Fisher Markets with Approximately Optimal Bundles and the Need for a PCP Theorem for PPAD

Argyrios Deligkas
argyrios.deligkas@rhul.ac.uk
Royal Holloway
Egham, UK

John Fearnley
john.fearnley@liverpool.ac.uk
University of Liverpool
Liverpool, UK

Alexandros Hollender
alexandros.hollender@cs.ox.ac.uk
University of Oxford
Oxford, UK

Themistoklis Melissourgos
themistoklis.melissourgos@essex.ac.uk
University of Essex
Colchester, UK

ABSTRACT

We study the problem of computing a competitive equilibrium with approximately optimal bundles in Fisher markets with separable piecewise-linear concave (SPLC) utility functions, meaning that every buyer receives a $(1-\delta)$-optimal bundle, instead of a perfectly optimal one. We establish the first intractability result for the problem by showing that it is PPAD-hard for some constant $\delta > 0$, assuming the PCP-for-PPAD conjecture. This hardness result holds even if all buyers have identical budgets (competitive equilibrium with equal incomes), linear capped utilities, and even if we also allow ε-approximate clearing instead of perfect clearing, for any constant $\varepsilon < 1/9$. Importantly, we show that the PCP-for-PPAD conjecture is in fact required to show hardness for constant δ: showing PPAD-hardness for finding such approximate market equilibria in a broad class of markets encompassing those generated by our hardness result would prove the conjecture. This is the first natural problem where the conjecture is provably required to establish hardness for it.

CCS CONCEPTS

• **Theory of computation → Problems, reductions and completeness**; **Market equilibria**; **Exact and approximate computation of equilibria**.

KEYWORDS

PPAD, market equilibria, PCP, inapproximability

1 INTRODUCTION

We study Fisher markets [4], which are a fundamental model of a market in which a set of buyers are interested in purchasing a set of goods. Unlike more general market models, buyers are not endowed with any goods, but instead come to the market with money and they spend this budget to buy goods in a way that maximizes their utility. Importantly, the buyers do not have any utility for money, but only for goods.

For Fisher markets, it is known that when the utility functions satisfy some standard sufficiency conditions, then a *competitive equilibrium*, or *market equilibrium*, is guaranteed to exist. A market equilibrium consists of prices for each of the goods and an allocation of the goods to the buyers such that (a) the market *clears*, i.e., for every good its supply is exactly equal to its demand, and (b) every buyer buys an *optimal bundle*, i.e., a set of goods that maximizes their utility under their budget constraint at the current prices.

The computational complexity of finding market equilibria. The problem of computing a market equilibrium in a Fisher market has received a lot of attention in prior work. When buyers have linear utility functions, a market equilibrium can always be computed in polynomial time [21, 36, 44]. However, the problem becomes intractable as soon as one considers *additive separable piecewise-linear concave* (SPLC) utilities, which are a generalization of linear utilities. A buyer with an SPLC utility function has a piecewise-linear concave utility for each good, and their utility for a bundle of goods is simply the sum of their utilities for each of the individual goods. Finding an equilibrium in a Fisher market with SPLC utilities is known to be PPAD-complete, i.e., the problem does not admit a polynomial-time algorithm, unless PPAD = P [9, 15, 42].

In fact, this hardness result continues to hold even for *ε-approximate* market equilibria. In an ε-approximate market equilibrium, the clearing constraint is relaxed and only requires every good to ε-clear, meaning that the discrepancy between the supply and the demand for any good is at most ε. Chen and Teng [9] and Vazirani and Yannakakis [42] showed that it is PPAD-complete to find an ε-approximate market equilibrium when ε is inversely polynomial in the size of the market, i.e., the number of buyers and the number of goods, and more recently, Deligkas et al. [15] proved PPAD-completeness even for *constant* $\varepsilon < 1/11$, ruling out the existence of a polynomial-time approximation scheme (PTAS) for ε-approximate market equilibria.

Approximately optimal bundles. The computational complexity lower bounds have so far only considered approximations that relax the clearing constraint of the market, while still insisting that each buyer should receive an optimal allocation. It is natural to ask what happens if the buyers' optimality is relaxed instead. In that case, we allow every buyer to receive a $(1-\delta)$-optimal bundle, that is, a bundle that gives them utility that is at least a $(1-\delta)$-fraction of the utility of an optimal bundle. This is arguably a more natural relaxation of the problem, since it only requires the buyers be willing to accept slightly sub-optimal bundles, whereas ε-approximate clearing requires the market to find a way to deal with the mismatch between supply and demand (e.g., by destroying an ε-fraction of some goods).

For approximately optimal bundles, [27] recently obtained a positive result. They proved that for a large family of utility functions, which include SPLC utilities, any allocation that maximizes Nash welfare can be coupled with appropriate prices so that the market exactly clears and every buyer receives a $1/2$-optimal bundle. The Nash welfare of an allocation is the product of the utilities of the buyers under this allocation, and it is known that such an allocation can be computed in polynomial time using convex optimization. So, in particular, this shows that it is possible to efficiently compute a market equilibrium with $(1-\delta)$-optimal bundles when $\delta = 1/2$. However, so far no lower bounds under this relaxation have been established. A very natural question is thus: does this problem admit a PTAS?

The PCP-for-PPAD conjecture. Unrelated to Fisher markets, there has been a line of work that has studied the PCP-for-PPAD conjecture. The ε-GCircuit problem is a well-known PPAD-complete problem that asks us to find an ε-approximate satisfying assignment to a fixed-point problem defined by arithmetic gates [8], and this problem is known to be PPAD-hard even when ε is constant [17, 40].

We can further weaken this to the (ε, δ)-GCircuit problem, which requires only that a $(1-\delta)$-fraction of the gates are ε-approximately satisfied, and so a δ-fraction of the constraints are allowed to be broken. The so-far unproven PCP-for-PPAD conjecture states that there exist small constants ε and δ such that (ε, δ)-GCircuit is PPAD-hard. This conjecture was first proposed by Babichenko et al. [2], where they hoped that it would be useful in showing that finding an approximate Nash equilibrium in a bimatrix game requires quasi-polynomial time. In the end, however, the conjecture was not used for this task, since the lower bound for bimatrix games was shown via alternate means [39].

The PCP-for-PPAD conjecture has found other uses, however, and it has recently been used to prove conditional hardness for computing approximate non-perfect Markov stationary coarse correlated equilibria in multiplayer games [38] and stationary CCE in two-player stochastic games [13].

Both of these hardness results show PPAD-hardness assuming that the PCP-for-PPAD conjecture holds. As such, they are open to the straightforward criticism that the PCP-for-PPAD conjecture may simply be false, meaning that no lower bound is actually shown. Another criticism is that the PCP-for-PPAD conjecture *may not be necessary at all* to show hardness for these problems, and there may be a direct PPAD-hardness reduction that we have so far missed, as was the case for bimatrix games.

1.1 Our Contribution

This paper provides the first intractability results for market equilibria with approximately optimal bundles, i.e., when the buyers are happy to receive $(1-\delta)$-optimal bundles. Our main hardness result is the following:

- Assuming that the PCP-for-PPAD conjecture holds, there exists a suitably small constant $\delta > 0$ such that it is PPAD-hard to find a market equilibrium with $(1-\delta)$-optimal bundles.

Furthermore, as a byproduct of the reduction establishing this result, we also obtain the following two unconditional PPAD-hardness results:

- If $\delta > 0$ is inverse-polynomial (i.e., given in the input in unary), then it is PPAD-hard to find a market equilibrium with $(1-\delta)$-optimal bundles.
- If buyers are not allowed to spend any money on goods for which they have zero utility, then there exists a constant $\delta > 0$ such that it is PPAD-hard to find a market equilibrium with $(1-\delta)$-optimal bundles.

Importantly, these three results continue to hold if we also *simultaneously* relax the equilibrium notion by allowing ε-approximate clearing, instead of perfect clearing. In fact, they hold for any constant $\varepsilon < 1/9$. In particular, our results improve on the state of the art even for the setting with perfectly optimal bundles, since PPAD-hardness was previously only known for $\varepsilon < 1/11$ [15].

Furthermore, our hardness results apply to a fairly simple class of markets. In particular, all buyers have *linear capped* utilities and identical budgets. This latter restriction corresponds to the setting of competitive equilibrium with equal incomes (CEEI), which is known to have desirable fairness properties. To the best of our knowledge, these are the first intractability results for CEEI, even for exact equilibria.[1]

By itself, our main hardness result for constant δ is open to the same criticisms as the other results that depend on the PCP-for-PPAD conjecture: perhaps the PCP-for-PPAD conjecture is false, or perhaps the hardness results can be shown without it. However, our second and perhaps more interesting result shows that this is not the case. Namely, we prove the following:

- If it is PPAD-hard to find a market equilibrium with $(1-\delta)$-optimal bundles for some constant $\delta > 0$ in a broad class of Fisher markets encompassing those generated by our hardness result, then the PCP-for-PPAD conjecture holds.

