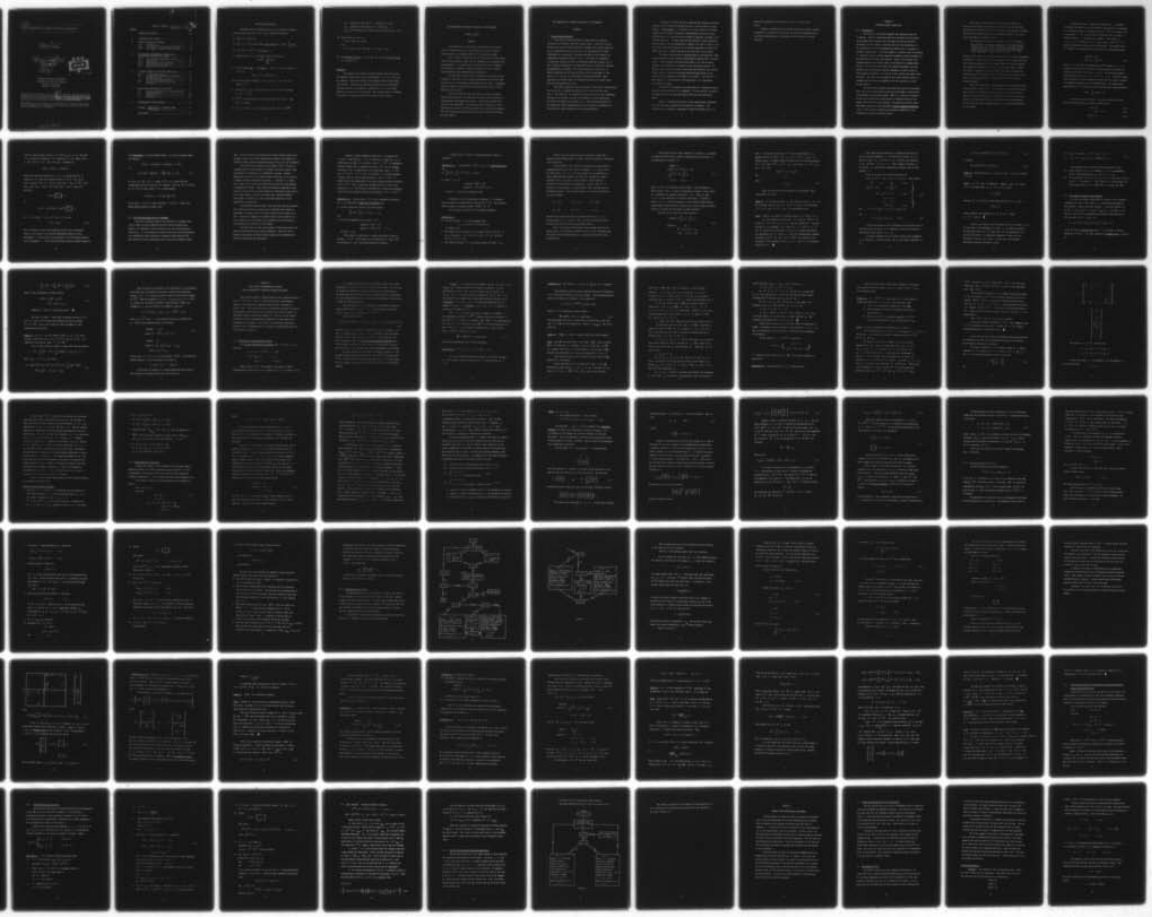
AD-A053 169 STANFORD UNIV CALIF SYSTEMS OPTIMIZATION LAB F/G 5/3
THE COMPUTATION OF ECONOMIC EQUILIBRIA BY PATH METHODS.(U)
OCT 77 T R ELKEN N00014-75-C-0267
UNCLASSIFIED SOL-77-26 NL

1 OF 2
AD
A053169



THE COMPUTATION OF ECONOMIC EQUILIBRIA BY PATH METHODS

BY

THOMAS R. ELKEN

TECHNICAL REPORT SOL 77-26

OCTOBER 1977

Systems Optimization Laboratory

Department of Operations Research

Stanford University

Stanford California 94305

(6) THE COMPUTATION OF ECONOMIC EQUILIBRIA BY PATH METHODS

by

(10) Thomas R. Elken

(9) TECHNICAL REPORT SOL-77-26 (14)

(11) October 1977

(12) 127p.

SYSTEMS OPTIMIZATION LABORATORY

DEPARTMENT OF OPERATIONS RESEARCH

Stanford University
Stanford, California

(15) Research and reproduction of this report were partially supported by the U.S. Energy Research and Development Administration Contract EY-76-03-0326 PA #18; the Office of Naval Research Contracts N00014-75-C-0267 and N00014-75-C-0865; the National Science Foundation Grants MCS76-81259 and MCS76-20019.

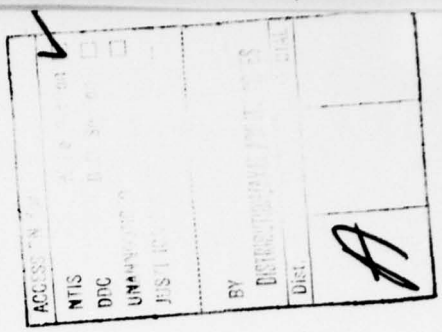


TABLE OF CONTENTS

NOTATION AND NUMBERING

We define some notation which may not be completely standard. All points and sets are in $\mathbb{R}^n$ unless indicated otherwise.

1) $\mathbb{R}^n_+ \equiv \{x \in \mathbb{R}^n | x \geq 0\}$

2) For $x, y \in \mathbb{R}^n$, denote their <u>inner product</u> by $\langle x,y \rangle = \sum_{i=1}^{n} x_i y_i$.

3) Let $\|x\| = \langle x,x \rangle^{1/2}$ be the <u>norm</u> of x.

4) Denote by $d(\cdot,\cdot)$ the distance function defined as

$$d(A,B) = \inf_{\substack{x \in A \\ y \in B}} \|x-y\|$$

5) Let the <u>ball about</u> A <u>of radius</u> ϵ, where A may be a point or a set, be

$$B(A,\epsilon) = \{x \in \mathbb{R}^n | d(x,A) \leq \epsilon\}$$

6) For any positive integer m, let $\underline{m} \equiv \{1,2,\ldots,m\}$. If $m = 0$, then $\underline{m} \equiv \emptyset$.

7) $\text{conv}[A_1,A_2,\ldots,A_n]$ is the convex hull of the sets (or points) $A_1, A_2, \ldots, A_n$.

8) $\mathbb{R}^{n \times m}$ is the space of all real-valued matrices with n rows and m columns.

9) If $\sigma \subset \underline{n}$ and $\mu \subset \underline{m}$ are index sets, then, for $A \in \mathbb{R}^{n \times m}$,

$A_{\sigma,\cdot}$ consists of the rows of A indexed by σ, and

$A_{\cdot,\mu}$ consists of the columns of A indexed by μ

$A_{\sigma,\mu}$ is the submatrix of elements whose indices are in $\sigma \times \mu$.

10) For two sets A and B,

$$A \setminus B = \{x | x \in A,\ x \notin B\}$$

and

$$A - B = \{z | z = x-y \text{ for some } x \in A, \text{ and } y \in B\}.$$

11) The boundary of a set C is called ∂C, and the interior of set C is called C^{o}.

Numbering

The chapters are numbered by Roman numerals and the sections are numbered consecutively within each chapter, i.e., II.1, II.2, etc. All theorems, lemmas, and examples are numbered consecutively within each section. Equations are numbered separately and are identified by being enclosed in parentheses. Equation (V.1.2) is the second equation in Section V.1, for instance. The chapter numeral is omitted for results or equations referred to within the same chapter.

THE COMPUTATION OF ECONOMIC EQUILIBRIA BY PATH METHODS

Thomas R. Elken

ABSTRACT

An introduction to the economic equilibrium model is given and it is demonstrated that a path method can be used to compute equilibria for pure exchange economies in a nonlinear setting.

Next, a model is described for an economy in which the utility functions are piecewise linear and the consumption and production sets are polyhedral. It is shown that an equilibrium for this economy is the solution to a system of bilinear equations subject to certain linear inequality and complementarity constraints.

Two approaches are discussed for computing equilibria for such economies. The first is the bilinear complementarity algorithm (BCA) and the second is the homotopy retraction algorithm (HRA). Convergence proofs are given for both methods using the general theory for path methods described above.

The BCA and HRA have been implemented as computer programs. Detailed descriptions of the algorithms are given, and the results of some numerical experiments are reported. Seven small problems were solved by both algorithms. No conclusion could be drawn as to which algorithm was superior, but both performed well enough that it appears that much larger equilibrium problems also can be solved efficiently by these methods.

THE COMPUTATION OF ECONOMIC EQUILIBRIA BY PATH METHODS

CHAPTER I

I. Introduction and Summary

This report is concerned with two algorithms for computing equilibria for a general piecewise linear economy. Proofs will be presented which show that these algorithms converge under certain conditions and the computational results from the first implementation of these algorithms will be reported. The first algorithm is the bilinear complementarity algorithm of Wilson [1976]. A new proof of convergence is given here using the results on subdivided complexes in Elken [1977], and a detailed description of an algorithm which is suitable for numerical implementation is presented. The second algorithm is new but it is heavily influenced by the ideas of Wilson [1976] and Kellogg, Li, and Yorke [1976]. We call it the homotopy retraction algorithm. Both algorithms show great promise in computing equilibria for larger models than has been possible up to now.

This report constitutes the second half of the author's dissertation. The first half is contained in Elken [1977]; we will call this work Part 1. The reader who is interested in the proofs for the two convergence theorems in this report will have to be familiar with the definitions and results in Chapter II of Elken [1977]. Those readers interested in the algorithms, their implementation, and the numerical experiments will find that this report contains all the relevant material.

In Chapter II of this report we introduce the economic equilibrium problem in its traditional form and show how it can be solved by a path method. (A path method is a procedure for solving a system of nonlinear equations by following a (piecewise) differentiable path from a known starting point to a solution for the system of equations; see Elken [1977] for more details.) We also introduce the piecewise linear formulation of an economy as it was developed by R. Mantel [1967] and R. Wilson [1976]. It is this formulation which we use to develop algorithms to exploit the linear structure to compute equilibria for this model of an economy.

In Chapter III we present a generalization of the piecewise linear equilibrium problem, the bilinear complementarity problem (BCP) of Wilson [1976]. Wilson defined a path and proved that it led from an easily obtained starting point to a solution of the equilibrium problem. We present a new proof of this and an algorithm which exploits the linearity of this model to reduce the nonlinear path-following problem to the lowest possible dimension ($\leq$ the number of consumers). The last part of this chapter is a description of the algorithm as it is implemented in a computer program.

In Chapter IV we present a new path method for computing economic equilibria and prove that it is convergent. We also describe, in detail, an algorithm based upon this path method which has been numerically tested.

Chapter V contains the results of some computational experiments with the algorithms presented in the previous two chapters. The conclusions are primarily a comparison of these two methods, but it is

hoped that comparisons with existing codes will be made in the future.

Chapter VI suggests some promising directions for future research. An Appendix is included which describes one method for generating piecewise linear approximations to nonlinear utility functions of several variables.

CHAPTER II

COMPUTING ECONOMIC EQUILIBRIA

II.1. Introduction

The concept "equilibrium" suggests that opposing forces are in balance. Economic equilibrium theory is concerned with the balance of the demand from consumers with the supply provided by producers. Each agent in the economy is concerned only with the maximization of his satisfaction or profit depending upon whether he is a consumer or producer, respectively. To be more explicit, a special class of economies is considered, namely, private ownership economies in which consumers own the resources and control the producers. Given a price system, each producer maximizes his profit, which is distributed to consumer-shareholders. The wealths of the latter are thereby determined, and they maximize their satisfaction from the consumption of goods subject to their wealth constraints. As a result of this process, each agent chooses an action. The state of an economy in which these optimal actions of consumers and producers are compatible with the resources is called an equilibrium.

The traditional questions which have been studied by economists are: Does a price system exist which puts the economy in equilibrium? Is an economy at equilibrium stable? That is, if there is a violation of one of the conditions of equilibrium, do rational actions of the agents tend to restore an equilibrium. An excellent work which deals with these questions is Arrow and Hahn's General Competitive Analysis [1971]. This work also provides an excellent introduction to the literature of economic equilibria theory.

This work is concerned with the existence question as it arises in the problem of computing equilibrium prices and activities. The question of stability will not be discussed. The ability to compute equilibria for large-scale models is important because it may allow a great improvement in economic modeling.

The difficulties inherent in economic model are well known.

> "The problem of collecting reliable data on the technological processes that are currently available is enormous, to say nothing of the difficulty of inventing appropriate input-output coefficients for productive techniques that remain to be discovered." (Scarf [1973], pp. 7-8).

Because of the large size of most economic models, the linear programming formulation is invariably used. The advantage of the equilibrium model over linear programming is that consumer demands and their dependence upon price are recognized and modeled with some rationality. Unless the system is prepared to tolerate strict rationing, consumers will respond not only to the availability of items but also to their price. Thus, we see the importance of computing equilibria as an aid in economic planning.

The first algorithms for the computation of economic equilibria were developed by Scarf [1967]. He developed an algorithm for solving fixed points of a function which maps the unit simplex into itself. Then the equilibrium problem was transformed into such a fixed point problem. To illustrate how this transformation can be accomplished we will deal with a pure exchange economy, i.e., we ignore production for the moment.

Suppose there are n goods in the economy of m consumers or traders, and that each of the consumer's preferences are represented by a utility function. To be precise, let $y \in \mathbb{R}^n$ be a vector of the n goods and $\pi \in \mathbb{R}^n$ be the vector of prices for the n goods. A bundle of goods x is preferred to a bundle y by consumer i, $i = 1,\ldots,m$ if and only if $u_i(x) > u_i(y)$ where $u_i: \mathbb{R}^n \to \mathbb{R}$ is a strictly concave and continuous utility function. The demands of the i^{th} consumer are determined by the solution to the following problem:

$$\begin{aligned} &\text{maximize} \quad u_i(y) \\ &\text{subject to} \quad \pi y \leq \pi w^i , \end{aligned} \tag{1.0}$$

where w^i is the initial endowment of the i^{th} consumer, $i = 1,\ldots,m$. We shall assume that the solution to this problem can be written as a continuous function of the prices π, $d_i(\pi)$. The individual trader's excess demand function is $d_i(\pi) - w^i$, $i = 1, \ldots, m$. The excess demand will be positive for those commodities whose stock he wishes to increase by exchange and negative for the remaining items. The market excess demand function g is the sum of the individual excess demand functions

$$g(\pi) = \sum_{i=1}^{m} (d_i(\pi) - w^i)$$

An equilibrium price vector $\bar{\pi}$ is one for which all of the market excess demands are less than or equal to zero

$$g(\pi) \leq 0, \tag{1.1}$$

$$\pi \geq 0, \quad \pi \neq 0 , \tag{1.2}$$

$$\pi^T g(\pi) = 0 \quad . \tag{1.3}$$

Equation (1.3) implies that there is a zero price for any commodity whose demand is strictly less than zero.

An important property of the excess demand function is that (1.3) will be satisfied for any price system π because the value of each consumer's excess demand is identically zero. This property is known as Walras' law. Another property which is easy to derive is that the excess demand function is homogeneous of degree zero, that is, if the prices are scaled up by a constant factor, the demands remain the same. This implies that it is sufficient to search for an equilibrium price vector on the unit simplex, S^n. The following function from S^n into S^n has the property that a fixed point is an equilibrium:

$$f_i(\pi) = \frac{x_i + \max(0, g_i(\pi))}{1 + \sum_{i=1}^{n} \max(0, g_i(\pi))} , \qquad i \in \underline{n} .$$

To see that a fixed point of f is an equilibrium point, the equation $\bar{\pi} = f(\bar{\pi})$ can be written

$$\alpha\bar{\pi}_i = \bar{\pi}_i + \max[0, g_i(\bar{\pi})]$$

with $\alpha = 1 + \sum_i \max[0, g_i(\pi)]$. If α is in fact greater than one, then the condition $\bar{\pi}_i(\alpha-1) = \max[0, g_i(\bar{\pi})]$ implies that $g_i(\bar{\pi}) > 0$ whenever $\bar{\pi}_i > 0$. Since some $\bar{\pi}_i$ are strictly positive this violates $\bar{\pi}^T g(\bar{\pi}) = 0$. Therefore $\alpha = 1$ and, hence, $g_i(\bar{\pi}) \leq 0$ for all i, and $\bar{\pi}$ is an equilibrium vector.

One procedure for moving towards equilibrium is a price adjustment in which the price of a good is increased if the excess demand for that good is positive, decreased if the excess demand is negative. This is the classical tâtonnement or "groping" for an equilibrium (Walras [1874]). Scarf [1960] has shown that for virtually arbitrary excess demand functions, this process can be unstable. This price adjustment is globally stable, however, if a certain gross substitutability between all commodities is satisfied (Arrow and Hurwicz [1958]). The differential equation which expresses this process is

$$\frac{d\pi}{dt} = g(\pi) \tag{1.4}$$

We will now show that a convergent price adjustment process can be defined which behaves like (1.4) initially. Suppose that the excess demand function g is differentiable. Since the excess demand function is homogeneous of degree zero, $g'(\pi)$ has rank n-1, in general; hence, we must reduce the dimension of the problem by one. This can be done by solving the problem on the unit simplex or the nonnegative portion of the unit sphere, $\{x \in \mathbb{R}^n | x \geq 0, \|x\| = 1\}$. We shall follow a certain amount of tradition in mathematical economics by considering a distinguished commodity or 'numeraire' (Quirk and Saposnik [1968]) so that the price of other goods are measured in terms of this numeraire good.

Suppose there are $n+1$ commodities with prices $\pi_0, \pi_1, \ldots, \pi_n$ where π_0 is the unit price of the numeraire good. We make the following assumption on the excess demand function $g_i(\pi)$, $i = 0, \ldots, n$.

Assumption 1.1. If $\pi_i = 0$, then $g_i(\pi) > 0$, $i = 0,1,\ldots,n$.

Thus, no equilibrium price vector will be on the boundary of $\mathbb{R}^{n+1}_+$. So we will work in the domain of price systems $(\pi_0,\ldots,\pi_n)$ which satisfy $\pi_0 > 0$. In this domain, by the homogeneity property of price systems, one can normalize a price vector by dividing by π_0. Thus a price system $(\pi_0,\ldots,\pi_\ell)$ can be represented by a unique point

$$(p_1,\ldots,p_n) \equiv (\pi_1/\pi_0, \ \ldots \ , \ \pi_n/\pi_0) \quad \text{in} \quad \mathbb{R}^n_+$$

With this interpretation, the space of price systems is $\mathbb{R}^n_+$. Suppose that the excess demand functions $g_0(\pi), \ \ldots \ , \ g_n(\pi)$ are C^2 and satisfy Walras' law. Then

$$g_0(\pi) = - \sum_{i=1}^{n} \frac{\pi_i}{\pi_0} g_i(\pi) \ . \tag{1.5}$$

This along with the boundary condition 1.1 implies that the equations

$$g_i(\pi) = 0 \ , \qquad i = 1,\ldots, \ n$$

are necessary and sufficient conditions for economic equilibrium.

Let $f_i(p) = g_i(1, \ \pi_1/\pi_0, \ \ldots \ , \ \pi_n/\pi_0) \ , \quad i = 1,\ldots,n \ .$

Then consider the application of the following version of Kellogg, Li, and Yorke's deformation. Pick some p^0 such that $p^0 \in \mathbb{R}^n_+$ and for exactly one $i \in n$, $p^0_i = 0$, say $i = 1$. Then the boundary condition

(III.1.1) can be shown to hold on $D = \{p \mid 0 \leq p_i \leq P,\ i \in n\}$ for some $P > 0$ because of Assumption 1.1 and equation (1.5) (cf. Smale [1976], p. 116). $M \equiv D \times [-1,1]$ and $F:M \to \mathbb{R}^n$ is denoted by

$$F(p,\theta) = \theta f(p) - (1-\theta)(p-p^0) .$$

Then by the theory of Section II.1, if 0 is a good value for F, there is a C^1 curve with boundary points at $(p^0,0)$ and $(p^*,1)$ and, of course, $f(p^*) = 0$. But if $(p(t),\theta(t)) = [0,T] \to F^{-1}(0)$ such that $(p(0),\theta(0)) = (p^0,0)$ and $(p(T),\theta(T)) = (p^*,1)$, then $\dot{p}(t)$ satisfies

$$F'(p,\theta) \begin{pmatrix} \dot{p}(t) \\ \dot{\theta}(t) \end{pmatrix} = 0$$

or

$$(\theta f'(p) - (1-\theta)I \mid f(p) - (p-p^0)) \begin{bmatrix} \dot{p}(t) \\ \dot{\theta}(t) \end{bmatrix} = 0 .$$

At $t = 0$ we have $\theta = 0$, $p = p^0$, $\dot{\theta}(t) = \lambda > 0$, and

$$\dot{p}(t) = \lambda f(p) \tag{1.6}$$

Thus, initially at least, the adjustment of prices by a retraction method, is the same as the classical tâtonnement method of price adjustments. It can be said, with reference to the equilibrium problem, that the parameter θ of the retraction method allows a smooth transition

from tâtonnement to a global Newton method. To see this consider again the relation

$$[\theta f'(p) - (1-\theta)I]p(t) = -\theta(t)(f(p) - p + p^0) ,$$

or

$$f'(p)\,\dot{p}(t) - \frac{1-\theta}{\theta}\,\dot{p}(t) = -\frac{\dot{\theta}(t)}{\theta}\,(f(p) - p + p^0) \qquad (1.7)$$

As $p(t) \to p^*$ and $\theta(t) \to 1$ along $F^{-1}(0)$ it is clear that the second term on the left side of (5.7) vanishes. Also for $(p,\theta) \in F^{-1}(0)$, $p - p^0 = \theta/(1-\theta)\, f(p)$, hence, (5.7) is approximately

$$f'(p)\,\dot{p}(t) = -\dot{\theta}(t)\,(\frac{1}{\theta} + \frac{1}{1-\theta})\, f(p) .$$

So for some $\lambda > 0$, $\dot{p}(t)$ almost satisfies $f'(p)\,\dot{p}(t) = -\lambda f(p)$, the global Newton equation of Smale [1976].

II.2. The Piecewise-Linear Model of Exchange

The usual assumptions made in the literature of economic equilibrium theory include assumptions that the utility functions are concave, the consumption sets are convex as are the production possibility sets. It is well-known that concave functions can be approximated to an arbitrarily small tolerance by piecewise linear functions and that convex sets can be approximated very closely by polyhedral convex

sets. In this section the discussion will center around a model of an economy in which the utility functions are assumed to be concave and piecewise linear, and the consumption and production sets are polyhedral.

This model was introduced by Rolf Mantel [1968]. He gave an ingenious proof of the existence of economic equilibria without using the methods of combinatorial or differential topology. Mantel's approach did not seem to be computationally efficient, for it involved a complex limiting operation at one point. G. Dantzig, B. C. Eaves, and D. Gale [1976] did use this model as the basis for a new approach to computing equilibria. They solve the problem by computing a fixed point of a point-to-set map whose values are determined by the solution to a linear program. We will not discuss this algorithm further, even though it promises to be one of the main competitors to the algorithms we present here.

Both Dantzig, Eaves and Gale [1976] and this work are concerned with solving equilibrium problems with a relatively small number of households or traders (about 3-10) and a large number of goods (up to 300). One obvious example of a problem of such a scale would be if 3 to 10 countries were considered to be consumers who were involved in the production and trade of up to 300 goods.

For simplicity, we will first present a model which allows only linear utility functions and ignores production. Later it will be shown how the general piecewise-linear economy can be formulated and solved in essentially the same manner.

Consider a simple exchange economy with m households and ℓ goods. Each household $i \in \underline{m}$ has available a finite set S_i of activities such that, if it chooses a vector $z^i \in \mathbb{R}^{S_i}$ of nonnegative activity levels, then it obtains the utility $\gamma^i z^i$ and it consumes the vector $B^i z^i$ of quantities of the commodities, where $\gamma^i \in \mathbb{R}^{S_i}$ and $B^i \in \mathbb{R}^{\ell \times S_i}$. If $w^i \in \mathbb{R}^{\ell}$ is household, i's initial endowment of goods, then its excess demand function is $g_i(z^i) = B^i z^i - w^i$. Notice that in the terminology of the pure exchange problem in Section 1, $U_i(z^i) = \gamma^i z^i$. Also, in that pure exchange problem, if all of the activities in S_i are merely the consumption of a particular commodity then all of the technology matrices are identities ($B^i = I \in \mathbb{R}^{\ell \times \ell}$).

