

Authors are encouraged to submit new papers to INFORMS journals by means of a style file template, which includes the journal title. However, use of a template does not certify that the paper has been accepted for publication in the named journal. INFORMS journal templates are for the exclusive purpose of submitting to an INFORMS journal and should not be used to distribute the papers in print or online or to submit the papers to another publication.

Satiation in Fisher Markets and Approximation of Nash Social Welfare

Jugal Garg

Dept. of Industrial & Enterprise Systems Engineering, Univ. of Illinois at Urbana-Champaign, jugal@illinois.edu

Martin Hoefer

Institut für Informatik, Goethe-Universität Frankfurt/Main, mhoefer@em.uni-frankfurt.de

Kurt Mehlhorn

Max-Planck-Institut für Informatik, mehlhorn@mpi-inf.mpg.de

We study linear Fisher markets with satiation. In these markets, sellers have *earning limits* and buyers have *utility limits*. Beyond applications in economics, they arise in the context of maximizing Nash social welfare when allocating indivisible items to agents. In contrast to markets with *either* earning *or* utility limits, markets with both limits have not been studied before. They turn out to have fundamentally different properties.

In general, the existence of competitive equilibria is not guaranteed. We identify a natural property of markets (termed *money clearing*) that implies existence. We show that the set of equilibria is not always convex, answering a question of [17]. We design an FPTAS to compute an approximate equilibrium and prove that the problem of computing an exact equilibrium lies in the complexity class CLS, i.e., the intersection of PLS and PPAD. For a constant number of buyers or goods, we give a polynomial-time algorithm to compute an exact equilibrium.

We show how (approximate) equilibria can be rounded and provide the first constant-factor approximation algorithm (with a factor of 2.404) for maximizing Nash social welfare when agents have capped linear (a.k.a. budget-additive) valuations. Finally, we significantly improve the approximation hardness for additive valuations to $\sqrt{8/7} > 1.069$.

Key words: Market Equilibrium, Equilibrium Computation, Nash Social Welfare, Budget-Additive Utilities
MSC2000 subject classification: Primary: 68Q25; secondary: 68W40

OR/MS subject classification: Primary: Analysis of algorithms: computational complexity; secondary: Games/group decisions: non-cooperative

History: Received June 7, 2019; revised August 2, 2020, December 1, 2021, November 14, 2022, and May 23, 2023.

1. Introduction Market equilibrium is a central solution concept in economics. It is useful to analyze and predict the outcomes of the interaction of strategic agents in large markets. Two of the most fundamental and extensively studied market models are (*Arrow-Debreu*) *exchange markets* and the special case of *Fisher markets*, introduced by Walras [56] and Fisher [9], respectively, in the late nineteenth century. Besides their importance in economics, market equilibria have also been a very fruitful domain for algorithm design – many novel algorithmic ideas have been developed in the context of computing market equilibria. In the majority of these works, the preferences of the agents are assumed to be nonsatiated, mostly in order to guarantee the existence of an

equilibrium. However, there are also a number of works that consider satiation of utilities, which appears naturally in many situations; see, e.g., [1, 5, 37, 41, 43].

Market equilibria have found many surprising applications even in non-market settings that do not involve an exchange of money. The reason is that they exhibit remarkable fairness and efficiency properties – the most prominent example is the popular fairness notion of competitive equilibrium with equal incomes (CEEI) [46].

In this paper, we study the Fisher model for markets with buyers and sellers that trade divisible goods. Buyers come to the market with money and have utility functions over allocations of goods. Given a price for each good, each buyer demands an affordable bundle of goods that maximizes his utility. At a market equilibrium, the prices are such that all goods are fully sold. Without loss of generality, we assume that each good comes in unit supply. Moreover, we assume that each single good is brought by a unique seller.¹

In a linear Fisher market, each buyer has a linear utility function, i.e., the utility of a buyer is additive over goods and scales linearly in the amount of each good received. Linear market models have been extensively studied since the 1950s [26, 30]. Recently, two natural generalizations of linear Fisher markets, based on satiation, were introduced. In these models, either *i*) buyers have utility limits or *ii*) sellers have earning limits.

In the first model, each buyer has an upper limit on the amount of utility that he wants to obtain. Each buyer spends the least amount of money on purchasing a bundle of goods that maximizes his utility up to the limit. He takes back any unused part of his money. We will refer to these utility functions as *capped linear utilities*. Equilibria in this model always exist, they can be captured by a convex program [17, 8], and there is a combinatorial polynomial-time algorithm to find an equilibrium [8].

Capped linear utilities are also known as budget-additive utilities in the literature. They represent a natural elementary case of functions with a concavity or submodularity structure. Utility functions of this form are studied frequently, e.g., in online advertising [45, 44], offline social welfare maximization [4, 6, 53, 13, 38], online algorithms [10, 22], mechanism design [11], Walrasian equilibria [51, 28], and market equilibria [8, 17].

In the second model, each seller has an upper limit on the amount of money that she wants to earn. Each seller sells the least amount of her good to earn the maximum amount of money up to the limit. She takes back any unsold portion of the good. This is a natural property and implies similar preferences for sellers as capped linear utilities imply for buyers. For further discussion and applications of earning limits, see [17]. Equilibria in this model may not always exist. However, the following are known: a necessary and sufficient condition for the existence, a convex programming formulation, and combinatorial polynomial-time algorithms for computing an equilibrium when it exists [17, 8].

The natural generalization of the two models, where both buyers and sellers have limits, was only briefly introduced in [17]. The authors posed an intriguing open question of obtaining a convex programming formulation for equilibria in this model. Markets with both utility and earning limits are the main subject of our paper. We study markets of this class that satisfy the sufficient condition for existence in the context of earning limits. In this class of markets, we show that the set of equilibria can be non-convex, thereby answering the question of [17]. We design an FPTAS to compute an approximate equilibrium and prove that the problem of computing an exact equilibrium lies in the complexity class **CLS** (Continuous Local Search), which represents the intersection of the classes **PLS** (Polynomial Local Search) and **PPAD** (Polynomial Parity Arguments on Directed Graphs) [27].

¹ For markets with linear utilities, the assumption that each agent brings a single good can be made without loss of generality. Fisher markets are sometimes defined without sellers, since their preferences are rather straightforward and can be expressed implicitly in the equilibrium conditions. Here we chose to explicitly model sellers for the goods.

For a constant number of buyers or goods, we present a polynomial-time algorithm to compute an exact equilibrium. To the best of our knowledge, this is the first market equilibrium problem that lies in **CLS** and for which no polynomial-time algorithm is known.

Beyond the economic interest in modeling buyer and seller preferences, generalized Fisher markets have also found further applications, especially for approximating the maximum Nash social welfare when allocating indivisible items to a set of agents. The Nash social welfare is defined as the geometric mean of agents' valuations, which provides an interesting trade-off between the extremal objectives of social welfare and egalitarian welfare. In social welfare, the objective is to maximize the sum of the valuations, while in egalitarian welfare, the objective is to maximize the minimum of the valuations. The Nash social welfare objective was proposed in the classic game theory literature by Nash [47] when solving the bargaining problem. It is closely related to the notion of proportional fairness studied in networking [40]. Nash social welfare satisfies a set of desirable axioms such as independence of unconcerned agents, the Pigou-Dalton transfer principle, and independence of common utility scale (see, e.g., [39, 46]). The latter implies that, in contrast to both social and egalitarian welfare, it is invariant to individual scaling of each agent valuation with possibly different constant factors.

The problem of maximizing the Nash social welfare objective is known to be **APX-hard** [42], even for additive valuations. In a remarkable result, Cole and Gkatzelis [18] gave the first constant-factor approximation algorithm for additive valuations. The constant was subsequently improved to 2 [17]. The algorithm computes and rounds an equilibrium of a Fisher market where sellers have earning limits. Moreover, the approach has been extended to provide a 2-approximation in multi-unit markets with agent valuations, which remain additive-separable over items [8], but might be concave in the number of copies received for each item [3].

In this paper, we show algorithms to compute (approximate and exact) market equilibria along with a rounding procedure. These algorithms yield the first constant-factor approximation algorithm for maximizing the Nash social welfare when agents have capped linear valuation functions. The analysis of these valuations significantly advances our understanding beyond additive-separable and towards non-separable submodular ones.

Finally, we also strengthen the existing hardness results for approximating Nash social welfare. We provide a new inapproximability bound of 1.069 that applies even in the case of *additive* valuations. This significantly improves the constant over 1.00008 in [42].

1.1. Contribution and Techniques

Money-Clearing Markets and an FPTAS We study Fisher markets with linear valuations and earning and utility limits. In markets with utility limits, a market equilibrium always exists. For markets with earning limits, equilibria exist if and only if the market satisfies a natural condition on budgets and earning limits (which we term *money clearing*). In particular, this condition holds for all market instances that arise in the context of computing approximate solutions for the Nash social welfare problem. In both markets models with *either* earning *or* utility limits, the set of equilibria is always convex.

For *both* earning *and* utility limits, we also concentrate on markets with the money-clearing condition, which we show is sufficient (but not necessary) for the existence of an equilibrium. We prove that the set of market equilibria can be *non-convex*. Hence, in contrast to the above cases, the toolbox for solving convex programs (e.g., ellipsoid [17] or scaling algorithms [18, 8]) is not directly applicable for computing an equilibrium.

Our main result is a new algorithm to compute an approximate equilibrium. Based on a constant $\varepsilon > 0$, it perturbs the valuations and rounds the parameters v_{ij} up to the next power of $(1 + \varepsilon)$. Then it computes an exact equilibrium of the perturbed market in polynomial time, which represents

an approximate equilibrium in the original market. This yields a novel FPTAS for markets with earning and utility limits. We note that the non-convexity of equilibria also applies to perturbed markets, which is surprising since we show an exact polynomial-time algorithm for computing an equilibrium.

To compute an exact equilibrium in the perturbed market, we first obtain an equilibrium (prices \mathbf{p} , allocation \mathbf{x}) of a market that results from ignoring all utility limits [17, 8]. This need not be an equilibrium of the market with both limits, because some buyers may be overspending. Let the surplus of a buyer be the money spent minus the money needed to obtain the utility limit, and similarly let the surplus of a good be the target income minus the actual income. In the precise definition of surplus below, negative values will be ruled out. We concentrate on buyers with positive surplus. Let S be the set of buyers who have a positive surplus at prices \mathbf{p} . Our idea is to pick a buyer, say k , in S , and decrease the prices of goods in a coordinated fashion. The goal is to make k 's surplus zero while keeping all zero surpluses at zero, namely those for all the goods and all the buyers not in S . We show that after a polynomial number of iterations of price decrease, either buyer k 's surplus becomes zero or we discover a good with price zero in equilibrium. Picking a particular buyer is crucial in the analysis because we rely on this buyer to show that a certain parameter strictly decreases. This guarantees a substantial price decrease and implies a polynomial running time.

Complexity of Exact Equilibria In addition to the FPTAS, we examine the complexity of computing an exact equilibrium in money-clearing markets. We show that this problem lies in $\text{PPAD} \cap \text{PLS}$. To show membership in PLS we first design a finite-time algorithm to compute an exact equilibrium. We define a finite *configuration* space such that the algorithm proceeds through a sequence of configurations. We show that configurations in the sequence do not repeat, and the algorithm terminates with an equilibrium. By defining a suitable potential function over configurations, we show that the problem is in PLS . As a refinement, whenever there are a constant number of buyers or sellers, we show that the number of configurations is polynomially bounded using a cell decomposition technique. This implies that our algorithm computes an equilibrium in polynomial time if the number of buyers or goods is constant.

For membership in PPAD we first derive a formulation as a linear complementarity problem (LCP). It captures all equilibria, but it also has non-equilibrium solutions. To discard the non-equilibrium solutions, we incorporate a positive lower bound on several variables. This turns out to be a non-trivial adjustment, because a subset of prices may be zero at all equilibria, so we must be careful not to discard equilibrium solutions. Then, we add a suitable auxiliary variable to the LCP and apply Lemke's algorithm [19]. We show that the algorithm is guaranteed to converge to an exact equilibrium under the money clearing condition. This, with a result of Todd [54], proves that the problem lies in PPAD .

Approximating Nash Social Welfare Finally, we consider the problem of maximizing Nash social welfare when allocating indivisible items to agents. We design an approximation algorithm that computes an equilibrium in a money-clearing market and rounds it to an integral allocation. Here we study the problem for agents with capped linear valuation functions. For these instances, money-clearing markets with earning and utility limits represent a natural fractional relaxation. We use our algorithms to compute an exact equilibrium (in the FPTAS for perturbed valuations). Given an exact equilibrium (for either perturbed or original valuations), we provide a rounding algorithm that turns the fractional allocation into an integral one. While the algorithm exploits a tree structure of the equilibrium allocation as in [18], the rounding becomes much more challenging, and we must be careful to correctly treat agents that reach their utility limits in the equilibrium. In particular, we first conduct several initial assignment steps to arrive at a solution where we have

a set of rooted trees on agents and items, and each item j has exactly one child agent i who gets at least half of the fractional valuation from j . In the main step of the rounding algorithm, we need to ensure that the root agent r receives one of his child items. Here we pick a child item j that generates the most value for r . A problem arises at the child agent i of j since r receiving j could decrease i 's valuation by a lot more than a factor of 2. Recursively, we again need to enforce an allocation for agent i , thereby “stealing” fractional value from one of i 's grandchildren agents, and so on. It may seem as if there is no hope for this approach to yield any reasonable approximation guarantee, but we show that in aggregate the agents only suffer a small constant-factor loss.

Our analysis of this rounding procedure provides a lower bound on the Nash social welfare obtained by the algorithm, which is complemented with an upper bound on the optimum solution. Both bounds crucially exploit the properties of agents (goods) that reach the utility (earning) limits in the market equilibrium. These bounds imply an approximation factor of $2e^{1/(2e)} < 2.404$. Since the equilibrium conditions apply with respect to perturbed valuations, we obtain a $(2e^{1/(2e)} + \varepsilon)$ -approximation in polynomial time, for any constant $\varepsilon > 0$.

In terms of lower bounds, we strengthen the existing inapproximability bound to $\sqrt{8/7} > 1.069$. Our improvement is based on the construction for the hardness of social welfare maximization for capped linear valuations from [13]. For the Nash social welfare objective, we observe how to drop the utility limits and apply the construction even for additive valuations.

A preliminary version of this paper appeared at the 29th ACM-SIAM Symposium on Discrete Algorithms (SODA 2018) [31].

1.2. Related Work

Market Equilibria The problem of computing market equilibria is an intensely studied problem, so we restrict our discussion to previous work that appears most relevant.

For linear Fisher markets, equilibria are captured by the Eisenberg-Gale convex program [26]. Later, Shmyrev [52] obtained another convex program for this problem. Cole et al. [17] provide a dual connection between these and other convex programs. A combinatorial polynomial-time algorithm for computing equilibrium in this model was obtained by Devanur et al. [23]. Orlin [49] gave the first strongly polynomial-time algorithm using a scaling technique. More recently, Végh [55] gave a different scaling-based algorithm that also runs in strongly polynomial time.

Fisher markets are a special case of the more general Arrow-Debreu exchange markets. There are many convex programming formulations for linear exchange markets; see [21] for details. The first polynomial-time algorithm was obtained by Jain [36] based on the ellipsoid method. Ye [57] obtained a polynomial-time algorithm based on the interior-point method. Duan and Mehlhorn [25] developed the first combinatorial polynomial-time algorithm, which was later improved in [24]. More recently, Garg and Végh [35] obtained the first strongly polynomial-time algorithm for this problem.

Linear Fisher markets with either utility or earning limits were studied only recently [17, 8], and equilibria in these models can be captured by extensions of Eisenberg-Gale and Shmyrev convex programs, respectively. In markets with utility limits, combinatorial polynomial-time algorithms are obtained in [8, 55], the set of equilibria forms a lattice, and equilibria with maximum or minimum prices can also be obtained efficiently [8]. In markets with earning limits, combinatorial polynomial-time algorithms are obtained in [18, 8]. Any equilibrium can be refined to one with minimal or maximal prices in polynomial time [8].

Nash Social Welfare The Nash social welfare is a classic objective for allocating goods to agents. Nash [47] proposed it for the bargaining problem as the unique objective that satisfies a collection of natural axioms. Since then, it has received significant attention in the literature on social choice and fair division (see, e.g., [12, 20, 50, 29] for a subset of notable recent work, and the references therein).

For divisible items, the problem of maximizing the Nash social welfare is solved by competitive equilibria with equal incomes (CEEI) [46]. However, CEEI can provide significantly more value in terms of Nash social welfare than optimal solutions for indivisible items. To obtain an improved bound on the indivisible optimum, Cole and Gkatzelis [18] introduced and rounded *spending-restricted equilibria*, i.e., equilibria in markets with an earning limit of 1 for every good. More generally, equilibria in linear markets with earning limits can be described by a convex program [17] similar to the one by Shmyrev.

For indivisible items and general non-negative valuations, the problem of maximizing the Nash social welfare is hard to approximate within any finite factor [48]. For additive valuations, the problem is APX-hard [42], and efficient 2-approximation algorithms based on market equilibrium [18, 17] and stable polynomials [2, 3] exist. These algorithms have been extended to give a 2-approximation in markets with multiple copies per item [8] and additive-separable concave valuations [3]. Barman et al. [7] introduced another technique based on limited envy and obtained a 1.45-approximation for additive valuations. Very recently, Chaudhury et al. [14] generalized this result to obtain a 1.45-approximation for a common generalization of both capped linear and additive-separable concave valuations.

1.3. Outline The rest of the paper is structured as follows. We introduce notation and preliminaries in Section 2. In Section 3.1, we discuss the existence of market equilibria under the money clearing condition. The FPTAS for perturbed markets is discussed in Section 3.2. The following sections contain our results on computing exact equilibria – membership in PLS (Section 3.3), the polynomial-time algorithms for a constant number of buyers or goods (Section 3.4), and membership in PPAD (Section 3.5). The rounding algorithm for maximizing the Nash social welfare and the analysis of its approximation factor are presented in Section 4.1. In Section 4.2, we present the improved hardness bound for the approximation of Nash social welfare with additive valuations. Finally, we conclude in Section 5 with a discussion of directions for future research.

2. Preliminaries

Fisher Markets with Earning and Utility Limits In such a market, there is a set B of n *buyers* and a set G of m *divisible goods*. Each good is owned by a separate seller and comes in unit supply. Each buyer $i \in B$ has a *utility value* $u_{ij} \geq 0$ for a unit of good $j \in G$ and a *budget* $m_i > 0$ of money. Suppose buyer i receives a bundle of goods $\mathbf{x}_i = (x_{ij})_{j \in G}$ with $x_{ij} \in [0, 1]$, then the *utility function* is capped linear if $u_i(\mathbf{x}_i) = \min(c_i, \sum_j u_{ij}x_{ij})$, where $c_i > 0$ is the *utility cap*.

The vector $\mathbf{x} = (\mathbf{x}_i)_{i \in B}$ with $\sum_{i \in B} x_{ij} = 1$ for every $j \in G$ denotes a *(fractional) allocation* of goods to buyers. For an allocation, we call i a *capped buyer* if $u_i(\mathbf{x}_i) = c_i$. We also maintain a vector $\mathbf{p} = (p_1, \dots, p_m)$ of *prices* for the goods. Given price p_j for good j , a buyer needs to pay $x_{ij}p_j$ when getting x_{ij} allocation of good $j \in G$. Given a vector of prices \mathbf{p} , a *demand bundle* \mathbf{x}_i^* of buyer i is an affordable bundle of goods that maximizes the utility of buyer i , i.e., $\mathbf{x}_i^* \in \arg \max_{\mathbf{x}_i} \{u_i(\mathbf{x}_i) \mid \sum_j p_j x_{ij} \leq m_i\}$. For price vector \mathbf{p} and buyer i , we let $\lambda_i = \min_j p_j / u_{ij}$ and denote by $\alpha_i = 1 / \lambda_i$ the *maximum bang-per-buck (MBB)* ratio (where we assume $0/0 = 0$). Given prices \mathbf{p} and allocation \mathbf{x} , the *money flow* f_{ij} from buyer i to seller j is given by $f_{ij} = p_j x_{ij}$. If price $p_j > 0$, then x_{ij} uniquely determines f_{ij} and vice versa.

For each seller j , let $x_j = \sum_i x_{ij}$; then the seller utility is $u_j(x_j, p_j) = \min(d_j, p_j x_j) = \min(d_j, \sum_i f_{ij})$, where $d_j > 0$ is the *earning or income cap*. We call seller j a *capped seller* if $u_j(x_j, p_j) = d_j$. An *optimal supply* e_j^* allows seller j to obtain the highest utility, i.e., $e_j^* \in \arg \max \{u_j(e_j, p_j) \mid e_j \leq 1\}$.

We assume that all parameters of the market, u_{ij} , c_i , d_j and m_i , for all $i \in B$ and $j \in G$, are non-negative integers. Let $U = \max_{i \in B, j \in G} \{u_{ij}, m_i, c_i, d_j\}$ be the largest integer in the representation of the market.

We consider three natural properties for allocation and supply vectors:

1. An allocation \mathbf{x}_i for buyer i is called *modest* if $\sum_j u_{ij}x_{ij} \leq c_i$. By definition, for uncapped buyers every demand bundle is modest. For capped buyers, if a bundle of goods is modest, then $c_i = \sum_j u_{ij}x_{ij}$.
2. A demand bundle \mathbf{x}_i is called *thrifty* or *MBB* if it consists only of MBB goods: $x_{ij} > 0$ only if $u_{ij}/p_j = \alpha_i$. For uncapped buyers every demand bundle is MBB.
3. A supply e_j for seller j is called *modest* if $e_j = \min(1, d_j/p_j)$.

