A Combinatorial Polynomial Algorithm for the Linear Arrow-Debreu Market

Ran Duan[1] and Kurt Mehlhorn[2]

[1] Max-Planck-Institut für Informatik, Saarbrücken, Germany
duanran@mpi-inf.mpg.de
[2] Max-Planck-Institut für Informatik, Saarbrücken, Germany
mehlhorn@mpi-inf.mpg.de

Abstract. We present the first combinatorial polynomial time algorithm for computing the equilibrium of the Arrow-Debreu market model with linear utilities. Our algorithm views the allocation of money as flows and iteratively improves the balanced flow as in [Devanur et al. 2008] for Fisher's model. We develop new methods to carefully deal with the flows and surpluses during price adjustments. Our algorithm performs $O(n^6 \log(nU))$ maximum flow computations, where n is the number of persons and U is the maximum integer utility. The flows have to be presented as numbers of bitlength $O(n \log(nU))$ to guarantee an exact solution. Previously, [Jain 2007, Ye 2007] have given polynomial time algorithms for this problem, which are based on solving convex programs using the ellipsoid algorithm and the interior-point method.

1 Introduction

We provide the first combinatorial polynomial algorithm for computing the model of economic markets formulated by Walras in 1874 [15]. In this model, every person has an initial distribution of some goods and a utility function of all goods. The market clears at a set of prices if each person sells its initial goods and then uses its entire revenue to buy a bundle of goods with maximum utility. We want to find the market equilibrium in which every good is assigned a price so that the market clears. In 1954, Arrow and Debreu [2] proved that the market equilibrium always exists if the utility functions are concave. The result is prominently mentioned in their Nobel prize laudation and the market is usually referred to as the "Arrow-Debreu market". However, their proof is based on Kakutani's fixed-point theorem and hence non-constructive. Since then, many algorithmic results studied the linear version of this model, that is, all utility functions are linear.

The first polynomial time algorithm for the linear Arrow-Debreu model is given by Jain et al [12]; it is based on solving a convex program using the ellipsoid algorithm. Another polynomial-time algorithm was given by Ye [16]; it is based on solving a convex program using the interior-point method. The latter algorithm has a time bound of $O(n^4 \log U)$ which is faster than our algorithm. However, our algorithm has the advantage of being quite simple (see Figure 1 for a complete listing) and combinatorial, and hence, gives additional insight in the nature of the problem. We obtain equilibrium prices by a simple procedure that iteratively adjusts prices and allocations in a carefully chosen, but intuitive manner. Previous to our algorithms, combinatorial algorithms were only known for computing an approximate equilibrium for the Arrow-Debreu model. Devanur and Vazirani [8] gave an approximation scheme for computing the Arrow-Debreu model with running time $O(\frac{n^4}{\epsilon}\log\frac{n}{\epsilon})$, improving [13]. Recently, Ghiyasvand and Orlin [11] improved the running time to $O(\frac{n}{\epsilon}(m+n\log n))$.

Many combinatorial algorithms consider a simpler model proposed by Fisher (see [3]), in which every buyer possesses an initial amount of money instead of some goods. Eisenberg and Gale [9] reduced the problem of computing the Fisher market equilibrium to a concave cost maximization problem and thus gave the first polynomial algorithm for the Fisher market by the ellipsoid algorithm. The first combinatorial polynomial algorithm for an exact linear Fisher market equilibrium is given by Devanur et al [7]. They use the maximum flow algorithm as a black box in their algorithm. When the input data are integral, their algorithm needs $O(n^5\log U + n^4\log e_{max})$ max-flow computations, where n is the number of buyers, U the largest integer utility, and e_{max} the largest initial amount of money of a buyer. If we use the common $O(n^3)$ max-flow algorithm (see [1]), their running time is $O(n^8\log U + n^7\log e_{max})$.

We next define the model we will use in this paper and then discuss our main contributions.

1.1 Model and Definitions

We make the following assumptions on the model as in Jain's paper [12]:

1. There are n persons in the system. Each person i has only one good, which is different from the goods other people have. The good person i has is denoted by good i.
2. Each person has only one unit of good. So, if the price of good i is p_i, person i will obtain p_i units of money when selling its good.
3. Each person i has a linear utility function $\sum_j u_{ij}z_{ij}$, where z_{ij} is the amount of good j consumed by i.
4. Each u_{ij} is an integer between 0 and U.
5. For all i, there is a j such that $u_{ij} > 0$. (Everybody likes some goods.)
6. For all j, there is an i such that $u_{ij} > 0$. (Every good is liked by somebody.)
7. For every proper subset P of persons, there exist $i \in P$ and $j \notin P$ such that $u_{ij} > 0$.

All these assumptions, with the exception of the last, are without loss of generality. The last assumption implies that all the equilibrium prices are nonzero [12], and it is only useful for the next section. In Section 4, we will discuss more about the last assumption.

Let $p = (p_1, p_2, ..., p_n)$ denote the vector of prices of goods 1 to n, so they are also the budgets of persons 1 to n. In this paper, we denote the set of all buyers to be $B = \{b_1, b_2, ..., b_n\}$ and the set of all goods to be $C = \{c_1, c_2, ..., c_n\}$. So, if the price of goods c_i is p_i, buyer b_i will have p_i amount of money. For a subset B' of persons or a subset C' of goods, we also use $p(B')$ or $p(C')$ to denote the total prices of the goods the persons in B' own or the goods in C'. For a vector $v = (v_1, v_2, ..., v_k)$, let:

- $|v| = |v_1| + |v_2| + ... + |v_k|$ be the l_1-norm of v.
- $\|v\| = \sqrt{v_1^2 + v_2^2 + ... + v_k^2}$ be the l_2-norm of v.

Each person only buys its favorite goods, that is, the goods with the maximum ratio of utility and price. Define its *bang per buck* to be $\alpha_i = \max_j\{u_{ij}/p_j\}$. The classical Arrow-Debreu [2] theorem says that there is a non-zero market clearing price vector.

For the current price vector p, the "equality graph" is a flow network $G = (\{s, t\} \cup B \cup C, E_G)$, where s is the source node and t is the sink node, then $B = \{b_1, ..., b_n\}$ denotes the set of buyers and $C = \{c_1, ..., c_n\}$ denotes the set of goods. E_G consists of:

- Edges from s to every node b_i in B with capacity p_i.
- Edges from every node c_i in C to t with capacity p_i.
- Edges from b_i to c_j with infinite capacity if $u_{ij}/p_j = \alpha_i$. Call these edges "equality edges".

