A Polynomial Time Algorithm for Computing an Arrow-Debreu Market Equilibrium for Linear Utilities

Kamal Jain
Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA
kamalj@microsoft.com

Abstract

We provide the first polynomial time exact algorithm for computing an Arrow-Debreu market equilibrium for the case of linear utilities. Our algorithm is based on solving a convex program using the ellipsoid algorithm and simultaneous diophantine approximation. As a side result, we prove that the set of assignments at equilibria is convex and the equilibria prices themselves are log-convex. Our convex program is explicit and intuitive, which allows maximizing a concave function over the set of equilibria. On the practical side, Ye developed an interior point algorithm [31] to find an equilibrium based on our convex program. We also derive separate combinatorial characterizations of equilibrium for Arrow-Debreu and Fisher cases. Our convex program can be extended for many non-linear utilities (section 8 and [4, 24]) and production models [21].

Our paper also makes a powerful theorem (theorem 6.4.1 in [19]) even more powerful (theorems 12 and 13) in the area of Geometric Algorithms and Combinatorial Optimization. The main idea in this generalization is to allow ellipsoids not to contain the whole convex region but a part of it. This theorem is of independent interest.

1. Introduction

We present the first polynomial time algorithm to compute the exact general market equilibrium for the linear utility case, thereby settling an open problem posed by Papadimitriou[28] and Deng-Papadimitriou-Safra [8]. A version of the problem was first formulated by a French economists, Leon Walras, in 1874 [30]. In this model, every person in an entire population has an initial endowment of divisible goods. Furthermore, every person in the population has a utility function for consuming goods. Every person sells the initial endowment and then buys an optimal bundle of goods with the entire revenue i.e., the market clears. Walras asked whether a price can be assigned to every good so that this is possible. Such a price vector, if exist, is called general market equilibrium. An answer was given by two Nobel Laureates Kenneth Arrow and Gerard Debreu in 1954 [2]. They showed that when the utility functions are concave then under some mild conditions a general market equilibrium always exist [2]. Their proof is non-constructive and does not suggest any polynomial time algorithm to find a general market equilibrium.

Irving Fisher was the first to consider the computability of equilibrium prices[3]. Independent of Walras work, he defined another model, which is a special case of Walras model, in 1891. In his model there are two kinds of people, producers and consumers. Consumers have money and utility functions for goods. Producers have initial endowments of goods and want to earn money. The equilibrium prices is defined as the assignment of prices to goods, so that when every consumer buys an optimal bundle then the market clears i.e., all the money is spent and all the goods are sold. If money is also considered a commodity then it is easy to see that Fisher model is a special case of Walras model, hence, Arrow-Debreu theorem [2] also implies the existence of a market equilibrium for Fisher model for the case of concave utilities.

Although Walras model and Fisher model seems very close, there are some fundamental differences between them. Walras model suggests that money has no intrinsic value except that it is a scale to measure the value of a good. Fisher model on the other hand assumes the value of the money. Fisher model can also be viewed as a generalization of the concept of demand and supply curves, where demand is a decreasing function of prices and supply is an increasing function of prices, a point where they meet is called an *equilibrium*. On the other hand Walras model allows the demand to be an increasing function of income besides a decreasing function of prices. In practice, at a macro level, when the prices increase, the demand

Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science (FOCS'04)
0272-5428/04 $20.00 © 2004 **IEEE**

decreases. But then the incomes also increase, because consumers and producers (i.e., workers in the global economy system) are the same, which has an increasing effect on demand. This feedback feature of the Walras model not only makes it more realistic but also puts it at a much higher difficulty level than the Fisher model. This is also evident by the results obtained about the Fisher model.

Eisenberg and Gale [14] gave a constructive proof using variational method for Fisher model in case of linear utilities. Their proof shows that the equilibrium assignment is the one which maximizes the product of utilities obtained by every dollar of the buyers. This gives a convex function which is minimized at the equilibrium. This implicitly implies a polynomial time (approximation) algorithm, in the sense of numerical computing i.e., polynomial in the input size and $\log(1/\epsilon)$, where ϵ is the precision in computation. The first exact algorithm is recently developed by Devanur et. al. [10].None of these algorithms is strongly polynomial time.

There was no polynomial time algorithm known for the Walras model, though, the problem was studied extensively and many heuristics are developed using Lemke method, Newton method, primal-dual method, convex programming and various other techniques in numerical optimization (A very few of them are [12, 29, 7, 15]). Deng, Papadimitriou and Safra [8] gave polynomial time algorithms for bounded number of agents or goods case and asked the question for general case. The importance of polynomial time computational results for computing equilibria was earlier mentioned in [27].

Recently two approximation schemes are developed for the Walras model, [22, 18, 11]. The first scheme [22] also works for some non-linear utility functions for those instances which satisfy the condition of gross-substitutability (i.e., if the price of some good is raised then the demand of other goods can only increase). An approximation scheme was also developed by Newman and Primak [26] by running Ellipsoid algorithm on an infinite linear program. The inequalities of the infinite linear program were given by Arrow, Block and Hurwicz [1] and hold for a very general class of utility functions satisfying gross substitutability. We refer the readers to [5] to get further details and pointers on this approach. Approximation schemes developed in [22, 18, 11] interpret the approximation factor in a physical sense. On the other hand the sense of approximation in [26] was not clear. One reason is that the linear inequalities used by [26] have both positive and negative coefficients with their signs not clear a priory. We again refer the readers to [5] for a physical interpretation of approximation in [26]. In some sense these approximation schemes [26, 22, 18, 11] are pseudo-approximation because they do not guarantee that an equilibrium is nearby.