Definition 2.1. A price vector $\bar{\pi} \neq 0$, and a consumption allocation, $\bar{z}_i$, $i \in \underline{m}$ constitute a competitive equilibrium iff

a) the net trades are feasible with free disposal:

$$\sum_{i=1}^{m} B^i \bar{z}^i \leq \sum_{i=1}^{m} w_i, \quad \bar{z}^i \geq 0, \quad i \in \underline{m},$$

and

b) Given the nonnegative price vector $\bar{\pi}$, $\bar{z}^i$

$$\text{maximizes } U(z^i) = \gamma^i z^i$$
$$\text{subject to } \bar{\pi} B^i z^i \leq \bar{\pi} w^i, \quad z^i > 0,$$

for each $i \in \underline{m}$.

This notion of equilibrium is clearly equivalent to that in Section 1. If $\bar{z}^i$ is considered to be a function of $\bar{\pi}$, then $B^i \bar{z}^i$ is equivalent to $d_i(\bar{\pi})$, the demand function of Section 1.

A similar notion is that of a quasi-equilibrium, which is defined as

Definition 2.2. A specification $(\bar{\pi}, \bar{z}^i, i \in \underline{m})$ is a quasi-equilibrium iff

a) $\sum_{i=1}^{m} B^i z^i \leq \sum_{i=1}^{m} w^i$, $\bar{z}^i \geq 0$, $i \in \underline{m}$ and

b) Given $\bar{\pi} \geq 0$, $\bar{z}^i$

$$\text{minimizes } \bar{\pi}(B^i z^i - w^i)$$

$$\text{subject to } \gamma^i z^i \geq \gamma^i \bar{z}^i,\ z^i \geq 0$$

for each $i \in \underline{m}$, and this minimum is zero.

Condition b) can be interpreted as choosing $\bar{z}^i$ to minimize the net expense of maintaining the utility level $U_i(\bar{z}^i)$ and requiring that the budget be balanced exactly $\bar{\pi}(B^i \bar{z}^i - w^i) = 0$.

The following conditions will be assumed throughout:

Assumption 2.3.

a) For each consumer $i \in \underline{m}$, the consumption set

$$X_i = \{x \in \mathbb{R}^\ell \mid (\exists\ z^i \geq 0)\ B^i z^i \leq x\}$$

is bounded below.

b) The induced utility function $U^i(x) = \max\{\gamma^i z^i \mid z^i \geq 0,\ B^i z^i \leq x\}$ on X_i is insatiable, i.e., for all $x^1 \in X_i$ $\exists$ $x^2 \in X_i$ such that $U^i(x^1) < U^i(x^2)$, $i \in \underline{m}$.

c) The initial endowment w^i is strictly positive for each $i \in \underline{m}$.

Debreu [1959] has shown that these assumptions imply that a competitive equilibrium exists, and that a quasi-equilibrium is equivalent to an equilibrium.

Let us consider, for a moment the complexity of the problem of computing equilibria for this class of economies. B. C. Eaves [1975] has shown that if $B^i = I$, $i \in \underline{m}$, this problem can be formulated as a linear complementarity problem, and with these assumptions. Lemke's algorithm [1965] will yield a solution after a finite number of additions, multiplications and comparisons. However, when B^i is allowed to be more general, a problem with rational data may have an irrational solution. The following example, due to Andreu Mas-Colell, is such a problem:

$$B^1 = \begin{pmatrix} .5 \\ 1 \end{pmatrix}, \qquad B^2 = \begin{pmatrix} 1 \\ .5 \end{pmatrix}, \qquad B^3 = \begin{pmatrix} .25 \\ .20 \end{pmatrix},$$

$$w^i = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad \gamma^i = (1), \qquad i = 1,2,3.$$

Mas-Colell has shown that every equilibrium price vector $\bar{\pi}$ is a positive scale of $(1 + \sqrt{3}, 1)$. Thus, no finite algorithm can hope to solve this class of equilibrium problems exactly.

Next, we introduce the auxiliary linear program which has the property that, if the correct constants are used in the right-hand side, the solution to the linear program is a quasi-equilibrium and, hence, an equilibrium.

The objective of the linear program is to maximize p, a measure of exports from the system, subject to maintaining utility levels v_i and feasibility of trades, i.e.,

$$\begin{aligned} &\text{maximize} \quad p \\ &\text{subject to} \quad \gamma^i z^i \geq v_i, \ i \in \underline{m} \\ &\sum_{i \in \underline{m}} B^i z^i + pe \leq \sum_{i=1}^{m} w_i \\ &z^i \geq 0, \quad i \in \underline{m}, \end{aligned} \tag{2.1}$$

where $e \in \mathbb{R}^{\ell}$ is a strictly positive vector. This formulation is due to Wilson [1976]. It also was influenced by Debreu's [1951] notion of a coefficient of resource utilization. If $e = \sum_{i=1}^{m} w^i$, then $\varphi = 1-p$ is the coefficient of resource utilization. In Debreu's theory, if $\varphi = 1$, then the economy is on the Pareto optimal frontier (any increase in one consumer's utility would require another consumer's utility to decrease for the trades to remain feasible).

The dual for the auxiliary linear program is

$$\text{minimize} \quad \pi\left(\sum_{i=1}^{m} w_i\right) - \sum_{i=1}^{m} \lambda_i v_i \tag{2.2}$$

$$\begin{aligned} \text{subject to} \quad & \pi e = 1 \\ & \pi B^i - \lambda_i \gamma^i \geq 0, \quad i \in \underline{m} \\ & \pi \geq 0, \quad \lambda_i \geq 0, \ i \in \underline{m}. \end{aligned}$$

Here, π is the price vector and λ_i^{-1} can be interpreted as i's marginal utility of income. Let $u_i \equiv \pi(w^i - B^i z^i)$, $i \in \underline{m}$, be i's budget surplus. If complementary slackness holds for the dual problems (2.1), (2.2) then $\pi B^i z^i = \lambda_i \gamma^i z^i = \lambda_i v_i$ and we have $u_i = \pi w^i - \lambda_i v_i$ for $i \in \underline{m}$. Also,

$$\pi(\sum_i B^i z^i + pe) = \pi(\sum_i w^i)$$

and

$$\pi e = 1$$

imply

$$p = \sum_i u_i \ . \tag{2.3}$$

Thus, at a solution to the primal and dual programs, these relationships will hold.

Lemma 2.4. If the utility levels v_i, are chosen so that $u_i = 0$, $i \in \underline{m}$, at an optimal solution $(\bar{\pi};\ \bar{\lambda}_i,\ i \in \underline{m};\ \bar{z}^i,\ i \in \underline{m},\ \bar{p})$ to the problems (2.1), (2.2), then $(\bar{\pi};\ \bar{z}^i,\ i \in \underline{m})$ is a quasi-equilibrium.

Proof. Clearly, the trades are feasible because (2.3) implies $p = 0$.

All that remains to show is that $\bar{z}^i$ minimizes $\bar{\pi}(B^i z^i - w^i)$ subject to $\gamma^i z^i \geq \gamma^i \bar{z}^i \equiv v_i$, $z^i \geq 0$. Call this problem (P). The dual problem (D) is to maximize $\lambda_i v_i$ subject to $\lambda_i \gamma^i \leq \bar{\pi} B^i$, $\lambda_i \geq 0$. $\bar{z}^i$ and $\bar{\lambda}_i$ satisfy the constraints of (P) and (D) because they satisfy the constraints for (2.1) and (2.2). Also, the complementary slackness conditions, $\bar{\lambda}_i \gamma^i \bar{z}^i = \bar{\lambda} v_i$ and $\lambda_i \gamma_i z^i = \bar{\pi} B^i \bar{z}^i$ are satisfied by the optimality of the variables for (2.1), (2.2). Thus, $\bar{z}^i$ is an optimal solution of (P). ∎

This lemma provides motivation to parametrically vary the initial utility assignments v_i and adjust the solutions of (2.1) and (2.2) until $u_i = 0$ for all $i \in \underline{m}$. Consider slack variables $t_i \geq 0$ such that $\gamma^i z^i = v_i + t_i$. Then, instead of varying v_i, we can vary t_i, ignoring the complementary slackness with the dual variable λ_i.

Thus, we can state the equilibrium problem as:

Find $(\pi, \lambda, \zeta^i \ (i \in \underline{m}), \rho;\ s,\ t,\ z^i \ (i \in \underline{m}), p)$ which satisfy

$$\left.\begin{aligned}
\gamma^i z^i \qquad - t_i \qquad &= v_i \qquad && i \in \underline{m}, \\
\sum_{i=1}^{m} B^i z^i + pe \qquad + s \qquad &= \sum_{i=1}^{m} w^i , \\
\pi B^i - \lambda_i \gamma^i - \zeta^i &= 0 , && i \in \underline{m}, \\
\pi e - \rho &= 1 , \\
\pi s \qquad &= 0 , \\
\zeta^i z^i &= 0 , && i \in \underline{m} ,
\end{aligned}\right\} \tag{2.3}$$

and $\quad (\pi, \ , \zeta^i \ (i \in \underline{m}), \rho;\ s,\ t,\ z^i \ (i \in \underline{m},\ p) \geq 0$

and

$$\pi w^i - \lambda_i (v_i + t_i) = 0 , \qquad i \in \underline{m} , \tag{2.4}$$

Solution procedures for this formulation of the problem will be discussed below, but first, it is important to discuss the choice of the initial utility levels v_i, $i \in \underline{m}$.

An obvious candidate for the initial utility level assignment v_i is consumer i's induced utility v_i^* for the initial endowment w^i, i.e.,

$$v_i^* = U_i(w^i) \equiv \max\{\gamma^i z^i \mid z^i \geq 0,\ B^i z^i \leq w^i\}\ , \tag{2.5}$$

or dually

$$v_i^* = \min\{\pi^i w^i \mid \pi^i \geq 0,\ \pi^i B^i \geq \gamma^i\} \tag{2.6}$$

Lemma 2.5. With the choice $v_i = v_i^*$, $u_i \geq 0$, for $i \in \underline{m}$ at a solution to (2.3).

Proof. $u_i = \pi w^i - \lambda_i v_i^*$ by definition. Suppose $\lambda_i v_i^* > \pi w^i$. Since $\pi w^i > 0$, we cannot have $\lambda_i = 0$. If $\lambda_i > 0$, we have

$$v_i^* > \lambda_i^{-1} \pi w^i\ .$$

However, $\lambda_i^{-1}\pi \geq 0$, and by the dual feasibility (2.2) of λ_i and π,

$$\lambda_i^{-1} \pi B^i \geq \gamma^i$$

which contradicts the optimality of v_i^* in (2.6). Hence, $u_i = \pi w^i - \lambda_i v_i^* \geq 0$. ∎

Since we are trying to find a solution of (2.5) for which $u_i = 0$, it would seem to be advantageous to choose v_i as large as possible. For arbitrary choices of v_i larger than v_i^* the constraints (2.5) may not be feasible. For theoretical purposes it turns out to be preferable to choose $v_i \in (0, v_i^*)$. In this case, the following equivalent properties are easy to verify

$$\text{(1)}\quad \lambda_i t_i = 0 \ \text{ implies } \ u_i = \pi w^i - \lambda_i(v_i + t_i) > 0$$
$$\text{(2)}\quad u_i = \pi w^i - \lambda_i(v_i + t_i) \leq 0 \ \text{ implies } \ \lambda_i t_i > 0 \ . \qquad (2.7)$$

This property will be exploited from two points of view:

(a) If the condition $u_i \geq 0$ is enforced, (2.7) is a complementary type of relationship between u_i and $\lambda_i \cdot t_i$, $i \in \underline{m}$,

(2) If one considers $f(\lambda;\pi,t) \equiv \pi w^i - \lambda_i \cdot (v_i + t_i)$ and one is solving $f(\lambda;\pi,t) = 0$ subject to $\lambda \in \mathbb{R}^m_+$, then (2.7) is the boundary condition that states, for $\bar{\lambda} \in \partial\mathbb{R}^m_+$, $f(\bar{\lambda};\pi,t)$ points into $\mathbb{R}^m_+$.

II.3. The General Piecewise Linear Economy

Now we allow further constraints to be added to each consumer's consumption set and a finite number q of firms which are owned by the consumers. Suppose that each consumer i owns the fraction σ^i_j of firm j, where $\sigma^i_j \geq 0$, $j \in \underline{q}$, and $\sum_{i\in\underline{m}} \sigma^i_j = 1$, for each $j \in \underline{q}$. Before we can define the consumption sets we must define production sets for each of the q firms. Let

$$Y_j = \{y \in \mathbb{R}^\ell \mid (\exists\, u^j \geq 0)\ D^j u^j \leq d^j,\ y \leq E^j u^j + \omega^j\}$$

be the j^{th} firms' production set where ω^j is a vector of initial endowments for each $j \in \underline{q}$. Each household's consumption set is defined as

$$X_i = \{x \in \mathbb{R}^{\ell} \mid (\exists z^i \geq 0)\ A^i z^i \leq a^i,\ x \geq B^i z^i - \sum_{j \in \underline{q}} \sigma_j^i y^j,\ y^j \in Y_j\}\ .$$

With these definitions, we can define an equilibrium for economies with production

Definition 3.1. A price vector $\bar{\pi}$, a consumption allocation $\bar{z}^i$, $i \in \underline{m}$, and a production allocation $\bar{y}^j$, $j \in \underline{q}$, constitute a competitive equilibrium if and only if

a) $\bar{\pi} \geq 0$ and $\bar{\pi} \neq 0$,

b) $\sum_{i=1}^{m} B^i \bar{z}^i \leq \sum_{i=1}^{q} \bar{y}^i + \sum_{i=1}^{m} w^i$

c) $\bar{y}^j$ maximizes $\bar{\pi} y$

subject to $y \in Y_j$, $j \in \underline{q}$, and

d) $\bar{z}^i$ maximizes $\succ^i z^i$

subject to $\bar{\pi} B^i \bar{z}^i \leq \bar{\pi}(w^i + \sum_{j \in \underline{q}} \sigma_j^i \bar{y}^j)$, $A^i z^i \leq a^i$, for every $i \in \underline{m}$.

A quasi-equilibrium (or the compensated equilibrium of Arrow and Hahn [1971]) replaces condition (d) with

d') $\bar{z}^i$ minimizes $\bar{\pi} B^i z^i$

subject to $\succ^i z^i \geq \succ^i \bar{z}^i$, and $\bar{\pi}(B^i \bar{z}^i - w^i - \sum_{j \in \underline{q}} \sigma_j^i \bar{y}^j) = C$, $i \in \underline{m}$.

If we combine Assumption 2.3 (using the new definition of consumption set) with the assumption below, then equilibria exist and coincide with quasi-equilibria (Debreu [1959]).

Assumption 3.2.

a) free disposal

b) production is irreversible

c) $(Y_j - \{\omega^j\}) \cap \mathbb{R}_+ = \{0\}$, i.e., there is no free production and the firm may produce nothing at all.

The auxiliary linear program is now defined as

$$\begin{aligned} &\text{maximize} \quad p \\ &\text{subject to} \quad \gamma^i z^i \geq \mathbf{v}_i \,, \quad i \in \underline{m} \\ &\qquad\qquad A^i z^i \leq a^i, \quad i \in \underline{m} \\ &\qquad\qquad D^j u^j \leq d^j, \quad j \in \underline{q} \\ &\sum_{i=1}^{m} B^i z^i - \sum_{j=1}^{q} E^j u^j + pe \leq \sum_{i=1}^{m} w^i + \sum_{j=1}^{q} \omega^j \\ &z^i \geq 0, \quad i \in \underline{m}, \quad u^j \geq 0, \quad j \in \underline{q} \end{aligned} \tag{3.1}$$

whose dual is

$$\begin{aligned} &\text{minimize} \quad \sum_{i=1}^{m} [\pi w^i + \nu^i a^i - \delta_h \mathbf{v}_h] + \sum_{j=1}^{q} \delta^j d^j \\ &\text{subject to} \quad \pi e = 1 \\ &\qquad\qquad \delta^j D^j - \pi E^j \geq 0 \\ &\qquad\qquad \nu^i A^i + \pi B^i - \lambda_i \gamma^i \geq 0 \\ &\qquad\qquad \pi, \ \delta^j \ (j \in \underline{q}), \ \nu^i, \ \lambda_i \ (i \in \underline{m}) \ \text{ all} \geq 0 \,. \end{aligned} \tag{3.2}$$

Next we prove the

Lemma 3.3. At a solution to (3.1) and (3.2) and for y^j a solution to $\max\{\pi y \mid y \in Y^j\}$, the following equations hold:

$$u_i \equiv \bar{\pi}[w^i + \sum_{j=1}^{q} \sigma_j^i y^j - B^i \bar{z}^i]$$

$$= \bar{\pi}(w^i + \sum_{j=1}^{q} \sigma_j^i \omega^j) + \bar{\nu}^i a^i + \sum_{j=1}^{q} \bar{\delta}^j(\sigma_j^i d^j) - \bar{\lambda}_i(v_i + \bar{t}_i), \text{ for all } i \in \underline{m}.$$

Proof. y^j satisfies $\bar{\pi} y^j = \max \bar{\pi} y$

$$\text{subject to} \qquad D^j u^j \leq d^j \qquad (P)$$

$$y - E^j u^j \leq \omega^j, \; u_j \geq 0 \; .$$

Hence, $\bar{\pi} y^j = \min \delta^j d^j + \pi \omega^j$

$$\text{s.t.} \; \delta^j D^j - \pi E^j \geq 0 \qquad (D)$$

$$\pi = \bar{\pi}, \quad \delta^j \geq 0 \; .$$

But, using the complementary slackness properties of the optimal solution $(\bar{z}^i, \bar{u}^j, \bar{p}; \bar{\pi}, \delta^j, \bar{\nu}^j)$ to (3.1) and (3.2), it is true that

$$\delta^j D^j \bar{u}^j - \bar{\pi} E^j \bar{u}^j = 0$$

and

$$\bar{\delta}^j D^j \bar{u}^j = \bar{\delta}^j d^j \; .$$

If y^j is chosen so that $y^j = \omega^j + E^j \bar{u}^j$, then it is clear that $(y^j, u^j; \bar{\pi}, \delta^j)$ is a solution for (P), (D). Hence,

$$\bar{\pi} y^j = \bar{\delta}^j d^j + \bar{\pi} \omega^j$$

Repeating this argument for each $j \in \underline{q}$, we get

$$\bar{\pi}\Big(\sum_{j=1}^{q} \sigma_j^i y^j\Big) = \bar{\pi}\Big(\sum_{j=1}^{q} \sigma_j^i d^j\Big) + \bar{\pi}\Big(\sum_{j=1}^{q} \sigma_j^i \omega^j\Big), \tag{3.3}$$

Again, using complementary slackness yields

$$\begin{aligned} -\bar{\pi}B^i\bar{z}^i &= \bar{\nu}^i A^i \bar{z}^i - \lambda_i r^i \bar{z}^i \\ &= \bar{\nu}^i a^i - \bar{\lambda}_i(v_i + t_i) \,. \end{aligned} \tag{3.4}$$

Summing (3.3) and (3.4) yields the result. ∎

The proof to Lemma 3.3 shows that an optimal solution to (3.1) and (3.2) yields profit maximizing production vectors by letting $y^j \equiv \omega^j + E\bar{u}^j$. Thus a proof similar to that for Lemma 2.4 would demonstrate the following

Lemma 3.4. If v_i, $i \in \underline{m}$ are chosen so that $\mu_i = 0$, $i \in \underline{m}$ at an optimal solution for (3.1), (3.2), $(\bar{\pi}, \bar{z}^i \ (i \in \underline{m}), y^j \ (j \in \underline{q}))$ is a competitive equilibrium, where $y^j = \omega^j + E\bar{u}^j$.

Also, a proof similar to Lemma 2.5 would prove the key property

$$\mu_i = \pi\Big(w^i + \sum_{j=1}^{q} \sigma_j^i \omega^j\Big) + \nu^i a^i + \sum_{j=1}^{q} \delta^j(\sigma_j^i d^j) - \lambda_i(v_i + t_i) > 0$$

when $\lambda_i \cdot t_i = 0$ if $v_i < v_i^*$ where

$$\begin{aligned} v_i^* = \max\{ & r^i z^i \,|\, z^i \geq 0,\ A^i z^i \leq a^i,\ B^i z^i \leq w^i + \sum_{j=1}^{q} (\sigma_j^i \omega^j + E^j u^j), \\ & D^j u^j \leq \sigma_j^i d^j, \quad \text{for all } j \in \underline{q}\}. \end{aligned} \tag{3.5}$$

Next we show how the freedom to add constraints to the households' sets allows the consideration of piecewise linear utility functions. Suppose $u_i(z^i)$ is a concave, piecewise linear utility function defined on $\mathbb{R}_+^{S_i}$. Then the epograph of u_i, $E_i = \{(z_0, z) | z_0 \leq u_i(z),\ z \in \mathbb{R}_+^{S_i}\}$, is a convex set which has a piecewise linear boundary. Hence, the epograph of u_i can be written as a polyhedral convex set

$$E_i = \{(z_0, z) | z_0' \leq g_{ij} z + c_{ij},\ z \in \mathbb{R}_+^{S_i},\ j \in \ell_i\}$$

where $q_{ij} \in \mathbb{R}^{S_i}$ and ℓ_i is the number of pieces of linearity for u_i. Thus, the following problems are equivalent

$$\begin{aligned} &\text{maximum} && u_i(z^i) \\ &\text{subject to} && \pi(B^i z^i) \leq \pi w^i,\ z^i \geq 0 \end{aligned} \tag{3.6}$$

$$\begin{aligned} &\text{maximum} && z_0^i \\ &\text{subject to} && z_0^i - g_{ij} z^i \leq d_{ij}, \quad j \in \underline{\ell_i} \\ & && \pi(B^i z^i) \leq w^i,\ z^i \geq 0 \ . \end{aligned}$$

In this case $\gamma^i = (1,0,\ldots,0)$ and the matrix $[A^i | a^i]$ in the auxiliary linear program (3.1) can be adjoined to the matrix

$$[e, -g_{i1}, -g_{i2}, \ldots, -g_{i\ell} | c_i] \ .$$

In the next two chapters we describe algorithms which utilize the structure of the models which were described here.

CHAPTER III

THE BILINEAR COMPLEMENTARITY PROBLEM AND AN ALGORITHM FOR COMPUTING ECONOMIC EQUILIBRIA

This chapter presents a generalization of the problem described in (II.2.3) and (II.2.4) which is called the bilinear complementarity problem. This approach to computing equilibria was introduced by Wilson [1976]. In the first section we essentially reproduce the results of that paper except that we use a new proof based on the concept of a subdivided complex introduced in Chapter II. The rest of the chapter describes a bilinear complementarity algorithm in enough detail to be implemented on a computer. The results of computational experiments using this first implementation of a bilinear complementarity algorithm are reported in Chapter V.

III.1. The Bilinear Complementarity Problem

The bilinear complementarity problem (BCP) is to find $x, y \geq 0$ such that

$$Ax + y = b, \quad A \in \mathbb{R}^{n \times n}, \quad b \in \mathbb{R}^n,$$

$$\langle x, C_{\cdot,i} \rangle - x_i y_i = 0, \qquad i \in \underline{m}$$

$$x_i y_i = 0, \qquad i \in \underline{n} \setminus \underline{m} .$$

Notice that if $m = 0$, the problem is the same as a linear complementarity problem (LCP) (cf. C. Lemke [1965] or B. C. Eaves [1971b]).

It is easy to verify that the equilibrium problem, as we have formulated it in (2.3), (2.4) can be stated in this form. It is an area of interest to determine what other problems can be formulated as a BCP other than the equilibrium problem or those which can be formulated as linear complementary problems. We present the general problem mainly to ease the notational burden.

The constraints (4.1) are weakened for algorithmic purposes by adding slack variables $u_i \geq 0$, $i \in \underline{m}$ to the bilinear constraints: $\langle x, C_{\cdot,i} \rangle - x_i y_i = u_i$, $i \in \underline{m}$, where $C \in \mathbb{R}^{n \times m}$.

Now we define the set

$$W = \{(x,y,u) \geq 0 \mid Ax + y = b, \langle x, C_{\cdot,i} \rangle + x_i y_i = u_i,\ i \in \underline{m},\ x_i, y_i = 0,\ i > m\}, \tag{1.2}$$

and for $k = 0,1,\ldots,m$ let W_k be the subset of W for which $u_i = 0$ for $i \leq k$, and $x_i y_i = 0$ for $i > k$. A solution to the LCP defined by (A,b) yields a point in W_0 if $x^T C \geq 0$. The solution to the BCP is a point in W_m. The bilinear complementarity algorithm (BCA), to be defined in the next section provides a procedure for tracing a path from W_0 to W_m provided that certain assumptions are satisfied.