So, our aim is to find a price vector p such that there is a flow in which all edges from s and to t are saturated, i.e., $(s, B \cup C \cup t)$ and $(s \cup B \cup C, t)$ are both minimum cuts. When this is satisfied, all goods are sold and all of the money earned by each person is spent.

In a flow f, define the surplus $r(b_i)$ of a buyer i to be the residual capacity of the edge (s, b_i), and define the surplus $r(c_j)$ of a good j to be the residual capacity of the edge (c_j, t). That is, $r(b_i) = p_i - \sum_j f_{ij}$, and $r(c_j) = p_j - \sum_i f_{ij}$, where f_{ij} is the amount of flow in the edge (b_i, c_j). Define the surplus vector of buyers to be $r(B) = (r(b_1), r(b_2), ..., r(b_n))$. Also, define the total surplus to be $|r(B)| = \sum_i r(b_i)$, which is also $\sum_j r(c_j)$ since the total capacity from s and to t are both equal to $\sum_i p_i$. For convenience, we denote the surplus vector of flow f' by $r'(B)$. In the network corresponding to market clearing prices, the total surplus of a maximum flow is zero.

1.2 Overview of our algorithm

The overall structure of our algorithm is similar to the ones of Devanur et al. [7] and Orlin [14] for computing equilibrium prices in Fisher markets, however, the details are quite different. The algorithm works iteratively. It starts with all prices equal to one. In each iteration it adjusts prices and allocations. The adjustment is guided by the analysis of a maximum flow in the equality graph.

In each iteration we first compute a balanced maximum flow [7]. A balanced maximum flow[3] is a maximum flow that minimizes the l_2-norm of the surplus

[3] In contrast to [7] the balanced maximum flow is not unique.

vector $r(B)$. We then order the buyers in order of decreasing surpluses: $b_1, \dots, b_n$. We find the minimal i such that $r(b_i)$ is substantially larger (by a factor of $1 + 1/n$) than $r(b_{i+1})$; $i = n$ if there is no such i. Let $B' = \{b_1, \dots, b_i\}$ and let $\Gamma(B')$ be the goods that are adjacent to a node in B' in the equality graph. There is no flow from the buyers in $B \setminus B'$ to the goods in $\Gamma(B')$; this is due to the fact that the flow is balanced. We raise the prices of and the flows[4] into the nodes of $\Gamma(B')$ by a common factor x. This affects the surpluses of the buyers, some go up and some go down. More precisely, there are four kind of buyers, depending on whether a buyer belongs to B' or not and on whether the good owned by the buyer belongs to $\Gamma(B')$ or not. We increase the prices until one of three events happens: (1) a new edge enters the equality graph[5] (2) the surplus of a buyer in B' and a buyer in $B \setminus B'$ becomes equal, or (3) x reaches a substantial value $(1 + 1/(n^3)$ in our algorithm).[6] This ends the description of an iteration.

In what sense are we making progress? The l_2-norm of the surplus vector does not decrease in every iteration.[7] In (3), the l_2-norm may increase. However, also at least one price increases significantly. Since we can independently upper bound the prices, we can bound the number of iterations in which event (3) occurs, and as a consequence, the total increase of the l_2-norm of the surplus vector. When event (1) or (2) occurs, the l_2-norm of the surplus vector decreases substantially, since surplus moves from a buyer in B' to a buyer in $B \setminus B'$ and buyers in these two groups have, by the choice of groups, substantially different surpluses.

We continue until the l_2-norm of the surplus vector is sufficiently small, so that a simple rounding procedure yields equilibrium prices.

1.3 Other Results

Recently, Orlin [14] improved the running time for computing the linear Fisher's model to $O(n^4 \log U + n^3 \log e_{max})$ and also gave the first strongly polynomial algorithm with running time $O(n^4 \log n)$. The problem of finding strongly polynomial algorithm for the linear Arrow-Debreu model is still open.

There are also algorithms considering Arrow-Debreu model with non-linear utilities [6,5]. The CES (constant elasticity of substitution) utility functions have drawn much attention, where the utility functions are of the form $u(x_1, ..., x_n) = (\sum_{j=1}^{n} c_j x_j^p)^{1/p}$ for $-\infty < p < 1$ and $p \neq 0$. [5] has shown that for $p > 0$ and $-1 \leq p < 0$, there are polynomial algorithms by convex program. However, Chen, Paparas and Yannakakis [4] have shown that it is PPAD-hard to solve market equilibrium of CES utilities for $p < -1$. They also define a new concept

[4] In [7,14] only prices are raised and flows stay the same. This works for Fisher's model because budgets are fixed. However, in the Arrow-Debreu model, an increase of prices of goods will also increase the budgets of their owners.

[5] The increase of prices of goods in $\Gamma(B')$ makes the goods in $C \setminus \Gamma(B')$ more attractive and hence an equality edge connecting a buyer in B' with a good in $C \setminus \Gamma(B')$ may come into existence. This event also exists in [7,14].

[6] Events (2) and (3) have no parallel in [7,14].

[7] In [7] the balance is strictly decreasing.

"Non-monotone utilities", and show the PPAD-hardness to solve the markets with non-monotone utilities. It remains open to find the exact border between tractable and intractable utility functions.

2 The algorithm

As in [7], our algorithm finds a balanced flow and increases the prices in the "active subgraph". But, in the Arrow-Debreu model, when we increase the prices of some good i, the budget of buyer i will also increase. So, we need to find a careful way to prevent the total surplus from increasing.

2.1 Balanced flow

As in [7], we define the concept of balanced flow to be a maximum flow that balances the surpluses of buyers. (However, unlike in their paper, the surpluses of goods can be positive here, which are not supposed to be balanced, so the balanced flow is not necessarily unique.)

Definition 1. *In the network G of current p, a* balanced flow *is a maximum flow that minimizes $\|r(B)\|$ over all choices of maximum flows.*

For flows f and f' and their surplus vectors $r(B)$ and $r'(B)$, respectively, if $\|r(B)\| < \|r'(B)\|$, then we say f is *more balanced* than f'. The next lemma shows why it is called "balanced".

