Fisher Equilibrium Price with a Class of Concave Utility Functions

Ning Chen[1], Xiaotie Deng[2]*, Xiaoming Sun[3]**, and Andrew Chi-Chih Yao[4]***

[1] Department of Computer Science, Fudan University, China
nchen@fudan.edu.cn
[2] Department of Computer Science, City University of Hong Kong
csdeng@cityu.edu.hk
[3] Department of Computer Science and Technology, Tsinghua University, China
sun_xm97@mails.tsinghua.edu.cn
[4] Department of Computer Science, Princeton University
yao@cs.princeton.edu

Abstract. In this paper we study efficient algorithms for computing equilibrium price in the Fisher model for a class of nonlinear concave utility functions, the logarithmic utility functions. We derive a duality relation between buyers and sellers under such utility functions, and use it to design a polynomial time algorithm for calculating equilibrium price, for the special case when either the number of sellers or the number of buyers is bounded by a constant.

1 Introduction

Equilibrium price is a vital notion in classical economic theory, and has been a central issue in computational economics. For the Fisher model, there are n buyers each with an initially endowed amount of cash and with a non-decreasing concave utility function, and there are m goods (w.l.o.g., each of a unit amount) for sale. At the equilibrium price, all goods are sold, all cash are spent, and the goods purchased by each buyer maximizes its utility function for the equilibrium price vector as constrained by its initial endowment. It can be viewed as a special case of the more widely known model of Arrow-Debreu.

Arrow and Debreu [1] proved the existence of equilibrium price assuming goods are divisible. Their proof was based on the fixed point theorem and thus was not constructive. Gale made an extensive study for linear utility functions [7]. On the algorithmic side, Scarf's pioneering work [11] pointed to the possibility

* Research work presented in this work is partially supported by an SRG grant of City University of Hong Kong (No. 7001514) and a research grant (CityU 1081/02E) from Hong Kong Research Grants Council, China.

** Research is supported by research grants (60223004, 60321002) of National Natural Science Foundation of China.

*** The research of this author was supported in part by US National Science Foundation under grant CCR-0310466

S. Albers and T. Radzik (Eds.): ESA 2004, LNCS 3221, pp. 169–179, 2004.

of obtaining the equilibrium price in the limit through an iterative combinatorial algorithm. However, after Scarf's work the algorithmic study in computation of equilibrium price has for a long time seen no significant progress.

Recently, Deng, Papadimitriou, and Safra [3] started a study of algorithmic complexity for the equilibrium price problem in the *Arrow-Debreu model.* Complexity and algorithmic results for the equilibrium price are obtained. In addition, an algorithmic concept of approximate equilibrium was introduced that led to a polynomial time approximation scheme for the equilibrium price of an economy with a bounded number of indivisible goods. Even though the work was presented for linear utility functions, it was pointed out many of the results hold for general concave utility functions. Still, a crucial open question remains: is there a polynomial time algorithm for the equilibrium price when the number of goods and agents are part of the input size?

Subsequently, a fair number of explorative works have followed for equilibrium and approximate equilibrium price under linear utility functions to study the problem with general input size. Devanur, et al. [4], obtained a polynomial time algorithm for a price equilibrium for the *Fisher model*, in which agents are initialed with certain amount of money. Later, on the basis of the duality type of algorithm proposed in [4], Jain et al. [8], Devanur and Vazirani [5], and Garg and Kapoor [9] showed different polynomial time approximation schemes for the linear utility functions in the Fisher model. Unfortunately, those results were for the linear utility functions, which still leaves open a gap towards the general concave utility functions.

In this paper, we develop a solution for a class of concave utility functions, the logarithmic utility functions, by making use of an interesting duality relationship between the amount of cash of buyers and the amount of commodities of the sellers. Polynomial time algorithms are obtained when the number of sellers (or buyers) is bounded by a constant.

In Section 2, we introduce a formal formulation of the problem. In Section 3, we derive and discuss a duality relationship between buyers and sellers. In Section 4, we present our polynomial time algorithms. We conclude our work with remarks and discussion in Section 5.

2 A Fisher Equilibrium Problem

We study a market consisting of a set A of n buyers and a set B of m divisible goods (*i.e.*, $|A| = n$ and $|B| = m$). Let e_i be the initial amount of endowment of buyer i, q_j be the quantity of goods j. Though it is common to scale quantities of goods to the units, for clarity we keep the notation q_j's for the goods, especially because of the duality relationship we are going to obtain.

