CHAPTER 9

THE SIMPLEX METHOD USING MULTIPLIERS

While each iteration of the simplex method requires that a whole new tableau be computed and recorded, it may be observed that only the modified cost row and the column corresponding to the variable entering the basic set play any role in the decision process. The idea behind the "Simplex Method Using Multipliers" is to use a set of numbers called *simplex multipliers* (*prices*) and the *inverse of the basis* to generate directly from the original equations just the information required for these decisions. This method is also referred to in the literature as the *revised simplex method* [Dantzig and Orchard-Hays, 1953-1].

The modified cost equation, obtained by eliminating the basic variables from the cost form, can be obtained directly from the original system by multiplying the original equations by weights, summing, and then subtracting from the objective equation. It is these weights that are called simplex multipliers and, in a somewhat broader context (see Chapter 12) are called "prices." From a theoretical point of view they are most important as they are related to the variables of the dual system and they play a role analogous to Lagrange multipliers in the calculus (Chapter 6). They are most valuable, as we shall see in Chapter 12, for determining the bottlenecks in a program, the payoff from increasing availabilities of certain stocks, the effect of an increase in capacity, or the value of a proposed new process.

The computational advantages of this approach are:

(a) Less data is recorded from one iteration to the next, which permits more significant figures to be carried or a larger problem to be solved within the limited memory capacity of an electronic computer.
(b) Where the original data has a high percentage of zero coefficients (90 percent or higher is quite common), there are less multiplications (see the computational remarks at the end of this section).
(c) A simple device exists that avoids degeneracy and hence the possibility of "circling" in the simplex algorithm (see *Lexicographic Rule* at the end of § 9-3). (Because the inverse of the basis is part of the full tableau, this device is equally applicable to the original procedure as well.)

The simplex method using multipliers is based on theory already covered in § 8-5. Accordingly, our purpose will be to bring out its operational features. Because some readers might find that the matrix notation of § 8-5 obscures the computational aspects, we have tended to avoid its use here.

9-1. AN ILLUSTRATION USING MULTIPLIERS

To illustrate the technique, consider problem (1)

Cycle 0

(1)

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	$-z$	Constants
1		1		1		1					3
	1		1		1		1				2
1			1					1			2
	1			1					1		1
−8	−9	−7	−6	−8	−9					1	−90

Basis B Cycle-k Initial Basis

which is in canonical form relative to the variables (x_7, x_8, x_9, x_{10}, $-z$). After several iterations of the simplex algorithm, this can be written in equivalent canonical form (2) relative to, say, the variables (x_1, x_2, x_3, x_4, $-z$).

Cycle k

(2)

Basic Variables	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	$-z$	Constants
x_1	1				1	−1		−1	1	1		1
x_2		1			1					1		1
x_3			1			1	1	1	−1	−1		2
x_4				1	−1	1		1		−1		1
$-z$					3	−4	7	5	1	4	1	−53
	●	●	●	○		★						

Inverse of the Basis $[\beta_{ij}]$

Multipliers $\pi = (-7, -5, -1, -4)$

The basis B for cycle k, see (3), is the square array of coefficients associated with the basic variables in the original system (1) where, for this discussion, we exclude z and the z-equation.

The first column of B corresponds to that basic variable in (2) with unit coefficient in the first row, . . . the k^{th} column of B corresponds to the one with unit coefficient in the k^{th} row, etc. (In other words the columns of B must be ordered to correspond to whatever basic variables are listed in the first column of (2); see § 4-2.) For the case at hand (see § 8-4-(11) for definition of inverse),

$$(3)\qquad B = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}; \quad B^{-1} = [\beta_{ij}] = \begin{bmatrix} 0 & -1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & -1 & -1 \\ 0 & 1 & 0 & -1 \end{bmatrix}$$

The inverse of the basis, denoted B^{-1} or $[\beta_{ij}]$, is the square array of coefficients in (2) (excluding the z-equation) associated with the variables (x_7, x_8, x_9, x_{10}) where the latter form the initial set of basic variables in (1). To prove this assertion, let us note, according to (§ 8-4-(20)), that *the inverse of the basis can be used to compute the coefficients $\bar{a}_{ij}$ in any column of* (2) *from the corresponding column j of the original system* (1) *by the formulas*

$$(4)\qquad \begin{aligned} \bar{a}_{1j} &= \beta_{11}a_{1j} + \beta_{12}a_{2j} + \beta_{13}a_{3j} + \beta_{14}a_{4j} \\ \bar{a}_{2j} &= \beta_{21}a_{1j} + \beta_{22}a_{2j} + \beta_{23}a_{3j} + \beta_{24}a_{4j} \\ \bar{a}_{3j} &= \beta_{31}a_{1j} + \beta_{32}a_{2j} + \beta_{33}a_{3j} + \beta_{34}a_{4j} \\ \bar{a}_{4j} &= \beta_{41}a_{1j} + \beta_{42}a_{2j} + \beta_{43}a_{3j} + \beta_{44}a_{4j} \end{aligned}$$

Since column $\langle a_{17}, a_{27}, a_{37}, a_{47}\rangle = \langle 1, 0, 0, 0\rangle$, substitution into (4) yields $\langle \bar{a}_{17}, \bar{a}_{27}, \bar{a}_{37}, \bar{a}_{47}\rangle = \langle \beta_{11}, \beta_{21}, \beta_{31}, \beta_{41}\rangle$ i.e., the column of coefficients of x_7 in (2) is the *same* as the first column of B^{-1}. In general, if x_j is any variable in (1) whose coefficients form a unit vector with unity in the i^{th} equation, then by substitution in (4), the corresponding column of coefficients in (2) is the same as the i^{th} column of B^{-1}. Hence, the inverse of the basis for cycle k is the set of coefficients in the tableau of cycle k of the variables which were basic in cycle 0.

The simplex multipliers or prices are defined as numbers $\pi_1, \pi_2, \pi_3, \pi_4$ such that the weighted sum formed by multiplying the first equation of (1) by π_1, the second by π_2, etc., and adding, will, when subtracted from the z-equation, eliminate the basic variables and yield the modified z-equation of (2). In particular it is obvious, since the only non-zero coefficient of x_7 in (1) is unity (from the first equation), that the resulting coefficient of x_7 in the z-equation of (2) is $-\pi_1$. Similarly, the coefficients of x_8, x_9, x_{10} in (2) must be $-\pi_2, -\pi_3, -\pi_4$. Thus $\pi_1 = -7$, $\pi_2 = -5$, $\pi_3 = -1$, $\pi_4 = -4$. The fact that these values are correct can be directly verified by multiplying them by the corresponding equations of (1), summing, and subtracting from the z-equation to reproduce the z-equation of (2). Thus the *simplex multipliers $\pi_1, \pi_2, \pi_3, \pi_4$ can be used to compute the relative cost factor $\bar{c}_j$ in* (2) *from the corresponding column of the original system by the formula* (see § 8-5-(16)):

$$(5)\qquad \bar{c}_j = c_j - (\pi_1 a_{1j} + \pi_2 a_{2j} + \pi_3 a_{3j} + \pi_4 a_{4j})$$

In our discussion so far we have excluded $-z$ from the set of basic variables and the z-equation from the basis and its inverse. If, alternatively, we include them, the basis B^* associated with the basic variables x_1, x_2, x_3, x_4 and $-z$ for some cycle k is composed of the coefficients of these variables in the original system (1).

$$(6)\qquad B^* = \left[\begin{array}{cccc|c} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ \hline -8 & -9 & -7 & -6 & 1 \end{array}\right]; \quad [B^*]^{-1} = \left[\begin{array}{cccc|c} 0 & -1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & -1 & -1 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ \hline 7 & 5 & 1 & 4 & 1 \end{array}\right].$$

The inverse of B^* then will be the coefficients of the initial basic variables x_7, x_8, x_9, x_{10}, and $-z$ in the canonical form for cycle k. According to § 8-4-(20), if the elements that appear in the k^{th} row of $[B^*]^{-1}$ are used to multiply respectively the five equations of (1), their sum also will reproduce the k^{th} equation of (2). From this point of view, the equations of (4) should result from using the first m rows of $[B^*]^{-1}$, and (5) should result from using the last row of $[B^*]^{-1}$. This is true because B^* differs from B in (6) by the border column of zeros, the border row of costs, and $+1$ in the lower right hand corner; similarly $[B^*]^{-1}$ differs from B^{-1} by the border column of zeros, the border row of the negative prices, and $+1$ in the lower right hand corner.

In the simplex method using multipliers, only certain key columns of the simplex tableau for cycle k are assumed known at the start of the cycle, namely:

(a) *The inverse of the basis B^** for cycle k, which numerically is the same as the columns of cycle k corresponding to the basic variables of cycle 0.

(b) *The basic feasible solution for cycle k*, which is expressed as the constant values $\bar{b}_i$, $-\bar{z}_0$ and the basic variables to which they correspond, the values of all other variables being zero. *All other data required to carry out steps of the standard simplex process are computed directly from the initial tableau as needed.*

To illustrate, the unshaded part of (7) shows the *recorded* part of the tableau at the start of cycle k.

Start of Cycle k

(7)

Basic Variables	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	$-z$	Constants
x_1								-1	1	1		1
x_2										1		1
x_3							1	1	-1	-1		2
x_4								1		-1		1
$-z$							7	5	1	4	1	-53

← Inverse, $[B^*]^{-1}$ →

The next step is to compute the relative cost factors $\bar{c}_j$, which are the values appearing in the bottom row of (2), from the data appearing in the tableau of cycle 0 and the known multipliers ($\pi_1 = -7$, $\pi_2 = -5$, $\pi_3 = -1$, $\pi_4 = -4$). From (5) we have, for example,

$$(8)\quad \bar{c}_1 = c_1 - \pi_1 a_{11} - \pi_2 a_{21} - \pi_3 a_{31} - \pi_4 a_{41} = -8 + 7 \cdot 1 + 5 \cdot 0 + 1 \cdot 1 + 4 \cdot 0 = 0$$

. . . .

$$\bar{c}_6 = c_6 - \pi_1 a_{16} - \pi_2 a_{26} - \pi_3 a_{36} - \pi_4 a_{46} = -9 + 7 \cdot 0 + 5 \cdot 1 + 1 \cdot 0 + 4 \cdot 0 = -4$$

As a check, the values of $\bar{c}_j$ for basic variables computed in this manner should be zero, while for those j corresponding to the basic variables of cycle 0, the values of $\bar{c}_j$ should equal the negative of the price, $-\pi_i$, corresponding to the i^{th} basic variable. If the computed $\bar{c}_j$'s are entered in the tableau and the value of s determined such that $\bar{c}_s = \text{Min } \bar{c}_j$, indicated by a ★ entered below column $s = 6$, we are in the situation shown in (9).

Step I of Cycle k

(9)

Basic Variables	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	$-z$	Constants
x_1								-1	1	1		1
x_2										1		1
x_3							1	1	-1	-1		2
x_4								1		-1		1
$-z$	0	0	0	0	3	-4	7	5	1	4	1	-53
	●	●	●	●		★						

We now know that x_6 is the candidate for the new basic variable. To determine which basic variable to drop, the value of $\bar{a}_{i6}$ for each i is computed and entered into column 6 using the coefficients a_{16}, a_{26}, a_{36}, a_{46} appearing in column 6 of the original tableau for cycle 0 and the i^{th} row of the inverse of the basis. According to (4),

$$(10)\quad \bar{a}_{16} = \beta_{11}a_{16} + \beta_{12}a_{26} + \beta_{13}a_{36} + \beta_{14}a_{46} = 0 \cdot 0 - 1 \cdot 1 + 1 \cdot 0 + 1 \cdot 0 = -1$$

$$\bar{a}_{26} = \beta_{21}a_{16} + \beta_{22}a_{26} + \beta_{23}a_{36} + \beta_{24}a_{46} = 0 \cdot 0 + 0 \cdot 1 + 0 \cdot 0 + 1 \cdot 0 = 0$$

$$\bar{a}_{36} = \beta_{31}a_{16} + \beta_{32}a_{26} + \beta_{33}a_{36} + \beta_{34}a_{46} = 1 \cdot 0 + 1 \cdot 1 - 1 \cdot 0 - 1 \cdot 0 = 1$$

$$\bar{a}_{46} = \beta_{41}a_{16} + \beta_{42}a_{26} + \beta_{43}a_{36} + \beta_{44}a_{46} = 0 \cdot 0 + 1 \cdot 1 + 0 \cdot 0 - 1 \cdot 0 = 1$$

If the computed $\bar{a}_{i6}$ are entered in the tableau and the value of r determined, such that $\bar{b}_r/\bar{a}_{rs} = \text{Min } \bar{b}_i/\bar{a}_{is}$ for $\bar{a}_{is} > 0$, we are then in the situation shown in (11), where $r = 4$, so that $\bar{a}_{rs} = \bar{a}_{46}$ becomes the *pivot* position.

Step II of Cycle k

(11)

Basic Variables	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	$-z$	Constants
x_1						-1		-1	1	1		1
x_2										1		1
x_3						1	1	1	-1	-1		2
x_4						1		1		-1		1
$-z$	0	0	0	0	3	-4	7	5	1	4	1	-53
	●	●	●	●		★						

The next step of the standard simplex method would be to replace x_4 by x_6 as a basic variable by reducing (2) to canonical form by pivoting on $\bar{a}_{46}$. We do the same in (11), except that here we are restricted to the completed columns, namely those corresponding to the pivot, the inverse of the basis B^*, and the constants. After elimination of x_6 in (11) using $\bar{a}_{46}$ as pivot, the situation is as shown in (12). Omitting the computed relative costs of the last cycle and the coefficients of x_6, we are ready to start cycle $k + 1$.

Start of Cycle $k + 1$

(12)

Basic Variables	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	$-z$	Constants
x_1	Omit all entries in the shaded portion					0			1			2
x_2						0				1		1
x_3						0	1		-1			1
x_6						1		1		-1		1
$-z$						0	7	9	1	0	1	-49
	●	●	●			●						

New Inverse of the Basis

New Multipliers $\pi = (-7, -9, -1, 0)$

Computational Remarks.

In the standard simplex method, each cycle requires the recording of at least $(m + 1)(n + 1)$ entries (or more if there are artificial variables). Here, however, by use of *cumulative multiplications*,[1] the amount of recorded information is reduced to $(m + 1)(m + 2)$ entries, actually $(m + 1)^2$ if we ignore the $(-z)$ column.

To illustrate, the values of $\{-\pi_1, -\pi_2, -\pi_3, -\pi_4, 1\}$ can be placed vertically on a strip of paper and moved alongside the j^{th} column as in (13a). It is now convenient to compute $\bar{c}_j$ by multiplying the corresponding entries and forming the cumulative sum.

DEFINITION: The operation (13a) of multiplying the simplex multipliers

[1] Desk calculators and electronic computers have special double-length registers that permit convenient forming of the cumulative products

$$a_1b_1,\ a_1b_1 + a_2b_2,\ (a_1b_1 + a_2b_2) + a_3b_3, \ldots$$

to double the number of places of a typical memory register of the machines.

of the basis by the vector of coefficients of x_j to determine its *coefficient* in the modified objective form is called *pricing out*[2] the j^{th} activity in terms of the basic set of activities.

(13a)
$$\begin{array}{|c|l} -\pi_1 & a_{1j} \\ -\pi_2 & a_{2j} \\ -\pi_3 & a_{3j} \\ -\pi_4 & a_{4j} \\ 1 & c_j \end{array}$$

(13b)
$$\begin{array}{cccc} \hline a_{16} & a_{26} & a_{36} & a_{46} \\ \hline \\ \beta_{k1} & \beta_{k2} & \beta_{k3} & \beta_{k4} \end{array}$$

Similarly, the values of a_{16}, a_{26}, a_{36}, a_{46} appearing in column $s = 6$ can be placed horizontally on a piece of paper and moved alongside the k^{th} row of the inverse of the basis B. It is now convenient to compute $\bar{a}_{k6}$ by multiplying the corresponding entries and forming their cumulative sum as in (13b).

DEFINITION: The operation of multiplication of the rows of the inverse of the basis by the vector of coefficients of x_j is called *representing the j^{th} activity in terms of the basic set of activities*. (See § 8-2-(21) and following discussion.)

Less machine memory is needed using the multiplier method for recording because the original coefficients are often given in fixed decimal of three to five places. This is considerably less than that required for $\bar{a}_{ij}$, when the standard simplex method is used, for this avoids round-off error difficulties in the passage from iteration to iteration. Moreover, using the multiplier method, it is convenient to cumulate the full products without round-off in the machine for both the pricing and representation operations and then to round the resulting sum. This results usually in considerably less round-off error than with the standard method which must round *each* product before recording.

In order to reduce recording still further, most electronic computer instruction codes compute successive $\bar{c}_j$ values by (13a), but keep a record only of the value and location of the *smallest* $\bar{c}_j$ attained up to that point in the calculations.

Starting with an $m \times n$ system in a feasible canonical form, the total number of multiplications required per iteration is

$$(14)\quad t(n - m)(m + 1) + tm(m + 1) + (m + 1)^2 = tn(m + 1) + (m + 1)^2$$

where the fraction of non-zero coefficients in the original tableau and in the column entering in the basis are assumed on the average to be both equal to t. The three terms on the left are the number of multiplications (or

[2] The reason for this term is that the simplex multipliers can be interpreted as prices (see Chapter 12); multiplying these prices by the input-output coefficients of an activity and summing evaluates or "prices out" the activity in terms of the substitute processes, the basic activities. The sum when compared with the direct cost c_j tells us whether or not it pays to consider introducing a non-basic activity j into the basic set.

divisions) used (a) in "pricing out," (b) in representing the new column, and (c) in pivoting. On the other hand the standard simplex procedure requires

$$\{(n - m) + 1\}(m + 1) = (n - 2m)(m + 1) + (m + 1)^2 \qquad (15)$$

operations on each cycle. Therefore, if the fraction of non-zeros,

$$t < 1 - 2m/n \qquad (16)$$

the simplex method using multipliers will require less effort. For example, if $n > 3m$, the fraction of non-zero coefficients required is $t < \frac{1}{3}$. Fig. 9-1-I

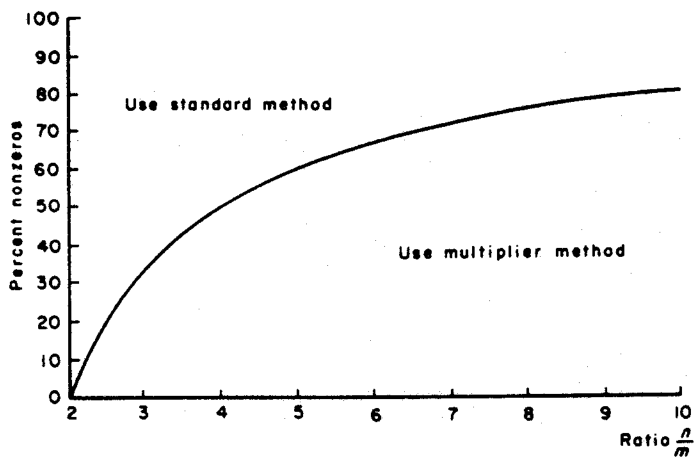


Figure 9-1-I. Condition for choosing the multiplier method over the standard method (starting in canonical form with no artificial variables).

can be used to decide whether to use the standard simplex or the multiplier method.

9-2. THE GENERAL METHOD USING MULTIPLIERS

Consider the system of equations in *canonical form* for Phase I of the simplex method as in § 5-2-(7), except that we use here m, n for M, N.

Cycle 0

(1)

Admissible Variables	Artificial Variables	
$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n$	$+ x_{n+1}$	$= b_1$
$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n$	$+ x_{n+2}$	$= b_2$
$\vdots$	$\ddots$	$\vdots$
$a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n$	$+ x_{n+m}$	$= b_m$
$c_1x_1 + c_2x_2 + \cdots + c_nx_n$	$-z$	$= 0$
$d_1x_1 + d_2x_2 + \cdots + d_nx_n$	$-w$	$= -w_0$

where $b_1, b_2, \ldots, b_m$ are made nonnegative by changing, if necessary, the signs of all terms in the original equations *prior to augmentation with artificial variables*, and where

(2) $$d_j = -\sum_{i=1}^{m} a_{ij}; \quad w_0 = \sum_{i=1}^{m} b_i$$

Thus, the sum of the first m equations when added to the w-form, implies

(3) $$x_{n+1} + x_{n+2} + \ldots + x_{n+m} - w = 0$$

The problem is to find w, z, and nonnegative x_j satisfying (1), such that $w = 0$ and z is a minimum. Tableau (4) is the canonical form with basic variables $x_{j_1}, x_{j_2}, \ldots, x_{j_m}, -z, -w$ for the regular simplex method for some cycle k. The basic feasible solution is obtained by setting $x_{j_1} = \bar{b}_1, \ldots, x_{j_m} = \bar{b}_m$; $z = \bar{z}_0$; $w = \bar{w}_0$; and $x_j = 0$ otherwise.

At the start of any cycle, using the multiplier method, the only recorded information from the tableau of the regular simplex method consists of the coefficients of the artificial variables, the constant terms, and the names of their corresponding basic variables. During the cycle, part of the missing data in the simplex tableau is generated as required; these are the values of $\bar{c}_j$ or $\bar{d}_j$ for $j = 1, 2, \ldots, n$ and the values in column $j = s$. The purpose of this section is to review, in general, how parts of the simplex tableau for cycle k can be generated *directly* from cycle 0 using the *inverse* of the basis. In the next section, we shall make use of this to give the computational rules of the simplex method using multipliers.

Tableau of Regular Simplex Method—Cycle k

(4)

Basic Variables	Admissible Variables					Artificial Variables			$-z$	$-w$	Constants
	x_1	. . .	x_s	. . .	x_n	x_{n+1}	. . .	x_{n+m}			
x_{j_1}	$\bar{a}_{11}$	. . .	$\bar{a}_{1s}$	. . .	$\bar{a}_{1n}$	$\bar{a}_{1,n+1}$	. . .	$\bar{a}_{1,n+m}$			$\bar{b}_1$
.	.		.		.	.		.			.
x_{j_r}	$\bar{a}_{r1}$	. . .	$\bar{a}_{rs}$	. . .	$\bar{a}_{rn}$	$\bar{a}_{r,n+1}$	. . .	$\bar{a}_{r,n+m}$			$\bar{b}_r$
.	.		.		.	.		.			.
x_{j_m}	$\bar{a}_{m1}$	. . .	$\bar{a}_{ms}$	. . .	$\bar{a}_{mn}$	$\bar{a}_{m,n+1}$	. . .	$\bar{a}_{m,n+m}$			$\bar{b}_m$
$-z$	$\bar{c}_1$	. . .	$\bar{c}_s$	. . .	$\bar{c}_n$	$\bar{c}_{n+1}$	. . .	$\bar{c}_{n+m}$	1		$-\bar{z}_0$
$-w$	$\bar{d}_1$	. . .	$\bar{d}_s$	. . .	$\bar{d}_n$	$\bar{d}_{n+1}$	. . .	$\bar{d}_{n+m}$		1	$-\bar{w}_0$

$B^{-1} = [\bar{a}_{i,n+j}]$, $-\pi = [\bar{c}_{n+1}, \ldots, \bar{c}_{n+m}]$
$-\sigma = [\bar{d}_{n+1}, \ldots, \bar{d}_{n+m}]$

Since the first m equations of (1) are in canonical form with respect to

$x_{n+1}, x_{n+2}, \ldots, x_{n+m}$ and the equivalent system (4) is in canonical form with respect to $x_{j_1}, x_{j_2}, \ldots, x_{j_m}$, it follows from (9) and (10) of § 8-4:

(a) if the basis, B, is the set of coefficients $x_{j_1}, x_{j_2}, \ldots, x_{j_m}$ in (1), then its inverse, B^{-1}, is the set of coefficients $x_{n+1}, x_{n+2}, \ldots, x_{n+m}$ in (4),

excluding the z- and w-equations. Moreover, since the entire system (1) is in canonical form with respect to $x_{n+1}, x_{n+2}, \ldots, x_{n+m}, -z, -w$ and the entire system (4) is in canonical form with respect to $x_{j_1}, x_{j_2}, \ldots, x_{j_m}, -z, -w$, it also follows that:

(b) if the basis, B^*, is the set of coefficients $x_{j_1}, x_{j_2}, \ldots, x_{j_m}, -z, -w$ in (1), then its inverse, $[B^*]^{-1}$, is the set of coefficients $x_{n+1}, x_{n+2}, \ldots, x_{n+m}, -z, -w$ in (4).

According to § 8-4-(20), an element in a given row and column of (4) can be generated from (1) by forming the scalar product of the corresponding row in the inverse and the corresponding column of (1). Thus $\bar{a}_{ij}$ can be generated for, say, $j = s$ by forming the scalar product of the i^{th} row of the inverse B by the j^{th} column of (1) excluding the z- and w-equations, i.e.,

(5) $$\bar{a}_{ij} = \beta_{i1}a_{1j} + \beta_{i2}a_{2j} + \cdots + \beta_{im}a_{mj}$$

where we have designated the elements of B^{-1} by β_{ij} and have shown by (a) and (b) above

(6) $$\beta_{ik} = \bar{a}_{i,n+k} \qquad (i, k = 1, 2, \ldots, m)$$

Similarly, $\bar{c}_j$ or $\bar{d}_j$ can be generated by the scalar product of the z- or w-row of the inverse $[B^*]^{-1}$ with the j^{th} column of (1) including the z- and w-rows. Upon rearrangement of terms

(7) $$\bar{c}_j = c_j - (\pi_1 a_{1j} + \pi_2 a_{2j} + \cdots + \pi_m a_{mj})$$

(8) $$\bar{d}_j = d_j - (\sigma_1 a_{1j} + \sigma_2 a_{2j} + \cdots + \sigma_m a_{mj})$$

where we have designated by $-(\pi_1, \pi_2, \ldots, \pi_m)$ and $-(\sigma_1, \sigma_2, \ldots, \sigma_m)$, the coefficients of the artificial variables in the z- and w-equations of (4); thus

(9) $$\pi_k = -\bar{c}_{n+k}, \; \sigma_k = -\bar{d}_{n+k} \qquad (k = 1, 2, \ldots, m)$$

Finally, the constants $\bar{b}_i, \bar{z}_0, \bar{w}_0$ can be generated by forming the scalar product of the corresponding row of the inverse with the constant column of (1):

(10) $$\begin{aligned} \bar{b}_i &= \beta_{i1}b_1 + \beta_{i2}b_2 + \cdots + \beta_{im}b_m \\ \bar{z}_0 &= \pi_1 b_1 + \pi_2 b_2 + \cdots + \pi_m b_m \\ \bar{w}_0 &= \sigma_1 b_1 + \sigma_2 b_2 + \cdots + \sigma_m b_m + w_0 \end{aligned}$$

Since the pivoting process of the multiplication method generates new values for the basic variables, formulas (10) are not used, except for *check* purposes.

DEFINITION: Multipliers $\pi = (\pi_1, \pi_2, \ldots, \pi_m)$ are called "simplex multipliers" *relative to the z-equation*, if multiplying the first equation of (1) by π_1, the second equation by $\pi_2, \ldots$, the m^{th} equation by π_m and subtracting their sum from the z-equation, eliminates the basic variables. A set of simplex multipliers *relative to the w-equation* is denoted $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_m)$.

THEOREM 1: *The simplex multipliers are unique, and are equal to the coefficients of the artificial variables of the z- and w-equation of the canonical form* (4).

PROOF: Using the particular values of $\pi_k = -\bar{c}_{n+k}$, $\sigma_k = -\bar{d}_{n+k}$ for $(k = 1, \ldots, m)$, the coefficients $\bar{c}_j$ and $\bar{d}_j$ of x_j, as given by the right-hand side of (7) and (8), vanish for columns j corresponding to basic variables. Hence these values satisfy the definition of simplex multipliers.

To show uniqueness, suppose for convenience that $x_1, x_2, \ldots, x_m$ are the basic variables; then π_i must be chosen so that

$$(11)\qquad \begin{aligned} \pi_1 a_{11} + \pi_2 a_{21} + \cdots + \pi_m a_{m1} &= c_1 \\ \pi_1 a_{12} + \pi_2 a_{22} + \cdots + \pi_m a_{m2} &= c_2 \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\ \pi_1 a_{1m} + \pi_2 a_{2m} + \cdots + \pi_m a_{mm} &= c_m \end{aligned}$$

This system of m equations in m unknowns should be contrasted with the system of m equations in m unknowns that the basic variables must satisfy

$$(12)\qquad \begin{aligned} x_1 a_{11} + x_2 a_{12} + \cdots + x_m a_{1m} &= b_1 \\ x_1 a_{21} + x_2 a_{22} + \cdots + x_m a_{2m} &= b_2 \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\ x_1 a_{m1} + x_2 a_{m2} + \cdots + x_m a_{mm} &= b_m \end{aligned}$$

System (11) interchanges rows and columns in (12) and replaces the constants by cost coefficients of the basic variables. The coefficients of x_j in (12) form the basis B (in this case) and hence the coefficients of π_i in (11) form the *transpose of the basis*. By § 8-4, Theorem 7, the inverse of the transpose of the basis exists and is the transpose of B^{-1}. This implies that $\pi_1, \pi_2, \ldots, \pi_m$ can be expressed uniquely in terms of $c_1, c_2, \ldots, c_m$ by

$$(13)\qquad \begin{aligned} \pi_1 &= \beta_{11} c_1 + \beta_{21} c_2 + \cdots + \beta_{m1} c_m \\ \pi_2 &= \beta_{12} c_1 + \beta_{22} c_2 + \cdots + \beta_{m2} c_m \\ \cdots\cdots&\cdots\cdots\cdots\cdots\cdots\cdots \\ \pi_m &= \beta_{1m} c_1 + \beta_{2m} c_2 + \cdots + \beta_{mm} c_m \end{aligned}$$

where $x_1, x_2, \ldots, x_m$ are basic variables or, more generally, if $x_{j_1}, \ldots, x_j$

are the basic variables, by replacing c_i above by the cost coefficient of the i^{th} basic variable $c_{j_i} = \gamma_i$ (see § 8-5-(17)).

9-3. COMPUTATIONAL RULES USING MULTIPLIERS

Preliminary Remarks.

Write out the system of equations in canonical form for Phase I of the simplex method, as described in § 9-2-(1). The full system is written in detached coefficient form in Table 9-3-Ia at the end of this section.

The *tableau of the simplex method using multipliers*, Table 9-3-IIb, changes from cycle to cycle. [The entries for the starting cycle 0 are shown in Table 9-3-IIa.] Its entries, excluding the last column, are the coefficients of the artificial variables $x_{n+1}, x_{n+2}, \ldots, x_{n+m}$, of $-z$, and $-w$, and the constant terms of the tableau of the regular simplex method for that cycle. It may also be interpreted as composed of the "Inverse of the Basis," $[\beta_{ij}]$, two rows for the negative of the "simplex multipliers," π_i and σ_i, a column for the values of the basic variables in the basic solution, and a column for a variable x_s. *At the start of a cycle, all entries in the tableau except the last column "x_s" are known.*

During some cycle, say cycle t, the values of the relative cost factors $\bar{d}_j$ for Phase I or $\bar{c}_j$ for Phase II are computed and entered in Table 9-3-Ib. Except at the end of Phase I or Phase II, this is followed by the computation of each $\bar{a}_{is}$, which are entered in the last column of Table 9-3-IIb. At the end of the cycle the entries of this tableau are used to compute the corresponding starting tableau of the next cycle, cycle $t + 1$. Table 9-3-IIc shows how the starting tableau of cycle $t + 1$ is related to the ending tableau of cycle t.

Computational Rules.

These apply to all cycles but differ slightly depending on whether the computations are in Phase I or in Phase II. They are the same as the standard method given in § 5-2 with the following modifications:

Step Ia: Use values of σ_i (if Phase I) or π_i (if Phase II) from Table 9-3-IIb to compute relative cost factors $\bar{d}_j$ (Phase I) or $\bar{c}_j$ (Phase II) for $j = 1, 2, \ldots, n; \ldots, n + m$ by (7), (8), (9) and record in Table 9-3-Ib in the row corresponding to the cycle.

Step Ib: Same as Step I of Standard Method.

Step IIa: Compute for $i = 1, 2, \ldots, m$, the coefficients $\bar{a}_{is}$ of x_s in the canonical form by (5) and (6) and record in the last column of tableau, Table 9-3-IIb.

Step IIb: Same as Step II of Standard Method. Instead of the *random rule* for resolving ties, one may use as an *alternative, the lexicographic rule* given below Step III.

Step III: Same as Step III of the Standard Method, except: Pivot using pivot element in $\bar{a}_{rs}$ in Table 9-3-IIb (instead of Table 5-2-II) and record entries in Table 9-3-IIc. Leave column "x_s" blank. Leave the list of basic variables in the left column the same as Table 9-3-IIb except change j_r to the value of s determined in Step I. *Return to Step Ia to initiate cycle* $t + 1$.

Lexicographic Rule for Resolving Degeneracy.

If two or more indices $r_1, r_2, \ldots$ are tied for the minimum, form the ratios of the corresponding entries in the *first column of the inverse* to $\bar{a}_{r_1s}, \bar{a}_{r_2s}, \ldots$, respectively:

$$\begin{gathered}\beta_{r_1,1}/\bar{a}_{r_1s}\\ \beta_{r_2,1}/\bar{a}_{r_2s}\\ \cdot\\ \cdot\\ \cdot\end{gathered}$$

and take the index of the row with the minimizing ratio for r. If there still remain ties, repeat for those indices that are still tied using instead the ratio of the corresponding entries in the *second column of the inverse* to their respective $\bar{a}_{r_1s}$. In this manner, ratios are formed from successive columns of the inverse until all ties are resolved uniquely (this always occurs on or before the last column is reached). A proof that the simplex algorithm always terminates in a finite number of steps using this rule will be the subject of the next chapter. (The proof that the random rule terminates in a finite number of steps with probability one was given at the end of § 6-1.)

TABLE 9-3-Ia

SIMPLEX METHOD USING MULTIPLIERS

Detached Coefficients, Original System

Equation i	Admissible Variables								Artificial Variables				$-z$	$-w$	Constants
	x_1	x_2	...	x_j	...	x_n			x_{n+1}	x_{n+2}	...	x_{n+m}			
1	a_{11}	a_{12}	...	a_{1j}	...	a_{1n}			1						b_1
2	a_{21}	a_{22}	...	a_{2j}	...	a_{2n}				1					b_2
.	.	.		.		.					.				.
.	.	.		.		.					.				.
.	.	.		.		.					.				.
m	a_{m1}	a_{m2}	...	a_{mj}	...	a_{mn}						1			b_m
z-form	c_1	c_2	...	c_j	...	c_n			0	0	...	0	1		0
w-form	d_1	d_2	...	d_j	...	d_n			0	0	...	0		1	$-\sum_1^m b_i$

$d_j = -\sum_{i=1}^m a_{ij};\ b_1 \geq 0, \ldots, b_m \geq 0$

TABLE 9-3-Ib

Relative Cost Factors d_j or $\bar{c}_j$

Cycle	Variable j: 1, 2, . . .	n,	$n + 1$	. . .	$n + m$
Phase I 0	d_1 d_2 . . .	d_n,	0	. . .	0
Phase I 1, ·, ·, ·, k	Using $\begin{Bmatrix} \sigma_i \text{ if in Phase I} \\ \text{or} \\ \pi_i \text{ if in Phase II} \end{Bmatrix}$ from Table 9-3-IIb, same cycle,				
Phase II k, $k + 1$, ·, ·, ·	*compute* $\begin{Bmatrix} d_j = d_j - [a_{1j}\sigma_1 + a_{2j}\sigma_2 + \dots + a_{mj}\sigma_m] \\ \text{or} \\ \bar{c}_j = c_j - [a_{1j}\pi_1 + a_{2j}\pi_2 + \dots + a_{mj}\pi_m] \end{Bmatrix}$				
	record d_j or $\bar{c}_j$ on row corresponding to cycle and choose pivot column s such that $\begin{Bmatrix} d_s = \text{Min } d_j \text{ (Phase I)} \\ \bar{c}_s = \text{Min } \bar{c}_j \text{ (Phase II)} \end{Bmatrix}$				

TABLE 9-3-IIa

SIMPLEX METHOD USING MULTIPLIERS

Tableau Start of Cycle 0

Basic Variables	(Columns of Canonical Form) x_{n+1}	. . .	x_{n+m}	$-z$	$-w$	Value of Basic Variable	x_s (*see note*)
	←—— Inverse of Basis ——→						
x_{n+1}	1					b_1	
·		·				·	
·		·				·	
·		·				·	
x_{n+r}		1				b_r	
·		·				·	
·		·				·	
·		·				·	
x_{n+m}			1			b_m	
$-z$	0	. . .	0	1		0	
$-w$	0	. . .	0		1	$-\sum_1^m b_i$	

Note: The x_s column is blank at start of cycle 0.

TABLE 9-3-IIb

SIMPLEX METHOD USING MULTIPLIERS

Tableau End of Some Cycle t

Basic Variables	(Columns of Canonical Form)					Value of Basic Variable	x_s (*see note*)
	x_{n+1}	. . .	x_{n+m}	$-z$	$-w$		
	$[\beta_{jk}] = [\bar{a}_{j,n+k}]$ ← Inverse of Basis →						Compute ↓
x_{j_1}	β_{11}	. . .	β_{1m}			$\bar{b}_1$	$\sum_1^m \beta_{1i} a_{is} = \bar{a}_{1s}$
.	.		.			.	.
.	.		.			.	.
.	.		.			.	.
x_{j_r}	β_{r1}	. . .	β_{rm}			$\bar{b}_r$	$\sum_1^m \beta_{ri} a_{is} = \boldsymbol{\bar{a}_{rs}}$
.	.		.			.	.
.	.		.			.	.
.	.		.			.	.
x_{j_m}	β_{m1}	. . .	β_{mm}			$\bar{b}_m$	$\sum_1^m \beta_{mi} a_{is} = \bar{a}_{ms}$
	$(-\pi_k = \bar{c}_{n+k})$						
$-z$	$-\pi_1$	. . .	$-\pi_m$	1		$-\bar{z}_0$	$c_s - \sum_1^m \pi_i a_{is} = \bar{c}_s$
	$(-\sigma_k = \bar{d}_{n+k})$						
$-w$	$-\sigma_1$	. . .	$-\sigma_m$		1	$-\bar{w}_0$	$d_s - \sum_1^m \sigma_i a_{is} = \bar{d}_s$

Note: Last column is blank at start of cycle; see Table 9-3-Ib for choice of s; see Step IIb for r; Table 9-3-IIc is obtained by pivoting on $\bar{a}_{rs}$ and omitting entries in last column. The bold-faced $\boldsymbol{\bar{a}_{rs}}$ indicates position of pivot.

TABLE 9-3-IIc

Tableau Start of Cycle $t + 1$

x_{j_1}	$\beta_{11} - \bar{a}_{1s}\beta^*_{r1}$	. . .	$\beta_{1m} - \bar{a}_{1s}\beta^*_{rm}$			$\bar{b}_1 - \bar{a}_{1s}\bar{b}^*_r$	(*see note above*)
.	.		.			.	
.	.		.			.	
.	.		.			.	
x_s	β^*_{r1}	. . .	β^*_{rm}			b^*	
.	.		.			.	
.	.		.			.	
.	.		.			.	
x_{j_m}	$\beta_{m1} - \bar{a}_{ms}\beta^*_{r1}$	. . .	$\beta_{mm} - \bar{a}_{ms}\beta^*_{rm}$			$\bar{b}_m - \bar{a}_{ms}\bar{b}^*_r$	
$-z$	$-\pi_1 - \bar{c}_s\beta^*_{r1}$	. . .	$-\pi_m - \bar{c}_s\beta^*_{rm}$	1		$-\bar{z}_0 - \bar{c}_s\bar{b}^*_r$	
$-w$	$-\sigma_1 - \bar{d}_s\beta^*_{r1}$	. . .	$-\sigma_m - \bar{d}_s\beta^*_{rm}$		1	$-\bar{w}_0 - \bar{d}_s\bar{b}^*_r$	

$\beta^*_{r1} = \beta_{ri}/\bar{a}_{rs}\ (i = 1, 2, \ldots, m);\ \bar{b}^*_r = \bar{b}_r/\bar{a}_{rs}$

EXAMPLE: To illustrate the computational procedures of the simplex method using multipliers, let us return again to the example of § 5-2, Table 5-2-V. In tableau form, the problem is given by Table 9-3-IIIa.

TABLE 9-3-IIIa

Detached Coefficients of Original System

x_1	x_2	x_3	x_4	x_5	x_6	x_7	$-z$	$-w$	Constants
5	−4	13	−2	1	1				20
1	−1	5	−1	1		1			8
1	6	−7	1	5			1		0
−6	5	−18	3	−2				1	−28

TABLE 9-3-IIIb

Cycle		Relative Cost Factors (1)	(2)	(3)	(4)	(5)	(6)	(7)	
d_j	0	−6	5	−18 ★	3	−2	0 ●	0 ○	
	1	12/13	−7/13	0 ●	3/13	−8/13 ★	18/13 ○	0	
	2	0	0	0 ●	0	0 ●	1	1	End of Phase I
$\bar{c}_j$	2	12	−1 ★	0 ●	2	0 ○	Drop		
	3	72/7	0 ●	0 ●	11/7	8/7			End of Phase II

TABLE 9-3-IV

Cycle 0

Basic Variables	Columns of Canonical Form x_6	x_7	$-z$	$-w$	Constants	$x_s = x_3$
x_6	1				20	**13**
x_7		1			8	5
$-z$			1		0	−7
$-w$				1	−28	−18

Cycle 1 $(x_s = x_5)$

x_3	1/13				20/13	1/13
x_7	−5/13	1			4/13	**8/13**
$-z$	7/13	0	1		140/13	72/13
$-w$	18/13	0		1	−4/13	−8/13

TABLE 9-3-IV (continued)

Cycle 2 $(x_s = x_2)$

x_3	1/8	−1/8			3/2	−3/8	End of Phase I, $d_j \geq 0$, $w = 0$
x_5	−5/8	13/8			1/2	7/8	
$-z$	+4	−9	1		8	−1	Drop x_6 and x_7 since $(d_6, d_7) = (1, 1) > 0$; drop w-row
$-w$	1	1		1	0	0	

Cycle 3

x_3	−1/7	4/7			12/7		End of Phase II, $\bar{c}_j \geq 0$.
x_2	−5/7	13/7			4/7		
$-z$	23/7	−50/7	1		60/7		
$-w$	(w-row dropped)						

9-4. PROBLEMS

1. Solve by the simplex method using multipliers:

$$\begin{aligned} 3x_1 - 3x_2 + 4x_3 + 2x_4 - x_5 + x_6 \quad &= 0 \\ x_1 + x_2 + x_3 + 3x_4 + x_5 \quad + x_7 &= 2 \\ 2x_1 + 3x_2 + 2x_3 - x_4 + x_5 \quad &= z \\ x_6 + x_7 &= w \end{aligned}$$

and minimize z, where $x_j \geq 0$ and $w = 0$.

2. Solve, using the simplex method using multipliers:

$$\begin{aligned} x_1 \quad + 2x_3 - x_4 \quad &\geq 3 \qquad (x \geq 0) \\ 2x_1 + x_2 \quad + 2x_4 - 2x_5 &\leq 1 \\ + x_2 - x_3 \quad &\geq 0 \\ -x_1 \quad + 3x_5 &\leq 2 \\ -x_1 - x_2 - x_3 \quad + x_5 &= \text{Min } z \end{aligned}$$

3. Discuss the relationships between the regular simplex method and the revised simplex method.

4. Set up the dual of the problem of finding x_j that minimizes $x_1 + x_2 = z$ subject to

$$\begin{aligned} x_1 + 2x_2 &\geq 3 \\ x_1 - 2x_2 &\geq -4 \\ x_1 + 7x_2 &\leq 6 \end{aligned}$$

where $x_1 \geq 0$, and x_2 is unrestricted in sign. Determine the simplex multipliers of the optimum solution of the primal, and verify that it satisfies the dual and gives the same value for the objective form.

5. Solve the Blending Problem II, § 3-4, by the revised simplex method.

6. Solve Waugh's problem using simplex multipliers. (See Problem 28, Chapter 5.)
7. Prove that

$$c_1\bar{a}_{1j} + c_2\bar{a}_{2j} + \ldots + c_m\bar{a}_{mj} = \pi_1 a_{1j} + \pi_2 a_{2j} + \ldots + \pi_m a_{mj}$$

where $\pi_1, \pi_2, \ldots, \pi_m$ are simplex multipliers associated with the basic set of variables $x_1, x_2, \ldots, x_m$. See Chapter 12 for an economic interpretation of this relation.
8. Prove that if P_s replaces P_{j_r}, the r^{th} column in a basis, and if π^* is the new vector of simplex multipliers, then

$$\pi^* = \pi + k\beta_r, \qquad k = \bar{c}_s/\bar{a}_{rs}$$

where β_r is the row of B^{-1} corresponding to P_{j_r}.
9. In § 8-5, the product form of the inverse was developed. Review this discussion and rework the exercises. Discuss how you would compute the simplex prices, π^*, if the inverse of the basis in product form were given.

REFERENCES

Dantzig and Orchard-Hays, 1953-1
Dantzig, Orden, and Wolfe, 1954-1
Garvin, 1960-1
Gass, 1958-1
Hadley, 1961-2
Orchard-Hays, 1955-1, 1956-1
Orden, 1952-1, 1955-1

CHAPTER 10

FINITENESS OF THE SIMPLEX METHOD UNDER PERTURBATION

10-1. THE POSSIBILITY OF CIRCLING IN THE SIMPLEX ALGORITHM

We have seen that if degeneracy does occur, then it is possible to have a sequence of iterations with no decrease in the value of z. Under such circumstances, may it not happen that a basic set will be repeated, thereby initiating an endless circle of such repetitions? If so, can we devise an efficient procedure to prevent such a circling possibility? In the early days of linear programming this was an unsolved problem.

In 1951, A. J. Hoffman constructed an example, shown in Table 10-1-I, involving three equations and eleven variables. He showed that if one *resolved* the ambiguity of choice regarding which variable to drop from the basic set *by selecting the first among them*, then the tableau at cycle 9 would by the same as at cycle 0. It follows in this case that the same basic set would be repeated every nine iterations and the simplex method would never terminate. This phenomenon is usually referred to as *cycling in the simplex algorithm*. We prefer, however, the term "circling," because we use the term "cycle" for a single iteration of the simplex algorithm.

Later, E. M. L. Beale [1955-1] constructed a second example, a version of which is shown in Table 10-1-II, that is remarkable for its simplicity. It also has three equations but only seven variables. Using the same rule for resolving a tie, the tableau at cycle 6 is the same at cycle 0. It is conjectured that this is the simplest example; to be precise, it is believed that no other example of circling can be constructed involving fewer variables regardless of the number of equations.

Since circling in the simplex algorithm is only possible under degeneracy, it is pertinent to ask how degeneracy can occur, how frequently it is encountered in practice and how often it implies circling. Degenerate solutions are possible only when the constants, b_i, of the original right-hand side bear a special relation to the coefficients of the basic variables. This is clear since the process of reduction to one of the finite set of canonical forms depends only on the coefficients and *not* on the right-hand side; the final values, $\bar{b}_i$, are weighted sums of the original b_i's where the weights depend only on the coefficients. If all the b_i's were selected at random, it would be something of a miracle if one or more of the constants $\bar{b}_i$ of the canonical system should vanish.

TABLE 10-1-I

A. J. Hoffman's Example of Circling in the Simplex Algorithm

$[\theta = 2\pi/5,\ w > (1 - \cos\theta)/(1 - 2\cos\theta)]$

(Cycle 0)

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	Constant
1											1
	1		$\boxed{\cos\theta}$	$-w\cos\theta$	$\cos 2\theta$	$-2w\cos^2\theta$	$\cos 2\theta$	$2w\cos^2\theta$	$\cos\theta$	$w\cos\theta$	0
		1	$\sin\theta\tan\theta/w$	$\cos\theta$	$\tan\theta\sin 2\theta/w$	$\cos 2\theta$	$-2\sin^2\theta/w$	$\cos 2\theta$	$-\tan\theta\sin\theta/w$	$\cos\theta$	0
			$-(1-\cos\theta)/\cos\theta$	$+w$		$+2w$	$+4\sin^2\theta$	$-2w\cos 2\theta$	$+4\sin^2\theta$	$w(1-2\cos\theta)$	z (Min)
●	○	●	★								

(Cycle 1)

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	Constant
1											1
	$\sec\theta$		1	$-w$	$4\cos^2\theta - 3$	$-2w\cos\theta$	$4\cos^2\theta - 3$	$2w\cos\theta$	1	w	0
	$-\tan^2\theta/w$	1		$\boxed{\sec\theta}$	$\tan^2\theta/w$	1	$2\sin\theta\tan\theta/w$	$4\cos^2\theta - 3$	$-2\sin\theta\tan\theta/w$	$4\cos^2\theta - 3$	0
	$(1-\cos\theta)/\cos^2\theta$			$w(2\cos\theta - 1)/\cos\theta$	$-2\sin\theta\tan\theta$	$2w\cos\theta$	$(\cos\theta - 1)/\cos\theta$	$3w$	$2\sin\theta\tan\theta$	$-w(4\cos^2\theta - 3)$	z
●		○	●	★							

(Cycle 2)

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	Constant
1											1
	$\cos\theta$	$w\cos\theta$	1		$\boxed{\cos\theta}$	$-w\cos\theta$	$\cos 2\theta$	$-2w\cos^2\theta$	$\cos 2\theta$	$2w\cos^2\theta$	0
	$-\tan\theta\sin\theta/w$	$\cos\theta$		1	$\sin\theta\tan\theta/w$	$\cos\theta$	$\tan\theta\sin 2\theta/w$	$\cos 2\theta$	$-2\sin^2\theta/w$	$\cos 2\theta$	0
	$4\sin^2\theta$	$w(1-2\cos\theta)$			$-(1-\cos\theta)/\cos\theta$	$+w$		$2w$	$4\sin^2\theta$	$-2w\cos 2\theta$	z
●			○	●	★						

Notice that columns (2, 3, 4, . . ., 11) of cycle 0 are the same as columns (4, 5, 6, . . ., 11; 2, 3) respectively of cycle 2; hence, 8 more iterations will repeat cycle 0.
The □ indicates the position of the pivot.

TABLE 10-1-II

BEALE'S EXAMPLE OF CIRCLING IN THE SIMPLEX ALGORITHM

x_1	x_2	x_3	x_4	x_5	x_6	x_7	$-z$	Constant
				(Cycle 0)				
1/4	−60	−1/25	9	1				0
1/2	−90	−1/50	3		1			0
		1				1		1
−3/4 ★	150	−1/50	6	●	●	●	1	0
				(Cycle 1)				
1	−240	−4/25	36	4				0
	30	3/50	−15	−2	1			0
		1				1		1
●	−30 ★	−7/50	33	3	●	●	1	0
				(Cycle 2)				
1		**8/25**	−84	−12	8			0
	1	1/500	−1/2	−1/15	1/30			0
		1				1		1
●	●	−2/25 ★	18	1	1	0 ●	1	0
				(Cycle 3)				
25/8		1	−525/2	−75/2	25			0
−1/160	1		**1/40**	1/120	−1/60			0
−25/8			525/2	75/2	−25	1		1
1/4	●	●	−3 ★	−2	3	●	1	0
				(Cycle 4)				
−125/2	10,500	1		**50**	−150			0
−1/4	40		1	1/3	−2/3			0
125/2	−10,500			−50	150	1		1
−1/2	120	●	●	−1 ★	1	●	1	0
				(Cycle 5)				
−5/4	210	1/50		1	−3			0
1/6	−30	−1/150	1		**1/3**			0
		1				1		1
−7/4	330	1/50	●	●	−2 ★	●	1	0

Note: Cycle 6 must be the same as Cycle 0, as it has the same basic variables in the same order.

Nevertheless, it is common experience, based on the solutions of thousands of practical linear programming problems by the simplex method, that nearly every problem at some stage of the process is degenerate. It might be thought that, since degeneracy happens all the time, there would be many observed cases of circling. However, to date, there has not been one single case of circling, except in the specially constructed examples of Hoffman and Beale. Apparently, circling is a very rare phenomenon in practice. For this reason, most instruction codes for electronic computers use no special device for perturbing the problem to avoid degeneracy and the possibility of circling. The cells of the computer's high-speed memory, when not entirely reserved for the data of a large problem, are occupied by subroutines designed to increase accuracy by means of arithmetical checks and multiple precision arithmetic.

From a mathematical point of view, the phenomenon of circling is an interesting one. Long before Hoffman discovered his example, simple devices were proposed to avoid degeneracy. The main problem was to devise a way of avoiding degeneracy that involved *as little extra work as possible*. The first proposal along these lines was presented by the author in the fall of 1950 in his Linear Programming Course at the Graduate School of the U.S. Department of Agriculture. Students were assigned exercises involving the proofs of the method along the lines given in this section [Edmondson, 1951-1; Dantzig, 1951-2]. Later A. Orden, P. Wolfe, and the author published a proof of this method based on the concept of *lexicographic ordering* of vectors [Dantzig, Orden, and Wolfe, 1954-1]. A. Charnes [1952-1] independently developed a technique of perturbation that is described in one of the problems at the end of the chapter.

As an alternative to the random choice rule established in § 6-1, we shall show in the next section that it is possible to perturb slightly the constant terms in such a way that

(a) *the basic feasible solutions become nondegenerate, and*
(b) *moreover, the corresponding basic solutions for the unperturbed problem will remain feasible.*

In effect, the perturbation simply *guides* the proper choice of variables to drop from the basic set in case of ties.

10-2. PERTURBING CONSTANTS TO AVOID DEGENERACY[1]

Let us begin with Phase I of the simplex method and assume, as in § 9-2, that all $b_i \geq 0$ and that a set of variables has been augmented by the

[1] The method given in this section is based on perturbing the constant terms. In [Dantzig, Orden, and Wolfe, 1954-1] an alternative proof, based on *lexicographic ordering*, closely parallels the one given here. For further discussion, see Problems 11, 12, and 13 at the end of this chapter and § 13-4.

artificial variables $x_{n+1}, x_{n+2}, \ldots, x_{n+m}$, so that the basic problem of Phase I is to find $x_j \geq 0$ and Min w, such that

$$\text{(1)} \qquad \sum_{j=1}^{n} a_{ij}x_j + x_{n+i} = b_i \quad (b_i \geq 0; i = 1, 2, \ldots, m)$$
$$\sum_{i=1}^{m} x_{n+i} = w$$

Let the initial basic set of variables be $x_{n+1}, x_{n+2}, \ldots, x_{n+m}$. If one or more of the constants, b_i, equal zero, the corresponding solution will be degenerate. We shall avoid this by considering the *perturbed problem*:

$$\text{(2)} \qquad \sum_{j=1}^{n} a_{ij}x_j + x_{n+i} = b_i + \varepsilon^i \quad (b_i \geq 0; \varepsilon > 0\, i; = 1, 2, \ldots, m)$$
$$\sum_{i=1}^{m} x_{n+i} = w \text{ (Min)}$$

It is obvious that the initial basic solution is nondegenerate, because for all i

$$\text{(3)} \qquad x_{n+i} = b_i + \varepsilon^i > 0 \qquad (b_i \geq 0; \varepsilon > 0)$$

and that, by setting $\varepsilon = 0$, the basic solution is feasible for the *unperturbed* problem.

On subsequent iterations, the values of basic variables will become general polynomial expressions in ε. Indeed, suppose for cycle t that $(x_{j_1}, x_{j_2}, \ldots, x_{j_m})$ is some basic set of variables, then by § 8-4-(21), the values of the basic variables which we denote by $\bar{b}_i(\varepsilon)$ are

$$\text{(4)} \qquad \bar{b}_i(\varepsilon) = \sum_{k=1}^{m} \beta_{ik}(b_k + \varepsilon^k) \qquad (i = 1, 2, \ldots, m)$$
$$= \bar{b}_i + \beta_{i1}\varepsilon + \beta_{i2}\varepsilon^2 + \cdots + \beta_{im}\varepsilon^m$$

where $[\beta_{ij}]$ is the inverse of the basis and $\bar{b}_i$ are the values of x_{j_i} for $\varepsilon = 0$.

EXERCISE: Show for each i there exists a $\beta_{ik} \neq 0$.

EXERCISE: Show that it is not possible for two rows of the inverse of a matrix to be proportional.

In (4) it is no longer possible to guarantee that the values of the basic variables will remain positive for *all* positive ε and nonnegative for $\varepsilon = 0$. However, we shall prove the following:

LEMMA 1: *Given a polynomial*

$$\text{(5)} \qquad f(\varepsilon) = a_0 + a_1\varepsilon + \cdots + a_m\varepsilon^m$$

then $f(\varepsilon) > 0$ *for all* $0 < \varepsilon <$ *some* h_0, *if and only if it has a non-zero term and the first such has a positive coefficient.*

PROOF: Let the first term of $f(\varepsilon)$ with non-zero coefficient (called the *leading term*) be k; then by assumption

$$a_0 = a_1 = \cdots = a_{k-1} = 0 \quad \text{and} \quad a_k > 0 \tag{6}$$

Let, for $j = k + 1, \ldots, m$,

$$M = \text{Max}\,(0, \underset{j}{\text{Max}} - a_j) \tag{7}$$

Then for $0 < \varepsilon < 1$ and $p \geq 1$,

$$a_j \varepsilon^p \geq -M\varepsilon^p \geq -M\varepsilon \qquad (j = k + 1, \ldots, m)$$

and, therefore,

$$\begin{aligned} f(\varepsilon) &= \varepsilon^k[a_k + a_{k+1}\varepsilon + \cdots + a_m\varepsilon^{m-k}] \\ &\geq \varepsilon^k[a_k - M\varepsilon - M\varepsilon - \cdots - M\varepsilon] \\ &\geq \varepsilon^k[a_k - M(m-k)\varepsilon] \end{aligned} \tag{8}$$

If we let

$$h_0 = \text{Min}\,[1, a_k/M(m-k)] \tag{9}$$

where $h_0 = 1$ if $M = 0$ or $M = k$, then it follows from (8) that $f(\varepsilon) > 0$ for all ε in the interval $0 < \varepsilon < h_0$.

EXERCISE: Prove the "only if" part of Lemma 1; see Problem 4.

LEMMA 2: *Given two polynomials $f(\varepsilon)$ and $g(\varepsilon)$, where*

$$f(\varepsilon) = \sum_{i=0}^{m} a_i\varepsilon^i, \qquad g(\varepsilon) = \sum_{i=0}^{m} b_i\varepsilon^i \tag{10}$$

such that

$$\begin{aligned} &a_i = b_i \quad \text{for } i = 1, 2, \ldots, k-1 \\ &a_k < b_k \\ &a_i, \quad b_i \quad \text{arbitrary for } i \neq k \end{aligned} \tag{11}$$

then for some $h_0 > 0$, $f(\varepsilon) < g(\varepsilon)$ for all $0 < \varepsilon < h_0$.