In this paper we propose a mathematical program for the Walras model for the case of linear utilities. We show that the program is valid for non-linear utilities too. The program is simple and does not have complicated constraints like "optimality of the bundles purchased" and "money spent equals to money earned". Instead these constraints follow from the feasibility of the program. In general constraints written do not have simple economic interpretation, which probably is the reason that they were not discovered so far. For the case of linear utilities, they do have some economic interpretation and also the program turns out to be convex. The program turns out to be convex for some cases of non-linear utilities too. We show that all the general market equilibria are feasible points of this convex program and vice-versa, hence, a general market equilibria can be obtained using ellipsoid method and simultaneous diophantine approximation. This leads us to develop theorems 12 and 13, which make a powerful geometric algorithm even more powerful. These theorems are about general convex programs and not restricted to game theory.

Our convex program is explicitly given and can easily be converted into an optimization program. Using this convex program Ye [31] developed practical algorithms for computing market equilibria. Besides, computability, we get some insight into the structure of the equilibrium itself. Our convex program shows that the set of all the equilibria assignments is convex. It also shows that the set of all the equilibria prices on a logarithmic scale is convex too. Indeed, algorithms via convex programs have distinct theoretical advantage. Almost any known basic fact about Fisher model with linear utilities can be easily observed from Eisenberg-Gale's convex program. Convexity, which gives dual interpretation to prices, is one of them. Uniqueness of equilibrium utilities and prices, rationality of prices, proportionate fairness of the equilibrium [25] and combinatorial characterization (theorem 11) are some others. Eisenberg-Gale's program gives the fastest known algorithm [31] for the problem. Their convex program can be generalized for homogeneous concave utilities [13] and homogeneous quasi-concave utilities [23] too. As shown in many thought experiments in [31], their convex program is quite natural and implies a proof of Arrow-Debreu's case by using a fixed point theorem. Furthermore, this convex program allows Jain-Vazirani-Ye [23] to define equilibrium for production model with some economies of scales. Note that Arrow-Debreu theorem breaks down as soon as we have any kind of economies of scales.

Indeed most of these features are also shared by our convex program in this paper. The only main difference is that the Eisenberg-Gale's convex program gives a constructive proof of the existence of an equilibrium whereas our convex program does not. The fact that no constructive proof has been developed for Arrow-Debreu's theorem in the last half century hints toward an intrinsic hardness of the model.

IEEE COMPUTER SOCIETY

Intuitively, one reason that the use of fixed point theorem can't be done away with easily is the two sided feedback of prices - demand decreases with prices whereas income increases which in turn has positive effect on demand. We are not aware of any other problem where the only proof of existence is via fixed point theorems and yet a polynomial time algorithm is successfully developed. Existence of taxes in selfish routing in [6] was one of them. In a surprising result [16] proposed a fairly straightforward proof of this result and its various generalizations.

In other related works, [21] generalizes the results in this paper to include production planning constraints and [4] generalizes the results in this paper to include utility functions with constant elasticity of substitution (CES). In an ongoing work [24] extends the convex programs to include quasi-concave functions. We conjecture that our convex program remains convex for the utility functions satisfying the gross substitution property. One reason for this conjecture is that the set of equilibria is known to be convex for the utility functions satisfying the gross substitution property [1, 5]. The convexity is proved via an infinite linear program which may not allow efficient interior point methods. An advantage of our approach is we get explicit polynomial sized convex programs.

Finally, we would like to close this section by emphasizing the need of efficient computational results in economics of equilibria, not only for the development of Computer Science but for the development of Economics itself. If a Turing machine can't compute then an economic system can't compute either. An economic system by converging at equilibria is also computing it. If equilibria is not computable with a Turing machine then it is unlikely that an economic system will be able to compute it either. Hence, a lack of computational result decreases the applicability of the equilibrium itself. In this regard, computing Nash Equilibrium is a big open question. Even approximating it with a nontrivial factor should be a good start.

2. Related Work

3. Model

There are n people in the system. They each have some initial endowment of divisible goods. Without loss of generality (in the linear utility case) we can assume that each person has only one good which is different from the goods which other people have. Further we can assume that each person has only one unit of good. Each person also has a linear utility function. For i-th person we denote this utility function by $\sum_j u_{ij}x_{ij}$, where x_{ij} is the amount of good j consumed by i. To be able to keep full precision in digital computers we assume that each u_{ij} is an integer. Each person maximizes her utility by buying an optimal bundle of goods with the revenue made by selling her own endowment. The classical Arrow-Debreu [2] theorem says that there is a price vector, $(p_1, p_2, \cdots, p_n)$, not all equal to zero, such that the buying and selling can be done at the prices in $(p_1, p_2, \cdots, p_n)$ in such a way that the market clears. This price vector is called *general market equilibrium*.

