

Contents

6 Primal Algorithm for the Transportation Problem	245
6.1 The Balanced Transportation Problem	245
6.2 An Application at a Bus Rental Company	246
6.3 Special Properties of the Problem	249
6.4 Notation Used to Display the Data	252
6.5 Routine for Finding an Initial Feasible Basic Vector and its BFS	253
6.6 How to Compute the Dual Basic Solution and Check Optimality	262
6.7 A Pivot Step: Moving to an Improved Adjacent Basic Vector	264
6.8 The Primal Simplex Algorithm for the Balanced Trans- portation Problem	274
6.9 Marginal Analysis in the Balanced Transportation Prob- lem	278
6.10 What to do if There is Excess Supply or Demand . . .	280
6.11 Exercises	282

Chapter 6

Primal Algorithm for the Transportation Problem

This is Chapter 6 of “Junior Level Web-Book for Optimization Models for decision Making” by Katta G. Murty.

6.1 The Balanced Transportation Problem

We consider the transportation problem with the following data

- m = number of sources where material is available
- n = number of sinks or demand centers where material is required
- a_i = units of material available at source i , $a_i > 0$, $i = 1$ to m
- b_j = units of material required at sink j , $b_j > 0$, $j = 1$ to n
- c_{ij} = unit shipping cost (\$/unit) from source i to sink j , $i = 1$ to m , $j = 1$ to n

The transportation problem with this data is said to satisfy the **balance condition** if it satisfies

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

If this condition holds, the problem is known as a **balanced transportation problem**. Letting x_{ij} denote the amount of material transported from source i to sink j , $i = 1$ to m , $j = 1$ to n , the problem is (6.1.1). It is known as an **uncapacitated balanced transportation problem** (uncapacitated because there are no specified upper bounds on the decision variables x_{ij} s). It is a **transportation problem of order $m \times n$** (m is the number of sources, and n is the number of sinks here).

$$\begin{aligned} \text{Minimize } z(x) &= \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \\ \text{subject to } \sum_{j=1}^n x_{ij} &= a_i, \quad i = 1 \text{ to } m \\ \sum_{i=1}^m x_{ij} &= b_j, \quad j = 1 \text{ to } n \quad (6.1.1) \\ x_{ij} &\geq 0, \quad \text{for all } i, j \end{aligned}$$

Let $x = (x_{ij})$ be a feasible solution for it. Summing the set of first m constraints (those corresponding to the sources), and the set of last n constraints (those corresponding to the sinks) in it separately, we see that $\sum_{i=1}^m a_i = \sum_{i=1}^m \sum_{j=1}^n x_{ij} = \sum_{j=1}^n b_j$.

So, we see that the balance condition is a **necessary condition for the feasibility of** this problem. So, we assume that the data satisfies the balance condition.

6.2 An Application at a Bus Rental Company

The transportation model finds many many applications, and often in contexts that do not involve shipping of any commodities. We will dis-

cuss one such application at a bus rental company in Seoul, South Korea, which involves allocation of buses to trips (see: K. G. Murty, and W. J. Kim, “An i-DMSS Based on Bipartite Matching and Heuristics for Rental Bus Allocation”, Chapter 12 in *Intelligent Decision-Making Support Systems*, J. N. D. Gupta, G. A. Forgionne, and M. Mora (ed.), Decision Engineering Series, Springer-Verlag, UK, 2006). .

This company rents buses with drivers to customers who request them. Seoul is a big city and a popular destination for many tourists from all over the world. Requests for the company’s buses come from visiting student groups, business teams, wedding groups, etc. If there are tour requests of smaller durations, the same bus may be able to handle them one after the other. In this way, the company tries to combine requests of smaller durations into a bus trip for the whole day.

So, each bus trip involves the driver reporting with the bus to the first group for the day at a specified location (origin of this trip) in the city, at a specified time in the morning (trip start time), and driving the group along the route laid out for their tour (the route may involve some stops of varying durations along the way, the driver waits in the bus during such stops). After finishing this group’s tour, the driver may handle the next group’s tour on schedule for the day in the same way. And so on. The bus trip for the day ends with the last group on its schedule, at the specified end location and specified time in the evening.

The company stations buses at two depots in the city, call them Depot 1, Depot 2 (D_1, D_2). A driver allotted to a bus trip starts the bus at the depot and picks up the first group on the trip schedule in the morning. In the evening after the trip is over, the driver takes the bus from the ending location of the trip back to its depot.

The customers pay for all the driving in their tours; but the company has to pay for the drive of the bus from the depot to the starting location of the trip in the morning, and from the ending location of the trip back to the depot in the evening. That’s why these drives are called *empty load drives*. It has been estimated that the time spent in empty load driving costs the company at the rate of \$40/hour (it includes driver’s wages, fuel, maintenance, lost opportunity for the bus to make profit during that time, etc.). This is the only cost we consider

in allocating buses to trips, this cost is to be minimized.

If the company finds that the number of trips on a day is more than the number of buses that they have, they can borrow additional buses on a daily basis from other vendors. For these borrowed buses, the company has to pay an agreed upon daily rate that is much higher than the cost of empty load drives if they use their own bus. That's why the company uses their own buses as far as possible.

The number of trips handled daily varies between 30 to 50. But in this illustrative example, we will consider only 8 trips.

The sources for the buses for these trips are D_1, D_2 (Depots 1, 2), and OV (Outside vendors). Each trip needs a bus, so each of them is a sink with a demand for one bus. The total demand on this day is for 8 buses. The company has two buses stationed at D_1 , and 4 at D_2 . So, they need to borrow $8 - 2 - 4 = 2$ buses from OV on this day.

All the data is shown in the following table. In it, j is a trip, and when i is a depot, c_{ij} = cost in \$ of empty load drives from i to the starting location and back to the depot from the ending location of trip j ; based on estimated average driving speed. When i is an OV, c_{ij} = cost in \$ of borrowing a bus from i to handle trip j .

Source i	c_{ij} for trip $j =$								Availability a_i
	1	2	3	4	5	6	7	8	
1 = D_1	44	20	60	30	25	50	40	30	2
2 = D_2	45	60	80	15	35	28	43	15	4
3 = OV	300	400	250	350	450	275	250	350	2
Demand b_j	1	1	1	1	1	1	1	1	

Let x_{ij} denote the number of buses allocated from source i to trip j . This bus allocation problem is clearly the balanced transportation problem with the data given in the above table, with the restriction that x_{ij} can take only integer values. However, the integer property of the transportation problem discussed in Chapter 3 guarantees that this integer restriction can be ignored because the remaining LP has integer optimum solutions that can be found by LP algorithms.

The optimum solution (x_{ij}) for this problem computed using the algorithm discussed in later sections is given in the following table

(blank entries are zero, i.e.; only if the value of an x_{ij} is nonzero, that value is entered in cell (i, j)).

Source	x_{ij} for trip $j =$							
	1	2	3	4	5	6	7	8
D_1		1			1			
D_2	1			1		1		1
OV			1				1	

So, in the optimum solution the two buses at D_1 are allocated to trips 2, 5; the four buses from D_2 are allocated to trips 1, 4, 6, and 8; and buses borrowed from OV are allocated to trips 3, 7; resulting in a minimum cost of \$648 for these trips.

Every day the company solves the same model with the data for the trips to be handled on that day to determine the bus allocations for these trips.