PROOF: This will follow from Lemma 1, since conditions (11) are a restatement of conditions (6) for $g(\varepsilon) - f(\varepsilon)$. Hence, there must exist an h_0, such that

$$g(\varepsilon) - f(\varepsilon) = \sum_{i=0}^{m} (b_i - a_i)\varepsilon^i > 0 \qquad \text{for all } 0 < \varepsilon < h_0 \tag{12}$$

THEOREM 1: *For cycle t, each polynomial expression in ε in (4) has at least one non-zero term; if the first such is positive for every i, then there is some range of values $0 < \varepsilon < h_t$, such that for any fixed ε in the range, the values of all basic variables are positive.*

The first part of the theorem holds because not all $\beta_{ij} = 0$ for fixed i. The second part follows from Lemma 1 and the fact that, if there are different ranges of values for ε within which the values of the various basic variables stay positive as a function of ε, then the smallest of these ranges will do for all x_{j_i}.

For our purposes, it is only important for cycle t that a range of values $0 < \varepsilon < h_t$ exists for which all basic variables remain positive as a function of ε. An explicit value for h_t is not needed, so that, *in computing work, h_t is never evaluated.*

THEOREM 2: *There exists a common range of values $0 < \varepsilon < h_t^*$ such that for any finite number of iterations of the simplex method as applied to any perturbed problem within the range, the values of all basic variables remain positive and the choice of the variable entering and leaving the basic set is unique and independent of the particular value of ε in the range.*

PROOF: For some cycle t let us apply the simplex algorithm to improve a basic solution for a perturbed problem (1), in which we assume for *inductive purposes*, that $\bar{b}_i(\varepsilon)$ will be positive for some range $0 < \varepsilon < h_t$ (hence, its leading term has a positive coefficient). Clearly the assumption is true for cycle 0 by (3). The choice of the *new variable* x_s entering the basic set depends only on the coefficients of the variables in the basic set and is *independent* of $\bar{b}_i$ and ε. On the other hand, the choice of the r^{th} basic variable *to be dropped is dependent on* $\bar{b}_i$ and ε. However, we shall now show that for the class of perturbed problems whose ε is within a sufficiently small range, the same variable x_r will be dropped from the basic set. In fact, by § 5-1-(21), the maximum value x_s^* of the variable x_s entering the basic set and the choice of the r^{th} basic variable to drop is determined through the relations

$$x_s^* = \frac{\bar{b}_r(\varepsilon)}{\bar{a}_{rs}} = \underset{\bar{a}_{is}>0}{\text{Min}} \{(\bar{b}_i + \beta_{i1}\varepsilon + \beta_{i2}\varepsilon^2 + \cdots + \beta_{im}\varepsilon^m)/\bar{a}_{is}\} \tag{13}$$

where x_s^* is positive for any ε in some range $0 < \varepsilon < h_t$ by the assumption that $\bar{b}_i(\varepsilon) > 0$ in this range.

The Lexicographic Rule.

Applying Lemma 2, the minimum of the several polynomial expressions (for sufficiently small range of values for ε) is found by first comparing their constant terms, i.e., by choosing r, such that

$$\frac{\bar{b}_r}{\bar{a}_{rs}} = \text{Min}\,\frac{\bar{b}_i}{\bar{a}_{is}} \qquad (\bar{a}_{rs} > 0;\ \bar{a}_{is} > 0) \tag{14}$$

If, however, there are several $i = r_1, r_2, \ldots$ satisfying (14), then for these i, choose $i = r$, such that

$$\frac{\beta_{r1}}{\bar{a}_{rs}} = \text{Min}\,\frac{\beta_{i1}}{\bar{a}_{is}} \qquad \text{for } i = r_1, r_2, \ldots \tag{15}$$

That is to say, the coefficients of the ε^1 power terms of these various polynomials are compared for those i that are tied in (15). If again r is not unique, then for those remaining i which are again tied for the minimum, the corresponding coefficients of ε^2; ε^3; and as many powers as necessary are compared *in turn* until a *unique* r is determined. This will always occur *on or before* comparison of the coefficients of ε^m because, if two (or more) polynomial expressions had equal coefficients for all powers, it would mean that *two (or more) rows of the inverse of the basis were constant multiples of each other*, which is *not* possible; see earlier exercise and Problem 2. By this means a unique r can be chosen corresponding to the unique smallest ratio (13) where the *same* choice of r can be made for all $\varepsilon > 0$ in some range. This also means that the values of *all* the basic variables for the next iteration, as given by the polynomial expressions in ε, must remain positive (zero excluded) in some range $0 < \varepsilon < h_{t+1}$, thereby completing the induction. Theorem 2 follows, if we let h_t^* be the smallest h_p for all $0 \leq p \leq t$.

THEOREM 3: *The simplex algorithm as applied to the perturbed problem terminates in a finite number of iterations.*

PROOF: For any fixed number of iterations, N, the values of the basic variables are all positive for *any* fixed ε in some range $0 < \varepsilon < h_N^*$. It follows that *there is a positive (zero excluded) decrease in the value of the objective form.* Therefore, *no basic set of variables could be the same as one obtained in earlier iterations.* Since there are only a finite number of basic sets of variables, not larger than the number of combinations of n things taken m at a time, $\binom{n}{m} = \frac{n!}{(n-m)!\,m!}$, it is not possible that $N > \binom{n}{m}$.

It is easy to see also

THEOREM 4: *The minimal basic feasible solution of the perturbed problem will yield the corresponding solution for the unperturbed problem by setting* $\varepsilon = 0$ *in* (4).

Phase I—Phase II Considerations.

The perturbed problem for Phase II must be suitably chosen so as to be a natural extension of Phase I. At the same time, the setup must be such that any artificial variables remaining in the basis must have zero values in basic solutions in subsequent iterations, when $\varepsilon = 0$. Let $\bar{d}_j \geq 0$ be the relative coefficients of the infeasibility form at the end of Phase I; then the perturbed problem for Phase II for $i = 1, 2, \ldots, m$ becomes

$$(16)\qquad \begin{aligned} \sum_{j=1}^{n} a_{ij}x_j + \delta_i x_{n+i} \qquad\qquad &= b_i + \varepsilon^i \\ \sum_{j=1}^{n} \bar{d}_j x_j \qquad + (-w) &= \quad + \varepsilon^{m+1} \\ \sum_{j=1}^{n} c_j x_j \qquad\qquad &= z \text{ (Min)} \end{aligned}$$

where for $i = 1, 2, \ldots, m$, $\delta_i = 1$ or 0 according to whether or not x_{n+i} is in the basic set at the end of Phase I. It will be noted that, if B is the basis associated with the first m equations, and we now include the w-equation of (16) and $(-w)$ as a basic variable, the extended basis and its inverse for the first $(m+1)$ equations become, respectively,

$$\text{(17)} \qquad \begin{bmatrix} B & 0 \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} B^{-1} & 0 \\ 0 & 1 \end{bmatrix}$$

The values of the basic variables for the *initial solution* for Phase II are now

$$\text{(18)} \qquad \begin{aligned} x_{j_i} &= \bar{b}_i + \beta_{i1}\varepsilon + \beta_{i2}\varepsilon^2 + \ldots + \beta_{im}\varepsilon^m + 0\,\varepsilon^{m+1} > 0 \\ (-w) &= \varepsilon^{m+1} \end{aligned}$$

where the leading terms are positive by Theorem 2.

During Phase II all variables $x_j \geq 0$, $x_{n+i} \geq 0$, and $(-w) \geq 0$ are treated as admissible variables and z is minimized. This, of course, follows precisely the procedure of Phase I. It remains only to show that if ε is set equal to zero, the value of any artificial basic variable is *zero* in any feasible solution. Let σ_i be the simplex multipliers associated with the final basis of Phase I; then, recalling that $d_j = -\sum_i a_{ij}$,

$$\text{(19)} \qquad \bar{d}_j = d_j - \sum_{i=1}^{m} a_{ij}\sigma_i = -\sum_{i=1}^{m} a_{ij}(\sigma_i + 1)$$

Hence, if the i^{th} equation of (16) is multiplied by $(\sigma_i + 1)$ and their sum for $i = 1, 2, \ldots, m$ is *added* to the w-equation of (16), one obtains, using (19) and the fact that $\sigma_i = -\bar{d}_{n+i}$

$$\text{(20)} \qquad \sum_{i=1}^{m} \delta_i x_{n+i} + (-w) = \sum_{i=1}^{m} (\sigma_i + 1)b_i + \sum_{i=1}^{m} (\sigma_i + 1)\varepsilon^i + \varepsilon^{m+1}$$

Now the constant term in the polynomial in ε of the right member of (20) *vanishes and the leading term is positive.*

PROOF: Equation (20) must hold for *every* solution of Phase II; in particular, it must hold for the values given to the artificial variables in the initial solution for Phase II given by (18); but these have the property that their constant terms are all zero and their leading terms are positive. Hence, substituting their polynomial expressions on the *left* in (20), and noting $\delta_i \geq 0$, the same property must hold for the polynomial expressions on the right. Conversely, if the expression on the right has a zero constant term, then so do x_{n+i} and $(-w)$ in any subsequent solution, because their leading terms are maintained positive. It follows that $x_{n+i} = 0$, if $\varepsilon = 0$.

THEOREM 5: *If a minimal feasible solution $x_j = x_j^*$, $z = z^*$ exists, then* *at the end of Phase II, a system of multipliers* $\pi_i = \pi_i^*$ *is obtained with the properties that*

(21)
$$\bar{c}_j^* = c_j - \sum_{i=1}^{m} \pi_i^* a_{ij} \geq 0 \qquad (j = 1, 2, \ldots, n)$$
$$\bar{c}_j^* > 0 \Rightarrow x_j^* = 0, \; x_j^* > 0 \Rightarrow \bar{c}_j^* = 0$$
$$\text{Min } z = z^* = \sum_{i=1}^{m} \pi_i^* b_i$$

Note: The symbol $\Rightarrow$ means "implies."

PROOF: The simplex multipliers π_i obtained at the end of Phase II for the extended problem satisfy the above conditions, *providing* a multiplier k is included for the w-equation of (16). However, dropping the artificial variables and the perturbation, the w-equation, $\Sigma \bar{d}_j x_j$, can be formed from the first m equations, using the multipliers σ_i obtained at the end of Phase I; noting (19), the required multipliers are

(22)
$$\pi_j^* = \pi_i - k(\sigma_i + 1)$$

10-3. PROBLEMS

1. Prove the exercise in § 10-2 that at least one element of each row and each column of the inverse of a basis is non-zero.
2. Prove the exercise in § 10-2 that it is impossible for the elements in a row of the inverse of a basis to be proportional to the elements of another row. (By use of the transpose, prove that the same is true for columns.)
3. Prove that if $\bar{b}_i(\varepsilon) \geq 0$ for $0 < \varepsilon < h$, then there exists an $h' < h$, such that $\bar{b}_i(\varepsilon)$ is *positive* for $0 < \varepsilon < h'$, where $\bar{b}_i(\varepsilon) = \bar{b}_i + \sum_j \beta_{ij}\varepsilon^j$ and $B^{-1} = [\beta_{ij}]$.
4. If $a_k < 0$ and $a_0 = a_1 = \ldots = a_{k-1} = 0$, prove there exists an $h > 0$, such that
$$f(\varepsilon) = a_0 + a_1\varepsilon + a_2\varepsilon^2 + \ldots + a_m\varepsilon^m$$
is negative for *all* $0 < \varepsilon < h$.
5. Prove that if all basic variables for perturbed solutions for each cycle t remain positive in a range $0 < \varepsilon < h_t$, then there exists a common range for all cycles up to t within which all basic variables for the first t iterations remain positive.
6. Solve Hoffman's example using a perturbation method. (See § 10-1, Table 10-1-I.)
7. Solve Beale's example using a perturbation method. (See § 10-1, Table 10-1-II.)
8. (a) Is it possible to construct a class of perturbed problems which are infeasible, but the corresponding class of unperturbed problems are feasible?

(b) Can the class of perturbed problems be feasible, but the unperturbed problem infeasible?

(c) Can the class of perturbed problems have a finite lower bound for z, but not the unperturbed?

(d) Can the class of perturbed problems have a lower bound of $-\infty$ for z, but not the unperturbed?

9. (Charnes.) Develop an alternative perturbation procedure based on replacing b_i by $b_i(\varepsilon) = b_i + \sum_j a_{i_j} \varepsilon^j$ for $i = 1, 2, \ldots, m$. Express the selection rules in terms of the full tableau of the regular simplex method.

10. (Unsolved.) It is conjectured that Beale's example has the least number of variables of any for which circling can occur in the simplex algorithm. Is this true? If not, construct an example with the least.

Problems Based on the Lexicographic Method. [Dantzig, Orden, and Wolfe, 1954-1]

11. An m-component vector A is said to be *lexico-positive*, denoted $A \succ 0$ (see § 13-3), if at least one component is non-zero and the first such is positive. The term "lexico" is short for "lexicographically." A vector A is said to be "lexico-greater" than B, written $A \succ B$, if $A - B \succ 0$. The smallest of several vectors will be denoted Lexico-Min. Prove that this lexicographic ordering of vectors is transitive, in other words

$$A \succ B \text{ and } B \succ C \Rightarrow A \succ C$$

12. Instead of perturbing constants, suppose the constants, b_i, in § 9-2-(1), are replaced by vectors

$$\begin{aligned} b_1^v &= [b_1, 1, 0, \ldots, 0] \qquad (b_i \geq 0) \\ b_2^v &= [b_2, 0, 1, \ldots, 0] \\ &\vdots \\ b_m^v &= [b_m, 0, 0, \ldots, 1] \end{aligned}$$

where the superscript v denotes "vector."

(a) Show, analogous to § 9-2-(4), that the values of the basic variables on some subsequent iteration are replaced by

$$\bar{b}_i^v = [\bar{b}_i, \beta_{i1}, \beta_{i2}, \ldots, \beta_{im}] \qquad (i = 1, 2, \ldots, m)$$

$$z^v = [\bar{z}_t, \pi_1, \pi_2, \ldots, \pi_m]$$

(b) Show, analogous to § 9-2-(10), that the variable chosen to be dropped is selected so that

$$\bar{b}_r^v/\bar{a}_{rs} = \text{Lexico-Min } \{\bar{b}_i^v/\bar{a}_{is}\} \succ 0 \qquad (\bar{a}_{is} > 0)$$

where $\bar{b}_i^v/\bar{a}_{is}$ is a vector formed by dividing the components of $\bar{b}_i^v$ by the scalar $\bar{a}_{is}$.

Prove $\bar{b}_i^v \succ 0$ for all iterations and $z_0^v \succ z_1^v \succ z_2^v \ldots$

13. Define a partial order relation between n-component vectors as follows:

If $x = (\xi_1, \ldots, \xi_n)$ and $y = (\eta_1, \ldots, \eta_n)$, then $x \geq y$ and $y \leq x$ if $\xi_i \geq \eta_i$ for $i = 1, \ldots, n$.

Letting x, y, x_1, y_1, a, denote n-component vectors, prove:

If $x \geq y$ and $y \geq z$, then $x \geq z$;
If $x_1 \geq y_1$ and $x_2 \geq y_2$, then $x_1 + x_2 \geq y_1 + y_2$;
If $x \geq y$, then $\lambda x \geq \lambda y$, where $\lambda \geq 0$ is a scalar and $\lambda x \leq \lambda y$, if $\lambda \leq 0$;
If $x \geq y$ and $a \geq 0$, then $a^T x \geq a^T y$, where a^T is the transpose of a.

REFERENCES

Beale, 1955-1
Charnes, 1952-1
Dantzig, 1951-2, 3, 1959-1
Dantzig, Orden, and Wolfe, 1954-1
Edmondson, 1951-1
Hadley, 1961–2
Hoffman, 1953-1
Nelson, 1957-1

CHAPTER 11

VARIANTS OF THE SIMPLEX ALGORITHM

Introduction.

By a *variant* of the simplex method (in this chapter) is meant an algorithm consisting of a sequence of pivot steps in the primal system, but using alternative rules for the selection of the pivot. Historically these variants were developed to take advantage of a situation where an infeasible basic solution of the primal is available. Often in applications, for example, there occurs a set of problems differing from one another only in their constant terms and cost factors. In such cases, it is convenient to omit Phase I and to use the optimal basis of one problem as the initial basis for the next.

Several methods have been proposed for varying the simplex algorithm so as to reduce the number of iterations. This is especially needed for problems involving many equations in order to reduce the cost of computation. It is also needed for problems involving a large number of variables n, for the number of iterations appears to grow proportionally to n.

As an alternative to using the selection rule $\bar{c}_s = \text{Min } \bar{c}_j$ one could select $j = s$ such that introducing x_s into the basic set gives the largest decrease in the value of z in the next basic solution. This rule is obviously not practical when using the simplex method with multipliers; see Chapter 9. Even using the standard canonical form, considerably more computations would be required per iteration. It is possible, however, to develop a modification of the canonical form in which the coefficient of the i^{th} basic variable is allowed to be different from unity in the i^{th} equation but $\bar{b}_i = 1$. In this form the selection of s by the steepest descent criterion would require little effort; moreover (by means of a special device), no more effort than that for the standard simplex algorithm would be required to maintain the tableau in proper form from cycle to cycle.

Starting in 1960-1961, a number of investigations have been systematically gathering empirical data on the comparative efficiency of various proposals such as the above. Harold Kuhn of Princeton and Philip Wolfe of RAND have been particularly active. Based on their preliminary findings, criteria independent of the units of the activities or of the items appear to be well worth the additional effort.

An important sub-case occurs when a new problem differs from the original in the constant terms alone. The optimal basis of the first problem will still "price out" optimal for the second (i.e., $\bar{c}_j \geq 0$), but the associated solution may not be feasible. Note, however, that optimality implies that the associated solution of the dual is feasible. For this situation, C. Lemke [1954-1] developed the *Dual-Simplex* algorithm as a variant of the standard primal simplex; see § 11-2. Computationally similar variants, the "Method of Leading Variables," by E. M. L. Beale [1954-1] and "PLP (Parametric Linear Programming)," by W. Orchard-Hays [1956-1], [Orchard-Hays, Cutler, and Judd, 1956-1] were developed. These are subsumed in the *Primal-Dual* method of § 11-4 developed first by Ford and Fulkerson for transportation problems (see Chapter 20), and later extended to the general linear program [Dantzig, Ford, and Fulkerson, 1956-1]. These alterations of the algorithm apply when the old basis still prices out optimally in the new system, and thus constitutes a feasible starting solution for the new dual. In contrast Gass and Saaty [1955-1], in their paper on the parametric objective, studied the case of fixed constant terms and varying cost coefficients.

However, when the problems differ by more than just the constant terms, the old basis may not price out optimal in the new problem, and other methods must be sought. When neither the basic solution nor the dual solution generated by its simplex multipliers remains feasible, the corresponding algorithm is called *composite* [Orchard-Hays, 1954-1 and 1956-1]. The *Self-Dual* algorithm of § 11-3 is an example of this.

11-1. COMPLEMENTARY PRIMAL AND DUAL BASES

Lemke [1954-1] discovered a certain complementarity between bases of the primal and dual systems that made it possible to interpret the simplex algorithm as applied to the dual as a sequence of basis changes in the primal; in this case, however, the associated basic solutions of the primal are not feasible, but the simplex multipliers continue to price out optimal (hence, yield a basic feasible solution to the dual). It is well to understand this complementarity, for it provides a means of easily dualizing a problem without the formality of actually restating it.

It will be convenient to take $x_1, x_2, \ldots, x_m$ as basic variables for the primal problem and to show that $(\pi_1, \pi_2, \ldots, \pi_m)$ and $(\bar{c}_{m+1}, \bar{c}_{m+2}, \ldots, \bar{c}_n)$ constitute a basic feasible solution for the dual. This may be shown clearly by use of a Tucker Diagram, Table 11-1-I. The smaller, bold-line square contains the basis, B, of the primal system, while the larger, double-line square gives the transpose of the dual basis, $\bar{B}$. It may easily be shown (and this is left as an exercise) that the determinant of B has the same absolute value as that of $\bar{B}$, so that if B^{-1} exists, then $\bar{B}^{-1}$ exists. With the aid of Table 11-1-I it is easy to see the correspondences given in Table 11-1-II.

TABLE 11-1-I

TUCKER DIAGRAM OF THE PRIMAL DUAL-SYSTEMS

	Primal									
	Variables	x_1	x_2	$\cdots$	x_m	x_{m+1}	$\cdots$	x_n	Relation	Constants
	$\bar{c}_1 \geq 0$	1							$\geq$	0
	$\bar{c}_2 \geq 0$		1						$\geq$	0
	$\cdot$			$\cdot$					$\cdot$	$\cdot$
	$\cdot$				1				$\cdot$	$\cdot$
	$\bar{c}_{m+1} \geq 0$					1			$\geq$	0
Dual	$\cdot$						$\cdot$		$\cdot$	$\cdot$
	$\bar{c}_n \geq 0$							1	$\geq$	0
	π_1	a_{11}	a_{12}	$\cdots$	a_{1m}	a_{1m+1}	$\cdots$	a_{1n}	$=$	b_1
	π_2	a_{21}	a_{22}	$\cdots$	a_{2m}	a_{2m+1}	$\cdots$	a_{2n}	$=$	b_2
	$\cdot$	$\cdots$				$\cdots$			$\cdot$	$\cdot$
	π_m	a_{m1}	a_{m2}	$\cdots$	a_{mm}	a_{mm+1}	$\cdots$	a_{mn}	$=$	b_m
	Relation	$=$	$=$	$\cdots$	$=$	$=$	$\cdots$	$=$	$\leq$ Max v	
	Constants	c_1	c_2	$\cdots$	c_m	c_{m+1}	$\cdots$	c_n	$\geq$ Min z	

TABLE 11-1-II

PRIMAL-DUAL CORRESPONDENCES

	Primal	Dual
Basis	B	$\bar{B}$
Basic Variables	$x_1, x_2, \ldots, x_m$	$\bar{c}_{m+1}, \ldots, \bar{c}_n; \pi_1, \pi_2, \ldots, \pi_m$
Non-Basic Variables	$x_{m+1}, \ldots, x_n$	$\bar{c}_1, \bar{c}_2, \ldots, \bar{c}_m$
Feasibility Condition	$x_i = \bar{b}_i \geq 0$ for all i	$\bar{c}_j \geq 0$ for all j

	Primal Simplex Method	Dual Simplex Method
Optimality Criterion	$\bar{c}_j \geq 0$ for all j	$\bar{b}_i \geq 0$ for all i
Introduction Rule (selection of the new basic variable)	if $\bar{c}_s = \text{Min } c_j < 0$, then choose x_s (pivot in *column s*)	if $\bar{b}_r = \text{Min } \bar{b}_i < 0$, then choose $\bar{c}_r$ (pivot in *row r*)
Representation of the new vector in terms of the basis	$\bar{a}_{1s}, \bar{a}_{2s}, \ldots, \bar{a}_{ms}$	$-\bar{a}_{r,m+1}, -\bar{a}_{r,m+2}, \ldots, -\bar{a}_{r,n}; \beta_{r1}, \beta_{r2}, \ldots, \beta_{rm}$
Rejection Rule (choice of the variable to be dropped from basis)	if $\bar{b}_r/\bar{a}_{rs} = \underset{\bar{a}_{is}>0}{\text{Min}}\ b_i/\bar{a}_{is} \geq 0$ drop x_r (pivot in *row r*)	if $\bar{c}_s/-\bar{a}_{rs} = \underset{-\bar{a}_{rj}>0}{\text{Min}}\ \bar{c}_j/-\bar{a}_{rj} \geq 0$ drop $\bar{c}_s$ (pivot in *column s*)
Pivot Element	$\bar{a}_{rs}$	$\bar{a}_{rs}$
Effect on the Objective Function	z decreases	v increases

11-2. THE DUAL SIMPLEX METHOD

The dual simplex operates with the same tableau as the primal method. However, the relative cost factors are nonnegative from iteration to iteration ($\bar{c}_j \geq 0$ instead of $\bar{b}_i \geq 0$). If it also happens that all the $\bar{b}_i$ are nonnegative, the associated solution will be *optimal* as well as feasible. If not, a pivot row r is chosen where $\bar{b}_r = \text{Min}\, \bar{b}_i < 0$; secondly, the pivot column s, is chosen so that $\bar{c}_s/-\bar{a}_{rs} = \text{Min}\, \bar{c}_j/-\bar{a}_{rj}$ for $\bar{a}_{rj}$ negative. If all $\bar{a}_{rj}$ are nonnegative, it is easy to see that the primal has no feasible solution. Thus, in the dual simplex method, when viewed in terms of the primal variables, one decides first which basic variable to *drop* and then decides which non-basic variable to *introduce*.

EXAMPLE: Suppose a system has been transformed to yield

Cycle 0

(1)
$$
\begin{array}{rrrrrrrl}
x_1 & & & +\,4x_4 & -\,5x_5 & +\,7x_6 & = & 8 \\
 & x_2 & & -\,2x_4 & +\,4x_5 & -\,2x_6 & = & -2 \\
 & & x_3 & +\,x_4 & -\,3x_5 & +\,2x_6 & = & 2 \\
\hline
 & & & x_4 & +\,3x_5 & +\,2x_6 & = & z-4 \\
\bullet & \circ & \bullet & \star & & & &
\end{array}
$$

Since all $\bar{c}_j$, but not all constant terms, are nonnegative, drop the basic variable, x_2, corresponding to $\bar{b}_2 = \text{Min}\, \bar{b}_i = -2$; and introduce x_4 into the next basic set, since $j = 4$ is determined by the criterion, $\text{Min}\, \bar{c}_j/-\bar{a}_{2j} = \bar{c}_4/-\bar{a}_{24} = \frac{1}{2}$, for $\bar{a}_{2j} < 0$. After pivoting, the system becomes (2). Since all $\bar{b}_i$ and $\bar{c}_j$ are nonnegative, the basic solution is now optimal.

Cycle 1

(2)
$$
\begin{array}{rrrrrrrl}
x_1 & +\,2x_2 & & & +\,3x_5 & +\,3x_6 & = & 4 \\
 & -\,\frac{1}{2}x_2 & & +\,x_4 & -\,2x_5 & +\,x_6 & = & 1 \\
 & +\,\frac{1}{2}x_2 & +\,x_3 & & -\,x_5 & +\,x_6 & = & 1 \\
\hline
 & +\,\frac{1}{2}x_2 & & & +\,5x_5 & +\,x_6 & = & z-5 \\
\bullet & & \bullet & \bullet & & & &
\end{array}
$$

Artificial Variables in the Dual Simplex.

Suppose, for the preceding example, that x_2 and x_3 are artificial; we shall proceed as before; we shall, however, disregard all artificial variables once they drop out of the basic set. Thus, x_2 will be dropped from the system in (2). The basic solution is still not feasible because x_3 is artificial. Conceptually, any artificial basic variable, x_j, whose value is positive in the basic solution, may be replaced by $-x_j' = x_j$, so that the basic solution becomes "infeasible," allowing application of the dual simplex rules.

It is clear that the algorithm cannot terminate as long as artificial variables with non-zero values remain in the basic solution. It must terminate either with a proof of the primal's infeasibility or with a primal feasible solution whose artificial variables are all zero or dropped.

In practice, it is probably better not to make the formal substitution, $x_j = -x_j'$, for artificial variables of positive value, but to modify the rules of procedure to produce the same effect. Proceeding with the example, however, dropping x_2 from the system and replacing x_3 by $-x_3'$, we have

Cycle 1 (x_2 dropped, $x_3' = -x_3$ artificial)

$$
\begin{array}{rccccccl}
(3) \qquad & x_1 & & & + 3x_5 & + 3x_6 & = & 4 \\
& & & + x_4 & - 2x_5 & + x_6 & = & 1 \\
& & + x_3' & & + x_5 & - 1x_6 & = & -1 \\
\hline
& & & & 5x_5 & + x_6 & = & z - 5 \\
& \bullet & \circ & \bullet & & \star & &
\end{array}
$$

Cycle 2 (Optimal)

$$
\begin{array}{rccccccl}
(4) \qquad & x_1 & + 3x_3' & & + 6x_5 & & = & 1 \\
& & + x_3' & + x_4 & - x_5 & & = & 0 \\
& & - x_3' & & - x_5 & + x_6 & = & 1 \\
\hline
& & x_3' & & + 6x_5 & & = & z - 6 \\
& \bullet & & \bullet & & \bullet & &
\end{array}
$$

As we have pointed out, many problems have a feasible solution to the dual readily available. For example, if the equations are weighted by the multipliers of a previously optimized system having the same matrix of coefficients, a_{ij}, and if the weighted sum is subtracted from the z-equation, the coefficients, c_j', of the transformed z-equation are nonnegative. Upon augmentation of the new system with artificial variables, the system is (a) in canonical form with respect to the artificial basis, and (b) its relative cost factors, c_j', are nonnegative. Hence, optimizing via the dual algorithm provides an optimum to the primal system without the usual Phase I.

Even in cases where the minimizing form has a *few* negative coefficients, it would appear expedient to replace each negative c_j by $c_j' = 0$ and then optimize. This will provide a basic feasible solution to the original system (not necessarily optimum) which may then be used with the true values of c_j to initiate the usual Phase II of the simplex process.

EXERCISE: Discuss how to recover the true values of $\bar{c}_j$ in this case.

EXERCISE: Prove that no more than k iterations are required to eliminate k artificial variables from a basic set while maintaining feasibility of the dual.

11-3. A SELF-DUAL PARAMETRIC ALGORITHM

Suppose that neither the basic solution nor its complementary dual is feasible. It is a simple matter to increase all the negative $\bar{b}_i$ and $\bar{c}_j$ to non-negative values by adding some constant θ to all of them.

The modified problem is now optimal. Next we will consider ways to maintain the feasibility of the primal and dual systems as the constants and cost coefficients are gradually changed toward their original values. Either the primal or the dual choice criterion will be employed, depending upon whether the basic solutions of the dual or the primal become infeasible.

For example, in the canonical system below, the original problem is obtained by setting $\theta = 0$; the associated basic solutions are infeasible for both the primal and dual.

$$
\text{(1)}\qquad
\begin{array}{ccccccl}
x_1 & & & + 2x_4 & + 2x_5 & & = 10 \\
 & x_2 & & - x_4 & + 1x_5 & & = -1 + \theta \\
 & & x_3 + & x_4 & - 2x_5 & & = -1 + \theta \\
\hline
 & & & 3x_4 & + (-3 + \theta)x_5 & & = z \\
\bullet & \bullet & \bullet & & \star & &
\end{array}
$$

On the other hand, if $\theta \geq 3$, the associated solutions are both feasible. If we start with $\theta = 4$, say, and then let θ approach zero, the associated solutions will remain feasible down to the critical value $\theta = 3$. Just below $\theta = 3$, the primal solution still remains feasible, but the dual solution becomes infeasible since $\bar{c}_3 = -3 + \theta$ is negative. Hence, for θ less than 3 but "very close" to it, we use the *primal* simplex algorithm, introducing x_5 while maintaining the feasibility of both systems. The variable to be dropped is determined from the minimum of the ratios $\bar{b}_i/\bar{a}_{is}$ for $\bar{a}_{is}$ positive. Since

$$\bar{b}_1/\bar{a}_{15} = 5, \qquad \bar{b}_2/\bar{a}_{25} = -1 + \theta$$

in the neighborhood of $\theta = 3$ the second ratio is minimal; thus $\bar{x}_2$ is to be dropped from the basic set in the next cycle. The new canonical system is

$$
\text{(2)}\qquad
\begin{array}{cccccl}
x_1 & - 2x_2 & & + 4x_4 & & = 12 - 2\theta \\
 & x_2 & & - x_4 & + x_5 & = -1 + \theta \\
 & 2x_2 & + x_3 & - 1x_4 & & = -3 + 3\theta \\
\hline
 & (3 - \theta)x_2 & & + \theta x_4 & & = z + (3 - \theta)(\theta - 1) \\
\bullet & & \circ & \star & \bullet &
\end{array}
\qquad (1 \leq \theta \leq 3)
$$

which remains feasible in the range, $1 \leq \theta \leq 3$. *Below* the critical value $\theta = 1$, the primal basic solution becomes infeasible. Hence, for θ less than 1 but very close to it, we use the *dual* simplex algorithm to drop x_3 as a basic variable and maintain the feasibility of both systems. The variable to be introduced is given by the minimum of the ratios $\bar{c}_j/-\bar{a}_{1j}$ for $\bar{a}_{1j}$ negative;

in this case, the only variable with a negative coefficient is x_4. Pivoting, we obtain

$$
\begin{array}{llr}
(3) & x_1 + 6x_2 + 4x_3 = 0 + 10\theta & (0 \le \theta \le 1) \\
& -x_2 - x_3 + x_5 = 2 - 2\theta & \\
& -2x_2 - x_3 + x_4 = 3 - 3\theta & \\
\hline
& (\theta + 3)x_2 + \theta x_3 = z + (3 - \theta)(\theta - 1) + \theta(3\theta - 3) &
\end{array}
$$

which is feasible for both the primal and dual systems at $\theta = 0$. Hence, the *optimal* solution is obtained by setting $\theta = 0$.

In general, it is not necessary to add the same parameter, θ, to all of the negative constants, $\bar{b}_i$ and $\bar{c}_j$, as was done in (1). Several different parameters could be added and each allowed *separately* to tend toward zero. Either way, the net result is the successive application of either the primal or dual simplex rules to change the basis.

How can one be certain that such a process will terminate in a finite number of steps? To answer this, we prove two theorems for the case of a single parameter, θ. First, we note that the values of a basic variable are linear functions of θ, so that, clearly, when a variable is nonnegative for both $\theta = \theta_1$ and $\theta = \theta_2$, then it is nonnegative throughout the interval $\theta_1 \le \theta \le \theta_2$; therefore

THEOREM 1: *It is not possible to have the same basis feasible in the primal and dual for two values of θ, with $\theta_1 < \theta_2$, unless it is also feasible for all values in the range, $\theta_1 \le \theta \le \theta_2$.*

Second, note that, if a change to basis B permits at some critical value $\theta = \theta_1$ a positive (non-zero) decrease in θ, this B is not a repeat of an earlier basis associated with some $\theta_2 > \theta_1$ because at the critical value of θ_1 where the basis change occurred, B would give an infeasible basic solution just above θ_1. Hence, also,

THEOREM 2: *If each change in basis is accompanied by a positive decrease in θ, there can only be a finite number of iterations.*

THEOREM 3: *If there is only one degeneracy in the primal and dual solutions before and after pivoting at a critical value of θ, there will be a positive decrease in θ.*

The latter theorem is due to Gass and Saaty [1955-1] for the case of degeneracy in the *dual* basic solution and to Orchard-Hays [1956-1] for degeneracy in the *primal*. If we prove one of them, the other will follow by duality. Suppose that, corresponding to x_s at a critical value of $\theta = \theta_0$, we have $\bar{c}_s = k(\theta - \theta_0) = \varepsilon$; however, for all *other* non-basic x_j, let $\bar{c}_j$ be expressed linearly in ε by $\bar{c}_j = \alpha_j + \varepsilon\beta_j$ where, by hypothesis, α_j is *positive* (not zero) for $j \ne s$. Also, assume that, for $\varepsilon = 0$, the primal solution is *nondegenerate* before and after x_s displaces some variable, x_{j_r}, in the basic set (actually, we need only require that the basic solution of the primal remain feasible for some *positive* decrease of ε). Under these conditions the new values of the relative cost factors $\bar{c}_j^*$ will be

$$(4) \qquad \begin{aligned} \bar{c}_j^* &= \bar{c}_j - (\bar{a}_{rj}/\bar{a}_{rs})\varepsilon = \alpha_j + [\beta_j - \bar{a}_{rj}/\bar{a}_{rs}]\varepsilon \\ \bar{c}_{j_r}^* &= -(1/\bar{a}_{rs})\varepsilon \end{aligned}$$

Since α_j is positive for all non-basic x_j except x_s, there is a range of values, $\varepsilon_0 < \varepsilon < 0$, with $\varepsilon_0 < 0$, for which $\bar{c}_j^*$ remains positive. In this range $c_{j_r}^* > 0$, and the theorem follows.

THEOREM 4: *If a feasible solution to the primal and dual systems exists for $\theta = \theta_0$ and $\theta = 0$, then feasible solutions exist for all θ in the interval $0 \leq \theta \leq \theta_0$.*

EXERCISE: Prove Theorem 4. This theorem also implies that the solution set generated by all vectors of constant terms, b_i, and cost terms, c_j, for which both the primal and dual problems remain feasible simultaneously, is a convex polyhedron. Prove this too.

11-4. THE PRIMAL-DUAL ALGORITHM

Experiments indicate that the "primal-dual" technique, developed by Fulkerson and Ford (Chapter 20) is very efficient for solving distribution problems. It is closely related to the work of H. Kuhn, who developed a special routine for solving assignment problems called the "Hungarian Method," based on investigations by the Hungarian mathematician Egerváry [1931-1]; see [Kuhn, 1955-1]. Our purpose is to extend this process to the solution of general linear programming problems. As stated here, it is a simplex variant whose number of iterations is quite often fewer than that required by the dual simplex [Dantzig, Ford, and Fulkerson, 1956-1].

Any feasible solution to the dual system may be used to initiate the proposed method. Associated with the dual solution is a *restricted primal* requiring optimization. When the solution of the restricted primal problem has been accomplished, an improved solution to the dual system can be obtained. This in turn gives rise to a *new* restricted primal to be optimized. After a finite number of improvements, an optimal solution is obtained for the original, unrestricted problem.

What markedly distinguishes the Ford-Fulkerson algorithm for distribution problems from the more general case discussed here is that the former method uses a method of optimization of the restricted primal, which appears to be more efficient than the simplex process, whereas the generalization uses the simplex process because it appears to be the most efficient one available. [According to R. Gomory, the former is actually a spiral sequence of simplex pivot steps.]

As in the preceding sections, the entire process may be considered to be a way of starting with an infeasible basic solution and using a feasible solution to the dual already at hand to *decrease the infeasibility form* of the primal in such a manner that, when a feasible basic solution is obtained, it will already be optimal.

The initial canonical form for the primal-dual algorithm is the same as for Phase I of the regular simplex method: see § 5-2-(7). Let

$$
\begin{array}{llr}
(1) & a_{11}x_1 + \cdots + a_{1n}x_n + x_{n+1} & = b_1 \\
 & \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\
 & a_{m1}x_1 + \cdots + a_{mn}x_n \qquad + x_{n+m} & = b_m \\
 & d_1x_1 + \cdots + d_nx_n & = w - w_0 \\
 & c_1x_1 + \cdots + c_nx_n & = z - z_0
\end{array}
$$

where b_i are made nonnegative before insertion of artificial variables, and

$$(2) \qquad d_j = -\sum_i a_{ij}, \text{ and } w_0 = \sum_i b_i$$

so that the sum of the first $m + 1$ equations yields

$$(3) \qquad x_{n+1} + x_{n+2} + \cdots + x_{n+m} = w$$

As before, it is *assumed* that a feasible solution to the dual is available and that, by applying the associated multipliers and summing, *the* c_j *have been adjusted before augmentation by artificial variables*, so that now

$$(4) \qquad c_j \geq 0 \qquad \text{for } j = 1, 2, \ldots, n$$

The problem is to find $x_j \geq 0$, $w = 0$, and Min z satisfying (1).

Suppose that on cycle t, the tableau has the format of Table 11-4-I

TABLE 11-4-I

TABLEAU OF THE PRIMAL-DUAL ALGORITHM

Cycle t

(Basis: $x_1, \ldots, x_m$)

x_1	. . .	x_q	x_{q+1}	. . .	x_m	x_{m+1}	. . .	x_{m+p}	x_{m+p+1}	. . .	x_{n+q}	Constants	π	σ
1						$\bar a_{1,m+1}$	. . .	$\bar a_{1,m+p}$	. . .	. . .	$\bar a_{1n+q}$	$b_1 \geq 0$	π_1	σ_1
	.					.		.			.	.	.	.
		1				.		.			.	.	.	.
			1			.		.			.	.	.	.
				.		.		.			.	.	.	.
					1	$\bar a_{m,m+1}$	. . .	$\bar a_{m,m+p}$	. . .	. . .	$\bar a_{m,n+q}$	$b_m \geq 0$	π_m	σ_m
0	. . .	0	0	. . .	0	d_{m+1}	. . .	d_{m+p}	d_{m+p+1}	. . .	d_{n+q}	$w - \bar w_0$		
*	. . .	*	0	. . .	0	0	. . .	0	$\bar c_{m+p+1}$	. . .	$\bar c_{n+q}$	$z - z_0$		

(Artificial: $x_1, \ldots, x_q$; $\bar c_j = 0$: $x_{q+1}, \ldots, x_{m+p}$; $\bar c_j > 0$: $x_{m+p+1}, \ldots, x_{n+q}$; Restricted primal: $x_1, \ldots, x_{m+p}$)

after relabeling and rearrangement of variables. Artificial variables not in the basic set are dropped from the system. We remark that

(a) The associated primal solution, including artificial variables, is feasible; $\bar b_i \geq 0$.

(b) The multipliers, σ, *are* simplex multipliers of the basis relative to the infeasibility form and generate $\bar{d}_j$.
(c) The multipliers, π, are *not* the simplex multipliers of the basis.
(d) The multipliers π constitute a feasible solution to the dual *excluding* artificial x_j and hence generate $\bar{c}_j \geq 0$.
(e) The values of $\bar{c}_j$ for artificial basic x_j may have either sign and may be omitted; otherwise $\bar{c}_j = 0$ for x_j basic.

THEOREM 1: *If $\bar{w}_0 = 0$, then the basic solution is optimal.*

When $\bar{w}_0 = 0$, the artificial variables all have zero values in the basic solution. Upon dropping them, the feasible solution has $\bar{c}_j$ equal to zero for x_j positive, which fulfills the condition for optimality.

Step 1: Minimizing Infeasibility of the Restricted Primal. At the start of cycle t, it is assumed that there are one or more non-basic x_j whose $\bar{c}_j = 0$. These x_j, together with the basic variables, constitute the *restricted primal* problem. Using only these variables for pivot-choice, the simplex algorithm is applied to minimize w. Usually artificial variables are dropped from the system when they become non-basic. During this subroutine, the values of the multipliers, π_i, are not modified. The simplex multipliers, σ_i, change, of course, at each iteration until w is "minimized," that is, until $\bar{d}_j$ is nonnegative for each x_j of the *restricted primal.*

Step 2: (a) If $\bar{w}_0 = 0$, terminate—the basic solution is feasible and minimal; (b) if $\bar{w}_0 > 0$ and all $\bar{d}_j \geq 0$ $(j = 1, 2, \ldots, n)$, terminate—no primal feasible solution exists. Otherwise, take Step 3.

Step 3: Improving the Dual Solution (Finding a New Restricted Primal). An improved solution of the dual and a new restricted primal is found by using new multipliers,

$$\pi_i^* = \pi_i + k\sigma_i \qquad (i = 1, 2, \ldots, m) \tag{5}$$

which generate nonnegative cost factors,

$$\bar{c}_j^* = \bar{c}_j + k\bar{d}_j \tag{6}$$

where k is a positive number defined by

$$k = \bar{c}_s/(-\bar{d}_s) = \operatorname*{Min}_{\bar{d}_j < 0} \bar{c}_j/(-\bar{d}_j) > 0 \tag{7}$$

The new restricted primal is obtained by using all the basic variables and those non-basic variables whose cost factors, $\bar{c}_j^*$, are zero. This completes the steps of the algorithm.

It should be noted under Step 3 that at least one new variable appears in the restricted primal, namely, x_s, as determined by (7). Note also that $\bar{d}_s < 0$, so that at least one iteration must take place before w is minimized within the new restricted primal. Assuming nondegeneracy, each iteration will decrease infeasibility; hence, no basis can be repeated, and an optimal solution will be obtained in a finite number of iterations.

It should also be noted under Step 2b, that, if the infeasibility factors, $\bar{d}_j$, are all nonnegative, *but* $\bar{w}_0$ *is still positive*, then (5) and (6) constitute a class of feasible solutions to the dual whose objective,

$$v = \bar{z}_0 + k\bar{w}_0 \tag{8}$$

tends to $+\infty$ with increasing k. At the same time, the nonnegativity of *all* the $\bar{d}_j$ and $\bar{w}_0 > 0$ implies that Min w is positive, so that no feasible solutions to the primal exist.

The Initial Restricted Primal.

At first glance it may appear that (1) is not in proper form to initiate the algorithm if *all* c_j are positive. However, if $c_j > 0$ for $j = 1, 2, \ldots, n$, let $\pi = 0$ and view the basic set of artificials as the full set of variables of the restricted primal with multipliers $\sigma = (1, 1, \ldots, 1)$. The algorithm, in this case, is initiated with the finding of an improved dual solution by means of Step 3.

To illustrate the procedure, we consider the problem of finding $x_1 \geq 0$, $x_2 \geq 0, \ldots, x_5 \geq 0$, Min z, and artificial variables, $x_6 = x_7 = x_8 = w = 0$, satisfying

$$\begin{aligned}
x_1 + 4x_2 - 5x_3 + 7x_4 - 4x_5 + x_6 \qquad\qquad &= 8 \\
-4x_2 + 4x_3 - 4x_4 + 4x_5 \qquad + x_7 \qquad &= 2 \\
x_2 - 3x_3 + 4x_4 - 2x_5 \qquad\qquad + x_8 &= 2 \\
-x_1 - x_2 + 4x_3 - 7x_4 + 2x_5 \qquad\qquad &= w - 12 \\
x_1 + 4x_2 + 8x_3 + 8x_4 + 23x_5 \qquad\qquad &= z
\end{aligned} \tag{9}$$

The w-equation is generated in a manner such that the sum of the first four equations is

$$x_6 + x_7 + x_8 = w \tag{10}$$

The first step is to determine the largest number, k, such that $z + kw$ has all its coefficients nonnegative. In this case, $k = 1$ according to (7), so that by simply adding the z and w equations, we obtain (11); for convenience, we have dropped the letter w, since all we are really doing is adding to the z-equation a linear combination of the original equations without the artificial variables.

$$0x_1 + 3x_2 + 12x_3 + x_4 + 25x_5 = z - 12 \tag{11}$$

The first *restricted primal* is obtained by choosing our variables *only* among x_6, x_7, x_8; and x_1 (since the first three are already basic and since only x_1 has a relative cost factor of zero in (11)); we now proceed to minimize w.

Cycle 0 (First Restricted Primal: $(x_6, x_7, x_8; x_1)$)

(12)
$$\begin{array}{rrrrrrrrl}
x_1 & +4x_2 & -5x_3 & +7x_4 & -4x_5 & +x_6 & & & = 8 \\
 & -4x_2 & +4x_3 & -4x_4 & +4x_5 & & +x_7 & & = 2 \\
 & x_2 & -3x_3 & +4x_4 & -2x_5 & & & +x_8 & = 2 \\
-x_1 & -x_2 & +4x_3 & -7x_4 & +2x_5 & & & & = w - 12 \\
\star & & & & & \circ & \bullet & \bullet &
\end{array}$$

Pivoting on x_1 and dropping x_6 from further consideration (since it is artificial), we obtain the next cycle, (13). The value of w has been minimized on the restricted primal $(x_6, x_7, x_8; x_1)$ since all the corresponding $\bar{d}_j$ are nonnegative. This terminates our concern with the first restricted primal.

(13) Cycle 1 $\begin{bmatrix}\text{First Restricted Primal, } (x_6, x_7, x_8; x_1)\text{, is terminated} \\ \text{Second Restricted Primal, } (x_1, x_7, x_8; x_3)\text{, is initiated}\end{bmatrix}$

$$\begin{array}{rrrrrrrl}
x_1 & +4x_2 & -5x_3 & +7x_4 & -4x_5 & & & = 8 \\
 & -4x_2 & +4x_3 & -4x_4 & +4x_5 & +x_7 & & = 2 \\
 & x_2 & -3x_3 & +4x_4 & -2x_5 & & +x_8 & = 2 \\
 & +3x_2 & -x_3 & & -2x_5 & & & = w - 4 \\
\bullet & & \star & & & \circ & \bullet &
\end{array}$$

To determine the new restricted primal, we adjust the z-equation again by determining the largest value of k such that $z + kw$, for z and w as given in (11) and (13), has all its coefficients nonnegative. We find that $k = 12$ is such a value, so that the new cost equation (upon omitting artificial variables as explained above) is

(14)
$$0x_1 + 39x_2 + 0x_3 + x_4 + x_5 = z - 60$$

Since $\bar{c}_3 = 0$ in this equation, the variables of the new restricted primal are x_1, x_7, x_8; and x_3. Introducing x_3 and dropping x_7 from the basic set (and from the system because it is artificial), we have (15). We have now minimized w for the restricted primal, $(x_1, x_7, x_8; x_3)$.

(15) Cycle 2 $\begin{bmatrix}\text{Second Restricted Primal, } (x_1, x_7, x_8; x_3)\text{, is terminated} \\ \text{Third Restricted Primal, } (x_1, x_3, x_8; x_4, x_5)\text{, is initiated}\end{bmatrix}$

$$\begin{array}{rrrrrrl}
x_1 & -x_2 & & +2x_4 & +x_5 & & = 10\tfrac{1}{2} \\
 & -x_2 & +x_3 & -x_4 & +x_5 & & = \tfrac{1}{2} \\
 & -2x_2 & & +1x_4 & +x_5 & +x_8 & = 3\tfrac{1}{2} \\
 & +2x_2 & & -x_4 & -x_5 & & = w - 3\tfrac{1}{2} \\
\bullet & & \bullet & \star & & \circ &
\end{array}$$

Once more we are ready to adjust the z-equation, so as to determine a new restricted primal, this time by determining the largest value of k such that $z + kw$, for z and w as given in (14) and (15), has nonnegative coefficients. This value turns out to be $k = 1$, giving (upon dropping of artificial variables)

(16)
$$0x_1 + 41x_2 + 0x_3 + 0x_4 + 0x_5 = z - 63\tfrac{1}{2}$$

The corresponding restricted primal is $(x_1, x_3, x_8; x_4, x_5)$, since both x_4 and x_5 have zero cost factors. Incidentally, we see that, except for x_2, all the original variables have been brought back into the problem.

To minimize w for the new restricted primal, we now introduce x_4 into the basic set, obtaining the system

Cycle 3 (Optimal)

$$\begin{array}{lrrrrrl} (17) & x_1 + 3x_2 & & & - & x_5 = & 3\tfrac{1}{2} \\ & - 3x_2 + & x_3 & & + & 2x_5 = & 4 \\ & - 2x_2 & & + x_4 + & & x_5 = & 3\tfrac{1}{2} \\ & 0x_1 + 0x_2 + & 0x_3 + & 0x_4 + & & 0x_5 = & w - 0 \\ & \bullet & \bullet & \bullet & & & \end{array}$$

whose associated solution, $(3\frac{1}{2}, 0, 4, 3\frac{1}{2}, 0)$, and $w = 0$ is feasible for the original *unrestricted* problem. Since the coefficients of the z-equation have been kept nonnegative throughout our procedure, this solution is evidently minimal.

Note in (15) that if x_5 had been introduced instead of x_4, it would have taken *two* iterations to minimize w, since x_3 would have dropped out instead of x_8.

The minimum value of z, $63\frac{1}{2}$, is obtained from (16).

11-5. AN ALTERNATIVE CRITERION FOR PHASE I

This criterion, first suggested informally by H. Markowitz, has many points in common with the primal-dual algorithm [Dantzig, Ford, and Fulkerson, 1956-1] treated in the last section and with the dual algorithm [Lemke, 1954-1].

Like the standard simplex, this method uses basic feasible solutions but changes the criterion for choice of new basic variables in Phase I. The standard criterion selects x_s in such a way that w, which measures primal infeasibility, decreases at the maximum rate when x_s is increased. Since this criterion is not influenced by the objective form, z, the feasible solution provided by Phase I may be quite different from the one required to minimize z. To correct this, *it is proposed that x_j be chosen in such a way that there is a maximum decrease (least increase) of the objective form per unit decrease of the infeasibility form.*

For some iteration, let the canonical tableau be the same as (3). The variables $x_1, x_2, \ldots, x_m$, some of which will be artificial, are assumed by rearrangement and relabeling to constitute the basic set. The standard criterion for Phase I chooses s in such a way that

$$(1) \qquad \bar{d}_s = \text{Min } \bar{d}_j < 0$$

Instead, the present proposal is to choose $j = s$ such that

$$(2) \qquad \bar{c}_s/(-\bar{d}_s) = \underset{\bar{d}_j<0}{\text{Min}}\ \bar{c}_j/(-\bar{d}_j) \qquad (\bar{d}_s < 0)$$

with no other change in the algorithm. In the event that several j minimize this ratio, the choice is made among those tied by the standard criterion, (1). For Phase II, s is chosen in the usual manner (i.e., such that $\bar{c}_s = \text{Min } \bar{c}_j < 0$).

(3)

x_1	x_2	. . .	x_m	x_{m+1}	. . .	x_n	. . .	x_{n+m}	Constants
1				$\bar{a}_{1,m+1}$	. . .	$\bar{a}_{1,n}$	. . .	$\bar{a}_{1,n+m}$	$\bar{b}_1$
	1			$\bar{a}_{2,m+1}$	. . .	$\bar{a}_{2,n}$	. . .	$\bar{a}_{2,n+m}$	$\bar{b}_2$
		.		.		.		.	.
		.		.		.		.	.
		.		.		.		.	.
			1	$\bar{a}_{m,m+1}$	. . .	$\bar{a}_{m,n}$	. . .	$\bar{a}_{m,n+m}$	$\bar{b}_m$
				$\bar{c}_{m+1}$	. . .	$\bar{c}_n$	. . .	$\bar{c}_{n+m}$	$z - z_0$
				$\bar{d}_{m+1}$	. . .	$\bar{d}_n$	. . .	$\bar{d}_{n+m}$	$w - w_0$

11-6. PROBLEMS

1. Review the following results from § 6-3:

   (a) Show that if a linear programming problem has a finite lower bound for some given set of constant terms, then it has a finite lower bound for *any* set of constant terms, providing a feasible solution exists.

   (b) Suppose that a linear programming problem is augmented with artificial variables whose sum is bounded by a constant (not necessarily zero). If z is minimized, allowing the artificial variables to enter the basic set with nonnegative value, prove that the minimum is finite or infinite, depending on whether Min z of the *original* problem is finite or infinite.

2. Show that no basis can re-occur in the parametric linear programming procedure. What assumption is made about degeneracy?
3. Develop lexicographic (perturbation) schemes for the dual simplex; for the self-dual parametric algorithm. What is a lexicographic scheme for the primal-dual algorithm and the Phase I alternative of Markowitz?
4. Re-solve the blending problem illustrated in § 5-2 (see Table 5-2-V), applying the different variants discussed in this chapter.
5. Show that, if no artificial variables remain in the basic set using the primal-dual algorithm, the solution is optimal.

REFERENCES

Beale, 1954-1
Dantzig, 1954-1, 2
Dantzig, Ford, and Fulkerson, 1956-1
Egerváry, 1931-1
Gass and Saaty, 1955-1
Gunther, 1955-1
Kuhn, 1955-1
Lemke, 1954-1
Orchard-Hays, 1954-1, 1956-1
Orchard-Hays, Cutler, and Judd, 1956-1
Orden, 1952-1, 1955-1

CHAPTER 12

THE PRICE CONCEPT IN LINEAR PROGRAMMING

In previous chapters we have frequently referred to the simplex multipliers as "prices." In this chapter, we shall discuss economic examples, which not only show how this viewpoint of the multipliers arises naturally, but also how it permits an economic interpretation of the simplex method itself. As we have seen, these multipliers are themselves the solution to a second linear programming problem which is called the *dual* of the first. The first example shows how a price can arise in a situation where there are no prices to begin with; the second and third examples, in § 12-2, show how the dual system of competitive prices for new items arises "naturally" in a situation where prices for old products and methods already exist. A fuller treatment of the relation of linear programming to economic theory can be found in several excellent books; see references at the end of this chapter.

12-1. THE PRICE MECHANISM OF THE SIMPLEX METHOD

The Manager of the Machine Tool Plant.[1]

Consider the problem of a manager of a machine tool plant, say, in an economy which has just been socialized. The central planners have allocated to this manager *input* quantities $+b_1, \ldots, +b_k$ of materials which we designate by $1, \ldots, k$ and have instructed this manager to produce *output* quantities $-b_{k+1}, \ldots, -b_m$ of the machine tools numbered $k+1$ through m (the $b_{k+1}, \ldots, b_m$, being outputs, are negative numbers by our conventions). The planners further direct the manager that he shall use as little labor as possible to meet his required production goals and that he must pay the workers with labor certificates, one certificate for each hour of labor. The central planners have declared old prices of items to be of no use and have not provided any new prices to the manager to guide him.

The manager has at his disposal many production activities, say, n of them, each of which he can describe by a column vector, $\langle a_{1j}, \ldots, a_{mj}\rangle$. If the j^{th} process inputs a_{ij} units of the i^{th} item per unit level of operation, a_{ij} is positive. If, on the other hand, the j^{th} process outputs a_{ij} units of item i per unit level of operation, a_{ij} is negative. The j^{th} process also requires c_j

[1] This subsection was contributed by C. Almon, Jr.

units of labor per unit level of operation. The manager's problem then is to find levels of operation for all the processes, $x_1, \ldots, x_n$, which satisfy

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n &= b_2 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n &= b_m \end{aligned} \tag{1}$$

and minimize the total amount of labor used,

$$c_1x_1 + c_2x_2 + \ldots + c_nx_n = z \text{ (Min)}$$

The x's must, of course, be nonnegative. In matrix notation $Ax = b$, $cx = z$.

The manager knows of m old reliable processes, namely $1, \ldots, m$, with which he is sure he can produce the required outputs with the given inputs though the labor requirements may be considerable. Thus, he knows he can find nonnegative $x_1, \ldots, x_m$, such that

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1m}x_m &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2m}x_m &= b_2 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mm}x_m &= b_m \end{aligned} \tag{2}$$

or, in matrix notation, $Bx = b$. We shall assume that B is a feasible basis for (1).

This manager has learned, however, that his workers are prone to be extravagant with materials, using far more inputs than are called for in (1). Unless he can keep this tendency in check, he knows he will fail to meet his quotas. Formerly, he deducted the cost of the extra materials from the workers' wages; but now that all prices have been swept away, he lacks a common denominator for materials and wages. Suddenly, in a moment of genius, it occurs to him that he can make up his own prices in terms of labor certificates, charge the operators of each process for the materials they use, credit them for their products, and give them the difference as their pay. Being a fair man, he wants to set prices such that the efficient workers can take home a certificate for each hour worked. That is, he wants to set product and raw material prices $\pi_1, \ldots, \pi_m$, such that the net yield on a unit level of each basic activity j is equal to the amount of labor c_j which it requires:

$$\begin{aligned} \pi_1a_{11} + \pi_2a_{21} + \ldots + \pi_ma_{m1} &= c_1 \\ \pi_1a_{12} + \pi_2a_{22} + \ldots + \pi_ma_{m2} &= c_2 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \pi_1a_{1m} + \pi_2a_{2m} + \ldots + \pi_ma_{mm} &= c_m \end{aligned} \tag{3}$$

or, using matrix notation,

$$\pi B = c$$

where π and c are row vectors.