Given a set of prices, a thrifty and modest demand bundle for buyer i minimizes the amount of money required to obtain optimal utility. A modest supply for seller j minimizes the amount of supply required to obtain optimal utility in equilibrium. Our interest lies in market equilibria that have thrifty and modest demands and modest supplies. Note that they also emerge when earning and utility caps are not satiation points but *limits in the form of hard constraints* on the utility in equilibrium (c.f. [17]).

DEFINITION 2.1 (THRIFTY AND MODEST EQUILIBRIUM). A *thrifty and modest (market) equilibrium* is a pair (\mathbf{x}, \mathbf{p}) , where \mathbf{x} is an allocation and \mathbf{p} a vector of *prices* such that the following conditions hold: (1) $\mathbf{p} \geq 0$ (prices are nonnegative), (2) e_j is a modest supply for every $j \in G$, (3) $x_j \leq e_j$ for every $j \in G$ (no overallocation), (4) \mathbf{x}_i is a thrifty and modest demand bundle for every $i \in B$, and (5) Walras' law holds: $p_j(e_j - x_j) = 0$ for every $j \in G$.

Note that in equilibrium, if $x_j < e_j$, then $p_j = 0$, due to Walras' law.

Consider the following condition termed *money clearing*: For each subset of buyers and the goods these buyers are interested in, there is a feasible allocation of the buyer money that does not violate the earning caps. More formally, let $\hat{B} \subseteq B$ be a set of buyers, and $N(\hat{B}) = \{j \in G \mid u_{ij} > 0 \text{ for some } i \in \hat{B}\}$ be the set of goods such that there is at least one buyer in \hat{B} with positive utility for the good.

DEFINITION 2.2 (MONEY CLEARING). A market is *money clearing* if

$$\sum_{i \in \hat{B}} m_i \leq \sum_{j \in N(\hat{B})} d_j, \quad \text{for all } \hat{B} \subseteq B. \quad (1)$$

The notion of money clearing has been used in previous work. In particular, when there are only earning limits, money clearing is a precise characterization of markets that have thrifty and modest equilibria [8]. For markets with both limits, it is sufficient for existence (see Section 3.1).

Perturbed Markets Our FPTAS in Section 3.2 computes a thrifty and modest equilibrium in a *perturbed market* $\widetilde{\mathcal{M}}$.

DEFINITION 2.3 (PERTURBED UTILITY, PERTURBED MARKET). For a market \mathcal{M} and a parameter $\varepsilon > 0$, the *perturbed utility* of buyer i is given by $\tilde{u}_i(\mathbf{x}_i) = \sum_j \tilde{u}_{ij}x_{ij}$, where $\tilde{u}_{ij} \in \{0, (1 + \varepsilon)^k \mid \text{integer } k \geq 0\}$ such that

$$\tilde{u}_{ij}/(1 + \varepsilon) \leq u_{ij} \leq \tilde{u}_{ij}, \quad \text{for all } i \in B, j \in G. \quad (2)$$

The *perturbed market* $\widetilde{\mathcal{M}}$ is exactly the market \mathcal{M} in which every buyer $i \in B$ has perturbed utilities \tilde{u}_i . We define $\tilde{U} = \max_{i \in B, j \in G} \tilde{u}_{ij}$ as the largest perturbed utility.

In Section 3.2 we observe that an exact equilibrium in $\widetilde{\mathcal{M}}$ is an ε -approximate equilibrium for the unperturbed market \mathcal{M} .

Nash Social Welfare There is a set B of n *agents* and a set G of m *indivisible items*, where we assume $m \geq n$. We allocate the items to the agents, and we represent an allocation $S = (S_1, \dots, S_n)$ using a characteristic vector \mathbf{x}^S with $x_{ij}^S = 1$ iff $j \in S_i$ and 0 otherwise. Agent $i \in B$ has a *value* $v_{ij} \geq 0$ for item j and a global *utility cap* $c_i > 0$. The *budget-additive* or *capped linear valuation* of agent i for an allocation S of items is $v_i(\mathbf{x}_i^S) = \min(c_i, \sum_{j \in G} v_{ij} x_{ij}^S)$. The goal is to find an allocation that approximates the optimal Nash social welfare, i.e., the optimal geometric mean of valuations

$$\max_S \left(\prod_{i \in B} v_i(\mathbf{x}_i^S) \right)^{1/n}.$$

Our approximation algorithm in Section 4 relies on rounding an equilibrium for a linear Fisher market with earning and utility limits. Our rounding algorithm deteriorates the Nash social welfare only by a constant factor. More precisely, we round an exact equilibrium of the perturbed market $\tilde{\mathcal{M}}$. The fact that this equilibrium satisfies the properties in Definition 2.1 with respect to perturbed utilities increases the approximation factor only by a small constant (see Section 4.1.3).

3. Computing Equilibria

3.1. Existence and Structure of Equilibria In this section, we briefly discuss the existence and structure of thrifty and modest equilibria in markets with utility and earning limits. The set of equilibria in these markets has interesting and non-trivial structure. For markets with utility limits, an equilibrium always exists [8]. For markets with earning limits, an equilibrium may not exist, because uncapped buyers always spend all their money. In these markets, the money-clearing condition is necessary and sufficient for the existence of a thrifty and modest equilibrium [8] (see also [17] for the case that $u_{ij} > 0$ for all $i \in B, j \in G$).

We observe that in a market \mathcal{M} with both limits, money clearing is sufficient but not necessary for the existence of a thrifty and modest equilibrium. Our FPTAS below gives an ε -approximate equilibrium in money-clearing markets, for arbitrarily small ε . Since all market parameters are assumed to be finite integers, for sufficiently small ε this implies the existence of an exact equilibrium.

This is interesting since the structure of equilibria in such markets can be quite complex. For example, in money-clearing markets \mathcal{M} there can be no convex program describing thrifty and modest equilibria. This holds even if we restrict to the ones that are Pareto-optimal with respect to the set of all thrifty and modest equilibria. Equilibria for the corresponding markets *without* caps, or with *either* earning *or* utility caps might not remain equilibria in the market with *both* sets of caps. Hence, the existence of a thrifty and modest equilibrium in money-clearing markets \mathcal{M} follows neither from a convex program nor by a direct application of existing algorithms for markets with only one set of either utility or earning caps. The following proposition summarizes our observations.

PROPOSITION 3.1. *There are markets \mathcal{M} with utility and earning limits such that the following hold:*

1. \mathcal{M} is not money-clearing and has a thrifty and modest equilibrium.
2. \mathcal{M} is money-clearing, and the set of thrifty and modest equilibria is not convex. Among these equilibria, there are multiple Pareto-optimal equilibria, and their set is also not convex.
3. For a money-clearing market \mathcal{M} and the three related markets – (1) with only utility caps, (2) with only earning caps, (3) without any caps – the sets of equilibria are mutually disjoint.

Proof. We provide an example market for each of the three properties.

Property 1: Consider a linear market with one buyer and one good. The buyer has $m_1 = 2$, utility $u_{11} = 2$, and utility cap $c_1 = 1$. The good has earning cap $d_1 = 1$. The unique thrifty and modest equilibrium has price $p_1 = 2$ and allocation $x_{11} = 1/2$. The income of the seller equals the earning cap. Due to price 2, the supply is 1/2, for which the achieved utility equals the utility cap. Both seller and buyer exactly reach their caps and obtain an optimal utility. Note that the money clearing condition (1) is violated.

Property 2: Consider the following example. There are two buyers and two goods. The buyer budgets are $m_1 = 2$ and $m_2 = 32$. The utility caps are $c_1 = \infty$, $c_2 = 32$, the earning caps are $d_1 = 8$, $d_2 = 26$. The linear utilities are given by the parameters $u_{11} = u_{22} = 32$, $u_{12} = 128$, and $u_{21} = 2$. Note that the total available money $2 + 32$ equals the sum of the earning caps $8 + 26$.

Consider any equilibrium (\mathbf{x}, \mathbf{p}) . We use b_1 and b_2 to refer to buyers 1 and 2, respectively. If both earning caps are met, the total spending equals the available money. Thus, b_2 needs to spend money on both goods, and we have $p_2 = 16p_1$. Hence, b_1 spends only on good 1, so the money flow must be $f_{11} = 2$, $f_{12} = 0$, $f_{21} = 6$, and $f_{22} = 26$. Since the earning cap $d_1 = 8$, we have $p_1 \geq 8$. With prices $(8y, 128y)$ and $y \geq 1$, the buyer utilities are $(8/y, 8/y)$.

We turn to the case that some earning cap is not met and, as a consequence, not all money is spent. As b_1 has utility cap ∞ , b_2 must not be spending in full. So b_2 must reach his utility cap of 32. Also, any good not meeting its earning cap must be sold in full. We now distinguish cases according to which buyer buys which goods.

b_1 buys some of both goods: Then $p_2 = 4p_1$ and b_2 only buys the second good. b_1 buys some of good 2, so b_2 does not buy all of good 2 and does not reach his utility cap, a contradiction.

b_2 buys some of both goods: Then $p_2 = 16p_1$. Since b_1 spends all money on the first good and b_2 buys some of the first good, $p_1 > 2$ and hence $p_2 > 32$. Thus, the bang-per-buck ratio of b_2 is less than 1. Hence, b_2 cannot reach his utility cap, a contradiction.

Each buyer buys exactly one of the goods: It is impossible that b_1 only buys the second good and b_2 only buys the first good because $p_2 \leq 4p_1$ in the former case and $p_2 \geq 16p_1$ in the latter case. Thus, b_1 only buys the first good and b_2 only buys the second good, and hence $4p_1 \leq p_2 \leq 16p_1$. As $m_1 = 2$, the earning cap for the first good is not met, and so the first good is sold entirely. The price of the first good must be such that b_1 spends all money and the first good is completely sold, i.e., $p_1 = 2$ and hence $8 \leq p_2 \leq 32$. Since $m_2 = 32$, $u_{22} = 32$, and $c_2 = 32$, b_2 will buy as much of good 2 as he possibly can. Since the second good has an earning cap of 26, $p_2 \leq 26$. So the possible equilibria have prices $(2, y)$ with $8 \leq y \leq 26$. The utilities are $(32, 32)$ and both goods are completely allocated.

We summarize the discussion: The equilibria form two disjoint convex sets, either prices $(2, y)$ and buyer utilities $(32, 32)$, for $y \in [8, 26]$; or prices $(8y, 128y)$ and buyer utilities $(8/y, 8/y)$, for $y \geq 1$.

There are exactly two Pareto-optimal equilibria: prices $(2, 8)$ (which also represents the income for the sellers) and buyer utilities $(32, 32)$; and prices $(8, 128)$ (with income $(8, 26)$ for the sellers) and buyer utilities $(8, 8)$. The first equilibrium is strictly better for both buyers, the second one strictly better for both sellers.

Property 3: Consider the following market with 2 buyers and 2 goods. The buyer budgets are $m_1 = 100$ and $m_2 = 11$. The utility caps are $c_1 = 0.9$, $c_2 = \infty$. The earning caps are $d_1 = 9$, $d_2 = \infty$. The utilities are $u_{11} = u_{22} = u_{12} = u_{21} = 1$.

First note that in all equilibria, the prices are equal as all the utilities are the same. If we ignore all caps, the price p of each good is $(\text{total budget})/2 = 55.5$. If we ignore the utility caps and consider only earning caps, we have $p = (\text{total budget}) - (\text{cap of good 1}) = 102$. If we ignore the earning caps and consider only utility caps, we have $0.9p + 11 = 2p$, resulting in $p = 10$. With both utility and earning caps, we have $\text{spending} = 0.9p + 11 = \text{income} = 9 + p$, resulting in $p = 20$. \square

3.2. Computing Equilibria in Perturbed Markets In this section, we describe and analyze Algorithm 1, an FPTAS for computing an approximate equilibrium in money-clearing markets \mathcal{M} . The input parameters for such a market are u_{ij}, m_i, c_i, d_j , for all $i \in B$ and $j \in G$, where u_{ij} is the utility derived by buyer i for a unit amount of good j , m_i is the budget of buyer i , c_i is the utility cap of buyer i , and d_j is the earning cap of seller j . For any $\varepsilon > 0$, Algorithm 1 computes an exact equilibrium in a perturbed market $\tilde{\mathcal{M}}$, where we increase every non-zero parameter u_{ij} to the next-larger power of $(1 + \varepsilon)$.

Additional Concepts Our algorithm steers prices and money flow towards equilibrium by monitoring the surplus of buyers and sellers. Note that a buyer i is capped if $m_i \alpha_i \geq c_i$.

DEFINITION 3.1 (ACTIVE BUDGET AND SUPPLY, INCOME, SURPLUS). Given prices \mathbf{p} and flow \mathbf{f} , the *active budget* of buyer i is $m_i^a = \min(m_i, c_i/\alpha_i)$, and the *active supply* of seller j is $e_j^a = \min(1, d_j/p_j)$. The *active price* $p_j^a = p_j e_j^a = \min(p_j, d_j)$ is the *income* of seller j . The *surplus of buyer i* is $s(i) = \sum_{j \in G} f_{ij} - m_i^a$, and the *surplus of good j* is $s(j) = p_j^a - \sum_{i \in B} f_{ij}$.

Several graphs connected to the MBB ratio are useful here.

DEFINITION 3.2 (MBB GRAPH, MBB EDGE, MBB RESIDUAL GRAPH). Given prices \mathbf{p} , we define the *MBB graph* $\mathcal{G}(\mathbf{p}) = (B \cup G, E)$, an undirected bipartite graph with node sets composed of buyers and sellers. The set of edges E are all *MBB edges*, where an undirected pair $\{i, j\}$ is an MBB edge if $i \in B$, $j \in G$, and $u_{ij}/p_j = \alpha_i$. Given prices \mathbf{p} and money flow \mathbf{f} , the *MBB residual graph* $\mathcal{G}_r(\mathbf{f}, \mathbf{p}) = (B \cup G, A)$ is a directed graph with the following arcs: If $\{i, j\}$ is MBB, then (i, j) is an arc in A ; if $\{i, j\}$ is MBB and $f_{ij} > 0$, then (j, i) is an arc in A .

As argued in [49, 24], we can assume without loss of generality that the MBB graph is *non-degenerate*, i.e., it is a forest.

In the MBB residual graph, we interpret the money flow \mathbf{f} to originate at the buyers and to flow to the goods, where all edges have infinite capacity. Edges (i, j) and (j, i) in the MBB residual graph indicate that flow from $i \in B$ to $j \in G$ can be increased and decreased, respectively. In contrast, let us also define a *reverse flow network* $N^-(\mathbf{p}, Z)$, where \mathbf{p} is a vector of prices and $Z \subseteq B$ a subset of buyers. $N^-(\mathbf{p}, Z)$ is constructed by adding a sink t to the MBB graph. The network has nodes $G \cup B \cup \{t\}$, edges (i, t) for $i \in B \setminus Z$, and the reverse MBB edges (j, i) if $\{i, j\}$ is an MBB edge. All edges have infinite capacity. The supply at node $j \in G$ is p_j^a , demand at node $i \in B$ is m_i^a , and demand at node t is $\sum_j p_j^a - \sum_i m_i^a$. The flow in the network corresponds to money. It originates at the *goods* and flows to the *buyers* (and possibly further to the sink t). Given a money flow \mathbf{f} in the network $N^-(\mathbf{p}, Z)$, the surplus of buyer $i \in B \setminus Z$ corresponds to flow on (i, t)

$$s(i) = \sum_{j \in G} f_{ji} - m_i^a = f_{it} .$$

Buyers in Z do not have edges to the sink. Hence, their surplus is fixed to 0 for every feasible flow.

Algorithm and Invariants Algorithm 1 starts by finding a min-price equilibrium $(\mathbf{x}^{min}, \mathbf{p}^{min})$ with utility caps but no earning caps. A min-price equilibrium has coordinate-wise smallest prices, i.e., for every good j the price p_j^{min} is the smallest price of good j in all equilibria. Such an equilibrium can be computed in polynomial time using the algorithm of [8]. The algorithm then removes the set S of goods with price 0 at \mathbf{p}^{min} and the set $\Gamma(S)$ of buyers having positive utility for a good in S . Note that buyers in $\Gamma(S)$ are capped and only buy the goods in S at $(\mathbf{x}^{min}, \mathbf{p}^{min})$. The algorithm sets the prices of goods in S to zero and the allocation of buyers in $\Gamma(S)$ as per \mathbf{x}^{min} , and then it proceeds with the remaining set of buyers and goods. Lemma 3.9 (in Section 3.5) shows that the price of each remaining good j in any equilibrium with both utility and earning caps is at least $p_j^{min} \geq 1/(n\tilde{U}^n)$. Recall that $\tilde{U} = \max_{i,j} \tilde{u}_{ij}$ is the largest perturbed utility.

For convenience, the algorithm maintains a money flow \mathbf{f} . For goods with non-zero prices, \mathbf{f} is equivalent to an allocation \mathbf{x} . Next, it finds an equilibrium with earning caps but no utility caps.² Such an equilibrium exists because the market is money-clearing. The equilibrium can be computed in polynomial time [17, 8] and consists of a pair (\mathbf{f}, \mathbf{p}) of flow and prices such that the outflow of every good j is p_j^a and the inflow of every buyer i is m_i . Given this equilibrium, the algorithm then initializes Z to the set of buyers with zero surplus in (\mathbf{f}, \mathbf{p}) .

There are two possible approaches at this point: to increase prices or to reduce them. Raising prices does not work, e.g., consider two buyers b_1 and b_2 and a good g . Both buyers have a budget of 1 and utility 1 for the entire good g , and they have a utility cap of $1/4$ and ∞ , respectively. The earning cap of g is ∞ . The starting price of g is 2 if we ignore the utility caps, and clearly, it is not possible to find an equilibrium with both utility and earning caps by raising the prices.

Consequently, we proceed by reducing prices in a controlled way. More specifically, the algorithm proceeds in a series of iterations. In each iteration, prices are reduced on a carefully chosen subset of goods, and for capped buyers³ interested in these goods, budgets are reduced in proportion. The idea is that uncapped buyers will be able to buy more goods, displacing capped buyers who need to reduce their spending.

The following **Invariants** are maintained during the run of Algorithm 1:

- no price ever increases.
- if $s(i) = 0$ for a buyer i , it remains 0. Z is monotonically increasing.
- $N^-(\mathbf{p}, Z)$ allows a feasible flow, i.e., $s(i) \geq 0$ for every buyer $i \in B$ and $s(j) = 0$ for every good $j \in G$.

LEMMA 3.1. *The Invariants hold during the run of Algorithm 1.*

The lemma is quite straightforward to see. Towards a proof, we explain the algorithm in more technical detail. The algorithm uses a descending-price approach. There is always a flow in $N^-(\mathbf{p}, Z)$ with outflow of a good $j \in G$ equal to p_j^a , in-flow into buyer $i \in B \cap Z$ equal to m_i^a , and in-flow into buyer $i \in B \setminus Z$ at least m_i^a . These imply that if a good (buyer) becomes uncapped (capped), it remains uncapped (capped).

The algorithm ends when $Z = B$, i.e., all buyers have zero surplus, and hence (\mathbf{f}, \mathbf{p}) is an equilibrium of \mathcal{M} . In the body of the outer while-loop, we first pick a buyer k whose surplus is positive. The inner while loop ends when the surplus of k becomes zero. This increases the size of Z (in line 19).

In the body of inner while-loop, we construct the set \hat{B} of buyers and \hat{G} of goods that can reach buyer k in the MBB residual graph (see Definition 3.2). We then continuously decrease the prices of all goods in \hat{G} by a common factor γ , starting from $\gamma = 1$. This may destroy MBB edges connecting buyers in \hat{B} with goods in $G \setminus \hat{G}$. However, by the definition of \hat{G} there is no flow on such edges. For uncapped goods in \hat{G} (capped buyers in \hat{B}), this decreases the active price (budget) by a factor of γ . We stop if one of the two events happens: (1) a new MBB edge appears, or (2) γ is equal to the minimum factor possible that allows a feasible flow with the current MBB edges, i.e., in-flow into a good $j \in \hat{G}$ is equal to p_j^a , outflow from a buyer in $\hat{B} \cap Z$ is equal to m_i^a , and outflow from a buyer in $\hat{B} \setminus Z$ is at least m_i^a . While the value of γ for event (1) results from ratios of \tilde{u}_{ij} , the value of γ for event (2) is found by Algorithm 2 based on a linear program (LP). Observe that the flow \mathbf{f} and $\gamma = 1$ are a feasible initial solution for the LP.

After an event happened, we update to a new feasible flow \mathbf{f} using Algorithm 3. For prices \mathbf{p} and the set Z of zero-surplus buyers, the in-flow into a good $j \in G$ must be equal to p_j^a , the

² We remark that an alternate approach may also work that starts from an equilibrium with utility caps but no earning caps, and then proceeding analogously by *raising* prices to find an equilibrium with both utility and earning caps.

³ Recall that a buyer is capped if his current utility equals his cap.