This shows that resolving the PCP-for-PPAD conjecture is *required* to resolve the computational complexity of Fisher markets with approximately optimal bundles: the problem is hard for constant δ if and only if the PCP-for-PPAD conjecture holds. We note that Fisher markets are the first *natural* problem for which this property is known to hold. By this, we mean that the problem does not have a "PCP-like" structure in which a δ-fraction of the constraints are allowed to fail. Indeed, the only other problem whose hardness is known to be equivalent to the PCP-for-PPAD conjecture is the (ε, δ)-weak approximate Nash equilibrium problem for polymatrix games [2]. However, this problem allows a δ-fraction of the players to not be in ε-equilibrium, namely to play an arbitrarily bad response. On the other hand, a market equilibrium with approximately optimal bundles requires that *all* goods exactly clear, and that *all* buyers receive a $(1-\delta)$-approximate bundle.

We argue that this revitalizes the PCP-for-PPAD conjecture, and provides a compelling need for its proof or disproof. Before our work, the conjecture had only been used to show conditional hardness results, and for those works the criticism that perhaps the PCP-for-PPAD conjecture could be bypassed always applied. Since we now have a natural problem that is hard if and only if the conjecture holds, resolving the conjecture now appears necessary.

[1] Some works [2, 37, 40] prove PPAD-hardness for approximate CEEI in a setting with indivisible goods, which should not be confused with our divisible setting.

1.2 Further related work

The computation of market equilibria, for both exact and approximate equilibria, has received significant attention over the years. For Fisher markets, Vazirani and Yannakakis [42] and Chen and Teng [9] have established PPAD-hardness for SPLC utilities albeit for a sub-constant ε. On the other hand, polynomial-time algorithms were derived for the cases where the utility functions of the buyers are linear [21, 36, 44], homogeneous [23], or weak gross substitutes [11].

Furthermore, when the number of goods is constant Kakade et al. [33] gave a PTAS while Devanur and Kannan [20] gave a polynomial-time algorithm for exact equilibria. In fact, the algorithm of [20] also works when the number of buyers is constant in the SPLC utility setting. For non-separable PLC utilities Garg et al. [26] derived a fixed parameter approximation scheme that has the number of buyers as a parameter.

Market equilibria where buyers get approximately optimal bundles (both with and without approximate clearing) have been studied in the context of obtaining approximation algorithms for various classes of utilities or variants of the problem [6, 10–12, 18, 24, 31, 34].

Matching markets are an interesting variant of general Fisher markets. Alaei et al. [1] designed a polynomial-time algorithm for markets with a constant number of goods or buyers, while Vazirani and Yannakakis [43] derived a polynomial algorithm when the buyers have dichotomous utilities. For one-sided matching markets, the most famous problem is the Hylland-Zeckhauser market, for which the existence of an equilibrium was initially established in [29] and was recently simplified by Braverman [5]. Even more recently, Chen et al. [7] have established PPAD-completeness for the problem.

There has also been interest in Fisher markets with additional constraints. Birnbaum et al. [3], Devanur [19], and Vazirani [41] considered the case where the utilities of the buyers depend on prices of goods through spending constraints. Jalota et al. [32] considered additional linear constraints that include matching markets, and they gave a tâtonnement process which was found to converge to a market equilibrium in experiments.

The Pure-Circuit problem was recently introduced in [17] , where it was used to prove strong, improved, PPAD-hardness results for a variety of problems related mainly to approximate Nash equilibria. Since then it was further used in [14] to prove tight PPAD-hardness for approximate Nash equilibria in graphical games, in [30] and in [22] to prove stronger PPAD-hardness in the problem of clearing financial networks, and in [28] to derive improved inapproximability PPAD-hardness for Markov equilibria in stochastic games.

2 PRELIMINARIES

2.1 Fisher Markets

Fisher markets. A Fisher market is given by a tuple $(G, B, (e_i)_{i\in B}, (u_i)_{i\in B})$, where:

- G is a set of (divisible) goods. Without loss of generality, we assume that there is one unit of each good available.[2]
- B is a set of buyers.
- For every $i \in B$, $e_i > 0$ is the budget of buyer i.
- For every $i \in B$, $u_i : \mathbb{R}^{|G|}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is the utility function of buyer i. For any allocation $x_i \in \mathbb{R}^{|G|}_{\geq 0}$ of goods to buyer i (where $x_{i,j} \geq 0$ denotes the amount of good j allocated to buyer i), $u_i(x_i)$ denotes the utility derived by the buyer. We assume that the utility functions are separable piecewise-linear concave (SPLC), meaning that $u_i(x_i)$ can be written as $\sum_{j\in G} u_{i,j}(x_{i,j})$, where each $u_{i,j} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ satisfies
  (1) $u_{i,j}(0) = 0$,
  (2) $u_{i,j}$ is continuous and piecewise-linear,
  (3) $u_{i,j}$ is concave and non-decreasing.

  In particular, we represent each piecewise-linear concave utility function $u_{i,j}$ as a sequence $\langle(s_{i,j,1}, \ell_{i,j,1}), (s_{i,j,2}, \ell_{i,j,2}), \ldots, (s_{i,j,m_{i,j}}, \ell_{i,j,m_{i,j}})\rangle$ where for each k we have that $s_{i,j,k}$ gives a slope of a linear piece and $\ell_{i,j,k}$ gives the length of that piece, and where $m_{i,j}$ is the number of pieces used by $u_{i,j}$. The length of the last piece can be infinite. So to compute $u_{i,j}(x)$ we find the largest value a such that $\sum_{k=1}^{a} \ell_{i,j,k} \leq x$ and then we set

  $$u_{i,j}(x) = \sum_{k=1}^{a}(s_{i,j,k} \cdot \ell_{i,j,k}) + (x - \sum_{k=1}^{a} \ell_{i,j,k}) \cdot s_{i,j,a+1}.$$

  The concavity of the function implies that $s_{i,j,k} \geq s_{i,j,k+1}$ for all k, and we can without loss of generality assume that $s_{i,j,k} > s_{i,j,k+1}$ for all a, since we can simply merge adjacent pieces with identical slopes.

  A notable special case of SPLC utilities are *linear capped* utilities, where $u_{i,j}$ is of the form $u_{i,j}(x_{i,j}) = \min\{a \cdot x_{i,j}, b\}$ for some $a \in \mathbb{R}_{\geq 0}$ and $b \in \mathbb{R}_{\geq 0} \cup \{+\infty\}$. Our hardness result will apply to these utilities.

Approximately optimal bundles. Given $\delta \in [0, 1]$ and a price vector $p \in \mathbb{R}^{|G|}_{\geq 0}$, where p_j denotes the price of good j, the set of $(1 - \delta)$-optimal bundles for buyer i, denoted $\mathrm{OPT}^{\delta}_i(p) \subseteq \mathbb{R}^{|G|}_{\geq 0}$, is the set of all $(1-\delta)$-optimal solutions of the following optimization problem:

$$\begin{aligned} \max \quad & u_i(x_i) \\ \text{s.t.} \quad & \sum_{j\in G} p_j x_{i,j} \leq e_i \\ & x_{i,j} \geq 0 \quad \forall j \in G. \end{aligned} \tag{1}$$

In other words, letting $F_i(p) \subseteq \mathbb{R}^{|G|}_{\geq 0}$ denote the feasible set of optimization problem (1), and $U^*_i := \max_{x_i\in F_i(p)} u_i(x_i) \in \mathbb{R}_{\geq 0} \cup \{+\infty\}$ its optimal value, we can write

$$\mathrm{OPT}^{\delta}_i(p) := \{x_i \in F_i(p) : u_i(x_i) \geq (1 - \delta) \cdot U^*_i\}.$$

For $\delta = 0$, this simply corresponds to the set of all optimal solutions to (1). Note that it is possible for $\mathrm{OPT}^{\delta}_i(p)$ to be empty. Indeed, it is possible to have $U^*_i = \infty$, when some good has price zero and the buyer is never satiated with that good.

For SPLC utilities, we will often be interested in the *bang-per-buck* of a particular utility-function segment. The bang-per-buck of the kth segment of $x_{i,j}$ under the price vector p is defined as $s_{i,j,k}/p_j$.

[2]This can be achieved by a simple normalization, and it simplifies the expression for the clearing constraint below.

Approximate competitive equilibrium. For any $\varepsilon, \delta \in [0,1]$, an (ε, δ)-approximate market equilibrium is a price vector p and an allocation vector $x = (x_i)_{i \in B}$ satisfying the following conditions:

(1) For each buyer i, x_i is a $(1-\delta)$-optimal bundle at prices p, i.e., $x_i \in \mathrm{OPT}_i^{\delta}(p)$.
(2) For each good j, the market clears approximately up to ε units of good, i.e.,

$$\left| \sum_{i \in B} x_{i,j} - 1 \right| \leq \varepsilon.$$

When $\varepsilon = \delta = 0$, this corresponds to an exact market equilibrium.

Existence of equilibria. The following condition is sufficient to guarantee the existence[3] of a market equilibrium [35, 42]:

> **Sufficient Condition:** For every buyer $i \in B$, there exists a good $j \in G$ such that $u_{i,j}$ is a strictly increasing function (i.e., buyer i is never satiated with good j).

