Computing Equilibria in Markets with Budget-Additive Utilities

Xiaohui Bei[1], Jugal Garg[2], Martin Hoefer[3], and Kurt Mehlhorn[4]

1 **Nanyang Technological University, Singapore**
xhbei@ntu.edu.sg
2 **University of Illinois at Urbana-Champaign, USA**
jugal@illinois.edu
3 **MPI für Informatik and Saarland University, Saarbrücken, Germany**
mhoefer@mpi-inf.mpg.de
4 **MPI für Informatik, Saarbrücken, Germany**
mehlhorn@mpi-inf.mpg.de

Abstract

We present the first analysis of Fisher markets with buyers that have budget-additive utility functions. Budget-additive utilities are elementary concave functions with numerous applications in online adword markets and revenue optimization problems. They extend the standard case of linear utilities and have been studied in a variety of other market models. In contrast to the frequently studied CES utilities, they have a global satiation point which can imply multiple market equilibria with quite different characteristics. Our main result is an efficient combinatorial algorithm to compute a market equilibrium with a Pareto-optimal allocation of goods. It relies on a new descending-price approach and, as a special case, also implies a novel combinatorial algorithm for computing a market equilibrium in linear Fisher markets. We complement this positive result with a number of hardness results for related computational questions. We prove that it is NP-hard to compute a market equilibrium that maximizes social welfare, and it is PPAD-hard to find any market equilibrium with utility functions with separate satiation points for each buyer and each good.

1998 ACM Subject Classification F.2.2 Nonnumerical Algorithms and Problems

Keywords and phrases Budget-Additive Utility, Market Equilibrium, Equilibrium Computation

Digital Object Identifier 10.4230/LIPIcs.ESA.2016.8

1 Introduction

The concept of market equilibrium is a fundamental and well-established notion in economics to analyze and predict the outcomes of strategic interaction in large markets. Initiated by Walras in 1874, the study of market equilibrium has become a cornerstone of microeconomic analysis, mostly due to general results that established existence under very mild conditions [2]. Since efficient computation is a fundamental criterion to evaluate the plausibility of equilibrium concepts, the algorithmic aspects of market equilibrium are one of the central domains in algorithmic game theory. Over the last decade, several new algorithmic approaches to compute market equilibria were discovered. Efficient algorithms based on convex programming techniques can compute equilibria in a large variety of domains [12, 22, 25]. More importantly, several approaches were proposed that avoid the use of heavy algorithmic machinery and follow combinatorial strategies [17, 26, 29, 32, 20, 19], or even work as a tâtonnement process in unknown market environments [13, 10, 4]. Designing such combinatorial algorithms

24rd Annual European Symposium on Algorithms (ESA 2016).
Editors: Piotr Sankowski and Christos Zaroliagis; Article No. 8; pp. 8:1–8:14
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

is useful also beyond the study of markets, since the underlying ideas can be applied in other areas. Variants of these algorithms were shown to solve scheduling [23, 24] and cloud computing problems [15], or can be used for fair allocation of indivisible items [14].

In this paper, we design a new combinatorial polynomial time algorithm for computing equilibria in Fisher markets with budget-additive utilities. In a Fisher market, there is a single seller with a set $G = \{1, \dots, m\}$ of goods. W.l.o.g. we assume that the total quantity of each good is 1. There is a set $B = \{1, \dots, n\}$ of buyers. Each buyer i has a budget $M_i > 0$ of money and a utility function u_i. For budget-additive utilities, $u_{ij} \geq 0$ is the utility of buyer i if one unit of good j is allocated to her. There is a *happiness cap* $c_i > 0$, and the utility function is

$$u_i(\mathbf{x}_i) = \min \left\{ c_i, \sum_{j \in G} u_{ij} x_{ij} \right\} \quad ,$$

where $\mathbf{x}_i = (x_{ij})_{j \in G}$ is any bundle of goods assigned to buyer i. If $u_i(\mathbf{x}_i) = c_i$, then buyer i is called *capped buyer* for allocation $\mathbf{x}$. We assume all u_{ij}, c_i, M_i are rational numbers.

Our goal is to compute an *allocation* $\mathbf{x} = (\mathbf{x}_i)_{i \in B}$ of goods and *prices* $\mathbf{p} = (p_j)_{j \in G}$ such that the pair $(\mathbf{x}, \mathbf{p})$ is a market equilibrium. Given prices $\mathbf{p}$, a *demand bundle* $\mathbf{x}_i^*$ of buyer i is a bundle of goods that maximizes the utility of buyer i for its budget, i.e., $\mathbf{x}_i^* \in \arg\max \left\{ u_i(\mathbf{x}_i) \mid \sum_j p_j x_{ij} \leq M_i \right\}$. Note that $\sum_j u_{ij} x_{ij}^* > c_i$ is allowed. A *market equilibrium* $(\mathbf{x}, \mathbf{p})$ is a pair such that

- $\mathbf{p} \geq 0$ (prices are nonnegative),
- $\sum_i x_{ij} \leq 1$ for every $j \in G$ (no overallocation),
- $\mathbf{x}_i$ is a demand bundle for every $i \in B$, and
- Walras' law holds: $p_j(1 - \sum_i x_{ij}) = 0$ for every $j \in G$.

Note that if $\sum_i x_{ij} < 1$, then $p_j = 0$. An equilibrium $(\mathbf{x}, \mathbf{p})$ is *Pareto-optimal* if there is no equilibrium $(\mathbf{x}', \mathbf{p}')$ such that $u_i(\mathbf{x}) \leq u_i(\mathbf{x}')$ for all i and $u_i(\mathbf{x}) < u_i(\mathbf{x}')$ for at least one i.

Budget-additive utility functions are a simple class of submodular and concave functions and a natural generalization of the standard and well-understood case of linear utilities. These utility functions arise naturally in cases where agents have an intrinsic upper bound on their utility. For example, if the goods are food and the utility of a food item for a particular buyer is its calorie content, calories above a certain threshold do not increase the utility of the buyer. In addition, there are a variety of further applications in adword auctions and revenue maximization problems [1, 3, 8, 6]. Recently, market models where agents have budget-additive utilities attracted a significant amount of research interest, e.g., for the allocation of indivisible goods in offline [1, 3, 8] and online [6, 27] scenarios, for truthful mechanism design [7], and for the study of Walrasian equilibrium with quasi-linear utilities [21, 18, 30]. As simple variants of submodular functions, they capture many of the inherent difficulties of more general domains. Given this amount of interest, it is perhaps surprising that they are not well-understood within the classic Fisher and exchange markets.