We consider a class of logarithmic utility functions:

$$u_i(x_{i,1}, \dots, x_{i,m}) = \sum_{j=1}^{m} a_{i,j} \ln(x_{i,j}/\epsilon_{i,j} + 1),$$

for each buyer i, where $a_{i,j}, \varepsilon_{i,j} > 0$ are given constants. For each i, j, define the functions $u_{i,j}^{(1)}(t) = \frac{a_{i,j}}{t+\epsilon_{i,j}}$. Clearly, the first order derivative of the utility function u_i with respect to goods j depends only on one variable $x_{i,j}$, and is in fact equal to $u_{i,j}^{(1)}(x_{i,j})$.

The functions $u_{i,j}^{(1)}(t)$ are non-negative and strictly decreasing in t (for $t > -\epsilon_{i,j}$). It represents the marginal utility of buyer i when the amount of j held by i is t, $0 \le t \le q_j$. In comparison, linear utility functions correspond to the case the marginal utility functions, $u_{i,j}^{(1)}(t)$, are all constant.

Given a price vector $P = (p_1, \dots, p_m) > 0$, each buyer i has a unique *optimal allocation* $\left(x_{i,1}^*(P), \dots, x_{i,m}^*(P)\right)$ (see, e.g., [10]), that maximizes the utility function u_i of buyer i:

$$\begin{aligned} \max \quad & u_i(x_{i,1}, \dots, x_{i,m}) \\ s.t. \quad & p_1 x_{i,1} + \cdots + p_m x_{i,m} = e_i \\ & x_{i,j} \ge 0, \quad \forall\, 1 \le j \le m \end{aligned} \tag{1}$$

The *Fisher equilibrium* problem is to find a price vector $P = (p_1, \dots, p_m)$ (and the corresponding optimal allocations $x^*(P)$) such that all money are spent and all goods are *cleared*, *i.e.*,

$$\begin{cases} \forall\, i \in A,\ \sum_{j \in B} p_j x_{i,j}^*(P) = e_i \\ \forall\, j \in B,\ \sum_{i \in A} x_{i,j}^*(P) = q_j \end{cases}.$$

Such a price vector P is called a *(market) equilibrium price vector.*

We relax the last constraint of (1) to obtain a relaxed linear program. For each buyer i,

$$\begin{aligned} \max \quad & u_i(x_{i,1}, \dots, x_{i,m}) \\ s.t. \quad & p_1 x_{i,1} + \cdots + p_m x_{i,m} = e_i, x_{i,j} > -\epsilon_{i,j} \text{ for all } j. \end{aligned} \tag{2}$$

Note that the value of utility u_i approaches $-\infty$, as any $x_{i,j}$ approaches $-\epsilon_{i,j}$. Furthermore, the feasible region of $(x_{i,1}, x_{i,2}, \dots, x_{i,m})$ in R^m is bounded. It is clear that the maximum of u_i is achieved in some interior point of the region, and can be found by applying the Lagrange Multiplier Method. Let

$$F(x_{i,1}, \dots, x_{i,m}, \eta) = u_i(x_{i,1}, \dots, x_{i,m}) - \eta(p_1 x_{i,1} + \cdots + p_m x_{i,m} - e_i).$$

Setting the derivatives to zero, we obtain

$$\begin{aligned} F_\eta = e_i - (p_1 x_{i,1} + \cdots + p_m x_{i,m}) \ &= \ 0 \\ F_{x_{i,j}} = \frac{a_{i,j}}{x_{i,j} + \epsilon_{i,j}} - \eta p_m \ &= \ 0, j = 1, 2, \dots, m. \end{aligned}$$

The solution $(x'_{i,1}(P), \ldots, x'_{i,m}(P), \frac{1}{\eta})$ is then:

$$\frac{1}{\eta} = \frac{e_i + \sum_{j=1}^{m} \epsilon_{i,j} p_j}{\sum_{j=1}^{m} a_{i,j}}$$
$$x'_{i,j} = \frac{a_{i,j}}{p_j} * \frac{e_i + \sum_{j=1}^{m} \epsilon_{i,j} p_j}{\sum_{j=1}^{m} a_{i,j}} - \epsilon_{i,j}, j = 1, 2, \ldots, m.$$

Note that $x'_{i,j} > -\epsilon_{i,j}$, and thus $(x'_{i,1}(P), \ldots, x'_{i,m}(P))$ is indeed an interior point within the domain within which the utility function u_i is defined (as a real-valued function).

Recall that we use $(x^*_{i,1}(P), \ldots, x^*_{i,m}(P))$ to denote the optimal solution of (1). In addition, in case of no ambiguity, we drop P in the notation of x' and x^*.