We shall define the "algorithm" by defining a mapping F from an $n+m+1$-dimensional subdivided complex into $\mathbb{R}^{n+m}$ and proving that $F^{-1}(0)$ is a path with the desired properties when 0 is a good value. First we define a collection of sets and show that it is a subdivided complex.

A basis β is a subset of the variables $\{X_i, Y_i,\ i \in \underline{n},\ U_i,\ i \in \underline{m}\}$ (capital X_i can be considered as an index or a name for x_i, etc.) consisting of (X_i, Y_i) for $i \leq k$ for some $0 \leq k \leq m$, (X_i, U_i) or (Y_i, U_i) for $k < i \leq m$, and exactly one member of (X_i, Y_i) for $i > m$. This definition is slightly more restrictive than that in Wilson [1976]. A path-basis $\beta(i)$ for $i = k+1$ is obtained by adding the $(k+1)^{st}$ element to a basis β. A sub-basis $\beta(i,j)$ for $i \neq j$ is obtained by deleting a j^{th} member of $\beta(i)$.

Now if $\bar{\beta}$ is a basis, path basis or subbasis, then $\mathcal{O}(\bar{\beta})$ is the non-negative orthant of $\mathbb{R}^{2n+m}$ indexed by the set of variables $\bar{\beta}$. Note that if $\hat{\beta}$ is a basis or sub-basis, then $\mathcal{O}(\hat{\beta})$ is an n+m-dimensional set, and if $\hat{\beta}$ is a path basis, then $\mathcal{O}(\bar{\beta})$ is an n+m+1-dimensional set. Also, a facet of $\hat{\beta}$ is an orthant corresponding to a basis or path basis. Define the collection of cells

$$\mathcal{M}^0 \equiv \{\mathcal{O}(\bar{\beta}) \mid \bar{\beta} \text{ is a path basis}\} . \tag{1.3}$$

The following assumptions are of vital importance.

Assumption 1.1: In W, $\langle x, C_{\cdot,i} \rangle > 0$ for each $i \in \underline{m}$.

In the equilibrium problem, this property is satisfied by Lemma 2.5, which depended upon the positivity of w^i and the choice of v^i, $i \in \underline{m}$.

Assumption 1.2: The subset $W^* = \{(x,y,u) \in w \mid \sum_{i=1}^{m} u_i > 0\}$ is bounded.

The conditions on the economic problem sufficient for assumption 1.2 to hold will be studied in the next chapter. This assumption implies that we can confine our attention to the compact set

$$K = \{(x,y,u) \in \mathbb{R}^{2n+m} \mid 0 \leq (x,y,u) \leq ke\}$$

where $k>0$ is a sufficiently large constant.

$$\mathcal{M} = \{\mathcal{O}(\bar{\beta}) \cap K \mid \bar{\beta} \text{ is a path basis}\} . \tag{1.4}$$

and assume that there are no redundancies in the problem so that each cell C of $\mathcal{M}$ is m+n+1 dimensional. Define $M = \cup_{C \in \mathcal{M}} C$. Now we can prove the

Lemma 1.3. $(M, \mathcal{M})$ is a finite, compact subdivided (m+n+1)-complex.

Proof. The number of path bases is less than $\binom{2n+m}{n-1}$ which is finite. $\mathcal{O}(\bar{\beta})$ is closed and K is compact so, by (1.4) each cell $C \in \mathcal{M}$ is a compact (m+n+1)-cell. All that remains to show is that $(M, \mathcal{M})$ satisfies the definition for a subdivided complex (II.3.2, Part 1).

Since there are only a finite number of (m+n+1)-cells in $\mathcal{M}$, property (c) is satisfied.

By the definition of W, the facets of a cell C of $\mathcal{M}$ correspond to some variable x_i (or y_i or u_i) being held at zero, where $X_i \in \bar{\beta}$ and $C = \mathcal{O}(\bar{\beta}) \cap \bar{W}^*$. Thus, since we can associate

each cell $C \subset \mathcal{M}$ with a basis of sub-basis, we can associate each cell $C' \subset C \subset \mathcal{M}$ with a set of variables β not constrained to be zero. Define $\varphi(C) = \bar{\beta}$ to be the set of variables associated with C and let $\varphi^{-1}(\bar{\beta}) = C$ be the cell associated with $\bar{\beta}$.

a) (any two (m+n+1)-cells of $\mathcal{M}$ are disjoint or meet in a common face). Let $C_1, C_2 \in \mathcal{M}$, and $\varphi(C_i) = \beta_i$, $i = 1,2$. If $\beta_1 \cap \beta_2 = \emptyset$, then $C_1 \cap C_2 = \emptyset$, a common face. Otherwise, it is clear from the definition of face, that if $\beta_1 \cap \beta_2 = \bar{\beta}$, then $\bar{C} = \varphi^{-1}(\bar{\beta})$ is a face for both C_1 and C_2.

b) (Each (m+n)-cell of $\mathcal{M}$ lies in at most two (m+n+1)-cells.)

By definition, any (m+n)-cell, say C_2 of $\mathcal{M}$ is the face of some (m+n+1)-cell of $\mathcal{M}$, say C_1. Suppose $\beta_2 = \varphi(C_2)$ and $\beta(k) = \varphi(C_1)$ then there are four cases to consider concerning the type of variable which is in $\beta(k)$ but not β_2.

1) X_i or Y_i for $i < k$. In this case there is no path-basis which can be formed by adding a variable to β_3. Hence, there is no (m+n+1)-cell, other than C_1, which contains C_2. (Note that $C_3 \subset \partial M$ in this case.)

2) X_k, Y_k, or U_k.

2a) If X_k or Y_k is in $\beta(k)$ but not β_2, then β_2 is a basis. By definition the only path basis which can be formed is by adding U_{k-1} to β_3; let $\beta_3 = \beta_2 \cup \{U_{k-1}\}$. Then $C_3 = \varphi(\beta_3)$ is the only other cell containing C_2.

2b) If $\{U_k\} = \beta(k) \setminus \beta_3$, then the only adjacent cell containing C_2, other than C_1, corresponds to the path-basis formed by adding the

$(k+1)^{st}$ variable (X_{k+1} or Y_{k+1}) which is not in β_2.

3) X_i, Y_i, or U_i for $k < i \leq m$.

3a) If X_i or Y_i is in $\beta(k)$ but not in β_2, then since $U_i \in \beta_2$, the adjacent cell is identified with the path-basis formed by adding the i^{th} variable which was not in $\beta(k)$.

3b) If U_i is in $\beta(k)$ but not in β_3, then there is no adjacent cell to C_1 which contains C_3, i.e., $C_3 \subset \partial M$.

4) X_i or Y_i for $m < i \leq n$.

If $\{X_i\} = \beta(k) \setminus \beta_3$, then $C_2 = \varphi(\beta_2)$, where $\beta_2 = \beta_3 \cup \{Y_i\}$, is the only $(m+n+1)$-cell, other than C_1, which contains C_3. The situation is analogous when $\beta(k) \setminus \beta_2 = \{Y_i\}$. ∎

Notice that the adjacency of cells and indeed the definition of $(M, \mathcal{M})$ is dependent upon the ordering of the households $i = 1,2,\ldots,m$. Thus, the efficiency of the algorithm is influenced by the ordering of the households.

Let the function $F = M \to \mathbb{R}^{n+m}$ be denoted by

$$F(x,y,u) = \begin{pmatrix} Ax + y - b \\ \langle C_{\cdot,i}, x \rangle - x_i y_i - u_i, \; i \in \underline{m} \end{pmatrix} .$$

F is smooth on each $(m+n+1)$-cell $C \subset \mathcal{M}$. For the next theorem, we require the

Assumption 1.4. The cardinality of W_0 is finite and odd.

Conditions sufficient to ensure that Assumption 1.4 hold have been supplied by various authors (e.g., B. C. Eaves [1971b]). Now we prove the main result.

Theorem 1.5. If $0 \in \mathbb{R}^{m+n}$ is a good value for F with respect to $(M, \mathcal{H})$ and Assumptions 1.1, 1.2, and 1.4 are satisfied, then

i) $F^{-1}(0)$ is a subdivided 1-complex neat in $(M, \mathcal{H})$.

ii) The cardinality of the set of solutions to the BCP is finite and odd.

iii) There is at least one connected component of $F^{-1}(0)$ with one boundary point in W_0 and the other in W_m.

Proof. i) follows immediately from Theorem II.3.17 (Part 1).

Due to Proposition II.3.18 (Part 1), and Assumption 1.4, ii) will be demonstrated if it can be shown that all boundary points of $F^{-1}(0)$ are in W_0 and W_m.

The details of the proof of Lemma 1.3 will be used here. Remember that $\beta(1)$ refers to a path basis with $t_1 \cdot \lambda_1 > 0$ and $t_i \cdot \lambda_i = 0$ for $i > 1$. If the face C_2 of $C_1 = \varphi(\beta(0))$ has $t_i \cdot \lambda_i = 0$, then by case 2a), C_1 is the only cell containing C_2. Hence, $C_2 \subset \partial M$. The union of all such facets will be called f^0. Similarly, if C_2 is a face of $C_1 = \varphi(\beta(m))$ such that $u_i = 0$, $i \in \underline{m}$, then C_1 is the only cell containing C_2 (see Case 2b). These facets in the boundary of M will be called f^m. It is clear from the definitions that $W_0 = F^{-1}(0) \cap f^0$ and $W_m = F^{-1}(0) \cap f^m$. Since

$F^{-1}(0)$ is neat in M, if we can show that $F^{-1}(0) \cap \partial M = W_0 \cup W_m$, then since $|W_0|$ is odd and $|\partial F^{-1}(0)|$ is even, we will have shown that $|W_m|$ is odd.

From the proof of Lemma 1.3 it is evident that the only facets in ∂M are either in f^0, f^m, or of the type described in 1) or 3b). Suppose $(\bar{x},\bar{y},\bar{u}) \in F^{-1}(0)$ is in a cell of the type described in 1). Then $\bar{x}_i \cdot \bar{y}_i = 0$ and $0 = u_i = \langle \bar{x}, C_{i,.} \rangle$ which contradicts Assumption 1.1. Suppose $(\bar{x},\bar{y},\bar{u}) \in F^{-1}(0)$ is in a cell of the type described in 3b). Again, this is impossible because $\bar{x}_i \cdot \bar{y}_i = 0$ and $\bar{u}_i = 0$ is a contradiction.

Thus $F^{-1}(0) \cap \partial M = W_0 \cup W_m$ and ii) is proved.

Since each connected component of $F^{-1}(0)$ with a boundary point in f^0 has another boundary point in either f^0 or f^m, a simple counting argument will demonstrate (iii). ∎

We now show how the linear pure exchange equilibrium problem can be formulated as a BCP; the corresponding formulation for the model which includes production is analogous.

The linear constraints in (2.3) define a system of n linear equations in 2n variables $Ax + y = b$ where $n = m + \ell + \sum_{i=1}^{m} s_i + 1$ and A is a square skew-symmetric matrix of the form

$$A \equiv \begin{bmatrix} 0 & D \\ -D^T & 0 \end{bmatrix} \tag{1.5}$$

and

$$D = \begin{bmatrix} -r^1 & 0 & \cdots\cdots & 0 \\ & -r^2 & & 0 \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ & & & -r^m \quad 0 \\ B^1 & B^2 & \cdots\cdots & B^m \quad e \end{bmatrix} ,$$

$$b = \begin{bmatrix} -v_1 \\ \vdots \\ -v_m \\ \sum_{i=1}^{m} w_i \\ \hline 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} .$$

The variables $x, y \in \mathbb{R}^n$ have the form

$$x = [\lambda_i \ (i \in \underline{m}), \ \pi; \ z^i \ (i \in \underline{m}), \ p]$$

$$y = [t_i \ (i \in \underline{m}), \ s; \ \zeta^i \ (i \in \underline{m}), \ \rho]$$

In this formulation p is constrained to be non-negative, so in the dual problem

$$\pi e - \rho = 1 .$$

But since $\sum u_i = p > 0$ except for points in W_m, $\rho = 0$ is satisfied until a solution to the BCP has been attained.

The bilinear constraints (2.4) are expressed by

$$\langle C_{.,i}, x \rangle - x_i y_i = 0, \qquad i \in \underline{m}$$

where

$$C_{.,i} = [0, \ldots, \underset{\substack{\uparrow \\ i^{th} \text{ element}}}{-v_i}, 0, \ldots, 0, w^i; \underset{\substack{\uparrow \\ \in \mathbb{R}^{\sum s_i + 1}}}{0}]^T$$

Corollary 1.6. If the primal and dual linear programs (V.2.1) and (V.2.2) have unique solutions, then, that solution $(x^0, y^0, u^0) \in W_0$ is a boundary point of some path $\gamma \subset F^{-1}(0)$. The other boundary point (x^*, y^*, u^*) of γ is a solution to the equilibrium problem (V.2.3, 2.4).

Proof. After observing that any point in W_0 is a solution to the linear programs (2.1) and (2.2), iii) of Theorem 1.5 shows that $(x^*, y^*, u^*) \in W_m$. Hence $u^* = 0$ and the constraints (2.3) and (2.4) are satisfied. ∎

III.2. The Bilinear Complementarity Algorithm

In the remainder of this chapter we explain in detail an algorithm for following the path, described in Section 1, which leads to an equilibrium.

The algorithm described here solves an equilibrium problem as formulated in Sections 2 and 3 of Chapter V, not the bilinear complementarity problem in its most general form. The reasons for this restriction are twofold: 1) we have not found any other classes of problems which can be formulated as BCP's, 2) The dimension of the linear systems of equations can be reduced significantly by dealing with D rather that A in equation (1.5). We still refer to this algorithm as the Bilinear Complementarity Algorithm (BCA) because the complementary property (II.2.7) involving $\lambda_i t_i$ and u_i is what motivates the definition of the algorithm.

In the remainder of this section we give a more general description of a bilinear complementarity algorithm. In the following sections we describe in detail the linear operations we use to reduce the dimension of the path following subproblem and a specific method for solving that subproblem. The algorithm we describe here is implemented in a computer program (BCA). The details of the implementation and a report on the numerical results using this code will be in given in Chapter V.

Next we describe in more detail the overall structure of the algorithm. This structure can be inferred from the proof that $(M, \mathcal{M})$ is a subdivided m-complex and the proof that $F^{-1}(0)$ leads through the cells of $\mathcal{M}$ to an equilibrium point in Section 1. However, to make the strategy for determining which cell (or path-basis) is adjacent a little clearer we will now give a verbal description with some visual aids.

A basic point in W is in W_d for some $d = 0,1,\ldots,m$. It is characterized by the fact that $u_i = 0$ for $i \leq d$ and $u_i > 0$

for $i > d$ and that $x_i y_i > 0$ for $i \leq d$ and $x_i y_i = 0$ for $i = d+1, \ldots, n$. It is useful to depict the variables which are positive and zero in a chart:

	1	2	3	4	5	6	7	8	9	10
x	+	+	+	0	+	+	0	0	0	+
y	+	+	0	+	0	0	+	+	+	0
u	0	0	+	+						

FIGURE 2.1

The difficult transitions to understand are those for which the current facet which $F^{-1}(0)$ intersects is a basis. Suppose that the last cell which the algorithm passed through had $u_2 > 0$ and Figure 2.1 represents the fact that u_2 just hit zero. The next step is to introduce either X_3 or Y_3 into the path-basis, whichever is not in the current basis. In this case, y_3 is now allowed to be positive.

	1	2	3	4	5	6	7	8	9	10
x	+	+	+	0	+	+	0	0	0	+
y	+	+	+	+	0	0	+	+	+	0
u	0	0	+	+						

FIGURE 2.2

Now we follow $F^{-1}(0)$ through the cell defined by the current path basis until some other variable goes to zero. We hope that u_3 hits zero so that we move towards an equilibrium where $u_i = 0$, $i \in \underline{m}$. Suppose that x_i or y_i hits zero for some $i \geq 4$. Then the adjacent path basis is formed by adding the variable complementary to the one that hit zero, i.e., if y_8 hits zero, introduce x_8. It is impossible for x_i or y_i to hit zero for $i < 3$ because $u_i = 0$ implies $x_i \cdot y_i > 0$. Similarly, u_4 cannot hit zero because $x_4 \cdot y_4 = 0$ implies $u_4 > 0$ (recall II(2.7)). There is one more possibility. If x_3 or y_3 hits zero, then we are again at a basic point, and, according to the definition of a path-basis, the only adjacent path-basis is formed by introducing u_2. Since u_2 is no longer constrained to zero, one could say that we have "back-tracked" in our goal of forcing all the budget surpluses to zero. However, the theory of Section 1 says that this goal will be reached, eventually. In fact, by the proof of Proposition II.3.18 (Part 1), only a finite number of cells need be traversed before an equilibrium point is reached.

Below is a more complete specification of the decisions involved in choosing the adjacent cell.

Bilinear Complementarity Algorithm

0. Initialize: $\omega_0 = (x^0, y^0, u^0)$ is determined from the solution of the linear program (II.2.1), with the associated basis β_0. m is the number of consumers. Let $j := 0$, $d := 1$.

1. If $d = m+1$, STOP - ω_j is an equilibrium point. Otherwise let $\bar{\beta}_j := \beta_j \cup (\{X_d\}$ or $\{Y_d\})$, whichever is not in β_j. Go to Step 4.

2. $\bar{\beta}_j := \beta_j \cup U_d$, go to 4.

3. If $\{Y_i\} = \bar{\beta}_{j-1} \setminus \beta_j$, then $\bar{\beta}_j = \beta_j \cup \{X_i\}$.

   If $\{X_i\} = \bar{\beta}_{j-1} \setminus \beta_j$, then $\bar{\beta}_j = \beta_j \cup \{Y_i\}$.

4. Follow the path $(F|_{\mathcal{O}(\bar{\beta}_j)})^{-1}(0)$ from ω_j into the interior of $\mathcal{O}(\bar{\beta}_j)$ until the curve intersects a facet, call it $\mathcal{O}(\beta_{j+1})$. Let ω_{j+1} be the point in that intersection. $j := j+1$.

5. a) If $\omega_j \in W_d$, $d := d+1$, go to 1; else,

   b) If $\omega_j \in W_{d-1}$, $d := d-1$, go to 2; else,

   c) ω_j is a sub-basic point, go to 3.

III.3. Exploiting the LP Structure

We shall now change from the notation of the bilinear complementarity problem to a linear programming type of notation. It will be important to distinguish between the primal and dual structural variables contained in x and the primal and dual slack variables in y.

The piecewise linear equilibrium problem can be formulated as follows:

find $(x,t,\lambda,\zeta) \geq 0$

such that

$$Dx + It = b \tag{4.1}$$

$$\lambda D - \zeta I = c$$

$$u_i = \langle C_{\cdot,i}, \lambda \rangle - \lambda_i t_i = 0, \quad i \in \underline{m}$$

$$\lambda_i t_i = 0, \quad i \in \underline{k} \setminus \underline{m}$$

$$\langle x, \zeta \rangle = 0,$$

where

$$x, \zeta \in \mathbb{R}^{\ell}, \quad t, \lambda \in \mathbb{R}^{k}, \quad \text{and} \quad D \in \mathbb{R}^{k \times \ell}.$$

It is clear that either the constraints of II(2.3) and (2.4) or the equilibrium conditions in II.3.4 may be expressed as above. The size of k is determined by the number of consumers, the number of goods and the number of exogenous constraints on the consumers and producers.

The first m components of the primal right-hand side are the initial utility levels v_i, $i \in \underline{m}$. To determine these values, one can either solve m smaller linear programs as described in (2.6) or (3.5) of Chapter V or let $v_i = 0$, $i \in \underline{m}$. It is strongly suggested that the former course be taken because good starting values will significantly reduce the run time of the BCA, and the choice of $v_i = 0$ will usually result in a highly degenerate solution to the linear program, which may cause problems with the BCA. Once these parameters have been established, the initial point $\omega^0 = (x^0, \lambda^0, t^0, \zeta^0, u^0)$ is determined by solving the auxiliary linear program

$$\begin{aligned} &\text{maximize} && cx \\ &\text{subject to} && Dx + t = b \\ & && x, t \geq 0, \end{aligned} \tag{4.2}$$

to find (x^0, t^0). If the revised simplex method (Dantzig [1963]) is used, the optimal shadow prices λ^0 and relative costs ζ^0 can be extracted. Finally, the initial budget surpluses

$$u_i^0 = \langle C_{\cdot,i}, \lambda^0 \rangle - \lambda_i^0 t_i^0 \quad , \quad i \in \underline{m}$$

can be determined. By the complementary slackness condition of the optimal primal and dual variables, $\lambda_i^0 \cdot t_i^0 = 0$, $i \in \underline{m}$, and, hence, $u_i^0 > 0$ for each $i \in \underline{m}$. Let σ^0 be the index set of the k basic primal variables and $\bar{\sigma}^0 = \underline{n} \setminus \sigma^0$ be the ℓ basic dual variables. Then the initial (BCA) basis β^0 consists of $(x,t)_{\sigma^0}$, the primal basic variables, $(\lambda,\zeta)_{\bar{\sigma}^0}$, the dual basic variables, and u_i, $i \in \underline{m}$.

The non-basic variable t_1 will be added to β^0 to form the first path basis $\bar{\beta}^0$. The path which the BCA follows is determined by $(F|_{\mathcal{O}(\bar{\beta}^0)})^{-1}(0)$ where $F|_{\mathcal{O}(\bar{\beta}^0)} = \mathbb{R}_+^{m+n+1} \to \mathbb{R}^{m+n}$ is defined in Section 1. It would be possible to follow this path through the $(m+n+1)$ dimensional cell $\mathcal{O}(\bar{\beta}^0)$, but we can use the largely linear structure of (4.1) and the positivity of $\langle C_{\cdot,i}, \lambda \rangle$, $i \in \underline{m}$, to reduce the dimension of the path following portion of the algorithm. In a typical path-basis $\bar{\beta}$ there are n variables which are basic for either the primal or dual problem and $m+1$ other variables. There are d variables, where d has the same value as in the description of the BCA, among $(\lambda_i, i \in \underline{m})$, $(t_i, i \in \underline{m})$ which are in β^0 but not in the linear programming basis, call them λ_μ and t_ν. Also in β^0 are the $m-d+1$ budget surpluses u_i which are positive. Using the primal and dual basis inverses, we can represent the n basic variables in terms of the d variables λ_μ and t_ν, so we have reduced the dimension by n. The variables $u_{d+1}, \ldots, u_m$ must stay positive because $\lambda_i \cdot t_i = 0$, $i = d+1, \ldots, m$, so we can ignore them.

The variable u_d and the constraint $u_d = \langle C_{\cdot,d}, \lambda \rangle - \lambda_d \cdot t_d$ can be ignored if we treat $\langle C_{\cdot,d}, \lambda \rangle - \lambda_d t_d \geq 0$ as a constraint helping to define the cell we are in. Thus, we have reduced the path following problem to one in the d variables λ_μ and t_ν and the $d-1$ equations, $\langle C_{\cdot,i}, \lambda \rangle - \lambda_i t_i = 0$, $i \in \underline{d-1}$, corresponding to $u_i = 0$, $i \in \underline{d-1}$. Next we describe the details of how this reduction of dimension is carried out in practice.

In linear programming codes it is common to maintain two integer vectors which can pick out the basic variables and tell which row of the matrix they pivot on. The row a basic column "pivots on" is the index corresponding to the winner of the min-ratio test performed when that column entered the basis. We shall also maintain analogous index vectors for the dual variables. To summarize, the following information is maintained and revised during the course of the BCA.

$\sigma(i)$ = the primal basic variable which pivots on row i, $i \in \underline{k}$

$\bar{\sigma}(i)$ = the dual basic variable which pivots on row i, $i \in \underline{\ell}$.

$$\tau(j) = \begin{cases} 0 & \text{if } j \notin \sigma \\ i & \text{if } j \in \sigma \text{ and pivots on row i,} \end{cases} \qquad j \in \underline{n}.$$

$$\bar{\tau}(j) = \begin{cases} 0 & \text{if } j \notin \bar{\sigma} \\ i & \text{if } j \in \bar{\sigma} \text{ and column } j \text{ pivots on row i,} \end{cases} \qquad j \in \underline{n}.$$

μ = index set of those λ-variables not in $\bar{\sigma}$, but allowed to be positive

ν = index set of those t-variables not in σ, but allowed to be positive

d = the number of pairs (λ_i, t_i) which are both allowed to be positive.

