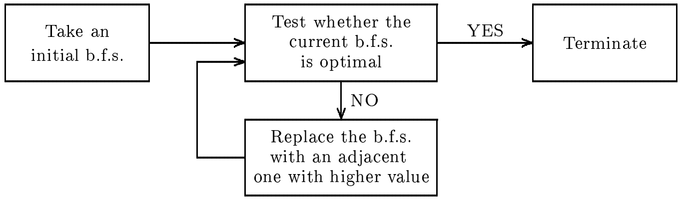
3. THE SIMPLEX ALGORITHM

3.1 Introduction

We know that, if a linear programming problem has a finite optimal solution, it has an optimal solution at a basic feasible solution (b.f.s.). The simplex algorithm is a systematic method of searching through the basic feasible solutions in such a way that, at each iteration, it moves to a better solution in the sense of having higher value of the objective function (or 'payoff' as it is often called); this would be in the case of a maximization problem, we would want a lower value if we were minimizing. The better solution is an 'adjacent' one which is reached by moving along an edge of the feasible set. As there are only a finite number of basic feasible solutions (bounded above by $\binom{n}{m}$) this process will terminate in a finite number of steps. Schematically, the algorithm proceeds as follows:



We will illustrate how this goes by considering the example of the previous chapter. In this case we will take as the initial b.f.s. the point A in the diagram corresponding to the values

$$x_1 = 0,\ x_2 = 0,\ z_1 = 4,\ z_2 = 6.$$

In general it may not be obvious how to determine an initial b.f.s. (we will come to that question later when we introduce the two-phase algorithm) but for any example which we can write with inequality constraints $A\boldsymbol{x} \leqslant \boldsymbol{b}$ where $\boldsymbol{b} \geqslant 0$, then when we put it in equality form $A\boldsymbol{x} + \boldsymbol{z} = \boldsymbol{b}$ with slack variables $\boldsymbol{z} \geqslant 0$ we can obtain an initial b.f.s. by setting $\boldsymbol{x} = 0$

and $\boldsymbol{z} = \boldsymbol{b} \geqslant 0$ (as we have done here). The basic variables at A are z_1 and z_2 and the non-basic variables are x_1 and x_2. Let us write the problem as

$$z_1 = 4 - 2x_1 - x_2 \qquad (1)$$

$$z_2 = 6 - 2x_1 - 3x_2 \qquad (2)$$

$$\text{Payoff} \quad f = 3x_1 + 2x_2, \qquad (3)$$

with the feasibility conditions $x_1 \geqslant 0$, $x_2 \geqslant 0$, $z_1 \geqslant 0$ and $z_2 \geqslant 0$. This representation of the problem expresses the basic variables z_1, z_2, and the objective function f, parametrically in terms of the non-basic variables x_1 and x_2. At this b.f.s., the non-basic variables $x_1 = x_2 = 0$, and we can read off the values of the basic variables, $z_1 = 4$, $z_2 = 6$, and of the objective function, $f = 0$, at A.

It is clear that this b.f.s. is not optimal because we can increase either x_1 or x_2 from 0 (by a sufficiently small amount so as to keep z_1 and z_2 non-negative) and thus strictly increase the objective function this is because the coefficients of both x_1 and x_2 in the expression for the objective function are positive. The method of the simplex algorithm is to move from this b.f.s. by increasing exactly one of the non-basic variables from zero (keeping the others equal to zero) by as large an amount as possible, while retaining feasibility (that is, the current basic variables $\geqslant 0$). In our case, since the coefficient of x_1 in the objective function is larger than that of x_2 let us choose x_1 to increase, as it will give the larger rate of increase of the payoff; this is a useful rule-of-thumb but it is not necessary to the algorithm – choosing any non-basic variable with a positive coefficient in the objective function would do.

Now as we increase x_1 from zero, keeping $x_2 = 0$, the current basic variables are $z_1 = 4 - 2x_1$, $z_2 = 6 - 2x_1$, and we see that we are moving along the edge AB of the feasible set; note that z_1 becomes negative as x_1 passes through the value 2 while z_2 does not change sign until $x_1 = 3$, so the largest value of x_1 that we may take and remain within the feasible set is $x_1 = 2$. This gives a new b.f.s. $x_1 = 2$, $z_2 = 2$, $x_2 = z_1 = 0$, which is the point B in the diagram. This new b.f.s. has x_2, z_1 as non-basic variables and x_1, z_2 basic.

We now rewrite the problem parametrically in terms of the new non-basic variables x_2 and z_1. Divide (1) by 2 and rearrange to get x_1 in (4) below, then substitute x_1 into

(2) (which is the same as subtracting (1) from (2) and rearranging to get the expression for z_2 in (5), and substitute for x_1 in (3), which corresponds to adding $\frac{3}{2}$ times (1) to (3) to get

$$x_1 = 2 - \tfrac{1}{2}z_1 - \tfrac{1}{2}x_2 \tag{4}$$

$$z_2 = 2 + \ z_1 - 2x_2 \tag{5}$$

$$\text{Payoff} \quad f = 6 - \tfrac{3}{2}z_1 + \tfrac{1}{2}x_2. \tag{6}$$

At this stage we have gone from z_1, z_2 as the basic variables to x_1 and z_2 basic (the terminology is 'x_1 has entered the basis' and 'z_1 has left the basis'). We are now at a better b.f.s. since the value of the objective function is now $f = 6$, and we start the loop again.