Lemma 1. *[7] If $a \geq b_i \geq 0, i = 1, 2, ..., k$ and $\delta \geq \sum_{i=1}^{k} \delta_i$, where $\delta, \delta_i \geq 0$, $i = 1, 2, ..., k$, then:*

$$\|(a, b_1, b_2, ..., b_k)\|^2 \leq \|(a + \delta, b_1 - \delta_1, b_2 - \delta_2, ..., b_k - \delta_k)\|^2 - \delta^2. \tag{1}$$

Proof.

$$(a + \delta)^2 + \sum_{i=1}^{k}(b_i - \delta_i)^2 - a^2 - \sum_{i=1}^{k} b_i^2 \tag{2}$$

$$\geq 2a\delta + \delta^2 - 2\sum_{i=1}^{k} b_i\delta_i \geq \delta^2 + 2a(\delta - \sum_{i=1}^{k} \delta_i) \geq \delta^2. \tag{3}$$

Lemma 2. *[7] In the network G for a price vector p, given a maximum flow f, a balanced flow f'' can be computed by at most n max-flow computations.*

Proof. In the residual graph G_f w.r.t. to f, let $S \subseteq B \cup C$ be the set of nodes reachable from s, and let $T = (B \cup C) \setminus S$ be the remaining nodes. Then, there are no edges from $S \cap B$ to $T \cap C$ in the equality graph, and there is no flow from $T \cap B$ to $S \cap C$. The buyers in $T \cap B$ and the goods in $S \cap C$ have no surplus w.r.t. f, and this holds true for every maximum flow. Let G' be the network spanned by $s \cup S \cup t$, and let f' be the balanced maximum flow in G'. The f' can be computed by n max-flow computations. (Corollary 8.8 in [7] is applicable since $(s \cup S, t)$ is a min-cut in G'.) Finally, f' together with the restriction of f to $s \cup T \cup t$ is a balanced flow f'' in G.

The surpluses of all goods in f'' are the same as those in f since we only balance the surpluses of buyers.

2.2 Price adjustment

We need to increase the prices of some goods to get more equality edges ([7,14]). For a subset of buyers B_1, define its neighborhood $\Gamma(B_1)$ in the current network to be: $\Gamma(B_1) = \{c_j \in C | \exists b_i \in B_1, s.t.\ (b_i, c_j) \in E_G\}$. Clearly, there is no edge in G from B_1 to $C \setminus \Gamma(B_1)$. In a balanced flow f, given a surplus bound $S > 0$, let $B(S)$ denote the subset of buyers with surpluses at least S, that is, $B(S) = \{b_i \in B | r(b_i) \geq S\}$. We can see the goods in $\Gamma(B(S))$ have no surplus.

Lemma 3. *In a balanced flow f, given a surplus bound S, there is no edge that carries flow from $B \setminus B(S)$ to $\Gamma(B(S))$.*

Proof. Suppose there is such an edge (b_i, c_j) that carries flow such that $b_i \notin B(S)$ and $c_j \in \Gamma(B(S))$. Then, in the residual graph, there are directed edges (b_k, c_j) and (c_j, b_i) with nonzero capacities in which $b_k \in B(S)$. However, $r(b_k) \geq S > r(b_i)$, so we can augment along this path and get a more balanced flow, contradicting that f is already a balanced flow.

From Lemma 3, we can increase the prices in $\Gamma(B(S))$ by the same factor x without inconsistency. There is no edge from $B(S)$ to $C \setminus \Gamma(B(S))$, and the edges from $B \setminus B(S)$ to $\Gamma(B(S))$ are not carrying flow, and hence, there will be no harm if they disappear from the equality graph. If there are edges (b_i, c_j) and (b_i, c_k) where $b_i \in B(S), c_j, c_k \in \Gamma(B(S))$, then $u_{ij}/p_j = u_{ik}/p_k$. Since the prices in $\Gamma(B(S))$ are multiplied by a common factor x, u_{ij}/p_j and u_{ik}/p_k remain equal after a price adjustment. However, the goods in $C \setminus \Gamma(B(S))$ will become more attractive, so there may be edges from $B(S)$ to $C \setminus \Gamma(B(S))$ entering the network, and the increase of prices needs to stop when this happens. Define such a factor to be $X(S)$, that is,

$$X(S) = \min\{\frac{u_{ij}}{p_j} \cdot \frac{p_k}{u_{ik}} | b_i \in B(S), (b_i, c_j) \in E_G, c_k \notin \Gamma(B(S))\}. \tag{4}$$

So, we need $O(n^2)$ multiplications/divisions to compute $X(S)$. When we increase the prices of the goods in $\Gamma(B(S))$ by a common factor $x \leq X(S)$, the equality edges in $B(S) \cup \Gamma(B(S))$ will remain in the network. We will also need the following theorem to prevent the total surplus from increasing.

Theorem 1. *Given a balanced flow f in the current network G and a surplus bound S, we can multiply the prices of goods in $\Gamma(B(S))$ with a parameter $x > 1$. When $x \leq \min_i\{p_i/(p_i - r(b_i)) | b_i \in B(S), c_i \notin \Gamma(B(S))\}$ and $x \leq X(S)$, we obtain a flow f' in the new network G' of adjusted prices with the same value of total surplus by:*

$$f'_{ij} = \begin{cases} x \cdot f_{ij} & \text{if } c_j \in \Gamma(B(S)); \\ f_{ij} & \text{if } c_j \notin \Gamma(B(S)). \end{cases}$$

Then, the surplus of each good remains unchanged, and the surpluses of the buyers become:

$$r'(b_i) = \begin{cases} x \cdot r(b_i) & \text{if } b_i \in B(S), c_i \in \Gamma(B(S)); \\ (1-x)p_i + x \cdot r(b_i) & \text{if } b_i \in B(S), c_i \notin \Gamma(B(S)); \\ (x-1)p_i + r(b_i) & \text{if } b_i \notin B(S), c_i \in \Gamma(B(S)); \\ r(b_i) & \text{if } b_i \notin B(S), c_i \notin \Gamma(B(S)). \end{cases}$$

We call these kinds of buyers type 1 to type 4 buyers, respectively.

Proof. Since the flows on all edges associated with goods in $\Gamma(B(S))$ are multiplied by x, the surplus of each good in $\Gamma(B(S))$ remains zero. Only the surplus of type 2 buyers decreases because the flows from a type 2 buyer b_i are multiplied by x, but its budget p_i is not changed. The flow after adjustment is $x(p_i - r(b_i))$. We need this to be at most p_i, so $x \leq p_i/(p_i - r(b_i))$ for all type 2 buyers b_i, and in f', the new surplus $r'(b_i) = (1-x)p_i + xr(b_i)$.

Since both money and flows are multiplied by x for a type 1 buyer, their surplus is also multiplied by x. For a type 3 buyer b_i, their flows are not changed, but their money is multiplied by x, so the new surplus is $xp_i - (p_i - r(b_i))$.