Proposition 1. *For any i, j, if $x'_{i,j} \le 0$, then $x^*_{i,j} = 0$.*

Proof. Let i be fixed. Assume to the contrary that there exists $j_1 \in B$ such that $x'_{i,j_1} \le 0$ and $x^*_{i,j_1} > 0$. Because of the budget constraint, there must exist $j_2 \in B$ such that $x'_{i,j_2} > x^*_{i,j_2} \ge 0$. In the following we focus only on the local allocation to j_1 and j_2.

First observe that one unit of j_1 is equivalent to p_{j_1}/p_{j_2} unit of j_2 in price. By definition of optimality, it would not increase the utility if x'_{i,j_2} is decreased with a corresponding increase of x'_{i,j_1}. Therefore, the marginal utility gain by purchasing j_2 with one unit money must be no less than that of purchasing j_1:

$$\frac{p_{j_1}}{p_{j_2}} \cdot u^{(1)}_{i,j_2}(x'_{i,j_2}) \ge u^{(1)}_{i,j_1}(x'_{i,j_1}).$$

Otherwise, we may decrease x'_{i,j_2} (and increase x'_{i,j_1} accordingly) to get higher utility for buyer i.

Using the last inequality and the monotonicity of $u^{(1)}_{i,j}$, we obtain

$$\frac{p_{j_1}}{p_{j_2}} \cdot u^{(1)}_{i,j_2}(x^*_{i,j_2}) > \frac{p_{j_1}}{p_{j_2}} \cdot u^{(1)}_{i,j_2}(x'_{i,j_2}) \ge u^{(1)}_{i,j_1}(x'_{i,j_1}) > u^{(1)}_{i,j_1}(x^*_{i,j_1}).$$

The inequality between the first and the last term implies that an increase in x^*_{i,j_2} (and a corresponding decrease in x^*_{i,j_1}) will gain buyer i a higher utility, which is a contradiction to the optimality of $x^*(P)$. □

3 Dual Relation Between Buyers and Sellers

For the sake of presentation, we associate a seller with each goods, following [4]. We introduce a bipartite graph to describe the transactions between buyers and sellers.

Definition 1. *Give a price vector P, the* transaction graph $G(P) = (A, B, E)$ *consists of vertices A, B, which represent the collection of buyers and sellers, respectively, and the edge set E. For any $i \in A$, $j \in B$, $(i, j) \in E$ if and only if $x^*_{i,j}(P) > 0$.*

For $i \in A$, denote its neighborhood by $\Gamma(i) = \{j \in B \mid (i,j) \in E\}$. For $j \in B$, denote its neighborhood by $\Phi(j) = \{i \in A \mid (i,j) \in E\}$.

It follows immediately from Proposition 1 that, for any price vector, we can compute the incident edge set to i, $\Gamma(i)$, as follows: First compute the optimal solution x' of (2). Then, drop the non-positive variables $x'_{i,j}$ in the solution. For the remaining variables, repeat the above procedure until all solutions are positive. The remaining variables (with positive solutions) yield the corresponding neighborhood set $\Gamma(i)$. We summarize in the following proposition.

Proposition 2. *If $x_{i,j} > 0$ in the last iteration, then $x^*_{i,j} > 0$.*

Note that in each iteration, at least one variable is dropped. Thus there can be at most m iterations. Hence, for any given price vector the above algorithm computes $\Gamma(i)$ in $O(m^3)$ time.

Now, for any given graph G, we define for each buyer i the "bias factor" λ_i as follows:

$$\lambda_i = \frac{e_i + \sum_{j \in \Gamma(i)} \epsilon_{i,j} p_j}{\sum_{j \in \Gamma(i)} a_{i,j}} \tag{3}$$

Note that in the last iteration of the algorithm to determine the transaction graph, we have $\eta = 1/\lambda_i > 0$, and for any $j \in \Gamma(i)$, $x_{i,j} = \frac{a_{i,j}}{\eta p_j} - \epsilon_{i,j} = \frac{a_{i,j}}{p_j}\lambda_i - \epsilon_{i,j} > 0$.

At any time during the iteration, define the current value of $1/\eta$ as the value of

$$\frac{e_i + \sum_{j \in \Gamma(i)} \epsilon_{i,j} p_j}{\sum_{j \in \Gamma(i)} a_{i,j}},$$

where $\Gamma(i)$ is the set of remaining vertices in B at that time. We argue that the value of η can only increase in the iterative process. Note that, after each iteration except the last one, some variables $x'_{i,j}$ become non-positive and thus j is dropped from $\Gamma(i)$. That means the new value of η must have become larger to decrease $\frac{a_{i,j}}{\eta p_j} - \epsilon_{i,j}$ from being positive to zero or negative.