Test for optimality: there is a coefficient of a non-basic variable, x_2, in the expression for the payoff (6) which is positive, so that the current b.f.s. is not optimal. We can increase x_2 from 0 ('x_2 enters the basis') to get a higher value of the objective function.

Choice of variable to leave the basis: the first of the two basic variables to hit 0 as x_2 increases (and z_1 is kept fixed at 0) is z_2 (when $x_2 = 1$).

The new b.f.s. has x_1, x_2 basic and z_1, z_2 non-basic, which corresponds to the point D in the diagram. Again, we express the problem in terms of the non-basic variables. Divide (5) by 2 and rearrange to get (8). Take $\frac{1}{4} \times (5)$ from (4) (to eliminate x_2, or equivalently substitute for x_2 from (8) into (4)) and rearrange to get (7); finally, add $\frac{1}{4} \times (5)$ to (6) to get (9) (again, to eliminate x_2), and we have the problem represented as

$$x_1 = \ \tfrac{3}{2} - \tfrac{3}{4}z_1 + \tfrac{1}{4}z_2 \tag{7}$$

$$x_2 = \ 1 + \tfrac{1}{2}z_1 - \tfrac{1}{2}z_2 \tag{8}$$

$$\text{Payoff} \quad f = \tfrac{13}{2} - \tfrac{5}{4}z_1 - \tfrac{1}{4}z_2. \tag{9}$$

We see that this b.f.s. with $x_1 = \frac{3}{2}$, $x_2 = 1$, $z_1 = z_2 = 0$ and the objective function $f = \frac{13}{2}$ is optimal because in (9) the coefficients of both the non-basic variables are negative so increasing either from 0 would give a lower value of the payoff.

3.2 The simplex tableau

The discussion in the previous section contains all the essential elements of the simplex algorithm, but the somewhat adhoc transition through the steps from b.f.s. to b.f.s. that we gave needs to be systematized and formalized to cope with larger problems. The way this is done is by writing the problem in the form of the simplex tableau, which is a form of bookkeeping for the procedures we have used. We will work through the problem using the notation of the simplex tableau. Firstly the problem in equality form is:

$$\begin{aligned} 2x_1 + x_2 + z_1 \qquad &= 4 \\ 2x_1 + 3x_2 \qquad + z_2 &= 6 \\ \text{Payoff} \qquad 3x_1 + 2x_2 \qquad\qquad &= f, \end{aligned}$$

with, of course, the conditions x_1, x_2, z_1, $z_2 \geqslant 0$ always assumed. This is written in the form

	x_1	x_2	z_1 *	z_2 *	
z_1	2	1	1	0	4
z_2	2	3	0	1	6
Payoff	3	2	0	0	0

with the * to indicate the basic variables; the entries z_1 and z_2 on the left-hand side are to remind you to which basic variable each row refers, including them is a bit of overkill but a little redundancy does not hurt if it helps to make it clearer. This table is essentially equations (1)-(3). In general if the constraints are in the form $A\boldsymbol{x} = \boldsymbol{b}$ where A is an $m \times n$ matrix , $\boldsymbol{x} = (x_1, \ldots, x_n)^\top$ and $\boldsymbol{b} = (b_1, \ldots, b_m)^\top$, the tableau looks like

	x_1	x_2	$\cdots$	x_n	
	$a_{1,1}$	$a_{1,2}$	$\cdots$	$a_{1,n}$	b_1
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
	$a_{m,1}$	$a_{m,2}$	$\cdots$	$a_{m,n}$	b_m
Payoff	$a_{0,1}$	$a_{0,2}$	$\cdots$	$a_{0,n}$	$a_{0,0}$

but within the box (rows 1 to m, columns 1 to n) there is the identity matrix (as there is under the basic variables in our example).

Step 1 Test for optimality. Look along the payoff row, if all the entries are $\leqslant 0$ (apart from the bottom right-hand entry $a_{0,0}$ which is $-f$, where f is the value of the objective function at the current b.f.s.), the current b.f.s. is optimal (we need all entries $\geqslant 0$ if the problem is a minimization). Otherwise, we proceed as follows.

Step 2 Identify the non-basic variable to enter the basis; this is called 'choosing the **pivot column**'. We look along the payoff row $(a_{0,1}, \ldots, a_{0,n})$ and choose a positive element (a negative element if we are minimizing); our rule-of-thumb is to choose the largest positive element $a_{0,j}$ indicated by $\uparrow$ below to give the steepest ascent, (but this is no guarantee that we will minimize the number of iterations of the algorithm). The choice of pivot column is to ensure that the objective function is higher (lower for minimization) at the next b.f.s..

Step 3 Choose the basic variable to leave the basis; this is called 'choosing the **pivot row**'. Within the pivot column j we choose the row element $a_{i,j}$ which minimizes the ratio $b_k/a_{k,j}$ over those rows k for which $a_{k,j} > 0$. If all $a_{k,j} \leqslant 0$ for all rows $k = 1, \ldots, m$ then the problem is unbounded above (the maximum of the problem $= \infty$) because the variable x_j may be increased indefinitely without making any of the basic variables go negative. The element $a_{i,j}$ chosen in this way is the **pivot element** and boxed in the tableau below. The variable x_j enters the basis and the variable that leaves the basis may be determined from the ith row label; it corresponds to the basic variable which has a 1 in the ith row in the identity matrix below the basic variables. This choice of pivot row is made to ensure that at the next step the basic solution is feasible.