Remark. a) $\mu \cup \nu = \underline{d}$

b) The current path basis $\bar{\beta}$ can be written

$$\bar{\beta} = \{(x,t)_{\sigma},\ (\lambda,\zeta)_{\bar{\sigma}},\ \lambda_{\mu}, t_{\nu},\ u_{d},\ u_{d+1},\ \ldots\ ,\ u_{m}\}.$$

The variables λ_{μ} and t_{ν} will be referred to as superbasic variables (to follow the terminology of Murtagh and Saunders [1977] in reference to the nonbasic but positive variables in their GRG algorithm). They are independent variables which determine the values of the dependent or LP-basic variables. Next we show how the dependence can be numerically calculated.

Let the primal $k \times k$ basis matrix P be partitioned as

$$\begin{array}{c} \quad x_{\sigma} \quad t_{\sigma} \\ \bar{P} = \left[\begin{array}{c|c} B & 0 \\ \hline A & I \end{array}\right] \end{array}$$

with the columns of P permuted so the basic slack columns are on the right and the rows they pivot on are at the bottom. One can write

$$\bar{D} = \left[\begin{array}{c|c} B & E \\ \hline A & F \end{array}\right] \qquad \text{and} \qquad \bar{P}^{-1} = \left[\begin{array}{c|c} B^{-1} & 0 \\ \hline -AB^{-1} & I \end{array}\right] \tag{4.2}$$

Dividing the primal system into basic and non-basic variables, we have

$$\left[\begin{array}{c|c} B & 0 \\ \hline A & I \end{array}\right] \begin{pmatrix} x_{\sigma} \\ t_{\sigma} \end{pmatrix} + \left[\begin{array}{c|c} E & I \\ \hline F & 0 \end{array}\right] \begin{pmatrix} x_{\bar{\sigma}} \\ t_{\bar{\sigma}} \end{pmatrix} = b\ .$$

If we want to see the effect of t_j, $j \in \nu$ on the basic variables,

one merely updates e_j, a column of I, in the usual manner. That is solve

$$\bar{P}s = e_j \ , \qquad \bar{P}\bar{b} = b \ , \tag{4.3}$$

so that

$$\begin{pmatrix} x_{\bar{\sigma}} \\ t_{\sigma} \end{pmatrix} = - [S] \, t_j + \bar{b} \ .$$

Clearly, the permutation of the rows and columns of P was not necessary in this case, because we are assuming that the primal basis is factored so that it is easy to solve systems such as (4.3). It is useful, however, to use the permuted from of D to describe the calculations in the dual system. One must keep in mind that, in practice, D is not actually permuted: the index sets σ and γ allow one to pick out the elements needed to perform the calculations below. We write the dual system in terms of its basic variables in $\bar{\sigma}$ and its non-basic variables in σ.

$$(\lambda_{\bar{\sigma}}, \zeta_{\bar{\sigma}}) \left[\begin{array}{c|c} B & E \\ \hline 0 & -I \end{array}\right] + (\lambda_{\sigma}, \zeta_{\sigma}) \left[\begin{array}{c|c} A & F \\ \hline -I & 0 \end{array}\right] = c = (c^1, c^2) \quad .$$

Multiplying by the dual basis inverse,

$$\left[\begin{array}{c|c} B & E \\ \hline 0 & -I \end{array}\right]^{-1} = \left[\begin{array}{c|c} B^{-1} & B^{-1}E \\ \hline 0 & -I \end{array}\right] \ ,$$

yields the updated tableau

$$(\underline{\lambda}_{\sigma}, \underline{\zeta}_{\sigma}) + (\lambda_{\sigma}, \zeta_{\sigma}) \left[\begin{array}{c|c} AB^{-1} & AB^{-1}E-F \\ \hline -B^{-1} & -B^{-1}E \end{array}\right] \geq (c^1B^{-1},\ c^1B^{-1}E - c^2) \ . \qquad (4.4)$$

Suppose we want to determine the effect of λ_j, $j \in \mu$, upon the basic variables; i.e., we want to calculate the appropriate row of $(AB^{-1},\ AB^{-1}E - F)$. Now, $-AB^{-1}$ is part of the primal basis. Let e_j be the j^{th} unit vector and solve $s^TP \geq e_j^T$. Then pick out the components of s^T which correspond to the x_σ columns of P $(\sigma(i) \leq \ell)$ and call the result $\bar{s}_1^T$. To get the appropriate row of $AB^{-1}E-F$, just calculate

$$-\bar{s}_2^T \equiv - \bar{s}_1^TE - F_j \ .$$

Then we have

$$(\underline{\lambda}_{\sigma}, \underline{\zeta}_{\sigma}) = \lambda_j(\bar{s}_1^T, \bar{s}_2^T) + (c_1B^{-1},\ c_1B^{-1}E - c_2) \ , \qquad (4.5)$$

If we want to update the row corresponding to ζ_j for some $j \in \sigma$, the procedure is very similar. We want to calculate the appropriate row of $(B^{-1}, B^{-1}E)$. The desired row of B^{-1} is that row corresponding to the column of B associated with x_j, row $\gamma(j)$. Suppose this is the q^{th} row of P. Then $(B^{-1})_{q.}$ is found by solving

$$s^TP = e_q^T$$

and extracting the components s_i^T such that $\sigma(i) \leq \ell$ to form $\bar{s}_1^T$. Let $\bar{s}_2^T = \bar{s}_1^TE$ and we have

$$(\lambda_{\sigma}, \zeta_{\sigma}) = \zeta_j(\bar{s}_1^T, \bar{s}_2^T) + (c_1 B^{-1},\ c_1 B^{-1} E - c_2) \quad . \tag{4.6}$$

The set of updated columns corresponding to t_ν will be $G1 \in \mathbb{R}^{k \times |\nu|}$ and the transpose of the updated rows corresponding to λ_μ will be $G2 \in \mathbb{R}^{\ell \times |\mu|}$. The following equations give us the basic variables in terms of the superbasics

$$\begin{pmatrix} x_\sigma \\ t_\sigma \end{pmatrix} = G1 t_\nu + \bar{b} \geq 0$$

$$\begin{pmatrix} \lambda_{\bar{\sigma}} \\ \zeta_{\sigma} \end{pmatrix} = G2 \lambda_\mu + \bar{c} \geq 0 \tag{4.7}$$

We have described how G_1 and G_2 can be computed from scratch; below we will describe how G_1 and G_2 are updated as the basis changes and superbasic variables are added and dropped.

First, however, we describe how the first d-1 bilinear equations (4.1) are expressed as $f(\lambda_\mu, t_\nu)$, a function of the superbasic variables. Only the bilinear equations which are binding ($u_i = 0$, $i \in \underline{d\text{-}1}$) are included in f. Hence, f maps $\mathbb{R}^d$ into $\mathbb{R}^{d-1}$. The inequality, $\langle C_{\cdot,d}, \lambda \rangle - \lambda_d t_d \geq 0$, referred to on page 42 is called the <u>bilinear inequality</u>. It also will be reduced to a function

$$q(\lambda_\mu, t_\nu) \geq 0 \tag{4.8}$$

of the superbasics. This inequality, along with the inequalities in (1.7) determine the cell which the algorithm is currently concerned with.

By substituting the correct expression in (4.7) for the basic primal and dual variables involved in the first d bilinear functionals, we can write

$$\begin{aligned} u_\mu &= h_1 + D_1\lambda_\mu - \mathrm{diag}(\lambda_\mu)(F_1 t_\nu + e_1) \\ u_\nu &= h_2 + D_2\lambda_\mu - \mathrm{diag}(t_\nu)(F_2\lambda_\mu + e_2) \ , \end{aligned} \tag{4.9}$$

where $\mathrm{diag}(y)$, $y \in \mathbb{R}^n$, is the diagonal $n \times n$ matrix with y as the diagonal. Basically, $f(\lambda_\mu, t_\nu)$ could be written as $f(\lambda_\mu, t_\nu) = (u_\mu, u_\nu)$, but we eliminate the functional corresponding to u_d and let $q(\lambda_\mu, t_\nu) = u_d$ as it is expressed in (4.9).

Now we are in a position to describe a method for executing Step 4 of the BCA.

III.4. The Path-Following Subroutine

It has been shown that the path defined by

$$f^{-1}(0) = \{(\lambda_\mu, t_\nu) \mid f(\lambda_\mu, t_\nu) = 0\}$$

in the cell S as defined in (4.7) and (4.8) is identical to the path $(F|_{\mathcal{O}(\bar{\beta}_j)})^{-1}(0)$ described in Step 4 of the BCA. Thus, we consider an algorithm for following $f^{-1}(0)$ from an initial point $(\lambda_\mu^0, t_\nu^0) \in \partial S$ into the cell S until the opposite boundary point of $f^{-1}(0) \cap S$ is reached.

The algorithm to be described here is an adaptation of the path following algorithm in Section III.3. The adaptations are intended to

speed the progress across a cell on the assumption that a) $f^{-1}(0)$ is nearly linear and b) the cells S are in general so small that the non-linearities of $f^{-1}(0)$ are not significant. The linear approximation to $f^{-1}(0)$ at (λ_μ^0, t_ν^0) is used to make a guess at which facet τ of S $f^{-1}(0)$ will intersect. Newton's method will be used to calculate the intersection of τ with $f^{-1}(0)$.

For simplicity let the variables (λ_μ, t_ν) be represented by $z \in \mathbb{R}^d$. Since the superbasic variables are changing with every change of cell, the association of (λ_μ, t_ν) with z is only temporary. In Chapter III (Part 1) we discuss in detail how the tangent to $f^{-1}(0)$ can be calculated. Suppose that $z_0 \in \partial S$ is our initial point. There calculate $\dot{z}_0$ which satisfies

$$f'(z_0)\dot{z}_0 = 0 ,$$

$$\|\dot{z}_0\| = 1 ,$$

and $\dot{z}_0$ points into S.

We only require that $\dot{z}_0$ have norm one so that we can easily measure distance along

$$T(\alpha) = z_0 + \alpha\dot{z}_0 , \qquad \alpha \geq 0 ,$$

the linear approximation to $f^{-1}(0)$ at z_0.

The next step is to determine how far one can move along $T(\alpha)$ until some facet τ of S is hit. For the linearly defined facets this corresponds to a ratio test in linear programming. We want the smallest positive α such that

$$(G1 \cdot T(\alpha))_i + \bar{b}_i = 0 , \qquad i \in \underline{k} ,$$

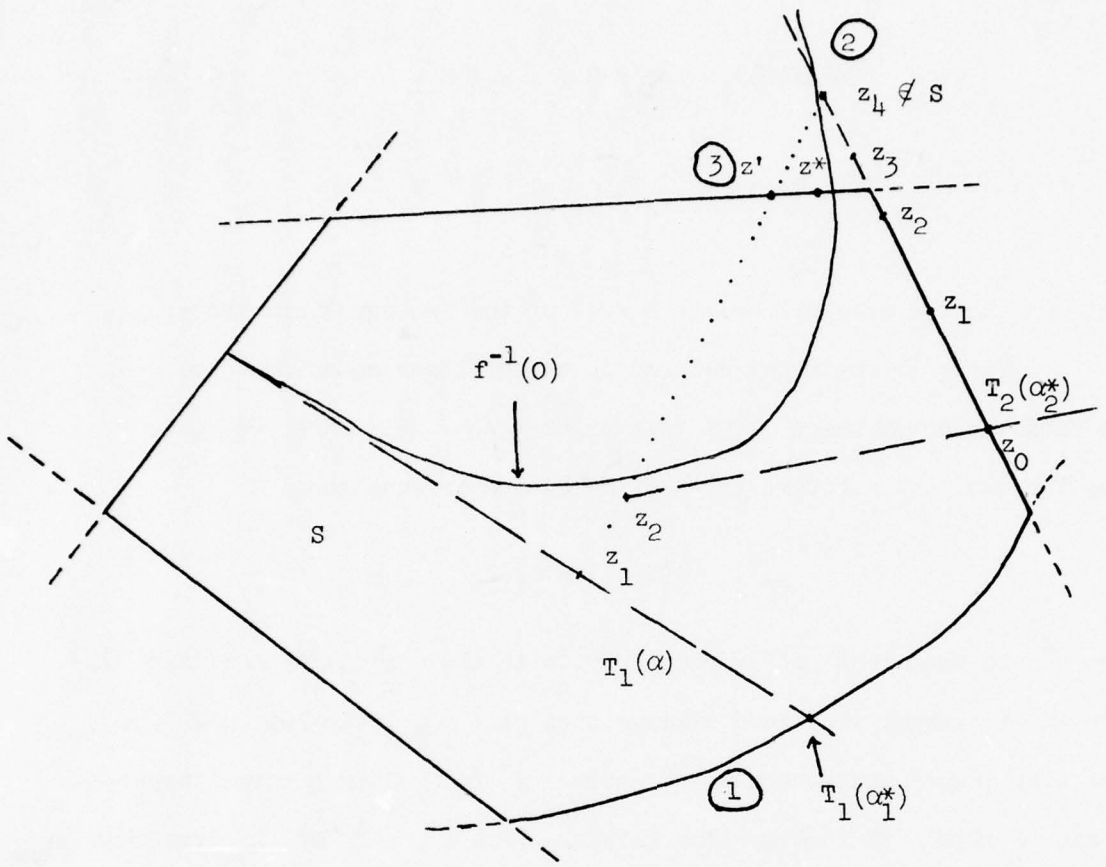
$$(G2 \cdot T(\alpha))_i + \bar{c}_j = 0, \qquad j \in \underline{\ell} .$$

Let α^* be the smallest positive root of the n equations above.

There is one facet defined by the bilinear equation $q(z) = 0$. To find the point where $T(\alpha)$ first intersects this facet we solve for the smallest positive root of the quadratic equation

$$q(T(\alpha)) = 0 .$$

Let $\bar{\alpha}$ be that root, and replace α^* with the minimum of α^* and $\bar{\alpha}$. If α^* is larger than some maximum step size α_{max}, we let $\alpha^*/2$ be the step length and return to the curve $f^{-1}(0)$ along a normal hyperplane to $T(\alpha)$ as in Algorithm III.3.1 (Part 1). If α^* is less than α_{max} we include the equation defining the facet containing $T(\alpha^*) = z^0$ with the $d-1$ functionals in $f : \mathbb{R}^d \to \mathbb{R}^{d-1}$ and use Newton's method to solve that system of equations. If the resulting solution is in S (or nearly so), the desired endpoint of $f^{-1}(0)$ and the desired facet of S have been found. If $z^* \notin S$, then a point z' on the segment $\text{conv}[z_1, z^*]$ is found which is on one of the violated constraints. Newton's method is again initiated at z_3 to find the intersection of $f^{-1}(0)$ with this facet of S. In practice, it is rare for $T(\alpha)$ to pick out the wrong facet, and, if that happens, the procedure described above usually has to be performed only once.



1. $\alpha_1^* > \alpha_{max}$, stepsize is reduced
2. $z_4 \notin S$
3. $z' \in \text{conv}[z_2, z^*]$ is determined.

FIGURE 4.1

Above is an example of the corrective mechanisms of the algorithm for following $f^{-1}(0)$. To find the correct endpoint. It required 2 tangent calculations and 8 Newton iterations. Our computational experience suggests that for most applications of this algorithm only one tangent calculation and one Newton step are required.

Next we give a precise description of the algorithm.

Algorithm 4.2.

0. We are given σ, μ, ν, d, G_1, G_2, f and q as defined in Section 2. Let $z_1 = (\lambda_\mu, t_\nu)$ determine the initial point in a facet of S. The particular facet of S is determined by the pair (pd,s) where

$$pd = \begin{cases} 1\,, & \text{if } \langle ((G_1)_{s,.}, z_1\rangle + \bar{b}_s = 0\,, \\ 0\,, & \text{if } q(z_1) = 0\,, \\ -1, & \text{if } \langle ((G_2)_{s,.}, z_1\rangle + \bar{c}_s = 0\,. \end{cases} \tag{4.1}$$

Call the binding constraint $b_s(z) = 0$. ϵ_1, ϵ_2, $\epsilon_3 > 0$ are fixed parameters. Let $i := 1$.

1. Calculate $f'(z_i) = [H|h] \in \mathbb{D}^{d-1 \times d}$.

2. Solve $Hy = -h$. (Since $f'(z_i)$ is of full rank, if $\det H = 0$, choose another $(d-1) \times (d-1)$ submatrix of $f'(z_i)$, H', and let h' be the vector left over. Set $H := H'$, $h := h'$ and repeat this step.)

3. Let $\bar{\omega} := \operatorname{sgn}(\det H)$, if $i > 1$ go to 5.

4. $\delta := \operatorname{sgn} \langle \nabla b_s(z_1), (y,1)\rangle$.
$\omega := \delta \cdot \bar{\omega}$, go to 6.

5. $\delta := \omega \cdot \bar{\omega}$.

6. $\dot{z}_i := \delta \dfrac{(y,1)}{\|(y,1)\|}$, define $T_i(\alpha) = z_i + \dot{z}_i$.

7. Solve the n linear equations in α defined by

$$\langle (G_1)_{i,.}, T(\alpha) \rangle + \bar{b}_i = 0, \qquad i \in \underline{k}$$

$$\langle (G_2)_{i,.}, T(\alpha) \rangle + \bar{c}_i = 0, \qquad i \in \underline{\ell},$$

and the quadratic defined by

$$q(Q(\alpha)) = 0 .$$

Let α^* be the minimum positive real root of these equations and let (pd,r) define the facet which $Q(\alpha^*)$ is contained in in the same manner as (4.1). Let $b_r(z) = 0$ be the equation defining this facet.

Let $\Delta = 1$ and $z^0 = T(z^*)$

8. Find the minimum positive scalar $\bar{\alpha}$ such that

$$T_i(\alpha) = 0 , \qquad i \in \underline{d}.$$

If $\bar{\alpha} > \alpha^*$ go to 9. Otherwise, let r be the index such that $T_r(\bar{\alpha}) = 0$, and let pd = 1 or -1 depending on whether z_r corresponds to t_r on λ_r. Let $\Delta = 2$, and $\alpha^* := \bar{\alpha}$, $z^0 = T(\alpha^*)$, and $b_r(z) = z_r$.

9. If $\alpha^* < \alpha_{max}$ go to Step 10.

9.5 Otherwise, let $\alpha^* := \alpha^*/2$,

$$z^0 = T(\alpha^*),$$

$$b_r(z) = \langle \dot{z}_i, z^0 - z \rangle ,$$

and $\Delta = 3$.

10. Define

$$g(z) = \begin{pmatrix} f(z) \\ b_r(z) \end{pmatrix}$$

and iterate

$$z^{\ell+1} = z^{\ell} - g'(z^{\ell})^{-1}\, g(z^{\ell}), \qquad \ell = 0,1,\ldots,K$$

until $\|g(z^{\ell+1})\| < \epsilon_2$. If the termination criterion is never realized go to Step 9.5.

11. If $\Delta = 1$ or 2 go to 12, else $\Delta = 3$. Let $i := i+1$, $z_i := z^{\ell+1}$. and go to 1.

12. Let $(z^*) = z^{\ell+1}$. Check that

$$\langle (G_1)_{i.,}, z^* \rangle + \bar{b}_i > -\epsilon_3 , \qquad i \in \underline{k},$$

$$\langle (G_2)_{i,.}, z^* \rangle + \bar{c}_i > -\epsilon_3 , \qquad i \in \underline{\ell} ,$$

and $q(z^*) > -\epsilon_3$. If z^* satisfies these inequalities, go to 13. Otherwise, define pd and r to correspond to the most infeasible constraint and represent that constraint by $b_r(z) \geq 0$. Now solve

$$b_r(\theta z^* + (1-\theta) z_i) = 0$$

for $\theta \in [0,1]$. Let $z^0 = \theta z^* + (1-\theta) z_i$, $\Delta = 2$, and go to Step 10.

13. If pd = 0 and d = m go to Step 14.
Otherwise STOP.

14. To get a more accurate final solution calculate

$$z^* := z^* - g'(z^*)^{-1}\, g(z^*)$$

and repeat until

$$\|g(z^*)\| < \epsilon_1,$$

and then STOP.

We shall call this algorithm the endpoint finding subroutine. Several remarks can be made about this subroutine.

a) The theory underlying steps 1 through 6 is contained in Section III.2 on the orientation of paths.

b) The principal computational effort involved with this subroutine can be divided into two parts: the calculation of Jacobian matrices and the solution of linear systems, and those operations involving the constraints defining S--the calculation of α^* and checking that $z^* \in S$.

c) The former operations are of order $\mathcal{O}(d^3)$ while the latter are $\mathcal{O}(d \cdot n)$. If n is very large in comparison with d (as is usually the case), then the latter type of operation takes more time than the former. Since d increases as the BCA runs its course, the work done in this endpoint subroutine increases.

d) One could apply Theorem III.3.16 (Part 1) to state that if α_{max} is chosen small enough, this subroutine will follow the right path and converge to the right endpoint. In general we choose $\alpha_{max} = 10$ or 100

depending on the problem. The Newton iterations cannot be guaranteed to converge with such large step lengths, but they always have.

3) Since we are using exact partial derivatives in Newton's method we can state that these subroutines converge quadratically to the endpoint z^* of $f^{-1}(0)$ (cf. Ortega and Rheinboldt, 10.2.2, [1970]). This means that

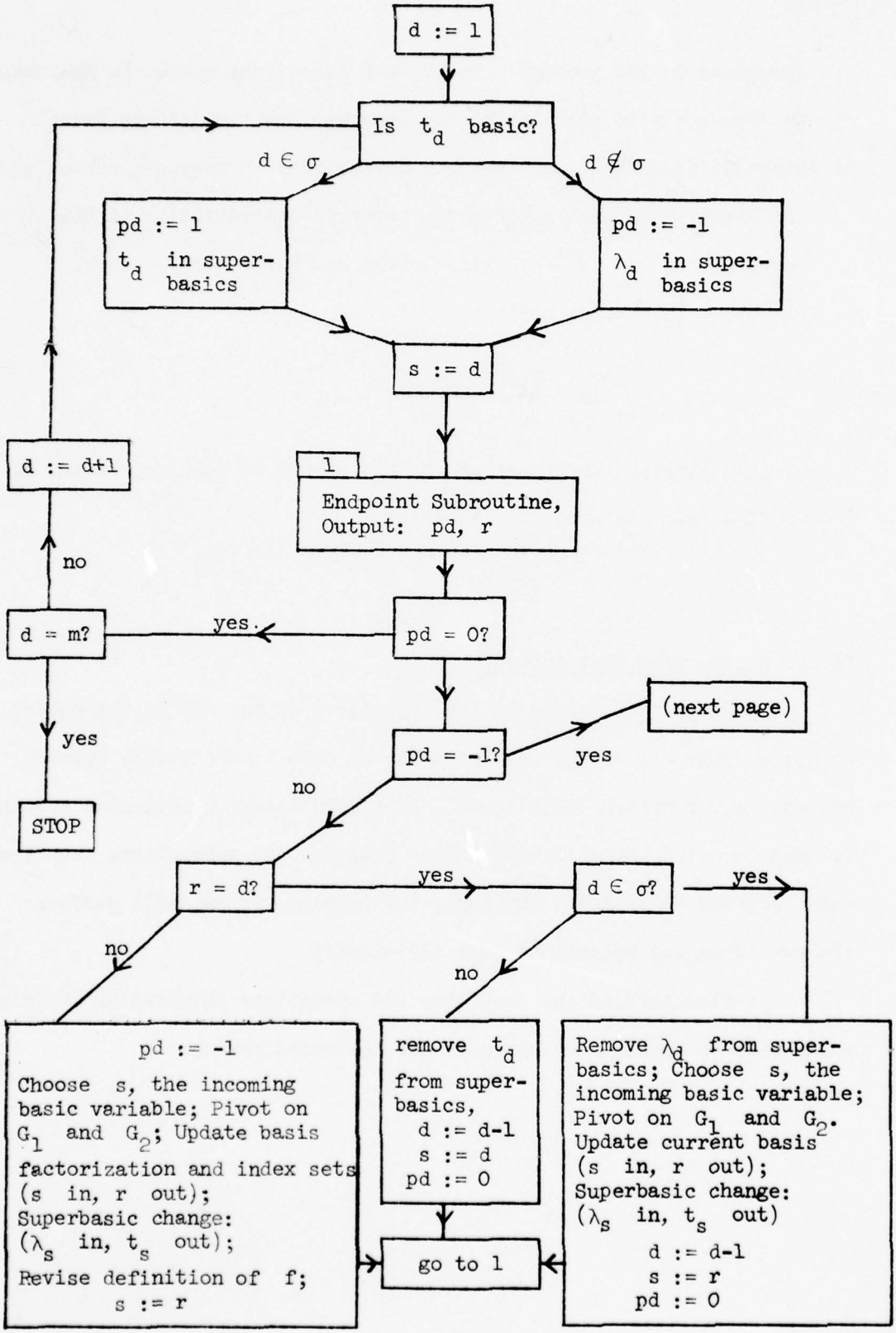
$$\lim_{\ell \to \infty} \frac{\|z^{\ell+1} - z^*\|}{\|z^{\ell} - z^*\|^2} < +\infty ,$$

or intuitively, the number of decimal places of accuracy eventually doubles from iteration to iteration.