After each price adjustment, in the new network, we will find a maximum flow by augmentation on the adjusted flow f' and then find a balanced flow by Lemma 2. This will guarantee that when the surplus of a good becomes zero, it will not change back to non-zero anymore. Thus, the prices of the goods with non-zero surpluses will not be adjusted.

Property 1. The prices of goods with non-zero surpluses remain unchanged in the algorithm.

2.3 Whole procedure

The whole algorithm is shown in Figure 1, where K is a constant we will set later. In this section, one iteration denotes one execution of the loop body.

In the first iteration, we construct a balanced flow f in the network where all prices are equal to 1. In the equality graph, we have at least one edge incident to every buyer. The total surplus will be bounded by n, actually $n-1$ as at least one good will be sold completely. In each iteration, we first update the prices and the flows as described in Theorem 1. By Theorem 1, in the execution of the algorithm, the total surplus will never increase. After having updated the flows and prices, we round them. The purpose of rounding is to control the bitlength of the numbers handled by the algorithm. Once the total surplus is sufficiently small, we stop and round the current solution to an exact solution. The rounding procedures and termination conditions are given by the following two lemmas, whose proofs will be discussed in Section 3.

Lemma 4. *[Restated and proven as Corollary 1 in Section 3] For $\Delta = n^{O(1)}U^{O(n)}$ and a surplus bound $\epsilon \geq n^5/\Delta$, we can adjust prices and flows to bitlength $O(n\log(nU))$, so that the l_2-norm of the surplus vector only increases by a factor of $1 + O(1/n^4)$.*

Lemma 5. *[Restated and proven as Lemma 11 in Section 3] When the total surplus is* $< \frac{1}{4n^4U^{3n}} = \epsilon$ *in a flow* f*, we can obtain an exact solution from the current equality graph.*

Initially set $p_i = 1$ for all goods i;
Repeat
  Construct the network G for the current p, and compute the balanced flow f in it;
  Sort all buyers by their surpluses in decreasing order: $b_1, b_2, ..., b_n$;
  Find the first i in which $\frac{r(b_i)}{r(b_{i+1})} > 1 + 1/n$, and $i = n$ when there is no such i;
  Let $S = r(b_i)$ and obtain $B(S), \Gamma(B(S)), X(S)$; $(B(S) = \{b_1, b_2, ..., b_i\})$
  Multiply the prices in $\Gamma(B(S))$ by a gradually increasing factor $x > 1$ until:
  (Let f' be the flow corresponding to x which is constructed according to Theorem 1.)
    New equality edges emerge (x reaches $X(S)$);
    OR the surplus of a buyer $\in B(S)$ and a buyer $\notin B(S)$ equals in f';
    OR x reaches $1 + \frac{1}{K \cdot n^3}$
  Round the prices in $\Gamma(B(S))$ according Lemma 4 with $\Delta = 4n^9U^{3n}$;
Until $|r(B)| < \epsilon$, where $\epsilon = \frac{1}{4n^4U^{3n}}$;
Finally, round the prices according to Lemma 5 to get an exact solution.

Fig. 1. The whole algorithm

To ensure that the algorithm will terminate in a polynomial number of steps, we will require the following lemmas. From Property 1, the prices of goods with non-zero surpluses stay one during the whole algorithm, so there is still a good with price one in the end. And, we need to bound the largest price:

Lemma 6. *The prices of goods are at most* $(nU)^{n-1}$.

Proof. It is enough to show that during the entire algorithm, for any non-empty and proper subset $\hat{C}$ of goods, there are goods $c_i \in \hat{C}, c_j \notin \hat{C}$ such that $p_i/p_j \leq nU$. So, when we sort all the prices in decreasing order, the ratio of two adjacent prices is at most nU. Since there is always a good with price 1, the largest price is $\leq (nU)^{n-1}$.

If $\hat{C}$ contains goods with surpluses, then their price is 1. The claim follows.

Let $\hat{B} = \Gamma(\hat{C})$ be the set of buyers adjacent to goods in $\hat{C}$ in the equality graph. If there exist b_i, c_j s.t. $b_i \in \hat{B}, c_j \notin \hat{C}$ and $u_{ij} > 0$, let $c_k \in \hat{C}$ be one of the goods adjacent to b_i in the equality graph, and then $u_{ij}/p_j \leq u_{ik}/p_k$. So, $p_k/p_j \leq u_{ik}/u_{ij} \leq U$.

If there do not exist such b_i, c_j, then there is no edge between $\hat{B}$ and $C \setminus \hat{C}$, and there is $b_k \notin \hat{B}$, but $c_k \in \hat{C}$. Otherwise the persons whose goods are in $\hat{C}$ will not like any goods not in $\hat{C}$, contradicting assumption (7). Let $B' = \{j | b_j \in \hat{B}, c_j \notin \hat{C}\}$ and $B'' = \{j | b_j \notin \hat{B}, c_j \in \hat{C}\}$. We have:

$$p_k \leq p(B'') = p(\hat{C}) - p(\{j | b_j \in \hat{B}, c_j \in \hat{C}\}) \tag{5}$$

$$\leq p(\hat{B}) - p(\{j | b_j \in \hat{B}, c_j \in \hat{C}\}) = p(B'). \tag{6}$$

The inequality of the second line holds since goods in $\hat{C}$ have surplus 0 and all of the flows to $\hat{C}$ come from $\hat{B}$. Thus, B' must be non-empty, and hence, there is a $j \in B'$ with $p_j \geq p(B')/n$. We conclude $p_k \leq np_j$.

By Lemma 5, we can round to the exact solution when the algorithm terminates. To analyze the correctness and running time, we need the following lemma:

Lemma 7. *After every price adjustment by x, the l_2-norm of the surplus vector $\|r(B)\|$ will either*

- *be multiplied by a factor of $1 + O(1/n^3)$ when $x = 1 + \frac{1}{Kn^3}$, or*
- *be divided by a factor of $1 + \Omega(1/n^3)$.*

Note that by Lemma 4, the rounding procedure can only increase $\|r(B)\|$ by a factor of $1 + O(1/n^4)$. Thus, the statement of Lemma 7 also holds after rounding.