6.3 Special Properties of the Problem

Redundancy in the constraints

Add the first m constraints in (6.1.1), and from the sum subtract the sum of the last n constraints. By the balance condition, this leads to the equation “0 = 0”. Hence there is a redundant constraint among the equality constraints in (6.1.1), and any one of the equality constraints in (6.1.1) can be treated as a redundant constraint and deleted from the system without affecting the set of feasible solutions. We treat the constraint corresponding to sink n

$$\sum_{i=1}^m x_{in} = b_n$$

as the redundant constraint to eliminate from (6.1.1) (one could have chosen any of the other equality constraints as being redundant instead of this one). After this constraint is deleted from (6.1.1), we

obtain the following problem in which all the equality constraints are nonredundant

$$\begin{aligned}
 \text{Minimize} \quad z(x) &= \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \\
 \text{s. to} \quad \sum_{j=1}^n x_{ij} &= a_i, i = 1 \dots m \\
 \sum_{i=1}^m x_{ij} &= b_j, j = 1 \dots n - 1 \quad (6.3.2) \\
 x_{ij} &\geq 0, \text{ for all } i, j
 \end{aligned}$$

The coefficient matrix of the system of equality constraints in (6.3.2) is of order $(m+n-1) \times mn$ and its rank is $(m+n-1)$. So, every basic vector for the balanced transportation problem of order $m \times n$ consists of $(m+n-1)$ basic variables.

The Dual Problem

Associating the dual variable u_i to the constraint corresponding to source i , $i = 1$ to m ; and the dual variable v_j to the constraint corresponding to sink j , $j = 1$ to n ; from Section 5.2, 5.3 we know that the dual of (6.3.2) is the one given below. Deleting the constraint corresponding to $j = n$ in (6.1.1) has the effect of setting $v_n = 0$ in the dual problem. So, the dual problem is

$$\begin{aligned}
 \text{Maximize} \quad w(u, v) &= \sum_{i=1}^m a_i u_i + \sum_{j=1}^n b_j v_j \\
 \text{subject to} \quad u_i + v_j &\leq c_{ij}, \text{ for all } i, j \quad (6.3.3) \\
 v_n &= 0
 \end{aligned}$$

The Complementary Slackness Optimality Conditions

Given the dual solution (u, v) , the relative cost coefficient of x_{ij} WRT it, i.e., the dual slack variable associated with it, is $\bar{c}_{ij} = c_{ij} - u_i - v_j$, for $i = 1$ to m , $j = 1$ to n . The various pairs (x_{ij}, \bar{c}_{ij}) are the complementary pairs in (6.3.2) and its dual (6.3.3). And from Chapter 5, we know that the complementary slackness conditions for optimality for a primal feasible solution $x = (x_{ij})$ and dual feasible solution $(u = (u_i), v = (v_j))$ to be optimal to the respective problems, are

$$x_{ij}\bar{c}_{ij} = x_{ij}(c_{ij} - u_i - v_j) = 0 \quad \text{for all } i, j$$

The Algorithm that We Will Discuss

Here we will discuss the version of the **primal simplex algorithm** for the balanced transportation problem that is executed without the canonical tableaus, using the special structure of this problem.

This version begins with a primal feasible basic vector obtained by a special initialization routine. The corresponding dual basic solution is then computed. If it is dual feasible, i.e., if all the relative cost coefficients \bar{c}_{ij} WRT it are ≥ 0 , the present solutions are optimal to the respective problems and the algorithm terminates.

If some $\bar{c}_{ij} < 0$, the present basic vector is not optimal (i.e., is dual infeasible). In this case the algorithm selects exactly one nonbasic variable x_{ij} corresponding to a negative \bar{c}_{ij} , and brings it into the basic vector; thus generating a new primal feasible basic vector with which the whole process is repeated.

Since the algorithm moves only among basic vectors, the complementary slackness optimality conditions hold automatically throughout the algorithm because in a BFS only basic primal variables can be nonzero, and basic relative cost coefficients are always zero. Thus the primal simplex algorithm maintains primal feasibility and complementary slackness property throughout, and in each step it tries to move closer to dual feasibility.

Because of its special structure, we can implement the primal simplex algorithm for solving the balanced transportation problem without

using the inverse tableaus, but doing all the computations on transportation arrays instead. We discuss this simpler implementation.

Forbidden Cells

In most applications involving a large number of sources and sinks, a source may not be able to transport material to all the sinks. Some of the sinks may be too far away from it, or there may be no direct route from it to all the sinks. In such applications, a set of **forbidden cells** in the transportation array is specified with the condition that there should be no transportation among cells in it. Let \mathbf{F} denote the set of all forbidden cells. For each $(i, j) \in \mathbf{F}$ we need to make sure that $x_{ij} = 0$ under this condition.

In the minimization problem (6.3.2) one way to force a variable x_{ij} to be zero in the optimum solution is to make its cost coefficient $c_{ij} = +\infty$, or a very large positive number α (taking $\alpha > (\sum_{i=1}^m a_i)(\max\{|c_{ij}| : i = 1 \text{ to } m, j = 1 \text{ to } n\})$ would do). So, we make $c_{ij} = \alpha$ for all forbidden cells $(i, j) \in \mathbf{F}$. With this change all forbidden cells have cost coefficient α and vice versa. If no forbidden cells are specified, all entries in the cost matrix will be as specified in the original data.

6.4 Notation Used to Display the Data

Computer implementations of the primal simplex algorithm for the transportation problem are usually based on the representation of the problem as a minimum cost flow problem on a bipartite network.

But for hand computation on small problems, it is convenient to work with transportation arrays discussed in Sections 3.7, 5.3. In this chapter we will discuss the procedure for applying the primal simplex method on the balanced transportation problem using transportation arrays.

Here we describe the various entries entered in these arrays in the examples given in the following sections.

Each row in the array corresponds to a source, and each column corresponds to a sink. The variable x_{ij} in the problem is associated with

cell (i, j) in the array. Forbidden cells (these correspond to variables x_{ij} which are required to be 0) have very large positive cost coefficients, and they are essentially crossed out and ignored in the algorithm (i.e., the values of the variables in them remain zero) once they become nonbasic.

The original cost coefficient c_{ij} in cell (i, j) will be entered in the lower right corner of the cell using small size numerals. The relative cost coefficients, \bar{c}_{ij} , will be entered in the upper left corner of the cells, also using small size numerals. The relative cost coefficient in every nonbasic forbidden cell will always be $+\infty$ if c_{ij} was defined to be $+\infty$, or some large positive number if c_{ij} was defined to be a large positive number.

Basic cells will have a small square in their center, with the value of the corresponding variable in the present BFS entered inside the square in normal size numerals. So, after an initial basic vector is selected, the basic vector at any stage consists of the set of cells with little squares in their center.

The availabilities at the sources and the requirements at the sinks are typeset using larger size numerals to distinguish them from the cost data. These are maintained on the array until a BFS to the problem is obtained.

The dual solution $((u_i), (v_j))$ is entered on the array again using smaller size numerals.

6.5 Routine for Finding an Initial Feasible Basic Vector and its BFS

This special routine for finding a feasible basic vector for a balanced transportation problem selects one basic cell per step, and hence needs $(m + n - 1)$ steps on a problem of order $m \times n$.

Initially, all cells in the transportation array are open for selection as basic cells. In each step, all the remaining cells in either a row or a column of the basic cell selected in that step will be crossed out from selection in subsequent steps. Also, the row and column totals will be modified after each step. The current row and column totals will

be denoted by a'_i, b'_j respectively; these will always be ≥ 0 , and they represent the remaining quantity of material still to be shipped from a source, or unfulfilled demand at a sink, at that stage. A row or column will always have an uncrossed cell not yet selected as a basic cell, that is open for selection as a basic cell, as long as the current total in it is > 0 . Initially, $a'_i = \text{original } a_i, b'_j = \text{original } b_j$, for all i, j .

Routine for Finding An Initial BFS

BEGIN

Initialization All cells in the $m \times n$ transportation array are open for selection as basic cells initially, and $a'_i = \text{original } a_i, b'_j = \text{original } b_j$, for all i, j . With these go to first step. We describe the general step.

General Step If all the remaining cells open for selection as basic cells, are all in a single row (column), select each of them as a basic cell; and make the value of the basic variable in each of them equal to the modified column (row) total at this stage. Terminate.

