Cowles Foundation Paper 308
Reprinted from
American Economic Review, 59(4)(1), 1969

An Example of an Algorithm for Calculating General Equilibrium Prices

By HERBERT SCARF*

During the past several years a number of writers have been involved in the development of an algorithm applicable to a class of problems in mathematical economics, not previously treated from a computational point of view. Perhaps the most interesting and potentially useful of these applications is to the calculation of equilibrium prices for a general Walrasian model of a competitive economy.

The version of the algorithm described in the present paper is a specialization due to T. Hansen [1], [2] of the more general but computationally less useful approach introduced in [4] and [5]. Hansen's algorithm was independently discovered by H. W. Kuhn [3], who also provided a illuminating geometrical interpretation, previously not noticed by either Hansen or myself.

The algorithm itself is not particularly subtle, and for the most part involves operations to which we have become accustomed in the solution of linear programming problems by means of the simplex method. There are, however, some novel features which may be obscured in a mathematical treatment, and which the general reader may most easily comprehend by means of a simple numerical example.

The present paper will describe in detail an application of the algorithm to an example with only three sectors. This small problem has been selected for illustrative purposes only; any practical application of these techniques will necessarily involve a substantially larger number of commodities.

A considerable body of computational experience with larger models has already been gathered. Over one hundred examples have been tested, ranging from three to twenty sectors. The computation time, which is dependant on the number of sectors, has never exceeded five minutes on an IBM 7094, and in most cases is substantially smaller. Given the increase in speed of the newer computers, and the reduction in computation time achievable by adroit modifications of the basic algorithm, it seems clear that problems involving as many as 30 or 40 sectors will eventually be feasible, should they be justified on intellectual grounds.

I. The Example

A general equilibrium model requires for its specification, a description of the productive technology available to the economy, demand functions of the consuming units, and the distribution of ownership of the real resources in the economy among these units. In our example the technology will be described by the following activity analysis matrix:

$$A = \begin{bmatrix} -1 & 0 & 0 & 4 & 4 & 4 & 0 \\ 0 & -1 & 0 & -8 & -6 & -4 & -2.4 \\ 0 & 0 & -1 & -1 & -2 & -3 & 1 \end{bmatrix} \begin{matrix} \text{Consumer Goods} \\ \text{Labor} \\ \text{Capital} \end{matrix}$$

in which inputs are represented by negative entries and outputs by positive entries. The first three activities permit the free disposal of consumer goods, labor and capital, respectively, and are incorporated in the model to allow for the possibility of a zero price at equilibrium. Activities 4, 5 and 6 represent three distinct techniques which utilize labor and capital to produce consumer goods. In each of these activities the third entry must be interpreted as the *decrease* in capital stock caused by depreciation and wear, if the activity is run at a unit level. The seventh activity describes a net increase in capital stock as a consequence of investment. The seven activities can be used simultaneously, at arbitrary nonnegative levels.

My own preference, even in a single period model, would be to consider the capital stock

* Cowles Foundation. The research described in this paper was carried out under grants from the National Science Foundation and from the Ford Foundation.

at the beginning of the period as a commodity distinct from that available at the end of the period. Each activity would then reduce the capital stock at the beginning of the period and produce, as a joint output, a depreciated stock of capital at the end of the the period. Aside from recognizing explicitly that production takes time, this approach leads naturally to a fully dynamic model in which the own rate of interest on capital would be determined by the equilibrium price calculation.

In our example we shall assume that there are two types of consumers, appearing in equal numbers in the economy, and differentiated by their ownership of resources and demand functions. The following table describes the patterns of ownership of each type.

TABLE 1—OWNERSHIP OF RESOURCES

	Consumer Goods	Labor	Capital
Type 1	0	10	8
Type 2	0	10	1

Each of the consumers has 10 units of potential labor, some of which will be engaged in production, and the remainder consumed as leisure. If the relative prices in the three sectors are given by π_1, π_2 and π_3, for consumer goods, labor and capital respectively, the potential wealth of the first consumer is $10\pi_2+8\pi_3$, and that of the second consumer $10\pi_2+\pi_3$.