Computational problem. Let $\varepsilon, \delta \in [0,1]$. The computational problem of computing an (ε, δ)-approximate market equilibrium is defined as follows:

> **Input:** A Fisher market $(G, B, (e_i)_{i \in B}, (u_i)_{i \in B})$ with SPLC utilities satisfying the sufficient condition for the existence of equilibria. For each $i \in B$ and $j \in G$, $u_{i,j}$ is explicitly described in the input, i.e., for each linear affine piece we are given the positions and values at its endpoints.
>
> **Output:** An (ε, δ)-approximate market equilibrium (p, x).

Given (p, x), the equilibrium conditions can be verified in polynomial time, because, for SPLC utilities, the optimization problem (1) used to define $\mathrm{OPT}_i^{\delta}(p)$ can be solved in polynomial time using a simple greedy approach; see, e.g., [25]. Together with the existence of solutions guaranteed by the sufficient condition, this puts the problem in the complexity class TFNP of total NP search problems. Prior work [42] has shown that the problem lies in the subclass PPAD of TFNP, even for $\varepsilon = \delta = 0$. In particular, exact rational solutions are guaranteed to exist. The problem is known to be PPAD-complete when $\delta = 0$ and $\varepsilon < 1/11$ [9, 15, 42]. No hardness result is known for any constant $\delta > 0$, even when $\varepsilon = 0$.

2.2 Pure-Circuit, GCircuit, and the PCP conjecture for PPAD

In this section, we formally present the PCP conjecture for PPAD and two new equivalent formulations. The original formulation is in terms of the generalized circuit problem. The first new equivalent formulation is in terms of the Pure-Circuit problem and it makes it easier to prove hardness results assuming the conjecture. The other new formulation makes it easier to prove that the conjecture is needed to prove some hardness result. We prove that the three formulations are equivalent and so can be used interchangeably.

[3] In fact, existence of a market equilibrium is guaranteed even without this condition. However, the condition guarantees the very desirable property that any market equilibrium is Pareto optimal, see e.g. [27].

2.2.1 The Pure-Circuit *Problem.* An instance of the Pure-Circuit problem is given by a set of nodes (or *variables*) $V = [n]$ and a set C of gate-constraints (or just *gates*). Each gate $g \in C$ is of the form $g = (T, u, v, w)$ where $u, v, w \in V$ are distinct nodes, and $T \in \{\mathsf{NAND}, \mathsf{PURIFY}\}$ is the type of the gate, with the following interpretation.

- If $T = \mathsf{NAND}$, then u and v are the inputs of the gate, and w is its output.
- If $T = \mathsf{PURIFY}$, then u is the input of the gate, and v and w are its outputs.

We require that each node is the output of exactly one gate. A node can be used as an input by multiple gates.

A solution to instance (V, C) is an assignment $\mathbf{x} : V \to \{0, 1, \perp\}$ that satisfies all the gates (see Figure 1), i.e., for each gate $g = (T, u, v, w) \in C$ we have the following.

- If $T = \mathsf{NAND}$ in $g = (T, u, v, w)$, then $\mathbf{x}$ satisfies

$$\mathbf{x}[u] = \mathbf{x}[v] = 1 \implies \mathbf{x}[w] = 0$$
$$(\mathbf{x}[u] = 0) \lor (\mathbf{x}[v] = 0) \implies \mathbf{x}[w] = 1$$

- If $T = \mathsf{PURIFY}$, then $\mathbf{x}$ satisfies

$$\{\mathbf{x}[v], \mathbf{x}[w]\} \cap \{0, 1\} \neq \emptyset$$
$$\mathbf{x}[u] \in \{0, 1\} \implies \mathbf{x}[v] = \mathbf{x}[w] = \mathbf{x}[u].$$

Theorem 2.1 ([17]). Pure-Circuit *is* PPAD*-complete, even when every node is the input to exactly one gate.*

For any constant $\delta \in [0,1]$, the δ-Pure-Circuit problem is defined as: given a Pure-Circuit instance (V, C), find an assignment $\mathbf{x} \in [0,1]^{|V|}$ such that at least a $(1-\delta)$ fraction of the gates in C are satisfied. The PCP-for-PPAD conjecture can be formulated as follows.

PCP-for-PPAD Conjecture (Pure-Circuit version)**.** *There exists a constant $\delta > 0$ such that δ-*Pure-Circuit *is* PPAD*-hard, even when every node is the input to exactly one gate.*

Furthermore, as we show in Theorem 2.3, removing the restriction on every node being the input to exactly one gate does not change the conjecture, i.e., the versions with and without that restriction are equivalent.

The conjecture was originally formulated in terms of the generalized circuit problem, GCircuit, which we define next. In Theorem 2.3 we show that these formulations are equivalent. We expect the new simplified formulation in terms of Pure-Circuit to be useful for future work.

2.2.2 The GCircuit *Problem.* A generalized circuit is defined in the following way.

Definition 2.2 (Generalized Circuit [8]). A generalized circuit is a tuple (V, T), where V is a set of nodes, and T is a set of gates. Each gate $t \in T$ is a five-tuple (G, u, v, w, c), where G is a gate type from the set $\{G_c, G_{\times c}, G_=, G_+, G_-, G_<, G_\lor, G_\land, G_\neg\}$, $u, v \in V \cup \{\mathsf{nil}\}$ are input variables, $w \in V$ is an output variable, and $c \in [0,1] \cup \{\mathsf{nil}\}$ is a rational constant.

The following requirements must be satisfied for each gate $(G, u, v, w, c) \in T$.

u	v	w
1	1	0
0	$\{0, 1, \perp\}$	1
$\{0, 1, \perp\}$	0	1
Else		$\{0, 1, \perp\}$

NAND gate

u	v	w
0	0	0
1	1	1
$\perp$	At least one output in $\{0, 1\}$	

PURIFY gate

Figure 1: The truth tables of the two gates of Pure-Circuit.

- G_c gates take no input variables and uses a constant in $[0, 1]$. So $u = v = \mathsf{nil}$ and $c \in [0, 1]$ whenever $G = G_c$.
- $G_{\times c}$ gates take one input variable and a constant. So $u \in V$, $v = \mathsf{nil}$, and $c \in [0, 1]$ whenever $G = G_{\times c}$.
- $G_=$ and $G_\neg$ gates take one input variable and do not use a constant. So $u \in V$, $v = c = \mathsf{nil}$, whenever $G \in \{G_=, G_\neg\}$.
- All other gates take two input variables and do not use a constant. So $u \in V$, $v \in V$, and $c = \mathsf{nil}$ whenever $G \notin \{G_c, G_{\times c}, G_=, G_\neg\}$.
- Every variable in V is the output variable for exactly one gate. More formally, for each variable $w \in V$, there is exactly one gate $t \in T$ such that $t = (G, u, v, w, c)$.

For constants $\varepsilon, \delta \in [0, 1]$, the (ε, δ)-GCircuit problem is defined as follows. Given a generalized circuit (V, T), find a vector $\mathbf{x} \in [0, 1]^{|V|}$ such that a $(1 - \delta)$ fraction of the gates in T satisfy the constraints in Figure 2.

The PCP-for-PPAD conjecture was originally introduced by [2] and formulated as follows.[4]

PCP-for-PPAD Conjecture (GCircuit version)**.** *There exist constants $\varepsilon > 0$ and $\delta > 0$ such that (ε, δ)-GCircuit is PPAD-hard.*

As mentioned above, in Theorem 2.3 we show that this is equivalent to our formulation in terms of Pure-Circuit presented earlier. For the purpose of proving that some problem is PPAD-hard assuming the PCP-for-PPAD conjecture, it is easier to use the Pure-Circuit version of the conjecture, since the gates are easier to implement. However, for the purpose of showing that the PCP-for-PPAD conjecture is *necessary* to show PPAD-hardness for some problem of interest, it is more convenient to have a formulation of the conjecture that uses a more expressive problem. Next, we present a third equivalent version of the conjecture in terms of a generalization of the GCircuit problem.

2.2.3 The GCircuit+ *problem.* When we reduce to GCircuit, it will be more convenient to express our circuits using more general gates. Specifically, we define the (ε, δ)-GCircuit+ problem to be the same as the (ε, δ)-GCircuit problem but with the following modifications.