If the remaining cells open for being selected as basic cells are in two or more rows and two or more columns of the array at this stage, select one of them as a basic cell. Two popular rules for making this selection are given below. If (r, s) is the selected cell, make $x_{rs} = \min\{a'_r, b'_s\} = \beta$ say. It is possible for β to be zero. Subtract β from both a'_r and b'_s , this updates them. If new $a'_r = 0 < b'_s$ ($b'_s = 0 < a'_r$) cross out all remaining cells in row r (column s) from being selected as basic cells in subsequent steps. If new $a'_r = \text{new } b'_s = 0$, cross out all remaining cells in either row r or column s , but not both, from being selected as basic cells in subsequent steps. Go to the next step.

END

Rules for Selecting an Open Cell as a Basic Cell

Here we discuss two rules that are commonly used for making this selection in the above routine.

The Greedy Choice Rule Under this rule, the cell (r, s) selected as the basic cell is one which has the smallest cost coefficient among all cells open for selection at that stage.

Vogel's Choice Rule Let line refer to a row or column of the array that contains some cells open for selection at this stage. In each line compute the **cost difference**, which is the second minimum cost coefficient – minimum cost coefficient, among all open cells in this line. Identify the line that has the maximum cost difference at this stage, and select a least cost open cell in it as the basic cell in this step. The rationale for this selection is the following: If that cell is not selected, any remaining supply or demand in this line has to be shipped using an open cell with the second minimum cost or higher cost in that line, and hence results in the highest increase in unit cost at this stage.

If forbidden cells are specified in the problem, it is possible that some of them may be selected as basic cells in this routine, and the basic variables corresponding to them may have positive values in the initial BFS. If the original problem has a feasible solution in which all the forbidden variables are zero, when the simplex algorithm is applied to solve the problem beginning with the initial BFS, the forbidden basic variables will become 0 in the BFS before the algorithm terminates with an optimum solution.

Example 6.5.1: Finding an initial feasible basic vector using the greedy choice rule: Consider the iron ore shipping problem discussed in Example 3.7.1. The array for this problem containing all the data is given above. The smallest cost coefficient in the entire array is $2 = c_{13}$, so we select $(1, 3)$ as the first basic cell and make $x_{13} = \min\{200, 800\} = 200$. With this the demand in Column 3 is fully satisfied, and we cannot ship any more ore to Plant 3 corresponding to Column 3. So, we cross out cell $(2, 3)$ in column 3 from being selected as a basic cell, and enter CR in it to indicate this fact. We also change

the amount still to be shipped from mine 1 to $800 - 200 = 600$. The array at this stage is given below.

Array 6.5.1

	Plant			a'_i
	1	2	3	
Mine 1			200	600
	11	8	2	
Mine 2			CR	300
	7	5	4	
b'_j	400	500		

The least cost cell among the remaining open cells is $(2, 2)$ with cost coefficient 5, which is selected as the next basic cell, and we make $x_{22} = \min\{500, 300\} = 300$. As before, we change the remaining requirement at plant 2 to $500 - 300 = 200$, cross out the remaining cell $(2, 1)$ in the saturated row 2 from being selected as a basic cell, and get the situation in the next array.

	Plant			a'_i
	1	2	3	
Mine 1			200	600
	11	8	2	
Mine 2	CR	300	CR	
	7	5	4	
b'_j	400	200		

Now the remaining open cells are $(1, 1)$, $(1, 2)$, both in row 1, so we select both of them as basic cells and make $x_{11} = 400$, and $x_{12} = 200$, and obtain the basic vector and associated BFS marked in the following array. The transportation cost in this BFS is $\sum \sum c_{ij}x_{ij} = 11 \times 400 + 8 \times 200 + 5 \times 300 + 2 \times 200 = \7900 .

Array 6.5.2: The basic vector and BFS

		Plant			
		1	2	3	
Mine 1	400	200	200		
	11	8	2		
Mine 2		300			
	7	5	4		

Example 6.5.2: : Finding an initial basic vector using Vogel's choice rule: Here we will find an initial feasible basic vector for the iron ore transportation problem in Example 3.7.1 using Vogel's rule for selecting basic cells in each step. In row 1, the smallest and second smallest cost coefficients are 2, 8, and hence the cost difference in row 1 is $8 - 2 = 6$. In the same way, the cost differences for all the rows and columns in the array are computed and given below.

Line	Cost difference	
Row 1	8	- 2 = 6
Row 2	5	- 4 = 1
Col. 1	11	- 7 = 4
Col. 2	8	- 5 = 3
Col. 3	4	- 2 = 2

The highest cost difference occurs in row 1, and hence we select the least cost cell $(1, 3)$ in it as the first basic cell, and get the same

situation as in Array 6.5.1 given above. Now column 3 is done, and we recompute the cost difference for the remaining lines using only the data from the remaining open cells. These are given below.

Line	Cost difference
Row 1	$11 - 8 = 3$
Row 2	$7 - 5 = 2$
Col. 1	$11 - 7 = 4$
Col. 2	$8 - 5 = 3$

The highest cost difference occurs in column 1, and hence we select the least cost open cell in it, $(2, 1)$, as the next basic cell and make $x_{21} = \min\{400, 300\} = 300$. This leads to the array given below.

	Plant			a'_i
	1	2	3	
Mine 1			200	600
	11	8	2	
Mine 2	300	CR	CR	
	7	5	4	
b'_j	100	500		

Now the only remaining open cells, $(1, 1), (1, 2)$, are in row 1, so we select both of them as basic cells and make $x_{11} = 100, x_{12} = 500$, leading to the basic vector given in the following array. The transportation cost in this BFS is $11 \times 100 + 8 \times 500 + 7 \times 300 + 2 \times 200 = \7600 . Verify that this BFS is better than the BFS obtained with the greedy choice rule in Example 6.5.1.

		Plant		
		1	2	3
Mine 1	100	500	200	
	11	8	2	
Mine 2	300			4
	7	5		

The computation of cost differences, and finding the maximum among them, imposes additional work in each step when using Vogel's selection rule. The effort needed to do this additional work is very worthwhile, as Vogel's rule usually produces a much better BFS than the simple greedy selection rule. Unfortunately, neither rule can guarantee that the BFS produced will be optimal, hence it is necessary to check the BFS for its optimality. Empirical tests show that the BFS produced by Vogel's rule is usually near optimal. So, some practitioners do not bother to obtain a true optimum solution to the problem, instead they implement the initial BFS obtained by using Vogel's selection rule. When used this way, the method is called **Vogel's approximation method** (or **VAM** in short) for the balanced transportation problem.

Nondegenerate, Degenerate BFSs

As discussed in Chapter 4, a BFS corresponding to a feasible basic vector for the uncapacitated balanced transportation problem is primal nondegenerate if all primal basic variables are > 0 in it, primal degenerate otherwise. In both the BFSs obtained in Examples 6.5.1 and 6.5.2 for the iron ore transportation problem, all the 4 basic variables are > 0 , hence they are both primal nondegenerate for that problem. We will now consider an example which leads to a primal degenerate BFS.

Example 6.5.3: : Example of a primal degenerate BFS:
 Consider the following balanced transportation problem with data given

		Sink			a_i
		1	2	3	
Source 1		1	2	3	15
	2	4	4	10	19
3		6	5	15	11
	b_j	7	8	30	

		Sink			a'_i
		1	2	3	
Source 1		7			8
	2	1	2	3	
3	CR	4	4	10	19
	CR	6	5	15	11
b'_j			8	30	

in the 1st array above. We will use the greedy selection rule for selecting basic cells in each step to get an initial BFS. The least cost cell (1, 1) is

selected as the first basic cell, and $x_{11} = \min\{7, 15\} = 7$. So, all other cells in column 1 are crossed out from being selected. The position at this stage is indicated in the 2nd array above.