The manager now proceeds to solve (2) for x by finding B^{-1} and setting

$$x = B^{-1}b = \bar{b}$$

Turning to (3), he notes that the solution is almost at hand, for

$$\pi = cB^{-1}$$

Common sense tells the manager that by using his pricing device he would have to pay out exactly as many labor certificates as he would if he paid the labor by the hour and all labor worked efficiently. Indeed, this is easily proved since the total cost using his calculated prices for all activities is $\pi b = cB^{-1}b = cx$, where cx is the cost of paying wages directly.

The manager harbors one qualm about his pricing device, however. He remembers that there are other processes besides the m he is planning to use and suspects that among the remainder there may be some for which his pricing device would require him to pay

$$c'_j = \pi_1 a_{1j} + \pi_2 a_{2j} + \cdots + \pi_m a_{mj} \tag{4}$$

which is greater than the direct wages c_j. If such a process comes to the attention of the workers, they will see that by using it they can, if they work efficiently, get more than one labor certificate for one hour's work. The wily men will then try to substitute these processes in such a way as not to affect the material inputs, yet achieve the same outputs. If they can do so, they will pocket excess wages, and before long, the secret will be out that he is paying for labor not performed. On looking over the list of processes in (1), the manager finds several for which the inequality $c'_j > c_j$ holds. Denoting the excess wages of the j^{th} process by $\bar{c}_j$,

$$\bar{c}_j = c_j - (\pi_1 a_{1j} + \pi_2 a_{2j} + \cdots + \pi_m a_{mj})$$

the manager singles out process s, the one offering the most excess wages:

$$\bar{c}_s = \text{Min } \bar{c}_j < 0 \tag{5}$$

Before devising repressive measures to keep the workers from using processes that yield excess wages, the manager, a meditative sort of fellow, pauses to reflect on the meaning of these excess wages. Having always had a bent for mathematics, he soon discovers a relation which, mathematically, we express by saying that the vector of coefficients $\bar{a}_{ij}$, for any activity j in the canonical form, can be used as weights to form a linear combination of the *original* vectors of the basic activities, which has the *same* input and output coefficients as that of activity j for all items, except possibly the cost item. In particular, he finds that he can represent activity s, the one yielding the most excess wages, as a linear combination of his "old reliables" as follows:

$$(6)\qquad \begin{bmatrix} a_{11} \\ a_{21} \\ \cdot \\ \cdot \\ \cdot \\ a_{m1} \end{bmatrix} \bar{a}_{1s} + \begin{bmatrix} a_{12} \\ a_{22} \\ \cdot \\ \cdot \\ \cdot \\ a_{m2} \end{bmatrix} \bar{a}_{2s} + \cdots \begin{bmatrix} a_{1m} \\ a_{2m} \\ \cdot \\ \cdot \\ \cdot \\ a_{mm} \end{bmatrix} \bar{a}_{ms} = \begin{bmatrix} a_{1s} \\ a_{2s} \\ \cdot \\ \cdot \\ \cdot \\ a_{ms} \end{bmatrix}$$

where $\bar{a}_{is}$ are the coefficients of x_s in the canonical form. In words, (6) tells him that x_s units of activity s can be *simulated* by a combination of $\bar{a}_{1s}x_s$, $\bar{a}_{2s}x_s$, . . ., $\bar{a}_{ms}x_s$ units of the basic set of activities $(1, 2, \ldots, m)$. Thus, if the workers introduce x_s units of activity s, the levels of the basic activities must be *adjusted* by these amounts (up or down, depending on sign) if the material constraints and output quotas are to remain satisfied. Now the labor cost of simulating one unit of activity s by the m old reliables is

$$c_1\bar{a}_{1s} + c_2\bar{a}_{2s} + \cdots + c_m\bar{a}_{ms}$$

This amount is precisely what the manager would pay for the various inputs and outputs of one unit of the real activity s if he were to use the prices π_i. For, considering the vector equation (6) as m equations and multiplying the first equation through by π_1, the second by π_2, etc., and summing, one obtains immediately from (3)

$$(7)\qquad c_1\bar{a}_{1s} + c_2\bar{a}_{2s} + \cdots + c_m\bar{a}_{ms} = \pi_1 a_{1s} + \pi_2 a_{2s} + \cdots + \pi_m a_{ms}$$

It is now readily shown that the fact that the process s yields excess wages means to the manager that it takes less labor to operate s directly than to simulate it with the m old activities. This is clear from (4), (5), and (7), which yield

$$(8)\qquad c_1\bar{a}_{1s} + \cdots + c_r\bar{a}_{rs} + \cdots + c_m\bar{a}_{ms} > c_s$$

Hence, he reasons, s must be in a sense more efficient than at least one of these old processes. Recalling that the planners instructed him to use as little labor as possible, the manager decides to use activity s in place of one of the original m. He soon discovers that if he wishes to avoid the non-sensical situation of planning to use some activity at a negative level, the process r to be replaced by process s must be chosen, as we have seen in § 5-1-(18), so that

$$\bar{b}_r/\bar{a}_{rs} = \text{Min } \bar{b}_i/\bar{a}_{is} \qquad (\bar{a}_{is}, \bar{a}_{rs} > 0)$$

Because $\bar{a}_{rs} > 0$, it follows from (8) that

$$(9)\qquad \{(c_1\bar{a}_{1s} + \cdots + c_{r-1}\bar{a}_{r-1,s} + c_{r+1,s}\bar{a}_{r+1,s} + \cdots + c_m\bar{a}_{ms} - c_s)/(-\bar{a}_{rs})\} < c_r$$

The coefficients of $c_1, c_2, \ldots, c_s, \ldots, c_m$ in (9) are precisely the weights required to simulate activity r out of the activities in the new basis, as can be seen by re-solving (6) for column r in terms of the others. But, e.g.,

$c_1\bar{a}_{1s}/-\bar{a}_{rs}$ is the labor cost of the first activity in the simulation of activity r, so that the left-hand side of (9) represents the total labor cost of simulating a unit level of activity r by the activities in the new basis, while c_r is of course the labor cost of one unit of the real activity r. Hence (9) shows that activity r is indeed less efficient in its use of labor than those in the new basis.

In summary, the manager now knows that, if there exist processes for which his pricing device yields more labor certificates than are actually required, then he can substitute one of these processes for one in the original set and thereby bring about a more efficient use of labor. Since the planners instructed him to use as little labor as possible, it is clearly wise for him to plan production using activity s instead of one of the m he had originally intended to use, to readjust the levels of use of the remaining ones, and to change the prices, so that none of the processes that will then be in use gives excess wages.

Having learned this lesson, the manager proceeds again to look for processes offering excess wages, to put into operation the worst offender, to readjust prices, to look for excess wages, and so on until he finds a set of prices π^0 under which no process offers excess wages. Fortunately for him, it turns out (as we know) that in a finite number of steps he will find such a set of prices.

Let us pause for a moment to consider the meaning of one of these prices, say π_i. Suppose we introduce into the manager's (A) matrix equation (1), a fictitious activity which consists simply of increasing his allotment of item i if $b_i > 0$ or of decreasing his quota on i if $b_i < 0$. Such an activity will be represented by a column which has all zeros except for a one in the i^{th} row. Thus the labor cost of simulating this activity with those of the final basis is, by (4), precisely π_i. Thus, π_i is the *labor value*, the labor which can be replaced by one additional unit of item i.

The manager has now achieved his objective of finding a set of prices to charge for raw materials and to pay for finished goods which will keep his workers from wasting inputs and yet offer no possibilities for excess wages. But he now begins to wonder if he is truly safe from the planners' criticism for the amount of labor he uses. He begins by specifying explicitly what he intends to do. His operating plan consists of a set of activity levels $x^o = (x_1^o, x_2^o, \ldots, x_n^o)$ satisfying

$$(10) \qquad Ax^o = b \qquad (x_j^o \geq 0)$$
$$cx^o = z^o$$

a set of prices $\pi^o = (\pi_1^o, \pi_2^o, \ldots, \pi_m^o)$, and, for each activity, an excess labor cost

$$(11) \qquad \bar{c}_j^o = c_j - \sum_{i=1}^{m} a_{ij}\pi_i^o \geq 0$$

with the property that,

$$\text{(12)} \qquad \text{if} \quad \bar{c}_j^o = c_j - \sum_{i=1}^{m} \pi_i^o a_{ij} > 0, \qquad \text{then } x_j^o = 0$$

We shall now prove that the manager's operating plan has minimized his labor costs. Writing $\bar{c}^o = (c - \pi^o A) = (\bar{c}_1^o, \bar{c}_2^o, \ldots, \bar{c}_n^o)$, we have from (10) that

$$\text{(13)} \qquad \bar{c}^o x^o = (c^o - \pi^o A)x^o = z^o - \pi^o b$$

where by (10), z^o is the total labor requirement of the manager's operating plan. But because of (12), $\bar{c}x^o = 0$, and therefore

$$\text{(14)} \qquad z^o = \pi^o b$$

Now let $x = (x_1, x_2, \ldots, x_n)$ be *any other feasible operating plan, and let z be its labor requirements; then*

$$\text{(15)} \qquad Ax = b \qquad (x_j \geq 0)$$
$$cx = z$$

It follows by multiplying $Ax = b$ by π^o and subtracting from $cx = z$ and noting (14):

$$\text{(16)} \qquad (c - \pi^o A)x = z - \pi^o b = z - z^o$$
$$\text{or} \quad \sum_j \bar{c}_j^o x_j = z - z^o$$

But the left member is the sum of nonnegative terms and therefore $z \geq z^o$. Hence, no other feasible operating plan exists whose labor requirement is less than the one found by the manager.

At this point, we can imagine the manager's delight at his genius, for *as a by-product of his search for prices that will cause his workers to work efficiently, he has also discovered those processes which minimize his labor requirements.* Without explicitly trying, he has solved his assigned task of keeping his use of labor to a *minimum*!

The Dual Problem.

Let us review the requirements satisfied by the prices found by the manager. First, there will be no excess wages in any activity; that is

$$\text{(17)} \qquad \pi A \leq c$$

Second, the total amount of wages to be paid for all activities should be the same whether they are paid directly or by use of the pricing device; that is

$$\text{(18)} \qquad z^o = cx^o = \pi b$$

where x^o is an optimal solution to (1).

Let us now show that these prices, π, themselves represent the optimal solution to another linear programming problem—specifically, to the dual problem of our manager's original production problem. By multiplying the j^{th} equation of (17) by x_j^o and summing, we find that

$$(19) \qquad \pi_1 \sum_{j=1}^{n} a_{1j}x_j^o + \pi_2 \sum_{j=1}^{n} a_{2j}x_j^o + \cdots + \pi_m \sum_{j=1}^{n} a_{mj}x_j^o \leq \sum_{j=1}^{n} c_j x_j^o$$

Substituting from (1)

$$(20) \qquad b_i = \sum_{j=1}^{n} a_{ij}x_j^o \qquad (i = 1, 2, \ldots, m)$$

gives

$$\pi_1 b_1 + \pi_2 b_2 + \cdots + \pi_m b_m \leq c_1 x_1^o + c_2 x_2^o + \cdots + c_n x_n^o$$

Thus, $\pi b \leq cx^o$ for any π that satisfies (17). The prices, π^o, found by the manager give $\pi^o b = cx^o$ and thus π^o maximizes πb, subject to the constraints (17). Hence, $\pi = \pi^o$ may be viewed as an optimal solution to the *dual* linear programming problem, namely,

$$(21) \qquad \begin{aligned} \pi A &\leq c \\ \pi b &= v \text{ (Max)} \end{aligned}$$

The relation $\pi^o b = cx^o$, where π^o is an optimal solution to (21), and x^o, an optimal solution to (1), agrees with the *duality theorem* established in § 6-3, Theorem 1. The reader should interpret for himself the economic meaning of maximizing πb in the case of the tool plant manager.

Let us now consider two other examples showing how the dual problem arises in other ways and how it may be interpreted.

12-2. EXAMPLES OF DUAL PROBLEMS

The Scheme of the Ambitious Industrialist.

In this section we shall formulate a problem whereby the dual problem arises "naturally." Consider a defense plant which has just been built by the government. The plant has been designed to produce certain definite amounts, $-b_i$, $i = k + 1, \ldots, m$, of certain defense items and to use only certain definite amounts, $+b_i$, $i = 1, 2, \ldots, k$, of certain scarce materials which will be provided without cost by other government plants. The consulting engineers who designed the plant provided the government with a list of the various processes available in the plant and their input and output coefficients. Somewhat confused by this mass of data, the civil servants who were supposed to operate the plant decide to call in a private industrialist to consult on how they should plan their production. The industrialist realizes that it would be good training for his men and a feather in his cap actually to operate the plant. Accordingly, once he gets the information and studies the data, he proposes a flat fee for which he will manage the plant, turn over to the government the required amounts of output and use no more than the alloted quantities of the scarce materials. The civil service men declare that, all other things being equal, they think it would be best for the government to operate the plant, but if he can convince them that his proposal is a good one (meaning that if the government operates the plant, it is unlikely it could do so less expensively), they will accept his offer.

The industrialist takes the data back to his office, gets out his linear programming book, and uses the data on input-output coefficients to form a matrix, A, similar to that of the manager of the machine tool plant, but with this difference: he includes the cost of purchased materials needed per unit of process j in the process cost, c_j.

To determine the minimum fee for which he can afford to operate the defense plant, the industrialist has only to solve the following linear program: find $x \geq 0$, such that

$$\begin{aligned} Ax &= b \\ cx &= z \text{ (Min)} \end{aligned} \tag{1}$$

He calls in his computer man, gives him the problem, and the next morning the results are on his desk: z^o is the minimum cost and x^o is the vector of optimal process utilization levels. His first thought is to explain the linear programming technique to his civil service friends, show them the final tableau, and thereby convince them that they can do no better than to accept his offer and pay him z^o. But then he realizes that this plan will give away his secret; the civil servants will have no further need for him; they will take his x^o vector and operate the plant themselves. Hence, he must convince them that z^o is minimal without giving away his plan x^o.

To this end, he decides to invent a system of prices which he will offer to pay for the materials, provided he is paid certain prices for outputs. He wants these prices to be such that there are no profits on any individual activity, for if there were profits, the government would spot them and would want to run this particular activity itself. On the other hand, given these restraints, he wants to make as much money as possible. That is, he wants his price vector π, a row, to satisfy

$$\pi A \leq c \tag{2}$$

and

$$\pi b = v \text{ (Max)}$$

Again he calls the computer man, who recognizes this problem as the dual of the one he solved the night before and immediately produces the solution: optimal $\pi = \pi^o$, the simplex multipliers from the last stage of the previous problem, and maximal $v = v^o$. Fortunately, they note with relief, $v^o = z^o$.

With these results under his arm, the industrialist goes back to see the civil servants and presents his offer in price terms. The bureaucrats check to be sure that every one of the inequalities (2) is satisfied, and, of course, calculate the total cost using these prices: $\pi^o b = v^o$. The industrialist then invites them to consider any program, x, satisfying (1). Its cost to them, if they operate the plant themselves, is cx. But, replacing π by π^o in (2) and multiplying both sides by any *feasible* x yields

$$\pi^o A x \leq cx \tag{3}$$

or, by (1),

$$\pi^o b \leq cx \tag{4}$$

Hence,

$$v^o \leq cx \tag{5}$$

so that the cost of the program will be at least v^o. This argument convinces the civil servants that they can do no better than to accept the industrialist's flat fee offer of v^o. With one last hope of operating the plant themselves, they try to pry out of him just how much of each process he intends to operate; but he feigns ignorance of such details and is soon happily on his way with his contract signed.

The Nutrition Pill Manufacturer.[2]

A housewife can buy foods in the grocery which vary in cost and nutritional elements. For simplicity, let us assume five foods and only two nutritional elements, calories and vitamins. The housewife's problem is to determine a minimum cost diet that has at least 21×100 calories and 12×100 vitamin units per person per day. The data for the simple linear programming model for this problem are given in Table 12-2-I.

TABLE 12-2-I

PRIMAL PILL PROBLEM

Items	Activities							Constants
	Buying Food					Having Excess		
						Cal.	Vit.	
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
Calories	-1		-1	-1	-2	1		$-21 \times (100)$
Vitamins		-1	-2	-1	-1		1	$-12 \times (100)$
Cost	20	20	31	11	12			z (Min)

[2] This example is a variant of one given in the book, *Linear Programming and Economic Analysis*, by Dorfman, Samuelson, and Solow [1958-1]. Our discussion can be regarded as a supplement to theirs.

A nutrition pill manufacturer wishes to supply the entire dietary requirements by marketing in the drug stores a pure calorie pill and a pure vitamin pill at prices that will not only compete with similar "foods" 1 and 2 offered in the grocery store but will be a cheaper source of nutritional needs than any food on the market. What prices should he charge in order to maximize his revenues?

Let π_1 be the price he charges per calorie pill and π_2 be the price per vitamin pill (each pill = 100 units). Then the dual problem takes the form shown in (6a). By substituting for π_i,

$$\pi_i = -y_i$$

it takes on the form (6b) which is more convenient for plotting; see Fig. 12-2-I.

Dual Pill Problem

(6a)
$$\begin{aligned}
-\pi_1 \qquad\quad &\leq 20 \\
-\quad \pi_2 &\leq 20 \\
-\pi_1 - 2\pi_2 &\leq 31 \\
-\pi_1 - \quad \pi_2 &\leq 11 \\
-2\pi_1 - \quad \pi_2 &\leq 12 \\
\pi_1 \qquad\quad &\leq 0 \\
\pi_2 &\leq 0 \\
-21\pi_1 - 12\pi_2 &= v \text{ (Max)}
\end{aligned}$$

(6b)
$$\begin{aligned}
y_1 \qquad\quad &\leq 20 \\
y_2 &\leq 20 \\
y_1 + 2y_2 &\leq 31 \\
y_1 + \quad y_2 &\leq 11 \\
2y_1 + \quad y_2 &\leq 12 \\
y_1 \qquad\quad &\geq 0 \\
y_2 &\geq 0 \\
21y_1 + 12y_2 &= v \text{ (Max)}
\end{aligned}$$

In (6b) the sum of the terms to the left of the inequality (such as $y_1 + 2y_2$ in the third constraint) represents the cost to the housewife if she *simulates* the type of food in question by purchasing nutrition pills with equal amounts of nutritional elements; the quantity to the right represents the cost to her if, instead, she buys the food. In each case it is required that it cost no more to buy the simulated food.

The inequalities (6b) are plotted in Fig. 12-2-I, and it is evident that the optimum choice of prices is to charge 1 cost unit for the calorie pill and 10 cost units for the vitamin pill.

(7) Optimum Prices: $\pi_1^* = -1, \pi_2^* = -10$

Maximum Revenue: $v^* = \pi_1^* b_1 + \pi_2^* b_2 = -(-21) - 10(-12) = 141$

It should be noted that there is a built-in assumption that the drug manufacturer will supply all dietary needs. Granted this, it is clear that his prices must be competitive with the price of each food, for otherwise the housewife would buy part of the diet in the grocery store and part in the drug store. Another point worth noting is that foods 4 and 5 are still competitive with the pills; that is to say, no more costly than the pills, as can be seen by substituting these values of π_i^* in (6a). Thus, pill prices must be set slightly below the optimum in order to overcome any residual bias toward pills and thereby guarantee the market. In the nutrition case, it is obvious from Fig. 12-2-I (and true in general when all $a_{ij} \leq 0$) that a slight

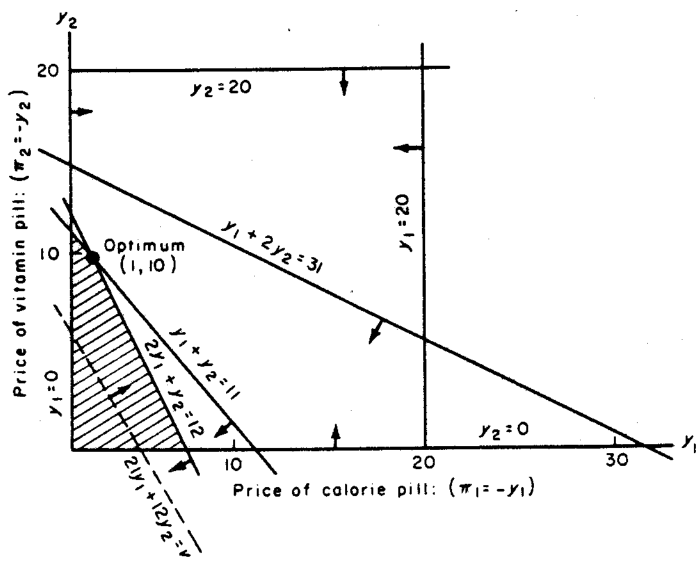


Figure 12-2-I. The dual pill problem.

decrease in all y_i (or increase in π_i) from the optimum is sufficient to guarantee the entire market, if the decisions of all housewives are determined strictly by minimum cost. Let us hope that this is not the case.

12-3. THE SIGN CONVENTION ON PRICES

DEFINITION: The *price* of an item is its exchange ratio relative to some standard item. If the item measured by the objective function is taken as a standard, then the price π_i of item i is the change it induces in the objective z per change of b_i, for small changes of b_i. *Asset* items will have *negative values* for prices because an increase in their quantity should bring about a *decrease* in the objective function, since we are minimizing.

The optimal dual "prices" π_i satisfy the relation

$$\bar{z}_0 = \pi_1 b_1 + \pi_2 b_2 + \cdots + \pi_m b_m$$

where Min $z = \bar{z}_0$. This same relation yields the value of z in any basic solution if π_i are the corresponding simplex multipliers. If the basic solution *is nondegenerate*, then, for small changes in any b_i, the basis and hence the simplex multipliers will remain constant. Thus the change in value of z per change of b_i for small changes in b_i is clearly π_i; hence by the above definition, π_i can be interpreted as the price.

Let us introduce into the program a fictitious "procurement" activity—"increasing the allotment of item i" whose coefficients are zero except for minus unity in row i and p_i in the cost row. Query: How low must the cost p_i be before it pays to increase the allotment of i? Pricing out this activity, we see it pays if

$$p_i + \pi_i < 0 \text{ or } p_i < -\pi_i$$

Hence, $-\pi_i$ is the break-even cost of the item i procurement activity.

Now, according to our interpretation of the price scheme developed in § 12-1 and § 12-2, for each unit of activity j, the input, $a_{ij} > 0$, would induce a payment to the manager of $\pi_i a_{ij}$. If the item is an asset, $\pi_i < 0$ and $\pi_i a_{ij} < 0$. In other words, *the flow of money is out* (negative). Similarly, if $a_{ij} < 0$, then the flow of money is toward the activity $\pi_i a_{ij} > 0$.

The total flow of money into the activity by the price device is given by pricing it out, i.e.,

$$\sum_i \pi_i a_{ij}$$

If this value exceeds c_j, the amount that would be received by the alternative of direct payment, then this activity (or some other with the same property) will, as we have seen for the example of § 12-1, be used in lieu of a basic activity now in use. This in turn will generate a new set of prices, etc.

12-4. SENSITIVITY ANALYSIS ILLUSTRATED[3]

The term *sensitivity analysis* refers to an analysis of the effect on the optimal solution to a linear programming problem of changes in the input-output coefficients, cost coefficients, and constant terms. We shall discuss these effects in terms of an illustration. The reader will find no difficulty in extending the results to the general case.

Consider the product mix problem as stated in § 3-5-(6):

$$\begin{array}{llr}
(1) & 4x_1' + 9x_2' + 7x_3' + 10x_4' + x_5' \qquad = 6000 & \text{Carpentry Shop} \\
& x_1' + x_2' + 3x_3' + 40x_4' \qquad + x_6' = 4000 & \text{Finishing Dept.} \\
& -12x_1' - 20x_2' - 18x_3' - 40x_4' \qquad = z' \text{ (Min)} &
\end{array}$$

For computational ease, let us scale the problem so that the production activities are stated in units of 1,000's of desks, the capacities in units of 1,000's of hours, and cost in units of $1,000. Letting $x_j' = 1{,}000x_j$, $z' = 1{,}000z$, system (1) in simplex tableau form, becomes (2).

[3] This section was contributed by W. O. Blattner.

(2)

Basic Variables	Admissible Variables (Including Slacks)							Constants
	x_1	x_2	x_3	x_4	x_5	x_6	$-z$	
x_5	4	9	7	10	1	0	0	6
x_6	1	1	3	40	0	1	0	4
$-z$	−12	−20	−18	−40	0	0	1	0
					●	●	●	

Since this is already in canonical form, addition of artificial variables is unnecessary, and we can proceed directly with Phase II of the simplex method. After several cycles we arrive at the optimum solution (3).

(3)

Basic Variables	Admissible Variables (Including Slacks)							Constants
	x_1	x_2	x_3	x_4	x_5	x_6	$-z$	
x_1	1	7/3	5/3	0	4/15	−1/15	0	4/3
x_4	0	−1/30	1/30	1	−1/150	2/75	0	1/15
$-z$	0	20/3	10/3	0	44/15	4/15	1	56/3
	●			●			●	

From the information contained in this tableau (3) we see that the optimum product mix for the problem as stated is at the rate of $1{,}333\frac{1}{3}$ desks of type 1 and $66\frac{2}{3}$ desks of type 4 per time period for a total profit rate of $z' = \$18{,}667$ per period.

In addition to giving us the point of most profitable operation, it is possible to obtain from the optimum tableau a wealth of information concerning a wide range of operations in the neighborhood of this optimum by making a sensitivity analysis. In many applications, the information thus obtained is as valuable as the specification of the optimum solution itself.

Sensitivity analysis is important for several reasons:

(a) Stability of the optimum solution under changes of parameters may be critical. For example, using the old optimum solution point, a slight variation of a parameter in one direction may result in a large unfavorable difference in the objective function relative to the new minimum, while a large variation in the parameter in another direction may result in only a small difference. In an industrial situation where there are certain inherent variabilities in processes and materials not taken account of in the model, it may be desirable to move away from the optimum solution in order to achieve a solution less likely to require essential modification.

(b) Values of the input-output coefficients, objective function coefficients, and/or constraint constants may be to some extent *controllable*, and in this case we want to know the effects which would result from changing these values.

(c) Even though the input-output and objective function coefficients and constraint constants are *not* controllable, the estimates for their values may be only approximate, making it important to know for what ranges of these values the solution is still optimum. If it turns out that the optimum solution is extremely sensitive to their values, it may become necessary to obtain better estimates.

Optimality Range for Cost Coefficients: Non-Basic Activities.

Problem 1: A new desk called Type 7 has been designed which will require 6 man hours of Carpentry Shop time and 2 man hours of Finishing Department labor per desk. Based on an estimated profit of $18 per desk, would it pay to produce this desk?

Note that the negatives of the values of the simplex multipliers, $\frac{44}{15}$, $\frac{4}{15}$, 1, for the last iteration can be obtained from the bottom row vector of the inverse of the final basis, and it is obvious that these are simply the updated relative cost factors corresponding to the initial basic variables x_5, x_6, $(-z)$. This yields, after pricing out, $\bar{c}_7 = \frac{44}{15}(6) + \frac{4}{15}(2) - 18 = \frac{2}{15}$. Since $\bar{c}_7 > 0$, it does *not* pay to produce this desk.

Problem 2: How much would the profit for desk Type 7 have to change before it becomes worthwhile to produce?

In order for a non-basic activity to be a candidate to enter the basis, its relative cost factor must be ≤ 0. If any non-basic activity has a relative cost factor exactly equal to zero it can be brought into the basis without changing the value of the objective function. Therefore, if the cost coefficient c_j for any non-basic activity is decreased by the value of its relative cost factor $\bar{c}_j$ in the optimum solution, it becomes a candidate to enter the basis. In this case, desk Type 7 becomes a candidate for production if its *profit* per unit can be increased by $\bar{c}_7 = \frac{2}{15}$.

EXERCISE: How much must the profit on desk Type 2 be increased to bring it into an optimum solution? How much would you have to raise the selling price on desk Type 3 in order to make its production profitable, assuming that all desks of Type 3 produced can be sold at the new price? How would you modify the model if the amount that can be sold is a function of selling price? See Chapter 18; see also Chapter 24.

Effect of Changing Input-Output Coefficients: Non-Basic Activities.

Problem 3: How much would the Carpentry Shop labor requirement for desk Type 7 have to change for it to become profitable to produce?

For non-basic activities the effect of changing the input-output coefficients or a_{ij} in the initial tableau can be easily calculated using the negative of the simplex multipliers. Replacing the original value of $a^o_{17} = 6$ by the parameter $a_{1,7}$ in the $\bar{c}_j$ calculation, we have

$$\bar{c}_7 = [\tfrac{44}{15} \quad \tfrac{4}{15} \quad 1] \cdot \begin{bmatrix} a_{17} \\ 2 \\ -18 \end{bmatrix} = \tfrac{44}{15} a_{1,7} - \tfrac{262}{15} \tag{4}$$

In order for activity $j = 7$ to become a candidate to enter the solution, $\bar{c}_7$ must be ≤ 0 or $a_{17} \leq \frac{131}{22}$.

EXERCISE: To what value would the Carpentry Shop hours for desk Type 2 have to be reduced to make it competitive?

Problem 4: Suppose that we are not really sure of either the labor requirements *or* profit for desk Type 2. Give a formula for these parameters that may be used to determine if it is profitable to produce desk Type 2.

For activity $j = 2$ to become a candidate to enter the solution:

$$\bar{c}_2 = [\tfrac{44}{15} \quad \tfrac{4}{15} \quad 1] \cdot \begin{bmatrix} a_{12} \\ a_{22} \\ c_2 \end{bmatrix} = \tfrac{44}{15} a_{12} + \tfrac{4}{15} a_{22} + c_2 \leq 0$$

If, for example, it turns out that $a_{12} = 8$, $a_{22} = 2$, $c_2 = -25$, substitution in the above formula gives $\bar{c}_2 = -1$ so that it pays to produce desk Type 2.

The Substitution Effect of Non-Basic Activities on Basic Activities.

Problem 5: How many units of the entering activity $j = s$ can be brought into the solution and what will be the effect upon the quantities of the other basic activities? Here again the answer is given by the simplex method:

$$\text{Max } x_s = \underset{\bar{a}_{is} > 0}{\text{Min}} \left(\frac{\bar{b}_i}{\bar{a}_{is}} \right) \tag{5}$$

Let us review the information directly available from the optimum tableau (3). The rows express the basic variables in terms of the non-basic variables, while the columns express the non-basic activities in terms of the basic activities. The column vector of matrix coefficients, $\bar{a}_{ij}$, under each variable x_j can be interpreted as "substitution" factors. For example, for each unit of activity $j = 2$ we bring into the solution we must remove $\frac{7}{3}$ units of basic activity $j = 1$ and add $\frac{1}{30}$ unit of basic activity $j = 4$ for a resulting net increase of $\frac{20}{3}$ units in the objective function. From (5), when (Max x_2) units of activity $j = 2$ are brought into the solution, then the corresponding i^{th} basic activity drops out.

Observe that the relative cost factor $\bar{c}_j$ for each variable can be calculated from the "substitution" factors, $\bar{a}_{ij}$, and the original cost coefficients c_j; namely

$$\bar{c}_j = c_j - \sum_i \bar{a}_{ij} c_{j_i} \tag{6}$$

where j_i is the index of the i^{th} basic activity. This is an alternative way to do the "pricing out" calculations. For each unit of $j = 2$ added, the following changes in the basic variables result:

(7)

Variable	Quantity Change		Cost per Unit		Cost Change
x_1:	$(-\frac{7}{3})$	·	(-12)	$=$	$+28$
x_4:	$(+\frac{1}{30})$	·	(-40)	$=$	$-\frac{4}{3}$
x_2:	$(+1)$	·	(-20)	$=$	-20

Relative cost factor: $\bar{c}_2 = +\frac{20}{3}$ per unit of x_2 introduced

EXERCISE: If the profit on desk Type 2 is increased by exactly $\bar{c}_2 = \frac{20}{3}$ per desk, show that up to $571\frac{3}{7}$ desks of Type 2 can be produced per period without reducing total profit. What is the resulting product mix?

Effect of Changing Constraint Constants.

Problem 6: What is the effect of increasing Finishing Department capacity?

An increase in the value of a constraint constant can be considered equivalent to introducing a fictitious procurement activity with coefficients equal to but opposite in sign to the corresponding slack variable in the original formulation (2). It follows that the effect of increasing Finishing Department capacity is to *increase* net profit by $\$\frac{4}{15}$ per hour of increase, up to 20,000 hours increase in the period, because the value of $\bar{c}_6 = \frac{4}{15}$ for the corresponding slack variable.

EXERCISE: If Finishing Department capacity is increased by 20,000 hours per time period, what is the resulting product mix? Which basic activity has dropped out of the solution?

EXERCISE: Necessary equipment to increase the capacity of the Carpentry Shop by 10 per cent can be rented for $5,000. Also, overtime hours up to 20 per cent of the rated capacity of either shop can be obtained at a premium of $1.50 per hour. Above this figure the premium is estimated to be about $3.00 per hour because of loss of productive efficiency. What would you do?

EXERCISE: Show that if a slack variable is in the basic set with value $\bar{x}_j$ in the optimum solution, then the corresponding constraint constant b_i in the initial tableau can take on any value $b_i \geq b_i^0 - \bar{x}_j$ where b_i^0 was its original value, with no change in the values of the objective function or the other basic variables in the optimum solution. In this range, is b_i actually constraining the solution?

Optimality Range for Cost Coefficients: Basic Activities.

Problem 7: For what range of costs of the basic activities does the present optimum solution still remain optimum?

In particular, consider basic activity $j = 1$. The present solution will remain optimal until cost coefficient c_1 is increased or decreased sufficiently so that the relative cost factor on one of the non-basic activities goes to zero, at which point that non-basic activity becomes a candidate to enter the solution.

Referring to (6) we have

$$\bar{c}_2 = -20 - [+\tfrac{7}{3}(c_1) - \tfrac{1}{30}(-40)] = -\tfrac{7}{3}c_1 - \tfrac{64}{3} \tag{8}$$

We see that the value of the cost coefficient c_1 of $-(64/3)/(7/3)$ is required to make the relative cost factor $\bar{c}_2 = 0$. We must, however, also investigate the effect of a c_1 change on all *other* non-basic activities; we find in general that the *change* in c_1 necessary to make the value of $\bar{c}_j = 0$ is given by

$$c_1 - c_1^o = \frac{\bar{c}_j^o}{\bar{a}_{1j}} \qquad (c_1^o = -12) \tag{9}$$

Hence,

$$\bar{c}_2 = 0 \text{ if } c_1 \text{ is increased by } \frac{\frac{20}{3}}{\frac{7}{3}} = \tfrac{20}{7}$$

$$\bar{c}_3 = 0 \text{ if } c_1 \text{ is increased by } \frac{\frac{10}{3}}{\frac{5}{3}} = 2$$

$$\bar{c}_5 = 0 \text{ if } c_1 \text{ is increased by } \frac{\frac{44}{15}}{\frac{4}{15}} = 11$$

$$\bar{c}_6 = 0 \text{ if } c_1 \text{ is increased by } \frac{\frac{4}{15}}{-\frac{1}{15}} = -4$$

Thus the present solution (3) is still optimal, for $-12 - 4 \leq c_1 \leq -12 + 2$ where $c_1^o = -12$ was the original value of c_1. The computational rule can be summarized:

$$\text{Max } c_{j_i} = c_{j_i}^o + \underset{\bar{a}_{ij}>0}{\text{Min}} \left(\frac{\bar{c}_j}{\bar{a}_{ij}}\right) \qquad j \text{ non-basic} \tag{10}$$

$$\text{Min } c_{j_i} = c_{j_i}^o + \underset{\bar{a}_{ij}<0}{\text{Max}} \left(\frac{\bar{c}_j}{\bar{a}_{ij}}\right) \qquad j \text{ non-basic}$$

Exercise: For what range of profit for desk Type 4 is the present solution (3) still optimal? Determine what activity enters the solution if c_1 is decreased to -20, increased to $-\frac{19}{2}$. What activity leaves the solution in each case?

Exercise: Construct an example by changing b_1 in the original problem (2) to show that if c_1 is increased to $-\frac{19}{2}$, desk Type 1 is not necessarily the activity that leaves the solution.

EXERCISE: Prove that if the cost of a basic activity is reduced, it will not be dropped from the optimum solution.

EXERCISE: The reason for the Irish Rebellion. The average Irishman has 27 pence per day to spend on food and requires a diet of 2,000 calories to live. Irish potatoes cost 3 pence per 1,000 calories and meat costs 24 pence per 1,000 calories. Since he detests potatoes, he eats 1,000 calories of meat and 1,000 calories of potatoes per day. Show that if the price of potatoes goes up to 10 pence per 1,000 calories (which will, of course, be blamed on the English), the average Irishman must increase his daily potato consumption by 50 per cent.

Effect of Changing Input-Output Coefficients: Basic Activities.

Problem 8: What happens if the Carpentry Shop requirements for desk Type 1 change? Changing an input-output coefficient for a basic activity results in changes to the negatives of the simplex multipliers and other elements of the updated inverse of the initial basis. To evaluate the effect of varying a_{11} we must find:

$$[\tilde{B}]^{-1} = \begin{bmatrix} a_{11} & 10 & 0 \\ 1 & 40 & 0 \\ -12 & -40 & 1 \end{bmatrix}^{-1} \tag{11}$$

already knowing that

$$[B]^{-1} = \begin{bmatrix} 4 & 10 & 0 \\ 1 & 40 & 0 \\ -12 & -40 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} \frac{4}{15} & -\frac{1}{15} & 0 \\ -\frac{1}{150} & \frac{2}{75} & 0 \\ \frac{44}{15} & \frac{4}{15} & 1 \end{bmatrix} \tag{12}$$

We can write $[\tilde{B}]$ as $[A] \cdot [B]$, where the matrix

$$[A] = [\tilde{B}] \cdot [B]^{-1} = \begin{bmatrix} \dfrac{4a_{11} - 1}{15} & \dfrac{4 - a_{11}}{15} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{13}$$

Since $[A]$ is an elementary matrix (see § 8-5) we can easily find its inverse. Let the reciprocal of the diagonal element in the first row be γ, thus

$$\gamma = \frac{15}{4a_{11} - 1} \quad \text{and} \quad a_{11} = \frac{15 + \gamma}{4\gamma} \tag{14}$$

We shall show that $[A]^{-1}$ and hence $[\tilde{B}]^{-1}$ are *linear* in γ.

$$[A] = \begin{bmatrix} \frac{1}{\gamma} & \left(\frac{1}{4} - \frac{1}{4\gamma}\right) & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad [A]^{-1} = \begin{bmatrix} \gamma & \dfrac{1 - \gamma}{4} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{15}$$

Then

$$(16)\qquad [\bar{B}]^{-1} = [[A][B]]^{-1} = [B]^{-1}\cdot[A]^{-1} = \begin{bmatrix} \frac{4\gamma}{15} & \frac{-\gamma}{15} & 0 \\ \frac{-\gamma}{150} & \frac{\gamma+15}{600} & 0 \\ \frac{44\gamma}{15} & \frac{15-11\gamma}{15} & 1 \end{bmatrix}$$

where by (14) $\gamma^o = 1$ corresponds to $a_{11}^o = 4$. Note that in order for $[\bar{B}]$ to be a basis, $[\bar{B}]^{-1}$ must exist. This means that $[A]^{-1}$ must exist which requires $1/\gamma \neq 0$, $\gamma \neq 0$. We shall show later that for feasibility $\gamma \geq 0$. Together these two restrictions require $\gamma > 0$. The question now becomes one of finding for what range of γ the present basic activities are still optimal. We first determine what values of γ will insure $\bar{c}_j \geq 0$ for each of the non-basic activities.

$$(17)\quad \bar{c}_2 = \begin{bmatrix} \frac{44\gamma}{15} & \frac{15-11\gamma}{15} & 1 \end{bmatrix} \begin{bmatrix} 9 \\ 1 \\ -20 \end{bmatrix} = \frac{77\gamma}{3} - 19 \geq 0 \text{ if } \gamma \geq \frac{57}{77}$$

$$\bar{c}_3 = \begin{bmatrix} \frac{44\gamma}{15} & \frac{15-11\gamma}{15} & 1 \end{bmatrix} \begin{bmatrix} 7 \\ 3 \\ -18 \end{bmatrix} = \frac{55\gamma}{3} - 15 \geq 0 \text{ if } \gamma \geq \frac{9}{11}$$

$$\bar{c}_5 = \begin{bmatrix} \frac{44\gamma}{15} & \frac{15-11\gamma}{15} & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \frac{44\gamma}{15} \geq 0 \text{ if } \gamma \geq 0$$

$$\bar{c}_6 = \begin{bmatrix} \frac{44\gamma}{15} & \frac{15-11\gamma}{15} & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = -\frac{11\gamma}{15} + 1 \geq 0 \text{ if } \gamma \leq \frac{15}{11}$$

In order to maintain *feasibility* we must *also* determine what range of values of γ will maintain each of the elements $\bar{b}_i \geq 0$:

$$(18)\qquad \begin{bmatrix} \bar{b}_1 \\ \bar{b}_2 \\ -z \end{bmatrix} = \begin{bmatrix} \frac{4\gamma}{15} & -\frac{\gamma}{15} & 0 \\ -\frac{\gamma}{150} & \frac{\gamma+15}{600} & 0 \\ \frac{44\gamma}{15} & \frac{15-11\gamma}{15} & 1 \end{bmatrix} \begin{bmatrix} 6 \\ 4 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{4\gamma}{3} \\ \frac{3-\gamma}{30} \\ \frac{12+44\gamma}{3} \end{bmatrix}$$

From this we see that $\bar{b}_1 \geq 0$ if $\gamma \geq 0$; and $\bar{b}_2 \geq 0$ if $\gamma \leq 3$. Noting that the first row of $[\bar{B}]^{-1}$ is simply γ times the first row of $[B]^{-1}$, we see that $\bar{b}_1 = \gamma \bar{b}_1^o$.

Assuming nondegeneracy, $\bar{b}_1^o > 0$. Then the requirement $\bar{b}_1 \geq 0$ requires $\gamma \geq 0$. Since $\gamma \neq 0$, $\gamma > 0$ and $\bar{b}_1 > 0$. Taking the most restrictive of the γ calculated by (17) and (18) above we find that the objective function z, and the values, $\bar{b}_i$, of the basic variables are all *linear* in γ and the adjusted basic solution is feasible and optimal for the range $\frac{9}{11} \leq \gamma \leq \frac{15}{11}$ or $\frac{29}{6} \geq a_{11} \geq 3$.

EXERCISE: What is the effect of increasing Carpentry Shop time on desk Type 1 to $4\frac{1}{2}$ hours per desk? To 5 hours per desk?

EXERCISE: Under what conditions can the value of an input-output coefficient for a basic activity be changed without any cost effect?

Problem 9: What is the effect of simultaneous changes to Carpentry Shop and Finishing Department requirements and profit for desk Type 1? Here the problem is to find

$$[\tilde{B}']^{-1} = \begin{bmatrix} a_{11} & 10 & 0 \\ a_{21} & 40 & 0 \\ c_1 & -40 & 1 \end{bmatrix}^{-1} \tag{19}$$

Writing $[\tilde{B}']$ this time as $[B] \cdot [A']$ we have

$$[A'] = [B]^{-1} \cdot [\tilde{B}'] = \begin{bmatrix} \dfrac{4a_{11} - a_{21}}{15} & 0 & 0 \\ \dfrac{4a_{21} - a_{11}}{150} & 1 & 0 \\ \dfrac{44a_{11} + 4a_{21} + 15c_1}{15} & 0 & 1 \end{bmatrix} \tag{20}$$

Letting $\dfrac{4a_{11} - a_{21}}{15} = p$, $\dfrac{4a_{21} - a_{11}}{150} = q$, and $\dfrac{44a_{11} + 4a_{21} + 15c_1}{15} = r$

$$[A']^{-1} = \begin{bmatrix} \dfrac{1}{p} & 0 & 0 \\ \dfrac{-q}{p} & 1 & 0 \\ \dfrac{-r}{p} & 0 & 1 \end{bmatrix} \tag{21}$$

and

$$[\tilde{B}']^{-1} = [A']^{-1}[B]^{-1} = \begin{bmatrix} \dfrac{4}{15p} & \dfrac{-1}{15p} & 0 \\ \dfrac{-p - 40q}{150p} & \dfrac{2p + 5q}{75p} & 0 \\ \dfrac{44p - 4r}{15p} & \dfrac{4p + r}{15p} & 0 \end{bmatrix} \tag{22}$$

where, as before, the pivot element $p \neq 0$ and $1/p \neq 0$.

Observe that $p^o = 1$, $q^o = r^o = 0$ corresponds to $a_{11}^o = 4$, $a_{21}^o = 1$, $c_1^o = -12$. We now determine what values of p, q, r will insure $\bar{c}_j \geq 0$ for each of the non-basic activities,

$$(23)\quad [\bar{c}_2] = \begin{bmatrix} \dfrac{44p - 4r}{15p} & \dfrac{4p + r}{15p} & 1 \end{bmatrix} \begin{bmatrix} 9 \\ 1 \\ -20 \end{bmatrix} = \frac{20}{3} - \frac{7r}{3p} \geq 0 \text{ if } \frac{r}{p} \leq \frac{20}{7}$$

$$[\bar{c}_3] = \begin{bmatrix} \dfrac{44p - 4r}{15p} & \dfrac{4p + r}{15p} & 1 \end{bmatrix} \begin{bmatrix} 7 \\ 3 \\ -18 \end{bmatrix} = \frac{10}{3} - \frac{5r}{3p} \geq 0 \text{ if } \frac{r}{p} \leq 2$$

$$[\bar{c}_5] = \begin{bmatrix} \dfrac{44p - 4r}{15p} & \dfrac{4p + r}{15p} & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \frac{44}{15} - \frac{4r}{15p} \geq 0 \text{ if } \frac{r}{p} \leq 11$$

$$[\bar{c}_6] = \begin{bmatrix} \dfrac{44p - 4r}{15p} & \dfrac{4p + r}{15p} & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \frac{4}{15} + \frac{r}{15p} \geq 0 \text{ if } \frac{r}{p} \geq -4$$

and also maintain each of the elements $\bar{b}_i \geq 0$:

$$(24)\quad \begin{bmatrix} \bar{b}_1 \\ \bar{b}_2 \\ -z \end{bmatrix} = \begin{bmatrix} \dfrac{4}{15p} & \dfrac{-1}{15p} & 0 \\ \dfrac{-p - 40q}{150p} & \dfrac{2p + 5q}{75p} & 0 \\ \dfrac{44p - 4r}{15p} & \dfrac{4p + r}{15p} & 1 \end{bmatrix} \begin{bmatrix} 6 \\ 4 \\ 0 \end{bmatrix} = \begin{bmatrix} \dfrac{4}{3p} \\ \dfrac{1}{15} - \dfrac{4q}{3p} \\ \dfrac{56}{3} - \dfrac{4r}{3p} \end{bmatrix}$$

whence $\bar{b}_1 \geq 0$ if $\frac{1}{p} \geq 0$, $\bar{b}_2 \geq 0$ if $\frac{q}{p} \leq \frac{1}{20}$, and the value of the objective function $z = -\frac{56}{3} + \frac{4r}{3p}$. Note that assuming nondegeneracy as before, $\bar{b}_1 = \frac{1}{p}\bar{b}_1^o > 0$ and $\frac{1}{p} > 0$.

Taking the most restrictive of the limits calculated above, we see that z and the $\bar{b}_i$ are *linear* in $\frac{1}{p}$, $\frac{q}{p}$, and $\frac{r}{p}$ for the range $\frac{1}{p} > 0$, $\frac{q}{p} \leq \frac{1}{20}$, and $2 \geq \frac{r}{p} \geq -4$. These restrictions are equivalent to $p > 0$, $p \geq 20q$, and $2p \geq r \geq -4p$, which upon substitution yield the conditions on the input-output coefficients and cost coefficients of basic activity $j = 1$ for which the basic variables given in (3) remain in the optimum solution:

(25)
$$4a_{11} - a_{21} \geq 0$$
$$2a_{11} - 3a_{21} \geq 0$$
$$-\frac{12a_{11} + 2a_{21}}{5} \geq c_1 \geq -4a_{11}$$

The effect of varying the a_{i1} and c_1 within these limits upon the $\bar{b}_i$ and z of the optimum solution is given by (24).

EXERCISE: Show that if a_{11} and a_{21} are varied within the limits of the first two inequalities in (25) and if c_1 is correspondingly varied so that $c_1 = -\frac{44a_{11} + 4a_{21}}{15}$, then there will be no change in the value of the objective function.

EXERCISE: Prove the following:

THEOREM: *Given a general linear program, the domain of all possible variations of a column P in an optimal feasible basis is convex in the space of the components of P.*

12-5. PROBLEMS

1. Show in § 12-1 that if, in an optimal solution, there are surpluses of certain items, their prices are zero. Show that the price of an item is zero if there is no cost associated with the activity of storing it and there is a surplus of the item.
2. Show that the above case might lead to excessive use of the raw material inputs, unless the central planners place some value on excess raw material in terms of labor.
3. Show that it would be better to also introduce activities for procurement of additional inputs and to place a labor value on these as well.
4. (Review) Show that the price π_i represents the change in the total costs z per infinitesimal change in the availability of item i.
5. Which of the various properties associated with the duality theorems of Chapter 6 explains why the manager of the tool plant discovered the process which minimizes his labor requirements in the course of developing a pricing system?
6. Given an optimal basic feasible solution and the corresponding system in canonical form, show that $\bar{c}_j$ represents the change necessary in the unit cost of the j^{th} non-basic activity before it would be a candidate to enter the basis. If the other coefficients as well as cost coefficients can vary, show that
$$\bar{c}_j = c_j - \sum_i \pi_i a_{ij}$$
is the amount of change where π_i are the prices associated with the basic set of variables.

7. Develop a formula for the change in cost c_j of a basis activity before it is a candidate for being dropped from the basis. Which activity would enter in its place?

REFERENCES

Allen, 1956-1
Arrow, 1951-1
Arrow, Karlin, and Suppes, 1960-1
Baumol, 1959-1, 1961-1
Beckmann, 1955-1
Boulding and Spivey, 1960-1
Dantzig, 1955-2
Dorfman, 1951-1
Dorfman, Samuelson, and Solow, 1958-1
Fisher and Schruben, 1953-1
Frisch, 1954-1
Gale, 1960-1
Gale, Kuhn, and Tucker, 1950-1, 2, 1951-1
Hicks, 1960-1
Karlin, 1959-1
Koopmans, 1951-1, 2, 1954-1
Leontief, 1951-1
Massé, 1946-1
Massé and Gibrat, 1957-1
Mills, 1956-1
Morgenstern, 1954-1
Solow, 1952-1
Uzawa, 1960-1
von Neumann, 1937-1
Wagner, 1957-2
Wald, 1935-1
Walras, 1874-1

CHAPTER 13

GAMES AND LINEAR PROGRAMS

13-1. MATRIX GAMES

Background.[1]

According to Gale [1960-1, p. 216], "One of the most striking events in connection with the emergence of modern linear economic model theory was the simultaneous but independent development of linear programming on the one hand and game theory on the other, and the eventual realization of the very close relationship that exists between these two subjects."

This relationship between linear programming and games was first pointed out by J. von Neumann in the fall of 1947 in informal discussions with the author. He showed that the central mathematical problem associated with a matrix game could be stated as a linear program and he conjectured that the converse was true. A. W. Tucker and his group at Princeton in early 1948 undertook a systematic study of the interrelations between the two fields in order to place the theory on a rigorous foundation [Gale, Kuhn, and Tucker, 1951-1]. It is the purpose of this chapter to bring out these connections.

Game theory is concerned with finding the best "strategies" for solving conflict situations. In the abstract these may be characterized as situations where the participants of the contest each control some but not all the actions that can take place. This, together with chance events (if present), determines the outcome upon which the participants may place widely differing values. The mathematical foundations of game theory are found in certain papers by J. von Neumann in 1928 and 1937 and less conclusive contributions by Borel in the early 1920's [von Neumann, 1928-1, 1937-1; Borel, 1921-1, 1924-1, and 1927-1]. Actually, until 1944 there were almost no papers; it was then that von Neumann and Morgenstern [1944-1] published their famous book, *Theory of Games and Economic Behavior*.

Matrix Games Defined.

In a matrix game there are two players whom we will refer to as "R" and "C." Each is supposed to make one choice from a set of possible choices

[1] For a popular explanation of Game Theory, the reader is referred to *The Compleat Strategyst* by John D. Williams [Williams, 1954-1]; a more formal introduction can be found in Luce and Raiffa, *Games and Decisions* [Luce and Raiffa, 1957-1] and in M. Dresher, *Games of Strategy: Theory and Applications* [Dresher, 1961-1]. Fundamental papers on the subject can be found in *Annals of Mathematics Studies*, Nos. 24 (1950) and 28 (1953) entitled "Contributions to the Theory of Games," edited by Kuhn and Tucker, [Kuhn and Tucker, 1950-1, and 1953-1].

without knowledge as to the choice of the other player. For each such choice by the two players, say i for R and j for C, there is a resulting outcome that can be specified by a single number a_{ij}, the *payoff* R stands to receive (positive, negative, or zero) according as R wins from C, loses, or draws; at the same time, C is in *diametric opposition* to R and stands to receive what R loses—namely $-a_{ij}$. If choices i for R range from $i = 1, 2, \ldots, m$ and choices j for C from $j = 1, 2, \ldots, n$, the possible payoff to R may be displayed as a matrix:

Payoff to R

if R chooses row i and C chooses column j

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

The problem for each player is this: What choice should he make in order that his partial influence over the outcome benefits him the most?

Our immediate purpose is to show that a variety of competitive situations can be cast into the form of a matrix game if we interpret "choice" in these cases to mean a selection among the "pure strategies" available to each player.

DEFINITION: *A pure strategy* is a plan so complete that it cannot be upset by the opponent or nature [Williams, 1954-1]. It is a complete set of advance instructions that specifies a definite choice for every conceivable situation in which the player may be required to act [Kuhn and Tucker, 1955-1].

Example 1, A Calling Game: Player R has two cards, one black and one red. He selects one. Without showing it to his opponent, he lays it down on the table. Player C then calls it. The card is turned over. If the card is called, R pays C a penny, otherwise he loses a penny. The pure strategies open to R are listed in the left *row* margin of the two-way array (1); we refer to R for this reason as the *row player*. In a similar manner, those for C are listed in the *column* margin, shown across the top, and C is called the *column player*. The table entries are the payoff to R if C calls the card correctly or incorrectly.

(1)

Player R's Pure Strategies	Player C's Pure Strategies	
	Call Black	Call Red
Choose black	−1	+1
Choose red	+1	−1

This matrix game points up how important it is to both R and C that their plans not be discovered in advance. The result could be disastrous. No matter how R chooses, his highest "floor," the maximum gain that R can guarantee himself, if he is discovered, is −1. Similarly, C's lowest "ceiling," the minimum loss that C can guarantee himself, if he is discovered, is +1. We shall show, however, that there is a way to play this game so that R can have an expected gain of 0, and C can have an expected loss of 0.

Example 2, Morra [Williams, 1954-1]: Two players simultaneously throw out one or two fingers and call out their guess as to what the total sum of the outstretched fingers will be. If a player guesses right, but his opponent does not, he receives payment equal to his guess. In all other cases, it is a draw.

The pure strategies open to each player are show 1, call 2; show 1, call 3; show 2, call 3; show 2, call 4. If we abbreviate these combinations (1, 2), (1, 3), etc., the matrix game takes the (skew-symmetric) form

(2)

		C (1, 2)	(1, 3)	(2, 3)	(2, 4)
R	(1, 2)	0	2	−3	0
	(1, 3)	−2	0	0	3
	(2, 3)	3	0	0	−4
	(2, 4)	0	−3	4	0

The maximum floor for R, if his pure strategy is discovered, is −2 and similarly the minimum ceiling for C is +2. Again we see how important it is not to reveal the pure strategy in advance.

Example 3, The Campers: John D. Williams [1954-1] in his humorous elementary introduction to game theory, *The Compleat Strategyst*, supplies the following example and discussion, which we quote with minor changes.[2]

"It may help to fix these ideas if we give a specific physical realization. When the payoffs are all positive, we may interpret them as the altitudes of points in a mountainous region. The various R and C strategies then correspond to the latitudes and longitudes of these points.

"To supply some actors and motivation for a game, let's suppose that a man and wife—Ray and Carol—are planning a camping trip, and that Ray likes high altitudes and Carol likes low altitudes.[3] The region of interest to them is criss-crossed by a network of fire divides, or roads, four running in each direction. The campers have agreed to camp at a road junction.

[2] We have changed the actress' name to Carol to correspond to C and have deleted a few phrases.

[3] Williams implicitly assumes that the interests of the man and wife, as far as altitude is concerned, are diametrically opposed and the "payoff" to R can be measured in feet of altitude.

They have further agreed that Ray will choose the east-west road and that Carol will choose the north-south road, which jointly identify the junction.

"Let us suppose the junctions on the roads i available to Ray and roads j available to Carol have the following altitudes, in thousands of feet:

(3)

	Carol (j)	(1)	(2)	(3)	(4)
Ray (i)	(1)	7	2	5	1
	(2)	2	2	3	4
	(3)	5	3	4	4
	(4)	3	2	1	6

Ray, being a reasonable person, who simply wants to have as much as possible, is naturally attracted to the road Ray 1—with junctions at altitudes of 7, 2 5, and 1—for it alone can get him the 7-thousand-foot peak. However, he does not dare undertake a plan which would realize him a great deal if it succeeds, but which would lead to disaster if Carol is skillful in her choice. Not anticipating that she will choose carelessly, his own interests compel him to ignore the peaks; instead, he must attend particularly to the sinks and lows which blemish the region. This study leads him finally to the road Ray 3, which has as attractive a low as the region affords, namely, one at an altitude of 3-thousand feet. By choosing Ray 3, he can ensure that the camp site will be at least 3-thousand feet up; it will be higher, if Carol is a little careless.

"His wife—as he feared—is just as bright about these matters as he is. As she examines these, she knows better than to waste time mooning over the deep valleys of Carol 3 and Carol 4, much as she would like to camp there. Being a realist, she examines the peaks which occur on her roads, determined to choose a road which contains only little ones. She is thus led, finally, to Carol 2, where a 3-thousand-foot camp site is the worst that can be inflicted on her.

"We now note that something in the nature of a coincidence has occurred. Ray has a strategy (Ray 3) which guarantees that the camp site will have an altitude of 3-thousand feet or more, and Carol has one (Carol 2) which ensures that it will be 3-thousand feet or less. In other words, either player can get a 3-thousand-foot camp site by his own efforts, in the face of a skillful opponent; and he will do somewhat better than this if his opponent is careless."

DEFINITION: When the guaranteed maximum floor for R and the minimum ceiling for C are exactly equal (as they are above), the game is said to have a *saddle point* and is also called a *strictly determined game* because the players should use the strategies which correspond to it. If either alone departs from the saddle-point strategy, he will suffer unnecessary loss. If both depart from it, the situation becomes completely fluid and someone will suffer. Note too, this consequence of having a saddle point: security measures are not strictly necessary. Either can openly announce a choice (if it is the proper one), and the other will be unable to exploit the information and force the other beyond the guaranteed maximum floor or minimum ceiling.