- In this version of the problem, each variable $v \in V$ comes equipped with a rational lower bound v_l and a rational upper bound v_u. Then, instead of seeking a solution $\mathbf{x} \in [0, 1]^{|V|}$, we seek a solution $\mathbf{x}$ where each x_v is constrained to lie in the range $[v_l, v_u]$.
- We extend the $G_{\times c}$ gate to permit c to be an arbitrary rational rather than restricting c to lie in $[0, 1]$.
- We replace the G_+, G_-, and $G_{\times c}$ gates with versions that do not take a min or max with 1 or 0, but instead use the bounds for the output variable. That is, we use the following definitions.
  - The $(G_+, u, v, w, \mathsf{nil})$ gate now requires that
  $$\mathbf{x}[w] = \max(\min(\mathbf{x}[u] + \mathbf{x}[v], w_u), w_l) \pm \varepsilon$$
  - The $(G_-, u, v, w, \mathsf{nil})$ gate now requires that
  $$\mathbf{x}[w] = \max(\min(\mathbf{x}[u] - \mathbf{x}[v], w_u), w_l) \pm \varepsilon$$
  - The $(G_{\times c}, u, \mathsf{nil}, w, c)$ gate now requires that
  $$\mathbf{x}[w] = \max(\min(\mathbf{x}[u] \cdot c, w_u), w_l) \pm \varepsilon$$

  Note that we now need to bound these operations from above and below, since we allow negative variables in GCircuit+, and we also allow negative multiplicands in our $G_{\times c}$ gates.
- We introduce explicit gates that allow us to take a min or max with a constant rational value c, and we do not require $c \in [0, 1]$.
  - A $(G_{\max}, u, \mathsf{nil}, w, c)$ gate requires that
  $$\mathbf{x}[w] = \max(\mathbf{x}[u], c) \pm \varepsilon$$
  - A $(G_{\min}, u, \mathsf{nil}, w, c)$ gate requires that
  $$\mathbf{x}[w] = \min(\mathbf{x}[u], c) \pm \varepsilon$$
- For the comparison gate $G_<$, we require that the output node has lower bound 0 and upper bound 1. For the logical gates $G_\vee$, $G_\wedge$, and $G_\neg$, we require that the input and output nodes have lower bound 0 and upper bound 1. For the constant gate G_c, we require that the constant c lies between the lower and upper bound of the output node. Finally, for the equality gate $G_=$, we require that the bounds on the input and output are the same.

So long as the bounds on all of the variables are constant, and all $G_{\times c}$ gates also multiply by constants, it is straightforward to reduce this new version of the problem to the original (see Theorem 2.3). As a result, we obtain the following third equivalent formulation of the conjecture.

PCP-for-PPAD Conjecture (GCircuit+ version)**.** *There exist constants $\varepsilon > 0$ and $\delta > 0$ such that (ε, δ)-GCircuit+ is PPAD-hard.*

[4] Babichenko et al. [2] actually give a more specific conjecture regarding the existence of a quasi-linear reduction from End-of-the-Line, which is the canonical PPAD-complete problem, to (ε, δ)-GCircuit. However, all subsequent works use the weaker conjecture that we also use in this work [13, 38, 39].

Gate	Constraint
$(G_c, \mathsf{nil}, \mathsf{nil}, w, c)$	$\mathbf{x}[w] = c \pm \varepsilon$
$(G_{\times c}, u, \mathsf{nil}, w, c)$	$\mathbf{x}[w] = \min(\mathbf{x}[u] \cdot c, 1) \pm \varepsilon$
$(G_=, u, \mathsf{nil}, w, \mathsf{nil})$	$\mathbf{x}[w] = \mathbf{x}[u] \pm \varepsilon$
$(G_+, u, v, w, \mathsf{nil})$	$\mathbf{x}[w] = \min(\mathbf{x}[u] + \mathbf{x}[v], 1) \pm \varepsilon$
$(G_-, u, v, w, \mathsf{nil})$	$\mathbf{x}[w] = \max(\mathbf{x}[u] - \mathbf{x}[v], 0) \pm \varepsilon$
$(G_<, u, v, w, \mathsf{nil})$	$\mathbf{x}[w] = \begin{cases} 1 \pm \varepsilon & \text{if } \mathbf{x}[u] < \mathbf{x}[v] - \varepsilon \\ 0 \pm \varepsilon & \text{if } \mathbf{x}[u] > \mathbf{x}[v] + \varepsilon \end{cases}$
$(G_\vee, u, v, w, \mathsf{nil})$	$\mathbf{x}[w] = \begin{cases} 1 \pm \varepsilon & \text{if } \mathbf{x}[u] \geq 1 - \varepsilon \text{ or } \mathbf{x}[v] \geq 1 - \varepsilon \\ 0 \pm \varepsilon & \text{if } \mathbf{x}[u] \leq \varepsilon \text{ and } \mathbf{x}[v] \leq \varepsilon \end{cases}$
$(G_\wedge, u, v, w, \mathsf{nil})$	$\mathbf{x}[w] = \begin{cases} 1 \pm \varepsilon & \text{if } \mathbf{x}[u] \geq 1 - \varepsilon \text{ and } \mathbf{x}[v] \geq 1 - \varepsilon \\ 0 \pm \varepsilon & \text{if } \mathbf{x}[u] \leq \varepsilon \text{ or } \mathbf{x}[v] \leq \varepsilon \end{cases}$
$(G_\neg, u, \mathsf{nil}, w, \mathsf{nil})$	$\mathbf{x}[w] = \begin{cases} 1 \pm \varepsilon & \text{if } \mathbf{x}[u] \leq \varepsilon \\ 0 \pm \varepsilon & \text{if } \mathbf{x}[u] \geq 1 - \varepsilon \end{cases}$

Figure 2: Constraints that need to be satisfied by a $(1-\delta)$ fraction of the gates in a solution to the (ε, δ)-GCircuit problem. Here the notation $a = b \pm \varepsilon$ is used as a shorthand for $a \in [b - \varepsilon, b + \varepsilon]$.

We will use this version to prove that the PCP-for-PPAD conjecture is *necessary* to obtain PPAD-hardness for our market equilibrium problem.

2.2.4 Equivalence of the three formulations. We prove the following theorem which shows that all these formulations are indeed equivalent.

Theorem 2.3. *The following are all equivalent formulations of the* PCP-for-PPAD *conjecture:*

(1) *There exists a constant $\delta > 0$ such that δ-Pure-Circuit is* PPAD*-hard, even when every node is the input to exactly one gate.*
(2) *There exists a constant $\delta > 0$ such that δ-Pure-Circuit is* PPAD*-hard.*
(3) *There exist constants $\varepsilon > 0$ and $\delta > 0$ such that (ε, δ)-*GCircuit *is* PPAD*-hard.*
(4) *There exist constants $\varepsilon > 0$ and $\delta > 0$ such that (ε, δ)-*GCircuit+ *is* PPAD*-hard.*

The proof can be found in the full version of the paper.

3 TECHNICAL OVERVIEW

3.1 Hardness for Fisher Markets from the PCP-for-PPAD conjecture

We show hardness results for Fisher markets by reducing from the δ-Pure-Circuit problem, whose PPAD-hardness for some constant δ is equivalent to the PCP-for-PPAD conjecture (by Theorem 2.3). The starting point for obtaining such a reduction is to examine the existing reduction from Pure-Circuit to Fisher markets with approximate clearing (but exact optimal bundles) [16]. Unfortunately, in order for that reduction to work with $(1-\delta_m)$-optimal bundles, we would need to set δ_m to an inverse-polynomial value. Indeed, setting δ_m to be a constant fails due to various parameters in the reduction being set to inverse-polynomial values.

This, in turn, is due to the fact that this construction uses a so-called *reference good*. This is a good that is desired by a buyer with an enormous budget (compared to the other buyers' budgets), who thus spends all their money on that good. As a result, the price of that good is very stable, as it barely moves (in relative terms) depending on whether any of the other buyers spend any money on it or not. This is very useful for the reduction, as it provides a stable reference price with respect to which other prices can be defined. However, this imbalance between an enormous budget and smaller budgets yields inverse-polynomial parameters.

We bypass this obstacle by avoiding the use of a reference good altogether. This yields a simpler, more direct reduction where all parameters are constant, including the slopes of the utility functions and the degrees in the graph of interactions between buyers and goods. As a result, we can set δ_m to be a sufficiently small constant and thus obtain PPAD-hardness assuming the PCP-for-PPAD conjecture. Furthermore, by setting δ_m to a sufficiently small inverse-polynomial value, we also obtain unconditional PPAD-hardness for that regime.

The instances we construct have two further properties that yield improvements over prior work, even for the case where $\delta_m = 0$.

- The reduction works for any clearing parameter $\varepsilon_m < 1/9$, improving upon the previously best $\varepsilon_m < 1/11$ from [16], even when $\delta_m = 0$.
- The constructed market satisfies the CEEI property, i.e., all the buyers have identical budget. No hardness result for this setting was known, even when $\varepsilon_m = \delta_m = 0$.

To summarize, this reduction is a significant simplification of a construction appearing in prior work that yields stronger results. For the details, see Section 4.

The intuition for why we need the PCP-for-PPAD conjecture in order to show hardness for constant δ_m is the following. If δ_m is some constant, say 0.1, then that means that every buyer can spend 10% of their budget on arbitrary goods, as long as they spend their remaining 90% optimally. Indeed, doing so will still guarantee 90% of the optimal total utility they could have achieved by spending their whole budget optimally. But now this means that

10% of the total amount of money available in the market can be spent in a completely arbitrary manner. In particular, this "rogue budget" can completely disrupt the functionality of some gadgets implementing gates. However, because it is only a constant fraction of the total amount of money, we can show that this rogue budget can only disrupt a constant fraction of the gadgets. This is where the PCP-for-PPAD conjecture comes in, as it precisely states that Pure-Circuit remains hard even if a constant fraction of the gates malfunction. In the next section, we show that the conjecture is in fact needed to prove hardness, at least for a family of markets with some nice structure.