Theorem 2. *In total, we need to compute $O(n^6 \log(nU))$ maximum flows, and the length of numbers is bounded by $O(n \log(nU))$. Thus, if we use the common $O(n^3)$ max-flow algorithm (see [1]), the total running time is $O(n^{10} \log^2(nU))$.*

Proof. By Lemma 6, every price can be multiplied by $x = 1 + \frac{1}{Kn^3}$ for $O(\log_{1+1/(Kn^3)}(nU)^n) = O(n^4 \log(nU))$ times, so the total number of iterations of the first type is $O(n^5 \log(nU))$. The total factor multiplied to $\|r(B)\|$ by the first type iterations is $(1 + O(1/n^3))^{O(n^5 \log(nU))}$.

At the beginning, $\|r(B)\| \leq \sqrt{n}$. When the algorithm terminates, $\|r(B)\| < \epsilon = \frac{1}{4n^4U^{3n}}$, so the number of second type iterations is bounded by

$$\log_{1+\Omega(1/n^3)}(\frac{1}{\epsilon}\sqrt{n}(1 + O(1/n^3))^{O(n^5 \log(nU))}) = O(n^5 \log(nU)). \qquad (7)$$

Thus, the total number of iterations performed is bounded by $O(n^5 \log(nU))$. Since we need to compute n max-flows for the balanced flow in every iteration, we need $O(n^6 \log(nU))$ maximum flow computations in total. By Lemma 6 and Lemma 4, the length of the numbers to be handled is bounded by $O(n \log(nU))$. Note that max-flow computations only need additions and subtractions. We perform multiplications and divisions when we scale prices and when we set up the max-flow computation in the computation of balanced flow. The numbers of multiplications/divisons is by a factor n less than the numbers of additions/subtractions, and hence, it suffices to charge $O(n \log(nU))$ per arithmetic operation.

Next we will prove Lemma 7. When we sort all the buyers by their surpluses $b_1, b_2, ..., b_n$ in decreasing order, b_1 is at least $|r(B)|/n$ (where $|r(B)|$ is the total surplus). So, for the first i in which $\frac{r(b_i)}{r(b_{i+1})} > 1+1/n$, we can see $\frac{r(b_j)}{r(b_{j+1})} \leq 1+1/n$ for $j < i$, so $r(b_i) \geq r(b_1)(1 + 1/n)^{-n} > |r(B)|/(e \cdot n)$. When such an i does not exist, each $r(b_i)$ is larger than $|r(B)|/(e \cdot n)$, and all goods in $\Gamma(B)$ must have zero surplus because the flow is otherwise not maximum. Thus, there are goods that have no buyers, and hence, either new equality edges emerge, or x reaches $1 + \frac{1}{Kn^3}$ (condition (3a) below).

From the algorithm, in every iteration, x satisfies the following conditions:

1. $x \le 1 + \frac{1}{Kn^3}$.
2. In f', $r'(b) \ge r'(b')$ for all $b \in B(S), b' \notin B(S)$. Here, $r'(b)$ is the surplus of b w.r.t. f', the flow corresponding to x by Theorem 1.
3. If $x < 1 + \frac{1}{Kn^3}$, the following possibilities arise:
   (a) There is a new equality edge (b_i, c_j) with $b_i \in B(S), c_j \notin \Gamma(B(S))$. By Lemma 8 below, we can obtain a flow f'' in which either $r''(b_i) = r'(b_i) - p_j$, or there is a $b_k \notin B(S)$ with $r''(b_i) = r''(b_k)$ (same as (b)).
   (b) When x satisfies the second requirement in the algorithm, it satisfies: there exist $b \in B(S)$ and $b' \notin B(S)$ such that $r'(b) = r'(b')$ in f'.

Lemma 8. *If there is a new equality edge (b_i, c_j) with $b_i \in B(S), c_j \notin \Gamma(B(S))$, we can obtain a flow f'' from f' (without increasing the total surplus) in which either $r''(b_i) = r'(b_i) - p_j$, or there is a $b_k \notin B(S)$ with $r''(b_i) = r''(b_k)$.*

Proof. Let $B' \subseteq B \setminus B(S)$ be the set of buyers with flows to c_j in f', and let w be the largest surplus of a buyer in $B \setminus B(S)$. Run the following procedure (f'' denotes the current flow in the algorithm):

Augment along (b_i, c_j) gradually until:
    $r''(b_i) = w$ or $r''(c_j) = 0$;
If $r''(b_i) = w$ then Exit;
For all $b_k \in B'$ in any order
    Augment along (b_i, c_j, b_k) gradually until:
        $r''(b_i) = \max\{r''(b_k), w\}$ or $f''(b_k, c_j) = 0$;
    Set $w = \max\{r''(b_k), w\}$;
    If $r''(b_i) = w$ then Exit.

During the procedure, the surplus of b_i decreases but cannot become less than the surplus of a buyer in $B \setminus B(S)$, so condition (2) holds. In the end, if $r''(b_i) = w$, then there is a $b_k \in B \setminus B(S)$ s.t. $r''(b_i) = r''(b_k)$; otherwise, c_j has no surplus, and the flows to it all come from b_i, so $r''(b_i) = r'(b_i) - p_j$.

From Theorem 1, the surpluses in f' will increase for type 1 and 3 buyers, will decrease for type 2 buyers, and will stay unchanged for type 4 buyers. Note that the surplus of a type 1 or 2 buyer cannot be smaller than the surplus of any type 3 or 4 buyer. From Theorem 1 and Lemma 8, we infer that the total surplus will not increase, type 2 and 3 buyers will get more balanced, and $r'(b) = x \cdot r(b)$ for type 1 buyers b, so $\|r'(B)\| \le x\|r(B)\| = (1 + O(1/n^3))\|r(B)\|$.

In (3a), there is a new equality edge (b_i, c_j). After the procedure described in Lemma 8, if there is no $b_k \notin B(S)$ such that $r''(b_i) = r''(b_k)$, then $r''(b_i) = r'(b_i) - p_j$ $(p_j \ge 1)$. For all $b_k \notin B(S)$, $r''(b_i) > r''(b_k)$, and $r''(b_k) = r'(b_k) + \delta_k$, where $\delta_k \geq 0$ and $\sum_{b_k \notin B(S)} \delta_k \leq p_j$. Because $|r(B)| \leq n$, $\|r(B)\|^2 \leq n^2$. By Lemma 1,

$$\|r''(B)\|^2 \leq \|r'(B)\|^2 - p_j^2 \tag{8}$$
$$\leq x^2\|r(B)\|^2 - 1 \tag{9}$$
$$\leq x^2\|r(B)\|^2 - \frac{1}{n^2}\|r(B)\|^2 \tag{10}$$
$$= (1 - \Theta(1/n^2))\|r(B)\|^2. \tag{11}$$

So, we have $\|r''(B)\| = (1 - \Omega(1/n^2))\|r(B)\|$.