The algorithm places no restriction on the individual demand functions other than the customary assumptions of homogeneity of degree zero in all relative prices, continuity, and satisfaction of the budget constraint. In the interest of simplicity of exposition, I shall assume that both individuals spend a constant proportion of their potential wealth on each commodity, independently of the relative prices, with the proportion given by:

TABLE 2—PERCENTAGES OF WEALTH SPENT IN EACH CATEGORY

	Consumer Goods	Leisure	Capital
Type 1	25	10	65
Type 2	60	20	20

These demand functions, which are derivable from utility functions having a Cobb-Douglas form, can be generalized to include income effects and more elaborate sensitivities to prices.

It should be understood that capital appears in the consumer demand functions as a proxy for savings. This assumes that investment activity is motivated entirely by private saving decisions, and does not reflect the eventual profitability of newly produced capital. A substantial improvement would be obtained by extending the example over several periods, and including markets for intermediary goods.

II. *Equilibrium Prices and Activity Levels*

Before discussing the algorithm in detail we shall describe the answer to our specific example. The equilibrium prices are given by

$$\pi = (13/30,\ 5/30,\ 12/30),$$

normalized arbitrarily so that their sum is unity. Of the seven activities, all are run at a zero level with the exception of activity number four, which operates at a level of 1.42, and activity number seven, at a level of 1.36. In the following table the above prices are used to compute the profitability of each of the available techniques:

Activity	Level	Profit
4	1.42	0
5	0	−.067
6	0	−.133
7	1.36	0

As we see, all activities make a nonpositive profit at the equilibrium prices, and those in use make a profit equal to zero. The total supply is given by the stock of initially owned commodities, minus the factors used up in production, and augmented by the outputs of production,

$$\begin{bmatrix} 0 \\ 20 \\ 9 \end{bmatrix} + 1.42\begin{bmatrix} 4 \\ -8 \\ -1 \end{bmatrix} + 1.36\begin{bmatrix} 0 \\ -2.4 \\ 1 \end{bmatrix} = \begin{bmatrix} 5.67 \\ 5.40 \\ 8.94 \end{bmatrix}.$$

At the above prices the consumers' demands may be shown to be

	Consumer Goods	Leisure	Capital
Type 1	2.808	2.920	7.908
Type 2	2.862	2.480	1.033
Market Demand	5.670	5.400	8.941

so that for each market, demand is equal to supply, and the suggested price and activity levels are indeed in equilibrium.

III. *How Not To Solve the Problem*

Our example is somewhat deceptive in that the equilibrium prices and activity levels can be determined by a very simple procedure and the reader may erroneously suspect that this technique is capable of being generalized. The procedure is based on the observation that the equilibrium price vector is determined up to a scale factor (aside from exceptional cases) when it is known which two activities are to be operated at a positive level.

If activities 4 and 7 appear in equilibrium, the competitive assumption that each of these activities earn a profit of zero, yields the pair of equations

$$4\pi_1 - 8\pi_2 - \pi_3 = 0$$
$$-2.4\pi_2 + \pi_3 = 0,$$

which have as a unique solution, (13/30, 5/30, 12/30), normalized to sum to unity. The market demand functions can be evaluated at this price vector, and the appropriate levels of activities 4 and 7 then determined so as to equate supply and demand.

If the wrong pair of activities is selected, this procedure may not work for one of several reasons: the price vector determined by the zero profit conditions may have several negative components; some of the remaining activities may make a positive profit; and finally it may be impossible to equate supply and demand by using this pair of activities alone. This does however suggest that we may base an algorithm upon the systematic examination of all pairs of activities (and in the general case with n sectors, all subsets of n-1 activities), until one is found satisfying all three of the above conditions.

It is easy to see however that this approach, reminiscent of the simplex method for linear programming, cannot be successful in general. If, in our example, the seventh activity is replaced by one in which a large quantity of labor is needed to produce a single unit of new capital, then the competitive equilibrium will require only one activity, the fourth, to be operated at a positive level. It will therefore be impossible to determine relative prices by the zero profitability conditions alone. This phenomenon is quite general and indicates that a completely different type of algorithm is required for the solution of this problem.