Results and Contribution. We study Fisher markets with budget-additive utilities. Our initial observations about these markets reveal that they have different properties than the ones with CES utilities usually studied in the literature. Due to the satiated nature of the utilities, capped buyers might not spend all their money or spend money on goods that do not give them maximum utility per unit of money, so prices and utilities in market equilibrium are not unique and can be quite different. It is possible to simply ignore the satiation and assume linear utilities. Then a variety of existing algorithms [17, 29, 20, 19, 4] can be used to compute a market equilibrium. It continues to be a market equilibrium for the market with budget-additive utilities. However, this equilibrium may be undesirable, as in many cases it does not even satisfy Pareto-optimality of the allocation.

▶ **Example 1.** Consider a *linear* market with two buyers and two goods, $u_{11} = 5$, $u_{21} = 2$, and $u_{12} = u_{22} = 1$. The budgets are $M_1 = 3$ and $M_2 = 1$. For the unique equilibrium we allocate good 1 completely to buyer 1 and good 2 completely to buyer 2, i.e., $x_{11} = x_{22} = 1$. The buyers' utilities amount to 5 and 1, resp., and the prices are $p_1 = 3$ and $p_2 = 1$.

Now suppose buyer 1 has a budget-additive utility function with cap $c_1 = 1$. Then $(\mathbf{x}, \mathbf{p})$ described above remains an equilibrium, since both buyers obtain a demand bundle (buyer 1 now has utility 1 instead of 5). Alternatively, suppose we allocate good 1 completely to buyer 2 and good 2 completely to buyer 1, i.e., $x_{12} = x_{21} = 1$. The utilities amount to 1 and 2, resp., and the prices can be chosen as $p_1 = 1$ and $p_2 \in [0.5, 3]$. Here buyer 1 buys a bundle of goods with optimal utility of 1. Buyer 2 buys a demand bundle since he spends all its budget on a good that gives him the maximum *bang-per-buck ratio*. All goods are exactly allocated, and Walras' law holds. Thus, it represents another market equilibrium. Note if $p_2 < 3$, buyer 1 does not spend all of its money, but it is still a demand bundle for because he achieves the maximum utility. Furthermore, such an equilibrium Pareto-dominates the one derived from the linear case in terms of utilities. ◀

We strive to compute a market equilibrium with a Pareto-optimal allocation and focus on a subset of market equilibria, in which we restrict the allocation to demand bundles which we call thrifty and modest – buyers spend the least amount of money that can achieve their optimal utilities and receive a bundle of goods that has a minimality property. In Section 2, we show that such *modest MBB equilibria* can be captured by a generalization of the classic Eisenberg-Gale convex program, their utilities are unique, and their allocation is always Pareto-optimal (w.r.t. all possible allocations, attainable in market equilibrium or not). We highlight that the set of modest MBB equilibria can be partially ordered with respect to their price vectors and forms a lattice. As such, there are modest MBB equilibria with pointwise largest and smallest prices, resp. Among all modest MBB equilibria they yield maximum and minimum revenue for the seller, resp.

Section 3 contains our main contribution – a combinatorial algorithm that computes price and allocation vectors of a modest MBB equilibrium in time $O(mn^6(\log(m+n) + (m+n)\log U))$, where n is the number of agents, m the number of goods, and U the largest integer in the market parameters. The computed equilibrium has a Pareto-optimal allocation, as well as pointwise largest prices and maximum revenue among all modest MBB equilibria.

Our algorithm represents a novel approach to compute market equilibria based on the idea of descending prices. While some parts of our algorithm are in spirit of combinatorial algorithms for linear markets [17, 20, 19, 4], all these approaches are ascending-price algorithms. This technique and its usual analysis based on the 1-norm of excess money does not apply in our case, since the norm is non-monotonic and cannot be used to measure progress towards equilibrium. Surprisingly, our novel descending-price approach overcomes the 1-norm issue, but we need to address additional challenges in establishing polynomial running time due to varying and non-increasing active budgets, and in showing that intermediate prices remain polynomially bounded. Note that, as a special case, this also yields a new combinatorial descending-price algorithm for linear Fisher markets.

In Section 4 we exploit the lattice structure of modest MBB equilibria and design a procedure, using which we can turn any modest MBB equilibrium into one with smallest prices and minimum revenue. In combination with the descending-price algorithm, it computes a modest MBB equilibrium with minimum revenue within the same asymptotic time bound.

Finally, we study two extensions in Section 5. Facing multiple equilibria, a natural goal is to compute an allocation that maximizes utilitarian social welfare. We prove that this problem is NP-hard, even when social welfare is measured by a k-norm of the vector of buyer utilities, for any constant $k > 0$. Moreover, we consider a variant of budget-additive utilities with a satiation point for *each buyer and each good.* They constitute a special class of separable piecewise-linear concave (SPLC) utilities, where each piecewise-linear component consists of two segments with the second one being constant. We show that even in this very special case computing any market equilibrium becomes PPAD-hard.

Related Work. The computation of market equilibria is a central area in algorithmic game theory. There are a variety of polynomial-time algorithms to compute approximate market equilibria based on solving different convex programming formulations [12, 22, 25]. Our paper is closer to work on markets with linear utilities and combinatorial algorithms that compute an exact equilibrium in polynomial time [17, 29, 20, 19]. Directly related to our approach is the classic combinatorial algorithm for linear Fisher markets [17]. In contrast, our algorithm is based on a new descending price approach where buyers are always saturated and goods have non-negative surplus. Further, the *active* budgets of buyers vary with the price change, which creates new challenges in establishing a polynomial bound on the number of iterations and the representation size of intermediate prices.

Independently of our work, Devanur et al [16] very recently presented the same convex program for Fisher markets with satiated buyers. They propose a polynomial-time algorithm for finding an arbitrary modest MBB equilibrium, but it is based on the ellipsoid method without any explicit running time bound.

Recently, algorithmic work has also started to address unknown markets, where utilities and budgets of buyers are unknown. Instead, algorithms iteratively set prices and query a demand oracle. In this domain, tâtonnement dynamics have been studied for Fisher markets and extensions with concave utilities. For many classes of these markets, a notion of $(1+\epsilon)$-approximate market equilibrium can be reached after a convergence time polynomial in $1/\epsilon$ and other market parameters [13, 10, 11, 5]. In some cases, the convergence time can even be reduced to $\log(1/\epsilon)$ [10]. A similar convergence rate is obtained by a more general algorithm even for general unknown exchange markets with weak gross-substitutes property, and even for linear markets with non-continuous demands and oracles using suitable tie-breaking [4].

Allocation of indivisible items to agents with budget-additive utilities is an active area of research interest. There are constant-factor approximation algorithms for optimizing the allocation in offline [1, 3, 8] and online [6, 27] scenarios. Closer to our work is the study of markets with money. The existence of Walrasian equilibrium with quasi-linear utilities and algorithmic issues of bundling items were studied in [21, 18, 30]. Strategic agents and truthful mechanisms for budget-additive markets have been analyzed in [7]. There are strong lower bounds for the approximation ratio of certain classes of truthful mechanisms, and a truthful mechanism with constant-factor approximation for budget-additive utilities is one of the most interesting open problems in combinatorial auctions.