In (3a), after the procedure described in Lemma 8, if there is $b_k \notin B(S)$ such that $r''(b_i) = r''(b_k)$, then we are in a similar situation as in (3b), possibly with an even smaller total surplus. So, we can prove this case by the proof of (3b).

In (3b), let $u_1, u_2, ..., u_k$ and $v_1, v_2, ..., v_{k'}$ be the list of original surpluses of type 2 and 3 buyers, respectively. Define $u = \min\{u_i\}, v = \max\{v_j\}$, so $u_i \geq u$ for all i, and $v_j \leq v$ for all j, and $u > (1 + 1/n)v$. After the price and flow adjustments in Theorem 1, the list of surpluses will be $u_1 - \delta_1, u_2 - \delta_2, ..., u_k - \delta_k$ and $v_1 + \delta'_1, v_2 + \delta'_2, ..., v_k + \delta'_{k'}$ (here $\delta_i, \delta'_j \geq 0$ for all i, j), and there exist I, J such that $u_I - \delta_I = v_J + \delta'_J$, where $u_I - \delta_I$ is the smallest among $u_i - \delta_i$, and $v_J + \delta'_J$ is the largest among $v_j + \delta'_j$ by condition (2). Since the surpluses of type 1 buyers also increase (and the total surplus may decrease), we have $\sum_i \delta_i \geq \sum_j \delta'_j$, $\delta_I \leq \sum_i \delta_i$, and $\delta'_J \leq \sum_j \delta'_j$. Compute:

$$\sum_i (u_i - \delta_i)^2 + \sum_j (v_j + \delta'_j)^2 - (\sum_i u_i^2 + \sum_j v_j^2) \tag{12}$$
$$= -2\sum_i u_i\delta_i + 2\sum_j v_j\delta'_j + \sum_i \delta_i^2 + \sum_j {\delta'}_j^2 \tag{13}$$
$$\leq -u\sum_i \delta_i + v\sum_j \delta'_j - \sum_i \delta_i(u_i - \delta_i) + \sum_j \delta'_j(v_j + \delta'_j) \tag{14}$$
$$\leq -(u - v)\sum_i \delta_i - (u_I - \delta_I)\sum_i \delta_i + (v_J + \delta'_J)\sum_j \delta'_j \tag{15}$$
$$\leq -(u - v)\sum_i \delta_i \tag{16}$$
$$\leq -(u - v)\max\{\delta_I, \delta'_J\} \tag{17}$$
$$\leq -\frac{1}{2}(u - v)^2 \tag{18}$$
$$< -\frac{1}{2(n + 1)^2}u^2. \tag{19}$$

Let $w_1, w_2, ...w_{k''}$ be the list of surpluses of type 1 buyers; all of them are $\leq e \cdot u$. After price adjustment, the surpluses will be $x \cdot w_1, x \cdot w_2, ...x \cdot w_{k''}$ from Theorem 1. Compute:

$$\sum_i (xw_i)^2 \leq (1 + \frac{1}{Kn^3})^2 \sum_i w_i^2 \tag{20}$$

$$\leq \sum_i w_i^2 + (\frac{2}{Kn^3} + \frac{1}{K^2n^6}) \cdot ne^2u^2 \tag{21}$$

$$= \sum_i w_i^2 + (\frac{2}{Kn^2} + \frac{1}{K^2n^5})e^2u^2. \tag{22}$$

Let $K = 32e^2$, then the change to the sum of squares of surpluses for type 2 and 3 buyers is less than $-\frac{1}{8n^2}u^2 = -\frac{4}{Kn^2}e^2u^2$. The total change to $\|r(B)\|^2$ is:

$$< (-\frac{2}{Kn^2} + \frac{1}{K^2n^5})e^2u^2. \tag{23}$$

Since $u \geq \frac{1}{e}r(b_i)$ for all buyers b_i, $nu^2 \geq \frac{1}{e^2}\|r(B)\|^2$. Since the change is negative:

$$\|r'(B)\|^2 < \|r(B)\|^2 + (-\frac{2}{Kn^2} + \frac{1}{K^2n^5})\frac{1}{n}\|r(B)\|^2 \tag{24}$$

$$= \|r(B)\|^2 - \frac{2}{Kn^3}\|r(B)\|^2 + \frac{1}{K^2n^6}\|r(B)\|^2 \tag{25}$$

$$= \|r(B)\|^2(1 - \frac{1}{Kn^3})^2. \tag{26}$$

Thus, Lemma 7 is proved.

3 Rounding and termination condition

In this section, we will show how to round the prices to rational numbers with denominators of length $O(n\log(nU))$. Also, we need the rounding process to obtain an exact market equilibrium when the surplus is very small. Here, we define the *undirected equality graph* F on $B \cup C$ of *undirected* equality edges between buyers and goods, and we consider every connected component in this undirected equality graph.

Lemma 9. *In a connected component Ψ containing k goods in the undirected equality graph, if we know that p_j is a rational number with denominator N, where $c_j \in \Psi \cap C$, then all the prices of goods in $\Psi \cap C$ are rational numbers with denominator $\leq N \cdot U^k$.*

Proof. Find a tree that connects all the goods in Ψ. The tree will contain $k+k'-1$ edges if it contains k' buyers. Then, we can get $k-1$ linear independent equations $p_j/u_{ij} = p_{j'}/u_{ij'}$ when both (b_i, c_j) and $(b_i, c_{j'})$ are tree edges. Together with the equation $p_j = I/N$ for some integer I, we can see that all the prices of goods in Ψ have denominator $\leq N \cdot U^k$.

For an integer Δ, we call a connected component in the undirected equality graph *consistent* if it has a good whose price is a rational number with denominator Δ. Then, by Lemma 9, the prices of goods in a consistent connected component are rational numbers with denominator $\leq \Delta \cdot U^n$.