Without loss of generality we assume that everybody likes something, that is for every i there is a j such that $u_{ij} > 0$. If somebody does not like anything then the price of her good can be anything and she can be given any bundle. Again without loss of generality we assume that every good is liked by somebody, that is for every j there is a i such that $u_{ij} > 0$. If some good is not liked by anybody then its price must be zero. So not much to discover about it. These assumptions we make only for convenience.

In the next two sections we assume that for every proper subset, S, of persons (i.e., neither empty nor everybody) there is a person, i, outside S, who likes some good posses by S, i.e., $u_{ij} > 0$ for some $i \notin S$ and some $j \in S$ (this assumption also implies our previous assumption that everybody likes something). This assumption implies that all the equilibrium prices are non-zero. If not, then consider S as the set of persons having zero priced good. Then somebody outside S will demand infinite quantity of something possessed by S. This assumption is not without loss of generality. So, we will justify this assumption after the next two sections.

4. Non-Convex Program

In this section we give a non-convex program which has all and only general market equilibria as feasible points.

$$\begin{aligned}
&\forall j: \ \sum_i x_{ij} = 1 \\
&\forall i,j: \ x_{ij} \geq 0 \\
&\forall i,j: \ \frac{u_{ij}}{p_j} \leq \frac{\sum_k u_{ik}x_{ik}}{p_i} \\
&\forall i: \ p_i > 0
\end{aligned} \tag{1}$$

The first two lines of this program says that x_{ij} are feasible assignments. The third line says that the goods purchased by i by spending p_i (which is her revenue) has the highest utility. The fourth line says that the prices are nonzero. Arrow-Debreu theorem only guarantees that at least one price is non-zero but our assumption before the section makes all the prices non-zero. It is easy to see that any general market equilibrium will satisfy all the lines of this nonconvex program. We claim that in fact the inverse is also true.

Theorem 1 *The feasible region of non-convex program 1 has all and only general market equilibria.*

0272-5428/04 $20.00 © 2004 **IEEE**
IEEE COMPUTER SOCIETY

Proof : It is clear that all the market equilibria satisfy the program. So we only need to show that any feasible point is a market equilibrium. Line 3 of the program, when multiplied with $x_{ij}p_j$ gives:

$$\forall i, j: \; u_{ij}x_{ij} \le \frac{\sum_k u_{ik}x_{ik}}{p_i} x_{ij}p_j$$

When we add these inequalities for all j we get:

$$\forall i: \; \sum_j u_{ij}x_{ij} \le \frac{\sum_k u_{ik}x_{ik}}{p_i} \sum_j x_{ij}p_j$$

Note that our assumption that everybody likes something implies that $\sum_k u_{ik}x_{ik}$ is not zero. So the inequality above after a simplification gives:

$$\forall i: \; p_i \le \sum_j x_{ij}p_j$$

When we add these inequalities for all i we get:

$$\sum_i p_i \le \sum_i \sum_j x_{ij}p_j$$

When we interchange the order of summation on the right hand side, we get:

$$\sum_i p_i \le \sum_j p_j \sum_i x_{ij}$$

Note that the second summation on the right hand side is 1, so we get:

$$\sum_i p_i \le \sum_j p_j$$

This should have been an equality. Which means that all the inequalities added to obtain this must have been equalities. This implies two facts, which we are writing as lemmas to be used in the later sections. The theorem follows from the following two lemmas. Q.E.D.

Lemma 2 *Every feasible point of the non-convex program 1 satisfies the constraint of money earned equals to money spend for every user, i.e.,*

$$\forall i: \; p_i = \sum_j x_{ij}p_j.$$

Lemma 3 *Every feasible point of the non-convex program 1 satisfies that the money of every person is spent optimally i.e., whenever $x_{ij} > 0$, the corresponding constraint on the third line of the program is tight.*

In the section of non-linear utility case, section 8, we will show that the theorem 1 still holds.

5. Solving Non-Convex Program 1

Note that the third line of non-convex program 1 is useful only when $u_{ij} > 0$. So we can rewrite the third line as:

$$\forall i, j \text{ such that } u_{ij} > 0: \; \frac{u_{ij}}{p_j} \le \frac{\sum_k u_{ik}x_{ik}}{p_i},$$

which in turn can rewritten as:

$$\forall i, j \text{ such that } u_{ij} > 0: \; \frac{p_i}{p_j} \le \frac{\sum_k u_{ik}x_{ik}}{u_{ij}}$$

Now we construct a directed graph, G, with the n persons as the set of vertices. We draw an edge from i to j when $u_{ij} > 0$ (i and j may be the same vertex in that case the edge is a loop). We assign two kinds of weight to each edge, ij. The first is denoted by w and the second is denoted by $LOGw$. For an edge from i to j, $w(ij) = \frac{\sum_k u_{ik}x_{ik}}{u_{ij}}$ and $LOGw(ij) = \log(w(ij))$. By Farkas lemma, non-convex program 1 is feasible if and only if the product of w_{ij} is at least one over any cycle of the graph, G. In other words we have the following theorem:

Theorem 4 *Non-convex program is feasible if and only if there is no negative cycle in G with respect to the weight function, $LOGw$.*

Proof : The problem of finding an equilibrium assignment is finding those x_{ij}'s for which there is a feasible solution of non-convex program 1. Suppose we have a feasible assignment of goods to people i.e., values for variable x_{ij}'s satisfying the first two sets of constraints of non-convex program 1. We want to find out whether there exist a feasible assignments of values to price variables satisfying the last two sets of constraints in non-convex program 1. Since x_{ij}'s are given they form the coefficients of a linear program whose variables correspond to prices. We can apply Farkas lemma to write conditions on the coefficients (i.e., on assignments variables) which allow a feasible solution to price variables. The best way to do is to take the logarithm of the third set of inequalities and assume that the variables are the $LOG(p)$ not p. The theorem now easily follows from Farkas lemma. Q.E.D.

This theorem is similar in flavor as the "no-negative-cycle" theorem for the classical problem of minimum cost flow. In fact, non-deterministically this has the same functionality too. It tells us when an assignment is an equilibrium assignment. A flow is minimum cost if there is no negative cycle. Similarly, an assignment of goods is an equilibrium solution if there is no negative cycle. This analogy gives a promising hope of a combinatorial algorithm for the general equilibrium problem. Later in the paper we will see that the theorem holds for the concave utilities functions too.

0272-5428/04 $20.00 © 2004 IEEE

Theorem 4 also gives us the following convex program for the general equilibrium problem, which again makes it a functional theorem for computational purpose.

$$\forall j: \sum_i x_{ij} = 1$$
$$\forall i,j: x_{ij} \geq 0$$
$$\text{For every cycle, } C \text{, of } G : \prod_{ij \in C} w(ij) \geq 1 \quad (2)$$

The separation oracle for the last set of inequalities can be derived using an algorithm for finding a negative cycle in a graph. Using the usual inequality of the arithmetic mean is at least the geometric mean, the last set of inequalities can also be converted into an infinite number of linear equalities as follows.

In the following program we denote by $\boldsymbol{\alpha}$ a vector of non-negative real numbers. The number of co-ordinates will be clear by the context. A subscripted $\boldsymbol{\alpha}$ will denote a coordinate of $\boldsymbol{\alpha}$.

$$\forall j: \sum_i x_{ij} = 1$$
$$\forall i,j: x_{ij} \geq 0$$
$$\text{For every cycle, } C \text{, of } G \text{ and for every } \boldsymbol{\alpha} :$$
$$\frac{1}{|C|}\sum_{ij \in C} \frac{w(ij)}{\boldsymbol{\alpha}_{ij}} \geq \left(\frac{1}{\prod_{ij\in C}\boldsymbol{\alpha}_{ij}}\right)^{\frac{1}{|C|}} \quad (3)$$

Lemma 5 *Convex program 2 is equivalent to linear program 3.*

Proof : Note that the third set of inequalities of the linear program follows from the third set of inequalities of the convex program using the inequality of arithmetic mean and geometric mean. So we only need to show the converse. Suppose one of the inequalities on the third line of the convex program is violated by an assignment vector $\mathbf{x}^*$ of x^*_{ij}. We show that $\mathbf{x}^*$ violates one of the inequality of the linear program too.

Suppose the inequality corresponding to a cycle C is violated, i.e., we have:

$$\prod_{ij\in C} w^*(ij) < 1,$$

where $\mathbf{w}^*$ denotes the value of the weight function at $\mathbf{x}^*$. We claim that the inequality of the linear program corresponding to the same cycle and $\boldsymbol{\alpha} = \mathbf{w}^*$ is violated too. Indeed, the left hand side is one when evaluated at $\mathbf{x}^*$ whereas the right hand side is bigger than one. This also shows that the convex program 2 is indeed convex. Q.E.D.

Corollary 6 *Ellipsoid algorithm finds a market equilibrium in polynomial time.*

Proof : Eaves [12] showed that the problem of finding a market equilibrium with linear utilities can be written as a linear complementarity program. This implies that there is a market equilibrium with rational numbers of polynomial sized denominator. Proof follows from theorem 12 or 13. Q.E.D.

Corollary 7 *The set of all possible assignments of goods to people (x_{ij} variables) at equilibria is convex.*

For the purpose of using ellipsoid algorithm, linear program 3 does not offer any advantage over the convex program 2. But a linear program can be useful for designing primal-dual algorithms. The infinite size of the linear program should not be a concern in designing a primal-dual algorithm. In the past, exponential sized linear programs are used for designing primal-dual algorithms. A cleverly designed primal-dual algorithm identifies a polynomial number of dual variables to be used.

Convex program 2 is of exponential size and if it is converted into a linear program then it is of infinite size. So convex program 2 is not suitable for developing more efficient interior point methods. In the next section we develop a new polynomial size convex program which gives a promising hope of developing interior point methods.