Sink			a'_i
	1	2	3
Source 1	7 1	8 2	CR 3
2	CR 4	4	10 19
3	CR 6	5	15 11
b'_j		0	30

Array 6.5.4

Sink			
	1	2	3
Source 1	7 1	8 2	3
2		0 4	19 10
3		5 6	11 15

The least cost cell among open cells now, (1, 2), with a cost coefficient of 2, is selected as the next basic cell, and we make $x_{12} = \min\{8,$

$8\} = 8$. At this stage we modify the totals in both row 1 and column 2 to 0, and have to cross out all remaining cells in one of them from being selected in subsequent stages. Suppose we select row 1 for this. This leads to the next array at the top of previous page.

Next we select $(2, 2)$ as a basic cell, and make $x_{22} = \min\{0, 19\} = 0$, and cross out the remaining cell in column 2. The remaining open cells are both in column 3, so we select both of them as basic cells. This leads to the BFS in Array 6.5.4

In this BFS the basic variable $x_{22} = 0$, hence it is primal degenerate. It is necessary to record the zero valued basic variables clearly so as to distinguish them from nonbasic variables which are always 0 in every BFS. For the $m \times n$ balanced transportation problem, every basic vector must have exactly $(m + n - 1)$ basic variables or cells.

6.6 How to Compute the Dual Basic Solution and Check Optimality

As discussed in Chapter 5, given a feasible basic vector \mathcal{B} for (6.3.2), the dual basic solution associated with it can be computed by solving the following system of equations. This system is obtained by treating all the dual constraints in the dual (6.3.3) corresponding to basic variables in \mathcal{B} as equations. The last equation $v_n = 0$ is associated with the constraint corresponding to sink n which we have treated as a redundant constraint in (6.3.2) and eliminated.

$$\begin{aligned} u_i + v_j &= c_{ij} \text{ for each basic cell } (i, j) \in \mathcal{B} \\ v_n &= 0 \end{aligned}$$

For the $m \times n$ transportation problem, there are $m+n$ dual variables, and $m+n-1$ basic variables in every basic vector. So, the above system of $m+n$ equations in $m+n$ unknowns, is a square nonsingular system of equations with a unique solution. This is the reason for requiring that all the zero valued basic variables be recorded carefully, without them the above system will not be a square system for computing the dual basic solution uniquely.

The special structure of the transportation problem makes it possible to solve the above system very easily by back substitution beginning with $v_n = 0$. Since we know $v_n = 0$, from these equations corresponding to basic cells in column n of the array, we can get the values of u_i for rows i of these basic cells. Now column n is processed. Knowing the values of these u_i , again from these equations corresponding to basic cells in the remaining columns in these rows, we can get the values of v_j for columns j of these basic cells. Now the rows of basic cells in column n are processed, and we continue the method with the columns of the newly computed v_j in the same way, until all the u_i and v_j are computed.

Having obtained the dual basic solution (u, v) corresponding to \mathcal{B} , we compute the relative cost coefficients $\bar{c}_{ij} = c_{ij} - u_i - v_j$ in all nonbasic cells (i, j) . The optimality criterion is

Optimality criterion: $\bar{c}_{ij} \geq 0$ for all nonbasic (i, j)

If the optimality criterion is satisfied, then (u, v) is dual feasible and hence \mathcal{B} is a dual feasible basic vector. Since \mathcal{B} is also primal feasible, by the results discussed in Chapter 5 it is an optimal basic vector, and the primal and dual basic solutions associated with it are optimal to the respective problems.

Example 6.6.1: Consider the basic vector in Array 6.5.2 for

Array 6.6.1

		Plant			u_i
		1	2	3	
Mine 1	400	200	200		
	11	8	2	2	
Mine 2	-1		5		
	7	300	5	4	-1
v_j	9	6	0		

the iron ore shipping problem. To compute the dual basic solution, we start with $v_3 = 0$. Since $(1, 3)$ is a basic cell in this basic vector, we have

$u_1 + v_3 = c_{13} = 2$, so $u_1 = 2$, and the processing of column 3 is done. As $(1, 1)$, $(1, 2)$ are basic cells, we have $u_1 + v_1 = c_{11} = 11$, $u_1 + v_2 = c_{12} = 8$, and since $u_1 = 2$ these equations yield $v_1 = 9$, $v_2 = 6$; and the processing of row 1 is done. As $(2, 2)$ is a basic cell, we have $u_2 + v_2 = c_{22} = 5$, and from $v_2 = 6$ this yields $u_2 = -1$. Now we have the complete dual solution; it is entered in the array given above.

The relative cost coefficients of the nonbasic cells $(2, 1)$, $(2, 3)$ are $\bar{c}_{21} = c_{21} - u_2 - v_1 = 7 - (-1) - 9 = -1$, $\bar{c}_{23} = c_{23} - u_2 - v_3 = 4 - (-1) - 0 = 5$. These are entered in the upper left corners of these cells. Since $\bar{c}_{21} < 0$, the optimality criterion is not satisfied in this basic vector.

6.7 A Pivot Step: Moving to an Improved Adjacent Basic Vector

When we have a feasible basic vector \mathcal{B} associated with the BFS $\bar{x} = (\bar{x}_{ij})$, which does not satisfy the optimality criterion, the primal simplex algorithm obtains a better solution by moving to an adjacent basic vector by replacing exactly one basic variable with a nonbasic variable x_{ij} whose relative cost coefficient $\bar{c}_{ij} < 0$, i.e., for which the optimality criterion is violated. That's why the set of nonbasic cells $\mathbf{E} = \{(i, j) : \bar{c}_{ij} < 0\}$, is called **the set of cells eligible to enter the present basic set**.

The method selects exactly one of these eligible nonbasic cells as the **entering cell**. This selection can be made arbitrarily, but a couple of rules that are used most commonly for solving small size problems by hand computation are the following.

First eligible cell encountered When computing the relative cost coefficients, the moment the first negative one turns up, select the corresponding cell as the entering cell. You don't even have to compute the relative cost coefficients in the remaining nonbasic cells.

Most negative \bar{c}_{ij} rule Under this rule, you compute the relative

cost coefficients in all the nonbasic cells, and if the optimality criterion is not satisfied, select as the entering cell the eligible cell (i, j) with the most negative \bar{c}_{ij} (break any ties arbitrarily). This rule is also called **the minimum \bar{c}_{ij} rule**, or **Dantzig's rule**.

Since every basic vector for the $m \times n$ transportation problem has exactly $m + n - 1$ basic cells, when an entering cell is brought into the basic vector, some present basic cell should be dropped from the basic vector, this cell is called **the dropping basic cell**, and the variable corresponding to it is called **the dropping basic variable**.

To determine the dropping basic variable and the new BFS, the following procedure is used. All the nonbasic variables other than the entering variable are fixed at the present value of 0, and the value of the entering variable is changed from 0 (its present value) to a value denoted by θ . So, if the entering cell is (p, q) , the procedure changes the value of x_{pq} from its present 0 (since it is a nonbasic variable) to θ . Now to make sure that exactly a_p units are shipped out of source p , and b_q units are shipped to sink q , we have to add a $-\theta$ to one of the basic values in row p , and another $-\theta$ to one of the basic values in column q . These subsequent adjustments have to be made among basic values only, because every nonbasic variable other than the entering variable is fixed at its present value of 0. There is a unique way of continuing these adjustments among basic values, adding alternately a $-\theta$ to the basic value in one basic cell, and then a balancing $+\theta$ to the basic value in another basic cell; until all the adjustments cancel each other in every row and column, so that the new solution again satisfies all the equality constraints in the problem. All the cells which have the value in them modified by the adjustment process belong to a loop called the **θ -loop**. It satisfies the following properties.

- (i): Every row and column of the array either has no cells in the θ -loop; or has exactly two cells, one with a $+\theta$ adjustment, and the other with a $-\theta$ adjustment.
- (ii): All the cells in the θ -loop other than the entering cell are present basic cells.
- (iii): No proper subset of a θ -loop satisfies property (i).