III.5. Moving from Cell to Cell

An important factor in the efficiency of the BCA is the matter of how quickly all of the revisions can be made to the basis, updated columns, and function definitions. If a good linear programming routine is used to solve the auxiliary linear program, the subroutines from that code involved in updating the basis and solving systems will perform the cell changing operations very efficiently.

A flowchart of the decisions and operations involved in deciding which cell is adjacent is given on the following pages.


d := 1
Is t_d basic?
$d \in \sigma$
$d \notin \sigma$
pd := 1
t_d in super-basics
pd := -1
λ_d in super-basics
s := d
1
Endpoint Subroutine, Output: pd, r
d := d+1
no
d = m?
yes
pd = 0?
yes
(next page)
pd = -1?
yes
no
STOP
r = d?
yes
$d \in \sigma$?
yes
no
no
pd := -1
Choose s, the incoming basic variable; Pivot on G_1 and G_2; Update basis factorization and index sets (s in, r out);
Superbasic change: (λ_s in, t_s out);
Revise definition of f;
s := r
remove t_d from super-basics,
d := d-1
s := d
pd := 0
Remove λ_d from super-basics; Choose s, the incoming basic variable; Pivot on G_1 and G_2. Update current basis (s in, r out);
Superbasic change: (λ_s in, t_s out)
d := d-1
s := r
pd := 0
go to 1

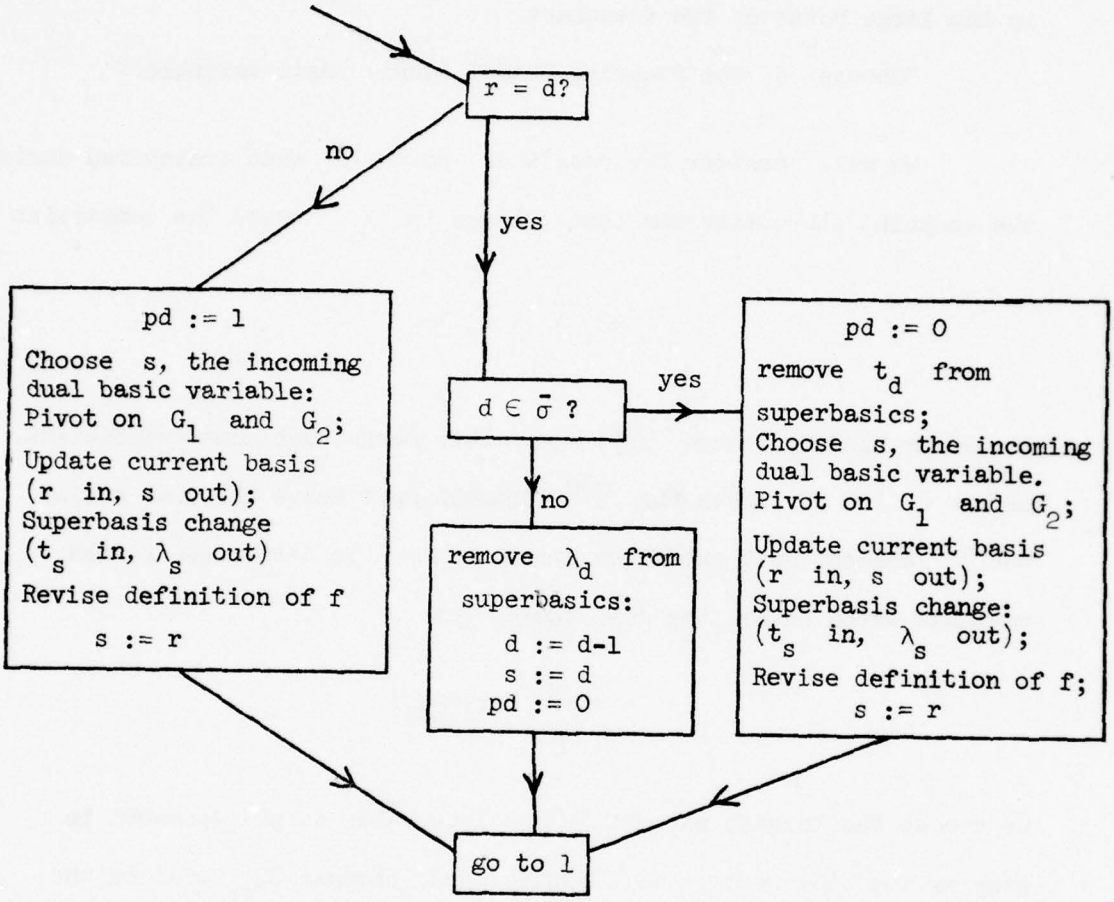

r = d?
no
yes
pd := 1
Choose s, the incoming dual basic variable:
Pivot on G_1 and G_2;
Update current basis (r in, s out);
Superbasis change (t_s in, λ_s out)
Revise definition of f
s := r
$d \in \bar{\sigma}$?
yes
no
pd := 0
remove t_d from superbasics;
Choose s, the incoming dual basic variable.
Pivot on G_1 and G_2;
Update current basis (r in, s out);
Superbasis change: (t_s in, λ_s out);
Revise definition of f;
s := r
remove λ_d from superbasics:
d := d-1
s := d
pd := 0
go to 1

FIGURE 5.1

Next we expand upon some of the operations described briefly in the large boxes of the flowchart.

"Choose s, the incoming primal (dual) basic variable."

We will consider the case when pd = -1. What transpired during the endpoint subroutine was that changes in λ_μ caused the constraint

$$G2_{\ell.}\lambda_\mu + \bar{c}_\ell \geq 0$$

to become binding, where $\bar{\gamma}(\bar{j}) = \ell$. This means that some coefficient, $G2_{\ell i} \neq 0$, $i \in \mu$. Since the $\bar{j}^{th}$ variable must leave the dual basis, the $\bar{j}^{th}$ element must enter the primal basis. To determine s, the variable which enters the dual basis, let

$$s = \arg \max_{i \in \mu} |G2_{\ell i}|$$

We choose the largest element in absolute value to aid somewhat in keeping the basis matrix well conditioned, because $G2_{\ell s}$ will be the pivot element in a straight forward update of the primal basis matrix. If p = 1 and $\ell = \gamma(\bar{j})$ we similarly find

$$s = \arg \max_{i \in \nu} |G1_{\ell i}|$$

and bring the column corresponding to t_s into the basis matrix and remove the column corresponding to the j^{th} primal variable.

"Pivot on G1 and G2"

Since the basis is to change, Gl and G2 must be updated. Obviously, one could update the basis and recalculate Gl and G2 as described in Section 1, but if there are several columns in Gl and G2, this would be rather expensive. Also the current right-hand sides $\bar{b}$ and $\bar{c}$ must be updated. We will describe the operations performed if $p = 1$ and $\ell = \gamma(r)$ and $s = \arg\max_{i \in \nu} |Gl_{\ell i}|$ has been chosen.

First we pivot to update Gl:

Calculate the eta vector defined by $Gl_{.s}$

$$\zeta_\ell := 1/Gl_{\ell s}$$

$$\zeta_i := -Gl_{is}/Gl_{\ell s}\ , \qquad i \neq \ell.$$

Update the columns, $Gl_{.j}$ and $\bar{b}$.

$$\begin{aligned} v &:= Gl_{\ell j} \\ Gl_{\ell j} &:= 0 \\ Gl_{.j} &:= Gl_{.j} + v\cdot\zeta\ , \qquad j \neq s. \end{aligned} \tag{6.1}$$

$$\begin{aligned} v &:= \bar{b}_\ell \\ \bar{b}_\ell &:= 0 \\ \bar{b} &:= \bar{b} + v\cdot\zeta\ . \end{aligned} \tag{6.2}$$

In the ℓ^{th} row we now have

$$\sum_{\substack{j \notin \nu \\ j \neq s}} Gl_{\ell,}\cdot t_j + 1\cdot t_s + \bar{b}_\ell = 0$$

But since $t_s \geq 0$ is now basic we want

$$t_s = \sum_{\substack{j \in \nu \\ j \neq s}} Gl_{\ell j} t_j + \bar{b}_\ell \geq 0$$

so we must change the signs of the ℓ^{th} row's coefficients:

$$Gl_{\ell j} := -Gl_{\ell j} , \qquad j \in \nu,\ j \neq s \tag{6.2}$$

$$\bar{b}_\ell := -\bar{b}_\ell .$$

Now the r^{th} dual variable is entering the dual basis, but since $r > d$, this is not a superbasic variable. So we must calculate the updated column corresponding to the r^{th} dual variable as described in (4.5) or (4.6) depending on whether the variables is a λ- or a ζ-variable. Call this updated column Y.

$n = \bar{\gamma}(s)$ is the row we are pivoting on because the λ_s is leaving the dual basis. Define the eta vector

$$\zeta_n = 1/Y_n ,$$

$$\zeta_i = -Y_i/Y_n , \qquad i \neq n ,$$

as usual and repeat the operations in (6.1), (6.2) and (6.3) with G2 replacing Gl, $\bar{c}$ replacing $\bar{b}$, r replacing s, and n replacing ℓ.

"Update current basis (r in, s out)"

Use the subroutines of the linear programming code to update the current basis factorization when column $\bar{j}$ enters the basis and column s leaves. Also revise the index sets σ, $\bar{\sigma}$, γ, $\bar{\gamma}$ to account for the change as follows:

Let k be a dummy variable and

$$k := \gamma(s) \qquad\qquad k := \bar{\gamma}(r)$$

$$\sigma(k) := r \qquad\qquad \bar{\sigma}(k) := s$$

$$\gamma(s) := 0 \qquad\qquad \bar{\gamma}(r) := 0$$

$$\gamma(r) := k \qquad\qquad \bar{\gamma}(s) := k\ .$$

"Superbasis change (λ_s in, t_s out)"

Update the index sets:

$$\mu := \mu \cup s\ ,$$

$$\nu := \nu \setminus s.$$

Add the column

$$\begin{pmatrix} s_1 \\ s_2 \end{pmatrix}$$

corresponding to λ_5 as calculated in (4.5) to the matrix G2, using the new basis factorization to calculate s_1. Remove the column of G1 corresponding to t_s.

"Revise the definition of $f(\lambda_\mu, t_\nu)$"

Actually we also revise the definition of $q(\lambda_\mu, t_\nu)$, the bilinear inequality, here too. This is just a recalculation of the bilinear equations of (1.9) using the updated matrices G1 and G2,

and the revised right-hand sides $\bar{b}$ and $\bar{c}$ to again express the basic variables in terms of the new superbasics.

The major work done in the change from cell to cell is the pivot on G1 and G2, the calculation of two new superbasic columns, and the update of the basis matrix. Clearly, the efficiency of this portion of the algorithm depends upon the efficiency of the particular subroutines of the LP code which is used.

It appears to be that an interesting area for algorithmic research and experimentation lies in studying the endpoint algorithm itself. Many variants of Newton's method could be used to solve the nonlinear equations involved. Perhaps sophisticated differential equation methods could be used to follow the path.

In Chapter V we shall report on the numerical results from the testing of a code which implements the algorithm described in this chapter.

CHAPTER IV

A HOMOTOPY METHOD FOR COMPUTING EQUILIBRIA

IV.1. A Convergence Theorem

The bilinear complementarity algorithm described in the last chapter is essentially a systematic procedure for balancing the budgets of the m consumers sequentially. One might guess that a rather large number of cells would be traversed before equilibrium is reached. For example, at least m cells must be traversed in which the curve $F^{-1}(0)$ hits a budget constraint. In this chapter, we investigate a path method which immediately relaxes $\lambda_i \cdot t_i = 0$, $i \in \underline{m}$, and adjusts the variables so as to solve for all of the budget surpluses at once. We use the homotopy retraction method from Section IV.1 (Part 1) to motivate the construction of the deformation which defines the path of interest.

We will again utilize the pure exchange model of Section II.2 as the generic example to ease an already cumbersome notational load. The more general economy of Section II.3 can be dealt with in an analogous manner. The assumptions which guarantee that equilibria exist and coincide with quasi-equilibria, II.2.3, will be in effect here. The utility values v_i will be chosen in accordance with (2.7). The same primal and dual auxiliary linear programs (II.2.1) and (2.2) will again serve as a vehicle for treating the equilibrium problem as an equation solving problem.

By Lemma II.2.4, solving the equilibrium problem is equivalent to solving for

$$\omega \equiv (\pi, \lambda, \zeta^i (i \in \underline{m}), \rho; s, t, z^i (i \in \underline{m}), p)$$

which satisfies

$$f(\omega) = 0 , \qquad \omega \in D , \tag{1.1}$$

where

$$f(\omega) = \begin{pmatrix} \pi(\omega) w^1 - \lambda_1(\omega)(t_1(\omega) + v_1) \\ \\ \pi(\omega) w^m - \lambda_m(\omega)(t_m(\omega) + v_m) \end{pmatrix} \tag{1.2}$$

$\pi(\omega)$ refers to the first m components of ω, etc., and

$$\begin{aligned} D \equiv \{\omega \mid \quad & \gamma^i z^i - t_i = v_i , && i \in \underline{m} \\ & \sum_{i=1}^{m} B^i z^i + pe + s = \sum_{i \in \underline{m}} w^i \\ & \pi B^i - \lambda_i \gamma^i - \zeta^i = 0 , && i \in \underline{m} \\ & \pi e - \rho = 1 \\ & \pi s = 0 \\ & \zeta^i z^i = 0 , && i \in \underline{m} \\ & \omega \geq 0 , && \} . \end{aligned} \tag{1.3}$$

Since f maps $(\pi, \lambda, t) \in \mathbb{R}^{3m}$ into $\mathbb{R}^m$, it does not appear that the path methods such as the homotopy retraction or strong path methods can be used to solve this problem. But, under some quite reasonable assumptions, it can be shown that D is a collection of m-dimensional polyhedral sets which form a subdivided m-complex. By a careful choice

of an initial point, an analogue of the homotopy retraction method will be used to define a path leading to an equilibrium point.

In the definition of D (1.3) the complementary relation $\lambda_i \cdot t_i = 0$, $i \in \underline{m}$, is not enforced. For this reason, none of these variables will be constrained to zero during the course of the algorithm. By V.2.7 we have that if λ_i or $t_i = 0$ then $f_i(\omega) > 0$, $i \in \underline{m}$. Thus, the retraction function which maps points in $\mathbb{R}^m_+$ into $\partial\mathbb{R}^m_+$ can be defined in terms of $\lambda(\omega) \in \mathbb{R}^m$ or $t(\omega) \in \mathbb{R}^m$. In either case, the boundary condition that $f(\omega)$ points into $\mathbb{R}^m_+$ for $\lambda(\omega) \in \partial\mathbb{R}^m_+$ (or $t(\omega) \in \partial\mathbb{R}^m_+$) will be satisfied. Since D is not a bounded convex set, we must make some assumptions to allow us to conclude that the path defined by the deformation is bounded. The assumptions required are more natural when the λ-variables are kept nonnegative by the intrinsic properties of the path, rather than the t-variables. The sign of the t-variables will be unrestricted, but if a solution of (1.1) can be found, then $f_i(\omega) = 0$ implies that $\lambda_i(\omega)$ and $t_i(\omega)$ are positive by 2.7 of Chapter II.

The preceding discussion motivates consideration of the set $K \supset D$, defined below. If we define

$$\omega^1 \equiv (z^i(\omega),\ i \in \underline{m},\ p(\omega),\ \pi(\omega)),$$

$$\omega^2 \equiv (\zeta^i(\omega),\ i \in \underline{m},\ \rho(\omega),\ s(\omega)),$$

and let

$$A = \left[\begin{array}{c|c} & \begin{matrix} -B^{1T} & \gamma^1 & & 0 \\ \cdot & & \cdot & & \\ \cdot & & & \cdot & \\ \cdot & & & & \cdot \\ -B^{mT} & & & & \gamma^m \\ -e^T & & 0 & & \end{matrix} \\ \hline \begin{matrix} B^1 \ \ldots\ldots\ B^m & e \\ -\gamma^1 & \\ \quad \cdot & \\ \quad\ \cdot & \\ \quad\quad \cdot & \\ \quad\quad\ -\gamma^m & \end{matrix} & \end{array}\right], \quad b = \left[\begin{array}{c} 0 \\ \\ \\ \\ \\ -1 \\ \hline \Sigma w^i \\ -v_i \\ \vdots \\ -v_m \end{array}\right]$$

then

$$K \equiv \left\{ \omega \middle| A \begin{pmatrix} \omega^1 \\ \lambda(\omega) \end{pmatrix} + I \begin{pmatrix} \omega^2 \\ t(\omega) \end{pmatrix} = b,\ \omega^1,\ \omega^2 \geq 0,\ \langle \omega^1, \omega^2 \rangle = 0 \right\} . \qquad (1.4)$$

A set $\Sigma = (\omega^1_\sigma, \omega^2_\sigma, \lambda_\sigma, t_\sigma)$ of n variables $(n = \Sigma s_i + \ell + m + 1)$ consisting of exactly one of each pair (ω^1_i, ω^2_i), $i \in \underline{n-m}$ or (λ_i, t_i), $i \in \underline{m}$, is a <u>feasible basis</u> if the columns of $[A|I]$ corresponding to Σ ($[A|I]_{.\Sigma}$) is a nonsingular matrix and the system

$$[A|I]_{.\Sigma} \begin{bmatrix} \omega^1_\sigma \\ \lambda_\sigma \\ \omega^2_\sigma \\ t_\sigma \end{bmatrix} + [A_{.\mu} | I_{.\nu}] \begin{bmatrix} \lambda_\mu \\ t_\nu \end{bmatrix} = b , \qquad (1.5)$$

$$\omega^1_\sigma,\ \omega^2_\sigma \geq 0$$

has a solution, where $\lambda_\mu \equiv \{\lambda_i | \lambda_i \notin \Sigma\}$ and $t_\nu \equiv \{t_i | t_i \notin \Sigma\}$.

A characteristic set of variables is a set $\Gamma = \Sigma \cup \mu \cup \nu$ for some feasible basis Σ. Note that a basis determines a characteristic set (c-set), but not vice versa. Since a c-set Γ always contains all of the λ- and t-variables, it can always be determined by $\gamma = \Gamma \setminus \{\lambda_i, t_i,\ i \in \underline{m}\}$. Let $\bar{\gamma}$ be those variables complementary to γ.

Corresponding to each c-set of variables Γ, we define the set

$$C_\Gamma = \left\{ \omega \middle| [A|I]_{.\gamma} \begin{bmatrix} \omega^1_\gamma \\ \omega^2_\gamma \end{bmatrix} + [A_{.\lambda}|I_{.t}] \begin{bmatrix} \lambda \\ t \end{bmatrix} = b_\gamma,\ \omega^1_\gamma,\ \omega^2_\gamma \geq 0,\ \omega^1_{\bar{\gamma}},\ \omega^2_{\bar{\gamma}} = 0 \right\}, \tag{1.6}$$

where

$$A_{.\lambda} \equiv \begin{bmatrix} \gamma^1 & & 0 \\ & \ddots & \\ 0 & & \gamma^n \\ \hline & \bigcirc & \end{bmatrix} \quad \text{and} \quad I_t \equiv \begin{bmatrix} & \bigcirc & \\ \hline 1 & & \bigcirc \\ & \ddots & \\ \bigcirc & & 1 \end{bmatrix}$$

are the columns corresponding to the λ- and t-variables. These sets will satisfy the definition of a cell (II.3, Part 1) if we assume that each cell corresponding to a characteristic set of variables satisfies the constraint qualification. This will imply that there is a point $\bar{\omega} \in C_p$ such that $\bar{\omega}^1_\gamma \gg 0$ and $\bar{\omega}^2_\gamma \gg 0$. Let $\mathcal{X} = \{C_\Gamma | \Gamma \text{ is a c-set}\}$. (We will occasionally refer to elements of $\mathcal{X}$ as characteristic cells or c-cells to emphasize their correspondence with some c-set of variables.)

Define $K = \bigcup_{C_\Gamma \in \mathcal{X}} C_\Gamma$.

An important step in applying the theory of Chapter II (Part 1) is to show that $(K, \mathcal{X})$ is a subdivided m-complex.

Lemma 1.1. $(K, \mathcal{X})$ is a subdivided m-complex.

Proof. We make use of the one-to-one correspondence between c-cells and c-sets. Property (c) if I.3.2 follows because there are only a finite number of characteristic sets.

The proof follows the lines of Lemma III.4.3 (Part 1) because, again, any cell $C \in \mathcal{X}$ can be associated with a set of variables $(\omega_\beta^1, \omega_\beta^2)$ not constrained to be zero. Define $\varphi(c) = \beta$ to be the set of variables associated with C and let $\varphi^{-1}(\beta) = C$ be the cell associated with $\bar{\beta}$. Property (a) now follows from Part a) of the proof of Lemma III.4.3. Property (b) follows from Case 4) of part b) of the proof of the same lemma. ∎

Below we will expand the subdivided m-complex $(K, \mathcal{X})$ by adding the parameter θ needed to define the deformation. Clearly if $\mathcal{L} \equiv \{\sigma | \sigma = c_\Gamma \times [0,1],\ c_\Gamma \in \mathcal{X}\}$ and $L \equiv \bigcup_{\sigma \in \mathcal{L}} \sigma$, then $(L, \mathcal{L})$ is a subdivided (m+1)-complex. Define $F: L \to \mathbb{R}^m$ as

$$F(\omega, \theta) = \theta f(\omega) - (1-\theta)(\lambda(\omega) - \lambda^0) \tag{1.8}$$

It will be assumed that $\lambda^0 = \lambda(\omega^0)$ for some $\omega^0 \in K$ which is easy to compute. For theoretical purposes it will be useful to choose ω^0 such that $\lambda(\omega^0) \equiv 0 \in \mathbb{R}^m$. This appears to be necessary in order to show that $F^{-1}(0)$ is bounded. The process of following $F^{-1}(0)$ to an equilibrium point will be called the homotopy retraction method.

Next we describe the assumptions necessary to prove that we can consider a compact subset of L.

Assumption 1.2. (Bounded utility). Even household's utility is bounded for a fixed level of exports. That is, for fixed p and any $i \in \underline{m}$, the linear program

$$\begin{aligned} &\text{maximize} && r^i z^i \\ &\text{subject to} && \sum_{i \in \underline{m}} B^i z^i + pe \leq \sum_{i \in \underline{m}} w_i' && z^i \geq 0, \end{aligned}$$

has a finite objective value. We also make the assumption that the utility is bounded below for fixed p.

The first part of the assumption is fairly standard in the literature (cf. Dantzig, Eaves, Gale [1976]). The second part seems as innocuous as the first. Even though there might be an unattractive activity $j \in S_i$ for some i $(r^i_j < 0)$ it seems reasonable to assume that household i can perform only a finite amount of that activity, even if the resources of the other households are made available to him.

Assumption 1.3. (Bounded production)

a) The amount of exports the system is capable of producing is bounded: the optimal value of

$$\text{maximize} \quad p$$
$$\text{subject to} \quad \sum_{i \in \underline{m}} B^i z^i + pe \leq \sum_{i \in \underline{m}} w^i, \; z^i \geq 0 \; .$$

is finite (call this optimal value p^+).

b) The solution to this linear program and its dual is unique.

Part b) will not necessarily be satisfied for an arbitrary problem, but by perturbing the objective vector and the sum of endowments this property can be satisfied.

Assumption 1.4. $\langle \pi(\omega), w^i \rangle > 0, \quad \forall i \in \underline{m}$ for $\omega \in K$.

This assumption is a weakening of Assumption IV.2.3 (Part 1), but it also implies the existence of equilibria and that equilibria and quasi-equilibria coincide (cf. Arrow and Hahn [1970]). In models with production, this assumption can be weakened to

$$\langle \pi(\omega), \; w^i + \sum_{j=1} \sigma^i_j \omega^j \rangle > 0 \qquad \forall \; \omega \in K.$$

This assumption implies that the value of each consumer's assets is positive for any realization of $\omega \in K$. This condition is very difficult to verify for any particular problem, in contrast to the assumption that $w^i \gg 0$, for any $i \in \underline{m}$. However, computational experience

indicates that Assumption 1.4 is often satisfied in practice.