2 Preliminaries

For a given price vector $\mathbf{p}$ and buyer i, we denote the *maximum bang-per-buck (MBB)* ratio by $\alpha_i = \max_j u_{ij}/p_j$, where we make the assumption that $0/0 = 0$. Budget-additive utilities strictly generalize linear utilities: when all c_i's are large enough, they are equivalent to linear utilities. If buyer i is uncapped in a market equilibrium $(\mathbf{x}, \mathbf{p})$, it behaves as in the linear case, spends all its budget, and buys only MBB goods ($x_{ij} > 0$ only if $u_{ij}/p_j = \alpha_i$). Otherwise, if buyer i is capped in $(\mathbf{x}, \mathbf{p})$, it might buy non-MBB goods and not spend all of its budget. This implies that unlike the case of linear utilities, market equilibrium prices and utilities are not unique with budget-additive utilities.

It is easy to see that we can obtain one market equilibrium by simply ignoring the happiness caps and treating the market as a linear one. However, this equilibrium is often undesirable since it is not always Pareto-optimal.

Our main goal in this paper is to find a market equilibrium that is Pareto-optimal. More generally, we will also be concerned with finding a (Pareto-optimal) market equilibrium that can maximize social welfare $\sum_{i \in B} u_i(\mathbf{x}_i)$. For the former we provide a polynomial-time algorithm, the latter we prove it to be NP-hard.

Modest MBB Equilibria, Pareto-Optimality, and Uniqueness. The main challenges in budget-additive markets arise from capped buyers, who may possibly have multiple choices for the demand bundle. Let us introduce two convenient restrictions on the allocation to capped buyers.

- An allocation $\mathbf{x}_i$ for buyer i is called *modest* if $\sum_j u_{ij}x_{ij} \le c_i$. By definition, for uncapped buyers every demand bundle is modest. For capped buyers, a modest bundle of goods $\mathbf{x}_i$ is such that utility breaks even between the linear part and c_i, i.e., $c_i = \sum_j u_{ij}x_{ij}$.
- A demand bundle $\mathbf{x}_i$ is called *thrifty* or *MBB* if it consists of only MBB goods: $x_{ij} > 0$ only if $u_{ij}/p_j = \alpha_i$. As noted above, for uncapped buyers every demand bundle is MBB.

We call a market equilibrium $(\mathbf{x}, \mathbf{p})$ a *modest MBB equilibrium* if $\mathbf{x}_i$ is modest and MBB for every buyer $i \in B$. We show an algorithm to compute in polynomial time such an equilibrium where $\mathbf{x}$ is also Pareto-optimal. Such an equilibrium is also desirable because it captures the behavioral assumption that each buyer is thrifty and spends the least amount of money in order to obtain a utility maximizing bundle of goods.

Consider the following Eisenberg-Gale program (1), which allows us to find a modest and Pareto-optimal allocation.

$$\begin{array}{rrcll} \text{Max.} & \displaystyle\sum_{i\in B} M_i \log \sum_{j\in G} u_{ij}x_{ij} & & & \\ \text{s.t.} & \displaystyle\sum_{j\in G} u_{ij}x_{ij} & \le & c_i & i \in B \\ & \displaystyle\sum_{i\in B} x_{ij} & \le & 1 & j \in G \\ & x_{ij} & \ge & 0 & i \in B,\ j \in G \end{array} \tag{1}$$

By standard arguments, we consider the dual for (1) using dual variables γ_i and p_j for the first two constraints, resp., and the KKT conditions read:

1. $p_j/u_{ij} \ge M_i/u_i - \gamma_i$

2. $x_{ij} > 0 \Rightarrow p_j/u_{ij} = M_i/u_i - \gamma_i$

3. $p_j \ge 0$ and $p_j > 0 \Rightarrow \sum_{i\in B} x_{ij} = 1$

4. $\gamma_i \ge 0$ and $\gamma_i > 0 \Rightarrow u_i = c_i$

Observe that the Lagrange multiplier γ_i indicates if the cap c_i represents a tight constraint in the optimum solution. The dual variables p_j can be interpreted as prices. Note that conditions 1 and 2 imply that $x_{ij} > 0$ if and only if $j \in \arg\min_{j'} p_{j'}/u_{ij'} = \arg\max_{j'} u_{ij'}/p_{j'}$, i.e., all agents purchase goods with maximum bang-per-buck. Hence, similarly as for linear markets [31], the KKT conditions imply that an optimal solution to the EG program (1) and corresponding dual prices constitute a market equilibrium, in which every agent buys goods that have maximum bang-per-buck. The KKT conditions postulate this also for agents whose utility reaches the cap. Thus, the optimal solution to this program is a modest MBB equilibrium. Furthermore, we obtain the following favorable analytical properties.

▶ **Proposition 2.** *The optimal solutions to* (1) *are exactly the modest MBB equilibria. The utility vector is unique across all such equilibria and each such equilibrium is Pareto-optimal. In particular, there is a unique set of capped buyers. Non-capped buyers spend all their money. Capped buyers do not overspend.*

While utilities are unique, allocation and prices of modest MBB equilibria might not be unique. Consider a market with two identical buyers and two goods, where $u_{11} = u_{12} = u_{21} = u_{22} = 1$, $c_1 = c_2 = 1$, and $M_1 = M_2 = 5$. The unique equilibrium utility of both buyers is $u_1 = u_2 = 1$, which can be obtained for any $p_1 = p_2 = p$, where $p \in [0, 5]$ and allocation $\mathbf{x}$ satisfying $x_{11} + x_{12} = 1$; $x_{21} + x_{22} = 1$; $x_{11} + x_{21} = 1$; $x_{12} + x_{22} = 1$.

▶ **Example 1** (continued)**.** For our example above, the modest MBB equilibrium obtained from solving the convex program is $x_{11} = 1/5$, $x_{12} = 0$, $x_{21} = 4/5$ and $x_{22} = 1$ with prices $p_1 = 10/13$ and $p_2 = 5/13$. Buyer 1 spends 2/13, buyer 2 spends the entire budget. The utilities are 1 and 13/5. It is easy to see that the KKT conditions hold. This equilibrium is Pareto-optimal and also the best equilibrium in terms of social welfare. ◀

Structure of Modest MBB Equilibria. Let us characterize the set of price vectors of modest MBB equilibria, which we denote by $\mathcal{P} = \{\mathbf{p} \mid (\mathbf{x}, \mathbf{p}) \text{ is modest MBB equilibrium }\}$. We consider the coordinate-wise comparison, i.e., $\mathbf{p} \geq \mathbf{p}'$ iff $p_j \geq p'_j$ for all $j \in G$.