EXERCISE: Show for a general matrix game that the maximum floor for R is less than or equal to the minimum ceiling for C.

Example 4, Chance and Bluffing: A deck consisting of one Ace, one King, one Queen is shuffled and the three cards are dealt face down one to each player and the last card placed in the middle of the table. R inspects his card and can either bet an amount b (including the ante) that he has the higher card or he can "fold." In the latter case, he loses an amount a, the ante, to his opponent. Player C can then inspect the card in the middle of the table and can either call the bet or he can fold, in which case he loses an amount a. If C calls, the player with the higher card receives the amount b from his opponent

The pure strategies for R are four in number (if we assume he always bets if he has an Ace). For example, he might state (to himself) two of his four complete alternative plans this way:

			Abbreviation
Pure Strategy	(1)	"If I receive Ace, I will bet." " " King, " " " " Queen, " "	(b, b, b)
Pure Strategy	(2)	" " Ace, " " " " King, " " " " Queen, I will fold."	(b, b, f)

The pure strategies for C depend on whether R folds *or*, if he does not, on the outcome of the middle card. He has eight pure strategies, two of which he might state (to himself) as follows:

		Abbreviation
(1)	"If R folds, I will collect ante." "If R bets, and middle card is Ace, I will call." " " " " King, " " " " " " Queen, " "	(b, b, b)
(2)	"If R folds, I will collect ante." "If R bets, and middle card is Ace, I will call." " " " " King, " " " " " " Queen, I will fold."	(b, b, f)

Let us now turn to the payoff under the pure strategy (b, b, b) for R and (b, b, b) for C. We examine the payoffs in the six cases arising out of the random shuffle.

R receives	Middle Card	Probability	The Play R	The Play C	The Payoff
Ace,	King	$\frac{1}{6}$	bet	bet	b
Ace,	Queen	$\frac{1}{6}$	bet	bet	b
King,	Ace	$\frac{1}{6}$	bet	bet	b
King,	Queen	$\frac{1}{6}$	bet	bet	$-b$
Queen,	Ace	$\frac{1}{6}$	bet	bet	$-b$
Queen,	King	$\frac{1}{6}$	bet	bet	$-b$

Expected Payoff: $\frac{0}{6} = 0$

In this problem, the payoff for given pure strategies for R and C cannot be stated with certainty because of the random elements which are beyond the control of either player. In this situation, the *expected payoff* is defined to be the "payoff" entry in the game matrix.

With this definition the game matrix (multiplied by 6 to avoid fractions) can be calculated to be

(4)

		C							
		(bbb)	(bbf)	(bfb)	(bff)	(fbb)	(fbf)	(ffb)	(fff)
R	(bbb)	0	$2a$	$2a$	$4a$	$2a$	$4a$	$4a$	$6a$
	(bbf)	$2b - 2a$	$2b$	$b - a$	$b + a$	$b - a$	$b + a$	0	$2a$
	(bfb)	$-2a$	$-b - a$	0	$-b + a$	$b - a$	0	$a + b$	$2a$
	(bff)	$2b - 4a$	$b - 3a$	$b - 3a$	$-2a$	$2b - 4a$	$b - 3a$	$b - 3a$	$-2a$

EXERCISE: Verify the entries in the game matrix. Which player has the advantage?

EXERCISE: Introduce the full set of pure strategies for R into the matrix game, i.e., include (fbb), (fbf), (ffb), (fff). Do you feel R's decision not to consider the latter possibilities was a good one?

Mixed Strategies.

As R weighs the consequences of various courses of action open to him in a matrix game, he becomes more aware of the need to keep secret his choice of pure strategy. In situations where there will be many repetitions of the same game, any obvious pattern of choice could lead to disastrous consequences.

If R is *completely pessimistic*, i.e., assumes that C knows his plan and will counter it to limit his gain to a bare minimum, then R will select his

Max-Min pure strategy, namely the row r whose row minimum is maximum. As we have seen in the matrix game

(5)
$$\begin{array}{cc} & (j=1) \quad (j=2) \\ \begin{array}{c}(i=1)\\(i=2)\end{array} & \begin{bmatrix} -1 & +1 \\ +1 & -1 \end{bmatrix} \end{array}$$

this highest floor is -1 attained when R chooses either $(i = 1)$ or $(i = 2)$. If C is also *completely pessimistic*, he will employ his Min-Max pure strategy, namely the column s whose column maximum is minimum, in this case $+1$. The payoff of a matrix game between two complete pessimists is a_{rs} which is somewhere in between R's highest floor and C's lowest ceiling, i.e.,

(6)
$$\operatorname*{Max}_{i}\,(\operatorname*{Min}_{j}\,a_{ij}) \leq a_{rs} \leq \operatorname*{Min}_{j}\,(\operatorname*{Max}_{i}\,a_{ij})$$

When there is a *saddle point* so that Max-Min = Min-Max, the pure strategy choices r, s are completely satisfactory as a solution to the game because whether R is pessimistic and assumes his opponent knows his choice or, if R is pessimistic and it turns out that his opponent is also, R can do no better. In any case, C can *force* R to accept a maximum gain of a_{rs} and if R deviates, he can only lose and if C deviates, R can only gain in general.

When there is *no saddle point*, the contradictory assumptions on the part of the two players lead to an *unstable* situation. In the event of repetitions of the same game with the same players, we can expect sooner or later one of the players will become bold and change his assumptions about his opponent's finding out and taking advantage of his plan. *Depending on what assumptions are made by the opponents about each other's knowledge of their respective plans, there can be, in general, different solutions to the game.*

If R is less than completely pessimistic (we will call him *conservative*) i.e., assumes that C can never be sure what his plan is, but can guess with what *probabilities* he will use one or the other pure strategy, then R will select the row probabilities so that the smallest "average" payoff in a column is a maximum. (For example, in repeated trials of the same game, R may seek to protect his plan by varying his choice of row and is willing to assume that his opponent cannot discover his particular selection, but at best can only detect his frequency of choice of the various rows.)

In the matrix game above, if row 1 is selected by R with probability x_1 and row 2 with probability $1 - x_1$, and if his opponent chooses strategy j, the *expected payoffs* are

(7)
$$\begin{array}{cc} (j=1) & (j=2) \\ -1x_1 + 1(1 - x_1), & 1x_1 - 1(1 - x_1) \end{array}$$

which simplifies to $1 - 2x_1$, and $-1 + 2x_1$. From $0 \leq x_1 \leq \frac{1}{2}$ the right term is the smaller and its largest value 0 occurs at $x_1 = \frac{1}{2}$. From $\frac{1}{2} < x_1 \leq 1$

the left term is the smaller and its largest value 0 also occurs at $x_1 = \frac{1}{2}$. Hence, in this case if R selects his pure strategies with equal probability, and if C should discover these probabilities, then R can assure himself an expected gain of at least 0.

Because of symmetry, it is obvious that if C makes the same kinds of assumptions with regard to his opponent, C can assure himself an expected payoff of no more than 0 if he randomizes his selections by choosing columns $j = 1$ and $j = 2$ with equal probability. Since the expected maximum floor for R equals the expected minimum ceiling for C, it is clear to C that even if he changes his mind and assumes that R is playing conservatively against him, he cannot take advantage and pay less than zero, and may have to pay more if he deviates. On the other hand, C also notes that he may pay less if he sticks to his optimal conservative mix of pure strategy choices and R deviates from his.

Definition: A *mixed strategy* is a selection among pure strategies with fixed probabilities.

Definition: A *conservative* player is one who assumes that his mixed strategy is known to his opponent.

As a second example of the optimal selection of probabilities for a mixed strategy, let us change the value of $a_{42} = 2$ to $a_{42} = 6$ in Williams' example (3) to avoid a saddle-point solution. Let us now consider the simpler problem of determining an optimal mix between two strategies $i = 3$ and $i = 4$ for the matrix

(8)

	$(j = 1)$	$(j = 2)$	$(j = 3)$	$(j = 4)$	Probabilities
$(i = 3)$	5	3	4	4	x_3
$(i = 4)$	3	6	1	6	$1 - x_3$

It is clear that any choice of a pure strategy on the part of R which becomes known to C (say through many repetitions of the same game using the same pure strategy each time) could at best achieve 3 units (strategy $i = 3$) because his opponent will surely choose $j = 2$. On the other hand, if he chooses $i = 3$ with probability x_3 and $i = 4$ with probability $1 - x_3 = x_4$, then his expected payoff for various j of his opponent would be the weighted average of his former payoffs:

(9) $\quad (j = 1) \qquad (j = 2) \qquad (j = 3) \qquad (j = 4)$

$$5x_3 + 3(1 - x_3),\ 3x_3 + 6(1 - x_3),\ 4x_3 + 1(1 - x_3),\ 4x_3 + 6(1 - x_3)$$

If $x_3 = 1$, then his opponent will select road $j = 2$ and his payoff will be 3; if $x_3 = 0$, then his opponent will select road $j = 3$ and his payoff will be 1. However, if $\frac{2}{3} < x_3 < 1$, then the *expected value* for any j will be greater than 3. To determine the best value we plot these expectations for different j strategies as a function of x_3. It is clear that C will choose $j = 3$ for $0 \leq x_3 < \frac{5}{8}$ and will choose $j = 2$ for $\frac{5}{8} < x_3 \leq 1$. For $x_3 = \frac{5}{8}$ he can choose either $j = 2$ or $j = 3$ and the expected payoff to R will be $\frac{7}{2}$ which is the best he can do if C knows his mixed strategy. See Fig. 13-1-I.

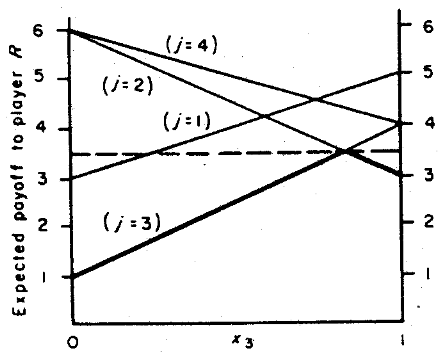


Figure 13-1-I. Graphical solution of a 2 × 4 matrix game.

On the other hand, if C randomizes his pure strategies, choosing j with probability y_j where $y_2 = y_3 = \frac{1}{2}$ and $y_1 = y_4 = 0$, then all the alternatives facing R reduce to a single dotted line at *constant* height $\frac{7}{2}$. In other words, C can limit his maximum gain to $\frac{7}{2}$ without knowing R's mixed strategy and could very well limit his gain to less if R deviates from his optimal mixed strategy.

The Mathematical Problem.

It thus appears that a conservative player can increase his expectation against an opponent who (he believes) knows he is conservative, by choosing at random among his pure strategies with certain *probabilities* $x_1, x_2, \ldots, x_m$. Any particular choice of x_i values is called a *mixed strategy* for player R. The mathematical problem for him thus becomes one of choosing $x_i \geq 0$, $\Sigma x_i = 1$, so that his expected payoff is maximum. This is referred to as the *optimum mixed strategy* for R. Just how to choose the x_i in general will be the subject of the next section. We will consider also mixed strategies for C which will be denoted by $(y_1, y_2, \ldots, y_n)$ where $y_j \geq 0$ and $\Sigma y_j = 1$.

A remarkable theorem due to von Neumann states that R can always assure himself a value v, the *value* of the game, if he plays his optimal mixed strategy and cannot hope to get more than this same value v even when playing to take advantage of a conservative opponent. What is more, any deviation by R, the other holding firm, runs the risk of a loss. The same statement holds, of course, for C and so we can expect that both players will play their optimal mixed strategies.

13-2. EQUIVALENCE OF MATRIX GAMES AND LINEAR PROGRAMS; THE MINIMAX THEOREM

Let $x_1, x_2, \ldots, x_m$ denote the probabilities that the row player R selects his pure strategies $i = 1, 2, \ldots, m$. We *assume* that R is *conservative*, that is to say R believes that C knows only R's probability of choice (but not his particular choice) and that C will play to take full advantage of this knowledge. If C chooses pure strategy $j = 1, 2, \ldots, n$, for any particular choice of x_i the expected *payoff* to R is $\sum_{i=1}^{m} a_{ij}x_i$ where, by definition, a_{ij} is the payoff to R if he chooses i and C chooses j. Thus R expects C to choose that pure strategy j corresponding to the *minimum* of these n expressions. Alternatively, if we *let L be any lower bound* for these expressions so that $\sum_{i=1}^{m} a_{ij}x_i \geq L$, then another way to state this is to say that C will choose (for fixed x_i) his pure strategy j to correspond to $j = j_0$ such that $\sum_{i=1}^{m} a_{ij_0}x_i = \text{Max } L$. Since R is interested that L be as large as possible, he will try to choose his x_i such that this Max L is as large as possible. Thus Max L is the largest floor that R can assure himself.

The Row Player's Problem: Choose $x_i \geq 0$ and Max L satisfying

$$\begin{array}{lr} a_{11}x_1 + a_{21}x_2 + \cdots + a_{m1}x_m \geq L & (x_i \geq 0) \\ a_{12}x_1 + a_{22}x_2 + \cdots + a_{m2}x_m \geq L & \\ \cdots\cdots\cdots\cdots\cdots\cdots & \\ a_{1n}x_1 + a_{2n}x_2 + \cdots + a_{mn}x_m \geq L & \\ x_1 + x_2 + \cdots + x_m = 1 & \end{array} \tag{1}$$

It is clear that we have reduced the solution of a matrix game for a conservative row player to a linear program. Let us now suppose that his opponent C also plays conservatively. In an analogous manner C is interested in choosing M as small as possible in the problem below. Thus Min M is the smallest ceiling that C can assure himself.

The Column Player's Problem: Choose $y_j \geq 0$ and Min M satisfying

$$\begin{array}{lr} a_{11}y_1 + a_{12}y_2 + \cdots + a_{1n}y_n \leq M & (y_j \geq 0) \\ a_{21}y_1 + a_{22}y_2 + \cdots + a_{2n}y_n \leq M & \\ \cdots\cdots\cdots\cdots\cdots\cdots & \\ a_{m1}y_1 + a_{m2}y_2 + \cdots + a_{mn}y_n \leq M & \\ y_1 + y_2 + \cdots + y_n = 1 & \end{array} \tag{2}$$

THEOREM 1: *If each player plays conservatively, each solves a linear program that is the dual of the other; this implies that the largest floor for* R *equals the lowest ceiling for* C.

PROOF: To see that (1) and (2) are duals of one another, transpose L and M to the left and let $M = z$ be the objective equation for (2) where z is to be minimized and $L = \underline{z}$ be the objective equation for (1) where $\underline{z}$ is to be maximized. Because any values $x_i \geq 0$, $\Sigma x_i = 1$; $y_j \geq 0$, $\Sigma y_j = 1$

may be used to obtain *feasible solutions* to the primal and dual problems, the *duality theorem states that optimal solutions exist* and Max L = Min M (§ 6-3, Theorem 1).

In substance, Theorem 1 tells us that if C plays conservatively, he can always hold R to the maximum assured gain whether or not R plays his optimal mixed strategy. Any deviation by R from his optimal runs risk of loss if C sticks to his optimal mixed strategy and there is a possibility of gain to R if C deviates from his. In such a situation we can expect that both players will use their optimal mixed strategies.

DEFINITIONS: The pair of optimal mixed strategies, if both players play conservatively, is called *the solution of the matrix game*; the resulting guaranteed expected payoff to the row player is called *the value of the game.*

THEOREM 2 (*von Neumann's Minimax Theorem*): *Given*
$\Sigma x_i = \Sigma y_j = 1, x_i \geq 0, y_j \geq 0,$

$$\text{(3)} \qquad \underset{x}{\text{Max}} \underset{y|x}{\text{Min}} \sum_i \sum_j a_{ij} x_i y_j = \underset{y}{\text{Min}} \underset{x|y}{\text{Max}} \sum_i \sum_j a_{ij} x_i y_j$$

The symbol $y|x$ is to be read "y given x." The left-hand side of (3) means: For some fixed (given) x, minimize the sum with respect to y; this results in a value that is a function of x; now choose x so that this value is maximum. Part of the term appearing on the right above may be rewritten (because $\Sigma x_i = 1, x_i \geq 0$),

$$\underset{x|y}{\text{Max}} \sum_i x_i \sum_j a_{ij} y_j = \underset{i}{\text{Max}} \sum_j a_{ij} y_j$$

and is the payoff to R if R knows C's mixed strategy and makes use of it. If C plays conservatively, he minimizes the entire right expression. Hence the right-hand side is a restatement of C's problem, and its value is Min M. Similarly, the left side is a restatement of R's problem and its value is Max L. In short, Theorem 2 is a concise statement of Theorem 1, since it includes (1) and (2).

COROLLARY 1: *If* C's *optimal strategy yields a strict inequality in the* i^{th} *relation of* (2), *then* R's *optimal strategy must have* $x_i = 0$; *if* R's *optimal strategy yields a strict inequality in the* j^{th} *relation of* (1), *then* C's *optimal strategy must have* $y_j = 0$. (For proof see Theorem 4, § 6-4.)

Equivalent Linear Programs.

We will now discuss various linear programs equivalent to the game problem [Dantzig, 1951-1; Gale, Kuhn, and Tucker, 1951-1]. Introducing into (2) slack variables $y_{n+i} \geq 0$, we have

$$\text{(4)} \qquad \begin{array}{l} a_{11} y_1 + a_{12} y_2 + \cdots + a_{1n} y_n + y_{n+1} = M \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ a_{m1} y_1 + a_{m2} y_2 + \cdots + a_{mn} y_n + y_{n+m} = M \\ y_1 + y_2 + \cdots + y_n = 1 \end{array}$$

By subtracting the first equation from each of the others (except the last), the resulting system is equivalent to a linear program in standard form. Find $y_j \geq 0$ and Min M satisfying

$$
\begin{array}{lllll}
(5) \quad a_{11}y_1 + & a_{12}y_2 + \dots + & a_{1n}y_n + & y_{n+1} & = M \\
(a_{21} - a_{11})y_1 + & (a_{22} - a_{12})y_2 + \dots + & (a_{2n} - a_{1n})y_n + & (y_{n+2} - y_{n+1}) & = 0 \\
\dots & \dots & \dots & \dots & \dots \\
(a_{m1} - a_{11})y_1 + & (a_{m2} - a_{12})y_2 + \dots + & (a_{mn} - a_{1n})y_n + & (y_{n+m} - y_{n+1}) & = 0 \\
y_1 + & y_2 + \dots + & y_n & & = 1
\end{array}
$$

EXERCISE: Show that solutions to primal and dual systems of (1), (2) and hence to (5) always exist.

EXERCISE: Show that if a constant K is added to each element of the payoff matrix, the optimal mixed strategy for either player is unchanged.

Another more symmetric way of effecting the reduction to standard form is found by substituting $y'_j M = y_j$ in (2), thus obtaining

$$
\begin{array}{ll}
(6) \quad a_{11} y'_1 + a_{12} y'_2 + \dots + a_{1n} y'_n \leq 1 & (y'_j \geq 0) \\
\dots & \\
a_{m1}y'_1 + a_{m2}y'_2 + \dots + a_{mn}y'_n \leq 1 & \\
y'_1 + \quad y'_2 + \dots + \quad y'_n = \dfrac{1}{M} \text{ (Max)} &
\end{array}
$$

Note $y'_j \geq 0$. *This substitution is valid* only if the value of the game, M, is known to be positive. However, since an arbitrary constant K can be added to all the elements of a payoff matrix without affecting the optimal mixed strategy, the restriction $M > 0$ presents no difficulties.

EXERCISE: How large must K be to guarantee $M > 0$?

Reduction of a Game to a Symmetric Game.

A two-person game is symmetric if the *same* number of pure strategies is open to both players and if the payoff to player R, when he selects his strategy i and his opponent selects j, is the same as the payoff to player C when the latter selects i and his opponent selects j. In other words, if $a_{ij} = -a_{ji}$ for all i, j. A matrix with this property is called *skew-symmetric*.

EXERCISE: Prove that the value of a symmetric matrix game is zero.

Von Neumann and Morgenstern [1944-1] first showed how to find a symmetric game matrix which is equivalent to any given game matrix. Later Gale, Kuhn, and Tucker [1951-1] showed a more compact way to symmetrize a game based on the same device by which a linear program is reduced to a game (see (11)). To reduce a game with matrix A to another game which is symmetric, we assume all $a_{ij} > 0$ (by adding a sufficiently large constant if necessary) and consider the dual programs which yield the optimum mixed strategies x and y of the two players:

$$\text{(7)}\qquad \begin{cases} u^T x = 1 \\ A^T x \geq vL \\ \quad L = z\,(\text{Max}), \end{cases} \qquad \begin{cases} v^T y = 1 \\ Ay \leq Mu \\ M = z\,(\text{Min}) \end{cases} \qquad (x \geq 0,\ y \geq 0)$$

where T, as in Chapter 8, denotes the transpose and $u^T = (1, 1, \ldots, 1)$, $v^T = (1, 1, \ldots, 1)$ are m- and n-component row vectors; thus $u^T x = \Sigma x_i = 1$ and $v^T y = \Sigma y_j = 1$.

We now consider a new matrix game with skew-symmetric matrix (8) of $m + n + 1$ columns and rows, and with a *new* pair of players:

$$\text{(8)}\qquad \begin{array}{c} \\ \bar{x} \\ \bar{y} \\ t \end{array} \begin{array}{c} \begin{array}{ccc} \bar{x} & \bar{y} & t \end{array} \\ \begin{bmatrix} 0 & A & -u \\ -A^T & 0 & v \\ u^T & -v^T & 0 \end{bmatrix} \end{array}$$

where $(\bar{x}, \bar{y}, t) = (\bar{x}_1, \ldots, \bar{x}_m; \bar{y}_1, \ldots, \bar{y}_n; t)$ denotes the optimal mixed strategy of either new player, and $\Sigma \bar{x}_i + \Sigma \bar{y}_j + t = 1$.

Exercise: Prove that if $\bar{x}, \bar{y}, t$ is optimal for player R, it is also optimal for player C.

Theorem 3: *If a game with matrix A has all positive elements, then any optimal mixed strategy $(\bar{x}, \bar{y}, t)$ for either player of the symmetric game* (8) *will yield optimal mixed strategies x, y and payoff M for players of the original game, namely*

$$\text{(9a)}\qquad y = \bar{y}/\sum \bar{y}_j, \qquad x = \bar{x}/\sum \bar{x}_i, \qquad M = 2t/(1 - t)$$

or conversely,

$$\text{(9b)}\qquad \bar{y} = y/(M + 2), \qquad \bar{x} = x/(M + 2), \qquad t = M/(M + 2)$$

Proof: Since the value of the symmetric game must necessarily be zero, the optimal mixed strategy satisfies

$$\text{(10)}\qquad \begin{aligned} A\bar{y} \quad - ut \qquad &\leq 0; \\ -A^T\bar{x} \quad + vt \qquad &\leq 0; \\ \sum \bar{x}_i \quad - \sum \bar{y}_j \qquad &\leq 0; \\ \sum \bar{x}_i \quad + \sum \bar{y}_j + t &= 1; \end{aligned} \qquad (\bar{x} \geq 0, \bar{y} \geq 0, t \geq 0)$$

where (10) is a special case of (2) when the skew-symmetric S replaces the A matrix of (2) and the value of the game is replaced by zero. We now observe that $t > 0$ if matrix A has all positive elements, because $t = 0$ implies $\bar{y} = 0$ from the first inequality, but then $\Sigma \bar{x}_i = 0$ by the next to the last inequality and $\Sigma \bar{x}_i = 1$ by the last equation—a contradiction. Next we observe from Corollary 1 that $\Sigma \bar{x}_i = \Sigma \bar{y}_j$ because $t > 0$ implies the "first" player of the symmetric game selects the last row with *positive* probability which means the expected payoff $\Sigma\bar{x}_i - \Sigma\bar{y}_j$ is maximal, i.e., zero. Finally, we note that $\bar{x} \neq 0$, for assuming $\bar{x} = 0$ would contradict the second equality; from $\Sigma\bar{x}_i = \Sigma\bar{y}_j$ follows $\bar{y} \neq 0$, and thus $t < 1$ from the last equality. It is therefore possible to define mixed strategies x and y and $M = L$ by (9) and observe that (10) reduces to (7) with $M = L$; hence x and y are optimal solutions to the dual programs and therefore optimal mixed strategies for the matrix A. Q.E.D.

Reduction of a Program to a Symmetric Game.

Analogous to the above we consider the dual programs

$$\text{(11)} \qquad \begin{cases} A^T x \leq c \\ b^T x = \underline{z} \text{ (Max)} \end{cases} \qquad \begin{cases} Ay \geq b \\ c^T y = z \text{ (Min)} \end{cases} \qquad (x \geq 0,\ y \geq 0)$$

where b and c are column vectors and study their relationship to the skew-symmetric game matrix,

$$\text{(12)} \qquad \begin{array}{c} \\ \bar{x} \\ \bar{y} \\ t \end{array} \begin{array}{c} \begin{array}{ccc} \bar{x} & \bar{y} & t \end{array} \\ \begin{bmatrix} 0 & -A & +b \\ +A^T & 0 & -c \\ -b^T & c^T & 0 \end{bmatrix} \end{array}$$

THEOREM 4: *A necessary and sufficient condition that solutions to linear programs* (11) *exist is that there exists an optimal mixed strategy* $(\bar{x}, \bar{y}, t)$ *to the symmetric game* (12) *with* $t > 0$. *The optimal solution to the programs is* $x = \bar{x}/t$, *and* $y = \bar{y}/t$.

The reduction of a linear program to a game was first established in 1948, based on conversations between the author and G. W. Brown. Soon thereafter, both Brown and Tucker noted its skew-symmetric game matrix [Dantzig, 1951-1].

PROOF: The optimal mixed strategy satisfies

$$\text{(13)} \qquad \begin{aligned} - A\bar{y} + bt &\leq 0 \\ A^T\bar{x} \qquad - ct &\leq 0 \\ -b^T\bar{x} + c^T\bar{y} \qquad &\leq 0 \end{aligned} \qquad (\bar{x} \geq 0,\ \bar{y} \geq 0)$$

Setting $\bar{y} = yt$, $\bar{x} = xt$, and noting $t > 0$, yields

$$\text{(14)} \qquad \begin{aligned} Ay &\geq b \\ A^T x &\leq c \\ c^T y &\leq b^T x \end{aligned} \qquad (x \geq 0,\ y \geq 0)$$

However, if the first inequality of (14) is multiplied on the left by x^T and the second by y^T, we obtain $b^T x \leq x^T A y \leq c^T y$. This together with the last inequality of (14) implies

$$\text{(15)} \qquad z = c^T y = b^T x = \underline{z}$$

Hence, by the Duality Theorem (§ 6-3, Theorem 1), x and y are optimal solutions to the dual programs.

Conversely, if optimal solutions x, y exist for the dual programs, then we can reverse our steps by defining $t > 0$ by

$$t\left(\sum x_i + \sum y_j + 1\right) = 1 \tag{16}$$

and setting $tx = \bar{x}$ and $ty = \bar{y}$.

The reduction of a linear program to a game depends on finding a solution of a game with $t > 0$. If $t = 0$ in a solution, it does not necessarily mean that an optimal feasible solution to the linear program does not exist. See Problems 3 through 7 at the end of the chapter.

13-3. CONSTRUCTIVE SOLUTION TO A MATRIX GAME (ALTERNATIVE PROOF OF MINIMAX THEOREM)

The matrix game is solved by a sequence of pivot operations on the linear program representing either the row or column player's problem. The rules of pivot choice will be reviewed so that the reader is provided with a constructive proof of the Minimax Theorem independent of the earlier chapters.[4]

The linear program § 13-2-(4) is set up in detached coefficient form (1) with constant terms in the first column; next, the columns y_{n+i} corresponding to slack variables; next, $(-M)$, the variable to be maximized, and then the main variables y_j.

(1)

Constants	y_{n+1}	. . .	y_{n+m-1}	y_{n+m}	$(-M)$	y_1	y_2	. . .	y_n
1					0	1	1	. . .	1
0	1				1	a_{11}	a_{12}	. . .	a_{1n}
.		.			.	.	.		.
.		.			.	.	.		.
.			.		.	.	.		.
0			1		1	$a_{m-1,1}$	$a_{m-1,2}$	. . .	$a_{m-1,n}$
0				1	1	a_{m1}	a_{m2}	. . .	a_{mn}

For convenience we assume the rows and columns of (1) are arranged so that

$$a_{m1} = \underset{j}{\text{Min}}\,[\underset{i}{\text{Max}}\, a_{ij}]$$

By multiplying the last row by -1 and the first row by $(a_{m1} - a_{k1})$ and adding their sum to row $k = 2, 3, \ldots, m - 1$ results in the equivalent tableau (2), where $a'_{kj} = a_{kj} - a_{mj} - a_{k1} + a_{m1}$. The last row is obtained

[4] Material in this section is drawn from Dantzig [1956-1], Tucker [1960-3], and Dorfman [1951-2].

by multiplying the top row by $-a_{m1}$ and adding to the last row. Heavy dots are placed below the unit vector columns $y_{n+1}, \ldots, y_{n+m-1}, (-M), y_1$.

(2)

Cycle 0

Constants	y_{n+1}	$\ldots$	y_{n+m-1}	y_{n+m}	$-M$	y_1	y_2	$\ldots$	y_n
1				0		1	1	$\ldots$	1
$a_{m1} - a_{11}$	1			-1		0	a'_{12}	$\ldots$	a'_{1n}
$\cdot$		$\cdot$		$\cdot$		$\cdot$	$\cdot$		$\cdot$
$\cdot$		$\cdot$		$\cdot$		$\cdot$	$\cdot$		$\cdot$
$\cdot$		$\cdot$		$\cdot$		$\cdot$	$\cdot$		$\cdot$
$a_{m1} - a_{m-1,1}$			1	-1		0	$a'_{m-1,2}$	$\ldots$	$a'_{m-1,n}$
$-a_{m1}$				1	1	0	$a_{m2} - a_{m1}$	$\ldots$	$a_{mn} - a_{m1}$
	●	$\ldots$	●		●	●			

Each cycle provides an improved mixed strategy for the column player C. The probability of choosing column j of the matrix game is zero, unless the corresponding column for y_j in some cycle is a unit vector column with a dot below. The value of y_j for such a column with a unit coefficient in row i is the i^{th} component of the constant column. Thus for cycle 0, $y_1 = 1$. Let us suppose the equivalent system in tableau form for cycle p is given by (3) which always includes some $m + 1$ unit vector columns (usually indicated by dots below these columns).

Cycle p

(3)

Constants	y_{n+1}	y_{n+2}	$\ldots$	y_{n+m}	$-M$	y_1	$\ldots$	y_s	$\ldots$	y_n
$\bar{b}_1$	β_{11}	β_{12}	$\ldots$	β_{1m}	0	$\bar{a}_1$	$\ldots$	$\bar{a}_{1s}$	$\ldots$	$\bar{a}_{1n}$
$\cdot$	$\cdot$	$\cdot$		$\cdot$	$\cdot$	$\cdot$		$\cdot$		$\cdot$
$\cdot$	$\cdot$	$\cdot$		$\cdot$	$\cdot$	$\cdot$		$\cdot$		$\cdot$
$\cdot$	$\cdot$	$\cdot$		$\cdot$	$\cdot$	$\cdot$		$\cdot$		$\cdot$
$\bar{b}_r$	β_{r1}	β_{r2}	$\ldots$	β_{rm}	0	$\bar{a}_{r1}$	$\ldots$	$\bar{a}_{rs}$	$\ldots$	$\bar{a}_{rn}$
$\cdot$	$\cdot$	$\cdot$		$\cdot$	$\cdot$	$\cdot$		$\cdot$		$\cdot$
$\cdot$	$\cdot$	$\cdot$		$\cdot$	$\cdot$	$\cdot$		$\cdot$		$\cdot$
$\cdot$	$\cdot$	$\cdot$		$\cdot$	$\cdot$	$\cdot$		$\cdot$		$\cdot$
$\bar{b}_m$	β_{m1}	β_{m2}	$\ldots$	β_{mm}	0	$\bar{a}_{m1}$	$\ldots$	$\bar{a}_{ms}$	$\ldots$	$\bar{a}_{mn}$
$-M_0$	$\bar{c}_{n+1}$	$\bar{c}_{n+2}$	$\ldots$	$\bar{c}_{n+m}$	1	$\bar{c}_1$	$\ldots$	$\bar{c}_s$	$\ldots$	$\bar{c}_n$

THEOREM 1: *If $\bar{c}_j \geq 0$ for all j and $\bar{b}_i \geq 0$ for all i, then an optimal mixed strategy for* R *is $x_i = \bar{c}_{n+i}$ for $i = 1, 2, \ldots, m$, and an optimal mixed strategy for* C *is to choose columns j of the matrix game with probability* 0 *unless the corresponding j in* (3) *is a dotted i^{th} unit column, $j = j_i$, in which case $y_{j_i} = \bar{b}_i$; moreover, M_0 is the value of the matrix game.*

PROOF: Note that the linear combination of rows of (1) with weights $M, x_1, x_2, \ldots, x_m$ that forms the last equation of (3) is $-M_0, \bar{c}_{n+1}, \ldots, \bar{c}_{n+m}$. These weights applied to the coefficients of $-M$ yield $\Sigma x_i = 1$, and to any other column j yield $\Sigma x_i a_{ij} = \bar{c}_j + M_0 \geq M_0$. Since $y_{j_i} = \bar{b}_i$ obviously solves C's problem and the payoff is also equal to M_0, the solutions are optimal.

Rules for Pivoting.

Step 1. Choose pivot column s (and dot column) by

$$\bar{c}_s = \text{Min } \bar{c}_j \qquad (j = 1, 2 \ldots, n + m) \tag{4}$$

If $\bar{c}_s \geq 0$ exit via step 4.

Step 2. Choose pivot row r (and remove dot from column with unity in row r) from among rows i with $\bar{a}_{is} > 0$ such that the row vector

$$[\bar{b}_r, \beta_{r1}, \beta_{r2}, \ldots, \beta_{rm}]/\bar{a}_{rs} \tag{5}$$

is the *lexicographic minimum* of the row vectors

$$[\bar{b}_i, \beta_{i1}, \beta_{i2}, \ldots, \beta_{im}]/\bar{a}_{is} \tag{6}$$

This is done by first comparing the leading components of these vectors, thus

$$\bar{b}_r/\bar{a}_{rs} = \text{Min } \bar{b}_i/\bar{a}_{is} \qquad (\bar{a}_{rs}, \bar{a}_{is} > 0)$$

where $\bar{a}_{is} = \beta_{ik}$ if $s = n + k$.

The Rule for Resolving Ties

If the choice of r is not unique so that $r = r_1, r_2, \ldots$ yields the minimum ratio above, then pass to column y_{n+1} and choose pivot row r by

$$\frac{\beta_{r1}}{\bar{a}_{rs}} = \text{Min } \frac{\beta_{i1}}{\bar{a}_{is}} \qquad (r, i = r_1, r_2, \ldots)$$

If the choice of r is still not unique so that $r = r_1', r_2', \ldots$ minimizes both ratios above, pass to column y_{n+2} and choose row r by

$$\frac{\beta_{r2}}{\bar{a}_{rs}} = \text{Min } \frac{\beta_{i2}}{\bar{a}_{is}} \qquad (r, i = r_1', r_2', \ldots)$$

Continue in this manner, passing to columns $y_{n+3}, y_{n+4}, \ldots$ until r is uniquely chosen.

Step 3. Pivot; cycle p is complete. Return to Step 1.

Step 4. Terminate by computing an *optimal mixed strategy* for player R:

$$x_1 = \bar{c}_{n+1}, x_2 = \bar{c}_{n+2}, \ldots, x_m = \bar{c}_{n+m} \tag{7}$$

and an optimal mixed strategy for player C: choose column j of the matrix game with probability 0 for $j = 1, 2, \ldots, n$ unless corresponding to j in cycle p is a dotted i^{th} unit vector column, in which case for each such i

$$y_{j_i} = \bar{b}_i \tag{8}$$

Note on Lexicographic Ordering.

We now digress for a moment and discuss a way to arrange or order a set of vectors (each with the same number of components) in sequence in much the same way that one orders a set of words in a dictionary; we will therefore refer to this way of ordering vectors as "lexicographic." We define a vector R as greater than zero in the lexicographic sense or, more simply, *lexico-positive*[5] and denoted $R \succ 0$, if it has at least one non-zero component, the first of which is positive. We next define a vector R to be *greater than* S in the lexicographic sense, denoted $R \succ S$, if $R - S \succ 0$.

EXERCISE:

(1) Show that the lexicographic ordering relation is transitive, i.e., $R \succ S$, $S \succ T$, implies $R \succ T$.

(2) Show that any two vectors R and S with the same number of components satisfy either $R \succ S$, $S \succ R$, or $R = S$; show also that $R \succ S$ implies $R \neq S$.

(3) Let $R \succeq S$ mean that either $R \succ S$ or $R = S$; show that $R \succeq S$ and $S \succeq R$ implies $R = S$.

(4) If $R \succ 0$ and $S \succ 0$, then $R + S \succ 0$. If $k > 0$ is a scalar and $R \succ 0$, then $kR \succ 0$.

Example: A. W. Tucker in his [1960-3] paper, "Solving a Matrix Game by Linear Programming," provides us with the following matrix game

$$\begin{bmatrix} 1 & -1 & 0 \\ -6 & 3 & -2 \\ 8 & -5 & 2 \end{bmatrix}. \tag{9}$$

The column players problem in detached coefficient form is:

(10)

Constants	y_4	y_5	y_6	$-M$	y_1	y_2	y_3
1				0	1	1	1
0	1			1	1	−1	0
0		1		1	−6	3	−2
0			1	1	8	−5	2*

By analogy with (2), the starred entry $a_{33} = 2$ is $\operatorname{Min}_j [\operatorname{Max}_i a_{ij}]$, hence the cycle-0 tableau (11) is obtained from (10) by two easy pivot operations such that the y_4, y_5, $-M$, y_3 columns are unit vectors.

[5] The abbreviated term is due to A. W. Tucker.

Cycle 0

(11)

Constants	y_4	y_5	y_6	$-M$	y_1	y_2	y_3	
1					1	1	1	
2	1		−1		−5	6		Tie
4		1	−1		−10	12		Tie
−2			1	1	6	−7		
	●	●		●		★	●	

Note that $r = 2$ and $r = 3$ are both tied for pivot since $\frac{2}{6} = \frac{4}{12}$. However, lexicographically the vector $\frac{1}{6}[2\ 1\ 0\ \text{-}1\ 0] \succ \frac{1}{12}[4\ 0\ 1\ \text{-}1\ 0]$. Hence $r = 3$ is chosen for pivot row. Pivoting on 12 yields

Cycle 1

(12)

Constants	y_4	y_5	y_6	$-M$	y_1	y_2	y_3
8/12		−1/12	1/12		22/12		1
0	1	−6/12	−6/12		0		
4/12		1/12	−1/12		−10/12	1	
+4/12		7/12	5/12	1	2/12		
	●			●		●	●

Since all entries in the bottom row are nonnegative, the cycling is terminated. The optimal solution for R is taken from the bottom row in the y_4, y_5, y_6 columns; it is $x_1 = 0$, $x_2 = \frac{7}{12}$, $x_3 = \frac{5}{12}$. The optimal solution for C is taken from the constant column; it is $y_3 = \frac{8}{12}$, $y_2 = \frac{4}{12}$ and all other non-slack y_j are zero. The value of the game is $M_0 = -\frac{4}{12}$.

Proof of Algorithm.

In (2), the tableau for cycle 0, entries in columns for $y_1, y_{n+1}, y_{n+2}, \cdots, y_{n+m-1}$, $-M$ form an identity matrix. The effect of a pivot operation in column s, row r is to transform column s into a unit vector with unity in the r^{th} position. The effect of several cycles is to shift the position of the identity matrix into the columns $y_{j_1}, y_{j_2}, \cdots, y_{j_m}$ and the $-M$ column. (The column for $-M$ remains the same from cycle to cycle.) The set of columns of the cycle-0 tableau corresponding to these unit columns in the cycle p tableau is referred to as the *basis* for cycle p. We shall denote it by B. It is easy to see that B is non-singular and, because under the pivot operations B transforms into the identity matrix (see § 8-4-(9), (10), and sequel), the identity matrix consisting of the first $m + 1$ columns of cycle 0 transforms into B^{-1} of cycle p.

We now show that the choice of pivot row r given in Step 2 always results in a unique value for r. Since r must be chosen among i such that $\bar{a}_{is} > 0$ we first show that at least one $\bar{a}_{is}$ is positive (so that this class is non-empty). Notice in the tableau of cycle p that the columns corresponding to $y_{j_1}, y_{j_2}, \ldots, y_{j_m}, -M$, form an identity matrix; hence, the linear combination of these columns using weights $\bar{a}_{is}$ results in column s and therefore this same relation among columns must hold also for the original tableau (1) (since pivoting does not effect a linear-combination relation among columns). However, assuming on the contrary that all $\bar{a}_{is} \leq 0$ would imply that some nonpositive linear combination of certain columns of (1) could yield column s. But this is clearly impossible since each column $j = 1, 2, \ldots, n$ has unity in the top row and the linear combination must include at least one such j.

We now show that the rule for deciding in which row to choose the pivot term results in a unique choice because non-uniqueness would imply at least two row vectors (5) and (6) to be equal component by component; but then, the square submatrix of (3) consisting of the first $m + 1$ rows and columns (starting with the constants column) would be singular—a contradiction, since this submatrix is obtained from (1) by a sequence of pivot operations and the corresponding submatrix of (1) is an identity matrix and, hence, non-singular.

The rule for selection of r also prevents the repetition of a basis obtained on an earlier cycle. To demonstrate this, let us denote for cycle p the rows of the tableau by $R_1, R_2, \ldots, R_m; R_{m+1}$, and those for cycle $p + 1$ by $R_1^*, R_2^*, \ldots, R_m^*; R_{m+1}^*$. We will now show

LEMMA 1: $R_i \succ 0$ *for all* i *and any cycle* p.

LEMMA 2: $R_{m+1}^* \succ R_{m+1}$.

The Proof is Inductive. The relation between R_i and R_i^* is

$$R_i^* = R_i - R_r \frac{\bar{a}_{is}}{\bar{a}_{rs}} \qquad (i \neq r) \tag{13}$$

$$R_r^* = R_r \frac{1}{\bar{a}_{rs}}$$

$$R_{m+1}^* = R_{m+1} - R_r \frac{\bar{c}_s}{\bar{a}_{rs}} \qquad \text{where } \bar{c}_s < 0$$

where r is the pivot row and $\bar{a}_{rs} > 0$. It follows at once that $R_r^* \succ 0$ if $R_r \succ 0$; moreover if $\bar{a}_{is} \leq 0$, then $R_i^* \succ 0$ since R_i and $-R_r(\bar{a}_{is}/\bar{a}_{rs})$ are each lexico-positive by our inductive assumptions (the same argument applies to R_{m+1}^* since $\bar{c}_s < 0$ and thus establishes Lemma 2). On the other hand, if $\bar{a}_{is} > 0$, then we write

$$R_i^* = [R_i/\bar{a}_{is} - R_r/\bar{a}_{rs}]\bar{a}_{is}$$

We now observe that the rule for selecting the pivot row in Step 2 is the same as selecting the lexico-smallest vector $R_i/\bar{a}_{is}$ among i such that $\bar{a}_{is} > 0$.

Thus $R_i^* \succeq 0$. As observed earlier, the square submatrix of the first $m + 1$ rows and columns of (3) is non-singular so that a non-zero component exists for each vector R_i somewhere among its first $m + 1$ components and hence on the next iteration for $R_i^* \neq 0$ also. Since we have already shown $R_i^* \succeq 0$ it follows that $R_i^* \succ 0$.

To show finiteness of the algorithm note that the last row is strictly increasing in the lexicographic sense; hence *no tableau can be the same as one obtained earlier*. We get a contradiction by assuming an infinite number of iterations because *the number of different bases is finite*, and, therefore, on some iteration, there would be a basis that is the same as one obtained earlier. But in this case the entire tableau would have to be repeated because the $\bar{a}_{ij}$ in the tableau simply express the combination of columns in the basis that form column j and this linear combination is unique and invariant under pivoting.

13-4. PROBLEMS

1. Suppose each of two players has a penny, a nickel, and a dime that he can select from as his pure strategies. If both players select the same type coin, Player 1 wins Player 2's coin; if the two coins *are not the same type*, then Player 2 wins Player 1's coin. How much should Player 2 give Player 1 before the game in order to make the game fair?
2. What is the analogue for a linear program of the saddle point solution of a game, if it exists?
3. (a) Show that, if $t = 0$ in Theorem 4 of § 13-2, this does not imply that a program does not have a solution.
   (b) Show that, if $t = 0$ and if the optimal solution has *positive slack* in the complementary relation of the dual system using the solution to the symmetric game, then there exists no solution to the linear program.
   (c) Show that, if $t = 0$ and there is *zero slack* in the complementary relation, it does not imply that a program does not have a solution.
   (d) If the coefficients $a_{ij} > 0$, show that $t > 0$.
   (e) (Shapley) Show that no feasible solution exists for the primal program below but that $t = 0$ and its complementary slack is zero for the equivalent game.

$$\text{Primal Program:} \begin{cases} 0y_1 \geq +1 \\ 0y_1 = z \text{ (Min)} \end{cases} \qquad \begin{matrix} \text{Solution} \rightarrow & 0 & 1 & 0 & \\ \text{Equivalent Game:} & \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} & & & \begin{matrix} 0 \\ 1 \\ 0 \\ \uparrow \\ \text{Solution} \end{matrix} \end{matrix}$$

(f) (Shapley) Show that feasible solutions exist and no lower bound for v exists for the dual of the above program (shown below) but that $t = 0$ and its complementary slack is zero for the equivalent game.

Dual Program: $\begin{cases} 0x_1 \leq 0 \\ +1x_1 = v \text{ (Max)} \end{cases}$

Equivalent Game:

$$\begin{matrix} \text{Solution} \rightarrow & 1 & 0 & 0 & \\ & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} & & & \begin{matrix} 1 \\ 0 \\ 0 \end{matrix} \\ & & & & \uparrow \\ & & & & \text{Solution} \end{matrix}$$

where $x_1 \geq 0$, $y_1 \geq 0$.

4. Construct an example to show that feasible solutions to both the primal and dual systems can exist, but that the solution to the equivalent game can have $t = 0$.
5. Prove: If a solution to a game is unique when variables corresponding to positive complementary slack values are dropped, then the remaining variables have positive values.
6. (Wolfe) Prove the following theorem: *All solutions of the corresponding game have $t > 0$ if, and only if, there exist u, $x > 0$ such that $uA > c$ and $Ax < b$.*
7. (Wolfe) Prove the following theorem: *If the set of $x \geq 0$, $Ax \leq b$ is non-empty, bounded and has an interior, and A has no zero row, then any solution $[x, u, t]$ of the corresponding game has $t > 0$.*

REFERENCES

Bellman and Blackwell, 1951-1
Blackwell and Girshick, 1954-1
Bohnenblust, Karlin, and Shapley, 1950-1
Borel, 1921-1, 1924-1, 1927-1, 1953-1
Brown, 1951-1
Brown and von Neumann, 1950-1
Dantzig, 1951-1, 1956-1, 1958-2
Dorfman, 1951-2
Dresher, 1961-1
Dresher, Tucker, and Wolfe, 1957-1
Gale, 1956-2, 1960-1
Gale, Kuhn, and Tucker, 1951-1
Gass, 1958-1
Kaplansky, 1945-1
Karlin, 1959-1
Kemeny, Snell, and Thompson, 1957-1
Kuhn and Tucker, 1950-1, 1953-1, 1955-1, 1958-1
Luce and Raiffa, 1957-1
MacDonald, 1950-1
McKinsey, 1952-1
Mills, 1956-1
Morgenstern, 1949-1
Raiffa, Thompson, and Thrall, 1952-1
Richardson, 1958-1
Robinson, 1951-1
Shapley and Snow, 1950-1
Shubik, 1959-1
Tucker, 1950-1, 1955-2, 1960-3
Vajda, 1956-1
Ville, 1938-1
von Neumann, 1928-1, 1937-1
von Neumann and Morgenstern, 1944-1
Williams, 1954-1

CHAPTER 14

THE CLASSICAL TRANSPORTATION PROBLEM

14-1. HISTORICAL SUMMARY

An important class of linear programming problems of economic and physical origin can be formulated in terms of a network composed of points (nodes) connected by routes (arcs), over which various kinds of transport (flow) take place (see Chapter 19).

The classical transportation problem arises when we must determine an optimal schedule of shipments that:

(a) originate at sources (supply depots) where fixed stockpiles of a commodity are available;

(b) are sent directly to their final destinations (demand depots) where various fixed amounts are required;

(c) exhaust the stockpiles and fulfill the demand; hence, total demand equals total supply;

and finally, the cost must

(d) satisfy a linear objective function; that is, the cost of each shipment is proportional to the amount shipped, and the total cost is the sum of the individual costs.

In this chapter we will take up this problem and show how it may be solved by the simplex method. (Succeeding chapters will explore some of its important variations.)

Although he awakened little interest at the time, L. V. Kantorovich [1939-1] showed that a class of problems closely related to the classical transportation case has a remarkable variety of applications concerned typically with the allotment of tasks to machines whose costs and rates of production vary by task and machine type. He gave a useful but incomplete algorithm for solving such problems (see Chapter 21). Again, in 1942, he wrote a mathematical paper concerned with a *continuous* version of the transportation problem, and in 1948, he authored an applicational study, jointly with Gavurin, on the *capacitated* transportation problem (see Chapter 18).

The now standard form of the problem was first formulated, along with a constructive solution, by Frank L. Hitchcock [1941-1]. His paper, "The Distribution of a Product from Several Sources to Numerous Localities," sketched out the partial theory of a technique foreshadowing the simplex method; it did not exploit special properties of the transportation problem except in finding starting solutions. This paper also failed to attract much attention.

Still another investigator, T. C. Koopmans, as a member of the Combined Shipping Board during World War II, became concerned with using solutions of the transportation problem to help reduce over-all shipping times, for the shortage of cargo ships constituted a critical bottleneck.

In 1947, Koopmans began to spearhead research on the potentialities of linear programs for the study of problems in economics. His historic paper, "Optimum Utilization of the Transportation System" [1947-1], was based on his wartime experience. Because of this and the work done earlier by Hitchcock, the classical case is often referred to as the Hitchcock-Koopmans Transportation Problem.

Another whose work anticipated the recent era of development in linear programming was E. Egerváry, a mathematician. His 1931 paper considered the problem of finding a permutation of ones in a matrix composed of zero and one elements. Based on this investigation, Kuhn [1955-1] developed an efficient algorithmic method for solving assignment problems (see § 15-1). Kuhn's approach, in its turn, underlies the Ford-Fulkerson Method for solution of the classical transportation problem (Chapter 20).

The method to be described in this chapter was developed independently, by specializing the general simplex method [Dantzig, 1951-2].

14-2. ELEMENTARY TRANSPORTATION THEORY

Suppose that m warehouses (origins) contain various amounts of a commodity which must be allocated to n cities (destinations). Specifically, the i^{th} warehouse must dispose of exactly the quantity a_i, while the j^{th} city must receive exactly the quantity b_j. It is assumed that the *total demand equals the total supply*, that is,

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \tag{1}$$

Besides the numbers, a_i and b_j, which are *nonnegative*, we are also given a set of numbers, c_{ij}, which may be *unrestricted* (although usually nonnegative under the present interpretation). The number c_{ij} represents the cost (or profit, if negative) of shipping a unit quantity from origin i to destination j. Our problem is to determine the number of units to be shipped from i to j in order that stockpiles will be depleted and needs satisfied at an over-all minimum cost.

The special structure of the matrix is evident when the equations are written in standard form, as in (2).

(2)

$$
\begin{array}{l}
x_{11} + x_{12} + \dots + x_{1n} \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad = a_1 \\
\qquad\qquad\qquad + x_{21} + x_{22} + \dots + x_{2n} \qquad\qquad\qquad\qquad = a_2 \\
\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\
\qquad\qquad\qquad\qquad\qquad\qquad\qquad x_{m1} + x_{m2} + \dots + x_{mn} = a_m \\
\hline
x_{11} \qquad\qquad\qquad + x_{21} \qquad\qquad\qquad + x_{m1} \qquad\qquad = b_1 \\
\quad + x_{12} \qquad\qquad\qquad + x_{22} \qquad\qquad\qquad + x_{m2} \qquad = b_2 \\
\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\
\qquad\quad + x_{1n} \qquad\qquad\qquad + x_{2n} \qquad\qquad\qquad + x_{mn} = b_n \\
\hline
c_{11}x_{11} + \dots + c_{1n}x_{1n} + c_{21}x_{21} + \dots + c_{2n}x_{2n} \dots + c_{m1}x_{m1} + \dots + c_{mn}x_{mn} = z
\end{array}
$$

EXERCISE: Condition (1) renders the system dependent since the sum of the first m-equations is the same as the sum of the last n. Prove that the rank of the system is exactly $m + n - 1$. Also show that each equation is a linear combination of the other $m + n - 1$ so that any one equation may be called redundant and discarded.

The Standard Transportation Array.

An important feature of this model is that it can be abbreviated in the form of a rectangular array, which displays the values of x_{ij} and c_{ij} in row i and column j, and the values of the constants and corresponding multipliers for the first m equations in a marginal column and for the remaining n equations in a marginal row.

(3)

						Row Totals
	x_{11} c_{11}	x_{12} c_{12}	x_{13} c_{13}	x_{14} c_{14}	x_{15} c_{15}	a_1 u_1
	x_{21} c_{21}	x_{22} c_{22}	x_{23} c_{23}	x_{24} c_{24}	x_{25} c_{25}	a_2 u_2
	x_{31} c_{31}	x_{32} c_{32}	x_{33} c_{33}	x_{34} c_{34}	x_{35} c_{35}	a_3 u_3
Column Totals	b_1 v_1	b_2 v_2	b_3 v_3	b_4 v_4	b_5 v_5	Implicit ← Prices ↑

At any stage of the algorithm, the square (i, j), situated in the i^{th} row and the j^{th} column, contains c_{ij} in its lower right-hand corner while, at its upper left corner, we find x^o_{ij}, the current numerical *value* assigned to x_{ij}. The lack of such an entry implies that x_{ij} is non-basic and hence of zero value. Zero-valued *basic* variables are indicated by a zero entry (degeneracy).

Any square along the right-hand or bottom margin, however, differs in that it contains the row or column *totals*, a_i or b_j, at upper left, and the corresponding current value, u_i^o or v_j^o, of the simplex multipliers at lower right. Each row and column of the array represents an equation. Specifically,

(4) The Row Equations: $(i = 1, 2, \ldots, m)$ $$\sum_{j=1}^{n} x_{ij} = a_i$$

(5) The Column Equations: $(j = 1, 2, \ldots, n)$ $$\sum_{i=1}^{m} x_{ij} = b_j$$

In order that these equations continue to be satisfied during the course of the algorithm, we must keep the sum of the entries in each row and column equal to the appropriate row or column total, a_i or b_j, which appear in the margins.

Finding a Basic Feasible Solution.

As candidate for the first basic variable, choose any variable, x_{pq}, and make it as large as possible, consistent with row and column totals, i.e., set

$$x_{pq} = \text{Min}\,[a_p, b_q]$$

Case 1: If a_p is less than b_q, then all the other variables in the p^{th} row are to be given the value zero and designated as non-basic. Next delete the p^{th} row, reduce the value of b_q to $(b_q - a_p)$, and proceed in the same manner to evaluate a variable in the *reduced array* composed of the $m - 1$ rows and n columns remaining.

Case 2: If a_p is greater than b_q, then, similarly, the q^{th} column is to be deleted and a_p replaced by $(a_p - b_q)$, etc.

Case 3: If a_p equals b_q, then delete either the row or the column, but *not both*. However, if several columns, but only one row, remain in the reduced array, then drop the q^{th} column, and conversely, if several rows and one column, drop the p^{th} row.

This rule will select as many variables for the basic set as there are rows plus columns, less one, $m + n - 1$, since on the last step, when one row and one column remain, both must be dropped after the last variable is evaluated. The important fact is that, as we shall show, *all basic solutions are of this form*, so that in defining a specific algorithm for achieving optimality, it is unnecessary to consider other forms of solution.

EXERCISE: We have assumed that $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$. Show that every reduced array retains this property, so that the last remaining variable can acquire a value consistent with the totals for the single row and column still remaining in the final reduced array.

EXERCISE: Let B be a square matrix. Show that B is nonsingular if and only if $Bx = b$ has a unique solution for every b.

THEOREM 1: *The candidate variables chosen by the rule for initial solution constitute a basic set.*

PROOF: Since the rank of the system is $m + n - 1$, a set of variables, x_{pq}, constitutes a basic set if its values, satisfying all $m + n$ equations, are given *uniquely* as linear combinations of any $m + n - 1$ of the $m + n$ marginals $a_1, a_2, \ldots, a_m; b_1, b_2, \ldots, b_n$, when the remaining variables are set equal to zero. Now, the value given each basic variable by the starting rule is the same as a marginal total of some reduced array, but this total, because of the way in which it was derived, is equal to the difference between partial sums of the original row and column totals. This shows the values are uniquely determined by some *particular* set of $m + n - 1$ marginals. However, because of condition (1), any one of these totals may be re-expressed as a linear combination of the others; for example,

$$b_n = \sum_{i=1}^{m} a_i - \sum_{j=1}^{n-1} b_j$$

COROLLARY 1: *The row totals of each reduced array are expressible as partial sums of the a_i minus partial sums of the b_j, whereas every column total is expressible as the negative of such a difference.*

EXERCISE: Show the above by induction.

The Property of Basis Triangularity.

When a system has the property that every basis is triangular (see § 4-2), then basic solutions are easily obtained. Let numbers, for example zeros, be substituted for all the non-basic variables. The resulting system of equations will, of course, involve only the basic variables. By definition, if triangular, the subsystem will contain at least one equation having exactly one variable, and this variable may, of course, be immediately evaluated by means of a single division. When the value so determined is substituted in the remaining equations, there will again be at least one equation with exactly one variable in the reduced system, and so forth. Thus, all of the basic variables may be evaluated in an analogous way. In particular, since each equation corresponds to a row or a column, this is the same thing as saying that all basic sets of variables can be generated by the starting rule of solution [Dantzig, 1951-2].

THEOREM 2: *Fundamental Theorem for the Transportation Problem: All bases are triangular.*

PROOF: Suppose we have a standard transportation array, similar to (3), with m rows and n columns and with arbitrary constants, a_i and b_j. Consider any particular set of basic variables and substitute the value zero for each of the non-basic variables.