Now we are in a position to prove that major result of this section, that the component of $F^{-1}(0)$ leading from $(\omega^0,0)$ terminates at a point $(\omega^*,1)$ which is an equilibrium point. First, we must show how to choose an initial point ω^0 such that $\lambda(\omega^0)$ 0 and $(\omega^0,0) \in 1$.

Let $(z_0^i,\ i \in \underline{m},\ p_0,\ s_0)$ solve the problem

$$\begin{aligned} &\text{maximize} && p \\ &\text{subject to} && \sum_{i\in\underline{m}} B^i Z^i + pe + s = \sum_{i\in\underline{m}} w^i \\ & && p,\ s \geq 0,\ z^i \geq 0,\ i \in \underline{m} \end{aligned} \tag{1.9}$$

and let $(\zeta_0^i,\ i \in \underline{m},\ \rho_0,\ \pi_0)$ solve the dual problem

$$\begin{aligned} &\text{minimize} && \pi\Big(\sum_{i\in\underline{m}} w^i\Big) \\ &\text{subject to} && \pi e - \rho = 1 \\ & && \pi B^i - \zeta^i = 0, \qquad i \in \underline{m} \\ & && \rho,\ \pi \geq 0,\ \zeta^i \geq 0, \qquad i \in \underline{m}. \end{aligned}$$

Then let $t_{i0} = \gamma^i z_0^i - v_i$, $i \in \underline{m}$ and $\lambda_0 = 0 \in \mathbb{R}^m$. It is easy to see that $(\omega^0,0) = (\pi_0,\ \lambda_0,\ \zeta_0^i\ (i \in \underline{m}),\ \rho_0;\ s_0,\ t_0,\ z_0^i\ (i \in \underline{m}),\ \rho_0;0)$ is the only point in L with $\lambda(\omega) = 0$ due to assumption 1.3(b).

Our deformation $F:L \to \mathbb{R}^m$ has the simple form

$$F(\omega,\theta) = \theta f(\omega) - (1-\theta)\,\lambda(\omega)\ , \qquad (\omega,\theta) \in L\ .$$

We will be assuming that 0 is a good value for F w.r.t. $(L, \mathcal{L})$.

Lemma 1.6. If P is the component of $F^{-1}(0)$ containing ω^0, and Assumptions 1.2 and 1.3 are satisfied, then P is a bounded set.

Proof. Suppose that $P(\alpha):[0,T] \to L$ is a continuous parametrization of P where $T \leq +\infty$. First we show that $\lambda(P(\alpha)) \gg 0$ for any $\alpha > 0$. Since we begin at the facet of L for which $\theta = 0$, it is clear that

$$\dot{\theta}(0) = \left.\frac{d\theta(\alpha)}{d\alpha}\right|_{\alpha=0} > 0\ .$$

Since $\lambda(0) = 0$, Assumption 1.4 gives us that $f(\omega^0) \gg 0$. Also, we can express f solely as a function of $\lambda(\omega)$, initially, because the t variables are basic at the start. Thus,

$$F'(\omega^0) = [\ \partial f'(\lambda) - (1-\theta)I \,|\, f(\lambda) + \lambda\ ]\ .$$

At $\alpha = 0$, given that $\dot{\theta}(\alpha) > 0$, it must be true that $\dot{\lambda}(0)$ satisfies

$$-I\dot{\lambda}(0) = -f(\lambda(0))$$

or

$$\left.\frac{d\lambda(\alpha)}{d\alpha}\right|_{\alpha=0} = f(\lambda(0)) \gg 0\ .$$

Hence, there is some $\epsilon > 0$ such that for any $\alpha \in (0,\epsilon]$, $\lambda(\alpha) \gg 0$. Suppose there is an $\bar{\alpha} \in (\epsilon,T]$ such that $\lambda_i(\bar{\alpha}) \leq 0$ for some $i \in \underline{m}$.

Then by the continuity of $\lambda_i(\alpha)$, there must be some $\bar{\alpha} \in (\epsilon, \bar{\alpha}]$ such that $\lambda_i(\alpha) = 0$. Then, since $P(\hat{\alpha}) \in F^{-1}(0)$,

$$\theta(\hat{\alpha}) \cdot f_i(\omega(\hat{\alpha})) = 0 \ .$$

This is impossible because (a) $\theta(\hat{\alpha}) = 0$ implies that $\lambda(\hat{\alpha}) = 0$ and, hence $P(0) = P(\hat{\alpha})$ which contradicts the fact that 0 is a good value for F, and (b) $\lambda_i(\hat{\alpha}) = 0$ implies $f_i(\omega(\hat{\alpha})) > 0$. Thus $\lambda(\alpha) \gg 0$ for any $\alpha > 0$.

The fact that $\lambda(\alpha) \gg 0$ for any $\alpha \in (0,T]$ immediately yields that $f(\omega) \geq 0$ for any $\omega \in P$ because

$$f_i(\omega) = \frac{1 - \theta(\omega)}{\theta(\omega)} \cdot \lambda_i(\omega) \geq 0 \ , \qquad i \in \underline{m} \ .$$

From equation (IV.2.3, Part 1), we have

$$p(\omega) = \sum_{i=1}^{m} f_i(\omega) \geq 0 \ , \qquad \omega \in P \ .$$

Thus, by Assumption 1.3(a), $0 \leq p(\omega) \leq p^+$ for any $\omega \in P$.

It is well known that the optimal value of a linear program is a continuous function of the right-hand sides for which the linear program has a feasible solution and finite optimal value. Consider the following continuous functions of p.

$$\Phi_i(p) = \max\{\gamma^i z^i \mid \sum_{i=1}^{m} B^i z^i \leq \sum_{i=1}^{m} w^i - pe,\ z^i \geq 0,\ i \in \underline{m}\}, \quad i \in \underline{m},$$

$$\varphi_i(p) = \min\{\gamma^i z^i \mid \sum_{i=1}^{m} B^i z^i \leq \sum_{i=1}^{m} w^i - pe,\ z^i \geq 0,\ i \in \underline{m}\}, \quad i \in \underline{m}.$$

By Assumption 1.2, $\Phi_i(p)$ and $\varphi_i(p)$ are finite for any $p \in [0,p^+]$, and, by continuity, $\Phi_i(p)$ attains its maximum, t_i^+, and $\varphi_i(p)$ attains its minimum, t_i^-, on $[0,p^+]$ for any $i \in \underline{m}$. From the definition of $D \supset P$ (1.3) we have for $\omega \in P$.

$$t_i^- - v_i \leq t_i(\omega) \leq t_i^+ - v_i\ , \qquad i \in \underline{m}\ .$$

Next we show that $\lambda(\omega)$ is bounded for $\omega \in P$.

$\pi(\omega) \in \{\pi \mid \pi \geq 0,\ \pi e = 1\}$, a compact set. Clearly $b_i(\pi) = \pi B^i$ is a continuous function of π. Thus b_i is bounded above by $k \cdot e^i$ for some $k > 0$ and $e^i \in \mathbb{R}^{s_i}$ is a vector of ones.

Since $\gamma^i z^i$ is an insatiable utility function, $\gamma^i_j > 0$ for some $j \in \underline{s_i}$. Pick $j = \arg\min_{k \in \underline{s_i}}(\gamma^i_k \mid \gamma^i_k > 0)$. Then the constraint $\lambda_i \gamma^i \leq \pi B^i \leq ke^i$ implies that $\lambda_i \leq k/\gamma^i \equiv \bar{\lambda}_i < \infty$ for any $i \in \underline{m}$. Since $\lambda_i \geq 0$, $\{\lambda(\omega) \mid \omega \in P\}$ is a bounded set. Since $t(\omega)$ and $\lambda(\omega)$ are bounded in their respective subspaces, we can use (1.5) to claim that all other variables are bounded. Given a feasible basis Σ, we have

$$\begin{bmatrix} \omega^1_\sigma \\ \lambda_\sigma \\ \omega^2_\sigma \\ t_\sigma \end{bmatrix} = [A|I]^{-1}_{\cdot\Sigma} \left[b - [A_{\cdot\mu} | I_{\cdot\nu}] \begin{pmatrix} \lambda_\mu \\ t_\nu \end{pmatrix} \right] .$$

Since ω_σ^1 and ω_σ^2 are continuous functions of $\lambda(\omega)$ and $t(\omega)$ for each of the finite number of bases Σ, we conclude that ω_i is bounded for any $i \in \underline{2n}$ when $\omega \in P$. Therefore P is bounded. ∎

Once again we redefine our subdivided $(m+1)$-complex. Choose a constant $Q > 0$ large enough that for any $\omega \in P$, $\omega_i \in (-Q,Q)$ for any $i \in \underline{2n}$. Define $\bar{C} \equiv C \cap \{(\omega,\theta) | \omega_i \in [-Q,Q],\ \theta \in [0,1]\}$ for any cell $C \in \mathcal{L}$. Let $\mathcal{M}$ be the collection of all such cells, and let $M = \bigcup_{\bar{C} \in \mathcal{M}} \bar{C}$. Then $(M, \mathcal{M})$ is clearly a subdivided $(m+1)$-complex which is compact. We can now state and prove the main result.

Theorem 1.7. If 0 is a good value of F with respect to $(M, \mathcal{M})$ and Assumptions 1.2, 1.3, and 1.4 are satisfied, then the component of $F^{-1}(0)$ containing $(\omega^0,0)$ leads to the boundary point $(\omega^*,1)$ and ω^* is an equilibrium point for the piecewise linear economy.

Proof. The facets of the cells of $\mathcal{M}$ which are contained in the boundary of M are those corresponding to $\omega_i = \pm Q$, $i \in \underline{2n}$ or the $\theta = 0$ and $\theta = 1$ boundaries. By Corollary II.3.19 (Part 1), the opposite boundary point to $(\omega^0,0)$ in P must be in the boundary of M. Here we are using the fact that 0 is a good value for the compact complex M. We have already shown that $\theta(\alpha) > 0$ for $\alpha > 0$ and the path P never hits a boundary corresponding to $|\omega_i| = Q$ for any $i \in \underline{2n}$. Thus, P must lead to a boundary point $(\omega^*,1)$. By the definition of F, $f(\omega^*) = 0$. Also the proof of Lemma 1.6 shows that $\lambda(\omega^*) \gg 0$, so by Assumption 1.4,

$t(\omega^*) \gg 0$. Finally, since $\omega^* \in K$, we appeal to Lemma IV.2.4 to conclude that ω^* is an equilibrium point. ∎

IV.2. The Homotopy Retraction Algorithm For Solving the Piecewise Linear Equilibrium Problem.

In these sections we describe how the theory of Section 1 can be implemented as an algorithm with some nice convergence properties. Much of the description here depends upon methods and notation that was introduced in the previous section of this chapter.

We are putting the problem into the form

find $(x,t;\lambda,\zeta) \geq 0$

such that

$$Dx + It = b$$

$$\lambda D - \zeta I = c$$

$$f(x,t;\lambda,\zeta) = (f_i)_{i\in\underline{m}} = (\langle C_{i,.}, \lambda\rangle - \lambda_i t_i)_{i\in\underline{m}} = 0$$

$$\lambda_i \cdot t_i = 0 , \qquad i \in \underline{k} \setminus \underline{m}$$

$$\langle x, \zeta\rangle = 0 .$$

Where again $D \in \mathbb{R}^{k\times\ell}$, and $C \in \mathbb{R}^{k\times m}$ contains the appropriate constants determining the income of the consumers as discussed in Lemma III.3.3 (Part 1).

Again, we reduce the problem to deal only with the superbasic variables λ_μ and t_ν. We will write the defining inequalities for a typical cell in terms of the matrices G1 and G2 and the updated right-hand sides calculated as in Section 4. Thus (4.7) of Chapter III can be written

$$C_\Gamma = \left\{ \lambda, t, x, \zeta \;\middle|\; \begin{pmatrix} x_\sigma \\ t_\sigma \end{pmatrix} = G1\ t_\nu + \bar{b} \geq 0, \right.$$

$$\left. x_{\bar{\sigma}}, t_{\bar{\sigma}}, \zeta_\sigma, \lambda_\sigma = 0, \ \begin{pmatrix} \lambda_{\bar{\sigma}} \\ \zeta_{\bar{\sigma}} \end{pmatrix} = G2\ \lambda_\mu + \bar{c} \geq 0 \right\}$$

The initial point $\omega^0 = (\lambda^0, t^0, x^0, \zeta^0)$ is determined by solving the auxiliary linear program II.3.1 rather than the procedure suggested in Chapter III which results in an initial point ω' such that $\lambda_i(\omega') = 0$, $i \in \underline{m}$. Solving the auxiliary linear program will result in a point which is much closer to the final solution than a solution for which $\lambda_i(\omega) = 0$, $i \in \underline{m}$. Thus, for reasons of algorithmic efficiency we choose an initial point which is not theoretically guaranteed to converge. However, in nearly all examples which have been run on the computer convergence was achieved.

The method still makes use of some of the desirable feasibility properties of the homotopy retraction method by calculating the boundary point which $f(\omega^0)$ points at from $\lambda_i(\omega^0)$, $i \in \underline{m}$. To do this we let

$$\alpha = \min_{i \in \underline{m}} \left(\lambda_i(\omega^0) / f_i(\omega^0) \right),$$

and $\lambda_i^0 = \lambda_i(\omega^0) - \alpha f_i(\omega))$, $i \in \underline{m}$. Then $\lambda^0 \in \partial \mathbb{R}_+^m$ is the "initial point" of the algorithm even though the algorithm never was at a point ω such that $\lambda(\omega) = \lambda^0$.

The deformation which, along with the subdivided m-complex (K, γ), defines the path the algorithm follows is given below.

$$F(\omega,\theta) = \theta f(\omega) - (1-\theta)(\lambda(\omega) - \lambda^0)$$

$$\omega \in K, \ \theta \in [0,1].$$

$$\theta^0 = \alpha/(1+\alpha)$$

Again, the variables in ω are expressed in terms of λ_μ, and t_ν as in (1.9).

$$F(\omega,\theta) = \left\{ \begin{array}{l} \theta(h_1 + D_1 \lambda_\mu - \text{diag}(\lambda_\mu)(F_1 \cdot t_\nu + e_1) - (1-\theta)(\lambda_\mu - \lambda^0_\mu) \ , \\ \theta(h_2 + D_2 \lambda_\mu - \text{diag}(t_\lambda)(F_2 \lambda_\mu + e_2) \\ \qquad - (1-\theta)(G2_{\mu\cdot} \lambda_\mu + \bar{c}_\mu - \lambda^0_\mu) \end{array} \right\} \quad (5.1)$$

It is important to choose the parameters v_i^*, $i \in \underline{m}$ be chosen large enough that $\lambda_i(\omega^0) > 0$. Often the choice of v_i suggested, in Section III.3 will result in positive dual multipliers, but if not, v_i must be increased (parametrically) until all λ_i are positive, $i = 1,2,\ldots, m$. If some $\lambda_i(\omega^0) = 0$, then $\alpha = 0$, $\theta^0 = 0$, and $\lambda^0 = \lambda(\omega^0)$. In the optimal solution to the auxiliary linear program, the slack variables on the first m rows, t_i, $i \in \underline{m}$ are never basic, so $t_\nu = (t_1, t_2, \ldots, t_m)$, and $\mu = \emptyset$. Thus, for the first cell, the function is

$$F(\omega,\theta) = \theta(h_2 - \text{diag}(t_\nu)(e_2)) - (1-\theta)\ (\bar{c}_\mu - \lambda^0_\ell)$$

and

$$F'(\omega,\theta)\Big|_{\theta=\theta^0} = [-\theta \ \text{diag}(e_2) \,|\, h_2 - \text{diag}(t_\nu)e_2] \ .$$

But if $\theta^0 = 0$, the matrix $F'(\omega,\theta)\Big|_{\theta=\theta^0}$ is clearly not of full rank, and hence the first step of the path following algorithm cannot be executed. Thus, the importance of choosing v_i so that $\lambda_i(\omega^0) > 0$, $i \in \underline{m}$ is clear.

In Section II.1 (Part 1) on the discussion of the homotopy retraction method the dimension of the problem was increased by one by the homotopy parameter θ. This was done primarily for theoretical reasons--it is easier to analyze the behavior of a path which ends at a boundary point of a complex, rather than a point at which the deformation function is undefined. However, one might suspect that it would be worthwhile to formulate the deformation in the lower dimension for computational purposes. For example, if $\lambda_1^0 = 0$, the deformation could be defined

$$h(\omega) = \lambda(\omega) - \frac{\lambda_1(\omega)}{f_1(\omega)} f(\omega)$$

and the path would be $h^{-1}(\lambda^0)$. However, $f_1(\omega) \to 0$ and $\lambda_1(\omega)/f_1(\omega) \to +\infty$ as the algorithm nears a solution and $h'(\omega)$ becomes unstable. A tail routine such as Newton's method can be used when the current point is near an equilibrium, but the decision as to when to institute this tail routine is by no means an easy one. Both approaches were experimented with and the homotopy approach was found to be much more stable.

IV.3. The Path-Following Algorithm

Next we present the path following algorithm which was implemented in the HRA code to be described in Chapter V. The algorithm is essentially identical to that presented in Section III.4; we include this description for completeness. Motivation and a verbal explanation of the algorithm are given in Section III.4.

Again we are given the data contained in σ, μ, ν, Gl, G2, $\bar{c}$, $\bar{b}$, $z_0 = (\lambda_\mu, t_\nu)$, θ_0, f as defined in (1.9), and (pd, s) describing the variable currently at zero which is allowed to increase. Define

$$b_s(z,\theta) = \begin{cases} Gl_{\sigma(s),.}\, t_\nu + \bar{b} & \text{if } pd = 1 \\ G2_{\bar{\sigma}(s),.}\, \lambda_\mu + \bar{c}_s & \text{if } pd = -1 . \end{cases}$$

Algorithm 3.1. The homotopy retraction algorithm (HRA).

0. $i = 0$, User supplies α_{max} and ϵ_1, ϵ_2, ϵ_3.

1. Calculate $F'(z_i, \theta_i) = [H|h] \in \mathbb{R}^{m \times m+1}$.

2. Solve $Hy = -h$ (if $\det H = 0$, rearrange columns of $[H|h] \to [H'|h']$ and repeat Step 2).
   Let $\bar{\omega} := \operatorname{sgn}(\det H)$
   if $i > 0$ go to Step 4.

3. $\delta := \operatorname{sgn}(\langle \nabla b_s(z_i, \theta_i), (y,1) \rangle)$,
   $\omega := \delta \cdot \bar{\omega}$, go so Step 5.

4. $\delta := \bar{\omega} \cdot \omega$

5. Let $v_1 = \delta \cdot \frac{(y,1)}{\|(y,1)\|}$

6. The tangential approximation to $F^{-1}(0)$

$T(\alpha) = a_1 \alpha + a_2$, $\alpha \in \mathbb{R}^1_+$,

where $a_1 := v$

and $a_2 := (z_i, \theta_i)$.

7. Solve the n linear equations in α defined by

$$\langle G1_{i,.}, T(\alpha) \rangle + \bar{b}_i = 0, \quad i \in \underline{k}$$

$$\langle G2_{i,.}, T(\alpha) \rangle + \bar{c}_i = 0, \quad i \in \underline{\ell} \qquad (6.1)$$

$$a_{1,m+1} \alpha + a_{2,m+1} = 1$$

Let α^* be the minimum positive real root of all these equations.

If a primal variable hits zero, pd := 1

If a dual variable hits zero, pd := -1

Let r := the index in (6.1) corresponding to the row for which $T(\alpha^*)$ is binding.

Let $\Delta := 2$.

If $a_{1,m+1} \alpha^* + a_{2,m+1} = 1$, let $\Delta := 3$.

Let $(z^0, \theta^0) = T(\alpha^*)$

8. If $\alpha^* < \alpha_{max}$ go to Step 9. Otherwise, let $\alpha^* := \alpha^*/2$, $(z^0, \theta^0) = T(\alpha^*)$, $b_r(z,\theta) \equiv \langle a_1, (z^0, \theta^0) - (z,0) \rangle$, $\Delta := 1$, and go to Step 10.

9. If pd and r describe the primal variable "p", let $\Delta := 3$.

If $\Delta = 3$ go to Step 13.

10. Define

$$g(z,\theta) \quad \begin{pmatrix} F(z,\theta) \\ b_r(z,\theta) \end{pmatrix}$$

and iterate

$$(z^{\ell+1},\theta^{\ell+1}) = (z^{\ell},\theta^{\ell}) - g'(z^{\ell},\theta^{\ell})^{-1}\, g(z^{\ell},\theta^{\ell}), \qquad \ell = 0,1,2,\ldots,$$

until $\|g(z^{\ell+1})\| < \epsilon_2$.

11. If $\Delta = 2$ go to Step 12.

Otherwise, let i := i+1,

$(z_i,\theta_i) := (z^{\ell+1},\theta^{\ell+1})$, and go to Step 1.

12. Let $(z^*,\theta^*) = (z^{\ell+1},\theta^{\ell+1})$

Check that $G1\cdot t_\nu^* + \bar{b} \geq -\epsilon_3$,

and $G2\cdot\lambda_\mu^* + \bar{b} \geq -\epsilon_3$,

and $\theta^* \in [0,1]$.

If so, return the values (z^*,θ^*, pd, and r) to the main program.

Otherwise, let pd and r define the most infeasible constraint,

solve

$$b_r(\mu\cdot(z^*,\theta^*) + (1-\mu)a_2) = 0$$

for $\mu \in \mathbb{R}^1$, let

$$(z^0,\theta^0) = \mu(z^*,\theta^*) + (1-\mu)a_2.,$$

and go to Step 10.

13. (Tail Routine). Perform the Newton iteration,

$$z^{\ell+1} = z^{\ell} - f'(z^{\ell})^{-1} f(z^{\ell}) , \qquad \ell = 0,1,2,\ldots$$

until $\|f(z^{\ell+1})\| < \epsilon_1$. Let $(z^*,\theta^*) := (z^{\ell+1},1)$ and go to Step 12.

Again, several remarks are in order:

a) The sizes of ϵ_i, $i \in \underline{3}$, and α_{max} may be chosen to suit a particular problem. Typical choices might be $\epsilon_1 = 10^{-10}$, $\epsilon_2 = 10^{-5}$, $\epsilon_3 = 10^{-4}$, and $\alpha_{max} = 5$. The choice of α_{max} will radically affect algorithm efficiency. If it is too small, many Newton iterations will be used to cross larger cells when it would probably work to take big jumps across in the facet which the curve seem to hit. If α_{max} is too large when $F^{-1}(0)$ takes a sharp curve, Step 10 may not converge.

b) (Step 9) "p" is the variable which the original auxiliary linear program was maximizing. Recall from the discussion in Section V.2 that $p = \sum_{i=1}^{m} u_i = \sum_{i=1}^{m} f_i(\omega)$ and if our path is "near" that of the theoretically desirable path described in Section IV.1, then $f_i(\omega) > 0$, $i \in \underline{m}$. Thus, if the path hits the "p = 0" facet we are close to an equilibrium and the tail routine can be implemented.

c) It is worth discussing how $F'(z,\theta)$ is computed in Step 10 assuming that a subroutine is available for computing $f'(\lambda_\mu, t_\nu)$. From the definition of $F(z,\theta)$ in (5.1) it is apparent that

$$F'(\lambda_\mu, t_\nu, \theta)$$

$$= \left[\vartheta \cdot f'(\lambda_\mu, t_\nu) - (1-\vartheta) \cdot \left[\frac{I}{G2_{\mu\cdot}} \middle| 0 \right] \middle| f(\lambda_\mu, t_\nu) + \left(\frac{\lambda_\mu}{G2_{\mu\cdot}\lambda_\mu + \bar{c}_\mu} \right) - \lambda^0 \right] \qquad (5.2)$$

The calculation is not much worse than calculating $f'(\lambda_\mu, t_\nu)$ if the values of $f(\lambda_\mu, t_\nu)$ and $G2_{\mu\cdot}\lambda_\mu + \bar{c}_\mu$ are saved from the calculation of $F(\lambda_\mu, t_\nu, \theta)$ as suggested by (5.1).

d) If we replace the first line in Step 8 by

"If $\alpha^* < \alpha_{max}$ go to 9, otherwise, let $\alpha^* := \alpha_{max}$,"

Then this algorithm is essentially the path following algorithm of Chapter I, which was proven to be convergent given ϵ_2, and α_{max} are small enough. This course of action would result in an inordinate amount of work for large cells--work which is not necessary in most cases.