▶ **Theorem 3.** *The pair* $(\mathcal{P}, \geq)$ *is a lattice.*

Given $\mathbf{p}$ and $\mathbf{p}'$, we partition the set of goods into three sets: $S_0 = \{j \mid p_j = p'_j\}$, $S_1 = \{j \mid p_j < p'_j\}$ and $S_2 = \{j \mid p_j > p'_j\}$. Let $\Gamma(T, \mathbf{p}) = \{i \mid x_{ij} > 0 \text{ for some } j \in T\}$ denote the set of buyers who are allocated a nonzero amount of any good in set T in the equilibrium $(\mathbf{x}, \mathbf{p})$. The proof of Theorem 3 exploits the following properties about the sets S_0, S_1 and S_2.

▶ **Lemma 4.** *Given any two modest MBB equilibria* $(\mathbf{x}, \mathbf{p})$ *and* $(\mathbf{x}', \mathbf{p}')$*, we have*

(i) $\Gamma(S_i, \mathbf{p}) = \Gamma(S_i, \mathbf{p}')$ *for* $i = 0, 1, 2$*, i.e., the set of buyers who buy the goods of* S_i *with respect to prices* $\mathbf{p}$ *and* $\mathbf{p}'$*are same.*

(ii) $\Gamma(S_0, \mathbf{p}), \Gamma(S_1, \mathbf{p})$ *and* $\Gamma(S_2, \mathbf{p})$ *are mutually disjoint.*

(iii) *All buyers in* $\Gamma(S_1, \mathbf{p})$ *and* $\Gamma(S_2, \mathbf{p})$ *are capped buyers in both* $(\mathbf{x}, \mathbf{p})$ *and* $(\mathbf{x}', \mathbf{p}')$*.*

Thus, there exists a modest MBB equilibrium with coordinate-wise highest (resp. lowest) prices. It yields the maximum (resp. minimum) revenue for the seller among all modest MBB equilibria.

▶ **Example 5.** Consider the following market with two buyers and two goods. Let $u_{11} = u_{12} = u_{22} = 1$ and $u_{21} = 0$. Let $M_1 = M_2 = 1$ and $c_1 = 1$. Then $x_{11} = x_{22} = 1$, $x_{12} = 0$, $p_1 = p_2 = 1$ is a modest MBB equilibrium with maximum revenue. A modest MBB equilibrium with minimum revenue has the same allocation and $p_1 = 0$ and $p_2 = 1$.

3 Computing a Modest MBB Equilibrium with Maximum Revenue

In this section, we describe an efficient algorithm to compute a modest MBB equilibrium. In fact, we compute the one with coordinate-wise highest prices and maximum revenue among all modest MBB equilibria. We define the *active budget* of buyer i at prices $\mathbf{p}$ as $M_i^a = \min\{M_i, c_i/\alpha_i\}$, where $\alpha_i = \max_{j\in G} u_{ij}/p_j$ is the MBB ratio. The active budget of buyer i is the minimum of M_i and the minimum amount of money needed to buy a bundle of goods with utility c_i. If $M_i^a = c_i/\alpha_i$, then buyer i is capped, otherwise uncapped.

3.1 Flow Network and Initialization

Given prices $\mathbf{p}$, let $A = \{(i,j) \subseteq B \times G \mid u_{ij}/p_j = \alpha_i\}$ be the set of equality edges, and the bipartite graph $(B \cup G, A)$ be the *equality graph*. We set up the following flow network N_p using the equality graph by adding a source s and sink t. It has nodes $\{s,t\} \cup B \cup G$ and edges (s,i) for $i \in B$, (j,t) for $j \in G$ and the equality edges. The edge (s,i) has capacity M_i^a, and the edge (j,t) has capacity p_j. The equality edges have infinite capacity. The flow in the network corresponds to money. We will maintain the following invariants throughout the algorithm.

Invariants:

- The edges out of s are saturated.
- Prices and active budgets never increase.
- Total utility of a buyer never decreases. Once a buyer is capped, it stays capped.

We initialize the prices to large values, namely $p_j = \sum_i M_i$. W.l.o.g. we will assume that all budgets, caps, and utilities are integers.

The surplus (residual capacity) of good j is $r_j = p_j - f_{jt}$, where f_{jt} is the flow from good j to t. Then $1 - f_{jt}/p_j$ is the fraction of good j that is not sold. We also keep track of the allocations x_{ij}. There might be prices equal to zero, and then the allocation cannot be computed from the money flow. Goods that have price zero have no surplus. There is no money flowing through them, although they may be (partially) allocated.

A subset T of buyers is called *tight* with respect to prices $\mathbf{p}$ if $\sum_{i\in T} M_i^a = \sum_{j\in\Gamma(T)} p_j$, where $\Gamma(T) \subseteq S$ is the set of goods connected to T in the equality graph.

A *balanced flow* is a maximum flow in N_p which minimizes the 2-norm of surplus vector r. Let $|r| = |r_1| + \ldots + |r_n|$ and $\|r\| = (r_1^2 + \ldots + r_n^2)^{1/2}$ be the ℓ_1 and ℓ_2 norm of r, resp.

3.2 The Algorithm

The complete algorithm is shown in Figure 1. We initialize price p_j of each good j to $\sum_i M_i$. This ensures that the invariants are satisfied, namely a maximum flow in network N_p saturates all edges out of s. We initialize every active budget $M_i^a = \min\{M_i, \min_j c_i p_j/u_{ij}\}$, and flow f and allocation x equal to zero.

The algorithm is divided into a set of phases, and each phase is further divided into a set of iterations. A phase starts with the computation of a balanced flow in N_p. Let the surplus of good j be $p_j - f_{jt}$. We pick a good j with maximum surplus, and we compute a set of goods S containing j and the goods which can reach j in the residual network corresponding to N_p without using nodes s and t. The surplus of each good in S is the same, and maximum among all goods. We denote by B' the set of buyers who have equality edges to goods in S, and by B'_c and B'_u the sets of capped and uncapped buyers in B', resp. Note that $x_{ij} = 0$ for all $i \in B'$ and $j \notin S$, since $x_{ij} > 0$ would imply $j \in S$.