Contrary Assumption: Assume that no row or column has exactly one basic variable. We discount the possibility that some row or column might have no basic variables since this would mean that the left member of the corresponding equation would be zero while the marginal total, presumed arbitrary, could be non-zero. Hence, all columns under our assumption must have two or more basic entries. If k is the total number of basic entries in the array, then, since there are at least two such entries per column, we must have

$$k \geq 2n$$

Similarly, we have

$$k \geq 2m$$

since there are at least two basic variables per row. Summing, we get

$$k \geq m + n \tag{6}$$

Now, there are $m + n$ equations, but since condition (1) renders one of these redundant, the number of basic variables, k, must also satisfy

$$k < m + n \tag{7}$$

in direct contradiction to (6). Thus, we have proved

LEMMA 1: *Some row or column has exactly one basic entry.*

Now to establish the triangularity of the basis, we must show

LEMMA 2: *The subsystem resulting by the exclusion of any redundant equation from the original system still contains an equation in exactly one basic variable.*

Suppose we drop some equation as redundant, say, the last row equation, and again make the contrary assumption, then

$$k \geq 2n \tag{8}$$

$$k' \geq 2(m - 1)$$

where k' is the total number of basic variables in all but the last row. However, since there is at least one basic entry in the last row, we have

$$k \geq k' + 1 \tag{9}$$

Hence, adding both relations in (8) and (9), we have

$$2k \geq 2m + 2n - 1 \quad \text{or} \tag{10}$$

$$k \geq m + n - \tfrac{1}{2}$$

contradicting the fact that at least one equation is redundant. This proves the lemma.

Now, starting with the original array, we delete the row or column having a single entry and repeat the argument for the reduced array, etc., thereby establishing the theorem.

THEOREM 3: *The values of all basic variables are integers if the row and column totals are integers.*

PROOF: The value of each basic variable is the same as a marginal total of some reduced array, but this is expressible as the difference between partial sums of the original row and column totals (see Corollary 1).

Simplex Multipliers.

Instead of using the symbol π_i, we represent the multiplier of the i^{th} row equation as u_i, and that of the j^{th} column equation as v_j. Since *any equation may be considered redundant, we can assign an arbitrary value to one of the simplex multipliers* and then evaluate the set of multipliers, thereby rendered unique, which will cause the vanishing of all the relative cost factors corresponding to basic variables. For the present, we will suppose that

$$v_n = 0 \tag{11}$$

After multiplying the i^{th} row equation of (2) by u_i and the j^{th} column equation by v_j, we subtract the weighted sum from the objective form, to obtain a modified z-equation,

$$\sum_{i,j} \bar{c}_{ij} x_{ij} = z - z_0 \tag{12}$$

where

$$\bar{c}_{ij} = c_{ij} - (u_i + v_j) \quad \text{for } i = 1, 2, \ldots, m; \text{ and } j = 1, 2, \ldots, n \tag{13}$$

and

$$z_0 = \sum_{i=1}^{m} a_i u_i + \sum_{j=1}^{n} b_j v_j \tag{14}$$

The values for u_i and v_j are chosen to make the coefficients of the basic variables vanish, i.e.,

$$c_{ij} = u_i + v_j \qquad \text{for } x_{ij}, \text{ a basic variable} \tag{15}$$

Note that (15) defines a *system* of equations in which the simplex multipliers play the part of variables. This system of equations has a matrix which is the *transpose* of the particular basis for which we desire simplex multipliers. (The *basis*, it will be recalled, is the matrix of coefficients belonging to the basic set of variables (§ 4-2 and § 8-2).)

EXERCISE: Show that u_i and v_j may be replaced by $(u_i + k)$ and $(v_j - k)$ without affecting the value of $\bar{c}_{ij}$ and z_0 in (13) and (14); hence, any one of $m + n$ multipliers may be given an arbitrary value in determining the remainder.

THEOREM 4: *When the unit costs, c_{ij}, are integers and one multiplier is given an arbitrary integral value, then all the simplex multipliers will be integers.*

PROOF: Since the basis is triangular, so is its transpose. Moreover, the transpose is also of rank $m + n - 1$. This means that the values of u_i and v_j satisfying (15) can be obtained uniquely, after one of them is arbitrarily assigned, in the same manner as the values of the basic variables; i.e., by looking for one equation in one unknown, etc. Since the coefficients in the basis are either unity or zero, the values of u_i and v_j will be sums and differences of c_{ij} corresponding to basic variables.

Finding an Improved Basic Solution.

The simplex criterion for optimality is $\bar{c}_{ij} \geq 0$ for all (i, j), i.e.,

$$c_{ij} \geq u_i + v_j, \qquad \text{for } i = 1, 2, \ldots, m; \text{ and } j = 1, 2, \ldots, n \tag{16}$$

On the other hand, if for some s and t

$$c_{st} < u_s + v_t \tag{17}$$

then a new basic solution is obtained by increasing the value of the non-basic variable, x_{st}, if possible, and adjusting the values of the basic variables so as to compensate.

To determine the effect of increasing x_{st}, the terms involving x_{st} are moved to the right in (2), and the values of the basic variables are redetermined.

THEOREM 5: *If the value of a non-basic variable, x_{st}, is allowed to increase, with the other non-basic variables remaining at zero, the value of any basic variable, x_{ij}, will change from x^o_{ij} to*

$$x_{ij} = x^o_{ij} \pm \delta x_{st}, \qquad \text{where } \delta = 0 \text{ or } 1 \tag{18}$$

PROOF: When, in the *general* simplex process, the non-basic variable, x_s, is allowed to increase while the other independent variables remain at zero, the value of the i^{th} basic variable is given by

$$x_{j_i} = \bar{b}_i - \bar{a}_{is} x_s$$

where $\bar{b}_i$ is obtained by solving the basic system of equations when the right-hand side of the i^{th} equation is b_i. The value of $\bar{a}_{is}$ is obtained by solving the same system when the right-hand side is a_{is}. In this case, the coefficient of x_{st}, as given in (2), is unity for the s^{th} row equation and for the t^{th} column equation, and *zero* elsewhere. *Hence, the coefficient of x_{st} in the canonical form is obtained by solving for the values imposed on the basic variables when the constants are replaced by $a_s = 1$ and $b_t = 1$, while all other a_i and b_j are zero.* By Corollary 1, the value of a basic variable is the difference (positive or negative) between some partial sum of the a_i (which in this case can only be unity or zero), and some partial sum of the b_j (also unity or zero). This difference must clearly be $+1$, 0, or -1; (18) is thus established.

Degeneracy.

If all a_i are positive, then degeneracy in the transportation problem can be avoided by considering the class of perturbed problems:

$$
(19)\qquad
\begin{array}{|llcl|l}
\hline
x_{11} & x_{12} & \cdots & x_{1n} & a_1 \\
x_{21} & x_{22} & \cdots & x_{2n} & a_2 \\
\cdot & \cdot & & \cdot & \cdot \\
\cdot & \cdot & & \cdot & \cdot \\
\cdot & \cdot & & \cdot & \cdot \\
x_{m-1.1} & x_{m-1.2} & \cdots & x_{mn} & a_{m-1} \\
x_{m1} & x_{m2} & \cdots & x_{mn} & a_m + n\varepsilon \\
\hline
\multicolumn{4}{l}{b_1 + \varepsilon \quad b_2 + \varepsilon \ldots b_n + \varepsilon} &
\end{array}
$$

For our discussion, *we shall assume that the last row equation has been omitted as redundant.* An arbitrary basic solution is chosen by the process of Theorem 1, except that basic variables are not selected in the last row until all other rows have been eliminated. If, at any step, there is a tie for $\varepsilon = 0$ between a reduced row and column total, then that total with the smallest coefficient of ε is selected for the minimum.

THEOREM 6: *It is not possible that there be a tie for minimum and thus a degeneracy in the basic solution for the subsequent perturbed problem.*

PROOF: The coefficient of ε in any row total of the reduced array is either zero or negative, since it is composed of a non-vacuous partial sum of the a_i (excluding $a_m + n\varepsilon$), minus a (possibly vacuous) partial sum of the $b_j + \varepsilon$. Similarly, the coefficient of ε for any column total of the reduced array is always positive, since it is composed of a (non-vacuous) partial sum of the $b_j + \varepsilon$, minus a (possibly vacuous) partial sum of the a_i (excluding $a_m + n\varepsilon$).

The adjusted row totals in each reduced array will always remain positive (non-zero) for $\varepsilon = 0$, while the column totals will be nonnegative. We can see this inductively. Assume all a_i are positive and all b_j are nonnegative. Suppose this is still true for some reduced array, so that $a_i' = \alpha - p\varepsilon$ (where α is positive and p is nonnegative), and that $b_j' = \beta + q\varepsilon$ (where β is nonnegative and q is positive). Now, if x_{ij} becomes a basic variable, then its value is Min $[(\alpha - p\varepsilon), (\beta + q\varepsilon)]$. For $\alpha \leq \beta$, the row total is satisfied and the new column total becomes $(\beta - \alpha) + (p + q)\varepsilon$, where $(\beta - \alpha)$ is nonnegative and $(p + q)$ is positive. On the other hand, if $\beta < \alpha$, then, for ε in some range, $0 < \varepsilon < \varepsilon_0$, the column total is satisfied, and the new row total becomes $(\alpha - \beta) - (p + q)\varepsilon$, with $(\alpha - \beta)$ and $(p + q)$ both positive.

This establishes the assertion that the *initial* basic solution is nondegenerate for a positive range of ε in the neighborhood of $\varepsilon = 0$. If we now perform an iteration of the simplex method, the new basic solution is feasible for some positive range of ε. Since the basis (formed by excluding the last row equation) is *triangular*, we have (by a repetition of the same argument as above) that the values of the new basic variables are of the form $\alpha + \varepsilon\beta$, where either α is positive or it is zero and β is positive. Hence, the new basic solution must be nondegenerate for some range of ε, $0 < \varepsilon < \varepsilon_1$ and the first two basic solutions remain feasible and nondegenerate for any $0 < \varepsilon < \text{Min}\,(\varepsilon_0, \varepsilon_1)$.

In general, for any sufficiently small ε, there will be a positive (non-zero) decrease in the value imposed on z after each cycle. Orden [1956-1] has shown that no basic feasible solution can be degenerate if $0 < \varepsilon < 1/n$. (See Problems 14 and 15.) Thus, no basis can be repeated, and the algorithm will terminate in a finite number of steps.

14-3. COMPUTATIONAL ALGORITHM FOR THE TRANSPORTATION PROBLEM

A notable feature of this problem is that, whereas linear programs typically require hand-operated calculators or, for larger problems, high-speed computing machinery, the transportation problem frequently is best solved by nothing more sophisticated than pencil and paper, since additions and subtractions are the only calculations required.

The standard transportation array § 14-2-(3) is repeated here for convenience. The array appearing here, however, is slightly different in that it contains certain Theta symbols. These are part of the computational procedure and will be explained later in this section.

(1)

x_{11} ; c_{11}	$x_{12} - \theta$; c_{12}	c_{13}	c_{14}	$+\theta*$ (enter) ; c_{15}	a_1 ; u_1
c_{21}	$x_{22} + \theta$; c_{22}	c_{23}	$x_{24} - \theta$; c_{24}	c_{25}	a_2 ; u_2
c_{31}	c_{32}	x_{33} ; c_{33}	$x_{34} + \theta$; c_{34}	$x_{35} - \theta$; c_{35}	a_3 ; u_3
b_1 ; v_1	b_2 ; v_2	b_3 ; v_3	b_4 ; v_4	b_5 ; v_5	Implicit ← Prices ↑

Finding a Good Starting Solution.

Computationally, the starting rule of solution given in the last section is not too practical, since usually the number of iterations required to achieve optimality can be greatly reduced if the basic set is selected with some reference to the values of the coefficients in the objective form. Several rules for selection can be found in the literature; one is

The Least-Cost Rule: Scan the

$$\text{(2)} \qquad \text{Unit Cost Array:} \quad \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ c_{m1} & c_{m2} & \cdots & c_{mn} \end{bmatrix}$$

for the *smallest* c_{ij} and choose the first basic variable, x_{pq}, such that

$$\text{(3)} \qquad c_{pq} = \operatorname*{Min}_{(i,j)} c_{ij}$$

The value of x_{pq} is chosen to be the minimum of its row or column total, and the row or column where the minimum is attained is then ineligible for the assumption of further increases in the values of its variables; if both the row and the column totals are minimum simultaneously, then either the row or the column is made ineligible, but not both. For subsequent entries, find the smallest cost factor, c_{ij}, among those remaining in eligible squares, and set the corresponding value of x_{ij} as large as is consistent with row and column totals (and with the values already entered). In all, $m + n - 1$ entries will be made. If any of these $m + n - 1$ values is zero, it is important that a zero symbol be entered to distinguish this zero of a basic variable from the zero of a non-basic variable (the latter being indicated by the lack of any numerical entry). The steps that follow apply to the initial or to any subsequent basic solution.

Computing the Values of the Implicit Prices (Simplex Multipliers).

First of all, since one equation must be redundant, we may set one of the prices at an arbitrary value. A good convention is to find a row or column having a great many basic variables and to set its corresponding price at zero. The remaining u_i and v_j are then so chosen that

$$\text{(4)} \qquad c_{ij} - u_i - v_j = 0, \quad \text{if } x_{ij} \text{ is basic}$$

We can determine such prices by scanning the squares corresponding to basic variables until one is found for which either the row price, u_i, or the column price, v_j, has already been evaluated; subtracting either price from c_{ij} determines the other price. The fundamental theorem on the triangularity of the basis guarantees that repeated scanning will result in the evaluation of all u_i and v_j.

Optimality Criterion: By inspection, every unit cost, c_{ij}, is compared with the sum of the implicit prices of its row and column. If their differences, which are the relative cost factors, are all nonnegative, that is, if we have

$$\bar{c}_{ij} = c_{ij} - (u_i + v_j) \geq 0 \tag{5}$$

for every square in the array, then the basic solution is *optimal* and the problem is finished.

Finding a Better Basic Solution.

If some $\bar{c}_{ij}$ is negative, then a non-basic variable, x_{st}, is *entered* into the basic set, replacing one of the $m + n - 1$ basic variables which is dropped from the basic set and becomes just another non-basic variable.

Choose x_{st} to be the new basic variable, where

$$c_{st} - u_s - v_t = \operatorname*{Min}_{(i,j)} [(c_{ij} - u_i - v_j) < 0] \tag{6}$$

The symbol, $+\theta$, is entered in square (s, t) to indicate that a value, called θ, will be given to the non-basic variable, x_{st}. Next, assuming that $x_{st} = \theta$, the basic entries are symbolically adjusted to compensate; we will append $(+\theta)$ to some, $(-\theta)$ to others, and leave the rest unchanged. *Because of the property of basis triangularity, it will always be possible to evaluate the basic variables, whatever be the values assigned to the non-basic variables, by repetitive scanning of the rows and columns for one in which only a single basic entry remains undetermined.* This entry is symbolically adjusted (if necessary), and the scanning is repeated until all entries have been considered.

The symbol, θ, is replaced by the largest numerical value which does not require any basic entries to be nonnegative; that is, θ takes on the value, x^o_{pq}, of the smallest entry to which the symbol $(-\theta)$ is appended, so that $x^o_{pq} - \theta$ becomes zero. Thus, x_{pq} is the variable to be dropped from the basic set. (If several variables are tied for smallest entry,[1] only one of them is selected for rejection; the choice can be made randomly (see § 6-1) or by the special perturbation procedure given in § 14-2-(19).)

Using the value of θ so determined, all the basic entries are recomputed and will constitute a new basic solution. Thereafter, as necessary, we repeat the cycle.

Three examples follow. The first is the original example due to Hitchcock, the second is an "assignment" problem (a transportation problem in which each x_{ij} equals zero or one, while all a_i and b_j are unity (see Chapter 15)); the third illustrates the perturbation method for avoiding degeneracy.

[1] This is the case of degeneracy. It is not known whether circling can occur in the transportation case, but a simple procedure for avoiding degeneracy is illustrated in Example 3 and represents only a trivial amount of extra work.

TABLE 14-3-I
Example 1 (Hitchcock)
Cycle 0

					Row Totals
	10	5	6	**25** 7	**25** −1
	8	**$20 - \theta$** 2	7	**$5 + \theta$** 6	**25** −2
	15 9	**$+\theta$*** (enter) 3	**30** 4	**$5 - \theta$** (drop) 8	**50** 0
Column Totals	**15** 9	**20** 4	**30** 4	**35** 8	Implicit ← Prices ↑

Cycle 1 (*Optimal*)

	10	5	6	**25** 7	**25** 0
	8	**15** 2	7	**10** 6	**25** −1
	15 9	**5** 3	**30** 4	8	**50** 0
	15 9	**20** 3	**30** 4	**35** 7	Implicit ← Prices ↑

TABLE 14-3-II

Example 2 (4×4 Assignment Problem)

Cycle 0

					Row Totals
	 14	**0 + θ** 5	**0 − θ** (drop) 8	**1** 5	**1** 0
	1 2	 12	 6	 7	**1** −1
	 7	 8	**1** 3	 9	**1** −5
	0 2	**1 − θ** 4	**θ*** (enter) 6	 10	**1** −1
Column Totals	**1** 3	**1** 5	**1** 8	**1** 5	Implicit ← Prices ↑

Cycle 1 (*Optimal*)

 14	**0** 5	 8	**1** 5	**1** 0
1 2	 12	 6	 7	**1** −1
 7	 8	**1** 3	 9	**1** −4
0 2	**1** 4	**0** 6	 10	**1** −1
1 3	**1** 5	**1** 7	**1** 5	Implicit ← Prices ↑

TABLE 14-3-III
Example 3 (Perturbation)
Cycle 0

					Row Totals
	14	5	8	1 5	1 -5
	$1-\theta$ 2	12	6	$+\theta^*$ (enter) 7	1 0
	7	8	1 3	9	1 -3
	$\epsilon+\theta$ 2	$1+\epsilon$ 4	ϵ 6	$\epsilon-\theta$ (drop) 10	$1+4\epsilon$ 0
Column Totals	$1+\epsilon$ 2	$1+\epsilon$ 4	$1+\epsilon$ 6	$1+\epsilon$ 10	Implicit ← Prices ↑

Cycle 1 (*Optimal*)

14	5	8	1 5	1 -2
$1-\epsilon$ 2	12	6	ϵ 7	1 0
7	8	1 3	9	1 -3
2ϵ 2	$1+\epsilon$ 4	ϵ 6	10	$1+4\epsilon$ 0
$1+\epsilon$ 2	$1+\epsilon$ 4	$1+\epsilon$ 6	$1+\epsilon$ 7	Implicit ← Prices ↑

14-4. PROBLEMS

1. (Review.) Prove that $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$ is a necessary and sufficient condition for the feasibility of a transportational problem.
2. (Review.) Prove there are $m + n - 1$ independent equations in the classical transportation problem, and that any equation can be dropped as the redundant equation.
3. (Review.) What is the dual of the transportation problem?
4. (Review.) Prove that any basis of the transportation problem's dual is triangular.
5. (Review.) Show that the simplex multipliers u_i and v_j are integers if c_{ij} are integers and u_1 is an integer.
6. Prove that in a regular transportation problem ($u_m = 0$) the values of the implicit prices are always $+1$ or 0 or -1 if all $c_{ij} = 0$ except $c_{11} = 1$.
7. Prove for the classic transportation problem that the unit costs c_{il} of any column l can be replaced by $c_{il} + c_l$ without affecting the optimal solution; similarly, for any row k, c_{kj} may be replaced by $c_{kj} + r_k$.
8. Prove that the classic transportation problem with some (or all) $c_{ij} < 0$ can be replaced by an equivalent problem where all $c_{ij} > 0$.
9. Suppose corresponding values of c_{ij} in two rows differ by a constant; show that the two rows can be combined into a single row.
10. (a) Solve Example 1 (Table 14-3-I) using the perturbation method.
    (b) Solve the transportation problem given in Fig. 16-1-I.
11. Prove every $k \times k$ sub-determinant of coefficients of a transportation problem has value $+1$, 0, or -1.
12. Solve the transportation problem of Chapter 1.
13. Solve the transportation problem of Chapter 3.
14. (Orden [1956-1].) Prove that if a_i, b_j are integers for $i = 1, 2, \ldots, m$; $j = 1, 2, \ldots, n$ and if b_j are replaced by $b_j + (1/n)$ and a_m by $a_m + 1$, then the new problem is never degenerate for a basic feasible solution and the corresponding solution for the unperturbed problem is always feasible. How can this be used to guard against the possibility of circling?
15. (Orden [1956-1].) With reference to Problem 14, show that fractions can be avoided in applying the simplex algorithm if the original b_j are replaced by $nb_j + 1$ and the a_i by na_i except a_m by $na_m + n$.
16. (Unsolved.) Can a degenerate transportation problem ever circle? (See Chapter 10.) If the answer is no, is a perturbation scheme required such as that described in Chapter 10 or such as the simpler one given in § 14-2-(19) and in Problems 14 and 15?

REFERENCES

Beckmann, McGuire, and Winsten, 1956-1
Berge, 1958-1
Charnes and Cooper, 1959-1
Dantzig, 1951-2
Dantzig and Fulkerson, 1954-1
Dennis, 1959-1
Dwyer, 1955-1
Flood, 1953-1, 1960-1
Ford and Fulkerson, 1955-1, 1956-1, 1960-1
Gale, 1960-1
Garvin, 1960-1
Gass, 1958-1
Gerstenhaber, 1958-1
Hadley, 1961-2
Heller, 1957-1
Heller and Tompkins, 1956-1
Hitchcock, 1941-1
Houthakker, 1955-1
Iri, 1960-1
Kantorovich, 1939-1, 1942-1
Kantorovich and Gavurin, 1949-1
Koopmans, 1947-1
Koopmans and Reiter, 1951-1
Kuhn, 1955-1, 1957-1
Motzkin, 1952-1
Munkres, 1957-1
Prager, 1957-1
Shettz, 1959-1
Tucker, 1950-1

CHAPTER 15

OPTIMAL ASSIGNMENT AND OTHER DISTRIBUTION PROBLEMS

The transportation problem, as set forth in the preceding chapter, appears to treat a rather narrow situation. However, the method developed for dealing with the problem may be extended to certain cases which are different in appearance but which can all be shown to be equivalent to the classical case:

1. *Optimal Assignment*

   We shall take up the problem of optimally assigning tasks to operators.

2. *Allocation When Surplus and Deficit Are Allowed*

   We shall give a means for dealing with the transportation array when the assumption of exact sums for the rows and columns is relaxed.

3. *Fixed Values*

   We shall explore the problems that arise when some of the variables in the array must assume certain predetermined values (e.g., zero).

15-1. THE OPTIMAL ASSIGNMENT PROBLEM

The term "assignment" describes the problem concerned, typically, with finding the best way to assign n persons to n jobs, assuming that the individuals vary in their suitability for a particular job. We shall assume that, by means of performance tests, the "value" of assigning the i^{th} person to the j^{th} job can in some sense be determined. The negative of this value (i.e., the unit cost) will be denoted by c_{ij}. Suppose for each i the i^{th} person has been assigned to job p_i; the total cost, z, for this assignment of personnel will be, we assume, the sum of the individual costs, that is,

$$z = \sum_{i=1}^{n} c_{ip_i}$$

The numbers $p_1, p_2, \ldots, p_n$ constitute a *permutation* of $1, 2, \ldots, n$; hence, the optimal assignment problem is to find a minimizing permutation.

Stated in this form the problem is evidently combinatorial. There are n ways in which to choose p_1, $(n - 1)$ ways remaining to choose $p_2, \ldots,$ or

$$n! = (n)(n-1)(n-2)\ldots(2)(1)$$

different possibilities. For $n = 6$, $n! = 720$, and one might pick the smallest value of z after calculating the costs of all the 720 possible assignments. But the number of possibilities grows rapidly. For example, $12! \cong 4.79 \times 10^8$. To attempt the solution of even a 12×12 assignment problem by seeking all the permutations is not very practical even on present-day computers. However, the problem can be reformulated as a 12×12 transportation problem which, through the procedures described earlier, can be solved by hand in a few minutes. For this purpose, let

$$x_{ij} = \begin{cases} 1 & \text{if the } i^{\text{th}} \text{ individual is assigned to the } j^{\text{th}} \text{ job} \\ 0 & \text{if not} \end{cases} \tag{1}$$

Because (we assume) each person can be assigned only one job, we have

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad \text{for } i = 1, 2, \ldots, n \tag{2}$$

and, because each job is assigned to only one person,

$$\sum_{i=1}^{n} x_{ij} = 1 \qquad \text{for } j = 1, 2, \ldots, n \tag{3}$$

Square arrays of nonnegative numbers, x_{ij}, with the property that all row and all column sums are unity, frequently appear in statistical theory. They are called *doubly stochastic* matrices, and the x_{ij} are interpreted as probabilities (not necessarily zero or one). When such arrays have all x_{ij} zero or one, they are called *permutation matrices*.

The objective of the assignment problem is to choose x_{ij}, satisfying (1), (2), and (3), in such a way that the total cost,

$$z = \sum_{i=1}^{n} c_{ip_i} = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{4}$$

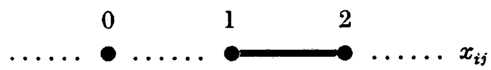
is minimized.

Condition (1), however, forestalls the direct application of linear programming methods to this formulation. Conditions of this type can, in themselves, make a problem very difficult to solve. The condition, "$x_{ij} = 0$ or 1," assigns x_{ij} a *disconnected* range composed of *discrete* values, as diagrammed in (5).

$$\begin{array}{c} \qquad 0 \qquad\quad 1 \\ \ldots\ldots \bullet \ldots\ldots \bullet \ldots\ldots\ x_{ij} \end{array} \tag{5}$$

Another example of a disconnected range which can take a problem out of the direct reach of linear programming methods is the condition, "$x_{ij} = 0$ or $1 \leq x_{ij} \leq 2$," depicted in (6).

(6)



In Chapter 26, we shall develop general methods for handling conditions such as these, which generate variables having disconnected or discrete ranges.

In lieu of the assignment problem formulated above, we shall show that we can obtain an equivalent transportation problem simply by replacing (1) with the condition,

$$0 \leq x_{ij} \tag{7}$$

Garrett Birkhoff [1946-1] showed that the set of permutation matrices is given by the extreme points of the convex set defined by the conditions for a doubly stochastic matrix with nonnegative entries; i.e.,

THEOREM 1: *An optimal solution of the assignment problem* {(1), (2), (3), *and* (4)} *is the same as an optimal solution of the linear programming problem given by* {(2), (3), (4), *and* (7)}.

PROOF: Each basic feasible solution has the property that the x_{ij} values are either zero or one, for Theorem 3, § 14-2, states that if the row and column totals are integers, then so are the values of the basic variables, and it is clear from (7) and (2), or (7) and (3), that the only integer values possible for x_{ij} are zero and one. It follows that an optimal basic feasible solution will be a permutation. Since all permutation solutions satisfy conditions (2), (3), (4), and (7), and since the minimizing solution is a permutation, it must also be a minimizing solution for the original assignment problem. Von Neumann [1953-1] establishes Birkhoff's theorem by reducing an assignment problem to an interesting matrix game. See also [Marcus, 1960-1].

Degeneracy.

Degeneracy, as earlier defined, occurs whenever one or more of the basic variables are zero. The linear program equivalent to an assignment problem has the property that every basic solution is degenerate, since exactly n basic variables must receive unit value, and the remaining $n - 1$ basic variables must, therefore, all be zero. If the number of basic variables with zero value is taken as measuring the "extent" of degeneracy, the equivalent linear program is seen to be highly degenerate. As pointed out earlier, it is not known whether circling can occur in transportation problems. (See Chapter 10.)

In practical work where, because of degeneracy, there is ambiguity as to which basic variable should be dropped, various procedures can be adopted. A simple rule would be to choose the variable, x_{rs}, whose corresponding c_{rs}

is maximal. The special ε-perturbation method developed in § 14-2-(19) for avoiding degeneracy in transportation problems, requires but a trivial amount of extra work and can be used whenever insurance against circling is necessary. A second method, guaranteed with probability one, is simply to resolve the degeneracy by random choice (see § 6-1).

Equivalence of Transportation and Assignment Problems.

We have already seen that the assignment problem is but a special case of the transportation problem. We shall now show that, *mutatis mutandis*, the transportation problem is, in its turn, a special case of the assignment problem, so that the two problems are completely equivalent. We shall assume that a_i and b_j are integers; if they were rational fractions, then, through a change of units, they could immediately be replaced by integers, while if irrational, they could be rationally approximated and then replaced.

A constructive proof will be indicated by example. Consider the transportation problem defined by tableau (8).

(8)

x_{11} $c_{11} = 10$	x_{12} $c_{12} = 5$	x_{13} $c_{13} = 6$	x_{14} $c_{14} = 7$	$2 = a_1$
x_{21} $c_{21} = 8$	x_{22} $c_{22} = 2$	x_{23} $c_{23} = 7$	x_{24} $c_{24} = 6$	$3 = a_2$
x_{31} $c_{31} = 9$	x_{32} $c_{32} = 3$	x_{33} $c_{33} = 4$	x_{34} $c_{34} = 8$	$1 = a_3$
$2 = b_1$	$1 = b_2$	$1 = b_3$	$2 = b_4$	

The first row equation

$$x_{11} + x_{12} + x_{13} + x_{14} = 2$$

is replaced by two equations (since $a_1 = 2$),

$$x'_{11} + x'_{12} + x'_{13} + x'_{14} = 1$$
$$x''_{11} + x''_{12} + x''_{13} + x''_{14} = 1$$

The second row equation

$$x_{21} + x_{22} + x_{23} + x_{24} = 3$$

is, similarly, replaced by three row equations (since $a_2 = 3$),

$$x'_{21} + x'_{22} + x'_{23} + x'_{24} = 1$$
$$x''_{21} + x''_{22} + x''_{23} + x''_{24} = 1$$
$$x'''_{21} + x'''_{22} + x'''_{23} + x'''_{24} = 1$$

Since $a_3 = 1$, the third row equation is left untouched. This results in an equivalent transportation problem with all row totals unity, as given in (9).

(9)

x'_{11} $c'_{11} = 10$	x'_{12} 5	x'_{13} 6	x'_{14} 7	$1 = a'_1$
x''_{11} $c''_{11} = 10$	x''_{12} 5	x''_{13} 6	x''_{14} 7	$1 = a''_1$
x'_{21} 8	x'_{22} 2	x'_{23} 7	x'_{24} 6	$1 = a'_2$
x''_{21} 8	x''_{22} 2	x''_{23} 7	x''_{24} 6	$1 = a''_2$
x'''_{21} 8	x'''_{22} 2	x'''_{23} 7	x'''_{24} 6	$1 = a'''_2$
x_{31} 9	x_{32} 3	x_{33} 4	x_{34} 8	$1 = a_3$
2	1	1	2	

In this table, $c_{11} = 10$ has been replaced by ($c'_{11} = 10$ *and* $c''_{11} = 10$), etc. Any solution of (9) yields a solution of (8) upon setting

$$\begin{aligned} x'_{11} + x''_{11} &= x_{11} \qquad & x'_{21} + x''_{21} + x'''_{21} &= x_{21} \\ x'_{12} + x''_{12} &= x_{12} & x'_{22} + x''_{22} + x'''_{22} &= x_{22} \\ x'_{13} + x''_{13} &= x_{13} & x'_{23} + x''_{23} + x'''_{23} &= x_{23} \\ x'_{14} + x''_{14} &= x_{14} & x'_{24} + x''_{24} + x'''_{24} &= x_{24} \end{aligned}$$

Because of the linearity of this relationship and the equality of the corresponding cost coefficients in (9), it is clear that the same values of z are obtained for (8). Moreover, if the original problem has a solution, then a solution of (9) can be obtained by apportioning the x_{ij} between the rows in any arbitrary manner, provided the row totals are unity. For example, the x_{ij} values in the first row of (8) can be divided equally between the two corresponding rows of (9); the second row values can be divided into three equal parts. The same value of z is obtained. From these observations, it follows that a minimizing solution of the first problem corresponds to a minimizing solution of the second, and conversely. Hence, the two problems are equivalent.

So far, however, only the row equations of (9) have been modified. To obtain the corresponding assignment problem, it is necessary to modify the column equations in a similar manner. The result is the assignment array given below, with y_{ij} denoting the new variables.

(10)

y_{11} 10	y_{12} 10	y_{13} 5	y_{14} 6	y_{15} 7	y_{16} 7	1
y_{21} 10	y_{22} 10	y_{23} 5	y_{24} 6	y_{25} 7	y_{26} 7	1
y_{31} 8	y_{32} 8	y_{33} 2	y_{34} 7	y_{35} 6	y_{36} 6	1
y_{41} 8	y_{42} 8	y_{43} 2	y_{44} 7	y_{45} 6	y_{46} 6	1
y_{51} 8	y_{52} 8	y_{53} 2	y_{54} 7	y_{55} 6	y_{56} 6	1
y_{61} 9	y_{62} 9	y_{63} 3	y_{64} 4	y_{65} 8	y_{66} 8	1
1	1	1	1	1	1	

Some Typical Uses of the Assignment Model.

Machine Set-up Time (See Problem 1). A job has n tasks to be assigned concurrently to n different machines. Each machine must receive an adjustment so as to adapt it to the particular task assigned. Cost is the time it takes to do this, and total cost is the sum of the man-hours thus consumed. The time it takes to set up a machine depends on what the machine was doing previously; if the same kind of task, it will not be necessary to reset the machine, or if the same raw material is used, it may not be necessary to remove residual material, and so forth.

The Marriage Game (See Problem 2). A pioneering colony of 10 bachelors is joined by 10 prospective brides. After a short period of courting, it is decided to have an immediate ceremony. Each bride is given a list of 10 names on which she is to list her preferences in a scale of 10, e.g., she may assign her first choice the number 10, her second choice the number 9, etc. She may also cross out names unacceptable to her. We assume that the sum of the assigned numbers constitutes a valid measure of the anticipated "happiness" of the colony in marital bliss [Halmos and Vaughan, 1950-1].

In taking leave of this example the reader may be amused by the following story:

In 1955, at the summer meeting of the Operations Research Society in Los Angeles, I (the author) was interviewed by the press. The reporter turned out to be the brother of my small daughter's piano teacher, and so we became quite friendly. I explained to him that linear programming models originated in the Air Force, and I described their growing application to industrial problems. It became obvious that this veteran Hollywood reporter was having a hard time seeing how to make the material into exciting news copy. In desperation I suggested, "How about something with sex appeal?" "Now you're talking," he said. "Well," I continued, "an interesting by-product of our work with linear programming models is a mathematical proof that of all the possible forms of marriage (monogamy, bigamy, polygamy, etc.), monogamy is the best." "You say monogamy is the best of all possible relations?" he queried. "Yes," I replied. "Man," he said, shaking his head in the negative, "you've been working with the wrong kind of models."

15-2. ALLOCATION WITH SURPLUS AND DEFICIT

It is often possible to identify one set of totals, say a_i, as the amount *available at origins* and b_j as the amount *required at destinations*, but in some applications it may be impossible (or unprofitable) to supply all that is required or to ship all that is available. Accordingly, the array takes the form (1), with $\Sigma a_i \geq \Sigma b_j$,

(1)

Origins i	Destinations j: 1 2 . . . n	Row Totals Available
1	x_{11} x_{12} . . . x_{1n}	$\leq a_1$
2	x_{21} x_{22} . . . x_{2n}	$\leq a_2$
. . .	. . . .	. . .
m	x_{m1} x_{m2} . . . x_{mn}	$\leq a_m$
Column Totals Required	$\leq$ $\leq$ . . . $\leq$ b_1 b_2 . . . b_n	

where the inequality symbols indicate that the row and column sums do not exceed the corresponding totals, a_i and b_j.

Let x_{i0} denote the *surplus* at the i^{th} origin and x_{0j} the *shortage* at the j^{th} destination, and let x_{00} be the total amount shipped from all origins to all destinations, so that

(2)
$$x_{00} = \sum_{i=1}^{m} \sum_{j=1}^{n} x_{ij}$$

Upon augmentation with a surplus column and a shortage row, to accommodate the slack variables, x_{i0} and x_{0j}, array (1) takes the form (3).

(3)

	Surplus Column	Destinations	Row Totals Available
Shortage Row	x_{00}	$x_{01} \ . \ . \ . \ x_{0n}$	$= \sum_1^n b_j$
Origins	x_{10} $\cdots$ x_{m0}	$x_{11} \ . \ . \ . \ x_{1n}$ $\cdots\cdots\cdots$ $x_{m1} \ . \ . \ . \ x_{mn}$	$= a_1$ $\cdots$ $= a_m$
Column Totals Required	$=$ $\sum_1^m a_i$	$= \ . \ . \ . \ =$ $b_1 \ . \ . \ . \ b_n$	

This is a standard transportation array, because it is clear from the way we have defined the variables that

(4) $$\sum_{i=1}^{m} a_i - \sum_{i=1}^{m} x_{i0} = \sum_{j=1}^{n} b_j - \sum_{j=1}^{n} x_{0j} = \sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij} = x_{00}$$

We have thus converted (1) to the standard format, but it should be noted that if there is no penalty associated with failure to deliver the required amounts, then there is really no problem at all; simply ship nothing. A meaningful problem exists only where failure to ship means a loss of revenue or good will, i.e., where positive cost factors, c_{i0} or c_{0j}, are assigned to surplus or shortage.

Surplus only: In case the availabilities exceed the requirements (i.e., $\Sigma a_i > \Sigma b_j$), but requirements must be met exactly, the array takes the form (5).

(5)

	Surplus Column	Destinations	Row Totals Available
Origins	x_{10} x_{20} $\cdots$ x_{m0}	$x_{11} \quad x_{12} \ . \ . \ . \ x_{1n}$ $x_{21} \quad x_{22} \ . \ . \ . \ x_{2n}$ $\cdots\cdots\cdots\cdots$ $x_{m1} \quad x_{m2} \ . \ . \ . \ x_{mn}$	$= a_1$ $= a_2$ $\cdots$ $= a_m$
Column Totals Required	$=$ $\Sigma a_i - \Sigma b_j$	$= \quad = \ . \ . \ . \ =$ $b_1 \quad b_2 \ . \ . \ . \ b_n$	

Shortages only: In case the requirements exceed the availabilities (i.e., $\Sigma b_j > \Sigma a_i$), but all available supplies must be shipped, the transportation array takes the form (6).

(6)

	Destinations	Row Totals Available
Shortage Row	x_{01} x_{02} . . . x_{0n}	$= \Sigma b_j - \Sigma a_i$
Origins	x_{11} x_{12} . . . x_{1n}	$= a_1$
	. . . .	. . .
	x_{m1} x_{m2} . . . x_{mn}	$= a_m$
Column Totals Required	$=$ $=$ . . . $=$ b_1 b_2 . . . b_n	

Theoretical Background.

Upon introduction of slack variables, x_{i0} and x_{0j}, the surplus-shortage problem may be displayed as in (7).

(7)

	x_{01} x_{02} . . . x_{0n}	
x_{10}	x_{11} x_{12} . . . x_{1n}	$= a_1$
. . .	. . . .	. . .
x_{m0}	x_{m1} x_{m2} . . . x_{mn}	$= a_m$
	$=$ $=$ . . . $=$ b_1 b_2 . . . b_n	

All $m + n$ equations are independent, in contrast to the classical transportation case, in which only $m + n - 1$ are independent. For example, it is easy to see that the slack variables constitute a basic set of $m + n$ variables.

Moreover, every basis is triangular. Proceeding as in the proof given earlier for the basic triangularity of transportation problems, let f be the number of basic variables among the surplus variables, x_{i0}, g among the shortage variables, x_{0j}, and h among the non-slack variables, so that

(8) $$n + m = f + g + h$$

	g
f	h

THEOREM 1: *Every basis contains at least one slack variable, i.e.,* $f + g \geq 1$.

PROOF: If some basis has no slack, then it would also constitute a basis for the analogous transportation problem, *without* slack. This can only happen if $\Sigma a_i = \Sigma b_j$, but in this case we know that the number of basic variables cannot exceed $m + n - 1$. We conclude that

$$f + g \geq 1 \tag{9}$$

THEOREM 2: *Every basis is triangular.*

PROOF: If not, then every row (except the shortage row) and every column (except the surplus column) has two or more basic variables. Thus,

$$\begin{aligned} 2m &\leq h + f, \quad \text{and} \\ 2n &\leq h + g, \quad \text{so that} \end{aligned} \tag{10}$$

$$n + m \leq h + \tfrac{1}{2}(f + g) \tag{11}$$

Since, according to (9), $f + g$ is *positive*, (11) implies the *strict* inequality,

$$n + m < h + f + g \tag{12}$$

which contradicts (8).

Thus, the assumption that all non-slack rows and columns contain at least two basic variables apiece must be false, and at least one of them must therefore contain exactly one variable. Upon deleting such a row or column, adjusting the totals as necessary, we may repeat the foregoing argument for the reduced system so derived, so that Theorem 2 is established by mathematical induction.

Pricing.

If u_i and v_j are the simplex multipliers (or "prices") associated with row i, column j, then the relative cost factors are

$$\begin{aligned} \bar{c}_{ij} &= c_{ij} - (u_i + v_j) && \text{for } i \neq 0, j \neq 0, \\ \bar{c}_{0j} &= c_{0j} - v_j && \text{for } j \neq 0, \text{ and} \\ \bar{c}_{i0} &= c_{i0} - u_i && \text{for } i \neq 0 \end{aligned} \tag{13}$$

It may be noted that we need not define slack multipliers, u_0 or v_0, since there is no equation pertaining to row zero or to column zero. For uniformity, however, it is convenient to assign fictitious prices,

$$u_0 = v_0 = 0 \tag{14}$$

We can then characterize the choice of prices as a selection of u_i and v_j such that, if x_{ij} is a basic variable, then

(15) $$u_i + v_j = c_{ij}$$

Optimal Allocation of Receivers to Transmitters. (An example of allocation with slack.)

A certain engine-testing facility is fully using four kinds of instruments: two hundred thermocouples, fifty pressure gauges, fifty accelerometers, and forty thrust meters. Each is measuring one type of characteristic and transmitting data about it over a separate communication channel. There are four types of receivers, each capable of recording one channel of information: two hundred cameras, one hundred fifty oscilloscopes, two hundred fifty-six test instruments called "Idiots," and fifty others called "Hathaways." The set-up time per channel varies among the different types and also according to the kind of data to be recorded. Assuming that all data must be recorded, the problem is to find an allocation of receivers to transmitters which minimizes the total set-up time.

The allocation table takes the form (16).

(16)

Recording Instrument i	Measuring Instrument j: Temp. 1	Pressure 2	Accel. 3	Thrust 4	Total Recording Channels Available
Cameras 1	x_{11} 1	x_{12} 3	x_{13} ∞	x_{14} 1	≤ 200
Oscilloscopes 2	x_{21} .5	x_{22} .5	x_{23} .5	x_{24} .5	≤ 150
"Idiots" 3	x_{31} 2	x_{32} 2	x_{33} 10	x_{34} 2	≤ 256
"Hathaways" 4	x_{41} 1.5	x_{42} 1.5	x_{43} 1.5	x_{44} 1.5	≤ 50
Total Channels to be recorded	200	50	50	40	

The number, c_{ij}, appearing at the lower right of square (i, j) is "cost" or set-up time of assigning a recording channel of the i^{th} type to a measuring channel of the j^{th} type The condition that $c_{13} = \infty$ means a camera is not to be used to record acceleration data. From a procedural point of view, square (1, 3) is to be avoided, if possible, in forming a starting solution, and if avoided initially, it will not thereafter be considered as a candidate in the basic set. (See § 15-3 for further details concerning inadmissible squares.) The objective is to choose nonnegative x_{ij} so as to minimize

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}x_{ij} \tag{17}$$

If slack variables are introduced to measure the number of unused recording channels, we have the transportation array (18).

(18)

Recording Instru-ment	Surplus 0	Measuring Instrument Temp. 1	Pressure 2	Accel. 3	Thrust 4	a_i / u_i
1	**10** / 0	**190** / 1	3	∞	1	**200** / .5
2	0	**10** / .5	**50** / .5	**50** / .5	**40** / .5	**150** / 0
3	**256** / 0	2	2	10	2	**256** / .5
4	**50** / 0	1.5	1.5	1.5	1.5	**50** / .5
b_j / v_j	**316** / −.5	**200** / .5	**50** / .5	**50** / .5	**40** / .5	

The total of the surplus column is the same as the number of channels available minus the number of channels required, i.e.,

$$(200 + 150 + 256 + 50) - (200 + 50 + 50 + 40) = 316$$

The costs for this column, c_{i0}, are all zero, because there is no set-up cost involved in not using a channel.

The basic solution shown was generated by using the rule discussed earlier in this section for choosing a good starting solution. When several c_{ij} were tied for minimum, squares were chosen which caused rows or columns having high c_{ij} values to be deleted. For example, because the set-up costs for recording most transmissions are highest with the instruments in rows *3* and *4*, allocations to surplus were made in these rows first. This left rows *1* and *2*, and columns *1*, *2*, *3*, and *4*. Because the costs in the second row are all exceeded by the corresponding first-row costs for these columns, allocations to surplus were made next in row *1*. The remaining allocations follow the rule given earlier. Computing the implicit prices, one can easily establish that this basic solution is optimal.

When recording instruments are in short supply (or are not of the most suitable types), a decision must be reached as to how much of each kind of data *not to record*. Consider the following unit-costs which the engineers assigned to the shortage row and surplus column:

$$c_{01} = 10,\ c_{02} = 10,\ \ c_{03} = 4,\ c_{04} = 100,$$
$$c_{10} = 0,\ \ c_{20} = -1,\ c_{30} = 0,\ c_{40} = 0,\ \text{and}\ c_{00} = 0$$

For example, it is 25 times more costly to neglect thrust data ($c_{04} = 100$) than to neglect acceleration data ($c_{03} = 4$). In general, however, it is less costly to *record* data than to *neglect* it.

The *negativity* of c_{20} is an expression of the fact that unused oscilloscopes may be profitably employed outside the model; how profitably, may be difficult at times to determine. As noted in § 14-4, Problem 7, the optimal solution is unaffected when we increase all the c_{ij} in a row or column of the equivalent transportation array by a constant. Accordingly, in experimenting on the effect of changes in the c_{ij}, it is advisable to hold at least one cost factor at a fixed value in a row and some column.

This problem may be treated by means of the standard transportation array, (19), with a shortage row and a surplus column. *Alternatively*, it may

(19)

Recording Instru-ment i	Surplus 0	Measuring Instrument				a_i
		Temp. 1	Pressure 2	Accel. 3	Thrust 4	u_i
Shortage 0	**340** 0	10	10	4	100	**340** .5
1	**10** 0	**190** 1	3	∞	1	**200** .5
2	−1	**10** .5	**50** .5	**50** .5	**40** .5	**150** 0
3	**256** 0	2	2	10	2	**256** .5
4	**50** 0	1.5	1.5	1.5	1.5	**50** .5
b_j v_j	**656** −.5	**200** .5	**50** .5	**50** .5	**40** .5	

be set up as in (7). From a computational viewpoint, the two methods are almost identical, as may be seen by perusal of (20) and (21).

Phase II—Cycle 0

(20)

	10	10	4	100	
10 − θ / 0	**190 + θ** / 1	3	∞	1	**200** / 0
θ* / −1	**10 − θ** / 0.5	**50** / 0.5	**50** / 0.5	**40** / 0.5	**150** / −0.5
256 / 0	2	2	10	2	**256** / 0
50 / 0	1.5	1.5	1.5	1.5	**50** / 0
	200 / 1	**50** / 1	**50** / 1	**40** / 1	

Phase II—Cycle 1—(Optimal)

(21)

	10	10		100	
0	**200** / 1	3	∞	1	**200** / −5
10 / −1	**0** / 0.5	**50** / 0.5	**50** / 0.5	**40** / 0.5	**150** / −1.0
256 / 0	2	2	10	2	**256** / 0
50 / 0	1.5	1.5	1.5	1.5	**50** / 0
	200 / 1.5	**50** / 1.5	**50** / 1.5	**40** / 1.5	

15-3. FIXED VALUES AND INADMISSIBLE SQUARES

In solving transportation problems, it quite often happens that some of the variables must assume predetermined values. In the preceding section, for example, it was not possible to assign cameras to record acceleration data, so that x_{13} had to be zero. Similarly, if there exists no route from an origin i to a destination j in a network, then the variable x_{ij} must be zero. In the problem of assigning people to jobs, certain assignments may be mandatory; for example, assigning a physician to a medical position. Note, however, that *a fixed variable can always be replaced by a zero-restricted variable after subtracting its predetermined value from the corresponding row and column totals*. We will designate the squares associated with zero-restricted variables as *inadmissible* and will shade such squares when they appear in a tableau.

If only a few squares are inadmissible, the best practical procedure is to attempt an initial basic feasible solution by using the *least-cost rule* discussed earlier, § 14-3-(3), but there are a great many problems in which inadmissible squares can be avoided only at the expense of selecting basic variables having higher unit-costs than would be suggested by this rule. Moreover, if too many squares are inadmissible, there may be no solution, or no readily discernible basic solution, using a set of variables selected from the admissible squares.

If there appears to be no way of avoiding inadmissible squares, they can be used to furnish artificial variables for a simplex Phase I, in which a basic feasible solution will be constructed if possible. For this purpose, the infeasibility form, $w = \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij}x_{ij}$, is used, so that the c_{ij} entries in the array are replaced by

$$d_{ij} = \begin{cases} 1 & \text{if } (i,j) \text{ is inadmissible} \\ 0 & \text{if } (i,j) \text{ is admissible} \end{cases}$$

If a feasible solution exists, then Min w is zero, but if Min w is positive, then the problem is infeasible.

Even if the problem is feasible, it may happen that some inadmissible variables remain in the basic set at the end of Phase I. If so, they will be zero in value, and they must remain so throughout the remaining procedure. Before initiation of Phase II, any non-basic x_{ij} will be dropped from further consideration if its relative infeasibility factor $\bar{d}_{ij}$ is positive, i.e., if

$$d_{ij} - u_i - v_j > 0$$

where u_i and v_j are the implicit prices associated with the infeasibility form at the end of Phase I.

EXAMPLE: Find an optimal feasible solution to the transportation problem (1). The algorithm is initiated with any basic solution such as the one shown in (2). The latter has two inadmissible basic variables x_{42}, x_{34} indicated by heavy-bordered squares.

(1)

	$c_{12} = 3$	$c_{13} = 3$		$7 = a_1$
$c_{21} = 2$		$c_{23} = 9$	$c_{24} = 6$	$25 = a_2$
$c_{31} = 7$		$c_{33} = 4$		$8 = a_3$
$c_{41} = 8$			$c_{44} = 5$	$3 = a_4$
$14 = b_1$	$7 = b_2$	$5 = b_3$	$17 = b_4$	

(2)

Phase I—Cycle 0
Starting Solution
(c_{ij} replaced by d_{ij})

	7 0	 0		7 −1
14 − θ 0		 0	11 + θ 0	25 0
θ* 0		5 0	3 − θ 1	8 1
 0	0 1		3 0	3 0
14 0	7 1	5 −1	17 0	

(3)

Phase I—Cycle 1 (Feasible)

	7 0	(Drop in Phase II) 0		**7** −1
11 0		0	**14** 0	**25** 0
3 0		**5** 0		**8** 0
0	**0** 1		**3** 0	**3** 0
14 0	**7** 1	**5** 0	**17** 0	

(4)

Phase II—Cycle 1 (Optimal)
(d_{ij} replaced by c_{ij})

	7 3	(Drop in Phase II)		**7** 3
11 2		9	**14** 6	**25** 1
3 7		**5** 4		**8** 6
8	**0** 0		**3** 5	**3** 0
14 1	**7** 0	**5** −2	**17** 5	

Note: In Phase II, x_{13} is dropped because $d_{13} > 0$ at end of Phase I. The value of c_{42} for the artificial variable is arbitrary, for example, $c_{42} = 0$.

15-4. PROBLEMS

1. Find the optimal assignment of 12 tasks to 12 machines if the time, c_{ij}, needed to set up the i^{th} task on the j^{th} machine is that given by the first table on page 333. (See § 15-1.)
2. Find an assignment which gives the greatest total "happiness," where the rating of the j^{th} bachelor by the i^{th} bride is that given in the second table on page 333 [Halmos and Vaughan, 1950-1]. (See § 15-1.)

The Machine-Task Problem

Machines	Tasks											
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)
(1)	79	24	13	53	47	66	85	17	92	47	46	13
(2)	43	59	33	95	55	97	34	55	84	94	26	56
(3)	29	52	26	27	13	33	70	11	71	86	6	76
(4)	88	83	64	72	90	67	27	47	83	62	35	38
(5)	65	90	56	62	53	91	48	23	6	89	49	33
(6)	44	79	86	93	71	7	86	59	17	56	45	59
(7)	35	51	9	91	39	32	3	12	79	25	79	81
(8)	50	12	59	32	23	64	20	94	97	14	11	97
(9)	25	17	39	0	38	63	87	14	4	18	11	45
(10)	68	45	99	0	94	44	99	59	37	18	38	74
(11)	93	36	91	30	44	69	68	67	81	62	66	37
(12)	19	36	5	50	49	94	95	17	63	41	84	1

The Marriage Problem

	Bill	John	Egbert	Cuthbert	Joe	Gaston	Chauncey	Clyde	Newt	Waldo
Jane	9	6	3	×	2	8	7	4	1	5
Mary	3	7	8	2	1	×	5	4	×	6
Chloe	4	2	1	6	×	8	3	9	7	5
Beulah	6	3	5	7	9	×	1	4	2	8
Phoebe	7	5	6	9	1	8	3	×	2	4
Octavia	1	10	8	4	5	3	6	9	2	7
Juliet	6	8	10	9	4	3	5	1	7	2
Myrtle	7	8	4	3	2	6	1	9	5	×
Olga	3	9	4	2	5	6	7	×	8	1
Mabel	9	3	1	8	×	4	2	7	6	5

Note: According to the description in the text, each bride is given a list of 10 names on which she is to list her preferences in a scale of 10, e.g., she may assign her first choice the number 10, her second choice the number 9, etc. She may also cross out names unacceptable to her. Thus, it should be safe to assume that all of the brides will have a number 10 (for first choice), which they do not, according to this problem. It seems that instead they have assigned the number 9 to first choice if one name is crossed off, 8 if two names are crossed off. Is this the same problem?

3. Prove that, for the alternative procedure of § 15-2, the $m + n$ multipliers, u_i and v_j, are determined uniquely by $m + n$ of the equations (14) and (15).
4. (Review.) Show that the system of equations (14), (15) of § 15-2 is triangular.
5. In § 15-2, show that the u_i and v_j are integers if the c_{ij} are integers.
6. In a transportation problem with one price set at zero, say $u_m = 0$, or in a surplus-shortage problem treated by the alternative procedure of § 15-2, prove that, if all the cost factors are zero in value except one which is unity, the implicit prices are always $+1$, or 0, or -1.
7. Why is x_{13} dropped on the first cycle of Phase II for the example in § 15-3? (See the rules of Phase I-Phase II transition of the simplex method in § 5-2.)
8. (a) Show that an extra row (or column) of slack variables with arbitrary unit costs may be introduced into a classical transportation problem without changing the optimal solution.
   (b) Also, show that the solution is unaffected when both an extra surplus row and an extra "over-supply" column of slack variables with arbitrary unit costs are introduced.
   (c) Show in this form there are no redundant equations in the system or multipliers with arbitrary values associated with a basic solution.

REFERENCES

Assignment Problem

Berge, 1958-1
Birkhoff, 1946-1
Dennis, 1959-1
Easterfield, 1946-1
Egerváry, 1931-1
Ford and Fulkerson, 1960-1
Gale, 1960-1
Gass, 1958-1
Gross, 1959-1
Hadley, 1961-2
Halmos and Vaughan, 1950-1
Hoffman, 1956-1, 1960-1
Hoffman and Kuhn, 1956-1
Kendall, 1960-1
Kuhn, 1955-1
Kuhn and Tucker, 1958-1
Marcus, 1960-1
Motzkin, 1956-1
Munkres, 1957-1
Vajda, 1961-1
von Neumann, 1953-1
Votaw, 1952-1
Votaw and Orden, 1952-1

CHAPTER 16

THE TRANSSHIPMENT PROBLEM

16-1. EQUIVALENT FORMULATION OF THE MODEL

In the Hitchcock transportation problem, cities where goods are produced (origins) ship only to cities where goods are consumed (destinations); shipments do not take place between origins or between destinations, nor from destinations to origins. However, while only shipments from origin to final destination appeared in the Hitchcock model, actual shipments might in practice be routed through many intermediate cities.

It is tacitly assumed that shipments between any two cities are always transported via the least-cost routes where cost, c_{ij}, may be in terms of distance, time, or money. In some instances, the amount that can be shipped on a link between two cities may be limited, in which case it may not always be possible to fulfill our tacit assumption of a shortest route. Another point worth noting is that the determination of a shortest route from each origin to every destination, as is necessary for the Hitchcock formulation, might in itself be quite a chore. It would be desirable to have an algorithm which develops this information automatically.

A. Orden [1956-1] proposed a generalized transportation model in which *transshipment* through intermediate cities is permitted. For every city, there is a material-balance equation stating that the amount shipped out minus that shipped in is equal to the net amount produced there (if positive), or net amount consumed there (if negative).

We shall consider therefore a generalized transportation model in which transshipment through intermediate cities is permitted. For every city, there will be a material-balance equation stating:

Gross Supply = Amount Shipped In + Produced
= Amount Shipped Out + Consumed,

or, in equation form,

(1) $$\sum_{i \neq j} x_{ij} + a_j^* = \sum_{k \neq j} x_{jk} + b^* = x_{jj}^* \qquad (j = 1, 2, \ldots, n)$$

where

x_{ij} = total quantity shipped from i to j for $i \neq j$,
x_{jj}^* = gross supply at j,
a_j^* = the production at city j, and
b_j^* = the consumption at city j

If local production for local consumption is excluded from the model, so that either a_j^* or b_j^* is zero, we shall use the symbols a_j, b_j without the star. In general, the net production a_j and consumption b_j are related to a_j^* and b_j^* by

$$(2) \qquad a_j = a_j^* - \text{Min}\,(a_j^*, b_j^*); \qquad\qquad b_j = b_j^* - \text{Min}\,(a_j^*, b_j^*)$$

We shall refer to a_j and b_j as the (net) amounts available and required.

The transshipment problem consists in finding $x_{ij} \geq 0$ and Min z satisfying (1) and the objective equation

$$3) \qquad \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} = z \qquad\qquad \text{where } i \neq j$$

Upon equating the expressions for gross supply given in (1), we obtain a complete transshipment model for n cities. The array of detached coefficients is shown in Table 16-1-I. Excluding the cost factor, each column contains only two non-zero coefficients, $+1$ and -1; more generally, if we allow surplus or shortage, then the rows include slack variables whose corresponding columns contain only one non-zero coefficient $+1$ or -1. (In Chapter 21, systems are considered that have at most two non-zero coefficients in a column but not necessarily equal and opposite in sign.)

