CHAPTER 8

On the Computation of Equilibrium Prices

HERBERT SCARF*

1. Introduction

In *Mathematical Investigations in the Theory of Value and Prices*, published in 1892, Irving Fisher described a mechanical and hydraulic analogue device intended to calculate equilibrium prices for a general competitive model. This chapter takes up the same problem and discusses an algorithm for a digital computer which approximates equilibrium prices to an arbitrary degree of accuracy.

At least two versions of Fisher's device were actually constructed and apparently performed successfully. The devices themselves have unfortunately been lost, but there are several photographs, which may be seen in the edition of Fisher's volume reprinted in 1961 by Yale University Press.

The equipment seems remarkably quaint and old-fashioned in this era of high-speed digital computers. Immersed in a large tub filled with water are a number of canisters whose irregular profiles are related to the consumers' marginal utilities for the various commodities. Each canister is constructed partly of flexible leather, looking somewhat like a bellows that expands and contracts in response to changes in prices. The canisters are connected by an elaborate system of rods, hinges, and tubes filled with water.

In order to specify the consumers' initial dollar incomes, a row of plungers is adjusted to specific heights, and in the pure exchange model, a similar series of adjustments is made to provide information about the initial stocks of commodities before trade takes place. The competitive price levels and allocations are then determined when the system reaches a physical state of equilibrium.

* The research described in this paper was carried out under a grant from the National Science Foundation.

Reprint From
Ten Economic Studies in the
Tradition of Irving Fisher
Published by John Wiley & Sons, Inc. 1967

To avoid elaborate engineering problems, Fisher found it necessary to assume that the utility functions could be written in a separable form so that the marginal utility of any commodity was independent of the level of consumption of the remaining commodities. In the model to which the algorithm of this paper is applied, there is no need for an assumption of separable utility. Each consumer will have a set of demand functions, which are continuous and homogeneous of degree zero in all prices, and, in addition, a given vector of commodities which are owned prior to production and trade. Fisher assumes that a specific dollar income appears on the right-hand side of each individual's budget constraint; recent authors have preferred to work with a more general model in which income is derived from the sale of factors whose prices are to be determined at equilibrium.

Assuming, in addition, that no income is generated by profits arising from production, the market demand functions satisfy Walras's law, to the effect that the market value of demand at any set of prices is equal to the value of the stock of initially owned commodities.

Let the market demand functions be denoted by

$$\begin{gathered}\xi_1(\pi_1, \ldots, \pi_n)\\ \vdots\\ \xi_n(\pi_1, \ldots, \pi_n)\end{gathered}$$

with $(\pi_1, \ldots, \pi_n)$ the vector of prices. Since the demand functions are naturally homogeneous of degree zero, it is sufficient to assume that they are defined only for prices which are nonnegative and sum to 1, and continuous on this set of prices. If the total stock of commodities prior to production and trade is given by the nonnegative vector $(w_1, \ldots, w_n)$, then the Walras law states that

$$\pi_1\xi_1(\pi) + \cdots + \pi_n\xi_n(\pi) \equiv \pi_1 w_1 + \cdots + \pi_n w_n$$

identically for all prices.

Whereas the consumption side of the economy is treated by Fisher in an essentially modern fashion, the model of production seems quite inadequate. There is no production function or transformation set, but instead Fisher uses a notion of the "marginal disutility of production," which is to be equated to the negative of the marginal utility of consumption for each consumer and commodity. Moreover, factors do not seem to be used up in production nor is income generated by the sale of productive factors.

The algorithm of this chapter permits production to be described by an arbitrary activity analysis model with a finite list of activities. Each column of the matrix

$$A = \begin{bmatrix} -1 & 0 & \cdots & 0 & a_{1,n+1} & \cdots & a_{1,m} \\ 0 & -1 & \cdots & 0 & \cdot & & \cdot \\ & & \cdot & & \cdot & & \cdot \\ & & & \cdot & \cdot & & \cdot \\ & & & & \cdot & & \cdot \\ 0 & 0 & \cdots & -1 & a_{n,n+1} & & a_{n,m} \end{bmatrix}$$

will refer to a specific productive process with inputs indicated by negative numbers and outputs by positive numbers.

The economy is assumed to be completely described by the market demand functions $\{\xi_i(\pi)\}$, the technology matrix A, and the stock of factors $(w_1, \ldots, w_n)$ prior to production. With this notation a competitive equilibrium is defined by a vector of prices $\pi_1, \ldots, \pi_n$, and a collection of nonnegative activity levels, $x_1, \ldots, x_m$, such that the following two sets of conditions are met:

1. Supply equals demand in each market, or, mathematically,

$$\xi_i(\pi) - \sum_{j=1}^{m} a_{ij}x_j = w_i \qquad \text{for} \quad i = 1, \ldots, n$$

2. Profit is maximized at the prices π, or

$$\sum_{i=1}^{n} \pi_i a_{ij} \leq 0 \qquad \text{for all} \quad j = 1, \ldots, m$$

with equality if $x_j > 0$.