The conventional price adjustment mechanism, in which prices are modified in proportion to the excess demand for the corresponding commodity, may also be conceived of as an algorithm for the computation of equilibrium prices. Aside from the relatively minor problem of a non-unique supply response to a given price vector, the gradient method has the serious drawback that very stringent conditions on the demand functions such as "gross substitutability," are required for its convergence. While the demand functions of our particular example do exhibit gross substitutability, this occurs rarely and the gradient method is not of wide applicability.

Virtually any algorithm which can be shown to be successful in the general Walrasian model will, at the same time, be capable of providing a proof that equilibrium prices do exist. The existence problem has been recognized during the last two decades as one of the most difficult areas of mathematical economics, and requires the use of techniques known as "fixed point theorems." An effective algorithm must, therefore, be intimately related to this branch of mathematics, and involve considerations whose economic interpretation is not immediate.

IV. *Introducing the Algorithm*

Since the equilibrium prices are determined up to a scale factor it is sufficient to restrict our attention to those price vectors

$\pi = (\pi_1, \pi_2, \pi_3)$ which are nonnegative and sum to unity. We begin by selecting a positive integer D, and consider only those price vectors on the unit simplex, whose components are rational numbers with denominators equal to D. In any particular example, the accuracy of the computation will be improved by increasing the value of D, but at the same time the number of basic iterations and therefore the total computation time will be increased, roughly proportionately to D. A typical example with ten sectors might require a D of 150 or more, in order to obtain sufficient accuracy. The number of price vectors potentially under examination would then be exceptionally high, but the algorithm will typically terminate with fewer than two or three thousand iterations.

Let us begin our example by selecting D to be equal to 10, a number which is too small for serious computation, but which is sufficient to illustrate the algorithm.

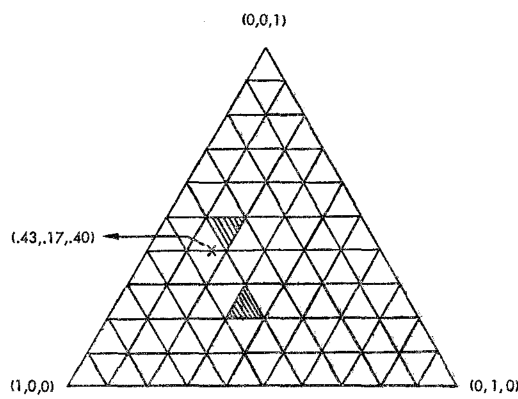

FIGURE 1

The vectors on the simplex with denominators equal to 10, are used to partition the simplex into the small triangles shown in the figure.

Each such price vector (π_1, π_2, π_3) will be associated with a specific commodity vector, which is either the vector of market demands at the set of prices (π_1, π_2, π_3) or the negative of a particular activity, according to the following rules:

1. If the price vector has a zero component, it will be associated with a "slack" vector containing a 1 in the place of the first zero price and 0's elsewhere. If all of the components are positive, rules 2, 3, and 4 are followed.

2. Examine all of the available activities to determine which yields the maximum profit at the price vector (π_1, π_2, π_3).

3. If this profit is greater than or equal to zero, then (π_1, π_2, π_3) is associated with a vector whose components are the *negatives* of the profit maximizing activity.

4. If the maximum profit at these prices is negative, then (π_1, π_2, π_3) is associated with the vector whose components are the market demands at these prices.

These rules, which may seem quite arbitrary to the reader, are of crucial importance to the algorithm. A few examples may be in order. At the prices (.8, .1, .1), the profits associated with activities 4 through 7 are given by

$$4(.8) - 8(.1) - 1(.1) = 2.3$$
$$4(.8) - 6(.1) - 2(.1) = 2.4$$
$$4(.8) - 4(.1) - 3(.1) = 2.5$$
$$-2.4(.1) + 1(.1) = -.14,$$

and since the maximum profit is positive (.8, .1, .1) is associated with (−4, 4, 3). On the other hand, the price vector (.4, .3, .3) yields as profits

$$4(.4) - 8(.3) - 1(.3) = -1.1$$
$$4(.4) - 6(.3) - 2(.3) = -.8$$
$$4(.4) - 4(.3) - 3(.3) = -.5$$
$$-2.4(.3) + 1(.3) = -.42,$$

and since all of these profits are negative the vector (.4, .3, .3) is associated with (8.325, 4.0, 13.9), which as the reader may verify, are the market demands at these prices.