			*	*	
	x_1	x_2	z_1	z_2	
z_1	$\boxed{2}$	1	1	0	4
z_2	2	3	0	1	6
Payoff	3	2	0	0	0
	$\uparrow$				

Step 4 We need to perform the **pivot** operation, when the pivot element is $a_{i,j}$. This is the process of rewriting the problem in terms of the new basic variables, as we did going from the representation (1)-(3) to the representation (4)-(6). The procedure is to

(i) divide row i by $a_{i,j}$;

(ii) add $-(a_{k,j}/a_{i,j}) \times$ row(i) to row k for each $k \neq i$ (including payoff row).

Notice that this operation preserves the property that the identity matrix is always embedded in the tableau under the basic variables (possibly after a rearrangement of the columns).

In the case of our example we obtain (the left-hand) tableau below, which should be compared with equations (4)-(6). At this stage we return to Step 1 and repeat the cycle. In the right-hand tableau we identify that the current b.f.s. is not optimal, that x_2 should enter the basis and that z_2 should leave:

	*			*	
	x_1	x_2	z_1	z_2	
x_1	1	$\frac{1}{2}$	$\frac{1}{2}$	0	2
z_2	0	2	-1	1	2
Payoff	0	$\frac{1}{2}$	$-\frac{3}{2}$	0	-6

	*			*	
	x_1	x_2	z_1	z_2	
x_1	1	$\frac{1}{2}$	$\frac{1}{2}$	0	2
z_2	0	$\boxed{2}$	-1	1	2
Payoff	0	$\frac{1}{2}$	$-\frac{3}{2}$	0	-6
		$\uparrow$			

Performing the pivot operation we obtain

	*	*			
	x_1	x_2	z_1	z_2	
x_1	1	0	$\frac{3}{4}$	$-\frac{1}{4}$	$\frac{3}{2}$
x_2	0	1	$-\frac{1}{2}$	$\frac{1}{2}$	1
Payoff	0	0	$-\frac{5}{4}$	$-\frac{1}{4}$	$-\frac{13}{2}$

which corresponds to equations (7)-(9), and which is optimal from Step 1 above. We read off the optimal solution $x_1 = \frac{3}{2}$, $x_2 = 1$, $z_1 = z_2 = 0$, and the optimal value of the objective function is $\frac{13}{2}$, obtained as minus the bottom right-hand entry of the tableau.

Remarks

1. By comparing the various simplex tableaux for this problem with the enumeration of the solutions to the dual problem in the previous chapter, it may be noted that, at each stage, in the payoff row of the tableau the dual variables at the relevant b.f.s. may be identified. This will be the case when the constraints in the primal problem are inequalities, however the order in which the dual variables appear, and their sign, may have to be deciphered, depending on the original formulation of the problem. In the typical formulation, as we have here, where we start from the b.f.s. where the slack variables are the basic variables then the optimal dual

variables will be minus the entries under the slack variables in the payoff row of the final tableau (see below and later). In this example, from the final tableau we can see that the optimal solution to the dual is $\lambda_1 = \frac{5}{4}$, $\lambda_2 = \frac{1}{4}$, and $\lambda_3 = v_1 = 0$, $\lambda_4 = v_2 = 0$. This can provide a useful check on your arithmetic as we must have, at the optimal solution, that $\boldsymbol{b}^\top \boldsymbol{\lambda} = \boldsymbol{c}^\top \boldsymbol{x}$, and here $4\left(\frac{5}{4}\right) + 6\left(\frac{1}{4}\right) = \frac{13}{2}$.

2. Each step of the simplex algorithm just involves elementary operations on the rows of the tableau: dividing through a row by a constant or adding/subtracting a multiple of a row to/from another. If we compare the initial and final tableaux

	x_1	x_2	$\overset{*}{z_1}$	$\overset{*}{z_2}$	
z_1	2	1	1	0	4
z_2	2	3	0	1	6
Payoff	3	2	0	0	0

	$\overset{*}{x_1}$	$\overset{*}{x_2}$	z_1	z_2	
x_1	1	0	$\frac{3}{4}$	$-\frac{1}{4}$	$\frac{3}{2}$
x_2	0	1	$-\frac{1}{2}$	$\frac{1}{2}$	1
Payoff	0	0	$-\frac{5}{4}$	$-\frac{1}{4}$	$-\frac{13}{2}$

of the problem we have looked at, because the identity matrix is initially below the slack variables z_1, z_2, we can identify easily the total effect of these operations. By looking under z_1 and z_2 in the final tableau, we see that the first two rows have been pre-multiplied by the matrix

$$\begin{pmatrix} \frac{3}{4} & -\frac{1}{4} \\ -\frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

and from considering the entries under z_1, z_2 in the payoff row we see that the final payoff row has been formed by adding $(-\frac{5}{4}\times$ (row 1) $-\frac{1}{4}\times$ (row 2)) to the initial payoff row. Suppose now that the initial right-hand sides in the constraints are changed to $4+\epsilon_1$ and $6+\epsilon_2$, then the new right-hand sides in the final tableau will be determined by