For any $j \notin \Gamma(i)$, when j is dropped from $\Gamma(i)$ during an iteration, $\frac{a_{i,j}}{\eta p_j} - \epsilon_{i,j} \leq 0$ for the value of η at that time. Since the value of η can only increase (reaching $1/\lambda_i$ at the end), we conclude that $\frac{a_{i,j}}{p_j}\lambda_i - \epsilon_{i,j} \leq 0$. The above discussions lead to the following characterization of the transaction graph.

Lemma 1. *Given a price vector $P > 0$, the transaction graph $G(P)$ satisfies the following conditions:*

(a) $\forall j \in \Gamma(i)$, $\lambda_i > \frac{\epsilon_{i,j}}{a_{i,j}} p_j$, and $\forall j \notin \Gamma(i)$, $\lambda_i \leq \frac{\epsilon_{i,j}}{a_{i,j}} p_j$. i.e. $\Gamma(i) = \{j \in B \mid \lambda_i > \frac{\epsilon_{i,j}}{a_{i,j}} p_j\}$.

(b) $\forall j \in \Gamma(i)$, $x^*_{i,j}(P) = \frac{a_{i,j}}{p_j}\lambda_i - \epsilon_{i,j}$.

Moreover, of all the subsets of A, $\Gamma(i)$ is the only subset that can satisfy Equation (3) and Condition (a) .

Proof. Conditions (a) and (b) follow from the above discussion. We only need to prove the uniqueness of $\Gamma(i)$. Assume without loss of generality that $\frac{\epsilon_{i,1}}{a_{i,1}}p_1 \le \cdots \le \frac{\epsilon_{i,m}}{a_{i,m}}p_m$. Suppose there is another subset $\Gamma'(i) \subseteq A$ satisfying Equation (3) and Condition (a). Assume $\Gamma(i) = \{1, \ldots, j\}$ and w.l.o.g. assume $\Gamma'(i) = \{1, \ldots, j, \ldots, j+k\}$, $k > 0$ (Discussion for the case $k < 0$ is symmetric.). Let

$$\lambda_i = \frac{e_i + \sum_{j\in\Gamma(i)} \epsilon_{i,j}p_j}{\sum_{j\in\Gamma(i)} a_{i,j}}, \qquad \lambda_i' = \frac{e_i + \sum_{j\in\Gamma'(i)} \epsilon_{i,j}p_j}{\sum_{j\in\Gamma'(i)} a_{i,j}}.$$

Thus,

$$\frac{\epsilon_{i,j}}{a_{i,j}}p_j < \lambda_i \le \frac{\epsilon_{i,j+1}}{a_{i,j+1}}p_{j+1}, \qquad \frac{\epsilon_{i,j+k}}{a_{i,j+k}}p_{j+k} < \lambda_i' \le \frac{\epsilon_{i,j+k+1}}{a_{i,j+k+1}}p_{j+k+1}.$$

Note that $\lambda_i \le \frac{\epsilon_{i,j+1}}{a_{i,j+1}}p_{j+1}$ indicates that

$$\frac{e_i + \sum_{j\in\Gamma(i)\cup\{j+1\}} \epsilon_{i,j}p_j}{\sum_{j\in\Gamma(i)\cup\{j+1\}} a_{i,j}} \le \frac{\epsilon_{i,j+1}}{a_{i,j+1}}p_{j+1}$$

Since $\frac{\epsilon_{i,j+1}}{a_{i,j+1}}p_{j+1} \le \frac{\epsilon_{i,j+2}}{a_{i,j+2}}p_{j+2} \le \cdots \le \frac{\epsilon_{i,j+k}}{a_{i,j+k}}p_{j+k}$, we have

$$\frac{e_i + \sum_{j\in\Gamma(i)\cup\{j+1,\ldots,j+k\}} \epsilon_{i,j}p_j}{\sum_{j\in\Gamma(i)\cup\{j+1,\ldots,j+k\}} a_{i,j}} \le \frac{\epsilon_{i,j+k}}{a_{i,j+k}}p_{j+k}.$$

The left hand side of the above inequality is λ_i', a contradiction to $\frac{\epsilon_{i,j+k}}{a_{i,j+k}}p_{j+k} < \lambda_i'$. □

Lemma 1 leads to an improved algorithm to compute λ_i and the transaction graph $G(P)$ for any given price vector P.

Lemma 2. *For any given price vector $P > 0$ and buyer i, λ_i and $\Gamma(i)$ can be computed in $O(m \log m)$ time. Therefore, we can compute G(P) in time $O(nm\log(m))$.*

Proof. By Lemma 1, we know that $\forall j \in \Gamma(i)$, $\lambda_i > \frac{\epsilon_{i,j}}{a_{i,j}}p_j$, and $\forall j \notin \Gamma(i)$, $\lambda_i \le \frac{\epsilon_{i,j}}{a_{i,j}}p_j$. Thus we design an algorithm as follows.