This set of cells is called the θ -loop in $\mathcal{B} \cup \{(p, q)\}$. On small problems being solved by hand, the θ -loop in $\mathcal{B} \cup \{(p, q)\}$ can easily be found by **trial and error**, starting with a $+\theta$ entry in cell (p, q) , and alternately adding one new $-\theta$, $+\theta$ entry among basic cells, backtracking on the selection when necessary. There is a very efficient procedure for finding θ -loops directly, which can be programmed easily; but to learn this procedure you need to know spanning trees in networks and their properties. This procedure is discussed in graduate level textbooks, for example [K. G. Murty, 1983].

Cells in the θ -loop with a $+\theta$ adjustment are called the **recipient cells**. The only nonbasic recipient cell is the entering cell. Cells with a $-\theta$ adjustment are called the **donor cells**. All donor cells are basic cells.

So, the new solution obtained by fixing all nonbasic variables other than the entering variable at their present value of 0, changing the value of the entering variable x_{pq} from its present 0 to θ , and then reevaluating the values of the basic variables so as to satisfy all the equality constraints in the problem, is $x(\theta) = (x_{ij}(\theta))$ where

$$x_{ij}(\theta) = \begin{cases} \bar{x}_{ij} & \text{the value of the basic variable in the} \\ & \text{present BFS, if } (i, j) \text{ is not in the } \theta\text{-loop} \\ \bar{x}_{ij} + \theta & \text{if } (i, j) \text{ is a recipient cell in the } \theta\text{-loop} \\ \bar{x}_{ij} - \theta & \text{if } (i, j) \text{ is a donor cell in the } \theta\text{-loop} \end{cases}$$

Since the shipments in all the recipient cells have to be increased, and in all the donor cells have to be decreased, the net cost of making a unit adjustment along the θ -loop in $\mathcal{B} \cup \{(p, q)\}$ is

$$\sum \left(c_{ij} : \begin{array}{c} \text{over recipient} \\ \text{cells } (i, j) \text{ in the} \\ \theta\text{-loop} \end{array} \right) - \sum \left(c_{ij} : \begin{array}{c} \text{over donor cells} \\ (i, j) \text{ in the } \theta\text{-loop} \end{array} \right)$$

and this will always be equal to the relative cost coefficient \bar{c}_{pq} of the nonbasic entering cell (p, q) in the θ -loop. We state this fact in the following theorem.

THEOREM 6.7.1 *Let \mathcal{B} be a basic set of cells for the $m \times n$ transportation problem WRT which the relative cost coefficients for nonbasic cells are (\bar{c}_{ij}) . Let (p, q) be a nonbasic cell. Then the set of cells $\mathcal{B} \cup \{(p, q)\}$ contains a unique θ -loop which can be obtained by putting a $+\theta$ entry in the nonbasic cell (p, q) , and alternately entries of $-\theta$ and $+\theta$ among basic cells as described above, until these adjustments cancel out in each row and column. This θ -loop satisfies conditions (i), (ii), (iii) stated above. And the net cost of this θ -loop as defined above is \bar{c}_{pq} , the relative cost coefficient in (p, q) WRT \mathcal{B} .*

For a proof of this theorem see [K. G. Murty, 1983]. Now considering the present BFS \bar{x} WRT the basic set \mathcal{B} , and the new solution $x(\theta)$ obtained as above, with the nonbasic cell (p, q) as the entering cell, we find from Theorem 6.7.1 that the objective value of $x(\theta)$ is $z(x(\theta)) = z(\bar{x}) + \theta(\text{net cost of the } \theta\text{-loop in } \mathcal{B} \cup \{(p, q)\}) = z(\bar{x}) + \theta\bar{c}_{pq}$.

Thus the relative cost coefficient \bar{c}_{pq} in the nonbasic cell (p, q) is the rate of change in the objective value, per unit change in the value of the nonbasic variable x_{pq} from its present value of 0, while all the other nonbasic variables stay fixed at their present value of 0.

This is the reason for selecting the entering cell to be one with a negative relative cost coefficient, since it can lead to an improved solution with reduced objective value. If the relative cost coefficient of the entering cell is 0(> 0), you get a solution with the same (higher) objective value. This also explains the rationale behind the optimality criterion. If all nonbasic relative cost coefficients are ≥ 0 , there is no way we can get a new feasible solution with a strictly better objective value by increasing the values of any nonbasic variables from their present values of 0.

Since $z(x(\theta)) = z(\bar{x}) + \theta\bar{c}_{pq}$, and $\bar{c}_{pq} < 0$, as θ increases, the objective value of $x(\theta)$ decreases. To get the best solution in this step, we should give θ the maximum value it can have. As θ increases, the value of $x_{ij}(\theta)$ decreases in donor cells (i, j) . So, for $x(\theta)$ to remain feasible to the problem, we need $\bar{x}_{ij} - \theta \geq 0$ for all donor cells (i, j) in the θ -loop. Thus the maximum value that θ can have while keeping $x(\theta)$ feasible is

$$\theta = \min\{\bar{x}_{ij} : (i, j) \text{ a donor cell in the } \theta\text{-loop}\}$$

The value of θ defined above is called **the minimum ratio** for the operation of bringing the entering cell (p, q) into the present basic vector \mathcal{B} .

All donor cells (i, j) which tie for the minimum in the above equation are said to be **eligible to drop from the present basic vector when (p, q) enters it**. When θ is made equal to the minimum ratio defined in the above equation in $x(\theta)$, $x_{ij}(\theta)$ becomes zero in all these cells (i, j) . One of these eligible to drop basic cells is selected as the dropping cell to be replaced by the entering cell, leading to the new feasible basic vector. $x(\theta)$ with the value of θ defined by the above equation is the BFS associated with it; its objective value is $z(x(\theta)) = z(\bar{x}) + \theta \bar{c}_{pq} \leq z(\bar{x})$ since $\bar{c}_{pq} < 0$ and $\theta \geq 0$. Since $x_{ij}(\theta) = 0$ for the dropping cell (i, j) , it becomes a nonbasic cell now. Any other donor cells which tied for the minimum in the above equation will stay as basic cells with the value of the basic variable in them zero in the new BFS.

After selecting the entering cell (p, q) , all the work involved in finding the the new basic vector and its BFS is called a **pivot step**. In a pivot step the basic vector changes by exactly one variable.

Example 6.7.1: : Example of a pivot step: Consider the feasible basic vector and the BFS displayed in Array 6.6.1 for the problem discussed in Example 6.5.1. The nonbasic cell $(2, 1)$ with relative cost coefficient $\bar{c}_{21} = -1$ is the only cell eligible to enter this basic vector. $x(\theta)$ is marked with $+\theta$, $-\theta$ entries in the following array.

	Plant			
	1	2	3	u_i
Mine 1	$400 - \theta$	$200 + \theta$	200	2
	11	8	2	2
Mine 2	-1 θ	$300 - \theta$	5	4
	7	5	4	-1
v_j	9	6	0	

The recipient cells in this θ -loop are (2, 1), (1, 2); and the donor cells are (1, 1), (2, 2). The net cost of making a unit adjustment along this θ -loop is $c_{21} + c_{12} - c_{11} - c_{22} = 7 + 8 - 11 - 5 = -1 = \bar{c}_{21}$, verifying the statement in Theorem 6.1. For $x(\theta)$ to be feasible, we need $400 - \theta \geq 0, 300 - \theta \geq 0$, i.e., the maximum value that θ can have is $\min\{400, 300\} = 300$ which is the minimum ratio. When $\theta = 300$, $x_{22}(\theta)$ becomes zero, it is the only basic cell with this property, so it is the dropping basic cell. So, we put $\theta = 300$ and replace the basic cell (2, 2) by the entering cell (2, 1) leading to the next basic vector given in Array 6.7.1. Since $\bar{c}_{21} = -1$, and minimum ratio $\theta = 300$, the change in the objective value in this pivot step is $-1 \times 300 = -300$ (it drops from \$7900 to \$7600).