The existence of equilibrium prices for models of this type has been investigated with great thoroughness by a number of writers (for example, [1] or [3]), and we now know that if certain relatively mild conditions are placed upon the specification of the model, there will indeed be an equilibrium. In our model the conditions are quite simple. In addition to the assumptions already made, we require that the set of activity levels which gives rise to a nonnegative *net* supply of all commodities forms a bounded set. In symbols, the set of nonnegative $(x_1, \ldots, x_m)$ for which

$$\sum_{j=1}^{m} a_{ij}x_j + w_i \geq 0 \qquad \text{for all } i$$

will be a bounded set. The assumption $w_i > 0$ for all i will also be used occasionally even though it can be replaced by more realistic requirements.

To demonstrate the existence of equilibrium prices in a model of this generality, it has been customary to make use of what are known as "fixed point" theorems, which describe conditions under which a continuous mapping leaves at least one point unchanged. The arguments leading from a fixed point theorem to the existence of equilibrium prices are often quite direct and economically suggestive; they have, however, a major drawback of offering no reasonable suggestion for the computation of such prices.

It may seem somewhat surprising, in view of the substantial body of work in mathematical programming, that no techniques have been proposed for what is one of the central problems in economic theory: the computation of equilibrium prices. This is undoubtedly due to the preoccupation with models which are exclusively on the production side of the economy and make no reference to the role played by consumers in the determination of equilibrium prices.*

There is a sense, well-known to economists, in which the model of competitive behavior does give rise to a nonlinear maximization problem similar to those encountered in the theory of production. If each consumer has a concave utility function, then the maximization of a weighted sum of utilities subject to the constraints of the technology and the existing stock of commodities does produce a set of prices which have many of the properties of equilibrium prices. Producers are maximizing profit at these prices, and no consumer can receive a higher utility at lower cost. There is, however, one serious drawback: unless the utility weights are selected in precisely the right way, the consumption of each consumer is in no way related to the income generated by the sale of his productive factors. Unless we are willing to neglect this vital link in the economic system completely, the problem has merely been shifted from the determination of equilibrium prices to the determination of the appropriate utility weights, and the latter problem is no simpler than the former.

* In his thesis, Rolf Mantel [4], gives a proof of the existence of equilibrium prices which is similar in many respects to the arguments of this chapter. I have also received an unpublished manuscript by H. Houthakker describing an algorithm which should be very effective under certain severe assumptions about the demand functions and the technology.

Nor is any general computational approach to be found in the literature about the stability of equilibrium, in which the process of adjusting prices to excess demands may be viewed as a gradient method for the computation of equilibrium prices. Even though gradient methods are successful on the production side of the economy, they need not be stable in a model involving consumers unless some relatively stringent assumptions, such as "gross substitutability," are placed on the market demand functions.

The basis for an effective algorithm for the computation of equilibrium prices has come from a rather unexpected source. Until recently, the existence of Nash or Cournot equilibrium points in a finite, two-person, nonzero sum game has been treated by the same nonconstructive topological methods as those used in equilibrium analysis. But Lemke, working with a student, Howson, has devised a most ingenious algorithm, based on pivot steps as used in linear programming, for calculating a Nash equilibrium point for a two person game [2]. Even though these problems have only a mathematical connection, Lemke's basic idea may be combined with a different notion of pivoting to give a constructive algorithm for approximating fixed points of a continuous mapping, for finding a point in the core of an economy, and for the algorithm used here [5, 6].

The next three sections describe the details of our algorithm. The reader whose interest is less in technical matters than in applications may prefer to jump to Section 5, in which some examples are given.

2. Setting the Stage for the Algorithm

As we have seen, it is sufficient to consider only those price vectors $\pi = (\pi_1, \ldots, \pi_n)$ which lie on the simplex $\pi_i \geq 0$, and $\sum_{i=1}^{n} \pi_i = 1$. Rather than examine all vectors on this simplex, we shall assume that a large but finite list of vectors $\pi^{n+1}, \ldots, \pi^k$ has been selected and restrict our attention to these vectors as potential equilibrium prices. (The reason for beginning the list with π^{n+1} rather than π^1 should become clear later.) Since the actual equilibrium vector need not be found in this list, the algorithm will provide only an approximation, but one whose accuracy can be increased by enlarging the list of vectors.

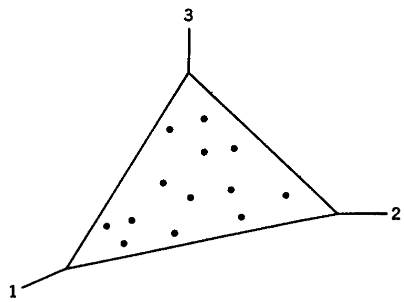

Figure 1

The algorithm has been applied to a number of specific examples, some of which involved a set of vectors containing as many as 10^{16} members. Of course, if all of the vectors in such a list were to be examined in order to determine an approximate equilibrium price vector, we would substantially exceed the capabilities of even the fastest electronic computers; the algorithm, however, has rarely required an examination of more than 1500 such vectors and has generally terminated with a far smaller number.

After the vectors $\pi^{n+1}, \ldots, \pi^k$ have been selected, the next step is to construct a particular matrix B with n rows and k columns. The first n columns of B will consist of a unit matrix. Column j, with $n + 1 \leq j \leq k$, will be related to the vector π^j, in the list of vectors, according to the following specific rules:

1. Let

$$\begin{matrix} a_{1l} \\ \cdot \\ \cdot \\ \cdot \\ a_{nl} \end{matrix}$$

be an activity in the technology matrix A which yields a maximum profit at the prices π^j. If there are several activities which give the same maximum profit, then an arbitrary selection of one of these is made.