$$\begin{pmatrix} \frac{3}{4} & -\frac{1}{4} \\ -\frac{1}{2} & \frac{1}{2} \end{pmatrix}\begin{pmatrix} 4+\epsilon_1 \\ 6+\epsilon_2 \end{pmatrix} = \begin{pmatrix} \frac{3}{2}+\frac{3}{4}\epsilon_1-\frac{1}{4}\epsilon_2 \\ 1-\frac{1}{2}\epsilon_1+\frac{1}{2}\epsilon_2 \end{pmatrix}$$

and the new final tableau is

	$\overset{*}{x_1}$	$\overset{*}{x_2}$	z_1	z_2	
x_1	1	0	$\frac{3}{4}$	$-\frac{1}{4}$	$\frac{3}{2}+\frac{3}{4}\epsilon_1-\frac{1}{4}\epsilon_2$
x_2	0	1	$-\frac{1}{2}$	$\frac{1}{2}$	$1-\frac{1}{2}\epsilon_1+\frac{1}{2}\epsilon_2$
Payoff	0	0	$-\frac{5}{4}$	$-\frac{1}{4}$	$-\frac{13}{2}-\frac{5}{4}\epsilon_1-\frac{1}{4}\epsilon_2$

One thing to note is that (apart from the changed value of the objective function) the payoff row is unchanged so the entries are still $\leqslant 0$; this tableau will be optimal for the revised problem so long as the new right-hand sides are $\geqslant 0$, that is the basic solution given by this tableau is feasible for the revised problem. So long as

$$\frac{3}{2}+\frac{3}{4}\epsilon_1-\frac{1}{4}\epsilon_2 \geqslant 0, \quad \text{and} \quad 1-\frac{1}{2}\epsilon_1+\frac{1}{2}\epsilon_2 \geqslant 0,$$

the tableau is optimal; this will be true irrespective of whether the application of the simplex algorithm has taken you outside the feasible set in the intervening steps. This is because of the three necessary and sufficient conditions for optimality we had in the previous chapter. The other thing to note that the change in the optimum value is $\frac{5}{4}\epsilon_1 + \frac{1}{4}\epsilon_2$, which is consistent with the shadow price interpretation of the optimal dual variables $\lambda_1 = \frac{5}{4}$ and $\lambda_2 = \frac{1}{4}$, discussed in Chapter 1.

3.3 The two-phase method

There remains the question of how the simplex algorithm should be started if there is no obvious b.f.s from which to start the algorithm. The answer to this, somewhat paradoxically, is that the simplex algorithm itself may be used to generate an initial b.f.s.. This is done by setting up a linear programming problem (the Phase I problem) for which there is an obvious initial b.f.s. and which yields an optimal solution which will be an initial b.f.s. for the original problem (if one exists). Phase II is then the application of the simplex algorithm to the original problem starting from this initial b.f.s.. We will consider the example:

$$\begin{aligned} \text{maximize} \quad & x_1 - 3x_2 + 5x_3 \\ \text{subject to} \quad & x_1 + x_2 + x_3 \leqslant 30 \\ & -x_2 + 2x_3 = 20 \\ & -x_1 + 2x_2 + x_3 \geqslant 40 \\ & x_1,\ x_2,\ x_3 \geqslant 0. \end{aligned}$$

Add slack variables to the first and third constraints to put it in equation form and we obtain

$$
\begin{array}{rl}
\text{maximize} & x_1 - 3x_2 + 5x_3 \\
\text{subject to} & x_1 + \ x_2 + \ x_3 + z_1 \qquad = 30 \\
& \quad - \ x_2 + 2x_3 \qquad\qquad = 20 \\
& -x_1 + 2x_2 + \ x_3 \quad - z_2 = 40 \\
& \qquad x_1,\ x_2,\ x_3,\ z_1,\ z_2 \geqslant 0.
\end{array}
$$

In the previous section we could start from the feasible solution with just the slack variables as basic; this cannot work here, for a start there is no slack variable appearing in the second constraint and, if we take $x_1 = x_2 = x_3 = 0$, then we obtain $z_2 = -40$ which is not feasible. The answer to the problem is to introduce what are known as **artificial** variables y_1, y_2 into the constraints

$$
\begin{array}{rcl}
x_1 + \ x_2 + \ x_3 + z_1 & & = 30 \\
- \ x_2 + 2x_3 & + y_1 & = 20 \\
-x_1 + 2x_2 + \ x_3 \quad - z_2 & + y_2 & = 40 \\
x_1,\ x_2,\ x_3,\ z_1,\ z_2, y_1,\ y_2 \geqslant 0, & &
\end{array}
$$

and then minimize $y_1 + y_2$ subject to these constraints, starting from the b.f.s. to this enlarged problem obtained by taking z_1, y_1 and y_2 basic, that is

$$x_1 = x_2 = x_3 = z_2 = 0,\ z_1 = 30,\ y_1 = 20,\ y_2 = 40;$$

if there is a solution to this Phase I problem with $y_1 = y_2 = 0$ (y_1, y_2 non-basic) then there is a b.f.s. to the original problem from which we can initiate the simplex algorithm for the original problem. If the optimal solution to the Phase I problem has optimal value which is strictly positive then there is no feasible solution to the original problem. We set this up in tableau form, and as a useful piece of bookkeeping during the Phase I algorithm, we include a row corresponding to the Phase II objective function (the objective function for the original problem). This avoids having to calculate the Phase II objective function in

terms of the non-basic variables at the end of Phase I.