3.2 Hardness for Fisher Markets Implies the PCP-for-PPAD conjecture

We now give a high-level overview of our result that if $(0, \delta)$-approximate market equilibrium is PPAD-hard for some constants ε and δ, then the PCP-for-PPAD conjecture holds. Full details and proofs can be found in Section 5.

Reducible markets. We say that a market is *reducible* if it satisfies the following properties.

- It has constant degree d, meaning that each buyer is interested in at most d goods, and each good is desired by at most d buyers.
- The budgets of all buyers lie in a range $[e_{\min}, e_{\max}]$, where $e_{\min}$ and $e_{\max}$ are both constants.
- Each buyer has an SPLC utility function where each utility-function uses a constant number of segments, and the slope of each utility-function segment is either zero or lies in a range $[1, \kappa]$ where κ is a constant.

We show that if it is PPAD-hard to find an (ε, δ)-approximate market equilibrium in a reducible Fisher market for some constants ε and δ, then the PCP-for-PPAD conjecture holds. We note in particular that the family of instances generated by our hardness results is reducible, so we get that the PCP-for-PPAD conjecture holds if and only if it is PPAD-hard to find an (ε, δ)-approximate market equilibrium in a reducible Fisher market.

We prove this result using a four-step procedure, where we first make a single query to $(\varepsilon_c, \delta_c)$-GCircuit+ for some suitably small constants ε_c and δ_c, and then use the result of this query to compute, in polynomial time, an (ε, δ)-approximate market equilibrium of the Fisher market. Thus if it is PPAD-hard to find an (ε, δ)-approximate market equilibrium for constants ε and δ, it must also be PPAD-hard to find a $(\varepsilon_c, \delta_c)$-GCircuit+ solution for constants ε_c and δ_c, which implies the PCP-for-PPAD conjecture. We give a high-level overview of the reduction here. More details can be found in Section 5, while the proofs are deferred to the full version of the paper.

3.2.1 *Step 1: Reduce to $(\varepsilon_c, \delta_c)$-GCircuit+.*

We start by formulating the (ε, δ)-approximate market equilibrium problem as an $(\varepsilon_c, \delta_c)$-GCircuit+ instance for some constants ε_c and δ_c.

The variables of this instance will encode a price vector p that assigns a price p_j to each good j, and a vector q such that for each buyer i, good j, and utility-function segment k, the variable $q_{i,j,k}$ gives the total amount of money that buyer i spends on that utility-function segment. Setting $x_{i,j} = \sum_k q_{i,j,k}/p_j$ then allows us to recover an allocation from the GCircuit+ solution.

We show that a system of constraints can be imposed on these variables to ensure that, in an $(\varepsilon_c, \delta_c)$-GCircuit+ solution, we have that there is a constant c such that the following properties hold.

- A $(1 - c \cdot \delta_c)$-fraction of the goods $(c \cdot \varepsilon_c)$-clear.
- A $(1 - c \cdot \delta_c)$-fraction of the buyers have $(1 - c \cdot \varepsilon_c)$-optimal allocations and satisfy their budget constraints.

The fact that only a $(1 - c \cdot \delta_c)$-fraction of the goods and buyers satisfy their constraints arises from the fact that a δ_c-fraction of the constraints in our GCircuit+ instance fail. The number of constraints used for each buyer is proportional to the number of non-zero utility-function segments that the buyer has, which is a constant in a reducible market. Likewise, the number of constraints used for each good is proportional to the non-zero utility-function segments for that good, which is also constant in a reducible market. So the δ_c-fraction of constraints that fail then translate to a $(c \cdot \delta_c)$-fraction of buyers and goods for which at least one constraint fails for some constant c.

3.2.2 *Step 2: Fix the broken buyers.*

In Step 2 we address the $(c \cdot \delta_c)$-fraction of buyers that do not receive a $(1 - c \cdot \varepsilon_c)$-optimal allocation at the end of Step 1. We do this simply by making those buyers buy an optimal allocation at the current price vector p, while leaving all other buyers unchanged. This then ensures that all buyers receive a $(1 - c \cdot \varepsilon_c)$-optimal allocation, and that all buyers satisfy their budget constraints.

The cost of doing this is that any buyer who changes their allocation in Step 2 may affect the clearing constraint of the goods that they move money from or to. This is fine, however, because the degree d of the market is constant, and at most a $(c \cdot \delta_c)$-fraction of the buyers change their allocation during Step 2. So at most a $(c \cdot \delta_c)$-fraction of the goods violated their clearing constraints at the start of Step 2, and at most $(d \cdot c \cdot \delta_c)$-fraction of the goods have their clearing constraints broken by the shifting of allocations in Step 2.

To summarize, at the end of Step 2, we have a price vector p, a revised allocation x, and a constant c (larger than the constant used in Step 1) such that the following hold.

- A $(1 - c \cdot \delta_c)$-fraction of the goods $(c \cdot \varepsilon_c)$-clear.
- All buyers have a $(1 - c \cdot \varepsilon_c)$-optimal allocation and satisfy their budget constraint.

3.2.3 *Step 3: Fix the over-clearing.*

In Step 3 we receive a price vector p, an allocation x and a constant c such that a $(1 - c \cdot \delta_c)$-fraction of the goods $(c \cdot \varepsilon_c)$-clear. We say that a good j *over-clears* if $\sum_i x_{i,j} > 1 + c \cdot \varepsilon_c$, and we say that a good *under-clears* if $\sum_i x_{i,j} < 1 - c \cdot \varepsilon_c$. In Step 3, our goal is to remove all of the over-clearing in the current allocation, and this is by far the most technically involved step.

Burning. In Step 3 we will temporarily allow each buyer to *burn* up to ε_c money. When a buyer burns money, they do not spend it on any goods, and they instead just destroy it. Technically, this is implemented by loosening the budget constraint so that buyer i is required to spend at least $e_i - \varepsilon_c$ and at most e_i money.

This is only a temporary measure. We will use burning as a technical tool to help us reduce over-clearing goods in Step 3, and we will later make use of the burned money in Step 4 to fix the under-clearing goods, and thereby restore the budget constraints for all buyers.

The Step 3 invariant. Throughout Step 3, we maintain the following invariant. For each buyer i, we define $\text{marginal-bpb}_i(p, x)$ to be the bang-per-buck of the most desirable utility-function segment (under prices p) that buyer i does not fully buy. We say that buyer i satisfies the invariant if they do not spend any money on segments that have bang-per-buck that is strictly worse than $\text{marginal-bpb}_i(p, x)/(1 + c_3 \cdot \varepsilon_c)$ for some constant $c_3 > 0$. It is relatively easy to show that any buyer that satisfies the invariant will receive an approximately optimal allocation, and so will maintain this invariant for all buyers throughout Step 3.

We also define the *window* for each buyer to be the interval $[\text{marginal-bpb}_i(p, x)/(1+c_3 \cdot \varepsilon_c), \text{marginal-bpb}_i(p, x)]$. We say that a utility-function segment is *in the window* if its bang-per-buck lies in the window interval. An important property of the window is that a buyer can shift money between utility-function segments that lie within the window without affecting the invariant.

Although we do not necessarily satisfy the invariant at the start of Step 3, the properties that we prove for Step 1 ensure that each buyer buys their segments in approximate bang-per-buck order, and we show that a straightforward preprocessing step can use this to transform the allocation at the end of Step 2 into an allocation that meets the invariant, without significantly altering the clearing constraints of the goods.

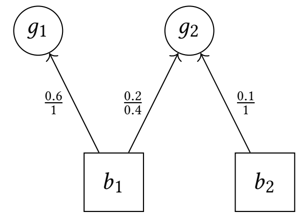
Shift-and-burn. Our procedure for Step 3 involves alternating two algorithms, the first of which we call shift-and-burn. The purpose of this algorithm is to shift money from the over-clearing goods on to buyers, who then burn that money.

In Figure 3a we show a simple scenario in which there is a good g_1 that over-clears, and a good g_2 that does not over- or under-clear. The two buyers in the example have utility-function segments to the goods that all lie in their buyer's windows, which are shown as edges in the figure. The labels of the edges show how much of those utility function segments is bought in the current allocation. So for example, buyer b_1 has an in-window utility function segment to good g_1 with length 1, and is currently buying 0.6 units of that segment, and another in-window segment to good g_2, of which 0.2 of the maximum 0.4 units are currently bought.

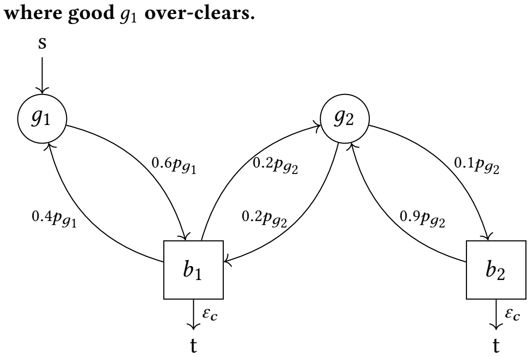
Note that buyer b_1 can *shift* up to ε_c money from g_1 and burn it instead, and doing so reduces the over-clearing at g_1, since fewer units of g_1 would now be bought. Buyer b_1 can also shift ε_c money from g_1 to g_2, which reduces the over-clearing at g_1 while increasing the amount of money spent on g_2. If, at the same time, b_2 then shifts ε_c money from g_2 and burns it instead, then the over-clearing at g_1 is reduced, while the clearing constraint at g_2 remains unaffected.