2. If the largest profit obtainable at the prices π^j is positive, then the jth column of B is defined to be

$$\begin{matrix} -a_{1l} \\ \cdot \\ \cdot \\ \cdot \\ -a_{nl} \end{matrix}$$

3. If the largest profit at the prices π^j is less than or equal to zero, then the jth column of B is defined to be

$$\begin{matrix} \xi_1(\pi^j) \\ \cdot \\ \cdot \\ \cdot \\ \xi_n(\pi^j) \end{matrix}$$

The general appearance of the matrix B is as follows (the vector π^j is written above column j to indicate the connection between the two.):

$$\begin{array}{cc} & \begin{matrix} & \pi^{j_1} \cdots & \pi^{j_2} \end{matrix} \\ B = & \begin{bmatrix} 1 & 0 & \cdots & 0 & -a_{1l} & \xi_1(\pi^{j_2}) \\ 0 & 1 & \cdots & 0 & \cdot & \cdot \\ & & & & \cdot & \cdot \\ & & & & \cdot & \cdot \\ 0 & 0 & \cdots & 1 & -a_{nl} & \xi_n(\pi^{j_2}) \end{bmatrix} \end{array}$$

Aside from the first n columns, which will play the role of slack· variables, the columns of B will be composed either of the market demands at a given set of prices or the negative of that activity vector which maximizes profit for the price vector appearing above the column. Of course some of the activity vectors will be repeated a substantial number of times.

The number of columns of B is apt to be very large, and we are fortunate that the algorithm never requires an explicit representation of this matrix in the memory units of the computer.

We shall be concerned with nonnegative solutions of the equations $Bx = w$, where

$$w = (w_1, \ldots, w_n)'$$

the vector of factors available prior to production. When these equations are written out explicitly they become

$$-\sum_j a_{il}x_j + \sum_j \xi_i(\pi^j)y_j = w_i$$

where in the first set of terms, the subscript l depends on j and refers to that activity which maximizes profit at the prices π^j, should that profit be positive. In the second set of terms the x_j's have been replaced by y_j's to emphasize the distinction between the two types of columns.

From the way in which the matrix B is constructed the following conditions are satisfied:

1. If $y_j > 0$, then $\sum_{i=1}^{n} \pi_i^j a_{il} \leq 0$ for *every l*.
2. If any activity l, other than a disposal activity, has a positive weight x_j, then $\sum_{i=1}^{n} \pi_i^j a_{il} > 0$.

The basic idea of our algorithm is to approximate an equilibrium price vector by determining a nonnegative solution to the equations $Bx = w$ with the property that *all* of the prices π^j corresponding to *positive* x_j or y_j are *close to each other*, and that the ith coordinate of all of these prices is *close* to zero if the ith slack variable is positive.

In order to see that such a solution would indeed represent an approximation to a competitive equilibrium, let us imagine that the prices π^j corresponding to positive x_j and y_j are so close to each other that they can be replaced by a common price π in the preceding equations, and in conditions 1 and 2.* In addition, π_i will be zero if the ith slack is positive. Since all of the $\xi(\pi^j)$ with positive weights y_j are replaced by $\xi(\pi)$, the equations become

$$-\sum a_{il}x_j + \left(\sum_j y_j\right)\xi_i(\pi) = w_i$$

which would describe the equality of supply and demand in all markets if it could be demonstrated that $\Sigma\, y_j = 1$.

* A more precise mathematical treatment would involve taking successively more refined grids on the simplex and letting π be a limit point of those π^j corresponding to positive x_j or y_j.

Moreover, condition 1 becomes 1′, as follows:

1′. If $y_j > 0$, then $\Sigma\, \pi_i a_{il} \leq 0$

for all l, so that the fact that no activity makes a positive profit at prices π follows from the positivity of at least one y_j.

Condition 2 in conjunction with the fact that $\pi_i = 0$ if the ith slack variable is positive may be restated as 2′:

2′. If any activity l, including disposal activities, has a positive weight x_j, then $\Sigma\, \pi_i a_{il} \geq 0$.

In order to show that we do indeed have an equilibrium price, we first show that at least one y_j is strictly positive, for then conditions 1′ and 2′ imply that no activity makes a positive profit, and that those activities which are operated at a positive level have a zero profit. Moreover, if the preceding equations are multiplied through by π_i and added we would then see that

$$\left(\sum_j y_j\right)\left[\sum_i \pi_i \xi_i(\pi)\right] = \sum_i \pi_i w_i$$

and it follows from the Walras law and the positivity of $\Sigma\, \pi_i w_i$ that $\Sigma\, y_j = 1$.

The only missing link in our argument is therefore the demonstration that at least one y_j is strictly positive. But if all y_j are zero, it follows that

$$-\sum a_{il} x_j = w_i$$

and

$$\sum \pi_i a_{il} \geq 0$$

for all l with a positive x_j. But then

$$0 \leq \sum \sum \pi_i a_{il} x_j = -\sum \pi_i w_i$$

which contradicts the positivity of $\Sigma\, \pi_i w_i$. (The assumption $w_i > 0$ for all i is used here and in the previous paragraph.) This concludes our argument that π represents an equilibrium price.

Of course the fact that the vectors π^j corresponding to positive x_j and y_j are close to each other does not permit a literal replacement by a common vector π, but it should be clear from the continuity of the demand functions that an average of π^j will serve as an *approximate* equilibrium price vector, since supply will be approximately equal to demand in all markets, profits if positive will be small, and the profits of those activities used at a positive level will be close to zero.