			*		*	*	
	x_1	x_2	x_3	z_1	z_2	y_1	y_2	
	1	1	1	1	0	0	0	30
	0	−1	2	0	0	1	0	20
	−1	2	1	0	−1	0	1	40
Phase II	1	−3	5	0	0	0	0	0
Phase I	0	0	0	0	0	1	1	0

We have written the Phase I objective function as $y_1 + y_2$ so the problem is a minimization; if you feel more comfortable maximizing, you may replace it by $-y_1 - y_2$.

Preliminary Step Notice first of all that the tableau above is not yet in the correct form to apply the simplex algorithm. The payoff row(s) should express the objective functions in terms of the non-basic variables so we need to subtract the second and third rows from the Phase I row to eliminate y_1 and y_2.

				*		*	*	
	x_1	x_2	x_3	z_1	z_2	y_1	y_2	
z_1	1	1	1	1	0	0	0	30
y_1	0	−1	$\boxed{2}$	0	0	1	0	20
y_2	−1	2	1	0	−1	0	1	40
Phase II	1	−3	5	0	0	0	0	0
Phase I	1	−1	−3	0	1	0	0	−60
			↑					

All we are doing here is expressing

$$y_1 + y_2 = 60 + x_1 - x_2 - 3x_3 + z_2$$

from the second and third constraint. We are now ready to roll and solve this Phase I problem (remembering that we have set it up as a minimization and so we choose a negative element in the payoff row).

Phase I We take two pivots, first introducing x_3 and dropping y_1 (pivot element marked above) to get:

	x_1	x_2	x_3	z_1	z_2	y_1	y_2	
			*	*			*	
z_1	1	$\frac{3}{2}$	0	1	0	$-\frac{1}{2}$	0	20
x_3	0	$-\frac{1}{2}$	1	0	0	$\frac{1}{2}$	0	10
y_2	-1	$\boxed{\frac{5}{2}}$	0	0	-1	$-\frac{1}{2}$	1	30
Phase II	1	$-\frac{1}{2}$	0	0	0	$-\frac{5}{2}$	0	-50
Phase I	1	$-\frac{5}{2}$	0	0	1	$\frac{3}{2}$	0	-30
		$\uparrow$						

and then then pivoting on $\frac{5}{2}$ to introduce x_2 and drop y_2:

	x_1	x_2	x_3	z_1	z_2	y_1	y_2	
		*	*	*				
z_1	$\frac{8}{5}$	0	0	1	$\frac{3}{5}$	$-\frac{1}{5}$	$-\frac{3}{5}$	2
x_3	$-\frac{1}{5}$	0	1	0	$-\frac{1}{5}$	$\frac{2}{5}$	$\frac{1}{5}$	16
x_2	$-\frac{2}{5}$	1	0	0	$-\frac{2}{5}$	$-\frac{1}{5}$	$\frac{2}{5}$	12
Phase II	$\frac{4}{5}$	0	0	0	$-\frac{1}{5}$	$-\frac{13}{5}$	$\frac{1}{5}$	-44
Phase I	0	0	0	0	0	1	1	0

We are now at an optimal solution to Phase I; all the entries in the payoff row are $\geqslant 0$ (we are minimizing) and the optimal value is 0 so $y_1 = y_2 = 0$ and we have a b.f.s. of the original problem. If we had obtained an optimal solution to Phase I and the value was > 0, so that at least one of $y_1 > 0$ or $y_2 > 0$, then there is no feasible solution to the original problem.

Phase II We may now drop the Phase I row and the columns corresponding to the artificial variables y_1 and y_2,

	x_1	x_2	x_3	z_1	z_2	
		*	*	*		
z_1	$\boxed{\frac{8}{5}}$	0	0	1	$\frac{3}{5}$	2
x_3	$-\frac{1}{5}$	0	1	0	$-\frac{1}{5}$	16
x_2	$-\frac{2}{5}$	1	0	0	$-\frac{2}{5}$	12
Phase II	$\frac{4}{5}$	0	0	0	$-\frac{1}{5}$	-44
	$\uparrow$					

and we proceed to apply the simplex algorithm to this reduced tableau, introducing x_1 and dropping z_1 from the basis (remember we are now maximizing, so we choose a positive element in the payoff row).

	x_1	x_2	x_3	z_1	z_2	
	*	*	*			
x_1	1	0	0	$\frac{5}{8}$	$\frac{3}{8}$	$\frac{5}{4}$
x_3	0	0	1	$\frac{1}{8}$	$-\frac{1}{8}$	$\frac{65}{4}$
x_2	0	1	0	$\frac{1}{4}$	$-\frac{1}{4}$	$\frac{25}{2}$
Phase II	0	0	0	$-\frac{1}{2}$	$-\frac{1}{2}$	-45

This tableau is now optimal for the original problem with the optimal value being 45 occurring when $x_1 = \frac{5}{4}$, $x_2 = \frac{25}{2}$ and $x_3 = \frac{65}{4}$.