6. Convex Program

Note that we need to write the third line in the nonconvex program 1 for only those i and j's for which $u_{ij} > 0$. Further note that this implies that $\sum_k u_{ik}x_{ik} > 0$. We already have that $p_i, p_j > 0$. So we can take the $\log$ of the whole inequalities to get:

$$\forall i,j \text{ such that } u_{ij} > 0: \ \log(p_i) - \log(p_j) \leq \log\left(\frac{\sum_k u_{ik}x_{ik}}{u_{ij}}\right)$$

We substitute every $\log(p_i)$ with a new variable $LOGp_i$. We then get:

$$\forall i,j \text{ such that } u_{ij} > 0: \ LOGp_i - LOGp_j \leq \log\left(\frac{\sum_k u_{ik}x_{ik}}{u_{ij}}\right)$$

Note that $\log$ is a concave function i.e., $\log(\frac{x+y}{2}) \geq \frac{\log(x)+\log(y)}{2}$. This means that if two feasible point satisfy the above inequality then their average will also satisfy the inequality. So the equivalent convex program we get for the non-convex program 1 is:

$$\forall j: \sum_i x_{ij} = 1$$
$$\forall i,j: x_{ij} \geq 0$$
$$\forall i,j \text{ such that } u_{ij} > 0:$$

IEEE COMPUTER SOCIETY

$$LOGp_i - LOGp_j \leq \log(\frac{\sum_k u_{ik}x_{ik}}{u_{ij}}) \quad (4)$$

Theorem 8 *Non-convex program 1 is equivalent to convex program 4.*

Corollary 9 *The set of all possible equilibria prices,on a logarithmic scale ($LOGp_i$) is convex.*

7. General case

In the model section we made an assumption that for every proper subset, S, of persons there is a $i \notin S$ and $j \in S$ such that $u_{ij} > 0$. This assumption is not without loss of generality. We will justify the assumption in this section. In this section we show that even without this assumption there is an equilibrium consisting of only non-zero prices. Hence, the convex program 4 remains valid and give all such equilibria. We also give a way to find other equilibria where some of the prices are zero.

We draw the *non-zero liking graph* of the problem. This graph has a node for every person in the economy. There is a directed edge from i to j whenever $u_{ij} > 0$. If i and j are the same then we put a loop on i. If this graph is disconnected, i.e., there is a proper subset S, such that there is no edge between S and $\bar{S}$ (S complement), then S and $\bar{S}$ can be considered separate economies. One can write the convex programs of two separate economies together and call it a convex program for the joint economy. So we assume that the graph is connected.

If the graph is strongly connected then it satisfies the assumption and we are done. Else we compute the strongly connected component decomposition of the graph. First, write the convex program for the equilibria of each component. For each component we know that equilibria prices are all non-zero. Now consider the underline acyclic structure on the strongly connected components. We say that a component, S_1, is *lower* than another component, S_2, if there is an edge from S_1 to S_2. Take the transitive closure of this *lower* relation. This will be a partial order. Again denote it by *lower*. Note that if S_1 is lower than S_2 then S_2 is not lower than S_1. Hence if the goods in S_1 are non-zero priced then they can't move from S_1 to S_2. On the other hand if goods are heavily priced in S_2 then they can't move from S_2 to S_1 either. So we find non-zero equilibria for each component. If S_1 is lower than S_2 then we scale up the prices in S_2 by a huge number so that every person in S_1 likes something in S_1 in comparison with anything in S_2. Hence all the prices are non-zero. For all such equilibria vectors we can write the convex program 4.

In case one wants to allow zero-prices then note that only a lower ideal can have zero prices. So take any lower ideal put the zero prices for these. For rest of the economy write the convex-program 4. Also note that the corollaries in the previous two sections remain valid whenever they are meaningful.

8. Non-linear utilities

In this section we explore the case when the utility functions are non-linear but concave. We assume that the utility functions are differentiable[1] Let $u_i(x_i)$ be the utility function of i, where x_i is her consumption vector. We assume that u_i is concave, i.e.,

$$\frac{u_i(x_i) + u_i(y_i)}{2} \leq u_i(\frac{x_i + y_i}{2}),$$

for every consumption vectors x_i and y_i. Let $u_{ij}(x_i)$ be the partial derivative of u_i at point x_i with respect to the consumption of j^{th} good (consumption of j^{th} good by i^{th} person is denoted by x_{ij}.). Now replace the u_{ij} in the non-convex program 1 by $u_{ij}(x_i)$ (where $x_i = (x_{i1}, x_{i2}, \cdots, x_{in})$. For brevity, $u_{ij}(x_i)$ is written as u_{ij}, since the argument is understood by the context. Now we claim that the non-convex program 1 is valid for the non-linear utility's case too.

Theorem 10 *The feasible region of non-convex program 1 has all and only general market equilibria even if utilities are general differentiable concave functions.*

Proof :

First, fix a price vector. At this price whatever a person can potentially buy is a convex set. But the person will buy that bundle of good which will maximizes her utility. Since the utility function is concave, any local minima will also be a global minima, or more generally, the set of maximum utility bundles is a convex set. The conditions of local minima are straight forward, the marginal utility per unit of additional money for all the consumed goods is the same and for other goods it is no bigger.

This time we will prove the harder part first, i.e., every feasible solution of the non-convex program 1 is an equilibrium point. Note that we did not use the fact that u_{ij} is constant when we proved lemmas 2 and 3. So these lemmas are still valid. Hence the harder side of the theorem follows.