3. The Main Theorem

In Section 2 we constructed a matrix

$$\begin{array}{cc} & \begin{array}{ccccccc} & & & & \pi^{n+1} & & \pi^{k} \end{array} \\ B = & \begin{bmatrix} 1 & 0 & \cdots & 0 & b_{1,n+1} & \cdots & b_{1,k} \\ & & & & \cdot & & \\ & & & & \cdot & & \\ & & & & \cdot & & \\ 0 & 0 & \cdots & 1 & b_{n,n+1} & \cdots & b_{n,k} \end{bmatrix} \end{array}$$

whose jth column was either the market demand at prices π^j or the negative of the profit maximizing activity at these prices, with π^j a specific price on the simplex $\left\{\pi \mid \pi_i \geq 0, \sum_i \pi_i = 1\right\}$. We then showed how an approximate equilibrium price vector could be determined by finding a nonnegative solution to the equations $Bx = w$ such that all of the vectors π^j with positive x_j are close to each other, and with the ith coordinate of each of these vectors close to zero, if the ith slack is positive. To be specific about this statement, we must formulate a precise definition of the concept of closeness.

It will be useful to begin by making the following assumption which can easily be brought about by a perturbation of the vectors $\pi^{n+1}, \ldots ; \pi^k$.

Nondegeneracy Assumption. No two vectors in the set $\pi^{n+1}, \ldots, \pi^k$ have the same ith coordinate for any i, and no vector has a zero coordinate.

Consider n vectors $\pi^{j_1}, \ldots, \pi^{j_n}$ selected from the list of vectors. These vectors may be used to generate a subsimplex in the following way: Begin by finding that one of the n vectors which has the smallest first coordinate. This will yield a unique vector because of the nondegeneracy assumption. Pass a hyperplane with constant first coordinate through that vector. Then find that one of the n vectors with smallest second coordinate and pass a hyperplane with constant second coordinate through that vector. If we continue in this fashion with each coordinate, the subsimplex which is then generated consists of all points $\pi = (\pi_1, \ldots, \pi_n)$ with

$$\pi_1 \geq \min(\pi_1^{j_1}, \ldots, \pi_1^{j_n})$$
$$\vdots$$
$$\pi_n \geq \min(\pi_n^{j_1}, \ldots, \pi_n^{j_n})$$

and

$$\pi_1 + \cdots + \pi_n = 1$$

In Figure 2 the list of vectors is given by $\pi^4, \ldots, \pi^k$, and two subsimplices have been drawn, one generated by π^4, π^5, and π^6, and the other by π^7, π^8, and π^9. The first triple of vectors are fairly far apart, whereas the three vectors in the second triple are all quite close, which is indicated by the fact that there are several vectors in the list $\pi^4, \ldots, \pi^k$ *interior* to the first subsimplex but none *interior* to the second. In general, if there are no vectors in the list $\pi^{n+1}, \ldots, \pi^k$ *interior* to the subsimplex generated by $\pi^{j_1}, \ldots, \pi^{j_n}$, then these n vectors must be close to each other. This is the concept of closeness we shall adopt.

Before giving a formal definition, let us make one extension to accommodate the possibility of forming subsimplices some of whose

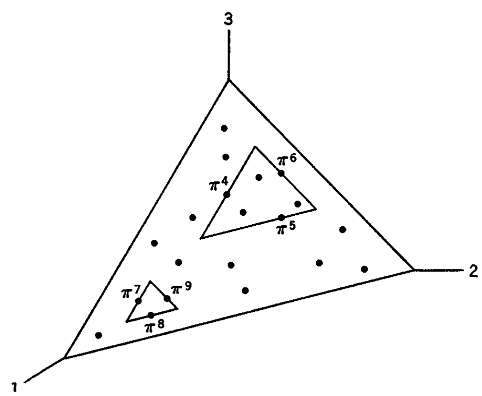

Figure 2

edges are given by the coordinate hyperplanes $\pi_i = 0$. This can be accomplished by the device of defining n new vectors

$$\begin{aligned}\pi^1 &= (0, M_1, \ldots, M_1)\\ \pi^2 &= (M_2, 0, \ldots, M_2)\\ &\vdots\\ \pi^n &= (M_n, M_n, \ldots, 0)\end{aligned}$$

with M_i different from each other and greater than one. The vectors, which are not on the simplex, are associated with the n slack variables in the matrix B. If we now consider n vectors $\pi^{j_1}, \ldots, \pi^{j_n}$ from the extended list $\pi^1, \ldots, \pi^k$, and define the associated subsimplex as before to be the set of $\pi = (\pi_1, \ldots, \pi_n)$ with

$$\begin{aligned}\pi_1 &\geq \min(\pi_1^{j_1}, \ldots, \pi_1^{j_n})\\ &\vdots\\ \pi_n &\geq \min(\pi_n^{j_1}, \ldots, \pi_n^{j_n})\end{aligned}$$

and

$$\pi_1 + \cdots + \pi_n = 1$$

then this subsimplex will be bounded by an edge $\pi_i = 0$ if π^i is one of the first n vectors, but otherwise the definition of the subsimplex will be as before.

