Economics Letters 43 (1993) 15–21

0165-1765/93/$06.00 © 1993 Elsevier Science Publishers B.V. All rights reserved

The coverability problem in input–output systems

Giacomo Bonanno *

Department of Economics, University of California, Davis, CA 95616-8578, USA

Received 12 April 1993
Accepted 29 June 1993

Abstract

The notion of a *vector replacement system*, developed in the computer science literature, is used to analyze the reachability problem in integer-valued input–output models. An *integer-valued input–output system with initial resources* is a 5-tuple $\langle n, m, A, B, \mu \rangle$, where n is the number of commodities, m the number of production processes, A is the $n \times m$ input matrix, B is the $n \times m$ output matrix and $\mu \in \Re^n$ is the initial vector of resources. The elements of A, B and μ are non-negative integers. A vector μ' is *reachable* from μ if it is possible to transform μ into μ' through a sequence of production processes, without ever violating the feasibility constraint represented by the available resources.

1. Introduction

The notion of an input–output system was first introduced by von Neumann in 1937 [von Neumann (1945)]. The open Leontief system [Leontief (1941)] is a special case of it. An *input–output system* is a 4-tuple $\langle n, m, A, B \rangle$, where n is the number of commodities, m the number of production processes, A is an $n \times m$ *input matrix* and B is an $n \times m$ *output matrix*. The elements of A and B are non-negative real numbers. Thus for $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, m\}$, the element a_{ij} of A is the quantity of commodity i *used* by production process j, when the latter is operated at unit intensity, while the element b_{ij} of B is the quantity of commodity i *produced* by process j (when operated at unit intensity). It is assumed that each production process requires at least one input (every column of A contains at least one positive entry). When $n = m$ and B is the identity matrix, then A is called a *Leontief matrix*. An important assumption that is usually made when dealing with input–output systems is that all the production processes can be scaled up or down by an arbitrary factor, that is, the corresponding technology displays *constant returns to scale*. We shall see later that for the analysis of section 2 the assumption of constant returns to scale is *not* needed.

An *input–output system with initial resources* is an input–output system together with a non-negative vector $\mu_0 \in \Re^n$ of initial resources. A question that has been addressed only indirectly and implicitly in the literature is the following: Given an input–output system with initial resources μ_0, what are all the possible commodity vectors μ into which μ_0 can be transformed? This question motivates the following definition.

* I am grateful to Kanapathipillai Sanjeevan for introducing me to the computer science literature referenced in this paper

Definition. Given an input–output system with initial resources μ_0, we say that the non-negative commodity vector $\mu \in \Re^n$ is *reachable* from μ_0 *through the sequence* $\langle x_1, x_2, \ldots, x_k \rangle$ of non-negative intensity vectors in $\Re^m$ if

$$\mu = \mu_0 + (B - A)(x_1 + x_2 + \cdots + x_k) , \tag{1}$$

$$\mu_0 + (B - A)(x_1 + x_2 + \cdots + x_r) \geq 0 , \quad \text{for every } r = 1, \ldots, k , \tag{2}$$

$$Ax_1 \leq \mu_0 , \tag{3}$$

and

$$Ax_r \leq \mu_0 + (B - A)(x_1 + x_2 + \cdots + x_{r-1}) , \quad \text{for every } r = 2, \ldots, k . \tag{4}$$

Finally, we say that $\mu' \in \Re^n$, $\mu' \geq 0$, is *coverable from* μ_0 if either (1) $\mu' \leq \mu_0$, or (2) there exists a $\mu \geq \mu'$ and a sequence of intensity vectors $\langle x_1, x_2, \ldots, x_k \rangle$ such that μ is reachable from μ_0 through this sequence.

The interpretation of the above definition is clear. Starting from μ_0, by activating x_1 one first obtains $\mu_1 = \mu_0 + (B - A)x_1$. Then from μ_1, by activating x_2, one obtains $\mu_2 = \mu_1 + (B - A)x_2$, etc. Condition (1) requires that at the end of the sequence, by activating x_k, one obtains $\mu_k = \mu$. Condition (2) requires that, for every $r = 1, \ldots, k$, μ_r be non-negative. Condition (3) requires that the initial activity vector x_1 be feasible, given μ_0. Finally, condition (4) requires that, for every $r = 2, \ldots, k$, activity vector x_r be feasible given μ_{r-1}.

The reachability or coverability problem has only been addressed implicitly in the input–output literature. For example, an implicit answer to the coverability problem in a Leontief framework is implied by the following theorem [for a proof see Gale (1960, ch. 9)].