Array 6.7.1

	Plant			u_i
	1	2	3	
Mine 1	100 11	500 8	200 2	2
Mine 2	300 7	1 5	6 4	-2
v_j	9	6	0	

We computed the dual basic solution and the nonbasic relative cost coefficients WRT the basic vector in Array 6.7.1 and entered them. Since all the nonbasic relative cost coefficients are > 0 , the optimality criterion holds in Array 6.7.1, hence the BFS there is an optimum solution to the problem and its cost is \$7600. This solution requires shipping

- 100 tons of ore from mine 1 to plant 1
- 500 tons of ore from mine 1 to plant 2
- 200 tons of ore from mine 1 to plant 3
- 300 tons of ore from mine 2 to plant 1

Example 6.7.2: An example of the trial and error method

for finding the θ -loop: Here we give another example of using the trial and error method to find the θ -loop in $\mathcal{B} \cup \{(p, q)\}$ where \mathcal{B} is a given feasible basic set for a balanced transportation problem, and (p, q) is the selected nonbasic entering cell. In the following array, the basic vector consists of all the cells with a square in the middle, with the value of the corresponding basic variable in the BFS entered inside this square.

Array 6.7.2

	1	2	3	4	5	u_i
1	1 6	$18 - \theta$	$20 + \theta$	7 4 5	-6 5	11
2	$27 - \theta$ 10	7 15	6 15	11 14	$35 + \theta$ 16	16
3	$0 + \theta$ 8	-1 5	$27 - \theta$ 7	10 11	-5 9	14
4	-6 13	-13 θ 4	-2 16	25 12	$65 - \theta$ 25	25
v_j	-6	-8	-7	-13	0	

Relative cost coefficients in nonbasic cells are entered in the upper left corner of the cell as usual. Every cell with a negative cost coefficient is eligible to enter this basic vector; of these we selected the cell (4, 2) as the entering cell.

We make the value of $x_{42} = \theta$ by putting a θ in the center of cell (4, 2). All other nonbasic variables remain fixed at their present value of 0. To continue to satisfy the equality constraints in the problem, we need to add a $-\theta$ to the value in one of the basic cells in row 4, i.e., in cells (4, 4) or (4, 5). If we add $-\theta$ to \bar{x}_{44} , since this is the only basic cell in column 4, we cannot make the next balancing correction of adding a $+\theta$ in another basic cell in this column. So, the basic cell

(4, 4) is the wrong cell to make the $-\theta$ adjustment in row 4, hence this adjustment must be made in the basic cell (4, 5). This is the trial and error feature of this procedure. Continuing in this manner, we get the entire θ -loop in this example, marked in the above array.

Exercises

6.7.1.

- (i): Write the donor, recipient cells in the θ -loop in Array 6.7.2.
- (ii): Verify that the net cost of making a unit adjustment along the θ -loop in Array 6.7.2 is $-13 = \bar{c}_{42}$, the relative cost coefficient of the entering cell.
- (iii): Compute the cost of the present BFS (remember that $\theta = 0$ in it).
- (iv): Find the minimum ratio, and select a dropping basic cell when (4, 2) enters this basic vector. Draw another array, and mark the new basic vector and the new BFS in it.
- (v): Compute the cost of the new BFS and verify that it is = cost of the old BFS + $\theta\bar{c}_{42}$.
- (vi): Is the new basic vector optimal? Why?
- (vii): If the new basic vector is not optimal, continue the process by selecting an entering variable into it and performing a pivot step. Repeat until you get an optimum solution to the problem.

How to Find the θ -loop in a Pivot Step

We only discussed a trial and error procedure for finding the θ -loop in a pivot step. This trial and error procedure is fine for solving small problems by hand computation, but it is very inefficient for solving large real world problems on a computer. Efficient methods for finding θ -loops are based on predecessor labeling schemes for storing tree structures; see [K. G. Murty, 1983] for details on them. Using these efficient schemes, large scale transportation problems with thousands of sources and sinks can be solved in a matter of seconds on modern digital computers.

Nondegenerate, Degenerate Pivot Steps

A BFS for (6.3.2) is said to be **nondegenerate** if all the $m + n - 1$ basic variables are strictly > 0 in it; otherwise it is said to be **degenerate**. The BFSs found in Examples 6.5.1 and 6.5.2 are all nondegenerate, while the BFS found in Example 6.5.3 is degenerate.

Let \mathcal{B} be a feasible basic vector for the problem, associated with the BFS $\bar{x} = (\bar{x}_{ij})$. If the nonbasic cell (p, q) is selected as the entering cell into \mathcal{B} , the ensuing pivot step is said to be a **nondegenerate pivot step** if the minimum ratio in it, θ is > 0 ; **degenerate pivot step** if this minimum ratio is 0.

If \bar{x} is a nondegenerate BFS, since the minimum ratio in this step is the minimum of \bar{x}_{ij} over all donor cells (i, j) in the θ -loop, all of which are basic cells, it is strictly > 0 , and hence this pivot step will be a nondegenerate pivot step. So, a pivot step in a basic vector can only be degenerate if that basic vector is primal degenerate. Even if the BFS \bar{x} is degenerate, if all the donor cells (i, j) in the θ -loop satisfy $\bar{x}_{ij} > 0$, the pivot step will be nondegenerate.

If \bar{c}_{pq} is the relative cost coefficient of the entering cell, we have seen that the objective value of the new BFS obtained at the end of this pivot step is $=$ (objective value of the old BFS) $+ \theta\bar{c}_{pq}$. Since $\bar{c}_{pq} < 0$ and $\theta > 0$ in a nondegenerate pivot step, the change in the objective value in it, $\theta\bar{c}_{pq}$, is < 0 . Thus after a nondegenerate pivot step, we will obtain a new BFS with a strictly better objective value. In a degenerate pivot step, $\theta = 0$, and hence the BFS and its objective value do not change, but we get a new basic vector corresponding to the same old BFS with a different set of zero valued basic variables in it.

The pivot steps discussed in Examples 6.7.1 and 6.7.2 are nondegenerate pivot steps, since the minimum ratios are > 0 in them.

Example 6.7.3: : Example of a degenerate pivot step
 Consider the degenerate BFS associated with the basic vector in Array 6.5.4 derived in Example 6.5.3. The dual basic solution and the relative cost coefficients WRT this basic vector are given below.

Array 6.7.3

		Sink			u_i
		1	2	3	
Source 1		$7 - \theta$	$8 + \theta$	-5	
	1	1	2	3	8
2	1 4	$0 - \theta$	4	$19 + \theta$	10 10
3	-2 θ 6	-4	5	$11 - \theta$	15 15
v_j	-7	-6	0		

The nonbasic cell $(3, 1)$ with relative cost coefficient -2 is selected as the entering cell. The θ -loop is entered on the array. The minimum ratio = $\min\{7, 0, 11\} = 0$, hence this is a degenerate pivot step. The entering cell $(3, 1)$ replaces the basic cell $(2, 2)$ as the new zero valued basic cell, leading to the new basic vector given below.

Array 6.7.4

		Sink			
		1	2	3	
Source 1		7	8		
	1	1	2	3	
2				19	10
3	0			11	15

Even though the basic vector is different, the BFS and the objective value corresponding to it are exactly the same as before. This is what happens in a degenerate pivot step. Thus in a degenerate pivot step there is no change in the primal solution or objective value; but in every pivot step, whether degenerate or not, the basic vector changes by one variable.

Verify that in the basic vector in Array 6.7.3, if we had selected the entering cell to be $(1, 3)$ instead of $(3, 1)$, it would have resulted in a nondegenerate pivot step with a strict decrease in the objective value.