In general, the algorithm will be confronted, at each iteration, not with a single price vector, but with *three* such vectors which form the vertices of one of the small triangles in the above figure. Consider, as an example, the lower of the two shaded triangles whose vertices are given by the three vectors

$$\begin{bmatrix} .4 \\ .3 \\ .3 \end{bmatrix}, \quad \begin{bmatrix} .4 \\ .4 \\ .2 \end{bmatrix}, \quad \begin{bmatrix} .5 \\ .3 \\ .2 \end{bmatrix}.$$

According to our rules, these vectors will be associated with

$$\begin{bmatrix} 8.325 \\ 4.0 \\ 13.9 \end{bmatrix}, \quad \begin{bmatrix} 9.8 \\ 3.5 \\ 22.4 \end{bmatrix}, \quad \begin{bmatrix} -4 \\ 4 \\ 3 \end{bmatrix}$$

respectively. The first two price vectors correspond to columns of market demands, and the third to the negative of activity number 6.

At each iteration the algorithm will ask whether there is a nonnegative linear combination of the three associated columns equal to the vector of initially owned factors, prior to production. In the present triangle the question is whether the equations

$$y_1 \begin{bmatrix} 8.325 \\ 4.0 \\ 13.9 \end{bmatrix} + y_2 \begin{bmatrix} 9.8 \\ 3.5 \\ 22.4 \end{bmatrix} + y_3 \begin{bmatrix} -4 \\ 4 \\ 3 \end{bmatrix} = \begin{bmatrix} 0 \\ 20 \\ 9 \end{bmatrix}$$

have a solution with $y_1, y_2, y_3 \geq 0$, which is not possible for this triangle since the unique solution is given by $(4.81, -2.96, 2.77)$.

In order to provide a triangle for which the answer to this question is affirmative, and to show the relationship between this question and the determination of equilibrium prices, consider the sub-simplex whose vertices are

$$\begin{bmatrix} .3 \\ .2 \\ .5 \end{bmatrix}, \quad \begin{bmatrix} .4 \\ .1 \\ .5 \end{bmatrix}, \quad \begin{bmatrix} .4 \\ .2 \\ .4 \end{bmatrix},$$

the upper shaded triangle in the above figure. The vectors associated with these three vertices are given by

$$\begin{bmatrix} .0 \\ 2.4 \\ -1.0 \end{bmatrix} \quad \begin{bmatrix} -4 \\ 8 \\ 1 \end{bmatrix} \quad \begin{bmatrix} 6.85 \\ 5.0 \\ 9.65 \end{bmatrix},$$

and the equations

$$\begin{bmatrix} .0 \\ 2.4 \\ -1.0 \end{bmatrix} y_1 + \begin{bmatrix} -4 \\ 8 \\ 1 \end{bmatrix} y_2 + \begin{bmatrix} 6.85 \\ 5.0 \\ 9.65 \end{bmatrix} y_3 = \begin{bmatrix} 0 \\ 20 \\ 9 \end{bmatrix}$$

have as a solution (1.24, 1.54, .90). which is nonnegative.

The sole purpose of the algorithm is to produce a triangle with this particular property, which can immediately be used to provide an approximate equilibrium price vector. To see this, let us rewrite the above equations as

$$.9 \begin{bmatrix} 6.85 \\ 5.0 \\ 9.65 \end{bmatrix} = \begin{bmatrix} 0 \\ 20 \\ 9 \end{bmatrix} + 1.54 \begin{bmatrix} 4 \\ -8 \\ -1 \end{bmatrix} + 1.24 \begin{bmatrix} 0 \\ -2.4 \\ 1 \end{bmatrix}.$$

The right hand side of this equality is the net supply if the 4th activity is operated at a level of 1.54, and the 7th activity at a level of 1.24. Aside from the factor of .9, the left hand side represents the market demand at the price vector (.4, .2, .4), so that these equations describe a situation of approximate equality between supply and demand in all markets. As the grid size D is increased, the factor of .9 will become closer to 1.0 and the approximation will become increasingly accurate. (In the general case in which more than one vertex of the subsimplex corresponds to a column of demands, it is the sum of the weights associated with these columns which tends to unity with increasing D.)