First, we calculate all $\frac{\epsilon_{i,j}}{a_{i,j}}p_j$, $j = 1, \ldots, m$, and sort them into non-decreasing order. Thus, the set $\Gamma(i)$ has at most m different choices. For each candidate set of $\Gamma(i)$, calculate the current "bias factor" λ_i according to Equation (3), and check whether $\Gamma(i)$ is equal to the set $\{j \in B \mid \lambda_i > \frac{\epsilon_{i,j}}{a_{i,j}}p_j\}$. The computation of λ_i can be done in $O(1)$ steps per candidate set, and thus the dominating computation time is in the sorting of $\frac{\epsilon_{i,j}}{a_{i,j}}p_j$, $j = 1, 2, ..., m$, and the Lemma follows. □

From now on in this section, we restrict our discussion to the equilibrium price vector.

Lemma 3. *For the equilibrium price vector P, we have*

$$p_j = \frac{\sum_{i \in \Phi(j)} a_{i,j} \lambda_i}{q_j + \sum_{i \in \Phi(j)} \epsilon_{i,j}} \tag{4}$$

and $\Phi(j) = \{i \in A \mid \frac{a_{i,j}}{\epsilon_{i,j}} \lambda_i > p_j\}$.

Proof. By Lemma 1, we have $(i,j) \in E \Leftrightarrow a_{i,j}\lambda_i > \epsilon_{i,j} p_j$, thus $\Phi(j) = \{i \in A \mid \frac{a_{i,j}}{\epsilon_{i,j}} \lambda_i > p_j\}$. And also from Lemma 1, $\forall (i,j) \in E$, $x_{i,j} = \frac{a_{i,j}}{p_j} \lambda_i - \epsilon_{i,j}$. According to the market clearance condition, $q_j = \sum_{i \in \Phi(j)} x_{i,j}$. Thus $q_j = \sum_{i \in \Phi(j)} (\frac{a_{i,j}}{p_j} \lambda_i - \epsilon_{i,j}) \Rightarrow p_j = \frac{\sum_{i \in \Phi(j)} a_{i,j} \lambda_i}{q_j + \sum_{i \in \Phi(j)} \epsilon_{i,j}}$. □

Lemma 3 presents a description of the optimal behavior of sellers. It allows use to obtain an algorithm to calculate p_j and $\Phi(j)$ on the basis of Λ, where $\Lambda = (\lambda_1, \ldots, \lambda_n)$.

Lemma 4. *Assume we know the vector Λ, then there exists the unique solution p_j and $\Phi(j)$ for seller j, and it can be computed in $O(n \log n)$ time.*

Proof. By Lemma 3, $\Phi(j) = \{i \in A \mid \frac{a_{i,j}}{\epsilon_{i,j}} \lambda_i > p_j\}$. We first sort all $\frac{a_{i,j}}{\epsilon_{i,j}} \lambda_i$, $i = 1, \ldots, n$, in a non-increasing order. Then $\Phi(j)$ has at most n choices. For each candidate of $\Phi(j)$, calculate p_j according to Equation (4), where the sums can be obtained by preprocessing the prefix sum series in $O(m)$ time. $\Phi(j)$ is then determined by whether it is equal to $\{i \in A \mid \frac{a_{i,j}}{\epsilon_{i,j}} \lambda_i > p_j\}$. □

A dual relation is now established between the price vector P and the "bias factor" Λ:

$$P \leftrightarrow \Lambda$$

$$p_j = \frac{\sum_{i \in \Phi(j)} a_{i,j} \lambda_i}{q_j + \sum_{i \in \Phi(j)} \epsilon_{i,j}} \leftrightarrow \lambda_i = \frac{e_i + \sum_{j \in \Gamma(i)} \epsilon_{i,j} p_j}{\sum_{j \in \Gamma(i)} a_{i,j}}$$

$$\Phi(j) = \{i \in A \mid \frac{a_{i,j}}{\epsilon_{i,j}} \lambda_i > p_j\} \leftrightarrow \Gamma(i) = \{j \in B \mid \lambda_i > \frac{\epsilon_{i,j}}{a_{i,j}} p_j\}$$

Although the "bias factor" was first defined by the price vector (Equation (3)), we can treat it as a natural parameter for buyers as a dual to the price vector P for sellers: Each seller j decides which collection of buyers to sell his goods, according to the quantity q_j and the buyers' "bias factor" Λ (dually, each buyer i decides which collection of goods to buy, according to his total endowment e_i and the sellers' price P). This property will be used in the next section to design an algorithm for finding the equilibrium solution.

4 Solution for Bounded Number of Buyers or Sellers

Our solution depends on the following lemma.