We now state the steps in the primal simplex algorithm for the balanced transportation problem.

6.8 The Primal Simplex Algorithm for the Balanced Transportation Problem

BEGIN

Initialization Obtain an initial primal feasible basic vector for the problem and the BFS associated with it, as discussed in Section 6.5. With this basic vector go to the first iteration.

General Iteration Find the dual basic solution and the nonbasic relative cost coefficients corresponding to the present basic vector, as discussed in Section 6.6. If all the nonbasic relative cost coefficients are ≥ 0 , the optimality criterion is satisfied, and the present primal and dual solutions are optimal to the respective problems. In this case, if some forbidden cells are still in the final basic vector with positive values for the basic variables in them in the BFS, it is an indication that there is no feasible solution for the original problem with $x_{ij} = 0$ for all forbidden cells $(i, j) \in F$. On the other hand, if all forbidden variables are zero in the final BFS when the optimality criterion is satisfied, that BFS is an optimum feasible solution for the original problem with the constraints that all forbidden variables be zero. Terminate.

If the optimality criterion is not satisfied, select a nonbasic cell with a negative relative cost coefficient as the entering cell, and perform the pivot step as in Section 6.7. With the new basic vector and the BFS associated with it, go to the next iteration.

END

Example 6.8.1: Consider the following balanced problem faced by a truck rental agency. They have some free trucks available at Detroit, Washington DC, and Denver; and need additional trucks in Orlando, Dallas, and Seattle. Data on the cost of transportation per truck (c_{ij} in coded units of money) and other data is given below.

Source city i	Sink city j			No. trucks available, a_i
	1	2	3	
Detroit 1	9	6	8	6
Washington DC 2	10	5	12	11
Denver 3	11	13	20	4
No. trucks needed, b_j	3	4	14	

Let x_{ij} denote the number of trucks sent from source city i to sink city j ; $i, j = 1$ to 3. The transportation cost is $z(x) = \sum_{i=1}^3 \sum_{j=1}^3 c_{ij} x_{ij}$. The objective is to find an x that meets the requirements at minimal cost.

To solve this problem we determine an initial primal feasible basic vector and the associated BFS as discussed in Section 6.5 using the greedy rule to select basic cells in each step. We show this basic vector in the following array. The associated basic solution and the relative

cost coefficients of nonbasic variables, computed as shown in Section 6.6, are also entered in the array.

		Sink			
		1	2	3	u_i
Source 1	3	5		6	
	9		6		8
2	$3 - \theta$	4	$4 + \theta$		12
	10	5			12
3	-7 θ	0	$4 - \theta$		
	11	13		20	20
v_j	-2	-7		0	

The optimality criterion is violated since $\bar{c}_{31} = -7 < 0$. (3, 1) is selected as the entering cell, and the θ -loop is already entered on the array. The minimum ratio is $\min\{3, 4\} = 3$. So, this is a nondegenerate pivot step, and the basic cell (2, 1) is the dropping cell. The next BFS is given in the following array.

Array 6.8.1

		Sink			
		1	2	3	u_i
Source 1	10	5		6	
	9		6		8
2	7	4	7		
	10	5		12	12
3	3	0	1		
	11	13		20	20
v_j	-9	-7		0	

Now the optimality criterion is satisfied, so the present BFS is an optimum solution. It requires the following shipments and has the minimum cost of 205 units of money.

6 trucks	from Detroit to Seattle
4 trucks	from Washington DC to Dallas
7 trucks	from Washington DC to Seattle
3 trucks	from Denver to Orlando
1 truck	from Denver to Seattle

Initiating the Primal Simplex Algorithm With a Given Primal Feasible Basic Vector

Consider a balanced transportation problem for which a primal feasible basic vector \mathcal{B} is provided. We can initiate the primal simplex algorithm with \mathcal{B} . First we have to compute the primal basic solution corresponding to \mathcal{B} . All variables x_{ij} not contained in \mathcal{B} are nonbasic variables, they are fixed at 0 in this basic solution. In the system of equality constraints in (6.3.2), when all these nonbasic variables are fixed at 0 and deleted, the remaining system can be solved by back substitution for the values of the basic variables; leading to the BFS corresponding to \mathcal{B} . Once this BFS is computed, the primal simplex algorithm can be applied beginning with it.

As an example, consider the iron ore shipping problem discussed in Section 6.5 with data shown in Array 6.5.1. Consider the basic vector $\mathcal{B} = (x_{11}, x_{12}, x_{22}, x_{23})$ for this problem. Fixing the nonbasic variables $x_{13} = x_{21} = 0$, the system of equality constraints in this problem becomes

$$\begin{array}{rcl}
 x_{11} & + x_{12} & = 800 \\
 & x_{22} & + x_{23} = 300 \\
 x_{11} & & = 400 \\
 x_{12} & + x_{22} & = 500 \\
 & x_{23} & = 200
 \end{array}$$

When this system is solved by back substitution, it leads to the BFS corresponding to \mathcal{B} given below.

		Plant		
		1	2	3
Mine 1	400	400		
	11	8		2
Mine 2		100	200	
	7	5		4

With this BFS, the transportation simplex algorithm discussed above can be initiated to solve this problem.

6.9 Marginal Analysis in the Balanced Transportation Problem

Marginal analysis deals with the rate of change in the optimum objective value, per unit change in the RHS constants (i.e., the availabilities and requirements, a_i and b_j) in the problem. In the balanced transportation model (6.1.1), the balance condition is necessary for feasibility. Since the balance condition holds originally, if only one quantity among $a_1, \dots, a_m; b_1, \dots, b_n$ changes while all the others remain fixed, the modified problem will be infeasible. So, if changes occur, at least two quantities among $a_1, \dots, a_m; b_1, \dots, b_n$ must change, and the changes must be such that the modified data also satisfies the balance condition.

We will consider three fundamental types of changes in the availability and requirement data that preserve the balance condition: (i) increased demand at sink j and a balancing increase in availability at source i (i.e., same increase in both an a_i and a b_j), (ii) increase a_p and

decrease a_i by the same amount (this shifts the supply from source i to source p), and (iii) increase b_q and decrease b_j by the same amount. In each of these categories, all the other data in the problem is assumed to remain fixed at present values. The marginal value of each type is the rate of change in the optimum objective value, per unit change of this type.

Let $\bar{x} = (\bar{x}_{ij})$, and (\bar{u}, \bar{v}) be optimum primal, dual solutions for original (6.3.2). Assume that \bar{x} is a nondegenerate BFS. Then by Results 5.5.2, 5.5.3, the marginal values associated with the three types of changes discussed above are as given below.

	Change		Marginal value
(i)	b_j and a_i increase by the same amount		$v_j + u_i$
(ii)	a_p increases and a_i decreases by the same amount		$u_p - u_i$
(iii)	b_q increases and b_j decreases by the same amount		$v_q - v_j$

Example 6.9.1: Consider the balanced transportation problem with the following data, and an optimum BFS for it marked in the following array.

	1	2	3	4	a_i	u_i
1	10 25		15 20	20 10	45	10
2		40 47		30 20	70	20
3	50 20				50	5
b_j	60	40	15	50		
v_j	15	20	10	0		

The optimum transportation cost in this problem is \$3950.

What will the rate of change in the optimum objective value be if b_2 were to increase from its current value of 40, and a corresponding change were made in a_3 to keep the problem balanced? It is $v_2 + u_3 = 20 + 5 = \$25$ per unit change.

From answers to the above questions, it is clear that if demand were to increase at any demand center, the best place to create additional supplies to satisfy that additional demand, purely from a transportation cost point of view, is source 3 (it is the source with the smallest u_i). This results in the smallest growth in the total transportation cost to meet the additional demand.