We view this shifting process as a flow problem.

- The source s of the flow problem has an edge to each over-clearing good j, with the capacity of that edge being the amount of money that would need to be removed from good j to stop it over-clearing.
- Each buyer has an edge to the target t of the flow problem with capacity ε_c, which represents the amount of money that they can burn. In later iterations, when the buyer may have already burned some money, this capacity will be reduced to ensure that no buyer burns more than ε_c money.
- For each utility-function segment for buyer i and good j, if the segment lies in buyer i's window, then we add an edge from j to i whose capacity is the total amount of money that is currently being spent on that utility-function segment, and an edge from i to j whose capacity is equal to the total amount of extra money that could be spent on that utility-function segment before it is fully used.



(a) Two buyers with in-window segments to two goods, where good g_1 over-clears.



(b) The resulting flow graph.

Figure 3: The construction of the flow graph.

Figure 3b shows the flow problem associated with the scenario from Figure 3a. Note that the edge from g_1 to b_1 has capacity $0.6 \cdot p_{g_1}$, since that is the total amount of money that b_1 is currently spending on g_1 for this utility-function segment, and thus the largest amount of money that b_1 could shift off this utility-function segment. Meanwhile, the edge from b_1 to g_2 has capacity $0.2 \cdot p_{g_2}$ since that is the total amount of money that b_1 could shift on to that utility-function segment without exhausting it.

Computing a maximal s-t flow then tells us the maximum amount of money that we can shift from the over-clearing goods to money that is burned by the buyers. In particular, since we have a flow, any intermediate goods through which money is routed do not have their clearing constraints changed, since exactly as much money is spent on them after applying the flow as was beforehand.

We will show that applying the shift-and-burn procedure maintains the invariant. Indeed, by construction, the flow is only able to change the amount of money spent on segments that lie within their buyer's window. However, it is possible that marginal-bpb$_i(p, x)$ changes after we apply the flow, which occurs whenever we fully buy all of the segments that defined marginal-bpb$_i(p, x)$ before the flow is applied. This changes the windows that we must consider in the new allocation, but we will show that this is not an issue, and that the invariant is maintained even if marginal-bpb$_i(p, x)$ changes for some buyer i. It will however be technically convenient to re-run shift-and-burn whenever this occurs, and we show that only a polynomial number of iterations are required to reach a shift-and-burn step in which no buyer's window changes.

The restricted flow graph. After the last shift-and-burn step, we then construct the *restricted flow graph*, which is the result of applying the following iterative algorithm to the flow graph that was considered in the last shift-and-burn iteration. If f is the maximal s-t flow that we computed in the flow graph, then we do the following.

(1) Add all over-clearing goods to the restricted flow graph.
(2) If j is a good in the restricted flow graph, and there is an edge from j to buyer i that is not saturated by f, then we add buyer i to the restricted flow graph.
(3) If i is a buyer in the restricted flow graph, and there is an edge from i to good j that is not saturated by f, then we add good j to the restricted flow graph.

The restricted flow graph is obtained by repeating points 2 and 3 above until convergence.

An important property of the restricted flow graph is that, if there are still over-clearing goods in the market, and if a $(c \cdot \delta_c)$-fraction of the goods are over-clearing for some constant c, then the restricted flow graph contains at most a $(c' \cdot \delta_c/\varepsilon_c)$-fraction of the goods for some constant c'. This follows from the following two observations.

- Every buyer in the restricted flow graph fully burns, i.e., they burn exactly ε_c money. This can easily be shown by contradiction: if a buyer i in the restricted flow graph did not fully burn then by construction there is a path in the restricted flow graph from s to i that only uses unsaturated edges, so we could increase the flow on those edges in order to burn more at i, which would contradict the maximality of f.
- The total amount of money spent on an over-clearing good is at most some constant M because each buyer's budget is less than $e_{\max}$, which is constant, and since the market has constant degree.

So at most $M \cdot (c \cdot \delta_c) \cdot |G|$ money is spent on over-clearing goods, and therefore if the restricted flow graph contains more than

$$\frac{M \cdot c \cdot \delta_c}{\varepsilon_c} \cdot |G|$$

goods, then all of the over-clearing must have been burned, meaning that there are no longer any over-clearing goods.

Pump-and-shift. The second procedure that we use in Step 3 is called the *pump-and-shift* procedure. Here we *pump* the prices of all goods in the restricted flow graph by multiplying them by $(1 + \varepsilon_c)$.

After the pump, we then need to repair the allocation by shifting money. In particular, if a utility-function segment (j, k) was above its buyer's window, then buyer i will need to spend more money in order to continue to fully buy that segment and satisfy the invariant.

To address this, we simply instruct each buyer who is interested in at least one good that was pumped to shift money from their worst bang-per-buck utility-function segments on to their best non-fully-bought utility-function segments, and we order this process so we first shift on to the segments that have the highest bang-per-bucks. We will show that, after this shifting operation, all buyers will continue to satisfy the invariant.

The pump-and-shift step does come with another cost, however, which is that as the buyers shift, they may introduce under-clearing in goods that are adjacent to the restricted flow graph. That is, if a buyer shifts money from a segment for their worst bang-per-buck good j onto another segment for a pumped good j', then the clearing constraint for j may be affected, and in particular we may have that j under-clears even if it was not under-clearing beforehand.

Here our bounds on the size of the restricted flow graph play a crucial role. The scenario described above only occurs when there is a buyer i that is interested in goods j and j', and j' is in the restricted flow graph. Since the market has constant degree d, and since at most a $(c' \cdot \delta_c/\varepsilon_c)$-fraction of the goods lie in the restricted flow graph for some constant c', we get that at most a $(d^2 \cdot c' \cdot \delta_c/\varepsilon_c)$-fraction of the goods can under-clear as a result of the pump step, which keeps the under-clearing small enough for our purposes.

Combining into Step 3. Step 3 consists of alternating shift-and-burn and pump-and-shift until no over-clearing remains. We prove that this will terminate after a constant number of rounds. The argument for this uses the fact that all goods have prices lying in the range $[P_{\min}, P_{\max}]$ for some constants $P_{\min}$ and $P_{\max}$ after after Step 1. We can multiply $P_{\min}$ by $(1 + \varepsilon_c)$ at most constantly many times before it exceeds $P_{\max}$, because $P_{\min}$ and $P_{\max}$ are both constant, and ε_c is also constant. We then argue that if a good has price $P_{\max}$ then it cannot possibly over-clear, because $P_{\max}$ was selected to be so high that even if all buyers that are interested in a good spend all of their money on it, the good still cannot over-clear. Thus, after constantly many rounds all over-clearing goods will have been pumped to the point where they no longer over-clear.

At the end of Step 3, we arrive at a price vector p, an allocation x, and a constant c (again, larger than the constant from Step 2) such that

- Every buyer receives a $(1 - c \cdot \varepsilon_c)$-optimal allocation, and burns at most ε_c money.
- At least $(1 - c \cdot (\delta_c/\varepsilon_c))$-fraction of the goods $(c \cdot \varepsilon_c)$-clear, and no good over-clears.

3.2.4 Step 4: Fix the under-clearing. There are only two tasks left at this point: fix the under-clearing and deal with the money that was burned in Step 3. We can in fact use one of these problems to fix the other. We instruct each buyer that has burned money to spend that money on goods that under-clear. In general this will deliver zero extra utility to those buyers, since they may not be interested in those goods, but we already have that these buyers are given $(1 - c \cdot \varepsilon_c)$-optimal allocations, so this does not harm us.

There may be too much or too little burned money to do this. We use the following procedures to deal with this.

- If there is too much burned money, then after fixing all of the under-clearing, we then instruct the buyers to spend their money equally on each of the goods in the market. This will push up the demand of all goods slightly, but we will show bounds on the amount of money burned in Step 3, which will be sufficient to show that all goods $(c' \cdot \varepsilon_c)$-clear after this operation, for some constant $c' > c$.
- If there is too little burned money, then after we exhaust the burn pool, we then instruct all buyers in the market to shift a small amount of money off their worst bang-per-buck utility segments, and then to use that money to clear the under-clearing goods. We will show bounds on the amount of money needed to clear the under-clearing goods that are sufficient to argue that only a small amount of money needs to be shifted in this way. In particular, each buyer will still have a $(1 - g(\varepsilon_c, \delta_c))$-optimal allocation after this operation for some function g, where ε_c and δ_c now appear in this bound because at the end of Step 3 the fraction of under-clearing goods depended on both ε_c and δ_c.
  Likewise, since the market has constant degree, and each buyer shifts only a small amount of money, in the worst case the total amount of money shifted off each good during this operation is still small, and we maintain that each good $f(\varepsilon_c, \delta_c)$-clears for some function f.