Algorithm 1: FPTAS for \mathcal{M} with Earning and Utility Caps

Input : Market \mathcal{M} given by budgets m_i , utility caps c_i , earning caps d_j , utilities u_{ij} , for all $i \in B$ and $j \in G$, approximation parameter ε ;

Output: Equilibrium (\mathbf{x}, \mathbf{p}) of the perturbed market $\tilde{\mathcal{M}}$

- 1 Construct $\tilde{\mathcal{M}}$ by increasing each non-zero u_{ij} to the next-larger power of $(1 + \varepsilon)$, set $\tilde{U} \leftarrow \max_{ij} \tilde{u}_{ij}$, and run the rest of the algorithm on $\tilde{\mathcal{M}}$
- 2 $(\mathbf{x}^{\min}, \mathbf{p}^{\min}) \leftarrow$ min-price equilibrium of $\tilde{\mathcal{M}}$ when ignoring all earning caps
- 3 $S \leftarrow \{j \in G \mid p_j^{\min} = 0\}$ and $\Gamma(S) \leftarrow \{i \in B \mid u_{ij} > 0, j \in S\}$
- 4 Set $p_j \leftarrow 0$, for all $j \in S$ and $x_{ij} \leftarrow x_{ij}^{\min}$, for all $i \in \Gamma(S)$ and $j \in S$
- 5 Remove goods in S and buyers in $\Gamma(S)$ from $\tilde{\mathcal{M}}$
- 6 $(\mathbf{f}, \mathbf{p}) \leftarrow$ equilibrium of $\tilde{\mathcal{M}}$ when ignoring all utility caps // $s(j) = 0, \forall j \in G; s(i) \geq 0, \forall i \in B$
- 7 $Z \leftarrow \{i \in B \mid s(i) = 0\}$ // set of zero surplus buyers
- 8 **while** $Z \neq B$ **do**
- 9 $k \leftarrow$ a buyer in $B \setminus Z$ // $s(k) > 0$
- 10 **while** $(s(k) > 0)$ **do**
- 11 $\hat{B} \leftarrow \{k\} \cup \{i \in B \mid i \text{ can reach } k \text{ in the MBB residual graph}\}$
- 12 $\hat{G} \leftarrow \{j \in G \mid j \text{ can reach } k \text{ in the MBB residual graph}\}$
- 13 $\mathbf{p}' \leftarrow \mathbf{p}$ and $\gamma \leftarrow 1$
- 14 Set $p_j \leftarrow \gamma \cdot p_j$, for all $j \in \hat{G}$
- 15 Decrease γ continuously down from 1 until one of these events occurs:
- 16 **Event 1:** A new MBB edge appears
- 17 **Event 2:** $\gamma = \text{MinFactor}(\mathbf{p}', \mathbf{f}, \hat{B}, \hat{G}, Z)$ // Algorithm 2
- 18 $\mathbf{f} \leftarrow \text{FeasibleFlow}(\mathbf{p}, Z)$ // Algorithm 3
- 19 $Z \leftarrow Z \cup \{i \in B \mid s(i) = 0\}$
- 20 Assign \mathbf{x}_i according to \mathbf{f} for all buyers $i \in B$ that have not been assigned yet.
- 21 **return** (\mathbf{x}, \mathbf{p})

outflow from a buyer in Z must be equal to m_i^a , and the outflow from a buyer in $B \setminus Z$ must be at least m_i^a . Algorithm 3 sets up a feasibility LP to find such a feasible flow. When being invoked by Algorithm 1, Algorithm 3 is guaranteed to be able to return a feasible flow.

Running Time We bound the running time of Algorithm 1. Event 1 provides a new MBB edge between a buyer in $B \setminus \hat{B}$ and a good in \hat{G} . Event 2 restricts the price decrease in γ ensuring that the invariants are maintained. Event 2 happens under two conditions: First, when the input configuration changes, i.e., either the set of capped buyers or the set of uncapped goods increases, and second, when further decrease would violate one of the invariants. These are captured in the LP (Algorithm 2) through $\gamma \lambda_i c_i \geq m_i$ and $\gamma p_j \geq d_j$, respectively. Observe that the remaining inequalities of the LP precisely capture feasible flows in $N^-(\mathbf{p}, Z)$ when varying prices of goods in \hat{G} using γ . This implies that the only possibility for the second condition is that $N^-(\mathbf{p}, Z)$ no longer has a feasible flow. At such a critical value of γ , (1) there is a subset of buyers $S \subseteq \hat{B}$ such that $\sum_{i \in S} m_i^a = \sum_{j \in \Gamma(S)} p_j^a$, where $\Gamma(S)$ is the set of goods to which buyers in S have MBB edges, and (2) further decrease of prices would make the total active budget of buyers in S larger than the total active prices of $\Gamma(S)$. Observe that the first condition of Event 2 can happen at most $m + n$ times during the entire run of the algorithm. In order to show polynomial running time, it suffices to bound the number of instances of Event 2 occurring under the second condition.

Since $\sum_{i \in S} m_i^a \leq \sum_{i \in S, j \in \Gamma(S)} f_{ij} \leq \sum_{j \in \Gamma(S)} p_j^a$, when equality holds, the buyers in S all have zero surpluses. If the subset S contains buyer k , then the surplus of k in every feasible flow is zero

Algorithm 2: MinFactor

Input : Prices \mathbf{p} , flow \mathbf{f} , set of buyers \hat{B} , set of goods \hat{G} , set of zero-surplus buyers Z
Output: Smallest price parameter γ s.t. prices are consistent with the input configuration

- 1 $E \leftarrow$ set of MBB edges at prices \mathbf{p} between \hat{B} and \hat{G}
- 2 $\tilde{G}_c \leftarrow$ set of goods from \hat{G} that are capped at (\mathbf{f}, \mathbf{p})
- 3 $\tilde{B}_c \leftarrow$ set of buyers from \hat{B} that are capped at (\mathbf{f}, \mathbf{p})
- 4 $\lambda_i \leftarrow \min_{k \in G} p_k / u_{ik}$, for all $i \in \hat{B}$
- 5 Solve the following LP in flow variables \mathbf{g} and γ :

$\min \gamma$ s.t.

$$\begin{aligned} \sum_{i \in \hat{B}} g_{ij} &= d_j, & \text{for all } j \in \tilde{G}_c \\ \sum_{i \in \hat{B}} g_{ij} &= \gamma p_j, & \text{for all } j \in \hat{G} \setminus \tilde{G}_c \\ \gamma p_j &\geq d_j, & \text{for all } j \in \tilde{G}_c \\ \sum_{j \in \hat{G}} g_{ij} &= \gamma c_i \lambda_i, & \text{for all } i \in \tilde{B}_c \cap Z \\ \sum_{j \in \hat{G}} g_{ij} &\geq \gamma c_i \lambda_i, & \text{for all } i \in \tilde{B}_c \setminus Z \\ \sum_{j \in \hat{G}} g_{ij} &= m_i, & \text{for all } i \in \hat{B} \setminus \tilde{B}_c \\ \gamma \lambda_i c_i &\geq m_i, & \text{for all } i \in \hat{B} \setminus \tilde{B}_c \\ g_{ij} &= 0, & \text{for all } \{i, j\} \in (\hat{B} \times \hat{G}) \setminus E \\ g_{ij} &\geq 0, & \text{for all } i \in \hat{B}, j \in \hat{G} \end{aligned}$$

- 6 **return** Optimal solution γ of above LP

Algorithm 3: FeasibleFlow

Input : Perturbed market $\tilde{\mathcal{M}}$, prices \mathbf{p} , and set of zero-surplus buyers Z
Output: Feasible flow consistent with the input configuration

- 1 $E \leftarrow$ set of MBB edges at prices \mathbf{p}
- 2 $\lambda_i \leftarrow \min_{k \in G} p_k / u_{ik}$, for all $i \in B$
- 3 $B_c \leftarrow$ set of capped buyers at \mathbf{p}
- 4 $G_c \leftarrow$ set of capped goods at \mathbf{p}
- 5 Solve the following feasibility LP in flow variables \mathbf{f} :

$$\begin{aligned} \sum_{i \in \hat{B}} f_{ij} &= d_j, & \text{for all } j \in G_c \\ \sum_{i \in \hat{B}} f_{ij} &= p_j, & \text{for all } j \in G \setminus G_c \\ \sum_{j \in \hat{G}} f_{ij} &= c_i \lambda_i, & \text{for all } i \in B_c \cap Z \\ \sum_{j \in \hat{G}} f_{ij} &\geq c_i \lambda_i, & \text{for all } i \in B_c \setminus Z \\ \sum_{j \in \hat{G}} f_{ij} &= m_i, & \text{for all } i \in B \setminus B_c \\ f_{ij} &= 0, & \text{for all } \{i, j\} \in (B \times G) \setminus E \\ f_{ij} &\geq 0, & \text{for all } i \in B, j \in G \end{aligned}$$

- 6 **return** Feasible solution f of above LP

at such a minimum γ , and hence the inner while-loop ends. Otherwise, the MBB edges between buyers in $B \setminus S$ and goods in $\Gamma(S)$ will become non-MBB in the next iteration. Observe that there must be at least one such edge because these buyers can reach k in the MBB graph. So in each event of the inner while-loop, either a new MBB edge appears (Event 1) or an existing MBB edge vanishes (Event 2). Next, we show that for a given buyer k , the total number of iterations of the inner while-loop is polynomially bounded. For this, we first show that the price of a good strictly decreases during each iteration of the inner while-loop.

LEMMA 3.2. *In each iteration of the inner while-loop, the MBB ratio of buyer k strictly increases.*

Proof. Each iteration of the inner while-loop ends with one of the two events. Clearly, Event 1 can occur only when the prices of goods in \hat{G} strictly decrease, and this implies that the MBB of buyer k strictly increases. In case of Event 2, as argued above, there is a subset $S \subseteq \hat{B}$ of buyers such that $\sum_{i \in S} m_i^a = \sum_{j \in \Gamma(S)} p_j^a$, where $\Gamma(S)$ is the set of goods to which S have MBB edges.

If $k \in S$, then $s(k) = 0$ at the end of this iteration. As $s(k) > 0$ at the start of the iteration, we deduce that $\sum_{i \in S} m_i^a < \sum_{j \in \Gamma(S)} p_j^a$ at the beginning of this iteration, and since equality emerges, the prices must have strictly decreased and the MBB of k strictly increased.

If $k \notin S$, then $S \neq \hat{B}$ and the flow on all MBB edges from $\hat{B} \setminus S$ to $\Gamma(S)$ has become zero. Note that there is at least one such edge due to the construction of \hat{B} and \hat{G} . The fact that there was a non-zero flow on these edges implies that $\sum_{i \in S} m_i^a < \sum_{j \in \Gamma(S)} p_j^a$ at the beginning of this iteration. We conclude that the prices of goods must have strictly decreased and the MBB of k strictly increased. \square

Next we show that the price of a good decreases substantially after a certain number of iterations. For this, we partition the iterations into *phases*, where every phase has $3m$ iterations of the inner while-loop.

LEMMA 3.3. *Let \mathbf{p} and \mathbf{p}' be the prices at the beginning and end of a phase, respectively. Then $p'_j \leq p_j$ for all $j \in G$, and there exists a good ℓ such that $p'_\ell \leq p_\ell / (1 + \varepsilon)$.*

Proof. By Lemma 3.1, $p'_j \leq p_j$ for all $j \in G$. For the second part, note that \hat{B} always contains buyer k during an entire run of the inner while-loop. Since prices decrease monotonically, the MBB α_k of buyer k increases monotonically. Further, if there is an MBB path (formed by MBB edges) from buyer k to a good j , say $\{k, j_1\}, \{j_1, i_1\}, \{i_1, j_2\}, \dots, \{i_{k-1}, j_k\}, \{j_k, i_k\}, \{i_k, j\}$, then for an integer c , we have

$$\alpha_k p_j = \frac{\prod \tilde{u}_{kj_1} \tilde{u}_{i_1 j_2} \dots \tilde{u}_{i_{k-1} j_k} \tilde{u}_{i_k j}}{\prod \tilde{u}_{i_1 j_1} \tilde{u}_{i_2 j_2} \dots \tilde{u}_{i_k j_k}} = (1 + \varepsilon)^c.$$

In each iteration, either a new MBB edge appears or an existing MBB edge vanishes. When a new MBB edge appears, a new MBB path from buyer k to a good j gets established. When an existing MBB edge vanishes, then an old MBB path from k to a good j gets destroyed. Further, if there is an MBB path from k to j , then the price of j decreases monotonically. If there is no MBB path from k to j , then the price of j does not decrease. After at most $3m$ events, there has to be a good j such that initially there is an MBB path from k to j , then no MBB path between them for some iterations, then again an MBB path between them. Let p_j be the price of good j at the time when there is no path from k to j . Let α_k and α'_k be, resp., the MBB for buyer k at the time the MBB path from k to j was broken and when it was subsequently reestablished. Since p_j does not change unless there is a path from k to j , we have

$$\alpha_k p_j = (1 + \varepsilon)^{c_1} \text{ and } \alpha'_k p_j = (1 + \varepsilon)^{c_2}, \text{ for some integers } c_1 \text{ and } c_2.$$

Since $\alpha'_k > \alpha_k$, due to Lemma 3.2, we have $\alpha'_k \geq \alpha_k (1 + \varepsilon)$. Let good l be an MBB good for buyer k when he has multiplier α'_k , and let p_l and p'_l be the prices of good l when the MBB path from k to j was broken and when it was later established. This implies

$$u_{il}/p'_l = \alpha'_k \geq \alpha_k (1 + \varepsilon) \geq (1 + \varepsilon) u_{il}/p_l$$

and, thus, $p'_l \leq p_l / (1 + \varepsilon)$. \square

Recall that \tilde{U} is the largest rounded utility.

LEMMA 3.4. *The number of iterations of the inner while-loop of Algorithm 1 is at most $O(nm \log_{1+\varepsilon}(n\tilde{U}^n \sum_i m_i))$.*

Proof. By Lemma 3.3, in each phase the price of a good decreases by a factor of $1 + \varepsilon$. The number of iterations in a phase is in $O(m)$. The starting price of any good is at most $\sum_i m_i$. Note that the price of a good j cannot go down below $p_j^{\min} \geq 1/(n\tilde{U}^n)$ (Lemma 3.9), otherwise it would contradict that p_j^{\min} is the minimum possible price of j in any equilibrium. This implies that the inner while-loop ends for a particular buyer k before the price of some good j is less than p_j^{\min} . Hence, the number of phases is at most $n \log_{1+\varepsilon}(n\tilde{U}^n \sum_i m_i)$, and the number of iterations of the inner while-loop is at most $O(nm \log_{1+\varepsilon}(n\tilde{U}^n \sum_i m_i))$. \square

THEOREM 3.1. *For every $\varepsilon > 0$, Algorithm 1 computes a thrifty and modest equilibrium in the perturbed market $\tilde{\mathcal{M}}$ in time polynomial in m, n, U and $1/\varepsilon$.*

Proof. From Lemma 3.1, all invariants are maintained throughout the algorithm. Hence, the surplus of each good is 0, the surplus of each buyer is non-negative, and all prices decrease monotonically. The algorithm ends when the surplus of each buyer is zero.

Lemma 3.4 shows that there are at most $O(nm \log_{1+\varepsilon}(n\tilde{U}^n \sum_i m_i))$ iterations. Since rounded utilities exceed utilities by a factor of at most $1 + \varepsilon$, this can be upper bounded by $O(\frac{n^2 m}{\varepsilon} \log(nU))$, where U denotes the largest integer in the representation of the market. Each iteration can be implemented in polynomial time. \square

Approximate Equilibrium Our algorithm computes an exact equilibrium in $\tilde{\mathcal{M}}$ in polynomial time. We show that such an exact equilibrium of $\tilde{\mathcal{M}}$ is an ε -approximate equilibrium of \mathcal{M} , thereby obtaining an FPTAS for the problem. Next, we define the precise notion of an ε -approximate market equilibrium. It is based on a notion of an ε -approximate demand bundle.

DEFINITION 3.3 (APPROXIMATE DEMAND). For a vector \mathbf{p} of prices, let \mathbf{x}_i^* be a demand bundle for buyer i . An allocation \mathbf{x}_i for buyer i is called an ε -approximate (thrifty and modest) demand bundle if (1) $\sum_j u_{ij} x_{ij} \leq c_i$, (2) $\sum_j x_{ij} p_j \leq m_i^a$, and (3) $u_i(\mathbf{x}_i) \geq (1 - \varepsilon)u_i(\mathbf{x}_i^*)$.

An ε -approximate (thrifty and modest) equilibrium differs from an exact equilibrium only by a relaxation of condition (4) to ε -approximate demand (c.f. Definition 2.1)

DEFINITION 3.4 (APPROXIMATE EQUILIBRIUM). An ε -approximate (thrifty and modest) equilibrium is a pair (\mathbf{x}, \mathbf{p}) , where \mathbf{x} is an allocation and \mathbf{p} a vector of prices such that conditions (1)-(3), (5) from Definition 2.1 hold, and (4) \mathbf{x}_i is an ε -approximate demand bundle for every $i \in B$.

Note that our definition is rather demanding – there are many further possible relaxations which we do not allow (e.g., we require exact market clearing, modest supplies, exact earning and utility caps), some of which are found in other notions of approximate equilibrium in the literature.

LEMMA 3.5. *An exact equilibrium (\mathbf{x}, \mathbf{p}) of $\tilde{\mathcal{M}}$ is an ε -approximate equilibrium of \mathcal{M} .*

Proof. Conditions (1) – (3) and (5) hold for $\tilde{\mathcal{M}}$ and are unchanged for \mathcal{M} . To see the ε -approximate property, let α_i and $\tilde{\alpha}_i$ be the MBB of buyer i at prices \mathbf{p} w.r.t. utility u_i and perturbed utility \tilde{u}_i , respectively. Formally, $\alpha_i = \max_{k \in G} u_{ik}/p_k$ and $\tilde{\alpha}_i = \max_{k \in G} \tilde{u}_{ik}/p_k$. Since $(1 + \varepsilon)u_{ij} > \tilde{u}_{ij} \geq u_{ij}$ for all i, j , we have $(1 + \varepsilon)\alpha_i > \tilde{\alpha}_i \geq \alpha_i$ for all i . At prices \mathbf{p} , let x_i^* be an optimal bundle of buyer i . Clearly, $u_i(x_i^*) = \min\{c_i, m_i \alpha_i\}$, and $\tilde{u}_i(x_i) = \min\{c_i, m_i \tilde{\alpha}_i\}$. Therefore,

$$u_i(x_i) > \frac{\tilde{u}_i(x_i)}{1 + \varepsilon} = \frac{\min\{c_i, m_i \tilde{\alpha}_i\}}{1 + \varepsilon} \geq \frac{\min\{c_i, m_i \alpha_i\}}{1 + \varepsilon} = \frac{u_i(x_i^*)}{1 + \varepsilon} \geq (1 - \varepsilon)u_i(x_i^*) .$$

Condition (1) of the ε -approximate demand holds for \mathcal{M} as $u_{ij} \leq \tilde{u}_{ij}$ for all i, j . Condition (2) holds as $m_i^a = \min\{m_i, c_i/\alpha_i\}$ and $\tilde{\alpha}_i \geq \alpha_i$. Finally, (3) holds as argued above. \square

Algorithm 4: Finite-Time Algorithm for Money-Clearing Markets

Input : Market \mathcal{M} given by budgets m_i , utility caps c_i , earning caps d_j , utilities u_{ij} , for all $i \in B$ and $j \in G$;

Output: Equilibrium prices \mathbf{p} and allocation \mathbf{x}

$(\mathbf{f}, \mathbf{p}) \leftarrow$ equilibrium of \mathcal{M} when ignoring all utility caps

while $\sum_i s(i) > 0$ **do**

$\mathbf{f} \leftarrow$ balanced flow in $N^-(\mathbf{p})$ $\//$ surpluses change similarly

$\delta \leftarrow \max_i s(i)$

$\hat{B} \leftarrow$ set of buyers with surplus δ $\// \delta > 0$

$\hat{G} \leftarrow \{k \in G \mid f_{ki} > 0, i \in \hat{B}\}$

Set $\gamma \leftarrow 1$, and $p_j \leftarrow \gamma \cdot p_j$ for all $j \in \hat{G}$

Decrease γ continuously down from 1 until one of these events occurs:

Event 1: An uncapped buyer becomes capped

Event 2: A capped good becomes uncapped

Event 3: A new MBB edge appears

Event 4: A subset of \hat{B} becomes *tight* $\// N^-(\mathbf{p})$ is feasible

$\mathbf{f} \leftarrow$ feasible flow in $N^-(\mathbf{p})$

$(\mathbf{f}, \mathbf{p}) \leftarrow \text{MinPrices}(\mathbf{f}, \mathbf{p})$

$\mathbf{x} \leftarrow \text{FindAllocation}(\mathbf{f}, \mathbf{p})$

return (\mathbf{x}, \mathbf{p})

COROLLARY 3.1. *Algorithm 1 is an FPTAS for computing an ε -approximate equilibrium for money-clearing markets with earning and utility limits.*

3.3. Membership in PLS In this section we show that the problem of computing an exact equilibrium in a money-clearing market \mathcal{M} is in the class PLS. We first design Algorithm 4, a finite-time descending-price algorithm. It again relies on the *reverse flow network* $N^-(\mathbf{p}) = N^-(\mathbf{p}, \emptyset)$ defined in the previous section for which we use $Z = \emptyset$.

Algorithm and Invariants The algorithm starts by computing a market equilibrium ignoring the utility caps of the buyers. This equilibrium exists since the market is money-clearing. It is a pair (\mathbf{f}, \mathbf{p}) of flow and prices for which the outflow of good j is equal to p_j^a and the inflow into buyer i is m_i .

We will maintain the following Invariants during the while-loop in Algorithm 4:

- No price ever increases.
- $N^-(\mathbf{p})$ allows a feasible flow.