Lemma 5. *Given a bipartite graph $G(A,B)$, if for all $i \in A, j \in B$, $\Gamma(i) \neq \emptyset$, $\Phi(j) \neq \emptyset$, then the linear equation system*

$$\begin{cases} \lambda_i = \frac{e_i + \sum_{j \in \Gamma(i)} \epsilon_{i,j} p_j}{\sum_{j \in \Gamma(i)} a_{i,j}}, \; i = 1, \dots, n \\ p_j = \frac{\sum_{i \in \Phi(j)} a_{i,j} \lambda_i}{q_j + \sum_{i \in \Phi(j)} \epsilon_{i,j}}, \quad j = 1, \dots, m \end{cases} \tag{5}$$

is non-degenerate.

Proof. From the equations, we have

$$\begin{aligned} \lambda_i &= \frac{e_i + \sum_{j \in \Gamma(i)} \epsilon_{i,j} p_j}{\sum_{j \in \Gamma(i)} a_{i,j}} \\ &= \frac{e_i}{\sum_{l \in \Gamma(i)} a_{i,l}} + \frac{1}{\sum_{l \in \Gamma(i)} a_{i,l}} \sum_{j \in \Gamma(i)} \epsilon_{i,j} \frac{\sum_{k \in \Phi(j)} a_{k,j} \lambda_k}{q_j + \sum_{k \in \Phi(j)} \epsilon_{k,j}} \\ &= \frac{e_i}{\sum_{l \in \Gamma(i)} a_{i,l}} + \frac{1}{\sum_{l \in \Gamma(i)} a_{i,l}} \sum_{j \in \Gamma(i)} \left(\sum_{k \in \Phi(j)} \frac{a_{k,j} \epsilon_{i,j} \lambda_k}{q_j + \sum_{l \in \Phi(j)} \epsilon_{l,j}} \right) \end{aligned}$$

Now we change the order of dummy j,k is the right side of the equation:

$$\begin{aligned} \lambda_i &= \frac{e_i}{\sum_{l \in \Gamma(i)} a_{i,l}} + \frac{1}{\sum_{l \in \Gamma(i)} a_{i,l}} \sum_{k : \Gamma(i) \cap \Gamma(k) \neq \emptyset} \left(\sum_{j : j \in \Gamma(i) \cap \Gamma(k)} \frac{a_{k,j} \epsilon_{i,j} \lambda_k}{q_j + \sum_{l \in \Phi(j)} \epsilon_{l,j}} \right) \\ &= \frac{e_i}{\sum_{l \in \Gamma(i)} a_{i,l}} + \frac{1}{\sum_{l \in \Gamma(i)} a_{i,l}} \sum_{k : \Gamma(i) \cap \Gamma(k) \neq \emptyset} \lambda_k \left(\sum_{j : j \in \Gamma(i) \cap \Gamma(k)} \frac{a_{k,j} \epsilon_{i,j}}{q_j + \sum_{l \in \Phi(j)} \epsilon_{l,j}} \right) \end{aligned}$$

i.e.

$$\left(\sum_{j \in \Gamma(i)} a_{i,j} \right) \lambda_i = e_i + \sum_{k : \Gamma(i) \cap \Gamma(k) \neq \emptyset} \lambda_k \left(\sum_{j : j \in \Gamma(i) \cap \Gamma(k)} \frac{a_{k,j} \epsilon_{i,j}}{q_j + \sum_{l \in \Phi(j)} \epsilon_{l,j}} \right)$$

Move λ_k to the left, we get an equation system of λ: $F\lambda = e$, here $F = [f_{i,k}]_{m \times m}$ is a $m \times m$ matrix, $e = [e_1, \dots, e_m]^T$, where

$$\begin{cases} f_{i,i} = \left(\sum_{j \in \Gamma(i)} a_{i,j} \right) - \sum_{j : j \in \Gamma(i)} \frac{a_{i,j} \epsilon_{i,j}}{q_j + \sum_{l \in \Phi(j)} \epsilon_{l,j}}, & \\ f_{i,k} = - \sum_{j : j \in \Gamma(i) \cap \Gamma(k)} \frac{a_{k,j} \epsilon_{i,j}}{q_j + \sum_{l \in \Phi(j)} \epsilon_{l,j}}, & \text{if } \Gamma(i) \cap \Gamma(k) \neq \emptyset, k \neq i \\ f_{i,k} = 0, & \text{if } \Gamma(i) \cap \Gamma(k) = \emptyset \end{cases}$$