Now we want to prove the easier side. In the linear case, the third line constraints, which represented optimally, were obvious. This time we will also have to use an additional fact that the "money earned is equal to money spent" at the equilibrium. We know that for every i, whenever $x_{ij} > 0$, marginal utility per unit of money is the same, i.e., for every i, the quantity $\frac{u_{ij}}{p_j}$ is the same whenever $x_{ij} > 0$. Since

1 If the utility functions are not differentiable than it may be possible to use sub differentials instead of differentials.

$x_{ij} > 0$, we can say that $\frac{x_{ij}u_{ij}}{x_{ij}p_j}$ is the same. Add all the numerator and denominator together and note that the denominator is p_i by "money earned equal to money spent" constraint. Hence the easier part of the theorem also follows. Q.E.D.

Above theorem shows that the non-convex program 1 remains valid. In fact, the program 4 is still valid too, but it may not always be convex. For some utility functions, e.g., if the utility of the i^{th} person is $\sum_j \sqrt{x_{ij}}$ then the program 4 is convex. Another interesting case is if the utility of the i^{th} person is $\sum_j \log(1 + x_{ij})$, then also the program 4 is convex. The program is convex for $\sum_j \frac{x_{ij}}{1+x_{ij}}$ too. One example where the program is not convex is $\sum_j (1 - e^{-x_{ij}})$. So a natural open question is when program 4 is convex. We think the answer includes the utility functions with weak gross substitutability property. Unlike in Fisher case, the answer does not include the homogeneous utility function [17]. Gjerstad in [17] gave an example with homogeneous utilities in which the set of market equilibria is not even connected.

It is worth mentioning that program 4 simplifies for the Fisher case. In Fisher case we do not need to take logs. One can analogously prove that the following program 5 characterizes equilibria in Fisher case even for non-linear utilities. It is convex for linear utilities.

$$
\begin{aligned}
&\forall j: \ \sum_i x_{ij} = 1 \\
&\forall i, j \ : \ m_i u_{ij} \le p_j \sum_k u_{ik} x_{ik} \\
&\sum_j p_j \le \sum_i m_i \\
&\forall i, j: \ x_{ij} \ge 0 \\
&\forall j: \ p_j \ge 0
\end{aligned} \tag{5}
$$

x_{ij} has the same meaning as in the program 4. Here m_i is the amount of money the i-th buyer has. p_j is a variable for the price of good j.

9. Toward Combinatorial Algorithms

The algorithm in this paper is neither combinatorial nor strongly polynomial. As discussed in the introduction algorithms using convex programming has many theoretical advantage. Convex programs help us understand the problem by revealing basic structures of the problem. The convex program in section 6 is no different. It also leads to an efficient practical algorithm using interior point methods [31]. And as shown in theorem 4 the convex programs in this paper also leads to a combinatorial characterization of the equilibria. This combinatorial characterization is kind of passive. When we discover a negative cycle the characterization does not tell us how to fix it. On the other hand a negative cycle in minimum cost flow problem tells us how to decrease the cost of flow.

We have such a characterization for Fisher model using Eisenberg-Gale's LP [14]. This characterization may help us develop a strongly polynomial combinatorial algorithm or a strongly polynomial time algorithm for the Fisher model. The following theorem, which is an active characterization of equilibria in Fisher model may play a role. This characterization does not only tell us when an assignment is not in equilibrium but also tells us how to fix it. To our knowledge there is no other combinatorial characterization known for the equilibria in Fisher model, which can tell us when an assignment is in equilibria. The algorithm in [10] neither use nor imply any combinatorial characterization for the problem.

Theorem 11 *Consider an assignment x_{ij}'s from goods to buyers. x_{ij} is an equilibrium if and only if there does not exist a good j and two buyers i and i' such that i has a non-zero quantity of good j and when i gives a sufficiently small but non-zero quantity of good j to i' then the product of $U_i^{m_i} U_{i'}^{m_{i'}}$ increases, where U_i and $U_{i'}$ are the utilities of i and i' and m_i and $m_{i'}$ are their initial endowments of money.*

Proof : The forward direction is immediately implied by Eisenberg-Gale LP [14]. For the reverse direction let x is the assignment which is not in equilibrium. Let us say that x' is the assignment in equilibrium. So Eisenberg-Gale's objective function is higher at x' than x. Consider the straight line segment between x and x'. Let $z = x' - x$. Let $\epsilon \in [0, 1]$. Eisenberg-Gale's objective function on any point on this line segment is:

$$\sum_i m_i \log \left(\sum_j u_{ij} \left(x_{ij} + \epsilon z_{ij} \right) \right)$$

Note that Eisenberg-Gale's objective function is strictly concave. So its value is strictly higher at positive ϵ than at $\epsilon = 0$. So its derivative at $\epsilon = 0$ is positive.