LEMMA 3.6. *The Invariants hold during the run of Algorithm 4.*

The lemma is quite straightforward to see. Towards a proof, we explain the algorithm in more technical detail. The algorithm uses a descending-price approach. There is always a flow in $N^-(\mathbf{p})$ with out-flow of good $j \in G$ equal to p_j^a and in-flow into buyer $i \in B$ at least m_i^a . The first invariant implies that once a good becomes uncapped, it remains uncapped, and once a buyer becomes capped, he remains capped.

We proceed to discuss the algorithm. It will then be straightforward to see that the lemma holds. In the body of the while-loop, we first compute a balanced flow \mathbf{f} . A *balanced flow* is a maximum feasible flow in $N^-(\mathbf{p})$ which minimizes the 2-norm of surplus vector $\mathbf{s} = (s(1), s(2), \dots, s(|B|))$. The notion of balanced flow was introduced in [23] for equilibrium computation in linear Fisher

Algorithm 5: MinPrices

Input : Market \mathcal{M} , prices \mathbf{p} , flow \mathbf{f}

Output: Minimum prices consistent with input configuration, feasible money flow

- 1 $E \leftarrow$ set of MBB edges at prices \mathbf{p}
- 2 $G_c \leftarrow$ set of capped goods at (\mathbf{f}, \mathbf{p})
- 3 $B_c \leftarrow$ set of capped buyers at (\mathbf{f}, \mathbf{p})
- 4 Solve the following LP in price variables \mathbf{q} and flow variables \mathbf{g} :

$$\begin{aligned}
 \min & \sum_j q_j \\
 u_{ij} q_k &= u_{ik} q_j, && \text{for each pair of edges } \{i, j\}, \{i, k\} \in E \\
 u_{ij} q_k &\geq u_{ik} q_j, && \text{for each pair of edge } \{i, j\} \in E \text{ and non-edge } \{i, k\} \notin E \\
 q_j &\leq d_j, && \text{for all } j \in G \setminus G_c \\
 q_j &\geq d_j, && \text{for all } j \in G_c \\
 \sum_i g_{ij} &= d_j, && \text{for all } j \in G_c \\
 \sum_i g_{ij} &= q_j, && \text{for all } j \in G \setminus G_c \\
 \sum_j g_{ij} &= m_i, && \text{for all } i \in B \setminus B_c \\
 m_i &\geq \sum_j g_{ij} \geq c_i q_k / u_{ik}, && \text{for all } i \in B_c \text{ where } \{i, k\} \in E \\
 g_{ij} &= 0, && \text{for all } \{i, j\} \in (B \times G) \setminus E \\
 q_j &\geq 0; g_{ij} \geq 0 && \text{for all } i \in B, j \in G
 \end{aligned}$$

- 5 **return** Optimal solution $(\mathbf{g}^*, \mathbf{q}^*)$ of the above LP

Algorithm 6: FindAllocation

Input : Market \mathcal{M} , prices \mathbf{p} , flow \mathbf{f}

Output: Allocation \mathbf{x}

- 1 $\hat{G} \leftarrow \{j \in G \mid p_j = 0\}$
- 2 $\hat{B} \leftarrow \{i \in B \mid u_{ij} > 0, j \in \hat{G}\}$
- 3 Solve the following feasibility LP in allocation variables $(x_{ij})_{i \in \hat{B}, j \in \hat{G}}$:

$$\begin{aligned}
 \sum_{j \in \hat{G}} u_{ij} x_{ij} &= c_i, && \text{for all } i \in \hat{B} \\
 \sum_{i \in \hat{B}} x_{ij} &\leq 1, && \text{for all } j \in \hat{G} \\
 x_{ij} &\geq 0 && \text{for all } i \in \hat{B}, j \in \hat{G}
 \end{aligned}$$
- 4 $x_{ij} \leftarrow f_{ij} / p_j$ for all $i \in B \setminus \hat{B}, j \in G \setminus \hat{G}$
- 5 **return** \mathbf{x}

markets. It can be computed by n maxflow computations. Consider two buyers i and k with different surpluses, say $s(i) > s(k)$. If there is a good j connected to i and k by MBB edges, then there is no flow from j to i [23]. Otherwise, we could decrease f_{ji} , increase f_{jk} by the same amount, and thus decrease the 2-norm of the surplus vector.

Let δ be the maximum surplus of any buyer, and let \hat{B} be the set of buyers with surplus δ . We let \hat{G} be the set of goods k that have non-zero flow to some buyer in \hat{B} . We then decrease the prices of all goods in \hat{G} by a common factor γ . Starting with $\gamma = 1$, we decrease it continuously. This may destroy MBB edges connecting buyers in \hat{B} with goods in $G \setminus \hat{G}$, but, by definition of \hat{G} , there is no flow on such edges. For uncapped goods and capped buyers, this decreases the active price, respectively budget by a factor of γ . We stop if one of four events happens: (1) an uncapped buyer becomes capped, (2) a capped good becomes uncapped, (3) a new MBB edge appears, or (4) a subset of \hat{B} becomes tight. A subset T of buyers is called *tight* with respect to prices \mathbf{p} if $\sum_{i \in T} m_i^a = \sum_{j \in \Gamma(T)} p_j^a$, where $\Gamma(T) \subseteq \hat{G}$ is the set of goods connected to T in the MBB graph.

Observe that in a feasible flow in $N^-(\mathbf{p})$, each buyer $i \in S$ receives flow m_i^a from $\Gamma(S)$, but some of the flow from $\Gamma(S)$ may go to buyers outside S . Therefore, we must have

$$\sum_{i \in S} m_i^a \leq \sum_{j \in \Gamma(S)} p_j^a, \quad \text{for all } S \subseteq B.$$

We can determine the value of γ at which Event 4 happens using n max-flow computations [23]. Next we obtain a feasible flow \mathbf{f} in $N^-(\mathbf{p})$, which is guaranteed to exist by Event 4. We then use an LP (Algorithm 5) to compute the pair (\mathbf{g}, \mathbf{q}) of flow and prices which minimizes $\sum_j q_j$ subject to the constraints that (1) the same buyers are capped, (2) the same goods are capped, and (3) the same edges are MBB as for (\mathbf{f}, \mathbf{p}) . The ratio of any two prices in a connected component of the MBB graph is constant, and \mathbf{f} is a feasible solution to the LP. Hence, $\mathbf{q} \leq \mathbf{p}$ component-wise. This implies that, indeed, the invariants hold throughout the while-loop.

Finally, we find an equilibrium allocation using Algorithm 6. Here, we first obtain the set \hat{G} of zero-priced goods and the set \hat{B} of buyers who have non-zero utilities for some good in \hat{G} . Clearly, the buyers of \hat{B} must be capped. For the buyers and goods in \hat{B} and \hat{G} respectively, we find an allocation by solving a feasibility LP which allocates each buyer i a bundle of goods worth c_i amount of utility. Note that the set of feasible solutions for this LP is non-empty, because we always maintain all the invariants (Lemma 3.6) throughout the algorithm.

Running Time and PLS We call the tuple (E, B_c, G_c) a *configuration*, where $E \subseteq B \times G$ is a set of MBB edges, $B_c \subseteq B$ is a set of capped buyers, and $G_c \subseteq G$ is a set of capped sellers. At the beginning of each iteration, we have a configuration based on the current prices. The following lemma ensures that our algorithm makes progress towards an equilibrium.

LEMMA 3.7. *During the run of Algorithm 4, no configuration repeats.*

Proof. An iteration ending with Event 1 or 2 grows the set of capped buyers or the set of uncapped goods. Since these sets never lose members, no preceding configuration can repeat. If the sum of prices is strictly decreased before an event, i.e., $\gamma < 1$, none of the preceding configurations can repeat, since we find the minimum possible prices for the current configuration at the end of each iteration. We will show below that the prices of the goods in \hat{G} are strictly decreased when an iteration ends with Event 3 or 4.

In the case of Event 3, a new MBB edge appears from a buyer k in $B \setminus \hat{B}$ to a good j in \hat{G} . For such an edge to become MBB, γ must be strictly less than 1: Since $k \notin \hat{B}$, $s(k) < \delta$ in the balanced flow. If $\gamma = 1$, then using this MBB edge from k to j , we could increase the surplus of k and decrease the surplus of a buyer in \hat{B} . This would decrease the 2-norm of the surplus vector, a contradiction.

Next consider an iteration that ends due to Event 4, and suppose the prices of goods in \hat{G} are not decreased, i.e., $\gamma = 1$. Note that the surplus of each buyer in \hat{B} is $\delta > 0$ and the surplus of each good is 0. Hence, before we decrease prices in lines 8–12 of Algorithm 4,

$$\sum_{j \in \hat{G}} p_j^a - \sum_{i \in \hat{B}} m_i^a = \delta \cdot |\hat{B}| .$$

If Event 4 occurs at $\gamma = 1$, then there is already a tight subset $T \subseteq \hat{B}$, i.e., $\sum_{j \in \Gamma(T)} p_j^a - \sum_{i \in T} m_i^a = 0$, where $\Gamma(T)$ is the set of goods connected to T . However, for each $i \in T$, $\delta = s(i) = \sum_{j \in \Gamma(T)} f_{ji} - m_i^a$. Together, they imply that $\delta \cdot |T| = \sum_{i \in T} (\sum_{j \in \Gamma(T)} f_{ji} - m_i^a) = \sum_{i \in T, j \in \Gamma(T)} f_{ji} - \sum_{j \in \Gamma(T)} p_j^a = -\sum_{i \notin T, j \in \Gamma(T)} f_{ji}$, which is a contradiction. \square

THEOREM 3.2. *For any money-clearing market, Algorithm 4 computes a thrifty and modest equilibrium in exponential time.*

Proof. In each iteration, the balanced flow can be obtained in polynomial time [23]. Consider the maximum γ at which an event occurs. The maximum γ for the first three events can be easily obtained in polynomial time. For Event 4, we need to find the maximum γ when a set of buyers become tight, which can be computed using at most a linear number of max-flow computations [23, 8]. Finally, the LP in Algorithm 5 can be solved in polynomial time, hence each iteration can be implemented in polynomial time.

By Lemma 3.7, each iteration begins with a different configuration. There are $2^{O(n \cdot m \cdot (nm))}$ distinct configurations, so Algorithm 4 terminates with an equilibrium. The running time depends polynomially on n , m , U , and the number of distinct configurations. \square

Observe that our algorithm constructs an initial configuration in polynomial time. Then, for each configuration, we can interpret the sum of consistent prices as an objective function, which can be computed by algorithm MinPrices. Furthermore, we can define a suitable neighborhood among configurations. Algorithm 4 can be interpreted as doing this. Also, we can compute in polynomial time an equilibrium for a market \mathcal{M}^s as a starting configuration for our algorithm. As such, our algorithm implements the oracles for the class PLS.

COROLLARY 3.2. *The problem of computing a thrifty and modest equilibrium in money-clearing markets is in the class PLS.*

Proof. We call a configuration *feasible* if the LP in Algorithm 5 is feasible and its output makes the feasibility-LP of Algorithm 6 non-empty. Otherwise, we call the configuration *infeasible*. For membership in PLS, we construct neighborhood and cost functions on the set of configurations that are computable in polynomial time and satisfy the following property: A configuration has lowest cost among all its neighbors (local optimum) if and only if it is an equilibrium.

For each feasible configuration, let the cost be the optimum value of the corresponding LP. For each infeasible configuration, we define its cost to be a prohibitively high value of $(n + m)U^{n+m+1} \sum_{i \in B} m_i$. Each infeasible configuration has a unique neighbor – the starting configuration of Algorithm 4 (in line 1). Observe that we can take any feasible configuration as the starting configuration in Algorithm 4. Accordingly, we define the unique neighbor of each feasible configuration \mathcal{C} as the next configuration in Algorithm 4 when it is started with \mathcal{C} . Clearly, both cost and neighborhood functions are polynomial-time computable, and a configuration is a local optimum if and only if it is a thrifty and modest equilibrium. This proves the claim. \square

REMARK 3.1. It is not obvious how to use Algorithm 1 to show membership in the class PLS. The difficulty lies in defining a suitable configuration space and a potential function.

3.4. Constant Number of Buyers or Goods In this section, we show that Algorithm 4 runs in polynomial time when either the number of buyers or the number of sellers is a constant. Consider the number of MBB graphs for a fixed set of capped buyers and capped sellers. Using a cell decomposition technique, we show it is polynomial when $|B|$ or $|G|$ is constant. We create regions in a constant-dimensional space by introducing polynomially many hyperplanes. The number of non-empty regions formed by N hyperplanes in \mathbb{R}^d is $O(N^d)$. Thus, we get a polynomial bound on the number of regions.

We then show that each MBB graph maps to a particular region thus created. Since the number of regions is polynomial, we get a polynomial bound on the number of different MBB graphs. This implies that for any given set of capped buyers and capped sellers, Algorithm 4 examines only polynomially many configurations. Since the set of capped buyers only grows and the set of capped sellers only shrinks, this implies a polynomial running time for Algorithm 4.

THEOREM 3.3. *For any money-clearing market with a constant number of buyers or sellers, Algorithm 4 computes a thrifty and modest equilibrium in polynomial time.*

Proof. For a constant number of goods, consider the following set of hyperplanes in $(p_1, \dots, p_{|G|})$ -space, where p_j denotes the price of good j :

$$u_{ij}p_{j'} - u_{ij'}p_j = 0, \quad \text{for all } i \in B \text{ and all } j, j' \in G.$$

These hyperplanes partition the space into cells, and each cell has one of the signs $<, =, >$ for each hyperplane. Further, each MBB graph $(B \cup G, E)$ satisfies the following constraints in \mathbf{p} -variables:

$$\begin{aligned} \text{For all } \{i, j\}, \{i, j'\} \in E : \quad & u_{ij}p_{j'} - u_{ij'}p_j = 0 \\ \text{For all } \{i, j\} \in E \text{ and all } \{i, j'\} \notin E : \quad & u_{ij}p_{j'} - u_{ij'}p_j > 0. \end{aligned}$$

Now, for a constant number of buyers consider the following set of hyperplanes in $(\lambda_1, \dots, \lambda_{|B|})$ -space, where $1/\lambda_i$ denotes the MBB of buyer i :

$$\lambda_i u_{ij} - \lambda_{i'} u_{i'j} = 0, \quad \text{for all } i, i' \in B \text{ and all } j \in G.$$

These hyperplanes partition the space into cells, and each cell has one of the signs $<, =, >$ for each hyperplane. Further, each MBB graph $(B \cup G, E)$ satisfies the following constraints in $\boldsymbol{\lambda}$ -variables:

$$\begin{aligned} \text{For all } \{i, j\}, \{i', j\} \in E : \quad & \lambda_i u_{ij} - \lambda_{i'} u_{i'j} = 0 \\ \text{For all } \{i, j\} \in E \text{ and all } \{i', j\} \notin E : \quad & \lambda_i u_{ij} - \lambda_{i'} u_{i'j} > 0. \end{aligned}$$

In both cases, each MBB graph maps to a particular cell in the cell decomposition. Since the number of cells is polynomially bounded for constant $|G|$ or $|B|$, this implies a polynomial bound on the number of different MBB graphs. Thus, since the set of capped buyers only grows and the set of capped sellers only shrinks, we get a polynomial running time for Algorithm 4. \square

3.5. Membership in PPAD In this section, we show that computing a thrifty and modest equilibrium in money-clearing markets \mathcal{M} is in the class PPAD. We first derive a formulation as a linear complementarity problem (LCP). It captures all thrifty and modest equilibria of \mathcal{M} , but also has non-equilibrium solutions. To discard the non-equilibrium solutions, we incorporate a positive lower bound on variables representing the prices of the goods and the MBB ratio of each buyer. This turns out to be a non-trivial adjustment, because a subset of prices may be zero at all equilibria, so we must be careful not to discard equilibrium solutions. Our approach is based on our previous work [8], in which we gave a polynomial-time algorithm for markets \mathcal{M}^b with utility limits (but without earning limits). This algorithm finds an equilibrium whose prices are coordinate-wise smallest among all equilibria.

Our approach can be summarized as follows. Consider a money-clearing market \mathcal{M} . Now suppose we remove all earning caps to obtain a market \mathcal{M}^b . To this market we apply the algorithm of [8] and obtain a min-price equilibrium $(\mathbf{x}^{min}, \mathbf{p}^{min})$. We show that using \mathbf{p}^{min} , the market \mathcal{M} can be partitioned into two separate markets \mathcal{M}_1 and \mathcal{M}_2 . Market \mathcal{M}_1 consists of all goods with price 0 in \mathbf{p}^{min} and all buyers having non-zero utility for these goods. \mathcal{M}_2 consists of the remaining buyers and goods. Since all buyers in \mathcal{M}_2 have no utility for goods in \mathcal{M}_1 , we have that \mathcal{M}_2 is money clearing if and only if \mathcal{M} is money clearing.

Based on these two markets, we show that there is⁴ an equilibrium of \mathcal{M} that is an equilibrium of \mathcal{M}_1 and an equilibrium of \mathcal{M}_2 . We already know an equilibrium for \mathcal{M}_1 with a price of 0 for every good. For \mathcal{M}_2 we show that at every equilibrium, the price of a good j is at least p_j^{min} . Using this lower bound on the equilibrium prices in \mathcal{M}_2 , we construct a modified LCP formulation

⁴ In fact, it can be shown that every equilibrium of \mathcal{M} has this property, but this is not necessary for membership in PPAD.

\mathcal{M} -LCP which exactly captures all equilibria of \mathcal{M}_2 . We add a suitable auxiliary scalar variable to \mathcal{M} -LCP and apply Lemke's algorithm. If \mathcal{M}_2 is money clearing, we show that Lemke's algorithm is guaranteed to converge to an equilibrium of \mathcal{M}_2 . Combining this with the equilibrium of \mathcal{M}_1 gives an equilibrium of \mathcal{M} . Further, using a result of Todd [54] specified later, this proves that computing an equilibrium in money-clearing markets \mathcal{M} is in PPAD.

3.5.1. LCP Formulation We start our analysis by deriving an LCP formulation to capture the equilibria of \mathcal{M} . Let \perp denote a complementarity constraint between an inequality and a variable, e.g., $u_{ij}\lambda_i - p_j \leq 0 \perp f_{ij} \geq 0$ is shorthand for $u_{ij}\lambda_i - p_j \leq 0$ and $f_{ij} \geq 0$ and $f_{ij}(u_{ij}\lambda_i - p_j) = 0$. The LCP has the following variables and constraints:

- $\mathbf{p} = (p_j)_{j \in G}$, where p_j is the price of good j ,
- $\mathbf{f} = (f_{ij})_{i \in B, j \in G}$, where f_{ij} is the money spent on good j by buyer i ,
- $\boldsymbol{\lambda} = (\lambda_i)_{i \in B}$, where $1/\lambda_i$ is the MBB of buyer i at prices \mathbf{p} ,
- $\boldsymbol{\delta} = (\delta_i)_{i \in B}$, where $(m_i - \delta_i)$ is the active budget of buyer i ,
- $\boldsymbol{\beta} = (\beta_j)_{j \in G}$, where $(p_j - \beta_j)$ is the active price of good j .

$$\text{For all } (i, j) \in (B, G) : \quad u_{ij}\lambda_i - p_j \leq 0 \quad \perp \quad f_{ij} \geq 0 \quad (3)$$

$$\text{For all } i \in B : \quad m_i - c_i\lambda_i \leq \delta_i \quad \perp \quad \delta_i \geq 0 \quad (4)$$

$$\text{For all } i \in B : \quad -\sum_j f_{ij} \leq -(m_i - \delta_i) \quad \perp \quad \lambda_i \geq 0 \quad (5)$$

$$\text{For all } j \in G : \quad p_j - d_j \leq \beta_j \quad \perp \quad \beta_j \geq 0 \quad (6)$$

$$\text{For all } j \in G : \quad \sum_i f_{ij} \leq p_j - \beta_j \quad \perp \quad p_j \geq 0 \quad (7)$$

LEMMA 3.8. *The LCP defined by (3)–(7) captures all the equilibria of \mathcal{M} .*

Proof. Let (\mathbf{f}, \mathbf{p}) be an equilibrium of \mathcal{M} . Let $1/\lambda_i$ be the MBB ratio of buyer i at prices \mathbf{p} . Clearly, $\lambda_i = \min_{j:u_{ij}>0} p_j/u_{ij}$. In equilibrium, each buyer only spends money on goods with largest MBB ratio, so $f_{ij} > 0$ only if $u_{ij}\lambda_i - p_j = 0$. This implies that $(\mathbf{p}, \mathbf{f}, \boldsymbol{\lambda})$ satisfies (3). The active budget m_i^a of buyer i is $\min\{m_i, c_i\lambda_i\}$, and $\sum_j f_{ij} = m_i^a$. Setting $\delta_i = \max\{0, m_i - c_i\lambda_i\}$ satisfies (4). This further implies that $m_i^a = m_i - \delta_i$, and (5) follows. Similarly, the active price p_j^a of good j is $\min\{p_j, d_j\}$, and $\sum_i f_{ij} = p_j^a$. Setting $\beta_j = \max\{0, p_j - d_j\}$ satisfies (6). This further implies that $p_j^a = p_j - \beta_j$, and (7) follows. This proves the claim. \square

Lemma 3.8 shows that all equilibria of \mathcal{M} are captured by the LCP (3)–(7). However, there are solutions to this LCP which are not market equilibria. For example, $\lambda_i = p_j = \beta_j = f_{ij} = 0$ for all i and j and $\delta_i = m_i$ for all i is an LCP solution but not an equilibrium. To discard these non-equilibrium solutions in the LCP, we strive to include a positive lower bound to all p_j and λ_i .

