, we first find clearing vectors to the set $\mathcal{S}_1$ of SCCs that have no incoming arc in D. For such SCCs, the assets and liabilities of the nodes are not influenced by external arcs pointing into the SCC. We subsequently find clearing rates for the set of SCCs $\mathcal{S}_2$ that succeed $\mathcal{S}_1$ in the topological order defined by D. In general, we define $\mathcal{S}_j$ inductively as the set of SCCs that directly succeed $\mathcal{S}_{j-1}$ in the topological order defined by D, and we iteratively find clearing rates to the set of SCCs $\mathcal{S}_j$, given the clearing rates computed for $\mathcal{S}_1, \ldots, \mathcal{S}_{j-1}$, until we have obtained a clearing vector covering all nodes in the system. A crucial observation that motivates this approach is that the absence of any weakly switched cycle of $G_{I,\text{aux}}$ causes all SCCs to satisfy the dedicated CDS debtor property, and that therefore the clearing vector computation problem considered in each iteration lies in PPAD. At each iteration, we are thus guaranteed that there are rational recovery rates, and finding them requires solving a PPAD-complete problem. However, there are quite a few details required to turn the above ideas into a rigorous proof, and we defer these to the full version of this paper [18]. ◀

The procedure outlined in the proof of Theorem 14 requires solving a PPAD-complete problem in each iteration, and the number of such iterations is at most linear in the instance size. Since solving each of these problems in PPAD yields a rational solution of size polynomial in the input, one might be tempted to think that the procedure in its entirety is capable of finding a polynomial size rational solution for any financial system that has no weakly switched cycles. Unfortunately, the latter is not true: Observe that in each iteration of the procedure, the PPAD-complete problem instance that is solved, is actually constructed using the rational recovery rate vectors that are computed in the preceding iterations. The coefficients in the PPAD-complete problem instance that is to be solved in any given iteration, are thus polynomially sized in the output recovery rates of the previous iteration. Altogether, this means that the coefficient sizes potentially grow by a polynomial factor in each iteration, and that the final recovery rates output by the procedure are potentially of exponential size.

Indeed, there are examples of financial systems without weakly switched cycles for which the rational clearing vector has recovery rates that require an exponential number of bits to write down. A simple example is obtained by taking some of the gadgets in the reduction used in our FIXP-completeness result (Theorem 5). By taking a duplication gadget followed by a multiplication gadget that is connected to the two output nodes of the duplication gadget. We may then take multiple copies of these, and chain them together to form an acyclic financial system. If we now give the first node in the chain (i.e., the input node of the first duplication gadget) some small amount of positive external assets $c < 1$, this acyclic financial system essentially performs a sequence of successive squaring operations on the number c, under the unique clearing vector. The resulting recovery rates on the output nodes of the multiplication gadgets are then doubly exponentially small in magnitude, with respect to the number of squaring repetitions. Thus, the resulting clearing recovery rates require a number of bits that is exponential in the size of the financial system.

If one is willing to discard the complexity issues that arise from working with large-size rational numbers, it is possible to study the procedure in the proof of Theorem 14 in the Blum-Shub-Smale model of computation. Under this computational model, any real number takes one unit of space to store, regardless of its size. Moreover, standard arithmetic operations are assumed to take unit time.[1] The proof of Theorem 14 then implies that when one has oracle access to PPAD, it is possible to find rational clearing vectors in polynomial time under this model of computation. The class of problems polynomial time solvable under the Blum-Shub-Smale model is commonly denoted by $\mathsf{P}_{\mathbb{R}}$. Hence, we obtain the following corollary.

▶ **Corollary 17.** *The exact computation version of* CDS-CLEARING*, restricted to instances without weakly switched cycles, is in the complexity class* $\mathsf{P}_{\mathbb{R}}^{\mathsf{PPAD}}$.

6 Conclusions

In this paper we study two questions of significance related to the systemic risk in financial networks with CDSes, a widely used and potentially disruptive class of financial derivatives. Firstly, we settle the computational complexity of computing strong approximations of each bank's exposure to systemic risk, arguably the right notion of approximation of interest to industry – a conceptual point so far overlooked in the literature. We show that this problem is FIXP-complete. Secondly, we initiate the study of the rational fragment of FIXP by studying the conditions under which rational solutions for CDS-CLEARING exist. Our results here are not conclusive in that there is a gap between our necessary and sufficient conditions, the cycles which involve both switched on and switched off nodes being not fully understood. [2] We conjecture that for any network with a weakly switched cycle there exist rational values for assets and liabilities that lead to irrational solutions; however, our arguments and scheme cannot be easily generalised to those instances. We leave providing a full characterisation as an open problem.

Further research directions are suggested by our work. It would be interesting to study whether Corollary 17's connection between CDS-CLEARING, PPAD, and $\mathsf{P}_{\mathbb{R}}$ (i.e. polynomial time under the Blum-Shub-Smale model of computation [5]) holds more generally for the entire rational subset of problems in FIXP. Furthermore, it is interesting to pursue finding

[1] For a formal and more accurate definition of the Blum-Shub-Smale model, see the book [5].