In the following figure the subsimplex is generated by π^2, π^4, and π^5, and since there are no vectors in the list interior to this subsimplex

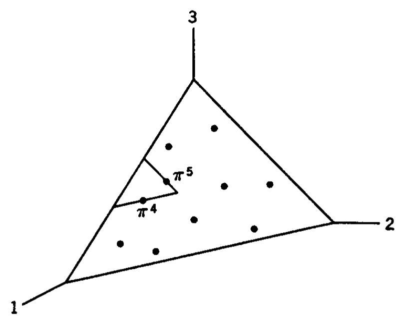

Figure 3

we are justified in saying that the vectors π^4 and π^5 are close to each other and have a second coordinate close to zero.

The following formal definition makes use of a term "primitive set," which I have used elsewhere, to describe the concept of closeness under discussion.

Definition. A set of n vectors $\pi^{j_1}, \ldots, \pi^{j_n}$ from the list $\pi^1, \ldots, \pi^k$ will be said to form a *primitive set* if no vectors in the list are interior to the subsimplex

$$\begin{aligned}
\pi_1 &\geq \min(\pi_1^{j_1}, \ldots, \pi_1^{j_n}) \\
&\;\;\vdots \\
\pi_n &\geq \min(\pi_n^{j_1}, \ldots, \pi_n^{j_n}) \\
\pi_1 + \cdots + \pi_n &= 1
\end{aligned}$$

The main theorem of this paper, which will be demonstrated by means of a constructive algorithm, follows.

Theorem. There exists a primitive set $\pi^{j_1}, \ldots, \pi^{j_n}$, such that the equations

$$Bx = w$$

have a nonnegative solution with $x_j = 0$ if j is different from $(j_1, \ldots, j_n)$.

The algorithm behind this theorem will provide us with precisely the right type of solution to the problem discussed in Section 2. It will yield a nonnegative solution to $Bx = w$, with all of the π^j corresponding to positive x_j or y_j close to each other, and with the ith coordinate of each of these prices close to zero if the ith slack is positive.

4. The Algorithm

The reason for introducing the notion of a primitive set of vectors is not only to define specifically when n vectors are to be considered close—many other constructions would serve just as well for this—but also because a type of operation, similar to a pivot step in linear programming, can be performed on a primitive set of vectors, and this operation is crucial for the development of our algorithm.

The operation consists of removing a specific vector from a primitive set of vectors and attempting to replace it by some other vector so that the new set of vectors is also a primitive set. As the following lemma indicates, this operation can, with one exception, always be performed and the replacement is uniquely determined.

Lemma. Let $\pi^{j_1}, \ldots, \pi^{j_n}$ be a primitive set and π^{j_α} a specific one of these vectors. Then, aside from one exceptional case, there is a *unique* vector π^j in the list $\pi^1, \ldots, \pi^k$, so that if π^j replaces π^{j_α}, the resulting collection of vectors forms a primitive set. The exceptional case arises when all of the vectors π^{j_i} with $i \neq \alpha$ are from the first n vectors in the list, and in this case no replacement is possible.

The vector π^j that replaces π^{j_α} may be found by a simple geometrical construction. To illustrate this construction let us assume that

$$\pi_i^{j_i} = \min(\pi_i^{j_1}, \ldots, \pi_i^{j_n})$$

so that π^{j_i} is on that face of the subsimplex on which the ith coordinate is constant. Assume moreover that π^{j_1} is being removed.

Let $\pi^{j_{i^*}}$ be that vector in the primitive set with the *second* smallest value of its first coordinate. To find the vector to replace π^{j_1} we move the face containing $\pi^{j_{i^*}}$ parallel to itself, lowering the i^*th coordinate

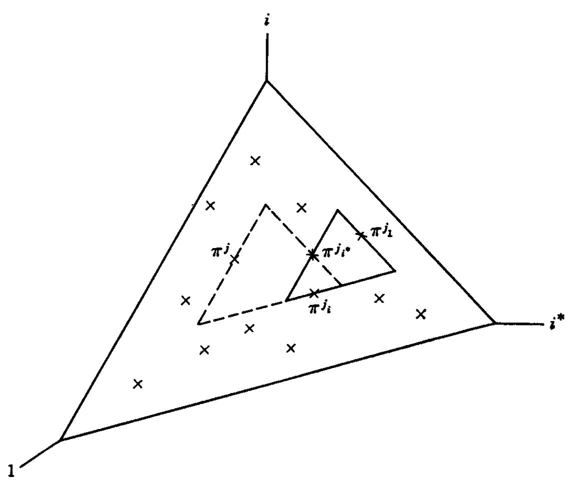

Figure 4

until we *first* intersect a vector π^j in the list with

$$\pi_i^j > \pi_i^{j_i} \qquad \text{for} \quad i \neq 1, i^*$$

and

$$\pi_1^j > \pi_1^{j_{i^*}}$$

or the face of the simplex S in which $\pi_{i^*} = 0$.

The rule is applicable except when the vectors π^{j_i} with $i \neq 1$ are all selected from the first n vectors of P_k, and it clearly produces a new primitive set. The details of the proof that π^j is the only possible replacement, and that no replacement is possible in the exceptional case, may be found in [6].