At the end of Step 4 we have a price vector p, an allocation x such that the following hold.

- Every buyer receives a $(1 - g(\varepsilon_c, \delta_c))$-optimal allocation, and satisfies their budget constraint.
- All goods $f(\varepsilon_c, \delta_c)$-clear.

3.2.5 Step 5: Obtain exact clearing. With the results from Step 4, by picking ε_c and δ_c to be suitably small constants, it is possible to show that if it is PPAD-hard to find a (ε, δ)-approximate market equilibrium of a reducible market for some constants ε and δ, then the PCP-for-PPAD conjecture holds. We can however show the stronger theorem that if it is PPAD-hard to find a $(0, \delta)$-approximate market equilibrium of a reducible market for some constant δ, then the PCP-for-PPAD conjecture holds. We do this in Step 5, where we transform the allocation x and price vector p from Step 4 so that all goods exactly clear, while ensuring that not too much is lost from the buyer's optimality during the process.

We do this in a three step process.

(1) The first step is to remove money from the allocation so that every good has demand exactly equal to $1 - f(\varepsilon_c, \delta_c)$. Note that Step 4 ensures that all goods $f(\varepsilon_c, \delta_c)$-clear, so this can be achieved by removing at most $2 \cdot f(\varepsilon_c, \delta_c)$ demand from each good. Since $f(\varepsilon_c, \delta_c)$ will be chosen to be small, we show that each buyer does not lose too much of their utility during this operation, although they are now underspending their budget.
(2) Next, we take the money that was removed in the previous step, and we instruct all buyers to distribute it evenly across all of the goods in such a way as to ensure that the demand of all goods is equal to some value $D \in [1 - f(\varepsilon_c, \delta_c), 1 + f(\varepsilon_c, \delta_c)]$. This can only increase the buyer's utilities, so the optimality of their bundles is not affected, and this also reestablishes the budget constraint of each of the buyers.
(3) Finally, we multiply the prices by a factor of D, while keeping the amount of money spent on each good the same. This normalizes the demand of each good to 1, giving us the exact clearing that we desire. This step can affect the optimality of the bundles, but we show that since $D \in [1 - f(\varepsilon_c, \delta_c), 1 + f(\varepsilon_c, \delta_c)]$, and since $f(\varepsilon_c, \delta_c)$ is very small, the bundles remain approximately optimal.

At the end of this process we arrive at an allocation x' and a price vector p' such that the following hold.

- Every good exactly clears.
- Every buyer receives a $1 - g'(\varepsilon_c, \delta_c)$-optimal bundle for some function g'.

As a final step, we then choose constant values for ε_c and δ_c to ensure that $g'(\varepsilon_c, \delta_c) \le \delta$, which then allows us to recover the $(0, \delta)$-approximate market equilibrium that we seek.

4 HARDNESS FOR FISHER MARKETS FROM THE PCP-for-PPAD CONJECTURE

In this section we prove that, if we assume the PCP-for-PPAD conjecture, then, for any $\varepsilon < 1/9$, there exist some sufficiently small constant $\delta > 0$ such that computing an (ε, δ)-approximate market equilibrium in a Fisher market with SPLC utilities is PPAD-hard.

Our reduction also yields two further results that hold unconditionally, without needing to assume the PCP-for-PPAD conjecture. Namely, we obtain PPAD-hardness for the problem, if we either

- set $\delta > 0$ to be inverse-polynomial instead of constant, or
- do not allow buyers to spend any money on goods for which their utility function is the zero function.

Furthermore, all of our hardness results hold for a class of Fisher markets that are particularly "simple", in the following sense.

Definition 4.1. A family of Fisher markets is *simple* if the utility functions are SPLC *linear capped*, all buyers have budget $e_i = 1$ (CEEI), and there exist constants such that for any market in the family:

- Every buyer i has non-zero utility function u_{ij} for at most a constant number of goods j.
- For every good j, the utility function u_{ij} is non-zero for at most a constant number of buyers i.
- For every buyer i, the non-zero slopes in the utility functions u_{ij} have ratio bounded by a constant.

Furthermore, the sufficient condition for existence of equilibrium holds, i.e., for every buyer i, there exists at least one good j such that i is not satiated with good j (in other words, the function u_{ij} is linear (uncapped) with strictly positive slope).

Theorem 4.2. *Fix any constant $\varepsilon < 1/9$. Assuming the* PCP-for-PPAD *conjecture, there exists a constant $\delta > 0$ such that finding an (ε, δ)-approximate market equilibrium in* simple *Fisher markets is* PPAD-*hard.*

The reduction establishing this theorem also yields the following two unconditional results.

Theorem 4.3. *Fix any constant $\varepsilon < 1/9$. For inverse-polynomial $\delta > 0$ given as part of the input, finding an (ε, δ)-approximate market equilibrium in* simple *Fisher markets is* PPAD*-hard.*

Theorem 4.4. *Fix any constant $\varepsilon < 1/9$. Then, there exists a constant $\delta > 0$ such that finding an (ε, δ)-approximate market equilibrium in* simple *Fisher markets is* PPAD*-hard,* if buyers are not allowed to spend any money on goods for which their utility function is the zero function.

These three theorems also hold for the *Arrow-Debreu exchange market* setting, where buyers do not have a budget and are instead endowed with goods. This follows by a known direct reduction. The proofs for all these results can be found in the full version of the paper.

5 HARDNESS FOR FISHER MARKETS IMPLIES THE PCP-for-PPAD CONJECTURE

In this section we will prove that PPAD-hardness of finding market equilibria with approximately optimal bundles would imply that the PCP-for-PPAD conjecture holds. We will do so for a class of markets that we call the *reducible* markets, which are defined as follows. A Fisher market $(G, B, (e_i)_{i \in B}, (u_i)_{i \in B})$ is reducible if all of the following hold.

- The market has *constant* degree d. We say that a market has degree d if, for each buyer $i \in B$, there are at most d goods j such that $u_{i,j}$ is a non-zero function, and for each good j there are at most d buyers i such that $u_{i,j}$ is a non-zero function.
- The budgets are upper and lower bounded by some constants $e_{\max}$ and $e_{\min}$, i.e., $\max_i e_i \leq e_{\max}$ and $\min_i e_i \geq e_{\min}$.
- The utilities are SPLC, each piecewise linear utility function $u_{i,j}$ has a constant number of pieces, and the non-zero slopes of each piece lie in the range $[1, \kappa]$ for some constant κ. Note that so long as, for any given buyer, the ratios between all non-zero slopes are bounded by a constant, the latter condition can be achieved by a simple re-normalization step.

  More formally, for each i and j, if $\langle (s_{i,j,1}, \ell_{i,j,1}), (s_{i,j,2}, \ell_{i,j,2}), \ldots, (s_{i,j,m}, \ell_{i,j,m}) \rangle$ is the representation of the utility function $u_{i,j}$, then we require that m is upper bounded by a constant, and that for all k, $s_{i,j,k}$ is either zero or lies in $[1, \kappa]$.

In this section we show the following theorem.

Theorem 5.1. *If there exists $\delta > 0$ such that finding a $(0, \delta)$-approximate market equilibrium in reducible Fisher markets is* PPAD*-hard, then the* PCP-for-PPAD *conjecture holds.*

An important point is that the *simple* markets (Definition 4.1) for which our hardness result holds (Theorem 4.2) are in particular also *reducible* markets, as defined above. As a result, Theorem 5.1 shows that the PCP-for-PPAD conjecture is necessary to obtain PPAD-hardness for equilibrium with approximately optimal bundles in simple markets. The proof of Theorem 5.1 can be found in the full version of the paper.

ACKNOWLEDGMENTS

We thank the reviewers for useful comments and suggestions. Argyrios Deligkas was supported by EPSRC Grant EP/X039862/1 “NAfANE: New Approaches for Approximate Nash Equilibria”. John Fearnley was supported by EPSRC Grant EP/W014750/1 “New Techniques for Resolving Boundary Problems in Total Search”.

REFERENCES

[1] Saeed Alaei, Pooya Jalaly Khalilabadi, and Eva Tardos. 2017. Computing equilibrium in matching markets. In *Proceedings of the 2017 ACM Conference on Economics and Computation*. 245–261.

[2] Yakov Babichenko, Christos H. Papadimitriou, and Aviad Rubinstein. 2016. Can Almost Everybody be Almost Happy?. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*. ACM, 1–9. https://doi.org/10.1145/2840728.2840731

[3] Benjamin Birnbaum, Nikhil R Devanur, and Lin Xiao. 2010. New convex programs and distributed algorithms for Fisher markets with linear and spending constraint utilities. *Unpublished manuscript* (2010).

[4] William C Brainard and Herbert E Scarf. 2005. How to compute equilibrium prices in 1891. *American Journal of Economics and Sociology* 64, 1 (2005), 57–83.

[5] Mark Braverman. 2021. Optimization-friendly generic mechanisms without money. *arXiv preprint arXiv:2106.07752* (2021).