We begin with an iteration, where we use a factor x to set the price of each good $j \in S$ to xp_j and the active budget of each buyer $i \in B'_c$ to xM_i^a. The prices and active budgets of the remaining goods and buyers remain unchanged. We decrease $x \leq 1$ continuously until

```
Input: A market with a set of buyers $B$ and a set of goods $G$;
        Budget $M_i$, happiness cap $c_i$, and utility parameters $u_{ij}$, $\forall i \in B, j \in G$;
Output: Equilibrium prices $\mathbf{p}$, allocation $x$;
$n \leftarrow |B|$;  $m \leftarrow |G|$;  $U \leftarrow \max_{i \in B, j \in G}\{M_i, c_i, u_{ij}\}$;  $\epsilon \leftarrow 1/((m+n)U^{4(m+n)})$;
Initialize price $p_j \leftarrow \sum_i M_i$ for each good $j$;
Initialize active budget $M_i^a \leftarrow \min\{M_i, \min_j c_i p_j / u_{ij}\}$ for each buyer $i$;
$f_{ij} \leftarrow 0$, $x_{ij} \leftarrow 0$, $\forall i \in B, j \in G$;
Repeat // phase
    $f \leftarrow$ balanced flow in $N_p$;  $x_{ij} \leftarrow f_{ij}/p_j$ if $p_j \neq 0$;  $r_j \leftarrow p_j - f_{jt}$;
    $\delta \leftarrow \max_j r_j$; Pick a good $j$ with surplus $\delta$;
    $S \leftarrow \{j\} \cup \{k \in G \mid k$ can reach $j$ in the residual network w.r.t. $f$ in $N_p \setminus \{s,t\}\}$;
    Repeat // iteration
        $B' \leftarrow$ Set of buyers who have incident equality edges to $S$;
        $B'_c \leftarrow$ Set of capped buyers in $B'$ (a buyer $i$ is capped if $M_i^a = \min_j {}^{c_i p_j}/_{u_{ij}}$);
        $B'_u \leftarrow B' \setminus B'_c$ (set of uncapped buyers);
        $x \leftarrow 1$; Set prices and active budgets as follows:
            $p_j \leftarrow x p_j$, $\forall j \in S$; $M_i^a \leftarrow x M_i^a$, $\forall i \in B'_c$;
        Decrease $x$ continuously down from 1 until one of the following events occurs
        Event 1: An uncapped buyer becomes capped
        Event 2: A new equality edge appears
            Recompute $N_p$;
            $f \leftarrow$ balanced flow in $N_p$;  $x_{ij} \leftarrow f_{ij}/p_j$ if $p_j \neq 0$;
            $S \leftarrow S \cup \{j \in G \mid j$ can reach $S$ in the residual network w.r.t. $f$ in $N_p \setminus \{s,t\}\}$;
        Event 3: A subset of $B'$ becomes tight // phase ends
    Until Event 3 occurs;
Until $|r| \leq \epsilon$;
Recompute $N_p$;
$f \leftarrow$ balanced flow in $N_p$;  $x_{ij} \leftarrow f_{ij}/p_j$ if $p_j \neq 0$;
```

Figure 1 The complete algorithm.

some structural change happens. Our goal here is to decrease prices as much as possible. By changing prices in this manner, all the equality edges between B' and S stay intact and the equality edges between B' and $G \setminus S$ become non-equality.

A possible structural change is that an uncapped buyer becomes capped. When a buyer $i \in B'$ is uncapped, $M_i < \min_j c_i p_j / u_{ij}$. Prices are decreasing, so this may become an equality. We term the first such change Event 1. Then we move buyer i from B'_u to B'_c.

Another possible change is that a new equality edge appears from a buyer in $B \setminus B'$ to a good in S. Prices of goods in S are decreasing, so goods in S are becoming attractive to buyers outside B'. Note that there cannot be a new equality edge from a buyer in B' to a good outside S. We term the first such change Event 2. Then we recompute the flow network N_p and a balanced flow in N_p. Next, we compute the set S' of goods $j \in G \setminus S$ that can reach a good in S in the residual graph corresponding to N_p without using the nodes s and t. Due to the property of balanced flows, the surplus of each good in S' is at least the surplus of some good in S. Finally, we add goods in S' to S.

Apart from the structural changes, we also maintain the invariants. The only invariant that can become violated with these changes is that the edges out of s are saturated. Hence, we need to stop when a subset T of B' becomes tight. Clearly, if prices are decreased further, then buyers in T will not be saturated, so we stop decreasing prices at this stage. We term this Event 3, and then the phase ends. We show in Lemma 10 below that during a phase, the 2-norm of the surplus vector decreases geometrically. The last phase ends when the total surplus becomes tiny. In fact, we will show that the surplus is actually zero at this point. We recompute a balanced flow and terminate.

When the prices of a set of goods hit zero in an iteration of the algorithm, then we do not change the allocation of these goods, and all the buyers interested in these goods must be capped. Since each buyer gets a modest allocation before the prices hit zero, the same allocation remains modest. None of the goods in the set is completely allocated. We delete these goods and the buyers to which they are allocated from consideration.

▶ **Example 1** (continued). Consider our algorithm applied to the example market above. We initialize p_1 and p_2 to $M_1 + M_2 = 4$. The active budgets become $M_1^a = \min_j c_1 p_j / u_{ij} = 4/5$ and $M_2^a = 1$. The edges $(1,1)$ and $(2,1)$ are equality edges and the balanced flow is $f_{11} = 4/5$, $f_{21} = 1$, and $f_{12} = f_{22} = 0$. The surpluses are $r_1 = 4 - 9/5 = 11/5$ and $r_2 = 4 - 0 = 4$. Thus $S = \{2\}$. We decrease p_2 to xp_2. At $x = 1/2$, the edge $(2,2)$ becomes an equality edge. Now $p_2 = 2$. The balanced flow does not change and hence $r_1 = 11/5$ and $r_2 = 2$. Thus $S = \{1\}$. We decrease M_1^a to $4x/5$ and p_1 to $4x$. At $x = 5/16$, B' becomes tight. We now have $M_1^a = 1/4$ and $p_1 = 5/4$. The balanced flow is $f_{11} = 1/4$ and $f_{21} = 1$. Thus $r_1 = 0$ and $r_2 = 2$. So $S = \{2\}$. We change p_2 to $p_2 x$. At $x = 5/16$, the edge $(2,2)$ becomes an equality edge. Now $p_2 = 5/8$. The balanced flow is $f_{11} = 1/4$, $f_{21} = 11/16$, and $f_{22} = 5/16$. Then $r_1 = r_2 = 5/16$. Thus $S = \{1,2\}$. We now decrease M_1^a to $x \cdot 1/4$, p_1 to $5x/4$ and p_2 to $5x/8$. At $x = 8/13$, B' becomes tight and we have $p_1 = 10/13$, $p_2 = 5/13$, $M_1^a = 2/13$, $x_{11} = 1/5$, $x_{21} = 4/5$, $x_{22} = 1$, $f_{11} = 2/13$, $f_{21} = 8/13$, and $f_{22} = 5/13$. ◀

3.3 Analysis

▶ **Lemma 6.** *The invariants hold during the run of the algorithm.*

Phases consist of iterations, which end with Event 1, 2, or 3. A phase ends with Event 3.