But we can say even more. The three price vectors, being the vertices of a small triangle are quite close to each other, and will be even closer as D is increased. According to the rules used to associate columns with vertices, one of these price vectors yields a profit less than zero for all activities, and the others provide a nonnegative profit for activities 4 and 7 respectively. Any price vector in the neighborhood of these three can therefore serve as an approximate equilibrium price vector, since in addition to permitting the approximate equality of supply and demand, it will yield profits approximately zero for the activities in use, and profits which are either negative or quite small for all of the activities. As the figure indicates, the true equilibrium prices are quite close to this triangle.

Before proceeding to a discussion of the algorithm which produces such a subsimplex whose associated columns can be combined, with nonnegative weights, so as to equal the vector of initial holdings, let us see how the approximation can be improved by increasing D from 10 to 100.

With this fine a grid the division of the simplex is awkward to reproduce in a figure, but the algorithm of the next section terminates with a sub-simplex whose vertices are

$$\begin{bmatrix} .42 \\ .17 \\ .41 \end{bmatrix}, \quad \begin{bmatrix} .43 \\ .16 \\ .41 \end{bmatrix}, \quad \begin{bmatrix} .43 \\ .17 \\ .40 \end{bmatrix},$$

and which are associated with the columns

$$\begin{bmatrix} 0 \\ 2.4 \\ -1.0 \end{bmatrix}, \quad \begin{bmatrix} -4 \\ 8 \\ 1 \end{bmatrix}, \quad \begin{bmatrix} 5.779 \\ 5.353 \\ 9.013 \end{bmatrix}.$$

Solving the resulting equations, we obtain

$$.99\begin{bmatrix} 5.779 \\ 5.353 \\ 9.013 \end{bmatrix} = \begin{bmatrix} 0 \\ 20 \\ 9 \end{bmatrix} + 1.43\begin{bmatrix} 4 \\ -8 \\ -1 \end{bmatrix} + 1.35\begin{bmatrix} 0 \\ -2.4 \\ 1 \end{bmatrix},$$

a very close approximation to the actual equilibrium price and activity levels.

In a problem involving a larger number of commodities, this degree of accuracy is usually obtained by treating the demand functions as locally linear in the neighborhood of the final subsimplex, and then solving a linear programming problem whose objective is to minimize some measure of the distance between supply and demand. A terminal linear programming step is much less expensive in terms of computer time, than increasing the grid size to obtain additional accuracy.

V. *The Algorithm for Determining the Appropriate Sub-Simplex*

As the following figure indicates, the algorithm begins at a triangle located at a vertex of the large simplex, say the vertex (1, 0, 0), and at each iteration moves to an adjacent triangle obtained by eliminating one of the three vertices. The specific sequence of triangles is completely determined by the decisions as to which of the three (and in the general case, n) vertices is to be eliminated at each iteration.

At each stage of the algorithm a record is kept of the specific vector which has just been introduced. For example, in the 10th triangle whose vertices are given by

$$\begin{bmatrix} .5 \\ .1 \\ .4 \end{bmatrix} \quad \begin{bmatrix} .5 \\ .2 \\ .3 \end{bmatrix} \quad \begin{bmatrix} .4 \\ .2 \\ .4 \end{bmatrix},$$

the vector $(.4, .2, .4)'$ has just been introduced as a new vertex.

The algorithm is attempting to determine a triangle whose three associated columns give rise to a system of linear equations with a unique nonnegative solution. While it is difficult to find such a triangle directly, it is quite easy to locate a triangle in which these conditions are almost fulfilled, in the sense that the equations based on *two* of the associated columns and the *first* slack vector have a unique nonnegative solution. For example, triangle 2 has this property since the columns

$$\begin{bmatrix} .9 \\ 0 \\ .1 \end{bmatrix} \quad \begin{bmatrix} .9 \\ .1 \\ 0 \end{bmatrix} \quad \begin{bmatrix} .8 \\ .1 \\ .1 \end{bmatrix}$$

are associated by virtue of Rule 1 with

$$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \begin{bmatrix} -4 \\ 4 \\ 3 \end{bmatrix}$$

and the equations

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} y_1 + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} y_2 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} y_3 = \begin{bmatrix} 0 \\ 20 \\ 9 \end{bmatrix},$$

have a trivial nonnegative solution. It is for this reason that the algorithm begins at a vertex of the large simplex.