Remark

If the dual variables corresponding to the three constraints for the problem of this section are λ_1, λ_2 and λ_3 then we must have $\lambda_1 \geqslant 0$, $\lambda_3 \leqslant 0$ while λ_2 is unconstrained in sign. We can read off the optimal $\lambda_1 = \frac{1}{2}$ and $\lambda_3 = -\frac{1}{2}$ from under the slack variables in the final tableau and you may like to think about how you would see that $\lambda_2 = \frac{5}{2}$.

3.4 Extreme points and basic feasible solutions

We will now fill in some of the details and proofs that were promised earlier in the course. We will discuss the case of the standard linear programming problem

$$\text{maximize} \quad \boldsymbol{c}^\top \boldsymbol{x} \quad \text{subject to} \quad A\boldsymbol{x} = \boldsymbol{b},\ \boldsymbol{x} \geqslant 0.$$

Recall that we are making the following

Assumptions

(i) A is an $m \times n$ matrix, $n > m$, with rank $(A) = m$;

(ii) any m columns of A are linearly independent; and

(iii) the problem is **non-degenerate**: $\boldsymbol{b}$ cannot be expressed as a linear combination of fewer than m columns of A (so that any b.f.s. has exactly m non-zero entries).

Degeneracy The importance of non-degeneracy for the simplex algorithm is the following: if, when we introduce a new variable into the basis, it has a positive value in the new b.f.s., because we have a strictly positive coefficient (when maximizing) for this variable in the expression for the objective function (the bottom row of the tableau), we get a strictly positive improvement in the objective function at the new b.f.s.. Hence, if the problem is non-degenerate, during each iteration the simplex algorithm will move to a strictly better b.f.s; there are only a finite number of basic feasible solutions and so the algorithm terminates after a finite number of steps. If the problem is degenerate and a basic variable is zero, then it is possible to go through an iteration and change the basis and get no improvement in the objective function – it is theoretically possible to cycle through a sequence of basic feasible solutions and return to the starting point, so the algorithm gets stuck. While this is a theoretical possibility, the situation is easily resolved, as even such a crude approach as randomizing the choice of pivot column will ensure that the algorithm breaks out of the loop.

We turn to the proofs of two results postponed from Chapter 2.

Theorem 2.2 *A point $\boldsymbol{x}$ is a basic feasible solution of $A\boldsymbol{x} = \boldsymbol{b}$ if and only if it is an extreme point of the feasible set $\{\boldsymbol{x} : \ A\boldsymbol{x} = \boldsymbol{b},\ \boldsymbol{x} \geqslant 0\}$ for the linear programming problem.*

Proof. Let $X_{\boldsymbol{b}} = \{\boldsymbol{x} : \ A\boldsymbol{x} = \boldsymbol{b},\ \boldsymbol{x} \geqslant 0\}$. Suppose that $\boldsymbol{x}$ is a b.f.s., so that $\boldsymbol{x} \in X_{\boldsymbol{b}}$ and $\boldsymbol{x}$ has exactly m non-zero entries. Suppose that

$$\boldsymbol{x} = \theta \boldsymbol{y} + (1-\theta)\boldsymbol{z}, \quad \boldsymbol{y},\ \boldsymbol{z} \in X_{\boldsymbol{b}}, \quad 0 < \theta < 1.$$

Since $y_i \geqslant 0$ and $z_i \geqslant 0$, $x_i = \theta y_i + (1-\theta) z_i$ implies that if $x_i = 0$ then necessarily $y_i = z_i = 0$, so that, by Assumption (iii), $\boldsymbol{y}$ and $\boldsymbol{z}$ are basic feasible solutions each with exactly m non-zero entries, which occur in the same positions. Since $A\boldsymbol{y} = \boldsymbol{b} = A\boldsymbol{z}$ we have $A(\boldsymbol{y} - \boldsymbol{z}) = 0$; but this give a linear combination of m columns of A equal to zero and, using Assumption (ii), we deduce that $\boldsymbol{y} = \boldsymbol{z}$ so that $\boldsymbol{x}$ is extreme in $X_{\boldsymbol{b}}$.

Conversely, suppose $\boldsymbol{x} \in X_{\boldsymbol{b}}$ is not a b.f.s., so that it has exactly r non-zero entries $x_{i_1}, \ldots, x_{i_r} > 0$ for some $r > m$; by Assumption (i), the corresponding columns of A, $\boldsymbol{a}_{i_1}, \ldots, \boldsymbol{a}_{i_r}$, must be linearly dependent so there exist $(y_{i_1}, \ldots, y_{i_r}) \neq 0$ with $\sum_{j=1}^{r} y_{i_j} \boldsymbol{a}_{i_j} = 0$. We may extend $(y_{i_1}, \ldots, y_{i_r})$ to the point $\boldsymbol{y} = (y_1, \ldots, y_n)^{\top}$ (by setting the other entries