▶ **Lemma 7.** *Each phase has at most $2n$ iterations.*

Our next goal is to show that the 2-norm of the surplus vector decreases substantially during a phase. Let r and r' be the surplus vectors at the beginning and at the end of a phase, resp. For the purpose of our analysis we also maintain an intermediate flow f continuously as we change prices in each iteration; this flow is *not* maintained by the algorithm. When we recompute a balanced flow during Event 2, then f will be reset to the balanced flow. It is defined as $\forall i \in B'_c : f_{ij} \leftarrow x f_{ij}$ and $\forall i \in B'_u : f_{ij} \leftarrow f_{ij}$. f ensures that all buyers are saturated. If the surplus of a good j becomes zero corresponding to f, then we keep its surplus equal to zero and reroute extra flow from j to some other good with positive surplus, using a path in the residual network corresponding to f. If there is no such path, then this implies Event 3 has occurred, in which case the current phase is done. Consider an intermediate iteration t. With respect to f, let $r^t = (r_1^t, \dots, r_m^t)$ be the surplus vector at the beginning of iteration t, and let $\tilde{r}^t = (\tilde{r}_1^t, \dots, \tilde{r}_m^t)$ be the surplus vector before we recompute a balanced flow in iteration t if Event 2 occurs.

▶ **Lemma 8.** *$\tilde{r}_j^t \le r_j^t$, $\forall j \in G$, and $\|r^{t+1}\| \le \|\tilde{r}^t\| \le \|r^t\|$.*

▶ **Lemma 9.** *[17] Suppose f and f^* are a feasible and a balanced flow in N_p, resp., and r and r^* are the surplus vectors w.r.t. f and f^*, resp. If $r_j^* = r_j - \delta$ for some good j and $\delta > 0$, then $\|r^*\|^2 \le \|r\|^2 - \delta^2$.*

▶ **Lemma 10.** $\|r'\|^2 \leq (1 - 1/4mn)\|r\|^2$.

Proof. Consider the value of $\gamma = \min\{r_j \mid j \in S\}$ during a phase. When the phase begins, $\gamma = \delta$, and when it ends $\gamma = 0$. Recall that S only grows, and when we add a new good k to S, then the surplus of k is at least the surplus of some good already in S. This implies that γ does not change when we add new goods to S.

Let $t_1, \ldots, t_l$ be the iterations where γ decreases, and let $\delta_i > 0$ be the amount of decrease in iteration t_i. Further we break each δ_i into two parts δ_{i1} and δ_{i2} such that $\delta_i = \delta_{i1} + \delta_{i2}$. Here δ_{i1} is the amount of decrease due to the flow change before we recompute balanced flow, and δ_{i2} is the amount of decrease due to recomputation of balanced flow. Next consider only positive δ_{i1}'s and δ_{i2}'s. Clearly, $l \leq 2n$ and $\sum_{i:\delta_{i1}>0,\delta_{i2}>0}(\delta_{i1} + \delta_{i2}) \geq \delta$. Using Lemmas 8 and 9, we have $\|r'\|^2 \leq \|r\|^2 - (\delta_{11}^2 + \delta_{12}^2 + \cdots + \delta_{l1}^2 + \delta_{l2}^2) \leq \|r\|^2 - \delta^2/4n$. Since $\|r\|^2 \leq m\delta^2$, we have $\|r'\|^2 \leq (1 - 1/4mn)\|r\|^2$. ◀

Polynomial Running Time. In each iteration, the prices of goods in S are multiplied by a value that itself depends on the prices. It is not obvious why the size of the numbers in the computation is polynomially bounded. Here we show that, indeed, the sizes of all intermediate prices and flows in our algorithm remain polynomially bounded.

▶ **Lemma 11.** *All goods in S are connected by equality edges at all times. There is no flow from buyers in B' to goods outside S.*

Cap-events occur only at a cap-event prices. A cap-event price is any price p with $M_i = c_i p/u_{ij}$ for some i and j. Let $P_c = \{M_i u_{ij}/c_i \mid 1 \leq i \leq n, 1 \leq j \leq m\}$.

Let A' be any subset of the edge set with positive utility such that the graph formed by it is connected. Let B' and G' be the buyers and goods in this connected graph. The prices in the component have only one degree of freedom, i.e., we can select one of the prices, say p, as a base price and express any other price in the component as αp, where α is a rational whose numerator and denominator are products of at most m utilities. Consider an arbitrary partition of B' into capped buyers B'_c and uncapped buyers B'_u; B'_u must be nonempty. The budget of a capped buyer is of the form $c\alpha p$, where c is a cap and α is as above. If there are no surpluses, p must satisfy that (budget of capped buyers + budget of uncapped buyers) equals sum of the prices (in the component). We call a price that can be obtained in this way a submarket price; note that not all submarket prices can actually occur. Let P_m be the set of *submarket prices.*

Let P_i be the set consisting of the initial price and zero. A price is *1-linked* if it is of the form $(U_1/U_2)p$ where $p \in P_c \cup P_m \cup P_i$ and U_1 and U_2 are products of at most n utilities each. A price is *2-linked* if it of the form $(U_1/U_2)p$, where p is 1-linked and U_1 and U_2 are products of at most n utilities each.

▶ **Lemma 12.** *Assuming that all budgets, happiness caps and utilities are integers bounded by U, 1-linked and 2-linked prices are rational numbers whose bit-length is at most* $\log(m+n) + 3(m+n)\log U$.

▶ **Lemma 13.** *At the beginning of a phase, all prices are 1-linked. During a phase, prices outside S are 1-linked. At the end of each iteration, prices in S are 2-linked.*

▶ **Theorem 14.** *The algorithm in Figure 1 computes a modest MBB equilibrium.*

Proof. When the algorithm terminates, we claim that at this stage total surplus $\sum_j r_j = 0$. This will imply that the algorithm in Figure 1 computes a market equilibrium. Consider any good j and the component C of the equality graph containing good j. The total surplus

$\sum_{j \in C} r_j$ in the component is $\sum_{j \in C} p_j - \sum_{i \in C} M_i^a$. This is non-negative and less than ϵ. All prices and active budgets of capped buyers can be expressed in terms of one price variable p using equality relations. By Lemma 12, p is a rational number with bounded denominator, and the above inequalities imply $\sum_{j \in C} r_j = 0$. Thus $r_j = 0$ for all j. ◀

Let x_{ts} denote the value of x when Event 3 occurs in the algorithm. Next we show that x_{ts} can be computed using at most n max-flow computations. This is a generalization of a procedure in [17] for computing tight set in case of *linear* Fisher markets.