Theorem. Let A be an $n \times n$ Leontief matrix. Then the following conditions are equivalent:

(1) *There exists a positive vector $\bar{x} \in \Re^n$ such that $\bar{x} > A\bar{x}$ (if w and z are two vectors in $\Re^p$, $p \geq 1$, $w > z$ means that every component of w is greater than the corresponding component of z).*

(2) *The matrix $(I - A)$ is invertible and its inverse is non-negative (where I denotes the identity matrix).*

(3) $\lambda < 1$, *where λ is the maximum eigenvalue of A (recall that, by the Perron–Frobenius theorem, there exists a positive real number λ such that*: (i) *λ is an eigenvalue of A, and* (ii) *if λ' is a real or complex eigenvalue of A, then $|\lambda'| \leq \lambda$).*

A Leontief matrix that satisfies any of the above conditions is called *productive*. Using the above theorem it is easy to prove the following.

Proposition. Let A be a productive *Leontief matrix. Let $\mu_0 > 0$ be a vector of initial resources and let $y \in \Re^n$ be an arbitrary non-negative vector. Then y is coverable from μ_0.*

Sketch of proof. Consider first the case where A is irreducible. Let λ be the maximum eigenvalue of A and $\hat{x}$ be a corresponding eigenvector. Then by the Perron–Frobenius theorem, $\hat{x} > 0$. Choose a scalar $\varepsilon > 0$ small enough so that $A\varepsilon\hat{x} \leq \mu_0$. By activating the production vector $\varepsilon\hat{x}$, μ_0 can be transformed into

$$\mu = \mu_0 - A\varepsilon\hat{x} + \varepsilon\hat{x} > \mu_0 - A\varepsilon\hat{x} + \lambda\varepsilon\hat{x} \text{ (since } \lambda < 1) = \mu_0 .$$

Repeating a sufficiently large number of times we obtain a commodity vector $\mu' \geq y$.

If A is not irreducible, change some of the zero entries of A to $\delta > 0$, small enough so that the resulting matrix is still productive and is irreducible. Then apply the above reasoning to the new matrix. □

Two things are crucial for the validity of the above proposition: (1) the hypothesis that $\mu_0 > 0$, and (2) the fact that there are constant return to scale. For example, let

$$A = \begin{pmatrix} 0.4 & 0.5 \\ 0.1 & 0.3 \end{pmatrix} \quad \text{and} \quad \mu_0 = \begin{pmatrix} 10 \\ 0 \end{pmatrix} .$$

Then, even though A is productive, the only vectors that can be covered from μ_0 are those of the form $(\alpha, 0)$, with $0 \leq \alpha \leq 10$. On the other hand, if $\mu_0 > 0$, but small, and all the process have some minimum scale of operation, it may not be possible to activate any production process. Notice also that von Neumann's technological expansion rate for this input–output system (where the output matrix is the identity matrix) is 1.727 with corresponding optimal intensity vector (0.941, 0.337).

In the next section, drawing from the computer science literature, we provide an answer to the coverability problem for arbitrary vectors μ_0 and μ, within a framework that does *not* require the assumption of constant return to scale.

2. The Karp–Miller coverability tree

We shall analyze input–output systems (with initial resources) $\langle n, m, A, B, \mu_0 \rangle$ with the added restriction that all the elements of A, B and μ_0 be (non-negative) *integers* [such a system is known as a *vector replacement system* in the computer science literature and was introduced by Keller (1972); it is a generalization of the notion of a *vector addition system* that was first introduced by Karp and Miller (1969); if the two matrices A and B are replaced by the matrix $B-A$, whose entries are therefore allowed to be negative integers, then one obtains a vector addition system]. The restriction that the elements of A, B and μ_0 belong to the set $\mathbb{N}$ of non-negative integers is not an important restriction for the following reason. In any application the entries of the input and output matrices will be rational numbers. It is therefore possible to redefine the units of measurement for some or all the commodities in such a way that all the entries become integers. For example, consider the (Leontief) system examined before, where

$$A = \begin{pmatrix} 0.4 & 0.5 \\ 0.1 & 0.3 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} .$$

For each commodity define a new unit which is equal to 1/10 of the old unit. Then the input and output matrices become

$$A = \begin{pmatrix} 4 & 5 \\ 1 & 3 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 10 & 0 \\ 0 & 10 \end{pmatrix} .$$

Notice that we *do not assume constant returns to scale*. Thus it may not be possible to scale a production process up or down by an arbitrary factor $\alpha > 0$. However, it is certainly possible to *repeat* a production process several times. For instance, the first production process in the above example can be repeated three times leading to a total output of 30 units of good 1 and a total consumption of 12 units of good 1 and 3 units of good 2.