TABLE 16-1-I

TRANSSHIPMENT MODEL—NETWORK FORMULATION

(Detached Coefficients)

Amounts Shipped	x_{12}	x_{13}	...	x_{1n}	x_{21}	x_{23}	...	x_{2n}	...	x_{n1}	x_{n2}	...	$x_{n,n-1}$	Net Produced or Consumed
City 1	1	1	...	1	−1					−1				$a_1^* - b_1^*$
City 2	−1				1	1	...	1			−1			$a_2^* - b_2^*$
		−1				−1								⋮
			⋱				⋱					⋱		⋮
City n				−1				−1		1	1	...	1	$a_n^* - b_n^*$
Cost	c_{12}	c_{13}	...	c_{1n}	c_{21}	c_{23}	...	c_{2n}	...	c_{n1}	c_{n2}	...	$c_{n,n-1}$	z

The general transshipment model is characterized by a cost function and a system of equations in nonnegative variables, each column of which contains at most two non-zero coefficients ($+1$ or -1 or both). The standard transportation model is clearly a special case of this formulation. However, under mildly restrictive assumptions to be discussed in § 16-2, the general transshipment problem will be proved equivalent to the classical transportation problem.

The Network Representation.

The array in Table 16-1-I contains $n(n-1)$ columns corresponding to the number of ways to ship from each city to any other city. If, however, all shipments are routed from one city to another by means of a *chain* of links between neighboring cities, then we need consider only the network composed of such local links. All the variables dealing with shipments to non-neighboring cities can be ignored.

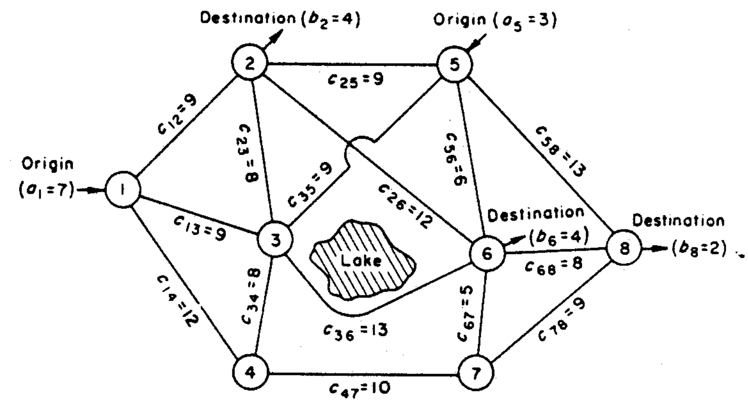


Figure 16-1-I. An example of the transshipment problem.

In the network shown in Fig. 16-1-I, the cost, c_{ij}, of shipping a ton of goods from i to a neighboring point, j, is shown on the relevant link: thus c_{36}, the cost from 3 to 6 is 13. We have not shown c_{63}, the cost from 6 to 3, because in this example each c_{ij} happens to equal c_{ji}. *The theory we will develop, however, is valid even when* $c_{ij} \neq c_{ji}$.

Indeed, although freight rates between two cities are often the same regardless of the direction of shipment, there might be a good economic reason why they should be different. A situation in which c_{ij} is not equal to c_{ji} might actually arise in a pipeline system if i is at the top of a mountain and j is in a valley, for it costs less to pump downhill than up. As a stabilizing influence in certain economic applications, Koopmans [1947-1] and Koopmans and Reiter [1951-1], have suggested that it would be in the public interest to have differing rates to encourage demands in the direction of least use between two cities.

We will show that, with minor modification, the simplex technique developed for solving the classical transportation problem may be used in the transshipment case as well. In Chapter 17, we shall present a computationally convenient procedure, using the network diagram itself, to exhibit, in an elegant way, the underlying geometrical structure.

Reduction to the Classical Case by the Direct Shipment Procedure.

In formulating the transshipment model, we assumed no knowledge of costs except between neighboring cities, but we do presume that the shipping costs between any pair of non-neighboring cities can be obtained by finding the minimum sum-of-costs along chains of local links which connect the two cities through all possible intermediate points (actual freight rates often do not satisfy this additivity assumption). For small problems, it may not be too difficult to determine all the minimum costs merely by inspecting the network. The cheapest ways to ship from origins *1* and *5* to the three destinations, *2*, *6*, and *8*, in the network example of Fig. 16-1-I, are given by the classical transportation array, Table 16-1-II.

TABLE 16-1-II

Origins	Destinations			Available (a_i)
	2	*6*	*8*	
1	x_{12} $c_{12} = 9$	x_{16} $c_{16} = 21$	x_{18} $c_{18} = 29$	7
5	x_{52} $c_{52} = 9$	x_{56} $c_{56} = 6$	x_{58} $c_{58} = 13$	3
Required (b_j)	4	4	2	10

For instance, the cheapest way to ship from *1* to *6* is along the link from *1* to *2* and then to *6*. Hence, we set $c_{16} = c_{12} + c_{26} = 9 + 12 = 21$.

Although one can in this way solve the transshipment problem by the classical transportation technique, our present purpose is to show an alternative approach which has certain advantages:

(a) It avoids the necessity of determining a least-cost route for every origin-destination pair.
(b) It permits treatment of problems in which certain arcs of the network have fixed capacities bounding the flows over these arcs.
(c) It may involve fewer variables, since the number of arcs of a network often is considerably less than the number of origin-destination pairs.

Reduction to the Classical Case by the Transshipment Procedure.

In Table 16-1-III, an array analogous to the classical transportation tableau is shown for the transshipment network shown in Fig. 16-1-I. Applying equations (1), the row equations are obtained by equating the gross supply to the amount consumed plus the amount shipped out; the

TABLE 16-1-III

TRANSSHIPMENT MODEL WITH DIAGONAL SUPPLY VARIABLES

Origin i	Destination j								Row Sum (required) $-b_j^*$
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	
(1)	$-x_{11}^*$ 0	x_{12} c_{12}	x_{13} c_{13}	x_{14} c_{14}					0
(2)	x_{21} c_{21}	$-x_{22}^*$ 0	x_{23} c_{23}		x_{25} c_{25}	x_{26} c_{26}			-4
(3)	x_{31} c_{31}	x_{32} c_{32}	$-x_{33}^*$ 0	x_{34} c_{34}	x_{35} c_{35}	x_{36} c_{36}			0
(4)	x_{41} c_{41}		x_{43} c_{43}	$-x_{44}^*$ 0			x_{47} c_{47}		0
(5)		x_{52} c_{52}	x_{53} c_{53}		$-x_{55}^*$ 0	x_{56} c_{56}		x_{58} c_{58}	0
(6)		x_{62} c_{62}	x_{63} c_{63}		x_{65} c_{65}	$-x_{66}^*$ 0	x_{67} c_{67}	x_{68} c_{68}	-4
(7)				x_{74} c_{74}		x_{76} c_{76}	$-x_{77}^*$ 0	x_{78} c_{78}	0
(8)					x_{85} c_{85}	x_{86} c_{86}	x_{87} c_{87}	$-x_{88}^*$ 0	-2
Col. Sum (available) $-a_j^*$	-7	0	0	0	-3	0	0	0	-10

column equations by equating the gross supply to the amount shipped in plus produced.

Standard Tableau for the Transshipment Model.

In continuing our analysis, it will be convenient to replace the gross-supply variables, x_{jj}^*, by a new set of diagonal variables, x_{jj}, representing the net amount transshipped *through* point j. These are related to the gross supply by

(4) $$x_{jj}^* = x_{jj} + [a_j^* + b_j^* - \text{Min}\,(a_j^*, b_j^*)]$$

To justify the use of the term *transshipment variable* for x_{jj}, we rewrite (4)

$$(5)\quad x_{jj} = [x_{jj}^* - \text{Min}\,(a_j^*, b_j^*)] - [a_j^* - \text{Min}\,(a_j^*, b_j^*)] - [b_j^* - \text{Min}\,(a_j^*, b_j^*)]$$

The subtraction of Min (a_j^*, b_j^*) from each term eliminates the local production for local consumption from the problem. If $a_j^* \geq b_j^*$, the last term drops, and $x_{jj} = x_{jj}^* - a_j^*$ is that part of the gross supply which originated elsewhere and is being transshipped. If $a_j^* \leq b_j^*$, the second term drops and $x_{jj} = x_{jj}^* - b_j^*$ is that part of the gross supply not locally consumed, hence transshipped. From (1) it also follows that $x_{jj}^* - a_j^* \geq 0$, $x_{jj}^* - b_j^* \geq 0$ and therefore $x_{jj} \geq 0$.

If x_{jj}^* is replaced by (4), and if we let a_j, b_j be the net production and consumption as defined by (2), the transshipment problem can be restated in the following standard diagonal form:

Transshipment Problem. Determine nonnegative numbers, x_{ij}, and the minimum z satisfying (6), (7), and (8);

$$(6)\ \text{Column Equations:}\qquad \sum_{i \neq j} x_{ij} - x_{jj} = b_j \qquad (j = 1, 2, \ldots, n)$$

(Total shipped into j minus amount transshipped = Net Consumption)

$$(7)\ \text{Row Equations:}\qquad \sum_{k \neq j} x_{jk} - x_{jj} = a_j \qquad (i = 1, 2, \ldots, n)$$

(Total shipped from j minus amount transshipped = Net Production)

and the

$$(8)\ \text{Cost Form:}\qquad \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} = z \qquad \text{with } c_{jj} = 0 \quad \text{for all } j$$

A general array for this standard form is displayed in Table 16-1-IV.

THEOREM 1: *Any basis for the transshipment problem is triangular.*

PROOF: If the variables, x_{ij} with $i \neq j$, are replaced by $x'_{ij} = -x_{ij}$, then equations (1) and (3) are the same as for a standard transportation problem. Because the proof of triangularity for that problem, as given in § 14-2, did not depend on the signs of the variables, it applies to the transshipment case as well. A similar argument applies to the system (6), (7), and (8) if x_{jj} is replaced by $-x'_{jj}$.

THEOREM 2: *The diagonal variables, x_{jj} or x_{jj}^*, can be made to form part of every basic feasible set.*

PROOF: Consider a new transshipment problem for which each a_i and b_j in (1) is replaced by $a_i + \varepsilon$ and $b_j + \varepsilon$, respectively, where ε is an arbitrary positive number, and x''_{ij} are the new variables. It is clear that, in every feasible solution, $x''_{jj} \geq \varepsilon$, so that the diagonal variables are positive and therefore must form part of every basic set. Any feasible solution of the

TABLE 16-1-IV

STANDARD TABLEAU FOR A TRANSSHIPMENT PROBLEM

Origin i	Destination j (1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	a_i: Avail. Price:
(1)	$-x_{11}$ 0	x_{12} c_{12}	x_{13} c_{13}	x_{14} c_{14}					7 $-\pi_1$
(2)	x_{21} c_{21}	$-x_{22}$ 0	x_{23} c_{23}		x_{25} c_{25}	x_{26} c_{26}			0 $-\pi_2$
(3)	x_{31} c_{31}	x_{32} c_{32}	$-x_{33}$ 0	x_{34} c_{34}	x_{35} c_{35}	x_{36} c_{36}			0 $-\pi_3$
(4)	x_{41} c_{41}		x_{43} c_{43}	$-x_{44}$ 0			x_{47} c_{47}		0 $-\pi_4$
(5)		x_{52} c_{52}	x_{53} c_{53}		$-x_{55}$ 0	x_{56} c_{56}		x_{58} c_{58}	3 $-\pi_5$
(6)		x_{62} c_{62}	x_{63} c_{63}		x_{65} c_{65}	$-x_{66}$ 0	x_{67} c_{67}	x_{68} c_{68}	0 $-\pi_6$
(7)				x_{74} c_{74}		x_{76} c_{76}	$-x_{77}$ 0	x_{78} c_{78}	0 $-\pi_7$
(8)					x_{85} c_{85}	x_{86} c_{86}	x_{87} c_{87}	$-x_{88}$ 0	0 $-\pi_8$
b_j: Req. Price:	0 $+\pi_1$	4 $+\pi_2$	0 $+\pi_3$	0 $+\pi_4$	0 $+\pi_5$	4 $+\pi_6$	0 $+\pi_7$	2 $+\pi_8$	

original problem determines a feasible solution of the new one (and conversely). In fact, if one sets

$$x''_{jj} = x^*_{jj} + \varepsilon, \text{ and } x''_{ij} = x_{ij} \qquad \text{for } i \neq j \tag{9}$$

then optimal solutions must correspond, since the value of z is invariant under the transformation.

From a procedural point of view, it is not desirable to transform the problem explicitly, since we can accomplish the same end simply by allowing the supply variables, x^*_{jj}, an *unrestricted range* of values. They will then be retained in the basic set, once they have entered it, even though their values may be zero. (See Chapter 18.) The same applies to the transshipment variables since they are in one-to-one correspondence, and we have shown that $x_{jj} \geq 0$ is implied by $x_{ij} \geq 0$ for $i \neq j$ (refer to discussion following (4)).

THEOREM 3: *The implicit prices, u_i and v_j, for the transshipment problem can be made to satisfy the relation,*

$$-u_j = v_j \qquad \text{for } j = 1, 2, \ldots, m \tag{10}$$

PROOF: Since $c_{ij} = u_i + v_j$ for all basic variables, x_{ij}, and since, according to Theorem 2, x_{jj} may be assumed to be in every basic feasible set, imposing the condition, $c_{jj} = 0$, will establish the desired relation.

Let us denote the value common to v_j and $-u_j$ by the symbol, π_j. Koopmans and Reiter [1951-1] call π_j the "potential" of point i in the network, in analogy with the electrostatic potential of an electrical network. (In particular, both kinds of "potential" are such that,

$$\begin{aligned} &\text{if } x_{ij} > 0, \text{ then } c_{ij} = \pi_j - \pi_i, \text{ and} \\ &\text{if } c_{ij} > \pi_j - \pi_i, \text{ then } x_{ij} = 0 \end{aligned} \tag{11}$$

at equilibrium. In other words, positive flow from i to j can occur if and only if the voltage drop, c_{ij}, is equal and opposite to the potential difference from i to j.)

16-2. THE EQUIVALENCE OF TRANSPORTATION AND TRANSSHIPMENT PROBLEMS

There is a fundamental difference between transportation and transshipment problems: In the transportation case, each variable is bounded by the smaller of the row and column totals, whereas transshipment allows the values in a 2×2 diagonal submatrix $(x_{ij}, x_{ji}, x_{ii}, x_{jj})$ to be increased by an arbitrary constant, k, since the row and column sums of the resulting subarray remain unchanged, as in

$$\begin{bmatrix} -x_{ii} & x_{ij} \\ x_{ji} & -x_{jj} \end{bmatrix} \rightarrow \begin{bmatrix} -(x_{ii} + k) & (x_{ij} + k) \\ (x_{ji} + k) & -(x_{jj} + k) \end{bmatrix} \tag{1}$$

In the case where all costs are positive, it clearly never pays to transship an amount greater than the total available from all sources. However, if some of the c_{ij} were negative, it might be that no lower bound for z would exist. For example, if $c_{ij} + c_{ji} < 0$, then $z \rightarrow -\infty$ for the class of solutions generated by $k \rightarrow +\infty$ in (1). More generally, it would pay to have such a *circulation* in the flow of the network whenever the sum of the c_{ij} around some loop is negative.[1] For a formal proof of these intuitive statements, see the Chain-Decomposition Theorem, § 19-1, Theorem 2. (Also see Problem 7 in the present chapter.) The latter theorem implies

[1] There are times, it is said, when the exchange rates between various currencies of the world are such that there is a net gain in exchanging between $A, B, C, D, \ldots$, and back to A again. In theory, one could make a fortune by recycling again and again. In practice some people have amassed a considerable profit before the exchange rates changed.

THEOREM 1: *If the sum of c_{ij} around every loop in the network is positive, then in any optimal solution, if one exists, the amount transshipped, x_{jj}, is bounded, and*

(2) $$x_{jj} \leq \sum a_i = \sum b_j = L$$

Defining transshipment *slack*,

(3) $$\bar{x}_{jj} = L - x_{jj}$$

we can reduce the transshipment problem to a standard transportation problem. For our example, $L = \sum a_i = \sum b_j = 10$, so that the transshipment problem given in Table 16-1-IV can be reduced to the one in Table 16-2-I.

TABLE 16-2-I

THE TRANSPORTATION EQUIVALENT OF A TRANSSHIPMENT PROBLEM

Origin i	Destination j (1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	$L + a_i$ / $-\pi_i$
(1)	$\bar{x}_{11}$ 0	x_{12} c_{12}	x_{13} c_{13}	x_{14} c_{14}					17 $-\pi_1$
(2)	x_{21} c_{21}	$\bar{x}_{22}$ 0	x_{23} c_{23}		x_{25} c_{25}	x_{26} c_{26}			10 $-\pi_2$
(3)	x_{31} c_{31}	x_{32} c_{32}	$\bar{x}_{33}$ 0	x_{34} c_{34}	x_{35} c_{35}	x_{36} c_{36}			10 $-\pi_3$
(4)	x_{41} c_{41}		x_{43} c_{43}	$\bar{x}_{44}$ 0			x_{47} c_{47}		10 $-\pi_4$
(5)		x_{52} c_{52}	x_{53} c_{53}		$\bar{x}_{55}$ 0	x_{56} c_{56}		x_{58} c_{58}	13 $-\pi_5$
(6)		x_{62} c_{62}	x_{63} c_{63}		x_{65} c_{65}	$\bar{x}_{66}$ 0	x_{67} c_{67}	x_{68} c_{68}	10 $-\pi_6$
(7)				x_{74} c_{74}		x_{76} c_{76}	$\bar{x}_{77}$ 0	x_{78} c_{78}	10 $-\pi_7$
(8)					x_{85} c_{85}	x_{86} c_{86}	x_{87} c_{87}	$\bar{x}_{88}$ 0	10 $-\pi_8$
$L + b_j$ / π_j	10 π_1	14 π_2	10 π_3	10 π_4	10 π_5	14 π_6	10 π_7	12 π_8	

Conversely, a classical transportation problem can readily be converted to the transshipment format. Let us consider the 3×4 transportation problem, Table 16-2-II, where destinations have been distinguished from origins by assigning them the numerically larger indices.

TABLE 16-2-II

Origins	Destinations				Available
	(4)	(5)	(6)	(7)	
(1)	x_{14} c_{14}	x_{15} c_{15}	x_{16} c_{16}	x_{17} c_{17}	a_1
(2)	x_{24} c_{24}	x_{25} c_{25}	x_{26} c_{26}	x_{27} c_{27}	a_2
(3)	x_{34} c_{34}	x_{35} c_{35}	x_{36} c_{36}	x_{37} c_{37}	a_3
Required	b_4	b_5	b_6	b_7	Totals

The original tableau may be rewritten in standard row-column form,

(4) $$\sum_{j=4}^{7} x_{ij} = a_i \qquad \text{for } i = 1, 2, 3$$

(5) $$\sum_{i=1}^{3} x_{ij} = b_j \qquad \text{for } j = 4, 5, 6, 7$$

and then reinterpreted as in (1) and (6), § 16-1; more conveniently, it may be displayed in the transshipment tableau, Table 16-2-III, where the possibility of shipping over links between origins or between destinations is to be excluded.

TABLE 16-2-III

Origins	Destinations							Available
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	
(1)	$-x_{11}$			x_{14}	x_{15}	x_{16}	x_{17}	a_1
(2)		$-x_{22}$		x_{24}	x_{25}	x_{26}	x_{27}	a_2
(3)			$-x_{33}$	x_{34}	x_{35}	x_{36}	x_{37}	a_3
(4)				$-x_{44}$				0
(5)					$-x_{55}$			0
(6)						$-x_{66}$		0
(7)							$-x_{77}$	0
Required	0	0	0	b_4	b_5	b_6	b_7	Totals

16-3. SOLVING A TRANSSHIPMENT PROBLEM BY THE SIMPLEX METHOD

Finding an Initial Basic Feasible Solution.

In any connected network,[2] such as the one diagrammed in Fig. 16-1-I, it is possible to find a starting solution by inspection.

Begin by selecting a route along the network from any origin to any destination, and specify the largest flow which does not exceed the total production or consumption. Next, either by a direct path or by branching off from a previous path, connect any origin whose gross supply is not yet exhausted to destinations whose gross demand is not yet satisfied, and increase the flow along the path to the maximum possible. (If two paths intersect, as in Fig. 16-3-I, there may sometimes be a *reversal of flow* in the

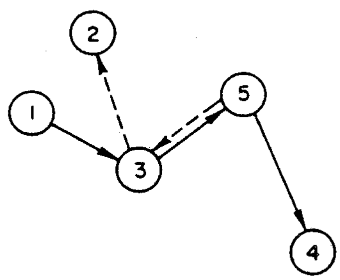


Figure 16-3-I.

common arc.) Repeat this procedure until all the demands have been satisfied.

In forming a basic solution, it is important to do the branching without forming loops and to have each point joined to every other over the constructed paths. For the latter purpose, it may be necessary to include some extra paths having *zero flow*. (The proof that these steps are always possible, and the more precise meaning of such network terms as "loop," will form part of the discussion in the next chapter and hence are omitted here.) For example, one feasible way to perform the required shipping in network Fig. 16-1-I is shown by Fig. 16-3-II below. Another feasible way is given by Fig. 16-3-III, but it would not qualify as a basic solution since it contains the loop corresponding to arcs and nodes of Fig. 16-3-IV.

[2] A network is connected if, given any pair of nodes, i and j, it is possible to find a *chain* of arcs, joining i to j_1, j_1 to j_2, j_2 to j_3, . . ., j_k to j. When the network consists of several separate parts and each part is a connected subnetwork, it is clear that the problem may be decomposed into independent sub-problems that can be solved in the same way.

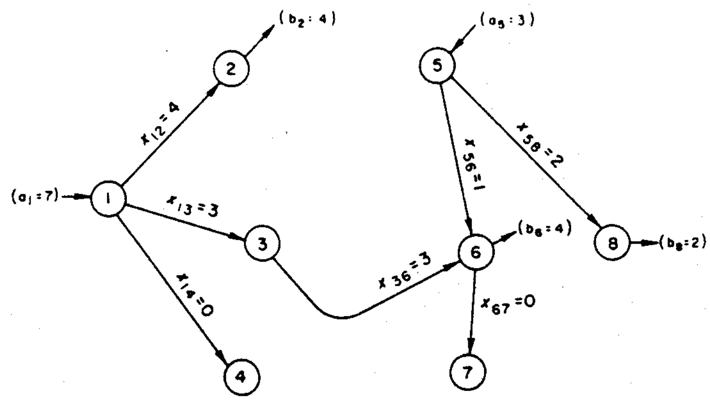


Figure 16-3-II. Graph of a basic feasible solution (see Table 16-3-I, Cycle 0).

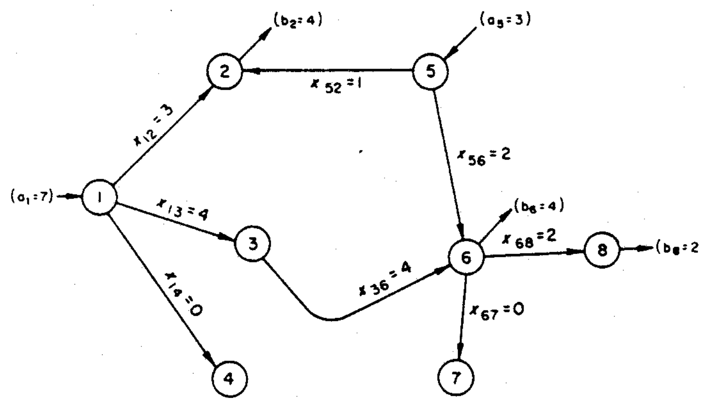


Figure 16-3-III. Graph of a feasible but not basic solution.

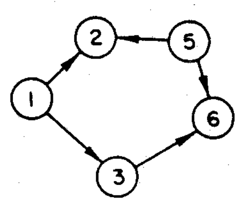


Figure 16-3-IV.

Iterative Procedure.

The basic solution depicted in Fig. 16-3-II is used as a starting solution in Table 16-3-I, cycle 0. The steps of a solution by the simplex method are shown in the tables for subsequent cycles. Since the technique is practically the same as the classical transportation algorithm given in Chapter 14, detailed discussion of the steps will not be given.

TABLE 16-3-I

TRANSSHIPMENT TABLEAU

Cycle 0

Origins	Destinations (1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	a_i / $-\pi_i$
(1)	**0** / 0	**4** / 9	**3 − θ** / 9	**0 + θ** / 12					**7** / 0
(2)	9	**0** / 0	8		9	12			**0** / −9
(3)	9	8	**−3 + θ** / 0	8	9	**3 − θ** / 13			**0** / −9
(4)	12		8	**0 − θ** / 0			**θ*** (enter) / 10		**0** / −12
(5)		9	9		**0** / 0	**1** / 6		**2** / 13	**3** / −16
(6)		12	13		6	**0 + θ** / 0	**0 − θ** (drop) / 5	8	**0** / −22
(7)				10		5	**0** / 0	9	**0** / −27
(8)					13	8	9	**0** / 0	**0** / −29
b_j / π_j	**0** / 0	**4** / 9	**0** / 9	**0** / 12	**0** / 16	**4** / 22	**0** / 27	**2** / 29	

Variable entering basic set is x_{47} because Min $\bar{c}_{ij} = \bar{c}_{47} = c_{47} - (\pi_7 - \pi_4) = -5$.
Variable leaving basic set is x_{67} because Max $\theta = 0$ for $x_{67} = 0 - \theta$.

TABLE 16-3-I (continued)

Cycle 1

Origins	Destinations (1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	a_i / $-\pi_i$
(1)	**0** / 0	**4 + θ** / 9	**3 − θ** / 9	**0** / 12					**7** / 0
(2)	9	**0 − θ** / 0	8		9	**θ*** (enter) / 12			**0** / −9
(3)	9	8	**−3 + θ** / 0	8	9	**3 − θ** (drop) / 13			**0** / −9
(4)	12		8	**0** / 0			**0** / 10		**0** / −12
(5)		9	9		**0** / 0	**1** / 6		**2** / 13	**3** / −16
(6)		12	13		6	**0** / 0	5	8	**0** / −22
(7)				10		5	**0** / 0	9	**0** / −22
(8)					13	8	9	**0** / 0	**0** / −29
b_j / π_j	**0** / 0	**4** / 9	**0** / 9	**0** / 12	**0** / 16	**4** / 22	**0** / 22	**2** / 29	

Variable entering basic set is x_{26} because Min $\bar{c}_{ij} = \bar{c}_{26} = c_{26} - (\pi_6 - \pi_2) = -1$.
Variable leaving basic set is x_{36} because Max $\theta = 3$ for $x_{36} = 3 - \theta$.

TABLE 16-3-I (continued)

Cycle 2 (Optimal)

Origins	Destinations (1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	a_i / $-\pi_i$
(1)	**0** 0	**7** 9	**0** 9	**0** 12					**7** 0
(2)	9	**−3** 0	8		9	**3** 12			**0** −9
(3)	9	8	**0** 0	8	9	13			**0** −9
(4)	12		8	**0** 0			**0** 10		**0** −12
(5)		9	9		**0** 0	**1** 6		**2** 13	**3** −15
(6)		12	13		6	**0** 0	5	8	**0** −21
(7)				10		5	**0** 0	9	**0** −22
(8)					13	8	9	**0** 0	**0** −28
b_j / π_j	**0** 0	**4** 9	**0** 9	**0** 12	**0** 15	**4** 21	**0** 22	**2** 28	

16-4. PROBLEMS

1. Show that no feasible solutions exist for the transshipment model shown in Table 16-1-I unless the total production equals the total consumption.
2. Generalize the equation model to allow for the storing of excesses at a city when the total of amounts shipped-in plus produced may possibly exceed the total of amounts shipped-out plus consumed.
3. Show that, in this generalized model, no feasible solution exists if $\Sigma a_i^* < \Sigma b_j^*$, and interpret the result.
4. Formulate the transshipment model in network form (Table 16-1-I) for the example given by Fig. 16-1-I, omitting all variables x_{ij} such that the network has no arc connecting city i to city j.
5. If cities are allowed to consume and produce simultaneously (so that a_i^* and b_i^* may both be positive), review the proof that the amount transshipped is

$$x_{ii} = x_{ii}^* - a_i - b_i$$

   Show that the standard form for transshipment model (Table 16-1-IV) results from the original when we define new constants and variables as follows:

$$\begin{aligned} a_i &= a_i^* - \text{Min}\,(a_i^*, b_i^*) \\ b_i &= b_i^* - \text{Min}\,(a_i^*, b_i^*) \\ x_{ii} &= x_{ii}^* - \text{Min}\,(a_i^*, b_i^*) \end{aligned}$$

6. Why is $x_{ii} \geq 0$ implied by the standard transshipment form, Table 16-1-IV?
7. In any transshipment problem, prove that if x_{ij} exceeds Σa_i, then there is a circularity in the flow pattern, and show that such a solution cannot be optimal if all c_{ij} are positive.
8. Solve the problem given in Fig. 16-1-I by setting up its transportation equivalent (Table 16-2-I) and applying the methods of Chapters 14 and 15.
9. The post office wishes to send a package from Los Angeles to Boston via the least-cost route. The cost of shipment between neighboring points of the transportation network are proportional to the numbers shown on connecting links of the map shown in Fig. 17-3-I. Find the route.

REFERENCES

Berge, 1958-1
Dennis, 1959-1
Dilworth, 1950-1
Ford and Fulkerson, 1960-1
Gale, 1960-1
Kantorovich and Gavurin, 1949-1
Koopmans, 1947-1
Koopmans and Reiter, 1951-1
Orden, 1956-1

CHAPTER 17

NETWORKS AND THE TRANSSHIPMENT PROBLEM

17-1. GRAPHS AND TREES

T. C. Koopmans, in his pioneering work on transportation problems, was the first to interpret properties of optimal and non-optimal solutions with respect to the linear graph associated with a network of routes [1947-1].

A *linear graph* or *network* consists of a number of *nodes* or junction points, each joined to some or all of the others by *arcs* or links. The diagrammed circles containing the labels 1, 2, . . ., 6 in Fig. 17-1-I are the

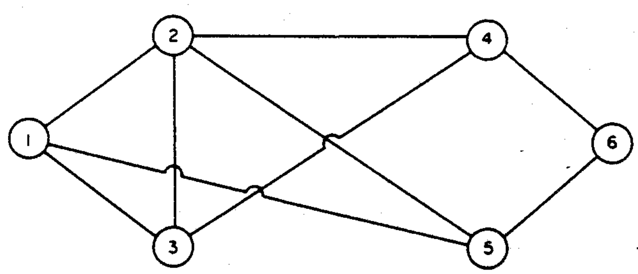


Figure 17-1-I. Example of a linear graph (network).

nodes. The arcs are indicated by straight or curved line segments, each of which links just two nodes, e.g. (1, 2), (1, 3), (1, 5), (2, 3), (2, 4), (2, 5), (3, 4), (4, 6), (5, 6). The crossing of two of these lines does not indicate intersection of the corresponding arcs except at nodes. We shall sometimes use the symbol ⌒ commonly seen in electrical diagrams to indicate a non-nodal crossing.

In transportation problems, the nodes often represent cities and the arcs represent routes between them. The unidirectional nature of flows in goods or traffic over routes leads to consideration of a directed graph (a network made up of directed arcs). The symbol $i \rightarrow j$ is used to denote a directed arc and represents an allowable precedence between i and j. Suppose, for example, in a given situation, we are allowed to proceed, if at 1, to 2 or 3; if at 2, to 3, 4, or 5; if at 4, to 3 or 6; if at 5, to 1 or 6; if at 6, to 5. This can be represented by the directed arcs, $1 \rightarrow 2$, $1 \rightarrow 3$, $2 \rightarrow 3$, $2 \rightarrow 4$, $2 \rightarrow 5$, $4 \rightarrow 3$, $4 \rightarrow 6$, $5 \rightarrow 1$, $5 \rightarrow 6$, and $6 \rightarrow 5$, as shown in Fig. 17-1-II.

There may be several distinct directed arcs joining the same two nodes, i and j; for our present discussion, however, we need consider only two: one associated with a possible shipping activity from i to j, and the other

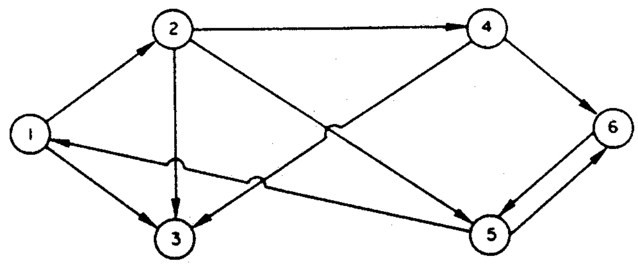


Figure 17-1-II. Example of a directed network.

j to i. We shall treat them as distinct arcs, although they may be diagrammed by a single line.

A sequence of arcs (i, i_1), (i_1, i_2), (i_2, i_3), . . ., (i_k, j), connecting the nodes, i and j, is called a *chain*, regardless of the particular ways in which these arcs may be directed (see Fig. 17-1-IIIa).

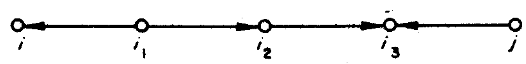
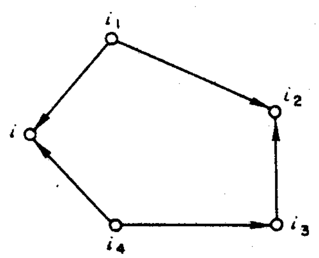


Figure 17-1-IIIa. Example of a chain.

A chain of arcs connecting i to itself is called a loop (a simple loop if the arcs are distinct) (see Fig. 17-1-IIIb).



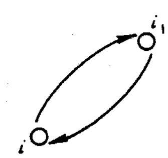


Figure 17-1-IIIb. Examples of chains that are loops.

A graph containing no loops in which every point is connected to every other point through a chain of arcs is called a *tree*. For example, dropping several of the arcs in Fig. 17-1-II, we are left, as in Fig. 17-1-IIIc, with a subnetwork which is a directed tree. Each point, such as 3, 4, or 6, joined to a network by a single link is known as an *end*.

Our eventual objective will be to show that the $(n - 1)$ basic, non-diagonal variables of a transshipment problem correspond to $(n - 1)$ arcs

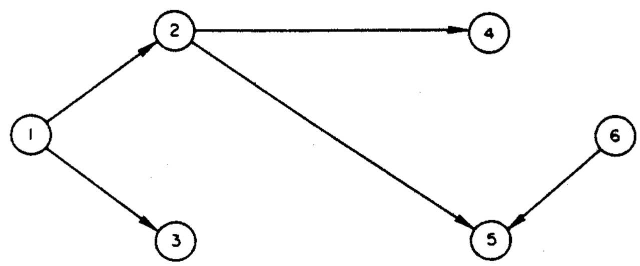


Figure 17-1-IIIc. Example of a tree.

which form a directed tree. The following theorem will be useful for this purpose.

THEOREM 1: *A network having n nodes is a tree if it has $(n - 1)$ arcs and no loops.*

In other words, such a graph is always connected; it cannot break up into *several* trees as in Fig. 17-1-IV.

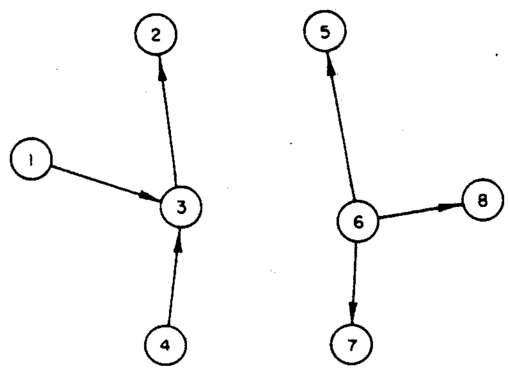


Figure 17-1-IV. Network of disconnected trees.

The theorem being clearly true for two nodes, assume it is true for 2, 3, . . ., $n - 1$.

For the case of n nodes, it will be convenient to establish the following:

LEMMA: *Under the hypotheses of Theorem 1 there exists at least one node which is an end, that is, a point p with only one arc (p, q) connecting it to the rest of the network.*

PROOF: To find a node which is an end, begin by selecting any node, say p_1; it is joined to at least one other node, say p_2, by an arc. (If this were not the case, deleting any arc (i, j) joining a pair of the $n - 1$ remaining nodes would leave $n - 2$ arcs containing no loop. By our inductive assumption, this forms a tree and there exists a chain of arcs joining p_i to p_j. Adjoining arc (p_i, p_j) to this chain would form a loop contrary to assumption.) Because there is an arc from p_1 to some p_2, move to p_2 along the arc (p_1, p_2). Leave p_2 on another arc (if possible) and move to p_3. Because the number of points is finite and there are no loops, by proceeding in this manner, a point p will be found which is an *end* point, with only one arc (p, q) linking it to the rest of the network.

PROOF OF THEOREM 1: If the end and its single arc are deleted, then the remaining network has $(n - 1)$ points, $(n - 2)$ arcs, and since it contains no loops, it is connected by the inductive assumption. If the deleted point, p, and its arc (p, q) are reinserted, it will be possible to connect p to any other point via q, which proves that a network having n points, $(n - 1)$ arcs, and no loops is connected, hence forms a tree.

We have shown that a tree contains at least one end. As an exercise, prove

THEOREM 2: *A tree contains at least two ends.*

The Graph Associated with a Transshipment Problem.

Let us consider the transshipment problem in the form

$$\sum_i x_{ik} - \sum_j x_{kj} = a_k - b_k \qquad (k = 1, 2, \ldots, n) \tag{1}$$

where the first summation is restricted to (i, k) corresponding to admissible arcs (i, k) and the second, to admissible arcs (k, j). The objective is to determine $x_{ij} \geq 0$ and Min z satisfying (1) and

$$\sum_i \sum_j c_{ij} x_{ij} = z \tag{2}$$

We set up our one-to-one correspondence ($\leftrightarrow$) with a network, as follows:

(3) Each equation $k \leftrightarrow$ node k of the graph
Each admissible $x_{ij} \leftrightarrow$ directed arc joining i to j

In the network of Fig. 17-1-V, the availabilities a_1 and a_5 at origins 1 and 5 are shown for convenience on the one-node arrows pointing into these points. By definition, these are not arcs of the network since they do not join pairs of nodes. Similarly, the requirements at destinations 2, 6, and 8 are shown on arrows pointing outward. Such arrows are omitted from points where $a_i = b_i = 0$. The arcs (i, j) of the network which correspond to basic variables, x_{ij}, are shown with heavy lines. The particular values of x_{ij} shown are those of the initial basic solution given in Table 16-3-I for the network problem shown in Fig. 16-1-I.

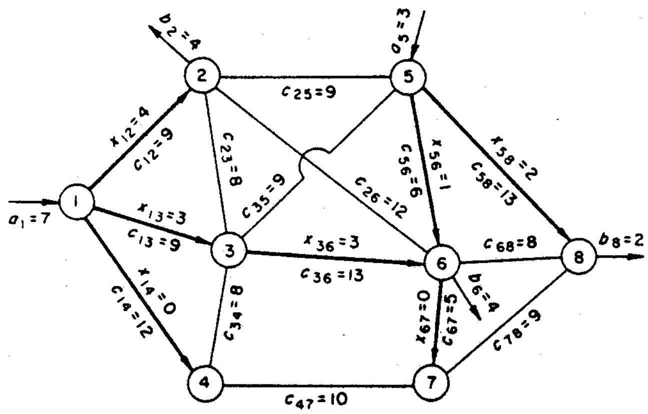


Figure 17-1-V. The tree of heavy arcs corresponds to a basic set of variables.

All solutions to the system of equations for this problem, both feasible and infeasible, can be represented on the corresponding directed arcs of the network. Consequently, at each node k it is necessary that the sum of values of x_{ik} and a_k on arrows pointing into the node equals the sum of values of x_{kj} and b_k on arrows pointing out of it (as in Table 16-1-IV, the common value of these two sums is assigned to the diagonal variable x_{ii}). Conversely, any set of values assigned to directed arcs of a network with this property may be extended to a (feasible or infeasible) solution of the equations, by determining the omitted diagonal variables by the common values of the sums described above.

Relationships Between Bases and Trees.

THEOREM 3: *The subnetwork corresponding to a basic set of variables is a tree.*

If the subnetwork of $(n - 1)$ arcs corresponding to the $(n - 1)$ basic, non-diagonal variables contains no loops, it is a tree by Theorem 1. It

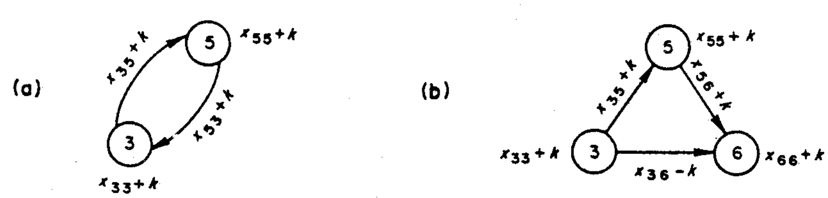


Figure 17-1-VIa, b. Why loops do not correspond to extreme points.

remains to be shown that no basic set can give rise to a loop. Suppose, for example, that x_{35} and x_{53} are both in the basic set as in Fig. 17-1-VIa or x_{35}, x_{56}, and x_{36}, as in Fig. 17-1-VIb. If this were the case, the values of the basic variables around the loop could be altered as indicated in Fig. 17-1-VIa, while the values of all other variables would remain the same, yielding a second solution of the transshipment equations. This, however, contradicts the uniqueness property of the basic solutions; therefore, the basic set cannot include loops.

EXERCISE: Formalize this proof for a general loop.

THEOREM 4: *Any subnetwork of a graph, which is a tree, corresponds to a basic set of variables.*

To prove that a tree corresponds to a basic set of variables, it will be sufficient to show that the $(n - 1)$ variables corresponding to the directed arcs of the tree, which we will call *tree variables*, can be uniquely evaluated for any choice of a_i and b_j, provided that $\Sigma a_i = \Sigma b_j$. To find these values, set all non-tree variables equal to zero. Starting with any end k, and the single point, i, to which it links, we see that there is only one non-zero tree variable in equation k, namely x_{ik} or x_{ki}. Hence, its value is $x_{ik} = (a_i - b_i)$ or $x_{ki} = (b_i - a_i)$.

Thus the triangularity property which makes possible the finding of an equation (say k) with one unknown, evaluating the variable and dropping of the k^{th} equation, corresponds to the tree property which makes possible the finding of an end (say node k) and its arc, evaluating the variable associated with the arc, and dropping the k^{th} node and its arc. What remains is also a tree and the procedure may be repeated, until (on the last step) two nodes and a connecting arc remain, corresponding to a single variable and two equations. Since $\Sigma a_i = \Sigma b_j$, the sum of the constants of (1) vanishes initially *and* after each deletion; this permits a consistent evaluation of the last step. From these observations, it is clear that the tree variables comprise a basic set.

17-2. INTERPRETING THE SIMPLEX METHOD ON THE NETWORK

Phase I, Finding an Initial Basic Feasible Solution.

In this section we shall discuss a simplex procedure for finding a starting solution if one is not readily available by inspection of the graph.

Step 1. Join each origin to other nodes j of the network using only admissible arcs directed away from origins i. Similarly, join various destinations i' to other nodes j' of the network using only admissible directed arcs pointing into final destinations i'. Repeat the process iteratively using nodes j and j' in place of i and i' *being sure at all times not to form loops.* Once a chain of arcs out of an origin joins with a chain of arcs from a destination, nodes along the chain may be joined to nodes not on the chain using arcs in either direction. If there still remain nodes not connected to others, use the arcs in either direction to make the connections. If, finally, there still remain sets of points isolated from other sets of points in the graph, this means that the original problem breaks up into two or more independent problems.

EXERCISE: Show that if the original network is connected, the procedure always yields a tree.

Step 2. Evaluate the basic variables corresponding to arcs of the tree. Since the tree does not necessarily correspond to a feasible set, some variables x_{ij} may be negative. Reverse the direction of the arrow on the arc and replace x_{ij} by x_{ji} of each negative variable, which will be positive in value. Some variables x_{ji} may now correspond to inadmissible arcs.

Step 3. If all directed arcs (i, j) of the tree *are* admissible, then the tree corresponds to a basic feasible solution. If not, then try to drive out the inadmissible arcs (i, j) by the usual Phase I, consisting, in this case, of an auxiliary transshipment problem in which the infeasibility form

$$w = \sum_{i,j} d_{ij} x_{ij} \tag{1}$$

(summed over both admissible and inadmissible arcs) has

$$d_{ij} = \begin{cases} 0 & \text{if } (i, j) \text{ is an admissible arc,} \\ 1 & \text{if } (i, j) \text{ is an inadmissible arc} \end{cases} \tag{2}$$

This is now in the proper format for applying the tree method which will be discussed in the next section.

Step 4. If it turns out that Min $w > 0$, then of course no feasible solution exists. If, on the other hand, feasibility is achieved at any stage with no inadmissible variables x_{ij} remaining, then a basic feasible solution and its corresponding tree have been constructed. Finally, if $w = 0$, but inadmissible x_{ij} still remain, there are two roads open:

(a) Drop all arcs (i, j) whose relative cost factors $\bar{d}_{ij}$ in the infeasibility form are positive, and continue with Phase II on the subnetwork of arcs for which $\bar{d}_{ij} = 0$; or

(b) Reverse the direction of the inadmissible arc (i, j), thus making it admissible, and replace the corresponding basic variable, x_{ij}, by x_{ji}, whose value in the basic solution is zero.

In either case, follow with Phase II, using the original c_{ij} in place of d_{ij}. We have thus shown

THEOREM 1: *If a connected network possesses a feasible solution, then there exists a tree corresponding to a basic feasible solution.*

Consider as an example, the transshipment problem treated in Chapter 16. A starting solution such as the one given in Fig. 17-1-V may be used to start Phase II. We could also construct another such solution using the procedure described in the preceding section. From origins 1 and 5 we can ship directly only to nodes 2, 3, 4; 6, 8. From (2, 3, 4, 6, 8), only one new point can be reached, namely 7. Hence, the tree diagram immediately takes the form of Fig. 17-2-I. The values of the variables on branches of the tree are

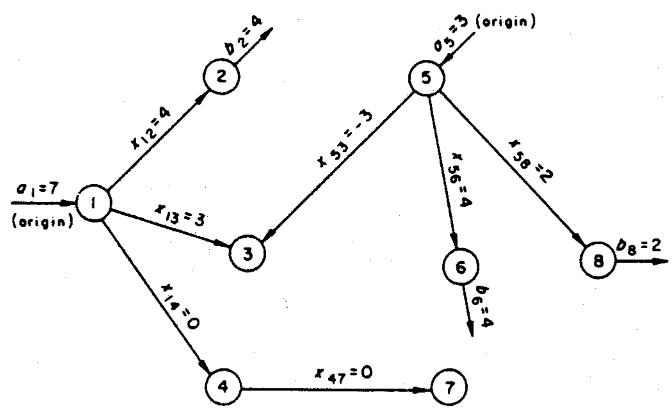


Figure 17-2-I. Starting tree and infeasible basic solution obtained by fanning out from the origins.

such that the algebraic sum at each point is zero (where the sign is determined by the arrows). Thus, the value of $x_{12} = 4$ is determined by the equation associated with end point 2 of the tree. End points 6, 7, 8 determine variables x_{56}, x_{47}, and x_{58}, respectively. This, in turn, permits evaluation of variables associated with arcs leading into ends of the subtree whose x_{ij} have not been evaluated. These are x_{53} and x_{14}. Finally, x_{13} is evaluated. The solution is not feasible, since $x_{53} = -3$. Reversing the arrow and replacing x_{53} by x_{35} produces the required basic feasible solution as shown in Fig. 17-2-II.

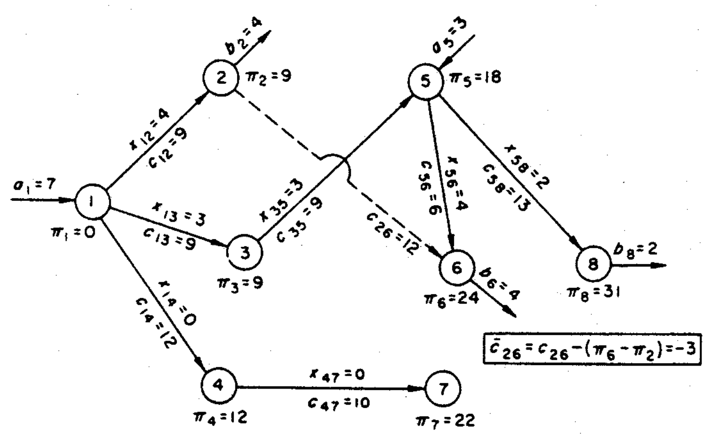


Figure 17-2-II. Graphically improving a basic feasible solution (cycle 0).

Phase II, Finding an Optimal Solution.

Compute the implicit price π_i satisfying the network Fig. 17-1-V with $\pi_j - \pi_i = c_{ij}$, for arcs (i, j) of the tree. For this purpose choose any point i and give it an arbitrary price, π_i. For example, in Fig. 17-2-II choose node 1 which is the focus of a large number of radiating arcs, and set $\pi_1 = 0$. The π_i such that $(1, i)$ or $(i, 1)$ is an arc of the tree can be evaluated next

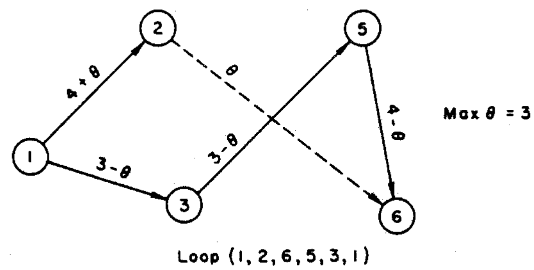


Figure 17-2-III. Adjusting the values of the basic variables around the loop.

from Fig. 17-1-V. In this case, the arcs are (1, 2), (1, 3), and (1, 4) leading to evaluation of π_2, π_3, π_4. From arcs (3, 5) and (4, 7), π_5 and π_7 can be evaluated. Finally, from (5. 6) and (5, 8), π_6 and π_8 can be evaluated.

To determine whether the basic solution shown in Fig. 17-2-II is optimal, compare $\pi_j - \pi_i$ with c_{ij}. This comparison can be made easily on the graph of the network Fig. 17-1-V by systematically scanning each arc (i, j) and forming

$$\bar{c}_{ij} = c_{ij} - (\pi_j - \pi_i) \tag{3}$$

If this is nonnegative for all arcs, then the solution is optimal. In Fig. 17-2-II the criterion is not met since $\bar{c}_{26} = c_{26} - (\pi_6 - \pi_2) = 12 - (24 - 9) = -3$. Hence, it pays to increase the flow along the arc (2, 6), indicated by

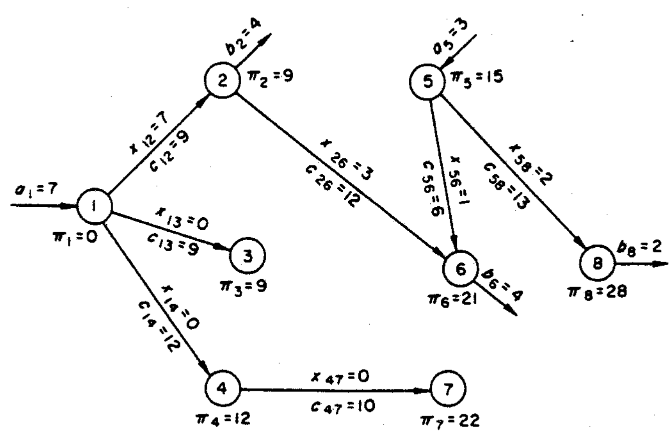


Figure 17-2-IV. Graph of the improved basic feasible solution (cycle 1, optimal).

the dotted arrow in Fig. 17-2-II. With arc (2, 6), the subnetwork is no longer a tree, since it contains the loop shown in Fig. 17-2-III. Only the values of the variables around the loop are affected by increasing the value of x_{26} to $\theta > 0$. It is clear from Fig. 17-2-III that $\theta = 3$ is the largest value that maintains feasibility. At this value, either arc (1, 3) or (3, 5) is dropped from the tree and is replaced by (2, 6). In Fig. 17-2-IV, arc (3, 5) has been dropped. The values of π_i are recomputed, but now all $\bar{c}_{ij} \geq 0$, and the solution is optimal and is the same as that given in Table 16-3-I, cycle 2.

17-3. THE SHORTEST ROUTE PROBLEM

A. An Iterative Solution. Let us suppose that there is a package originating in Los Angeles which can be delivered to Boston along any of several different routes, shown in Fig. 17-3-I. We are interested in having the package transshipped over the shortest route.

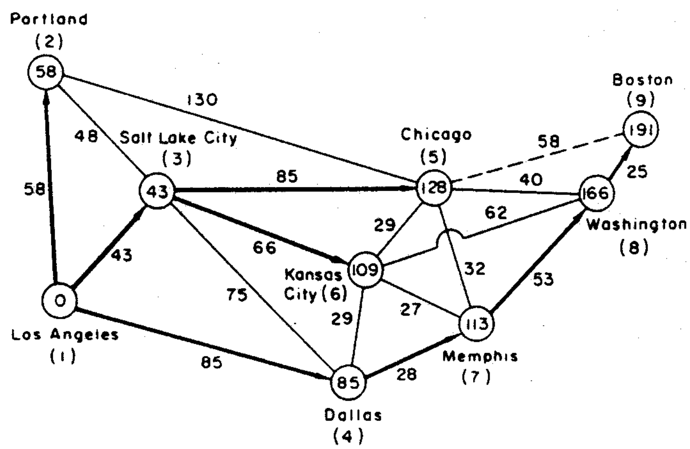


Figure 17-3-I. Starting tree for the initial guess of the shortest route from Los Angeles to Boston.

Now $x_{ij} = 1$ means that the package is shipped from city i to city j; $x_{ij} = 0$ means it is not (where $i \neq j$). Let x_{ii}^* ($i = 2, \ldots, 8$) be the total quantity transshipped through city i. The numbers appearing in the circles are the distances from Los Angeles along heavy arrow routes. This gives rise to a system of constraints

Shipped out

$$
\begin{array}{lr}
(1)\quad x_{12} + x_{13} + x_{14} & = 1 \\
x_{21} - x_{22}^* + x_{23} \qquad + x_{25} & = 0 \\
x_{31} + x_{32} - x_{33}^* + x_{34} + x_{35} + x_{36} & = 0 \\
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\
x_{85} + x_{86} + x_{87} - x_{88}^* + x_{89} & = 0
\end{array}
$$

Shipped in

$$\begin{aligned}(2)\quad & x_{12} - x_{22}^* + x_{32} + x_{52} = 0 \\ & x_{13} + x_{23} - x_{33}^* + x_{43} + x_{53} + x_{63} = 0 \\ & x_{14} + x_{34} - x_{44}^* + x_{64} + x_{74} = 0 \\ & \cdots\cdots\cdots\cdots\cdots\cdots \\ & x_{59} + x_{89} = 1\end{aligned}$$

$$(3)\qquad \sum_{i=1}^{9}\sum_{j=1}^{9} d_{ij}x_{ij} = z \text{ (to be minimized)}$$

where d_{ij}, in this case, is the distance between city i and city j.

The first equation states that the amount shipped *out* of Los Angeles is unity. The last equation states that the amount shipped *into* Boston is unity. Equating the x_{ii} in the i^{th} equation of (1) with the x_{ii} in the $(i-1)^{\text{st}}$ equation of (2) makes the amount shipped out equal to the amount shipped in for each city.

If we replace the condition ($x_{ij} = 0$ or 1) by ($x_{ij} \geq 0$) (see Theorem 1, Chapter 15-1), the problem can of course be solved by the transshipment method, using either a tableau, or a graph, as explained in the last section. However, there is a closely related, but even simpler graphical procedure: starting from Los Angeles, draw some route conjectured to be optimal. For example, ship along the southern route and then up along the east coast to Boston, indicating the route with arrows. In a similar manner, draw conjectured shortest routes and arrows from Los Angeles to all other cities, making sure that the arrows do not form loops. Note that the result must be a tree since the origin is connected to every other node. Note also that the chain from the origin to any node has only directed arcs of the form:

$$(4)\qquad \text{origin}\ \underset{i}{\cdot} \longrightarrow \cdot \longrightarrow \cdot \longrightarrow \cdot \longrightarrow \underset{j}{\cdot}$$

An example of such a tree is shown by the heavy arrows in Fig. 17-3-I. Each such tree corresponds to a basis and it is easy to verify that the corresponding basic solution is feasible. Because of (4), the prices, π_i, are computed in the following manner: set $\pi_1 = 0$ for the origin. At each city, put a number in the node circle that is its distance from the origin via the (unique) path of the tree, e.g. from Los Angeles to Memphis via the tree is 85 plus 28, or 113. The next step is to test whether the tree represents a solution to the shortest route problem. To do this, we are interested in whether the circled numbers are actually the shortest distances from Los Angeles, when we allow other possible paths. If not, the solution can be mproved in the sense that a shorter route can be found from terminal to node. The test of whether these are minimum distances goes like this. Notice that the total distance to Boston is 191; however, if the route went via Chicago, and then to Boston, it would be $128 + 58 = 186$, a decrease of 5 units, so that this particular tree is not an optimal solution. We can better the solution by inserting an arrow between Chicago and Boston, recording 186 at Boston and removing the arrow between Washington and Boston. Again we test whether a shorter route could be obtained via Portland-Chicago, say. However, $58 + 130 \geq 128$, so we try Kansas City-Chicago. But again, $109 + 29 \geq 128$. By continuing in this manner, we eventually arrive at a situation where it is not possible to improve the distance shown in any circle. Accordingly, we have arrived at the optimal solution. For the example at hand, the tree shown is optimal where the Washington-Boston arrow is dropped and the Chicago-Boston arrow inserted, changing the 191 at Boston to 186. The values of x_{ij} are unity along the path in the final tree from Los Angeles to Boston and zero elsewhere. Hence the optimal path is Los Angeles-Salt Lake-Chicago-Boston. The proof of these statements depends on the following observations:

(1) The method of scanning other cities to see if there is a shorter alternative route is the same as the test for optimality, $\pi_j - \pi_i \leq d_{ij}$.

(2) A tree in which all arcs are directed along the chains joining the origin to any node has the properties that (a) the corresponding basic solution is always feasible and (b) only one of the arcs having a given node as end-point points toward the node; the others point outward (Fig. 17-3-II).

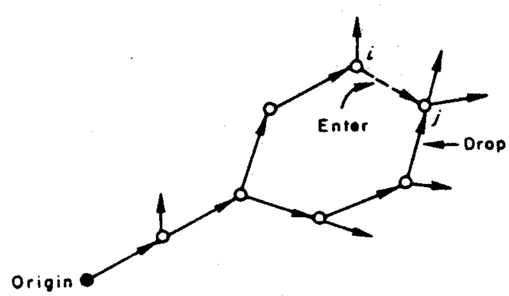


Figure 17-3-II. The special adjustment around a loop for the shortest route problem.

Observe that, if the arc (i, j) is entered into the tree to form an improved new tree, then the arc dropped (assuming non-degeneracy) must be the arc of the old tree pointing *into* j. Each new tree has the same properties.