To see the analogy between this type of replacement and a pivot step in linear programming, consider a system of linear equations in nonnegative variables, $Bx = w$, where

$$B = \begin{bmatrix} 1 & \cdots & 0 & b_{1,n+1} & \cdots & b_{1,k} \\ \cdot & & & \cdot & & \\ \cdot & & & \cdot & & \\ \cdot & & & \cdot & & \\ 0 & \cdots & 1 & b_{n,n+1} & \cdots & b_{n,k} \end{bmatrix}$$

A feasible basis for this system of equations is a collection of n linearly independent columns $j_1, \ldots, j_n$ from the matrix B, such that the equations $Bx = w$ have a nonnegative solution with $x_j = 0$ if j is different from $j_1, \ldots, j_n$.

In a pivot step, one takes a specific column outside of the basis and attempts to introduce it into the basis, while removing some column, so that the resulting collection of n columns is again a feasible basis. In linear programming one attempts to bring *into* the basis a specific column from outside, whereas with a primitive set one attempts to *remove* a column in the primitive set. If the set of nonnegative solutions to the equations $Bx = w$ forms a bounded set, then a pivot step can always be carried out, and if the problem is nondegenerate, in the sense used in linear programming, there is a *unique* column to be eliminated from the basis for the resulting collection of columns to form a feasible basis.

Our assumptions on the technology guarantee that the nonnegative solutions to $Bx = w$ form a bounded set when the matrix B is constructed as in Section 2. And nondegeneracy can be brought about by a small perturbation of w, or by using a lexicographic

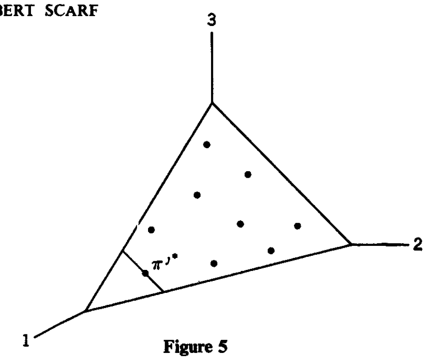

Figure 5

ordering, so that in our case a pivot step on the matrix B is always possible and unique.

The main theorem of the previous section can now be restated in a more specific and useful form.

Theorem (*Restatement*). There exists a primitive set $\pi^{j_1}, \ldots, \pi^{j_n}$, so that the columns $j_1, \ldots, j_n$ form a feasible basis for $Bx = w$.

Let us now turn to a proof of this theorem. Consider the set of vectors $(\pi^2, \ldots, \pi^n, \pi^{j^*})$ with π^{j^*} selected from those vectors beyond the first n so as to maximize its *first* coordinate. This collection forms a primitive set, as the above figure clearly indicates.

Moreover, the columns $1, 2, \ldots, n$ form a feasible basis for the matrix B since $w > 0$. Let us perform a pivot step on B by introducing column j^*. If column 1 is removed from the basis the problem is over since $(2, \ldots, n, j^*)$ would be both a primitive set and a feasible basis for $Bx = w$. Generally, this will not be the case and some column other than the first will be removed when j^* is introduced. The next step in the algorithm is to remove from the primitive set that price which corresponds to the column just removed from the feasible basis for $Bx = w$.

The algorithm alternates between pivot steps on the B matrix and the analogous operation on the primitive sets. We take into the feasible basis for B the column corresponding to that price just taken into the primitive set, followed by removing that price from the primitive set which corresponds to the column just removed from the basis for B.

It is easy to see that in any intermediary step in the computation, we shall be in a situation in which the feasible basis for B consists of column 1 and $n-1$ other columns $j_2, \ldots, j_n$, whereas the primitive set consists of vectors $\pi^{j_1}, \pi^{j_2}, \ldots, \pi^{j_n}$ with $j_1 \neq 1$. The algorithm retains this relationship in which $n-1$ of the indices are identical by performing either of *two* possible operations. If we are pivoting on the matrix B, then column j_1 must be introduced, and if we are replacing an element in the primitive set, then π^{j_1} must be removed. Except for the initial position where the primitive set is composed of the vectors $(\pi^2, \ldots, \pi^n, \pi^{j^*})$ and the feasible basis consists of the first n columns, both of these operations can be performed; in the initial position only one operation can be taken since removing π^{j^*} is the exceptional case referred to in the lemma.

In general, some step has been taken to arrive at the present state. The algorithm then takes that other continuation open to it. The algorithm cannot cycle, since if the first state that it returns to is not the initial position, there would have to be three ways to exit from this position rather than two, and if the first position which is repeated is the initial position, there would be two ways to exit from this position rather than one.

Since the algorithm does not cycle and there are a finite number of possible positions, the algorithm must terminate, and this can only happen when the prices corresponding to a feasible basis for $Bx = w$ also form a primitive set. This concludes the proof of the main theorem.

5. Some Examples

The algorithm has been programmed for an IBM 7094, and a number of examples have been tried. Before describing the results of a sample computation, it might be useful to indicate two of the special techniques that have been incorporated into the program.

In order to use the algorithm, some specific set of vectors $\pi^{n+1}, \ldots, \pi^k$ must be selected. I have found it convenient to form this list by taking all vectors on the simplex $\left\{\pi \mid \pi_i \geq 0, \ \sum_1^n \pi_i = 1\right\}$ whose coordinates are positive fractions with a given denominator. In other words, a denominator D is selected and we consider all vectors

$$\pi = \left(\frac{k_1}{D}, \ldots, \frac{k_n}{D}\right)$$

with k_i positive integers such that $k_1 + \cdots + k_n = D$.