[2] We regard the *simplicity* condition we made (i.e., about off-cycle paths between cycle nodes) as a technicality, which is less interesting and likely somewhat easier to deal with.

polynomial-time *constant* approximation algorithms of clearing recovery rate vectors: Also from an applied point of view, achieving a good approximation factor here (say with an additive approximation term of 1/100) might yield solutions that are useful in most practical circumstances and could be considered acceptable by financial institutions. We note here that a 1/2-strong approximation is easy to compute (a recovery rate vector of only 1/2s would indeed suffice).

References

1. Euro-parliament bans “naked” credit default swaps. EUBusiness. `https://www.eubusiness.com/news-eu/finance-economy-cds.dij`.
2. Lasting effects: The global economic recovery 10 years after the crisis. IMFBlogs. URL: `https://blogs.imf.org/2018/10/03/lasting-effects-the-global-economic-recovery-10-years-after-the-crisis/`.
3. Daron Acemoglu, Asuman Ozdaglar, and Alireza Tahbaz-Salehi. Systemic risk and stability in financial networks. *American Economic Review*, 105(2):564–608, 2015.
4. Nils Bertschinger, Martin Hoefer, and Daniel Schmand. Strategic payments in financial networks. In Thomas Vidick, editor, *11th Innovations in Theoretical Computer Science Conference, ITCS 2020, January 12-14, 2020, Seattle, Washington, USA*, volume 151 of *LIPIcs*, pages 46:1–46:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. `doi:10.4230/LIPIcs.ITCS.2020.46`.
5. Lenore Blum, Felipe Cucker, Michael Shub, and Steve Smale. *Complexity and real computation*. Springer Science & Business Media, 1998.
6. Rodrigo Cifuentes, Gianluigi Ferrucci, and Hyun Song Shin. Liquidity risk and contagion. *Journal of the European Economic association*, 3(2-3):556–566, 2005.
7. Larry Eisenberg and Thomas H Noe. Systemic risk in financial systems. *Management Science*, 47(2):236–249, 2001.
8. Matthew Elliott, Benjamin Golub, and Matthew O Jackson. Financial networks and contagion. *American Economic Review*, 104(10):3115–53, 2014.
9. Kousha Etessami and Mihalis Yannakakis. On the complexity of nash equilibria and other fixed points. *SIAM Journal on Computing*, 39(6):2531–2597, 2010.
10. Aris Filos-Ratsikas, Yiannis Giannakopoulos, Alexandros Hollender, Philip Lazos, and Diogo Poças. On the complexity of equilibrium computation in first-price auctions. *arXiv preprint arXiv:2103.03238*, 2021.
11. Aris Filos-Ratsikas, Kristoffer Arnsfelt Hansen, Kasper Høgh, and Alexandros Hollender. Fixp-membership via convex optimization: Games, cakes, and markets. *arXiv preprint*, 2021. `arXiv:2111.06878`.
12. Jugal Garg, Ruta Mehta, Vijay V. Vazirani, and Sadra Yazdanbod. Settling the complexity of leontief and PLC exchange markets under exact and approximate equilibria. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 890–901. ACM, 2017. `doi:10.1145/3055399.3055474`.
13. Paul Glasserman and H Peyton Young. How likely is contagion in financial networks? *Journal of Banking & Finance*, 50:383–399, 2015.
14. Paul W Goldberg and Alexandros Hollender. The hairy ball problem is ppad-complete. *Journal of Computer and System Sciences*, 2021.
15. Sebastian Heise and Reimer Kühn. Derivatives and credit contagion in interconnected networks. *The European Physical Journal B*, 85(4):1–19, 2012.
16. Brett Hemenway and Sanjeev Khanna. Sensitivity and computational complexity in financial networks. *Algorithmic Finance*, 5(3-4):95–110, 2016.
17. Daning Hu, J Leon Zhao, Zhimin Hua, and Michael CS Wong. Network-based modeling and analysis of systemic risk in banking systems. *MIS quarterly*, pages 1269–1291, 2012.

18 Stavros D. Ioannidis, Bart de Keijzer, and Carmine Ventre. Strong approximations and irrationality in financial networks with financial derivatives. *CoRR*, abs/2109.06608, 2021. `arXiv:2109.06608`.

19 Pál András Papp and Roger Wattenhofer. Default ambiguity: finding the best solution to the clearing problem. *arXiv preprint*, 2020. `arXiv:2002.07741`.

20 Pál András Papp and Roger Wattenhofer. Network-aware strategies in financial systems. In Artur Czumaj, Anuj Dawar, and Emanuela Merelli, editors, *47th International Colloquium on Automata, Languages, and Programming, ICALP 2020, July 8-11, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 168 of *LIPIcs*, pages 91:1–91:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. `doi:10.4230/LIPIcs.ICALP.2020.91`.

21 Leonard CG Rogers and Luitgard AM Veraart. Failure and rescue in an interbank network. *Management Science*, 59(4):882–898, 2013.

22 Steffen Schuldenzucker, Sven Seuken, and Stefano Battiston. Clearing payments in financial networks with credit default swaps. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pages 759–759, 2016.

23 Steffen Schuldenzucker, Sven Seuken, and Stefano Battiston. Finding clearing payments in financial networks with credit default swaps is PPAD-complete. *LIPIcs: Leibniz International Proceedings in Informatics*, 67, 2017.