How much can the company save in transportation dollars by shifting supply from source 2 to source 3? The rate of change in the optimum transportation cost per unit shift is $u_3 - u_2 = 5 - 20 = -\$15$, or a saving of \$15. \bowtie

Thus using this marginal analysis, we can determine if the transportation costs can be reduced by shifting production from existing centers to different places. However this analysis has not taken into account any differences in production costs between centers. To determine the net savings in shifting supplies, one has to take into account the differences in production costs between places too.

6.10 What to do if There is Excess Supply or Demand

The transportation problem (6.1) in which all the constraints are equations has a feasible solution iff the total supply $\sum a_i$ is equal to the total demand $\sum b_j$.

Suppose we have a situation in which the total supply $\sum a_i$ strictly exceeds the total demand $\sum b_j$. In this case, after all the demand is met, an amount $\Delta = \sum a_i - \sum b_j$ will be left unused at the sources. So, to solve this problem, we open a new $(n+1)$ th column in the array. In row i , the cell $(i, n+1)$ represents the material left unused at source i , i.e., not shipped out of source i . Since there is no cost for not shipping

the material, we make the cost coefficients for all the cells in the new column ($n + 1$) equal to zero. Make $b_{n+1} = \Delta$, the total amount of material that will be left unused at all the sources. Solve this modified $m \times (n + 1)$ problem as a balanced transportation problem. In the optimum solution of this modified problem, basic values in the cells of column ($n + 1$) represent unused material at the sources.

As an example, consider a company with three refineries with daily production capacities for gasoline as shown in the following array. The company has contracts to sell gasoline to four wholesalers daily as shown below. Total availability = $\sum_{i=1}^3 a_i = 55$ units, while the

Wholesaler	Transportation cost from refinery to wholesaler(money/unit)				Daily availability (unit = 10^6 gal.)
	1	2	3	4	
Refinery 1	5	8	10	11	23
	2	8	6	5	16
	3	9	6	5	16
Daily requirement	15	3	14	14	

	1	2	3	4	Dummy 5	a_i
1	[15]				[8]	23
	5	8	10	11	0	
2			[14]	[2]		16
	8	6	5	8	0	
3		[3]		[12]	[1]	16
	9	6	5	8	0	
b_j	15	3	14	14	9	

total requirement is $\sum_{j=1}^4 b_j = 46$ units with an excess supply of Δ

= 9 units daily. Open a fifth column (dummy sink) with a demand for 9 units (unused supply at the refineries) leading to the balanced transportation problem given above.

An optimum solution for this problem is also entered in the array. From the basic values in the dummy column, we find that in order to meet the existing demand at minimum transportation cost, it is best to cut down production at refinery 1 to $23 - 8 = 15$ units/day, and that at refinery 3 to also $16 - 1 = 15$ units/day, while operating refinery 2 at its full capacity of 16 units/day.

Consider the other situation where total demand $\sum b_j$ exceeds total supply $\sum a_i$. In this case there is a shortage of $d = \sum b_j - \sum a_i$, and there is no way we can meet all the demand with the existing supply only.

To meet all of the existing demand, we need to identify a new source that can supply d units. In this case, if it is only required to find how to distribute the existing supply to meet as much of the demand as possible at minimum transportation cost, we open a dummy source row (the $(m + 1)$ th), cells in which represent unfulfilled demand at the sinks. Make the cost coefficients of all the cells in this dummy row equal to zero (since they represent demand not fulfilled, i.e., not shipped), and make $a_{m+1} = d$, and solve the resulting $(m + 1) \times n$ balanced transportation problem.

6.11 Exercises

6.1: A company making canned foods has vegetable farms and canning facilities in cities FC1, FC2, FC3. They store the canned produce in warehouses located in cities denoted by W1, W2, W3, W4. They supply their canned produce to customers all over the country from these warehouses. At each canning facility, as soon as a full truckload is produced, it is sent to a warehouse for storing.

The forecasted output at each canning facility (in truckloads), the storage space availability at each warehouse (in truckloads), and the trucking cost c_{ij} (money units/truckload) between the pair (FC*i*, W*j*) for $i = 1, 2, 3$ and $j = 1$ to 4 is given in the following table.

Source i	c_{ij} to sink j				Output
	W1	W2	W3	W4	
FC1	3	7	6	4	150
FC2	2	4	3	2	250
FC3	4	3	8	5	300
Storage Space	200	150	200	250	

Determine how many truckloads to send from each canning facility to each warehouse to minimize the trucking cost incurred.

6.2: Production allocation to plants: A company makes and markets products P_1 to P_4 . They have 5 plants in the country to make these products. Each plant can make one or more of the products, but the manufacturing cost of each product varies from plant to plant. For $i = 1$ to 4 , $j = 1$ to 5 , let

c_{ij} = Production cost (in money units/unit, money unit is a coded unit) of producing P_i at plant j .

b_j = Total production capacity (units/year) of Plant j . Each plant can make any combination of products that they can make, upto their production capacity.

a_i = Demand for Product i (units/year).

The company would like to meet the demand for all the products. If they are unable to make any product to the full demand level, they can subcontract the unmet demand to a subsidiary. Here is the data. Blank entries indicate that the plant cannot make that product.

P_i	c_{ij} at Plant j					a_i
	1	2	3	4	5	
P_1		50	70	65		1500
P_2	40	51		65	50	1000
P_3	30		45		40	5000
P_4	12	15	30		20	10,000
b_j	600	400	600	1200	1000	

How many units of each product should they make at each of their plants in order to minimize their total production cost? What portion of the demand would this solution leave for the subsidiary?

6.3: Solve the following balanced transportation problem using $\{(1, 2), (1, 5), (2, 4), (2, 5), (3, 1), (3, 3), (4, 2), (4, 3)\}$ as the initial basic set of cells. c_{ij} is the shipping cost/unit from source i to market j .

Suppose the requirement at market 5, b_5 , increases from its current value of 19. What is the best source, say source p , at which to create additional supplies to meet this extra demand? Explain the reasons for the choice of p clearly.

Source i	1	2	c_{ij}	4	5	$a_i = \text{supply (units)}$
1	12	6	12	8	8	20
2	15	9	14	8	10	21
3	12	9	13	10	9	8
4	11	7	11	9	12	13
$b_j = \text{requirement}$	6	14	11	12	19	

Increase both b_5 and a_p by δ , and obtain the new optimum solution as a function of δ , and find its optimality range.

6.4: Consider the balanced transportation problem with $m = 4$ sources, $n = 6$ markets, $a = (a_i) = (13, 31, 51, 21)$, $b = (b_j) = (17, 4, 16, 13, 54, 12)$; where a_i, b_j are the amounts available to be shipped out of source i , required at market j respectively. c_{ij} = cost of transporting from source i to market j /unit, and

$$c = (c_{ij}) = \begin{pmatrix} 10 & 2 & 9 & 1 & 11 & 12 \\ 12 & 9 & 3 & 11 & 4 & 15 \\ 3 & 7 & 10 & 9 & 6 & 6 \\ 12 & 9 & 11 & 3 & 5 & 18 \end{pmatrix}$$

Find an initial BFS to this problem using Vogel's method. Solve the problem beginning with that BFS.

Index

For each index entry we provide the section number where it is defined or discussed first, and number of some sections where it appears prominently.

BFS 6.5

- Degenerate 6.5
- Nondegenerate 6.5
- Balance condition 6.1
- Balanced transportation problem 6.1
- Initial BFS for 6.5
- Primal simplex algo. for 6.4, 6.8

C S Optimality conditions 6.3

- Dropping cell 6.7
- Donor cell 6.7
- Dual basic sol. 6.6

Eligible cells 6.7

- Entering cell 6.7

Forbidden cells 6.4

Greedy choice rule 6.5

Marginal analysis 6.9

Optimality criterion 6.7

Pivot step 6.7

Recipient cells 6.7

- Redundency 6.3

θ -loop 6.7

- Trial and error method 6.7

VAM 6.5

- Vogel's Approx. method 6.5
- Vogel's choice rule 6.5