If the list of vectors has this special structure, the operation of replacing a vector in a primitive set can be carried out by a simple algebraic computation and does not involve a lengthy search through all of the vectors in the list. Of course, the set $\pi^{n+1}, \ldots, \pi^k$ will not satisfy the nondegeneracy assumption made in Section 3, since many vectors in the list have a common coordinate. There are a variety of techniques, however, for resolving degeneracy and insuring that the algorithm does not cycle.

The algorithm terminates with a primitive set $\pi^{j_1}, \ldots, \pi^{j_n}$ such that the columns $j_1, \ldots, j_n$ are a feasible basis for the equations $Bx = w$, and, as we have seen, an average of the vectors in the primitive set will serve as an approximation to an equilibrium price vector. To determine the specific weights to be used in forming an average, I have assumed that the demand functions are locally linear in the neighborhood of the primitive set and selected that vector which minimizes the maximum deviation between demand and supply. The efficiency of the algorithm is substantially increased by a device of this sort.

Let us consider, as a numerical example, with no pretense towards realism, an economy involving the following six commodities:

1. Capital available at the end of the current period.
2. Capital available at the beginning of the current period.
3. Skilled labor.
4. Unskilled labor.
5. Nondurable consumer goods.
6. Durable consumer goods.

During the particular time period, production may be carried out in each of three sectors: the construction of durable consumer goods, the production of nondurable consumer goods, and a sector for the construction of new capital available at the end of the period.

The durable consumer goods sector is assumed to be described by the two activities

$$\begin{array}{cc} 4 & 4 \\ -5.3 & -5 \\ -2 & -1 \\ -1 & -6 \\ 0 & 0 \\ 4 & 3.5 \end{array}$$

with the commodities in the order given. The first of these two activities represents a process which produces four units of durable consumer goods and uses five and three-tenths units of capital, two units of skilled labor, and one unit of unskilled labor. In addition, the five and three-tenths units of capital are partially depreciated during use and become four units of capital available at the end of the period. The second activity in this sector permits the substitution of unskilled labor for skilled labor.

There are three possible activities in the nondurable sector:

$$\begin{array}{ccc} 1.6 & 1.6 & 1.6 \\ -2 & -2 & -2 \\ -2 & -4 & -1 \\ -3 & -1 & -8 \\ 6 & 8 & 7 \\ 0 & 0 & 0 \end{array}$$

with varying degrees of substitution between skilled and unskilled labor.

Finally, the capital good sector involves the following three activities:

$$\begin{array}{ccc} 0.9 & 7 & 8 \\ -1 & -4 & -5 \\ 0 & -3 & -2 \\ 0 & -1 & -8 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array}$$

the first of which represents the rate of capital depreciation if no investment is undertaken.

In addition to this activity analysis model of production, our hypothetical economy will involve five consumers, each of whom has a distinct set of demand functions and vector of initial assets. The following matrix describes the initial assets of each consumer:

	C. End	C. Beg.	Skil. L.	Unskil. L.	Nondur.	Dur.
Consumer 1	0	3	5	0.1	0	1
Consumer 2	0	0.1	0.1	7	0	2
Consumer 3	0	2	6	0.1	0	1.5
Consumer 4	0	1	0.1	8	0	1
Consumer 5	0	6	0.1	0.5	0	2

As we see, no consumer owns, prior to production, either non-durable goods or capital available at the end of the period. Consumer 5 is the largest owner of capital at the beginning of the period, and there are varying degrees of ownership of the two varieties of labor and of consumer durables.

I have assumed that each consumer has a set of demand functions derivable from a utility function with constant elasticity of substitution. This implies that at the prices $\pi_1, \ldots, \pi_6$, the ith consumer will make the demands

$$\xi_j(\pi) = \frac{a_{ij} \cdot f_i(\pi_1, \ldots, \pi_6)}{\pi_j^{b_i}}$$

where b_i is the elasticity of substitution for consumer i, a_{ij} measures the intensity of the ith consumer's demand for commodity j, and $f_i(\pi_1, \ldots, \pi_6)$ is a complex function of the price vector π selected so that the budget constraint is satisfied for each individual. The specific values of a_{ij} are given by the following matrix:

	C. End	C. Beg.	Skil. L.	Unskil. L.	Nondur.	Dur.
Consumer 1	4	0	0.2	0	2	3.2
Consumer 2	0.4	0	0	0.6	4	1
Consumer 3	2	0	0.5	0	2	1.5
Consumer 4	5	0	0	0.2	5	4.5
Consumer 5	3	0	0	0.2	4	2

As we see, no consumer has a demand for capital at the beginning of the period, but there may be a substantial demand, depending of course on the prices, for capital at the end of the period. Since there is no explicit description of production after the end of the time period, this demand is to be interpreted as a demand for savings. The entries under the skilled and unskilled labor columns refer to a demand for leisure. Finally, the elasticities of substitution b_i are given by

Consumer	b_i
1	1.2
2	1.6
3	0.8
4	0.5
5	0.6

In the numerical solution of this example, the set $\pi^{n+1}, \ldots, \pi^k$ was assumed to consist of all vectors of the form

$$\frac{k_1}{100}, \frac{k_2}{100}, \ldots, \frac{k_6}{100}$$

with k_i positive integers summing to 100. There is an exceptionally large number of these vectors. The algorithm terminated after examining only 913 of them in a little over a minute of 7094 computing time, with the following primitive set:

π^{j_1}	π^{j_2}	π^{j_3}	π^{j_4}	π^{j_5}	π^{j_6}
22	22	22	22	22	23
22	21	22	22	24	22
20	19	19	19	19	20
7	7	7	6	6	7
12	12	12	12	11	12
19	19	18	19	18	16

These six vectors are related, one by one, to six columns in the matrix B, which form a feasible basis for the equations $Bx = w$. The first four of these vectors are associated with the 9th, 7th, 10th, and 11th activities, in the order in which they have been described. The vector π^{j_5} gives rise to a negative profit for all possible activities, and therefore corresponds, in the matrix B, to a column of demands. π^6 is associated with the thirteenth activity.