B. A Direct Solution. Our purpose is to give what is believed to be the shortest procedure for obtaining the shortest route from a given origin to all other nodes in the network or to a particular destination point. The method [Dantzig, 1960-1] can be interpreted as a slight refinement of the method given in A above, those reported by Bellman [1958-1], Moore [1957-1], Dantzig [1957-2], and those proposed informally by Gale and Fulkerson to the author. It is similar to Moore's method of "fanning out" from the origin. However, its special feature is that *the fanning out is done one point at a time and the distance assigned is final.*

It is assumed that (a) one can write without effort for each node the arcs leading to other nodes in increasing order of length and (b) it is no effort to ignore an arc of the list, if it leads to a node whose distance has been assigned earlier. It will be shown that no more than $n(n-1)/2$ comparisons are needed in an n-node network to determine the shortest routes from a given origin to all other nodes.

Suppose that, at some stage k in the computing process, the shortest paths to k of the nodes from some origin are known. Call this set of k points S. (See Fig. 17-3-III.)

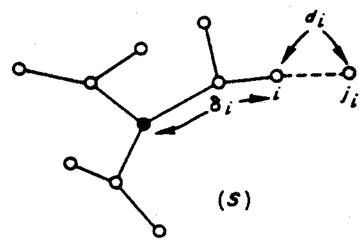


Figure 17-3-III. Finding the shortest route in at most $n(n-1)/2$ comparisons; all arcs selected in the spanning subtree are final.

(a) Let i be a node in S,
(b) let δ_i be its least distance to the origin ●,
(c) let j_i be the closest node to i not in S, if any, and
(d) let d_i be its distance from i.

Choose j_s as the $(k+1)^{st}$ point where

$$\delta_s + d_s = \text{Min } (\delta_i + d_i) \qquad (i = 1, 2, \ldots, k) \tag{5}$$

(In case of ties for minimum, the process could be made more efficient by determining several new nodes j at a time.) This choice implies that the minimum path to j_s from the origin, having a length of $\delta_s + d_s$, is via s. To see this, consider any other path from j_s to the origin. Eventually, the path must reach some node i of S from some node j not in S (where j may be j_s). We now assume that the distances along the path from j_s to j are nonnegative (see Problem 5) so that the total distance to the origin along the path is not less than $\delta_i + d_i$; by (5), however, $\delta_i + d_i \geq \delta_s + d_s$.

Note that the minimum requires only k comparisons for a decision as to the $(k+1)^{st}$ point; hence in an n-node network no more than $1 + 2 + \ldots + (n-1) = n(n-1)/2$ comparisons are needed.

In practice, the number of comparisons can be considerably less than this because, after several stages, one or more of the nodes in S only have arcs leading to points already in S. The 8-node graph shown in Fig. 17-3-IV, for instance, required only 16 comparisons instead of $(8 \times 7)/2 = 28$ comparisons.

If the problem is to determine the shortest path from a given origin to a given terminal, the number of comparisons can often be reduced in practice by fanning out from both the origin and the terminal simultaneously, adding one point at a time to sets about the origin and the terminal, as if they were two separate independent problems.

However, once the shortest path between a node and the origin or the terminal is found in one problem, the path is conceptually replaced by a single arc in the other problem. The algorithm terminates whenever the fan of one of the problems reaches its terminal in the other.

Example: Distances on links of the network are as in Fig. 17-3-IV.

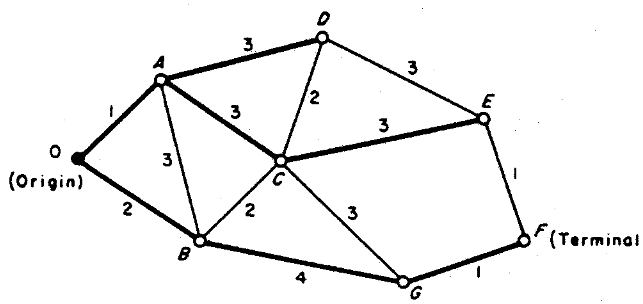


Figure 17-3-IV. An example of a shortest route problem (optimal solution shown by heavy arcs).

For each node, list arcs branching *out* of the node by ascending arc distances:

(O)	(A)	(B)	(C)	(D)	(E)	(F)	(G)
OA-1	AB-3	BC-2	CB-2	DC-2	EF-1	FE-1	GF-1
OB-2	AC-3	BA-3	CD-2	DA-3	EC-3	FG-1	GC-3
	AD-3	BG-4	CA-3	DE-3	ED-3		GB-4
			CG-3				
			CE-3				

Step 0. The set S consists of O initially.

Step 1. Choose arc OA; write its length, 1, above column A, deleting all arcs *into* A. (Delete OA, BA, CA, DA; add A to the set S.) ("Length" means least distance from O.)

Step 2. Compare OB-2 and AB $(1 + 3)$; choose path via OB and write its length, 2, above column B, deleting all arcs *into* B. (Delete OB, AB, CB, GB; add B to the set S.)

Step 3. Compare AC $(1 + 3)$, AD $(1 + 3)$, and BC $(2 + 2)$; and, because of ties, choose paths via AC (or BC) and AD and write their length, 4, above columns C and D, deleting all arcs *into* C and D. (Delete AC, AD, BC, DC, EC, ED, GC; add C and D to the set S.)

Step 4. Compare BG $(2 + 4)$, CG $(4 + 3)$, and DE $(4 + 3)$; choose path via BG, and write its length, 6, above column G, deleting all arcs into G. (Delete BG, CG, FG; add G to the set S.)

Step 5. Compare CE $(4 + 3)$, DE $(4 + 3)$, and GF $(6 + 1)$; choose path via CE (or DE) and GF and write its length, 7, above columns E and F, deleting all arcs into E and F. (Delete CE, DE, FE, GF; add E and F to the set S.)

Because of ties, many of the steps were performed simultaneously.

The shortest paths from the origin to other nodes are along paths OA, OB, AC, AD, BG, CE, GF (see heavy arcs, Fig. 17-3-IV) with alternative BC for AC, DE for CE, and EF for GF.

17-4. PROBLEMS

1. Consider a transshipment problem consisting of k independent networks, each feasible. Connect these k networks together by introducing $k - 1$ new arcs (i_s, j_s), where i_s is a point in the s^{th} network and j_s is a point in the $(s + 1)^{\text{st}}$ network $(s = 1, 2, \ldots, k - 1)$. Prove that $x_{i_s j_s} = 0$ in any feasible solution for the augmented network.
2. Show how the shortest route problem of Fig. 17-3-I can be solved by the transshipment method; show that the tree corresponding to a basis need not have the property that the arrows point away from the origin (Los Angeles).
3. Show that the technique simultaneously works out the shortest routes from Los Angeles to all cities. What are the shortest routes?
4. What are the best routes for distributing a_1 packages from the origin to $(n - 1)$ cities in the quantities $b_2, b_3, \ldots, b_n$, where $\sum_2^n b_1 = a_1$.
5. Show for the shortest route problem of § 17-3 that, if it is permissible for d_{ij} to be negative as well as positive, and if the sum of d_{ij} values around any loop is positive, then the iterative method and the direct solution method are valid, but that both methods fail if this is not true.
6. Solve the shortest route example of § 17-3, used to illustrate the direct method, by the iterative procedure and the method of Chapter 16.
7. *The Caterer Problem* [Jacobs, 1954-1]. A caterer has booked his services for the next T days. He requires r_t fresh napkins on the t^{th} day, $t = 1, 2, \ldots, T$. He sends his soiled napkins to the laundry which has three speeds of service, $f = 1$, 2, or 3 days. The faster the service, the higher the cost, c_f, of laundering a napkin. He can also purchase new napkins at a cost c_0. He has an initial stock of s napkins. The caterer wishes to minimize his total outlay. Formulate as a network problem. Define the caterer at time t as a "source point" in an abstract network for soiled napkins that are connected to "laundry points" $t + 1$, $t + 2$, $t + 3$. The reverse arc is not possible. The laundry point t is connected to a fresh napkin "destination point t" which in turn is connected to the

same type point for $t + 1$. Assign values to the various parameters and solve the numerical problem.

REFERENCES

Bellman, 1958-1
Berge, 1958-1
Dantzig, 1957-2, 1960-1
Dennis, 1959-1
Ford and Fulkerson, 1960-1
Gale, 1960-1
Hadley, 1961-1
Hoffman, 1960-1
Iri, 1960-1
Kelley, 1961-1
König, 1936-1
Koopmans, 1947-1
Koopmans and Reiter, 1951-1
Kruskal, 1956-1
Moore, 1957-1
Orden, 1956-1
Tucker, 1950-1
Vajda, 1961-1

The Caterer Problem

Gaddum, Hoffman, and Sokolowsky, 1954-1
Garvin, 1960-1
Gass, 1958-1
Hadley, 1961-2
Jacobs, 1954-1
Prager, 1956-1

CHAPTER 18

VARIABLES WITH UPPER BOUNDS

18-1. THE GENERAL CASE

Our purpose will be to develop short-cut computational methods for solving an important class of systems involving upper bound restraints on the variables.

In this connection it should be noted that with the growing use of linear programming models for both dynamic and static problems, the main obstacle to full application is the inability of current computational methods to cope with the magnitude of the matrices for even the simplest technological situations. However, in certain cases, such as the now classical Hitchcock-Koopmans transportation model (see Chapter 14), it has been possible to solve this linear inequality system in spite of size because of simple regularities of the system. This suggests that research be undertaken to exploit the properties of other special matrix structures in order to facilitate ready solution of larger systems.

The method described here [Dantzig, 1954-3] was first developed at The RAND Corporation to provide a short-cut computing routine for the following problem: The research personnel were dissatisfied with the long delays generally incurred between the time their request for computation was initiated and the time their work was completed. The main cause of dissatisfaction was quite clear, for there was one project that was both *top-priority* and *so large* in volume that it completely absorbed the entire computing capacity for many weeks. The research people, being human, were no longer interested in the computed answers to their problems when the computing lab finally got around to them. In this example we have a case where the priority method of scheduling was not necessarily the best.[1]

To develop a more flexible decision method than priority scheduling, a model was devised in which the value of a job decreased as its completion date was delayed. The final determination of the optimum schedule depended on solving the distribution problem defined below.

[1] "A Model for Optimum Scheduling of Projects on Punched Card Equipment" was developed by Clifford Shaw of RAND and the author, and reported jointly before the RAND-U.C.L.A. Seminar on Industrial Scheduling in the winter of 1952 (the latter, incidentally, being one of the forerunners of The Institute of Management Sciences).

Type Job	1st week	2nd week	3rd week	. . .	n^{th} week	Total hours required
Job 1	x_{11}	x_{12}	x_{13}	. . .	x_{1n}	$= r_1$
Job 2	x_{21}	x_{22}	x_{23}	. . .	x_{2n}	$= r_2$
.	.	.	.		.	.
.	.	.	.		.	.
.	.	.	.		.	.
Job m	x_{m1}	x_{m2}	x_{m3}	. . .	x_{mn}	$= r_m$
Total hours available	$\leq h_1$	$\leq h_2$	$\leq h_3$	. . .	$\leq h_n$	

The variable x_{ij} is the number of hours to be assigned the i^{th} job in the j^{th} week. Thus, nonnegative x_{ij} and Min z are to be chosen such that the total hours assigned to the i^{th} job equals the hours assigned, (1); the total hours assigned in the j^{th} week must not exceed the availability, (2); and the total cost (3) is minimum.

(1) $$\sum_{j=1}^{n} x_{ij} = r_i$$

(2) $$\sum_{i=1}^{m} x_{ij} \leq h_j$$

(3) $$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}x_{ij} = z \text{ (Min)}$$

where $(-c_{ij})$, the value to the customer of one hour expended on his project in the j^{th} week, satisfies

(3a) $$(-c_{i1}) \geq (-c_{i2}) \geq \cdots \geq (-c_{in})$$

In addition to these restrictions, this problem has the added wrinkle that for some i and j,

(3b) $$x_{ij} \leq \alpha_{ij}$$

or equivalently,

(3c) $$x_{ij} + y_{ij} = \alpha_{ij} \qquad (y_{ij} \geq 0)$$

In other words, the hours assigned the i^{th} project in the j^{th} week cannot exceed α_{ij}. If we were to proceed in the usual manner of adding equations and slack variables for the upper bound restraints, this could greatly enlarge the size of the problem. To illustrate, a problem with a schedule for 18 projects in 10 weeks has 28 equations in 180 unknowns without upper bounds;

with upper bound conditions, it would require an additional 180 equations like (3c), as well as 180 slack variables, y_{ij}.

General Case: Find x_j and Min z satisfying

$$\sum_{j=1}^{n} a_{ij}x_j = b_i \qquad (x_j \geq 0;\ i = 1, 2, \ldots, m) \tag{4}$$
$$\sum_{j=1}^{n} c_j\, x_j = z \text{ (Min)}$$

and, if x_j must also satisfy the inequality, $x_j \leq \alpha_j$, use a slack variable, y_j, and the new equation,

$$x_j + y_j = \alpha_j \qquad (x_j \geq 0;\ y_j \geq 0) \tag{5}$$

to account for each such restriction. We shall refer to (4) and (5) as a *capacitated* system.

The Technique Illustrated.

An idea which permits the solution of a capacitated system with little additional computational effort is based on a slight generalization of the simplex procedure. While the simplex algorithm ordinarily fixes the values of non-basic variables at zero, a little reflection makes it clear that they could be at any fixed value. The simplex criterion indicates that it pays to increase the value of a variable, x_s, if its corresponding relative cost factor, $\bar{c}_s$, is negative, and to decrease x_s, if its corresponding relative cost factor, $\bar{c}_s$, is positive. With the added lower and upper bound restraints on x_s, it will only pay to increase its value, if $\bar{c}_s < 0$ and x_s is not at its lower bound, or to decrease its value if $\bar{c}_s > 0$ and x_s is not at its upper bound.

The following example shows how the enlarged system may be solved by applying the simplex algorithm to the format of the original uncapacitated system, with very simple conventions to insure that the values assigned a variable remain in the range between its upper and lower bounds.

Example 1*:* Find numbers

$$\begin{aligned} &0 \leq x_1 \leq 4, \quad 0 \leq x_2 \leq 5, \quad 0 \leq x_3 \leq 1, \\ &0 \leq x_4 \leq 2, \quad 0 \leq x_5 \leq 3 \end{aligned} \tag{6}$$

and Min z, satisfying

$$\begin{array}{rrrrrrl} x_1 & & +\ x_3 & -\ 2x_4 & & & = 3 \\ & x_2 & -\ x_3 & +\ x_4 & +\ 2x_5 & & = 4 \\ & & -\ 2x_3 & -\ x_4 & +\ x_5 & -\ z & = 0 \\ \bullet & \bullet & & \star & & \bullet & \end{array} \tag{7}$$

Using x_1, x_2, and $(-z)$ as basic variables, the basic feasible solution is

$$\begin{array}{cccccc} [3, & 4, & 0, & 0, & 0, & 0] \\ \bullet & \bullet & & & & \bullet \end{array} \tag{8}$$

The values of the basic variables are dotted. Note that no variable is at its upper bound.

Since $\bar{c}_3$ is negative, it pays to increase the value of x_3. Holding the other non-basic variables fixed, the solution becomes,

$$(9) \qquad [\underset{\bullet}{3 - x_3},\ \underset{\bullet}{4 + x_3},\ \underset{\star}{x_3},\ 0,\ 0,\ \underset{\bullet}{-2x_3}]$$

indicating that x_3 can increase to 3. However, due to the upper bound restraint specified in (6), x_3 cannot be greater than 1. Therefore, we increase x_3 only to its upper bound and hold it fixed at this value, keeping the *same basic set*. The solution becomes

$$(10) \qquad [\underset{\bullet}{2},\ \underset{\bullet}{5},\ \underset{\star}{1},\ 0,\ 0,\ \underset{\bullet}{-2}]$$

Because the basic set is unchanged, the values of $\bar{c}_j$ given in the z-equation of (7) are still applicable. Because $\bar{c}_4 = -1$, we proceed to increase x_4 first, obtaining

$$(11) \qquad [\underset{\bullet}{2 + 2x_4},\ \underset{\bullet}{5 - x_4},\ 1,\ \underset{\star}{x_4},\ 0,\ \underset{\bullet}{-2 - x_4}]$$

Now x_4 cannot increase to 5, since its upper bound is 2, nor can it reach its upper bound value, because we would then have $x_1 = 2 + 2(2) = 6$, violating the upper bound of $x_1 \leq 4$. The largest value permissible to x_4 is thus $x_4 = 1$. Because adopting this value causes the basic variable x_1 to assume its upper bound value, we drop x_1 from the basic set, replacing it by x_4. The canonical form relative to the new basis is obtained by using as pivot, $-2x_4$, the bold faced term in (7), obtaining

$$(12) \qquad \begin{aligned} -\tfrac{1}{2}x_1 \phantom{{}+x_2} - \tfrac{1}{2}x_3 + x_4 \phantom{{}+2x_5 - z} &= -\tfrac{3}{2} \\ \tfrac{1}{2}x_1 + x_2 - \tfrac{1}{2}x_3 \phantom{{}+x_4} + 2x_5 \phantom{{}-z} &= \tfrac{11}{2} \\ -\tfrac{1}{2}x_1 \phantom{{}+x_2} - \tfrac{5}{2}x_3 \phantom{{}+x_4} + x_5 - z &= -\tfrac{3}{2} \end{aligned}$$

associated with the solution,

$$(13) \qquad [4,\ \underset{\bullet}{4},\ 1,\ \underset{\bullet}{1},\ 0,\ \underset{\bullet}{-3}]$$

The variable, x_1, enters the non-basic set at its upper bound value, $x_1 = 4$. It will be noted that the basic variables, x_2 and x_4, are *between* their upper and lower bounds while the non-basic variables, x_1, x_3, and x_5, are *at* their upper or lower bounds. Note that x_5, at its lower bound (zero), has a positive cost factor, while all the variables at their upper bound value have negative cost factors: $\bar{c}_1 = -\frac{1}{2}$ and $\bar{c}_3 = -\frac{5}{2}$. As we shall show in a moment, this satisfies our criterion of optimality for the bounded variable problem, and no further iterations are required.

To prove optimality in this case, substitute for those non-basic variables which are at their upper bounds in (12), the expressions

$$(14) \qquad x_1 = 4 - x_1' \qquad\qquad x_3 = 1 - x_3'$$

thus, deriving the system,

$$(15)\qquad \begin{aligned} \tfrac{1}{2}x_1' \qquad\; + \tfrac{1}{2}x_3' + x_4 \qquad\qquad &= 1 \\ -\tfrac{1}{2}x_1' + x_2 + \tfrac{1}{2}x_3' \qquad + 2x_5 \qquad &= 4 \\ \tfrac{1}{2}x_1' \qquad\; + \tfrac{5}{2}x_3' \qquad + \; x_5 - z &= 3 \end{aligned}$$

whose corresponding basic feasible solution

$$(16)\qquad [0,\ 4,\ 0,\ 1,\ 0,\ -3]$$

is clearly optimal, ignoring the capacity constraints $x_1' \leq 4$, $x_2 \leq 5$, $x_3' \leq 1$, $x_4 \leq 2$, $x_5 \leq 3$. Moreover, substituting $x_1' = 0$, $x_3' = 0$ into (14) yields a set of nonnegative values that satisfies all constraints (14) and (15). Since, in general, the addition of constraints like (14) can only increase or leave unchanged the minimum value that z can attain, the value $z = -3$ is minimum for the full system (14) and (15) or the equivalent system (6) and (7).

Theory of the Upper Bounding Technique.

A basic solution of the uncapacitated system (4) is defined in earlier chapters as a solution obtained by setting $(n - m)$ non-basic variables equal to zero and solving for the consequent values of the basic variables, $x_{j_1}, \ldots, x_{j_m}$. A solution is said to be feasible, if it assigns nonnegative values to all the variables. In each simplex iteration, a new basic feasible solution is derived in which one of the basic variables, x_{j_r} $(1 \leq r \leq m)$, is given a zero value and replaced in the basic set by a non-basic variable x_s.

For this chapter only, we shall use these terms in a slightly altered sense. A *feasible* solution will be one which assigns to each variable, x_j, a nonnegative value not exceeding its upper bound α_j. By a *basic feasible* solution we shall mean one in which the non-basic variables have values either zero or α_j.

Now we note that the original system (4) and (5) may be replaced by an equivalent, *uncapacitated* system in which all non-basic variables are zero. This is done by substituting $\alpha_j - x_j'$ for any non-basic variable, x_j, at its upper bound. After rearrangement, the canonical form relative to some set of basic variables, say $x_1, x_2, \ldots, x_m$, takes the form,

$$(17)\qquad \begin{aligned} x_1 \qquad &+ \bar{a}_{1,m+1}[x_{m+1} \text{ or } (\alpha_{m+1} - x_{m+1}')] + \cdots \\ &\qquad + \bar{a}_{1n}[x_n \text{ or } (\alpha_n - x_n')] = \bar{b}_1 \\ x_2 \qquad &+ \bar{a}_{2,m+1}[x_{m+1} \text{ or } (\alpha_{m+1} - x_{m+1}')] + \cdots \\ &\qquad + \bar{a}_{2n}[x_n \text{ or } (\alpha_n - x_n')] = \bar{b}_2 \\ \cdot \qquad & \cdots\cdots\cdots\cdots\cdots\cdots \\ x_m &+ \bar{a}_{m,m+1}[x_{m+1} \text{ or } (\alpha_{m+1} - x_{m+1}')] + \cdots \\ &\qquad + \bar{a}_{mn}[x_n \text{ or } (\alpha_n - x_n')] = \bar{b}_m \\ &+ \bar{c}_{m+1}\ [x_{m+1} \text{ or } (\alpha_{m+1} - x_{m+1}')] + \cdots \\ &\qquad + \bar{c}_n\ [x_n \text{ or } (\alpha_n - x_n')] = z - \bar{z}_0 \end{aligned}$$

Upon transposing the constants, $(\bar{a}_{ij}\alpha_j)$ and $\bar{c}_j\alpha_j$, to the right member, we get

$$(18)\quad \begin{aligned} x_1 \qquad & + \bar{a}_{1,m+1}[x_{m+1} \text{ or } -x'_{m+1}] + \cdots + \bar{a}_{1s}[x_s \text{ or } -x'_s] \\ & + \cdots + \bar{a}_{1n}[x_n \text{ or } -x'_n] = \bar{b}'_1 \\ & \cdots\cdots\cdots\cdots\cdots\cdots \\ x_m & + \bar{a}_{m,m+1}[x_{m+1} \text{ or } -x'_{m+1}] + \cdots + \bar{a}_{ms}[x_s \text{ or } -x'_s] \\ & + \cdots + \bar{a}_{mn}[x_n \text{ or } -x'_n] = \bar{b}'_m \\ & \bar{c}_{m+1}\ [x_{m+1} \text{ or } -x'_{m+1}] + \cdots + \bar{c}_s\ [x_s \text{ or } -x'_s] \\ & + \cdots + \bar{c}_n\ [x_n \text{ or } -x'_n] = z - \bar{z}'_0 \end{aligned}$$

where

(i) $\bar{a}_{ij}$ are the coefficients of the original canonical form;
(ii) x_{j_i}, with $i = 1, 2, \ldots, m$, is a basic variable ($j_i = i$ above);
(iii) a non-basic variable appears in (17) as either x_j or $(-x'_j)$ depending upon whether its assigned value was zero or α_j in the original solution;
(iv) $\bar{b}'_i \geq 0$ and $(-\bar{z}'_0)$ are, respectively, the values of the basic variables, x_{j_i} and $(-z)$, in the basic feasible solution resulting from this assignment.

Since the basic variables and their coefficients are unaffected by the transformation, the simplex multipliers remain unchanged.

Test for Optimality.

The basic feasible solution for the equivalent problem (18) is optimal if

$$(19)\quad \begin{cases} x_j = 0 \text{ implies } \bar{c}_j = c_j - \sum_{i=1}^{m} \pi_i a_{ij} \geq 0 \\ \qquad\qquad \text{and} \\ x'_j = 0 \text{ implies } \bar{c}_j = c_j - \sum_{i=1}^{m} \pi_i a_{ij} \leq 0 \end{cases}$$

where *the primed variables must satisfy the reverse inequality* because their coefficients are of the opposite sign from those of the unprimed variables.

THEOREM 1: *A basic feasible solution of* (18) *which satisfies* (19) *is equivalent to an optimal solution of* (4) *and* (5).

The proof is immediate.

Improving a Basic Feasible Solution.

If the solution is not optimal, then the variable to be selected for increase is the non-basic variable, x_s (or x'_s), such that

$$(20)\quad \bar{c}'_s = \text{Min } \bar{c}'_j < 0 \qquad \text{where } \bar{c}'_j = \begin{cases} \bar{c}_j \text{ for non-basic } x_j \\ -\bar{c}_j \text{ for non-basic } x'_j \end{cases}$$

Let θ represent the change (plus or minus) in the value of x_s.

Case I: The non-basic candidate is unprimed. The new value of x_s is θ, where $0 \leq \theta \leq \alpha_s$.

Case II: Here x'_s is the non-basic variable to be increased; the new value of x_s is $(\alpha_s + \theta)$, where $-\alpha_s \leq \theta \leq 0$ (since, in this case, the value of x_s must be decreased from its upper bound value, α_s).

In either case, if we fix the values of all other non-basic variables, then the adjusted values of the basic variables x_{j_i}, z, and x_s in terms of the change, θ, are

$$\begin{aligned} x_{j_1} &= \bar{b}'_1 - \bar{a}_{1s}\theta \qquad (0 \leq x_{j_i} \leq \alpha_{j_i}) \\ x_{j_2} &= \bar{b}'_2 - \bar{a}_{2s}\theta \\ &\cdots\cdots\cdots\cdots \\ x_{j_m} &= \bar{b}'_m - \bar{a}_{ms}\theta \\ z &= \bar{z}'_0 + \bar{c}_s\theta \end{aligned} \tag{21}$$

where

Case I: $\quad x_s = \theta$ and $\bar{c}_s < 0 \quad (0 \leq \theta \leq \alpha_s)$

Case II: $\quad x_s = \alpha_s + \theta$ and $\bar{c}_s > 0 \quad (-\alpha_s \leq \theta \leq 0)$

Case I: The greatest nonnegative *increase*, θ^*, that maintains feasibility is given by

$$\theta^* = \text{Min}\begin{cases} \alpha_s \\ \bar{b}'_i/\bar{a}_{is} & (\bar{a}_{is} > 0) \\ (\bar{b}'_i - \alpha_{j_i})/\bar{a}_{is} & (\bar{a}_{is} < 0) \end{cases} \tag{22}$$

Case II: The maximum nonnegative *decrease*, θ^*, that maintains feasibility is

$$-\theta^* = \text{Min}\begin{cases} \alpha_s \\ \bar{b}'_i/(-\bar{a}_{is}) & (-\bar{a}_{is} > 0) \\ (\bar{b}'_i - \alpha_{j_i})/(-\bar{a}_{is}) & (-\bar{a}_{is} < 0) \end{cases} \tag{23}$$

(a) If $\theta^* = \alpha_s$ in (22), then non-basic variable x_s appears as x'_s in the next iteration, or if $-\theta^* = \alpha_s$ in (23), then the non-basic variable x'_s appears as x_s in the next iteration. In either event, the basic set remains unchanged, but the variables acquire new values as determined by setting $\theta = \theta^*$ in (21).

(b) If, on the other hand, α_s is not the minimum in (22) or in (23), then x_s *replaces* some x_{j_r} as a basic variable. The new value of x_s is θ^* in Case I with x_{j_r} becoming zero or α_s, according as $\bar{a}_{rs} > 0$ or $\bar{a}_{rs} < 0$. In Case II, the new value of x_s is $(\alpha_s + \theta^*)$, with x_{j_r} becoming non-basic at zero or at α_s according as $-\bar{a}_{is} > 0$ or $-\bar{a}_{is} < 0$. The new values for the other basic variables $j_i \neq j_r$ are found by setting $\theta = \theta^*$ in (21).

Example 2: Find numbers

(24) $$0 \leq x_1 \leq 1, \quad 0 \leq x_2 \leq 2, \quad 0 \leq x_3 \leq 3, \quad 0 \leq x_4 \leq 4,$$

and Min z satisfying

(25)
$$\begin{aligned} x_1 + x_2 + x_3 + x_4 \qquad\qquad &= 6 \\ 2x_1 + x_2 - 4x_3 - 5x_4 + (-z) &= 0 \end{aligned}$$

Starting with the feasible solution $x_1 = 1$, $x_2 = 2$, $x_3 = 3$, $x_4 = 0$, and $(-z) = 8$, using x_1 as a basic variable, the initial canonical form is

(26) Cycle 0

$$\begin{cases} x_1 + x_2 + x_3 + x_4 \qquad\qquad = 6 \\ \quad\;\; - x_2 - 6x_3 - 7x_4 + (-z) = -12 \end{cases}$$

$$\begin{array}{ccccc} [1 - \theta_0, & 2, & 3, & 0 + \theta_0, & 8 + 7\theta_0] \\ \bullet & & & \star & \end{array}$$

Max $\theta_0 = 1$; x_4 replaces x_1 as a basic variable.

(27) Cycle 1

$$\begin{cases} x_1 + x_2 + x_3 + x_4 \qquad\qquad = 6 \\ 7x_1 + 6x_2 + x_3 \qquad + (-z) = 30 \end{cases}$$

$$\begin{array}{ccccc} [0, & 2 - \theta_1, & 3, & 1 + \theta_1, & 15 + 6\theta_1] \\ & \star & & \bullet & \end{array}$$

Max $\theta_1 = 2$; x_4 is still basic and x_2, non-basic, shifts from its upper to its lower bound.

(28) Cycle 2

$$\begin{cases} x_1 + x_2 + x_3 + x_4 \qquad\qquad = 6 \\ 7x_1 + 6x_2 + x_3 \qquad + (-z) = 30 \end{cases}$$

$$\begin{array}{ccccc} [0, & 0, & 3 - \theta_2, & 3 + \theta_2, & 27 + \theta_2] \\ & & \star & \bullet & \end{array}$$

Max $\theta_2 = 1$; x_3 replaces x_4 as a basic variable, which drops out at its upper bound.

(29) Cycle 3 (Optimal)

$$\begin{cases} x_1 + x_2 + x_3 + x_4 \qquad\qquad = 6 \\ 6x_1 + 5x_2 \qquad - x_4 + (-z) = 24 \end{cases}$$

$$\begin{array}{ccccc} [0, & 0, & 2, & 4, & 28] \\ & & \bullet & & \end{array}$$

The Revised Simplex Tableau.

Only minor changes in the simplex method using multipliers are necessary to account for upper bounds (see Chapter 9). These are discussed under headings (1), (2), (3) below.

(1) In Table 9-3-Ia, add an extra row and record in it the upper bound values of the variables.

(2) In Table 18-1-I, it is advisable not only to record the relative cost

factors for d_j or $\bar{c}_j$, but also to indicate which non-basic variables are at their upper bounds. In the table below, the following notation is used:

(a) A bar above a numerical cost factor is used to indicate that the corresponding variable is at its upper bound.

(b) Brackets [ ] are used to indicate that the corresponding variable is basic.

(c) An asterisk * indicates that corresponding x_j is to enter the basic set in the next iteration.

Referring to Example 1 (see (6), (7)), we have

TABLE 18-1-I

	Relative Cost Factors $\bar{c}_j$ or d_j				
$j =$	(1)	(2)	(3)	(4)	(5)
Cycle 0	[0]	[0]	-2^*	-1	1
1	[0]	[0]	$-\bar{2}$	-1^*	1
2	$-\overline{\tfrac{1}{2}}$	[0]	$-\overline{\tfrac{5}{2}}$	[0]	1

(3) There is no change in the layout of Table 9-3-IIa of § 9-3. The choice of pivot element $\bar{a}_{rs}$, however, is in accordance with (22) or (23) and sequel. If x_s replaces x_{j_r}, Table 9-3-IIb is replaced by Table 18-1-II where θ^* or $(\alpha_s + \theta^*)$ is the new value of x_s accordingly as $x_s = 0$ or $x_s = \alpha_s$ in this cycle. If there is no change in the basic set, then all β_{ij}, π_i, and σ_i remain unchanged; the new values for the basic variables, z and w are as shown in the next to last column of Table 18-1-II, except that row r becomes $x_{j_r} = (\bar{b}'_r - \bar{a}_{rs}\theta^*)$.

TABLE 18-1-II

Tableau at Start of Cycle $\ell + 1$ (if x_s Replaces x_{j_r} as Basic Variable)

Basic Variables	Columns of the Canonical Form					Values of Basic Variables	x_s
	x_{n+1}	. . .	x_{n+m}	$-z$	$-w$		
	←Inverse of basis→						
x_{j_1}	$\beta_{11} - \bar{a}_{1s}\beta^*_{r1}$	. . .	$\beta_{1m} - \bar{a}_{1s}\beta^*_{rm}$			$\bar{b}'_1 - \bar{a}_{1s}\theta^*$	
.	.		.			.	
x_s	β^*_{r1}	. . .	β^*_{rm}			θ^* or $\alpha_s + \theta^*$	
.	.		.			.	
x_{j_m}	$\beta_{m1} - \bar{a}_{ms}\beta^*_{r1}$	. . .	$\beta_{mm} - \bar{a}_{ms}\beta^*_{rm}$			$\bar{b}'_m - \bar{a}_{ms}\theta^*$	
$-z$	$-\pi_1 - \bar{c}_s\beta^*_{r1}$	. . .	$-\pi_m - \bar{c}_s\beta^*_{rm}$	1		$-\bar{z}'_0 - \bar{c}_s\theta^*$	
$-w$	$-\sigma_1 - d_s\beta^*_{r1}$	. . .	$-\sigma_m - d_s\beta^*_{rm}$		1	$-\bar{w}'_0 - d_s\theta^*$	

$\beta^*_{ri} = \beta_{ri}/\bar{a}_{rs}$ $(i = 1, 2, \ldots, m)$

θ^* determined by formula (22) or (23).

x_s column blank at start of cycle.

18-2. THE BOUNDED VARIABLE TRANSPORTATION PROBLEM AND GENERALIZATIONS

Solving a Bounded Variable Transportation Problem.

The theory of upper bounding will be applied to a capacitated transportation problem. In general form, the problem is that of finding nonnegative x_{ij} and minimum z satisfying

(1) The Row Equations: $(i = 1, 2, \ldots, m)$ $\qquad \sum_{j=1}^{n} x_{ij} = a_i$

(2) The Column Equations: $(j = 1, 2, \ldots, n)$ $\qquad \sum_{i=1}^{m} x_{ij} = b_j$

(3) The Upper Bounds: (for all i and j) $\qquad x_{ij} \leq \alpha_{ij}$, and

(4) The Objective Function: $\qquad \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} = z$

Referring to (5) as the *standard*, 3×4 *transportation array for bounded variables*, we observe that Theorem 1 below is a direct translation of the optimality criteria given in the preceding section.

(5)

x_{11} α_{11} c_{11}	x_{12} α_{12} c_{12}	x_{13} α_{13} c_{13}	x_{14} α_{14} c_{14}	a_1 u_1
x_{21} α_{21} c_{21}	x_{22} α_{22} c_{22}	x_{23} α_{23} c_{23}	x_{24} α_{24} c_{24}	a_2 u_2
x_{31} α_{31} c_{31}	x_{32} α_{32} c_{32}	x_{33} α_{33} c_{33}	x_{34} α_{34} c_{34}	a_3 u_3
b_1 v_1	b_2 v_2	b_3 v_3	b_4 v_4	Implicit ↑ ← Prices

THEOREM 1: *A feasible solution, $x_{ij} = x_{ij}^o$, for the capacitated transportation problem is optimal, if there is a set of implicit prices u_i and v_j and relative cost factors $\bar{c}_{ij} = c_{ij} - u_i - v_j$, such that*

(6)
$$\begin{aligned} 0 < x_{ij}^o < \alpha_{ij} &\Rightarrow \bar{c}_{ij} = 0 \\ x_{ij}^o = 0 \quad &\Rightarrow \bar{c}_{ij} \geq 0 \\ x_{ij}^o = \alpha_{ij} \quad &\Rightarrow \bar{c}_{ij} \leq 0 \end{aligned}$$

We will show how the methods of § 18-1 may be adapted efficiently to this class by solving a simple numerical example given in (7) below. Later in this section, we will show that the class of capacitated transportation problems is actually equivalent to the class of ordinary transportation problems, and also consider various generalizations of the capacity concept.

(7)

$x_{11} \leq 12$ $c_{11} = 10$	$x_{12} \leq 13$ 5	$x_{13} \leq 5$ 6	$x_{14} \leq 20$ 7	25
$x_{21} \leq 14$ 8	$x_{22} \leq 20$ 2	$x_{23} \leq 10$ 7	$x_{24} \leq 9$ 6	25
$x_{31} \leq 18$ 9	$x_{32} \leq 4$ 3	$x_{33} \leq 25$ 4	$x_{34} \leq 7$ 8	50
15	20	30	35	

Finding an Initial Basic Feasible Solution.

While simple rules have been devised for finding an initial solution in an uncapacitated transportation problem, it does not appear possible to construct such a rule in the capacitated case. If one were able to do this, one would thereby also have found a simple solution to the problem: find an assignment of m men to m jobs where certain men are excluded from certain jobs. Formulated in mathematical terms, the problem is: given an $m \times m$ incidence matrix (elements 0 or 1), pick out a permutation of ones or show none exists. So far; no one has been able to give a *non-iterative procedure* for solving this problem.

In attempting to find an initial solution for (7), it is generally useful to begin by selecting a box with the minimum c_{ij}, which in (7) is c_{22} with a value of 2, and to assign as high a value as possible to the corresponding variable without forcing any variable to exceed its upper bound. Here we set $x_{22} = 20$. If the size of this entry is finally limited by a row or column equation, consider it a basic variable and make no more entries in that row or column. If, on the other hand, the value of the variable is limited by its upper bound restriction, then consider the variable non-basic at its upper bound and place a bar above the entry. In case of a tie between the two types of limitations, always consider the row or column as limiting and the variable as basic. Repeat the procedure with the remaining boxes.

Applied to (7), this routine yields in order, the assignments: $x_{22} = 20$ (basic), $x_{33} = 25$ (bounded), $x_{13} = 5$ (basic), $x_{24} = 5$ (basic), $x_{14} = 20$ (basic), $x_{34} = 7$ (bounded), $x_{31} = 15$ (basic). Since the third row and fourth column still have 3 units unassigned, the solution is not feasible. Extra "short" boxes are added to the array: an $i = 0$ row and $j = 0$ column, and $d_{ij} = 0$

replaces the original c_{ij}, and $d_{ij} = 1$ in the shortage boxes. This is summarized in (8).

(8)

				Short: $3 - \theta_0$ 1	0
	12 $d_{11} = 0$	13 0	**5** 5 0	**20** 20 0	**25** -1
	14 0	$\mathbf{20 - \theta_0}$ 20 0	10 0	$\mathbf{5 + \theta_0}$ 9 0	**25** -1
Short: $3 - \theta_0$ 1	**15** 18 0	θ_0^* 4 0	$\overline{25}$ 25 0	$\overline{7}$ 7 0	**50** 1
0	**15** -1	**20** 1	**30** 1	**35** 1	u_i v_j

Note that $d_{30} = d_{04} = 1$ must equal u_3 and v_4 respectively, since we have shown in § 15-2-(14), that slack rows and columns can be regarded as having prices u_0 and v_0 equal to zero.

Proceeding now with Phase I, minimizing the sum of the artificial variables, in particular, $x_{04} + x_{30}$, we find that a single iteration furnishes a feasible solution as given by (9). The original cost factors, c_{ij}, are now restored.

(9)

12 10	13 5	**5** 5 6	**20** 20 7	**25** 0
14 8	$\mathbf{17 - \theta}$ 20 2	10 7	$\mathbf{8 + \theta}$ 9 6	**25** -1
15 18 9	$\mathbf{3 + \theta}$ 4 3	$\overline{25}$ 25 4	$\overline{7} - \theta^*$ 7 8	**50** 0
15 9	**20** 3	**30** 6	**35** 7	u_i v_j

However, this solution is not optimal, because x_{34} is a non-basic variable at its upper bound, whose relative cost factor should be nonpositive, while in reality, $\bar{c}_{34} = c_{34} - u_3 - v_4 = 8 - 0 - 7 = +1$. Thus, it pays to decrease x_{34} from its upper bound value, keeping the other non-basic variables fixed and adjusting the basic variables. The greatest decrease, θ, that maintains feasibility is $\theta = 1$, and at this value it is stopped by the upper bounding restriction, $x_{24} = 8 + \theta = 9$.

The new array, given in (10), is optimal.

(10)

12 10	13 5	**5** 5 6	**20** 20 7	**25** −1
14 8	**16** 20 2	10 7	**$\bar{9}$** 9 6	**25** −1
15 18 9	**4** 4 3	**$\overline{25}$** 25 4	**6** 7 8	**50** 0
15 9	**20** 3	**30** 7	**35** 8	u_i v_j

The foregoing method implies the following theorem whose proof for the general bounded transportation problem is left as an exercise.

THEOREM 2: *If the upper bounds, the quantities available, and the quantities required are all integers, every basic solution will be integral in a bounded transportation problem.*

On the Equivalence of a Bounded Transportation Problem and the Classical Transportation Problem.

It will be noted that each variable x_{ij} appears in *three* equations with non-zero coefficients; not only in (1) and (2), the row and column equations used in the classical problem, but in addition the upper bounding inequality (3), which may be rewritten

$$x_{ij} + y_{ij} = \alpha_{ij} \qquad (y_{ij} \geq 0) \tag{11}$$

where variable, y_{ij}, represents slack. The system can, however, be replaced by an obviously equivalent one in which each variable enters only two equations just as in the classical transportation form. Consider the problem of finding $x_{ij} \geq 0$ and Min z satisfying

(12)

Row: $(i = 1, \ldots, m)$ $\qquad \sum_{j=1}^{n} x_{ij} = a_i$

Column: (all i, j) $\qquad -x_{ij} - y_{ij} = -\alpha_{ij}$

Row: (all i, j) $\qquad y_{ij} + x'_{ij} = \alpha_{ij}$

Column: $\qquad\qquad\qquad\qquad -\sum_{i=1}^{m} x'_{ij} = -b_j$
$(j = 1, \ldots, n)$

$$\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \qquad\qquad = z\ (\text{Min})$$

An illuminating interpretation of this result is in terms of networks. The conventional graph for a capacitated transportation problem may be represented as in Fig. 18-2-I. The numbers α_{ij} on the directed arc joining

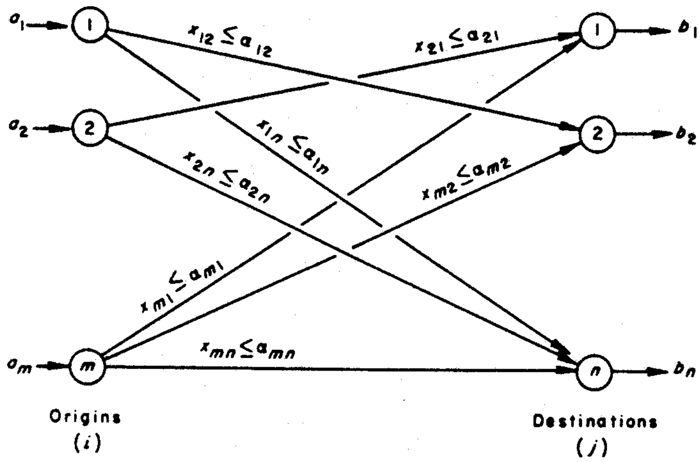
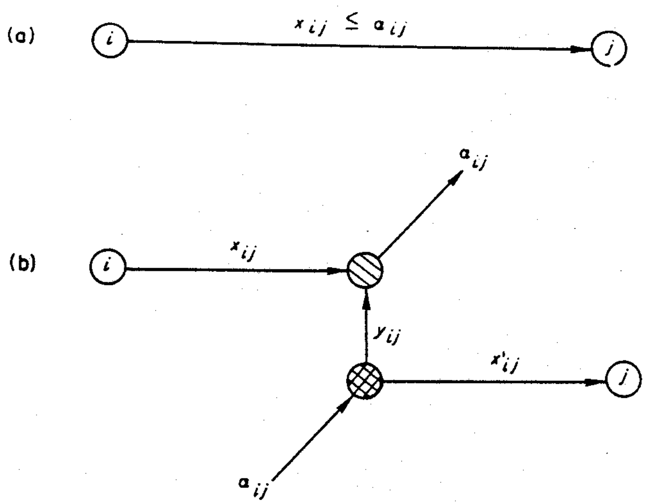


Figure 18-2-I. Bi-partite graph of a capacitated transportation problem.



Figures 18-2-IIa, b. How to replace a capacitated arc by unrestricted arcs.

origin i and destination j are the *arc capacities*. The device used in (12) replaces each *capacitated* arc of Fig. 18-2-IIa by the *set* of *unrestricted* arcs of Fig. 18-2-IIb.

Transportation Problems with Bounded Partial Sums of Variables.

An idea formalized by A. S. Manne deals with bounding, not only variables, but also partial sums of variables. For simplicity, let us consider a case with only one such partial sum. In the scheduling of jobs on computing machinery discussed earlier, the condition $x_{11} \leq 40$ might be interpreted to mean that no more than one man can be assigned to job 1 in week 1. In some problems a more involved condition might be desired, such as $x_{11} + x_{31} + x_{61} \leq 40$, expressing the circumstance that jobs 1, 3, and 6 can be assigned only to a certain individual. Similarly, a condition like $x_{11} + x_{12} + x_{13} \geq k$ might mean that at least k hours must be worked on job 1 during the first three weeks. Just as with the variables themselves, a transportation problem with a bounded partial sum of variables in either a row or a column can be reduced to a standard transportation problem. To see this, consider the system (1), (2) and the added condition

$$x_{11} + x_{12} + \cdots + x_{1k} \leq \alpha \tag{13}$$

This may be written in row-column format as in Table 18-2-I.

TABLE 18-2-I

x_{10}	x_{11}	$\cdots$	x_{1k}				α
y_{10}				$x_{1,k+1}$	$\cdots$	x_{1n}	a_1
	x_{21}	$\cdots$	x_{2k}	$x_{2,k+1}$	$\cdots$	x_{2n}	a_2
	$\vdots$		$\vdots$	$\vdots$		$\vdots$	$\vdots$
	x_{m1}	$\cdots$	x_{mk}	$x_{m,k+1}$	$\cdots$	x_{mn}	a_m
α	b_1	$\cdots$	b_k	b_{k+1}	$\cdots$	b_n	Totals

(The inadmissible boxes are shaded.) It is clear that any number of conditions like (13) can be added to the system by similarly treating each in turn. For example, the added condition on column 2

$$x_{12} + x_{32} + x_{62} \leq \beta \tag{14}$$

may be taken care of by splitting column 2 and using a second slack variable. Moreover, there may be other conditions on column 2, such as

$$x_{22} + x_{42} + x_{52} \leq \gamma \tag{15}$$

that do not involve the same variables. Also, there can be more than one condition on the same variables in the same column, for example, condition (14) and

$$x_{12} + x_{62} \leq \delta \tag{16}$$

could be taken care of by further splitting the column associated with the variables x_{12}, x_{32}, x_{62}.

THEOREM 3: *A transportation problem with added partial sum conditions in rows and columns can be reduced to a standard transportation problem, if any two conditions in a column (or row) either have no variables in common, or the variables of one of the conditions are a subset of the variables of the other condition.*

THEOREM 4: *If a bounded partial sum of variables includes variables in different columns or rows, the basis need not be triangular, so that non-integral basic solutions can be obtained.*

EXERCISE: Prove these last two theorems.

18-3. PROBLEMS

The General Case. (Refer to § 18-1.)

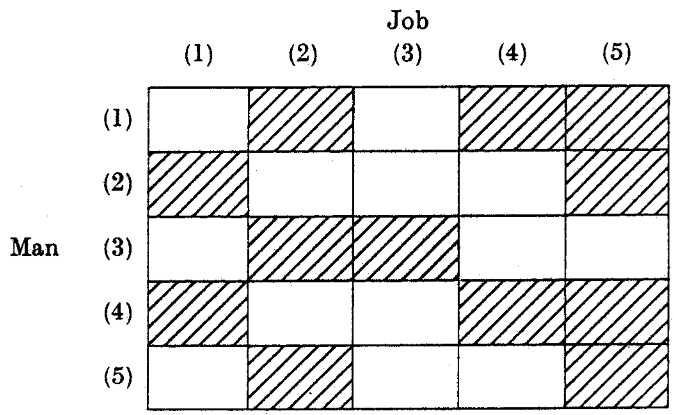
1. (a) Review the rules for determining the candidate for entering the basic set or shifting to upper or lower bound and the variable leaving the basic set.
   (b) Modify the procedure to improve a general feasible solution.
   (c) Modify the procedure to cover a problem where variables have lower bounds other than zero.
   (d) Does the lexicographic scheme for getting around degeneracy still apply to the bounded variable method; if not, what modifications are necessary?

The Bounded Transportation Problem and Generalization. (Refer to § 18-2.)

2. Give a direct proof of Theorem 1 of § 18-2.
3. Show for capacitated transportation problem of § 18-2-(1), (2), (3), and (4) that no feasible solution exists if there is a row p, such that $\sum_{j=1}^{n} \alpha_{pj} < a_p$ or a column q such that $\sum_{i=1}^{m} \alpha_{iq} < b_q$.
4. Construct an example to show that a feasible solution satisfying (1), (2), and (3) need not exist even if, for *all* $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n$,

$$\sum_{j=1}^{n} \alpha_{ij} \geq a_i, \ \sum_{i=1}^{m} \alpha_{ij} \geq b_j$$

5. Consider the example shown in § 18-2-(7); restate in the form § 18-2-(1), (2), (3), and (4).
6. In the 5 × 5 array below, the exclusion of men from certain jobs is indicated by shaded boxes. Show why this is a bounded variable problem. Use the methods of § 18-2 to find a feasible solution.



7. Construct an example where artificial variables x_{0j}, x_{i0} are required as part of every basis during Phase II. Amplify the discussion of the text to cover situations in which artificial variables form a part of the basis during Phase II.
8. Prove that the solution shown in § 18-2-(10) is optimal. Is it unique? If not, construct all other optimal solutions.
9. Given a capacitated transportation problem with $m = 5$ rows and $n = 7$ columns and the additional partial row sum condition $x_{32} + x_{35} + x_{37} \leq \alpha$, find an equivalent capacitated transportation problem with no side conditions.
10. Construct examples to show that if the sets of variables used in the partial sums are not *nested* or mutually exclusive in a row or column, then the basis need not be triangular.
11. If a bounded partial sum of variables includes variables in different rows or columns, show that the basis need not be triangular; in other words, it is not equivalent to a transportation problem.

REFERENCES

Berge, 1958-1
Cahn, 1948-1
Charnes and Cooper, 1955-1, 1959-1
Dantzig, 1949-1, 1954-3, 1956-2, 1957-3
Ford and Fulkerson, 1957-1, 1960-1
Hu and Prager, 1958-1
Prager, 1957-1
Vajda, 1958-1, 1961-1
Wagner, 1958-1, 1959-1

CHAPTER 19

MAXIMAL FLOWS IN NETWORKS

19-1. FORD-FULKERSON THEORY

Consider a network connecting two nodes, a *source* and a *destination*, by way of several intermediate nodes. Each arc of the network is assigned two numbers, representing the flow *capacity* along the arc in each direction. Assuming a steady state condition, *find a maximal flow from the source to the destination*. In this section we shall follow the theory developed by Ford and Fulkerson [1954-1; 1960-1].

In network Fig. 19-1-I, the source and destination are distinguished from the other nodes by double circles. The flow capacities, α_{ij}, in each direction

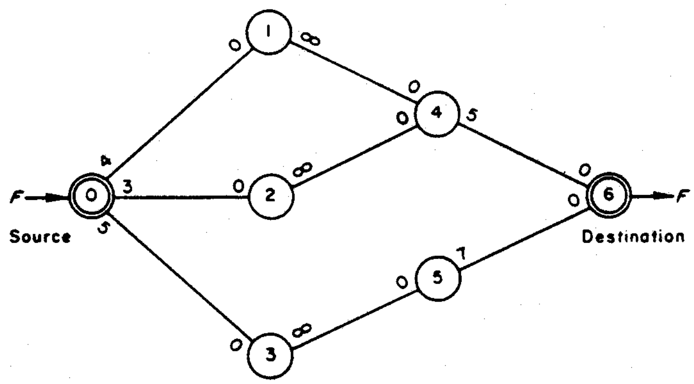


Figure 19-1-I. A maximal flow problem with directed arc capacities.

are shown by numbers along each arc near the node at which the flow might originate.

If $x_{ij} \geq 0$ denotes the quantity of flow from i to j, then the following *constraints on capacity* hold:

(1) $$0 \leq x_{ij} \leq \alpha_{ij}$$

(2) $$0 \leq x_{ji} \leq \alpha_{ji}$$

Thus in Fig. 19-1-I, for example, $0 \leq x_{01} \leq 4$, $0 \leq x_{10} \leq 0$; $0 \leq x_{14} \leq \infty$, $0 \leq x_{41} \leq 0$, etc. In addition, the following *conservation-of-flow* equations hold: except for the source where $k = 0$, and the destination where $k = n$, the sum of flows into node k balances the sum of flows out of it, so that

$$\sum_i x_{ik} - \sum_j x_{kj} = 0 \qquad (k = 1, 2, \ldots, n-1) \tag{3}$$

where all terms x_{ik} and x_{kj} are omitted from these sums, except those corresponding to arcs of the network. We denote by F the flow into the source from outside the network; then, by definition,

$$F + \sum_i x_{i0} - \sum_j x_{0j} = 0 \tag{4}$$

It is not difficult to show, in view of (3), that the flow out of the system at the destination also equals F for, if we sum the $n - 1$ relations appearing in (3) and (4), each variable, x_{ij}, appears in two equations with opposite signs (hence cancels), except for those representing flows into the destination. Reversing signs, one obtains

$$\sum_i x_{in} - \sum_j x_{nj} - F = 0 \tag{5}$$

The Maximal Flow Problem is to choose $x_{ij} \geq 0$ and Max F satisfying (1), (2), (3), (4), and (5).

Finding Feasible Solutions to Transportation and Transshipment Problems.

It is interesting to note that the problem of finding a feasible solution to an assignment problem or, more generally, to a transportation problem in which not all x_{ij} are admissible, is equivalent (as we shall see in a moment) to solving a maximal flow problem. Recall that the primal-dual algorithm (§ 11-4) seeks feasible solutions to a sequence of restricted primal problems. This implies that transportation problems could be solved by means of a sequence of maximal flow problems. This idea is developed into an efficient algorithm in the next chapter. The following transportation problem (actually not solvable) can be reduced to a network flow problem like Fig. 19-1-I.

(6)

Origins	Destinations (4)	Destinations (5)	Row Total
(1)	x_{14}		4
(2)	x_{24}		3
(3)		x_{35}	5
Column Total	5	7	

It is clear that a feasible solution to (6), if it exists, corresponds to finding $x_{ij} \geq 0$ in the array (7) that sum to unknown row and column totals x_{0i} and x_{j0} such that the sum, z, of these marginal totals is maximum and *equal* to the sum of column or row capacities.

(7)

Origins	Destinations (4)	Destinations (5)	Row Total	
(1)	x_{14}		x_{01}	$(x_{01} \leq 4)$
(2)	x_{24}		x_{02}	$(x_{02} \leq 3)$
(3)		x_{35}	x_{03}	$(x_{03} \leq 5)$
Column Total	x_{46}	x_{56}		

$$(x_{46} \leq 5)\ (x_{56} \leq 7)$$

$$x_{01} + x_{02} + x_{03} = x_{46} + x_{56} = z \text{ (Max)}$$

On the other hand, if the maximum value of z is less than the specified capacities, as is the case here, no feasible solution to (6) exists. It is now obvious that Fig. 19-1-I is the network representation of (7).

In the case of a transshipment problem, a modified network is formed by joining all nodes with surplus available to a fictitious common source node by arcs with capacities equal to the surpluses available, and by joining all nodes with unsatisfied needs to a common fictitious destination by arcs with capacities equal to the deficit. A feasible solution to the transshipment problem will then correspond to a maximal flow solution to the modified network problem, which equals the sum of the capacities of the arcs from the source node (or into the destination node).

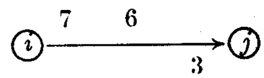
Properties of Network Flow Problems.

The following theorem is easily seen:

THEOREM 1: *A set of $x_{ij} \geq 0$ satisfying the capacity constraints and the conservation equations can be replaced by another set x'_{ij} with the same total flow F in which either x'_{ij} or x'_{ji} is zero by setting*

(8) $$x'_{ij} = x_{ij} - \text{Min}\,(x_{ij}, x_{ji})$$

We shall only consider flows where this is always the case. For example, if on a *directed arc* joining i to j a number 6 appears, this will mean $x_{ij} = 6$ and $x_{ji} = 0$.



$(x_{ij} \leq 7,\ x_{ji} \leq 3)$

It is now possible to replace all variables x_{ij} and x_{ji} by their difference

$$\bar{x}_{ij} = x_{ij} - x_{ji} \tag{9}$$

in which case $\begin{cases} \bar{x}_{ij} > 0 \text{ corresponds to } \bar{x}_{ij} = x'_{ij} \text{ and } x'_{ji} = 0 \\ \bar{x}_{ij} < 0 \text{ corresponds to } -\bar{x}_{ij} = x'_{ji} \text{ and } x'_{ij} = 0 \end{cases}$

The capacity constraints and conservation equations become

$$-\alpha_{ji} \leq \bar{x}_{ij} \leq \alpha_{ij} \tag{10}$$

$$\sum_i \bar{x}_{ik} = 0 \qquad (k = 1, 2, \ldots, n-1)$$

$$F + \sum_i \bar{x}_{i0} = 0$$

$$-F + \sum_i \bar{x}_{in} = 0$$

THEOREM 2: *A set of $\bar{x}_{ij}$ and $F > 0$ satisfying the capacity constraints and the conservation equations can be decomposed into a sum of positive chain flows from the source to the destination and a set of circular flows such that the direction of positive flows in any common arc is the same for all chains.*

DEFINITION: A *chain flow, K*, is a constant flow value $\bar{x}_{ij} = K$ for every arc $(i \to j)$ along a chain and $\bar{x}_{ij} = 0$ elsewhere. This theorem is an interesting one because it means that a solution to a flow problem or a transshipment problem corresponds to our intuitive notion that items start from nodes of surplus and move (flow) from one node to the next without losing their identity until arriving finally at some node of deficit.

The proof is straightforward. Assume $F > 0$. Choose a chain starting at 0 with initial arc $(0 \to i_1)$, where

$$\bar{x}_{0i_1} = \operatorname*{Max}_i \bar{x}_{0i} > 0 \tag{11}$$

That $\bar{x}_{0i_1} > 0$ follows from $F > 0$ and the conservation relation $F = \Sigma \bar{x}_{0i}$. We now repeat our procedure at node i, choosing the second arc $(i_1 \to i_2)$ of the chain by

$$\bar{x}_{i_1, i_2} = \operatorname*{Max}_i \bar{x}_{i_1, i} > 0 \tag{12}$$

Again by the conservation equation at i_1 and the fact that $\bar{x}_{0i_1} > 0$, it follows that $\bar{x}_{i_1, i_2} > 0$.