equal to 0), to get $A\boldsymbol{y} = 0$ and $\boldsymbol{y} \neq 0$. It follows that, for $\epsilon > 0$ sufficiently small, $\boldsymbol{x} \pm \epsilon \boldsymbol{y} \geqslant 0$ and $A(\boldsymbol{x} \pm \epsilon \boldsymbol{y}) = \boldsymbol{b}$ so that $\boldsymbol{x} \pm \epsilon \boldsymbol{y} \in X_{\boldsymbol{b}}$ and

$$\boldsymbol{x} = \frac{1}{2}(\boldsymbol{x} + \epsilon \boldsymbol{y}) + \frac{1}{2}(\boldsymbol{x} - \epsilon \boldsymbol{y}),$$

showing that $\boldsymbol{x}$ is not extreme. □

Theorem 2.1 *If the linear programming problem, maximize $\boldsymbol{c}^{\top}\boldsymbol{x}$ subject to $A\boldsymbol{x} = \boldsymbol{b}$, $\boldsymbol{x} \geqslant 0$, has a finite optimum then there is an optimal solution which is an extreme point of the feasible set.*

Proof. Let $\boldsymbol{x}$ be an optimal solution; firstly, suppose that $\boldsymbol{x}$ has exactly m non-zero entries then $\boldsymbol{x}$ is a b.f.s. and, by the previous result, it is necessarily an extreme point of $X_{\boldsymbol{b}}$. So suppose that $\boldsymbol{x}$ has r non-zero entries where $r > m$ and that $\boldsymbol{x}$ is not extreme in $X_{\boldsymbol{b}}$; that is, $\boldsymbol{x} = \theta\boldsymbol{y} + (1-\theta)\boldsymbol{z}$, for $\boldsymbol{y},\ \boldsymbol{z} \in X_{\boldsymbol{b}}$, $\boldsymbol{y} \neq \boldsymbol{z}$, $0 < \theta < 1$. We will show that we can find an optimal solution with fewer than r non-zero entries; then we can repeat the argument until we get down to an optimal solution with m non-zero entries which is extreme. Since $\boldsymbol{c}^{\top}\boldsymbol{x} \geqslant \boldsymbol{c}^{\top}\boldsymbol{y}$, $\boldsymbol{c}^{\top}\boldsymbol{x} \geqslant \boldsymbol{c}^{\top}\boldsymbol{z}$ and $\boldsymbol{c}^{\top}\boldsymbol{x} = \theta\boldsymbol{c}^{\top}\boldsymbol{y} + (1-\theta)\boldsymbol{c}^{\top}\boldsymbol{z}$ we must have $\boldsymbol{c}^{\top}\boldsymbol{x} = \boldsymbol{c}^{\top}\boldsymbol{y} = \boldsymbol{c}^{\top}\boldsymbol{z}$, so that $\boldsymbol{y}$ and $\boldsymbol{z}$ are also optimal. As in the previous proof, $x_i = 0$ implies that $y_i = 0$ and $z_i = 0$, so that $\boldsymbol{y}$ and $\boldsymbol{z}$ have r, or fewer, non-zero entries which occur in the same positions as those in $\boldsymbol{x}$. If either $\boldsymbol{y}$ or $\boldsymbol{z}$ have fewer than r non-zero entries we are done, otherwise we can choose a $\theta' \in \mathbb{R}$ so that $\boldsymbol{x}' = \theta'\boldsymbol{y} + (1-\theta')\boldsymbol{z} \geqslant 0$ and $\boldsymbol{x}'$ has $r-1$, or fewer, non-zero entries. To do this, write $\boldsymbol{x}' = \boldsymbol{z} + \theta'(\boldsymbol{y} - \boldsymbol{z})$; since $\boldsymbol{y} \neq \boldsymbol{z}$ we can move θ' from 0 (by the smallest amount possible, either in the positive or negative direction) until $\boldsymbol{x}'$ has $r-1$, or fewer, non-zero components. The point $\boldsymbol{x}'$ will be again be optimal. This completes the argument. □

Note In fact this theorem can be strengthened to the case of maximizing a convex function $f(\boldsymbol{x})$ over $\boldsymbol{x} \in X \subseteq \mathbb{R}^n$, a compact convex set, to show that the maximum occurs at an extreme point of the set X. It may be shown that any point $\boldsymbol{x} \in X$ may be written as a convex combination $\boldsymbol{x} = \sum_{i=1}^{k} \theta_i \boldsymbol{x}_i$, $\theta_i \geqslant 0$, $\sum_{i=1}^{k} \theta_i = 1$, of extreme points $\{\boldsymbol{x}_i\}$ of X. Then

$$f(\boldsymbol{x}) \leqslant \sum_{i=1}^{k} \theta_i f(\boldsymbol{x}_i) \leqslant \max_{1 \leqslant i \leqslant k} f(\boldsymbol{x}_i),$$

from which the conclusion follows.