$$\sum_i \frac{m_i \sum_j u_{ij} z_{ij}}{\sum_j u_{ij} x_{ij}} > 0$$

Let $w_{ij} = m_i u_{ij} / \sum_{j'} u_{ij'} x_{ij'}$. Note that w_{ij} does not depend upon z. The above inequality can then be written as $\sum_i \sum_j w_{ij} z_{ij}$. We construct a bipartite graph with i's on one side and j's on the other. For every $z_{ij} > 0$, we draw an edge from i to j with (fractional) multiplicity z_{ij}. For every $z_{ij} < 0$, we draw an edge from j to i with (fractional) multiplicity $-z_{ij}$. Note that the bipartite graph is eulerian on the right hand side, i.e., sum of z_{ij}'s on incoming edges to a node is same as the sum of z_{ij}'s on outgoing edges from the node. This property implies with a simple inductive proof that the above graph can be decomposed into graphs with two edges $i'j$ and ji with the same multiplicity. This means that the weight function i.e., $\sum_i \sum_j w_{ij} z_{ij}$ can be written as a positive combination of the weight function on these two edge graphs. This implies that there is a at least one two edge graph with positive weight function. Let one such graph be $i'j$ and ji, i.e., $w_{i'j} - w_{ij} > 0$. Note that since z_{ij} is negative i has some positive quantity of good j. Consider another assignment y which is obtained by x by giving all of i's good j to i'. Now consider the line segment between x and y. Let us consider the derivative of Eisenberg-Gale's objective function in the direction of x to y. This derivative is positive at x. There are two case. The first is that the derivative remains positive on this line segment, in this case we move to assignment y. The other case is that the derivative is zero somewhere on the line segment, say at y'. In that case we move to y'. In both cases we prove the theorem. Q.E.D.

10. Generalized convex feasibility testing algorithm via ellipsoid and simultaneous diophantine approximation

This section generalizes the Theorem 6.4.1 in Grotschel, Lovasz, Schrijver's book on Geometric Algorithms and Combinatorial Optimization [19]. The theorem says: "The strong non-emptiness problem for well-described polyhedra, given by a strong separation oracle, can be solved in oracle-polynomial time."

The theorem makes an assumption of well-described polyhedra which is not true for our convex program or for Eisenberg-Gale's convex program. Well-described polyhedra means that any facet of the polyhedra can be encoded with binary encoded length ϕ. Which implies that the coefficients used to describe a facet are rational numbers with binary encoded length ϕ. This also implies that the corner points of the polyhedra also use rational numbers with binary encoded length polynomial in ϕ and n, the dimension of the space. Binary encoded length of a rational number is the sum of the binary encoded lengths of the numerator and the denominator. We prove the following theorem:

Theorem 12 *Given a convex set via a strong separation oracle with a guarantee that the set contains a point with binary encoding length at most ϕ, a point in the convex set can be found in polynomial time.*

Proof :

Let P be the given convex set and $\mathcal{S}$ be the given strong separation oracle. Let Q be the convex hull of all the points in P with binary encoding length ϕ. Lemma 6.2.4(b) of [19] shows that Q is a well-described polyhedra. If we had a strong separation oracle for Q then we could have used Theorem 6.2.1 of [19] to obtain a point of Q, which will also be a point of P. Since $\mathcal{S}$ separates any point not in P from P, $\mathcal{S}$ also separates any point not in P from Q. The set of points which $\mathcal{S}$ fails to separate from Q is $P - Q$. In any case, let us try to fool the Theorem 6.2.1 of [19] by invoking it on incorrect inputs. We run the algorithm, used in the theorem, on Q and $\mathcal{S}$. The input is incorrect because $\mathcal{S}$ is not a separation oracle for Q. Note that the run of the algorithm will not notice the inconsistency of the input until it tries to separate a point of $P - Q$ from Q, but in this case we already have a point in P, which proves our theorem. The other case is that the run of the algorithm does not try to separate a point of $P - Q$ from Q, in this case $\mathcal{S}$ works fine and the successful completion of the run is guaranteed by the proof of Theorem 6.2.1 of [19].

Q.E.D.

Note that since the algorithm runs in polynomial time it finds a solution with binary encoded length $P(n)\phi$, where $P(n)$ is a polynomial. Hence the following theorem follows:

Theorem 13 *Given a convex set by a strong separation oracle and a prescribed precision ϕ, there is an oracle-polynomial time and ϕ-linear time algorithm which does one of the following:*

- *Concludes that there is no point in the convex set with binary encoded length at most ϕ.*
- *Produces a point in the convex set with binary encoded length at most $P(n)\phi$, where $P(n)$ is a polynomial.*

Note that $P(n)$ is a sort of approximation factor, which is an artifact of exponential approximation factor in simultaneous diophantine approximation. Finding an algorithm with $P(n) = 1$ or a matter of fact any constant should be a challenging problem.

11. Acknowledgment

I would like to thank Laszlo Lovasz for verifying a complicated proof and then inspiring a simplification. I would also like to thank him for providing an expert advice on the last section of the paper. I also like to thank Rakesh Vohra for exchanging many pieces of email and providing important references. I also like to thank Bruno Codenotti, Mohammad Mahdian, Vahab S. Mirrokni, Amin Saberi, Kasturi Varadarajan, Vijay V. Vazirani and Yinye Ye for helpful discussions.

References

[1] K. Arrow, H. D. Block, and L. Hurwicz. On the stability of the competitive equilibrium, II. *Econometrica* 27, 82–109, 1959.