▶ **Lemma 15.** *x_{ts} can be computed using at most n max-flow computation.*

Maximum Revenue. Finally, we show that our algorithm gives a modest MBB equilibrium with maximum revenue among all modest MBB equilibria.

▶ **Lemma 16.** *Consider the price vector $\mathbf{p}$ at the end of any phase of the algorithm. We have $\mathbf{p} \geq \mathbf{p}'$ for any price vector $\mathbf{p}' \in \mathcal{P}$ of a modest MBB equilibrium.*

For the main result in this section, assume that all budgets, happiness caps and utilities are integers bounded by U.

▶ **Theorem 17.** *The algorithm in Figure 1 computes a modest MBB equilibrium with maximum revenue in $O(mn^6(\log(m+n) + (m+n)\log U))$ time.*

Proof. In the beginning, the 2-norm of surplus vector r satisfies $\|r\|^2 \leq mn^2U^2$. By Theorem 14, the algorithm will terminate before the norm becomes $\|r'\|^2 = 1/m(m+n)^2U^{8(m+n)}$. Let k denote the number of phases when the surplus becomes r'. From Lemma 10, we have $\|r\|^2(1-1/4mn)^k = \|r'\|^2$, which implies that the total number of phases in the algorithm is $O(mn(\log(m+n) + (m+n)\log U))$.

In each phase, we have at most $2n$ iterations, and in each iteration we need to compute the maximum $0 \leq x \leq 1$ when one of the three events occurs. Let x_c, x_{eq} and x_{ts} respectively denote the maximum value of x where Event 1, 2 and 3 occurs. Clearly, x_c can be obtained in $O(n)$ time, x_{eq} can be obtained in $O(mn)$ time, and x_{ts} can be obtained using at most n max-flow computations due to Lemma 15. Further, we recompute a balanced flow in case of Event 2 which further requires at most n max-flow computations [17]. Since a max-flow can be obtained in $O(n^3)$ time, each iteration can be implemented in $O(n^4)$ time. Hence, the total running time of the algorithm is $O(mn^6(\log(m+n) + (m+n)\log U))$. ◀

We conjecture that the running time in Theorem 17 can be reduced by a factor of $\tilde{O}(n^2)$ using the perturbation technique from [19] which requires a max-flow to be computed only in a network with forest structure. We have not worked out the details.

4 Computing a Modest MBB Equilibrium with Minimum Revenue

In this section, we show how to transform in polynomial time any modest MBB equilibrium into one with minimum revenue using the postprocessing procedure in Fig. 2.

▶ **Theorem 18.** *The algorithm in Figure 2 computes a modest MBB equilibrium with minimum revenue.*

Proof. It is easy to check that throughout the algorithm, $(\mathbf{x}, \mathbf{p})$ always remains a modest MBB equilibrium. Assume by contradiction that at the end of the algorithm, $(\mathbf{x}, \mathbf{p})$ is not an equilibrium with smallest prices. Let $(\mathbf{x}', \mathbf{p}')$ be an equilibrium with smallest prices, and

```
Input: A market with a set of buyers $B$ and a set of goods $G$;
        Budget $M_i$, happiness cap $c_i$, and utility parameters $u_{ij}$, $\forall i \in B, j \in G$;
        Any modest MBB equilibrium $(\mathbf{x}, \mathbf{p})$;
Output: A modest MBB equilibrium $(\mathbf{x}, \mathbf{p})$ with minimum revenue;
Initialize active budget $M_i' \leftarrow \min\{M_i, \min_j c_i p_j / u_{ij}\}$ for each buyer $i$;
$S \leftarrow \{j | p_j > 0$ and $j$ does not have incident equality edges to any uncapped buyer$\}$;
While $S \neq \emptyset$
    $B' \leftarrow$ Set of buyers who have incident equality edges to $S$;
    $x \leftarrow 1$; Define prices and active budgets as follows:
        $p_j \leftarrow x p_j$, $\forall j \in S$; $M_i^a \leftarrow x M_i^a$, $\forall i \in B_c'$;
    Decrease $x$ continuously down from 1 until one of the following events occurs
    Event 1: $x$ becomes zero;
    Event 2: A new equality edge appears
    Recompute $N_p$ and $S$;
EndWhile
```

Figure 2 The postprocessing algorithm for an equilibrium with minimum revenue.

define $S_1 = \{j \mid p_j > p_j'\}$. By Lemma 4 property (3), all buyers in $\Gamma(S_1, \mathbf{p})$ are capped buyers. Because prices of goods in set S_1 decrease from $\mathbf{p}$ to $\mathbf{p}'$, every buyer i incident to S_1 in the equality graph with prices $\mathbf{p}$ will only have equality edges to S_1 with prices $\mathbf{p}'$. Therefore we have $i \in \Gamma(S_1, \mathbf{p}') = \Gamma(S_1, \mathbf{p})$ (the equality is again by Lemma 4). This implies $\Gamma(S_1, \mathbf{p})$ is also the set of buyers who have incident equality edges to S_1 with prices $\mathbf{p}$. Hence, set S is nonempty for the While loop, and the algorithm should not terminate. ◀

5 Extensions

In the previous section, we proposed an algorithm for computing a modest MBB equilibrium, which has a Pareto-optimal allocation. When we depart from the set of such equilibria, then utilities in market equilibrium are not uniquely determined. In fact, we show that market equilibria with maximum social welfare might not be modest MBB equilibria, and computing such optimal equilibria becomes NP-hard. As a corollary, we note that the proof can also be used to show NP-hardness for optimizing any constant norm of utility values.

▶ **Theorem 19.** *It is* NP*-hard to compute a market equilibrium that maximizes social welfare.*

▶ **Corollary 20.** *It is* NP*-hard to compute a market equilibrium* $(\mathbf{x}, \mathbf{p})$ *that maximizes* $\sum_i (u_i(\mathbf{x}))^\rho$*, for every constant* $\rho > 0$.

There are several ways of introducing satiation points into the utility function. Instead of a global cap, let us assume there is a cap c_{ij} for the utility buyer i can obtain from good j. A good-based budget-additive utility of buyer i is then $u_i(\mathbf{x}_i) = \sum_j \min(c_{ij}, u_{ij}x_{ij})$. This variant turns out to be an elementary special case of separable piecewise-linear concave (SPLC) utilities, in which every piece consists of a linear segment followed by a constant segment. We show that even finding a single market equilibrium here becomes PPAD-hard. The proof adjusts a construction put forward in [9].

▶ **Theorem 21.** *It is* PPAD*-hard to compute a market equilibrium in Fisher markets with good-based budget-additive utilities.*

References

1 Nir Andelman and Yishay Mansour. Auctions with budget constraints. In *Proc. 9th Scandinavian Workshop Algorithm Theory (SWAT)*, pages 26–38, 2004.