Only those triangles possessing this quite specific property will ever appear in an intermediary stage of the algorithm. The two vertices of the triangle, neither of which have just been introduced, will have two columns, either market demands or the negative of an activity level, associated with them. For the triangle to appear, it is necessary that there be one, and only one, nonnegative linear combination of these *two* associated columns, and of the first *slack vector* $(1, 0, 0)'$ equal to the total supply prior to production.

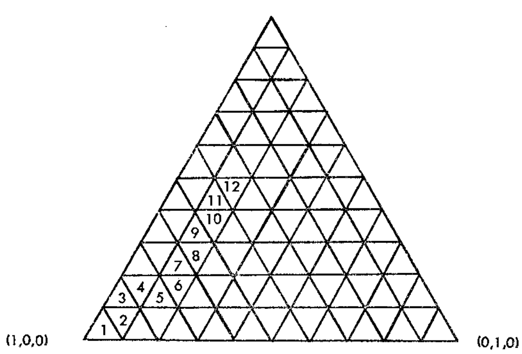

FIGURE 2

In triangle 10, for example, the vertices

$$\begin{bmatrix} .5 \\ .1 \\ .4 \end{bmatrix} \quad \begin{bmatrix} .5 \\ .2 \\ .3 \end{bmatrix} \quad \begin{bmatrix} .4 \\ .2 \\ .4 \end{bmatrix}$$

are associated with

$$\begin{bmatrix} -4 \\ 8 \\ 1 \end{bmatrix} \quad \begin{bmatrix} -4 \\ 4 \\ 3 \end{bmatrix} \quad \begin{bmatrix} 6.85 \\ 5.00 \\ 9.65 \end{bmatrix}$$

and we are therefore insisting that the equations

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} y_1 + \begin{bmatrix} -4 \\ 8 \\ 1 \end{bmatrix} y_2 + \begin{bmatrix} -4 \\ 4 \\ 3 \end{bmatrix} y_3 = \begin{bmatrix} 0 \\ 20 \\ 9 \end{bmatrix},$$

have a unique non-negative solution, which in fact is given by $(y_1, y_2, y_3) = (76/5, 6/5, 13/5)$.

Triangle 10 is obtained by following the sequence illustrated in Figure 2. If we wish to proceed from this triangle to a new triangle, then the decision as to which of the two older vertices is to be eliminated is uniquely determined if we desire the new triangle to have the same specific property. The equality

$$\frac{76}{5}\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + \frac{6}{5}\begin{bmatrix} -4 \\ 8 \\ 1 \end{bmatrix} + \frac{13}{5}\begin{bmatrix} -4 \\ 4 \\ 3 \end{bmatrix} = \begin{bmatrix} 0 \\ 20 \\ 9 \end{bmatrix},$$

may be considered as representing a feasible basis for a linear programming problem, and if the column $(6.85, 5.00, 9.65)'$ is introduced into this basis the conventional pivot operation will remove $(-4, 4, 3)'$ and yield

$$3.37\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + 2.05\begin{bmatrix} -4 \\ 8 \\ 1 \end{bmatrix} + .72\begin{bmatrix} 6.85 \\ 5.00 \\ 9.65 \end{bmatrix} = \begin{bmatrix} 0 \\ 20 \\ 9 \end{bmatrix}.$$

Therefore the vertex (.5, .2, .3)′, associated with the column (−4, 4, 3)′, is to be eliminated from triangle 10. We obtain triangle 11, whose new vertex is given by (.4, .1, .5)′ as can be seen from Figure 2. A version of this pivot operation occurs at every iteration of the algorithm.