REMARK 3.2. Previous constructions [32, 34] of LCPs for market equilibria use a positive lower bound of 1 for the prices. In our case this is not sufficient, for at least two reasons. First, there might be prices that are zero in all equilibria. A positive lower bound for these prices would discard all equilibria as solutions of the LCP. Second, there are solutions to the LCP, where $\lambda_i = 0$ and $\delta_i = m_i$ for all i and $f_{ij} = 0$ for all (i, j) . As a consequence, we need to establish positive lower bounds for all λ_i .

To handle these difficulties we use our polynomial-time algorithm [8] for markets \mathcal{M}^b with utility caps. Consider market \mathcal{M} and disregard all earning caps. The resulting market is a market \mathcal{M}^b , for which our algorithm from [8] can compute a price vector $\mathbf{p}^{min} = (p_j^{min})_{j \in G}$ of a min-price equilibrium. A min-price equilibrium has coordinate-wise smallest prices, i.e., for every good j the price p_j^{min} is the smallest price of good j in all equilibria. As a consequence, the set $S = \{j \in G \mid p_j^{min} = 0\}$ includes all goods that have price zero in every equilibrium of market \mathcal{M}^b . Let $\Gamma(S)$ be the set of buyers who derive non-zero utility from goods in S , i.e., $\Gamma(S) = \{i \in B \mid u_{ij} > 0 \text{ for some } j \in S\}$.

We partition the market into two disjoint markets. Market \mathcal{M}_1 includes exactly the goods of S and the buyers in $\Gamma(S)$. Market \mathcal{M}_2 has the remaining goods and buyers. By definition $u_{ij} = 0$ for every $j \in S$ and $i \notin \Gamma(S)$ and hence no buyer in \mathcal{M}_2 will ever spend on goods in \mathcal{M}_1 . The min-price equilibrium $(\mathbf{x}^{\min}, \mathbf{p}^{\min})$ yields an equilibrium for \mathcal{M}_1 , since all utility caps for all $i \in \Gamma(S)$ are reached and all earning caps for all $j \in S$ are satisfied. We will next establish that in every equilibrium for \mathcal{M}_2 every good $j \notin S$ has a price $p_j \geq p_j^{\min}$. Then no buyer in \mathcal{M}_1 will ever spend on goods from \mathcal{M}_2 , justifying the separation of the markets.

LEMMA 3.9. *In every equilibrium for market \mathcal{M}_2 , $p_j \geq p_j^{\min}$, for all $j \notin S$. Furthermore, $p_j^{\min} \geq 1/(nU_{\max}^n)$, where U_{\max} is the largest utility parameter.*

Proof. For the first part, consider the min-price equilibrium $(\mathbf{x}^{\min}, \mathbf{p}^{\min})$ without earning caps, restricted to buyers and goods in \mathcal{M}_2 . For a subset T of goods, let $\Gamma_{\mathbf{p}}(T)$ denote the set of buyers having MBB edges to at least one good in T at prices \mathbf{p} . We claim that for any non-empty set T of goods in \mathcal{M}_2 , either $\Gamma_{\mathbf{p}^{\min}}(T)$ contains an uncapped buyer or there is a capped buyer in $\Gamma_{\mathbf{p}^{\min}}(T)$ who spends money on a good outside T . Observe that if neither of the two conditions are true, then we can find another equilibrium by reducing the prices of goods in T and keeping the allocation the same, which contradicts that $(\mathbf{x}^{\min}, \mathbf{p}^{\min})$ is a *min-price* equilibrium without earning caps.

Now suppose there is an equilibrium (\mathbf{x}, \mathbf{p}) of \mathcal{M}_2 where $p_j < p_j^{\min}$ for some $j \notin S$. Let $\gamma = \min_{k \in G \setminus S} p_k / p_k^{\min} < 1$ and let T be the set of goods j in $G \setminus S$ for which $p_j / p_j^{\min} = \gamma$, i.e., $T = \{j \in G \setminus S \mid p_j / p_j^{\min} = \gamma\}$. Consider any buyer $i \in \Gamma_{\mathbf{p}^{\min}}(T)$ and let $m_i^a(\mathbf{p})$ and $m_i^a(\mathbf{p}^{\min})$ denote the active budget of i at prices \mathbf{p} and \mathbf{p}^{\min} , respectively. Then $m_i^a(\mathbf{p}^{\min}) = \min(m_i, \lambda_i c_i)$, where λ_i is the reciprocal of the MBB ratio of i at prices \mathbf{p}^{\min} . Since the prices of the goods outside T are scaled by a factor larger than γ when passing from \mathbf{p}^{\min} to \mathbf{p} , all MBB goods for i with respect to prices \mathbf{p} are entirely in T . Also λ_i is scaled by γ and hence $m_i^a(\mathbf{p}) = \min(m_i, \gamma \lambda_i c_i)$. Thus, if i is capped in $(\mathbf{x}^{\min}, \mathbf{p}^{\min})$, then i will be capped in (\mathbf{x}, \mathbf{p}) and $m_i^a(\mathbf{p}) = \gamma m_i^a(\mathbf{p}^{\min})$. If i is uncapped in the former equilibrium, he may or may not be uncapped in the latter, and $m_i^a(\mathbf{p}) = \min(m_i, \gamma \lambda_i c_i) > \gamma m_i \geq \gamma \min(m_i, \lambda_i c_i) = \gamma m_i^a(\mathbf{p}^{\min})$. Using these properties, we obtain

$$\sum_{i \in \Gamma_{\mathbf{p}^{\min}}(T)} m_i^a(\mathbf{p}) \geq \gamma \cdot \sum_{i \in \Gamma_{\mathbf{p}^{\min}}(T)} m_i^a(\mathbf{p}^{\min}) \geq \gamma \cdot \sum_{j \in T} p_j^{\min} = \sum_{j \in T} p_j \geq \sum_{j \in T} p_j^a. \quad (8)$$

If in equilibrium $(\mathbf{x}^{\min}, \mathbf{p}^{\min})$, $\Gamma_{\mathbf{p}^{\min}}(T)$ contains an uncapped buyer, the first inequality of (8) is strict, and if a capped buyer spends money on a good outside T , the second inequality of (8) is strict. In either case, we have

$$\sum_{i \in \Gamma_{\mathbf{p}^{\min}}(T)} m_i^a(\mathbf{p}) > \sum_{j \in T} p_j^a,$$

a contradiction because buyers in $\Gamma_{\mathbf{p}^{\min}}(T)$ only have MBB edges to goods in T at prices \mathbf{p} .

For the second part, consider the MBB graph \mathcal{G} restricted to buyers and goods in \mathcal{M}_2 at prices \mathbf{p}^{\min} . Observe that each connected component C of \mathcal{G} contains an uncapped buyer, otherwise we could reduce the prices of goods in C , which contradicts that $(\mathbf{x}^{\min}, \mathbf{p}^{\min})$ is a min-price equilibrium. Since the budget of each buyer is at least 1, there is a good, say g_C , in C that has price at least $1/n$. Moreover, since there is a path of length at most n of MBB edges connecting every other good g in C with g_C , the price of g is at least $1/(nU_{\max}^n)$, where $U_{\max} = \max_{i,j} u_{ij}$. \square

Next we derive an LCP for market \mathcal{M}_2 using the lower bound on the price of each good as given in Lemma 3.9. At this point, we need to solve the equilibrium problem for \mathcal{M}_2 only, so let $B_2 = B \setminus \Gamma(S)$ and $G_2 = G \setminus S$ denote the sets of buyers and goods in \mathcal{M}_2 , respectively. For the lower bound on λ_i 's we define

$$\Lambda = \frac{1}{2} \cdot \min_{i \in B_2, j \in G_2: u_{ij} > 0} \left\{ \frac{p_j^{\min}}{u_{ij}} \right\}. \quad (9)$$

Consider the following modified LCP in variables $(\boldsymbol{\lambda}', \mathbf{p}', \mathbf{f}, \boldsymbol{\delta}, \boldsymbol{\beta})$, where we substitute $\lambda_i \triangleq \lambda'_i + \Lambda$ and $p_j \triangleq p'_j + p_j^{\min}$. Together with $\lambda'_i \geq 0$ and $p'_j \geq 0$, this implies that $p_j \geq p_j^{\min}$ and $\lambda_i \geq \Lambda$.

$$\text{For all } (i, j) \in (B_2, G_2) : u_{ij}(\lambda'_i + \Lambda) - (p'_j + p_j^{\min}) \leq 0 \perp f_{ij} \geq 0 \quad (10)$$

$$\text{For all } i \in B_2 : m_i - c_i(\lambda'_i + \Lambda) \leq \delta_i \perp \delta_i \geq 0 \quad (11)$$

$$\text{For all } i \in B_2 : -\sum_j f_{ij} \leq -(m_i - \delta_i) \perp \lambda'_i \geq 0 \quad (12)$$

$$\text{For all } j \in G_2 : p'_j + p_j^{\min} - d_j \leq \beta_j \perp \beta_j \geq 0 \quad (13)$$

$$\text{For all } j \in G_2 : \sum_i f_{ij} \leq p'_j + p_j^{\min} - \beta_j \perp p'_j \geq 0 \quad (14)$$

The constraints (10)-(14) represent the \mathcal{M} -LCP. It is the same as the LCP (3)-(7) for \mathcal{M}_2 with additional constraints $p_j \geq p_j^{\min}$ for all $j \in G_2$ and $\lambda_i \geq \Lambda$ for all $i \in B_2$. We show that this LCP exactly captures all equilibria of \mathcal{M}_2 .

LEMMA 3.10. *A solution of \mathcal{M} -LCP is a thrifty and modest equilibrium of \mathcal{M}_2 and vice-versa.*

Proof. Lemmas 3.8 and 3.9 show that every equilibrium of \mathcal{M}_2 is a solution of \mathcal{M} -LCP. For the other direction, consider a solution $(\boldsymbol{\lambda}', \mathbf{p}', \mathbf{f}, \boldsymbol{\delta}, \boldsymbol{\beta})$ of \mathcal{M} -LCP. For the active prices, we show $p_j^a = \min(d_j, p'_j + p_j^{\min}) = p'_j + p_j^{\min} - \beta_j$. Indeed, if $\beta_j = 0$, we have $p'_j + p_j^{\min} \leq d_j$ by (13), and then $p_j^a = p'_j + p_j^{\min}$; otherwise $p'_j + p_j^{\min} = d_j + \beta_j \geq d_j$ and then $p_j^a = d_j$. Similarly, for the active budgets, we have $m_i^a = \min(m_i, c_i(\lambda'_i + \Lambda)) = m_i - \delta_i$. Indeed, if $\delta_i = 0$, we have $m_i \leq c_i(\lambda'_i + \Lambda)$ by (11), and then $m_i^a = m_i$; otherwise $m_i = c_i(\lambda'_i + \Lambda) + \delta_i \geq c_i(\lambda'_i + \Lambda)$, and then $m_i^a = c_i(\lambda'_i + \Lambda)$. This implies that the active price p_j^a of good j is $p'_j + p_j^{\min} - \beta_j$, and the active budget m_i^a of buyer i is $m_i - \delta_i$. Clearly, $m_i^a > 0$ for all $i \in B_2$, and $p_j^a > 0$ for all $j \in G_2$.

We next show that $\lambda'_i > 0$ for all i . Suppose $\lambda'_i = 0$ for a buyer i . Then the left inequality in (10) is not tight by (9) and hence $f_{ij} = 0$ for all j . However, $\sum_j f_{ij} = 0$ violates the left inequality of (12) since $m_i - \delta_i = m_i^a > 0$. Hence $\lambda'_i > 0$, for all $i \in B_2$.

The constraints (10) ensure that $f_{ij} > 0$ implies $u_{ij}/(p'_j + p_j^{\min}) = \max_{k \in G_2} u_{ik}/(p'_k + p_k^{\min})$, i.e., money flows only on MBB edges, and each buyer buys an optimal bundle.

The constraints (12) together with the fact that $\lambda'_i > 0$ ensure $\sum_j f_{ij} = m_i^a$ for each i , i.e., each buyer spends his entire active budget. Now we only need to show that each good receives money equal to its active price, i.e., $\sum_i f_{ij} = p_j^a$, for all $j \in G_2$.

Note that the prices \mathbf{p}^{\min} impose an equilibrium for \mathcal{M}_2 without the earning caps. Let $S' = \{j \in G_2 \mid p'_j = 0\}$. Clearly, $\sum_i f_{ij} = p_j^a$, for all $j \in G_2 \setminus S'$, due to (14). Let $\Gamma(S')$ be the set of buyers having at least one MBB good in S' at prices \mathbf{p}^{\min} . Since \mathbf{p}^{\min} is the min-price equilibrium without earning caps, the total active budget at prices \mathbf{p}^{\min} of buyers in $\Gamma(S')$ is at least the total prices of goods in S' , i.e., $\sum_{j \in S'} p_j^{\min} \leq \sum_{i \in \Gamma(S')} m_i^a(\mathbf{p}^{\min})$, where (as in the proof of the preceding lemma) we use $m_i^a(\mathbf{p}^{\min})$ to denote the active budget of i at prices \mathbf{p}^{\min} .

Now suppose we set prices $p_j = p'_j + p_j^{\min} \geq p_j^{\min}$ for all $j \in G_2$. Since $p_j > p_j^{\min}$ for all $j \in G_2 \setminus S'$ and $p_j = p_j^{\min}$ for all $j \in S'$, buyers in $\Gamma(S')$ have in S' all their MBB goods at prices \mathbf{p} . Hence, λ_i for buyers $i \in \Gamma(S')$ does not change. Thus, $m_i^a = m_i^a(\mathbf{p}) = m_i^a(\mathbf{p}^{\min})$ for every $i \in \Gamma(S')$, which implies

$$\sum_{j \in S'} p_j^{\min} \leq \sum_{i \in \Gamma(S')} m_i^a . \quad (15)$$

Further, summing the left-hand-side inequality of constraints (14) and (12) for buyers in $\Gamma(S')$ and using the fact that $\lambda'_i > 0$, for all i , we get

$$\sum_{i \in \Gamma(S')} m_i^a = \sum_{i \in \Gamma(S'), j \in S'} f_{ij} \leq \sum_{j \in S'} p_j^{\min} - \sum_{j \in S'} \beta_j = \sum_{j \in S'} p_j^a .$$

Together with (15), this implies that $\beta_j = 0$ for all $j \in S'$, and all inequalities are equalities. Therefore $\sum_i f_{ij} = p_j^a$ for all $j \in S'$ and hence for all $j \in G_2$. \square

3.5.2. Lemke's Algorithm In this section, we apply Lemke's algorithm (see Appendix A for details) on the \mathcal{M} -LCP. For this, we first add an auxiliary, non-negative, scalar variable z in (12) and consider

$$-\sum_j f_{ij} - z \leq -(m_i - \delta_i) \quad \perp \quad \lambda'_i \geq 0 \quad \text{and} \quad z \geq 0. \quad (16)$$

This is the conjunction of the inequality $z \geq 0$ with a complementarity constraint. We denote by \mathcal{M} -LCP2 the set of constraints (10-11), (13-14) and (16). The primary ray of \mathcal{M} -LCP2 is $z \geq m_i - \delta_i$, for all i . The other variables are set to $\boldsymbol{\lambda}' = \mathbf{0}$, $\mathbf{p}' = \mathbf{0}$, $\mathbf{f} = \mathbf{0}$, $\delta_i = \max\{0, m_i - \Lambda c_i\}$, for all i , and $\beta_j = \max\{0, p_j^{\min} - d_j\}$, for all j . In the proof of the following theorem we show that under the money-clearing condition, there are no secondary rays in \mathcal{M} -LCP2. Hence, Lemke's algorithm applied to this LCP will converge to an equilibrium.

THEOREM 3.4. *Lemke's algorithm applied to \mathcal{M} -LCP2 converges to an equilibrium in money-clearing markets.*

Proof. We prove the result by contradiction. Suppose Lemke's algorithm converges on a secondary ray R , which starts at a vertex $(\boldsymbol{\lambda}'_*, \mathbf{p}'_*, \mathbf{f}_*, \boldsymbol{\delta}_*, \boldsymbol{\beta}_*, z_*)$ where $z_* > 0$ (otherwise, the algorithm would have stopped in the vertex) and extends in direction $(\boldsymbol{\lambda}'_o, \mathbf{p}'_o, \mathbf{f}_o, \boldsymbol{\delta}_o, \boldsymbol{\beta}_o, z_o)$, i.e.,

$$R = \{(\boldsymbol{\lambda}'_*, \mathbf{p}'_*, \mathbf{f}_*, \boldsymbol{\delta}_*, \boldsymbol{\beta}_*, z_*) + \alpha(\boldsymbol{\lambda}'_o, \mathbf{p}'_o, \mathbf{f}_o, \boldsymbol{\delta}_o, \boldsymbol{\beta}_o, z_o) \mid \alpha \geq 0\}.$$

Observe that $(\boldsymbol{\lambda}'_o, \mathbf{p}'_o, \mathbf{f}_o, \boldsymbol{\delta}_o, \boldsymbol{\beta}_o, z_o) \geq 0$ since all variables of \mathcal{M} -LCP2 are constrained to be non-negative. We consider three cases and show a contradiction in each of them.

Case 1: $\mathbf{p}'_o > \mathbf{0}$: On R , the following hold:

- Since all prices are increasing, all β_j 's are increasing due to (13). Thus for large α , the left inequality of (13) is tight and hence $\mathbf{p} - \mathbf{d} = \boldsymbol{\beta}$. Thus $p_j^a = p_j' + p_j^{\min} - \beta_j = d_j$ for all j .
- Further, $\sum_i f_{ij} = p_j^a$ for all j by (14) since prices \mathbf{p}' are positive.
- Consider any pair (i, j) with $f_{ij} > 0$. Then $\lambda'_i > 0$ by (9) and (10).
- Let $S = \{i \in B_2 \mid \lambda'_i > 0\}$. For $i \in S$, the left inequality in (16) is an equality. Summing over $i \in S$, we obtain $\sum_{i \in S} (m_i - \delta_i) - z|S| = \sum_j d_j$. By the money clearing property (1), $\sum_{i \in S} m_i \leq \sum_j d_j$ and hence $z = 0$ and $z_* = 0$; a contradiction.

Case 2: $\mathbf{p}'_0 = \mathbf{0}$: On R , the following hold:

- The vector of all prices remains constant and equal to $\mathbf{p}^{\min} + \mathbf{p}'_*$.
- $\boldsymbol{\lambda}'_0 = \mathbf{0}$ since otherwise the left inequality in (10) does not hold for large α . Thus $\boldsymbol{\lambda}'$ is constant: $\boldsymbol{\lambda}' = \boldsymbol{\lambda}'_*$.
- $\mathbf{f}_0 = \mathbf{0}$ as otherwise the left inequality in (14) does not hold for large α . Thus, $\mathbf{f} = \mathbf{f}_*$.
- Since \mathbf{f} , \mathbf{p} , and $\boldsymbol{\lambda}$ are constant, $\boldsymbol{\beta}$ must be constant by (14) and $\boldsymbol{\delta}$ must be constant by (11). Thus, $\boldsymbol{\delta}_o = \mathbf{0}$ and $\boldsymbol{\beta}_o = \mathbf{0}$.
- Since a direction vector cannot be zero, we must have $z_o > 0$. Note that all other components of the direction vector are zero.
- Since $z_o > 0$, (16) implies $\boldsymbol{\lambda}' = \mathbf{0}$ and, hence, $\boldsymbol{\lambda}'_* = \mathbf{0}$.
- We further conclude $\mathbf{f}_* = \mathbf{0}$ from (9) and (10), then $\mathbf{p}'_* = \mathbf{0}$ from (14), $(\boldsymbol{\delta}_*)_i = \max\{0, m_i - c_i \Lambda\}$ from (11), and finally $(\boldsymbol{\beta}_*)_j = \max\{0, p_j^{\min} - d_j\}$ from (13).

This means that the ray is not a secondary ray, but the primary ray; a contradiction.

Case 3: $\mathbf{p}'_o \not\succ \mathbf{0}$ and $\mathbf{p}'_o \neq \mathbf{0}$: On R , the following hold:

- Some prices are increasing and some are constant on R . Let $S' = \{j \in G_2 \mid (p'_o)_j > 0\}$ be the set of goods with increasing prices.
- For $j \in S'$, β_j must be increasing by (13). Hence, the left inequality in (13) must be an equality. Thus, $p_j^a = d_j$, and by (14) $\sum_i f_{ij} = d_j$.

- The prices of all goods in S' are increasing to infinity, and these goods are sold up to their maximum possible revenue. Hence, the buyers who buy these goods have zero utility for the goods outside S' , because each buyer buys an optimal bundle and the prices of goods outside S' are constant. Let $\Gamma(S')$ be the set of buyers buying goods in S' on R . Then $\lambda'_i > 0$ for all $i \in \Gamma(S')$ due to (9) and (10), and hence the left inequality in (16) is an equality.
- Summing (16) over $i \in \Gamma(S')$, we obtain $\sum_{i \in \Gamma(S')} (m_i - \delta_i) - z|\Gamma(S')| = \sum_{j \in S'} d_j$. By the money clearing property $\sum_{i \in \Gamma(S')} m_i \leq \sum_{j \in S'} d_j$. Thus $z = 0$ and $z_* = 0$, a contradiction.

This completes the proof. \square

We note that the algorithm of [8] guarantees that the price vector \mathbf{p}^{min} is rational with a bit length that is polynomial in the size of the input (i.e., in the sum of the logarithm of every u_{ij}, m_i, d_j, c_i). This further implies that Λ is rational with polynomial bit length. Hence, the polyhedron associated with \mathcal{M} -LCP2 is defined by rational numbers with a bit length that is polynomial in the size of the input.