Lemma 10. *In a balanced flow f of total surplus $\geq \epsilon \geq n^5/\Delta$, for $\Delta > nU^n$, if all the connected components in $(B \setminus B(S), C \setminus \Gamma(B(S)))$ (for the $B(S)$ in the original algorithm) are consistent, we can adjust the prices in $\Gamma(B(S))$ so that all connected components in the equality graph are consistent. In the adjusted flow f' by Theorem 1, $\|r'(B)\| = \|r(B)\|(1+O(\frac{n}{\epsilon \cdot \Delta}))$, where $r'(B)$ is the surplus vector in f'.*

Proof. Here $B(S)$ and $\Gamma(B(S))$ are the original ones in every iteration of the algorithm. The procedure is shown below:

Set $B' = B(S)$;
Repeat
  Multiply the prices in $\Gamma(B')$ by $x > 1$ until:
    A price in $\Gamma(B')$ has denominator Δ;
    OR a new equality edge emerges;
  Update the flow f by Theorem 1;
  Remove new consistent components from $B' \cup \Gamma(B')$;
Until $B' = \emptyset$.

Since all the prices change by at most $1/\Delta$, the changes to the total surplus of type 2 buyers is at most $n/\Delta < S$, so the surplus of every type 2 buyer is still positive. Since $\|r(B)\|^2 \geq \frac{1}{n}|r(B)|^2 \geq \frac{\epsilon}{n}|r(B)| \geq \frac{\epsilon^2}{n}$,

$$\|r'(B)\|^2 \tag{27}$$

$$\leq \|r(B)\|^2 + \frac{2}{\Delta}|r(B)| + \frac{n}{\Delta^2} \tag{28}$$

$$\leq \|r(B)\|^2 + \frac{2n}{\epsilon\Delta}\|r(B)\|^2 + \frac{n^2}{\epsilon^2\Delta^2}\|r(B)\|^2 \tag{29}$$

$$= \|r(B)\|^2(1 + \frac{n}{\epsilon\Delta})^2. \tag{30}$$

During the algorithm, we can see that all the connected components in $(B \setminus B', C \setminus \Gamma(B'))$ are consistent since we move the new consistent components to it. When we find new equality edges connecting B' and $C \setminus \Gamma(B')$, some nodes in $B' \cup \Gamma(B')$ will connect to $(B \setminus B', C \setminus \Gamma(B'))$, so these nodes can be removed. When a price in $\Gamma(B')$ has denominator Δ, the component containing it will become consistent, so the loop will run for at most n times. In each loop, we need to compute $O(n^2)$ multiplications/divisions, so the running time for this rounding procedure is less than the computation of a balanced flow.

Combining Lemma 6 and Lemma 10, we have the following corollary.

Corollary 1. *[Restatement of Lemma 4] For $\Delta = n^{O(1)}U^{O(n)}$ and a surplus bound $\epsilon \geq n^5/\Delta$, we can adjust prices to length $O(n\log(nU))$, so that in its adjusted flow, the l_2-norm of the surplus vector only increases by a factor of $1 + O(1/n^4)$.*

Lemma 11. *[Restatement of Lemma 5] When the total surplus is $< \frac{1}{4n^4U^{3n}} = \epsilon$ in a flow f, we can obtain an exact solution from the current equality graph.*

Proof. Add the edge (b_i, c_i) for each person i to the undirected equality graph F to obtain F'. For a connected component of F', the sum of the prices on both sides are the same. For every component Φ of F' with no surplus node, increase its prices by a common factor until a new equality edge emerges; this will unite two components. Repeat this until all components in F' have a surplus node. We may assume w.l.o.g. that F' becomes connected by this process. Otherwise, the following argument can be applied independently to each component of F'. The total surplus is still less than ϵ. The following rounding procedure will be performed on these revised prices.

Denote the set of connected components in the undirected equality graph F (not F') by $\Lambda = \{\Psi_k\}$. For each component Ψ_k in F, find a spanning tree T_k in it, then write the following equations:

$$p_j/u_{ij} = p_{j'}/u_{ij'}, \forall (b_i, c_j), (b_i, c_{j'}) \in T_k. \tag{31}$$

Since we have one such equation if c_j and $c_{j'}$ are connected by one b_i, we can have $|\Psi_k \cap C| - 1$ linear independent equations. The total number of linear independent equations for all components in F is $n - |\Lambda|$.

Since there is no flow between components, for each component Ψ_k in F, the money difference between buyers and goods in Ψ_k is only the surplus difference. So, we can write

$$\sum_{b_i \in B \cap \Psi_k} p_i - \sum_{c_i \in C \cap \Psi_k} p_i = \epsilon_k, \forall \Psi_k. \tag{32}$$

Here, ϵ_k (positive or negative) comes from the surpluses of goods and buyers, so $\sum |\epsilon_k| \le 2\epsilon$. If b_i and c_i belong to distinct connected components Ψ_j and Ψ_k, the coefficient of p_i is $+1$ in the equation of Ψ_j, -1 in the equation for Ψ_k, and 0 in all other equations. If b_i and c_i belong to the same connected component, the coefficient of p_i is zero in all equations. Assume now that there is a proper subset of the equations that is linear dependent. Then, if b_i or c_i belongs to one of the components in the subset, both of them do. However, the subset of components is a proper subgraph of F', and hence, there is at least one i such that only one of b_i or c_i belongs to the subset of components. Thus, we have $|\Lambda| - 1$ independent equations.

Since there is a good c_i with non-zero surplus, we have $p_i = 1$. Thus, the current price vector p is the solution of these linear equations $Ap = X$ in which A is invertible.

Consider the following n linear equations of price vector p' with ϵ_k removed:

$$p'_j/u_{ij} = p'_{j'}/u_{ij'}, \forall\ (b_i, c_j), (b_i, c_{j'}) \in T_k \tag{33}$$

$$\sum_{b_i \in B \cap \Psi_k} p'_i - \sum_{c_i \in C \cap \Psi_k} p'_i = 0, \forall\ \Psi_k \tag{34}$$

$$p'_i = 1, \exists\ r(c_i) > 0. \tag{35}$$

They can be denoted by $Ap' = X'$, so there is also a unique solution. The solution will be rational numbers with a common denominator $D \le nU^n$ by Cramer's rule. Since $||X| - |X'|| < 2\epsilon$, the difference $|p'_i - p_i|$ of solutions of each price is at most $2\epsilon \cdot nU^n = \frac{1}{2n^3U^{2n}}$ by Cramer's rule. The difference between any two different numbers of denominators $D, D' \leq nU^n$ is a positive rational number of denominator $D \cdot D' < n^2U^{2n}$, which is larger than $2|p'_i - p_i|$. Since p'_i is a rational number with denominator $D \leq nU^n$, we can get p'_i by rounding p_i to the nearest rational number of denominator $\leq nU^n$. This can be done by continued fraction expansion, which needs $O(n \log^2 D) = O(n^3 \log^2(nU))$ time by Theorem 3.13 in [10]. We can also compute $D = det(A)$ directly and round every price to the nearest rational with denominator D or solve the linear equations $Ap' = X'$. By Theorem 5.12 in [10], computing the determinant of a matrix of dimension n with entries $\leq U$ takes $\tilde{O}(n^4 \log U)$ time, and solving $Ap' = X'$ also takes $\tilde{O}(n^4 \log U)$ time.