0272-5428/04 $20.00 © 2004 **IEEE**

[2] K. Arrow and G. Debreu. Existance of an equilibrium for a competitive economy. *Econometrica*, 22:265-290, 1954.

[3] W. C. Brainard and H. E. Scarf. How to compute equilibrium prices in 1891. *Cowles Foundation Discussion Paper*, (1270), 2000.

[4] B. Codenotti and K. Varadarajan. Equilibrium for Exchange Markets with CES Utility Functions. Manuscript July 26, 2004.

[5] B. Codenotti, S. Pemmaraju, and K. Varadarajan. On the Polynomial Time Computation of Equilibria for Certain Exchange Economies. Manuscript July 27, 2004.

[6] R. Cole, Y. Dodis, T. Roughgarden. Pricing network edges for heterogeneous selfish users. *STOC* 2003.

[7] S. P. Dirkse and M. C. Ferris. A pathsearch damped Newton method for computing general equilibria. *Annals of Operations Research*, pages 211-232, 1996.

[8] X. Deng, C. Papadimitriou, and S. Safra. On the complexity of equilibria. In *34th ACM Symposium on Theory of Computing (STOC 2002)*, Montreal, Quebec, Canada, May 2002.

[9] N. Devanur, C. Papadimitriou, A. Saberi, and V. Vazirani. Market equilibrium via a primal-dual-type algorithm. In *43rd Symposium on Foundations of Computer Science (FOCS 2002)*, pages 389-395, Nov. 2002.

[10] N. Devanur, C. Papadimitriou, A. Saberi, and V. Vazirani. Market equilibrium via a primal-dual-type algorithm. Manuscript made public in 2004.

[11] N. Devanur and Vijay V. Vazirani. An Improved Approximation Scheme for Computing Arrow-Debreu Prices for the Linear Case. In *Proc FSTTCS*, 2003.

[12] B. C. Eaves. A Finite algorithm for the linear exchange model, *Journal of Mathematical Economics*, Vol 3, Issue 2, July 1976, Pages 197-203.

[13] E. Eisenberg. Aggregation of Utility Functions. *Management Sciences*, Vol 7 (4), 337–350 (1961).

[14] E. Eisenberg and D. Gale. Consesus of subjective probabilities: The pari-mutuel method. *Annals of Mathematical Statistics*, 30:165-168, 1959.

[15] M. C. Ferris, S. P. Dirkse, and A. Meeraus. Mathematical Programs with Equilibrium Constraints: Automatic Reformulation and Solution via Constrained Optimization. *Numerical Analysis Group Research Report* NA-02/11,*Oxford University Computing Laboratory, Oxford University*, 2002.

[16] L. Fleisher, K. Jain and M. Mahdian. Taxes for heterogeneous selfish users in a multicommodity network. Submitted in this conference.

[17] S. Gjerstad. Multiple Equilibria in Exchange Economies with Homothetic, Nearly Identical Preference. *University of Minnesota, Center for Economic Research*, Discussion Paper 288, 1996.

[18] R. Garg and S. Kapoor. Auction Algorithms for Market Equilibrium. *IBM Research Report*, 2003.

[19] M. Grotschel, L. Lovasz, and A. Schrijver. Geometric Algorithms and Combinatorial Optimization, Second corrected edition, *Springer Verlag* Berlin Heidelberg, 1993.

[20] K. Jain. A Polynomial Time Algorithm for Computing the Arrow-Debreu Market Equilibrium for Linear Utilities. Manuscript distributed in 2003.

[21] K. Jain. A Polynomial Size Convex Program for Market Equilibria with Production Planning and Linear Utilities. Submitted.

[22] K. Jain, M. Mahdian, and A. Saberi. Approximating Market Equilibrium. *Appeared in the Proc. of APPROX*, 2003.

[23] K. Jain, V. V. Vazirani, and Y. Ye. Market equilibria for homothetic, quasi-concave utilities and economies of scale in production. Submitted.

[24] K. Jain and Y. Ye. About computing Market Equilibria for quasi-concave utilities. Ongoing work.

[25] F. P. Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 8:33–37, 1997.

[26] D. J. Newman and M. E. Primak. Complexity of Circumscribed and Inscribed Ellipsoid Methods for Solving Equilibrium Models. *Applied Mathematics and Computation* 52:223–231 (1992).

[27] C. H. Papadimitriou. On the Complexity of the Parity Argument and Other Inefficient Prood of Existence. *JCSS* 48(3),pp. 498-532 (1994)

[28] C. H. Papadimitriou. Algorithms, games, and the internet. In *Proceedings of the 33rd Annual ACM Symposium on the Theory of Computing*, pages 749-753, 2001.

[29] M. E. Primak. A converging algorithm for a linear exchange model. *Journal of Mathematical Economics*, Vol 22, Issue 2, 1993, pages 181-187.

[30] L. Walras. Elements of pure economics, or the theory of social wealth (in French). Lausanne, Paris, 1874.

[31] Y. Ye. A Path to the Arrow-Debreu Competitive Market Equilibrium. Manuscript made public in 2004.

0272-5428/04 $20.00 © 2004 **IEEE**