Upon iteration, we either (a) generate a chain that returns to a node arrived at earlier, thus forming a "loop," or (b) we complete a chain to the destination. If a loop is generated, subtract a constant $K > 0$ from each $\bar{x}_{ij}$ for arcs (i, j) of the loop where, letting $\in$ denote "belongs to,"

$$K = \text{Min } \bar{x}_{ij} \qquad [(i, j) \in \text{loop}] \tag{13}$$

It is clear that the new values of $\bar{x}_{ij}$ satisfy the capacity and conservation relations. Starting again at the node where the chain first formed a loop, the chain generation procedure is continued. Only a finite number of loops can be removed from the solution by the above procedure, since each new solution generated by a loop removal has at least one more $\bar{x}_{ij}$ that is zero.

Hence, after a finite number of steps, a chain from origin to destination can be constructed with positive flow along it. A value K is then assigned to the chain by setting

$$K = \text{Min } \bar{x}_{ij} > 0 \qquad \text{for all } (i, j) \in \text{chain} \tag{14}$$

A new solution to the flow problem is now constructed with flow value $F - K$, by subtracting K from each $\bar{x}_{ij}$ value corresponding to arcs (i, j) along the chain. The entire procedure can now be repeated with the new problem if $F - K > 0$. Again we note that there can only be a finite number of chain removals because each new solution has at least one more $\bar{x}_{ij}$ that is zero.

Finally, if $F = 0$ and some $\bar{x}_{ij} > 0$, starting with node i and arc $(i \to j)$, the above procedure can be followed to construct a loop which can be removed. In a finite number of steps all residual loops can be removed. This completes the constructive proof of the theorem.

THEOREM 3: *If there exists no chain of arcs, each with positive capacity, joining the source to the destination, then the maximal flow is zero.*

PROOF: Assume, on the contrary, that it is possible to have the maximal flow positive. By the previous theorem it is then possible to decompose it into *chains* of positive flows. Along any such chain with flow value $K > 0$, we must have $0 < K \leq \bar{x}_{ij} \leq \alpha_{ij}$, because the method of decomposition is such that each $\bar{x}_{ij} > 0$ is represented as a sum of nonnegative chain flows along the directed arc joining i to j. It follows that the selected chain has arcs of positive capacity.

We can argue, conversely, that if there exists a chain with arcs of positive capacity, we may choose $K = \text{Min } \alpha_{ij} > 0$ for arcs along the chain and thereby obtain a flow $F = K > 0$ along the chain; hence

THEOREM 4: *The maximal flow is positive if there exists a chain of arcs, each with positive capacity, joining the source to the destination.*

The following theorem permits us constructively to obtain a maximal flow in a network by seeking in associated networks a chain of arcs each with positive capacity joining the source to the destination.

THEOREM 5: *A solution $F = F_0$, $\bar{x}_{ij} = \bar{x}^0_{ij}$ is maximal if and only if the* *maximal flow F' is zero in a second network formed by replacing α_{ij} by $\alpha'_{ij} = \alpha_{ij} - \bar{x}^o_{ij}$.*

PROOF: Suppose, on the contrary, that there exists a solution, $F' = F'_0 > 0$, $-\alpha'_{ji} \leq \bar{y}^o_{ij} \leq \alpha'_{ij}$ to the associated problem. Then

$$(15) \qquad \begin{aligned} -(\alpha_{ji} + \bar{x}^o_{ij}) \leq \bar{y}_{ij} \leq \alpha_{ij} - \bar{x}^o_{ij}, \\ -\alpha_{ji} \leq \bar{y}_{ij} + \bar{x}^o_{ij} \leq \alpha_{ij} \end{aligned}$$

It follows that $\bar{x}_{ij} = \bar{y}^o_{ij} + \bar{x}^o_{ij}$ is an admissible solution to the original network with flow $F = (F'_0 + F_0) > F_0$, contradicting the hypothesis of

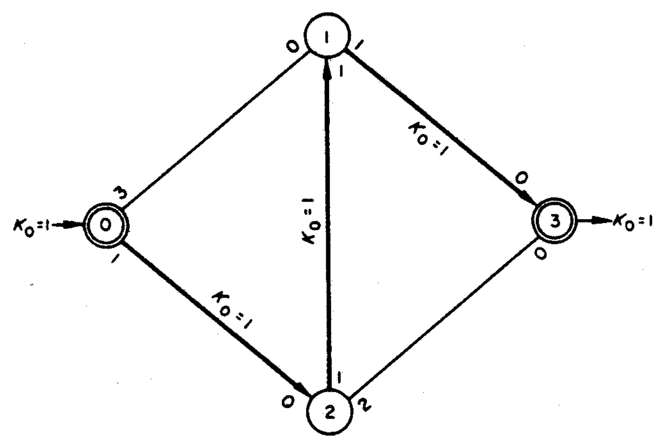


Figure 19-1-IIa. A maximal flow example with an initiating chain flow (cycle 0).

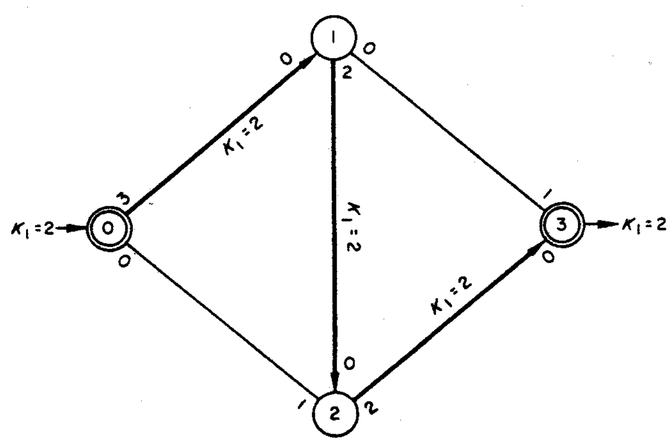


Figure 19-1-IIb. Adjusted arc capacities and an augmenting chain flow (cycle 1).

maximal flow. Thus, if $F = F_0$ is not maximal, an improved solution to the original system can be constructed.

EXERCISE: Show the necessity as well as the sufficiency of the hypothesis of Theorem 5.

A simple example, Fig. 19-1-IIa, illustrates this. To initiate the computation, seek a chain joining 0 to 3 with arcs of positive capacity. (Later we shall describe how to do this systematically.) One such is the chain $(0 \to 2)$, $(2 \to 1)$, $(1 \to 3)$ with capacities $(1, 1, 1)$; along it, initiate the flow $K_0 = 1$.

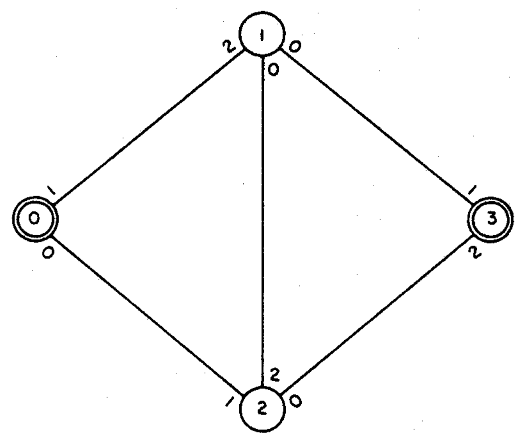


Figure 19-1-IIc. **Final adjusted arc capacities, no additional chain flow possible (cycle 2).**

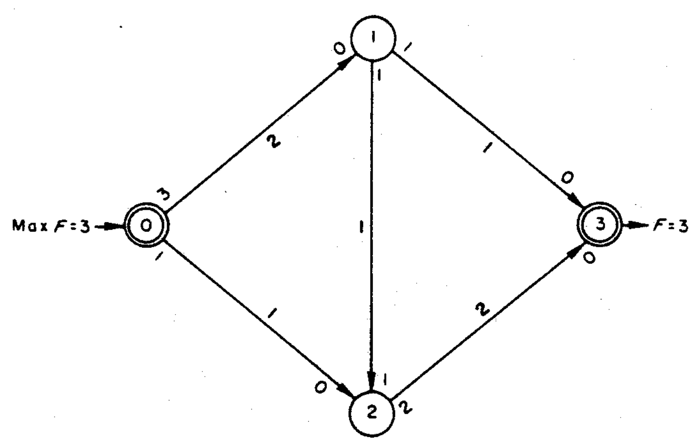


Figure 19-1-IId. **This maximal flow is the algebraic sum of the previous chain flows.**

This is represented by the numbers at the middle of the arcs, the arrows indicating the direction of flow along the chain. Adjust the capacity of each arc by subtracting $K_0 = 1$ from the capacity at the *base* of the arrow and adding it to the capacity at the *point* of the arrow. This results in Fig. 19-1-IIb.

In the new network there is only one chain of arcs with positive capacity; namely $(0 \to 1)$, $(1 \to 2)$, $(2 \to 3)$, with capacities (3, 2, 2). Hence a flow of $K_1 = 2$ can be set up along this chain. Again adjusting the capacities of the network, we have Fig. 19-1-IIc.

No chain of positive capacities joining 0 to 3 exists. We now form our maximal flow as ***the algebraic sum*** of the chain flows given in Fig. 19-1-IIa, b as shown in Fig. 19-1-IId.

Constructing a Chain of Positive Arc Capacities Joining Source to Destination.

This can be done systematically by forming a tree of all the nodes that can be reached from the source by such chains. Thus, all nodes that can be reached from the source by arcs of positive capacity are determined first. In Fig. 19-1-III, these are nodes 1 and 5; arcs $(0 \to 1)$ and $(0 \to 5)$ form part of the tree. The procedure is repeated with each new node in turn, omitting nodes reached earlier. It is easy to show that, if a chain of positive

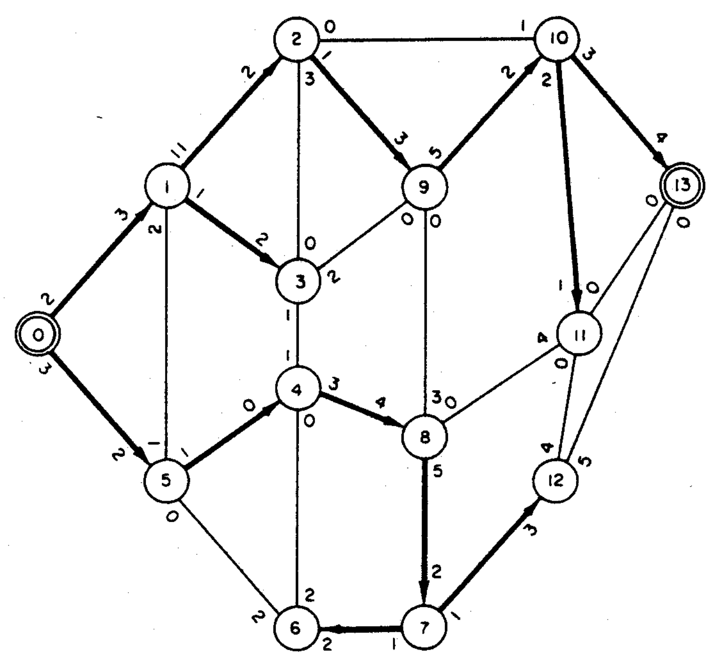


Figure 19-1-III. Fanning-out procedure for finding a positive chain flow.

arc capacities exists joining the source to any node, this procedure will construct at least one such chain.

EXERCISE: Show that the above procedure will always construct a chain of positive capacities from origin to destination, if one exists.

THEOREM 6: *If a maximal flow exists and the capacities are integers, the procedure will construct only a finite number of positive chain-flows, whose algebraic sum is the maximal flow.*

PROOF: Note that if α_{ij} are integers, then the chain flow, K, is also an integer, and the same is true of the successive adjusted arc-capacities, α'_{ij}. But each flow being positive implies that F must increase by at least unity on each iteration. Hence, only a finite number of iterations is possible since F is finite.

Properties of Cuts in Networks.

When the flow $F = F_0$ is maximal for $x_{ij} = x^o_{ij}$, it will be observed that certain of the directed arcs of the network are used to full capacity, or *saturated*, i.e., $x_{ii} = \alpha_{ii}$. It is easy to see that, if for this set all the saturated arcs are removed from the network, or more precisely, if their α_{ij} values are set equal to zero, no flow is possible. Indeed, if a positive flow over some chain of unsaturated arcs existed, the same chain would have *positive arc capacity* for the adjusted network, $\alpha'_{ij} = \alpha_{ij} - x^o_{ij}$, and by Theorem 5, $x_{ij} = x^o_{ij}$ would not be a maximal flow solution.

DEFINITION: A *cut* is any set of directed arcs containing at least one arc from every chain of positive capacity joining the source to the destination.

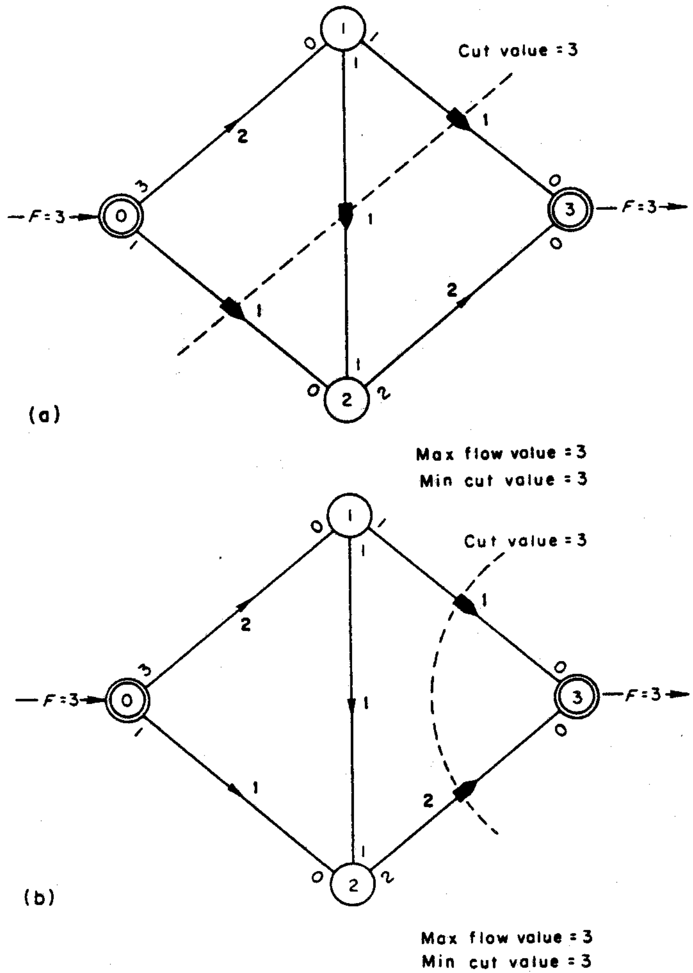
DEFINITION: The *cut value* is the sum of the capacities of the arcs of the cut.

From our remarks it is clear that the collection of saturated arcs in a maximal solution constitutes a cut. Thus, in Fig. 19-1-IId, the set of directed arcs, $(1 \rightarrow 2)$, $(1 \rightarrow 3)$, $(0 \rightarrow 2)$, $(2 \rightarrow 3)$, with capacities, (1, 1, 1, 2), constitutes a cut. The cut value, in this case, is $1 + 1 + 1 + 2 = 5$. It will be noted that this cut has two subsets which are also cuts. These are marked in Figs. 19-1-IVa and IVb with the bullet symbol indicating the direction of the arc belonging to the cut.

The cut value in both of these cases is less than before, namely, $1 + 1 + 1 = 3$ and $1 + 2 = 3$. Notice, however, that this is the same value as the maximum flow value. It is also the smallest value that can be obtained for any cut. Fulkerson first conjectured that the minimum cut value was always equal to the maximal flow value. This was first established for so-called "planar networks" (RAND Seminar, 1954). Later Ford and Fulkerson established this as true in general [Ford and Fulkerson, 1954-1].

THEOREM 7: *The Max-flow value equals the Min-cut value.*

PROOF: We first establish that, if $\{F, \bar{x}_{ij}\}$ is any flow and C is any cut value for some arbitrary cut, then $C \geq F$. Decompose the flow into a sum of r positive chain flows. Suppose that arc $(i_1 \rightarrow j_1)$ of the cut is shared with a



Figures 19-1-IVa, b. An example of the max flow-min cut theorem.

set S_1 of chains p_1 with flow values K_{p_1}; suppose that arc $(i_2 \rightarrow j_2)$ of the cut is shared with a set, S_2, of the chains where S_2 may have chains in common with S_1, etc. We may write

(16)
$$c_{i_1,j_1} \geq \bar{x}_{i_1,j_1} = \sum_{p_1 \epsilon S_1} K_{p_1}$$
$$c_{i_2,j_2} \geq \bar{x}_{i_2,j_2} = \sum_{p_2 \epsilon S_2} K_{p_2}$$
$$\cdots\cdots\cdots\cdots\cdots\cdots$$

Because every chain $p = 1, 2, \ldots, r$ of the decomposition must have at least one arc belonging to the cut, K_p must appear in at least one of the sums above. Hence, summing the entire set of inequalities,

$$C = c_{i_1,j_1} + c_{i_2,j_2} + \cdots \geq K_1 + K_2 + \cdots + K_r = F \tag{17}$$

We shall now show that, if $F = F_0$ is the maximal flow for $\bar{x}_{ij} = \bar{x}^o_{ij}$, a subset of the saturated arcs constitutes a cut with value $C_0 = F_0$. Since in general $C \geq F$, the theorem follows. Divide the nodes of the network into two classes. In the *first* class, place all nodes that can be reached from the source node by one or more chains composed of unsaturated arcs. In the *second* class, place all the remaining nodes (if any). All directed arcs $(i \to j)$, joining a node i of the first class to a node j of the second must be saturated. (Otherwise, j could be reached from the origin via some unsaturated chain through i.) First, we will show that the set of these arcs forms a cut and, second, that its cut value is minimal.

Call the set of these saturated arcs S, and suppose there exists some chain joining source to destination that avoids all the saturated arcs of S. Since the entire set of saturated arcs, S^*, forms a cut, *let a chain, p, be chosen that shares the least number of arcs with S^* and none with S.* Let arc $(i \to j)$ be the first such saturated arc along the chain, p. It follows that node i is in the first class. Hence, j is also in the first class. (Otherwise, arc $(i \to j)$ would belong to S.) But in this case, j can be reached from the source by a chain of unsaturated arcs. This chain can then be joined to the remainder of the chain from j to the destination; the new chain now has one less saturated arc of S^* than p, contrary to the assumption that p had the least number.

We wish now to show that the cut value of S is $C_0 = F_0$. Sum the conservation relations (3) over all nodes of the first class. Variables $\bar{x}_{ij}$ and $\bar{x}_{ji}$ cancel, if both i and j are in the first class. What remains is only the sum,

$$F_0 = \sum \bar{x}^o_{ij} \qquad \text{for } (i \to j) \in S \tag{18}$$

where the i belongs to the first class and the j to the second. Since the $(i \to j)$ are all the arcs of the cut S, and since these are all saturated, we have

$$F_0 = \sum \bar{x}^o_{ij} = \sum \alpha_{ij} = C_0 \qquad [(i \to j) \in S] \tag{19}$$

and the theorem follows. It is now easy to prove:

THEOREM 8: *Given any partition of the nodes into two classes, where the first class includes the source and the second class the destination, then a feasible solution ($F = F_0$, $\bar{x}_{ij} = \bar{x}^o_{ij}$) is maximal, if every arc $(i \to j)$ is saturated that joins a node of the first class to a node of the second class.*

Observe that, if we sum the conservation equations corresponding to nodes of the first class, we obtain for *any* feasible solution,

$$F = \sum \bar{x}_{ij} \tag{20}$$

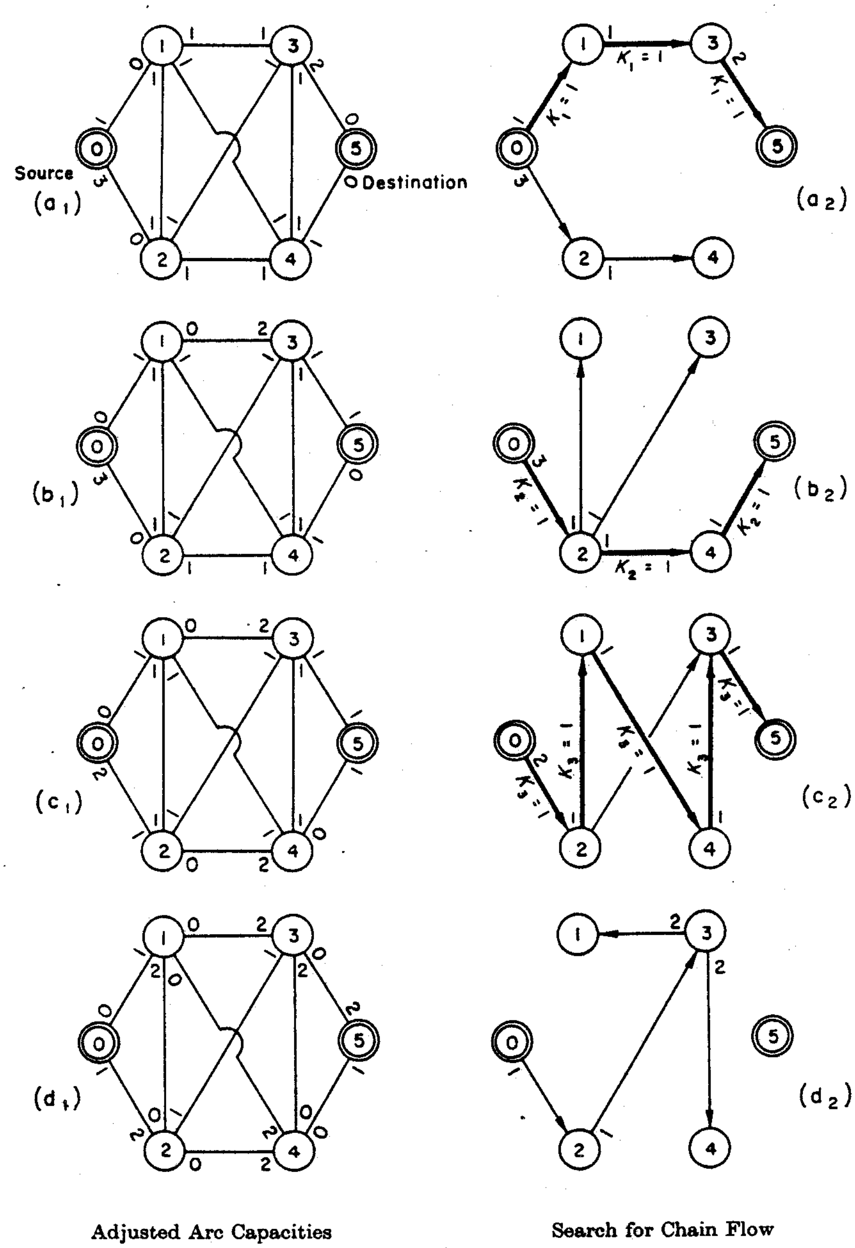


Figure 19-1-V. A second example of the Ford-Fulkerson max-flow algorithm.

where the summation extends over all arcs $(i \to j)$ such that i is in the first class and j is in the second. Because the solution is feasible, $\bar{x}_{ij} \leq \alpha_{ij}$. However, our particular solution has the property that $\bar{x}^{o}_{ij} = \alpha_{ij}$. It follows that

$$F = \sum \bar{x}_{ij} \leq \sum \alpha_{ij} = \sum \bar{x}^{o}_{ij} = F_0. \tag{21}$$

EXERCISE: Show that the set of arcs $(i \to j)$, defined above, forms a cut and its cut value is minimum.

To illustrate the method with a second example, consider Fig. 19-1-Va$_1$, where the capacities on the arcs in each direction are indicated. Thus the capacity on the arc $(0 \to 1)$, is 1, and in the reverse direction, 0, etc. Assume a starting flow, $x_{ij} = 0$, then Fig. 19-1-Va$_2$ represents a possible tree of positive arc capacities fanning out from the source. The flow can now be increased along the chain (0, 1, 3, 5) to $K_1 = 1$, at which point the arcs, (0, 1) and (1, 3), are saturated. The modified network, Fig. 19-1-Vb$_1$, is formed by setting $\alpha'_{ij} = \alpha_{ij} - K_1$ and $\alpha'_{ji} = \alpha_{ji} + K_1$ for arcs $(i \to j)$ of the chain. In Fig. 19-1-Vb$_2$ is a new tree of positive arc capacities fanning out from (0), resulting in the chain (0, 2, 4, 5). The successive solutions are shown in Fig. 19-1-Vb$_1$, Vc$_1$, and Vd$_1$, the various trees in Fig. 19-1-Vb$_2$, Vc$_2$, and Vd$_2$. Since it is not possible in the final Fig. 19-1-Vd$_2$ to reach the destination, the procedure is terminated. The sum of the chain flows found in Fig. 19-1-Va$_2$, Vb$_2$, and Vc$_2$ constitutes a maximal flow. This is shown on network Fig. 19-1-VI. Saturated arcs are marked with the symbol pointing

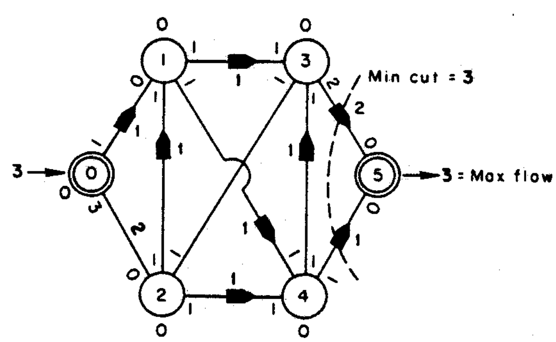


Figure 19-1-VI. Max flow-min cut solution for the second example.

in the direction of saturation. To find the cut with minimum value, choose saturated arcs leading from nodes of the first class to nodes of the second. The nodes in the first class can all be reached from the source along unsaturated chains. This set was determined by the nodes in the subtree of positive arc capacities Fig. 19-1-Vd$_2$. Hence, the nodes of the first class are 0, 2, 3, 1, 4 and the cut is made up of arcs $(3 \to 5)$ and $(4 \to 5)$ as shown in Fig. 19-1-VI.

19-2. THE TREE METHOD FOR SOLVING MAXIMAL FLOW PROBLEMS

This technique [Dantzig and Fulkerson, 1956-1; Fulkerson and Dantzig, 1955-1] is identical in principle with the one used earlier for solving capacitated transportation problems. We shall, however, illustrate a variation of it again giving a network interpretation for the maximal flow problem. Suppose we have the network in Fig. 19-2-Ia, with source A, destination B, and arc

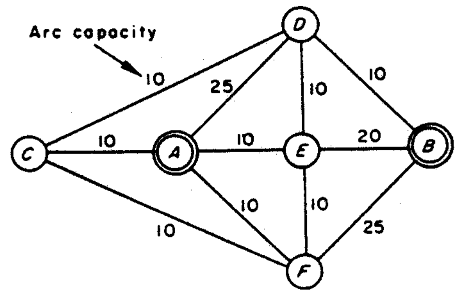


Figure 19-2-Ia. A max-flow example: The flow at start of cycle 0 is zero.

capacities as indicated; these are assumed equal in both directions. To start, select two subtrees[1] of arcs—one, T_A, branching out from the source A, the other, T_B, branching out from the destination, B, so that every intermediate node is reached by just one of the trees. For example, T_B might contain no arcs, and T_A might be made up of AC, CD, DE, EF (see Fig. 19-2-Ib). Notice

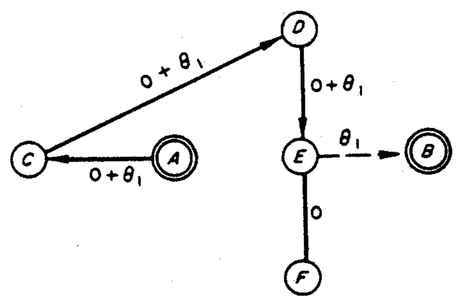


Figure 19-2-Ib. Heavy arcs form cycle 0 basis: The incremental chain flow is $\theta_1 = 10$.

that, since the network is connected, it is always possible to select two such trees. Next, introduce any arc which leads from T_A to T_B. There will be

[1] It is clear that, in a connected network with equal arc capacities in either direction, arcs may be removed until a tree is left. There is then a unique chain joining A and B. Elimination of any arc of this chain gives two trees of the kind described.

just one chain from A to B; flow as much as possible along this chain. In the example EB, is such an arc, and we have then the flow diagram of Fig. 19-2-IIa with the arcs AC, CD, DE saturated. Select any one of these

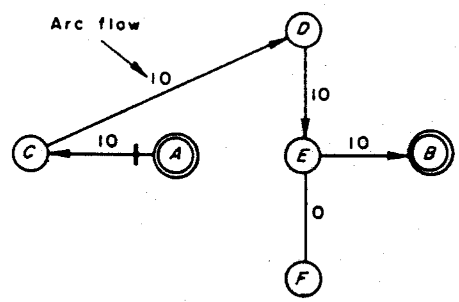


Figure 19-2-IIa. Feasible solution start of cycle 1.

saturated arcs, say AC, and place some identifying mark on it. In Fig. 19-2-IIa we have used a bar (|); this symbol will be used throughout to designate a subset of the saturated arcs. Now observe that, if the barred arc is dropped from the picture (Fig. 19-2-IIb), we again have two trees,

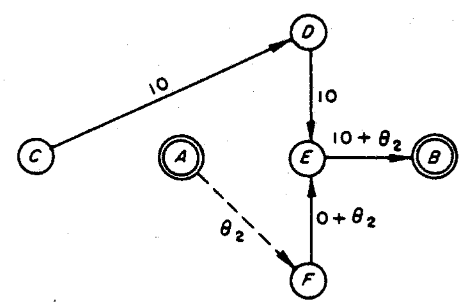


Figure 19-2-IIb. Cycle 1 basis: The incremental chain flow is $\theta_2 = 10$.

T_A consisting of no arcs and $T_B = \{EB, ED, DC, EF\}$. If the underlying basic solutions were nondegenerate, T_A and T_B would always consist of unsaturated arcs, i.e., we would always have $\alpha_{ji} < \bar{x}_{ij} < \alpha_{ij}$. Again introduce any unbarred arc leading from T_A to T_B, say AF. This creates a flow of 10 units along the chain AF, FE, EB, and saturates each arc of this chain. Select one of these, say AF and "bar" it. We now have the diagram given in Fig. 19-2-III, and we have achieved a flow of 20. Dropping barred arcs gives the same tree as in Fig. 19-2-IIb. Introduce arc AE from T_A to T_B. This leads to a situation we have not met previously, in that the chain thus constructed, namely AE, EB, cannot take any more flow because EB, though unbarred, is at its upper limit (the degenerate case). Bar EB and

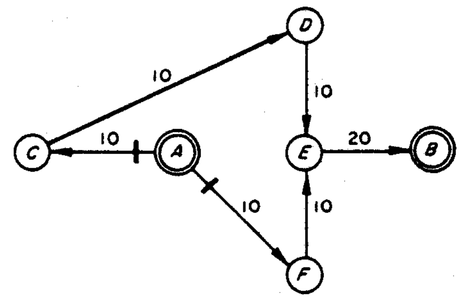


Figure 19-2-III. Feasible solution start of cycle 2.

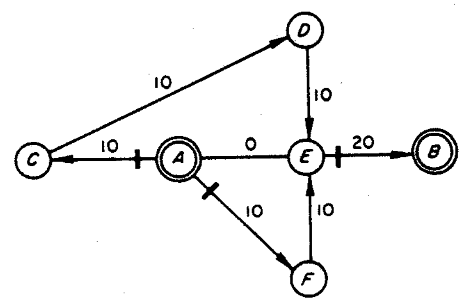


Figure 19-2-IVa. Feasible solution start of cycle 3.

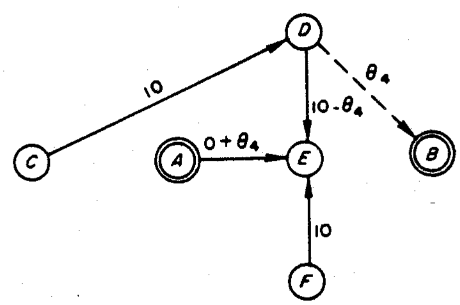


Figure 19-2-IVb. Cycle 3 basis: The incremental chain flow is $\theta_4 = 10$.

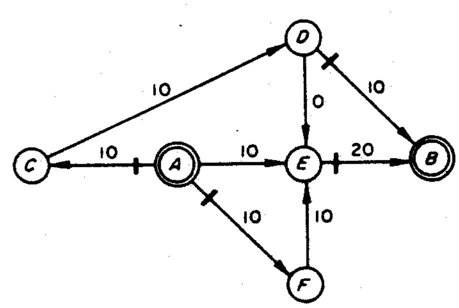


Figure 19-2-Va. Feasible solution start of cycle 4.

leave AE in with a flow of zero, obtaining Fig. 19-2-IVa, with new trees as shown in Fig. 19-2-IVb. Introduce DB to get the chain AE, ED, DB. This time we can get an increase even though DE is saturated, since the flow in Fig. 19-2-IVa is from D to E. Thus, if the flow from A to E is increased by $\theta_4 \geq 0$, the flow from D to E must be decreased by θ_4, and the flow from D to B increased by θ_4, in order to preserve the conservation equations at E and D (see Fig. 19-2-IVb). The largest possible value of θ_4 is 10, since the capacity of DB (and of AE) is 10. This cancels the flow from D to E. Bar DB and proceed.

Repeated application of this procedure produces the sequence of flows depicted in Figs. 19-2-Va, b and 19-2-VIa, b.

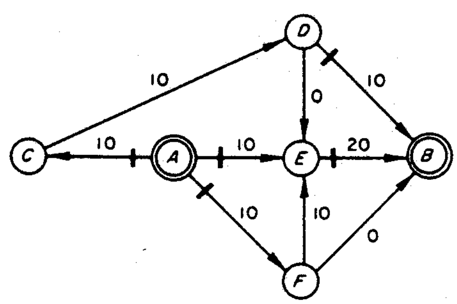


Figure 19-2-Vb. Feasible solution start of cycle 5.

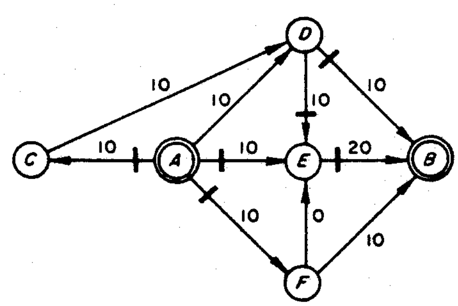


Figure 19-2-VIa. Feasible solution start of cycle 6.

Now in Fig. 19-2-VIb the trees are $T_A = \{AD, DC\}$, $T_B = \{FB, EF\}$, and there are no more arcs to introduce from T_A to T_B. At this stage examine the barred arcs connecting nodes of T_A to those of T_B. If the flow in each of these is in the right direction, that is, from T_A to T_B, an optimum has been reached, as we shall prove. If, on the other hand, the flow in one of the barred arcs which join T_A to T_B is in the wrong direction, an increase in total flow may possibly be obtained by decreasing the flow in this arc. To see this, notice that the arc in question, together with arcs of T_A and T_B,

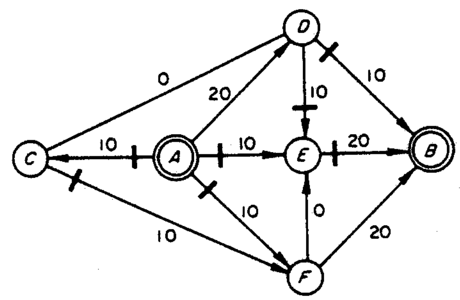


Figure 19-2-VIb. Feasible solution start of cycle 7.

will form a (unique) chain joining A and B which might look, for example, like Fig. 19-2-VII. In this case, the iterative process is continued.

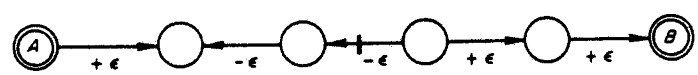


Figure 19-2-VII.

Proof of the Tree Method: Let us assume, for a general network, that the iterative process is finite and that the final subtrees, T_0 and T_n, are as shown in Fig. 19-2-VIII. Join the destination n to the source by a fictitious

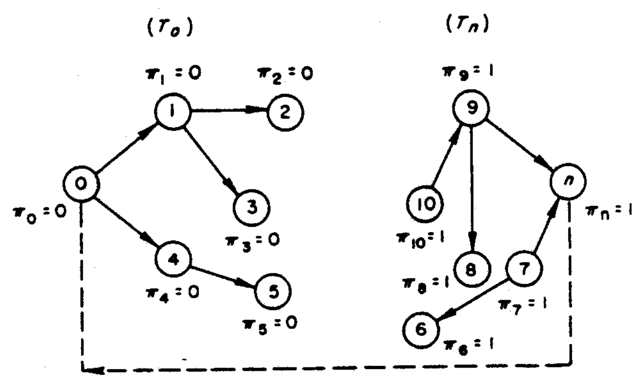


Figure 19-2-VIII. With a back flow arc, the graph of the basis forms a single tree.

back flow arc $(n, 0)$ so that now *the subtrees, T_0 and T_n, and the arc $(n, 0)$ form a single tree* in the network. The direction of the arrows on all arcs corresponds to the direction of positive flow. If the flow is zero, this direction is undefined and may be arbitrarily chosen. The capacitated transshipment problem is to maximize the flow, F, along the arc $(n, 0)$. The conditions of the problem are

$$\sum_j \bar{x}_{ij} = 0 \qquad (i = 1, 2, \ldots, n)$$
$$-\alpha_{ji} \le x_{ij} \le \alpha_{ij}$$
$$-x_{n,0} = -F \text{ (Min)}$$

Thus, in this case, $c_{n0} = -1$, while all the other c_{ij} are zero.

As shown in § 17-1, the $\bar{x}_{ij}$ corresponding to arcs (i, j) of a tree form a basic set. The simplex multipliers satisfy the equation, $\pi_j - \pi_i = c_{ij}$, for arcs (i, j), corresponding to basic variables. Since one multiplier may be arbitrarily chosen, we set $\pi_0 = 0$. Each node, i, in the subtree, T_0, will then have a multiplier, π_i, of zero, and each node in T_n will have a multiplier of unity. It is now easy to see that the final solution, $x_{ij} = x^0_{ij}$, is optimal, because all arcs (i, j) connecting T_0 to T_n, are saturated, i.e., all the corresponding non-basic variables are at their upper bound $[x_{ij} = \alpha_{ij}]$ and have nonpositive cost factors $[c_{ij} - (\pi_j - \pi_i) = -1]$, while, for all the basic variables, $c_{ij} - (\pi_j - \pi_i) = 0$. The conditions for optimality of a bounded variable problem are fulfilled. (See § 18-2-(6).)

We have assumed a finite number of iterations. It will be noted that each iteration generates two subtrees which, if joined by the arc $(n, 0)$ form a single tree. Hence, each successive solution corresponds to a basic solution. Assuming nondegeneracy, there would be a positive increase in flow on each iteration, hence it would not be possible to repeat a basis. In the case of degeneracy (which occurred frequently in the example), a randomized rule of rejection from the basis will insure with probability one against circling in the algorithm. (See Chapter 10; see also Problem 1 and § 6-1.)

19-3. PROBLEMS

1. Determine a perturbation scheme for avoiding degeneracy. Using Orden's approach (§ 14-4, Problems 14, 15), find a fixed value for ε in advance. Note that it will be necessary to use the equivalence of the capacitated flow problem with the transportation problem.
2. Solve the problem shown in Fig. 19-1-III by the Ford-Fulkerson Method, and the Simplex Method. Compare relative efficiencies.

REFERENCES

Berge, 1958-1
Boldyreff, 1955-1
Dantzig and Fulkerson, 1956-1
Egerváry, 1931-1
Flood, 1960-1
Ford and Fulkerson, 1954-1, 1955-1, 1956-1, 2, 1957-1, 1958-1, 1960-1
Fulkerson, 1956-1, 1959-1, 1960-1, 1961-1
Fulkerson and Dantzig, 1955-1
Gale, 1957-1, 1959-1, 1960-1
Gomory and Hu, 1960-1, 2
Hadley, 1961-2
Hoffman, 1960-1
Iri, 1960-1
Kuhn, 1955-1
Munkres, 1957-1
Ryser, 1961-1
Shapley, 1959-1

CHAPTER 20

THE PRIMAL-DUAL METHOD FOR TRANSPORTATION PROBLEMS

20-1. INTRODUCTION

Although the simplex method, as adapted to the transportation array by the techniques of Chapter 14, has been used successfully to solve large problems involving hundreds of equations in thousands of unknowns, the *primal-dual transportation method* presented in this chapter appears to have certain advantages. In an informal experiment by Ford, Fulkerson, and the author, this method has been compared with the simplex procedure in a number of small problems and was found to take roughly half the effort. For example, one 20×20 optimal assignment for which the simplex method required well over an hour of hand computation, was accomplished by the present method in about thirty minutes. With larger problems the advantage may be greater. However, the experience reported informally to the author has not been conclusive.

As in § 11-4, the technique keeps the relative cost factors nonnegative while it works toward feasibility, so that when a feasible solution is obtained it will *already* be optimal. Historically, the technique evolved from a combinatorial procedure called the "Hungarian Method," which was designed by H. Kuhn [1955-1] for solving a specialized assignment problem and is based on a proof, by a Hungarian mathematician, Egerváry [1931-1], for a linear graph theorem of König.

Ford and Fulkerson [1955-1] subsequently discovered a simplified algorithm for solving maximal flow problems in networks (see Chapter 19). This algorithm, when applied to the Hitchcock transportation network, serves as a substitute for part of Kuhn's procedure, enabling an analogous solution of the transportation problem and the least cost capacitated transshipment problem. A number of other authors have also developed methods for solving such problems based on Kuhn's algorithm; notably Munkres [1957-1] and Flood [1960-1].

The *out-of-kilter* method of Fulkerson [1961-2] for minimal cost flow problems generalizes the primal-dual method so that it may be initiated with an infeasible dual solution as well as an infeasible primal solution. Computer codes based on this algorithm (developed by Jack D. Little and Richard J. Clasen of RAND) are being successfully applied in several industries.

20-2. THE FORD-FULKERSON ALGORITHM

The Hitchcock problem is to find an $m \times n$ array, $x = (x_{ij})$, of nonnegative numbers, x_{ij}, which minimizes the objective function, $\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}x_{ij}$, subject to the constraints,

$$(1) \qquad \sum_{j=1}^{n} x_{ij} = a_i \qquad (i = 1, 2, \ldots, m)$$

$$\sum_{i=1}^{m} x_{ij} = b_j \qquad (j = 1, 2, \ldots, n)$$

where a_i and b_j, specified nonnegative integers, satisfy $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$. (If $m = n$ and all the a_i and b_j are equal to 1, this reduces to an optimal assignment problem.)

To describe the process, we will work through an example due to Ford and Fulkerson [1956-1]:

(2)

Surpluses	Shortages
$a_1 = 3$	$b_1 = 3$
$a_2 = 4$	$b_2 = 3$
$a_3 = 2$	$b_3 = 6$
$a_4 = 8$	$b_4 = 2$
	$b_5 = 1$
	$b_6 = 2$

Unit Shipping Costs c_{ij}

	(j) (1)	(2)	(3)	(4)	(5)	(6)
(i) (1)	5	3	7	3	8	5
(2)	5	6	12	5	7	11
(3)	2	8	3	4	8	2
(4)	9	6	10	5	10	9

The computational procedure carries along the original unit-cost matrix, (c_{ij}), and also an auxiliary array of the same size, which is the *restricted primal* array. Associated with each element, c_{ij}, of the cost matrix will be prices, u_i and v_j, such that $c_{ij} - u_i - v_j$ is nonnegative.

Associated with each row of the restricted primal array, at any stage of the procedure, will be a "surplus," and with each column a "shortage." These are the portions of the a_i and b_j still not allocated to routes.

Determining Nonnegative Values of the Relative Cost Factors, $\bar{c}_{ij}$, for Initial Selection of a Restricted Primal Array.

For each row i, assign the value, $u_i = \min_j c_{ij}$. Subtract this value from each entry in the row to form the i^{th} row of the $(c_{ij} - u_i)$ array. Then, for each column j, assign the value, $v_j = \min_i (c_{ij} - u_i)$. Subtract this value from each element, $c_{ij} - u_i$, in the column to form the j^{th} column of $(\bar{c}_{ij})$ (which equals $(c_{ij} - u_i - v_j)$), the relative-cost array. For our example, this produces (3), (4), and (5) in turn.

(3)

c_{ij}; $u_i = \operatorname{Min}_j c_{ij}$

						u_i
5	3	7	3	8	5	3
5	6	12	5	7	11	5
2	8	3	4	8	2	2
9	6	10	5	10	9	5

(4)

$c'_{ij} = c_{ij} - u_i$

	2	0	4	0	5	2
	0	1	7	0	2	6
	0	6	1	2	6	0
	4	1	5	0	5	4
$\operatorname{Min}_i c'_{ij} = v_j \rightarrow$	0	0	1	0	2	0

(5)

$\bar{c}_{ij} = c_{ij} - u_i - v_j$

2	0	3	0	3	2
0	1	6	0	0	6
0	6	0	2	4	0
4	1	4	0	3	4

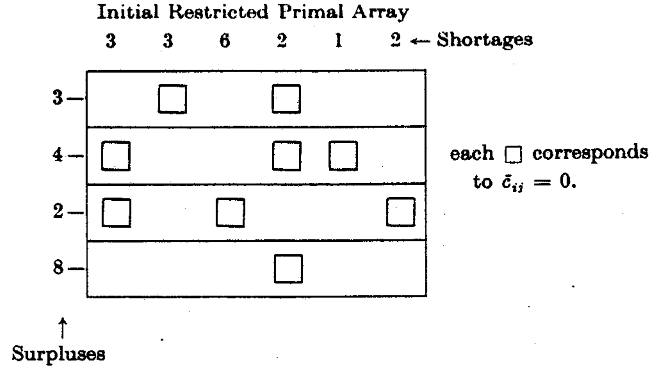
For the restricted primal problem, we seek to reduce the shortages by assigning positive values only to those x_{ij} for which $\bar{c}_{ij}$ is zero. Thus, for our example, entries must be made only in the squares with inscribed boxes of (6), since these correspond to the $\bar{c}_{ij}$ which are zero in (5).

Step 0: Initiate the Labeling

Start by labeling each row for which a surplus occurs, appending a minus sign (label) next to the surplus value.

Initial Restricted Primal Array

(6)



Step 1: Searching for a Chain

In each row containing a minus "label," append a plus "label" to each unlabeled inscribed box. If a row contains two or more minus labels, change all but one to plus, affixing minus to the box whose entry is largest. In each column containing a plus label, append a minus label to each unlabeled box containing a non-zero entry. Continue until either (a) a plus label is entered in a shortage column, in which case proceed to Step 2, or (b) it becomes impossible, under the rules, to enter any more labels—in which case proceed to Step 3.

Step 2: Allocating Shortage Along the Chain

Let k represent an amount to be determined, and begin by indicating that k is to be ***subtracted from the deficit*** in the column containing the plus label which terminated Step 1 according to rule 1(a). The procedure now consists of either (a) selecting a column where k has previously been subtracted from some entry or from the current shortage, and adding k to just one of the plus-labeled entries in the column (there is always at least one such entry), or (b) selecting the row where k has just been added to some entry, and subtracting k from the minus-labeled entry or from the minus-labeled surplus (there is only one minus label in a labeled row).

Continue until k is shown subtracted from some entry in a surplus row. Now substitute for k the value of the smallest entry which must be reduced by k and perform the indicated additions and subtractions of k. If a surplus remains in any row, erase all the plus and minus labels and return to Step 0. If no surplus remains, terminate, for feasibility has thus been achieved, and *the set of entries now constitutes an optimal solution* since all the relative cost factors have been kept nonnegative.

Step 3: Finding a New Restricted Primal

Let $(i, j) \in S$ mean that square (i, j) is in a *labeled row and an unlabeled column*. Similarly, let $(i, j) \in S'$ mean that square (i, j) is in an *unlabeled row and a labeled column*. (Note that neither S nor S' contains any labeled squares.)

Determine a constant, Δ, and a square (r, s) in S, such that

$$\Delta = \bar{c}_{rs} = \min_{(i,j)\in S} \bar{c}_{ij} \tag{7}$$

and new $\bar{c}_{ij}$ values,

$$\bar{c}^*_{ij} = \begin{cases} \bar{c}_{ij} - \Delta \text{ if } (i, j) \in S & \text{(Labeled row, unlabeled column)} \\ \bar{c}_{ij} + \Delta \text{ if } (i, j) \in S' & \text{(Unlabeled row, labeled column)} \\ \bar{c}_{ij} \quad \text{otherwise} & \end{cases}$$

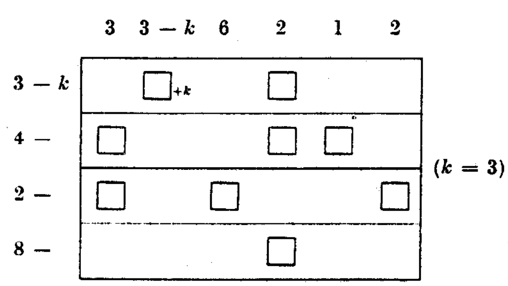
As an alternative method of effecting these changes, we may first adjust the values of u_i to $u^*_i = u_i + \Delta$ for rows i with labels and v_j to $v^*_j = v_j - \Delta$ in columns with labels. The values of c^*_{ij} may then be determined as

$$\bar{c}^*_{ij} = c_{ij} - u^*_i - v^*_j$$

In the restricted-primal array, correct the positions of the inscribed boxes to correspond to $\bar{c}^*_{ij} = 0$. There will be at least one new box in S, for $\Delta = \bar{c}_{rs}$. All the boxes in S' will drop (their entries are zero) and the others will remain unchanged. Leaving the labels intact, return to Step 1 and continue the labeling process by scanning those rows where new boxes were inscribed.

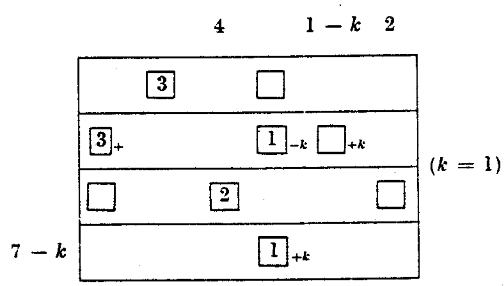
Illustration. In our example, having appended a minus label to the surpluses, we begin by scanning row 1 for inscribed boxes and by plus-labeling the x_{12} square of (6). Because the second column has surplus, rule 1(a) directs us to Step 2. This results in (8).

(8)

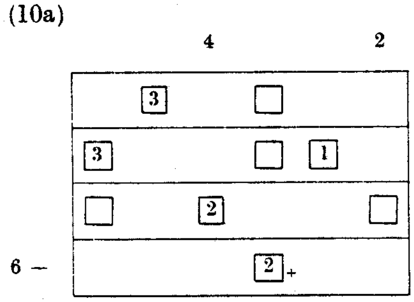


Setting $k = 3$, and adjusting the entries decreases the shortage by 3 units and completes Step 2. We now repeat Step 1 since the problem is still infeasible. After several repetitions of the Step 1-Step 2 cycle, the array appears as in (9) and then as in (10a).

(9)



(10a)



(10b)

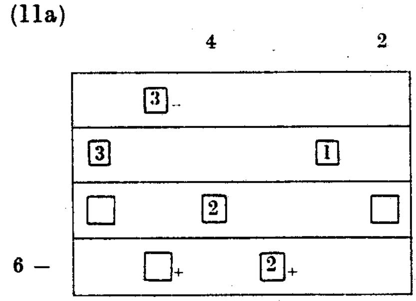
$\bar{c}_{ij}$

2	0	3	0	3	2	
0	1	6	0	0	6	
0	6	0	2	4	0	
4	1	4	0	3	4	$+\Delta_1$
			$-\Delta_1$			$(\Delta_1 = 1)$

As specified by rule 1(b), because no more plus or minus labels may be entered, we now proceed to Step 3, again using the array in (5) (which is repeated as (10b) above). The only labeled entry is in the fourth row and fourth column; S consists of the double-hatched squares and S' the single-hatched squares. Now, since the smallest element in the squares of S is 1, we subtract $\Delta = 1$ from the $\bar{c}_{ij}$ values of S and add $\Delta = 1$ to the $\bar{c}_{ij}$ values of S'. This has the effect of keeping $\bar{c}_{44}$ fixed at zero while reducing the cost factor, $\bar{c}_{42}$, to zero, as shown in (11b).

The new restricted primal problem appears in (11a), and the remainder of the procedure is shown by the subsequent displays.

(11a)

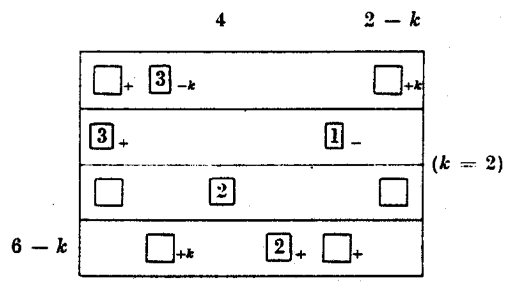


(11b)

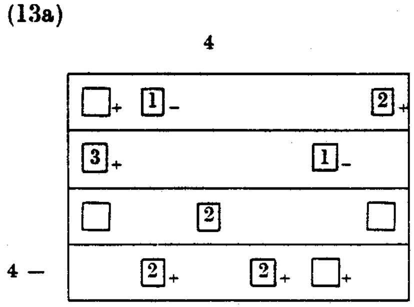
c_{ij}

2	0	3	1	3	2	$+\Delta_2$
0	1	6	1	0	6	
0	6	0	3	4	0	
3	0	3	0	2	3	$+\Delta_2$
	$-\Delta_2$		$-\Delta_2$			$(\Delta_2 = 2)$

(12)



(13a)

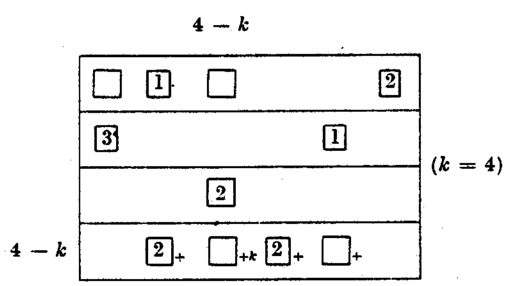


(13b)

$\bar{c}_{ij}$

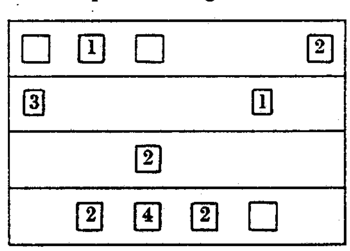
0	0	1	1	1	0	$+\Delta_3$
0	3	6	3	0	6	$+\Delta_3$
0	8	0	5	4	0	
1	0	1	0	0	1	$+\Delta_3$
$-\Delta_3$	$-\Delta_3$		$-\Delta_3$	$-\Delta_3$	$-\Delta_3$	$(\Delta_3 = 1)$

(14)



(15a)

Optimal Assignment



(15b)

Optimal $\bar{c}_{ij}$

							u_i
	0	0	0	1	1	0	6
	0	3	5	3	0	6	6
	1	9	0	6	5	1	2
	1	0	0	0	0	1	9
v_j	−1	−3	1	−4	1	−1	

The optimal set of prices, given in (15b), is formed by successively adjusting the initial values of u_i and v_j, as given in (3) and (4), by the corresponding values of $\pm\Delta$ (if any) given in (10b), (11b), (13b), and (15b).

Finiteness of the Primal-Dual Transportation Method.

Each application of Step 1 must make either Step 2 or Step 3 possible. There can be, at most, $N = \Sigma a_i = \Sigma b_j$ applications of Step 2 if a_i and b_j are integers, for, in that case, the successive k-parameters are all positive integers, and the surpluses and shortages are decreased by at least unity at each application of Step 2.

THEOREM 1: *The algorithm will not lead to Step 3 unless shortage columns (and possibly others) are unlabeled.*

This must certainly be true if any column still has shortage, since, if such a column contained a plus sign, Step 2 (and not Step 3) would have followed after Step 1, in accordance with rule 1(a); and if such a column contained a minus sign, this label would always have been preceded by a plus sign in the same column.

THEOREM 2: *There can be, at most, $n - 2$ applications of Step 3 between applications of Step 2* (where n is the number of columns).

To see this, note that a boxed entry having a label at the end of Step 1 will remain boxed after Step 3, since the only $\bar{c}_{ij}$ values modified are those situated in unlabeled rows or columns. In addition, *there will be at least one new label entered in a previously unlabeled column*, s, corresponding to $(r, s) \in S$, such that $\bar{c}_{rs} = \Delta$. Since rows of S are labeled (by definition) and every labeled row contains one minus label, we follow Step 3 by re-applying Step 1. The rules will, therefore, ascribe a plus label to the x_{rs} square. There can thus be no more than $n - 2$ successive returns to Step 3 after Step 1 before a labeling occurs in a shortage column (invoking Step 2).

THEOREM 3: *A transportation array will be optimized by this Primal-Dual Method in not more than $N(n - 2)$ cycles, where $N = \Sigma a_i = \Sigma b_j$, and n is the number of columns.*

20-3. PROBLEMS

1. Complete the proof of Theorem 3.
2. Modify the Primal-Dual Method for transportation problems to account for the inadmissibility of some x_{ij}.
3. How does the Primal-Dual Transportation Method relate to the more general primal-dual algorithm of § 11-4?
4. Show that Steps 1 and 2 constitute special application of the Ford-Fulkerson Method for achieving maximal flow in a network (see Chapter 19).
5. Explain the Primal-Dual Transportation Method by diagramming the procedure of Steps 1, 2, and 3 as a flow-chart.

6. How does the magnitude of Δ affect the amount by which the value of the dual objective function is changed during an iteration?
7. Show that, instead of starting our process of labeling by appending minus labels to surplus row, we could start by attaching minus labels to shortage columns, etc.
8. Show that the progress of the algorithm will be accelerated by double-labeling, using both of the possible labeling methods, as mentioned in the last problem, one after the other during a single application of Step 1.
9. *Warehouse problem:* Suppose a number of commodities i can be purchased (or sold) at prices p_{it} at the start of period t; it costs s_{it} to store it for one period. If the capacity, S, of a storage warehouse is fixed, what is the optimal storage, purchase, sales program? Show that an item is either used to completely stock a warehouse or, if in stock, is either completely held or sold out. References: Cahn [1948-1], Charnes and Cooper [1955-1], Dantzig [1949-1; 1957-3], Prager [1957-1].

REFERENCES

Cahn, 1948-1
Charnes and Cooper, 1955-1
Dantzig, 1949-1, 1957-3
Egerváry, 1931-1
Flood, 1960-1
Ford and Fulkerson, 1955-1, 1956-1, 1960-1
Fulkerson, 1959-2, 1961-2
Garvin, 1960-1
Hadley, 1961-2
Hoffman, 1960-1
Jewell, 1957-1
Kuhn, 1955-1
Munkres, 1957-1
Prager, 1957-1