Now we show that $\forall k,\ \sum_{i:i\neq k} |f_{i,k}| < f_{k,k}$: In fact,

$$\begin{aligned}
& f_{k,k} - \sum_{i:i\neq k} |f_{i,k}| \\
&= \left(\sum_{j\in\Gamma(k)} a_{k,j}\right) - \sum_{i:\Gamma(i)\cap\Gamma(k)\neq\emptyset} \left(\sum_{j:j\in\Gamma(i)\cap\Gamma(k)} \frac{a_{k,j}\epsilon_{i,j}}{q_j + \sum_{l\in\Phi(j)} \epsilon_{l,j}}\right) \\
&= \left(\sum_{j\in\Gamma(k)} a_{k,j}\right) - \sum_{j:j\in\Gamma(k)} \left(\sum_{i:i\in\Phi(j)} \frac{a_{k,j}\epsilon_{i,j}}{q_j + \sum_{l\in\Phi(j)} \epsilon_{l,j}}\right) \\
&= \left(\sum_{j\in\Gamma(k)} a_{k,j}\right) - \sum_{j:j\in\Gamma(k)} \left(\frac{a_{k,j}}{q_j + \sum_{l\in\Phi(j)} \epsilon_{l,j}} \sum_{i:i\in\Phi(j)} \epsilon_{i,j}\right) \\
&= \sum_{j\in\Gamma(k)} a_{k,j} \left(1 - \frac{\sum_{i:i\in\Phi(j)} \epsilon_{i,j}}{q_j + \sum_{l\in\Phi(j)} \epsilon_{l,j}}\right) \\
&= \sum_{j\in\Gamma(k)} \frac{a_{k,j} b_j}{q_j + \sum_{l\in\Phi(j)} \epsilon_{l,j}} > 0
\end{aligned}$$

So $rank(F) = m$, the linear equation system is non-degenerate. □

Proposition 3. *Given the transaction graph G, P and Λ can be computed in $O((m+n)^3)$ time.*

Proof. We establish $(m+n)$ linear equations of Λ and P from graph G:

$$\begin{cases} \lambda_i = \frac{e_i + \sum_{j\in\Gamma(i)} \epsilon_{i,j} p_j}{\sum_{j\in\Gamma(i)} a_{i,j}}, & i = 1, \dots, n \\ p_j = \frac{\sum_{i\in\Phi(j)} a_{i,j}\lambda_i}{q_j + \sum_{i\in\Phi(j)} \epsilon_{i,j}}, & j = 1, \dots, m \end{cases}$$

By Lemma 5, the set of equations is non-degenerate for e, q corresponding to the equilibrium solution. It can be computed in $O((m+n)^3)$ time. □

The number of bipartite graphs with m and n vertices on each side has in general 2^{mn} different choices, which is exponential for both m and n. In the following, we show how to reduce this number to polynomial, when either m or n is bounded by a constant.

Lemma 6. *There is an algorithm solving the Market equilibrium problem with logarithmic utility functions in $O((m + 2^m mn)^{2m+1}(m+n)^3)$ time.*

Proof. Due to Proposition 3, we only need to prove there are at most $O((m + 2^m mn)^m)$ different possibilities of graph G, or equivalently, different choices of $(\Gamma(1), \dots, \Gamma(n))$, and that it takes $O((m + 2^m mn)^{2m+1})$ time to generate all these graphs.

For each $i \in A$ and subset $\Delta \subseteq B$, let $\mathcal{I}_{i,\Delta}$ denote the system of m inequalities

$$\begin{cases} \frac{e_i+\sum_{k\in\Delta}\epsilon_{i,k}p_k}{\sum_{k\in\Delta}a_{i,k}} > \frac{\epsilon_{i,j}}{a_{i,j}}p_j, \forall j \in \Delta \\ \frac{e_i+\sum_{k\in\Delta}\epsilon_{i,k}p_k}{\sum_{k\in\Delta}a_{i,k}} \leq \frac{\epsilon_{i,j}}{a_{i,j}}p_j, \forall j \notin \Delta \end{cases} \tag{6}$$

If a $(\Gamma(1), \Gamma(2), \ldots, \Gamma(n))$ (where $\Gamma(i) \subseteq B$) is a possible candidate for the transaction graph, then by Lemma 3.1, there must exist some $P \in \mathbb{R}_+^m$ such that $\mathcal{I}_{i,\Gamma(i)}$ holds for every $i \in A$.