2 Kenneth Arrow and Gerard Debreu. Existence of an equilibrium for a competitive economy. *Econometrica*, 22(3):265–290, 1954.

3 Yossi Azar, Benjamin Birnbaum, Anna Karlin, Claire Mathieu, and C. Thach Nguyen. Improved approximation algorithms for budgeted allocations. In *Proc. 35th Intl. Coll. Automata, Languages and Programming (ICALP)*, volume 1, pages 186–197, 2008.

4 Xiaohui Bei, Jugal Garg, and Martin Hoefer. Ascending-price algorithms for unknown markets. In *Proc. 17th Conf. Economics and Computation (EC)*, 2016. To appear.

5 Benjamin Birnbaum, Nikhil Devanur, and Lin Xiao. Distributed algorithms via gradient descent for Fisher markets. In *Proc. 12th Conf. Electronic Commerce (EC)*, pages 127–136, 2011.

6 Niv Buchbinder, Kamal Jain, and Joseph Naor. Online primal-dual algorithms for maximizing ad-auctions revenue. In *Proc. 15th European Symp. Algorithms (ESA)*, pages 253–264, 2007.

7 Dave Buchfuhrer, Shaddin Dughmi, Hu Fu, Robert Kleinberg, Elchanan Mossel, Christos Papadimitriou, Michael Schapira, Yaron Singer, and Chris Umans. Inapproximability for VCG-based combinatorial auctions. In *Proc. 21st Symp. Discrete Algorithms (SODA)*, pages 518–536, 2010.

8 Deeparnab Chakrabarty and Gagan Goel. On the approximability of budgeted allocations and improved lower bounds for submodular welfare maximization and gap. *SIAM J. Comput.*, 39(6):2189–2211, 2010.

9 Xi Chen and Shang-Hua Teng. Spending is not easier than trading: On the computational equivalence of Fisher and Arrow-Debreu equilibria. In *Proc. 20th Intl. Symp. Algorithms and Computation (ISAAC)*, pages 647–656, 2009.

10 Yun Kuen Cheung, Richard Cole, and Nikhil Devanur. Tatonnement beyond gross substitutes? Gradient descent to the rescue. In *Proc. 45th Symp. Theory of Computing (STOC)*, pages 191–200, 2013.

11 Yun Kuen Cheung, Richard Cole, and Ashish Rastogi. Tatonnement in ongoing markets of complementary goods. In *Proc. 13th Conf. Electronic Commerce (EC)*, pages 337–354, 2012.

12 Bruno Codenotti and Kasturi Varadarajan. Computation of market equilibria by convex programming. In Nisan et al. [28], chapter 8.

13 Richard Cole and Lisa Fleischer. Fast-converging tatonnement algorithms for one-time and ongoing market problems. In *Proc. 40th Symp. Theory of Computing (STOC)*, pages 315–324, 2008.

14 Richard Cole and Vasilis Gkatzelis. Approximating the Nash social welfare with indivisible items. In *Proc. 47th Symp. Theory of Computing (STOC)*, pages 371–380, 2015.

15 Nikhil Devanur, Jugal Garg, Ruta Mehta, Vijay Vazirani, and Sadra Yazdanbod. A market for scheduling, with applications to cloud computing. CoRR abs/1511.08748, 2015.

16 Nikhil Devanur, Kamal Jain, Tung Mai, Vijay Vazirani, and Sadra Yazdanbod. New convex programs for Fisher's market model and its generalizations, 2016. CoRR abs/1603.01257.

17 Nikhil Devanur, Christos Papadimitriou, Amin Saberi, and Vijay Vazirani. Market equilibrium via a primal–dual algorithm for a convex program. *J. ACM*, 55(5), 2008.

18 Shahar Dobzinski, Michal Feldman, Inbal Talgam-Cohen, and Omri Weinstein. Welfare and revenue guarantees for competitive bundling equilibrium. In *Proc. 11th Conf. Web and Internet Economics (WINE)*, pages 300–313, 2015.

19 Ran Duan, Jugal Garg, and Kurt Mehlhorn. An improved combinatorial polynomial algorithm for the linear Arrow-Debreu market. In *Proc. 27th Symp. Discrete Algorithms (SODA)*, pages 90–106, 2016.

20 Ran Duan and Kurt Mehlhorn. A combinatorial polynomial algorithm for the linear Arrow-Debreu market. *Inf. Comput.*, 243:112–132, 2015.

21 Michal Feldman, Nick Gravin, and Brendan Lucier. Combinatorial Walrasian equilibrium. *SIAM J. Comput.*, 45(1):29–48, 2016.

22 Gagan Goel and Vijay Vazirani. A perfect price discrimination market model with production, and a rational convex program for it. *Math. Oper. Res.*, 36(4):762–782, 2011.

23 Sungjin Im, Janardhan Kulkarni, and Kamesh Munagala. Competitive algorithms from competitive equilibria: non-clairvoyant scheduling under polyhedral constraints. In *Proc. 46th Symp. Theory of Computing (STOC)*, pages 313–322, 2014.

24 Sungjin Im, Janardhan Kulkarni, and Kamesh Munagala. Competitive flow time algorithms for polyhedral scheduling. In *Proc. 56th Symp. Foundations of Computer Science (FOCS)*, pages 506–524, 2015.

25 Kamal Jain. A polynomial time algorithm for computing the Arrow-Debreu market equilibrium for linear utilities. *SIAM J. Comput.*, 37(1):306–318, 2007.

26 Kamal Jain and Vijay Vazirani. Eisenberg-Gale markets: Algorithms and game-theoretic properties. *Games Econom. Behav.*, 70(1):84–106, 2010.

27 Michael Kapralov, Ian Post, and Jan Vondrák. Online submodular welfare maximization: Greedy is optimal. In *Proc. 24th Symp. Discrete Algorithms (SODA)*, pages 1216–1225, 2013.

28 Noam Nisan, Éva Tardos, Tim Roughgarden, and Vijay Vazirani, editors. *Algorithmic Game Theory.* Cambridge University Press, 2007.

29 James Orlin. Improved algorithms for computing Fisher's market clearing prices. In *Proc. 42nd Symp. Theory of Computing (STOC)*, pages 291–300, 2010.

30 Tim Roughgarden and Inbal Talgam-Cohen. Why prices need algorithms. In *Proc. 16th Conf. Economics and Computation (EC)*, pages 19–36, 2015.

31 Vijay Vazirani. Combinatorial algorithms for market equilibria. In Nisan et al. [28], chapter 7.

32 László Végh. A strongly polynomial algorithm for generalized flow maximization. In *Proc. 46th Symp. Theory of Computing (STOC)*, pages 644–653, 2014.