COROLLARY 3.3. *The problem of computing a thrifty and modest equilibrium in money-clearing markets is in the class PPAD.*

Proof. By Theorem 3.4, Lemke's algorithm must converge to an equilibrium for money-clearing markets \mathcal{M} . Note that Lemke's algorithm traces a path on the 1-skeleton of a polyhedron. Let v be a vertex on the path found by Lemke's algorithm. To prove membership in PPAD, we need to show that the unique predecessor and successor of v on this path can be found efficiently. Clearly, these two vertices, say u and w , can be found simply by pivoting. To determine which vertex leads to the start of the path, i.e., the primary ray, and which leads to the end, we use a result by Todd [54] on the orientability of the path followed by a complementary pivot algorithm. It shows that the signs of the sub-determinants of tight constraints satisfied by the vertices u , v and w provide the orientation of the path. This concludes the proof of membership in PPAD. \square

REMARK 3.3. We note that a money-clearing market \mathcal{M} can be reduced to a more general Leontief-free market [33]. However, the agents in the reduced market remain satiated because buyers and sellers in \mathcal{M} are thrifty. The results for Leontief-free markets in [33] (such as membership in PPAD) require non-satiation of agents. Hence, these results are not directly applicable to markets \mathcal{M} via such a reduction.

4. Approximating the Nash Social Welfare

4.1. Constant-Factor Approximation for Capped Linear Valuations In this section, we present a $(2e^{1/(2e)} + \varepsilon)$ -approximation algorithm for maximizing Nash social welfare with capped linear (or budget-additive) valuations, for every constant $\varepsilon > 0$.

Consider maximizing Nash social welfare when allocating a set G of indivisible items to a set B of agents with capped linear valuations. As a first step, we execute a simple adjustment to the valuation functions. Note that if for some agent $i \in B$ and some item $j \in G$ we have $v_{ij} \geq c_i$, we can equivalently assume that $v_{ij} = c_i$ since the valuation of i can be at most c_i . More formally, let $v'_{ij} = \min(v_{ij}, c_i)$ and $v'_i(\mathbf{x}_i^S) = \min(c_i, \sum_{j \in G} v'_{ij} x_{ij}^S)$. The following lemma is straightforward and its proof is omitted.

LEMMA 4.1. $v'_i(\mathbf{x}) = v_i(\mathbf{x})$ for every integral allocation \mathbf{x} .

Henceforth, we will assume that $v_{ij} \leq c_i$, for all $i \in B$, $j \in G$.

To solve the integral maximization problem, consider the following convex program which constitutes a natural fractional relaxation.

$$\begin{aligned}
 & \text{maximize} \quad \left(\prod_{i \in B} \left(\sum_{j \in G} v_{ij} x_{ij} \right) \right)^{1/n} \\
 & \text{s.t.} \quad \sum_{j \in G} v_{ij} x_{ij} \leq c_i \quad \text{for all } i \in B, \\
 & \quad \sum_{i \in B} x_{ij} \leq 1 \quad \text{for all } j \in G, \\
 & \quad x_{ij} \geq 0 \quad \text{for all } i \in B, j \in G.
 \end{aligned} \tag{17}$$

The optimal solution to this program is the allocation vector of a thrifty and modest equilibrium in a Fisher market, in which agent i has a linear utility with $u_{ij} = v_{ij}$, a utility limit c_i , and a budget $m_i = 1$ (for details, see, e.g. [8, 17]).

Unfortunately, the optimal fractional allocation of this program can be significantly better in terms of the Nash social welfare than any integral solution with all $x_{ij} \in \{0, 1\}$ (see [18] for an example, in which the ratio is exponential in $|B|$). Hence, as in [8, 17, 18], we introduce additional earning limits $d_j = 1$ for all $j \in G$. We will see that this lowers the achievable objective function value in equilibrium and allows us to round the fractional equilibrium allocation to an integral one that approximates the optimal Nash social welfare.

Consider the resulting Fisher market \mathcal{M} with utility and earning limits. Our first observation is that non-trivial instances of the Nash social welfare problem give rise to a market \mathcal{M} that is money clearing.

LEMMA 4.2. *Consider the Fisher market \mathcal{M} resulting from an instance of the Nash social welfare problem. If the market \mathcal{M} is not money clearing, then the maximum Nash social welfare for indivisible items is 0.*

Proof. Obviously, if market \mathcal{M} is not money clearing, then there exists a subset B' of buyers such that the sum of the earning caps of goods in $\Gamma(B') = \{j \mid v_{ij} > 0, i \in B'\}$ is smaller than the sum of the budgets of buyers in B' . This implies that $|\Gamma(B')| < |B'|$. Hence, there is no allocation where each agent in B' gets at least one item of positive valuation. Thus, the Nash social welfare must always be 0. \square

When the market is not money clearing, every integral allocation has the optimal Nash social welfare. It is easy to check the money-clearing condition (1) by a max-flow computation.

Hence, for the remainder of this section, we assume that the instance of the Nash social welfare problem is non-trivial, i.e., the resulting Fisher market is money clearing. We have seen in Section 3.1 above that a money-clearing market \mathcal{M} always has a thrifty and modest equilibrium. Suppose we are given such an equilibrium (\mathbf{x}, \mathbf{p}) .

The Nash social welfare objective allows scaling the valuation function of every agent i by any factor $\gamma_i > 0$. This adjustment changes neither the equilibrium, the integral optimum solution of the Nash social welfare problem, nor the approximation factor. Given the equilibrium (\mathbf{x}, \mathbf{p}) , we want to normalize the valuation function for agent i based on the MBB ratio α_i of buyer i in the market equilibrium.

In equilibrium, there can be a set of goods $G_0 = \{j \mid p_j = 0\}$. All buyers $B_0 = \{i \mid u_{ij} > 0 \text{ for some } j \in G_0\}$ interested in any good $j \in G_0$ have an infinite MBB ratio. Due to our equilibrium conditions, every $i \in B_0$ must be capped and receive allocation only from G_0 , i.e., $u_i(\mathbf{x}) = c_i$ and $x_{ij} > 0$ only if $j \in G_0$ and $u_{ij} > 0$. Moreover, since no buyer $i \in B \setminus B_0$ has positive utility for

any of the goods G_0 , these goods are allocated only to B_0 . Therefore, we can treat items G_0 and agents B_0 separately in the analysis.

For all $i \in B \setminus B_0$, we normalize $v'_{ij} = v_{ij}/\alpha_i$ and $c'_i = c_i/\alpha_i$. This does not change the demand bundle for buyer i , and thus (\mathbf{x}, \mathbf{p}) remains an equilibrium. In the resulting instance, every such buyer has an MBB ratio of 1 in (\mathbf{x}, \mathbf{p}) . Consequently, $v'_{ij} \leq p_j$ for all $i \in B \setminus B_0, j \in G$, where equality holds if and only if j is an MBB good of buyer i . For simplicity we assume that v and c fulfill these conditions directly, i.e., $v_{ij} = v'_{ij}$ and $c_i = c'_i$. Together with the fact that $v_{ij} \leq c_i$ for all (i, j) this implies

$$v_{ij} \leq \min(p_j, c_i), \quad \text{for all } i \in B \setminus B_0 \text{ and all } j \in G. \quad (18)$$

The following lemma is a helpful insight on the structure of equilibria.

LEMMA 4.3. *Consider a money-clearing market \mathcal{M} with $m_i = 1$, $d_j = 1$, and $v_{ij} \leq c_i$, for all $i \in B, j \in G$. Suppose we normalize the utilities based on a thrifty and modest equilibrium (\mathbf{x}, \mathbf{p}) . Then the following properties hold.*

- (a) *A buyer $i \in B \setminus B_0$ spends $m_i^a = \min(1, c_i)$ units of money. His valuation $v_i(\mathbf{x})$ equals m_i^a . If i is capped, $c_i \leq 1$.*
- (b) *If i is capped, he is allocated at least one unit of goods.*
- (c) *If i is capped and j is an MBB good for buyer i , then $p_j \leq 1$.*
- (d) *If $p_j < 1$, j is completely sold.*
- (e) *The money spent on good j is $p_j^a = \min(p_j, 1)$.*

Proof. An uncapped buyer $i \in B \setminus B_0$ spends his entire budget as otherwise \mathbf{x}_i would not be a demand bundle. Since the MBB ratio is 1, the valuation $v_i(\mathbf{x})$ equals the money spent by i . If i is capped, his valuation equals c_i , and hence the money spent equals c_i , so $c_i \leq 1$.

Since $v_{ij} \leq c_i$ always and $c_i = v_i(\mathbf{x})$ for a capped buyer, $c_i = v_i(\mathbf{x}) \leq c_i \sum_j x_{ij}$ and hence $\sum_j x_{ij} \geq 1$.

If j is an MBB good for a capped buyer i , then $p_j = v_{ij} \leq \min(p_j, c_i)$ according to (18) and hence $p_j \leq c_i \leq 1$, where the last inequality was established in (a).

If $0 < p_j < 1$, the supply $e_j = \min(1, d_j/p_j) = \min(1, 1/p_j) = 1$. Thus j is completely sold.

Finally, the money spent on j is $p_j e_j = p_j \min(1, 1/p_j) = \min(p_j, 1) = p_j^a$. \square

Our subsequent analysis proceeds as follows. First, in Section 4.1.1, we describe an upper bound on the optimal Nash social welfare of any integral solution. The upper bound is based on the properties of any thrifty and modest equilibrium described above. Then, in Section 4.1.2, we show how to round an equilibrium and obtain an integral solution which is a $2e^{1/(2e)}$ -approximation to the optimal Nash social welfare. Finally, in Section 4.1.3, since our FPTAS from Section 3.2 computes an exact equilibrium in a perturbed market, we discuss the impact of perturbation on the approximation guarantee.

4.1.1. Upper Bound In this section, we describe an upper bound on the optimal Nash social welfare when valuations are normalized based on an equilibrium (\mathbf{x}, \mathbf{p}) . The bound relates to prices and utility caps of the capped buyers in (\mathbf{x}, \mathbf{p}) . We denote by B_c and B_u the set of capped and uncapped buyers in (\mathbf{x}, \mathbf{p}) , respectively. Recall that since (\mathbf{x}, \mathbf{p}) is a thrifty and modest equilibrium, buyers may not spend their entire budget and sellers may not sell their entire supply. We denote by $m_i^a = \min(1, c_i)$ the active budget of buyer i and by $p_j^a = \min(p_j, 1)$ the active price of good j . The following result is a generalization of a similar bound shown in [18].

THEOREM 4.1. *For valuations v and caps c normalized according to equilibrium prices \mathbf{p} ,*

$$\left(\prod_{i \in B} v_i(\mathbf{x}^*) \right)^{1/n} \leq \left(\prod_{i \in B_c} c_i \prod_{j: p_j > 1} p_j \right)^{1/n},$$

where x^* is an integral allocation that maximizes the Nash social welfare.

Proof. Consider an integral allocation \mathbf{x}^* that maximizes the Nash social welfare. For the agents $i \in B_0 \subseteq B_c$, a simple upper bound is $\prod_{i \in B_0} v_i(\mathbf{x}^*) \leq \prod_{i \in B_0} c_i$. Next, we obtain an upper bound on $\prod_{i \in B \setminus B_0} v_i(\mathbf{x}^*)$ using the market equilibrium (\mathbf{x}, \mathbf{p}) . The starting point is the observation that for a buyer who receives a good, due to the rescaling, his valuation is equal to the good's price.

Step 1. Consider the set G_h of goods with a price greater than 1. Let $\Gamma(G_h)$ be the set of buyers who buy some good in G_h at x , i.e., $\Gamma(G_h) := \{i \in B \mid x_{ij} > 0, j \in G_h\}$. Note that every agent of $\Gamma(G_h)$ is uncapped due to Lemma 4.3(c), i.e., $\Gamma(G_h) \cap B_c = \emptyset$. Next, allocate the goods in G_h to $\Gamma(G_h)$, one per buyer, but only to buyers whose valuation equals the price. This is possible because at equilibrium (\mathbf{x}, \mathbf{p}) , there is one unit of spending on each of these goods, and each buyer has one unit of money. Thus, for every subset $F \subseteq G_h$, the set of buyers interested in F is of size at least $|F|$, which implies by Hall's theorem that a perfect matching exists for these goods and buyers. Let $B_h \subseteq \Gamma(G_h)$ be the set of buyers who each gets one good of G_h in this matching. Clearly, $|B_h| = |G_h|$. Remove the goods in G_h and the buyers in B_h from further consideration.

Step 2. We now increase the valuations of the remaining buyers (i.e., $B \setminus (B_0 \cup B_h)$) for the remaining goods (i.e., $G \setminus (G_h \cup G_0)$) so that on every good they are equal to the prices, i.e., we set $v'_{ij} = p_j$ for all (i, j) . Note that the Nash social welfare can only increase. Arbitrarily allocate the remaining goods fractionally so that each remaining buyer i receives valuation $\min\{c_i, 1\}$. This is possible because the sum of the prices of all remaining goods is exactly equal to $\sum_{i \in B_c \setminus B_0} c_i + |B \setminus B_c| - |B_h|$.

Clearly, as shown in [18], the resulting Nash social welfare upper bounds the Nash welfare at \mathbf{x}^* and is equal to

$$\left(\prod_{i \in B_c} c_i \prod_{j: p_j > 1} p_j \right)^{1/n}.$$

□

4.1.2. Rounding Equilibria In this section, we give an algorithm to round a fractional allocation of a thrifty and modest equilibrium (\mathbf{x}, \mathbf{p}) to an integral one. Without loss of generality, we may assume that the allocation graph $(B \cup G, E)$ with $E = \{(i, j) \in B \times G \mid x_{ij} > 0\}$ is a forest [49, 24]. In the following, we only discuss how to round the trees in $(B \setminus B_0) \times (G \setminus G_0)$. For trees in $B_0 \times G_0$, the rounding and the analysis are very similar, but independent of prices and slightly simpler (see Appendix B). Consider the following procedure:

Preprocessing: It consists of three substeps.

- (a) For each tree component of the allocation graph, assign some agent to be a root node.
- (b) For every good j keep at most one child agent. This child-agent i must buy the largest amount of j among the child agents (ties are broken arbitrarily) and must have an active budget which is less than twice the price of j , i.e., $m_i^a/2 < p_j$. In other words, child agent i is cut off from good j if a sibling buys more of good j (ties are broken arbitrarily) or if $p_j \leq m_i^a/2$. Note that if a sibling buys more of good j , the sibling also spends more on good j .
- (c) Agents whose connection to their parent-good is severed in step (b) become roots.

Rounding: It consists of two substeps.

- (a) Goods with no child agent are assigned to their parent agent.
- (b) For each non-trivial tree component, do the following recursively: Assign the root agent a child good j that gives the maximum value (among all children goods) in the fractional solution. Except in the subtree rooted at j , assign each good to its child agent in the remaining tree. Make the child agent of good j the root node of a newly created tree.

Let us define the remaining valuation of a buyer after preprocessing as the sum of the fractional values for the goods at the ends of the remaining edges incident on the buyer.

LEMMA 4.4. *After preprocessing, the remaining valuation of each root agent r is at least $v_r(\mathbf{x})/2$. For all other agents i the valuation is at least $v_i(\mathbf{x})$. If good j has child agents and $p_j > 1$, then j keeps a child agent.*

Proof. Whenever an agent i loses allocation because the connection to the parent-good j is cut, a new tree component is being created, and i becomes its root node. If i is cut from j , then either $x_{ij} \leq 1/2$ or $p_j \leq m_i^a/2$. In either case, since $v_{ij} \leq c_i$ for all (i, j) , i receives at most half of his utility from j . For a good j with child agents and $p_j > 1$, the child agent i who buys most of j is kept as a child since $p_j > 1/2 \geq m_i^a/2$. \square

LEMMA 4.5. *After step (a) of the rounding, each tree component T has $k_T + 1$ agents and k_T goods for some $k_T \geq 0$. Suppose agent i in T is assigned a good j with $p_j > 1$ during step (a) of the rounding. Then T consists of only i and j , the valuation of i after rounding is p_j , and $B_c \cap T = \emptyset$.*

Proof. The first part is straightforward since after step (a) of the rounding, every remaining good has exactly one parent agent and one child agent. For the second part, consider any good j with price $p_j > 1$. If j has children-agents, the one that spends most on j stays as a child, because $m_{i'}^a \leq 1 < 2p_j$ for any agent i' . This implies that if j is a leaf of the initial forest, only its parent agent spends money on it, call it i . Since the money inflow into j is one and i has only one unit to spend, only i spends on j and i spends one unit on j . Thus T consists of only i and j . Further, $v_{ij} = p_j$, since j is an MBB good for i . Hence, $p_j \leq c_i$ by (18). The valuation of i after rounding is $p_j > 1$, and i is not capped in the equilibrium. This implies $B_c \cap T = \emptyset$. \square

LEMMA 4.6. *After rounding, each agent i who is assigned his parent good obtains a valuation of at least $v_i(\mathbf{x})/2$.*

Proof. Consider any good j in the tree in the rounding step. Since j was not assigned to its parent agent during rounding step (a), its price is at least half of the active budget of its child agent i , i.e., $p_j \geq m_i^a/2$. Since j is MBB for i , we see $v_{ij} = p_j$. From this good the child-agent obtains a valuation of at least half of the valuation in the equilibrium. \square

Consider a tree T at the beginning of step (b) of the rounding with $k_T + 1$ agents and k_T goods. Assume $k_T \geq 1$ first. Let $a_1, g_1, a_2, g_2, \dots, a_l, g_\ell, a_{\ell+1}$ be the *recursion path* in T starting from the root agent a_1 and ending at the leaf agent $a_{\ell+1}$ such that $a_1, \dots, a_{\ell+1}$ became root agents of the trees formed recursively during the rounding step, and good g_i is assigned to a_i in this process, for $1 \leq i \leq \ell$. Note that $a_{\ell+1}$ is not assigned any good in step (b) of the rounding. However, as the proof of the following lemma shows, $a_{\ell+1}$ must have been assigned some good during step (a) of the rounding. Let k_i denote the number of agent a_i 's children, for $1 \leq i \leq \ell$. If $k_T = 0$, then $\ell = 0$ and $a_1 = a_{\ell+1}$ is the root of a tree containing no goods after step (a) of the rounding.

LEMMA 4.7. *The product of the valuations of agents in T in the rounded solution is at least*

$$\left(\frac{1}{2}\right)^{k_T-\ell+1} \cdot \frac{1}{k_1 \cdots k_\ell} \cdot \prod_{i \in T \cap B_c} c_i \prod_{j \in T: p_j > 1} p_j \cdot$$

Proof. Let $\bar{c}_i = \min\{1, c_i\}$ for all $i \in B$.

We first deal with the case $k_T = 0$. Then $\ell = 0$. If a good j of price $p_j > 1$ is assigned to a_1 during step (a) of the rounding, then the valuation of a_1 after the rounding is p_j and $B_c \cap T = \emptyset$ by Lemma 4.5. This establishes the claim even without the leading factor $1/2$. If all goods assigned to a_1 during step (a) of the rounding have price at most one, then $\{j \in T : p_j > 1\} = \emptyset$ and $T \cap B_c \subseteq$

$\{a_1\}$. Moreover, the value of a_1 after the preprocessing is at least $v_{a_1}(\mathbf{x})/2 = \bar{c}_1/2$. The rounding does not decrease this value.

We turn to the case $k_T \geq 1$. Let $q_i = x_{a_i, g_i} > 0$ be the amount of good g_i bought by agent a_i in the equilibrium, for $1 \leq i \leq l$. Then a_i spends $q_i p_i$ on good g_i .

In the market equilibrium, the root agent a_1 receives at least half of the valuation from his children. Thus $q_1 p_1 \geq \bar{c}_1/(2k_1)$.

We next show that agent a_i , $2 \leq i \leq \ell + 1$, receives at least value $q_{i-1} \max(\bar{c}_i, p_{i-1})$ from his children in the equilibrium. Note that agent a_i can spend at most $\min\{p_{i-1}, 1\} - q_{i-1} p_{i-1} \leq \bar{c}_i - q_{i-1} p_{i-1}$ on good g_{i-1} : If a_i is capped, then $p_{i-1} \leq c_i = \bar{c}_i$ as good g_{i-1} is MBB for a_i , and if a_i is not capped, then, by (18), $\bar{c}_i = 1$. Thus a_i must spend $q_{i-1} p_{i-1}$ on direct children in the equilibrium. This establishes the claim if $\bar{c}_i \leq p_{i-1}$. So assume $p_{i-1} < \bar{c}_i$. Agent a_i can receive at most a fraction $1 - q_{i-1}$ of good g_{i-1} . Hence the value a_i receives from this good is at most $(1 - q_{i-1}) p_{i-1} \leq (1 - q_{i-1}) \bar{c}_i$. Thus a_i must receive value at least $q_{i-1} \bar{c}_i$ from children goods.

Now $q_i p_i \geq q_{i-1} \max(\bar{c}_i, p_{i-1})/k_i$ for $2 \leq i \leq \ell$, since agent a_i spends $q_i p_i$ on good g_i , and this is at least a fraction $1/k_i$ of what he spends totally on his children.

The product of the valuations of a_1 to $a_{\ell+1}$ in the rounded solution is at least $p_1 \dots p_\ell \cdot q_\ell \max(\bar{c}_{\ell+1}, p_\ell)$. This holds since g_i is assigned to a_i , for $1 \leq i \leq \ell$, and $a_{\ell+1}$ receives a value at least $q_\ell \max(\bar{c}_{\ell+1}, p_\ell)$ from his children in the equilibrium. Since these children are assigned to $a_{\ell+1}$ during step (a) of the rounding, he receives at least this value in the rounded solution.