For every $j = 1, \ldots, m$, let $e_j \in \Re^m$ be the unit intensity vector whose jth coordinate is 1 and all the other coordinates are 0. Given a vector of initial resources, $\mu_0 \in \mathbb{N}^n$, the definition of coverability of $\mu \in \mathbb{N}^n$ from μ_0 is the same as in section 1, except that we require each activity vector x_j in the sequence $\langle x_1, \ldots, x_k \rangle$ to belong to the set $\{e_1, e_2, \ldots, e_m\}$.

Karp and Miller (1969) constructed an algorithm that yields the so-called *coverability tree*. Associated with each node of the tree is an *extended vector*. While the initial vector of resources, μ_0, is a point in $\mathbb{N}^n$, an extended vector is a point in the set $(\mathbb{N} \cup \{\infty\})^n$. The symbol ∞ stands for 'infinity' and represents a number of units of a commodity that can be made arbitrarily large. For any integer k, we define:

$$\infty + k = \infty\,, \qquad \infty - k = \infty\,, \qquad k < \infty\,, \qquad \infty \leq \infty\,.$$

Let a_j denote the jth column of the input matrix A and b_j denote the jth column of the output matrix B ($j = 1, \ldots, m$). Given a vector of initial resources, μ_0, we associate μ_0 with the root of the tree. For every $j = 1, \ldots, m$ for which $\mu_0 \geq a_j$, we construct a new node and associate with it the commodity vector $\mu' = \mu_0 - a_j + b_j$. We then repeat the procedure starting from this new vector μ' (for every $j = 1, \ldots, m$ for which $\mu' \geq a_j$, we construct a new node and associate with it the vector $\mu'' = \mu' - a_j + b_j$, etc.). However, we introduce rules aimed at making the tree finite, so that, starting from the root, every path leads to a terminal node. Obvious terminal nodes are *dead ends* (a dead end is a vector $\hat{\mu}$ such that, for *no* $j = 1, \ldots, m$, $\hat{\mu} \geq a_j$), or nodes whose associated vectors are duplicates of vectors previously obtained. The symbol ∞ is used to obtain the remaining terminal nodes. Consider a feasible sequence of unit intensity vectors $\sigma = e_{j_1} e_{j_2} \cdots e_{j_k}$ which starts at a commodity vector μ' and leads to the commodity vector μ'', with $\mu'' \geq \mu'$, $\mu'' \neq \mu'$. Since the sequence σ was feasible starting from μ', it will still be feasible starting from μ'' and will lead to a new commodity vector $\mu''' = \mu'' + (\mu'' - \mu')$ [since the sequence σ always adds $(\mu'' - \mu')$]. If we repeat σ n times, we add the commodity vector $(\mu'' - \mu')$ n times. Thus, for those commodities that were increased by the sequence, we can create an arbitrarily large number of units simply by repeating the sequence σ as often as desired. During the construction of the tree, if at any time we obtain a vector μ'' with $\mu'' \geq \mu'$ and $\mu'' \neq \mu'$, we replace μ'' with an extended vector where there is the symbol ∞ in place of those components of μ'' that are greater than the corresponding components of μ' (the discussion based on Fig. 1 below will illustrate this).

The Karp–Miller coverability tree is constructed using the following algorithm. Every node v of the tree is assigned two labels: an extended vector $\mu[v] \in (\mathbb{N} \cup \{\infty\})^n$ and a label $\ell[v] \in$ {open, interior, duplicate, dead end, infinite}. The algorithm terminates when there are no nodes v such that $\ell[v] =$ 'open'.

STEP 1. Let μ_0 be the vector of initial resources. Label the root v_0 as follows: $\mu[v_0] = \mu_0$, $\ell[v_0] =$ 'open'.

STEP 2. While nodes v such that $\ell[v] =$ 'open' exist, do the following:

- STEP 2.1. Select a node v such that $\ell[v] =$ 'open'.
- STEP 2.2. If $\mu[v]$ is identical to $\mu[v']$ for some node $v' \neq v$ on the path from the root to v, then set $\ell[v] =$ 'duplicate' and go back to Step 2.1.
- STEP 2.3. If there is no $j = 1, \ldots, m$ for which $\mu[v] \geq a_j$, set $\ell[v] =$ 'dead end' and go back to Step 2.1.
- STEP 2.4. If each coordinate of $\mu[v]$ is the symbol ∞, set $\ell[v] =$ 'infinite' and go back to Step 2.1.