These six vectors were averaged by a set of weights obtained as the solution of a specific linear program, and the following price vector and activity levels were obtained:

$$\pi = (21.8, 21.8, 19.4, 7.4, 12.2, 17.4)$$

Activity	Level	Profit
7	0.86	−0.05
8	0.0	−0.25
9	0.10	0.03
10	1.41	0.04
11	1.31	−0.02
12	0.0	−0.02
13	0.47	0.00
14	0.0	−0.33

The profits in the final column are based on the price vector π normalized so that its sum is one.

As a final summary, let us compare the market demand at this set of prices with the net supply obtained by using these activity levels in conjunction with the initial stocks of commodities.

	C. End	C. Beg.	Skil. Lab.	Unsk. Lab.	Nondur.	Dur.
Demand	11.27	0.00	1.02	2.17	21.08	10.98
Supply	11.27	−0.01	1.01	2.14	21.06	10.96

The price vector π and the activity levels given seem to be a fine approximation to an equilibrium for the equality of supply and demand in each market. The profits, which should be zero for those activities in use and less than or equal to zero for the remaining activities, seem a bit less satisfactory. This is undoubtedly due to the preoccupation of the final linear programming problem with minimizing the maximum deviation between supply and demand, a goal which is not directly responsive to considerations of profit. Many other averaging processes can be used, and they deserve to be explored before substantially larger problems are tried. It should be pointed out that the final linear programming problem which takes no more than one or two seconds of computing time, is a very minor part of the algorithm. The important work in the algorithm is done in determining the primitive set whose associated columns form a feasible basis for $Bx = w$. It is this calculation that indicates the neighborhood in which approximate equilibrium prices are to be found.

In examining the preceding example, one sees that the price of capital available today is identical with the price to be paid today for capital delivered tomorrow, so that the real rate of interest should naturally be taken as zero. This is reflected in the fact that the initial stock of capital falls during the period from 12.1 units to 11.3 units, even though activity 13, a capital producing activity, is used at the level of 0.47.

Let us compare this model with one which differs from it by the introduction of one new productive activity in the capital goods sector. The activity

$$\begin{matrix} 6.4 \\ -3.5 \\ -1 \\ -5 \\ 0 \\ 0 \end{matrix}$$

has a profit of 0.13 at the previous equilibrium prices, normalized so that their sum is one, so that if this activity is available it will surely be used. It would also seem reasonable to suspect that the use of this activity would have a tendency to increase the interest rate from its previous level of zero.

If we adopt the same grid of prices on the simplex, the calculations for this model terminate after the examination of 1185 price vectors in about one minute and twenty seconds of 7094 computing time. After averaging, the following price vectors and activity levels are obtained:

$$\pi = (18.8, 22.0, 19.6, 7.1, 13.4, 19.1)$$

Activity	Level	Profit
7	0.69	−0.11
8	0.0	−0.30
9	0.0	0.06
10	1.79	0.08
11	0.76	0.04
12	0.0	−0.05
13	0.0	−0.22
14	0.0	−0.56
15	0.95	−0.12

The relation between demand and net supply is given by the table

	C. End	C. Beg.	Skil. L.	Unsk. L.	Nondur.	Dur.
Demand	12.98	0.00	1.03	2.41	19.73	10.29
Supply	12.93	−0.01	1.02	2.40	19.68	10.28

The new activity is used at the expense of activity 11, in which a substantial quantity of unskilled labor was required to produce nondurable goods. As might be imagined, the price of nondurable goods rises and its consumption falls. The expected rise in the interest rate takes place, along with an increase in savings.

These examples give an indication of the speed and accuracy with which the algorithm can solve a moderately difficult problem. I feel quite sure that the performance of the algorithm can be substantially improved in both of these dimensions by more subtle programming techniques, and that eventually problems involving as many as twenty commodities will be feasible without an excessive use of computing time.

REFERENCES

1. Debreu, Gerard, *The Theory of Value*, Wiley 1954.
2. Lemke, C. E. and J. T. Howson, "Equilibrium Points of Bi-Matrix Games," *SIAM Journal*, Vol. 12, July, 1964.
3. McKenzie, L. W., "On the Existence of General Equilibrium for a Competitive Model," *Econometrica*, Vol. 27, pp. 54–71.
4. Mantel, Rolf, "Toward a Constructive Proof of the Existence of Equilibrium in a Competitive Economy," Ph.D. dissertation, Yale Univ.
5. Scarf, H., "The Core of an *N* Person Game," *Econometrica*, January 1967.
6. Scarf, H. "The Approximation of Fixed Points of a Continuous Mapping," *SIAM Journal on Applied Math.*, September, 1967.