Combining the arguments above we obtain

$$\begin{aligned}
 & p_1 \dots p_\ell \cdot q_\ell \cdot \max(\bar{c}_{\ell+1}, p_\ell) \\
 & \geq \frac{\bar{c}_1}{2q_1 k_1} \cdot \frac{q_1 \max(\bar{c}_2, p_1)}{q_2 k_2} \dots \frac{q_{\ell-1} \max(\bar{c}_\ell, p_{\ell-1})}{q_\ell k_\ell} \cdot q_\ell \cdot \max(\bar{c}_{\ell+1}, p_\ell) \\
 & = \frac{1}{2} \left(\frac{1}{k_1 \dots k_\ell} \right) \bar{c}_1 \cdot \prod_{2 \leq i \leq \ell+1} \max(\bar{c}_i, p_{i-1}) \\
 & \geq \frac{1}{2} \left(\frac{1}{k_1 \dots k_\ell} \right) \prod_{1 \leq i \leq \ell+1} \bar{c}_i \cdot \prod_{\substack{1 \leq i \leq \ell, \\ p_i > 1}} p_i \\
 & \geq \frac{1}{2} \left(\frac{1}{k_1 \dots k_\ell} \right) \prod_{\substack{1 \leq i \leq \ell+1, \\ i \in T \cap B_c}} c_i \cdot \prod_{\substack{1 \leq i \leq \ell, \\ p_i > 1}} p_i,
 \end{aligned} \tag{19}$$

where the next to last inequality follows from $\max(\bar{c}_i, p_{i-1}) \geq \bar{c}_i \cdot \max(1, p_{i-1})$, for all i .

By Lemma 4.6, each of the remaining $k_T - \ell$ agents in T gets value at least $\max(v_i(\mathbf{x})/2, p) \geq \max(\bar{c}_i/2, p)$, where p is the price of the parent-good. This implies that each remaining agent i in T receives valuation at least $\bar{c}_i/2$. Further, every good j with $p_j > 1$ is assigned to a distinct agent, and the bound in (19) already applies for the agents on the recursive path. Combining these observations, we obtain the following lower bound for the product of the valuations of agents in T :

$$\left(\frac{1}{2} \right)^{k_T - \ell + 1} \left(\frac{1}{k_1 \dots k_\ell} \right) \prod_{i \in T \cap B_c} c_i \prod_{j \in T: p_j > 1} p_j.$$

□

THEOREM 4.2. *The rounding procedure gives a $2e^{1/2e}$ -approximation for the optimal Nash social welfare with capped linear valuations.*

Proof. Suppose there are trees T^1, T^2, \dots, T^a at the beginning of the rounding. Let $k^i + 1$ and k^i be the number of agents and goods in tree T^i , respectively. Let $l^i + 1$ be the number of agents on

the path in T^i traced during the rounding step, and let $k_1^i, \dots, k_{l_i}^i$ be the number of children goods for agents along that path.

The bound in Lemma 4.7 for trees $T \subseteq (B \setminus B_0) \times (G \setminus G_0)$ can also be obtained for our rounding of trees $T \subseteq B_0 \times G_0$ (Lemma B.4 in the Appendix). Thus, the Nash social welfare of the rounded solution is at least

$$\begin{aligned} & \left(\left(\frac{1}{2} \right)^{\sum_{i=1}^a (k^i - l^i + 1)} \left(\frac{1}{k_1^1 \dots k_{l_1}^1 k_1^2 \dots k_{l_2}^2 \dots k_1^a \dots k_{l_a}^a} \right) \prod_{i \in B_c} c_i \prod_{j: p_j > 1} p_j \right)^{1/n} \\ &= \frac{1}{2} \cdot 2^{\sum_{i=1}^a l^i / n} \cdot \left(\frac{1}{\prod_{i=1}^a \prod_{j=1}^{l^i} k_j^i} \right)^{1/n} \left(\prod_{i \in B_c} c_i \prod_{j: p_j > 1} p_j \right)^{1/n} \\ &\geq \frac{1}{2} \left(\frac{2^{\sum_{i=1}^a l^i}}{\sum_{i=1}^a \sum_{j=1}^{l^i} k_j^i} \right)^{\sum_{i=1}^a l^i / n} \left(\prod_{i \in B_c} c_i \prod_{j: p_j > 1} p_j \right)^{1/n} \geq \frac{1}{2e^{1/2e}} \left(\prod_{i \in B_c} c_i \prod_{j: p_j > 1} p_j \right)^{1/n}. \end{aligned}$$

The first equation uses $\sum_{i=1}^a (k_i + 1) = n$, which implies $\sum_{i=1}^a (k^i - l^i + 1) = n - \sum_{i=1}^a l^i$. With a $1/n$ -factor in the exponent stemming from the outer bracket, the first expression becomes $(1/2)^{1 - \sum_i l^i / n}$. The subsequent inequality follows from the standard relation of arithmetic and geometric means applied to the set of all k_j^i , i.e., $\left(\prod_i \prod_j k_j^i \right)^{1/n} \leq \left(\prod_i \prod_j k_j^i \right)^{1/\sum_i l^i} \leq \sum_i \sum_j k_j^i / \sum_i l^i$. The last inequality uses $\sum_{i=1}^a \sum_{j=1}^{l^i} k_j^i \leq n$ and the fact that $(2x)^x$ is minimal at $x = 1/2e$. \square

4.1.3. Rounding Equilibria of Perturbed Markets Given a parameter $\varepsilon' > 0$, our FPTAS in Section 3.2 computes an exact equilibrium for a perturbed market, which results when agents have perturbed valuations $\tilde{v}_i(\mathbf{x}) = \min(c_i, \sum_j \tilde{v}_{ij} x_{ij})$ with the same caps c_i and $\tilde{v}_{ij} \geq v_{ij} \geq \tilde{v}_{ij}/(1 + \varepsilon')$. Suppose we apply our rounding algorithm to the exact equilibrium for \tilde{v} . It obtains an allocation S such that

$$\begin{aligned} \left(\prod_i v_i(\mathbf{x}_i^S) \right)^{1/n} &\geq \frac{1}{(1 + \varepsilon')} \left(\prod_i \tilde{v}_i(\mathbf{x}_i^S) \right)^{1/n} \\ &\geq \frac{1}{(1 + \varepsilon')} \cdot \frac{1}{2e^{1/2e}} \left(\prod_i \tilde{v}_i(\mathbf{x}^*) \right)^{1/n} \\ &\geq \frac{1}{(1 + \varepsilon') \cdot 2e^{1/2e}} \cdot \left(\prod_i v_i(\mathbf{x}^*) \right)^{1/n}. \end{aligned}$$

If we apply the FPTAS with ε' , then this yields an approximation ratio of at most $2e^{1/2e} + \varepsilon$ for $\varepsilon = 2e^{1/(2e)} \varepsilon'$. We summarize our main result:

COROLLARY 4.1. *For every $\varepsilon > 0$ there is an algorithm with running time polynomial in $n, m, \log \max_{i,j} \{v_{ij}, c_i\}$, and $1/\varepsilon$ that computes an allocation which is a $(2e^{1/2e} + \varepsilon)$ -approximation to the optimal Nash social welfare.*

4.2. Hardness of Approximation In this section, we prove a hardness result for approximation of the maximum Nash social welfare with additive valuations. The best previous bound was a lower bound of 1.00008 [42]. Our improved bound of $\sqrt{8/7} > 1.069$ follows by adapting a construction in [13] for (sum) social welfare with capped linear valuations.

THEOREM 4.3. *For every constant $\delta > 0$, there is no $(\sqrt{8/7} - \delta)$ -approximation algorithm for maximizing Nash social welfare with additive valuations unless $P = NP$.*

For clarity, we first describe the proof for capped linear valuations. Subsequently, we show how to drop the assumption of caps.

LEMMA 4.8. *For every constant $\delta > 0$, there is no $(\sqrt{8/7} - \delta)$ -approximation algorithm for maximizing Nash social welfare with capped linear valuations unless $P = NP$.*

Proof. Chakrabarty and Goel [13] show hardness for (sum) social welfare by reducing from MAX-E3-LIN-2. An instance of this problem consists of n variables and m linear equations over $GF(2)$. Each equation consists of three distinct variables. For the Nash social welfare objective, we need slightly more structure in the optimal assignments. Therefore, we consider the stronger problem variant Ek-OCC-MAX-E3-LIN-2, in which each variable occurs exactly k times in the equations.

THEOREM 4.4 ([15]). *For every constant $\varepsilon \in (0, \frac{1}{4})$ there is a constant $k(\varepsilon)$, and a class of instances of Ek-OCC-MAX-E3-LIN-2 with $k \geq k(\varepsilon)$ such that $k \rightarrow \infty$ as $\varepsilon \rightarrow 0$, for which we cannot decide if the optimal variable assignment fulfills more than $(1 - \varepsilon)m$ equations or fewer than $(1/2 + \varepsilon)m$ equations, unless $P = NP$.*

Our reduction follows the construction in [13]. We only sketch the main properties here. For more details see [13, Section 4].

For each variable x_i we introduce two agents $\langle x_i : 0 \rangle$ and $\langle x_i : 1 \rangle$. Each of these agents has a cap of $c_i = 4k$, where k is the number of occurrences of x_i in the equations. Since in Ek-OCC-MAX-E3-LIN-2 every variable occurs exactly k times, we have $c_i = 4k$ for all agents.

For each variable x_i there is a *switch item*. The switch item has value $4k$ for agents $\langle x_i : 0 \rangle$ and $\langle x_i : 1 \rangle$, and value 0 for every other agent. It serves to capture the assignment of the variable – if x_i is set to $x_i = 1$, the switch item is given to $\langle x_i : 0 \rangle$ (for $x_i = 0$, the switch item goes to $\langle x_i : 1 \rangle$). Due to his cap, an agent cannot obtain additional value from receiving any other items.

The remaining items, called *equation items*, are defined as follows. For each equation $x_i + x_j + x_k = \alpha$ with $\alpha \in \{0, 1\}$, we introduce four classes of equation items – one class for each satisfying assignment. In particular, we get class $\langle x_i : \alpha; x_j : \alpha; x_k : \alpha \rangle$ as well as classes $\langle x_i : \bar{\alpha}, x_j : \bar{\alpha}, x_k : \alpha \rangle$, $\langle x_i : \bar{\alpha}, x_j : \alpha, x_k : \bar{\alpha} \rangle$ and $\langle x_i : \alpha, x_j : \bar{\alpha}, x_k : \bar{\alpha} \rangle$. For each of these classes, we introduce three items. Hence, for each equation we introduce twelve items in total. An item $\langle x_i : \alpha_i, x_j : \alpha_j, x_k : \alpha_k \rangle$ has a value of 1 for the three agents $\langle x_i : \alpha_i \rangle$, $\langle x_j : \alpha_j \rangle$, and $\langle x_k : \alpha_k \rangle$, and value 0 for every other agent.

Clearly, there is an optimal assignment of items to agents which assigns the switch item of every variable x_i to one of the two agents $\langle x_i : 0 \rangle$ or $\langle x_i : 1 \rangle$. If the switch item is assigned to a different agent, it has value 0 for that agent, so removing the item does not decrease the valuation and the Nash social welfare. Then giving it to any of the two agents $\langle x_i : 0 \rangle$ or $\langle x_i : 1 \rangle$ can only increase the Nash social welfare. Therefore, we can assume that an optimal assignment yields some variable assignment for the underlying instance of Ek-OCC-MAX-E3-LIN-2.

Consider an equation $x_i + x_j + x_k = \alpha$ that becomes satisfied by setting the variables $(x_i, x_j, x_k) = (\alpha_i, \alpha_j, \alpha_k)$. Then none of the agents $\langle x_i : \alpha_i \rangle$, $\langle x_j : \alpha_j \rangle$, and $\langle x_k : \alpha_k \rangle$ gets a switch item, and we can assign exactly four equation items to each of these agents (for details see [13]). Hence, all twelve equation items generate additional value. In particular, it follows that if x_i is involved in a satisfied equation, one of its agents gets a switch item, and the other one can receive at least four equation items.

Consider an equation $x_i + x_j + x_k = \alpha$ that becomes unsatisfied by setting the variables $(x_i, x_j, x_k) = (\alpha_i, \alpha_j, \alpha_k)$. Then for one class of equation items, all agents who value these items have already received switch items (for details see [13]). Hence, at most nine equation items generate additional value. They can be assigned to the agents who did not receive switch items so that each agent receives three items. In particular, it follows that if x_i is involved in an unsatisfied equation,

one of its agents gets a switch item, and the other one can receive at least three equation items. Hence, in every optimal solution the Nash social welfare is strictly larger than 0.

We now derive a lower bound on the optimal Nash social welfare when $(1 - \varepsilon)m$ equations can be satisfied. In this case, we obtain value $4k$ for the n agents that receive switch items. Moreover, we get an additional total value of $12m(1 - \varepsilon) + 9m\varepsilon$ generated by the equation items. Note that $m = kn/3$. We will lower bound the Nash social welfare of an optimal assignment in this case. For this, it suffices to consider the assignment indicated above – for each satisfied equation, all incident agents without switch items get four equation items. For each unsatisfied equation, all incident agents without switch items get 3 equation items. Then, every agent obtains a value in the range $[3k, 4k]$. A lower bound on the Nash social welfare when agents share the value $12m(1 - \varepsilon) + 9m\varepsilon$ is obtained when some agents have the maximum value of $4k$, while all others have the minimum value of $3k$. This occurs when $n(1 - \varepsilon)$ agents have value $4k$. Therefore, when an assignment of items to agents generates Nash social welfare of more than

$$((4k)^n \cdot (4k)^{n(1-\varepsilon)} \cdot (3k)^{n\varepsilon})^{\frac{1}{2n}} = k \cdot 4^{\frac{1}{2}} \cdot 4^{\frac{1}{2}} \cdot (3/4)^{\frac{\varepsilon}{2}},$$

we take this as an indicator that at least $m(1 - \varepsilon)$ equations can be fulfilled.

In contrast, now suppose only $(1/2 + \varepsilon)m$ equations can be fulfilled. In this case, we obtain value $4k$ for the n agents who receive the switch items. Moreover, we get an additional total value of at most $12m(1/2 + \varepsilon) + 9m(1/2 - \varepsilon) = 10.5m + 3\varepsilon m$ generated by the equation items. Next, we upper bound the Nash social welfare of such an assignment. To this end, we assume that all n agents who do not receive a switch item get an equal share of the value generated by equation items, i.e., a share of $3.5k + k\varepsilon$. Therefore, when an assignment of items to agents generates Nash social welfare of at most

$$((4k)^n \cdot (k(3.5 + \varepsilon))^n)^{\frac{1}{2n}} = k \cdot 4^{\frac{1}{2}} \cdot (3.5 + \varepsilon)^{\frac{1}{2}},$$

we can take this as an indicator that at most $m(1/2 + \varepsilon)$ equations can be fulfilled.

Hence, if we can approximate the optimal Nash social welfare by at most a factor of

$$\frac{4^{\frac{1}{2}} \cdot (3/4)^{\frac{\varepsilon}{2}}}{(3.5 + \varepsilon)^{\frac{1}{2}}} = \left(\frac{4 \cdot (3/4)^{\varepsilon}}{3.5 + \varepsilon} \right)^{\frac{1}{2}},$$

we can decide whether the instance of Ek-OCC-MAX-E3-LIN-2 has an optimal assignment with at least $m(1 - \varepsilon)$ or at most $m(1/2 + \varepsilon)$ satisfied equations. This shows that, for every constant $\delta > 0$, there is no $(\sqrt{8/7} - \delta)$ -approximation algorithm for maximizing Nash social welfare with capped linear valuations unless $P = NP$. \square

We now show how to adjust the construction for additive valuations without caps.

Proof of Theorem 4.3. We use the same construction as in Lemma 4.8 with the following adjustments. A switch item for variable x_i now has a large value $M \gg 4k$ for agents $\langle x_i : 0 \rangle$ or $\langle x_i : 1 \rangle$. All agent valuations are additive and have no caps (i.e., all $c_i = \infty$).

First, we again establish the lower bound on the optimal Nash social welfare when $(1 - \varepsilon)m$ equations can be satisfied. Then we can assign the switch items so that $12m(1 - \varepsilon) + 9m\varepsilon$ equation items are each assigned to an agent who values it (i.e., it represents a literal in the corresponding equation) and does not receive a switch item. Each of the remaining $3m\varepsilon$ equation items is assigned to an agent who values it (i.e., it represents a literal in the corresponding equation) and does receive a switch item. Instead, to construct a lower bound on the achievable Nash social welfare, we simply drop these items from consideration. Therefore, a necessary condition to fulfill $m(1 - \varepsilon)$ equations is that there is an assignment of items to agents that generates Nash social welfare of at least

$$(M^n \cdot (4k)^{n(1-\varepsilon)} \cdot 3k^{n\varepsilon})^{\frac{1}{2n}} = k^{\frac{1}{2}} \cdot M^{\frac{1}{2}} \cdot 4^{\frac{1}{2}} \cdot (3/4)^{\frac{\varepsilon}{2}}.$$

Now suppose only $(1/2 + \varepsilon)m$ equations can be fulfilled. To construct an upper bound on the optimal Nash social welfare, we apply the same reasoning as above. n agents receive switch items according to an optimal variable assignment. The $12m(1/2 + \varepsilon) + 9m(1/2 - \varepsilon) = 10.5m + 3\varepsilon m$ equation items are assigned in equal shares to agents without switch items. The remaining $3m(1/2 - \varepsilon)$ items are assigned in equal shares to agents with switch items. Therefore, a necessary condition for the case when at most $m(1/2 + \varepsilon)$ equations can be fulfilled is that the optimal assignment of items to agents generates Nash social welfare of at most

$$\left(\left(M + k \left(\frac{1}{2} - \varepsilon \right) \right)^n \cdot (k(3.5 + \varepsilon))^n \right)^{\frac{1}{2n}} = k^{\frac{1}{2}} \cdot \left(M + k \left(\frac{1}{2} - \varepsilon \right) \right)^{\frac{1}{2}} \cdot (3.5 + \varepsilon)^{\frac{1}{2}} .$$

Hence, approximating the optimal Nash social welfare to within a factor of at most

$$\left(\frac{M}{M + k \left(\frac{1}{2} - \varepsilon \right)} \right)^{\frac{1}{2}} \cdot \left(\frac{4 \cdot (3/4)^\varepsilon}{3.5 + \varepsilon} \right)^{\frac{1}{2}}$$

allows us to distinguish between the cases whether the instance of Ek-OCC-MAX-E3-LIN-2 has an optimal assignment with at least $m(1 - \varepsilon)$ or at most $m(1/2 + \varepsilon)$ satisfied equations. The second fraction clearly grows to $\sqrt{8/7}$ as $\varepsilon \rightarrow 0$. For a fixed number M , the first fraction decreases, since $k = k(\varepsilon)$ increases with decreasing ε . However, we can choose a number $M = \omega(k)$ since k is the number of occurrences of a single variable and, thus, the input size is polynomial in k . For example, with $M = k^2$ the first term approaches 1 as $\varepsilon \rightarrow 0$ (and $k \rightarrow \infty$). This proves the theorem. \square

5. Future Directions There are many interesting questions arising from our work. Equilibria in linear Fisher markets with both earning and utility limits turn out to have intriguing structure. Although an equilibrium may not always exist in these markets, we showed that it always exists when the market satisfies the money clearing condition, i.e., money clearing is a (polynomial-time checkable) sufficiency condition for existence. While existence is guaranteed in this case, the set of equilibria is non-convex.

Our algorithm initially ignores all utility caps and then incorporates them using a descending-price approach. Conceivably, an analogous algorithm could be obtained by first ignoring all earning caps and then carefully incorporating them using a suitable ascending-price approach. Some interesting directions are to show whether (1) such an ascending-price algorithm exists, (2) all equilibria can be obtained using one of the two algorithms, and (3) there is a convex program capturing the set of equilibria attainable by either approach.

We obtain an FPTAS to compute an approximate equilibrium. Beyond money-clearing markets, however, even deciding existence is not well-understood – is it NP-hard, or can it be tightly characterized by a simple condition that can be checked in polynomial time? In addition to deciding existence, can we efficiently *find* exact equilibria in general (if they exist) or in money-clearing markets? Is the problem of finding equilibria for money-clearing markets complete for CLS?

We showed that an approximate equilibrium can be rounded to obtain a constant-factor approximation of Nash social welfare for agents with capped linear (i.e., budget-additive) valuations. This shows that in order to obtain a constant-factor approximation algorithm for the Nash social welfare problem, we only need a constant-factor approximate equilibrium of an appropriate Fisher market. It would be interesting to see if equilibria in Fisher markets with limits can yield good approximation algorithms for the Nash social welfare problem for more general classes of valuations.