STEP 2.5. While there exist $j = 1, \ldots, m$ for which $\mu[v] \geq a_j$, do the following for every such j.

STEP 2.5.1. Set $\ell[v]$ = 'interior'. Draw a new vertex w and an arc from v to w. Label the arc with production process j. Obtain the commodity vector $\mu' = \mu[v] - a_j + b_j$.

STEP 2.5.2. **IF** on the path from the root to v there exists a node $z \neq v$ such that $\mu' \geq \mu[z]$ and $\mu' \neq \mu[z]$, then replace each component of μ' which is greater than the corresponding component of $\mu[z]$ with the symbol ∞; let μ'' be the resulting extended vector; set $\mu[w] = \mu''$ and $\ell[w]$ = 'open'; **ELSE** go to the next step.

STEP 2.5.3. Set $\mu[w] = \mu'$ and $\ell[w]$ = 'open'.

Karp and Miller proved that the algorithm terminates (all nodes are labeled as either interior or duplicate or dead-end or infinite) and therefore yields a finite tree. Thus the coverability problem is decidable.

Example. Let

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 2 & 0 & 4 \\ 0 & 1 & 0 \end{pmatrix}, \qquad B = \begin{pmatrix} 0 & 2 & 0 \\ 0 & 0 & 6 \\ 0 & 2 & 0 \\ 5 & 0 & 2 \end{pmatrix} \quad \text{and} \quad \mu_0 = \begin{pmatrix} 1 \\ 2 \\ 2 \\ 1 \end{pmatrix}.$$

Then using the Karp–Miller algorithm one obtains the coverability tree of Fig. 1. Activating process 1 at unit intensity leads from $(1, 2, 2, 1)$ to $(1, 2, 2, 1) - (1, 0, 2, 0) + (0, 0, 0, 5) = (0, 2, 0, 6)$. The only process that can be activated (at unit intensity: for brevity, from now on the clause 'at unit intensity' will be omitted) at $(0, 2, 0, 6)$ is process 2 which leads to $(2, 0, 2, 5)$. Here the only process that can be activated is process 1, leading to $(1, 0, 0, 10)$, which is a dead end, since no process can be activated. This explains the left branch of the tree. Similarly for the right branch. Now let us go back to the root. Activating first process 2 and then process 3 leads first to $(3, 0, 4, 0)$ and then to $(3, 6, 0, 2)$. Now activating process 2 (the only one that can be activated) we would get $(5, 4, 2, 1)$. This is greater than $(1, 2, 2, 1) = \mu[v_0]$. The first and second components are greater, hence we replace them with the symbol ∞. Thus $\mu[v_8] = (\infty, \infty, 2, 1)$. This means that be repeating the sequence of processes $\langle 2, 3, 2 \rangle$ a sufficiently large number of times one can increase the quantity obtained of commodities 1 and 2 to any desired level. Activating process 1 at $\mu[v_8] = (\infty, \infty, 2, 1)$ one would obtain $(\infty - 1 = \infty, \infty, 0, 6)$ which is greater than $\mu[v_5]$, the last component being greater. Therefore we replace the last component with ∞ and let $\mu[v_9] = (\infty, \infty, 0, \infty)$. The remaining nodes are obtained similarly.

3. Conclusion

The purpose of this paper was to show the relevance and usefulness of some concepts and techniques developed in the computer science literature for the analysis of the production possibility set of an economy, starting from an initial vector of resources. For a more extensive analysis of input–output systems, using the notion of a Petri net (also developed in the computer science literature), see Bonanno (1993).