3.5 Formal description of the simplex algorithm

We make the same three assumptions as in the previous section about the constraints $A\boldsymbol{x} = \boldsymbol{b}$. Each stage of the algorithm determines a basis which is a subset $B = \{i_1, \ldots, i_m\} \subseteq \{1, \ldots, n\}$, corresponding to the indices of the basic variables, with $N = \{1, \ldots, n\} \setminus B$ being the indices of the non-basic variables. After rearranging the columns of A and the elements of $\boldsymbol{x}$, if necessary, we can rewrite $A = (A_B, A_N)$, where A_B, A_N are the columns of A corresponding to B, N, respectively, and $\boldsymbol{x} = \begin{pmatrix} \boldsymbol{x}_B \\ \boldsymbol{x}_N \end{pmatrix}$ is the rearrangement of $\boldsymbol{x}$, the constraints become

$$A_B \boldsymbol{x}_B + A_N \boldsymbol{x}_N = \boldsymbol{b}.$$

By our assumptions, A_B is non-singular and we can determine the basic solution corresponding to B by setting $\boldsymbol{x}_N = 0$ and solving for $\boldsymbol{x}_B = A_B^{-1}\boldsymbol{b}$. So B gives a b.f.s. if $A_B^{-1}\boldsymbol{b} \geqslant 0$. If we partition $\boldsymbol{c} = \begin{pmatrix} \boldsymbol{c}_B \\ \boldsymbol{c}_N \end{pmatrix}$ in the same way the objective function becomes

$$f = \boldsymbol{c}_B^\top \boldsymbol{x}_B + \boldsymbol{c}_N^\top \boldsymbol{x}_N.$$

As we have seen the algorithm expresses the problem in terms of the non-basic variables at each stage. Rewriting the constraint we get $\boldsymbol{x}_B = A_B^{-1}\boldsymbol{b} - A_B^{-1} A_N \boldsymbol{x}_N$ and substituting that into the objective function yields

$$f = \boldsymbol{c}_B^\top A_B^{-1}\boldsymbol{b} + \left(\boldsymbol{c}_N - A_N^\top \left(A_B^{-1}\right)^\top \boldsymbol{c}_B\right)^\top \boldsymbol{x}_N. \tag{1}$$

The complementary slackness condition in this problem is

$$0 = \left(\boldsymbol{c} - A^\top \boldsymbol{\lambda}\right)^\top \boldsymbol{x} = \left(\boldsymbol{c}_B - A_B^\top \boldsymbol{\lambda}\right)^\top \boldsymbol{x}_B + \left(\boldsymbol{c}_N - A_N^\top \boldsymbol{\lambda}\right)^\top \boldsymbol{x}_N.$$

At this basic solution $\boldsymbol{x}_N = 0$ and the components of $\boldsymbol{x}_B$ are positive (if it is a b.f.s.) so to ensure complementary slackness we take $\boldsymbol{\lambda} = \left(A_B^{-1}\right)^\top \boldsymbol{c}_B$. Notice that this choice of $\boldsymbol{\lambda}$, as well as ensuring complementary slackness, gives the same values of the dual and primal objective solutions

$$h = \boldsymbol{b}^\top \boldsymbol{\lambda} = \boldsymbol{b}^\top \left(A_B^{-1}\right)^\top \boldsymbol{c}_B = f.$$

Dual feasibility for the problem is

$$0 \geqslant \boldsymbol{c} - A^\top \boldsymbol{\lambda} = \begin{pmatrix} \boldsymbol{c}_B \\ \boldsymbol{c}_N \end{pmatrix} - \begin{pmatrix} A_B^\top \\ A_N^\top \end{pmatrix} \left(A_B^{-1}\right)^\top \boldsymbol{c}_B = \begin{pmatrix} 0 \\ \boldsymbol{c}_N - A_N^\top \boldsymbol{\lambda} \end{pmatrix},$$

which reduces to the condition $\boldsymbol{c}_N - A_N^\top \boldsymbol{\lambda} \leqslant 0$. If we substitute for $\boldsymbol{\lambda}$ into (1) we see that the primal objective function in terms of the non-basic variables is then

$$f = \boldsymbol{c}_B^\top A_B^{-1}\boldsymbol{b} + \left(\boldsymbol{c}_N - A_N^\top \boldsymbol{\lambda}\right)^\top \boldsymbol{x}_N.$$

Schematically, the simplex tableau may be expressed as

	$\overset{*}{\boldsymbol{x}_B^\top}$	$\boldsymbol{x}_N^\top$	
	I	$A_B^{-1} A_N$	$A_B^{-1}\boldsymbol{b}$
Payoff	0	$\left(\boldsymbol{c}_N - A_N^\top \boldsymbol{\lambda}\right)^\top$	$-\boldsymbol{c}_B^\top \left(A_B^{-1}\boldsymbol{b}\right)$

The algorithm proceeds as follows:

(i) it maintains primal feasibility at each stage, that is, the right-hand sides in the tableau $A_B^{-1}\boldsymbol{b} \geqslant 0$;

(ii) it has complementary slackness built in at each stage by the choice of $\boldsymbol{\lambda}$;

(iii) it seeks to satisfy the third of the three necessary and sufficient conditions for optimality which is dual feasibility, $\boldsymbol{c}_N - A_N^\top \boldsymbol{\lambda} \leqslant 0$, which we see is the condition that the entries in the payoff row are $\leqslant 0$.

Finally, we can justify the assertion made earlier that, when the initial basic solution is obtained by taking the slack variables as the basic variables then the negative of the dual variables will be the entries under these variables in the tableau. Observe that in this case we would have $A_N = I$ and $\boldsymbol{c}_N = 0$ and it is then clear from the schematic tableau above that the entry in the payoff row under $\boldsymbol{x}_N^\top$ is $-\boldsymbol{\lambda}^\top$.