Appendix A: The Linear Complementarity Problem and Lemke's Algorithm Given an $n \times n$ matrix \mathbf{M} and a vector \mathbf{q} , the linear complementarity problem⁵ asks for a vector \mathbf{y} satisfying the following conditions:

$$\mathbf{M}\mathbf{y} \leq \mathbf{q}, \quad \mathbf{y} \geq 0 \quad \text{and} \quad \mathbf{y} \cdot (\mathbf{q} - \mathbf{M}\mathbf{y}) = 0. \quad (20)$$

The problem is interesting only when $\mathbf{q} \not\geq 0$, since otherwise $\mathbf{y} = 0$ is a trivial solution. Let us introduce slack variables \mathbf{v} to obtain the equivalent formulation

$$\mathbf{M}\mathbf{y} + \mathbf{v} = \mathbf{q}, \quad \mathbf{y} \geq 0, \quad \mathbf{v} \geq 0 \quad \text{and} \quad \mathbf{y} \cdot \mathbf{v} = 0. \quad (21)$$

The reason for imposing non-negativity on the slack variables is that the first condition in (20) implies $\mathbf{q} - \mathbf{M}\mathbf{y} \geq 0$. Let \mathcal{P} be the polyhedron in $2n$ -dimensional space defined by the first three conditions; we will assume that \mathcal{P} is non-degenerate⁶. There are standard ways to handle degeneracy in Lemke's scheme, namely *lexico-minimum ratio test* (see Section 4.3 of [19] and also [16]) to ensure termination in a finite number of steps. Under the non-degeneracy condition, any solution to (21) will be a vertex of \mathcal{P} , since it must satisfy $2n$ equalities. Note that the set of solutions may be disconnected.

An ingenious idea of Lemke was to introduce a new variable and consider the following system, which is called the *augmented LCP*:

$$\mathbf{M}\mathbf{y} + \mathbf{v} - z\mathbf{1} = \mathbf{q}, \quad \mathbf{y} \geq 0, \quad \mathbf{v} \geq 0, \quad z \geq 0 \quad \text{and} \quad \mathbf{y} \cdot \mathbf{v} = 0. \quad (22)$$

Let \mathcal{P}' be the polyhedron in $(2n + 1)$ -dimensional space defined by the first four conditions of the augmented LCP; again we will assume that \mathcal{P}' is non-degenerate. Since any solution to (22) must still satisfy $2n$ equalities, the set of solutions, say S , will be a subset of the one-skeleton of \mathcal{P}' , i.e., it will consist of edges and vertices of \mathcal{P}' . Any solution to the original system must satisfy the additional condition $z = 0$ and hence will be a vertex of \mathcal{P}' .

Now S turns out to have some nice properties. Any point of S is *fully labeled* in the sense that for each i , $y_i = 0$ or $v_i = 0$.⁷ We will say that a point of S has *double label* i if $y_i = 0$ and $v_i = 0$ are both satisfied at this point. Clearly, such a point will be a vertex of \mathcal{P}' , and it will have only one double label. Since there are exactly two ways of relaxing this double label, this vertex must have exactly two edges of S incident on it. Clearly, a solution to the original system (i.e., satisfying $z = 0$) will be a vertex of \mathcal{P}' that does not have a double label. On relaxing $z = 0$, we get the unique edge of S incident on this vertex.

As a result of these observations, it follows that S consists of paths and cycles. Of these paths, Lemke's algorithm explores a special one. A *ray* is an unbounded edge of S such that the vertex of \mathcal{P}' it is incident on has $z > 0$. Among the rays, one is special – the one on which $\mathbf{y} = 0$; the points on this ray have $\mathbf{v} = \mathbf{q} + z\mathbf{1}$ and $z \geq -\min_i q_i$. This is called the *primary ray*, and the rest are called *secondary rays*. Now Lemke's algorithm explores, via pivoting, the path starting with the primary ray. This path must end either in a vertex satisfying $z = 0$, i.e., a solution to the original system, or a secondary ray. In the latter case, the algorithm is unsuccessful in finding a solution to the original system; in particular, the original system may not have a solution.

⁵ We refer the reader to [19] for a comprehensive treatment of notions presented in this section.

⁶ A polyhedron in n -dimension is said to be *non-degenerate* if on its d -dimensional faces exactly $n - d$ of its constraints hold with equality. For example on vertices (0-dimensional face) exactly n constraints hold with equality. There are many other equivalent ways to describe this notion.

⁷ These are also known as *almost complementary solutions* in the literature.

REMARK A.1. The set-up for the primary ray can be somewhat streamlined. First, the variable z is only needed in rows in which \mathbf{q} is negative. For the other rows, one fixes the value of \mathbf{v} to the value of \mathbf{q} . In our application, we can use this streamlining for equations (10) and (14).

Second, assume that the constraint for y_n is of the form $y_n \geq E \perp y_n \geq 0$, where the expression E is a function of variables y_1 to y_{n-1} . The constraint expresses $y_n = \max(0, E)$. With a slack variable v_n , the constraint becomes $-y_n + v_n = -E$, and if $E > 0$ for $y_1 = y_2 = \dots = y_{n-1} = 0$, one would have to introduce a z into the equation. Instead, one can set y_n to $\max(0, E)$ and fix v_n to $y_n - E$ on the ray. In our application, we can use this streamlining for equations (11) and (13). Only, for equation (12) we introduce the z -variable and obtain (16).

Third, one can also do without introducing the vector \mathbf{v} of slack variables. Without them one moves in \mathbb{R}^{n+1} in the 1-skeleton of the polyhedron defined by the inequalities $\mathbf{M}\mathbf{y} - z\mathbf{1} \leq \mathbf{q}$, $\mathbf{y} \geq 0$, $z \geq 0$. On each edge or ray, n of the inequalities are satisfied with equality, and on a vertex, $n+1$ are. The streamlining described in the second paragraph still applies.

As mentioned above, if \mathbf{q} has no negative components, (20) has the trivial solution $\mathbf{y} = 0$. In this case Lemke's algorithm cannot be used for finding a non-trivial solution, since it is simply not applicable. However, the Lemke-Howson scheme is applicable in this case; it follows a complementary path in the original polyhedron (21) starting at $\mathbf{y} = 0$ and guarantees termination at a non-trivial solution if the polyhedron is bounded.

Appendix B: Rounding Trees with Zero Price Goods In this section, we give an algorithm to round trees $T_0 \subseteq B_0 \times G_0$ of the equilibrium (\mathbf{x}, \mathbf{p}) to an integral allocation. Recall that in such trees, all goods have price $p_j = 0$ and all buyers reach their caps c_i 's. Consider the following procedure which is similar to the procedure in Section 4.1.2. It uses only the allocation \mathbf{x} and does not rely on prices. In particular, the only price-based assignment rule is in the preprocessing step, and it can be replaced here with an equivalent, more direct criterion:

Preprocessing: It consists of three substeps.

- (a) For each zero-price tree component of the allocation graph, assign some agent to be a root node.
- (b) For every good j keep at most one child agent. This child agent i must receive the largest amount of j among the child agents (ties are broken arbitrarily) and must receive a utility that is more than half of his total utility, i.e., $u_{ij}x_{ij} > c_i/2$. In other words, child agent i is cut off from good j if a sibling buys more of good j (ties are broken arbitrarily) or if $u_{ij}x_{ij} \leq c_i/2$.
- (c) Agents whose connection to their parent goods is severed in step (b) become roots.

Rounding: It consists of two substeps.

- (a) Goods with no child agent are assigned to their parent agent.
- (b) For each non-trivial zero-price tree component, do the following recursively: Assign the root agent a child good j that gives him the maximum value (among all children goods) in the fractional solution. Except in the subtree rooted at j , assign each good to its child agent in the remaining tree. Make the child agent of good j the root node of the newly created tree.

Let us define the remaining valuation of an agent after preprocessing as the sum of the fractional values for the goods at the ends of the remaining edges incident on the agent. The following lemmas follow by construction.

LEMMA B.1. *After preprocessing, the remaining valuation of each root agent r is at least $c_r/2$. For all other agents i the valuation is at least c_i .*

LEMMA B.2. *After step (a) of the rounding, each tree component T has $k_T + 1$ agents and k_T goods for some $k_T \geq 0$.*

LEMMA B.3. *After the rounding, each agent i that is assigned his parent good obtains a valuation of at least $c_i/2$.*

Next, consider a zero-price tree T at the beginning of step (b) of the rounding with $k_T + 1$ agents and k_T goods. Assume $k_T \geq 1$ first. Let $a_1, g_1, a_2, g_2, \dots, a_\ell, g_\ell, a_{\ell+1}$ be the *recursion path* in T starting from the root agent a_1 and ending at the leaf agent $a_{\ell+1}$ where $a_1, \dots, a_{\ell+1}$ became root agents of the trees formed recursively during the rounding step, and good g_i is assigned to a_i in this process, for $1 \leq i \leq \ell$. Note that $a_{\ell+1}$ is not assigned any good in step (b) of the rounding. However, we can deduce from Lemma B.1 that $a_{\ell+1}$ must have been assigned some good during step (a) of the rounding. We denote by k_i the number of children for agent a_i , for $1 \leq i \leq \ell$. If $k_T = 0$, then $\ell = 0$ and $a_1 = a_{\ell+1}$ is the root of a tree containing no goods after step (a) of the rounding.

LEMMA B.4. *The product of the valuations of agents in T in the rounded solution is at least*

$$\left(\frac{1}{2}\right)^{k_T-\ell+1} \cdot \frac{1}{k_1 \dots k_\ell} \cdot \prod_{i \in T} c_i .$$

Proof. Let $q_i = x_{a_i, g_i} > 0$ denote the amount of good g_i bought by agent a_i in the equilibrium, for $1 \leq i \leq \ell$. Since the root agent a_1 receives at least half of the valuation from his children (including the ones before any edge removals), we have $u_{a_1 g_1} q_1 \geq c_{a_1} / (2k_1)$. Further, since $u_{ij} \leq c_i$ for all (i, j) , each agent a_i , $2 \leq i \leq \ell + 1$, receives value of at most $u_{a_i g_{i-1}} (1 - q_{i-1}) \leq c_{a_i} (1 - q_{i-1})$ from his parent-good, and thus value of at least $q_{i-1} c_{a_i}$ from children in the equilibrium. This implies that $u_{a_i, g_i} q_i \geq q_{i-1} c_{a_i} / k_i$ for $2 \leq i \leq \ell$.

The product of the valuations of a_1 to $a_{\ell+1}$ in the rounded solution is at least $\prod_{i=1}^{\ell} u_{a_i g_i} \cdot q_\ell c_{a_{\ell+1}}$. This holds since g_i is assigned to a_i , for $1 \leq i \leq \ell$, and $a_{\ell+1}$ receives a value of at least $q_\ell c_{a_{\ell+1}}$ from his children in the equilibrium. Since these children are assigned to $a_{\ell+1}$ during step (a) of the rounding, he receives at least this value in the rounded solution.

Combining the arguments above we obtain

$$\begin{aligned} \prod_{i=1}^{\ell} u_{a_i g_i} \cdot q_\ell c_{a_{\ell+1}} &\geq \frac{c_{a_1}}{2q_1 k_1} \frac{q_1 c_{a_2}}{q_2 k_2} \dots \frac{q_{\ell-1} c_{a_\ell}}{q_\ell k_\ell} \cdot q_\ell c_{a_{\ell+1}} \\ &= \frac{1}{2} \left(\frac{1}{k_1 \dots k_\ell} \right) \cdot \prod_{1 \leq i \leq \ell+1} c_{a_i} . \end{aligned}$$

Since each of the remaining $k_T - \ell$ agents i in T gets a value of at least $c_i / 2$, the product of the valuations of agents in T in the rounded solution is at least

$$\left(\frac{1}{2}\right)^{k_T-\ell+1} \left(\frac{1}{k_1 \dots k_\ell}\right) \prod_{i \in T} c_i .$$

□

Acknowledgments. Jugal Garg was supported by NSF grant CCF-1942321. Martin Hoefer was supported by DFG grants Ho 3831/5-1, 6-1, and 7-1.

The authors are highly grateful to the anonymous referee for their detailed reports suggesting numerous simplified constructions at many places that have helped improve the presentation of the paper immensely.

References

[1] Allouch N, Van CL (2007) Walras and dividends equilibrium with possibly satiated consumers. *J. Math. Econom.* 44(9):907–918.

- [2] Anari N, Gharan SO, Saberi A, Singh M (2017) Nash social welfare, matrix permanent, and stable polynomials. *Proc. 8th Symp. Innovations in Theoret. Comput. Sci. (ITCS)*.
- [3] Anari N, Mai T, Gharan SO, Vazirani V (2018) Nash social welfare for indivisible items under separable, piecewise-linear concave utilities. *Proc. 29th Symp. Discrete Algorithms (SODA)*, 2274–2290.
- [4] Andelman N, Mansour Y (2004) Auctions with budget constraints. *Proc. 9th Scandinavian Workshop Algorithm Theory (SWAT)*, 26–38.
- [5] Aumann R, Dreze J (1986) Values of markets with satiation or fixed prices. *Econometrica* 54(6):1271–1318.
- [6] Azar Y, Birnbaum B, Karlin A, Mathieu C, Nguyen CT (2008) Improved approximation algorithms for budgeted allocations. *Proc. 35th Intl. Coll. Automata, Languages & Programming (ICALP)*, volume 1, 186–197.
- [7] Barman S, Krishnamurthy SK, Vaish R (2018) Finding fair and efficient allocations. *Proc. 19th Conf. Econom. Comput. (EC)*, 557–574.
- [8] Bei X, Garg J, Hoefer M, Mehlhorn K (2019) Earning and utility limits in Fisher markets. *ACM Trans. Econom. Comput.* 7(2):10:1–10:35.
- [9] Brainard W, Scarf H (2000) How to compute equilibrium prices in 1891. *Cowles Foundation Discussion Paper* 1270.
- [10] Buchbinder N, Jain K, Naor J (2007) Online primal-dual algorithms for maximizing ad-auctions revenue. *Proc. 15th European Symp. Algorithms (ESA)*, 253–264.
- [11] Buchfuhrer D, Dughmi S, Fu H, Kleinberg R, Mossel E, Papadimitriou C, Schapira M, Singer Y, Umans C (2010) Inapproximability for VCG-based combinatorial auctions. *Proc. 21st Symp. Discrete Algorithms (SODA)*, 518–536.
- [12] Caragiannis I, Kurokawa D, Moulin H, Procaccia AD, Shah N, Wang J (2019) The unreasonable fairness of maximum Nash welfare. *ACM Trans. Econom. Comput.* 7(3):12:1–12:32.
- [13] Chakrabarty D, Goel G (2010) On the approximability of budgeted allocations and improved lower bounds for submodular welfare maximization and GAP. *SIAM J. Comput.* 39(6):2189–2211.
- [14] Chaudhury BR, Cheung YK, Garg J, Garg N, Hoefer M, Mehlhorn K (2022) Fair division of indivisible goods for a class of concave valuations. *J. Artif. Intell. Res.* 74:111–142.
- [15] Chlebík M, Chlebíková J (2003) Approximation hardness for small occurrence instances of NP-hard problems. *Proc. 5th Intl. Conf. Algorithms and Complexity (CIAC)*, 152–164, full version ECCC-TR02-073, 2002.
- [16] Chvátal V (1983) *Linear Programming* (W.H. Freeman and Company).
- [17] Cole R, Devanur N, Gkatzelis V, Jain K, Mai T, Vazirani V, Yazdanbod S (2017) Convex program duality, Fisher markets, and Nash social welfare. *Proc. 18th Conf. Econom. Comput. (EC)*, 459–460.
- [18] Cole R, Gkatzelis V (2018) Approximating the Nash social welfare with indivisible items. *SIAM J. Comput.* 47(3):1211–1236.
- [19] Cottle R, Pang JS, Stone R (1992) *The Linear Complementarity Problem* (Boston: Academic Press).
- [20] Darmann A, Schauer J (2015) Maximizing Nash product social welfare in allocating indivisible goods. *Europ. J. Oper. Res.* 247(2):548–559.
- [21] Devanur N, Garg J, Végh L (2016) A rational convex program for linear Arrow-Debreu markets. *ACM Trans. Econom. Comput.* 5(1):6:1–6:13.
- [22] Devanur N, Jain K, Sivan B, Wilkens C (2019) Near optimal online algorithms and fast approximation algorithms for resource allocation problems. *J. ACM* 66(1):7:1–7:41.
- [23] Devanur N, Papadimitriou C, Saberi A, Vazirani V (2008) Market equilibrium via a primal-dual algorithm for a convex program. *J. ACM* 55(5):22:1–22:18.
- [24] Duan R, Garg J, Mehlhorn K (2016) An improved combinatorial polynomial algorithm for the linear Arrow-Debreu market. *Proc. 27th Symp. Discrete Algorithms (SODA)*, 90–106.

- [25] Duan R, Mehlhorn K (2015) A combinatorial polynomial algorithm for the linear Arrow-Debreu market. *Inf. Comput.* 243:112–132.
- [26] Eisenberg E, Gale D (1959) Consensus of subjective probabilities: The Pari-Mutuel method. *Ann. Math. Stat.* 30(1):165–168.
- [27] Fearnley J, Goldberg P, Hollender A, Savani R (2021) The complexity of gradient descent: CLS = PPAD \cap PLS. *Proc. 53rd Symp. Theory of Comput. (STOC)*, 46–59.
- [28] Feldman M, Gravin N, Lucier B (2016) Combinatorial Walrasian equilibrium. *SIAM J. Comput.* 45(1):29–48.
- [29] Freeman R, Zahedi S, Conitzer V (2017) Fair and efficient social choice in dynamic settings. *Proc. 26th Intl. Joint Conf. Artif. Intell. (IJCAI)*, 4580–4587.
- [30] Gale D (1960) *Theory of Linear Economic Models* (McGraw Hill).
- [31] Garg J, Hoefer M, Mehlhorn K (2018) Approximating the Nash social welfare with budget-additive valuations. *Proc. 29th Symp. Discrete Algorithms (SODA)*, 2326–2340.
- [32] Garg J, Mehta R, Sohoni M, Vazirani V (2015) A complementary pivot algorithm for market equilibrium under separable, piecewise-linear concave utilities. *SIAM J. Comput.* 44(6):1820–1847.
- [33] Garg J, Mehta R, Vazirani V (2014) Dichotomies in equilibrium computation, and complementary pivot algorithms for a new class of non-separable utility functions. *Proc. 46th Symp. Theory of Comput. (STOC)*, 525–534.
- [34] Garg J, Mehta R, Vazirani V (2018) Substitution with satiation: A new class of utility functions and a complementary pivot algorithm. *Math. Oper. Res.* 43(3):996–1024.
- [35] Garg J, Végh L (2019) A strongly polynomial algorithm for linear exchange markets. *Proc. 51st Symp. Theory of Comput. (STOC)*, 54–65.
- [36] Jain K (2007) A polynomial time algorithm for computing the Arrow-Debreu market equilibrium for linear utilities. *SIAM J. Comput.* 37(1):306–318.
- [37] Kajii A (1996) How to discard non-satiation and free-disposal with paper money. *J. Math. Econom.* 25:75–84.
- [38] Kalaitzis C (2016) An improved approximation guarantee for the maximum budgeted allocation problem. *Proc. 27th Symp. Discrete Algorithms (SODA)*, 1048–1066.
- [39] Kaneko M, Nakamura K (1979) The Nash social welfare function. *Econometrica* 47(2):423–436.
- [40] Kelly F (1997) Charging and rate control for elastic traffic. *European Transactions on Telecommunications* 8:33–37.
- [41] Konovalov A (2005) The core of an economy with satiation. *Economic Theory* 25:711–719.
- [42] Lee E (2017) APX-hardness of maximizing Nash social welfare with indivisible items. *Inf. Process. Lett.* 122:17–20.
- [43] Mas-Colell A (1992) Equilibrium theory with possibly satiated preferences. Majumdar M, ed., *Equilibrium and Dynamics*, chapter 9 (Palgrave Macmillan, London).
- [44] Mehta A (2012) Online matching and ad allocation. *Foundations and Trends in Theoretical Computer Science* 8(4):265–368.
- [45] Mehta A, Saberi A, Vazirani U, Vazirani V (2007) Adwords and generalized online matching. *J. ACM* 54(5).
- [46] Moulin H (2003) *Fair Division and Collective Welfare* (MIT Press).
- [47] Nash J (1950) The bargaining problem. *Econometrica* 18(2):155–162.
- [48] Nguyen N, Nguyen TT, Roos M, Rothe J (2014) Computational complexity and approximability of social welfare optimization in multiagent resource allocation. *Autonom. Agents & Multi-Agent Syst.* 28(2):256–289.
- [49] Orlin J (2010) Improved algorithms for computing Fisher's market clearing prices. *Proc. 42nd Symp. Theory of Comput. (STOC)*, 291–300.

- [50] Ramezani S, Endriss U (2010) Nash social welfare in multiagent resource allocation. *Agent-Mediated Electronic Commerce. Designing Trading Strategies and Mechanisms for Electronic Markets*, 117–131 (Springer Verlag).
- [51] Roughgarden T, Talgam-Cohen I (2015) Why prices need algorithms. *Proc. 16th Conf. Econom. Comput. (EC)*, 19–36.
- [52] Shmyrev V (2009) An algorithm for finding equilibrium in the linear exchange model with fixed budgets. *J. Appl. Indust. Math.* 3(4):505–518.
- [53] Srinivasan A (2008) Budgeted allocations in the full-information setting. *Proc. 11th Workshop Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*, 247–253.
- [54] Todd M (1976) Orientation in complementary pivot algorithms. *Math. Oper. Res.* 1(1):54–66.
- [55] Végh L (2014) Concave generalized flows with applications to market equilibria. *Math. Oper. Res.* 39(2):573–596.
- [56] Walras L (1874) *Éléments d'économie politique pure, ou théorie de la richesse sociale (Elements of Pure Economics, or the theory of social wealth)* (Lausanne, Paris), (1899, 4th ed.; 1926, rev ed., 1954, Engl. transl.).
- [57] Ye Y (2007) Exchange market equilibria with Leontief's utility: Freedom of pricing leads to rationality. *Theoret. Comput. Sci.* 378(2):134–142.