Let us proceed for several additional iterations. The vertices of triangle 11

$$\begin{bmatrix} .5 \\ .1 \\ .4 \end{bmatrix} \begin{bmatrix} .4 \\ .1 \\ .5 \end{bmatrix} \begin{bmatrix} .4 \\ .2 \\ .4 \end{bmatrix}$$

are associated with

$$\begin{bmatrix} -4 \\ 8 \\ 1 \end{bmatrix} \begin{bmatrix} -4 \\ 8 \\ 1 \end{bmatrix} \begin{bmatrix} 6.85 \\ 5.00 \\ 9.65 \end{bmatrix}$$

Since the new vertex of this triangle has associated with it a column identical to one already in the basis, the pivot step will merely remove this pre-existing column. The vertex (.5, .1, .4)′ will therefore be eliminated from triangle 11, and we obtain triangle 12.

As figure 2 indicates, the three vertices of triangle 12, are given by

$$\begin{bmatrix} .3 \\ .2 \\ .5 \end{bmatrix} \begin{bmatrix} .4 \\ .1 \\ .5 \end{bmatrix} \begin{bmatrix} .4 \\ .2 \\ .4 \end{bmatrix}$$

which, according to our rules, are associated with the three columns

$$\begin{bmatrix} 0 \\ 2.4 \\ -1 \end{bmatrix} \begin{bmatrix} -4 \\ 8 \\ 1 \end{bmatrix} \begin{bmatrix} 6.85 \\ 5.00 \\ 9.65 \end{bmatrix}$$

If the new column is brought into the basis represented by the equations

$$3.37 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + 2.05 \begin{bmatrix} -4 \\ 8 \\ 1 \end{bmatrix} + .72 \begin{bmatrix} 6.85 \\ 5.00 \\ 9.65 \end{bmatrix} = \begin{bmatrix} 0 \\ 20 \\ 9 \end{bmatrix},$$

then the conventional pivot rules will eliminate the first slack column, and the basic equations become

$$1.24 \begin{bmatrix} 0 \\ 2.4 \\ -1 \end{bmatrix} + 1.54 \begin{bmatrix} -4 \\ 8 \\ 1 \end{bmatrix} + .9 \begin{bmatrix} 6.85 \\ 5.00 \\ 9.65 \end{bmatrix} = \begin{bmatrix} 0 \\ 20 \\ 9 \end{bmatrix}.$$

The algorithm now terminates, since the three columns associated with the vertices of triangle 12 form a feasible basis.

A formal demonstration that the algorithm will always terminate in a finite number of iterations with the desired solution may be found in [1], [2], and [5]. The fundamental ideas behind the algorithm are, however, quite simple. At each iteration a new vertex is determined and the column associated with that vertex is calculated. That column is brought into the basis consisting of the first slack vector and the two columns associated with the remaining two vertices of the triangle. The algorithm terminates if the first slack vector is eliminated by this pivot step. If not, some other column is removed and the vertex associated with that column is eliminated from the triangle, thereby producing a new triangle. The process is then repeated. When the algorithm eventually stops it provides us with a region on the simplex containing an approximate equilibrium price vector.

Some aspects of the algorithm are capable of a rudimentary economic interpretation. Each stage of the algorithm involves a production plan and a weighted sum of market demands evaluated at neighboring price vectors. A new price vector is proposed and the technology is examined to determine the activity which maximizes profit at this set of prices. If the maximum profit is nonnegative, the scale of this activity is increased from zero. On the other hand, if the maximum profit is negative, the market demands are calculated at these prices and introduced into the basis. A new price vector is then generated, depending upon the result of a pivot step. It is this last aspect of the algorithm which seems to be very difficult to interpret from an economic point of view.

REFERENCES

1. T. HANSEN, *On the Approximation of a Competitive Equilibrium*, Ph.D. Dissertation, Yale University 1968.
2. ———, and H. SCARF, "On the Applications of a Recent Combinatorial Algorithm," Cowles Foundation Discussion Paper No. 272, 1969.
3. H. W. KUHN, "Simplicial Approximation of Fixed Points," *Proceedings of the National Academy of Sciences*, Vol. 61, 1968.
4. H. SCARF, "The Approximation of Fixed Points of a Continuous Mapping," *SIAM Journal of Applied Mathematics*, Vol. 15 1967.
5. ———, "On the Computation of Equilibrium Prices," *Ten Economic Studies in the Tradition of Irving Fisher*, New York 1967.