IV.4. The Cell Switching Decisions and Operations

The decisions necessary in the "main program" of this algorithm for equilibrium calculation are very simple. Initially $t_\nu = 0$ and $\nu = \underline{m}$, so any one of the first m primal variables may be specified as binding--we choose $pd = 1$, $s = 1$. With the correct initial choice of v_i, $i \in \underline{m}$, all budget surpluses will be positive. An argument similar to that in the proof of Lemma 1.6 will show that all the variables t_i, $i \in \nu$ increase initially, and, hence, the initial tangent, a_1, points into the interior of the initial cell. Thus, no problems are encountered due to the fact that we are starting out at the vertex of the initial cell.

Flowchart for cell switching and basis updating.

(The same abbreviations will be used as in Section VI.4.)

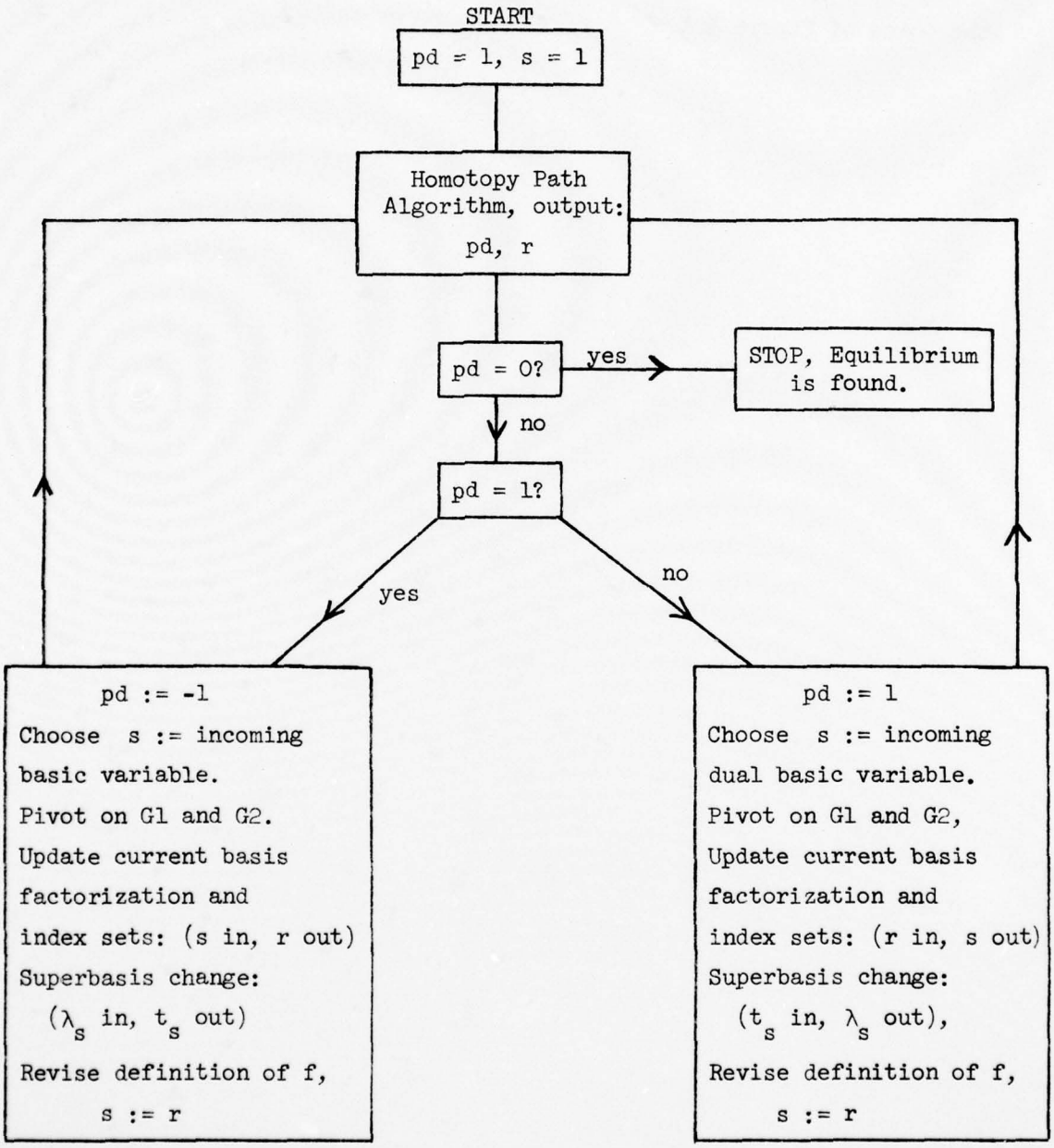


FIGURE 3.1

The reader is directed to the comments following Figure VI.5.1 for a discussion of the abbreviated description of the algorithm in the boxes of Figure 3.1.

CHAPTER V

RESULTS FROM COMPUTATIONAL EXPERIMENTS

In this chapter we report the results of numerical experiments with a code which implements the Bilinear Complementarity Algorithm (BCA) of Chapter III and Homotopy Retraction Algorithm (HRA) of Chapter IV. A large part of the work involved in this experimentation was in the construction of some test problems. We shall discuss how these problems were generated and what sort of preprocessing was necessary in order to solve the problems with these algorithms. The two codes are named BCA and HRA, respectively. The only differences in the codes are those absolutely necessary to implement the different path definitions of the two methods.

Both BCA and HRA use LPM1, an all-in-core FORTRAN linear programming code written at Stanford by J. A. Tomlin. LPM1 stores the problem matrix by columns packed in a vector of non-zeroes, a vector of the same dimension giving for each non-zero coefficient its row index, and a vector giving for each column the position of its first nonzero element in each of the two above vectors. The eta file is stored using the same principle. It uses an L-U factorization for inverting the basis followed by product form updates.

V.1. Computer Implementation of the Algorithms

The BCA and HRA codes are written in FORTRAN-IV and are compatible with both the WATFIV and FORTRAN-H compilers. The results to be reported here were done with the FORTRAN-H compiler with full optimization (OPT = 2). The testing was performed on an IBM 370-168 computer located at the Stanford Linear Accelerator Center. BCA consists of 3,837 lines of code while HRA is 3,632 lines long. LPM1 occupies 1,819 lines of each of these programs.

The form of the data input will not be described in detail here, but most of the data is input as the auxiliary linear program in MPS standard format. A small amount of additional data must be added to describe the vector $C(\cdot,I)$, for $I = 1,\ldots,m$, which contain the coefficients necessary to calculate the budget surplus of the I^{th} consumer. The details of input and output are contained in "A Programmer's Guide for BCA and HRA: Two programs for computing economic equilibria," to be published as a technical report.

V.2. Experimental Design

The primary objective of these numerical experiments is to demonstrate that a path-following philosophy along with an exploitation of the linear-programming structure can be implemented to solve some problems of moderate size (up to 6 consumers and 56 goods). It was hoped that the run times would be short enough for these problems that

it would be clear that these algorithms show promise for considerably larger problems. By larger problems we mean those concerned with about 10 traders and 250 goods. A model for which such a capability would be useful is a model of international trade where the traders are countries or groups of countries.

A secondary objective is to compare the performance of the BCA and HRA codes as they process the same set of test problems. The only conclusion we can make is that the BCA performs better on certain problems and worse on others, in comparison with the HRA algorithm.

One of the biggest difficulties in performing this experiment was in finding or manufacturing suitable test problems. Some small examples were given in Wilson [1976] which have been solved by hand. Two of these were solved by the BCA and HRA codes mainly to verify that the codes actually were computing equilibria. Three problems were adapted from the three equilibrium problems in Scarf [1973]. The last two problems appear for the first time here. The run times will follow the problem descriptions.

Problem Descriptions

Problem 1. This problem is due to Andreu Mas-Colell. There are three traders and two commodities. Each trader has one unit of each good and the utility functions are, respectively,

$$\min(x, 2y)$$

$$\min(2x, y)$$

$$\min(4x, 5y)$$

where x and y are the quantities of the two goods consumed.

One can easily verify that the demand function derived from these utility functions will have a ray as its range. Thus the activities of the traders can be reduced to one each and the utility functions are reduced to linear functions. The problem data is, now

$$r^i = (1), \qquad i \in \underline{3}$$

$$B^1 = \begin{pmatrix} .5 \\ 1 \end{pmatrix}, \qquad B^2 = \begin{pmatrix} 1 \\ .5 \end{pmatrix}, \qquad B^3 = \begin{pmatrix} .25 \\ .20 \end{pmatrix},$$

$$w^i = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad i \in \underline{3}.$$

$v^* = (1,1,4)$ is determined from the formula (V.2.6). We choose v_i to be slightly less than v_i^* as suggested in Section V.2:

$$v_1 = .9, \qquad v_2 = .95, \qquad v_3 = 3.92.$$

This example is used to show that an equilibrium problem of this form can have rational data and an irrational solution. The equilibrium prices are some scalar multiple of the vector

$$\pi = (1 + \sqrt{3}, 1).$$

The prices computed by both algorithms are correct to six decimal places,

$$\pi = (.732051, .267949).$$

AD-A053 169 STANFORD UNIV CALIF SYSTEMS OPTIMIZATION LAB F/G 5/3
THE COMPUTATION OF ECONOMIC EQUILIBRIA BY PATH METHODS.(U)
OCT 77 T R ELKEN N00014-75-C-0267
UNCLASSIFIED SOL-77-26 NL

2 OF 2
AD
A053169

END
DATE
FILMED
6-78
DDC

Problem 2. This example is due to Alan Whisman. It is reported in Wilson [1976]. It is concerned with four consumers and three goods. The utility functions are linear, but the B^i matrices are non-trivial. Again, there are no firms.

Problem data:

$$\gamma^1 = \left[\begin{array}{cc} 1 & 1 \end{array}\right] \qquad B^1 = \left[\begin{array}{cc} 1 & 2 \\ 2 & 2 \\ 3 & 2 \end{array}\right] \qquad w^1 = \left[\begin{array}{c} 2 \\ 2 \\ 1 \end{array}\right],$$

$$\gamma^2 = \left[\begin{array}{cc} 1 & 1 \end{array}\right] \qquad B^2 = \left[\begin{array}{cc} 4 & 3 \\ 3 & 2 \\ 1 & 2 \end{array}\right] \qquad w^2 = \left[\begin{array}{c} 1 \\ 3 \\ 3 \end{array}\right],$$

$$\gamma^3 = \left[\begin{array}{cccc} 1 & 1 & 1 & 1 \end{array}\right] \qquad B^3 = \left[\begin{array}{cccc} 0 & 1 & 0 & 3 \\ 1 & 2 & 3 & 0 \\ 0 & 0 & 4 & 3 \end{array}\right] \qquad w^3 = \left[\begin{array}{c} 1 \\ 5 \\ 4 \end{array}\right],$$

$$\gamma^4 = \left[\begin{array}{cc} 1 & 1 \end{array}\right] \qquad B^4 = \left[\begin{array}{cc} 1 & 2 \\ 1 & 0 \\ 0 & 0 \end{array}\right] \qquad w^4 = \left[\begin{array}{c} 1 \\ 1 \\ 1 \end{array}\right].$$

Problem 3. This is the first of three examples taken from Scarf's The Computation of Economic Equilibria [1973]. This is a pure exchange model, while the next two include production. This example involves 10 commodities and 5 traders. Each trader has 10 activities, the consumption of each of the 10 commodities, so the activity matrix $B^i = I \in \mathbb{R}^{10 \times 10}$ for $i = 1, \ldots, 5$. Each consumer has an initial stock $w^i \in \mathbb{R}^{10}$ of commodities prior to trade and a utility function.

$$u_i(y) = \text{sign}(b_i - 1) \cdot \sum_{j=1}^{10} a_{ij}^{1/b_i} y_{ij}^{1-1/b_i} \quad , \quad i \in \underline{5} \ ,$$

for final consumption. This is a special case of the class of constant elasticity of substitution or CES utility functions. For convenience we will refer to this functional form as a CES utility function.

Scarf's algorithm uses the demand functions derived by maximizing $u_i(y)$, subject to $\Sigma_1^{10} \pi_j y_j \leq \Sigma_1^{10} \pi_j w_{ij}$ to derive a mapping whose fixed point corresponds to an equilibrium. To implement either of our algorithms we must calculate a piecewise linear approximation to each consumer's utility function.

From an econometric point of view, it is probably just as easy and accurate to assess a utility function in a piecewise linear form as in some standard nonlinear form. However, if one is trying to approximate a nonlinear functional on $\mathbb{R}^{10}$ by linear pieces, it may take a very large number of pieces. For this reason, we are not trying to duplicate the answers which Scarf computed, although in some cases we come close. As a rule of thumb, we compute twice as many pieces of linearity as the dimension of the domain of the utility function.

In this example, we compute 20 pieces of linearity for each of the five consumers. The details of the calculations involved in this preprocessing are described in the Appendix. The prices obtained in the equilibrium solution were not very close to those computed by Scarf, but that is to be expected.

The problem data is given below.

Initial Stock of Commodities

Consumer										
1	.6	.2	.2	20	.1	2	9	5	5	15
2	.2	11	12	13	14	15	16	5	5	9
3	.4	9	8	7	6	5	4	5	7	12
4	1	5	5	5	5	5	5	8	3	17
5	8	1	22	10	.3	.9	5.1	.1	6.2	11

Utility parameters a_{ij}

Consumer										
1	1	1	3	.1	.1	1.2	2	1	1	.7
2	1	1	1	1	1	1	1	1	1	1
3	9.7	.1	5	.2	6	.2	8	1	1	.2
4	1	2	3	4	5	6	7	8	9	10
5	1	13	11	9	4	.9	8	1	2	10

Utility parameters b

Consumer	
1	2.0
2	1.3
3	3.0
4	0.2
5	0.6

Problem 4. This is the Walrasian equilibrium model on pp. 109-113 of Scarf [1973]. The problem involves 5 consumers, 6 commodities and a production sector. One can think of the production sector as being a single firm. The consumers share ownership of the firm in the sense that they own the factors of production--capital, unskilled labor, and skilled labor. The six commodities may be described as follows:

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

The activity matrix for the firm is

$$E^1 = \begin{matrix} 4 & 4 & 1.6 & 1.6 & 1.6 & .9 & 7 & 8 \\ -5.3 & -5 & -2 & -2 & -2 & -1 & -4 & -5 \\ -2 & -1 & -2 & -4 & -1 & 0 & -3 & -2 \\ -1 & -6 & -3 & -1 & -8 & 0 & -1 & -8 \\ 0 & 0 & 6 & 8 & 7 & 0 & 0 & 0 \\ 4 & 3.5 & 0 & 0 & 0 & 0 & 0 & 0 \end{matrix}$$

The consumers have initial endowments of commodities 2, 3, 4 and 6 given in the following matrix:

	Commodity			
Consumer	2	3	4	6
1	3	5	.1	1
2	.1	.1	7	2
3	2	6	.1	1.5
4	1	.1	8	1
5	6	.1	.5	.2

The consumer's utility functions are of the same form as in the previous problem. The specific values of a_{ij} are given in the following matrix:

	Commodity					
Consumer	1	2	3	4	5	6
1	4	0	.2	0	2	3.2
2	0.4	0	0	.6	4	1
3	2	0	.5	0	2	1.5
4	5	0	0	.2	5	4.5
5	3	0	0	.2	4	2

The activity matrix B^i is composed of zeroes except that $B^i_{jj} = 1$ if $a_{ij} \neq 0$. Of course zero columns can be omitted. The elasticities of substitution are given by

$$b_i = (1.2, 1.6, 0.8, 0.5, 0.6)$$

Problem 5. This problem, taken from pp. 113-119 of Scarf [1973] deals with four consumers and fourteen commodities:

1. Basic agricultural goods.
2. Processed food.
3. Textiles.
4. Housing services and heating.
5. Entertainment.
6. Housing, end of period.
7. Other capital, end of period.
8. Steel.
9. Coal.
10. Lumber
11. Housing, beginning of period.
12. Other capital, beginning of period.
13. Labor
14. Foreign exchange.

The consumers have initially a nonzero holding only of commodities 11-13. The following matrix describes this ownership pattern:

	Commodity		
Consumer	11	12	13
1	20	30	6
2	4	20	8
3	0	0	10
4	8	75	6

In this case the productive sectors involves import and export activities. The activity analysis matrix is given in Scarf.

The utility functions are of the form

$$u_j(y) = \prod_{i=1}^{7} y_{ij}^{a_i},$$

where

$$\sum_{i=1}^{7} a_{ij} = 1, \qquad j = 1, \ldots, 4 .$$

These Cobb-Douglas utility functions are homogeneous of degree one, i.e., for any $\lambda \geq 0$, $\lambda u_j(y) = u_j(\lambda y)$. This means that our piecewise linear approximations are tangent to the graph of u_j along a ray rather than at a single point. As one might expect, this means that our piecewise linear approximation will be better for this form of utility function than for the constant elasticity of substitution utilities.

The coefficients a_{ij} are given by

	Commodity						
Consumer	1	2	3	4	5	6	7
1	.1	.2	.1	.1	.1	.3	.1
2	.2	.2	.1	.1	.1	.1	.2
3	.3	.2	.3	.1	.1	0.0	0.0
4	.1	.2	.1	.1	.1	.1	.3

To make some sort of comparison between the Cobb-Douglas and CES (1) utility functions, we also solved this equilibrium problem using the former type of utility function. We used intensities a_{ij} proportional to those in the table above and let $b_i = 1.2$, $i = 1,2,3,4$. Below is a comparison of the equilibrium prices derived from this model using 1) Scarf's results, 2) our results using PL approximation to the Cobb-Douglas utility function, and 3) our results from a PL approximation to a CES utility function. (We use abbreviations for the names of the commodities.)

Equilibrium Prices

		SCARF	COBB-DOUGLAS	CES
1	AGRI	.0621	.0613	.0495
2	FOOD	.0583	.0595	.0500
3	TXTL	.0984	.0977	.0830
4	HSVH	.0714	.0706	.0479
5	ENTR	.0658	.0650	.0494
6	HEND	.0624	.0614	.0674
7	CEND	.0689	.0722	.0680
8	STEL	.0981	.0999	.0831
9	COAL	.0902	.0888	.0721
10	LUMB	.0795	.0773	.0654
11	HBEG	.0562	.0552	.1858
12	CBEG	.0620	.0650	.0613
13	LABO	.0365	.0300	.0328
14	FEXC	.0928	.0956	.0840

We note here that the HRA path passed through only 3 cells to reach the equilibrium prices in column 2 while it passed through 37 cells to reach the equilibrium prices in column 3. The computational results reported later for this example are concerned with the approximation to the Cobb-Douglas utility function.

The next two examples are new.

Problem 6. This example has 6 households and 4 goods. Five of the consumers have piecewise linear utility functions, the sixth has a linear utility function. Some of the initial endowments of the consumers are zero. In this case the entire auxiliary linear program matrix will be given to show how the piecewise linear utility functions are handled.

One notes that this matrix (Figure 2.1) could easily be put into a block diagonal form with coupling constraints, but since our LP code does not take advantage of such structure we leave it in the form such that the v_i, $i = 1,\ldots, m$ are in the first m rows. This allows us to consider the slack variables t_i, $i \in \underline{m}$ as the first m primal variables which eases the indexing problems in the code.

The initial endowments are given below.

	Commodity			
Consumer	1	2	3	4
1	2	2	1	2
2	1	4	3	2
3	0	0	3	4
4	1	1	3	0.5
5	1	2	0	2
6	2	3	2	1

D						x_n	b
-1							-.38 v_1
	-1						-4.06 v_2
		-1					-.66 v_3
			-1				-.93 v_4
				-1			-1.91 v_5
					-1		-.61 v_6
-4/3 -2/3 1							
-1/2 -3/2 1							
	-2 -1/2 -2/3 1						
	-3/4 -3/4 -3/4 1						
		-3/2 -.15 1					
		-1/2 -3/2 1					
		-1/2 -2/3 1					
			-1 -1 1				
				-2 -1/2 1			
				-1 -1 1			
					-2 -1/2 1		
					-3/4 -3/4 1		
-1 2	-1 1	4 -1	1 2	2	1	1	7
2 2	-.5 2	3 -2	1	.5	-.5 1	1	12
3 2	1.5	1 3	1	.7 -.2	3 2	1	12 $\Sigma_{i=1}^{4} w^i$
1	1 4	1	..5	1		1	11.5

The l.p. is max x_n s.t. $Dx \leq b$, $x \geq 0$

FIGURE 2.1

We will call this problem 6a. To create problem 6b we just changed three entries in the matrix of Figure 2.1.

Problem 7. This is a dynamic model in which three consumers attempt to maximize their discounted utility subject to a budget constraint while the producers maximize profit. The model is an adaptation of the Mananaguay model of W. P. Drews [1976]. In its original form it was a dynamic linear programming model of the economy of a developing country. The country was mythical so the data was not empirically supported, but the numbers were reasonable on the basis of past experience. This model was over five two-year periods and included sophisticated techniques to deal with distortions due to the finite planning horizon. The model offered a rich choice of objective functions, including one which maximized a weighted sum of consumption and one which minimized the dependence upon imports.

We substantially alter the model 1) to reduce the size of the problem and 2) to make the expansion in the consumption sector which an equilibrium formulation allows. We reduced the size of the model by dropping the number of periods to three, eliminating import and export activities and simplifying the end-correction mechanism. Arbitrary levels of capital stocks are defined which must be left to the society at the end of the third period. This is an admittedly poor method for handling finite horizon problems, but its simplicity suits our purpose.

We will not present the data of the problem here, just a few statistics. There are 8 perishable goods in each period (including three kinds of labor) and 8 capital goods in each of the three periods plus a fourth set of 8 capital goods which is bequeathed at the end of the planning horizon. Since the same good in two different time periods is considered to be two different goods, the model deals with 56 goods. In each period there are 8 production activities, 9 capital goods construction activities, and 18 consumption activities, 6 for each consumer. Hence, there is a total of 105 activities. The utilities are discounted at a rate of 15% between the two year periods for an annual rate of about 7.5%.

Preprocessing

Before we present the numerical results it would be useful to discuss how the initial utility levels v_i, $i \in \underline{m}$ were determined. For the pure exchange problems 1, 2, 3, and 6, we used the procedure suggested in Section V.2, i.e. solve the linear programs

$$v^* = \max\{\gamma^i z^i \mid A^i z^i \leq a^i,\ B^i z^i \leq w^i,\ z^i \geq 0\}, \qquad i \in \underline{m},$$

and let $v_i = v_i^*$, $i \in \underline{m}$. In general, the solution of these m linear programs took about as much computer time as the solution of the resulting auxiliary linear program. The initial utility levels v_i derived in this manner were very satisfactory for these models. For the models with production, many times the consumers are so dependent

upon the other consumers and the productive sector that it does not make sense to determine v_i^* as suggested in Section V.3. In these cases we either guessed at the initial utility levels or followed the procedure described below:

1. Guess at the equilibrium prices $\bar{\pi}$ of the consumers initially held commodities w^i.
2. Calculate their wealth $W^i = \bar{\pi} w^i$.
3. Calculate $v_i^* = \max u_i(z^i)$
   subject to $\bar{\pi} z^i \leq W^i$,
   using the nonlinear utility function u_i.

This value v_i^* will usually be less than the equilibrium utility level for consumer i because the piecewise linear utility overestimates the actual utility function (see Appendix A). If a poor choice of initial utility level causes the algorithm to fail for any reason, a better choice can usually be made by examining the output from the failure.

Several times the HRA algorithm failed to converge because of poor initial utility levels. The BCA failed only if the choice of some v_i was so large that $f_i(\omega) = \langle C_{\cdot,i}, \lambda \rangle - \lambda_i(v_i + t_i)$ is negative after the solution of the auxiliary linear program. We will report one example in which the HRA code failed but the BCA code worked.

V.3. Numerical Results

On the succeeding pages, tables will give the numerical results. First, though, we will describe the various measures used to evaluate how much work is being done to find an equilibrium solution.

1. L.P. iterations and L.P. time: The solution of the auxiliary linear program is definitely part of the HRA and BCA so the amount of work done to solve it must be measured. If the utility levels v_i are very good the largest percentage of work will be done in the LP portion of the algorithm. The LP time is measured using the subroutine LEFT1A supplied by the numerical analysis package at SLAC. The accuracy of this timer seems to be about $\pm$.02 sec.

2. Number of cells traversed and path-following time: The number of cells traversed is important because it counts the number of times the basis and the superbasic columns are updated. This work really outweighs the work done to follow the path through the cell. The path-following time measures the time spent after the auxiliary LP is computed and before the output of the final solution.

3. Scalar function calls: This refers to the number of times any of the consumers budget surplus functions are evaluated. In the BCA the work involved is on the order of the dot product of two vectors in $\mathbb{R}^d$ (see (III.4.9)), where d is the number of budget surpluses currently "at zero." In the HRA there are always m superbasic variables so, on the average, a little more work is done.