[6] Sourav Chakraborty, Nikhil R. Devanur, and Chinmay Karande. 2010. Market Equilibrium with Transaction Costs. In *Proceedings of the 6th Workshop on Internet and Network Economics (WINE)*. 496–504. https://doi.org/10.1007/978-3-642-17572-5_43

[7] Thomas Chen, Xi Chen, Binghui Peng, and Mihalis Yannakakis. 2022. Computational Hardness of the Hylland-Zeckhauser Scheme. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM, 2253–2268.

[8] Xi Chen, Xiaotie Deng, and Shang-Hua Teng. 2009. Settling the complexity of computing two-player Nash equilibria. *J. ACM* 56, 3 (2009), 14:1–14:57. https://doi.org/10.1145/1516512.1516516

[9] Xi Chen and Shang-Hua Teng. 2009. Spending is not easier than trading: on the computational equivalence of Fisher and Arrow-Debreu equilibria. In *International Symposium on Algorithms and Computation*. Springer, 647–656.

[10] Bruno Codenotti, Benton McCune, and Kasturi Varadarajan. 2005. Market equilibrium via the excess demand function. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*. 74–83.

[11] Bruno Codenotti, Sriram V Pemmaraju, and Kasturi R Varadarajan. 2005. On the polynomial time computation of equilibria for certain exchange economies. In *SODA*, Vol. 5. 72–81.

[12] Bruno Codenotti and Kasturi Varadarajan. 2007. Computation of market equilibria by convex programming. *Algorithmic Game Theory* (2007), 135–158.

[13] Constantinos Daskalakis, Noah Golowich, and Kaiqing Zhang. 2023. The complexity of Markov equilibrium in stochastic games. In *The Thirty Sixth Annual Conference on Learning Theory*. PMLR, 4180–4234.

[14] Argyrios Deligkas, John Fearnley, Alexandros Hollender, and Themistoklis Melissourgos. 2023. Tight inapproximability for graphical games. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 5600–5607.

[15] Argyrios Deligkas, John Fearnley, Alexandros Hollender, and Themistoklis Melissourgos. 2024. Constant Inapproximability for Fisher Markets. In *Proceedings of the 25th ACM Conference on Economics and Computation (EC)*. 13–39. https://doi.org/10.1145/3670865.3673533

[16] Argyrios Deligkas, John Fearnley, Alexandros Hollender, and Themistoklis Melissourgos. 2024. Constant Inapproximability for Fisher Markets. In *Proceedings of the 25th ACM Conference on Economics and Computation, EC 2024, New Haven, CT, USA, July 8-11, 2024*, Dirk Bergemann, Robert Kleinberg, and Daniela Sabán (Eds.). ACM, 13–39. https://doi.org/10.1145/3670865.3673533

[17] Argyrios Deligkas, John Fearnley, Alexandros Hollender, and Themistoklis Melissourgos. 2024. Pure-Circuit: Tight Inapproximability for PPAD. *JACM* 71, 5 (2024), 31:1–31:48. https://doi.org/10.1145/3678166

[18] Xiaotie Deng, Christos Papadimitriou, and Shmuel Safra. 2003. On the complexity of price equilibria. *J. Comput. System Sci.* 67, 2 (2003), 311–324. https://doi.org/10.1016/s0022-0000(03)00011-4

[19] Nikhil R Devanur. 2004. The spending constraint model for market equilibrium: Algorithmic, existence and uniqueness results. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*. 519–528.

[20] Nikhil R Devanur and Ravi Kannan. 2008. Market equilibria in polynomial time for fixed number of goods or agents. In *2008 49th Annual IEEE Symposium on Foundations of Computer Science*. IEEE, 45–53.

[21] Nikhil R Devanur, Christos H Papadimitriou, Amin Saberi, and Vijay V Vazirani. 2008. Market equilibrium via a primal–dual algorithm for a convex program. *Journal of the ACM (JACM)* 55, 5 (2008), 1–18.

[22] Simon Dohn, Kristoffer Arnsfelt Hansen, and Asger Klinkby. 2025. Improved Hardness Results for the Clearing Problem in Financial Networks with Credit Default Swaps. In *International Symposium on Algorithmic Game Theory*. Springer, 81–98.

[23] Edmund Eisenberg. 1961. Aggregation of utility functions. *Management Science* 7, 4 (1961), 337–350.

[24] Jugal Garg, Martin Hoefer, and Kurt Mehlhorn. 2024. Satiation in Fisher Markets and Approximation of Nash Social Welfare. *Mathematics of Operations Research* 49, 2 (2024), 1109–1139. https://doi.org/10.1287/moor.2019.0129

[25] Jugal Garg, Ruta Mehta, Milind Sohoni, and Vijay V. Vazirani. 2015. A Complementary Pivot Algorithm for Market Equilibrium under Separable, Piecewise-Linear Concave Utilities. *SIAM J. Comput.* 44, 6 (2015), 1820–1847. https://doi.org/10.1137/140971002

[26] Jugal Garg, Yixin Tao, and László A Végh. 2022. Approximating Equilibrium under Constrained Piecewise Linear Concave Utilities with Applications to Matching Markets. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM, 2269–2284.

[27] Jugal Garg, Yixin Tao, and László A. Végh. 2026. Approximating competitive equilibrium by Nash welfare. *Games and Economic Behavior* 158 (2026), 142–166. https://doi.org/10.1016/j.geb.2026.03.004

[28] Kristoffer Arnsfelt Hansen and Xinhao Nie. 2025. On the Complexity of Stationary Nash Equilibria in Discounted Perfect Information Stochastic Games. *arXiv preprint arXiv:2510.11550* (2025).

[29] Aanund Hylland and Richard Zeckhauser. 1979. The efficient allocation of individuals to positions. *Journal of Political economy* 87, 2 (1979), 293–314.

[30] Stavros D Ioannidis, Bart de Keijzer, and Carmine Ventre. 2023. Clearing Financial Networks with Derivatives: From Intractability to Algorithms. *arXiv preprint arXiv:2312.05139* (2023).

[31] Kamal Jain, Mohammad Mahdian, and Amin Saberi. 2003. Approximating Market Equilibria. In *Proceedings of the 6th Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*. 98–108. https://doi.org/10.1007/978-3-540-45198-3_9

[32] Devansh Jalota, Marco Pavone, Qi Qi, and Yinyu Ye. 2023. Fisher markets with linear constraints: Equilibrium properties and efficient distributed algorithms. *Games and Economic Behavior* 141 (2023), 223–260.

[33] Sham M Kakade, Michael Kearns, and Luis E Ortiz. 2004. Graphical economics. In *Learning Theory: 17th Annual Conference on Learning Theory, COLT 2004, Banff, Canada, July 1-4, 2004. Proceedings 17*. Springer, 17–32.

[34] Chinmay Karande and Nikhil Devanur. 2007. Computing Market Equilibrium: Beyond Weak Gross Substitutes. In *Proceedings of the 3rd Workshop on Internet and Network Economics (WINE)*. 368–373. https://doi.org/10.1007/978-3-540-77105-0_40

[35] Robert R. Maxfield. 1997. General equilibrium and the theory of directed graphs. *Journal of Mathematical Economics* 27, 1 (1997), 23–51. https://doi.org/10.1016/0304-4068(95)00763-6

[36] James B Orlin. 2010. Improved algorithms for computing Fisher's market clearing prices: Computing Fisher's market clearing prices. In *Proceedings of the forty-second ACM symposium on Theory of computing*. 291–300.

[37] Abraham Othman, Christos Papadimitriou, and Aviad Rubinstein. 2016. The complexity of fairness through equilibrium. *ACM Transactions on Economics and Computation (TEAC)* 4, 4 (2016), 1–19.

[38] Chanwoo Park, Kaiqing Zhang, and Asuman Ozdaglar. 2023. Multi-player zero-sum Markov games with networked separable interactions. *Advances in Neural Information Processing Systems* 36 (2023), 37354–37369.

[39] Aviad Rubinstein. 2016. Settling the Complexity of Computing Approximate Two-Player Nash Equilibria. In *Proceedings of the 57th IEEE Symposium on Foundations of Computer Science (FOCS)*. 258–265. https://doi.org/10.1109/focs.2016.35

[40] Aviad Rubinstein. 2018. Inapproximability of Nash equilibrium. *SIAM J. Comput.* 47, 3 (2018), 917–959. https://doi.org/10.1137/15M1039274

[41] Vijay V Vazirani. 2010. Spending constraint utilities with applications to the adwords market. *Mathematics of Operations Research* 35, 2 (2010), 458–478.

[42] Vijay V. Vazirani and Mihalis Yannakakis. 2011. Market equilibrium under separable, piecewise-linear, concave utilities. *J. ACM* 58, 3 (2011), 1–25.

[43] Vijay V. Vazirani and Mihalis Yannakakis. 2021. Computational Complexity of the Hylland-Zeckhauser Scheme for One-Sided Matching Markets. In *12th Innovations in Theoretical Computer Science Conference, ITCS (LIPIcs, Vol. 185)*. 59:1–59:19. https://doi.org/10.4230/LIPIcs.ITCS.2021.59

[44] László A Végh. 2012. Strongly polynomial algorithm for a class of minimum-cost flow problems with separable convex objectives. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*. 27–40.

Received 2025-11-04; accepted 2026-02-01