Now, all the prices p'_i are of the form q_i/D, where q_i, D are integers and $D \leq nU^n$ a common denominator. So, $|p_i - \frac{q_i}{D}| \leq \frac{1}{2n^3U^{2n}} = \frac{\epsilon'}{D}$, in which $\epsilon' = \frac{D}{2n^3U^{2n}} \leq \frac{1}{2n^2U^n}$. Construct the flow network G' for the new prices $q = (q_1, q_2, ..., q_n)$. Consider any $b_i \in B$ and $c_j, c_k \in C$ and assume $u_{ij}/p_j \leq u_{ik}/p_k$. Then,

$$u_{ij}q_k \leq u_{ij}(p_kD + \epsilon') \tag{36}$$
$$\leq u_{ik}p_jD + u_{ij}\epsilon' \tag{37}$$
$$\leq u_{ik}(q_j + \epsilon') + u_{ij}\epsilon' \tag{38}$$
$$\leq u_{ik}q_j + (u_{ik} + u_{ij})\epsilon' \tag{39}$$
$$< u_{ik}q_j + 1, \tag{40}$$

and hence, $u_{ij}q_k \leq u_{ik}q_j$ since $u_{ij}q_k$ and $u_{ik}q_j$ are integral. We conclude that the edges in G are all in G'.

Denote the size of the cuts $(s, B \cup C \cup t)$ and $(s \cup B \cup C, t)$ in G' by Z, which is an integer. Then, the size of this cut in G is $\geq (Z - n\epsilon')/D$. If there is another cut in G' of size $\leq Z - 1$, it is also a cut in G, and its size in G is $\leq \frac{Z-1}{D} + 2n\frac{\epsilon'}{D} = Z/D - \frac{1-2n\epsilon'}{D}$, so the maximum flow in G will have total surplus $\geq \frac{1-3n\epsilon'}{D} > \epsilon$. Thus, $(s, B \cup C \cup t)$ and $(s \cup B \cup C, t)$ are both min-cuts in G', so the prices reach a market equilibrium.

4 General Case

Here, we consider the case which does not satisfy the assumption (7) in Section 1.1, i.e., there may be a proper subset P of persons such that $u_{ij} = 0$ for all $i \in P$ and $j \notin P$. Note that all other assumptions are satisfied, so the equilibrium exists [2].

The following procedure resembles the one in Secton 6 of [12]. We draw the liking graph of persons in which there is a directed edge from i to j iff $u_{ij} > 0$. If the graph is strongly connected, then the case satisfies assumption (7). Otherwise, if we shrink every strongly connected component into one vertex, then the graph will be a DAG (Directed acyclic graph), and we can find a topological order of strongly connected components: $P_1, P_2, ...P_k$, in which there are only edges from a lower order to a higher order. We use the algorithm in Section 2 to compute the equilibrium for all the persons in every strongly connected component P_i ($i = 1, 2, ..., k$). For $i = 2, ..., k$, multiplying the prices in P_i by $(U+1)\cdot\max\{p_j|j \in P_{i-1}\}$ will ensure that there are no equality edges from P_i to P_j for $i < j$. Since the persons in P_j do not like any goods in P_i for $i < j$, this will not affect the equilibrium of every component, so we get a global equilibrium.

References

1. R. K. Ahuja, T. L. Magnati, and J. B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, 1993.
2. Kenneth J. Arrow and Gérard Debreu. Existence of an equilibrium for a competitive economy. *Econometrica*, 22:265–290, 1954.
3. William C. Brainard and Herbert E. Scarf. How to compute equilibrium prices in 1891. Cowles Foundation Discussion Papers 1272, Cowles Foundation for Research in Economics, Yale University, August 2000.
4. Xi Chen, Dimitris Paparas, and Mihalis Yannakakis. The complexity of non-monotone markets. *CoRR*, abs/1211.4918, 2012.
5. Bruno Codenotti, Benton McCune, Sriram Penumatcha, and Kasturi Varadarajan. Market equilibrium for CES exchange economies: existence, multiplicity, and computation. In *Proceedings of the 25th international conference on Foundations of Software Technology and Theoretical Computer Science*, FSTTCS '05, pages 505–516, Berlin, Heidelberg, 2005. Springer-Verlag.
6. Bruno Codenotti, Benton Mccune, and Kasturi Varadarajan. Market equilibrium via the excess demand function. In *In Proceedings STOC05*, pages 74–83, 2005.
7. Nikhil R. Devanur, Christos H. Papadimitriou, Amin Saberi, and Vijay V. Vazirani. Market equilibrium via a primal–dual algorithm for a convex program. *J. ACM*, 55(5):22:1–22:18, November 2008.
8. Nikhil R. Devanur and Vijay V. Vazirani. An improved approximation scheme for computing Arrow-Debreu prices for the linear case. In *FSTTCS*, pages 149–155, 2003.
9. E. Eisenberg and D. Gale. *Consensus of Subjective Probabilities: the Pari-mutuel Method*. Defense Technical Information Center, 1958.
10. Joachim Von Zur Gathen and Jurgen Gerhard. *Modern Computer Algebra*. Cambridge University Press, New York, NY, USA, 2 edition, 2003.
11. Mehdi Ghiyasvand and James B. Orlin. A simple approximation algorithm for computing Arrow-Debreu prices. *To appear in Operations Research*, 2012.
12. Kamal Jain. A polynomial time algorithm for computing an Arrow-Debreu market equilibrium for linear utilities. *SIAM J. Comput.*, 37(1):303–318, April 2007.
13. Kamal Jain, Mohammad Mahdian, and Amin Saberi. Approximating market equilibria, 2003.
14. James B. Orlin. Improved algorithms for computing Fisher's market clearing prices. In *Proceedings of the 42nd ACM symposium on Theory of computing*, STOC '10, pages 291–300, New York, NY, USA, 2010. ACM.
15. Leon Walras. *Elements of Pure Economics, or the theory of social wealth*. 1874.
16. Yinyu Ye. A path to the Arrow-Debreu competitive market equilibrium. *Math. Program.*, 111(1):315–348, June 2007.