4. Jacobian evaluations: To compensate for the changing dimension of the BCA jacobians, here we counted the number of partial derivatives calculated and divide by m^2. In the HRA we just count the number of times a jacobian is calculated. Since the formulas (III.4.4) make it relatively easy to compute partial derivatives exactly, that is what is done. Each time a jacobian is calculated a linear system is solved in either code, so this figure represents more work than the evaluation of a jacobian.

5. CPU time: This includes input time, which is not negligible, and the output time of the equilibrium solution which is negligible. Thus the LP time and the path-following time will not add up to the CPU time.

Problem Statistics

Problem	Consumers, Goods	LP size	Density	Iterations	LP time
1	3,2	6 x 4	50%	1	.005 sec
2	4,3	8 x 11	40.9%	5	.02 sec
3	5,10	116 x 56	13.9%	95	1.75 sec
4	5,6	52 x 36	15.1%	33	.54 sec
5	4,14	75 x 57	14.6%	82	.85 sec
6(a)	6,4	23 x 20	18.5%	12	.07 sec
(b)				16	.08 sec
7(a)	3,56	97 x 107	11.8%	96	1.63 sec
(b)				90	1.63 sec

TABLE 3.1

Comparison of Algorithms

BCA (Bilinear Complementarity Algorithm)

HRA (Homotopy Retraction Algorithm)

Problem	Number of Cells Traversed	Path Following Time	Scalar Function Calls	Jacobian Evaluations	CPU Time
1 BCA	4	.13 sec	37	4	.18 sec
HRA	2	.12	29	7	.17 sec
2 BCA	8	.15	45	2	.21
HRA	1	.10	16	2	.19
3 BCA	44	1.61	164	7	4.49
HRA	20	.92	319	55	3.90
4 BCA	21	.31	82	3	.91
HRA	19	.51	387	66	1.17
5 BCA	46	1.23	166	14	2.69
HRA	3	.09	44	8	1.44
6a BCA	14	.18	194	11	.45
HRA	3	.08	90	12	.35
6b BCA	14	.21	247	14	.45
HRA	5	.11	130	18	.38
7a BCA	41	1.83	126	13	4.54
HRA	69	3.47	629	173	6.19
7b BCA	35	1.45	70	4	4.18
HRA	60	FAILED TO COMPUTE A SOLUTION[1]			

TABLE 3.2

[1] 7b. uses the same data as 7a. A different starting point caused the HRA to diverge.

Comparison of HRA on Problem 5 with Two Different Utility Functions

Utility Function	Number of Cells Traversed	Path Following Time	Scalar Function Calls	Jacobian Evaluations	CPU Time
Cobb-Douglas	3	.09 sec	44	8	1.44 sec
CES	37	1.41 sec	559	118	2.55 sec

TABLE 3.3

The solutions calculated by the codes are equilibrium points if they satisfy the systems of equations (II.2.3) and (II.2.3) and the non-negativity of the variables. Of course, no finite algorithm can calculate exact equilibria, in general. During the course of the algorithm the linear inequalities are maintained to a tolerance of 10^{-4}, i.e., in the linear approximation Algorithm III.5.2, $\epsilon_2 = 10^{-4}$. This is the default tolerance setting in LPM1 for the non-negativity of the relative cost coefficients. The final application of Newton's method for calculating a zero of the budget surplus function $f(\omega)$ terminates when $\|f(\omega)\| < 10^{-10}$ ($\epsilon_1 = 10^{-10}$ in Algorithm III.5.2). These tolerances are among the user-supplied parameters. The choice of ϵ_1 is rather arbitrary; since Newton's routine is quadratically convergent, a smaller

tolerance will not appreciably affect run time. However, since the accuracy of the variables and function parameters are only on the order of 10^{-14}, it would not be appropriate to set ϵ_1 lower than that. To set ϵ_2 smaller than 10^{-4} may appreciably increase the run time of the simplex method for solving the auxiliary linear program. Another tolerance is of importance in these algorithms. The termination criterion for each Newton routine for returning to the curve is $\epsilon_3 = 10^{-5}$. Making this parameter larger may reduce the run time marginally, but one runs a risk of moving outside of the radius of convergence surrounding the curve with the next step along the tangent.

V.4. Conclusions

The conclusions that can be drawn from such a limited testing must be tentative at best. However, it has been shown that some medium-sized problems can be solved quite quickly using path methods. In our opinion this experience shows that these methods show promise for the solution of problems with up to 10 consumers and 200 goods. This is only speculation, though, at this point.

Referring to Table 3.2, we see that the HRA was faster 6 times and the BCA was faster on 3 problems. This indicates that there can be no conclusion drawn concerning which method is better. The HRA appears to be more erratic. Compare the results from problem 5 and 7 for example. The increase in CPU time for the BCA is more in proportion to the change in problem size than the increase for the HRA.

It may be that the HRA is superior for problems with Cobb-Douglas utility functions. We cite the evidence from Problem 5 in Table 3.2 and Table 3.3. However, more testing must be done with problems having Cobb-Douglas utility functions before this statement can be made with any confidence.

An interesting fact not reported in the tables above is that for the BCA, the number of bilinear equations which are binding increases throughout the course of the computation on such problem. The theory does not guarantee that this type of monotone improvement occurs, but it is fortunate that this behavior seems to be common. One way of measuring the progress of the HRA is to note how θ increases from $\theta^0 \in (0,1)$ to $\theta^* = 1$ at an equilibrium. In several examples θ increased and then decreased for several cells. In problem 7b, $\theta^0 = .256$ increased to .697 before decreasing to 0, causing the program to stop execution.

The HRA passed through fewer cells than the BCA in every problem but Number 7; this advantage must be large in order for the HRA to take less path following time because the BCA is concerned with cells of smaller dimension than m most of the time. This fact explains the relatively small number of scalar function calls and jacobian evaluations for the BCA in comparison with the number of cells traversed.

This comparison of algorithms is not extensive enough to allow us to make authoritative conclusions. It does allow us to say that these algorithms show promise for the goal of solving economically significant equilibrium models.

CHAPTER VI

SUGGESTIONS FOR FUTURE RESEARCH

We close this work with some ideas for continuing this investigation. Some of these hypotheses are more speculative than others but they all deserve to be mentioned.

1. Chapter IV gives a convergence proof for the Homotopy Retraction Algorithm (HRA) when the starting point λ^0 is the zero vector. For computational efficiency, we solve an auxiliary linear program which results in a starting point which is closer to equilibrium (Section IV.2), but, as problem 7b indicates, convergence is no longer guaranteed. Another difficulty is in choosing the initial utility levels v_i, $i \in \underline{m}$ in such a way that a) all dual multipliers λ_i, $i \in \underline{m}$, are positive, and b) all budget surpluses $f_i(\omega)$, $i \in \underline{m}$ are positive.

Although the author has tried unsuccessfully to merge other homotopy deformations with the piecewise linear economy to define a convergent path method, there still may be promise in such an enterprise. It is certainly possible that some other deformation could define a path which would have better computational properties than the HRA. One simple alternative would be to involve the t variables (slacks corresponding to the utility levels) rather than the λ-variables in the deformation $F(\omega, \theta)$, i.e.,

$$F(\omega, \theta) = \theta f(\omega) - (1-\theta)(t(\omega) - t^0) .$$

Other possible path methods which could be adapted to this problem are the strong path method (Section IV.3, Part 1) or any of the class of path methods discussed in Section IV.7 (Part 1).

One possible remedy for the initial point problems of the HRA would be to solve the auxiliary linear program (II.2.3) with rather conservative initial utility levels, and then parametrically increase those v_i for which $\lambda_i = 0$. One would continue this increase until λ_i was positive and stop before $f_i(\omega)$ went to zero ($\lambda_i = 0 \Rightarrow f_i > 0$). When these conditions were satisfied for $i = 1, 2, \ldots, m$, the HRA would be instituted.

2. We discussed earlier why it was reasonable to consider models with few consumers and piecewise linear utility functions, but interesting extensions to our theory are possible which would deal with the problems of many consumers and nonlinear utility functions.

In competitive economies the various agents make decisions independently, using only a little common information (prices). Perhaps some sort of decomposition principle could be used to account for the effects that the decisions made by a large number of households have on the system. Another method to ease the computational burden would be to take advantage of the block-diagonal structure of the linear program by using a GGUB algorithm (see Winkler [1972] for a survey and unification of these methods).

If one is trying to approximate a nonlinear utility function in many variables, better methods may be available than using a globally defined piecewise linear function. If one could deal with only the hyperplanes which are binding or near binding, the dimension of the problem could be reduced considerably. The rows corresponding to these hyperplanes are columns in the dual auxiliary program. Perhaps some sort of column generation algorithm could be used in the dual system to bring in a column whenever the current point is in an inaccurate portion of the current PL approximation.

3. Nothing has been said in this dissertation with reference to the problem of stability in equilibria theory and computation. If the economy is, for some reason, displaced from equilibrium, what process will bring it back to the desired point? It appears as though a path of the form of the homotopy retraction algorithm could be defined which would lead to the equilibrium for any point in a neighborhood of the equilibrium. A sensitivity analysis of the auxiliary linear program could yield insight into the stability of the equilibrium.

APPENDIX

THE CALCULATION OF PIECEWISE LINEAR UTILITY FUNCTIONS

There are clearly many possible ways to compute a piecewise linear approximation to a given utility function. We will present one method which is relatively simple and easy to refine. We will not give any detailed justification for this particular method.

The epograph of a concave utility function $u: \mathbb{R}^n \to \mathbb{R}^1$ is defined as

$$\text{epo}(u) \equiv \{(x,t) \mid x \in \mathbb{R}^n,\ t \in \mathbb{R}^1,\ t \leq u(x)\} .$$

One could evaluate $u(x^0)$ by finding

$$t^0 = \sup_{(x^0,t) \in \text{epo}(u)} t = u(x^0) .$$

We will define a polyhedral approximation $\text{epo}(\hat{u})$ to $\text{epo}(u)$ and the piecewise linear approximation $\hat{u}$ of u can be evaluated as

$$\hat{u}(x^0) = \sup_{(x^0,t) \in \text{epo}(\hat{u})} t .$$

We will find an outer approximation using hyperplanes tangent to the boundary of epo u. Assume that u is differentiable. Suppose $t^0 = u(x^0)$. Then the tangent hyperplane to $\text{epo}(u) = \{(x,t) \mid t - u(x) \leq 0\}$ at (x^0,t^0) is

$$\{(x,t) \mid \langle(-\nabla u(x^0),1),\ (x,t) - (x^0,t^0)\rangle = 0\} ,$$

or letting $c^0 = t^0 - \langle \nabla u(x^0), x^0 \rangle$ we have that

$$\text{epo}(u) \subset \{(x,t) | t - \langle \nabla u(x^0), x \rangle \leq c^0\} .$$

$\text{epo}(\hat{u})$ can be calculated by choosing a number of points of tangeny $\{x^i, i = 0,1,\ldots, m\}$ and calculating the appropriate gradients:

$$\text{epo}(\hat{u}) = \{(x,t) | t - \langle \nabla u(x^i), x \rangle \leq c^i, \ i = 0,\ldots,m\} . \qquad (1)$$

The problem which remains is to make some sort of choice of points of tangency. Pick two utility levels U_1 and U_2 which are likely to be in the range of those under consideration in the problem. Given a vector $v \in S^n = \{x \in \mathbb{R}^n | \Sigma_1^n x_i = 1, \ x \geq 0\}$, it is usually quite easy to find scalars λ_i such that $u(\lambda_i v) = U_i$. Our procedure, then, is to choose n vectors $v^1, \ldots, v^n$ scattered in some manner in the unit simplex and to find the corresponding points $x^i = \lambda_i v^i$ such that $u(x^i) = U_1$, for $i \in \underline{n}$. Then we choose n more vectors $v^{n+1}, \ldots, v^{2n}$ which yield x^i such that $u(x^i) = U_2$, $i \in \underline{2n} \setminus \underline{n}$. The points x^i, $i \in \underline{2n}$ will be the points of tangency in the definition of $\text{epo}(\check{u})$, (1).

One way to choose v^i, $i \in \underline{2n}$ is to distribute them around a point v^* which maximizes the utility subject to a budget constraint assuming all prices are equal, i.e.,

$$v^* \text{ solves maximize } u(v)$$
$$\text{subject to } \sum_1^n v_i \leq 1.$$

We arbitrarily choose $v^1, \ldots, v^n$ to be convex combinations of v^* and each of the vertices of S^n. If e^i is the i^{th} unit vector for some $i \in \underline{n}$, then

$$v^i = \alpha v^* + (1-\alpha)e^i , \qquad i \in \underline{n} .$$

Let $d^i = (1/(n-1))e - (1/(n-1))e^i$ where $e \in \mathbb{R}^n$ is a vector of ones. Then v^{n+i}, is chosen to be some convex combination of v^* and d^i, the centroid of $S_i^n = \{x \in S^n | x_i = 0\}$. Let

$$v^{n+i} = \alpha v^* + (1-\alpha)d^i , \qquad i \in \underline{n}.$$

We chose α to be $1/2$, but other choices are certainly possible.

To illustrate what our piecewise linear approximations look like we must use examples in three different dimensions.

The first illustrates why we choose points of tangency at two different utility levels (Fig. A.1): in general, concave utilities have decreasing marginal returns.

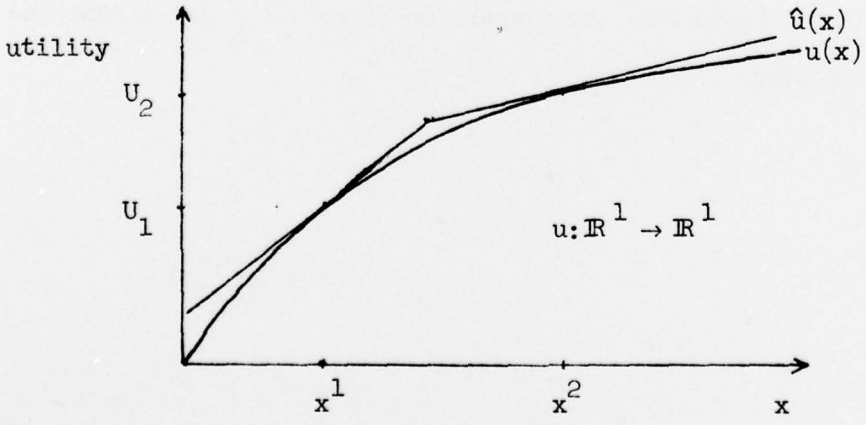


FIGURE A.1

In two dimensions we can see the piecewise linear approximation to two level curves.

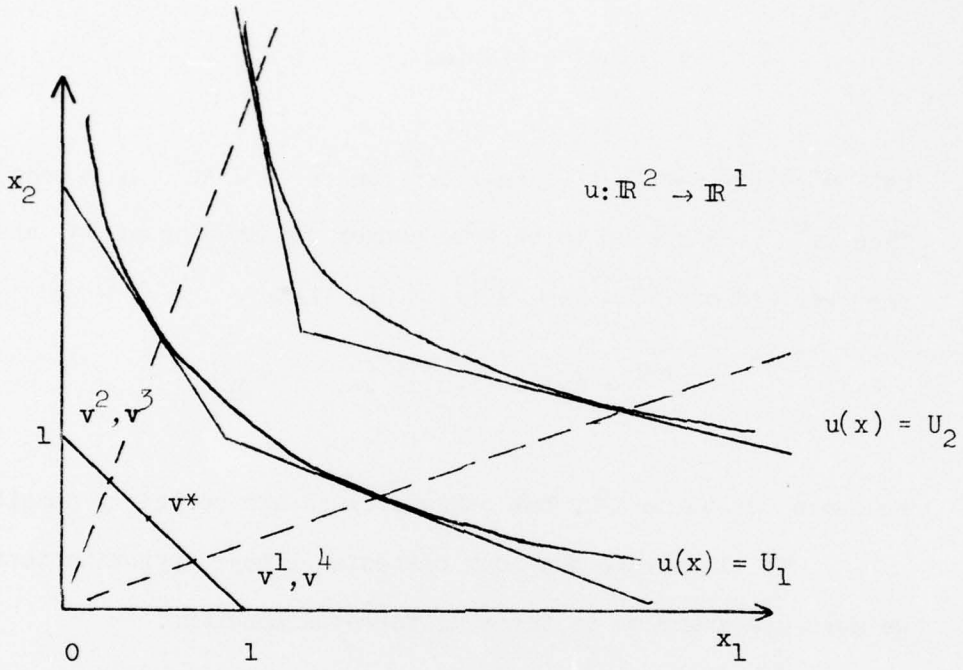


FIGURE A.2

In three dimensions the dispersion of v^i, $i \in \underline{2n}$ around the simplex is apparent.

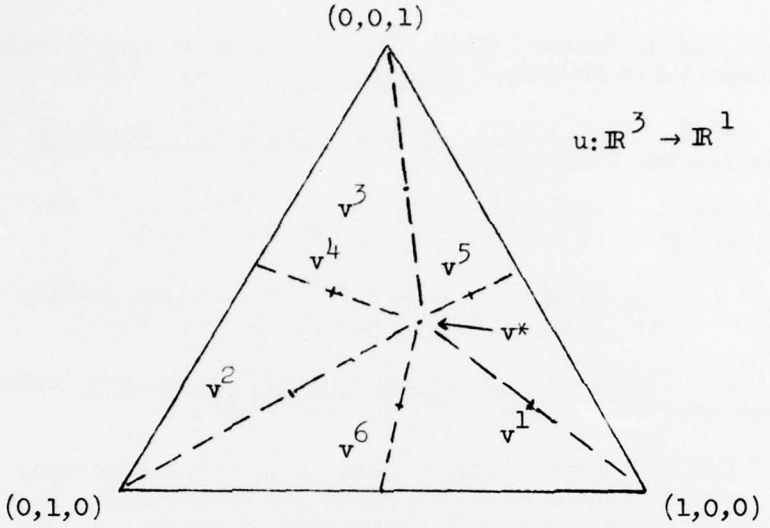


FIGURE A.3

It may be that the hyperplanes generated with this particular choice of U_1, U_2, and α yield answers which seem to be bad in some sense. In that case, information gained from the solution of that problem can be used to adjust the parameters v^*, U_1, U_2 and α so as to achieve a better approximation to u in the region of interest. With this refinement the equilibrium problem can be solved again.

BIBLIOGRAPHY

Arrow, K.J. and G. Debreu [1954], "Existence of an Equilibrium for a Competitive Economy," Econometrica, 22, pp. 265-90.

Arrow, K. and F. Hahn [1971]. General Competitive Analysis, San Francisco: Holden Day Publishing Company.

Arrow, K.J. and L. Hurwicz [1958], "On the Stability of the Competitive Equilibrium, I", Econometrica, 26, pp. 522-52.

Avila, J.H. [1974], "The Feasibility of Continuation Methods for Nonlinear Equations," SIAM J. Numer. Anal., 11, pp. 102-22.

Dantzig, G.B. [1963], Linear Programming and Extensions, Princeton, New Jersey: Princeton University Press.

Dantzig, G.B., B.C. Eaves, and D. Gale [1976], "An Algorithm for a Piecewise Linear Model of Trade and Production with Negative Prices and Bankruptcy," Department of Operations Research, Stanford University, August, 1976.

Debreu, G. [1951], "The Coefficient of Resource Utilization," Econometrica, 19, pp. 273-292.

Debreu, G. [1959], Theory of Value, New York: Wiley.

Drews, W.P. [1976], "Demonstration Examples of Trade-Payments Model," unpublished communication.

Eaves, B. C. [1971b], "The Linear Complementarity Problem," Management Sciences, 17, pp. 612-34.

Eaves, B.C. [1975], "A Finite Algorithm for the Linear Exchange Model," Technical Report SOL 75-4, Stanford University.

Elken, T. [1977], "On the Solution of Nonlinear Equations by Path Methods," SOL Technical Report 77-25, Department of Operations Research, Stanford University.

Ginsburgh, V. and J. Waelbroeck [1975], "A General Equilibrium Model of World Trade, Part I, Full Format Computation of Economic Equilibria," Cowles Foundation Discussion Paper No. 412, Yale University.

Ginsburgh, V. and J. Waelbroeck [1975], "A General Equilibrium Model of World Trade, Part II, the Empirical Specification," Cowles Foundation Discussion Paper No. 413, Yale University.

Lemke, C.E. [1965], "Bimatrix Equilibrium Points and Mathematical Programming," Management Sciences, 11, pp. 681-89.

Mantel, R. [1968], "Toward a Constructive Proof of the Existence of Equilibrium in a Competitive Economy," Yale Economic Essays, 8, pp. 155-196.

Murtagh, B.A. and M.A. Saunders [1977], "A Large-Scale Nonlinear Programming System, User's Guide," Technical Report SOL 77-9, Department of Operations Research, Stanford University.

Ortega, J.M. and W.C. Rheinboldt [1970], Iterative Solution of Nonlinear Equations in Several Variables, New York: Academic Press.

Quirk, J.R. and R. Saposnik [1968], Introduction to Equilibrium Theory and Welfare Economics, New York: McGraw-Hill.

Scarf, H. [1960], "Some Examples of the Global Instability of the Competitive Equilibrium," International Economic Review, 1, pp. 157-72.

Scarf, H. [1967a], "The Approximation of Fixed Points of a Continuous Mapping," SIAM J. Applied Math., 15, pp. 1328-43.

Scarf, H. [1967b], "On the Computation of Equilibrium Prices," in Ten Essays in Honor of Irving Fischer, ed. Fellner, et al., pp. 207-230, New York: John Wiley and Sons.

Scarf, H. [1973], The Computation of Economic Equilibria, New Haven: Yale University Press.

Smale, S. [1976], "A Convergent Process of Price Adjustment and Global Newton Methods," Journal of Math. Econ., 3, pp. 107-120.

Wald, A. [1935], "Uber die eindeutige positive Losbarkeit der neuen Produktionsgleichungen." Ergibnisse eines Mathematischen Kolloquiums, 6, pp. 12-20.

Walras, L. [1874], Elements d'Economie Politique Pure, Lausanne: Corbaz. [Translated as Elements of Pure Economics, trans. W. Jaffe, London: Allen and Unwin, 1954.]

Wilson, R. [1976], "The Bilinear Complementarity Problem and Competitive Equilibria of Linear Economic Models," Technical Report SOL 76-2, Department of Operations Research, Stanford University. To appear in Econometrica.

Winkler, C. [1974], "Basis Factorization for Block-Angular Linear Programs: Unified Theory of Partitioning and Decomposition Using the Simplex Method," SOL Technical Report 74-19, Department of Operations Research, Stanford University.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER SOL 77-26	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) THE COMPUTATION OF ECONOMIC EQUILIBRIA BY PATH METHODS		5. TYPE OF REPORT & PERIOD COVERED Technical report 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Thomas R. Elken		8. CONTRACT OR GRANT NUMBER(s) N00014-75-C-0267 N00014-75-C-0865
9. PERFORMING ORGANIZATION NAME AND ADDRESS Department of Operations Research, SOL Stanford University Stanford, CA 94305		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NR-047-064 NR-047-143
11. CONTROLLING OFFICE NAME AND ADDRESS Operations Research Program Office of Naval Research Arlington, VA 22217		12. REPORT DATE October 1977 13. NUMBER OF PAGES 119
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

This document has been approved for public release and sale; its distribution is unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

ECONOMIC EQUILIBRA
PIECEWISE LINEAR ECONOMY
BILINEAR COMPLEMENTARITY PROBLEM
PATH-FOLLOWING ALGORITHMS
HOMOTOPY RETRACTION METHOD
PIECEWISE LINEAR APPROXIMATION OF CONCAVE UTILITY
COMPUTATIONAL EXPERIMENTS

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

SEE ATTACHED

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

SOL 77-26: THE COMPUTATION OF ECONOMIC EQUILIBRIA BY PATH METHODS,
by Thomas R. Elken

An introduction to the economic equilibrium model is given and it is demonstrated that a path method can be used to compute equilibria for pure exchange economies in a nonlinear setting.

Next, a model is described for an economy in which the utility functions are piecewise linear and the consumption and production sets are polyhedral. It is shown that an equilibrium for this economy is the solution to a system of bilinear equations subject to certain linear inequality and complementarity constraints.

Two approaches are discussed for computing equilibria for such economies. The first is the bilinear complementarity algorithm (BCA) and the second is the homotopy retraction algorithm (HRA). Convergence proofs are given for both methods using the general theory for path methods described above.

The BCA and HRA have been implemented as computer programs. Detailed descriptions of the algorithms are given, and the results of some numerical experiments are reported. Seven small problems were solved by both algorithms. No conclusion could be drawn as to which algorithm was superior, but both performed well enough that it appears that much larger equilibrium problems also can be solved efficiently by these methods.