For each $i \in A$, subset $\Delta \subseteq B$ and $j \in \Delta$, let $f_{i,\Delta,j}(p_1, p_2, \ldots, p_m)$ denote the following linear function in variables $p_1, \ldots, p_m$:

$$\frac{e_i + \sum_{k\in\Delta}\epsilon_{i,k}p_k}{\sum_{k\in\Delta}a_{i,k}} - \frac{\epsilon_{i,j}}{a_{i,j}}p_j.$$

Let N be the number of all triplets (i, Δ, j), then $N \leq nm2^m$. Let $P \in \mathbb{R}_+^m$ be an unknown price vector, but that we know the answer $\xi(P) \in \{>, \leq\}^N$ to all the binary comparison queries "$f_{i,\Delta,j}(P) : 0$". Then for each i we can easily determine all the subsets $\Delta \subseteq B$ such that the system of inequalities $\mathcal{I}_{i,\Delta}$ holds. By Lemma 3.1, there is in fact exactly one such Δ, which we call $\Gamma(i)$. In this fashion, each possible $\xi(P)$ leads to one candidate graph $G(P)$, as specified by $(\Gamma(1), \ldots, \Gamma(n))$; this computation can be done in time $O(N)$.

To prove the Lemma, it suffices to show that there are at most $O(N^m)$ distinct $\xi(P)$'s, and that they can be generated in time $O(N^{2m+1})$. In $\mathbb{R}^m$, consider the $N+m$ hyperplanes $f_{i,\Delta,j}(p_1, p_2, \ldots, p_m) = 0$ and $p_i = 0$. They divide the space into regions depending on answers to the queries "$f_{i,\Delta,j}(p_1, p_2, \ldots, p_m) > 0$?" and "$p_i > 0$?". For any region in $\mathbb{R}_+^m$, all the points P in it share the same $\xi(P)$. It is well known that there are at most $O((N+m)^m)$ regions. We can easily enumerate all these regions and compute their $\xi(P)$'s in time $O((N+m)^m(N+m)^{m+1})$, by adding hyperplanes one at a time and updating the regions using linear programming in $\mathbb{R}^m$. □

Lemma 6 solves the Market equilibrium problem in the price space $\mathbb{R}^m$. By the dual relationship, we can also solve the problem in the "bias factor" space $\mathbb{R}^n$ in an almost verbatim manner.

Lemma 7. *There is an algorithm solving the Market equilibrium problem with logarithmic utility functions in $O((m + 2^n mn)^{2m+1}(m+n)^3)$ time.*

From Lemma 6 and Lemma 7, we have the following conclusion.

Theorem 1. *If the number of buyers, or goods, is bounded by a constant, then there is a polynomial time algorithm solving the Market equilibrium problem with logarithmic utility functions.*

5 Conclusion and Discussion

In this paper, we have derived a duality relation for the market equilibrium problem with logarithmic utility functions, and use it to develop a polynomial time algorithm when the number of either the buyers or the sellers is bounded by a constant. The techniques developed may be of some help for general concave utility functions.

We note that, recently, Devanur and Vazirani made some extension to design a PTAS for another special type of value functions for the buyers as they considered the concave utility function very difficult to deal with by their approach [6]. At the same time, Codenotti and Varadarajan [2] studied how to compute equilibrium price for Leontief utility function.

References

1. K. K. Arrow, G. Debreu, *Existence of An Equilibrium for a Competitive Economy*, Econometrica, V. 22, 265-290, 1954.
2. B. Codenotti, K. Varadarajan, *Efficient Computation of Equilibrium Prices for Market with Leontief Utilities*, To appear in ICALP 2004.
3. Xiaotie Deng, C. H. Papadimitriou, S. Safra, *On the Complexity of Equilibria*, STOC 2002, 67-71. Journal version: Xiaotie Deng, C. H. Papadimitriou, S. Safra, *On the Complexity of Price Equilibrium*, Journal of Computer and System Sciences, V. 67(2), 311-324, 2003.
4. N. Devanur, C. H. Papadimitriou, A. Saberi, V. V. Vazirani, *Market Equilibrium via a Primal-Dual-Type Algorithm*, FOCS 2002, 389-395.
5. N. Devanur, V. Vazirani, *An Improved Approximation Scheme for Computing Arrow-Debreu Prices for the Linear Case*, FSTTCS 2003, 149-155.
6. N. Devanur, V. Vazirani, *The Spending Constraint Model for Market Equilibrium: Algorithmic, Existence and Uniqueness Results*, STOC 2004.
7. D. Gale, *The Theory of Linear Economic Models*, McGraw Hill, N.Y., 1960.
8. K. Jain, M. Mahdian, A. Saberi, *Approximating Market Equilibria*, APPROX 2003, pp.98-108.
9. S. Kapoor, R. Garg, *Auction Algorithms for Market Equilibrium*, STOC 2004.
10. B.T. Polyak: "Introduction to Optimization", Opt. Software Inc., 1987.
11. H. E. Scarf, *The Computation of Economic Equilibria* (with collaboration of T. Hansen), Cowles Foundation Monograph No. 24, Yale University Press, 1973.