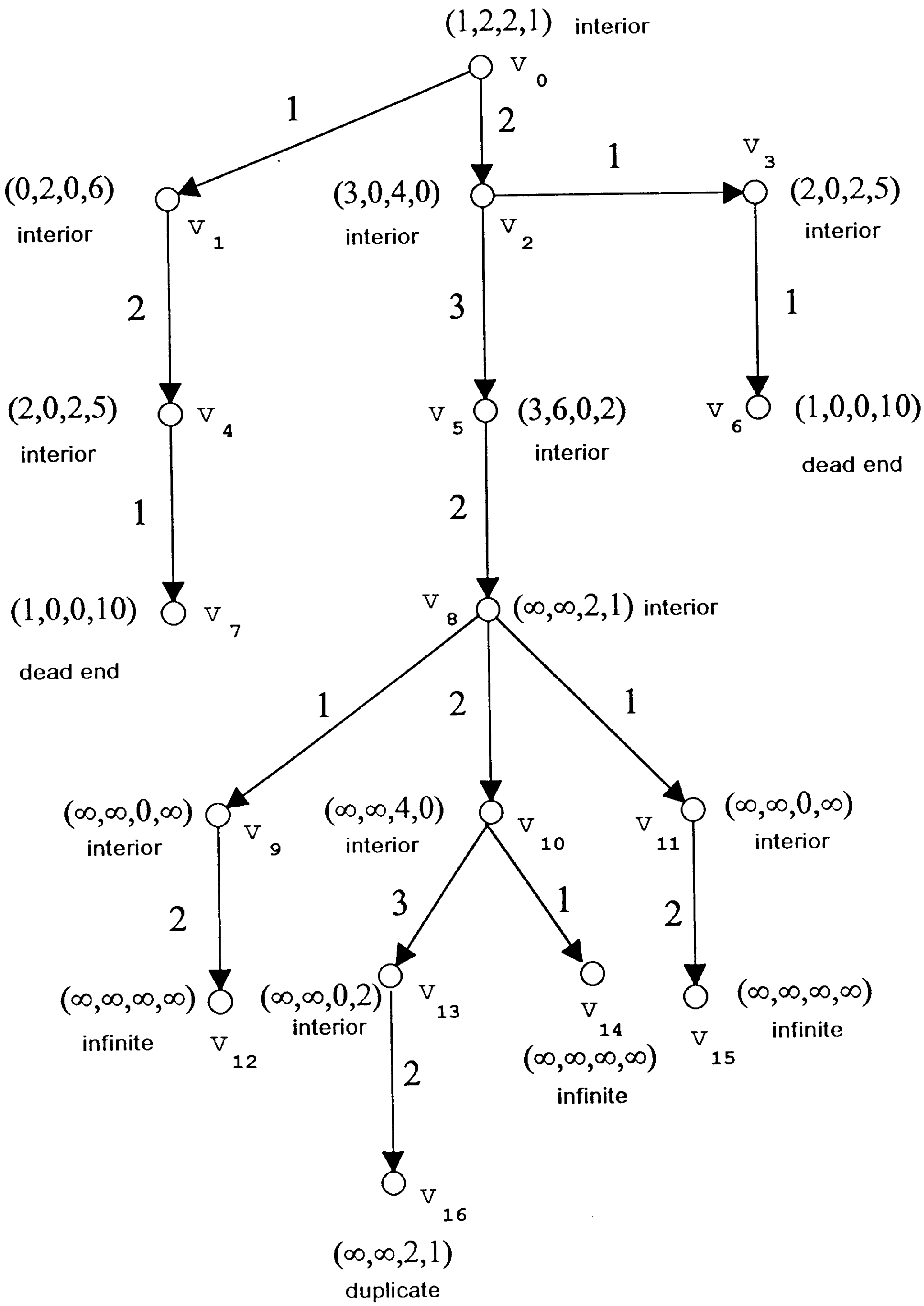
(1,2,2,1) interior
V 0
1
2
(0,2,0,6)
interior
V 1
(3,0,4,0)
interior
V 2
1
V 3
(2,0,2,5)
interior
2
3
1
(2,0,2,5)
interior
V 4
V 5
(3,6,0,2)
interior
V 6
(1,0,0,10)
dead end
1
2
(1,0,0,10)
V 7
dead end
V 8
(∞,∞,2,1) interior
1
2
1
(∞,∞,0,∞)
interior
V 9
(∞,∞,4,0)
interior
V 10
V 11
(∞,∞,0,∞)
interior
2
3
1
2
(∞,∞,∞,∞)
infinite
V 12
(∞,∞,0,2)
interior
V 13
V 14
(∞,∞,∞,∞)
infinite
V 15
(∞,∞,∞,∞)
infinite
2
V 16
(∞,∞,2,1)
duplicate

Fig. 1.

References

Bonanno, G., 1993, Petri nets and input–output analysis, Department of Economics Working Paper no. 93-5, University of California, Davis.

Gale, D, 1960, The theory of linear economic models (McGraw-Hill, New York).

Karp R.M. and R.E. Miller, 1969, Parallel program schemata, Journal of Computer and System Sciences 3, 247–295.

Keller, R., 1972, Vector replacement systems: A formalism for modeling asynchronous systems, Technical report 117, Computer Science Laboratory, Princeton University (revised 1974).

Leontief, W., 1941, The structure of the American Economy 1919–1929 (Harvard University Press, Harvard, MA).

von Neumann, J., 1945, A model of general economic equilibrium, Review of Economic Studies 13 (English translation of a paper published in German in 1937).